Plataforma

# AWS Well-Architected Framework



Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS Well-Architected Framework: Plataforma

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

# Table of Contents

Resumen e introducción ..... 1
Introducción ..... 1
Definiciones ..... 2
Sobre la arquitectura ..... 5
Principios de diseño generales ..... 6
Los pilares del marco ..... 8
Excelencia operativa ..... 8
Principios de diseño ..... 9
Definición ..... 10
Prácticas recomendadas ..... 11
Recursos ..... 21
Seguridad ..... 22
Principios de diseño ..... 22
Definición ..... 23
Prácticas recomendadas ..... 24
Recursos ..... 34
Fiabilidad ..... 35
Principios de diseño ..... 35
Definición ..... 36
Prácticas recomendadas ..... 37
Recursos ..... 42
Eficiencia del rendimiento ..... 43
Principios de diseño ..... 43
Definición ..... 44
Prácticas recomendadas ..... 44
Recursos ..... 50
Optimización de costos ..... 51
Principios de diseño ..... 51
Definición ..... 52
Prácticas recomendadas ..... 53
Recursos ..... 59
Sostenibilidad ..... 59
Principios de diseño ..... 60
Definición ..... 61

Prácticas recomendadas ..... 62
Recursos ..... 69
El proceso de revisión ..... 70
Conclusión ..... 73
Colaboradores ..... 74
Documentación adicional ..... 75
Revisiones del documento ..... 76
Apéndice: preguntas y prácticas recomendadas ..... 80
Excelencia operativa ..... 80
Organization ..... 80
Prepare ..... 141
Operación ..... 215
Evolución ..... 262
Seguridad ..... 282
Aspectos básicos de seguridad ..... 283
Identity and Access Management ..... 309
Detección ..... 372
Protección de la infraestructura ..... 389
Protección de datos ..... 417
Respuesta a incidentes ..... 454
Seguridad de las aplicaciones ..... 480
Fiabilidad ..... 505
Principios básicos ..... 505
Arquitectura de la carga de trabajo ..... 549
Administración de cambios ..... 602
Administración de errores ..... 653
Eficiencia del rendimiento ..... 766
Selección de la arquitectura ..... 766
Computación y hardware ..... 782
Administración de datos ..... 802
Redes y entrega de contenido ..... 829
Proceso y cultura ..... 861
Optimización de costos ..... 878
Práctica de administración financiera en la nube ..... 879
Conocimiento del gasto y del uso ..... 905
Recursos rentables ..... 951

Administración de la demanda y suministro de recursos ..... 995
Optimización a lo largo del tiempo ..... 1008
Sostenibilidad ..... 1018
Selección de región ..... 1018
Alineación con la demanda ..... 1020
Software y arquitectura ..... 1036
Datos ..... 1050
Hardware y servicios ..... 1071
Proceso y cultura ..... 1081
Avisos ..... 1093
Glosario de AWS ..... 1094

# AWS Well-Architected Framework

Fecha de publicación: 6 de noviembre de 2024 (Revisiones del documento)

El Marco de AWS Well-Architected le ayuda a comprender las ventajas y desventajas de las decisiones que toma al crear sistemas en AWS. El uso del marco le permitirá conocer las prácticas recomendadas de arquitectura para diseñar y operar sistemas en la nube que sean fiables, seguros, eficientes, rentables y sostenibles.

## Introducción

El Marco de AWS Well-Architected le ayuda a comprender las ventajas y desventajas de las decisiones que toma al crear sistemas en AWS. Mediante el uso del marco, podrá conocer las prácticas recomendadas de arquitectura para diseñar y operar cargas de trabajo en la Nube de AWS que sean seguras, fiables, eficientes, rentables y sostenibles. Proporciona un medio para medir sus arquitecturas de forma coherente comparándolas con las prácticas recomendadas e identificar áreas de mejora. El proceso para revisar una arquitectura es un diálogo constructivo sobre las decisiones de arquitectura y no es un mecanismo de auditoría. Creemos que contar con sistemas con una buena arquitectura aumenta en gran medida la probabilidad de éxito empresarial.

AWS Solutions Architects cuenta con años de experiencia en soluciones de diseño de arquitecturas para una amplia variedad de sectores empresariales y casos de uso. Hemos ayudado a diseñar y revisar miles de arquitecturas de clientes en AWS. A partir de dicha experiencia, hemos identificado las prácticas recomendadas y estrategias centrales para sistemas de diseño de arquitecturas en la nube.

El Marco de AWS Well-Architected documenta un conjunto de cuestiones fundamentales que le permiten comprender si una arquitectura específica se corresponde con las prácticas recomendadas de la nube. El marco proporciona un enfoque coherente para evaluar los sistemas frente a las cualidades que espera de los sistemas modernos basados en la nube y la solución necesaria para lograr dichas cualidades. A medida que AWS continúa evolucionando y que seguimos aprendiendo más al trabajar con nuestros clientes, continuaremos perfeccionando la definición de una buena arquitectura.

Este marco está destinado a aquellos que ocupan puestos en tecnología, como los directores de tecnología (CTO), arquitectos, desarrolladores y miembros del equipo de operaciones. Describe las prácticas recomendadas y las estrategias de AWS para diseñar y operar una carga de trabajo en la

nube y proporciona enlaces a más detalles sobre implementación y patrones arquitectónicos. Para obtener más información, consulte la página de inicio de AWS Well-Architected.

AWS también proporciona un servicio para revisar sus cargas de trabajo de forma gratuita. La Herramienta de AWS Well-Architected (AWS WA Tool) es un servicio en la nube que proporciona un proceso coherente para que revise y mida su arquitectura con el Marco de AWS Well-Architected. La Herramienta de AWS WA proporciona recomendaciones para que sus cargas de trabajo sean más fiables, seguras, eficientes y rentables.

Para ayudarle a aplicar las prácticas recomendadas, hemos creado los Laboratorios de AWS Well-Architected, que proporcionan un repositorio de código y documentación para ofrecerle experiencia práctica en la implementación de las prácticas recomendadas. También nos hemos unido a socios selectos de la Red de socios de AWS (APN) que son miembros del Programa para Socios de AWS Well-Architected. Estos socios de AWS tienen exhaustivos conocimientos sobre AWS y pueden ayudarle a revisar y mejorar sus cargas de trabajo.

## Definiciones

Cada día, los expertos de AWS ayudan a los clientes con la arquitectura de sistemas para aprovechar las prácticas recomendadas en la nube. Trabajamos conjuntamente para lograr compensaciones arquitectónicas a medida que sus diseños evolucionan. A medida que implementa estos sistemas en entornos reales, descubrimos el excelente rendimiento de dichos sistemas y las consecuencias de dichas compensaciones.

Con lo aprendido, hemos creado el Marco de AWS Well-Architected, que proporciona un conjunto coherente de prácticas recomendadas para que clientes y socios evalúen arquitecturas y cuenta con un conjunto de preguntas que puede utilizar para evaluar en qué medida una arquitectura está alineada con las prácticas recomendadas de AWS.

El Marco de AWS Well-Architected se basa en seis pilares: excelencia operativa, seguridad, fiabilidad, eficiencia del rendimiento, optimización de costos y sostenibilidad.

Tabla 1. Los pilares de AWS Well-Architected Framework

| Name | Descripción |
| --- | --- |
| Excelencia operativa | Capacidad de apoyar el desarrollo y ejecutar cargas de trabajo eficazmente, conocer sus operaciones y mejorar continuamente los |

| Name | Descripción |
| --- | --- |
| | procesos y procedimientos de soporte para ofrecer valor empresarial. |
| Seguridad | En el pilar de seguridad, se describe cómo aprovechar las tecnologías en la nube para proteger los datos, los sistemas y los activos de una manera que pueda mejorar su posición de seguridad. |
| Fiabilidad | El pilar de fiabilidad abarca la capacidad de una carga de trabajo para llevar a cabo su función prevista de forma correcta y coherente cuando se espera que lo haga. Esto incluye la capacidad de utilizar y probar la carga de trabajo a lo largo de todo su ciclo de vida. En este documento se incluye orientación de prácticas recomendadas para la implement ación de cargas de trabajo fiables en AWS. |
| Eficiencia del rendimiento | Capacidad para utilizar los recursos computaci onales de forma eficaz a fin de que satisfaga n los requisitos del sistema y para mantener dicha eficacia a medida que la demanda cambia y las tecnologías evolucionan. |
| Optimización de costos | Capacidad de ejecutar sistemas para proporcio nar valor comercial al menor precio. |
| Sostenibilidad | Es la capacidad de mejorar constantemente el impacto en la sostenibilidad mediante la reducción del consumo de energía y el aumento de la eficiencia en todos los componentes de una carga de trabajo, maximizando los beneficios de los recursos aprovisionados y minimizando el número total de recursos necesarios. |

En el Marco de AWS Well-Architected, usamos estos términos:

- Un componente es el código, la configuración y los recursos de AWS que, en conjunto, cumplen con un requisito. El componente suele ser la unidad de responsabilidad técnica y está desacoplado de otros componentes.
- El término carga de trabajo se usa para identificar un grupo de componentes que, en conjunto, proporcionan valor de negocio. La carga de trabajo suele ser el nivel de detalle sobre el que hablan los líderes tecnológicos y comerciales.
- Pensamos en la arquitectura como la forma en que los componentes trabajan juntos en una carga de trabajo. La forma en la que interactúan y se comunican los componentes es, a menudo, el foco de los diagramas de arquitectura.
- Los hitos marcan los cambios clave en la arquitectura a medida que evoluciona a lo largo del ciclo de vida del producto (diseño, implementación, prueba, lanzamiento y producción).
- En una organización, la cartera tecnológica es el conjunto de cargas de trabajo necesarias para que opere la empresa.
- El nivel de esfuerzo consiste en categorizar la cantidad de tiempo, esfuerzo y complejidad que requiere la implementación de una tarea. Cada organización tiene que considerar el tamaño y la experiencia del equipo y la complejidad de la carga de trabajo como contexto adicional a fin de determinar correctamente el nivel de esfuerzo de la organización.
  - Alto: el trabajo podría llevar varias semanas o meses. Esto podría desglosarse en múltiples historias, versiones y tareas.
  - Medio: el trabajo podría llevar varios días o semanas. Esto podría desglosarse en múltiples versiones y tareas.
  - Bajo: el trabajo podría llevar varias horas o días. Esto podría desglosarse en múltiples tareas.

Al diseñar la arquitectura de las cargas de trabajo, se hacen concesiones entre pilares según el contexto empresarial. Estas decisiones de negocio puede impulsar sus prioridades de diseño. Podría optimizarlas para mejorar el impacto en la sostenibilidad y reducir los costos en detrimento de la fiabilidad en los entornos de desarrollo o, si se trata de soluciones fundamentales, podría optimizar la fiabilidad con incremento de costos e impacto en la sostenibilidad. En las soluciones de comercio electrónico, el rendimiento puede afectar a los ingresos y a la tendencia de los clientes a comprar. Sin embargo, por lo general, la excelencia en la seguridad y las operaciones no afecta a los demás pilares.

# Sobre la arquitectura

En entornos en las instalaciones, los clientes suelen contar con un equipo central dedicado a la arquitectura tecnológica que está por encima de otros equipos de productos o características para verificar que sigan las prácticas recomendadas. En los equipos de arquitectura tecnológica suelen haber distintos roles como el de arquitecto técnico (infraestructura), arquitecto de soluciones (software), arquitecto de datos, arquitecto de redes y arquitecto de seguridad. A menudo, estos equipos utilizan TOGAF o el Zachman Framework como parte de una capacidad de arquitectura empresarial.

En AWS, preferimos distribuir las capacidades en equipos en lugar de tener un equipo centralizado con esa capacidad. Existen riesgos cuando se elige distribuir la autoridad de la toma de decisiones, por ejemplo, al verificar que los equipos cumplan con los estándares internos. Mitigamos estos riesgos de dos maneras. En primer lugar, tenemos prácticas (formas de hacer las cosas, procesos, estándares y normas aceptadas) que se centran en permitir que cada equipo tenga esa capacidad, además de que contamos con expertos que verifican que los equipos eleven el nivel de los estándares que deben cumplir. En segundo lugar, implementamos mecanismos que llevan a cabo comprobaciones automáticas para verificar que se cumplan los estándares.

> “Las buenas intenciones nunca funcionan; hacen falta buenos mecanismos para que todo suceda”, Jeff Bezos.

Esto significa reemplazar los esfuerzos humanos por mecanismos (a menudo automáticos) que controlan el cumplimiento de las normas y los procesos. Este enfoque distribuido está respaldado por los principios de liderazgo de Amazon y establece una cultura en todos los roles que funciona a partir del cliente. Pensar en el cliente es esencial en nuestro proceso de innovación. Comenzamos con el cliente y lo que quiere, y dejamos que eso defina y guíe nuestros esfuerzos. Los equipos centrados en el cliente crean productos como respuesta a una necesidad del cliente.

Para la arquitectura, esto significa que esperamos que cada equipo tenga la capacidad de crear arquitecturas y seguir las prácticas recomendadas. Para ayudar a los nuevos equipos a obtener estas capacidades o a los equipos existentes a subir el listón, activamos el acceso a una comunidad virtual de ingenieros principales que pueden revisar sus diseños y ayudarles a comprender cuáles son las prácticas recomendadas de AWS. La comunidad de ingenieros trabaja para que las prácticas recomendadas sean visibles y accesibles. Una forma de hacerlo es, por ejemplo, a través de charlas a la hora del almuerzo centradas en la aplicación de las prácticas recomendadas a casos reales.

Estas charlas se graban y pueden utilizarse como parte de los materiales de incorporación para los nuevos miembros del equipo.

Las prácticas recomendadas de AWS surgen de nuestra experiencia con miles de sistemas a escala de Internet. Preferimos usar datos para definir las prácticas recomendadas, pero también usamos expertos en la materia, como ingenieros principales, para establecerlas. A medida que los ingenieros principales ven emerger nuevas prácticas recomendadas, trabajan como una comunidad para verificar que los equipos las sigan. Con el tiempo, estas prácticas recomendadas se formalizan en nuestros procesos de revisión interna, y también en mecanismos que garantizan su cumplimiento. El Marco de Well-Architected representa la implementación orientada al cliente de nuestro proceso de revisión interna, donde hemos codificado la forma de pensar de nuestros ingenieros principales en roles de campo, como los de la arquitectura de soluciones y de los equipos de ingeniería internos. El Marco de Well-Architected es un mecanismo escalable que le permite aprovechar estos aprendizajes.

Al seguir el enfoque de una comunidad de ingeniería con propiedad distribuida de la arquitectura, creemos que puede surgir una arquitectura empresarial de Well-Architected impulsada por las necesidades del cliente. Los líderes tecnológicos (como los CTO o los administradores de desarrollo) que hagan revisiones de Well-Architected en todas sus cargas de trabajo podrán comprender mejor los riesgos de su cartera de tecnología. Con este enfoque, puede identificar temas en todos los equipos que su organización podría abordar mediante mecanismos, formaciones o charlas a la hora del almuerzo donde los ingenieros principales pueden compartir sus ideas sobre áreas específicas con múltiples equipos.

# Principios de diseño generales

El Marco de Well-Architected identifica un conjunto de principios generales de diseño que facilitan un buen diseño en la nube:

- No más conjeturas sobre la capacidad que necesita: si opta por poca capacidad al implementar una carga de trabajo, puede terminar con recursos inactivos caros o lidiando con las implicaciones de rendimiento de una capacidad limitada. Con los servicios de computación en la nube, estos problemas pueden desaparecer. Puede usar tanta capacidad como necesite y escalar automáticamente hacia arriba y hacia abajo.
- Prueba de sistemas a escala de producción: en la nube, puede crear un entorno de prueba a escala de producción bajo demanda, completar las pruebas y retirar los recursos. Debido a que solo paga por el entorno de prueba cuando se ejecuta, puede simular su entorno real por una fracción del costo de las pruebas en las instalaciones.

- Automatización según la experimentación arquitectónica: la automatización le permite crear y replicar sus cargas de trabajo a bajo costo, además de evitarle los gastos generados por el esfuerzo manual. Puede rastrear cambios en su automatización, auditar su impacto y volver a los parámetros anteriores cuando sea necesario.
- Consideración de la posibilidad de usar arquitecturas evolutivas: en un entorno tradicional, las decisiones arquitectónicas a menudo se implementan como eventos estáticos puntuales, y solo se desarrollan algunas versiones principales de un sistema durante su vida útil. A medida que una empresa y su contexto evolucionan, estas decisiones iniciales pueden dificultar la capacidad del sistema para cumplir con los requisitos cambiantes de la empresa. En la nube, la capacidad de automatizar y probar bajo demanda reduce el riesgo de impacto de los cambios en el diseño. Esto permite que los sistemas evolucionen con el paso del tiempo para que las empresas puedan aprovechar las innovaciones como una práctica habitual.
- Impulso de arquitecturas mediante el uso de datos: en la nube, puede recopilar datos sobre cómo sus decisiones arquitectónicas afectan al comportamiento de la carga de trabajo. Esto le permite tomar decisiones basadas en hechos sobre cómo mejorar su carga de trabajo. Su infraestructura en la nube es código, por lo que pueden utilizar esos datos para notificar sus elecciones de arquitectura y mejoras a lo largo del tiempo.
- Mejora mediante “días de juego”: ponga a prueba el rendimiento de su arquitectura y sus procesos al programar periódicamente días de juego para simular eventos en producción. Esto ayudará a comprender dónde se pueden efectuar mejoras y a desarrollar la experiencia organizacional en la gestión de eventos.

# Los pilares del marco

Crear un sistema de software se asemeja mucho a construir un edificio. Si los cimientos son endebles, pueden producirse problemas estructurales que minen la integridad y el funcionamiento del edificio. Al diseñar soluciones tecnológicas, si descuida los seis pilares de excelencia operativa, seguridad, fiabilidad, eficiencia del rendimiento, optimización de costos y sostenibilidad, puede resultar difícil crear un sistema que cumpla con sus expectativas y requisitos. La incorporación de dichos pilares en su arquitectura ayudará a generar sistemas estables y eficientes. Esto le permitirá centrarse en otros aspectos del diseño, como los requisitos funcionales.

Pilares

- Excelencia operativa
- Seguridad
- Fiabilidad
- Eficiencia del rendimiento
- Optimización de costos
- Sostenibilidad

# Excelencia operativa

La excelencia operativa (OE) es el compromiso de crear software correctamente, al mismo tiempo que se ofrece de forma coherente una excelente experiencia de cliente. El pilar de excelencia operativa contiene las prácticas recomendadas para organizar su equipo, diseñar su carga de trabajo, ejecutarla a escala y hacerla evolucionar con el tiempo.

El pilar de excelencia operativa proporciona información general sobre los principios de diseño, prácticas recomendadas y preguntas. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de excelencia operativa.

Temas

- Principios de diseño
- Definición
- Prácticas recomendadas
- Recursos

# Principios de diseño

A continuación, se presentan los principios de diseño para la excelencia operativa en la nube:

- Organización de los equipos en torno a los resultados empresariales: la capacidad de un equipo para lograr los resultados empresariales proviene de la visión del liderazgo, las operaciones efectivas y un modelo operativo alineado con el negocio. Los líderes deben invertir en la transformación de CloudOps y comprometerse plenamente con este proceso a través de un modelo operativo en la nube adecuado que incentive a los equipos a trabajar de la manera más eficiente y a alcanzar los objetivos empresariales. Un modelo operativo adecuado aprovechará las capacidades de las personas, los procesos y la tecnología para escalar, optimizar la productividad y diferenciarse a través de la agilidad y la capacidad de respuesta y adaptación. La visión a largo plazo de la organización se traduce en objetivos que se comunican a las partes interesadas de toda la empresa y a los consumidores de sus servicios en la nube. Los objetivos y los KPI operativos están alineados a todos los niveles. Esta práctica asegura el valor a largo plazo que se obtiene al aplicar los siguientes principios de diseño.
- Implementación de la observabilidad para obtener información práctica: conozca plenamente el comportamiento, el rendimiento, la fiabilidad, el costo y el estado de la carga de trabajo. Establezca indicadores clave de rendimiento (KPI) y utilice la telemetría de observabilidad para tomar decisiones informadas y medidas rápidas cuando los resultados empresariales estén en riesgo. Mejore de forma proactiva el rendimiento, la fiabilidad y los costos según los datos de observabilidad procesables.
- Automatización segura siempre que sea posible: en la nube, puede aplicar la misma disciplina de ingeniería que usa para el código de aplicación a todo el entorno. Puede definir toda la carga de trabajo y sus operaciones (aplicaciones, infraestructura, configuración y procedimientos) como código y actualizarla. A continuación, para proceder a automatizar las operaciones de su carga de trabajo, puede iniciarlas en respuesta a los eventos. En la nube, puede automatizar la seguridad mediante la configuración de barreras de protección, entre las que se incluyen el control de velocidad, los umbrales de error y las aprobaciones. Con una automatización eficaz, puede conseguir respuestas uniformes a los eventos, limitar los errores humanos y reducir el esfuerzo de los operadores.
- Cambios frecuentes, pequeños y reversibles: diseñe cargas de trabajo que sean escalables y tengan acoplamiento flexible para permitir que los componentes se actualicen con regularidad. Las técnicas de implementación automatizadas, en combinación con cambios graduales más pequeños, reducen el radio de repercusión y permiten una reversión más rápida cuando se producen fallos. Esto aumenta la confianza para aplicar cambios beneficiosos en su carga de

trabajo y, al mismo tiempo, se mantiene la calidad y es posible adaptarse rápidamente a los cambios en las condiciones del mercado.

- Refinamiento frecuente de los procedimientos operativos: a medida que evolucione las cargas de trabajo, evolucione sus operaciones de forma adecuada. A medida que vaya usando los procedimientos operativos, busque oportunidades para mejorarlos. Haga revisiones regulares y valide que todos los procedimientos sean efectivos y que los equipos estén familiarizados con ellos. Cuando se identifiquen lagunas, actualice los procedimientos en consecuencia. Comunique las actualizaciones de los procedimientos a todas las partes interesadas y equipos. Ludifique sus operaciones para compartir las prácticas recomendadas y formar a los equipos.
- Anticipación del fracaso: maximice el éxito operativo analizando los situaciones de error para comprender el perfil de riesgo de la carga de trabajo y su impacto en los resultados empresariales. Pruebe la eficacia de sus procedimientos y la respuesta de su equipo a estos fallos simulados. Tome decisiones fundamentadas para gestionar los riesgos existentes identificados por sus pruebas.
- Aprendizaje de todos los eventos y métricas operacionales: impulse las mejoras gracias a las lecciones que se aprendan después de todos los eventos operativos y errores. Comparta las conclusiones con los equipos y con toda la organización. Se deben destacar datos y anécdotas relacionados con la forma en la que las operaciones contribuyen a los resultados empresariales.
- Uso de los servicios administrados: reduzca la carga operativa mediante el uso de los servicios administrados de AWS siempre que sea posible. Desarrolle procedimientos operativos en torno a las interacciones con esos servicios.

## Definición

Hay cuatro áreas de prácticas recomendadas para la excelencia operativa en la nube:

- Organización
- Preparación
- Operación
- Evolución

La dirección de la organización define los objetivos empresariales. La organización debe comprender los requisitos y prioridades, y usarlos para organizar y llevar a cabo su trabajo para lograr los objetivos de la empresa. Su carga de trabajo debe emitir la información necesaria para apoyarlos. Al implementar servicios para conseguir la integración, la implementación y la entrega de su carga de

trabajo, creará un flujo creciente de cambios positivos para la producción al automatizar los procesos repetitivos.

Puede haber riesgos inherentes a la operativa de la carga de trabajo. Comprenda dichos riesgos y tome una decisión informada para iniciar la producción. Sus equipos deben poder prestar asistencia a su carga de trabajo. Las métricas empresariales y operativas derivadas de los resultados empresariales deseados le permitirán entender el estado de su carga de trabajo y sus actividades de operaciones, así como responder a los incidentes. Las prioridades cambiarán a medida que cambien sus necesidades empresariales y su entorno empresarial. Úselas como referencia para introducir mejoras continuamente en su organización y en la operativa de la carga de trabajo.

## Prácticas recomendadas

> **Note**
>
> Todas las preguntas sobre excelencia operativa llevan el prefijo OPS, que es la abreviatura del pilar.

Temas

- Organization
- Prepare
- Operación
- Evolución

### Organization

Sus equipos deben disponer de un entendimiento compartido de toda la carga de trabajo, su rol en ella y los objetivos empresariales compartidos para establecer las prioridades que permitan conseguir el éxito empresarial. Unas prioridades bien definidas maximizarán los beneficios de sus esfuerzos. Evalúe las necesidades de los clientes internos y externos, e involucre a las partes interesadas clave, incluidos los equipos de negocios, desarrollo y operaciones, para determinar dónde se deben centrar los esfuerzos. La evaluación de las necesidades de los clientes le permitirá verificar que tiene una comprensión profunda de la asistencia que se necesita para lograr los resultados empresariales. Verifique que conozca las directrices y las obligaciones definidas por la gobernanza de su organización y los factores externos, tales como los requisitos normativos de cumplimiento y los estándares del sector, para asegurarse de que pueda exigir o aplicar un enfoque

específico. Valide la existencia de mecanismos para identificar cambios en la gobernanza interna y en los requisitos de cumplimiento externos. Si no existen dichos requisitos, asegúrese de haber llevado a cabo una investigación exhaustiva para tomar las decisiones pertinentes. Revise sus prioridades con regularidad para poder actualizarlas a medida que cambien las necesidades.

Evalúe las amenazas a la empresa (por ejemplo, riesgos y responsabilidades empresariales, amenazas a la seguridad de la información) y mantenga dicha información en un registro de riesgos. Evalúe el impacto de los riesgos y las compensaciones entre los intereses opuestos o los enfoques alternativos. Por ejemplo, la aceleración de la velocidad de comercialización de las nuevas características puede primar sobre la optimización de los costos, o se puede elegir una base de datos relacional para los datos no relacionales para simplificar el esfuerzo de migración de un sistema. Administre los beneficios y los riesgos para tomar decisiones fundamentadas a la hora de determinar dónde centrar sus esfuerzos. Algunos riesgos u opciones son aceptables durante un tiempo, incluso se podrían mitigar los riesgos asociados, pero también podría ser inaceptable permitir que un riesgo persista, en cuyo caso se tomarán medidas para abordarlo.

Sus equipos deben comprender su papel en la consecución de los resultados empresariales. Los equipos deben comprender el rol que desempeñan en el éxito de otros equipos, así como el que desempeñan los demás equipos en su propio éxito, además de tener objetivos en común. Comprender la responsabilidad, la propiedad, cómo se toman las decisiones y quién tiene autoridad para tomarlas ayudará a centrar los esfuerzos y a maximizar los beneficios de sus equipos. Las necesidades de un equipo se verán determinadas por el cliente al que dan soporte, su organización, la composición del equipo y las características de su carga de trabajo. No es razonable esperar que un único modelo operativo sea capaz de respaldar a todos los equipos y sus cargas de trabajo en la organización.

Verifique que se haya identificado a los encargados de cada aplicación, carga de trabajo, plataforma y componente de la infraestructura, y de que cada proceso y procedimiento disponga de un encargado responsable de su definición y de encargados responsables de su rendimiento.

Las acciones de los miembros del equipo se fundamentarán en la comprensión del valor empresarial de cada componente, proceso y procedimiento, el motivo por el cual se establecieron los recursos o se llevan a cabo determinadas actividades, y la razón por la que esa propiedad existe. Defina claramente las responsabilidades de los miembros del equipo para que puedan actuar de forma adecuada y disponer de mecanismos para identificar la responsabilidad y la propiedad. Cuente con mecanismos para solicitar adiciones, cambios y excepciones para no limitar la innovación. Defina acuerdos entre los equipos que describan el trabajo conjunto para darse apoyo entre sí y respaldar los resultados de la empresa.

Preste asistencia a los miembros de su equipo para que puedan ser más eficaces a la hora de actuar y apoyar los resultado empresariales. Los líderes comprometidos deben establecer expectativas y medir el éxito. Los directivos deben ser los patrocinadores, defensores e impulsores de la adopción de las prácticas recomendadas y de la evolución de la organización. Deje que los miembros del equipo actúen cuando los resultados corran algún riesgo para, así, minimizar el impacto, y anímelos a derivar las cuestiones a los responsables de la toma de decisiones y las partes interesadas cuando crean que exista un riesgo, de manera que se pueda abordar y se eviten incidentes. Proporcione una comunicación oportuna, clara y procesable de los riesgos conocidos y de los eventos planificados para que los miembros del equipo puedan reaccionar de forma oportuna y adecuada.

Fomente la experimentación para acelerar el aprendizaje y mantener a los miembros del equipo interesados y comprometidos. Los equipos deben aumentar el conjunto de habilidades para adoptar nuevas tecnologías y para apoyar los cambios en la demanda y las responsabilidades. Debe apoyar y fomentar esto mediante un horario estructurado dedicado a la capacitación. Verifique que los miembros del equipo dispongan de los recursos (herramientas y miembros del equipo) para lograr el éxito y escalar con el fin de lograr los resultados empresariales. Aproveche la diversidad entre las organizaciones para buscar múltiples perspectivas únicas. Utilice esta perspectiva para aumentar la innovación, cuestionar sus suposiciones y reducir el riesgo de sesgo de confirmación. Fomente la inclusión, la diversidad y la accesibilidad en sus equipos para obtener perspectivas beneficiosas.

Si existen requisitos externos de regulación o cumplimiento que se aplican a su organización, debe utilizar los recursos proporcionados por el centro de conformidad en la nube de AWS para ayudar a instruir a sus equipos para que puedan considerar el impacto en sus prioridades operativas. El Marco de Well-Architected hace hincapié en aprender, medir y mejorar. Proporciona un enfoque coherente para evaluar las arquitecturas e implementar diseños que se escalarán con el tiempo. AWS proporciona la AWS Well-Architected Tool para ayudarle a revisar su enfoque antes del desarrollo, el estado de sus cargas de trabajo antes de la producción y el estado de sus cargas de trabajo durante la producción. Puede comparar las cargas de trabajo de las prácticas recomendadas de arquitectura de AWS más recientes, supervisar su estado global y obtener información sobre los riesgos potenciales. La AWS Trusted Advisor es una herramienta que proporciona acceso a un conjunto básico de comprobaciones que recomiendan optimizaciones que pueden ayudar a definir sus prioridades. Los clientes de Business y Enterprise Support reciben acceso a comprobaciones adicionales centradas en la seguridad, la fiabilidad, el rendimiento, la optimización de los costos y la sostenibilidad que pueden ayudar a configurar sus prioridades.

AWS puede ayudarlo a capacitar a sus equipos sobre AWS y sus servicios para aumentar su comprensión de cómo sus elecciones pueden tener un impacto en su carga de trabajo. Debe utilizar los recursos proporcionados por AWS Support (Centro de conocimientos de AWS, foros de debate

de AWS y Centro de AWS Support) y la documentación de AWS para capacitar a sus equipos. En caso de tener dudas sobre AWS, contacte con AWS Support a través del Centro de AWS Support. AWS también comparte los patrones y prácticas recomendadas que hemos aprendido a través del funcionamiento de AWS en la Amazon Builders' Library. Hay una gran variedad de información útil disponible a través del Blog de AWS y The Official AWS Podcast. AWS Training and Certification ofrece una formación gratuita a través de cursos digitales autoguiados sobre los conceptos básicos de AWS. También puede inscribirse en una formación adicional impartida por un instructor para respaldar el desarrollo de las habilidades de AWS de sus equipos.

Utilice herramientas o servicios que le permitan controlar de forma centralizada sus entornos en todas las cuentas, como, por ejemplo, AWS Organizations, para ayudarle a administrar los modelos operativos. Servicios como AWS Control Tower amplían esta capacidad de administración al permitirle definir esquemas (que respaldan sus modelos operativos) para la configuración de las cuentas, aplicar una gobernanza continua mediante AWS Organizations y automatizar el aprovisionamiento de nuevas cuentas. Los proveedores de servicios administrados tales como AWS Managed Services, los socios de AWS Managed Services o los proveedores de servicios administrados en la Red de socios de AWS proporcionan experiencia en la implementación de entornos en la nube y respaldan sus requisitos de seguridad y cumplimiento y sus objetivos empresariales. La incorporación de los servicios administrados a su modelo operativo puede ahorrarle tiempo y recursos, y le permite mantener a sus equipos internos centrados en los resultados estratégicos que diferenciarán a su empresa, en lugar de desarrollar nuevas competencias y capacidades.

Las siguientes preguntas se centran en estas consideraciones sobre la excelencia operativa. (Para ver una lista de preguntas y prácticas recomendadas relacionadas con la excelencia operativa, consulte el Apéndice.)

### OPS 1: ¿Cómo determina cuáles son sus prioridades?

Todos deben conocer el papel que desempeñan para lograr el éxito empresarial. Si se tienen objetivos compartidos, se podrán priorizar los recursos. Esto hará que los esfuerzos se traduzcan en un mayor rendimiento.

### OPS 2: ¿Cómo estructura su organización para lograr los objetivos empresariales?

Sus equipos deben comprender su papel en la consecución de los resultados empresariales. Los equipos deben comprender el rol que desempeñan en el éxito de otros equipos, así como el

OPS 2: ¿Cómo estructura su organización para lograr los objetivos empresariales?

que desempeñan los demás equipos en su propio éxito, además de tener objetivos en común. Comprender la responsabilidad, la propiedad, cómo se toman las decisiones y quién tiene autoridad para tomarlas ayudará a centrar los esfuerzos y a maximizar los beneficios de sus equipos.

OPS 3: ¿Cómo ayuda la cultura de su organización a alcanzar los objetivos empresariales?

Ayude a los miembros de su equipo para que puedan actuar de manera más eficaz y contribuir a avanzar hacia los objetivos de la empresa.

Es posible que, en algún momento, quiera hacer énfasis en un pequeño subgrupo de prioridades. Utilice un enfoque equilibrado a largo plazo para verificar el desarrollo de las capacidades necesarias y la administración de riesgos. Revise las prioridades con regularidad y actualícelas a medida que cambien las necesidades. Cuando la responsabilidad y la propiedad no están definidas o se desconocen, se corre el riesgo tanto de no actuar a tiempo, como de que se hagan esfuerzos repetidos y potencialmente conflictivos para abordar dichas necesidades. La cultura organizativa tiene un impacto directo en la satisfacción laboral y la retención de los miembros del equipo. Estimule el compromiso y las capacidades de los miembros de su equipo para lograr el éxito de la empresa. La experimentación es necesaria para innovar y convertir las ideas en resultados. Debe saber que un resultado no deseado es un experimento exitoso que ha identificado un camino que no llevará al éxito.

## Prepare

Para prepararse para la excelencia operativa hay que entender las cargas de trabajo y sus comportamientos esperados. Entonces, podrá diseñarlas para que proporcionen información sobre su estado y crear los procedimientos para respaldarlas.

Diseñe la carga de trabajo para que proporcione la información necesaria para que pueda comprender el estado interno (por ejemplo, métricas, registros, eventos y rastros) en todos los componentes en caso de problemas de investigación y observabilidad. La observabilidad va más allá de la simple supervisión, ya que proporciona una comprensión integral del funcionamiento interno de un sistema en función de sus resultados externos. La observabilidad, que se basa en métricas, registros y rastros, ofrece una visión profunda del comportamiento y la dinámica del sistema. Con

una observabilidad eficaz, los equipos pueden discernir patrones, anomalías y tendencias, lo que les permite abordar de forma proactiva los posibles problemas y mantener un estado óptimo del sistema. La identificación de los indicadores clave de rendimiento (KPI) es fundamental para garantizar la alineación entre las actividades de supervisión y los objetivos empresariales. Esta alineación garantiza que los equipos tomen decisiones basadas en datos mediante el uso de métricas que realmente sean relevantes, optimizando así tanto el rendimiento del sistema como los resultados empresariales. Además, la observabilidad permite que las empresas sean proactivas en lugar de reactivas. Los equipos pueden entender las relaciones de causa y efecto dentro de sus sistemas y predecir y prevenir los problemas en lugar de simplemente reaccionar ante ellos. A medida que las cargas de trabajo evolucionan, es esencial revisitar y refinar la estrategia de observabilidad para garantizar que esta siga siendo pertinente y eficaz.

Adopte enfoques que mejoren el flujo de cambios en producción y que ayuden a la refactorización, a la respuesta rápida sobre la calidad y a la corrección de errores. Estos enfoques aceleran los cambios positivos que se introducen en producción, limitan los problemas implementados y activan una rápida identificación y solución de los problemas introducidos a través de las actividades de implementación o descubiertos en sus entornos.

Adopte enfoques que proporcionen una respuesta inmediata sobre la calidad y logren una recuperación rápida de los cambios que no muestran los resultados deseados. El uso de estas prácticas ayuda a mitigar el impacto de los problemas generados con la implementación de cambios. Planifique para hacer frente a los cambios fallidos para que pueda responder rápidamente si es necesario. Además, pruebe y valide los cambios que haga. Debe conocer las actividades planificadas en sus entornos para poder administrar el riesgo de que los cambios afecten a dichas actividades. Haga cambios frecuentes, pequeños y reversibles para limitar el alcance del cambio. Al hacerlo, los problemas se solucionan de forma más rápida con la opción de revertir un cambio. También significa que podrá beneficiarse de unos cambios valiosos de forma más frecuente.

Evalúe la disponibilidad operativa de la carga de trabajo, los procesos y procedimientos, y el personal para comprender los riesgos operativos relacionados con la carga de trabajo. Use un proceso coherente (que incluya listas de verificación manuales y automáticas) para saber cuándo una carga de trabajo o cambio estarán listos para lanzarse. Esto también le ayudará a detectar cualquier área para la que sea necesaria la elaboración de un plan de tratamiento. Debe disponer de manuales de procedimientos que documenten las actividades rutinarias y guías de estrategias para aplicar los procesos de resolución de errores. Debe comprender los beneficios y los riesgos para tomar decisiones bien fundamentadas a fin de permitir que los cambios entren en la fase de producción.

AWS le permite ver toda su carga de trabajo (aplicaciones, infraestructura, política, gobernanza y operaciones) como código. Eso significa que puede aplicar la misma disciplina de ingeniería que usa para el código de las aplicaciones a cada elemento de su pila y compartirla entre los equipos u organizaciones para magnificar los beneficios de los esfuerzos de desarrollo. Use las operaciones como código en la nube y la capacidad de experimentar de manera segura para desarrollar la carga de trabajo, sus procedimientos operativos y poner en práctica los casos en los que se produzcan errores. Usar CloudFormation le permite tener entornos de producción, de pruebas y de desarrollo de entorno de pruebas coherentes y con formatos ya definidos, con un aumento de los niveles de control operativo.

Las siguientes preguntas se centran en estas consideraciones sobre la excelencia operativa.

### OPS 4: ¿Cómo implementa la observabilidad en su carga de trabajo?

Implemente la observabilidad en su carga de trabajo para que pueda comprender su estado y tomar decisiones basadas en datos en función de los requisitos empresariales.

### OPS 5: ¿Cómo reduce los defectos, facilita la reparación y mejora el flujo en producción?

Adopte métodos que mejoren el flujo de cambios en producción, que permitan la refactorización de la información rápida sobre la calidad y la corrección de errores. Esto acelerará los cambios positivos que se introducen en producción, limitará los problemas implementados y logrará una rápida identificación y solución de los problemas introducidos a través de las actividades de implementación.

### OPS 6: ¿Cómo mitiga los riesgos de implementación?

Adopte métodos que proporcionen una respuesta inmediata sobre la calidad y logren una recuperación rápida de los cambios que no obtengan los resultados deseados. El uso de estas prácticas ayuda a mitigar el impacto de los problemas generados con la implementación de cambios.

OPS 7: ¿Cómo sabe que está listo para dar respaldo a una carga de trabajo?

Evalúe la disponibilidad operativa de la carga de trabajo, los procesos y procedimientos, y el personal para comprender los riesgos operativos relacionados con la carga de trabajo.

Invierta en implementar actividades operativas como código para maximizar la productividad del personal de operaciones, minimizar las tasas de error y habilitar las respuestas automatizadas. Haga ensayos de fallos “pre mortem” para anticipar el fracaso y crear procedimientos cuando sea apropiado. Aplique metadatos mediante etiquetas de registro y Grupos de recursos de AWS mediante una estrategia de etiquetado coherente para permitir la identificación de sus recursos. Etiquete sus recursos para la organización, la contabilidad de costos, los controles de acceso y el objetivo de ejecución de actividades de operaciones automatizadas. Adopte las prácticas de implementación que aprovechan la elasticidad de la nube a fin de facilitar las actividades de desarrollo y la implementación previa de sistemas para que la implementación sea más rápida. Cuando haga cambios en las listas de control que utiliza para evaluar sus cargas de trabajo, planifique lo que hará con los sistemas activos que ya no cumplen los requisitos.

## Operación

La observabilidad le permite centrarse en datos significativos y comprender las interacciones y los resultados de su carga de trabajo. Al concentrarse en la información esencial y eliminar los datos innecesarios, mantiene un enfoque sencillo para comprender el rendimiento de las cargas de trabajo. No solo es esencial recopilar datos, sino también interpretarlos correctamente. Defina puntos de referencia claros, establezca umbrales de alerta adecuados y supervise activamente cualquier desviación. Un cambio en una métrica clave, especialmente cuando se correlaciona con otros datos, puede identificar áreas problemáticas concretas. Con la observabilidad, está mejor preparado para prever y abordar los posibles desafíos, lo que garantiza que su carga de trabajo funcione sin problemas y satisfaga las necesidades empresariales.

El éxito operativo de una carga de trabajo se mide por los logros de los resultados del cliente y del negocio. Defina los resultados esperados, decida cómo se medirá el éxito e identifique las métricas que se usarán en los cálculos para determinar si su carga de trabajo y las operaciones se efectúan con éxito. El estado de las operaciones incluye tanto el estado de la carga de trabajo como el éxito de las operaciones que se hacen para llevarlas a cabo (por ejemplo, la implementación y la respuesta frente a incidencias). Establezca puntos de referencia de métricas para las mejoras, la investigación y la intervención, y recopile y analice las métricas. A continuación, corrobore si comprende el éxito de las operaciones y cómo cambia con el tiempo. Utilice métricas recopiladas

para determinar si satisface las necesidades del cliente y del negocio. Identifique también las áreas de mejora.

Se requiere eficacia y eficiencia en la administración de los eventos operativos para lograr excelencia operativa. Se aplica tanto a los eventos operativos planificados como a los no planificados. Utilice los manuales de procedimientos establecidos para eventos bien conocidos y las guías de estrategia como ayuda para investigar y para resolver otros problemas. Priorice aquellos eventos que tengan mayor repercusión en el negocio y en el cliente. Verifique que, si se genera una alerta como respuesta a un evento, se ejecutará un proceso asociado con un encargado identificado de forma específica. Defina con antelación el personal necesario para resolver un evento e incluya procesos de escalado para que participe personal adicional, si es necesario, en función de la urgencia y el impacto. Identifique e implique a aquellos individuos que tengan autoridad para decidir sobre las acciones en aquellos casos en los que la respuesta a un evento que no se haya abordado previamente repercuta en el negocio.

Comunique el estado operativo de las cargas de trabajo mediante paneles y notificaciones adaptadas a la audiencia de destino (por ejemplo, cliente, negocio, desarrolladores, operaciones) para que puedan llevar a cabo las medidas adecuadas, administren sus expectativas y se les informe cuando se reanuden las operaciones habituales.

En AWS, puede generar vistas de panel de las métricas recopiladas a partir de cargas de trabajo y de AWS de forma nativa. Puede aprovechar CloudWatch o aplicaciones de terceros para agregar y presentar vistas de la empresa, la carga de trabajo y las operaciones de las actividades operativas. AWS proporciona información sobre cargas de trabajo mediante capacidades de registro, como AWS X-Ray, CloudWatch, CloudTrail y registros de flujo de VPC para identificar problemas de las cargas de trabajo a fin de ofrecer apoyo a la hora de analizar y corregir la causa raíz.

Las siguientes preguntas se centran en estas consideraciones sobre la excelencia operativa.

### OPS 8: ¿Cómo utiliza la observabilidad de la carga de trabajo en su organización?

Recurra a la observabilidad para garantizar un estado óptimo de la carga de trabajo. Utilice métricas, registros y rastros pertinentes para obtener una visión integral del rendimiento de su carga de trabajo y abordar los problemas de manera eficiente.

OPS 9: ¿Qué hace para comprender el estado de las operaciones?

Defina, capture y analice las métricas de las operaciones para obtener visibilidad de los eventos de operaciones y poder tomar las medidas adecuadas.

OPS 10: ¿Cómo administra la carga de trabajo y los eventos de operaciones?

Prepare y valide los procedimientos de respuesta a los eventos para minimizar la interrupción de la carga de trabajo.

Todas las métricas que recopile deben estar alineadas con una necesidad empresarial y los objetivos a los que estas contribuyen. Desarrolle respuestas con scripts para los eventos bien conocidos y automatice su rendimiento en respuesta al reconocimiento del evento.

## Evolución

Aprenda, comparta y mejore continuamente para mantener la excelencia operativa. Dedique ciclos de trabajo a hacer mejoras graduales de forma casi continua. Haga análisis posteriores al incidente de todos los eventos que afecten a los clientes. Identifique los factores que han contribuido a ello y actúe de forma preventiva para limitar o impedir que se repita. Comunique los factores que han contribuido a ello a las comunidades afectadas, según proceda. Evalúe y priorice las oportunidades de mejora de forma gradual (por ejemplo, solicitudes de características, solución de problemas y requisitos de cumplimiento), entre ellos, los procedimientos operativos y de cargas de trabajo.

Incluya bucles de comentarios en los procedimientos para identificar rápidamente las áreas de mejora y recoger lo aprendido durante la ejecución de operaciones.

Comparta lo aprendido con los equipos para enseñar los beneficios de dichas lecciones. Analice las tendencias de las lecciones aprendidas y haga un análisis retrospectivo de las métricas de las operaciones entre equipos para identificar oportunidades y métodos de mejora. Aplique aquellos cambios que traigan consigo mejoras y evalúe los resultados para determinar el éxito.

En AWS, puede exportar los datos de registro a Amazon S3 o enviar registros directamente a Amazon S3 para un almacenamiento a largo plazo. AWS Glue le permite descubrir y preparar los datos de registro en Amazon S3 para hacer análisis y almacenar los metadatos asociados en el AWS Glue Data Catalog. Puede utilizar Amazon Athena, a través de su integración nativa con AWS

Glue, para analizar los datos de registro y consultarlos mediante SQL estándar. Con una herramienta de inteligencia empresarial como Amazon Quick, puede visualizar, explorar y analizar sus datos. También puede descubrir las tendencias y los eventos de interés que pueden fomentar las mejoras.

La siguiente pregunta se centra en estas consideraciones acerca de la excelencia operativa.

**OPS 11: ¿Cómo desarrolla las operaciones?**

Dedique tiempo y recursos a la mejora gradual casi continua, para incrementar la eficacia y la eficiencia de sus operaciones.

Una correcta evolución de las operaciones depende de cambios pequeños, pero frecuentes, entornos seguros y tiempo para experimentar, desarrollar y probar mejoras, así como entornos en los que se anima a aprender de los errores. La asistencia operativa en entornos de producción, pruebas, desarrollo y entorno de pruebas, con un nivel creciente de controles operativos, facilita el desarrollo y aumenta la predictibilidad de resultados exitosos a partir de los cambios que se implementen en producción.

## Recursos

Consulte los siguientes recursos para obtener más información sobre nuestras prácticas recomendadas de excelencia operativa.

### Documentación

- DevOps y AWS

### Documento técnico

- Operational Excellence Pillar

### Video

- DevOps en Amazon

# Seguridad

El pilar de seguridad engloba la capacidad de proteger datos, sistemas y activos para sacar partido de las tecnologías de nube con el fin de mejorar su nivel de seguridad.

El pilar de seguridad ofrece una visión general de principios de diseño, prácticas recomendadas y preguntas. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de seguridad.

Temas


- Principios de diseño
- Definición
- Prácticas recomendadas
- Recursos


## Principios de diseño

Hay una serie de principios en la nube que pueden ayudarle a fortalecer la seguridad de la carga de trabajo:

- Implementación de sólidas bases de identidad: implemente un principio de privilegios mínimos y aplique una separación de tareas con la autorización adecuada para cada interacción con los recursos de AWS. Centralice la administración de identidades y busque eliminar la dependencia de las credenciales a largo plazo.
- Mantenimiento de la trazabilidad: supervise, audite y alerte de acciones y cambios en el entorno en tiempo real. Integre la recopilación de registros y métricas con sistemas para investigar y tomar medidas automáticamente.
- Aplicación de la seguridad en todas las capas: aplique un enfoque de defensa en profundidad con múltiples controles de seguridad. Impleméntelo en todas las capas (por ejemplo, red periférica, VPC, equilibrio de carga, cada instancia y servicio de computación, sistema operativo, aplicación y código).
- Automatización de las prácticas recomendadas de seguridad: los mecanismos automatizados de seguridad basados en software mejoran la capacidad de escalar de forma más segura, rápida y rentable. Cree arquitecturas seguras, como la implementación de controles definidos y administrados como código en plantillas controladas por versión.

- Protección de datos en tránsito y en reposo: clasifique los datos por niveles de confidencialidad y use mecanismos como el cifrado, la tokenización y el control de acceso cuando corresponda.
- Alejamiento de las personas respecto a los datos: use mecanismos y herramientas para reducir o eliminar la necesidad de acceder directamente a los datos o procesarlos manualmente. De esta forma, se reducen los errores humanos y el riesgo de una mala gestión o modificación al gestionar información confidencial.
- Preparación para eventos de seguridad: para prepararse para un incidente, tenga a su disposición procesos y políticas de investigación y administración de incidentes que se ajusten a los requisitos de su organización. Ejecute simulaciones de respuesta frente a incidencias y use herramientas con automatización para aumentar la velocidad de detección, investigación y recuperación.

## Definición

Existen siete áreas de prácticas recomendadas para la seguridad en la nube:

- Aspectos básicos de seguridad
- Identity and Access Management
- Detección
- Protección de la infraestructura
- Protección de datos
- Respuesta frente a incidencias
- Seguridad de las aplicaciones

Antes de diseñar una carga de trabajo, hay que adoptar prácticas que influyan en la seguridad. Debe controlar quién puede hacer qué. Además, debe poder identificar los incidentes de seguridad, proteger sus sistemas y servicios, y mantener la confidencialidad e integridad de los datos a través de la protección de datos. Debe tener un proceso ben definido y practicado para responder a los incidentes de seguridad. Estas herramientas y técnicas son importantes porque respaldan objetivos como la prevención de pérdidas económicas o el cumplimiento de las obligaciones reglamentarias.

El Modelo de responsabilidad compartida de AWS ayuda a que las organizaciones adopten la nube para alcanzar sus metas de seguridad y cumplimiento. Debido a que AWS asegura físicamente la infraestructura que permite nuestros servicios en la nube, puede enfocarse, como cliente de AWS, en utilizar servicios para lograr sus metas. La nube de AWS también ofrece más acceso a los datos de seguridad y un enfoque automatizado para responder a los eventos de seguridad.

# Prácticas recomendadas

Temas


- Aspectos básicos de seguridad
- Identity and Access Management
- Detección
- Protección de la infraestructura
- Protección de datos
- Respuesta a incidentes
- Seguridad de las aplicaciones


## Aspectos básicos de seguridad

La siguiente pregunta se centra en las consideraciones de seguridad. (Para ver una lista de preguntas y prácticas recomendadas sobre seguridad, consulte el Apéndice.).

**SEC 1: ¿Cómo utiliza la carga de trabajo de forma segura?**

Para gestionar su carga de trabajo de forma segura, debe aplicar las prácticas recomendadas generales en todas las áreas de seguridad. Tome los requisitos y procesos que ha definido en materia de excelencia operativa en los niveles organizativo y de carga de trabajo, y aplíquelos a todas las áreas.

Mantenerse al día con las recomendaciones de AWS, las fuentes del sector y la inteligencia sobre amenazas lo ayuda a desarrollar su modelo de amenazas y sus objetivos de control. La automatiz ación de los procesos de seguridad, las pruebas y la validación le permite escalar sus operacion es de seguridad.

En AWS, se recomienda separar las distintas cargas de trabajo por cuenta, según su función y los requisitos de cumplimiento o confidencialidad de los datos.

## Identity and Access Management

La administración de identidades y accesos representa una parte clave de un programa de seguridad de la información, ya que garantiza que solo los usuarios y los componentes autorizados

e identificados puedan acceder a sus recursos (y solo de la forma prevista). Por ejemplo, debería definir las entidades principales (es decir, cuentas, usuarios, roles y servicios que puedan intervenir en su cuenta), crear políticas que hagan referencia a estas entidades e implementar una administración sólida de credenciales. Estos elementos de administración de privilegios constituyen el núcleo de la autenticación y la autorización.

En AWS, la administración de privilegios se apoya principalmente en el servicio AWS Identity and Access Management (IAM), que permite controlar el acceso de usuarios y programas a los servicios y recursos de AWS. Procure aplicar políticas granulares que asignen permisos a cada usuario, grupo, rol o recurso. También puede exigir prácticas de contraseñas seguras; por ejemplo, puede establecer el nivel de complejidad, impedir la reutilización y emplear autenticación multifactor (MFA). Puede usar la federación con su servicio de directorio existente. Cuando las cargas de trabajo requieren que los sistemas tengan acceso a AWS, IAM permite un acceso seguro mediante la asignación de roles, perfiles de instancia, federación de identidades y credenciales temporales.

Las siguientes preguntas se centran en estas consideraciones acerca de la seguridad.

### SEC 2: ¿Cómo se administran las identidades de las personas y las máquinas?

Hay dos tipos de identidades que debe administrar al abordar la operación de cargas de trabajo de AWS seguras. Entender el tipo de identidad que tiene que administrar y a la que otorgar acceso ayuda a comprobar que las identidades adecuadas tengan acceso a los recursos correctos bajo las condiciones adecuadas.

Identidades humanas: sus administradores, desarrolladores, operadores y usuarios finales necesitan una identidad para acceder a sus entornos y aplicaciones de AWS. Son miembros de su organización o usuarios externos con los que colabora y que interactúan con sus recursos de AWS a través de un navegador web, una aplicación cliente o herramientas de línea de comandos interactivas.

Identidades de máquinas: las aplicaciones de servicio, las herramientas operativas y las cargas de trabajo requieren una identidad para hacer solicitudes a los servicios de AWS, como, por ejemplo, para leer datos. Estas identidades incluyen máquinas que se ejecutan en los entornos de AWS, como instancias de Amazon EC2 o funciones de AWS Lambda. También se podrían administrar identidades de máquina para las partes externas que necesiten acceso. Además, es posible que también tenga máquinas fuera de AWS que necesiten acceso al entorno de AWS.

SEC 3: ¿Cómo administra la autenticación para personas y máquinas?

Administre permisos para controlar el acceso a identidades de personas y de máquinas que requieran acceso a AWS y sus cargas de trabajo. Los permisos controlan a qué puede acceder cada usuario y en qué condiciones.

Las credenciales no se deben compartir entre usuarios o sistemas. El acceso de los usuarios se debe conceder empleando un enfoque de privilegio mínimo con prácticas recomendadas, como los requisitos de contraseña y la obligatoriedad de usar MFA. El acceso programático, incluidas las llamadas a la API de los servicios de AWS, debe hacerse mediante credenciales temporales y de privilegio limitado como las emitidas por AWS Security Token Service.

Los usuarios necesitan acceso programático si desean interactuar con AWS fuera de la Consola de administración de AWS. La forma de conceder el acceso programático depende del tipo de usuario que acceda a AWS.

Para conceder acceso programático a los usuarios, seleccione una de las siguientes opciones.

| ¿Qué usuario necesita acceso programático? | Para | Mediante |
|---|---|---|
| IAM | (Recomendado) Use credenciales de consola como credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS. | Siga las instrucciones de la interfaz que desea utilizar:<br>• Para la AWS CLI, consulte Iniciar sesión para el desarrollo local de AWS en la Guía del usuario de AWS Command Line Interface.<br>• Para los AWS SDK, consulte Iniciar sesión para el desarrollo local de AWS en la Guía de referencia de AWS SDK y herramientas. |
| Identidad del personal | Utiliza credenciales temporales para firmar las solicitudes | Siga las instrucciones de la interfaz que desea utilizar: |

| ¿Qué usuario necesita acceso programático? | Para | Mediante |
| --- | --- | --- |
| (Usuarios administrados en el IAM Identity Center) | programáticas a la AWS CLI, los AWS SDK y las API de AWS. | • Para utilizar la AWS CLI, consulte Configuring the AWS CLI to usa AWS IAM Identity Center en la Guía del usuario de AWS Command Line Interface.<br>• Para usar AWS SDK, las herramientas y las API de AWS, consulte IAM Identity Center authentication en la Guía de referencia del SDK y las herramientas de AWS. |
| IAM | Utiliza credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS. | Siguiendo las instrucciones de Uso de credenciales temporales con recursos de AWS de la Guía del usuario de IAM. |

| ¿Qué usuario necesita acceso programático? | Para | Mediante |
| --- | --- | --- |
| IAM | (No recomendado) Utilizar credenciales a largo plazo para firmar las solicitud es programáticas a la AWS CLI, los AWS SDK o las API de AWS. | Siga las instrucciones de la interfaz que desea utilizar:<br>• Para la AWS CLI, consulte Autenticación mediante credenciales de usuario de IAM en la Guía del usuario de AWS Command Line Interface.<br>• Para ver los AWS SDK y las herramientas, consulte Autenticar mediante credenciales a largo plazo en la Guía de referencia de AWS SDK y herramientas.<br>• Para las API de AWS, consulte Administración de claves de acceso para usuarios de IAM en la Guía del usuario de IAM. |

AWS ofrece recursos que pueden ayudar a administrar la identidad y el acceso. Para aprender las prácticas recomendadas, explore nuestros laboratorios prácticos sobre la administración de credenciales y la autenticación, el control del acceso humano y el control del acceso programático.

## Detección

Puede usar los controles detectores para identificar una incidencia o amenaza potencial de seguridad. Son una parte fundamental de los marcos de gobernanza y se pueden usar como complemento de procesos de calidad, para una obligación legal o de cumplimiento y para la identificación de amenazas y respuestas. Hay distintos tipos de controles de detección. Por ejemplo, hacer un inventario de activos y de sus atributos detallados facilita la toma de decisiones (y los controles del ciclo de vida) para ayudar a establecer líneas de base operativas. También puede usar

auditorías internas, un examen de los controles relacionados con los sistemas de información, para garantizar que las prácticas cumplan con las políticas y los requisitos, así como que haya establecido las notificaciones correctas de alertas automatizadas basadas en las condiciones definidas. Estos controles son factores reactivos importantes que ayudan a su organización a identificar la actividad anómala y comprender sus repercusiones.

AWS le permite aplicar controles de detección mediante un procesamiento de registros, eventos y supervisión que posibilita aplicar auditorías, análisis automatizados y alarmas. Los registros de CloudTrail, las llamadas a la API de AWS y CloudWatch proporcionan una supervisión de las métricas con alarmas, y AWS Config proporciona el historial de configuración. Amazon GuardDuty es un servicio administrado de detección de amenazas que supervisa de forma continua posibles comportamientos malintencionados o no autorizados y ayuda a proteger sus cargas de trabajo y cuentas de AWS. También dispone de registros de nivel de servicio; por ejemplo, puede utilizar Amazon Simple Storage Service (Amazon S3) para registrar las solicitudes de acceso.

La siguiente pregunta se centra en las consideraciones de seguridad.

**SEC 4: ¿Cómo detecta e investiga los eventos de seguridad?**

Capture y analice los eventos a partir de registros y métricas para obtener una mejor visibilidad. Actúe ante los eventos de seguridad y las posibles amenazas para proteger las cargas de trabajo.

La administración de registros es una parte importante de una carga de trabajo de Well-Architected por razones que van desde la seguridad y el análisis forense hasta los requisitos normativos o legales. Es fundamental que analice los registros y responda a ellos para poder identificar posibles incidentes de seguridad. AWS proporciona una funcionalidad que facilita la puesta en práctica de la administración de registros, ya que permite definir un ciclo de vida de retención de datos y decidir dónde se conservarán, archivarán y eliminarán los datos. Así es más sencillo y rentable lograr que el manejo de datos sea predecible y fiable.

## Protección de la infraestructura

La protección de la infraestructura abarca las metodologías de control, como la defensa en profundidad, necesarias para ajustarse a las prácticas recomendadas y las obligaciones organizativas o normativas. El uso de estas metodologías es fundamental para el éxito de las operaciones en curso, ya sea en la nube o en las instalaciones.

En AWS es posible implementar la inspección de paquetes con y sin estado, ya sea mediante tecnologías incluidas en AWS o mediante los productos y servicios de socios disponibles en AWS Marketplace. Es recomendable usar Amazon Virtual Private Cloud (Amazon VPC) para crear un entorno privado, asegurado y escalable en el que pueda definir su topología, incluidas puertas de enlace, tablas de enrutamiento y subredes públicas y privadas.

Las siguientes preguntas se centran en estas consideraciones acerca de la seguridad.

### SEC 5: ¿Cómo protege sus redes?

Cualquier carga de trabajo que tenga forma de conexión de red, ya sea internet o una red privada, requiere varias capas de defensa para protegerse de amenazas internas y externas basadas en la red.

### SEC 6: ¿Cómo protege sus recursos de computación?

Los recursos de computación de su carga de trabajo requieren varios niveles de defensa para protegerse de las amenazas externas e internas. Entre los recursos de computación se incluyen instancias de EC2, contenedores, funciones de AWS Lambda, servicios de bases de datos, dispositivos IoT, etc.

Tener varias capas de defensa es aconsejable en cualquier tipo de entorno. En el caso de la protección de la infraestructura, muchos de los conceptos y métodos son válidos para los modelos de nube y en las instalaciones. En un plan eficaz de seguridad de la información, es esencial imponer la protección de los límites, supervisar los puntos de entrada y salida y aplicar de forma exhaustiva registros, supervisión y alertas.

Los clientes de AWS pueden adaptar o reforzar la configuración de una nube de Amazon Elastic Compute Cloud (Amazon EC2), un contenedor de Amazon Elastic Container Service (Amazon ECS) o una instancia de AWS Elastic Beanstalk y mantener esta configuración de forma persistente en una Imagen de máquina de Amazon (AMI) inmutable. Así, todos los nuevos servidores virtuales (instancias) que se lancen con esta AMI reciben la configuración reforzada, tanto si los lanza Auto Scaling como si se lanzan de forma manual.

## Protección de datos

Antes de diseñar un sistema, hay que adoptar prácticas que influyen en la seguridad. Por ejemplo, la clasificación de datos ofrece una manera de categorizar los datos de la organización según los niveles de confidencialidad, mientras que el cifrado protege los datos haciéndolos ininteligibles para el acceso no autorizado. Estas herramientas y técnicas son importantes porque respaldan objetivos como la prevención de pérdidas económicas o el cumplimiento de las obligaciones reglamentarias.

En AWS, las siguientes prácticas facilitan la protección de los datos:

- Como cliente de AWS, controla por completo sus datos.
- AWS simplifica el cifrado de los datos y la administración de claves, además de la rotación regular de las claves, que puede gestionar de forma manual o automatizar fácilmente con AWS.
- Dispone de un registro detallado con contenidos importantes, como el acceso a los archivos y los cambios.
- AWS ha diseñado sistemas de almacenamiento con una resiliencia excepcional. Por ejemplo, Amazon S3 Standard, S3 Standard-IA, S3 One Zone-IA y Amazon Glacier están todos diseñados para proporcionar una durabilidad del 99,999999999 % de objetos en un año determinado. Este nivel de durabilidad corresponde a una pérdida promedio anual esperada del 0,000000001% de los objetos.
- El control de versiones, que puede formar parte de un proceso más amplio de administración del ciclo de vida de los datos, puede protegerlos contra sobrescrituras o eliminaciones accidentales y daños similares.
- AWS nunca inicia un movimiento de datos entre regiones. El contenido situado en una región permanece en esa región a menos que se utilice explícitamente una característica o se emplee un servicio que proporcione esa funcionalidad.

Las siguientes preguntas se centran en estas consideraciones acerca de la seguridad.

### SEC 7: ¿Cómo clasifica sus datos?

La clasificación proporciona una forma de categorizar los datos, basada en el nivel de importanc ia y la confidencialidad, para ayudarlo a determinar los controles de protección y de conservación adecuados.

SEC 8: ¿Cómo protege los datos en reposo?

Proteja sus datos en reposo mediante la implementación de varios controles para reducir el riesgo de acceso no autorizado o mala gestión.

SEC 9: ¿Cómo protege sus datos en tránsito?

Proteja sus datos en tránsito mediante la implementación de varios controles para reducir el riesgo de acceso no autorizado o pérdida.

AWS proporciona múltiples medios para cifrar los datos en reposo y en tránsito. Nuestros servicios incorporan características que facilitan el cifrado de los datos. Por ejemplo, hemos implementado el cifrado del servidor (SSE) en Amazon S3 para facilitar el almacenamiento de datos de forma cifrada. También puede hacer que Elastic Load Balancing (ELB) maneje todo el proceso de cifrado y descifrado HTTPS (generalmente conocido como terminación SSL).

## Respuesta a incidentes

Incluso con controles detectores y de prevención extremadamente eficaces, la organización debería continuar aplicando procesos para responder a incidencias de seguridad y mitigar su posible impacto. La arquitectura de la carga de trabajo afecta considerablemente a la capacidad de los equipos de operar de forma eficaz durante una incidencia, aislar o contener sistemas y restaurar operaciones a un estado conocido correcto. Si prepara las herramientas y el acceso en previsión de un incidente de seguridad, y practica periódicamente la respuesta a incidentes durante los días de juego, comprobará que su arquitectura va a posibilitar una investigación y una recuperación sin demoras.

En AWS, las siguientes prácticas facilitan una respuesta efectiva frente a cualquier incidencia:

- Dispone de un registro detallado con contenidos importantes, como el acceso a los archivos y los cambios.
- Los eventos pueden procesarse automáticamente y lanzar herramientas que automatizan las respuestas mediante el uso de las API de AWS.
- Puede preaprovisionar herramientas y una sala limpia gracias a AWS CloudFormation. Esto permite hacer análisis forenses en un ambiente seguro y aislado.

La siguiente pregunta se centra en las consideraciones de seguridad.

SEC 10: ¿Cómo anticipa los incidentes, responde a ellos y se recupera?

La preparación es fundamental para investigar de forma oportuna y efectiva, dar respuesta a incidentes de seguridad, así como para minimizar posibles interrupciones en su organización.

Asegúrese de disponer de un método para conceder rápidamente acceso a su equipo de seguridad y automatice tanto el aislamiento de las instancias como la captura de datos y de estados para el análisis forense.

## Seguridad de las aplicaciones

La seguridad de las aplicaciones (AppSec) describe el proceso general de diseño, compilación y comprobación de las propiedades de seguridad de las cargas de trabajo que desarrolla. Debe contar con personal debidamente formado en su organización, comprender las propiedades de seguridad de su infraestructura de creación y lanzamiento, y utilizar la automatización para identificar problemas de seguridad.

La adopción de pruebas de seguridad de las aplicaciones como parte habitual del ciclo de vida de desarrollo del software (SDLC) y de los procesos posteriores al lanzamiento contribuye a asegurar que se dispone de un mecanismo estructurado para identificar, corregir y evitar que los problemas de seguridad de las aplicaciones entren en el entorno de producción.

La metodología de desarrollo de las aplicaciones debe incluir controles de seguridad a medida que diseña, compila, implementa y opera las cargas de trabajo. Al hacerlo, ajuste el proceso a fin de reducir continuamente los defectos y minimizar la deuda técnica. Por ejemplo, el uso de modelos de amenazas en la fase de diseño ayuda a detectar defectos de diseño en una fase temprana, lo que hace que sea más fácil y menos costoso solucionarlos, en lugar de esperar y mitigarlos más adelante.

El costo y la complejidad que supone resolver los defectos suelen ser menores cuanto antes se detecten en el SDLC. La forma más fácil de resolver los problemas es no tenerlos en primer lugar, por lo que empezar con un modelo de amenazas ayuda a centrarse en los resultados correctos desde la fase de diseño. A medida que su programa de AppSec madura, puede aumentar la cantidad de pruebas que se llevan a cabo mediante la automatización, mejorar la fidelidad de los comentarios para los creadores y reducir el tiempo necesario para las revisiones de seguridad. Todas estas

medidas mejoran la calidad del software que crea y aumentan la velocidad de entrega de las características a la producción.

Estas directrices de implementación se centran en cuatro áreas: la organización y la cultura, la seguridad de la canalización, la seguridad en la canalización y la administración de las dependencias. Cada área proporciona un conjunto de principios que puede implementar y ofrece una visión integral de cómo diseñar, desarrollar, compilar, implementar y operar cargas de trabajo.

En AWS existen una serie de estrategias diferentes que puede utilizar a la hora de acometer su programa de seguridad de las aplicaciones. Algunas de ellas se basan en la tecnología, mientras que otras se centran en las personas y los aspectos organizativos de su programa de seguridad de las aplicaciones.

La siguiente pregunta se centra en las consideraciones de seguridad de las aplicaciones.

SEC 11: ¿Cómo incorpora y valida las propiedades de seguridad de las aplicaciones durante el ciclo de vida de diseño, desarrollo e implementación?

La capacitación de los usuarios, las pruebas mediante automatización, el conocimiento de las dependencias y la validación de las propiedades de seguridad de herramientas y aplicaciones contribuyen a reducir la probabilidad de que se produzcan problemas de seguridad en las cargas de trabajo de producción.

## Recursos

Consulte los siguientes recursos para obtener más información sobre nuestras prácticas recomendadas en materia de seguridad.

### Documentación

- AWS Seguridad en la nube de
- AWS Cumplimiento de
- AWS Blog de seguridad de
- AWS Security Maturity Model

### Documento técnico

- Pilar de seguridad
- AWS Security Overview
- AWS Risk and Compliance

### Video

- AWS Situación de la seguridad de
- Shared Responsibility Overview

## Fiabilidad

El pilar de fiabilidad abarca la capacidad de una carga de trabajo para llevar a cabo su función prevista de forma correcta y coherente cuando se espera que lo haga. Esto incluye la capacidad de utilizar y probar la carga de trabajo a lo largo de todo su ciclo de vida. En este documento se incluye orientación de prácticas recomendadas para la implementación de cargas de trabajo fiables en AWS.

El pilar de fiabilidad proporciona información general sobre los principios de diseño, prácticas recomendadas y preguntas. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de fiabilidad.

Temas

- Principios de diseño
- Definición
- Prácticas recomendadas
- Recursos

### Principios de diseño

Existen cinco principios de diseño para la fiabilidad en la nube:

- Recuperación automática de un error: al supervisar un sistema de indicadores clave de rendimiento (KPI), se puede iniciar la automatización cuando se supera un umbral. Estos KPI deben ser una medida del valor de negocio, no de los aspectos técnicos del funcionamiento del servicio. De este modo, se hace posible la notificación y el seguimiento automático de los errores,

así como los procesos de recuperación automatizada que pueden solucionar o corregir el error. Con una automatización más sofisticada, es posible anticipar y solucionar errores antes de que sucedan.

- Prueba de los procedimientos de recuperación: en un entorno en las instalaciones, a menudo se hacen pruebas para ver si una carga de trabajo funciona en una situación concreta. Normalmente, las pruebas no se usan para comprobar estrategias de recuperación. En la nube, puede probar los errores de la carga de trabajo y validar los procedimientos de recuperación. Puede usar la automatización para simular diferentes errores o recrear escenarios que anteriormente han producido algún error. Esto expone vías de error que puede probar y arreglar antes de que se produzca una situación de error real, lo que reduce el riesgo.
- Escalado horizontal para aumentar la disponibilidad agregada de la carga de trabajo: reemplace un gran recurso por varios recursos pequeños para reducir el efecto de un solo error en toda la carga de trabajo. Distribuya las solicitudes a través de varios recursos más pequeños para verificar que no compartan el mismo error.
- No más conjeturas sobre la capacidad: un factor común de error de los sistemas en las instalaciones es la saturación de recursos, cuando las demandas que se hacen a una carga de trabajo superan su capacidad (este es a menudo el objetivo de los ataques de denegación de servicio). En la nube, se puede supervisar la demanda y el uso de la carga de trabajo, además de automatizar la incorporación o eliminación de recursos de forma automatizada para mantener un nivel más eficiente y satisfacer la demanda sin tener un aprovisionamiento excesivo o insuficiente. Aún hay límites, pero algunas cuotas se pueden controlar, mientras que otras se pueden administrar (consulte Manage Service Quotas and Constraints).
- Administración de los cambios mediante la automatización: los cambios que se apliquen a la infraestructura deben hacerse mediante la automatización. Entre los cambios que se tienen que administrar se encuentran los de la automatización, de los que, posteriormente, se puede hacer un seguimiento y una revisión.

## Definición

Existen cuatro áreas de prácticas recomendadas para la fiabilidad en la nube:

- Principios básicos
- Arquitectura de la carga de trabajo
- Administración de cambios
- Administración de errores

Para lograr fiabilidad hay que empezar por los cimientos: un entorno en el que las Service Quotas y la topología de la red se adapten a la carga de trabajo. La arquitectura de la carga de trabajo del sistema distribuido debe estar diseñada para prevenir y mitigar los errores. La carga de trabajo debe gestionar los cambios en la demanda o los requisitos, y debe estar diseñada para detectar errores y repararse automáticamente.

## Prácticas recomendadas

Temas


- Principios básicos
- Arquitectura de la carga de trabajo
- Administración de cambios
- Administración de errores


### Principios básicos

Los requisitos fundamentales son aquellos cuyo ámbito va más allá de una única carga de trabajo o proyecto. Antes de diseñar la arquitectura de cualquier sistema, los requisitos básicos que afectan a la fiabilidad deberían estar aplicados. Por ejemplo, es preciso que haya suficiente ancho de banda de la red en el centro de datos.

En AWS, la mayor parte de estos requisitos básicos están incorporados o pueden abordarse según corresponda. La nube está diseñada para que sea, en esencia, ilimitada, por lo que la responsabilidad de brindar suficiente capacidad de computación y de red recae en AWS. Esto permite que pueda cambiar la asignación y el tamaño de los recursos según sea necesario.

Las siguientes preguntas se centran en estas consideraciones de fiabilidad. (Para ver una lista de preguntas y prácticas recomendadas sobre fiabilidad, consulte el Apéndice).

**REL 1: ¿Cómo administra las Service Quotas y las restricciones?**

Para las arquitecturas de carga de trabajo basadas en la nube, existen Service Quotas (que también se denominan límites de servicio). Estas cuotas existen para evitar el aprovisionamiento accidental de más recursos de los que se necesitan y para limitar las tasas de solicitudes en las operaciones de la API a fin de proteger los servicios contra el abuso. También hay limitaciones de recursos, por ejemplo, la velocidad a la que se pueden introducir bits por un cable de fibra óptica o la cantidad de almacenamiento en un disco físico.

REL 2: ¿Cómo planifica la topología de la red?

Las cargas de trabajo suelen existir en varios entornos. Estos incluyen varios entornos de nube (tanto de acceso público como privados) y, posiblemente, la infraestructura de su centro de datos existente. Los planes deben incluir consideraciones sobre la red, como la conectividad dentro y entre sistemas, la administración de direcciones IP públicas, la administración de direcciones IP privadas y la resolución de nombres de dominio.

## Arquitectura de la carga de trabajo

Una carga de trabajo fiable comienza por tomar decisiones de diseño anticipadas tanto para el software como para la infraestructura. Sus elecciones respecto a la arquitectura afectarán al comportamiento de su carga de trabajo en todos los pilares de Well-Architected. Para la fiabilidad, debe seguir patrones específicos.

En AWS, los desarrolladores de la carga de trabajo pueden elegir los lenguajes y las tecnologías que usar. Los SDK de AWS simplifican la codificación al proporcionar API específicas de lenguajes para los servicios de AWS. Estos SDK, más la elección del lenguaje, permiten a los desarrolladores implementar las prácticas recomendadas de fiabilidad enumeradas aquí. Los desarrolladores también pueden leer y aprender sobre cómo Amazon crea y opera el software en Amazon Builders' Library.

Las siguientes preguntas se centran en estas consideraciones de fiabilidad.

REL 3: ¿Cómo diseña la arquitectura de servicio de su carga de trabajo?

Desarrolle cargas de trabajo escalables y fiables mediante una arquitectura orientada a servicios (SOA) o una arquitectura de microservicios. La arquitectura orientada a servicios (SOA) es hacer que los componentes de software se puedan reutilizar mediante interfaces de servicio. La arquitectura de microservicios va más allá, para hacer que los componentes sean más pequeños y sencillos.

REL 4: ¿Cómo diseña las interacciones en un sistema distribuido para evitar errores?

Los sistemas distribuidos se basan en las redes de comunicaciones para interconectar component es, como servidores o servicios. Su carga de trabajo debe funcionar de manera fiable a pesar

REL 4: ¿Cómo diseña las interacciones en un sistema distribuido para evitar errores?

de la pérdida de datos o la latencia en estas redes. Los componentes del sistema distribuido deben funcionar de manera que no afecten negativamente a otros componentes o a la carga de trabajo. Estas prácticas recomendadas previenen los fallos y mejoran el tiempo medio entre errores (MTBD).

REL 5: ¿Cómo diseña las interacciones en un sistema distribuido para mitigar o tolerar errores?

Los sistemas distribuidos dependen de las redes de comunicaciones para interconectar componentes, como servidores o servicios. Su carga de trabajo debe funcionar de manera fiable aunque se pierdan datos o haya latencia en estas redes. Los componentes del sistema distribui do deben funcionar de manera que no afecten negativamente a otros componentes o a la carga de trabajo. Estas prácticas recomendadas permiten que las cargas de trabajo toleren el estrés o los errores, se recuperen más rápidamente de ellos y mitiguen el impacto de dichos errores. El resultado es un tiempo medio de recuperación (MTTR) mejor.

## Administración de cambios

Se deben prever y ajustar los cambios en la carga de trabajo o el entorno para lograr un funcionamiento fiable de la carga de trabajo. Los cambios incluyen aquellos impuestos a la carga de trabajo, como los picos de demanda, además de los inherentes a ella, como las implementaciones de características o las revisiones de seguridad.

AWS le permite supervisar el comportamiento de una carga de trabajo y automatizar la respuesta a los KPI. Por ejemplo, la carga de trabajo puede agregar servidores adicionales a medida que la carga de trabajo gane más usuarios. Puede controlar quién tiene permisos para aplicar cambios en la carga de trabajo e inspeccionar el historial de cambios.

Las siguientes preguntas se centran en estas consideraciones de fiabilidad.

REL 6: ¿Cómo supervisa los recursos de las cargas de trabajo?

Los registros y las métricas son herramientas poderosas para obtener información sobre el estado de su carga de trabajo. Puede configurar su carga de trabajo para supervisar los registros y las métricas y enviar notificaciones cuando se superen los umbrales o se produzcan eventos significa

REL 6: ¿Cómo supervisa los recursos de las cargas de trabajo?

tivos. La supervisión permite que su carga de trabajo reconozca cuándo se cruzan umbrales de bajo rendimiento o se producen errores, para que pueda recuperarse de los errores de forma automática una vez recibida una respuesta.

REL 7: ¿Cómo diseña su carga de trabajo para que se adapte a los cambios en la demanda?

Una carga de trabajo escalable proporciona elasticidad para agregar o eliminar recursos automáticamente, de modo que se ajusten perfectamente a la demanda actual en cualquier momento dado.

REL 8: ¿Cómo implementa los cambios?

Los cambios controlados son necesarios para implementar nuevas funcionalidades y comprobar que las cargas de trabajo y el entorno operativo ejecuten software conocido y que puedan recibir revisiones o reemplazos de manera predecible. Si estos cambios no se controlan, es difícil predecir su efecto o abordar los problemas que surjan a causa de ellos.

Cuando diseña la arquitectura de una carga de trabajo para agregar y eliminar recursos de forma automática como respuesta a los cambios solicitados, aumenta la fiabilidad a la par que se garantiza que el éxito del negocio no se convierta en una carga. Al contar con supervisión, el equipo recibirá alertas automáticas cuando los KPI se desvíen de las reglas esperadas. Los registros automáticos de los cambios aplicados en el entorno le permiten inspeccionar e identificar rápidamente aquellas medidas que hayan repercutido en la fiabilidad. Controlar la administración de cambios garantiza que se puedan aplicar reglas que ayuden a alcanzar el grado de fiabilidad deseado.

## Administración de errores

De cualquier sistema con una complejidad razonable se esperan errores. La fiabilidad requiere que la carga de trabajo conozca los errores a medida que ocurren y que actúe para evitar que afecten a la disponibilidad. Las cargas de trabajo deben ser capaces de tolerar errores y de repararlos de forma automática.

Gracias a AWS, podrá aprovechar la automatización para reaccionar a los datos de supervisión. Por ejemplo, cuando una métrica concreta pasa un umbral, podrá iniciar una acción automática para solucionar el problema. Además, puede reemplazar un recurso que genere un error y forme parte del entorno de producción por uno nuevo y analizar dicho recurso fuera de banda en lugar de intentar diagnosticar y arreglar el recurso del error. Ya que la nube permite soportar versiones temporales de todo un sistema a bajo costo, puede usar las pruebas automáticas para comprobar los procesos de recuperación completos.

Las siguientes preguntas se centran en estas consideraciones de fiabilidad.

### REL 9: ¿Cómo hace una copia de seguridad de los datos?

Realice copias de seguridad de los datos, las aplicaciones y la configuración para cumplir con los requisitos de objetivos de tiempo de recuperación (RTO) y objetivos de punto de recuperación (RPO).

### REL 10: ¿Cómo usa el aislamiento de errores para proteger su carga de trabajo?

El aislamiento de fallos limita el impacto de un fallo en un componente o sistema a un límite definido. Mediante un aislamiento adecuado, los componentes que se encuentran fuera del límite no se ven afectados por el error. Hacer que la carga de trabajo supere varios límites de aislamien to de fallos puede hacer que sea más resistente a los fallos.

### REL 11: ¿Cómo diseña su carga de trabajo para que soporte los errores de los componentes?

Las cargas de trabajo con un requisito de alta disponibilidad y un tiempo de recuperación (MTTR) bajo deben diseñarse para que sean resilientes.

### REL 12: ¿Cómo pone a prueba la fiabilidad?

Una vez diseñada la carga de trabajo para que sea resiliente al estrés de producción, las pruebas son la única forma de comprobar que funcionará según lo previsto y proporcionará la resiliencia esperada.

REL 13: ¿Cómo planifica la recuperación de desastres (DR)?

Disponer de copias de seguridad y de componentes de cargas de trabajo redundantes es el principio de su estrategia de DR. El RTO y el RPO son los objetivos de restauración de las cargas de trabajo. Estos se definen en función de las necesidades del negocio. Implemente una estrategia para satisfacer estos objetivos teniendo en cuenta las ubicaciones y la función de los recursos de las cargas de trabajo y los datos. La probabilidad de una interrupción y el costo de recuperación son también factores clave que ayudan a conocer el valor empresarial de proporcio nar recuperación de desastres para una carga de trabajo.

Haga una copia de seguridad de los datos de forma regular y ponga a prueba estos archivos para garantizar que pueda recuperarse tanto de los errores físicos como de los lógicos. Un factor clave para administrar los errores es probar de forma frecuente y automática las cargas de trabajo que causan errores para después observar cómo se recuperan. Haga esto de manera regular y asegúrese de que dichas pruebas también se inicien tras aplicar cambios importantes en la carga de trabajo. Haga un seguimiento activo de los KPI, el objetivo de tiempo de recuperación (RTO) y el objetivo de punto de recuperación (RPO) para evaluar la resiliencia de la carga de trabajo (especialmente, cuando se pongan a prueba situaciones en las que se produzca un error). Hacer el seguimiento de los KPI será de ayuda para identificar y mitigar los puntos únicos de error. El objetivo es someter los procesos de recuperación de la carga de trabajo a pruebas exhaustivas para que sepa que puede recuperar todos los datos y continuar brindando servicios a los clientes, aunque se experimenten problemas prolongados. Los procesos de recuperación deberían efectuarse igual de bien que los procesos de producción normales.

## Recursos

Consulte los siguientes recursos para obtener más información sobre nuestras prácticas recomendadas de fiabilidad.

### Documentación

- Documentación de AWS
- Infraestructura global de AWS
- AWS Auto Scaling: How Scaling Plans Work
- ¿Qué es AWS Backup?

### Documento técnico

- Reliability Pillar: AWS Well-Architected
- Implementing Microservices on AWS

# Eficiencia del rendimiento

El pilar de eficiencia del rendimiento incluye la capacidad para utilizar los recursos de la nube de forma eficaz a fin de que satisfagan los requisitos de rendimiento y para mantener dicha eficacia a medida que la demanda cambia y las tecnologías evolucionan.

El pilar de eficiencia del rendimiento ofrece una visión general de principios de diseño, prácticas recomendadas y preguntas. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de eficiencia del rendimiento.

Temas

- Principios de diseño
- Definición
- Prácticas recomendadas
- Recursos

## Principios de diseño

Existen cinco principios de diseño para la eficiencia del rendimiento en la nube:

- Democratización de las tecnologías avanzadas: facilite a su equipo la implementación de tecnologías avanzadas mediante la delegación de tareas complejas a su proveedor de servicios en la nube. En lugar de pedir a su equipo de TI que aprenda a alojar y ejecutar una tecnología nueva, considere la posibilidad de consumir la tecnología como un servicio. Por ejemplo, las bases de datos NoSQL, la transcodificación de medios y el machine learning son tecnologías que requieren conocimientos especializados. En la nube, estas tecnologías se convierten en servicios que su equipo puede consumir, lo que permite que se centre en el desarrollo de productos, y no en aprovisionar o administrar recursos.
- Adopción de un enfoque global en cuestión de minutos: la implementación de su carga de trabajo en varias regiones de AWS del mundo le permite ofrecer una menor latencia y una mejor experiencia a sus clientes con un costo mínimo.

- Uso de arquitecturas sin servidor: las arquitecturas sin servidor eliminan la necesidad de ejecutar y mantener servidores físicos para las actividades de computación tradicionales. Por ejemplo, los servicios de almacenamiento sin servidor pueden servir como sitios web estáticos, con lo que se elimina la necesidad de servidores web. Además, los servicios basados en eventos pueden alojar código. Esto elimina la carga operativa de administrar servidores físicos y puede reducir los costos de transacciones porque los servicios administrados operan a escala de la nube.
- Experimentación más frecuente: los recursos virtuales y automatizables permiten hacer pruebas comparativas con rapidez mediante diferentes tipos de instancias, almacenamiento y configuraciones.
- Consideración de la simpatía mecánica: descubra cómo se consumen los servicios en la nube y utilice siempre el enfoque tecnológico que mejor se adapte a sus objetivos de carga de trabajo. Por ejemplo, piense en los patrones de acceso a datos al elegir los enfoques de base de datos o de almacenamiento.

## Definición

Existen cinco áreas de prácticas recomendadas para la eficiencia del rendimiento en la nube:

- Selección de la arquitectura
- Computación y hardware
- Administración de datos
- Redes y entrega de contenido
- Proceso y cultura

Adopte un enfoque basado en datos para crear una arquitectura de alto rendimiento. Recopile datos sobre todos los aspectos de la arquitectura, desde el diseño general hasta la selección y configuración de los tipos de recursos.

Revisar periódicamente sus opciones validará que aprovecha la continua evolución de la nube de AWS. Mediante la supervisión verifica que es consciente de cualquier desviación del rendimiento esperado. Haga compensaciones en su arquitectura para mejorar el rendimiento, tales como el uso de la compresión o el almacenamiento en caché, o bien la mitigación de los requisitos de coherencia.

## Prácticas recomendadas

Temas

- Selección de la arquitectura
- Computación y hardware
- Administración de datos
- Redes y entrega de contenido
- Proceso y cultura

## Selección de la arquitectura

La solución óptima para una carga de trabajo concreta varía y las soluciones suelen combinar varios enfoques. Las cargas de trabajo de Well-Architected utilizan varias soluciones y admiten diferentes características para mejorar el rendimiento.

Los recursos de AWS están disponibles en muchos tipos y configuraciones, lo que facilita encontrar un enfoque que se ajuste a sus necesidades. También puede encontrar opciones que no se logran fácilmente con una infraestructura en las instalaciones. Por ejemplo, un servicio administrado como Amazon DynamoDB ofrece una base de datos NoSQL completamente administrada con una latencia de milisegundos de un solo dígito a cualquier escala.

La siguiente pregunta se centra en estas consideraciones para mejorar la eficacia del rendimiento. (Para ver una lista de preguntas sobre la eficiencia del rendimiento y las prácticas recomendadas, consulte el Apéndice.).

PERF 1: ¿Cómo selecciona los recursos y los patrones de arquitectura en la nube adecuados para su carga de trabajo?

A menudo, se requieren varios enfoques para obtener un rendimiento más eficaz en una carga de trabajo. Los sistemas Well-Architected utilizan varias soluciones y características para mejorar el rendimiento.

## Computación y hardware

La elección óptima de computación para una carga de trabajo concreta puede variar en función del diseño de la aplicación, los patrones de uso y los ajustes de configuración. Las arquitecturas pueden usar diferentes opciones de computación para varios componentes y admiten diferentes características para mejorar el rendimiento. No seleccionar la opción de computación correcta para una arquitectura puede disminuir la eficiencia del rendimiento.

En AWS, la computación está disponible de tres formas: instancias, contenedores y funciones.

- Las instancias son servidores virtualizados que hacen posible cambiar sus capacidades con un botón o una llamada a la API. Como las decisiones sobre los recursos en la nube no son fijas, puede experimentar con diferentes tipos de servidores. En AWS, estas instancias de servidor virtual se presentan en diferentes familias y tamaños, y ofrecen una amplia variedad de capacidades, incluidas unidades de estado sólido (SSD) y unidades de procesamiento gráfico (GPU).
- Los contenedores son un método de virtualización de sistemas operativos que permite ejecutar una aplicación y sus dependencias en procesos aislados de recursos. AWS Fargate es computación sin servidor para contenedores o puede utilizar Amazon EC2 si necesita controlar la instalación, la configuración y la administración de su entorno de computación. También puede elegir entre varias plataformas de orquestación de contenedores: Amazon Elastic Container Service (ECS) o Amazon Elastic Kubernetes Service (EKS).
- Las funciones extraen el entorno de ejecución del código que desea aplicar. Por ejemplo, AWS Lambda permite ejecutar código sin ejecutar una instancia.

La siguiente pregunta se centra en estas consideraciones para mejorar la eficacia del rendimiento.

PERF 2: ¿Cómo selecciona y utiliza los recursos de computación en su carga de trabajo?

La solución de computación más eficaz para cada carga de trabajo depende del diseño de las aplicaciones, de los patrones de uso y de las opciones de configuración. Las arquitecturas pueden usar diferentes soluciones de computación para varios componentes y activar diferentes caracterí sticas que mejoren el rendimiento. Seleccionar las soluciones de computación incorrectas para una arquitectura puede disminuir la eficiencia del rendimiento.

## Administración de datos

La solución de administración de datos óptima para un sistema concreto varía según el tipo de datos (bloque, archivo u objeto), patrones de acceso (aleatorio o secuencial), rendimiento requerido, frecuencia de acceso (en línea, fuera de línea, archivo), frecuencia de actualización (WORM, dinámica), y restricciones de disponibilidad y durabilidad. Las cargas de trabajo de Well-Architected utilizan almacenes de datos diseñados específicamente que admiten diferentes características para mejorar el rendimiento.

En AWS, el almacenamiento está disponible en tres formas: objeto, bloque y archivo.

- El almacenamiento de objetos proporciona una plataforma escalable y duradera que permite acceder a los datos desde cualquier ubicación de Internet para el contenido generado por los usuarios, el archivado activo, la computación sin servidor, el almacenamiento de macrodatos o las copias de seguridad y recuperación. Amazon Simple Storage Service (Amazon S3) es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos, seguridad y rendimiento líderes del sector. Amazon S3 se creó desde cero para ofrecer un 99,999999999 % (11 nueves) de durabilidad y almacena datos de millones de aplicaciones para empresas de todo el mundo.
- El almacenamiento en bloques proporciona un almacenamiento en bloques de alta disponibilidad, uniforme y de baja latencia para cada host virtual y es análogo al almacenamiento con conexión directa (DAS) o a una red de área de almacenamiento (SAN). Amazon Elastic Block Store (Amazon EBS) se ha diseñado para cargas de trabajo que requieren un almacenamiento persistente al que pueden acceder las instancias de EC2 y lo ayuda a optimizar las aplicaciones con la capacidad de almacenamiento, el rendimiento y el costo adecuados.
- El almacenamiento de archivos proporciona acceso a un sistema de archivos compartido en varios sistemas. Las soluciones de almacenamiento de archivos como Amazon Elastic File System (Amazon EFS) son perfectas para los casos de uso como grandes repositorios de contenido, entornos de desarrollo, almacenes de medios o directorios domésticos de usuario. Amazon FSx es eficiente y rentabiliza el lanzamiento y la ejecución de sistemas de archivos populares para que pueda aprovechar los conjuntos de características completos y el rápido rendimiento de los sistemas de archivos de código abierto y con licencia comercial más utilizados.

La siguiente pregunta se centra en estas consideraciones para mejorar la eficacia del rendimiento.

PERF 3: ¿Cómo almacena y administra los datos de su carga de trabajo y cómo accede a ellos?

La solución de almacenamiento más eficiente para un sistema varía según el tipo de operación de acceso (bloque, archivo u objeto), patrones de acceso (aleatorio o secuencial), rendimiento requerido, frecuencia de acceso (en línea, fuera de línea, archivo), frecuencia de actualización (WORM, dinámica) y restricciones de disponibilidad y durabilidad. Los sistemas Well-Architected utilizan varias soluciones de almacenamiento y activan diferentes características para mejorar el rendimiento y utilizar los recursos de manera eficiente.

## Redes y entrega de contenido

La solución de redes óptima para una carga de trabajo varía según los requisitos de latencia, rendimiento, fluctuaciones y ancho de banda. Las limitaciones físicas, como los recursos de usuario o en las instalaciones, determinan las opciones de ubicación. Estas limitaciones pueden compensarse con las ubicaciones periféricas o la ubicación de los recursos.

En AWS, las redes se virtualizan y están disponibles en diversos tipos y configuraciones. Esto facilita la adaptación de las redes a sus necesidades. AWS ofrece características de producto, como, por ejemplo, redes mejoradas, instancias optimizadas para redes de Amazon EC2, aceleración de la transferencia de Amazon S3 y Amazon CloudFront dinámico, con el fin de optimizar el tráfico de red. AWS también ofrece características de red, como enrutamiento de latencia de Amazon Route 53, puntos de conexión de Amazon VPC, AWS Direct Connect y AWS Global Accelerator, para reducir la distancia o las fluctuaciones de red.

La siguiente pregunta se centra en estas consideraciones para mejorar la eficacia del rendimiento.

**PERF 4: ¿Cómo selecciona y configura los recursos de red en su carga de trabajo?**

Esta parte incluye la guía y las prácticas recomendadas para diseñar, configurar y operar soluciones de redes y entrega de contenido eficientes en la nube.

## Proceso y cultura

Al diseñar cargas de trabajo, hay principios y prácticas que puede adoptar con el fin de ayudarle a ejecutar mejor cargas de trabajo en la nube eficientes y de alto rendimiento. Para adoptar una cultura que fomente la eficiencia del rendimiento de las cargas de trabajo en la nube, tenga en cuenta estos principios y prácticas clave.

Tenga en cuenta estos principios clave para crear esta cultura:

- Infraestructura como código: defina su infraestructura como código al usar enfoques como las plantillas de AWS CloudFormation. El uso de plantillas le permite colocar su infraestructura en un control fuente junto con su código de aplicación y configuraciones. Esto le permite aplicar las mismas prácticas que utiliza para desarrollar software en su infraestructura con la finalidad de que pueda iterar rápidamente.
- Canalización de implementación: utilice una canalización de integración continua/implementación continua (CI/CD), (por ejemplo, el repositorio del código fuente, los sistemas de diseño, la

implementación y la automatización de pruebas) para implementar su infraestructura. Esto le permite implementar de manera repetible, coherente y por un bajo costo mientras itera.

- Métricas bien definidas: configure y supervise métricas para capturar indicadores clave de rendimiento (KPI). Recomendamos que utilice tanto métricas técnicas como comerciales. Para aplicaciones móviles o sitios web, las métricas clave registran el tiempo para el primer byte o la renderización. Otras métricas que generalmente se aplican incluyen el recuento de subprocesos, la tasa de recopilación de elementos no utilizados y los estados de espera. Las métricas comerciales, como el costo acumulado agregado por solicitud, puede alertarle sobre formas de reducir costos. Considere con cuidado cómo planifica interpretar las métricas. Por ejemplo, podría elegir el percentil máximo o el 99.º, en vez del promedio.
- Prueba de rendimiento automática: como parte de su proceso de implementación, lance automáticamente las pruebas de rendimiento después de que las pruebas de ejecución más rápida se hayan aprobado con éxito. La automatización debería crear un nuevo entorno, establecer condiciones iniciales como datos de prueba y luego ejecutar una serie de puntos de referencia y pruebas de carga. Los resultados de estas pruebas deberían estar vinculados al diseño, para que pueda seguir los cambios del rendimiento en el tiempo. Para las pruebas de larga ejecución, puede hacer que esta parte de la canalización sea asíncrona al resto del diseño. Alternativamente, podría ejecutar las pruebas de rendimiento durante la noche con instancias de spot de Amazon EC2.
- Generación de cargas: debería crear una serie de scripts de prueba que repliquen trayectos de usuario sintéticos o pregrabados. Estos scripts deben ser idempotentes y no acoplados; podría necesitar incluir scripts de precalentamiento para rendir resultados válidos. En la medida de lo posible, sus scripts de prueba deben replicar el comportamiento de uso en producción. Puede utilizar soluciones de software o de software como servicio (SaaS) para generar la carga. Considere la posibilidad de utilizar soluciones de AWS Marketplace e instancias de spot, ya que pueden ser formas rentables de generar carga.
- Visibilidad de rendimiento: las métricas clave deben ser visibles para su equipo, especialmente las métricas para cada versión de diseño. Esto le permite ver cualquier tendencia significativa, sea positiva o negativa, con el paso del tiempo. También debería exponer métricas en la cantidad de errores o excepciones para garantizar que está poniendo a prueba un sistema de trabajo.
- Visualización: utilice técnicas de visualización que dejen claro dónde se presentan problemas de rendimiento, puntos críticos, estados de espera o un uso bajo. Superponga las métricas de rendimiento sobre los diagramas de arquitectura: los gráficos de llamadas o el código pueden ayudar a identificar problemas con mayor rapidez.

- Proceso de revisión periódico: el mal funcionamiento de las arquitecturas suele ser el resultado de un proceso de revisión del rendimiento inexistente o deficiente. Si su arquitectura tiene un bajo rendimiento, la implementación de un proceso de revisión del rendimiento le permitirá impulsar la mejora iterativa.
- Optimización continua: adopte una cultura que optimice continuamente la eficiencia del rendimiento de su carga de trabajo en la nube.

La siguiente pregunta se centra en estas consideraciones para mejorar la eficacia del rendimiento.

PERF 5: ¿Qué proceso utiliza para lograr una mayor eficiencia en el rendimiento de su carga de trabajo?

Al diseñar cargas de trabajo, hay principios y prácticas que puede adoptar con el fin de ayudarle a ejecutar mejor cargas de trabajo en la nube eficientes y de alto rendimiento. Para adoptar una cultura que fomente la eficiencia del rendimiento de las cargas de trabajo en la nube, tenga en cuenta estos principios y prácticas clave.

## Recursos

Consulte los siguientes recursos para obtener más información sobre nuestras prácticas recomendadas en la eficiencia del rendimiento.

### Documentación

- Amazon S3 Performance Optimization
- Rendimiento de los volúmenes de Amazon EBS

### Documento técnico

- Performance Efficiency Pillar

### Video

- AWS re:Invent 2019: Amazon EC2 foundations (CMP211-R2)
- AWS re:Invent 2019: Leadership session: Storage state of the union (STG201-L)

- AWS re:Invent 2019: Leadership session: AWS purpose-built databases (DAT209-L)
- AWS re:Invent 2019: Connectivity to AWS and hybrid AWS network architectures (NET317-R1)
- AWS re:Invent 2019: Powering next-gen Amazon EC2: Deep dive into the Nitro system (CMP303-R2)
- AWS re:Invent 2019: Scaling up to your first 10 million users (ARC211-R)

# Optimización de costos

El pilar de optimización de costos incluye la capacidad de ejecutar sistemas para ofrecer valor empresarial al precio más bajo posible.

El pilar de optimización de costos proporciona información general sobre los principios de diseño, prácticas recomendadas y preguntas. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de optimización de costos.

Temas


- Principios de diseño
- Definición
- Prácticas recomendadas
- Recursos


## Principios de diseño

Existen cinco principios de diseño para la optimización de costos en la nube:

- Implementación de la administración financiera en la nube: para alcanzar el éxito financiero y acelerar la materialización del valor empresarial en la nube, es necesario invertir en la administración financiera de la nube y en la optimización de costos. Su organización debe dedicar tiempo y recursos para desenvolverse bien en este nuevo ámbito de la tecnología y de administración del uso. De forma similar a su capacidad de seguridad o de excelencia operativa, debe desarrollar capacidades a través de la creación de conocimientos, programas, recursos y procesos que lo ayuden a convertirse en una organización rentable.
- Adopción de un modelo de consumo: pague solo por los recursos de computación que necesite y aumente o disminuya su uso en función de las necesidades de la empresa, sin recurrir a elaboradas previsiones. Por ejemplo, los entornos de desarrollo y pruebas se utilizan normalmente

solo ocho horas al día durante la semana laboral. Puede interrumpir estos recursos cuando no se utilicen y obtener así un posible ahorro de costos del 75 % (40 horas frente a 168 horas).

- Evaluación de la eficacia global: mida el rendimiento empresarial de la carga de trabajo y los costos asociados a su entrega. Use esta medición para conocer las ganancias que obtiene al aumentar la producción y reducir los costos.
- Fin del gasto de dinero en tareas pesadas poco diferenciadas: AWS se encarga del trabajo pesado de las operaciones del centro de datos, como el montaje de los servidores en bastidores, su apilamiento y su alimentación. También elimina la carga operativa de administrar sistemas operativos y aplicaciones con servicios administrados. De este modo, podrá centrarse en sus clientes y proyectos empresariales en lugar de hacerlo en la infraestructura de TI.
- Análisis y atribución de gastos: la nube facilita la identificación precisa del uso y el costo de los sistemas, lo que permite atribuir de forma transparente los costos de TI a los propietarios de cargas de trabajo individuales. Esto lo ayuda a medir el retorno de la inversión (ROI) y da a los propietarios de cargas de trabajo la oportunidad de optimizar sus recursos y reducir costos.

## Definición

Existen cinco áreas de prácticas recomendadas para la optimización de costos en la nube:

- Práctica de administración financiera en la nube
- Conocimiento del gasto y del uso
- Recursos rentables
- Administración de la demanda y suministro de recursos
- Optimización a lo largo del tiempo

Al igual que con los otros pilares del Marco de Well-Architected, hay compensaciones que se deben tomar en cuenta, por ejemplo, si se debe optimizar el tiempo de comercialización o el costo. En algunos casos, es más eficiente optimizar la velocidad (salida rápida al mercado, envío de nuevas características o cumplimiento de una fecha límite), en lugar de invertir en la optimización de costos iniciales. Las decisiones de diseño a veces se guían por la prisa en lugar de basarse en los datos y siempre existe la tentación de sobrecompensar “por si acaso” en lugar de dedicar tiempo a hacer un análisis comparativo para definir cuál es la implementación más rentable. Esto puede dar como resultado implementaciones poco optimizadas y con un aprovisionamiento excesivo. Sin embargo, es una opción razonable cuando hay que migrar recursos mediante lift-and-shift desde su entorno en las instalaciones hacia la nube y optimizarlos más adelante. Invertir el esfuerzo adecuado en una

estrategia de optimización de costos por adelantado le permite obtener los beneficios económicos de la nube con mayor facilidad, al lograr una adhesión constante a las prácticas recomendadas y evitar un aprovisionamiento excesivo innecesario. En las siguientes secciones se proporcionan técnicas y prácticas recomendadas para la implementación inicial y continua de la administración financiera de la nube y la optimización de costos para sus cargas de trabajo.

## Prácticas recomendadas

Temas


- Práctica de administración financiera en la nube
- Conocimiento del gasto y del uso
- Recursos rentables
- Administración de la demanda y suministro de recursos
- Optimización a lo largo del tiempo


### Práctica de administración financiera en la nube

Con la adopción de la nube, los equipos de tecnología innovan más rápido porque los ciclos de aprobación, aprovisionamiento e implementación de la infraestructura son más cortos. Se necesita un nuevo enfoque de la administración financiera en la nube para obtener valor empresarial y éxito financiero. Este enfoque es la administración financiera en la nube que permite desarrollar capacidades en toda la organización implementando su amplio conocimiento organizativo a la hora de diseñar programas, recursos y procesos.

Muchas empresas constan de distintas unidades con distintas prioridades. La capacidad para que la organización siga una serie acordada de objetivos financieros y que disponga de los mecanismos necesarios para cumplirlos hará que sea mucho más eficiente. Una organización capaz innovará y diseñará más rápidamente, será más ágil y se adaptará a factores internos o externos.

En AWS puede usar el Explorador de costos y, opcionalmente, Amazon Athena y Amazon QuickSight con el Informe de uso y costos (CUR) para que toda la organización sea consciente del uso y los costos. AWS Budgets envía notificaciones proactivas sobre el uso y los costos. Los blogs de AWS aportan información sobre nuevos servicios y funciones para verificar que está al día de las nuevas versiones de los servicios.

La siguiente pregunta se centra en estas consideraciones sobre la optimización de costos. (Para ver una lista de preguntas y prácticas recomendadas sobre la optimización de costos, consulte el Apéndice).

COST 1: ¿Cómo implementa la administración financiera en la nube?

Implementar la administración financiera en la nube ayuda a las empresas a obtener valor empresarial y éxito financiero al optimizar su costo y uso, y al escalar en AWS.

A la hora de diseñar una función de optimización de costos, trabaje con los miembros e implique a expertos en CFM y optimización de costos en los equipos. De este modo, los miembros del equipo comprenderán cómo funciona la empresa actualmente y cómo implementar mejoras rápidamente. Piense también en incluir a personas con habilidades adicionales o específicas, como analistas y gestores de proyectos.

Al crear conciencia de los costos en toda la organización, piense en mejorar o aprovechar los programas y procesos existentes. Es más rápido agregar a algo que ya existe que diseñar procesos y programas nuevos. De este modo verá resultados mucho rápidamente.

## Conocimiento del gasto y del uso

La mayor flexibilidad y agilidad que proporciona la nube fomenta la innovación, además de acelerar el desarrollo y la implementación. Disminuye los procesos manuales y el tiempo asociados al aprovisionamiento de la infraestructura en las instalaciones, incluida la identificación de especificaciones de hardware, la negociación de presupuestos de precios, la administración de órdenes de compra, la programación de envíos y la posterior implementación de los recursos. Sin embargo, la facilidad de uso y la capacidad bajo demanda prácticamente ilimitada requiere una nueva forma de pensar sobre los gastos.

Muchos negocios constan de varios sistemas ejecutados por varios equipos. La capacidad de atribuir los costos de los recursos a los propietarios de organizaciones individuales o productos impulsa el comportamiento de uso eficiente y ayuda a reducir el desperdicio. La atribución de costos precisa le permite saber qué productos son realmente rentables para así tomar decisiones más informadas sobre dónde asignar el presupuesto.

En AWS, puede crear una estructura de cuentas con AWS Organizations o AWS Control Tower, que le permite separar y lo ayuda a la hora de asignar los costos y el uso. También puede etiquetar los recursos para aplicar información de la organización y empresarial al uso y los costos. Use

AWS Cost Explorer para tener mayor visibilidad de los costos y el uso, o cree análisis y paneles personalizados con Amazon Athena y Amazon QuickSight. Los costos y el uso se controlan mediante notificaciones de AWS, y para los controles se usa AWS Identity and Access Management (IAM) y Service Quotas.

Las siguientes preguntas se centran en estas consideraciones sobre la optimización de costos.

### COST 2: ¿Cómo controla el uso?

Establezca políticas y mecanismos para validar que se incurre en costos apropiados mientras se alcanzan los objetivos. Al emplear un enfoque basado en evaluar la situación, puede innovar sin gastar de más.

### COST 3: ¿Cómo supervisa el uso y el costo?

Establezca políticas y procedimientos para supervisar y asignar adecuadamente sus costos. Esto le permite medir y mejorar la rentabilidad de esta carga de trabajo.

### COST 4: ¿Cómo retira los recursos?

Implemente el control de cambios y la administración de recursos desde el inicio del proyecto hasta su finalización. Esto facilita el cierre de los recursos no utilizados con el fin de reducir el desperdicio.

Puede usar etiquetas de asignación de costos para clasificar y hacer un seguimiento del uso y los costos de AWS. Cuando aplica etiquetas a sus recursos de AWS (como instancias de EC2 o buckets de S3), AWS genera un informe de uso y costos con su uso y sus etiquetas. Puede aplicar etiquetas que representen categorías de la organización (como centros de costos, nombres de cargas de trabajo o propietarios) para organizar sus costos en varios servicios.

Asegúrese de usar el nivel de detalle y especificación adecuados al supervisar y crear informes de los costos y el uso. En el caso de la información de alto nivel y las tendencias, use la información detallada diaria con el explorador de costos de AWS Cost Explorer. Si quiere una inspección y análisis en profundidad, use la información detallada por hora de AWS Cost Explorer o Amazon Athena y Amazon Quick con el informe de uso y costos (CUR) con información detallada por hora.

La combinación de recursos etiquetados con el seguimiento del ciclo de vida de las entidades (empleados, proyectos) hace posible identificar recursos o proyectos huérfanos que ya no generan valor para la organización y deberían retirarse. Puede establecer alertas de facturación para recibir notificaciones cuando se supere el gasto previsto.

## Recursos rentables

El uso de las instancias y los recursos adecuados para su carga de trabajo es clave para ahorrar costos. Por ejemplo, un proceso de informes puede tardar cinco horas en ejecutarse en un servidor pequeño, pero tardará una hora en un servidor más grande que es el doble de caro. Ambos servidores proporcionan el mismo resultado, pero el servidor más pequeño acarrea más costo a lo largo del tiempo.

Una carga de trabajo con una arquitectura adecuada usa los recursos más rentables, lo que puede suponer un impacto económico positivo notable. También tiene la oportunidad de usar servicios administrados para reducir los costos. Por ejemplo, en lugar de mantener servidores para entregar correos electrónicos, puede usar un servicio que cobre por mensaje.

AWS ofrece una variedad de opciones a precios rentables y flexibles para adquirir instancias de Amazon EC2 y otros servicios de la manera que mejor se adapte a sus necesidades. Las instancias bajo demanda permiten pagar la capacidad de computación por hora, sin que exista una tarifa mínima necesaria. Los Savings Plans e instancias reservadas ofrecen un ahorro de hasta el 75 % en los precios bajo demanda. Con las instancias de spot, puede aprovechar la capacidad sin utilizar de Amazon EC2 y ofrecer un ahorro de hasta el 90 % en los precios bajo demanda. Las instancias de spot son adecuadas cuando el sistema tolera el uso de una flota de servidores en la que los servidores individuales pueden ir y venir de forma dinámica, como los servidores web sin estado, el procesamiento por lotes o cuando se utilizan HPC y macrodatos.

Seleccionar el servicio apropiado también puede reducir el uso y los costos, como CloudFront para minimizar la transferencia de datos o reducir los costos, como el uso de Amazon Aurora en Amazon RDS para eliminar los caros costos de licencias de bases de datos.

Las siguientes preguntas se centran en estas consideraciones sobre la optimización de costos.

### COST 5: ¿Cómo evalúa el costo cuando selecciona servicios?

Amazon EC2, Amazon EBS y Amazon S3 son servicios de AWS básicos. Los servicios administr ados, como Amazon RDS y Amazon DynamoDB, son servicios de AWS de nivel superior o de aplicación. Al seleccionar los bloques de creación y los servicios administrados apropiados, puede

COST 5: ¿Cómo evalúa el costo cuando selecciona servicios?

optimizar esta carga de trabajo para el costo. Por ejemplo, al usar servicios administrados, puede reducir o eliminar gran parte de sus gastos administrativos y operativos, lo que le permite trabajar en aplicaciones y actividades relacionadas con el negocio.

COST 6: ¿Cómo cumple los objetivos de costos cuando selecciona el tipo, el tamaño y el número de recursos?

Compruebe que elija el tamaño y el número de recursos apropiados para la tarea en cuestión. Al seleccionar el tipo, el tamaño y el número más rentables, minimiza el desperdicio.

COST 7: ¿Cómo utiliza los modelos de fijación de precios para reducir los costos?

Use el modelo de fijación de precios más apropiado para sus recursos a fin de minimizar los gastos.

COST 8: ¿Cómo planifica los gastos de transferencia de datos?

Compruebe que planifique y supervise los cargos de transferencia de datos para que pueda tomar decisiones en cuanto al diseño y minimizar los costos. Un cambio de diseño pequeño, pero efectivo, puede reducir drásticamente sus costos operativos con el tiempo.

Al tener en cuenta el costo durante la selección del servicio y usar herramientas como el Explorador de costos y AWS Trusted Advisor para revisar regularmente el uso de AWS, puede supervisar activamente su uso y ajustar sus implementaciones de acuerdo con ello.

## Administración de la demanda y suministro de recursos

Al migrar a la nube, paga solo por lo que necesita. Puede proporcionar recursos para que se adapten a la demanda de carga de trabajo en el momento que se requieran, disminuyendo así la necesidad de sobreaprovisionamiento, que es algo caro e inútil. También puede modificar la demanda con un límite, un búfer o una cola para suavizar la demanda y usar menos recursos, lo que bajará el costo. O bien puede procesarla más tarde con un servicio por lotes.

En AWS, puede aprovisionar los recursos automáticamente para que coincidan con la demanda de la carga de trabajo. Auto Scaling y los enfoques basados en la demanda y el tiempo le permiten agregar y quitar recursos según sea necesario. Si puede anticipar los cambios en la demanda, puede ahorrar más dinero y asegurarse de que sus recursos coincidan con las necesidades de su carga de trabajo. Puede usar Amazon API Gateway para implementar una limitación o Amazon SQS para implementar una cola en la carga de trabajo. Ambas herramientas le permitirán modificar la demanda en los componentes de la carga de trabajo.

La siguiente pregunta se centra en estas consideraciones sobre la optimización de costos.

COST 9: ¿Cómo administra la demanda y aprovisiona los recursos?

Para que la carga de trabajo tenga un gasto y un rendimiento equilibrados, compruebe que se utiliza todo aquello en lo que invierte y evite desaprovechar significativamente las instancias. Una métrica de uso sesgada en cualquier dirección tiene un efecto adverso en su organización, ya sea en los costos operativos (rendimiento degradado debido al sobreúso) o en los gastos de AWS desperdiciados (debido al sobreaprovisionamiento).

Al pensar en modificar la demanda y aprovisionar recursos, tenga muy en cuenta los patrones de uso, el tiempo que se tarda en aprovisionar nuevos recursos y en la predictibilidad del patrón de demanda. Al administrar la demanda, compruebe que dispone de un búfer o una cola de tamaño adecuado y que responde a la demanda de carga de trabajo en el plazo requerido.

## Optimización a lo largo del tiempo

A medida que AWS presenta nuevos servicios y características, se recomienda que revise sus decisiones de diseño actuales para comprobar que sigan siendo las más rentables. A medida que cambian sus requisitos, retire de forma agresiva recursos, servicios enteros y sistemas que ya no necesite.

Implementar nuevas funciones o tipos de recurso puede optimizar la carga de trabajo poco a poco, a la vez que minimiza el esfuerzo requerido para implementar el cambio. Esto proporciona mejoras continuas en la eficiencia a lo largo de tiempo y le permite disponer de la tecnología más avanzada para reducir los costos operativos. También puede reemplazar o agregar componentes nuevos a la carga de trabajo con servicios nuevos. Esto puede mejorar significativamente la eficiencia, por lo que es esencial que revise regularmente su carga de trabajo e implemente nuevos servicios y funciones.

Las siguientes preguntas se centran en estas consideraciones sobre la optimización de costos.

COST 10: ¿Cómo evalúa los servicios nuevos?

A medida que AWS presenta nuevos servicios y características, se recomienda que revise sus decisiones de diseño actuales para comprobar que sigan siendo las más rentables.

Cuando revise regularmente sus implementaciones, evalúe cómo le pueden ayudar los nuevos servicios a ahorrar dinero. Por ejemplo, Amazon Aurora en Amazon RDS puede reducir los costos de las bases de datos relacionales. Usar un servicio sin servidor como Lambda puede eliminar la necesidad de operar y administrar instancias para ejecutar código.

COST 11: ¿Cómo evalúa el costo del esfuerzo?

Evalúe el costo del esfuerzo de las operaciones en la nube, revise las operaciones en la nube que requieran más tiempo y automatícelas para reducir el esfuerzo humano y los costos mediante la adopción de servicios AWS relacionados, productos de terceros o herramientas personalizadas.

## Recursos

Consulte los siguientes recursos para obtener más información sobre nuestras prácticas recomendadas sobre la optimización de costos.

### Documentación

- AWS documentación

### Documento técnico

- Pilar de optimización de costos

# Sostenibilidad

El pilar de la sostenibilidad se centra en los impactos medioambientales, sobre todo en la eficiencia y el consumo energéticos, ya que son impulsores importantes que ayudan a los arquitectos a promover la adopción de medidas directas destinadas a reducir el uso de los recursos. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de sostenibilidad.

Temas


- Principios de diseño
- Definición
- Prácticas recomendadas
- Recursos


## Principios de diseño

Existen seis principios de diseño para la sostenibilidad en la nube:

- Comprensión del impacto: mida la repercusión de su carga de trabajo en la nube y modele el impacto futuro de dicha carga. Incluya todas las fuentes de impacto, incluidas las repercusiones resultantes del uso de sus productos por parte de los clientes y de su eventual cierre y retirada. Revise los recursos y las emisiones que se requieren por unidad de trabajo para comparar el rendimiento productivo con el impacto total de sus cargas de trabajo en la nube. Use estos datos para establecer indicadores clave de rendimiento (KPI), evaluar formas de mejorar la productividad a la vez que se reduce el impacto y calcular la repercusión de los cambios propuestos a lo largo del tiempo.
- Establecimiento de objetivos de sostenibilidad: para cada carga de trabajo en la nube, establezca objetivos de sostenibilidad a largo plazo, como la reducción de los recursos de computación y almacenamiento requeridos por transacción. Modele el rendimiento de la inversión de las mejoras de sostenibilidad para las cargas de trabajo existentes y proporcione a los propietarios los recursos que necesitan para invertir en objetivos de sostenibilidad. Planifique el crecimiento y diseñe la arquitectura de las cargas de trabajo para que ese crecimiento se traduzca en una reducción de la intensidad del impacto y elija una unidad de medida adecuada, por ejemplo, por usuario o por transacción. Los objetivos en general ayudan a respaldar el cumplimiento de los objetivos de sostenibilidad más amplios de su organización o negocio, a identificar las regresiones y a priorizar las áreas susceptibles de mejora.
- Maximización del uso: aplique el tamaño adecuado a las cargas de trabajo e implemente un diseño eficaz para garantizar un alto uso y maximizar la eficiencia energética del hardware subyacente. Dos hosts que se ejecutan al 30 % son menos eficientes que uno solo que se ejecute al 60 %, debido al consumo energético base por host. Al mismo tiempo, reduzca o minimice los recursos inactivos, el procesamiento y el almacenamiento para reducir la energía total necesaria para ejecutar su carga de trabajo.

- Anticipación y adopción de ofertas de hardware y software nuevas y más eficientes: respalde las mejoras iniciales que sus socios y proveedores hagan para ayudarle a reducir el impacto de sus cargas de trabajo en la nube. Supervise y evalúe de forma continua las nuevas ofertas de hardware y software más eficaces. Aporte flexibilidad al diseño para permitir una adopción rápida de tecnologías nuevas y eficaces.
- Uso de servicios administrados: el uso compartido de servicios en una amplia base de clientes ayuda a maximizar la utilización de los recursos, lo cual reduce el volumen de infraestructura necesario para admitir las cargas de trabajo en la nube. Por ejemplo, los clientes pueden compartir el impacto de los componentes de un centro de datos común, como la potencia y las redes, mediante la migración de las cargas de trabajo a la Nube de AWS y la adopción de servicios administrados, como AWS Fargate, para contenedores sin servidor, en los que AWS opera a escala y es responsable de su funcionamiento eficiente. Use servicios administrados que le ayuden a minimizar su impacto, como pasar datos a los que no se accede con frecuencia a almacenamiento en frío de forma automática con configuraciones del ciclo de vida de Amazon S3 o Amazon EC2 Auto Scaling, a fin de ajustar la capacidad para satisfacer la demanda.
- Reducción del impacto descendente de las cargas de trabajo en la nube: reduzca la cantidad de energía o recursos necesarios para utilizar sus servicios. Reduzca la necesidad de que los clientes tengan que actualizar sus dispositivos para usar sus servicios. Pruebe a usar granjas de dispositivos para comprender el impacto esperado y haga pruebas con los clientes para que entiendan el impacto real del uso de sus servicios.

## Definición

Existen seis áreas de prácticas recomendadas para la sostenibilidad en la nube:

- Selección de región
- Alineación con la demanda
- Software y arquitectura
- Datos
- Hardware y servicios
- Proceso y cultura

La sostenibilidad en la nube es un esfuerzo casi continuo que se centra principalmente en mejorar la eficiencia y reducir el consumo de energía en todos los componentes de una carga de trabajo, obteniendo así el máximo beneficio de los recursos aprovisionados y minimizando los recursos

totales necesarios. Este esfuerzo puede abarcar desde la selección inicial de un lenguaje de programación eficiente hasta la adopción de algoritmos modernos, el uso de técnicas eficientes de almacenamiento de datos, la implementación de una infraestructura de computación eficiente y del tamaño correcto, y la minimización de los requisitos de hardware de alta potencia para el usuario final.

## Prácticas recomendadas

Temas


- Selección de región
- Alineación con la demanda
- Software y arquitectura
- Administración de datos
- Hardware y servicios
- Proceso y cultura


### Selección de región

La elección de la región para su carga de trabajo afecta significativamente a sus KPI, incluidos el rendimiento, el costo y la huella de carbono. Para mejorar estos KPI, debe elegir las regiones para las cargas de trabajo en función tanto de los requisitos empresariales como de los objetivos de sostenibilidad.

La siguiente pregunta se centra en estas consideraciones de sostenibilidad. (Para ver una lista de preguntas y prácticas recomendadas sobre sostenibilidad, consulte el Apéndice).

SUS 1: ¿Cómo se seleccionan las regiones para la carga de trabajo?

La elección de la región para su carga de trabajo afecta significativamente a sus KPI, incluidos el rendimiento, el costo y la huella de carbono. Para mejorar estos KPI, debe elegir las regiones para las cargas de trabajo en función tanto de los requisitos empresariales como de los objetivos de sostenibilidad.

## Alineación con la demanda

La forma en que los usuarios y las aplicaciones consumen las cargas de trabajo y otros recursos puede ayudarle a identificar las mejoras necesarias para alcanzar sus objetivos de sostenibilidad. Escale la infraestructura para adaptarla continuamente a la demanda y compruebe que solo utiliza los recursos mínimos necesarios para prestar asistencia a sus usuarios. Alinee los niveles de servicio con las necesidades de los clientes. Posicione los recursos de forma que se limite el uso de red necesario para que los usuarios puedan consumirlos. Elimine los activos que no se usan. Proporcione a los miembros de su equipo dispositivos que satisfagan sus necesidades con un impacto mínimo en la sostenibilidad.

La siguiente pregunta se centra en esta consideración de la sostenibilidad:

**SUS 2: ¿Cómo alinea los recursos en la nube a su demanda?**

La forma en que los usuarios y las aplicaciones consumen las cargas de trabajo y otros recursos puede ayudarle a identificar las mejoras necesarias para alcanzar sus objetivos de sostenibilidad. Escale la infraestructura para adaptarla continuamente a la demanda y compruebe que solo utiliza los recursos mínimos necesarios para prestar asistencia a sus usuarios. Alinee los niveles de servicio con las necesidades de los clientes. Posicione los recursos de forma que se limite el uso de red necesario para que los usuarios puedan consumirlos. Elimine los activos que no se usan. Proporcione a los miembros de su equipo dispositivos que satisfagan sus necesidades con un impacto mínimo en la sostenibilidad.

Escalado de la infraestructura con la carga del usuario: identifique los periodos de poco o ningún uso y escale los recursos en consonancia para reducir el exceso de capacidad y mejorar la eficiencia.

Alineación de los SLA con los objetivos de sostenibilidad: defina y actualice los acuerdos de nivel de servicio (SLA), por ejemplo, para los periodos de retención de datos o la disponibilidad, a fin de minimizar el número de recursos necesarios para admitir la carga de trabajo sin, por ello, dejar de satisfacer los requisitos empresariales.

Disminución de la creación y el mantenimiento de los recursos no utilizados: analice los recursos de aplicaciones (como los informes precompilados, los conjuntos de datos y las imágenes estáticas) y los patrones de acceso a los recursos para identificar cualquier tipo de redundancia, infrautilización y los posibles objetivos de retirada. Consolide los recursos generados con contenido redundante (por ejemplo, informes mensuales con conjuntos de datos o resultados superpuestos o comunes)

para reducir los recursos consumidos cuando se duplican las salidas. Retire los recursos que no se utilicen (por ejemplo, imágenes de productos que ya no se venden) para liberar recursos consumidos y reducir el número de recursos que se usan para admitir la carga de trabajo.

Optimización de la ubicación geográfica de las cargas de trabajo para las ubicaciones de los usuarios: analice los patrones de acceso a la red para identificar la ubicación geográfica desde la que se conectan los clientes. Seleccione regiones y servicios que acorten la distancia que debe recorrer el tráfico de red a fin de reducir el total de recursos de red necesarios para admitir su carga de trabajo.

Optimización de los recursos de los miembros del equipo para las actividades: optimice los recursos proporcionados a los miembros del equipo para minimizar el impacto en la sostenibilidad a la vez que se cubren sus necesidades. Por ejemplo, lleve a cabo operaciones complejas (como la representación y la compilación) en escritorios en la nube compartidos con un uso intensivo, en lugar de hacerlo en sistemas de usuarios únicos de gran potencia infrautilizados.

## Software y arquitectura

Implemente patrones que permitan suavizar la carga y mantener un uso elevado consistente de los recursos implementados para minimizar los recursos consumidos. Puede haber componentes que queden inactivos debido a la falta de uso relacionada con los cambios en el comportamiento de los usuarios a lo largo del tiempo. Revise los patrones y la arquitectura para consolidar los componentes infrautilizados a fin de incrementar el uso general. Retire los componentes que ya no son necesarios. Analice el rendimiento de los componentes de su carga de trabajo y optimice aquellos que consumen la mayor cantidad de recursos. Tenga en cuenta los dispositivos que usan los clientes para acceder a sus servicios e implemente patrones para minimizar la necesidad de llevar a cabo actualizaciones de los dispositivos.

La siguiente pregunta se centra en estas consideraciones de sostenibilidad:

SUS 3: ¿Cómo puede sacar partido de los patrones de software y de arquitectura para respaldar sus objetivos de sostenibilidad?

Implemente patrones que permitan suavizar la carga y mantener un uso elevado consistente de los recursos implementados para minimizar los recursos consumidos. Puede haber componentes que queden inactivos debido a la falta de uso relacionada con los cambios en el comportamiento de los usuarios a lo largo del tiempo. Revise los patrones y la arquitectura para consolidar los componentes infrautilizados a fin de incrementar el uso general. Retire los componentes que ya

SUS 3: ¿Cómo puede sacar partido de los patrones de software y de arquitectura para respaldar sus objetivos de sostenibilidad?

no son necesarios. Analice el rendimiento de los componentes de su carga de trabajo y optimice aquellos que consumen la mayor cantidad de recursos. Tenga en cuenta los dispositivos que usan los clientes para acceder a sus servicios e implemente patrones para minimizar la necesidad de llevar a cabo actualizaciones de los dispositivos.

Optimización del software y la arquitectura para los trabajos asíncronos y programados: use arquitecturas y diseños de software eficaces para minimizar el promedio de recursos necesarios por unidad de trabajo. Implemente mecanismos que deriven en un uso equilibrado de los componentes para reducir el número de recursos inactivos entre tareas y minimizar el impacto de los picos de carga.

Eliminación o refactorización de los componentes de cargas de trabajo con uso reducido o nulo: supervise la actividad de la carga de trabajo para identificar posibles cambios en el uso de los componentes individuales a lo largo del tiempo. Elimine los componentes que ya no se usan ni se necesitan y refactorice aquellos con un uso reducido para limitar los recursos desperdiciados.

Optimización de las áreas de código que consumen más tiempo o recursos: supervise la actividad de la carga de trabajo para identificar los componentes de aplicaciones que consumen la mayor cantidad de recursos. Optimización del código que se ejecuta en estos componentes para minimizar el uso de los recursos y, a la vez, maximizar el rendimiento.

Optimización del impacto en los dispositivos y equipos de los clientes: analice los dispositivos y equipos que usan los clientes para consumir sus servicios, el ciclo de vida que se espera que tengan y el impacto económico y en la sostenibilidad que supondría reemplazar esos componentes. Implemente patrones de software y arquitecturas que reduzcan al mínimo la necesidad de que los clientes tengan que reemplazar los dispositivos y actualizar los equipos. Por ejemplo, implemente características nuevas que usen código compatible con versiones de sistemas operativos y hardware anteriores o administre el tamaño de las cargas para que no superen la capacidad de almacenamiento del dispositivo de destino.

Uso de patrones y arquitecturas de software que admitan de manera más eficaz los patrones de almacenamiento y acceso a los datos: comprenda cómo se utilizan los datos dentro de la carga de trabajo, cómo los consumen los usuarios y cómo se transfieren y almacenan. Seleccione las tecnologías adecuadas para minimizar los requisitos de almacenamiento y procesamiento de los datos.

## Administración de datos

La siguiente pregunta se centra en estas consideraciones de sostenibilidad:

SUS 4: ¿Cómo puede aprovechar los patrones y las políticas de administración de datos para admitir sus objetivos de sostenibilidad?

Implemente prácticas de administración de datos para reducir el almacenamiento aprovisio nado que se necesita para admitir la carga de trabajo y los recursos necesarios para su uso. Comprenda sus datos y use las configuraciones y tecnologías de almacenamiento que respalden con mayor eficacia al valor empresarial de los datos y la forma en que se usan. Haga que el ciclo de vida de los datos incluya un almacenamiento más eficaz con un menor rendimiento cuando disminuyan los requisitos y elimine los datos que ya no se requieran.

Implemente una política de clasificación de datos: clasifique los datos para entender su importancia con respecto a los resultados empresariales. Use esta información para determinar cuándo puede mover los datos a un almacenamiento de más bajo consumo o bien eliminarlos de forma segura.

Uso de tecnologías que admitan patrones de almacenamiento y acceso a los datos: use el almacenamiento que mejor respalde la forma en que accede a los datos y los guarda a fin de minimizar los recursos aprovisionados para admitir la carga de trabajo. Por ejemplo, los dispositivos de estado sólido (SSD) requieren mucha más energía que las unidades magnéticas y solo deben utilizarse para los casos de uso de datos activos. Use almacenamiento de tipo de archivo de bajo consumo para los datos a los que se accede con poca frecuencia.

Uso de políticas de ciclo de vida para eliminar los datos innecesarios: administre el ciclo de vida de todos sus datos e imponga plazos de eliminación de forma automática para minimizar los requisitos de almacenamiento totales de la carga de trabajo.

Minimice el aprovisionamiento excesivo con el almacenamiento en bloque: para minimizar el almacenamiento total aprovisionado, cree almacenamiento en bloque con asignaciones de tamaño adecuadas para la carga de trabajo. Use volúmenes elásticos para expandir el almacenamiento a medida que crezcan los datos sin necesidad de ajustar el tamaño de almacenamiento asociado a los recursos de computación. Revise periódicamente los volúmenes elásticos y contraiga los volúmenes con un aprovisionamiento excesivo para adaptarlos al tamaño de datos actual.

Eliminación de datos innecesarios o redundantes: duplique los datos solo cuando sea necesario para minimizar el almacenamiento total consumido. Use tecnologías de copia de seguridad que

desdupliquen los datos en el nivel de archivo y de bloque. Limite el uso de configuraciones de matriz redundante de discos independientes (RAID), excepto cuando sea necesario para cumplir los SLA.

Uso del almacenamiento de objetos o sistemas de archivos compartidos para acceder a datos comunes: adopte el almacenamiento compartido y fuentes de confianza únicas para evitar la duplicación de datos y reducir los requisitos de almacenamiento total de la carga de trabajo. Recupere datos del almacenamiento compartido solo cuando sea necesario. Desconecte los volúmenes que no se utilizan para liberar recursos. Minimice el movimiento de datos entre las redes: use el almacenamiento compartido y acceda a los datos de los almacenes regionales correspondientes para minimizar el total de recursos de redes necesarios para admitir el movimiento de los datos de la carga de trabajo.

Creación de copias de seguridad de los datos solo cuando sea difícil volver a crearlos: para minimizar el consumo de almacenamiento, haga copias de seguridad únicamente de aquellos datos que tengan valor empresarial o que sean necesarios para satisfacer los requisitos de cumplimiento. Examine las políticas de copia de seguridad y excluya el almacenamiento efímero que no proporcione valor alguno en un escenario de recuperación.

## Hardware y servicios

Haga cambios en sus prácticas de administración de hardware como forma de reducir el impacto en la sostenibilidad de las cargas de trabajo. Minimice la cantidad de hardware necesario para aprovisionar e implementar y seleccione el hardware y los servicios más eficaces para su carga de trabajo individual.

La siguiente pregunta se centra en estas consideraciones de sostenibilidad:

SUS 5: ¿Cómo selecciona y usa el hardware y los servicios en la nube de su arquitectura para lograr sus objetivos de sostenibilidad?

Haga cambios en sus prácticas de administración de hardware como forma de reducir el impacto en la sostenibilidad de las cargas de trabajo. Minimice la cantidad de hardware necesario para aprovisionar e implementar y seleccione el hardware y los servicios más eficaces para su carga de trabajo individual.

Uso de la mínima cantidad de hardware para satisfacer sus necesidades: use las capacidades de la nube para hacer cambios frecuentes en las implementaciones de la carga de trabajo. Actualice los componentes implementados a medida que cambian sus necesidades.

Uso de los tipos de instancia con menor impacto: supervise de forma continuada el lanzamiento de nuevos tipos de instancia y aproveche las mejoras de la eficiencia energética; se incluyen los tipos de instancia diseñados para admitir cargas de trabajo específicas, como el entrenamiento y la inferencia en machine learning y la transcodificación de video.

Uso de servicios administrados: los servicios administrados traspasan a AWS la responsabilidad de mantener un uso medio elevado y de optimizar la sostenibilidad del hardware implementado. Use servicios administrados para distribuir el impacto en la sostenibilidad del servicio entre todos los inquilinos del mismo, lo que reduce su contribución individual.

Optimización del uso de las GPU: las unidades de procesamiento gráfico (GPU) pueden originar un alto consumo energético y muchas de las cargas de trabajo de GPU son sumamente variables, como la representación, la transcodificación y el entrenamiento y modelado de machine learning. Ejecute las instancias de GPU solo durante el tiempo que sea necesario y retírelas mediante automatización cuando no se requieran para minimizar los recursos consumidos.

## Proceso y cultura

Haga cambios en sus prácticas de desarrollo, prueba e implementación como forma de reducir el impacto en la sostenibilidad.

La siguiente pregunta se centra en estas consideraciones de sostenibilidad:

**SUS 6: ¿Cómo respaldan sus procesos organizativos sus objetivos de sostenibilidad?**

Haga cambios en sus prácticas de desarrollo, prueba e implementación como forma de reducir el impacto en la sostenibilidad.

Adopción de métodos que permitan introducir mejoras en la sostenibilidad rápidamente: pruebe y valide las posibles mejoras de sostenibilidad antes de implementarlas en producción. Tenga en cuenta el costo de las pruebas al calcular las posibles ventajas futuras de una mejora. Desarrolle operaciones de prueba de bajo costo para impulsar la oferta de pequeñas mejoras.

Mantenga actualizada la carga de trabajo: actualizar los sistemas operativos, las bibliotecas y las aplicaciones puede mejorar la eficiencia de la carga de trabajo y permitir una adopción más sencilla de tecnologías más eficaces. Un software actualizado también puede incluir características que midan el impacto de la carga de trabajo en la sostenibilidad de forma más precisa, ya que los proveedores ofrecen características para cumplir sus propios objetivos de sostenibilidad.

Aumento del uso de los entornos de compilación: use la automatización y la infraestructura como código para incorporar los entornos de preproducción cuando sea necesario y retirarlos cuando no se utilicen. Un patrón común consiste en programar periodos de disponibilidad que coincidan con las horas de trabajo de los miembros del equipo de desarrollo. La hibernación es una herramienta útil para preservar el estado y habilitar las instancias en línea de forma rápida solo cuando sea necesario. Use tipos de instancia con capacidad de ampliación, instancias de spot, servicios elásticos de base de datos, contenedores y otras tecnologías para alinear la capacidad de desarrollo y prueba con el uso.

Uso de granjas de dispositivos administrados para pruebas: las granjas de dispositivos administrados reparten el impacto en la sostenibilidad de la fabricación de hardware y del uso de los recursos en varios inquilinos. Las granjas de dispositivos administrados ofrecen diversidad en los tipos de dispositivos para que pueda ofrecer compatibilidad con hardware anterior y menos popular y evitar el impacto en la sostenibilidad para el cliente que tienen las actualizaciones innecesarias de los dispositivos.

## Recursos

Consulte los siguientes recursos para obtener más información sobre nuestras prácticas recomendadas en materia de sostenibilidad.

### Documento técnico

- Pilar de sostenibilidad

### Video

- The Climate Pledge

# El proceso de revisión

La revisión de las arquitecturas debe hacerse de manera consistente, con un enfoque sin culpa que fomente la inmersión profunda. Debe ser un proceso rápido (de horas, no días), es decir, una conversación y no una auditoría. El propósito de revisar una arquitectura consiste en identificar cualquier problema crítico que deba abordarse o áreas que podrían mejorarse. El resultado de la revisión es un conjunto de acciones que deberían mejorar la experiencia de un cliente que utiliza la carga de trabajo.

Como se debatió en la sección “Arquitectura local”, querrá que cada miembro del equipo asuma la responsabilidad de la calidad de su arquitectura. Recomendamos que los miembros del equipo que construyen una arquitectura usen el Marco de Well-Architected para comprobar continuamente su arquitectura, en lugar de llevar a cabo una reunión de revisión formal. Un enfoque casi continuo permite a los miembros del equipo actualizar las respuestas a medida que evoluciona la arquitectura y mejorar la arquitectura a medida que ofrece funciones.

El Marco de AWS Well-Architected se corresponde con la forma en que AWS revisa los sistemas y servicios internamente. Se basa en un conjunto de principios de diseño que influyen en el enfoque arquitectónico y en cuestiones que garantizan que las personas no descuiden las áreas que suelen aparecer en el análisis de causa raíz (RCA). Siempre que haya un problema importante con un sistema interno, un servicio de AWS o un cliente, observamos el RCA para comprobar si podemos mejorar los procesos de revisión que utilizamos.

Las revisiones deben aplicarse en los hitos clave del ciclo de vida del producto, al principio de la fase de diseño para evitar caminos sin retorno que sean difíciles de cambiar y, luego, antes de la fecha de puesta en funcionamiento. (Muchas decisiones son caminos reversibles de doble sentido. Para tomar esas decisiones se puede recurrir a un proceso ligero. Los caminos sin retorno son difíciles o imposibles de invertir y requieren una inspección más exhaustiva antes de crearlos). Tras entrar en producción, su carga de trabajo continuará evolucionando a medida que agregue nuevas funciones y cambie las implementaciones de tecnología. La arquitectura de una carga de trabajo cambia con el tiempo. Debe seguir unas buenas prácticas de higiene para evitar que las características arquitectónicas se degraden a medida que evoluciona. Si hace cambios significativos en la arquitectura, debe seguir un conjunto de procesos de higiene, incluida una revisión Well-Architected.

Si desea utilizar la revisión como una instantánea única o una medición independiente, le recomendamos que se asegure de incluir a todas las personas relevantes en la reunión. A menudo, descubrimos que las revisiones son la primera vez que un equipo comprende realmente lo que ha

implementado. Un enfoque que funciona bien cuando se revisa la carga de trabajo de otro equipo consiste en organizar una serie de reuniones informales sobre su arquitectura, donde pueda obtener respuestas a la mayoría de las preguntas. Puede hacer un seguimiento con una o dos reuniones para aclarar o profundizar en áreas de ambigüedad o riesgo percibido.

A continuación, se presentan algunos elementos sugeridos para facilitar sus reuniones:

- Una sala de reuniones con pizarras blancas
- Impresión de diagramas o notas de diseño
- Lista de preguntas para la toma de medidas que requieren una investigación fuera de banda para responderlas (por ejemplo, “¿hemos activado el cifrado o no?”)

Tras la revisión, debe disponer de una lista de problemas que puede priorizar según el contexto de su negocio. También querrá tener en cuenta el impacto de dichos problemas en el trabajo diario de su equipo. Si aborda estos problemas con antelación, podría dedicarle más tiempo a trabajar en la creación de valor empresarial en lugar de resolver problemas recurrentes. A medida que aborde los problemas, puede actualizar su revisión para ver cómo mejora la arquitectura.

Dado que el valor de una revisión queda claro tras completarla, es posible que, al principio, los equipos nuevos sean reticentes. Aquí hay algunas objeciones que se pueden tratar mediante la formación del equipo en los beneficios de las revisiones:

- “¡Estamos demasiado ocupados!”. (A menudo se dice cuando el equipo se está preparando para un lanzamiento importante).
  - Si se está preparando para un gran lanzamiento, deseará que vaya perfectamente. La revisión le permitirá detectar cualquier problema que pueda haber pasado por alto.
  - Recomendamos que haga revisiones al principio del ciclo de vida del producto para descubrir riesgos y desarrollar un plan de mitigación conforme a la hoja de ruta de entrega de características.
- “¡No tenemos tiempo para hacer nada con los resultados!”. (A menudo se dice que cuando hay un evento inamovible, como la Super Bowl, en el que participar).
  - Estos eventos no se pueden mover. ¿Realmente quiere participar sin conocer los riesgos para su arquitectura? Aunque no aborde todos estos problemas, todavía puede tener manuales de estrategias para afrontarlos si se materializan.
- “No queremos que otras personas conozcan los secretos de la implementación de nuestra solución”.

- Si señala al equipo las preguntas del Marco de Well-Architected, verá que ninguna de las preguntas revela información comercial ni técnica.

A medida que lleva a cabo múltiples revisiones con los equipos de su organización, puede identificar problemas temáticos. Por ejemplo, es posible que descubra que un grupo de equipos tiene problemas en un pilar o tema en particular. Querrá ver todas sus revisiones de manera integral e identificar cualquier mecanismo, formación o charla de ingeniería que puedan ayudar a abordar esas cuestiones temáticas.

# Conclusión

El Marco de AWS Well-Architected proporciona prácticas recomendadas sobre arquitectura en los seis pilares para diseñar y utilizar sistemas en la nube fiables, seguros, eficaces, rentables y sostenibles. El marco proporciona un conjunto de preguntas que le permiten revisar una arquitectura existente o propuesta. También proporciona un conjunto de prácticas recomendadas de AWS para cada pilar. El uso del marco en su arquitectura le ayudará a producir sistemas estables y eficaces, lo que le permite centrarse en sus requisitos funcionales.

# Colaboradores

Las siguientes personas y organizaciones han colaborado en este documento:

- Brian Carlson, Operations Lead Well-Architected, Amazon Web Services
- Ben Potter, Security Lead Well-Architected, Amazon Web Services
- Seth Eliot, Reliability Lead Well-Architected, Amazon Web Services
- Eric Pullen, Sr. Solutions Architect, Amazon Web Services
- Rodney Lester, Principal Solutions Architect, Amazon Web Services
- Jon Steele, Sr. Technical Account Manager, Amazon Web Services
- Max Ramsay, Principal Security Solutions Architect, Amazon Web Services
- Callum Hughes, Solutions Architect, Amazon Web Services
- Ben Mergen, Senior Cost Lead Solutions Architect, Amazon Web Services
- Chris Kozlowski, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Alex Livingstone, Principal Specialist Solutions Architect, Cloud Operations, Amazon Web Services
- Paul Moran, Principal Technologist, Enterprise Support, Amazon Web Services
- Peter Mullen, Advisory Consultant, Professional Services, Amazon Web Services
- Chris Pates, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Arvind Raghunathan, Principal Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Sam Mokhtari, Senior Efficiency Lead Solutions Architect, Amazon Web Services

# Documentación adicional

Centro de arquitectura de AWS

AWS Conformidad en la nube

Programa para socios de Well-Architected de AWS

AWS Well-Architected Tool

Página de inicio de AWS Well-Architected

Documento técnico sobre el pilar de excelencia operativa

Documento técnico sobre el pilar de seguridad

Documento técnico sobre el pilar de fiabilidad

Documento técnico sobre el pilar de eficiencia del rendimiento

Documento técnico sobre el pilar de optimización de costos

Documento técnico sobre el pilar de sostenibilidad

Amazon Builders' Library

# Revisiones del documento

Para recibir notificaciones sobre las actualizaciones de este documento técnico, suscríbase a la fuente RSS.

| Cambio | Descripción | Fecha |
| --- | --- | --- |
| Actualización importante | Se han actualizado las prácticas recomendadas con nuevas orientaciones en fiabilidad, seguridad, excelencia operativa, sostenibilidad y eficiencia del rendimiento. El pilar de la fiabilidad se ha renovado y actualizado a gran escala en muchas de las prácticas recomendadas. Las directrices sobre la excelencia operativa y de seguridad se han actualizado y perfeccionado con nuevos servicios y sugerencias de la IA generativa. La sostenibilidad recibió varias actualizaciones basadas en los servicios de AWS y en una nueva práctica recomendada. | 6 de noviembre de 2024 |
| Actualización importante | Se aplicaron actualizaciones a gran escala de las prácticas recomendadas en todo los pilares. Tanto la seguridad como los costos cuentan con nuevas prácticas recomendadas. | 27 de junio de 2024 |

| | | |
|---|---|---|
| Actualización importante | Se aplicaron actualizaciones importantes de los pilares. | 3 de octubre de 2023 |
| Actualización importante | Se actualizaron las prácticas recomendadas con una guía prescriptiva y se agregaron nuevas prácticas recomenda das. Se agregaron nuevas preguntas a los pilares de seguridad y optimización de costos. | 10 de abril de 2023 |
| Actualización menor | Se agregó la definición de nivel de esfuerzo y se actualizaron las prácticas recomendadas del apéndice. | 20 de octubre de 2022 |
| Actualización importante | Se agregó el pilar de sostenibi lidad y se actualizaron los enlaces. | 2 de diciembre de 2021 |
| Actualización importante | Se agregó el pilar de sostenibi lidad al marco. | 20 de noviembre de 2021 |
| Actualización menor | Se eliminó el lenguaje no inclusivo. | 22 de abril de 2021 |
| Actualización menor | Se corrigieron numerosos enlaces. | 10 de marzo de 2021 |
| Actualización menor | Se aplicaron cambios editorial es mínimos en todo el documento. | 15 de julio de 2020 |
| Actualización importante | Se revisaron y reescribieron casi todas las preguntas y respuestas. | 8 de julio de 2020 |

| | | |
|---|---|---|
| Documento técnico actualizado | Se agregaron la AWS Well-Architected Tool, enlaces a AWS Well-Architected Labs y socios de Well-Architected de AWS, y correcciones menores para permitir la versión del marco en múltiples idiomas. | 1 de julio de 2019 |
| Documento técnico actualizado | Se revisaron y reescribieron la mayoría de las preguntas y respuestas para garantizar que las preguntas se centren en un único tema. Esto provocó que algunas preguntas anteriores se dividieran en preguntas múltiples. Se agregaron términos comunes a las definiciones (carga de trabajo, componente, etc.). Se modificó la presentación de la pregunta en el cuerpo principal para incluir texto descriptivo. | 1 de noviembre de 2018 |
| Documento técnico actualizado | Se aplicaron actualizaciones para simplificar el texto de las preguntas, estandarizar las respuestas y mejorar la legibilidad. | 1 de junio de 2018 |
| Documento técnico actualizado | La excelencia operativa se trasladó al frente de los pilares y se reescribió para incluir otros pilares. Se actualizaron otros pilares para reflejar la evolución de AWS. | 1 de noviembre de 2017 |

| | | |
|---|---|---|
| Documento técnico actualizado | Se actualizó el marco para incluir el pilar de excelencia operativa y se revisaron y actualizaron los otros pilares para reducir la duplicación e incorporar los aprendizajes de las revisiones a miles de clientes. | 1 de noviembre de 2016 |
| Actualizaciones menores | Se actualizó el apéndice con información actual de Registros de Amazon CloudWatch. | 1 de noviembre de 2015 |
| Publicación inicial | Se publicó el Marco de AWS Well-Architected. | 1 de octubre de 2015 |

> **Note**
>
> Para suscribirse a las actualizaciones de RSS, debe tener un complemento de RSS habilitado para el navegador que esté utilizando.

Versiones del marco:

- 27-06-2024
- 03-10-2023
- 10-04-2023
- 31-03-2022

# Apéndice: preguntas y prácticas recomendadas

En este apéndice se resumen todas las preguntas y prácticas recomendadas del Marco de AWS Well-Architected.

Pilares

- Excelencia operativa
- Seguridad
- Fiabilidad
- Eficiencia del rendimiento
- Optimización de costos
- Sostenibilidad

## Excelencia operativa

La excelencia operativa (OE) es el compromiso de crear software correctamente, al mismo tiempo que se ofrece de forma coherente una excelente experiencia de cliente. El pilar de excelencia operativa contiene las prácticas recomendadas para organizar su equipo, diseñar su carga de trabajo, ejecutarla a escala y hacerla evolucionar con el tiempo. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de excelencia operativa.

Áreas de prácticas recomendadas

- Organization
- Prepare
- Operación
- Evolución

### Organization

Preguntas

- OPS 1. ¿Cómo determina cuáles son sus prioridades?
- OPS 2. ¿Cómo estructura su organización para respaldar los resultados empresariales?
- OPS 3. ¿Cómo ayuda la cultura de su organización a lograr los resultados empresariales?

## OPS 1. ¿Cómo determina cuáles son sus prioridades?

Todos deben comprender su parte para lograr el éxito empresarial. Si se tienen objetivos compartidos, se podrán priorizar los recursos. Esto hará que los esfuerzos se traduzcan en un mayor rendimiento.

Prácticas recomendadas

- OPS01-BP01 Evaluación de las necesidades de los clientes externos
- OPS01-BP02 Evaluación de las necesidades de los clientes internos
- OPS01-BP03 Evaluación de los requisitos de gobernanza
- OPS01-BP04 Evaluación de los requisitos de cumplimiento
- OPS01-BP05 Evaluación del panorama de amenazas
- OPS01-BP06 Evaluación de las compensaciones al administrar los beneficios y los riesgos

### OPS01-BP01 Evaluación de las necesidades de los clientes externos

Involucre a las partes interesadas clave, incluidos los equipos de negocio, desarrollo y operaciones, para determinar dónde centrar los esfuerzos en función de las necesidades de los clientes externos. De este modo, se asegurará de comprender a fondo el respaldo operativo que se requiere para lograr los resultados empresariales deseados.

Resultado deseado:

- Trabaja en sentido inverso a partir de los resultados de los clientes.
- Entiende cómo sus prácticas operativas respaldan los resultados y objetivos empresariales.
- Involucra a todas las partes pertinentes.
- Dispone de mecanismos para recopilar las necesidades de los clientes externos.

Patrones comunes de uso no recomendados:

- Ha decidido no ofrecer asistencia a los clientes fuera de las horas laborables centrales, pero no ha revisado los datos históricos de solicitud de asistencia. No sabe si esto afectará a sus clientes.
- Está desarrollando una característica nueva, pero no ha involucrado a sus clientes para saber si les interesa, y si les interesa, de qué forma, y tampoco ha experimentado para validar la necesidad y la forma de la entrega.

Beneficios de establecer esta práctica recomendada: es más probable que los clientes cuyas necesidades se satisfagan sigan siendo clientes. Evaluar y comprender las necesidades de los clientes externos le permitirá saber dónde centrar sus esfuerzos para aportar valor a la empresa.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Conozca los objetivos empresariales: el éxito de una empresa se consigue por medio de metas compartidas y un entendimiento entre las partes interesadas, incluidos los equipos de negocio, desarrollo y operaciones.

Revisión de los objetivos empresariales, las necesidades y las prioridades de los clientes externos: involucre a las partes interesadas clave, incluidos los equipos de negocios, desarrollo y operaciones, para analizar los objetivos, las necesidades y las prioridades de los clientes externos. Esto garantiza que comprenda a fondo la asistencia operativa que se requiere para lograr los resultados de la empresa y de los clientes.

Establecimiento de un entendimiento compartido: establezca un entendimiento compartido de las funciones empresariales de la carga de trabajo, los roles de cada uno de los equipos que manejan la carga de trabajo y cómo estos factores facilitan sus objetivos empresariales compartidos entre los clientes internos y externos.

Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP03 Implementación de bucles de retroalimentación

OPS01-BP02 Evaluación de las necesidades de los clientes internos

Para determinar dónde centrar los esfuerzos en función de las necesidades de los clientes internos, involucre a las partes interesadas clave, incluidos los equipos de negocios, desarrollo y operaciones. Así se asegurará de que comprende exhaustivamente la asistencia operativa que se requiere para lograr resultados comerciales.

Resultado deseado:

- Utiliza sus prioridades establecidas para centrar sus esfuerzos de mejora en las que tendrán mayor repercusión (por ejemplo, el desarrollo de las competencias del equipo, la mejora del

rendimiento de la carga de trabajo, la reducción de los costos, la automatización de los manuales de procedimientos o la mejora de la supervisión).

- Actualiza sus prioridades a medida que cambian las necesidades.

Patrones comunes de uso no recomendados:

- Ha decidido cambiar las asignaciones de direcciones IP de sus equipos de producto, sin consultar con ellos, para facilitar la administración de su red. No sabe cómo afectará esto a sus equipos de producto.
- Está implementando una nueva herramienta de desarrollo, pero no ha involucrado a sus clientes internos para averiguar si es necesaria o si es compatible con sus prácticas actuales.
- Está implementando un nuevo sistema de supervisión, pero no ha contactado con sus clientes internos para averiguar si tienen necesidades de supervisión o de elaboración de informes que deban tenerse en cuenta.

Beneficios de establecer esta práctica recomendada: evaluar y comprender las necesidades internas de los clientes sirve de base para priorizar sus esfuerzos por ofrecer valor empresarial.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

- Conozca los objetivos empresariales: el éxito de una empresa se consigue por medio de metas compartidas y un entendimiento entre las partes interesadas, incluidos los equipos de negocio, desarrollo y operaciones.
- Revise los objetivos empresariales, las necesidades y las prioridades de los clientes internos: involucre a las partes interesadas clave, incluidos los equipos de negocios, desarrollo y operaciones, para analizar los objetivos, las necesidades y las prioridades de los clientes internos. Esto garantiza que comprenda a fondo la asistencia operativa que se requiere para lograr los resultados de la empresa y de los clientes.
- Establezca un entendimiento compartido: establezca un entendimiento compartido de las funciones empresariales de la carga de trabajo, los roles de cada uno de los equipos que manejan la carga de trabajo y cómo estos factores facilitan sus objetivos empresariales compartidos entre los clientes internos y externos.

Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP03 Implementación de bucles de retroalimentación

OPS01-BP03 Evaluación de los requisitos de gobernanza

La gobernanza es el conjunto de políticas, normas o marcos que utiliza una empresa para conseguir sus objetivos empresariales. Los requisitos de gobernanza se generan en su organización. Pueden afectar a los tipos de tecnologías que elija o influir en la forma de utilizar su carga de trabajo. Incorpore los requisitos de gobernanza de la organización a su carga de trabajo. El cumplimiento es la capacidad de demostrar que ha implementado los requisitos de gobernanza.

Resultado deseado:

- Los requisitos de gobernanza se incorporan al diseño arquitectónico y al funcionamiento de su carga de trabajo.
- Puede aportar pruebas de que ha seguido los requisitos de gobernanza.
- Los requisitos de gobernanza se revisan y actualizan periódicamente.

Patrones comunes de uso no recomendados:

- Su organización exige que la cuenta raíz disponga de autenticación multifactor. No ha implementado este requisito y la cuenta raíz está comprometida.
- Durante el diseño de la carga de trabajo, elegirá un tipo de instancia que no ha aprobado el departamento de TI. No puede lanzar la carga de trabajo y debe llevar a cabo un rediseño.
- Debe disponer de un plan de recuperación de desastres. No ha creado ninguno y la carga de trabajo sufre una interrupción prolongada.
- Su equipo quiere utilizar nuevas instancias, pero sus requisitos de gobernanza no se han actualizado para permitirlo.

Beneficios de establecer esta práctica recomendada:

- Seguir los requisitos de gobernanza alinea su carga de trabajo con las políticas de la organización.
- Los requisitos de gobernanza reflejan los estándares del sector y las prácticas recomendadas para su organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Colabore con las partes interesadas y las organizaciones de gobernanza para identificar los requisitos de gobernanza. Incluya los requisitos de gobernanza en su carga de trabajo. Sea capaz de demostrar que ha seguido los requisitos de gobernanza.

Ejemplo de cliente

En AnyCompany Retail, el equipo de operaciones en la nube colabora con las partes interesadas de toda la organización para desarrollar los requisitos de gobernanza. Por ejemplo, prohíben el acceso SSH a las instancias de Amazon EC2. Si los equipos necesitan acceso al sistema, deberán utilizar AWS Systems Manager Session Manager. El equipo de operaciones en la nube actualiza periódicamente los requisitos de gobernanza a medida que hay disponibles nuevos servicios.

Pasos para la implementación

1. Identifique a las partes interesadas para su carga de trabajo, incluidos los equipos centralizados.
2. Colabore con las partes interesadas para identificar los requisitos de gobernanza.
3. Una vez generada la lista, priorice los elementos de mejora y comience a implementarlos en su carga de trabajo.
   a. Utilice servicios como AWS Config para crear una gobernanza como código y validar que se cumplan los requisitos de gobernanza.
   b. Si utiliza AWS Organizations, puede aprovechar las políticas de control de servicios para implementar los requisitos de gobernanza.
4. Proporcione documentación que valide la implementación.

Nivel de esfuerzo para el plan de implementación: medio. La implementación de los requisitos de gobernanza que faltan puede dar lugar a un reajuste de su carga de trabajo.

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP04 Evaluación de los requisitos de cumplimiento: el cumplimiento es como la gobernanza, pero proviene de fuera de la organización.

Documentos relacionados:

- AWS Management and Governance Cloud Environment Guide
- Best Practices for AWS Organizations Service Control Policies in a Multi-Account Environment
- Governance in the Nube de AWS: The Right Balance Between Agility and Safety
- ¿Qué es el enfoque de gobernanza, riesgo y cumplimiento (GRC)?

Videos relacionados:

- AWS Management and Governance: Configuration, Compliance, and Audit - AWS Online Tech Talks
- AWS re:Inforce 2019: Governance for the Cloud Age (DEM12-R1)
- AWS re:Invent 2020: Achieve compliance as code using AWS Config
- AWS re:Invent 2020: Agile governance on AWS GovCloud (US)

Ejemplos relacionados:

- AWS Config Conformance Pack Samples

Servicios relacionados:

- AWS Config
- AWS Organizations - Service Control Policies

OPS01-BP04 Evaluación de los requisitos de cumplimiento

Los requisitos de cumplimiento normativo, del sector e internos son importantes a la hora de definir las prioridades de la organización. Es posible que su marco de cumplimiento le impida utilizar determinadas tecnologías o ubicaciones geográficas. Aplique la diligencia debida si no se identifican marcos de cumplimiento externos. Genere auditorías o informes que validen el cumplimiento.

Si indica que su producto se ajusta a estándares de cumplimiento específicos, debe tener un proceso interno que garantice el cumplimiento continuo. Algunos ejemplos de estándares de cumplimiento son PCI DSS, FedRAMP e HIPAA. Las estándares de cumplimiento aplicables se determinan en función de diversos factores, como los tipos de datos que la solución almacena o transmite, o las regiones geográficas compatibles con la solución.

Resultado deseado:

- Los requisitos de cumplimiento normativo, sectorial e interno se incorporan a la selección de arquitectura.
- Puede validar el cumplimiento y generar informes de auditoría.

Patrones comunes de uso no recomendados:

- Algunas partes de su carga de trabajo entran dentro del marco del estándar de seguridad de datos de la industria de tarjetas de pago (PCI DSS), pero su carga de trabajo almacena los datos de las tarjetas de crédito sin cifrar.
- Sus desarrolladores y arquitectos de software desconocen el marco de cumplimiento al que debe adherirse su organización.
- La auditoría anual de sistemas y organizaciones de control (SOC2) de tipo II tendrá lugar en breve y no puede verificar que los controles están aplicados.

Beneficios de establecer esta práctica recomendada:

- Evaluar y comprender los requisitos de cumplimiento se aplican a su carga de trabajo determinarán cómo priorizar sus esfuerzos para ofrecer valor empresarial.
- Elige las ubicaciones y las tecnologías adecuadas que sean congruentes con su marco de cumplimiento.
- Diseñar su carga de trabajo para que pueda auditar lo ayuda a demostrar que se atiene a su marco de cumplimiento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La implementación de esta práctica recomendada significa que se incorporan los requisitos de cumplimiento a su proceso de diseño de la arquitectura. Los miembros de su equipo conocen el marco de cumplimiento necesario. Valida el cumplimiento de acuerdo con el marco.

Ejemplo de cliente

AnyCompany Retail almacena la información de las tarjetas de crédito de los clientes. Los desarrolladores del equipo de almacenamiento de tarjetas saben que deben cumplir el marco PCI-DSS. Han tomado medidas para verificar que la información de las tarjetas de crédito se almacena

y se accede a ella de forma segura de acuerdo con el marco PCI-DSS. Cada año colaboran con su equipo de seguridad para validar el cumplimiento.

Pasos para la implementación

1. Colabore con sus equipos de seguridad y gobernanza para determinar qué marcos de cumplimiento sectorial, normativo o interno debe cumplir su carga de trabajo. Incorpore los marcos de cumplimiento a su carga de trabajo.
   a. Valide el cumplimiento continuo de los recursos de AWS con servicios como AWS Compute Optimizer y AWS Security Hub CSPM.
2. Informe a los miembros de su equipo sobre los requisitos de cumplimiento para que puedan utilizar y hacer evolucionar la carga de trabajo de acuerdo con ellos. Los requisitos de cumplimiento deben incluirse en las opciones de arquitectura y tecnología.
3. En función del marco de cumplimiento, puede que deba generar un informe de auditoría o de cumplimiento. Colabore con su organización para automatizar este proceso en la medida de lo posible.
   a. Utilice servicios como AWS Audit Manager para validar el cumplimiento y generar informes de auditoría.
   b. Puede descargar documentos de seguridad y cumplimiento de AWS con AWS Artifact.

Nivel de esfuerzo para el plan de implementación: medio. La implementación de marcos de cumplimiento puede suponer un desafío. La generación de informes de auditoría o documentos de cumplimiento agrega complejidad adicional.

Recursos

Prácticas recomendadas relacionadas:

- SSEC01-BP03 Identificación y validación de los objetivos de control: los objetivos de control de seguridad son una parte importante del cumplimiento general.
- SEC01-BP06 Automatización de la implementación de controles de seguridad estándares: como parte de sus canalizaciones, valide los controles de seguridad. También puede generar documentación de cumplimiento para los nuevos cambios.
- SEC07-BP02 Definición de los controles de protección de datos: muchos marcos de cumplimiento se basan en políticas de control y almacenamiento de datos.
- SEC10-BP03 Preparación de las capacidades forenses: a veces se pueden utilizar capacidades forenses para auditar el cumplimiento.

Documentos relacionados:

- Centro de cumplimiento de AWS
- Recursos de conformidad de AWS
- Documento técnico sobre riesgo y cumplimiento de AWS
- Modelo de responsabilidad compartida de AWS
- Servicios de AWS en el ámbito del programa de conformidad

Videos relacionados:

- AWS re:Invent 2020: Achieve compliance as code using AWS Compute Optimizer
- AWS re:Invent 2021 - Cloud compliance, assurance, and auditing
- AWS Summit ATL 2022 - Implementing compliance, assurance, and auditing on AWS (COP202)

Ejemplos relacionados:

- PCI DSS and AWS Foundational Security Best Practices on AWS

Servicios relacionados:

- AWS Artifact
- AWS Audit Manager
- AWS Compute Optimizer
- AWS Security Hub CSPM

OPS01-BP05 Evaluación del panorama de amenazas

Evalúe las amenazas para la empresa (por ejemplo, las empresas de la competencia, los riesgos y responsabilidades empresariales, los riesgos operativos y las amenazas para la seguridad de la información) y mantenga la información actualizada en un registro de riesgos. Incluya la repercusión de los riesgos a la hora de determinar dónde centrar los esfuerzos.

El Marco de Well-Architected hace hincapié en aprender, medir y mejorar. Proporciona un enfoque coherente para evaluar las arquitecturas e implementar diseños que se escalarán con el tiempo. AWS proporciona la AWS Well-Architected Tool para ayudarle a revisar su enfoque antes del

desarrollo, el estado de sus cargas de trabajo antes de la producción y el estado de sus cargas de trabajo durante la producción. Puede compararlos con las prácticas recomendadas de arquitectura de AWS más recientes, supervisar el estado general de sus cargas de trabajo y obtener información sobre posibles riesgos.

Los clientes de AWS son elegibles para una revisión de Well-Architected de sus cargas de trabajo de importancia crítica para medir sus arquitecturas con las prácticas recomendadas de AWS. Los clientes de Enterprise Support son elegibles para una revisión de operaciones diseñada para ayudarlos a identificar las lagunas en su enfoque para operar en la nube.

La participación de todos los equipos en estas revisiones ayuda a establecer un entendimiento común de sus cargas de trabajo y de cómo los roles del equipo contribuyen al éxito. Las necesidades identificadas a través de la revisión pueden ayudar a dar forma a sus prioridades.

AWS Trusted Advisor es una herramienta que proporciona acceso a un conjunto básico de comprobaciones que recomiendan optimizaciones para su entorno y que pueden ayudar a dar forma a sus prioridades. Los clientes de Business y Enterprise Support reciben acceso a comprobaciones adicionales centradas en la seguridad, la fiabilidad, el rendimiento y la optimización de los costos que pueden ayudar a configurar sus prioridades.

Resultado deseado:

- Revisa y toma medidas regularmente en función de los resultados de Well-Architected y Trusted Advisor.
- Conoce las revisiones que se han aplicado más recientemente a sus servicios.
- Conoce el riesgo y la repercusión de las amenazas conocidas y actúa en consecuencia.
- Implementa mitigaciones de la forma necesaria.
- Comunica acciones y el contexto.

Patrones comunes de uso no recomendados:

- Está utilizando una versión antigua de una biblioteca de software en su producto. No está al corriente de las actualizaciones de seguridad de la biblioteca por problemas que pueden tener un impacto no deseado en la carga de trabajo.
- Su competidor acaba de lanzar una versión de su producto que resuelve muchas de las quejas de sus clientes sobre su producto. No ha priorizado la resolución de ninguno de estos problemas conocidos.

- Los reguladores han estado persiguiendo a empresas como la suya que no cumplen con los requisitos legales de conformidad de la normativa. No ha priorizado el cumplimiento de ninguno de sus requisitos pendientes.

Beneficios de establecer esta práctica recomendada: identifica y comprende las amenazas a su organización y carga de trabajo, lo que lo ayuda a determinar qué amenazas debe abordar, su prioridad y los recursos necesarios para hacerlo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Evaluación del panorama de amenazas: evalúe las amenazas a la empresa (por ejemplo, competencia, riesgos y responsabilidades comerciales, riesgos operativos y amenazas a la seguridad de la información), de modo que pueda incluir su impacto al determinar dónde centrar los esfuerzos.
  - Últimos boletines de seguridad de AWS
  - AWS Trusted Advisor
- Mantenimiento de un modelo de amenazas: establezca y mantenga un modelo de amenazas que identifique las amenazas potenciales, las mitigaciones planificadas y en curso, y su prioridad. Revise la probabilidad de que las amenazas se manifiesten en forma de incidentes, el costo de recuperación de dichos incidentes y el daño causado esperado, así como el costo de prevención de los incidentes. Revise las prioridades a medida que cambie el contenido del modelo de amenazas.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP07 Identificación de amenazas y priorización de mitigaciones con un modelo de amenazas

Documentos relacionados:

- Cumplimiento de Nube de AWS
- Últimos boletines de seguridad de AWS
- AWS Trusted Advisor

Videos relacionados:

- AWS re:Inforce 2023 - A tool to help improve your threat modeling

OPS01-BP06 Evaluación de las compensaciones al administrar los beneficios y los riesgos

Los intereses contrapuestos de diversas partes pueden complicar la priorización de esfuerzos, el desarrollo de capacidades y la obtención de resultados que se ajusten a las estrategias empresariales. Por ejemplo, es posible que se le pida que acelere la comercialización de nuevas características en lugar de optimizar los costos de la infraestructura de TI. Esto puede poner en conflicto a dos partes interesadas. En estas situaciones, es necesario trasladar las decisiones a una autoridad superior para resolver el conflicto. Los datos son necesarios para eliminar el factor emocional del proceso de toma de decisiones.

Podría presentarse el mismo reto en un nivel táctico. Por ejemplo, la elección entre utilizar tecnologías de bases de datos relacionales o no relacionales puede tener una importante repercusión en el funcionamiento de una aplicación. Es fundamental conocer los resultados predecibles de las distintas opciones.

AWS puede ayudarlo a capacitar a sus equipos sobre AWS y sus servicios para aumentar su comprensión de cómo sus elecciones pueden tener un impacto en su carga de trabajo. Debe utilizar los recursos proporcionados por Soporte (Centro de conocimientos de AWS, foros de debate de AWS y Centro de Soporte) y la documentación de AWS para capacitar a sus equipos. Si tiene más preguntas, contacte con Soporte.

AWS también comparte las prácticas recomendadas y las patrones operativos en la Amazon Builders' Library. Hay una gran variedad de información útil disponible a través del Blog de AWS y The Official AWS Podcast.

Resultado deseado: cuenta con un marco de gobierno de toma de decisiones claramente definido para facilitar las decisiones importantes en todos los niveles de su organización de entrega en la nube. Este marco incluye características como un registro de riesgos, roles definidos que están autorizados a tomar decisiones y modelos definidos para cada nivel de decisión que puede tomarse. Este marco define de antemano cómo se resuelven los conflictos, qué datos deben presentarse y cómo se priorizan las opciones para que, una vez tomadas las decisiones, pueda ponerlas en marcha sin demora. El marco de toma de decisiones incluye un enfoque estandarizado para revisar y sopesar los beneficios y los riesgos de cada decisión con el fin de entender si compensa. Esto puede incluir factores externos, como el cumplimiento de los requisitos normativos.

Patrones comunes de uso no recomendados:

- Sus inversores le piden que demuestre que cumple los estándares de seguridad de datos del sector de las tarjetas de pago (PCI DSS). No ha considerado las ventajas y desventajas de satisfacer su solicitud y de continuar con sus esfuerzos actuales de desarrollo. En su lugar, sigue adelante con sus esfuerzos de desarrollo sin demostrar el cumplimiento. Sus inversores dejan de apoyar a su empresa porque les preocupa la seguridad de la plataforma y las inversiones que han hecho en ella.
- Ha decidido incluir una biblioteca que uno de los desarrolladores ha encontrado en Internet. No ha evaluado los riesgos de adoptar esta biblioteca de una fuente desconocida y no sabe si contiene vulnerabilidades o código malicioso.
- La justificación empresarial original de su migración se basaba en la modernización del 60 % de las cargas de trabajo de sus aplicaciones. Sin embargo, debido a dificultades técnicas, se tomó la decisión de modernizar solo el 20 %, lo que se tradujo en una reducción de los beneficios previstos a largo plazo, un aumento de la carga de trabajo de los equipos de infraestructura para dar asistencia manual a los sistemas heredados y una mayor necesidad de desarrollar nuevas competencias en sus equipos de infraestructura, que no tenían previsto este cambio.

Beneficios de establecer esta práctica recomendada: alinear y respaldar plenamente las prioridades empresariales a nivel de los consejos de administración, comprender los riesgos que implica lograr el éxito, tomar decisiones informadas y actuar de forma adecuada cuando los riesgos impiden las posibilidades de éxito. Comprender las implicaciones y consecuencias que tienen sus decisiones le ayudará a priorizar las opciones que tiene y a lograr que los líderes lleguen a un acuerdo más rápidamente, lo que se traducirá en mejores resultados empresariales. Identificar los beneficios disponibles de sus opciones y ser consciente de los riesgos para su organización lo ayuda a tomar decisiones basadas en datos, en lugar de en anécdotas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La administración de los beneficios y los riesgos debe definirla un órgano de gobierno que determine los requisitos para la toma de decisiones clave. Desea que las decisiones se tomen y se prioricen en función de cómo benefician a la organización, con conocimiento de los riesgos que conllevan. Tener información precisa es fundamental para tomar las decisiones de la organización. Esto debe basarse en mediciones sólidas y definirse de acuerdo con las prácticas habituales del sector en materia de análisis de costos y beneficios. Para tomar este tipo de decisiones, busque un equilibrio entre la

autoridad centralizada y la descentralizada. Siempre hay un punto medio y es importante comprender cómo afecta cada elección a las estrategias definidas y a los resultados empresariales deseados.

Pasos para la implementación

1. Formalice las prácticas de medición de beneficios dentro de un marco integral de gobernanza en la nube.
   a. Equilibre el control central de la toma de decisiones con la autoridad descentralizada para algunas decisiones.
   b. Tenga en cuenta que los arduos procesos de toma de decisiones impuestos en cada decisión pueden ralentizarle.
   c. Incorpore factores externos en su proceso de toma de decisiones (por ejemplo, requisitos de cumplimiento).
2. Establezca un marco de toma de decisiones consensuado para los distintos niveles de decisiones, que incluya quién está obligado a desbloquear las decisiones que estén sujetas a intereses conflictivos.
   a. Centralice las decisiones unidireccionales que podrían ser irreversibles.
   b. Permita que los líderes de la organización de nivel inferior tomen decisiones bidireccionales.
3. Conozca y administre los beneficios y los riesgos. Equilibre los beneficios de las decisiones con los riesgos involucrados.
   a. Identificación de beneficios: identifique los beneficios en función de los objetivos, las necesidades y las prioridades comerciales. Entre algunos ejemplos, se incluyen la repercusión del caso empresarial, el tiempo de comercialización, la seguridad, la fiabilidad, el rendimiento y el costo.
   b. Identificación de riesgos: identifique los riesgos en función de los objetivos comerciales, las necesidades y las prioridades. Los ejemplos incluyen tiempo de comercialización, seguridad, fiabilidad, rendimiento y costo.
   c. Evaluación de los beneficios frente a los riesgos y toma de decisiones fundamentadas: determine el impacto de los beneficios y los riesgos en función de los objetivos, las necesidades y las prioridades de sus partes interesadas clave, incluidos el negocio, el desarrollo y las operaciones. Evalúe el valor del beneficio frente a la probabilidad de que el riesgo se materialice y el costo de su impacto. Por ejemplo, enfatizar la velocidad de comercialización sobre la fiabilidad podría suponer una ventaja competitiva. Sin embargo, podría dar lugar a una reducción del tiempo de actividad si hay problemas de fiabilidad.

4. Imponga de forma programada decisiones clave que automaticen el cumplimiento de los requisitos de conformidad.
5. Utilice marcos y capacidades conocidos del sector, como el análisis del flujo de valor y LEAN, para establecer una base de referencia del rendimiento actual y las métricas empresariales y definir las iteraciones del progreso hacia la mejora de estas métricas.

Nivel de esfuerzo para el plan de implementación: medio-alto

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP05 Evaluación del panorama de amenazas

Documentos relacionados:

- Elementos de la cultura del día 1 de Amazon | Tome decisiones de alta velocidad y alta calidad
- Gobernanza de la nube
- Management & Governance Cloud Environment
- Governance in the Cloud and in the Digital Age: Parts One & Two

Videos relacionados:

- Podcast | Jeff Bezos | On how to make decisions

Ejemplos relacionados:

- Make informed decisions using data (The DevOps Sagas)
- Using development value stream mapping to identify constraints to DevOps outcomes

## OPS 2. ¿Cómo estructura su organización para respaldar los resultados empresariales?

Sus equipos deben comprender su papel en la consecución de los resultados empresariales. Los equipos deben comprender la función que desempeñan en el éxito de otros equipos, así como la que desempeñan los demás equipos en su propio éxito, y tener objetivos en común. Comprender la

responsabilidad, la propiedad, cómo se toman las decisiones y quién tiene autoridad para tomarlas ayudará a centrar los esfuerzos y a maximizar los beneficios de sus equipos.

Prácticas recomendadas

- OPS02-BP01 Recursos con propietarios identificados
- OPS02-BP02 Procesos y procedimientos con propietarios identificados
- OPS02-BP03 Actividades operativas con propietarios identificados responsables de su rendimiento
- OPS02-BP04 Mecanismos existentes para administrar las responsabilidades y la propiedad
- OPS02-BP05 Mecanismos para solicitar adiciones, cambios y excepciones
- OPS02-BP06 Responsabilidades predefinidas o negociadas entre equipos

OPS02-BP01 Recursos con propietarios identificados

Los recursos para su carga de trabajo deben tener propietarios identificados para el control de cambios, la resolución de problemas y otras funciones. Se asignan propietarios para las cargas de trabajo, las cuentas, la infraestructura, las plataformas y las aplicaciones. La propiedad se registra mediante herramientas como un registro central o metadatos adjuntos a los recursos. El valor empresarial de los componentes determina los procesos y los procedimientos que se les aplican.

Resultado deseado:

- Los recursos tienen propietarios identificados mediante metadatos o un registro central.
- Los miembros del equipo pueden identificar a quién pertenecen los recursos.
- Las cuentas tienen un único propietario siempre que sea posible.

Patrones comunes de uso no recomendados:

- Los contactos alternativos para sus Cuentas de AWS no están asignados.
- Los recursos carecen de etiquetas que identifiquen a qué equipos pertenecen.
- Tiene una cola de ITSM sin una asignación de correo electrónico.
- Dos equipos tienen la propiedad solapada de un elemento fundamental de la infraestructura.

Beneficios de establecer esta práctica recomendada:

- El control de cambios de los recursos resulta sencillo con una propiedad asignada.

- Puede implicar a los propietarios adecuados a la hora de solucionar problemas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Defina qué significa la propiedad para los casos de uso de los recursos en su entorno. La propiedad puede indicar quién supervisa los cambios en el recurso, quién lo respalda durante la resolución de problemas o quién es responsable desde el punto de vista financiero. Especifique y registre los propietarios de los recursos, incluido el nombre, la información de contacto, la organización y el equipo.

Ejemplo de cliente

AnyCompany Retail define la propiedad como el equipo o la persona que se encarga de los cambios y de la asistencia a los recursos. Usa AWS Organizations para administrar sus Cuentas de AWS. Los contactos alternativos de la cuenta se configuran mediante bandejas de entrada de grupo. Cada cola de ITSM se asigna a un alias de correo electrónico. Las etiquetas identifican a quién pertenecen los recursos de AWS. Para otras plataformas e infraestructuras, tiene una página wiki en la que se identifican la propiedad y la información de contacto.

Pasos para la implementación

1. Empiece con la definición de la propiedad de su organización. La propiedad puede implicar quién es el propietario del riesgo del recurso, quién es el propietario de los cambios en el recurso o quién presta asistencia al recurso cuando se solucionan problemas. La propiedad también podría implicar la propiedad financiera o administrativa del recurso.
2. Se usa AWS Organizations para administrar cuentas. Puede administrar los contactos alternativos de sus cuentas de forma centralizada.
   a. Si utiliza las direcciones de correo electrónico y los números de teléfono de la empresa como información de contacto, podrá comunicarse aunque la persona a la que pertenezca dicha dirección o teléfono haya abandonado la organización. Por ejemplo, cree listas de correo electrónico diferentes para la facturación, las operaciones y la seguridad, y configure estos contactos como Facturación, Seguridad y Operaciones en cada Cuenta de AWS activa. Las notificaciones de AWS llegarán a diversas personas y se responderán incluso si alguien está de vacaciones, cambia de rol o deja la empresa.
   b. En caso de que AWS Organizations no administre una cuenta, los contactos alternativos de esta permitirán que AWS pueda ponerse en contacto con las personas adecuadas si fuera

necesario. Configure los contactos alternativos de la cuenta para que apunten a un grupo en lugar de a una persona.

3. Utilice etiquetas para identificar a los propietarios de los recursos de AWS. Puede especificar tanto los propietarios como su información de contacto en etiquetas independientes.
   a. Puede usar reglas de AWS Config para garantizar que los recursos tengan las etiquetas de propiedad requeridas.
   b. Para obtener instrucciones detalladas sobre cómo crear una estrategia de etiquetado para su organización, consulte el documento técnico sobre prácticas recomendadas de etiquetado de AWS.
4. Utilice Amazon Q Business, un asistente conversacional que utiliza IA generativa para mejorar la productividad de la fuerza laboral, responder preguntas y completar tareas en función de la información de los sistemas empresariales.
   a. Conecte Amazon Q Business al origen de datos de su empresa. Amazon Q Business ofrece conectores prediseñados para más de 40 orígenes de datos compatibles, incluidos Amazon Simple Storage Service (Amazon S3), Microsoft SharePoint, Salesforce y Atlassian Confluence. Para obtener más información, consulte Conectores de Amazon Q Business.
5. Para otros recursos, plataformas e infraestructuras, cree documentación que identifique la propiedad. Todos los miembros del equipo deben poder acceder a ella.

Nivel de esfuerzo para el plan de implementación: bajo. Utilice la información de contacto de la cuenta y las etiquetas para asignar la propiedad de los recursos de AWS. Para otros recursos puede utilizar algo tan simple como una tabla en un wiki para registrar la propiedad y la información de contacto o una herramienta ITSM para asignar la propiedad.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP02 Procesos y procedimientos con propietarios identificados
- OPS02-BP04 Mecanismos existentes para administrar las responsabilidades y la propiedad

Documentos relacionados:

- AWS Account Management - Updating contact information
- AWS Organizations - Updating alternative contacts in your organization
- Documento técnico sobre prácticas recomendadas de etiquetado de AWS

- Build private and secure enterprise generative AI apps with Amazon Q Business and AWS IAM Identity Center
- Amazon Q Business, now generally available, helps boost workforce productivity with generative AI
- Nube de AWS Operations & Migrations Blog - Implementing automated and centralized tagging controls with AWS Config and AWS Organizations
- AWS Security Blog - Extend your pre-commit hooks with AWS CloudFormation Guard
- AWS DevOps Blog - Integrating AWS CloudFormation Guard into CI/CD pipelines

Talleres relacionados:

- AWS Workshop: Tagging

Ejemplos relacionados:

- Reglas de AWS Config - Amazon EC2 with required tags and valid values

Servicios relacionados:

- Reglas de AWS Config - required-tags
- AWS Organizations

OPS02-BP02 Procesos y procedimientos con propietarios identificados

Conozca quién tiene la propiedad de la definición de procesos y procedimientos específicos, por qué se utilizan esos procesos y procedimientos y por qué existe esa propiedad. Comprender las razones por las que se utilizan procesos y procedimientos específicos permite identificar las oportunidades de mejora.

Resultado deseado: su organización cuenta con un conjunto de procesos y procedimientos bien definidos y mantenidos para las tareas operativas. El proceso y los procedimientos se almacenan en una ubicación central y están disponibles para los miembros de su equipo. El proceso y los procedimientos se actualizan con frecuencia, a través de la propiedad claramente asignada. Siempre que es posible, los scripts, las plantillas y los documentos de automatización se implementan como código.

Patrones comunes de uso no recomendados:

- Los procesos no se documentan. Pueden existir scripts fragmentados en estaciones de trabajo de operadores aisladas.
- Solo unas pocas personas o el equipo de manera informal saben cómo usar los scripts.
- Está previsto actualizar un proceso heredado, pero la propiedad de la actualización no está clara y el autor original ya no forma parte de la organización.
- Los procesos y los scripts no se pueden detectar, por lo que no están disponibles cuando son necesarios (por ejemplo, en respuesta a un incidente).

Beneficios de establecer esta práctica recomendada:

- Los procesos y procedimientos impulsan sus esfuerzos para gestionar sus cargas de trabajo.
- Los nuevos miembros del equipo se hacen eficaces más rápidamente.
- Reducción del tiempo para mitigar los incidentes.
- Los diferentes miembros del equipo (y equipos) pueden usar los mismos procesos y procedimientos de manera coherente.
- Los equipos pueden escalar sus procesos con procesos repetibles.
- Los procesos y procedimientos estandarizados ayudan a mitigar el impacto de transferir las responsabilidades de las cargas de trabajo entre los equipos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

- Los procesos y procedimientos tienen propietarios identificados que son responsables de su definición.
  - Identifique las actividades operativas que se han llevado a cabo para ayudar a sus cargas de trabajo. Documente estas actividades en un lugar accesible.
  - Identifique de forma exclusiva a la persona o equipo responsable de la especificación de una actividad. Son responsables de verificar que un miembro del equipo con la formación adecuada y con los permisos, el acceso y las herramientas correctas pueda hacerla con éxito. Si hay problemas para llevar a cabo la actividad, los miembros del equipo que la llevan a cabo son responsables de proporcionar la información detallada necesaria para mejorar la actividad.
  - Refleje la propiedad en los metadatos del artefacto de la actividad a través de servicios como AWS Systems Manager, documentos y AWS Lambda. Capture la propiedad de los recursos mediante etiquetas o grupos de recursos y especifique la propiedad y la información de contacto.

Utilice AWS Organizations para crear políticas de etiquetas y asegurarse de que se registra la información de contacto y de propiedad.

- Con el tiempo, estos procedimientos deberían evolucionar para que puedan ejecutarse como código, lo que reduce la necesidad de intervención humana.
  - Por ejemplo, considere la posibilidad de usar funciones de AWS Lambda, plantillas de CloudFormation o documentos de automatización de AWS Systems Manager.
  - Lleve a cabo el control de versiones en los repositorios apropiados.
  - Incluya un etiquetado de recursos adecuado para que los propietarios y la documentación puedan identificarse fácilmente.

Ejemplo de cliente

AnyCompany Retail define la propiedad como el equipo o individuo que posee los procesos de una aplicación o grupos de aplicaciones (que comparten prácticas y tecnologías arquitectónicas comunes). Inicialmente, los procesos y procedimientos se documentan como guías paso a paso en el sistema de administración de documentos y se pueden encontrar mediante etiquetas en la Cuenta de AWS que aloja la aplicación y en grupos específicos de recursos dentro de la cuenta. Usa AWS Organizations para administrar sus Cuentas de AWS. Con el tiempo, estos procesos se convierten en código y los recursos se definen con infraestructura como código (por ejemplo, plantillas de CloudFormation o Kit de desarrollo de la nube de AWS (AWS CDK)). Los procesos operativos se convierten en documentos de automatización en funciones de AWS Systems Manager o AWS Lambda, que pueden iniciarse como tareas programadas, en respuesta a eventos como alarmas de AWS CloudWatch o eventos de AWS EventBridge, o iniciarse mediante solicitudes dentro de una plataforma de administración de servicios de TI (ITSM). Todos los procesos tienen etiquetas para identificar la propiedad. La documentación de la automatización y el proceso se mantiene en las páginas wiki generadas por el repositorio de código del proceso.

Pasos para la implementación

1. Documente los procesos y procedimientos existentes.
   a. Revíselos y manténgalos actualizados.
   b. Identifique un propietario para cada proceso o procedimiento.
   c. Colóquelos bajo control de versiones.
   d. Siempre que sea posible, comparta procesos y procedimientos entre cargas de trabajo y entornos que compartan diseños arquitectónicos.
2. Establezca mecanismos para recibir comentarios y mejorar.

   a. Defina políticas sobre la frecuencia con la que se deben revisar los procesos.
   b. Defina los procesos de los revisores y aprobadores.
   c. Implemente problemas o una cola de tickets para que se proporcionen comentarios y se haga un seguimiento de ellos.
   d. Siempre que sea posible, los procesos y procedimientos deben contar con la aprobación previa y la clasificación de riesgos de una junta de aprobación de cambios (CAB).
3. Verifique que los procesos y procedimientos sean accesibles y fáciles de encontrar para quienes tienen que ejecutarlos.
   a. Utilice etiquetas para indicar dónde se puede acceder a los procesos y procedimientos de la carga de trabajo.
   b. Utilice mensajes de error y eventos significativos para indicar los procesos o procedimientos adecuados para abordar un problema.
   c. Use wikis y la administración de documentos, y haga que los procesos y procedimientos se puedan buscar de manera uniforme en toda la organización.
4. Utilice Amazon Q Business, un asistente conversacional que utiliza IA generativa para mejorar la productividad de la fuerza laboral, responder preguntas y completar tareas en función de la información de los sistemas empresariales.
   a. Conecte Amazon Q Business al origen de datos de su empresa. Amazon Q Business ofrece conectores prediseñados para más de 40 orígenes de datos compatibles, incluidos Amazon S3, Microsoft SharePoint, Salesforce y Atlassian Confluence. Para obtener más información, consulte Conectores de Amazon Q.
5. Automatice cuando sea necesario.
   a. Las automatizaciones deben desarrollarse cuando los servicios y las tecnologías proporcionan una API.
   b. Imparta formaciones adecuadas sobre los procesos. Desarrolle casos de usuario y los requisitos para automatizar esos procesos.
   c. Mida la utilización de los procesos y procedimientos correctamente, y cree problemas o tickets para respaldar la mejora iterativa.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP01 Recursos con propietarios identificados
- OPS02-BP04 Mecanismos existentes para administrar las responsabilidades y la propiedad
- OPS11-BP04 Administración de conocimientos

Documentos relacionados:

- AWS Documento técnico de : Introducción a DevOps en AWS
- Documento técnico de AWS: Prácticas recomendadas para el etiquetado de los recursos de AWS
- Documento técnico de AWS: Organizing Your AWS Environment Using Multiple Accounts
- Nube de AWS Operations and Migrations Blog - Using Amazon Q Business to streamline your operations
- Nube de AWS Operations & Migrations Blog - Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Nube de AWS Operations & Migrations Blog - Implementing automated and centralized tagging controls with AWS Config and AWS Organizations
- AWS Security Blog - Extend your pre-commit hooks with AWS CloudFormation Guard
- AWS DevOps Blog - Integrating AWS CloudFormation Guard into CI/CD pipelines

Talleres relacionados:

- AWS Well-Architected Operational Excellence Workshop
- AWS Workshop: Tagging

Videos relacionados:

- How to automate IT Operations on AWS
- AWS re:Invent 2020 - Automate anything with AWS Systems Manager
- AWS re:Inforce 2022 - Automating patch management and compliance using AWS (NIS306)
- Soportes You - Diving Deep into AWS Systems Manager

Servicios relacionados:

- AWS Systems Manager - Automation

- AWS Service Management Connector

OPS02-BP03 Actividades operativas con propietarios identificados responsables de su rendimiento

Averigüe quién tiene la responsabilidad de efectuar actividades específicas en las cargas de trabajo definidas y por qué existe esa responsabilidad. Conocer quién tiene la responsabilidad de efectuar las actividades sirve para saber quién llevará a cabo la actividad, validará el resultado y proporcionará información al propietario de la actividad.

Resultado deseado:

La organización define claramente las responsabilidades para llevar a cabo actividades específicas en cargas de trabajo definidas y responder a los eventos que genera la carga de trabajo. La organización documenta la propiedad y los procesos que se llevan a cabo y hace que esta información sea fácil de encontrar. Revisa y actualiza las responsabilidades cuando se producen cambios organizativos, y los equipos hacen un seguimiento y miden el rendimiento de las actividades de identificación de defectos e ineficiencias. Implementa mecanismos de obtener comentarios para hacer un seguimiento de los defectos y las mejoras y apoya la mejora iterativa.

Patrones comunes de uso no recomendados:

- No documenta las responsabilidades.
- Existen scripts fragmentados en estaciones de trabajo de operadores aisladas. Solo unas pocas personas saben cómo usarlos o se refieren a ellos de manera informal como conocimiento de equipo.
- Hay que actualizar un proceso heredado, pero nadie sabe quién es el propietario del proceso y el autor original ya no forma parte de la organización.
- Los procesos y los scripts no se pueden encontrar y no están disponibles cuando son necesarios (por ejemplo, en respuesta a un incidente).

Beneficios de establecer esta práctica recomendada:

- Sabe quién es responsable de llevar a cabo una actividad, a quién debe notificar cuando sea necesario tomar una medida y quién toma la medida, valida el resultado y proporciona comentarios al propietario de la actividad.
- Los procesos y procedimientos impulsan sus esfuerzos para gestionar sus cargas de trabajo.
- Los nuevos miembros del equipo se hacen eficaces más rápidamente.

- Reduce el tiempo necesario para mitigar los incidentes.
- Los diferentes equipos utilizan los mismos procesos y procedimientos para llevar a cabo las tareas de manera uniforme.
- Los equipos pueden escalar sus procesos con procesos repetibles.
- Los procesos y procedimientos estandarizados ayudan a mitigar la repercusión de transferir las responsabilidades de las cargas de trabajo entre los equipos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para empezar a definir las responsabilidades, comience por la documentación existente, como las matrices de responsabilidades, los procesos y procedimientos, los roles y responsabilidades, y las herramientas y la automatización. Revise y organice debates sobre las responsabilidades de los procesos documentados. Lleve a cabo una revisión con los equipos para identificar desajustes entre las responsabilidades de los documentos y los procesos. Analice los servicios que se ofrecen con los clientes internos de ese equipo para identificar las diferencias de expectativas entre los equipos.

Analice y aborde las discrepancias. Identifique oportunidades de mejora y busque las actividades que se solicitan con frecuencia y requieren muchos recursos, que suelen ser firmes candidatas a una mejora. Examine las prácticas recomendadas, los patrones y la orientación prescriptiva para simplificar y estandarizar las mejoras. Registre las oportunidades de mejora y haga un seguimiento de las mejoras hasta el final.

Con el tiempo, estos procedimientos deberían evolucionar para ejecutarse como código, lo que reduce la necesidad de intervención humana. Por ejemplo, los procedimientos se pueden iniciar como funciones de AWS Lambda, plantillas de CloudFormation o documentos de Automatización de AWS Systems Manager. Verifique que estos procedimientos estén controlados por versiones en los repositorios apropiados e incluyan el etiquetado de recursos adecuado para que los equipos puedan identificar fácilmente a los propietarios y la documentación. Documente la responsabilidad de llevar a cabo las actividades y, a continuación, supervise las automatizaciones para que se inicien y funcionen correctamente, así como la obtención de los resultados deseados.

Ejemplo de cliente

AnyCompany Retail define la propiedad como el equipo o individuo que posee los procesos de una aplicación o grupos de aplicaciones que comparten prácticas y tecnologías arquitectónicas

comunes. Inicialmente, la empresa documenta los procesos y procedimientos como guías paso a paso en el sistema de administración documental. Hacen que los procedimientos sean fáciles de encontrar mediante etiquetas en la Cuenta de AWS que aloja la aplicación y en grupos específicos de recursos de la cuenta, y utilizan AWS Organizations para administrar sus Cuentas de AWS. Con el tiempo, AnyCompany Retail convierte estos procesos en código y define los recursos con la infraestructura como código (a través de servicios como plantillas de CloudFormation o Kit de desarrollo de la nube de AWS (AWS CDK)). Los procesos operativos se convierten en documentos de automatización en funciones de AWS Systems Manager o AWS Lambda, que pueden iniciarse como tareas programadas, en respuesta a eventos como alarmas de Amazon CloudWatch o eventos de Amazon EventBridge, o mediante solicitudes dentro de una plataforma de administración de servicios de TI (ITSM). Todos los procesos tienen etiquetas para identificar a quién pertenecen. Los equipos administran la documentación para la automatización y el proceso dentro de las páginas wiki que genera el repositorio de código para el proceso.

Pasos para la implementación

1. Documente los procesos y procedimientos existentes.
    a. Revise y verifique que estén actualizados.
    b. Verifique que cada proceso o procedimiento tenga un propietario.
    c. Ponga los procedimientos bajo un control de versiones.
    d. Siempre que sea posible, comparta procesos y procedimientos entre cargas de trabajo y entornos que compartan diseños arquitectónicos.
2. Establezca mecanismos para recibir comentarios y mejorar.
    a. Defina políticas sobre la frecuencia con la que se deben revisar los procesos.
    b. Defina los procesos de los revisores y aprobadores.
    c. Implemente problemas o una cola de tickets para proporcionar comentarios y hacer un seguimiento de ellos.
    d. Siempre que sea posible, proporcione una aprobación previa y una clasificación de riesgos de los procesos y procedimientos que ha obtenido de una junta de aprobación de cambios (CAB).
3. Haga que los procesos y procedimientos sean accesibles y fáciles de encontrar para los usuarios que necesitan ejecutarlos.
    a. Utilice etiquetas para indicar dónde se puede acceder a los procesos y procedimientos de la carga de trabajo.
    b. Utilice mensajes de error y eventos fáciles de entender para indicar los procesos o procedimientos adecuados para abordar el problema.

   c. Use wikis o la administración de documentos para que los procesos y procedimientos se puedan buscar de manera uniforme en toda la organización.

4. Automatice cuando sea apropiado.
   a. Cuando los servicios y las tecnologías tengan una API, desarrolle automatizaciones.
   b. Verifique que los procesos se entiendan bien y desarrolle los casos de usuario y los requisitos para automatizar esos procesos.
   c. Determine si los procesos y procedimientos se utilizan de forma satisfactoria y haga un seguimiento de los problemas para facilitar una mejora iterativa.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP01 Recursos con propietarios identificados
- OPS02-BP02 Procesos y procedimientos con propietarios identificados
- OPS02-BP04 Mecanismos existentes para administrar las responsabilidades y la propiedad
- OPS02-BP05 Mecanismos para solicitar adiciones, cambios y excepciones
- OPS11-BP04 Administración de conocimientos

Documentos relacionados:

- Documento técnico de AWS: Introducción a DevOps en AWS
- Documento técnico de AWS: Best Practices for Tagging AWS Resources
- Documento técnico de AWS: Organizing Your AWS Environment Using Multiple Accounts
- Nube de AWS Operations & Migrations Blog: Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- AWS Workshop: Tagging
- AWS Service Management Connector

Videos relacionados:

- AWS Knowledge Center Live | Tagging AWS Resources

- AWS re:Invent 2020 | Automate anything with AWS Systems Manager
- AWS re:Inforce 2022 | Automating patch management and compliance using AWS (NIS306)
- Soportes You | Diving Deep into AWS Systems Manager

OPS02-BP04 Mecanismos existentes para administrar las responsabilidades y la propiedad

Conozca las responsabilidades de su rol y cómo contribuye a los resultados empresariales, ya que este conocimiento determina la prioridad de sus tareas y por qué su rol es importante. Esto ayuda a los miembros del equipo a reconocer las necesidades y responder de la forma adecuada. Cuando los miembros del equipo conocen su rol, pueden establecer la propiedad, identificar oportunidades de mejora y saber cómo influir o hacer los cambios apropiados.

En ocasiones, es posible que una responsabilidad no tenga un propietario claro. En estas situaciones, diseñe un mecanismo para resolver esta carencia. Cree una ruta de derivación bien definida a alguien con autoridad para asignar la propiedad o planificar la forma de satisfacer la necesidad.

Resultado deseado: los equipos de su organización tienen responsabilidades claramente definidas que incluyen su relación con los recursos, las acciones que se deben llevar a cabo, los procesos y los procedimientos. Estas responsabilidades se corresponden con las responsabilidades y objetivos del equipo, así como con las responsabilidades de otros equipos. Debe documentar las rutas de derivación de una manera uniforme y fácil de encontrar y debe introducir estas decisiones en artefactos de documentación, como matrices de responsabilidad, definiciones de equipos o páginas wiki.

Patrones comunes de uso no recomendados:

- Las responsabilidades del equipo son ambiguas o están mal definidas.
- Los roles del equipo no se corresponden con las responsabilidades.
- El equipo no ajusta sus metas y objetivos a sus responsabilidades, lo que dificulta la medición del éxito.
- Las responsabilidades de los miembros del equipo no se corresponden con las del equipo ni con la organización en general.
- Su equipo no mantiene actualizadas las responsabilidades, lo que las hace incoherentes con las tareas que lleva a cabo.
- Las rutas de derivación para determinar las responsabilidades no están definidas o no son claras.
- Las rutas de derivación no tienen un único propietario que garantice una respuesta oportuna.

- Los roles, las responsabilidades y las rutas de derivación no son fáciles de encontrar y no están disponibles cuando son necesarios (por ejemplo, en respuesta a un incidente).

Beneficios de establecer esta práctica recomendada:

- Cuando sepa quién tiene la responsabilidad o la propiedad, podrá contactar con el equipo o el miembro del equipo adecuado para presentar una solicitud o la transición de una tarea.
- Para reducir el riesgo de inacción y de que existan necesidades no atendidas, ha identificado a una persona que tiene la autoridad para asignar la responsabilidad o la propiedad.
- Cuando define claramente el alcance de una responsabilidad, los miembros de su equipo ganan autonomía y propiedad.
- Sus responsabilidades determinan las decisiones que toma, las acciones que emprende y las actividades que transfiere a sus propietarios adecuados.
- Es fácil identificar las responsabilidades abandonadas porque tiene una idea clara de lo que queda fuera de la responsabilidad de su equipo, lo que le ayuda a derivar los problemas para aclararlos.
- Los equipos evitan la confusión y la tensión, y pueden administrar más adecuadamente sus cargas de trabajo y sus recursos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Identifique los roles y responsabilidades de los miembros del equipo y asegúrese de que conozcan las expectativas de su rol. Haga que esta información sea fácil de encontrar para que los miembros de su organización puedan identificar con quién deben contactar, ya sea un equipo o una persona. Cuando las organizaciones tratan de aprovechar las oportunidades de migrar y modernizar en AWS, los roles y responsabilidades también podrían cambiar. Mantenga a sus equipos y a sus miembros al corriente de sus responsabilidades y proporcióneles la formación adecuada para llevar a cabo sus tareas durante este cambio.

Determine el rol o el equipo que debe recibir las derivaciones para identificar la responsabilidad y la propiedad. Este equipo puede interactuar con varias partes interesadas para tomar una decisión. Sin embargo, deben ser propietarios de la administración del proceso de toma de decisiones.

Proporcione mecanismos accesibles para que los miembros de su organización descubran e identifiquen la propiedad y la responsabilidad. Estos mecanismos les enseñan con quién contactar para necesidades específicas.

Ejemplo de cliente

AnyCompany Retail completó recientemente una migración de cargas de trabajo desde un entorno en las instalaciones a su zona de aterrizaje en AWS con un enfoque de migración mediante lift-and-shift. Revisó las operaciones para determinar cómo llevar a cabo las tareas operativas comunes y verificó que su matriz de responsabilidades existente refleja las operaciones en el nuevo entorno. Cuando migró de un entorno en las instalaciones a AWS, redujo las responsabilidades de los equipos de infraestructura relacionadas con el hardware y la infraestructura física. Esta medida también dio lugar a nuevas oportunidades para hacer evolucionar el modelo operativo de sus cargas de trabajo.

Aunque identificó, abordó y documentó la mayoría de las responsabilidades, también definió rutas de derivación para cualquier responsabilidad que se hubiera pasado por alto o que pudiera tener que cambiar a medida que evolucionaran las prácticas operativas. Para explorar nuevas oportunidades para estandarizar y mejorar la eficiencia de sus cargas de trabajo, proporcione acceso a herramientas operativas como AWS Systems Manager y herramientas de seguridad como AWS Security Hub CSPM y Amazon GuardDuty. AnyCompany Retail revisa las responsabilidades y la estrategia en función de las mejoras que quiere abordar en primer lugar. A medida que la empresa adopta nuevas formas de trabajo y patrones tecnológicos, actualiza su matriz de responsabilidad de la forma correspondiente.

Pasos para la implementación

1. Comience con la documentación existente. Estos son algunos documentos iniciales típicos:
    a. Matrices de responsabilidad o de responsable, encargado, consultado e informado (RACI)
    b. Definiciones de equipo o páginas wiki
    c. Definiciones y ofertas de servicios
    d. Descripciones de roles o puestos
2. Revise y organice debates sobre las responsabilidades documentadas:
    a. Lleve a cabo una revisión con los equipos para identificar desajustes entre las responsabilidades documentadas y las responsabilidades que el equipo suele desempeñar.
    b. Analice los posibles servicios que ofrecen los clientes internos para identificar las diferencias de expectativas entre los equipos.
3. Analice y aborde las discrepancias.
4. Identifique oportunidades de mejora.

   a. Identifique las solicitudes más frecuentes y que requieren más recursos, que suelen ser firmes candidatas a una mejora.
   b. Busque las prácticas recomendadas, conozca los patrones, siga las recomendaciones y simplifique y estandarice las mejoras.
   c. Registre las oportunidades de mejora y haga un seguimiento de ellas hasta el final.
5. Si un equipo aún no tiene la responsabilidad de administrar y hacer un seguimiento de la asignación de responsabilidades, identifique a alguien del equipo para que asuma esta responsabilidad.
6. Defina un proceso para que los equipos soliciten una aclaración de la responsabilidad.
   a. Revise el proceso y verifique que sea claro y fácil de usar.
   b. Asegúrese de que alguien sea el propietario de las derivaciones y haga un seguimiento de ellas hasta el final.
   c. Establezca métricas operativas para medir la eficacia.
   d. Cree mecanismos para obtener comentarios para verificar que los equipos puedan llamar la atención sobre las oportunidades de mejora.
   e. Implemente un mecanismo de revisión periódica.
7. Lleve a cabo la documentación en una ubicación accesible y reconocible.
   a. Las wikis o el portal de documentación son opciones comunes.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP06 Evaluación de las compensaciones
- OPS03-BP02 Preparación de los miembros del equipo para actuar cuando los resultados están en riesgo
- OPS03-BP03 Fomento de la derivación
- OPS03-BP07 Recursos adecuados para los equipos
- OPS09-BP01 Medición de los objetivos operativos y los KPI con métricas
- OPS09-BP03 Revisión de las métricas de las operaciones y priorización de las mejoras
- OPS11-BP01 Implementación de un proceso de mejora continua

Documentos relacionados:

- Documento técnico de AWS: Introducción a DevOps en AWS
- Documento técnico de AWS: Nube de AWS Cloud Adoption Framework: Operations Perspective
- Excelencia operativa del Marco de AWS Well-Architected: Representaciones 2 por 2 del modelo operativo
- Guía prescriptiva de AWS: Building your Cloud Operating Model
- Guía prescriptiva de AWS: Create a RACI or RASCI matrix for a cloud operating model
- Nube de AWS Operations & Migrations Blog - Delivering Business Value with Cloud Platform Teams
- Nube de AWS Operations & Migrations Blog - Why a Cloud Operating Model?
- AWS DevOps Blog - How organizations are modernizing for cloud operations

Videos relacionados:

- AWS Summit Online - Cloud Operating Models for Accelerated Transformation
- AWS re:Invent 2023 - Future-proofing cloud security: A new operating model

OPS02-BP05 Mecanismos para solicitar adiciones, cambios y excepciones

Puede presentar solicitudes a los propietarios de los procesos, procedimientos y recursos. Las solicitudes incluyen adiciones, cambios y excepciones. Estas solicitudes pasan por un proceso de administración de cambios. Tome decisiones fundamentadas para aprobar las solicitudes cuando sean viables y se determine que son adecuadas tras una evaluación de los beneficios y los riesgos.

Resultado deseado:

- Puede presentar solicitudes para cambiar procesos, procedimientos y recursos en función de la propiedad asignada.
- Los cambios se hacen de forma deliberada y se tienen en cuenta los beneficios y los riesgos.

Patrones comunes de uso no recomendados:

- Debe actualizar la forma de implementar su aplicación, pero no hay forma de solicitar un cambio en el proceso de implementación al equipo de operaciones.

- El plan de recuperación de desastres debe actualizarse, pero no hay ningún propietario identificado al que solicitar cambios.

Beneficios de establecer esta práctica recomendada:

- Los procesos, los procedimientos y los recursos pueden evolucionar a medida que cambian los requisitos.
- Los propietarios pueden tomar decisiones fundamentadas cuando hacen cambios.
- Los cambios se hacen de forma deliberada.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para implementar esta práctica recomendada, debe poder solicitar cambios en los procesos, los procedimientos y los recursos. El proceso de administración de los cambios puede ser ligero. Documente el proceso de administración de cambios.

Ejemplo de cliente

AnyCompany Retail utiliza una matriz de asignación de responsabilidades (RACI) para identificar a quién corresponden los cambios en los procesos, los procedimientos y los recursos. Dispone de un proceso de administración de cambios documentado, ligero y fácil de seguir. Con la matriz RACI y el proceso, cualquiera puede enviar solicitudes de cambio.

Pasos para la implementación

1. Identifique los procesos, los procedimientos y los recursos de su carga de trabajo y los responsables de cada uno de ellos. Documéntelos en su sistema de administración de conocimientos.
   a. Si no ha implementado OPS02-BP01 Recursos con propietarios identificados, OPS02-BP02 Procesos y procedimientos con propietarios identificados o OPS02-BP03 Actividades operativas con propietarios identificados responsables de su rendimiento, empiece con ellos en primer lugar.
2. Colabore con las partes interesadas de su organización para desarrollar un proceso de administración de cambios. El proceso debe abarcar las incorporaciones, los cambios y las excepciones de recursos, procesos y procedimientos.

   a. Puede utilizar el Administrador de cambios de AWS Systems Manager como plataforma de gestión de cambios para los recursos de carga de trabajo.
3. Documente el proceso de administración de cambios en su sistema de administración de conocimientos.

Nivel de esfuerzo para el plan de implementación: medio. El desarrollo de un proceso de administración de cambios requiere la coordinación con las múltiples partes interesadas de toda la organización.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP01 Recursos con propietarios identificados: es necesario identificar a los propietarios de los recursos antes de crear un proceso de gestión de cambios.
- OPS02-BP02 Procesos y procedimientos con propietarios identificados: es necesario identificar a los propietarios de los procesos antes de crear un proceso de gestión de cambios.
- OPS02-BP03 Actividades operativas con propietarios identificados responsables de su rendimiento: es necesario identificar a los propietarios de las actividades operativas antes de crear un proceso de gestión de cambios.

Documentos relacionados:

- Guía prescriptiva de AWS: Foundation playbook for AWS large migrations: Creating RACI matrices
- Documento técnico Change Management in the Cloud

Servicios relacionados:

- AWS Systems Manager Change Manager

OPS02-BP06 Responsabilidades predefinidas o negociadas entre equipos

Posibilite que se definan o negocien acuerdos entre equipos que describan cómo trabajan y se apoyan mutuamente (por ejemplo, tiempos de respuesta, objetivos de nivel de servicio o acuerdos de nivel de servicio). Los canales de comunicación entre equipos están documentados. Comprender el impacto del trabajo de los equipos en los resultados de la empresa y los resultados de otros

equipos y organizaciones fundamenta la priorización de sus tareas y contribuye a que respondan adecuadamente.

Cuando la responsabilidad y la propiedad no están definidas o se desconocen, se corre el riesgo de que no se aborden las actividades necesarias a tiempo y de que se hagan esfuerzos repetidos y potencialmente conflictivos para satisfacer esas necesidades.

Resultado deseado:

- Se acuerdan y documentan los acuerdos de trabajo o asistencia entre equipos.
- Los equipos que se prestan asistencia o colaboran entre sí tienen definidos los canales de comunicación y las expectativas de respuesta.

Patrones comunes de uso no recomendados:

- Se produce un problema en producción y dos equipos distintos empiezan a solucionar los problemas independientemente el uno del otro. Sus esfuerzos aislados prolongan la interrupción.
- El equipo de operaciones necesita ayuda del equipo de desarrollo, pero no hay un tiempo de respuesta acordado. La solicitud está atascada en la lista de tareas pendientes.

Beneficios de establecer esta práctica recomendada:

- Los equipos saben cómo interactuar y prestarse asistencia mutua.
- Se conocen las expectativas de capacidad de respuesta.
- Los canales de comunicación están claramente definidos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

La implementación de esta práctica recomendada significa que no existe ninguna ambigüedad sobre cómo colaboran los equipos. Los acuerdos formales codifican la forma en que los equipos trabajan juntos o se prestan asistencia mutua. Los canales de comunicación entre equipos están documentados.

Ejemplo de cliente

El equipo de SRE de AnyCompany Retail tiene un acuerdo de nivel de servicio con su equipo de desarrollo. Cada vez que el equipo de desarrollo presenta una solicitud en su sistema de tickets, puede esperar una respuesta en quince minutos. Si se produce una interrupción en el sitio, el equipo de SRE toma la iniciativa en la investigación con la ayuda del equipo de desarrollo.

Pasos para la implementación

1. En colaboración con las partes interesadas de toda su organización, desarrolle acuerdos entre los equipos basados en procesos y procedimientos.
   a. Si dos equipos comparten un proceso o un procedimiento, elabore un manual de procedimientos sobre cómo colaborarán.
   b. Si existen dependencias entre los equipos, acuerde un SLA de respuesta para las solicitudes.
2. Documente las responsabilidades en su sistema de administración de conocimientos.

Nivel de esfuerzo para el plan de implementación: medio. Si no existen acuerdos entre los equipos, puede resultar difícil llegar a un acuerdo con las partes interesadas de toda la organización.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP02 Procesos y procedimientos con propietarios identificados: se debe identificar la propiedad del proceso antes de establecer acuerdos entre los equipos.
- OPS02-BP03 Actividades operativas con propietarios identificados responsables de su rendimiento: se debe identificar la propiedad de las actividades operativas antes de establecer acuerdos entre los equipos.

Documentos relacionados:

- AWS Executive Insights: Estimulación de la innovación y la velocidad con los equipos de dos pizzas de Amazon
- Introducción a DevOps en AWS: Equipos de dos pizzas

## OPS 3. ¿Cómo ayuda la cultura de su organización a lograr los resultados empresariales?

Ayude a los miembros de su equipo para que puedan actuar de manera más eficaz y contribuir a avanzar hacia los objetivos de la empresa.

Prácticas recomendadas

- OPS03-BP01 Respaldo del área ejecutiva
- OPS03-BP02 Preparación de los miembros del equipo para actuar cuando los resultados están en riesgo
- OPS03-BP03 Fomento de la derivación
- OPS03-BP04 Comunicaciones oportunas, claras y procesables
- OPS03-BP05 Fomento de la experimentación
- OPS03-BP06 Fomento para que los miembros del equipo mantengan y aumenten su conjunto de competencias
- OPS03-BP07 Recursos adecuados para los equipos

### OPS03-BP01 Respaldo del área ejecutiva

En el nivel más alto, los líderes sénior actúan como patrocinadores ejecutivos para fijar de forma clara las expectativas y la dirección de los resultados de la organización, incluida la evaluación de su éxito. El patrocinador defiende e impulsa la adopción de las prácticas recomendadas y la evolución de la organización.

Resultado deseado: las organizaciones que se esfuerzan por adoptar, transformar y optimizar sus operaciones en la nube establecen líneas claras de liderazgo y responsabilidad para lograr los resultados deseados. La organización sabe cuáles son todas las capacidades que necesita para lograr un nuevo resultado y asigna la propiedad a los equipos funcionales para su desarrollo. Los líderes marcan activamente esta dirección, asignan la propiedad, asumen la responsabilidad y definen el trabajo. Como resultado, las personas de toda la organización pueden movilizarse, sentirse inspiradas y trabajar activamente para alcanzar los objetivos deseados.

Patrones comunes de uso no recomendados:

- Los propietarios de las cargas de trabajo tienen la obligación de migrarlas a AWS sin un patrocinador y un plan claros para las operaciones en la nube. El resultado es que los equipos no colaboran conscientemente para mejorar y madurar sus capacidades operativas. La falta de

prácticas recomendadas operativas estándar sobrecarga a los equipos (por ejemplo, trabajos de operadores, guardias y deuda técnica), lo que limita la innovación.

- Se ha fijado el objetivo de adoptar una tecnología emergente en toda la organización sin proporcionar ningún patrocinador de liderazgo ni estrategia. Los equipos interpretan los objetivos de manera diferente, lo que genera confusión sobre dónde centrar los esfuerzos, por qué son importantes y cómo medir la repercusión. En consecuencia, la organización pierde impulso a la hora de adoptar la tecnología.

Beneficios de establecer esta práctica recomendada: cuando el patrocinio ejecutivo comunica y comparte la visión, la dirección y los objetivos de una forma clara, los miembros del equipo saben lo que se espera de ellos. Cuando los líderes participan activamente, las personas y los equipos comienzan a centrar intensamente sus esfuerzos en la misma dirección para lograr los objetivos definidos. Como resultado, la organización aumenta al máximo la posibilidad de éxito. Si evalúa el éxito, podrá identificar mejor las barreras que impiden conseguirlo para abordarlas mediante la intervención del patrocinador ejecutivo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

- En cada fase del traspaso a la nube (migración, adopción u optimización), el éxito requiere una implicación activa al más alto nivel de liderazgo y que haya un patrocinador ejecutivo designado. El patrocinador ejecutivo ajusta la mentalidad, las competencias y las formas de trabajar del equipo con la estrategia definida.
  - Explicación del motivo: aclare y explique el razonamiento que hay detrás de la visión y la estrategia.
  - Definición de expectativas: defina y publique los objetivos de sus organizaciones, incluida la forma en que se miden el progreso y el éxito.
  - Seguimiento del logro de los objetivos: mida el logro gradual de los objetivos con regularidad (no solo la finalización de las tareas). Comparta las conclusiones para que puedan tomarse las medidas adecuadas si los resultados están en peligro.
  - Disposición de los recursos necesarios para alcanzar sus objetivos: reúna a las personas y los equipos para que colaboren y creen las soluciones adecuadas que dan lugar a los resultados definidos. Esto reduce o elimina la fricción organizativa.
  - Apoyo a los equipos: manténgase en contacto con los equipos para conocer su rendimiento y saber si hay factores externos que les afectan. Identifique los obstáculos que impiden que

el equipo avance. Actúe en nombre de sus equipos para ayudar a abordar los obstáculos y eliminar las cargas innecesarias. Cuando sus equipos se vean afectados por factores externos, vuelva a evaluar los objetivos y ajústelos según convenga.

- Impulso de la adopción de prácticas recomendadas: reconozca las prácticas recomendadas que proporcionan beneficios cuantificables y exprese su reconocimiento a sus creadores y a los que las han adoptado. Fomente una mayor adopción para incrementar los beneficios conseguidos.
- Fomento de la evolución de sus equipos: cree una cultura de mejora continua y aprenda de forma proactiva tanto de los avances logrados como de los fracasos. Fomente el crecimiento y el desarrollo tanto personal como de la organización. Utilice datos y anécdotas para desarrollar la visión y la estrategia.

Ejemplo de cliente

AnyCompany Retail está en proceso de transformación empresarial mediante la rápida reinvención de las experiencias de los clientes, la mejora de la productividad y la aceleración del crecimiento a través de la IA generativa.

Pasos para la implementación

1. Establezca un liderazgo de un solo enfoque y asigne un patrocinador ejecutivo principal para dirigir e impulsar la transformación.
2. Defina unos resultados empresariales claros de su transformación y asigne la propiedad y la responsabilidad. Otorgue al ejecutivo principal la autoridad para dirigir y tomar decisiones críticas.
3. Verifique que su estrategia de transformación sea muy clara y que el patrocinador ejecutivo la comunique de forma generalizada en todos los niveles de la organización.
    a. Defina con claridad los objetivos empresariales para las iniciativas de TI y la nube.
    b. Documente las métricas empresariales clave para impulsar la transformación de TI y la nube.
    c. Comunique la visión de manera uniforme a todos los equipos y personas responsables de alguna parte de la estrategia.
4. Desarrolle matrices de planificación de la comunicación en las que se especifique qué mensaje debe darse a los líderes, gerentes y colaboradores individuales específicos. Especifique la persona o el equipo que debe entregar este mensaje.
    a. Siga los planes de comunicación de manera uniforme y fiable.
    b. Marque y administre las expectativas a través de eventos presenciales de forma regular.

   c. Acepte los comentarios sobre la eficacia de las comunicaciones y ajuste las comunicaciones y el plan en consecuencia.
   d. Programe eventos de comunicación para conocer de manera proactiva los desafíos de los equipos y establezca un ciclo de comentarios uniforme que permita corregir el rumbo cuando sea necesario.
5. Comprométase activamente con cada iniciativa desde una perspectiva de liderazgo para verificar que todos los equipos afectados conocen los resultados que deben alcanzar.
6. En cada reunión de estado, los patrocinadores ejecutivos deben buscar los obstáculos, inspeccionar las métricas establecidas, las anécdotas o los comentarios de los equipos y medir el progreso hacia los objetivos.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS03-BP04 Comunicaciones oportunas, claras y procesables
- OP11-BP01 Implementación de un proceso de mejora continua
- OPS11-BP07 Revisiones de métricas de operaciones

Documentos relacionados:

- Untangling Your Organisational Hairball: Highly Aligned
- The Living Transformation: Pragmatically approaching changes
- Becoming a Future-Ready Enterprise
- 7 Pitfalls to Avoid When Building a CCOE
- Navigating the Cloud: Key Performance Indicators for Success

Videos relacionados:

- AWS re:Invent 2023: A leader's guide to generative AI: Using history to shape the future (SEG204)

Ejemplos relacionados:

- Prosci: Primary Sponsor's Role and Importance

OPS03-BP02 Preparación de los miembros del equipo para actuar cuando los resultados están en riesgo

Si el comportamiento cultural de propiedad lo inculcan los líderes, todos los empleados se sentirán preparados para actuar en nombre de toda la empresa más allá del ámbito definido de su rol y responsabilidad. Los empleados pueden actuar para identificar de forma proactiva los riesgos a medida que surjan y tomar las medidas adecuadas. Una cultura así permite a los empleados tomar decisiones de alto valor con conocimiento de la situación.

Por ejemplo, Amazon utiliza los Principios de liderazgo como directrices para que los empleados tengan el comportamiento deseado para desenvolverse en diferentes situaciones, resolver problemas, abordar conflictos y tomar medidas.

Resultado deseado: los líderes han influido en una nueva cultura que permite a las personas y los equipos tomar decisiones críticas, incluso en los niveles inferiores de la organización (siempre que las decisiones se definan con permisos y mecanismos de seguridad auditables). No se desalienta el fracaso y los equipos aprenden iterativamente a mejorar su toma de decisiones y sus respuestas para afrontar situaciones similares en el futuro. Si las acciones de alguien se traducen en una mejora que puede beneficiar a otros equipos, se comparten de forma proactiva los conocimientos derivados de dichas acciones. Los líderes miden las mejoras operativas e incentivan al individuo y a la organización para que adopten dichos patrones.

Patrones comunes de uso no recomendados:

- No hay directrices ni mecanismos claros en una organización sobre qué hacer cuando se identifica un riesgo. Por ejemplo, cuando un empleado se da cuenta de que ha sufrido ataque de phishing, no informa al equipo de seguridad, lo que hace que una gran parte de la organización caiga en la trampa del ataque. Esto da lugar a una vulneración de los datos.
- Sus clientes se quejan de la falta de disponibilidad del servicio, que se debe principalmente a fallos de implementación. Su equipo de SRE es responsable de la herramienta de implementación y figura la reversión automática de las implementaciones en su hoja de ruta a largo plazo. En una implementación reciente de una aplicación, a uno de los ingenieros se le ocurrió una solución para automatizar la reversión de su aplicación a una versión anterior. Aunque su solución puede convertirse en el patrón para los equipos de SRE, otros equipos no la adoptan, ya que no existe un proceso para hacer un seguimiento de dichas mejoras. La organización sigue plagada de implementaciones fallidas que afectan a los clientes y aumentan el sentimiento negativo.

- Para cumplir las normativas, su equipo de seguridad de la información supervisa un proceso establecido desde hace tiempo para rotar regularmente las claves SSH compartidas en nombre de los operadores que se conectan a sus instancias de Amazon EC2 en Linux. Los equipos de seguridad de la información tardan varios días en completar la rotación de claves y no puede conectarse a esas instancias. Nadie, dentro ni fuera del equipo de seguridad de la información, sugiere utilizar otras opciones en AWS para lograr el mismo resultado.

Beneficios de establecer esta práctica recomendada: al descentralizar la autoridad para tomar decisiones y preparar a sus equipos para que tomen decisiones clave, puede abordar los problemas más rápidamente con unos índices de éxito cada vez mayores. Además, los equipos empiezan a darse cuenta del sentido de propiedad, y los fracasos son aceptables. La experimentación se convierte en un pilar cultural. Los gerentes y directores no se sienten controlados al mínimo detalle en todos los aspectos de su trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

1. Desarrolle una cultura en la que se espere que puedan producirse fallos.
2. Defina claramente la propiedad y la responsabilidad de las distintas áreas funcionales de la organización.
3. Comunique la propiedad y la responsabilidad a todo el mundo para que las personas sepan quién puede ayudarles a tomar decisiones descentralizadas.
4. Defina sus decisiones unidireccionales y bidireccionales para ayudar a las personas a saber cuándo deben derivar a niveles más altos de liderazgo.
5. Conciencie a toda la organización de que todos los empleados están facultados para actuar a varios niveles cuando los resultados están en peligro. Proporcione a los miembros de su equipo documentación sobre gobernanza, niveles de permisos, herramientas y oportunidades para practicar las competencias necesarias para responder con eficacia.
6. Dé a los miembros de su equipo la oportunidad de practicar las competencias necesarias para responder a diversas decisiones. Una vez que se hayan definido los niveles de decisión, lleva a cabo eventos de GameDay para verificar que todos los colaboradores individuales conozcan y puedan demostrar el proceso.
   a. Proporcione entornos alternativos seguros en los que se puedan probar y entrenar los procesos y procedimientos.

   b. Confirme que los miembros del equipo tengan autoridad para tomar medidas cuando el resultado tenga un nivel de riesgo predefinido y conciencie sobre esa idea.
   c. Defina la autoridad de los miembros del equipo para tomar medidas mediante la asignación de permisos y acceso a las cargas de trabajo y los componentes que respaldan.
7. Proporcione a los equipos la capacidad de compartir lo que han aprendido (éxitos y fracasos operativos).
8. Prepare a los equipos para que desafíen el statu quo y proporcione mecanismos para rastrear y medir las mejoras, así como su repercusión en la organización.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP06 Evaluación de las compensaciones al administrar los beneficios y los riesgos
- OPS02-BP05 Mecanismos para solicitar adiciones, cambios y excepciones

Documentos relacionados:

- Entrada en el blog de AWS: The agile enterprise
- Entrada en el blog de AWS: Measuring success: A paradox and a plan
- Entrada en el blog de AWS: Letting go: Enabling autonomy in teams
- Centralize or Decentralize?

Videos relacionados:

- re:Invent 2023 | How to not sabotage your transformation (SEG201)
- re:Invent 2021 | Amazon Builders' Library: Operational Excellence at Amazon
- Centralization vs. Decentralization

Ejemplos relacionados:

- Using architectural decision records to streamline technical decision-making for a software development project

### OPS03-BP03 Fomento de la derivación

Los líderes animan a los miembros del equipo a derivar los problemas y preocupaciones a los responsables de la toma de decisiones de mayor nivel y las partes interesadas si creen que los resultados deseados están en peligro y no se cumplen los estándares esperados. Esta es una característica de la cultura de la organización y se impulsa en todos los niveles. La derivación debe hacerse de forma temprana y frecuente para poder identificar los riesgos y evitar que provoquen incidentes. Los líderes no reprenden a las personas por derivar un problema.

Resultado deseado: las personas de toda la organización se sienten cómodas al derivar los problemas a sus niveles de liderazgo inmediatos y superiores. Los líderes han establecido de forma deliberada y consciente expectativas para que sus equipos se sientan seguros para derivar cualquier asunto. Existe un mecanismo para derivar los problemas en cada nivel de la organización. Cuando los empleados recurren a su gerente, deciden conjuntamente el grado de repercusión y si el problema debe derivarse. Para iniciar una derivación, los empleados deben incluir un plan de trabajo recomendado para abordar el problema. Si el equipo directivo directo no toma las medidas oportunas, se anima a los empleados a que lleven los problemas al nivel más alto de liderazgo si creen firmemente que los riesgos para la organización justifican la derivación.

Patrones comunes de uso no recomendados:

- Los líderes ejecutivos no hacen suficientes preguntas de sondeo durante la reunión sobre el estado de su programa de transformación en la nube para averiguar dónde se producen los problemas y los obstáculos. Solo se les informa de las buenas noticias. La CIO ha dejado claro que solo le gusta escuchar buenas noticias, ya que cualquier reto que se plantee hace pensar al CEO que el programa está fracasando.
- Es un ingeniero de operaciones en la nube y observa que los equipos de aplicaciones no están adoptando de forma generalizada el nuevo sistema de administración del conocimiento. La empresa invirtió un año y varios millones de dólares para implementar este nuevo sistema de administración del conocimiento, pero la gente sigue creando sus manuales de procedimientos localmente y compartiéndolos en un recurso compartido en la nube de la organización, lo que hace difícil buscar información relativa a las cargas de trabajo asumidas. Trata de llamar la atención de los líderes sobre este asunto, porque un uso uniforme de este sistema puede mejorar la eficacia operativa. Cuando se lo plantea a la directora que dirigió la implementación del sistema de administración del conocimiento, esta le reprende porque pone en entredicho la inversión.
- El equipo de seguridad de la información responsable de reforzar los recursos de computación ha decidido implementar un proceso que exige llevar a cabo los análisis necesarios para garantizar que las instancias de EC2 estén totalmente protegidas antes de que el equipo de computación

publique el recurso para su uso. A causa de esto, se ha producido un retraso de una semana en la implementación de los recursos, lo que infringe su SLA. El equipo de computación tiene miedo de derivar esta situación al VP de la nube porque esto hace quedar mal al VP de seguridad de la información.

Beneficios de establecer esta práctica recomendada:

Los problemas complejos o críticos se abordan antes de que afecten al negocio. Se pierde menos tiempo. Los riesgos se minimizan. Los equipos se vuelven más proactivos y se centran en los resultados a la hora de resolver problemas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La voluntad y la capacidad de derivar los asuntos libremente en todos los niveles de la organización es una base organizativa y cultural que debe desarrollarse conscientemente a través de una mayor formación, las comunicaciones de los líderes, el establecimiento de expectativas y la implementación de mecanismos en todos los niveles de la organización.

Pasos para la implementación

1. Defina políticas, estándares y expectativas para su organización.
    a. Garantice la adopción y comprensión generalizada de las políticas, expectativas y estándares.
2. Anime, forme y prepare a los trabajadores para que deriven los incumplimientos de los estándares cuanto antes y con frecuencia.
3. Confirme en el nivel de la organización que la práctica recomendada es derivar cuanto antes y con frecuencia. Acepte que las derivaciones pueden ser infundadas y que es mejor tener la oportunidad de prevenir un incidente que perder esa oportunidad por no haber derivado.
    a. Cree un mecanismo de derivación (como un sistema Andon Cord).
    b. Disponga de procedimientos documentados que definan cuándo y cómo debe derivarse.
    c. Defina quiénes son las personas con mayor autoridad para tomar o aprobar acciones, así como la información de contacto de cada parte interesada.
4. Cuando se produce una derivación, esta debe continuar hasta que el miembro del equipo esté convencido de que el riesgo se ha mitigado mediante acciones impulsadas por los líderes.
    a. Las derivaciones deben incluir:

i. Descripción de la situación y naturaleza del riesgo
ii. Gravedad de la situación
iii. Quién o qué se ve afectado
iv. Cuál es la repercusión
v. Nivel de urgencia si hay alguna repercusión
vi. Soluciones sugeridas y planes de mitigación

b. Proteja a los empleados que derivan. Disponga de una política que proteja a los miembros del equipo frente a represalias si derivan a un responsable de la toma de decisiones o a una parte interesada que no responde. Disponga de mecanismos para identificar si está ocurriendo esto y responder de forma adecuada.

5. Fomente una cultura de bucles de retroalimentación de mejora continua en todo lo que produce la organización. Los bucles de retroalimentación sirven como derivaciones menores a las personas responsables e identifican oportunidades de mejora, incluso cuando la derivación no es necesaria. Las culturas de mejora continua obligan a todo el mundo a ser más proactivo.
6. Los líderes deben volver a insistir periódicamente en las políticas, los estándares, los mecanismos y el deseo de que se produzcan derivaciones abiertas y bucles de comentarios continuos sin retribución.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP05 Mecanismos para solicitar adiciones, cambios y excepciones

Documentos relacionados:

- How do you foster a culture of continuous improvement and learning from Andon and escalation systems?
- The Andon Cord (IT Revolution)
- AWS DevOps Guidance | Establish clear escalation paths and encourage constructive disagreement

Videos relacionados:

- Jeff Bezos on how to make decisions (& increase velocity)
- Toyota Product System: Stopping Production, a Button, and an Andon Electric Board
- Andon Cord in LEAN Manufacturing

Ejemplos relacionados:

- Working with escalation plans in Incident Manager

OPS03-BP04 Comunicaciones oportunas, claras y procesables

Los líderes son responsables de la creación de comunicaciones sólidas y eficaces, especialmente cuando la organización adopta nuevas estrategias, tecnologías o formas de trabajar. Los líderes deben marcar expectativas para que todo el personal trabaje en pos de los objetivos de la empresa. Diseñe mecanismos de comunicación que creen y mantengan la concienciación entre los equipos responsables de ejecutar los planes financiados y patrocinados por los líderes. Aproveche la diversidad interorganizativa y escuche atentamente las numerosas perspectivas únicas. Utilice esta perspectiva para aumentar la innovación, cuestionar sus suposiciones y reducir el riesgo de sesgo de confirmación. Fomente la inclusión, la diversidad y la accesibilidad en sus equipos para obtener perspectivas que sean beneficiosas.

Resultado deseado: su organización diseña estrategias de comunicación para abordar la repercusión del cambio en la organización. Los equipos se mantienen informados y motivados para seguir trabajando juntos y no unos contra los otros. Las personas entienden lo importante que es su rol para lograr los objetivos establecidos. El correo electrónico es solo un mecanismo pasivo de comunicación y se usa en consecuencia. Los directivos dedican tiempo a sus colaboradores individuales para ayudarlos a comprender cuál es su responsabilidad, las tareas que deben llevar a cabo y de qué forma su trabajo contribuye a la misión general. Cuando es necesario, los líderes se reúnen directamente con las personas en salas más pequeñas para transmitirles mensajes y verificar que estos se han transmitido de forma eficaz. Como resultado de las buenas estrategias de comunicación, la organización funciona a la altura o por encima de las expectativas de los líderes. Los líderes fomentan y buscan opiniones diversas dentro de los equipos y entre ellos.

Patrones comunes de uso no recomendados:

- Su organización tiene un plan quinquenal para migrar todas las cargas de trabajo a AWS. El argumento empresarial a favor de la nube incluye la modernización del 25 % de todas las cargas de trabajo para aprovechar la tecnología sin servidor. El CIO comunica esta estrategia

a sus subordinados directos y espera que cada líder se la transmita en cascada a los gerentes, directores y colaboradores individuales sin ninguna comunicación en persona. El CIO se queda a un lado y espera que su organización lleve a cabo la nueva estrategia.

- Los líderes no proporcionan ni utilizan un mecanismo de comentarios y la brecha en las expectativas crece, lo que conduce al estancamiento de los proyectos.
- Se le pide que haga un cambio en sus grupos de seguridad, pero no se le da ningún detalle sobre qué cambio hay que hacer, cuál podría ser la repercusión del cambio en todas las cargas de trabajo y cuándo debería producirse. El gerente reenvía un correo electrónico del vicepresidente de seguridad de la información y agrega el mensaje “Make this happen”.
- Se han hecho cambios en la estrategia de migración que reducen el número de modernizaciones previstas del 25 % al 10 %. Esto tiene efectos posteriores en la organización de las operaciones. No se les informó de este cambio estratégico y, por lo tanto, no están preparados ni tienen la suficiente capacidad cualificada para admitir un mayor número de cargas de trabajo migradas mediante lift-and-shift a AWS.

Beneficios de establecer esta práctica recomendada:

- Su organización está bien informada sobre las estrategias nuevas o modificadas y actúa en consecuencia con una fuerte motivación para ayudarse mutuamente a alcanzar los objetivos generales y las métricas que han fijado los líderes.
- Existen y se utilizan mecanismos para avisar a tiempo a los miembros del equipo de los riesgos conocidos y de los eventos planificados.
- La organización adopta de manera más eficaz las nuevas formas de trabajar (incluidos los cambios en las personas o la organización, los procesos o la tecnología), además de las competencias requeridas, y su organización obtiene beneficios empresariales con mayor rapidez.
- Los miembros del equipo tienen el contexto necesario de las comunicaciones que reciben y pueden ser más eficaces en su trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para implementar esta práctica recomendada, debe colaborar con las partes interesadas de toda su organización para acordar unos estándares de comunicación. Dé a conocer esos estándares a su organización. En cualquier transición de TI importante, un equipo de planificación establecido puede administrar con más éxito la repercusión del cambio en su personal que una organización que

no sigue esta práctica. A las organizaciones más grandes les podría resultar más difícil administrar el cambio porque es fundamental conseguir una aceptación sólida de una nueva estrategia por parte de todos los colaboradores individuales. En ausencia de un equipo de planificación de una transición de este tipo, los líderes tienen el 100 % de la responsabilidad de que se lleven a cabo comunicaciones eficaces. Cuando cree un equipo de planificación de la transición, asigne a los miembros del equipo la tarea de trabajar con todos los líderes de la organización para definir y administrar unas comunicaciones eficaces en todos los niveles.

Ejemplo de cliente

AnyCompany Retail se suscribió a AWS Enterprise Support y utiliza otros proveedores externos para sus operaciones en la nube. La empresa utiliza el chat y ChatOps como su principal medio de comunicación para las actividades operativas. Las alertas y otras informaciones se incluyen en canales específicos. Cuando alguien debe actuar, expone claramente el resultado deseado y, en muchos casos, recibe un manual de procedimientos o de estrategias para que los utilice. Programa los cambios importantes en los sistemas de producción con un calendario de cambios.

Pasos para la implementación

1. Establezca un equipo central dentro de la organización que tenga la responsabilidad de crear e iniciar planes de comunicación para los cambios que se produzcan en varios niveles de la organización.
2. Establezca una propiedad de un solo enfoque para la supervisión. Ofrezca a los equipos individuales la capacidad de innovar de forma independiente y equilibre el uso de mecanismos uniformes, lo que permite conseguir el nivel adecuado de inspección y visión direccional.
3. Colabore con las partes interesadas de toda su organización para acordar unos estándares, prácticas y planes de comunicación.
4. Verifique que el equipo central de comunicaciones colabore con los líderes de la organización y del programa para redactar mensajes para el personal apropiado en nombre de los líderes.
5. Cree mecanismos de comunicación estratégicos para administrar el cambio mediante anuncios, calendarios compartidos, reuniones generales y reuniones presenciales o individuales para que los miembros del equipo tengan expectativas adecuadas sobre las medidas que deben tomar.
6. Proporcione el contexto, los detalles y el tiempo necesarios (cuando sea posible) para determinar si es necesario tomar medidas. Cuando sea necesario tomar medidas, indique qué medida se debe tomar y cuál es su repercusión.
7. Implemente herramientas que faciliten las comunicaciones tácticas, como el chat interno, el correo electrónico y la administración del conocimiento.

8. Implemente mecanismos para medir y verificar que todas las comunicaciones conduzcan a los resultados deseados.
9. Establezca un bucle de retroalimentación que mida la eficacia de todas las comunicaciones, especialmente cuando estén relacionadas con la resistencia a los cambios en toda la organización.
10. Para todas las Cuentas de AWS, establezca contactos alternativos para la facturación, la seguridad y las operaciones. Lo ideal sería que cada contacto fuera una cuenta de distribución de correo electrónico en lugar de un contacto individual específico.
11. Establezca un plan de comunicación de derivaciones y derivaciones inversas para interactuar con los equipos internos y externos, incluidos los de AWS Support y otros proveedores externos.
12. Inicie y ejecute estrategias de comunicación de manera uniforme durante toda la vida de cada programa de transformación.
13. Priorice las acciones que se puedan repetir siempre que sea posible para automatizarlas de forma segura a escala.
14. Cuando se requieren comunicaciones en escenarios con acciones automatizadas, el propósito de la comunicación debe ser informar a los equipos para llevar a cabo auditorías o como parte del proceso de administración de cambios.
15. Analice las comunicaciones de sus sistemas de alertas para detectar falsos positivos o alertas que se generan constantemente. Elimine o cambie estas alertas para que se inicien cuando sea necesaria una intervención humana. Si se inicia una alerta, proporcione un manual de procedimientos o una guía de estrategias.
    a. Puede utilizar los documentos de AWS Systems Manager para crear manuales de procedimientos y de estrategias para las alertas.
16. Existen mecanismos para notificar sobre los riesgos o los eventos previstos de forma clara y procesable, con suficiente antelación para poder responder de forma adecuada. Utilice listas de correo electrónico o canales de chat para enviar notificaciones antes de los eventos previstos.
    a. AWS Chatbot puede utilizarse para enviar alertas y responder a eventos desde la plataforma de mensajería de su organización.
17. Proporcione una fuente de información accesible en la que se puedan consultar los actos programados. Proporcione notificaciones de eventos planificados desde el mismo sistema.
    a. El Calendario de cambios de AWS Systems Manager se puede utilizar para crear periodos en los que se pueden producir cambios. De este modo, se avisa a los miembros del equipo de que pueden hacer cambios con seguridad.

18. Supervise las notificaciones de vulnerabilidad y la información sobre revisiones para comprender las vulnerabilidades existentes y los riesgos potenciales asociados a los componentes de su carga de trabajo. Notifique a los miembros del equipo para que puedan actuar.
    a. Puede suscribirse a los boletines de seguridad de AWS para recibir notificaciones sobre vulnerabilidades en AWS.
19. Búsqueda de opiniones y perspectivas diversas: fomente las contribuciones de todos. Ofrezca oportunidades de comunicación a los grupos que tienen menos representación. Rote los roles y las responsabilidades en las reuniones.
    a. Ampliación de los roles y las responsabilidades: ofrezca a los miembros del equipo la oportunidad de asumir roles que de otro modo no podrían desempeñar. Ganarán experiencia y perspectiva gracias al rol y a las interacciones con nuevos miembros del equipo con los que, de otro modo, no podrían interactuar. También aportarán su experiencia y perspectiva al nuevo rol y a los miembros del equipo con los que interactúen. A medida que aumente la perspectiva, identifique oportunidades de negocio emergentes o nuevas oportunidades de mejora. Rote entre los miembros de un equipo las tareas comunes que suelen llevar a cabo los demás para que sepan cuáles son sus exigencias y su repercusión.
    b. Disposición de un entorno seguro y acogedor: establezca políticas y controles que protejan la seguridad mental y física de los miembros del equipo de su organización. Los miembros del equipo deben poder interactuar sin miedo a represalias. Si los miembros del equipo se sienten seguros y acogidos, es más probable que se comprometan y sean productivos. Cuanto más diversa sea su organización, mejor comprenderá a las personas a las que apoya, incluidos sus clientes. Cuando los miembros del equipo se sienten cómodos, se consideran libres para hablar y confían en que se los escuchará, por lo que es más probable que compartan ideas valiosas (por ejemplo, oportunidades de marketing, necesidades de accesibilidad, segmentos de mercado no atendidos y riesgos no reconocidos en el entorno).
    c. Interacción con los miembros del equipo para que participen plenamente: proporcione los recursos necesarios para que sus empleados participen plenamente en todas las actividades relacionadas con el trabajo. Los miembros del equipo que se enfrentan a retos diarios desarrollan competencias para trabajar en torno a ellos. Estas competencias que se han desarrollado de esta forma única pueden aportar un beneficio importante a su organización. Para ayudar a los miembros del equipo, lleve a cabo las adaptaciones necesarias para aumentar los beneficios que puede recibir de sus contribuciones.

## Recursos

Prácticas recomendadas relacionadas:

- OPS03-BP01 Respaldo del área ejecutiva
- OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos
- OPS07-BP04 Uso de manuales de estrategias para investigar problemas

Documentos relacionados:

- Entrada de blog de AWS: Accountability and empowerment are key to high-performing agile organizations
- AWS Executive Insights: Descubra cómo escalar la innovación, en lugar de la complejidad | Líderes de un solo enfoque
- Boletines de seguridad de AWS
- Open CVE
- Soporte App in Slack to Manage Support Cases
- Administración de los recursos de AWS en los canales de Slack con Amazon Q Developer en las aplicaciones de chat

Servicios relacionados:

- Amazon Q Developer en aplicaciones de chat
- AWS Calendario de cambios de Systems Manager
- AWS Documentos de Systems Manager

OPS03-BP05 Fomento de la experimentación

La experimentación es un catalizador para convertir nuevas ideas en productos y características. Acelera el aprendizaje y mantiene a los miembros del equipo interesados y comprometidos. Se anima a los miembros del equipo a experimentar con frecuencia para impulsar la innovación. Incluso cuando se produce un resultado no deseado, tiene valor saber lo que no hay que hacer. No se castiga a los miembros del equipo por experimentos hechos correctamente con resultados no deseados.

Resultado deseado:

- Su organización fomenta la experimentación para impulsar la innovación.
- Los experimentos se utilizan como una oportunidad de aprender.

Patrones comunes de uso no recomendados:

- Desea efectuar una prueba A/B, pero no existe ningún mecanismo para llevar a cabo el experimento. Implementa un cambio en la interfaz de usuario sin poder probarlo. El resultado es una experiencia negativa para el cliente.
- Su empresa solo tiene un entorno de prueba y producción. No existe un entorno de pruebas para experimentar con nuevas características o productos, por lo que deberá experimentar en el entorno de producción.

Beneficios de establecer esta práctica recomendada:

- La experimentación impulsa la innovación.
- Puede reaccionar más rápidamente a los comentarios de los usuarios mediante la experimentación.
- Su organización desarrolla una cultura de aprendizaje.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los experimentos se deben hacer de forma segura. Utilice múltiples entornos para experimentar sin poner en peligro los recursos de producción. Utilice las pruebas A/B y las marcas de características para probar experimentos. Proporcione a los miembros del equipo la posibilidad de hacer experimentos en un entorno de pruebas.

Ejemplo de cliente

AnyCompany Retail fomenta la experimentación. Los miembros del equipo pueden utilizar el 20 % de su semana laboral para experimentar o aprender nuevas tecnologías. Disponen de un entorno de pruebas en el que pueden innovar. Las pruebas A/B se utilizan para las nuevas características con el fin de validarlas con comentarios de usuarios reales.

Pasos para la implementación

1. Colabore con los directivos de su organización para respaldar la experimentación. Se debe animar a los miembros del equipo a llevar a cabo los experimentos de forma segura.
2. Proporcione a los miembros del equipo un entorno en el que puedan experimentar con seguridad. Deben tener acceso a un entorno similar al de producción.

   a. Puede usar una Cuenta de AWS independiente para crear un entorno de pruebas para la experimentación. AWS Control Tower puede utilizarse para aprovisionar estas cuentas.
3. Utilice marcas de características y pruebas A/B para experimentar con seguridad y recopilar los comentarios de los usuarios.
   a. AWS AppConfig Feature Flags ofrece la posibilidad de crear marcadores de características.
   b. Puede usar las versiones de AWS Lambda para implementar una nueva versión de una función para las pruebas beta.

Nivel de esfuerzo para el plan de implementación: alto. Proporcionar a los miembros del equipo un entorno en el que experimentar y una forma segura de llevar a cabo los experimentos puede requerir una inversión significativa. También es posible que deba modificar el código de la aplicación para utilizar las marcas de características o admitir pruebas A/B.

Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP02 Análisis después del incidente: aprender de los incidentes es un motor importante de la innovación, junto con la experimentación.
- OPS11-BP03 Implementación de bucles de retroalimentación: los circuitos de retroalimentación son una parte importante de la experimentación.

Documentos relacionados:

- An Inside Look at the Amazon Culture: Experimentation, Failure, and Customer Obsession
- Best practices for creating and managing sandbox accounts in AWS
- Create a Culture of Experimentation Enabled by the Cloud
- Enabling experimentation and innovation in the cloud at SulAmérica Seguros
- Experiment More, Fail Less
- Organizing Your AWS Environment Using Multiple Accounts - Sandbox OU
- Using AWS AppConfig Feature Flags

Videos relacionados:

- AWS On Air ft. Amazon CloudWatch Evidently | AWS Events

- AWS On Air San Fran Summit 2022 ft. AWS AppConfig Feature Flags integration with Jira
- AWS re:Invent 2022 - A deployment is not a release: Control your launches w/feature flags (BOA305-R)
- Programmatically Create an Cuenta de AWS with AWS Control Tower
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations

Ejemplos relacionados:

- Entorno de pruebas para innovación de AWS
- End-to-end Personalization 101 for E-Commerce

Servicios relacionados:

- Amazon CloudWatch Evidently
- AWS AppConfig
- AWS Control Tower

OPS03-BP06 Fomento para que los miembros del equipo mantengan y aumenten su conjunto de competencias

Los equipos deben aumentar el conjunto de competencias para adoptar nuevas tecnologías, así como para hacer cambios en la demanda y las responsabilidades en favor de sus cargas de trabajo. El aumento de las competencias en las nuevas tecnologías suele ser una fuente de satisfacción para los miembros del equipo y fomenta la innovación. Apoye a los miembros de su equipo para que obtengan y mantengan certificaciones del sector que validen y reconozcan sus competencias en constante crecimiento. Lleve a cabo una formación interdisciplinar para promover la transferencia de conocimientos y reducir el riesgo de que se produzca un impacto significativo cuando pierda a miembros del equipo cualificados y experimentados con conocimiento institucional. Ofrezca un tiempo estructurado dedicado al aprendizaje.

AWS proporciona recursos, como el Centro de recursos introductorios de AWS, los blogs de AWS, las charlas técnicas en línea de AWS, los eventos y seminarios web de AWS y los AWS Well-Architected Labs, que ofrecen orientación, ejemplos y tutoriales detallados para formar a sus equipos.

Recursos como Soporte, (AWS re:Post, el Centro de Soporte) y la documentación de AWS ayudan a eliminar los obstáculos técnicos y a mejorar las operaciones. Contacte con Soporte a través del Centro de Soporte para que le ayuden con sus preguntas.

AWS también comparte los patrones y prácticas recomendadas que hemos aprendido a través del funcionamiento de AWS en Amazon Builders' Library y una gran variedad de material educativo y útil a través del blog de AWS y The Official AWS Podcast.

Formación de AWS and Certification incluye formación gratuita a través de cursos digitales que puede llevar a cabo a su propio ritmo, además de planes de aprendizaje por rol o dominio. También puede inscribirse en una formación adicional impartida por un instructor para facilitar aún más el desarrollo de las competencias de AWS de sus equipos.

Resultado deseado: su organización evalúa constantemente las carencias de competencias y las soluciona con un presupuesto e inversiones estructurados. Los equipos alientan e incentivan a sus miembros con actividades de mejora de las competencias, como la adquisición de las principales certificaciones del sector. Los equipos aprovechan los programas dedicados al intercambio de conocimientos, como almuerzos de trabajo, jornadas de inmersión, encuentros de programadores y GameDays. Su organización mantiene sus sistemas de conocimiento actualizados y relevantes para dar a los miembros del equipo una formación cruzada, incluidas las formaciones de incorporación de nuevos empleados.

Patrones comunes de uso no recomendados:

- En ausencia de un programa de formación y un presupuesto estructurados, los equipos sienten incertidumbre cuando intentan mantenerse al día de la evolución de la tecnología, lo que se traduce en un aumento de las renuncias laborales.
- Como parte de la migración a AWS, demuestra la existencia de lagunas en las competencias y una fluidez variable en la nube entre los equipos. Si no se esfuerzan por mejorar sus competencias, los equipos se ven sobrecargados con tareas heredadas y una administración ineficaz del entorno de la nube, lo que provoca un aumento del trabajo de los operadores. Este agotamiento incrementa la insatisfacción de los empleados.

Beneficios de establecer esta práctica recomendada: cuando su organización invierte conscientemente en mejorar las competencias de sus equipos, también ayuda a acelerar y escalar la adopción y optimización de la nube. Los programas de aprendizaje específicos impulsan la innovación y fomentan la capacidad operativa para que los equipos estén preparados para hacer frente a cualquier acontecimiento. Los equipos invierten conscientemente en la implementación y

el desarrollo de las prácticas recomendadas. La moral del equipo es alta y los miembros del equipo valoran su contribución a la empresa.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para adoptar nuevas tecnologías, impulsar la innovación y seguir el ritmo de los cambios en la demanda y las responsabilidades para facilitar sus cargas de trabajo, invierta continuamente en el crecimiento profesional de sus equipos.

Pasos para la implementación

1. Uso de programas estructurados de fomento de la nube: AWS Skills Guild ofrece formación consultiva para aumentar la confianza en las competencias en la nube y fomentar una cultura de aprendizaje continuo.
2. Disposición de recursos para la formación: proporcione tiempo estructurado dedicado, acceso a materiales de formación y recursos de laboratorio, y respalde la participación en conferencias y acceso organizaciones profesionales que proporcionen oportunidades para aprender, tanto a los educadores como a los compañeros. Proporcione a los miembros de su equipo sin experiencia acceso a los miembros del equipo experimentados para que actúen como mentores, o permita que los miembros del equipo sin experiencia sigan de cerca el trabajo de los que tienen experiencia y se expongan a sus métodos y competencias. Fomente el aprendizaje de contenido no relacionado directamente con el trabajo para tener una perspectiva más amplia.
3. Fomento del uso de recursos técnicos expertos: aproveche recursos como AWS re:Post para acceder a conocimientos cuidadosamente seleccionados y a una comunidad dinámica.
4. Creación y mantenimiento de un repositorio de conocimientos actualizado: utilice plataformas para compartir conocimientos, como wikis y manuales de procedimientos. Cree su propia fuente de conocimiento experto reutilizable con AWS re:Post Private para agilizar la colaboración, mejorar la productividad y acelerar la incorporación de los nuevos empleados.
5. Formación de equipos e interacción entre equipos: planifique las necesidades de formación continua de los miembros de su equipo. Proporcione oportunidades para que los miembros del equipo se unan a otros equipos (temporal o permanentemente) para compartir competencias y prácticas recomendadas que beneficien a toda la organización.
6. Apoyo de la obtención y el mantenimiento de certificaciones del sector: apoye a los miembros de su equipo para que adquieran y mantengan certificaciones del sector que validen lo que han aprendido y reconozca sus logros.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- OPS03-BP01 Respaldo del área ejecutiva
- OPS11-BP04 Administración de conocimientos

Documentos relacionados:

- Documento técnico de AWS: Cloud Adoption Framework: People Perspective
- Investing in continuous learning to grow your organization's future
- AWS Skills Guild
- Formación de AWS and Certification
- Soporte
- AWS re:Post
- AWSCentro de recursos introductorios
- Blogs de AWS
- Nube de AWS Cumplimiento de
- AWS documentación
- The Official AWS Podcast.
- AWS Charlas de tecnología en línea de
- Eventos y seminarios web de AWS
- AWS Well-Architected Labs
- Amazon Builders' Library

Videos relacionados:

- AWS re:Invent 2023 | Reskilling at the speed of cloud: Turning employees into entrepreneurs
- AWS re:Invent 2023: Building a culture of curiosity through gamification

OPS03-BP07 Recursos adecuados para los equipos

Proporcione la cantidad adecuada de miembros competentes en el equipo y facilite las herramientas y los recursos necesarios para satisfacer sus necesidades de carga de trabajo. Sobrecargar a los miembros del equipo aumenta el riesgo de que se produzcan errores humanos. Las inversiones en herramientas y recursos, como la automatización, pueden aumentar la eficacia de su equipo y ayudarlo a soportar una mayor cantidad de cargas de trabajo sin la necesidad de capacidad adicional.

Resultado deseado:

- Ha dotado a su equipo del personal adecuado para que adquiera las competencias necesarias para administrar las cargas de trabajo en AWS de acuerdo con su plan de migración. A medida que su equipo se ha ido ampliando en el transcurso de su proyecto de migración, ha adquirido competencias en las tecnologías básicas de AWS que la empresa tiene previsto utilizar al migrar o modernizar sus aplicaciones.
- Ha alineado cuidadosamente su plan de dotación de personal para hacer un uso eficiente de los recursos aprovechando la automatización y el flujo de trabajo. Un equipo más pequeño puede administrar ahora más infraestructura en nombre de los equipos de desarrollo de aplicaciones.
- Dado que las prioridades operativas cambian, cualquier limitación de recursos de personal se identifica de manera proactiva para ayudar a que las iniciativas empresariales salgan adelante.
- Las métricas operativas que indican el esfuerzo operativo (como la fatiga del personal de guardia o un uso excesivo de localizadores) se revisan para verificar que el personal no esté sobrecargado.

Patrones comunes de uso no recomendados:

- Su personal no ha reforzado sus conocimientos de AWS a medida que se acerca a su plan plurianual de migración a la nube, lo que pone en riesgo que se atiendan las cargas de trabajo y reduce la moral de los empleados.
- Toda su organización de TI está adoptando formas de trabajo ágiles. La empresa está priorizando la cartera de productos y estableciendo métricas para las características que deben desarrollarse primero. El proceso ágil no requiere que los equipos asignen puntos escalonados a sus planes de trabajo. Como resultado, es imposible saber el nivel de capacidad que se necesita para la próxima cantidad de trabajo, o si tiene asignadas las competencias adecuadas al trabajo.
- Ha encargado a un socio de AWS la migración de sus cargas de trabajo y no dispone de un plan de transición de la asistencia para sus equipos una vez que el socio finalice el proyecto de

migración. Sus equipos se esfuerzan por atender las cargas de trabajo de manera eficiente y eficaz.

Beneficios de establecer esta práctica recomendada: cuenta con miembros del equipo que tienen las competencias adecuadas en su organización para atender las cargas de trabajo. La asignación de recursos puede adaptarse a las prioridades cambiantes sin afectar al rendimiento. El resultado es que los equipos son capaces de atender las cargas de trabajo y, al mismo tiempo, maximizar el tiempo para centrarse en la innovación para los clientes, lo que a su vez aumenta la satisfacción de los empleados.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La planificación de los recursos para la migración a la nube debe hacerse en un nivel de la organización que se ajuste a su plan de migración, así como al modelo operativo deseado que se está implementando para respaldar su nuevo entorno de nube. Esto debe incluir saber qué tecnologías de la nube se implementan para los equipos de negocio y desarrollo de aplicaciones. Los líderes de la infraestructura y las operaciones deben planificar el análisis de las carencias de competencias, la formación y la definición de roles para los ingenieros que lideran la adopción de la nube.

Pasos para la implementación

1. Defina criterios para el éxito del equipo con métricas operativas relevantes, como la productividad del personal (por ejemplo, el costo de atender una carga de trabajo o las horas empleadas por los operadores durante los incidentes).
2. Defina mecanismos de planificación e inspección de la capacidad de los recursos para verificar que haya un equilibrio adecuado de capacidad cualificada cuando sea necesario y se pueda ajustar con el tiempo.
3. Cree mecanismos (por ejemplo, enviar una encuesta mensual a los equipos) para conocer los retos relacionados con el trabajo que afectan a los equipos (como el aumento de responsabilidades, los cambios en la tecnología, la pérdida de personal o el incremento de clientes a los que se presta asistencia).
4. Utilice estos mecanismos para interactuar con los equipos y detectar tendencias que puedan agravar los desafíos de productividad de los empleados. Cuando sus equipos se vean afectados por factores externos, vuelva a evaluar los objetivos y ajústelos según convenga. Identifique los obstáculos que impiden que el equipo avance.

5. Revise periódicamente si los recursos de los que dispone en la actualidad siguen siendo suficientes, o si se necesitan recursos adicionales, y haga los ajustes oportunos para apoyar a los equipos.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS03-BP06 Fomento para que los miembros del equipo mantengan y aumenten su conjunto de competencias
- OPS09-BP03 Revisión de las métricas de las operaciones y priorización de las mejoras
- OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas
- OPS10-BP07 Automatización de las respuestas a eventos

Documentos relacionados:

- Nube de AWS Adoption Framework: People Perspective
- Becoming a Future-Ready Enterprise
- Prioritize your Employees' Skills to Drive Business Growth
- Estimulación de la innovación y la velocidad con los equipos de dos pizzas de Amazon
- How Cloud-Mature Enterprises Succeed

## Prepare

Preguntas

- OPS 4. ¿Cómo implementa la observabilidad en su carga de trabajo?
- OPS 5. ¿Cómo reduce los defectos, facilita la reparación y mejora el flujo en producción?
- OPS 6. ¿Cómo mitiga los riesgos de implementación?
- OPS 7. ¿Cómo sabe que está listo para admitir una carga de trabajo?

## OPS 4. ¿Cómo implementa la observabilidad en su carga de trabajo?

Implemente la observabilidad en su carga de trabajo para que pueda comprender su estado y tomar decisiones basadas en datos en función de los requisitos empresariales.

Prácticas recomendadas

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS04-BP03 Implementación de telemetría de la experiencia del usuario
- OPS04-BP04 Implementación de telemetría de dependencias
- OPS04-BP05 Implementación de rastreo distribuido

### OPS04-BP01 Identificación de los indicadores clave de rendimiento

La implementación de la observabilidad en su carga de trabajo comienza con la comprensión de su estado y la toma de decisiones basadas en datos en función de los requisitos empresariales. Una de las formas más eficaces de garantizar la alineación entre las actividades de supervisión y los objetivos empresariales consiste en definir y supervisar los indicadores clave de rendimiento (KPI).

Resultado deseado: prácticas de observabilidad eficientes que están estrechamente alineadas con los objetivos empresariales, lo que garantiza que los esfuerzos de supervisión siempre estén al servicio de resultados comerciales tangibles.

Patrones comunes de uso no recomendados:

- Indicadores clave de rendimiento indefinidos: trabajar sin indicadores clave de rendimiento claros puede llevar a una supervisión excesiva o insuficiente y a la pérdida de señales vitales.
- KPI estáticos: no se revisitan ni refinan los KPI a medida que evolucionan la carga de trabajo o los objetivos empresariales.
- Desalineación: centrarse en las métricas técnicas que no se correlacionan directamente con los resultados empresariales o que son más difíciles de correlacionar con problemas de la vida real.

Beneficios de establecer esta práctica recomendada:

- Facilidad de identificación de problemas: los KPI empresariales suelen mostrar los problemas con más claridad que las métricas técnicas. Una caída en un KPI empresarial puede identificar un problema de forma más eficaz que analizar numerosas métricas técnicas.

- Alineación empresarial: garantiza que las actividades de supervisión respalden directamente los objetivos empresariales.
- Eficiencia: priorice los recursos de supervisión y preste atención a las métricas que importan.
- Proactividad: detecte y aborde los problemas antes de que tengan implicaciones comerciales más amplias.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para definir de forma eficaz los KPI de la carga de trabajo:

1. Inicio con los resultados empresariales: antes de profundizar en las métricas, comprenda los resultados empresariales deseados. ¿Se trata de un aumento de las ventas, una mayor participación de los usuarios o unos tiempos de respuesta más rápidos?
2. Correlación de las métricas técnicas con los objetivos empresariales: no todas las métricas técnicas tienen un impacto directo en los resultados empresariales. Identifique los que sí lo tienen, pero a menudo es más sencillo identificar un problema mediante un KPI empresarial.
3. Uso de Amazon CloudWatch: utilice CloudWatch para definir y supervisar las métricas que representan sus KPI.
4. Revisión y actualización de los KPI con regularidad: a medida que su carga de trabajo y su empresa evolucionen, mantenga la relevancia de sus KPI.
5. Implicación de las partes interesadas: involucre a los equipos técnicos y empresariales en la definición y revisión de los KPI.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- the section called “OPS04-BP02 Implementación de telemetría de aplicaciones”
- the section called “OPS04-BP03 Implementación de telemetría de la experiencia del usuario”
- the section called “OPS04-BP04 Implementación de telemetría de dependencias”
- the section called “OPS04-BP05 Implementación de rastreo distribuido”

Documentos relacionados:

- AWS Observability Best Practices
- CloudWatch User Guide
- AWS Observability Skill Builder Course

Videos relacionados:

- Developing an observability strategy

Ejemplos relacionados:

- One Observability Workshop

OPS04-BP02 Implementación de telemetría de aplicaciones

La telemetría de aplicaciones sirve de base de la observabilidad de su carga de trabajo. Es crucial emitir telemetría que ofrezca información procesable sobre el estado de la aplicación y el logro de los resultados técnicos y empresariales. Desde la solución de problemas hasta la medición del impacto de una nueva característica o la garantía de la alineación con los indicadores clave de rendimiento (KPI) de la empresa, la telemetría de las aplicaciones informa sobre la forma de crear, operar y hacer evolucionar su carga de trabajo.

Las métricas, los registros y los rastreos forman los tres pilares principales de la observabilidad. Sirven como herramientas de diagnóstico que describen el estado de la aplicación. Con el tiempo, ayudan a crear puntos de referencia e identificar anomalías. Sin embargo, para garantizar la alineación entre las actividades de supervisión y los objetivos empresariales, es fundamental definir y supervisar los KPI. Los KPI empresariales suelen facilitar la identificación de los problemas en comparación con las métricas técnicas únicamente.

Otros tipos de telemetría, como la supervisión de usuarios reales (RUM) y las transacciones sintéticas, complementan estos orígenes de datos principales. La RUM ofrece información sobre las interacciones de los usuarios en tiempo real, mientras que las transacciones sintéticas simulan los posibles comportamientos de los usuarios, lo que ayuda a detectar los cuellos de botella antes de que los usuarios reales los encuentren.

Resultado deseado: obtenga información útil sobre el rendimiento de su carga de trabajo. Estos conocimientos le permiten tomar decisiones proactivas sobre la optimización del rendimiento, lograr

una mayor estabilidad de la carga de trabajo, optimizar los procesos de CI/CD y utilizar los recursos de manera eficaz.

Patrones comunes de uso no recomendados:

- Observabilidad incompleta: no incorporar la observabilidad en todos los niveles de la carga de trabajo, lo que resulta en puntos ciegos que pueden ocultar información vital sobre el rendimiento y el comportamiento del sistema.
- Vista de datos fragmentada: cuando los datos están dispersos en varias herramientas y sistemas, resulta difícil mantener una visión integral del estado y el rendimiento de la carga de trabajo.
- Problemas informados por los usuarios: una señal de que falta una detección proactiva de los problemas mediante la telemetría y la supervisión de los KPI empresariales.

Beneficios de establecer esta práctica recomendada:

- Toma de decisiones informadas: con la información de la telemetría y los KPI empresariales, puede tomar decisiones basadas en datos.
- Mejora de la eficiencia operativa: el uso de los recursos basada en datos conduce a la rentabilidad.
- Mejora de la estabilidad de la carga de trabajo: detección y resolución de problemas más rápidas, lo que mejora el tiempo de actividad.
- Procesos de CI/CD simplificados: la información obtenida de los datos de telemetría facilita el refinamiento de los procesos y la entrega fiable de código.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para implementar la telemetría de aplicaciones para su carga de trabajo, utilice servicios de AWS como Amazon CloudWatch y AWS X-Ray. Amazon CloudWatch proporciona un conjunto completo de herramientas de supervisión que le permiten observar sus recursos y aplicaciones en entornos en las instalaciones y de AWS. Recopila, sigue y analiza las métricas, consolida y supervisa los datos de registro y responde a los cambios en los recursos, lo que mejora su comprensión de cómo funciona su carga de trabajo. Al mismo tiempo, AWS X-Ray le permite rastrear, analizar y depurar sus aplicaciones, lo que le proporciona una comprensión profunda del comportamiento de su carga de trabajo. Con características como los mapas de servicios, las distribuciones de latencia y la cronología de rastreo, AWS X-Ray proporciona información sobre el rendimiento de su carga de trabajo y los cuellos de botella que le afectan.

Pasos para la implementación

1. Identificación de los datos que hay que recopilar: determine las métricas, los registros y los rastreos esenciales que podrían ofrecer información sustancial sobre el estado, el rendimiento y el comportamiento de su carga de trabajo.
2. Implementación del agente de CloudWatch: el agente de CloudWatch es fundamental a la hora de obtener métricas y registros del sistema y las aplicaciones de su carga de trabajo y su infraestructura subyacente. El agente de CloudWatch también se puede utilizar para recopilar rastreos de X-Ray o OpenTelemetry y enviarlos a X-Ray.
3. Implementación de la detección de anomalías para los registros y las métricas: utilice la detección de anomalías de los Registros de CloudWatch y la detección de anomalías de métricas de CloudWatch para identificar automáticamente las actividades inusuales en las operaciones de su aplicación. Estas herramientas utilizan algoritmos de machine learning para detectar anomalías y alertar sobre ellas, lo que mejora las capacidades de supervisión y acelera el tiempo de respuesta ante posibles interrupciones o amenazas de seguridad. Configure estas características para administrar de forma proactiva el estado y la seguridad de las aplicaciones.
4. Protección de los datos de registro confidenciales: utilice la protección de datos de los Registros de Amazon CloudWatch para ocultar la información confidencial de sus registros. Esta característica ayuda a mantener la privacidad y el cumplimiento mediante la detección automática y el enmascaramiento de los datos confidenciales antes de que se acceda a ellos. Implemente el enmascaramiento de datos para gestionar y proteger de forma segura los datos confidenciales, como la información de identificación personal (PII).
5. Definición y supervisión de los KPI empresariales: establezca métricas personalizadas que se ajusten a los resultados empresariales.
6. Instrumentación de su aplicación con AWS X-Ray: además de implementar el agente de CloudWatch, es fundamental instrumentar su aplicación para que emita datos de rastreo. Este proceso puede proporcionar más información sobre el comportamiento y el rendimiento de su carga de trabajo.
7. Estandarización de la recopilación de datos en toda su aplicación: estandarice las prácticas de recopilación de datos en toda la aplicación. La uniformidad ayuda a correlacionar y analizar los datos y proporciona una vista completa del comportamiento de la aplicación.
8. Implementación de la observabilidad entre cuentas: mejore la eficiencia de la supervisión entre Cuentas de AWS con la observabilidad entre cuentas de Amazon CloudWatch. Con esta característica, puede consolidar las métricas, los registros y las alarmas de diferentes cuentas

en una sola vista, lo que simplifica la administración y mejora los tiempos de respuesta para los problemas identificados en el entorno de AWS de su organización.

9. Análisis de los datos y actuación en consecuencia: una vez que la recopilación y la normalización de los datos estén en marcha, utilice Amazon CloudWatch para llevar a cabo el análisis de métricas y registros, y AWS X-Ray para el análisis de rastreos. Este análisis puede proporcionar información crucial sobre el estado, el rendimiento y el comportamiento de su carga de trabajo, lo que guiará su proceso de toma de decisiones.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Definición de los KPI de la carga de trabajo
- OPS04-BP03 Implementación de telemetría de actividades de usuario
- OPS04-BP04 Implementación de telemetría de dependencias
- OPS04-BP05 Implementación de rastreo distribuido

Documentos relacionados:

- AWS Observability Best Practices
- CloudWatch User Guide
- Guía para desarrolladores de AWS X-Ray
- Instrumentación de los sistemas distribuidos para obtener visibilidad operativa
- AWS Observability Skill Builder Course
- Novedades de Amazon CloudWatch
- Novedades de AWS X-Ray

Videos relacionados:

- AWS re:Invent 2022 - Observability best practices at Amazon
- AWS re:Invent 2022 - Developing an observability strategy

Ejemplos relacionados:

- One Observability Workshop
- Biblioteca de soluciones de AWS: Supervisión de aplicaciones con Amazon CloudWatch

OPS04-BP03 Implementación de telemetría de la experiencia del usuario

Es crucial obtener información detallada sobre las experiencias de los clientes y las interacciones con su aplicación. La supervisión de usuarios reales (RUM) y las transacciones sintéticas sirven como herramientas poderosas para este propósito. La RUM proporciona datos sobre las interacciones reales de los usuarios, lo que ofrece una perspectiva sin filtrar de la satisfacción del usuario, mientras que las transacciones sintéticas simulan las interacciones de los usuarios, lo que ayuda a detectar posibles problemas incluso antes de que afecten a los usuarios reales.

Resultado deseado: una visión integral de la experiencia del cliente, detección proactiva de problemas y optimización de las interacciones de los usuarios para ofrecer experiencias digitales fluidas.

Patrones comunes de uso no recomendados:

- Aplicaciones sin supervisión de usuarios reales (RUM):
  - Retraso en la detección de problemas: sin RUM, es posible que no se dé cuenta de los cuellos de botella o problemas de rendimiento hasta que los usuarios se quejen. Este enfoque reactivo puede provocar la insatisfacción de los clientes.
  - Falta de información sobre la experiencia del usuario: no usar RUM significa perder datos cruciales que muestran cómo los usuarios reales interactúan con su aplicación, lo que limita su capacidad de optimizar la experiencia del usuario.
- Aplicaciones sin transacciones sintéticas:
  - Omisión de casos de periferia: las transacciones sintéticas le ayudan a probar rutas y funciones que los usuarios habituales no suelen utilizar con frecuencia, pero que son fundamentales para determinadas funciones empresariales. Sin ellos, estas rutas podrían funcionar mal y el problema podría pasar desapercibido.
  - Comprobación de problemas cuando no se utiliza la aplicación: las pruebas sintéticas periódicas pueden simular momentos en los que los usuarios reales no interactúan activamente con la aplicación, lo que garantiza que el sistema siempre funcione correctamente.

Beneficios de establecer esta práctica recomendada:

- Detección proactiva de problemas: identifique y aborde los posibles problemas antes de que afecten a los usuarios reales.
- Experiencia de usuario optimizada: los comentarios continuos de la RUM ayudan a refinar y mejorar la experiencia general del usuario.
- Información sobre el rendimiento de los dispositivos y navegadores: comprenda el rendimiento de su aplicación en varios dispositivos y navegadores, lo que permitirá una mayor optimización.
- Flujos de trabajo empresariales validados: las transacciones sintéticas periódicas garantizan que las funcionalidades básicas y las rutas cruciales permanezcan operativas y eficientes.
- Mejora del rendimiento de las aplicaciones: utilice la información recopilada a partir de datos de usuarios reales para mejorar la capacidad de respuesta y la fiabilidad de las aplicaciones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para aprovechar RUM y las transacciones sintéticas para la telemetría de la actividad del usuario, AWS ofrece servicios como Amazon CloudWatch RUM y Amazon CloudWatch Synthetics. Las métricas, los registros y los rastreos, junto con los datos de actividad de los usuarios, proporcionan una vista completa tanto del estado operativo de la aplicación como de la experiencia del usuario.

Pasos para la implementación

1. Implementación de Amazon CloudWatch RUM: integre su aplicación con CloudWatch RUM para recopilar, analizar y presentar datos de usuarios reales.
   a. Utilice la biblioteca de JavaScript de CloudWatch RUM para integrar la RUM con su aplicación.
   b. Configure paneles para visualizar y supervisar los datos de los usuarios reales.
2. Configuración de CloudWatch Synthetics: cree canarios, o rutinas con scripts, que simulen las interacciones de los usuarios con su aplicación.
   a. Defina los flujos de trabajo y las rutas de las aplicaciones fundamentales.
   b. Diseñe canarios controlados por scripts de CloudWatch Synthetics para simular las interacciones de los usuarios en estas rutas.
   c. Programe y supervise los canarios para que se ejecuten a intervalos específicos, lo que garantiza controles de rendimiento coherentes.
3. Análisis y acción en consecuencia: utilice los datos de la RUM y las transacciones sintéticas para obtener información y tomar medidas correctivas cuando se detecten anomalías. Utilice paneles y alarmas de CloudWatch para mantenerse informado.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS04-BP04 Implementación de telemetría de dependencias
- OPS04-BP05 Implementación de rastreo distribuido

Documentos relacionados:

- Guía de Amazon CloudWatch RUM
- Guía de Amazon CloudWatch Synthetics

Videos relacionados:

- Optimize applications through end user insights with Amazon CloudWatch RUM
- AWS on Air ft. Real-User Monitoring for Amazon CloudWatch

Ejemplos relacionados:

- One Observability Workshop
- Git Repository for Amazon CloudWatch RUM Web Client
- Using Amazon CloudWatch Synthetics to measure page load time

OPS04-BP04 Implementación de telemetría de dependencias

La telemetría de dependencias es esencial para supervisar el estado y el rendimiento de los servicios y componentes externos de los que depende su carga de trabajo. Proporciona información valiosa sobre la accesibilidad, los tiempos de espera y otros eventos cruciales relacionados con dependencias como DNS, bases de datos o API de terceros. Al instrumentar su aplicación para que emita métricas, registros y rastreos sobre estas dependencias, entenderá más claramente cuáles son los posibles cuellos de botella, problemas de rendimiento o errores que podrían afectar a su carga de trabajo.

Resultado deseado: asegúrese de que las dependencias en las que se basa su carga de trabajo funcionan según lo previsto, lo que le permitirá abordar los problemas de forma proactiva y garantizar un rendimiento óptimo de la carga de trabajo.

Patrones comunes de uso no recomendados:

- Omisión de las dependencias externas: centrarse únicamente en las métricas internas de las aplicaciones y descuidar las métricas relacionadas con las dependencias externas.
- Falta de supervisión proactiva: esperar a que surjan problemas en lugar de supervisar continuamente el estado y el rendimiento de la dependencia.
- Supervisión en silos: uso de numerosas herramientas de supervisión dispares que pueden generar vistas fragmentadas e incoherentes del estado de la dependencia.

Beneficios de establecer esta práctica recomendada:

- Mejora de la fiabilidad de la carga de trabajo: al garantizar que las dependencias externas estén siempre disponibles y funcionen de manera óptima.
- Detección y resolución de problemas más rápidas: identificar y abordar de forma proactiva los problemas relacionados con las dependencias antes de que afecten a la carga de trabajo.
- Panorámica completa: obtener una visión integral de los componentes internos y externos que influyen en el estado de la carga de trabajo.
- Mejora de la escalabilidad de la carga de trabajo: mediante la comprensión de los límites de escalabilidad y las características de rendimiento de las dependencias externas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para implementar la telemetría de dependencias, empiece por identificar los servicios, la infraestructura y los procesos de los que depende su carga de trabajo. Cuantifique qué aspecto tienen las buenas condiciones cuando esas dependencias funcionan según lo esperado y, a continuación, determine qué datos se necesitan para medirlas. Con esa información, puede crear paneles y alertas que proporcionen información a sus equipos de operaciones sobre el estado de esas dependencias. Use herramientas de AWS para detectar y cuantificar el efecto cuando las dependencias no pueden satisfacer las necesidades. Revisite su estrategia para que tenga en cuenta los cambios en las prioridades, los objetivos y los conocimientos adquiridos.

Pasos para la implementación

Para implementar la telemetría de dependencias de manera eficaz:

1. Identificación de las dependencias externas: colabore con las partes interesadas para identificar las dependencias externas de las que depende su carga de trabajo. Las dependencias externas pueden abarcar servicios como bases de datos externas, API de terceros, rutas de conectividad de red a otros entornos y servicios de DNS. El primer paso para lograr una telemetría de dependencias eficaz es comprender a la perfección cuáles son esas dependencias.
2. Desarrollo de una estrategia de supervisión: una vez que tenga una idea clara de sus dependencias externas, diseñe una estrategia de supervisión adaptada a ellas. Esto implica comprender la importancia de cada dependencia, su comportamiento esperado y cualquier acuerdo u objetivo de nivel de servicio (SLA o SLT) asociado. Configure alertas proactivas que le notifiquen los cambios de estado o las desviaciones del rendimiento.
3. Uso de la supervisión de la red: utilice Internet Monitor y Network Monitor, que proporcionan información completa sobre las condiciones globales de Internet y la red. Estas herramientas le ayudan a conocer los cortes, interrupciones o degradaciones del rendimiento que afectan a sus dependencias externas y responder a ellos.
4. Manténgase informado con AWS Health: AWS Health es la fuente autorizada de información sobre el estado de los recursos de Nube de AWS. Use AWS Health para visualizar y recibir notificaciones sobre cualquier evento de servicio actual y cambios futuros, como los eventos del ciclo de vida planificados, de modo que pueda tomar medidas para mitigar los impactos.
    a. Cree notificaciones de eventos de AWS Health adaptadas al propósito para el correo electrónico y los canales de chat a través de AWS User Notifications e intégrelas mediante programación con las herramientas de supervisión y alerta a través de Amazon EventBridge o la API de AWS Health.
    b. Planifique y realice un seguimiento del progreso de los eventos de estado que requieran una acción mediante la integración con herramientas de administración de cambios o ITSM (como Jira o ServiceNow) que ya pueda utilizar a través de Amazon EventBridge o la API de AWS Health.
    c. Si usa AWS Organizations, habilite la vista de organización para AWS Health a fin de agregar eventos de AWS Health en todas las cuentas.
5. Instrumentación de su aplicación con AWS X-Ray: AWS X-Ray proporciona información sobre el rendimiento de las aplicaciones y sus dependencias subyacentes. Al rastrear las solicitudes de principio a fin, puede identificar cuellos de botella o errores en los servicios o componentes externos en los que se basa su aplicación.

6. Uso de Amazon DevOps Guru: este servicio basado en machine learning identifica problemas operativos, predice cuándo pueden producirse problemas críticos y recomienda medidas concretas. Tiene un valor incalculable para obtener información sobre las dependencias y determinar que no son el origen de los problemas operativos.
7. Supervisión periódica: supervise continuamente las métricas y los registros relacionados con las dependencias externas. Configure alertas en caso de que se produzca un comportamiento inesperado o una degradación del rendimiento.
8. Validación después de los cambios: siempre que se produzca una actualización o un cambio en alguna de las dependencias externas, valide su rendimiento y compruebe su conformidad con los requisitos de la aplicación.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Definición de los KPI de la carga de trabajo
- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS04-BP03 Implementación de telemetría de actividades de usuario
- OPS04-BP05 Implementación de rastreo distribuido
- OP08-BP04 Creación de alertas procesables

Documentos relacionados:

- Amazon Personal Panel de estado User Guide
- AWS Internet Monitor User Guide
- Guía para desarrolladores de AWS X-Ray
- AWS DevOps Guru User Guide

Videos relacionados:

- Visibility into how internet issues impact app performance
- Introduction to Amazon DevOps Guru
- Manage resource lifecycle events at scale with AWS Health

Ejemplos relacionados:

- AWS Health Aware
- Using Tag-Based Filtering to Manage AWS Health Monitoring and Alerting at Scale

OPS04-BP05 Implementación de rastreo distribuido

El rastreo distribuido ofrece una forma de supervisar y visualizar las solicitudes a medida que atraviesan varios componentes de un sistema distribuido. Al obtener datos de rastreo de numerosos orígenes y analizarlos en una vista unificada, los equipos pueden comprender mejor cómo fluyen las solicitudes, dónde existen los cuellos de botella y dónde deben centrarse los esfuerzos de optimización.

Resultado deseado: obtenga una visión integral de las solicitudes que fluyen por su sistema distribuido, lo que permite una depuración precisa, un rendimiento optimizado y una mejor experiencia del usuario.

Patrones comunes de uso no recomendados:

- Instrumentación incoherente: no todos los servicios de un sistema distribuido están instrumentados para el rastreo.
- Hacer caso omiso de la latencia: centrarse únicamente en los errores y no tener en cuenta la latencia o las degradaciones graduales del rendimiento.

Beneficios de establecer esta práctica recomendada:

- Información general completa del sistema: visualización de toda la ruta de las solicitudes, desde la entrada hasta la salida.
- Depuración mejorada: identificación rápida de dónde se producen errores o problemas de rendimiento.
- Mejora de la experiencia del usuario: supervisión y optimización en función de los datos reales del usuario, lo que garantiza que el sistema satisfaga las demandas de la vida real.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Comience por identificar todos los elementos de la carga de trabajo que requieren instrumentación. Una vez contabilizados todos los componentes, utilice herramientas como AWS X-Ray y OpenTelemetry para recopilar datos y analizarlos con herramientas como X-Ray y Amazon CloudWatch ServiceLens Map. Lleve a cabo revisiones periódicas con los desarrolladores y complemente estas conversaciones con herramientas como Amazon DevOps Guru, X-Ray Analytics y X-Ray Insights para sacar a la luz resultados más profundos. Establezca alertas a partir de los datos de rastreo para notificar cuando los resultados, tal como se definen en el plan de supervisión de la carga de trabajo, estén en peligro.

Pasos para la implementación

Implementación del rastreo distribuido de manera eficaz:

1. Incorporación de AWS X-Ray: integre X-Ray en su aplicación para obtener información sobre su comportamiento, comprender su rendimiento e identificar los cuellos de botella. Utilice X-Ray Insights para el análisis automático de rastreos.
2. Instrumentación de sus servicios: compruebe que todos los servicios, desde una función de AWS Lambda a una instancia de EC2, envíen datos de rastreo. Cuantos más servicios instrumente, más clara será la vista de principio a fin.
3. Integración del seguimiento de CloudWatch RUM y Synthetic Monitoring: integre el CloudWatch RUM y el Synthetic Monitoring con X-Ray. Esto permite recoger experiencias de usuario de la vida real y simular las interacciones de los usuarios para identificar posibles problemas.
4. Uso del agente de CloudWatch: el agente puede enviar rastreos tanto de X-Ray como de OpenTelemetry, lo que mejora la profundidad de la información obtenida.
5. Uso de Amazon DevOps Guru: DevOps Guru utiliza datos de X-Ray, CloudWatch, AWS Config y AWS CloudTrail para ofrecer recomendaciones prácticas.
6. Análisis de los rastreos: revise periódicamente los datos de rastreo para detectar patrones, anomalías o cuellos de botella que podrían afectar al rendimiento de su aplicación.
7. Configuración de alertas: configure las alarmas en CloudWatch para detectar patrones inusuales o latencias prolongadas, lo que permite abordar los problemas de forma proactiva.
8. Mejora continua: revisite su estrategia de rastreo a medida que se agregan o modifiquen servicios para recoger todos los puntos de datos pertinentes.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS04-BP03 Implementación de telemetría de la experiencia del usuario
- OPS04-BP04 Implementación de telemetría de dependencias

Documentos relacionados:

- Guía para desarrolladores de AWS X-Ray
- Guía del usuario del agente de Amazon CloudWatch
- Amazon DevOps Guru User Guide

Videos relacionados:

- Use AWS X-Ray Insights
- AWS on Air ft. Observability: Amazon CloudWatch and AWS X-Ray

Ejemplos relacionados:

- Instrumenting your application for AWS X-Ray

## OPS 5. ¿Cómo reduce los defectos, facilita la reparación y mejora el flujo en producción?

Adopte enfoques que mejoren el flujo de cambios en producción, que activen la refactorización, la respuesta rápida sobre la calidad y la corrección de errores. Esto acelerará los cambios positivos que se introducen en producción, limitará los problemas implementados y logrará una rápida identificación y solución de los problemas introducidos a través de las actividades de implementación.

Prácticas recomendadas

- OPS05-BP01 Uso del control de versiones
- OPS05-BP02 Prueba y validación de los cambios

- OPS05-BP03 Uso de sistemas de administración de la configuración
- OPS05-BP04 Uso de sistemas de administración de compilación e implementación
- OPS05-BP05 Administración de parches
- OPS05-BP06 Uso compartido de estándares de diseño
- OPS05-BP07 Implementación de prácticas para mejorar la calidad del código
- OPS05-BP08 Uso de varios entornos
- OPS05-BP09 Cambios frecuentes, pequeños y reversibles
- OPS05-BP10 Automatización completa de la integración y la implementación

### OPS05-BP01 Uso del control de versiones

Use el control de versiones para activar el seguimiento de cambios y versiones.

Muchos servicios de AWS ofrecen capacidades de control de versiones. Utilice un sistema de revisiones o de control de orígenes como Git para administrar el código y otros artefactos, como las plantillas de AWS CloudFormation controladas por versiones de la infraestructura.

Resultado deseado: sus equipos colaboran en el código. Cuando se fusiona, el código es coherente y no se pierde ningún cambio. Los errores se revierten fácilmente mediante el control de versiones correcto.

Patrones comunes de uso no recomendados:

- Ha estado desarrollando y almacenando el código en su estación de trabajo. Ha sufrido un error de almacenamiento irrecuperable en la estación de trabajo y el código se ha perdido.
- Después de sobrescribir el código existente con sus cambios, reinicia la aplicación y ya no está operativa. No puede revertir el cambio.
- Tiene un bloqueo de escritura en un archivo de informe que tiene que editar otra persona. Contacta con usted para pedirle que deje de trabajar en él para que puedan completar sus tareas.
- Su equipo de investigación ha estado trabajando en un análisis detallado que modela su trabajo futuro. Alguien ha guardado accidentalmente su lista de la compra sobre el informe final. No puede revertir el cambio y tiene que volver a crear el informe.

Beneficios de establecer esta práctica recomendada: mediante el uso de las capacidades de control de versiones puede revertir fácilmente los estados buenos conocidos y las versiones anteriores, y limitar el riesgo de que se pierdan los activos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Mantenga los activos en repositorios con control de versiones. Esto permite hacer un seguimiento de los cambios, implementar versiones nuevas, detectar cambios en las versiones existentes y volver a versiones anteriores (por ejemplo, revertir a un estado conocido correcto en caso de error). Integre en sus procedimientos las capacidades de control de versiones de sus sistemas de administración de la configuración.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP04 Uso de sistemas de administración de compilación e implementación

Videos relacionados:

- AWS re:Invent 2023 - How Lockheed Martin builds software faster, powered by DevSecOps
- AWS re:Invent 2023 - How GitHub operationalizes AI for team collaboration and productivity

OPS05-BP02 Prueba y validación de los cambios

Cada cambio implementado se debe probar para evitar errores en producción. Esta práctica recomendada se centra en probar los cambios desde el control de versiones hasta la creación de artefactos. Además de los cambios en el código de la aplicación, las pruebas deben incluir la infraestructura, la configuración, los controles de seguridad y los procedimientos operativos. Las pruebas adoptan muchas formas, desde las pruebas unitarias hasta el análisis de componentes de software (SCA). Mover las pruebas más a la izquierda en el proceso de integración y entrega del software se traduce en una mayor certeza de la calidad de los artefactos.

Su organización debe desarrollar estándares de prueba para todos los artefactos de software. Las pruebas automatizadas reducen el trabajo y evitan los errores de las pruebas manuales. En algunos casos puede ser necesario hacer pruebas manuales. Los desarrolladores deben tener acceso a los resultados de las pruebas automatizadas para crear bucles de comentarios que mejoren la calidad del software.

Resultado deseado: los cambios en el software se prueban antes de su entrega. Los desarrolladores tienen acceso a los resultados de las pruebas y las validaciones. Su organización tiene un estándar de pruebas que se aplica a todos los cambios de software.

Patrones comunes de uso no recomendados:

- Implementa un nuevo cambio de software sin hacer ninguna prueba. No funciona en producción, lo que provoca una interrupción del servicio.
- Los nuevos grupos de seguridad se implementan con AWS CloudFormation sin haberse probado en un entorno de preproducción. Los grupos de seguridad hacen que la aplicación sea inaccesible para los clientes.
- Se modifica un método, pero no hay pruebas unitarias. El software no funciona cuando se implementa en producción.

Beneficios de establecer esta práctica recomendada: se reduce la tasa de errores de cambio de las implementaciones de software. Se mejora la calidad del software. Los desarrolladores son más conscientes de la viabilidad de su código. Las políticas de seguridad se pueden implementar con confianza para respaldar el cumplimiento de la organización. Los cambios en la infraestructura, como las actualizaciones automáticas de las políticas de escalamiento, se prueban con antelación para satisfacer las necesidades de tráfico.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las pruebas se hacen en todos los cambios, desde el código de la aplicación hasta la infraestructura, como parte de su práctica de integración continua. Los resultados de las pruebas se publican para que los desarrolladores tengan comentarios rápidos. Su organización tiene un estándar de pruebas que deben superar todos los cambios.

Utilice el poder de la IA generativa con Amazon Q Developer para mejorar la productividad de los desarrolladores y la calidad del código. Amazon Q Developer incluye la generación de sugerencias de código (basadas en modelos de lenguaje de gran tamaño), la producción de pruebas unitarias (incluidas condiciones límite) y mejoras de seguridad del código mediante la detección y la corrección de las vulnerabilidades de seguridad.

Ejemplo de cliente

Como parte de su canalización de integración continua, AnyCompany Retail ejecuta varios tipos de pruebas en todos los artefactos de software. Practica el desarrollo basado en pruebas, por lo que todo el software tiene pruebas unitarias. Una vez creado el artefacto, ejecuta pruebas integrales. Una vez completada esta primera ronda de pruebas, ejecuta un examen estático de la seguridad de la aplicación, que busca vulnerabilidades conocidas. Los desarrolladores reciben mensajes a

medida que se supera cada puerta de prueba. Una vez completadas todas las pruebas, el artefacto de software se almacena en un repositorio de artefactos.

Pasos para la implementación

1. Colabore con las partes interesadas de su organización en el desarrollo de un estándar de pruebas para los artefactos de software. ¿Qué pruebas estándar deben superar todos los artefactos? ¿Hay requisitos de cumplimiento o gobernanza que deban incluirse en la cobertura de las pruebas? ¿Necesita efectuar pruebas de calidad del código? Cuando finalicen las pruebas, ¿quién tiene que saberlo?
    1. AWS Deployment Pipeline Reference Architecture incluye una lista autorizada de tipos de pruebas que pueden llevarse a cabo en artefactos de software como parte de una canalización de integración.
2. Instrumente su aplicación con las pruebas necesarias en función de su estándar de pruebas de software. Cada conjunto de pruebas debería completarse en menos de diez minutos. Las pruebas deben ejecutarse como parte de una canalización de integración.
    a. Utilice Amazon Q Developer, una herramienta de IA generativa que puede ayudar a crear casos de pruebas unitarias (incluidas las condiciones de límite), generar funciones mediante código y comentarios e implementar algoritmos conocidos.
    b. Utilice Revisor de Amazon CodeGuru para comprobar si el código de la aplicación presenta defectos.
    c. Puede utilizar AWS CodeBuild para hacer pruebas en artefactos de software.
    d. AWS CodePipeline puede orquestar sus pruebas de software en una canalización.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP01 Uso del control de versiones
- OPS05-BP06 Uso compartido de estándares de diseño
- OPS05-BP07 Implementación de prácticas para mejorar la calidad del código
- OPS05-BP10 Automatización completa de la integración y la implementación

Documentos relacionados:

- Adoptar un enfoque de desarrollo basado en pruebas

- Accelerate your Software Development Lifecycle with Amazon Q
- Amazon Q Developer, now generally available, includes previews of new capabilities to reimagine developer experience
- The Ultimate Cheat Sheet for Using Amazon Q Developer in Your IDE
- Shift-Left Workload, leveraging AI for Test Creation
- Amazon Q Developer Center
- 10 ways to build applications faster with Amazon CodeWhisperer
- Looking beyond code coverage with Amazon CodeWhisperer
- Best Practices for Prompt Engineering with Amazon CodeWhisperer
- Automated AWS CloudFormation Testing Pipeline with TaskCat and CodePipeline
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST, and DAST tools
- Getting started with testing serverless applications
- My CI/CD pipeline is my release captain
- Documento técnico de AWS: Práctica de integración y entrega continuas

Videos relacionados:

- Implement an API with Amazon Q Developer Agent for Software Development
- Installing, Configuring, & Using Amazon Q Developer with JetBrains IDEs (How-to)
- Mastering the art of Amazon CodeWhisperer - YouTube playlist
- AWS re:Invent 2020: Testable infrastructure: Integration testing on AWS
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development
- Testing Your Infrastructure as Code with AWS CDK

Recursos relacionados:

- AWS Deployment Pipeline Reference Architecture: Application
- AWS Kubernetes DevSecOps Pipeline
- Run unit tests for a Node.js application from GitHub by using AWS CodeBuild
- Use Serverspec for test-driven development of infrastructure code

Servicios relacionados:

- Amazon Q Developer
- Revisor de Amazon CodeGuru
- AWS CodeBuild
- AWS CodePipeline

OPS05-BP03 Uso de sistemas de administración de la configuración

Utilice sistemas de administración de la configuración para efectuar modificaciones en la configuración y hacer un seguimiento de ellas. Estos sistemas reducen tanto los errores causados por los procesos manuales como el nivel de esfuerzo requerido para implementar los cambios.

La administración de la configuración estática establece valores al inicializar un recurso que se espera que permanezcan constantes durante toda la vida del recurso. La administración de la configuración dinámica establece valores en la inicialización que pueden cambiar o se espera que cambien durante la vida de un recurso. Por ejemplo, podría establecer un conmutador de características para activar la funcionalidad en su código a través de un cambio de configuración o cambiar el nivel de detalle del registro durante un incidente.

Las configuraciones deben implementarse en un estado conocido y coherente. Debe utilizar la inspección automatizada para supervisar continuamente las configuraciones de los recursos en todos los entornos y regiones. Estos controles deben definirse como un código y una gestión automatizados para garantizar que las reglas se apliquen de forma coherente en todos los entornos. Los cambios en las configuraciones deben actualizarse mediante procedimientos de control de cambios acordados y aplicarse de manera coherente, a la vez que se respeta el control de versiones. La configuración de la aplicación debe gestionarse de forma independiente del código de la aplicación y la infraestructura. Esto permite una implementación uniforme en varios entornos. Los cambios de configuración no dan como resultado la reconstrucción o la reimplementación de la aplicación.

Resultado deseado: configure, valide e implemente como parte de su proceso de integración continua y entrega continua (CI/CD). Supervisa para validar que las configuraciones sean correctas. Esto minimiza cualquier impacto en los usuarios finales y los clientes.

Patrones comunes de uso no recomendados:

- Actualiza manualmente la configuración del servidor web en toda su flota y varios servidores dejan de responder debido a errores de actualización.

- Actualiza manualmente su flota de servidores de aplicaciones en el transcurso de muchas horas. La incoherencia en la configuración durante el cambio provoca comportamientos inesperados.
- Alguien ha actualizado sus grupos de seguridad y ya no se puede acceder a los servidores web. Sin saber lo que ha cambiado, se pierde mucho tiempo investigando el problema, lo que prolonga el tiempo de recuperación.
- Una configuración de preproducción se introduce en producción a través de CI/CD sin validación. Expone a los usuarios y clientes a datos y servicios incorrectos.

Beneficios de establecer esta política recomendada: la adopción de sistemas de administración de la configuración reduce el nivel de esfuerzo para hacer cambios y hacer un seguimiento de los mismos, así como la frecuencia de los errores provocados por los procedimientos manuales. Los sistemas de administración de la configuración ofrecen garantías con respecto a la gobernanza, el cumplimiento y los requisitos reglamentarios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los sistemas de administración de la configuración se utilizan para seguir e implementar cambios en las configuraciones de las aplicaciones y el entorno. Los sistemas de administración de la configuración también se utilizan para reducir los errores causados por los procesos manuales, hacer que los cambios de configuración sean repetibles y auditables y reducir el nivel de esfuerzo.

En AWS, puede utilizar AWS Config para supervisar de forma continua las configuraciones de los recursos de AWS entre todas las cuentas y regiones. Le ayuda a hacer un seguimiento de su historial de configuración, comprender cómo un cambio de configuración afectaría a otros recursos y auditarlos con respecto a las configuraciones esperadas o deseadas mediante Reglas de AWS Config y AWS Config Conformance Packs.

Para las configuraciones dinámicas de sus aplicaciones que se ejecutan en instancias, AWS Lambda, contenedores, aplicaciones móviles o dispositivos de IoT de Amazon EC2, puede utilizar AWS AppConfig para configurarlas, validarlas, implementarlas y supervisarlas en todos sus entornos.

Pasos para la implementación

1. Identifique a los propietarios de la configuración.
   a. Haga que los propietarios de las configuraciones estén al tanto de cualquier necesidad de cumplimiento, gobernanza o normativa.

2. Identifique los elementos de configuración y los resultados.
    a. Los elementos de configuración son todas las configuraciones de las aplicaciones y los entornos afectadas por una implementación dentro de su canalización de CI/CD.
    b. Los resultados incluyen los criterios de éxito, la validación y lo que se debe supervisar.
3. Seleccione herramientas para la administración de la configuración en función de los requisitos empresariales y el proceso de entrega.
4. Considere el uso de implementaciones ponderadas, como las implementaciones canario, para los cambios de configuración significativos a fin de minimizar el impacto de las configuraciones incorrectas.
5. Integre la administración de la configuración en su canalización de CI/CD.
6. Valide todos los cambios introducidos.

Recursos

Prácticas recomendadas relacionadas:

- OPS06-BP01 Planificación para hacer frente a los cambios infructuosos
- OPS06-BP02 Implementaciones de prueba
- OPS06-BP03 Uso de estrategias de implementación seguras
- OPS06-BP04 Automatización de las pruebas y la reversión

Documentos relacionados:

- AWS Control Tower
- Acelerador de zonas de aterrizaje de AWS
- AWS Config
- What is AWS Config?
- AWS AppConfig
- What is AWS CloudFormation?
- Herramientas para desarrolladores de AWS
- AWS CodeBuild
- AWS CodePipeline
- AWS CodeDeploy

Videos relacionados:

- AWS re:Invent 2022 - Proactive governance and compliance for AWS workloads
- AWS re:Invent 2020: Achieve compliance as code using AWS Config
- Manage and Deploy Application Configurations with AWS AppConfig

OPS05-BP04 Uso de sistemas de administración de compilación e implementación

Utilice sistemas de administración de compilación e implementación. Estos sistemas reducen tanto los errores causados por los procesos manuales como el nivel de esfuerzo requerido para implementar los cambios.

En AWS, puede crear canalizaciones de integración continua/implementación continua (CI/CD) a través de servicios como las herramientas para desarrolladores de AWS (por ejemplo, AWS CodeBuild, AWS CodePipeline y AWS CodeDeploy).

Resultado deseado: sus sistemas de administración de compilación e implementación respaldan el sistema de integración continua y entrega continua (CI/CD) de su organización, que proporciona capacidades para automatizar implementaciones seguras con las configuraciones correctas.

Patrones comunes de uso no recomendados:

- Después de compilar su código en el sistema de desarrollo, copia el ejecutable en los sistemas de producción y no se inicia. Los archivos de registro locales indican que ha fallado debido a la falta de dependencias.
- Crea con éxito su aplicación con nuevas características en su entorno de desarrollo y proporciona el código a control de calidad. No pasa el control de calidad porque le faltan activos estáticos.
- El viernes, después de mucho esfuerzo, crea con éxito su aplicación manualmente en su entorno de desarrollo que incluye las funcionalidades recién codificadas. El lunes, no puede repetir los pasos que le permitieron crear con éxito su aplicación.
- Lleva a cabo las pruebas que ha creado para su nueva versión. A continuación, dedica la siguiente semana a configurar un entorno de pruebas y a llevar a cabo todas las pruebas de integración existentes, seguidas de las pruebas de rendimiento. El nuevo código tiene un impacto inaceptable en el rendimiento y debe desarrollarse y probarse de nuevo.

Beneficios de establecer esta práctica recomendada: al proporcionar mecanismos para gestionar las actividades de desarrollo e implementación, se reduce el nivel de esfuerzo para llevar a cabo tareas

repetitivas, se libera a los miembros del equipo para que se centren en sus tareas creativas de alto valor y se limita la introducción de errores de procedimientos manuales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los sistemas de administración de compilación e implementación se utilizan para seguir e implementar cambios, reducir los errores causados por los procesos manuales y reducir el nivel de esfuerzo requerido para una implementación segura. Automatice completamente el proceso de integración e implementación, desde el registro del código hasta la compilación, prueba, implementación y validación. Esto reduce el tiempo de entrega, disminuye los costos, fomenta una mayor frecuencia de cambios, reduce el nivel de esfuerzo y aumenta la colaboración.

Pasos para la implementación

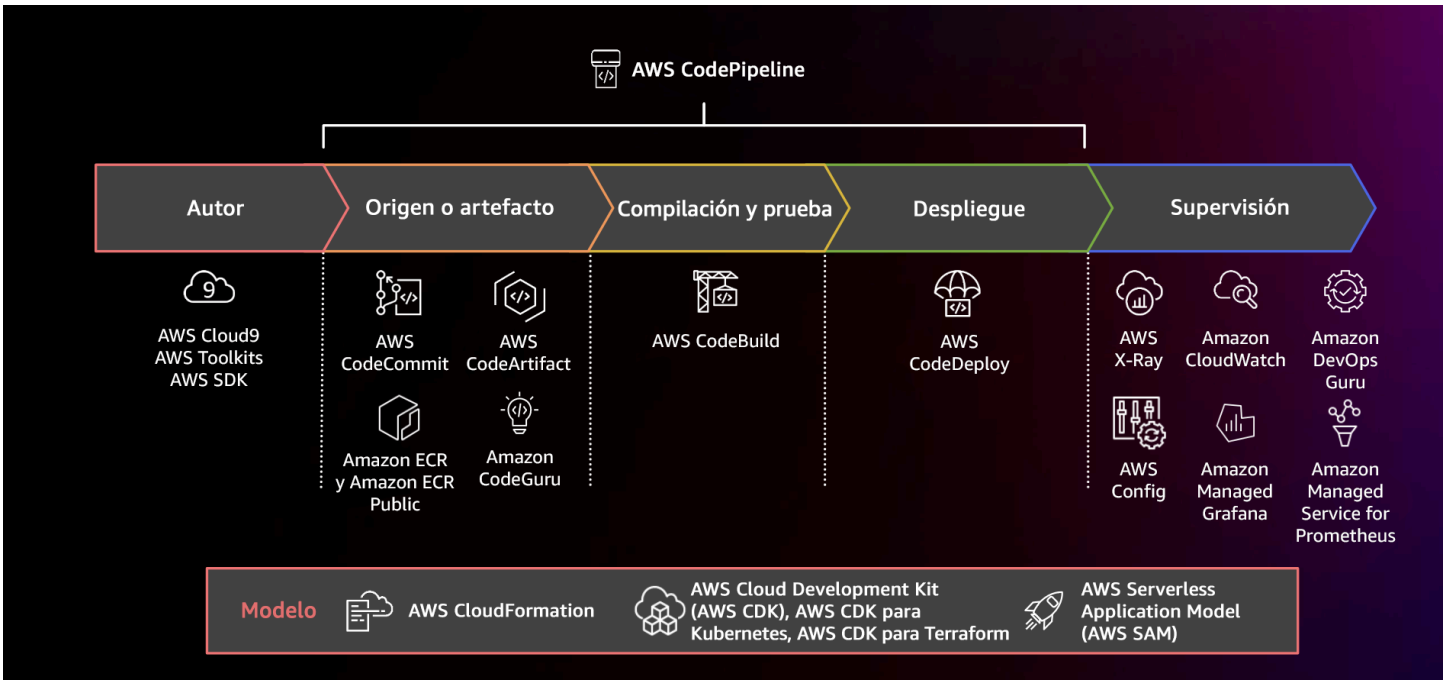



Diagrama que muestra el uso de una canalización de CI/CD con AWS CodePipeline y los servicios relacionados

1. Utilice un sistema de control de versiones para almacenar y administrar activos (como documentos, código fuente y archivos binarios).
2. Utilice CodeBuild para compilar su código fuente, ejecutar pruebas unitarias y producir artefactos listos para la implementación.

3. Utilice CodeDeploy como un servicio de implementación que automatiza las implementaciones de las aplicaciones en instancias de Amazon EC2, en instancias en las instalaciones o en funciones de AWS Lambda sin servidor o servicios de Amazon ECS.
4. Supervise sus implementaciones.

Recursos

Prácticas recomendadas relacionadas:

- OPS06-BP04 Automatización de las pruebas y la reversión

Documentos relacionados:

- Herramientas para desarrolladores de AWS
- ¿Qué es AWS CodeBuild?
- AWS CodeBuild
- ¿Qué es AWS CodeDeploy?

Videos relacionados:

- AWS re:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS

OPS05-BP05 Administración de parches

Administre parches para ampliar las características, resolver problemas y mantener la conformidad con la gobernanza. Automatice la administración de parches para reducir los errores causados por los procesos manuales, la escala y el nivel de esfuerzo requerido para aplicarlos.

La administración de parches y vulnerabilidades forma parte de sus actividades de administración de beneficios y riesgos. Es preferible tener infraestructuras inmutables e implementar las cargas de trabajo en estados en buenas condiciones conocidos y verificados. Cuando esto no es viable, la opción que queda es el parcheado in situ.

AWS Health es el origen autorizado de la información sobre eventos planificados del ciclo de vida y otros eventos que requieren acciones y que afectan el estado de los recursos de Nube de AWS. Debe estar al tanto de los próximos cambios y actualizaciones que deben realizarse. Los principales eventos planificados del ciclo de vida se envían con al menos seis meses de antelación.

El Generador de imágenes de Amazon EC2 proporciona canalizaciones para actualizar las imágenes de las máquinas. Como parte de la administración de parches, considere la posibilidad de utilizar Imagen de máquina de Amazon (AMI) con una canalización de imágenes de AMI o imágenes de contenedor con una canalización de imágenes de Docker, a la vez que AWS Lambda proporciona patrones para tiempos de ejecución personalizados y bibliotecas adicionales para eliminar las vulnerabilidades.

Debe gestionar las actualizaciones de las imágenes de Imagen de máquina de Amazon para Linux o Windows Server mediante el Generador de imágenes de Amazon EC2. Puede utilizar Amazon Elastic Container Registry (Amazon ECR) con su canalización existente para gestionar las imágenes de Amazon ECS y las de Amazon EKS. Lambda incluye características de administración de versiones.

La aplicación de parches no debe llevarse a cabo en los sistemas de producción sin antes hacer pruebas en un entorno seguro. Los parches solo deben aplicarse si sirven para mejorar los resultados operativos o empresariales. En AWS, puede utilizar AWS Systems Manager Patch Manager para automatizar el proceso de aplicación de parches en los sistemas administrados y programar la actividad con las Ventanas de mantenimiento de Systems Manager.

Resultado deseado: las imágenes del contenedor y AMI están parcheadas, actualizadas y listas para su lanzamiento. Puede hacer un seguimiento del estado de todas las imágenes implementadas y determinar el cumplimiento de los parches. Puede informar sobre el estado actual y disponer de un proceso que satisfaga sus necesidades de cumplimiento.

Patrones comunes de uso no recomendados:

- Se le encomienda la aplicación de todos los nuevos parches de seguridad en un plazo de dos horas, lo que da lugar a numerosas interrupciones debido a la incompatibilidad de las aplicaciones con los parches.
- Una biblioteca sin parches tiene consecuencias no deseadas, ya que partes desconocidas utilizan las vulnerabilidades de la misma para acceder a su carga de trabajo.
- Aplica parches a los entornos de los desarrolladores sin avisarles. Recibe múltiples quejas de los desarrolladores porque su entorno ha dejado de funcionar tal como se esperaba.
- No se ha parcheado el software comercial disponible en el mercado en una instancia persistente. Cuando tiene un problema con el software y contacta con el proveedor, este le notifica que la versión no es compatible y que tiene que aplicar un parche en un nivel específico para recibir asistencia.

- Ha utilizado un parche para el software de cifrado publicado recientemente que tiene importantes mejoras de rendimiento. Su sistema sin parches tiene problemas de rendimiento que continúan como resultado de no aplicar los parches.
- Se le notifica una vulnerabilidad de día cero que requiere una solución de emergencia y tiene que parchar todos sus entornos manualmente.
- No está al tanto de las acciones críticas necesarias para mantener los recursos, como las actualizaciones de versión obligatorias, porque no revisa los próximos eventos del ciclo de vida planificado y otra información. Pierde tiempo crítico para la planificación y la ejecución, lo que provoca cambios de emergencia para los equipos y un impacto potencial o un tiempo de inactividad inesperado.

Beneficios de establecer esta práctica recomendada: al establecer un proceso de administración de parches, que incluya sus criterios de aplicación de parches y la metodología de distribución en sus entornos, puede escalar e informar sobre los niveles de parches. Esto proporciona garantías en torno a la aplicación de parches de seguridad y garantiza una visibilidad clara del estado de las correcciones conocidas que se están aplicando. Esto fomenta la adopción de las características y capacidades deseadas, la rápida eliminación de problemas y el cumplimiento sostenido de la gobernanza. Implemente sistemas de administración de parches y automatización para reducir el nivel de esfuerzo en la implementación de parches y limitar los errores causados por los procesos manuales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Aplique parches a los sistemas para solucionar problemas, para obtener las características o capacidades deseadas y para mantener la conformidad con la política de gobernanza y los requisitos de soporte de los proveedores. En sistemas inmutables, implemente con el conjunto de parches adecuados para lograr el resultado deseado. Automatice el mecanismo de administración de parches para reducir el tiempo que tarda en aplicarlos, evitar los errores causados por los procesos manuales y reducir el nivel de esfuerzo requerido para aplicar los parches.

Pasos para la implementación

Para el Generador de imágenes de Amazon EC2:

1. Con el Generador de imágenes de Amazon EC2, especifique los detalles de la canalización:
   a. Cree una canalización de imágenes y asígnele un nombre.

   b. Defina el horario y la zona horaria de la canalización.
   c. Configure las dependencias.
2. Elija una receta:
   a. Seleccione una receta existente o cree una nueva.
   b. Seleccione el tipo de imagen.
   c. Asigne un nombre y versión a la receta.
   d. Seleccione la imagen base.
   e. Agregue componentes de compilación y agréguelos al registro de destino.
3. Opcional: defina la configuración de la infraestructura.
4. Opcional: defina los ajustes de configuración.
5. Revise la configuración.
6. Mantenga la higiene de las recetas con regularidad.

Para Systems Manager Patch Manager:

1. Cree una línea de base de revisiones.
2. Seleccione un método de operaciones de creación de revisiones.
3. Habilite el análisis y la generación de informes de cumplimiento.

Recursos

Prácticas recomendadas relacionadas:

- OPS06-BP04 Automatización de las pruebas y la reversión

Documentos relacionados:

- What is Amazon EC2 Image Builder
- Create an image pipeline using the Amazon EC2 Image Builder
- Create a container image pipeline
- AWS Systems Manager Patch Manager
- Uso de Patch Manager (consola)
- Trabajo con informes de conformidad de las revisiones

- Herramientas para desarrolladores de AWS

Videos relacionados:

- CI/CD for Serverless Applications on AWS
- Design with Ops in Mind

Ejemplos relacionados:

- Tutoriales de AWS Systems Manager Systems Manager Patch Manager

OPS05-BP06 Uso compartido de estándares de diseño

Comparta las prácticas recomendadas entre los equipos para aumentar la conciencia y maximizar los beneficios del trabajo de desarrollo. Documéntelas y manténgalas actualizadas a medida que evoluciona su arquitectura. Si se aplican los estándares compartidos en su organización, es fundamental que existan mecanismos para solicitar adiciones, cambios y excepciones a los estándares. Sin esta opción, los estándares se convierten en un obstáculo para la innovación.

Resultado deseado: los estándares de diseño se comparten entre los equipos de sus organizaciones. Se documentan y actualizan a medida que evolucionan las prácticas recomendadas.

Patrones comunes de uso no recomendados:

- Dos equipos de desarrollo distintos han creado, cada uno, un servicio de autenticación de usuarios. Sus usuarios tienen que mantener un conjunto de credenciales diferente para cada parte del sistema a la que quieran acceder.
- Cada equipo administra su propia infraestructura. Un nuevo requisito de conformidad obliga a cambiar la infraestructura y cada equipo lo aplica de forma distinta.

Beneficios de establecer esta práctica recomendada: el uso de estándares compartidos favorece la adopción de las prácticas recomendadas y maximiza las ventajas de los esfuerzos de desarrollo. La documentación y actualización de los estándares de diseño mantiene a su organización al día de las prácticas recomendadas y de los requisitos de seguridad y cumplimiento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Comparta entre los equipos las prácticas recomendadas, los estándares de diseño, las listas de verificación, los procedimientos operativos, las orientaciones y los requisitos de gobernanza. Disponga de procedimientos para solicitar cambios, adiciones y excepciones a los estándares de diseño para apoyar la mejora y la innovación. Asegúrese de que los equipos estén al tanto del contenido publicado. Disponga de un mecanismo para mantener al día los estándares de diseño a medida que surgen nuevas prácticas recomendadas.

Ejemplo de cliente

AnyCompany Retail cuenta con un equipo de arquitectura interfuncional que crea patrones de arquitectura de software. Este equipo construye la arquitectura con la conformidad y la gobernanza integradas. Los equipos que adoptan estos estándares compartidos se benefician de la conformidad y la gobernanza integradas. Pueden construir rápidamente sobre el estándar de diseño. El equipo de arquitectura se reúne trimestralmente para evaluar los patrones de arquitectura y actualizarlos en caso necesario.

Pasos para la implementación

1. Identifique un equipo interfuncional que se encargue de desarrollar y actualizar los estándares de diseño. Este equipo debe trabajar con las partes interesadas de toda la organización a fin de desarrollar estándares de diseño, procedimientos operativos, listas de verificación, guías y requisitos de gobernanza. Documente los estándares de diseño y compártalos dentro de su organización.
   a. AWS Service Catalog puede utilizarse para crear carteras que representen los estándares de diseño mediante la infraestructura como código. Puede compartir carteras entre cuentas.
2. Disponga de un mecanismo para mantener al día los estándares de diseño a medida que se identifiquen nuevas prácticas recomendadas.
3. Si los estándares de diseño se aplican de forma centralizada, cuente con un proceso para solicitar cambios, actualizaciones y exenciones.

Nivel de esfuerzo para el plan de implementación: medio. El desarrollo de un proceso para crear y compartir estándares de diseño precisa de coordinación y cooperación con las partes interesadas de toda la organización.

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP03 Evaluación de los requisitos de gobernanza: los requisitos de gobernanza influyen en los estándares de diseño.
- OPS01-BP04 Evaluación de los requisitos de cumplimiento: la conformidad es un elemento vital de la creación de estándares de diseño.
- OPS07-BP02 Garantía de una revisión sistemática de la preparación operativa: las listas de verificación de preparación operativa son un mecanismo para implementar los estándares de diseño a la hora de diseñar la carga de trabajo.
- OPS11-BP01 Implementación de un proceso de mejora continua: la actualización de los estándares de diseño forma parte de la mejora continua.
- OPS11-BP04 Administración de conocimientos: como parte de su práctica de administración del conocimiento, documente y comparta los estándares de diseño.

Documentos relacionados:

- Automate AWS Backups with AWS Service Catalog
- AWS Service Catalog Account Factory-Enhanced
- How Expedia Group built Database as a Service (DBaaS) offering using AWS Service Catalog
- Maintain visibility over the use of cloud architecture patterns
- Simplify sharing your AWS Service Catalog portfolios in an AWS Organizations setup

Videos relacionados:

- AWS Service Catalog – Getting Started
- AWS re:Invent 2020: Manage your AWS Service Catalog portfolios like an expert

Ejemplos relacionados:

- Arquitectura de referencia de AWS Service Catalog
- AWS Service Catalog Workshop

Servicios relacionados:

- AWS Service Catalog

### OPS05-BP07 Implementación de prácticas para mejorar la calidad del código

Adopte prácticas para mejorar la calidad del código y minimizar los defectos. Algunos ejemplos son el desarrollo basado en pruebas, las revisiones de código, la adopción de estándares y la programación en pareja. Integre estas prácticas a su proceso de integración y entrega continuas.

Resultado deseado: su organización utiliza las prácticas recomendadas, como las revisiones de código o la programación en pareja, para mejorar la calidad del código. Los desarrolladores y operadores adoptan las prácticas recomendadas de calidad del código como parte del ciclo de vida de desarrollo del software.

Patrones comunes de uso no recomendados:

- Envía código a la rama principal de su aplicación sin una revisión del código. El cambio se implementa automáticamente en producción y provoca una interrupción del servicio.
- Se desarrolla una nueva aplicación sin pruebas de unidad, integrales o de integración. No hay forma de probar la aplicación antes de la implementación.
- Los equipos hacen cambios manuales en producción para corregir defectos. Los cambios no se someten a pruebas ni revisiones de código y no se capturan ni registran en los procesos de integración y entrega continuas.

Beneficios de establecer esta práctica recomendada: al adoptar prácticas para mejorar la calidad del código, puede ayudar a minimizar los problemas introducidos en la producción. Las prácticas recomendadas para la calidad del código incluyen la programación en parejas, las revisiones del código y la implementación de herramientas de productividad de IA.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Adopte prácticas para mejorar la calidad del código y minimizar los defectos antes de la implementación. Utilice prácticas como desarrollo basado en pruebas, revisiones de código y programación en pareja para mejorar la calidad de su proceso.

Utilice el poder de la IA generativa con Amazon Q Developer para mejorar la productividad de los desarrolladores y la calidad del código. Amazon Q Developer incluye la generación de sugerencias de código (basadas en modelos de lenguaje de gran tamaño), la producción de pruebas unitarias (incluidas condiciones límite) y mejoras de seguridad del código mediante la detección y la corrección de las vulnerabilidades de seguridad.

Ejemplo de cliente

AnyCompany Retail adopta diversas prácticas para mejorar la calidad del código. Ha adoptado el desarrollo basado en pruebas como norma para escribir aplicaciones. Para algunas funciones nuevas, hace que los desarrolladores programen en pareja durante un sprint. Cada solicitud de extracción se somete a una revisión de código por parte de un desarrollador sénior antes de que se integre e implemente.

Pasos para la implementación

1. Adopte prácticas que fomenten la calidad del código, como el desarrollo basado en pruebas, las revisiones del código y la programación en parejas, en su proceso de integración y entrega continuas. Utilice estas técnicas para mejorar la calidad del software.
   a. Utilice Amazon Q Developer, una herramienta de IA generativa que puede ayudarle a crear casos de pruebas unitarias (incluidas las condiciones de límite), generar funciones mediante código y comentarios, implementar algoritmos conocidos, detectar infracciones y vulnerabilidades de las políticas de seguridad en su código, detectar secretos, escanear la infraestructura como código (IaC), documentar código y aprender bibliotecas de códigos de terceros con mayor rapidez.
   b. El Revisor de Amazon CodeGuru puede proporcionar recomendaciones de programación para código Java y Python mediante el uso de machine learning.

Nivel de esfuerzo para el plan de implementación: medio. Existen numerosas formas de implementar esta práctica recomendada, pero conseguir que la organización la adopte puede suponer un reto.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP02 Prueba y validación de los cambios
- OPS05-BP06 Uso compartido de estándares de diseño

Documentos relacionados:

- Adoptar un enfoque de desarrollo basado en pruebas
- Accelerate your Software Development Lifecycle with Amazon Q
- Amazon Q Developer, now generally available, includes previews of new capabilities to reimagine developer experience

- The Ultimate Cheat Sheet for Using Amazon Q Developer in Your IDE
- Shift-Left Workload, leveraging AI for Test Creation
- Amazon Q Developer Center
- 10 ways to build applications faster with Amazon CodeWhisperer
- Looking beyond code coverage with Amazon CodeWhisperer
- Best Practices for Prompt Engineering with Amazon CodeWhisperer
- Agile Software Guide
- My CI/CD pipeline is my release captain
- Automate code reviews with Amazon CodeGuru Reviewer
- Adoptar un enfoque de desarrollo basado en pruebas
- How DevFactory builds better applications with Amazon CodeGuru
- On Pair Programming
- RENGA Inc. automates code reviews with Amazon CodeGuru
- The Art of Agile Development: Test-Driven Development
- Why code reviews matter (and actually save time!)

Videos relacionados:

- Implement an API with Amazon Q Developer Agent for Software Development
- Installing, Configuring, & Using Amazon Q Developer with JetBrains IDEs (How-to)
- Mastering the art of Amazon CodeWhisperer - YouTube playlist
- AWS re:Invent 2020: Continuous improvement of code quality with Amazon CodeGuru
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development

Servicios relacionados:

- Amazon Q Developer
- Revisor de Amazon CodeGuru
- Generador de perfiles de Amazon CodeGuru

OPS05-BP08 Uso de varios entornos

Use diversos entornos para experimentar, desarrollar y poner a prueba su carga de trabajo. Utilice niveles crecientes de controles a medida que los entornos se acerquen a la fase de producción para asegurarse de que su carga de trabajo funcione según lo previsto cuando se implemente.

Resultado deseado: tiene varios entornos que reflejan sus necesidades de cumplimiento y gobernanza. Prueba y hace avanzar el código a través de entornos en su ruta hasta producción.

1. Su organización lo consigue mediante el establecimiento de una zona de aterrizaje, que proporciona gobernanza, controles, automatizaciones de cuentas, redes, seguridad y observabilidad operativa. Gestione estas capacidades de zona de aterrizaje mediante el uso de varios entornos. Un ejemplo común es una organización de un entorno de pruebas para desarrollar y probar cambios en una zona de aterrizaje basada en AWS Control Tower, que incluye AWS IAM Identity Center y políticas como las políticas de control de servicios (SCP). Todos estos elementos pueden afectar significativamente al acceso y al funcionamiento de Cuentas de AWS en la zona de aterrizaje.
2. Además de estos servicios, sus equipos amplían las capacidades de las zonas de aterrizaje con soluciones publicadas por socios de AWS y AWS o como soluciones personalizadas desarrolladas dentro de su organización. Algunos ejemplos de soluciones publicadas por AWS son Customizations for AWS Control Tower (cFCT) y AWS Control Tower Account Factory for Terraform (AFT).
3. Su organización aplica los mismos principios de pruebas, promoción del código y cambios de política para la zona de aterrizaje en los entornos que se encuentran en su camino hacia la producción. Esta estrategia proporciona un entorno de zona de aterrizaje estable y seguro para sus equipos de aplicaciones y cargas de trabajo.

Patrones comunes de uso no recomendados:

- Está desarrollando en un entorno compartido y otro desarrollador sobrescribe sus cambios de código.
- Los controles de seguridad restrictivos de su entorno de desarrollo compartido le impiden experimentar con nuevos servicios y características.
- Lleva a cabo pruebas de carga en sus sistemas de producción y provoca una interrupción a los usuarios.
- Se ha producido un error crítico que ha provocado la pérdida de datos en producción. En el entorno de producción, se intenta recrear las condiciones que condujeron a la pérdida de datos

para poder identificar cómo ocurrió y evitar que vuelva a suceder. Para evitar más pérdida de datos durante las pruebas, se ve obligado a hacer que la aplicación no esté disponible para los usuarios.

- Utiliza un servicio de inquilino múltiple y no puede atender la solicitud de un cliente de tener un entorno dedicado.
- Puede que no siempre pruebe, pero cuando lo hace, lo hace en su entorno de producción.
- Cree que la simplicidad de un entorno único anula el alcance del impacto de los cambios en el entorno.
- Actualiza una función clave de zona de aterrizaje, pero el cambio perjudica la capacidad de su equipo de vender cuentas para nuevos proyectos o para sus cargas de trabajo existentes.
- Aplica nuevos controles a sus Cuentas de AWS, pero el cambio repercute en la capacidad de su equipo de cargas de trabajo para implementar cambios en sus Cuentas de AWS.

Beneficios de establecer esta práctica recomendada: cuando implementa varios entornos, puede dar respaldo a varios entornos simultáneos de desarrollo, de pruebas y de producción sin crear conflictos entre los desarrolladores o las comunidades de usuarios. En el caso de funciones complejas, como las zonas de aterrizaje, se reduce considerablemente el riesgo de cambios, se simplifica el proceso de mejora y se reduce el riesgo de que se produzcan actualizaciones críticas en el entorno. Las organizaciones que utilizan zonas de aterrizaje se benefician naturalmente de tener múltiples cuentas en su entorno de AWS, con configuraciones de estructura de cuentas, gobierno, red y seguridad. Con el tiempo, a medida que su organización crece, la zona de aterrizaje puede evolucionar para proteger y organizar sus cargas de trabajo y sus recursos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Use varios entornos y proporcione a los desarrolladores entornos de pruebas con controles minimizados para ayudar con la experimentación. Proporcione entornos de desarrollo individuales para ayudar al trabajo en paralelo, que aumenta la agilidad del desarrollo. Implemente controles más rigurosos en los entornos que están cercanos a la producción para que los desarrolladores puedan innovar. Utilice infraestructura como código y sistemas de administración de la configuración para implementar entornos que estén configurados de forma coherente con los controles presentes en la producción y asegurarse de que los sistemas funcionarán como se espera cuando se implementen. Cuando los entornos no estén en uso (por ejemplo, sistemas de desarrollo durante la noche y los fines de semana), apáguelos para evitar los costos asociados a los recursos inactivos. Cuando haga pruebas de carga, implemente entornos semejantes al de producción para mejorar los resultados válidos.

Los equipos de ingeniería de plataformas, redes y operaciones de seguridad suelen gestionar las capacidades a nivel de la organización con requisitos distintos. La separación de las cuentas por sí sola no basta para proporcionar y mantener entornos separados para la experimentación, el desarrollo y las pruebas. En estos casos, cree instancias de AWS Organizations separadas.

Recursos

Documentos relacionados:

- Programador de instancias de AWS
- ¿Qué es AWS CloudFormation?
- Organización de su entorno de AWS mediante varias cuentas. Varias organizaciones. Pruebe los cambios en el entorno general de AWS
- Guía de AWS Control Tower

OPS05-BP09 Cambios frecuentes, pequeños y reversibles

Los cambios frecuentes, pequeños y reversibles tienen menos alcance y menos repercusiones. Cuando se utilizan junto con sistemas de administración de cambios, sistemas de administración de la configuración y sistemas de compilación y entrega, los cambios frecuentes, pequeños y reversibles reducen el alcance y el impacto de un cambio. Al hacerlo, los problemas se solucionan de forma más eficaz y rápida con la opción de revertir los cambios.

Patrones comunes de uso no recomendados:

- Implementa una nueva versión de su aplicación trimestralmente con una ventana de cambios que significa que un servicio principal está desactivado.
- Hace cambios frecuentes en el esquema de su base de datos sin hacer un seguimiento de los cambios en sus sistemas de administración.
- Lleve a cabo actualizaciones manuales in situ, y sobrescriba las instalaciones y configuraciones existentes y no tiene un plan de reversión claro.

Beneficios de establecer esta práctica recomendada: los esfuerzos de desarrollo son más rápidos al implementar pequeños cambios con frecuencia. Cuando los cambios son pequeños, es mucho más fácil identificar si tienen consecuencias no deseadas y es más fácil revertirlos. Cuando los cambios son reversibles, hay menos riesgo de aplicar el cambio, ya que la recuperación se simplifica. El proceso de cambio tiene un menor riesgo y el impacto de un cambio erróneo se reduce.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Utilice cambios frecuentes, pequeños y reversibles para reducir el alcance y las repercusiones del cambio. Esto facilita la resolución de problemas, ayuda a implementar correcciones rápidamente y permite revertir los cambios. También aumenta el ritmo con el que entrega valor a la empresa.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP03 Uso de sistemas de administración de la configuración
- OPS05-BP04 Uso de sistemas de administración de compilación e implementación
- OPS06-BP04 Automatización de las pruebas y la reversión

Documentos relacionados:

- Implementing Microservices on AWS
- Microservices - Observability

OPS05-BP10 Automatización completa de la integración y la implementación

Automatice la compilación, la implementación y la comprobación de la carga de trabajo. Esto reduce tanto los errores causados por los procesos manuales como el esfuerzo requerido para implementar los cambios.

Aplique metadatos mediante etiquetas de registro y Grupos de recursos de AWS mediante una estrategia de etiquetado coherente para permitir la identificación de sus recursos. Etiquete sus recursos para la organización, la contabilidad de costos, los controles de acceso y el objetivo de ejecución de actividades de operaciones automatizadas.

Resultado deseado: los desarrolladores utilizan herramientas para entregar código y progresar hasta producción. Los desarrolladores no tienen que iniciar sesión en la Consola de administración de AWS para entregar actualizaciones. Existe un registro de auditoría completo de los cambios y la configuración, que satisface las necesidades de gobernanza y cumplimiento. Los procesos son repetibles y están estandarizados en todos los equipos. Los desarrolladores pueden centrarse en el desarrollo y en la introducción de código, lo que aumenta la productividad.

Patrones comunes de uso no recomendados:

- El viernes finaliza con la creación del nuevo código para la ramificación de características. El lunes, después de ejecutar los scripts de pruebas de calidad del código y cada uno de los scripts de pruebas unitarias, comprueba el código para la siguiente versión programada.
- Se le asigna la tarea de codificar una solución para un problema crítico que afecta a un gran número de clientes en producción. Después de probar la corrección, confirma el código y envía un correo electrónico a la administración de cambios para solicitar la aprobación de su implementación en producción.
- Como desarrollador, debe iniciar sesión en la Consola de administración de AWS para crear un nuevo entorno de desarrollo utilizando métodos y sistemas no estándar.

Beneficios de establecer esta práctica recomendada: al implementar sistemas automatizados de administración de compilación e implementación, se reducen los errores causados por los procesos manuales y se reduce el esfuerzo para implementar los cambios, lo que ayuda a los miembros de su equipo a centrarse en la entrega de valor empresarial. Aumenta la velocidad de entrega a medida que progresa hasta producción.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Utilice sistemas de administración de compilación e implementación para hacer un seguimiento e implementar el cambio, a fin de reducir tanto los errores causados por los procesos manuales como el nivel de esfuerzo. Automatice completamente el proceso de integración e implementación, desde el registro del código hasta la compilación, prueba, implementación y validación. Esto reduce el tiempo de entrega, fomenta una mayor frecuencia de cambios, reduce el nivel de esfuerzo, aumenta la velocidad de comercialización, se traduce en un aumento de la productividad y aumenta la seguridad del código a medida que progresa hasta producción.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP03 Uso de sistemas de administración de la configuración
- OPS05-BP04 Uso de sistemas de administración de compilación e implementación

Documentos relacionados:

- ¿Qué es AWS CodeBuild?
- ¿Qué es AWS CodeDeploy?

Videos relacionados:

- AWS re:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS

## OPS 6. ¿Cómo mitiga los riesgos de implementación?

Adopte métodos que proporcionen una respuesta inmediata sobre la calidad y logren una recuperación rápida de los cambios que no obtengan los resultados deseados. El uso de estas prácticas ayuda a mitigar el impacto de los problemas generados con la implementación de cambios.

Prácticas recomendadas

- OPS06-BP01 Planificación para hacer frente a los cambios infructuosos
- OPS06-BP02 Implementaciones de prueba
- OPS06-BP03 Uso de estrategias de implementación seguras
- OPS06-BP04 Automatización de las pruebas y la reversión

OPS06-BP01 Planificación para hacer frente a los cambios infructuosos

Planifique la reversión a un estado óptimo conocido o la corrección en el entorno de producción si una implementación causa un resultado no deseado. Tener una política para establecer un plan de este tipo ayuda a todos los equipos a desarrollar estrategias para recuperarse de los cambios fallidos. Algunos ejemplos de estrategias son los pasos de implementación y reversión, las políticas de cambio, los indicadores de características, el aislamiento del tráfico y el cambio de tráfico. Una sola versión puede incluir varios cambios de componentes relacionados. La estrategia debe proporcionar la capacidad de resistir o recuperarse de un error de cualquier cambio de componente.

Resultado deseado: ha preparado un plan de recuperación detallado para su cambio en caso de que no tenga éxito. Además, ha reducido el tamaño de su versión para minimizar el impacto potencial en otros componentes de la carga de trabajo. Como resultado, ha reducido su impacto empresarial al acortar el posible tiempo de inactividad causado por un cambio infructuoso y ha aumentado la flexibilidad y la eficiencia de los tiempos de recuperación.

Patrones comunes de uso no recomendados:

- Ha llevado a cabo una implementación y la aplicación se comporta de forma inestable, aunque parece que hay usuarios activos en el sistema. Debe decidir si deshacer el cambio, lo que afectará a los usuarios activos, o esperar a revertir el cambio sabiendo que los usuarios pueden verse afectados igualmente.
- Después de hacer un cambio de rutina, sus nuevos entornos son accesibles, pero una de sus subredes ha quedado inaccesible. Tiene que decidir si revertirlo todo o intentar reparar la subred inaccesible. Mientras toma esa decisión, no se podrá acceder a la subred.
- Sus sistemas no tienen una arquitectura que permita actualizarlos con versiones más pequeñas. Como resultado, tiene dificultades para revertir esos cambios masivos durante una implementación infructuosa.
- No utiliza la infraestructura como código (IaC) y ha llevado a cabo actualizaciones manuales en su infraestructura que han dado lugar a una configuración no deseada. No puede hacer un seguimiento eficaz de los cambios manuales ni revertirlos.
- Como no ha medido el aumento de la frecuencia de sus implementaciones, su equipo no tiene incentivos para reducir el tamaño de los cambios y mejorar los planes de reversión para cada cambio, lo que genera más riesgos y mayores tasas de errores.
- No mide la duración total de una interrupción provocada por cambios infructuosos. Su equipo no puede establecer prioridades ni mejorar la eficacia del proceso de implementación y del plan de recuperación.

Beneficios de establecer esta práctica recomendada: tener un plan para recuperarse de cambios fallidos minimiza el tiempo medio de recuperación (MTTR) y reduce el impacto en la organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La adopción por parte de los equipos de lanzamiento de políticas y prácticas coherentes permite a la organización planificar lo que debe suceder si se producen cambios infructuosos. La política debe permitir aplicar correcciones temporales en circunstancias concretas. En cualquier situación, un plan de corrección temporal o reversión debe estar bien documentado y probado antes de implementarlo en producción en vivo para minimizar el tiempo que lleva revertir un cambio.

Pasos para la implementación

1. Documente las políticas que requieren que los equipos tengan planes efectivos para revertir los cambios dentro de un período específico.

   a. Las políticas deben especificar cuándo se permite una situación de corrección temporal.
   b. Exija un plan de reversión documentado al que puedan acceder todas las partes involucradas.
   c. Especifique los requisitos para la reversión (por ejemplo, cuando se descubra que se han implementado cambios no autorizados).
2. Analice el grado de impacto de todos los cambios relacionados con cada componente de una carga de trabajo.
   a. Permita que los cambios repetibles se estandaricen, se diseñen con plantillas y se autoricen previamente si siguen un flujo de trabajo coherente que aplique las políticas de cambio.
   b. Reduzca el impacto potencial de cualquier cambio mediante la reducción del tamaño del cambio para que la recuperación lleve menos tiempo y cause menos repercusión en la empresa.
   c. Asegúrese de que los procedimientos de reversión reviertan el código al estado correcto conocido para evitar incidentes siempre que sea posible.
3. Integre herramientas y flujos de trabajo para aplicar sus políticas mediante programación.
4. Haga que los datos sobre los cambios sean visibles para otros propietarios de cargas de trabajo para mejorar la velocidad de diagnóstico de cualquier cambio infructuoso que no se pueda revertir.
   a. Mida el éxito de esta práctica a través de datos de cambios visibles e identifique las mejoras iterativas.
5. Utilice herramientas de supervisión para verificar el éxito o el fracaso de una implementación a fin de acelerar la toma de decisiones sobre la reversión.
6. Mida la duración de la interrupción durante un cambio infructuoso para mejorar continuamente sus planes de recuperación.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS06-BP04 Automatización de las pruebas y la reversión

Documentos relacionados:

- AWS Builders' Library: Asegurar la seguridad en las restauraciones durante las implementaciones
- Documento técnico de AWS: Change Management in the Cloud

Videos relacionados:

- re:Invent 2019 | Amazon’s approach to high-availability deployment

OPS06-BP02 Implementaciones de prueba

Pruebe los procedimientos de lanzamiento en preproducción con la misma configuración de implementación, controles de seguridad, pasos y procedimientos que en producción. Valide que todos los pasos implementados se completen según lo esperado, como la inspección de archivos, configuraciones y servicios. Pruebe más a fondo todos los cambios con pruebas funcionales, de integración y de carga, junto con cualquier supervisión, como la comprobación de estado. Al llevar a cabo estas pruebas, puede identificar los problemas de implementación con prontitud y tiene la oportunidad de planificarlos y mitigarlos antes de llegar a producción.

Puede crear entornos paralelos temporales para probar cada cambio. Automatice la implementación de los entornos de prueba mediante la infraestructura como código (IaC) para reducir la cantidad de trabajo que implica y garantizar la estabilidad, la coherencia y una entrega de características más rápida.

Resultado deseado: su organización adopta una cultura de desarrollo basada en pruebas que incluye la implementación de pruebas. Esto garantiza que los equipos se centren en ofrecer valor empresarial en lugar de administrar las versiones. Los equipos participan desde el principio de la identificación de los riesgos de la implementación para determinar el curso de mitigación adecuado.

Patrones comunes de uso no recomendados:

- Durante las versiones de producción, las implementaciones no probadas provocan problemas frecuentes que requieren la solución de problemas y la escalada.
- Su versión contiene infraestructura como código (IaC) que actualiza los recursos existentes. No tiene la seguridad de si IaC se ejecuta correctamente o si afecta a los recursos.
- Implementa una característica nueva en su aplicación. No funciona según lo previsto y no hay visibilidad hasta que los usuarios afectados lo denuncien.
- Actualiza sus certificados. Instala accidentalmente los certificados en los componentes incorrectos, lo que pasa desapercibido y afecta a los visitantes del sitio web porque no se puede establecer una conexión segura con el sitio web.

Beneficios de establecer esta práctica recomendada: las exhaustivas pruebas en la preproducción de los procedimientos de implementación y los cambios introducidos por ellos minimizan la posible

repercusión en producción causada por las etapas de implementación. Esto aumenta la confianza durante el lanzamiento de producción y minimiza el soporte operativo sin ralentizar la velocidad de los cambios que se introducen.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Probar el proceso de implementación es tan importante como probar los cambios que resultan de la implementación. Esto se puede lograr probando los pasos de implementación en un entorno de preproducción que refleje la producción lo más fielmente posible. Los problemas más habituales, como pasos de implementación incompletos o incorrectos o configuraciones incorrectas, pueden detectarse antes de pasar a producción. Además, puede poner a prueba sus pasos de recuperación.

Ejemplo de cliente

Como parte de su canalización de integración y entrega continuas (CI/CD), AnyCompany Retail lleva a cabo los pasos definidos necesarios para lanzar actualizaciones de infraestructura y software para sus clientes en un entorno similar al de producción. La canalización se compone de comprobaciones previas para detectar desviaciones (detectar cambios en los recursos hechos fuera de su IaC) en los recursos antes de la implementación, así como para validar las acciones que la IaC emprende tras su inicio. Valida los pasos de la implementación, como verificar que determinados archivos y configuraciones estén en su sitio y que los servicios estén en estado de ejecución y respondan correctamente a las comprobaciones de estado del host local antes de volver a registrarse en el equilibrador de carga. Además, todos los cambios se someten a una serie de pruebas automatizadas, como pruebas funcionales, de seguridad, de regresión, de integración y de carga.

Pasos para la implementación

1. Haga comprobaciones previas a la instalación para reflejar el entorno de preproducción en producción.
   a. Use la detección de desviaciones para detectar cuándo se han cambiado los recursos fuera de CloudFormation.
   b. Use los conjuntos de cambio para validar que la intención de una actualización de la pila coincida con las acciones que CloudFormation lleva a cabo cuando se inicia el conjunto de cambios.
2. Esto desencadena un paso de aprobación manual en AWS CodePipeline para autorizar la implementación en el entorno de preproducción.

3. Utilice configuraciones de implementación, como los archivos AWS CodeDeploy AppSpec, para definir los pasos de implementación y validación.
4. Cuando proceda, integre AWS CodeDeploy con otros servicios de AWS o integre AWS CodeDeploy con los productos y servicios de los socios.
5. Supervise las implementaciones mediante Amazon CloudWatch, AWS CloudTrail y las notificaciones de eventos de Amazon SNS.
6. Lleve a cabo pruebas automatizadas posteriores a la implementación, incluidas pruebas funcionales, de seguridad, de regresión, de integración y de carga.
7. Solucione los problemas de implementación.
8. La validación correcta de los pasos precedentes debería iniciar un flujo de trabajo de aprobación manual para autorizar la implementación en producción.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP02 Prueba y validación de los cambios

Documentos relacionados:

- AWS Builders' Library: Automatización de implementaciones seguras y sin intervención | Implementaciones de prueba
- Documento técnico de AWS: Práctica de integración y entrega continuas en AWS
- The Story of Apollo - Amazon's Deployment Engine
- How to test and debug AWS CodeDeploy locally before you ship your code
- Integrating Network Connectivity Testing with Infrastructure Deployment

Videos relacionados:

- re:Invent 2020 | Testing software and systems at Amazon

Ejemplos relacionados:

- Tutorial: Deploy and Amazon ECS service with a validation test

### OPS06-BP03 Uso de estrategias de implementación seguras

Las implementaciones de producción seguras controlan el flujo de cambios beneficiosos con el objetivo de minimizar cualquier impacto percibido por los clientes como consecuencia de dichos cambios. Los controles de seguridad proporcionan mecanismos de inspección para validar los resultados deseados y limitar el alcance del impacto de cualquier defecto introducido por los cambios o por errores en la implementación. Las implementaciones seguras incluyen estrategias como: indicadores de características, caja individual, continuas (versiones de canarios), inmutables, división del tráfico e implementaciones azul-verde.

Resultado deseado: su organización utiliza un sistema de entrega continua e integración continua (CI/CD) que proporciona capacidades para automatizar implementaciones seguras. Los equipos deben utilizar estrategias adecuadas para implementaciones seguras.

Patrones comunes de uso no recomendados:

- Implementa un cambio sin éxito en toda la producción de una sola vez. Como resultado, todos los clientes resultan afectados simultáneamente.
- Un defecto introducido en una implementación simultánea en todos los sistemas requiere una versión de emergencia. Corregirlo para todos los clientes lleva varios días.
- La administración del lanzamiento de producción requiere la planificación y la participación de varios equipos. Esto limita su capacidad de actualizar con frecuencia las características para sus clientes.
- Lleva a cabo una implementación mutable al modificar los sistemas existentes. Tras descubrir que el cambio no ha tenido éxito, se ve obligado a modificar de nuevo los sistemas para restaurar la versión antigua, lo que prolonga el tiempo de recuperación.

Beneficios de establecer esta práctica recomendada: las implementaciones automatizadas equilibran la velocidad de las implementaciones con la entrega de cambios beneficiosos de manera coherente a los clientes. Limitar el impacto evita caros errores de implementación y maximiza la capacidad de los equipos de responder de manera eficiente a los errores.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

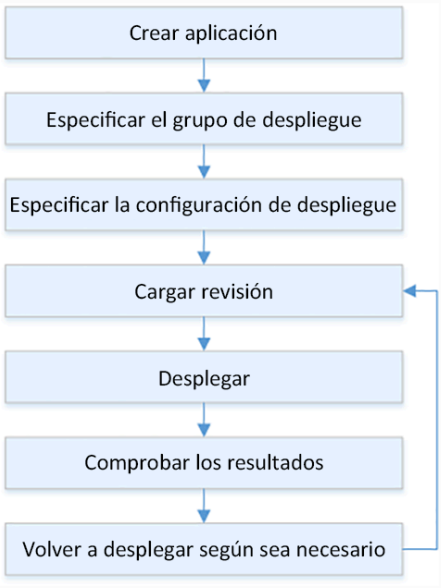
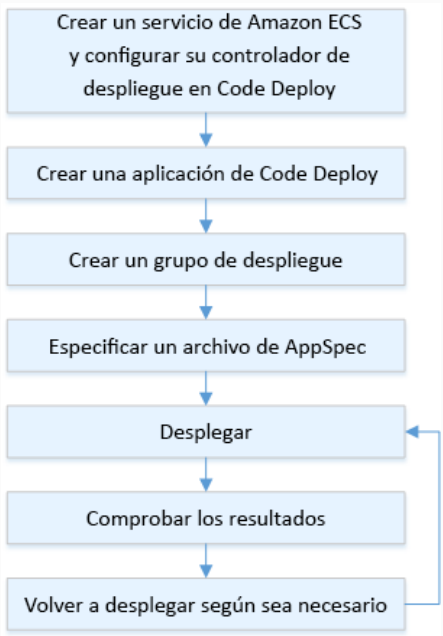
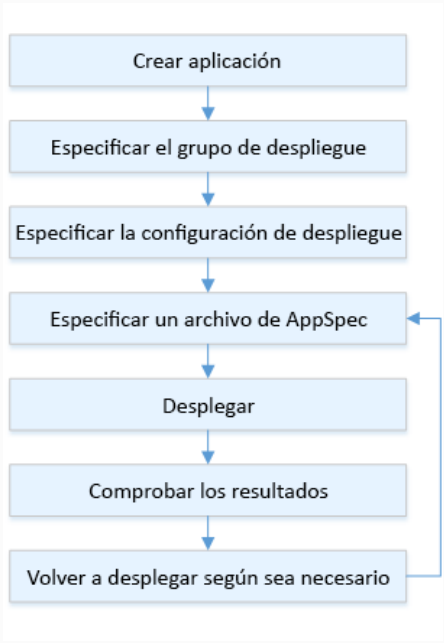
#### Guía para la implementación

Los errores continuos en la entrega pueden provocar una reducción de la disponibilidad del servicio y una mala experiencia para los clientes. Para maximizar la tasa de implementaciones satisfactorias,

implemente controles de seguridad en el proceso de lanzamiento de principio a fin de minimizar los errores de implementación, con el objetivo de lograr implementaciones sin ningún error.

Ejemplo de cliente

AnyCompany Retail tiene la misión de lograr implementaciones con un tiempo de inactividad mínimo o nulo, lo que significa que los usuarios no perciban ningún impacto durante las implementaciones. Para lograrlo, la empresa ha establecido patrones de implementación (consulte el siguiente diagrama de flujo de trabajo), como implementaciones azul-verde y continuas. Todos los equipos adoptan uno o más de estos patrones en su canalización de CI/CD.

| Flujo de trabajo de CodeDeploy para Amazon EC2 | Flujo de trabajo de CodeDeploy para Amazon ECS | Flujo de trabajo de CodeDeploy para Lambda |
|---|---|---|
|  |  |  |







Pasos para la implementación

1. Use un flujo de trabajo de aprobación para iniciar la secuencia de pasos de implementación de producción, de la promoción a la producción.
2. Use un sistema de implementación automatizado como AWS CodeDeploy. Las opciones de implementaciones locales de AWS CodeDeploy para EC2/en las instalaciones e implementaciones azul-verde para EC2/en las instalaciones, AWS Lambda y Amazon ECS (consulte el diagrama de flujo de trabajo anterior).

   a. Cuando proceda, integre AWS CodeDeploy con otros servicios de AWS o integre AWS CodeDeploy con los productos y servicios de los socios.
3. Utilice implementaciones azul/verde para bases de datos como Amazon Aurora y Amazon RDS.
4. Supervise las implementaciones mediante Amazon CloudWatch, AWS CloudTrail y las notificaciones de eventos de Amazon Simple Notification Service (Amazon SNS).
5. Haga pruebas automatizadas posteriores a la implementación, incluidas pruebas funcionales, de seguridad, de regresión, de integración y cualquier prueba de carga.
6. Solucione los problemas de implementación.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP02 Prueba y validación de los cambios
- OPS05-BP09 Cambios frecuentes, pequeños y reversibles
- OPS05-BP10 Automatización completa de la integración y la implementación

Documentos relacionados:

- AWS Builders' Library: Automatización de implementaciones seguras y sin intervención | Implementaciones de producción
- AWS Builders Library: My CI/CD pipeline is my release captain | Safe, automatic production releases
- Documento técnico de AWS: Práctica de integración y entrega continuas en AWS | Métodos de implementación
- AWS CodeDeploy Guía del usuario de
- Working with deployment configurations in AWS CodeDeploy
- Configuración de una implementación de un lanzamiento canario de API Gateway
- Amazon ECS Deployment Types
- Fully Managed Blue/Green Deployments in Amazon Aurora and Amazon RDS
- Blue/Green deployments with AWS Elastic Beanstalk

Videos relacionados:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon
- re:Invent 2019 | Amazon's Approach to high-availability deployment

Ejemplos relacionados:

- Try a Sample Blue/Green Deployment in AWS CodeDeploy
- Taller | Building CI/CD pipelines for Lambda canary deployments using AWS CDK
- Taller | Building your first DevOps Blue/Green pipeline with Amazon ECS
- Taller | Building your first DevOps Blue/Green pipeline with Amazon EKS
- Taller | EKS GitOps with ArgoCD
- Taller | CI/CD on AWS Workshop
- Implementing cross-account CI/CD with AWS SAM for container-based Lambda functions

OPS06-BP04 Automatización de las pruebas y la reversión

Para aumentar la velocidad, la fiabilidad y la confianza de su proceso de implementación, tenga una estrategia para automatizar las capacidades de prueba y reversión en los entornos de preproducción y producción. Automatice las pruebas al implementar en producción para simular las interacciones entre humanos y sistemas que verifican los cambios que se implementan. Automatice la reversión para volver rápidamente a un estado válido anterior conocido. La reversión debe iniciarse automáticamente en condiciones predefinidas, como cuando no se logra el resultado deseado del cambio o cuando la prueba automatizada fracasa. La automatización de estas dos actividades mejora la tasa de éxito de las implementaciones, minimiza el tiempo de recuperación y reduce el impacto potencial en la empresa.

Resultado deseado: sus pruebas automatizadas y sus estrategias de reversión se integran en el proceso de integración y entrega continuas (CI/CD). Su supervisión puede validarse según sus criterios de éxito e iniciar una reversión automática en caso de error. Esto minimiza cualquier impacto en los usuarios finales y los clientes. Por ejemplo, cuando se satisfacen todos los resultados de las pruebas, promociona el código al entorno de producción donde se inician las pruebas de regresión automatizadas, mediante el uso de los mismos casos de prueba. Si los resultados de la prueba de regresión no coinciden con las expectativas, se inicia una reversión automática en el flujo de trabajo de la canalización.

Patrones comunes de uso no recomendados:

- Sus sistemas no tienen una arquitectura que permita actualizarlos con versiones más pequeñas. Como resultado, tiene dificultades para revertir esos cambios masivos durante una implementación infructuosa.
- El proceso de implementación consta de una serie de pasos manuales. Tras implementar los cambios en la carga de trabajo, se inician las pruebas posteriores a la implementación. Tras las pruebas, se da cuenta de que no puede utilizar la carga de trabajo y los clientes están desconectados. A continuación, empieza a revertir a la versión anterior. Todos estos pasos manuales retrasan la recuperación general del sistema y provocan un impacto prolongado en sus clientes.
- Ha dedicado tiempo a desarrollar casos de prueba automatizados para funciones que no se utilizan con frecuencia en su aplicación, lo que minimiza el retorno de la inversión en su capacidad de llevar a cabo pruebas automatizadas.
- Su versión se compone de actualizaciones de aplicaciones, infraestructura, parches y configuración que son independientes entre sí. Sin embargo, tiene una única canalización de CI/CD que introduce todos los cambios a la vez. Un error en un componente le obliga a revertir todos los cambios, lo que hace que la reversión sea compleja e ineficiente.
- Su equipo completa el trabajo de codificación en el primer sprint y comienza el trabajo en el segundo, pero el plan no incluía las pruebas hasta el tercer sprint. Como resultado, las pruebas automatizadas revelaron defectos en el primer sprint que tenían que haberse resuelto antes de empezar a probar los resultados del segundo sprint, con lo que se retrasa todo el lanzamiento y se devalúan las pruebas automatizadas.
- Los casos de las pruebas de regresión automatizadas para el lanzamiento de producción se han completado, pero no está supervisando el estado de la carga de trabajo. Como no puede saber si el servicio se ha reiniciado o no, no está seguro de si es necesaria una reversión o si ya se ha producido.

Beneficios de establecer esta práctica recomendada: las pruebas automatizadas aumentan la transparencia del proceso de pruebas y su capacidad para abarcar más características en un intervalo más reducido. Al probar y validar los cambios en la producción, puede identificar los problemas de forma inmediata. La mejora de la coherencia con herramientas de prueba automatizadas permite una mejor detección de los defectos. Al revertir automáticamente a la versión anterior, se minimiza el impacto en los clientes. La reversión automatizada, en última instancia,

inspira más confianza en sus capacidades de implementación al reducir el impacto empresarial. En general, estas capacidades reducen el tiempo de entrega y, al mismo tiempo, garantizan la calidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Automatice las pruebas de los entornos implementados para confirmar los resultados deseados con más rapidez. Automatice la reversión a un estado conocido correcto anterior cuando no se logren resultados predefinidos para minimizar el tiempo de recuperación y reducir los errores causados por los procesos manuales. Integre las herramientas de prueba con el flujo de trabajo de la canalización para probar y minimizar las entradas manuales de manera coherente. Priorice la automatización de los casos de prueba, como aquellos que mitigan los mayores riesgos y que deben probarse con frecuencia con cada cambio. Esto le ayuda a medir el retorno de la inversión (ROI) y da a los propietarios de cargas de trabajo la oportunidad de optimizar sus recursos y reducir costos.

Pasos para la implementación

1. Establezca un ciclo de vida de pruebas para su ciclo de vida de desarrollo que defina cada etapa del proceso de prueba, desde la planificación de los requisitos hasta el desarrollo de los casos de prueba, la configuración de las herramientas, las pruebas automatizadas y el cierre de los casos de prueba.
   a. Cree un enfoque de pruebas específico para la carga de trabajo a partir de su estrategia general de pruebas.
   b. Considere una estrategia de pruebas continuas cuando sea apropiado durante todo el ciclo de vida de desarrollo.
2. Seleccione herramientas automatizadas para efectuar pruebas y reversiones en función de sus requisitos empresariales y de las inversiones en curso.
3. Decida qué casos de prueba quiere automatizar y cuáles se deberán llevar a cabo manualmente. Estos se pueden definir en función de la prioridad de valor empresarial de la característica que se está probando. Alinee a todos los miembros del equipo con este plan y verifique la responsabilidad de efectuar las pruebas manuales.
   a. Aplique capacidades de pruebas automatizadas a casos de prueba específicos que tengan sentido para la automatización, como los casos repetibles o que se ejecutan con frecuencia, los que requieren tareas repetitivas o los que se requieren en varias configuraciones.

   b. Defina los scripts de automatización de pruebas, así como los criterios de éxito en la herramienta de automatización, de modo que se pueda iniciar la automatización continua del flujo de trabajo cuando fracasan casos específicos.
   c. Defina criterios de error concretos para la reversión automática.
4. Priorice la automatización de las pruebas para obtener resultados coherentes con un desarrollo exhaustivo de casos de prueba en los que la complejidad y la interacción humana tengan un mayor riesgo de fracaso.
5. Integre las herramientas automatizadas de pruebas y reversión en la canalización de CI/CD.
   a. Desarrolle criterios de éxito claros para los cambios.
   b. Supervise y observe para detectar estos criterios y revertir automáticamente los cambios cuando se cumplan criterios de reversión específicos.
6. Lleve a cabo diferentes tipos de pruebas automatizadas de producción, como:
   a. Pruebas A/B para mostrar los resultados en comparación con la versión actual entre dos grupos de pruebas de usuarios.
   b. Pruebas canario que permiten implementar el cambio en un subconjunto de usuarios antes de lanzarlo para todos.
   c. Pruebas de marca de características que permiten activar y desactivar las características de la nueva versión de una en una desde fuera de la aplicación para que cada característica nueva se pueda validar por sí sola.
   d. Pruebas de regresión para verificar la nueva funcionalidad con los componentes interrelacionados ya existentes.
7. Supervise los aspectos operativos de la aplicación, las transacciones y las interacciones con otras aplicaciones y componentes. Redacte informes que muestren el éxito de los cambios por carga de trabajo, de modo que pueda identificar qué partes de la automatización y el flujo de trabajo se pueden optimizar aún más.
   a. Elabore informes de resultados de pruebas que le ayuden a tomar decisiones rápidas sobre si se deben invocar o no los procedimientos de reversión.
   b. Implemente una estrategia que permita la reversión automática en función de condiciones de error predefinidas que resulten de uno o más de sus métodos de prueba.
8. Desarrolle los casos de prueba automatizados para poder volver a usarlos en futuros cambios repetibles.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS06-BP01 Planificación para hacer frente a los cambios infructuosos
- OPS06-BP02 Implementaciones de prueba

Documentos relacionados:

- AWS Builders' Library: Asegurar la seguridad en las restauraciones durante las implementaciones
- Redeploy and rollback a deployment with AWS CodeDeploy
- 8 best practices when automating your deployments with AWS CloudFormation

Ejemplos relacionados:

- Serverless UI testing using Selenium, AWS Lambda, AWS Fargate, and AWS Developer Tools

Videos relacionados:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon
- re:Invent 2019 | Amazon's Approach to high-availability deployment

## OPS 7. ¿Cómo sabe que está listo para admitir una carga de trabajo?

Evalúe la disponibilidad operativa de la carga de trabajo, los procesos y procedimientos, y el personal para comprender los riesgos operativos relacionados con la carga de trabajo.

Prácticas recomendadas

- OPS07-BP01 Garantía de la capacidad del personal
- OPS07-BP02 Garantía de una revisión sistemática de la preparación operativa
- OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos
- OPS07-BP04 Uso de manuales de estrategias para investigar problemas
- OPS07-BP05 Toma de decisiones fundamentadas para implementar sistemas y cambios
- OPS07-BP06 Creación de planes de asistencia para cargas de trabajo de producción

### OPS07-BP01 Garantía de la capacidad del personal

Disponga de un mecanismo para comprobar que cuente con la cantidad adecuada de personal formado para atender la carga de trabajo. Deben recibir formación sobre la plataforma y los servicios que componen su carga de trabajo. Transmítales los conocimientos necesarios para poder gestionar la carga de trabajo. Debe disponer de suficiente personal formado para atender el funcionamiento normal de la carga de trabajo y solucionar las incidencias que se produzcan. Cuente con suficiente personal para que pueda rotar durante las guardias y vacaciones, a fin de evitar que el personal se sienta quemado.

Resultado deseado:

- Hay suficiente personal formado para atender la carga de trabajo cuando esta se encuentre disponible.
- El personal recibe formación sobre el software y los servicios que componen la carga de trabajo.

Patrones comunes de uso no recomendados:

- Se implementa una carga de trabajo sin miembros del equipo formados para operar la plataforma y los servicios en uso.
- Se carece de suficiente personal para facilitar las rotaciones de guardia o para que el personal se tome tiempo libre.

Beneficios de establecer esta práctica recomendada:

- Contar con miembros del equipo cualificados permite un apoyo eficaz para su carga de trabajo.
- Si hay suficientes miembros en el equipo, es posible atender la carga de trabajo y las rotaciones de guardia, al tiempo que disminuye el riesgo de síndrome de burnout.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Compruebe que haya suficiente personal formado para atender la carga de trabajo. Compruebe que cuenta con suficientes miembros del equipo para cubrir las actividades operativas, incluidas las rotaciones de guardia.

Ejemplo de cliente

AnyCompany Retail se asegura de que los equipos que atienden la carga de trabajo cuentan con la formación y el personal adecuados. Tienen suficientes ingenieros para tolerar una rotación de guardia. El personal recibe formación sobre el software y la plataforma en los que se basa la carga de trabajo y se le anima a obtener certificaciones. Hay personal suficiente para que los empleados puedan tomarse tiempo libre sin dejar de atender la carga de trabajo y la rotación de guardia.

Pasos para la implementación

1. Asigne un número adecuado de personal para operar y respaldar la carga de trabajo, incluidas las tareas de guardia, los problemas de seguridad y los eventos del ciclo de vida, como las tareas de fin de soporte y de rotación de certificados.
2. Forme a su personal sobre el software y las plataformas que componen su carga de trabajo.
    a. Capacitación y certificación de AWS tiene una biblioteca de cursos sobre AWS. Ofrece cursos gratuitos y de pago, en línea y presenciales.
    b. AWS organiza eventos y seminarios web en los que puede aprender de la mano de expertos de AWS.
3. Realice lo siguiente de forma periódica:
    - Evalúe el tamaño y las aptitudes del equipo a medida que cambien las condiciones operativas y la carga de trabajo.
    - Ajuste el tamaño y las competencias del equipo para que se ciñan a los requisitos operativos.
    - Verifique la habilidad y la capacidad para hacer frente a los eventos del ciclo de vida previstos, la seguridad no planeada y las notificaciones operativas a través de AWS Health.

Nivel de esfuerzo para el plan de implementación: alto. La contratación y la formación de un equipo que atienda la carga de trabajo puede suponer un esfuerzo considerable, pero promete importantes ventajas a largo plazo.

Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP04 Administración de conocimientos: los miembros del equipo deben disponer de la información necesaria para operar y atender la carga de trabajo. La administración del conocimiento es la clave para alcanzar este objetivo.

Documentos relacionados:

- Eventos y seminarios web de AWS
- Capacitación y certificación de AWS

OPS07-BP02 Garantía de una revisión sistemática de la preparación operativa

Utilice las revisiones de la preparación operativa (ORR) para validar que puede utilizar su carga de trabajo. ORR es un mecanismo desarrollado en Amazon para validar que los equipos puedan utilizar con seguridad sus cargas de trabajo. Una ORR es un proceso de revisión e inspección que utiliza una lista de verificación de requisitos. Una ORR es una experiencia de autoservicio que los equipos utilizan para certificar sus cargas de trabajo. Las ORR incluyen las prácticas recomendadas procedentes de las lecciones aprendidas en nuestros años de creación de software.

Una lista de verificación de ORR se compone de recomendaciones de arquitectura, proceso operativo, administración de eventos y calidad de lanzamiento. Nuestro proceso de corrección de errores (CoE) es uno de los principales impulsores de estos elementos. Su análisis posterior al incidente debe impulsar la evolución de su propia ORR. Una ORR no solo consiste en seguir las prácticas recomendadas, sino en evitar que se repitan sucesos ya vistos. Por último, los requisitos de seguridad, gobernanza y conformidad también pueden incluirse en una ORR.

Ejecute las ORR antes de que una carga de trabajo se lance a la disponibilidad general y, después, a lo largo del ciclo de vida de desarrollo del software. Ejecutar la ORR antes del lanzamiento aumenta su capacidad para utilizar la carga de trabajo de forma segura. Vuelva a ejecutar periódicamente su ORR en la carga de trabajo para detectar cualquier desviación de las prácticas recomendadas. Puede tener listas de verificación de ORR para el lanzamiento de nuevos servicios y ORR para las revisiones periódicas. Esto le ayuda a mantenerse al día en cuanto a las nuevas prácticas recomendadas que surgen y a incorporar las lecciones aprendidas del análisis posterior al incidente. A medida que madure su uso de la nube, podrá incorporar los requisitos de ORR en su arquitectura de forma predeterminada.

Resultado deseado: tiene una lista de verificación de ORR con las prácticas recomendadas para su organización. Las ORR se llevan a cabo antes de lanzar las cargas de trabajo. Las ORR se llevan a cabo periódicamente a lo largo del ciclo de vida de la carga de trabajo.

Patrones comunes de uso no recomendados:

- Lanza una carga de trabajo sin saber si puede utilizarla.
- Los requisitos de gobernanza y seguridad no se incluyen en la certificación de una carga de trabajo para su lanzamiento.

- Las cargas de trabajo no se revalúan periódicamente.
- Las cargas de trabajo se lanzan sin los procedimientos necesarios.
- Observa la repetición de los mismos errores de causa raíz en varias cargas de trabajo.

Beneficios de establecer esta práctica recomendada:

- Sus cargas de trabajo incluyen las prácticas recomendadas de arquitectura, procesos y administración.
- Las lecciones aprendidas se incorporan al proceso de ORR.
- Se aplican los procedimientos necesarios cuando se lanzan las cargas de trabajo.
- Las ORR se ejecutan a lo largo del ciclo de vida del software de sus cargas de trabajo.

Nivel de riesgo si no se establece esta práctica recomendada: alto

Guía para la implementación

Una ORR es dos cosas: un proceso y una lista de verificación. Su organización debe adoptar el proceso de ORR y contar con la asistencia de un patrocinador ejecutivo. Como mínimo, las ORR deben hacerse antes de que una carga de trabajo se lance a la disponibilidad general. Ejecute la ORR durante todo el ciclo de vida del desarrollo del software para mantenerla actualizada con las prácticas recomendadas o los nuevos requisitos. La lista de verificación de ORR debe incluir elementos de configuración, requisitos de seguridad y gobernanza, y las prácticas recomendadas de su organización. Con el tiempo, puede utilizar servicios, como AWS Config, AWS Security Hub CSPM y las barreras de protección de AWS Control Tower para convertir las prácticas recomendadas de la ORR en barreras de protección para la detección automática de las prácticas recomendadas.

Ejemplo de cliente

Tras varios incidentes de producción, AnyCompany Retail decidió implementar un proceso de ORR. Elaboró una lista de verificación compuesta de prácticas recomendadas, requisitos de gobernanza y conformidad, y lecciones aprendidas de las interrupciones. Las nuevas cargas de trabajo llevan a cabo las ORR antes de su lanzamiento. Cada carga de trabajo lleva a cabo una ORR anual con un subconjunto de prácticas recomendadas para incorporar nuevas prácticas y requisitos que se agregan a la lista de verificación de ORR. Con el tiempo, AnyCompany Retail utilizó AWS Config para detectar algunas prácticas recomendadas, lo que agilizó el proceso de ORR.

Pasos para la implementación

Para obtener más información sobre las ORR, lea el documento técnico sobre las revisiones de la preparación operativa (ORR). En él se ofrece información detallada sobre la historia del proceso ORR, cómo crear su propia práctica ORR y cómo desarrollar su lista de verificación de ORR. Los siguientes pasos son una versión abreviada de ese documento. Para conocer en profundidad qué son las ORR y cómo crear las suyas, le recomendamos que lea ese documento técnico.

1. Reúna a las principales partes interesadas, incluidos los representantes de seguridad, operaciones y desarrollo.
2. Pida a cada parte interesada que aporte al menos un requisito. Para la primera iteración, intente limitar el número de elementos a treinta o menos.
   - El Appendix B: Example ORR questions del documento técnico sobre las revisiones de la preparación operativa (ORR) contiene las preguntas de ejemplo que puede usar para empezar.
3. Recopile sus requisitos en una hoja de cálculo.
   - Puede usar enfoques personalizados en AWS Well-Architected Tool para desarrollar su ORR y compartirlos entre sus cuentas y su organización de AWS.
4. Identifique una carga de trabajo para llevar a cabo la ORR en ella. Lo ideal es una carga de trabajo previa al lanzamiento o una carga de trabajo interna.
5. Repase la lista de verificación de ORR y tome nota de las detecciones. Los descubrimientos pueden ser correctos si existe una mitigación. Agregue cualquier descubrimiento que carezca de una mitigación a su lista de tareas pendientes e impleméntelas antes de lanzarlas.
6. Siga agregando las prácticas recomendadas y los requisitos a su lista de verificación ORR con el tiempo.

Los clientes de Soporte con Enterprise Support pueden solicitar el taller de revisión de la preparación operativa a su gerente técnico de cuentas. El taller es una sesión de trabajo en sentido inverso interactiva para desarrollar su propia lista de verificación de ORR.

Nivel de esfuerzo para el plan de implementación: alto. La adopción de una práctica de ORR en su organización requiere el patrocinio ejecutivo y la aceptación de las partes interesadas. Cree y actualice la lista de verificación con las aportaciones de toda su organización.

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP03 Evaluación de los requisitos de gobernanza : los requisitos de gobernanza encajan de forma natural en una lista de verificación de ORR.
- OPS01-BP04 Evaluación de los requisitos de cumplimiento : los requisitos de conformidad se incluyen a veces en una lista de verificación de ORR. Otras veces son un proceso independiente.
- OPS03-BP07 Recursos adecuados para los equipos: la capacidad del equipo es un buen candidato para un requisito de ORR.
- OPS06-BP01 Planificación para hacer frente a los cambios infructuosos: antes de lanzar la carga de trabajo, debe establecerse un plan de restauración o de avance.
- OPS07-BP01 Garantía de la capacidad del personal: para respaldar una carga de trabajo hay que contar con el personal necesario.
- SEC01-BP03 Identificación y validación de los objetivos de control: los objetivos de control de seguridad son excelentes requisitos de ORR.
- REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos: los planes de recuperación de desastres son un buen requisito de ORR.
- COST02-BP01 Desarrollo de políticas basadas en los requisitos de su organización: las políticas de administración de costos son adecuadas para incluirlas en su lista de verificación de ORR.

Documentos relacionados:

- AWS Control Tower - Guardrails in AWS Control Tower
- AWS Well-Architected Tool - Custom Lenses
- Operational Readiness Review Template de Adrian Hornsby
- Operational Readiness Reviews (ORR) Whitepaper

Videos relacionados:

- AWS Supports You | Building an Effective Operational Readiness Review (ORR)

Ejemplos relacionados:

- Sample Operational Readiness Review (ORR) Lens

Servicios relacionados:

- AWS Config
- AWS Control Tower
- AWS Security Hub CSPM
- AWS Well-Architected Tool

OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos

Un manual de procedimientos es un proceso documentado para lograr un resultado específico. Los manuales de procedimientos consisten en una serie de pasos que alguien sigue para conseguir algo. Los manuales de procedimientos se han utilizado en operaciones que se remontan a los primeros días de la aviación. En las operaciones en la nube, utilizamos manuales de procedimientos para reducir el riesgo y lograr los resultados deseados. En su forma más simple, un manual de procedimientos es una lista de verificación para completar una tarea.

Los manuales de procedimientos son una parte esencial del funcionamiento de su carga de trabajo. Desde la incorporación de un nuevo miembro del equipo hasta la implementación de una versión importante, los manuales de procedimientos son los procesos codificados que proporcionan resultados coherentes independientemente de quién los utilice. Los manuales de procedimientos deben publicarse en una ubicación central y actualizarse a medida que el proceso evolucione, ya que la actualización de los manuales de procedimientos es un componente clave de un proceso de administración de cambios. También deben incluir directrices sobre la gestión de errores, las herramientas, los permisos, las excepciones y las escalaciones en caso de que se produzca un problema.

A medida que su organización madure, comience a automatizar los manuales de procedimientos. Comience con manuales de procedimientos que sean cortos y se utilicen con frecuencia. Utilice lenguajes de scripting para automatizar pasos o facilitar que se haga. A medida que automatice los primeros manuales de procedimientos, dedicará tiempo a automatizar manuales de procedimientos más complejos. Con el tiempo, la mayoría de sus manuales de procedimientos deberían estar automatizados de alguna manera.

Resultado deseado: su equipo dispone de una colección de guías paso a paso para llevar a cabo las tareas de la carga de trabajo. Los manuales de procedimientos contienen el resultado deseado, las herramientas y los permisos necesarios y las instrucciones para la gestión de errores. Se almacenan en una ubicación central (sistema de control de versiones) y se actualizan con frecuencia. Por ejemplo, sus manuales de procedimientos ofrecen a sus equipos la capacidad de supervisar,

comunicarse y responder a los eventos de AWS Health de las cuentas críticas durante las alarmas de las aplicaciones, los problemas operativos y los eventos planificados del ciclo de vida.

Patrones comunes de uso no recomendados:

- Depender de la memoria para completar cada paso de un proceso.
- Implementar manualmente los cambios sin una lista de verificación.
- Diferentes miembros del equipo llevan a cabo el mismo proceso, pero con diferentes pasos o resultados.
- Dejar que los manuales de procedimientos se desincronicen con los cambios del sistema y la automatización.

Beneficios de establecer esta práctica recomendada:

- Reducción de los índices de error en las tareas manuales.
- Las operaciones se llevan a cabo de forma coherente.
- Los nuevos miembros del equipo pueden empezar a efectuar tareas antes.
- Los manuales de procedimientos pueden automatizarse para reducir el trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los manuales de procedimientos pueden adoptar varias formas en función del nivel de madurez de su organización. Como mínimo, deben consistir en un documento de texto paso a paso. El resultado deseado debe indicarse claramente. Documente claramente los permisos o herramientas especiales necesarios. Proporcione directrices detalladas sobre la gestión de errores y las derivaciones en caso de que algo vaya mal. Indique el propietario del manual de procedimientos y publíquelo en una ubicación central. Una vez que el manual de procedimientos esté documentado, haga que otra persona de su equipo lo ejecute para validarlo. A medida que los procedimientos evolucionen, actualice sus manuales de procedimientos de acuerdo con su proceso de administración de cambios.

Sus manuales de procedimientos deben automatizarse a medida que su organización madura. Con servicios como la Automatización de AWS Systems Manager, puede transformar un texto plano en automatizaciones que pueden ejecutarse contra su carga de trabajo. Estas automatizaciones pueden ejecutarse en respuesta a eventos, lo que reduce la carga operativa para mantener su carga

de trabajo. AWS La Automatización de Systems Manager también proporciona una experiencia de diseño visual con poco código para crear manuales de automatización con mayor facilidad.

Ejemplo de cliente

AnyCompany Retail debe llevar a cabo actualizaciones del esquema de la base de datos durante implementaciones de software. El equipo de operaciones en la nube trabajó con el equipo de administración de bases de datos para crear un manual de procedimientos para implementar manualmente estos cambios. El manual de procedimientos enumeraba cada paso del proceso en forma de lista de verificación. Incluía una sección sobre la gestión de errores en caso de que algo saliera mal. Publicaron el manual de procedimientos en su wiki interna junto con sus otros manuales de procedimientos. El equipo de operaciones en la nube tiene previsto automatizar el manual de procedimientos en un futuro sprint.

Pasos para la implementación

Si no tiene un repositorio de documentos, un repositorio de control de versiones es un buen lugar para empezar a crear su biblioteca de manuales de procedimientos. Puede crear sus manuales de procedimientos con Markdown. Hemos proporcionado una plantilla de manual de procedimientos de ejemplo que puede utilizar para empezar a crear manuales de procedimientos.

```
# Runbook Title
## Runbook Info
| Runbook ID | Description | Tools Used | Special Permissions | Runbook Author | Last
 Updated | Escalation POC |
|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this runbook for? What is the desired outcome? | Tools | Permissions
 | Your Name | 2022-09-21 | Escalation Name |
## Steps
1. Step one
2. Step two
```

1. Si no tiene un repositorio de documentación o un wiki, cree un nuevo repositorio de control de versiones en su sistema de control de versiones.
2. Identifique un proceso que no tenga un manual de procedimientos. Un proceso ideal es aquel que se lleva a cabo de forma semirregular, es corto en número de pasos y tiene errores de bajo impacto.
3. En su repositorio de documentos, use la plantilla para crear un nuevo borrador de documento Markdown. Rellene el título del manual de procedimientos y los campos obligatorios en Información del manual de procedimientos.

4. En el primer paso, rellene la parte Pasos del manual de procedimientos.
5. Asigne el manual de procedimientos a un miembro del equipo. Pídales que utilicen el manual de procedimientos para validar los pasos. Si falta algo o hay que aclararlo, actualice el manual de procedimientos.
6. Publique el manual de procedimientos en su almacén de documentación interno. Una vez publicado, comuníquelo a su equipo y a otras partes interesadas.
7. Con el tiempo, creará una biblioteca de manuales de procedimientos. A medida que esa biblioteca crezca, comience a trabajar para automatizar los manuales de procedimientos.

Nivel de esfuerzo para el plan de implementación: bajo. El estándar mínimo para un manual de procedimientos es una guía de texto paso a paso. La automatización de manuales de procedimientos puede aumentar el esfuerzo de implementación.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP02 Procesos y procedimientos con propietarios identificados
- OPS07-BP04 Uso de manuales de estrategias para investigar problemas
- OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas
- OPS10-BP02 Implementación de un proceso por alerta
- OPS11-BP04 Administración de conocimientos

Documentos relacionados:

- Achieving Operational Excellence using automated playbook and runbook
- AWS Systems Manager: Creación de sus propios manuales de procedimientos
- Migration playbook for AWS large migrations - Task 4: Improving your migration runbooks
- Use AWS Systems Manager Automation runbooks to resolve operational tasks

Videos relacionados:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response
- How to automate IT Operations on AWS | Amazon Web Services
- Integrate Scripts into AWS Systems Manager

Ejemplos relacionados:

- Well-Architected Labs: Automating operations with Playbooks and Runbooks
- AWS Entrada en el blog de : Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- AWS Systems Manager: tutoriales de Automation
- AWS Systems Manager: Restauración de un volumen raíz a partir de la última instantánea del manual de procedimientos
- Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake
- Gitlab: Runbooks
- Rubix: una biblioteca de Python para crear manuales de procedimientos en cuadernos de Jupyter
- Uso del Generador de documentos para crear un manual de procedimientos

Servicios relacionados:

- AWS Systems Manager Automation

OPS07-BP04 Uso de manuales de estrategias para investigar problemas

Los manuales de estrategias son guías paso a paso que se utilizan para investigar un incidente. Cuando se producen incidentes, se usan para investigar, determinar el impacto e identificar la causa raíz. Los manuales de estrategias se utilizan en diversas situaciones, desde implementaciones erróneas hasta incidentes de seguridad. En numerosos casos, identifican la causa raíz para la que se usa un manual de procedimientos para mitigarla. Las guías de estrategias son un componente esencial de los planes de respuesta a incidentes de su organización.

Un buen manual de estrategias tiene varias características clave. Orienta al usuario, paso a paso, a través del proceso de descubrimiento. Viéndolo desde fuera, ¿qué pasos debería seguir alguien para diagnosticar un incidente? Defina de forma clara en el manual de estrategias si se necesitan herramientas especiales o permisos de alto nivel en ella. El hecho de contar con un plan de comunicación para informar a las partes interesadas sobre el estado de la investigación es un componente clave. En las situaciones en las que no se pueda identificar la causa raíz, la guía de estrategias debe tener un plan de derivación. Si se identifica la causa raíz, la guía de estrategias debe señalar un manual de procedimientos que describa cómo resolverla. Los manuales de estrategias deben almacenarse de forma centralizada y se debe hacer un mantenimiento periódico

de ellos. Si se utilizan para alertas específicas, facilite a su equipo indicaciones sobre cada guía de estrategias en cada alerta.

A medida que madure su organización, automatice los manuales de estrategias. Empiece con manuales de estrategias que cubran incidentes de poco riesgo. Utilice scripting para automatizar los pasos de descubrimiento. Asegúrese de que dispone de manuales de procedimientos complementarios para mitigar las causas raíz más habituales.

Resultado deseado: su organización dispone de manuales de estrategias para incidentes comunes. Dichos manuales de estrategias se almacenan en una ubicación central y están a disposición de los miembros del equipo. y se actualizan con frecuencia. Se crean manuales de procedimientos complementarios para cualquier causa raíz conocida.

Patrones comunes de uso no recomendados:

- No existe una forma estándar de investigar un incidente.
- Los miembros del equipo confían en la memoria muscular o en el conocimiento institucional para solucionar una implementación con errores.
- Los nuevos miembros del equipo aprenden a investigar los problemas con el método de ensayo y error.
- Las prácticas recomendadas para investigar los problemas no se comparten entre los equipos.

Beneficios de establecer esta práctica recomendada:

- Los manuales de estrategias impulsan sus esfuerzos para mitigar los incidentes.
- Los distintos miembros del equipo pueden utilizar el mismo manual de estrategias para identificar la causa raíz de forma coherente.
- Las causas raíz conocidas pueden tener manuales de procedimientos desarrollados para ellas, lo que acelera el tiempo de recuperación.
- Los manuales de estrategias permiten a los miembros del equipo empezar a contribuir antes.
- Los equipos pueden escalar sus procesos con manuales de estrategias repetibles.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

### Guía para la implementación

La forma de crear y utilizar las guías de estrategias depende de la madurez de su organización. Si es la primera vez que utiliza la nube, cree guías de estrategias en formato de texto en un repositorio de documentos central. A medida que madure su organización, los manuales de estrategias pueden semiautomatizarse con lenguajes de scripting como Python. Estos scripts pueden ejecutarse en un cuaderno de Jupyter para acelerar el descubrimiento. Las organizaciones avanzadas cuentan con manuales de estrategias completamente automatizados para los problemas más habituales que se solucionan de forma automática con manuales de procedimientos.

Elabore una lista de incidentes comunes que afectan a la carga de trabajo para empezar a crear los manuales de estrategias. Como punto de partida, elija manuales de estrategias para incidentes con poco riesgo y en los que la causa raíz se haya reducido a unos pocos problemas. Una vez que disponga de manuales de estrategias para las situaciones más sencillas, continúe con las de mayor riesgo o cuya causa raíz no se conozca bien.

Sus manuales de estrategias en texto deben automatizarse a medida que su organización madura. Con servicios como la Automatización de AWS Systems Manager, se puede transformar un texto plano en automatizaciones. Estas automatizaciones pueden ejecutarse en la carga de trabajo para acelerar las investigaciones. Se pueden activar en respuesta a los incidentes, lo que reduce el tiempo medio para descubrir y resolver los incidentes.

Los clientes pueden usar el Administrador de incidentes de AWS Systems Manager para responder a los incidentes. Este servicio proporciona una interfaz única para clasificar los incidentes, informar a las partes interesadas durante el descubrimiento y la mitigación y colaborar durante todo el incidente. Utiliza las automatizaciones de AWS para acelerar la detección y la recuperación.

### Ejemplo de cliente

La empresa AnyCompany Retail se ha visto afectada por un incidente de producción. El ingeniero de guardia utilizó un manual de estrategias para investigar el problema. A medida que iba siguiendo los pasos, informaba a las partes interesadas clave identificadas en el manual de estrategias. El ingeniero identificó la causa raíz como una condición de secuencia en un servicio backend. Mediante un manual de procedimientos, el ingeniero relanzó el servicio, con lo que AnyCompany Retail volvió a estar en línea.

### Pasos para la implementación

Si no tiene un repositorio de documentos, le sugerimos que cree uno de control de versiones para su biblioteca de manuales de estrategias. Puede crear los manuales de estrategias con Markdown, que

es compatible con la mayoría de los sistemas de automatización de este tipo de manuales. Si está empezando desde cero, utilice la siguiente plantilla de guía de estrategias de ejemplo.

```
# Playbook Title
## Playbook Info
| Playbook ID | Description | Tools Used | Special Permissions | Playbook Author | Last
 Updated | Escalation POC | Stakeholders | Communication Plan |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this playbook for? What incident is it used for? | Tools |
 Permissions | Your Name | 2022-09-21 | Escalation Name | Stakeholder Name | How will
 updates be communicated during the investigation? |
## Steps
1. Step one
2. Step two
```

1. Si no tiene un repositorio de documentos o un wiki, cree un nuevo repositorio de control de versiones para los manuales de estrategias en su sistema de control de versiones.
2. Identifique un problema común que requiera una investigación. Este debería ser un escenario en el que la causa raíz se limita a unos pocos problemas y la resolución conlleva poco riesgo.
3. Con la plantilla de Markdown, rellene la sección Nombre del manual de estrategias y los campos de información de la guía de estrategias.
4. Rellene los pasos de solución adicionales. Indique con la mayor claridad posible las acciones que se deben llevar a cabo o las áreas que debe investigar.
5. Entregue a un miembro del equipo la guía de estrategias y pídale que la revise para validarla. Si falta algo o no está claro, actualice la guía de estrategias.
6. Publique el manual de estrategias en el repositorio de documentos e informe al equipo y a las partes interesadas.
7. Esta biblioteca de manuales de estrategias crecerá a medida que vaya agregando más guías. Una vez que tenga varias guías de estrategias, empiece a automatizarlas con herramientas como las Automatizaciones de AWS Systems Manager para sincronizar la automatización y las guías de estrategias.

Nivel de esfuerzo para el plan de implementación: bajo. Los manuales de estrategias deben ser documentos de texto almacenados en una ubicación central. Las organizaciones más maduras se inclinarán por la automatización de los manuales de estrategias.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP02 Procesos y procedimientos con propietarios identificados
- OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos
- OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas
- OPS10-BP02 Implementación de un proceso por alerta
- OPS11-BP04 Administración de conocimientos

Documentos relacionados:

- Achieving Operational Excellence using automated playbook and runbook
- AWS Systems Manager: Creación de sus propios manuales de procedimientos
- Use AWS Systems Manager Automation runbooks to resolve operational tasks

Videos relacionados:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response (SEC318-R1)
- AWS Systems Manager Incident Manager - AWS Virtual Workshops
- Integrate Scripts into AWS Systems Manager

Ejemplos relacionados:

- AWS Customer Playbook Framework
- AWS Systems Manager: tutoriales de Automation
- Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake
- Rubix: una biblioteca de Python para crear manuales de procedimientos en cuadernos de Jupyter
- Uso del Generador de documentos para crear un manual de procedimientos

Servicios relacionados:

- AWS Systems Manager Automation
- AWS Systems Manager Incident Manager

### OPS07-BP05 Toma de decisiones fundamentadas para implementar sistemas y cambios

Disponga de procesos en caso de cambios fructíferos e infructuosos de la carga de trabajo. Un pre mortem es un ejercicio en el que un equipo simula un error para desarrollar estrategias de mitigación. haga ensayos de errores pre mortem para anticipar el fracaso y crear procedimientos cuando sea apropiado. Evalúe las ventajas y los riesgos de implementar cambios en la carga de trabajo. Verifique que todos los cambios cumplan con la gobernanza.

Resultado deseado:

- Tomará decisiones informadas cuando implemente cambios en la carga de trabajo.
- Los cambios cumplirán con la gobernanza.

Patrones comunes de uso no recomendados:

- implementación de un cambio en la carga de trabajo sin un proceso para gestionar una implementación infructuosa.
- Hacer cambios en el entorno de producción que incumplen los requisitos de gobernanza.
- Implementación de una nueva versión de la carga de trabajo sin establecer una línea de referencia para la utilización de recursos.

Beneficios de establecer esta práctica recomendada:

- Estará preparado para cambios infructuosos en su carga de trabajo.
- Los cambios en la carga de trabajo cumplirán con las políticas de gobernanza.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Utilice ejercicios pre mortem para desarrollar procesos en caso de cambios infructuosos. Documente los procesos para los cambios infructuosos. Asegúrese de que todos los cambios se ajusten a la gobernanza. Evalúe las ventajas y los riesgos de implementar cambios en la carga de trabajo.

Ejemplo de cliente

AnyCompany Retail lleva a cabo con regularidad ejercicios pre mortem para validar los procesos en caso de cambios infructuosos. Documenta los procesos en una wiki compartida y la actualiza con frecuencia. Todos los cambios se ajustan a los requisitos de gobernanza.

Pasos para la implementación

1. Tome decisiones fundamentadas cuando implemente cambios en la carga de trabajo. Establezca y revise los criterios para una implementación fructífera. Desarrolle escenarios o criterios que desencadenen la reversión de un cambio. Sopese las ventajas de la implementación de cambios frente a los riesgos de un cambio infructuoso.
2. Verifique que todos los cambios cumplan las políticas de gobernanza.
3. Utilice ejercicios pre mortem para desarrollar planes en caso de cambios infructuosos y documentar las estrategias de mitigación. Lleve a cabo un ejercicio de simulación para modelar un cambio infructuoso y validar los procedimientos de reversión.

Nivel de esfuerzo para el plan de implementación: moderado. La implementación de una práctica de pre mortem exige la coordinación y el esfuerzo de las partes interesadas de toda la organización.

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP03 Evaluación de los requisitos de gobernanza: los requisitos de gobernanza son un factor clave para determinar si se debe implementar un cambio.
- OPS06-BP01 Planificación para hacer frente a los cambios infructuosos: establezca planes para mitigar una implementación infructuosa y utilice actividades pre mortem para validarlas.
- OPS06-BP02 Implementaciones de prueba: cada cambio de software debe probarse adecuadamente antes de su implementación, a fin de reducir los defectos en producción.
- OPS07-BP01 Garantía de la capacidad del personal: disponer de suficiente personal formado para atender la carga de trabajo es esencial para tomar una decisión informada sobre la implementación de un cambio en el sistema.

Documentos relacionados:

- Amazon Web Services: Risk and Compliance
- Modelo de responsabilidad compartida de AWS

- Governance in the Nube de AWS: The Right Balance Between Agility and Safety

OPS07-BP06 Creación de planes de asistencia para cargas de trabajo de producción

Facilite la asistencia de cualquier software y servicio del que dependa su carga de trabajo de producción. Seleccione un nivel de asistencia adecuado para satisfacer sus necesidades de nivel de servicio de producción. Los planes de asistencia para estas dependencias son necesarios en caso de que se produzca una interrupción del servicio o un problema con el software. Documente los planes de asistencia y cómo solicitar asistencia de todos los proveedores de servicios y software. Implemente mecanismos que verifiquen que los puntos de asistencia de los contactos se mantengan actualizados.

Resultado deseado:

- Implemente planes de asistencia para el software y los servicios de los que dependen las cargas de trabajo de producción.
- Elija un plan de asistencia adecuado en función de las necesidades del nivel de servicio.
- Documente los planes de asistencia, los niveles de asistencia y la forma de solicitarla.

Patrones comunes de uso no recomendados:

- No dispone de un plan de asistencia para un proveedor de software fundamental. Su carga de trabajo se ve afectada por su proveedor y no puede hacer nada para acelerar una solución u obtener actualizaciones puntuales de él.
- Un desarrollador que era el principal punto de contacto para un proveedor de software ha abandonado la empresa. No puede ponerse en contacto directamente con el equipo de asistencia del proveedor. Debe dedicar tiempo a investigar y recorrer los sistemas de contacto genéricos, lo que aumenta el tiempo necesario para responder cuando sea necesario.
- Se produce una interrupción de la producción con un proveedor de software. No hay documentación sobre cómo presentar un caso de asistencia.

Beneficios de establecer esta práctica recomendada:

- Con el nivel de asistencia adecuado, podrá obtener una respuesta en el plazo necesario para satisfacer las necesidades de servicio.
- Como cliente con asistencia puede remitir a un nivel superior si hay problemas de producción.

- Los proveedores de software y servicios pueden ayudar en la resolución de problemas durante un incidente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Facilite planes de asistencia de cualquier proveedor de software y servicio del que dependa su carga de trabajo de producción. Configure planes de asistencia adecuados para satisfacer las necesidades de nivel de servicio. Para los clientes de AWS, esto significa activar AWS Business Support o superior en cualquier cuenta en la que tenga cargas de trabajo de producción. Reúnase con los proveedores de asistencia con regularidad para obtener información actualizada sobre las ofertas de asistencia, los procesos y los contactos. Documente cómo solicitar asistencia a los proveedores de software y servicios, incluida la forma de remitir a un nivel superior si se produce una interrupción. Implemente mecanismos para mantener actualizados los contactos de asistencia.

Ejemplo de cliente

En AnyCompany Retail, todas las dependencias de software y servicios comerciales disponen de planes de asistencia. Por ejemplo, tienen activado AWS Enterprise Support en todas las cuentas con cargas de trabajo de producción. Cualquier desarrollador puede abrir un caso de asistencia cuando surja un problema. Hay una página wiki con información sobre cómo solicitar asistencia, a quién notificarlo y las prácticas recomendadas para agilizar un caso.

Pasos para la implementación

1. Colabore con las partes interesadas de su organización para identificar a los proveedores de software y servicios en los que se basa su carga de trabajo. Documente estas dependencias.
2. Determine las necesidades de nivel de servicio de su carga de trabajo. Seleccione un plan de asistencia que se ajuste a ellas.
3. Para el software y los servicios comerciales, establezca un plan de asistencia con los proveedores.
   a. Al suscribirse a AWS Business Support o un plan superior en todas las cuentas de producción, disfrutará de tiempos de respuesta más rápidos por parte de AWS Support, lo que resulta muy recomendable. Si no dispone de Premium Support, deberá tener un plan de acción para administrar los problemas que requieran la ayuda de AWS Support. AWS Support le ofrece una combinación de herramientas, tecnología, personal y programas diseñados para ayudarlo de forma proactiva a optimizar el rendimiento, rebajar los costos e innovar rápidamente. Además,

AWS Business Support proporciona beneficios adicionales, incluido el acceso de API a AWS Trusted Advisor y AWS Healthpara la integración mediante programación con los sistemas, junto con otros métodos de acceso como la Consola de administración de AWS y los canales de Amazon EventBridge.

4. Documente el plan de asistencia en su herramienta de administración de conocimientos. Incluya la forma de solicitar asistencia, a quién notificar si se presenta un caso de asistencia y cómo derivar el caso a un nivel superior durante un incidente. Un wiki es un buen mecanismo para que cualquiera pueda llevar a cabo las actualizaciones necesarias en la documentación cuando tenga conocimiento de cambios en los procesos de asistencia o en los contactos.

Nivel de esfuerzo para el plan de implementación: bajo. La mayoría de los proveedores de software y servicios ofrecen planes de asistencia opcionales. Al documentar y compartir las prácticas recomendadas de asistencia en su sistema de administración de conocimientos, verifica que su equipo sabe qué hacer cuando se produce un problema de producción.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP02 Procesos y procedimientos con propietarios identificados

Documentos relacionados:

- AWS Support Plans

Servicios relacionados:

- AWS Business Support
- AWS Enterprise Support

## Operación

Preguntas

- OPS 8. ¿Cómo utiliza la observabilidad de la carga de trabajo en su organización?
- OPS 9. ¿Cómo hace para comprender el estado de las operaciones?
- OPS 10. ¿Cómo administra la carga de trabajo y los eventos de operaciones?

## OPS 8. ¿Cómo utiliza la observabilidad de la carga de trabajo en su organización?

Recurra a la observabilidad para garantizar un estado óptimo de la carga de trabajo. Utilice métricas, registros y rastros pertinentes para obtener una visión integral del rendimiento de su carga de trabajo y abordar los problemas de manera eficiente.

Prácticas recomendadas

- OPS08-BP01 Análisis de las métricas de la carga de trabajo
- OPS08-BP02 Análisis de los registros de la carga de trabajo
- OPS08-BP03 Análisis de los rastreos de la carga de trabajo
- OPS08-BP04 Creación de alertas procesables
- OPS08-BP05 Creación de paneles

### OPS08-BP01 Análisis de las métricas de la carga de trabajo

Después de implementar la telemetría de la aplicación, analice periódicamente las métricas recopiladas. Si bien la latencia, las solicitudes, los errores y la capacidad (o las cuotas) proporcionan información sobre el rendimiento del sistema, es fundamental dar prioridad la revisión de las métricas de resultados empresariales. Esto garantiza que tome decisiones basadas en datos alineadas con sus objetivos empresariales.

Resultado deseado: información veraz sobre el rendimiento de la carga de trabajo que genera decisiones basadas en datos y garantiza la alineación con los objetivos empresariales.

Patrones comunes de uso no recomendados:

- Analizar las métricas de forma aislada sin tener en cuenta su impacto en los resultados empresariales.
- Confiar de forma excesiva en las métricas técnicas y, al mismo tiempo, dejar de lado las métricas empresariales.
- Revisar infrecuentemente las métricas, lo que hace que se pierdan oportunidades de toma de decisiones en tiempo real.

Beneficios de establecer esta práctica recomendada:

- Comprensión mejorada de la correlación entre el rendimiento técnico y los resultados empresariales.

- Proceso de toma de decisiones mejorado basado en datos en tiempo real.
- Identificación y mitigación proactivas de los problemas antes de que afecten a los resultados empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Utilice herramientas como Amazon CloudWatch para llevar a cabo análisis de métricas. Los servicios de AWS como detección de anomalías de CloudWatch y Amazon DevOps Guru pueden utilizarse para detectar anomalías, especialmente cuando se desconocen los umbrales estáticos o cuando los patrones de comportamiento son más adecuados para la detección de anomalías.

Pasos para la implementación

1. Análisis y revisión: revise e interprete periódicamente las métricas de carga de trabajo.
   a. Priorice las métricas de resultados empresariales sobre las métricas puramente técnicas.
   b. Comprenda la importancia de los picos, las caídas o los patrones en sus datos.
2. Uso de Amazon CloudWatch: utilice Amazon CloudWatch para obtener una vista centralizada y un análisis exhaustivo.
   a. Configure paneles de CloudWatch para visualizar sus métricas y compararlas a lo largo del tiempo.
   b. Utilice percentiles de CloudWatch para obtener una vista clara de la distribución de métricas, lo que puede ayudar a definir los SLA y comprender los valores atípicos.
   c. Configure la detección de anomalías de CloudWatch para identificar patrones inusuales sin depender de umbrales estáticos.
   d. Implemente la observabilidad entre cuentas de CloudWatch para supervisar y solucionar problemas en las aplicaciones que abarcan varias cuentas de una región.
   e. Utilice Información de métricas de CloudWatch para consultar y analizar datos de métricas en cuentas y regiones, identificando tendencias y anomalías.
   f. Aplique calculadora de métricas para transformar, agregar o hacer cálculos en sus métricas a fin de obtener información más detallada.
3. Uso de Amazon DevOps Guru: integre Amazon DevOps Guru por su detección de anomalías mejorada con machine learning para identificar los primeros signos de problemas operativos en sus aplicaciones sin servidor y solucionarlos antes de que afecten a sus clientes.

4. Optimización basada en información: tome decisiones fundamentadas en función de su análisis de métricas para ajustar y mejorar sus cargas de trabajo.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS04-BP02 Implementación de telemetría de aplicaciones

Documentos relacionados:

- The Wheel Blog: Emphasizing the importance of continually reviewing metrics
- Percentiles are important
- Uso de AWS Cost Anomaly Detection
- Observabilidad entre cuentas de CloudWatch
- Consulte sus métricas con Información de métricas de CloudWatch

Videos relacionados:

- Enable Cross-Account Observability in Amazon CloudWatch
- Introduction to Amazon DevOps Guru
- Continuously Analyze Metrics using AWS Cost Anomaly Detection

Ejemplos relacionados:

- One Observability Workshop
- Gaining operation insights with AIOps using Amazon DevOps Guru

OPS08-BP02 Análisis de los registros de la carga de trabajo

El análisis periódico de los registros de la carga de trabajo es esencial para adquirir una comprensión exhaustiva de los aspectos operativos de su aplicación. Al examinar, visualizar e interpretar de

manera eficiente los datos de registro, puede optimizar continuamente el rendimiento y la seguridad de las aplicaciones.

Resultado deseado: amplios conocimientos sobre el comportamiento y las operaciones de las aplicaciones derivados de un análisis exhaustivo de los registros, lo que garantiza la detección y mitigación proactivas de los problemas.

Patrones comunes de uso no recomendados:

- Descuidar el análisis de los registros hasta que surja un problema crítico.
- No utilizar el conjunto completo de herramientas disponibles para el análisis de registros, lo que significa perder información crucial.
- Confiar únicamente en la revisión manual de los registros sin utilizar las capacidades de automatización y consulta.

Beneficios de establecer esta práctica recomendada:

- Identificación proactiva de los cuellos de botella operativos, las amenazas a la seguridad y otros posibles problemas.
- Uso eficiente de los datos de registro para la optimización continua de las aplicaciones.
- Mejor comprensión del comportamiento de las aplicaciones, lo que ayuda a depurar y solucionar problemas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Registros de Amazon CloudWatch es una herramienta potente para el análisis de registros. Las características integradas, como Información de registros de CloudWatch e Información de colaboradores, hacen que el proceso de obtener información significativa de los registros sea intuitivo y eficiente.

Pasos para la implementación

1. Configuración de Registros de CloudWatch: configure aplicaciones y servicios para enviar registros a Registros de CloudWatch.
2. Uso de la detección de anomalías en los registros: utilice la detección de anomalías de Registros de Amazon CloudWatch para identificar y alertar automáticamente sobre patrones de registros

inusuales. Esta herramienta le ayuda a administrar de forma proactiva las anomalías en sus registros y a detectar posibles problemas con antelación.

3. Configuración de Información de registros de CloudWatch: use Información de registros de CloudWatch para buscar y analizar de forma interactiva los datos de registro.
   a. Cree consultas para extraer patrones, visualizar datos de registro y obtener información procesable.
   b. Utilice el análisis de patrones de Información de registros de CloudWatch para analizar y visualizar los patrones de registro frecuentes. Esta característica le ayuda a conocer las tendencias operativas comunes y los posibles valores atípicos en sus datos de registro.
   c. Utilice la comparativa (diff) de Registros de CloudWatch para llevar a cabo análisis diferenciales entre distintos periodos de tiempo o entre distintos grupos de registros. Utilice esta capacidad para identificar los cambios y evaluar su repercusión en el rendimiento o el comportamiento del sistema.
4. Supervisión de los registros en tiempo real con Live Tail: utilice Live Tail de Registros de Amazon CloudWatch para ver los datos de registro en tiempo real. Puede supervisar activamente las actividades operativas de su aplicación a medida que se producen, lo que proporciona una visibilidad inmediata del rendimiento del sistema y de los posibles problemas.
5. Aproveche Información de colaboradores: utilice Información de colaboradores de CloudWatch para identificar a los principales interlocutores en dimensiones de alta cardinalidad, como las direcciones IP o los agentes de usuario.
6. Implementación de filtros de métricas de Registros de CloudWatch: configure los filtros de métricas de Registros de CloudWatch para convertir los datos de registro en métricas procesables. Esto le permite configurar alarmas o analizar más a fondo los patrones.
7. Implementación de la observabilidad entre cuentas de CloudWatch: supervise y solucione problemas en las aplicaciones que abarcan varias cuentas de una región.
8. Revisión y perfeccionamiento periódicos: revise periódicamente sus estrategias de análisis de registros para recoger toda la información pertinente y optimizar continuamente el rendimiento de las aplicaciones.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento

- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS08-BP01 Análisis de las métricas de la carga de trabajo

Documentos relacionados:

- Analyzing Log Data with CloudWatch Logs Insights
- Using CloudWatch Contributor Insights
- Creating and Managing CloudWatch Log Metric Filters

Videos relacionados:

- Analyze Log Data with CloudWatch Logs Insights
- Use CloudWatch Contributor Insights to Analyze High-Cardinality Data

Ejemplos relacionados:

- CloudWatch Logs Sample Queries
- One Observability Workshop

OPS08-BP03 Análisis de los rastreos de la carga de trabajo

El análisis de los datos de rastreo es crucial para lograr una visión integral del recorrido operativo de una aplicación. Al visualizar y comprender las interacciones entre varios componentes, se puede ajustar el rendimiento, identificar los cuellos de botella y mejorar las experiencias de los usuarios.

Resultado deseado: logre una visibilidad clara de las operaciones distribuidas de su aplicación, lo que permite una resolución de problemas más rápida y una mejor experiencia del usuario.

Patrones comunes de uso no recomendados:

- Pasar por alto los datos de rastreo y confiar únicamente en los registros y las métricas.
- No se correlacionan los datos de rastreo con los registros asociados.
- Hacer caso omiso de las métricas derivadas de los rastreos, como la latencia y las tasas de errores.

Beneficios de establecer esta práctica recomendada:

- Mejore la solución de problemas y reduzca el tiempo medio de resolución (MTTR).
- Obtenga información sobre las dependencias y su impacto.
- Identifique y corrija rápidamente los problemas de rendimiento.
- Utilice las métricas derivadas de los rastreos para tomar decisiones informadas.
- Mejore la experiencia del usuario mediante interacciones de componentes optimizadas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

AWS X-Ray ofrece un conjunto completo para el análisis de datos de rastreo, que proporciona una visión integral de las interacciones del servicio, supervisa las actividades de los usuarios y detecta problemas de rendimiento. Características como ServiceLens, X-Ray Insights, X-Ray Analytics y Amazon DevOps Guru mejoran la profundidad de la información procesable derivada de los datos de rastreo.

Pasos para la implementación

Los siguientes pasos ofrecen un enfoque estructurado para implementar de manera eficaz el análisis de datos de rastreo mediante servicios de AWS:

1. Integración de AWS X-Ray: asegúrese de que X-Ray esté integrado con sus aplicaciones para obtener datos de rastreo.
2. Análisis de las métricas de X-Ray: profundice en las métricas obtenidas de los rastreos de X-Ray, como la latencia, las tasas de solicitudes, las tasas de errores y las distribuciones del tiempo de respuesta mediante el mapa de servicios para supervisar el estado de las aplicaciones.
3. Uso de ServiceLens: aproveche el mapa de ServiceLens para mejorar la observabilidad de sus servicios y aplicaciones. Esto permite la visualización integrada de rastreos, métricas, registros, alarmas y otra información de estado.
4. Activación de X-Ray Insights:
    a. Active X-Ray Insights para la detección automática de anomalías en los rastreos.
    b. Examine la información para identificar patrones y determinar las causas raíz, como el aumento de tasas de errores o latencias.
    c. Consulte el cronograma de información para obtener un análisis cronológico de los problemas detectados.

5. Uso de X-Ray Analytics: X-Ray Analytics le permite explorar a fondo los datos de rastreo, identificar patrones y extraer información.
6. Uso de grupos en X-Ray: cree grupos en X-Ray para filtrar los rastreos en función de criterios como la alta latencia, lo que permite un análisis más específico.
7. Integración de Amazon DevOps Guru: utilice Amazon DevOps Guru para beneficiarse de los modelos de machine learning que identifican anomalías operativas en los rastreos.
8. Uso de CloudWatch Synthetics: utilice CloudWatch Synthetics para crear canarios para supervisar continuamente sus puntos de enlace y flujos de trabajo. Estos canarios pueden integrarse con X-Ray para proporcionar datos de rastreo para un análisis en profundidad de las aplicaciones que se están probando.
9. Uso de Real User Monitoring (RUM): con AWS X-Ray y CloudWatch RUM, puede analizar y depurar la ruta de solicitud a partir de los usuarios finales de su aplicación y hasta los servicios administrados de AWS posteriores. Eso le ayuda a identificar las tendencias de latencia y los errores que afectan a sus usuarios finales.
10. Correlación con registros: correlacione los datos de seguimiento con los registros relacionados en la vista de rastreo de X-Ray para obtener una perspectiva detallada del comportamiento de las aplicaciones. Esto le permite ver los eventos de registro directamente asociados con las transacciones rastreadas.
11. Implementación de la observabilidad entre cuentas de CloudWatch: supervise y solucione problemas en las aplicaciones que abarcan varias cuentas de una región.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS08-BP01 Análisis de las métricas de la carga de trabajo
- OPS08-BP02 Análisis de los registros de la carga de trabajo

Documentos relacionados:

- Using ServiceLens to Monitor Application Health
- Exploring Trace Data with X-Ray Analytics
- Detecting Anomalies in Traces with X-Ray Insights

- Continuous Monitoring with CloudWatch Synthetics

Videos relacionados:

- Analyze and Debug Applications Using Amazon CloudWatch Synthetics & AWS X-Ray
- Use AWS X-Ray Insights

Ejemplos relacionados:

- One Observability Workshop
- Implementación de X-Ray con AWS Lambda
- Plantillas de canarios de CloudWatch Synthetics

OPS08-BP04 Creación de alertas procesables

Es crucial detectar y responder rápidamente a las desviaciones en el comportamiento de su aplicación. Es especialmente vital reconocer cuándo están en peligro los resultados basados en los indicadores clave de rendimiento (KPI) o cuándo surgen anomalías inesperadas. Basar las alertas en los KPI garantiza que las señales que reciba estén directamente relacionadas con el impacto empresarial u operativo. Este enfoque de alertas procesables promueve respuestas proactivas y ayuda a mantener el rendimiento y la fiabilidad del sistema.

Resultado deseado: reciba alertas oportunas, pertinentes y procesables para identificar y mitigar rápidamente los posibles problemas, especialmente cuando los resultados de los KPI están en peligro.

Patrones comunes de uso no recomendados:

- Configurar demasiadas alertas que no son críticas, lo que provoca un exceso de alertas.
- No dar prioridad a las alertas en función de los KPI, lo que dificulta la comprensión del impacto empresarial de los problemas.
- No abordar las causas raíz, lo que genera alertas repetitivas sobre el mismo problema.

Beneficios de establecer esta práctica recomendada:

- Se ha reducido el exceso de alertas al poner el foco en las alertas pertinentes y procesables.

- Se ha mejorado el tiempo de actividad y la fiabilidad del sistema gracias a la detección y mitigación proactivas de problemas.
- Se ha mejorado la colaboración en equipo y se ha agilizado la resolución de problemas mediante la integración con herramientas de alerta y comunicación populares.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para crear un mecanismo de alerta eficaz, es fundamental utilizar métricas, registros y datos de rastreo que indiquen cuándo los resultados basados en los KPI están en peligro o se detectan anomalías.

Pasos para la implementación

1. Definición de los indicadores clave de rendimiento (KPI): identifique los KPI de su aplicación. Las alertas deben estar vinculadas a estos KPI para reflejar el impacto empresarial con precisión.
2. Implementación de la detección de anomalías:
   - Uso de la detección de anomalías de Amazon CloudWatch: configure la detección de anomalías de Amazon CloudWatch para detectar automáticamente patrones inusuales, lo que le ayuda a generar alertas únicamente para anomalías auténticas.
   - Uso de AWS X-Ray Insights:
     a. Configure X-Ray Insights para detectar anomalías en los datos de rastreo.
     b. Configure las notificaciones de X-Ray Insights para recibir alertas sobre los problemas detectados.
   - Integración con Amazon DevOps Guru:
     a. Use Amazon DevOps Guru por sus capacidades de machine learning para detectar anomalías operativas con los datos existentes.
     b. Vaya a la configuración de notificaciones en DevOps Guru para configurar alertas de anomalías.
3. Implementación de alertas procesables: diseñe alertas que proporcionen la información adecuada para tomar medidas de inmediato.
   1. Supervise los eventos de AWS Health con las reglas de Amazon EventBridge o intégrelos mediante programación con la API de AWS Health para automatizar las acciones cuando reciba eventos de AWS Health. Puede tratarse de acciones generales, como el envío de todos los mensajes de eventos del ciclo de vida planificado a una interfaz de chat, o de acciones

específicas, como el inicio de un flujo de trabajo en una herramienta de administración de servicios de TI.

4. Reducción de la fatiga de alertas: minimice las alertas no críticas. Cuando los equipos se sienten abrumados porque reciben numerosas alertas insignificantes, podrían dejar pasar problemas críticos, lo que disminuye la eficacia general del mecanismo de alertas.
5. Configuración de alarmas compuestas: utilice alarmas compuestas de Amazon CloudWatch para consolidar varias alarmas.
6. Integración con herramientas de alerta: incorpore herramientas como Ops Genie y PagerDuty.
7. Participación de Amazon Q Developer en las aplicaciones de chat: integre Amazon Q Developer en las aplicaciones de chat para transmitir alertas a Amazon Chime, Microsoft Teams y Slack.
8. Alerta basada en registros: utilice filtros de métricas de registro en CloudWatch para crear alarmas basadas en eventos de registro específicos.
9. Revisión e iteración: revisite y perfeccione periódicamente las configuraciones de las alertas.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS04-BP03 Implementación de telemetría de la experiencia del usuario
- OPS04-BP04 Implementación de telemetría de dependencias
- OPS04-BP05 Implementación de rastreo distribuido
- OPS08-BP01 Análisis de las métricas de la carga de trabajo
- OPS08-BP02 Análisis de los registros de la carga de trabajo
- OPS08-BP03 Análisis de los rastreos de la carga de trabajo

Documentos relacionados:

- Uso de las alarmas de Amazon CloudWatch
- Crear una alarma compuesta
- Crear una alarma de CloudWatch en función de la detección de anomalías

- DevOps Guru Notifications
- X-ray insights notifications
- Monitoree, opere y resuelva problemas en sus recursos de AWS con ChatOps interactivos
- Amazon CloudWatch Integration Guide | PagerDuty
- Integrate Opsgenie with Amazon CloudWatch

Videos relacionados:

- Create Composite Alarms in Amazon CloudWatch
- Información general de Amazon Q Developer en aplicaciones de chat
- AWS On Air ft. Mutative Commands in Amazon Q Developer in chat applications

Ejemplos relacionados:

- Alarms, incident management, and remediation in the cloud with Amazon CloudWatch
- Tutorial: Creating an Amazon EventBridge rule that sends notifications to Amazon Q Developer in chat applications
- One Observability Workshop

OPS08-BP05 Creación de paneles

Los paneles son la perspectiva centrada en las personas de los datos de telemetría de sus cargas de trabajo. Si bien proporcionan una interfaz visual vital, no deben reemplazar los mecanismos de alerta, sino complementarlos. Cuando se diseñan con cuidado, no solo pueden ofrecer información rápida sobre el estado y el rendimiento del sistema, sino que también pueden presentar a las partes interesadas información en tiempo real sobre los resultados empresariales y el impacto de los problemas.

Resultado deseado:

información clara y procesable sobre el estado del sistema y la empresa mediante representaciones visuales.

Patrones comunes de uso no recomendados:

- Paneles demasiado complicados con demasiadas métricas.

- Confiar en los paneles sin alertas de detección de anomalías.
- No actualizar los paneles a medida que evolucionan las cargas de trabajo.

Beneficios de esta práctica recomendada:

- Visibilidad inmediata de las métricas y los KPI cruciales del sistema.
- Mejora de la comunicación y la comprensión de las partes interesadas.
- Información rápida sobre el impacto de los problemas operativos.

Nivel de riesgo si no se establece esta práctica recomendada: medio

Guía para la implementación

Paneles centrados en la empresa

Los paneles adaptados a los KPI empresariales implican a un mayor número de partes interesadas. Si bien es posible que estas personas no estén interesadas en las métricas del sistema, están interesadas en comprender las implicaciones empresariales de estas cifras. Un panel centrado en la empresa garantiza que todas las métricas técnicas y operativas que se supervisan y analizan estén en sintonía con los objetivos empresariales generales. Esta alineación proporciona claridad y garantiza que todo el mundo coincida en lo que es esencial y lo que no. Además, los paneles que destacan los KPI empresariales suelen ser más procesables. Las partes interesadas pueden comprender rápidamente el estado de las operaciones, las áreas que requieren atención y el impacto potencial en los resultados empresariales.

Con esto en mente, al crear sus paneles, asegúrese de que haya un equilibrio entre las métricas técnicas y los KPI empresariales. Ambos son vitales, pero se dirigen a públicos diferentes. Lo ideal sería disponer de paneles que proporcionen una visión integral del estado y el rendimiento del sistema y, al mismo tiempo, hagan hincapié en los resultados empresariales clave y sus implicaciones.

Los paneles de Amazon CloudWatch son páginas de inicio personalizables en la consola de CloudWatch que puede utilizar para supervisar sus recursos en una vista única, incluso aquellos que se reparten entre diferentes Regiones de AWS y cuentas.

Pasos para la implementación

1. Creación de un panel básico: cree un panel nuevo en CloudWatch y asígnele un nombre descriptivo.

2. Uso de los widgets de Markdown: antes de sumergirse en las métricas, use widgets de Markdown para agregar contexto textual en la parte superior de su panel de control. Debe explicar lo que cubre el panel, la importancia de las métricas representadas y también puede contener enlaces a otros paneles y herramientas de solución de problemas.
3. Creación de variables de panel: integre variables de panel cuando sea necesario para permitir vistas de panel dinámicas y flexibles.
4. Creación de widgets de métricas: agregue widgets de métricas para visualizar las diversas métricas que emite su aplicación. Adapte estos widgets para que representen de forma eficaz el estado del sistema y los resultados empresariales.
5. Consultas de Información de registros: utilice Información de registros de CloudWatch para obtener métricas procesables de sus registros y mostrar esta información en su panel de control.
6. Configuración de alarmas: integre las alarmas de CloudWatch en su panel de control para ver rápidamente cualquier métrica que supere los umbrales.
7. Uso de Información de colaboradores: integre Información de colaboradores de CloudWatch para analizar los campos de alta cardinalidad y comprender mejor a los principales contribuyentes de su recurso.
8. Diseño de widgets personalizados: para necesidades específicas que no satisfagan los widgets estándar, considere la posibilidad de crear widgets personalizados. Pueden proceder de varios orígenes de datos o representar los datos de formas únicas.
9. Utilice AWS Health: AWS Health es el origen de información sobre el estado de los recursos de Nube de AWS. Utilice Panel de AWS Health directamente o utilice los datos de AWS Health en los paneles y las herramientas propios para disponer de la información adecuada para tomar decisiones con conocimiento de causa.
10. Iteración y ajuste: a medida que evolucione la aplicación, revisite periódicamente el panel para asegurarse de que siga siendo relevante.

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS08-BP01 Análisis de las métricas de la carga de trabajo
- OPS08-BP02 Análisis de los registros de la carga de trabajo
- OPS08-BP03 Análisis de los rastreos de la carga de trabajo
- OPS08-BP04 Creación de alertas procesables

Documentos relacionados:

- La creación de paneles para la visibilidad operativa
- Uso de paneles de Amazon CloudWatch

Videos relacionados:

- Create Cross Account & Cross Region CloudWatch Dashboards
- AWS re:Invent 2021 - Gain enterprise visibility with Nube de AWS operation dashboards)

Ejemplos relacionados:

- One Observability Workshop
- Supervisión de aplicaciones con Amazon CloudWatch
- AWS Health Events Intelligence Dashboards and Insights
- Visualize AWS Health events using Amazon Managed Grafana

## OPS 9. ¿Cómo hace para comprender el estado de las operaciones?

Defina, capture y analice las métricas de las operaciones para obtener visibilidad de los eventos de operaciones y poder tomar las medidas adecuadas.

Prácticas recomendadas

- OPS09-BP01 Medición de los objetivos operativos y los KPI con métricas
- OPS09-BP02 Comunicación del estado y las tendencias para garantizar la visibilidad de la operación
- OPS09-BP03 Revisión de las métricas de las operaciones y priorización de las mejoras

### OPS09-BP01 Medición de los objetivos operativos y los KPI con métricas

Obtenga objetivos y KPI que definan el éxito de las operaciones de su organización y determine las métricas que los reflejen. Establezca líneas de base como puntos de referencia y reevalúelas periódicamente. Desarrolle mecanismos para recopilar estas métricas de los equipos para su evaluación. Las métricas de DevOps Research and Assessment (DORA) proporcionan un método de uso extendido para medir el progreso en el uso de prácticas de DevOps de entrega de software.

Resultado deseado:

- La organización publica y comparte los objetivos y los KPI de los equipos de operaciones.
- Establece métricas que reflejan estos KPI. Algunos ejemplos podrían ser:
  - Profundidad de la cola de tickets o antigüedad media de los tickets.
  - Recuento de tickets agrupado por tipo de problema.
  - Tiempo dedicado a resolver problemas con o sin un procedimiento operativo estandarizado (SOP).
  - Cantidad de tiempo empleado en recuperarse de un error producido al introducir código.
  - Volumen de llamadas

Patrones comunes de uso no recomendados:

- No se cumplen los plazos de implementación porque los desarrolladores se ven obligados a llevar a cabo tareas de solución de problemas. Los equipos de desarrollo abogan por más personal, pero no pueden indicar cuántas personas necesitan porque no se puede medir el tiempo empleado.
- Se configuró un servicio de asistencia de nivel 1 para gestionar las llamadas de los usuarios. Con el tiempo, se agregaron más cargas de trabajo, pero no se asignó personal al servicio de asistencia de nivel 1. La satisfacción de los clientes se resiente a medida que aumenta la duración de las llamadas y los problemas tardan más en resolverse, pero la administración no ve ningún indicador de ello, lo que impide tomar medidas.
- Una carga de trabajo problemática se ha transferido a un equipo de operaciones independiente para su gestión. A diferencia de otras cargas de trabajo, esta nueva carga no se suministró con la documentación y los manuales de procedimientos adecuados. Por lo tanto, los equipos dedican más tiempo a solucionar problemas y hacer frente a errores. Sin embargo, no hay métricas que lo documenten, lo que dificulta la rendición de cuentas.

Beneficios de establecer esta práctica recomendada: mientras que la supervisión de la carga de trabajo muestra el estado de nuestras aplicaciones y servicios, la supervisión de los equipos de operaciones proporciona a los propietarios información sobre los cambios que se producen entre los consumidores de esas cargas de trabajo, como los cambios en las necesidades empresariales. Mida la eficacia de estos equipos y evalúelos con respecto a los objetivos empresariales mediante la creación de métricas que puedan reflejar el estado de las operaciones. Las métricas pueden resaltar los problemas de asistencia o identificar cuándo se producen desviaciones respecto a un objetivo de nivel de servicio.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Programe tiempo con la dirección empresarial y las partes interesadas para determinar los objetivos generales del servicio. Determine cuáles deberían ser las tareas de los distintos equipos de operaciones y qué desafíos podrían presentárseles. Con estos, haga una lluvia de ideas sobre los indicadores clave de rendimiento (KPI) para reflejar los objetivos operativos. Podría ser la satisfacción del cliente, el tiempo transcurrido desde la concepción de la característica hasta la implementación, el tiempo medio de resolución de problemas o las eficiencias de costes.

A partir de los KPI, identifique las métricas y los orígenes de datos que podrían reflejar mejor estos objetivos. La satisfacción del cliente podría ser una combinación de varios indicadores, como los tiempos de espera o respuesta de las llamadas, las puntuaciones de satisfacción y los tipos de problemas planteados. Los tiempos de implementación podrían ser la suma del tiempo necesario para las pruebas y la implementación, además de las correcciones posteriores a la implementación que deban agregarse. Las estadísticas que muestran el tiempo dedicado a diferentes tipos de problemas (o el recuento de esos problemas) pueden proporcionar una panorámica de dónde se necesita un esfuerzo específico.

Recursos

Documentos relacionados:

- Quick: uso de KPI
- Uso de métricas de Amazon CloudWatch
- Creación de paneles
- How to track your cost optimization KPIs with KPI Dashboard
- Guía sobre AWS DevOps

Ejemplos relacionados:

- Monitoree el rendimiento de su entrega de software mediante herramientas nativas de AWS de monitoreo y observabilidad
- Equilibre la velocidad y la estabilidad de la implementación con las métricas de DORA
- Ejemplo de métricas operativas de MLOP en el sector de servicios financieros
- How to track your cost optimization KPIs with the KPI Dashboard

OPS09-BP02 Comunicación del estado y las tendencias para garantizar la visibilidad de la operación

Es necesario conocer el estado de sus operaciones y la dirección de sus tendencias para identificar qué resultados corren peligro, si se puede respaldar o no el trabajo adicional o los efectos que los cambios han tenido en sus equipos. Durante los eventos de operaciones, disponer de páginas de estado que los usuarios y los equipos de operaciones puedan consultar para obtener información puede reducir la presión sobre los canales de comunicación y difundir la información de forma proactiva.

Resultado deseado:

- La dirección de operaciones puede ver de un vistazo el volumen de llamadas que reciben sus equipos y las actividades que se están llevando a cabo, como las implementaciones.
- Las alertas se difunden a las partes interesadas y las comunidades de usuarios cuando se producen repercusiones en las operaciones normales.
- La dirección de la organización y las partes interesadas pueden consultar una página de estado en respuesta a una alerta o una repercusión y obtener información sobre un evento operativo, como puntos de contacto, información de tickets y tiempos de recuperación estimados.
- Los informes se ponen a disposición de la dirección y otras partes interesadas para mostrar las estadísticas de las operaciones, como el volumen de llamadas durante un periodo de tiempo, las puntuaciones de satisfacción de los usuarios, el número de entradas pendientes y su antigüedad.

Patrones comunes de uso no recomendados:

- Una carga de trabajo deja de funcionar y un servicio no está disponible. El volumen de llamadas aumenta a medida que los usuarios quieren saber qué pasa. Los administradores contribuyen al aumento del volumen de solicitudes, pues quieren saber quién está trabajando en el problema. Varios equipos de operaciones duplican sus esfuerzos al tratar de investigar.
- El interés por una nueva capacidad lleva a la reasignación de varios miembros del personal a tareas de ingeniería. No se proporcionan refuerzos y los tiempos de resolución de problemas aumentan. Esta información no se recopila, y la dirección no se da cuenta del problema hasta después de varias semanas y de que los usuarios muestren su insatisfacción.

Beneficios de establecer esta práctica recomendada: durante los eventos operativos que afectan a la empresa, se puede desperdiciar mucho tiempo y energía solicitando información a varios equipos para intentar comprender la situación. Al establecer paneles y páginas de estado ampliamente difundidos, las partes interesadas pueden obtener rápidamente información sobre si se detectó o

no un problema, quién se encarga del problema o cuándo se espera que las operaciones vuelvan a la normalidad. Esto evita que los miembros del equipo dediquen demasiado tiempo a comunicar su estado a los demás y dediquen más tiempo a abordar los problemas.

Además, los paneles y los informes pueden proporcionar información a los responsables de la toma de decisiones y a las partes interesadas para que evalúen cómo los equipos de operaciones pueden responder a las necesidades empresariales y cómo se asignan sus recursos. Esto es crucial para determinar si se cuenta con los recursos adecuados para respaldar a la empresa.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cree paneles en los que se muestren las métricas clave actuales de sus equipos de operaciones y póngalos a disposición tanto de la dirección de operaciones como de la administración.

Cree páginas de estado que se puedan actualizar rápidamente para mostrar cuándo se produce un incidente o evento, quién es el propietario y quién coordina la respuesta. Comparta en esta página todos los pasos o soluciones que los usuarios deberían tener en cuenta y difunda ampliamente la ubicación. Anime a los usuarios a comprobar primero esta ubicación cuando se enfrenten a un problema desconocido.

Recopile y proporcione informes que muestren el estado de las operaciones a lo largo del tiempo y distribúyalos entre la dirección y los responsables de la toma de decisiones para ilustrar el trabajo de operaciones junto con los desafíos y las necesidades.

Comparta con los equipos las métricas e informes que mejor reflejen los objetivos y los KPI y en qué aspectos han influido a la hora de impulsar el cambio. Dedique tiempo a estas actividades para aumentar la importancia de las operaciones dentro de los equipos y entre ellos.

Utilice AWS Health junto a paneles propios o integre eventos de AWS Health en ellos para que los equipos puedan correlacionar los problemas de la aplicación con el estado del servicio AWS.

Recursos

Prácticas recomendadas relacionadas:

- OPS09-BP01 Medición de los objetivos operativos y los KPI con métricas

Documentos relacionados:

- Measure Progress
- La creación de paneles para la visibilidad operativa

Ejemplos relacionados:

- Operaciones de datos
- How to track your cost optimization KPIs with KPI Dashboard
- La importancia de los indicadores clave de rendimiento (KPI) para las migraciones a gran escala a la nube

OPS09-BP03 Revisión de las métricas de las operaciones y priorización de las mejoras

Destinar tiempo y recursos dedicados a revisar el estado de las operaciones garantiza que atender la línea empresarial diaria siga siendo una prioridad. Reúna a la dirección de operaciones y las partes interesadas para revisar periódicamente las métricas, reafirmar o modificar las metas y los objetivos y dar prioridad a las mejoras.

Resultado deseado:

- La dirección y el personal de operaciones se reúnen periódicamente para revisar las métricas durante un periodo de informe determinado. Se comunican los desafíos, se celebran las victorias y se comparten las lecciones aprendidas.
- Las partes interesadas y la dirección empresarial reciben información periódica sobre el estado de las operaciones y se les pide su opinión sobre los objetivos, los KPI y las iniciativas futuras. Se analizan y contextualizan las compensaciones entre la prestación de servicios, las operaciones y el mantenimiento.

Patrones comunes de uso no recomendados:

- Se lanza un nuevo producto, pero los equipos de operaciones de nivel 1 y nivel 2 no cuentan con la formación adecuada para ofrecer soporte ni cuentan con personal adicional. La dirección no ve las métricas que muestran el empeoramiento de los tiempos de resolución de los tickets y el aumento del volumen de incidentes. No se toman medidas hasta que han transcurrido varias semanas, cuando el número de suscriptores comienza a caer porque los usuarios descontentos abandonan la plataforma.

- Hace mucho tiempo que existe un proceso manual para efectuar el mantenimiento de una carga de trabajo. Si bien había interés por automatizar, esta era una prioridad baja dada la poca importancia del sistema. Sin embargo, con el tiempo, el sistema ha ido ganando importancia y ahora estos procesos manuales consumen la mayor parte del tiempo de las operaciones. No hay recursos programados para proporcionar más herramientas a las operaciones, lo que provoca el agotamiento del personal a medida que aumentan las cargas de trabajo. La dirección se da cuenta cuando se les informa que el personal se va a la competencia.

Beneficios de establecer esta práctica recomendada: en algunas organizaciones, puede ser desafiante asignar el mismo tiempo y atención que se dedica a la prestación de servicios y a los nuevos productos u ofertas. Cuando esto ocurre, la línea empresarial puede resentirse a medida que el nivel de servicio esperado se deteriora lentamente. Esto se debe a que las operaciones no cambian ni evolucionan con el crecimiento de la empresa y pronto pueden quedarse rezagadas. Sin una revisión periódica de la información que recopilan las operaciones, es posible que el riesgo para la empresa solo resulte evidente cuando sea demasiado tarde. Al asignar tiempo para revisar las métricas y los procedimientos tanto entre el personal de operaciones como con la dirección, el papel crucial que desempeñan las operaciones permanece visible y los riesgos se pueden identificar mucho antes de que alcancen niveles críticos. Los equipos de operaciones obtienen una mejor perspectiva de los cambios e iniciativas empresariales inminentes, lo que permite llevar a cabo esfuerzos proactivos. La visibilidad de la dirección de las métricas de las operaciones muestra el papel que desempeñan estos equipos en la satisfacción del cliente, tanto interna como externa, y les permite sopesar mejor las opciones en función de las prioridades, o garantizar que las operaciones tengan el tiempo y los recursos para cambiar y evolucionar con las nuevas iniciativas empresariales y de carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Dedique tiempo a revisar las métricas de las operaciones entre las partes interesadas y los equipos de operaciones y a revisar los datos de los informes. Analice estos informes en el contexto de las metas y los objetivos de la organización para determinar si se están cumpliendo. Identifique los orígenes de ambigüedad en los que las metas no estén claras o en las que pueda haber conflictos entre lo que se pide y lo que se da.

Identifique dónde pueden ayudar el tiempo, las personas y las herramientas a obtener resultados operativos. Determine a qué KPI afectaría esto y cuáles deberían ser los objetivos de éxito.

Revisite todo esto periódicamente a fin de garantizar que las operaciones cuenten con los recursos suficientes para respaldar la línea empresarial.

Recursos

Documentos relacionados:

- Amazon Athena
- Referencia de métricas y dimensiones de Amazon CloudWatch
- Amazon Quick
- AWS Glue
- AWS Glue Data Catalog
- Recopilación de métricas y registros de instancias de Amazon EC2 y en los servidores en las instalaciones con el agente de Amazon CloudWatch
- Uso de métricas de Amazon CloudWatch

## OPS 10. ¿Cómo administra la carga de trabajo y los eventos de operaciones?

Prepare y valide los procedimientos de respuesta a los eventos para minimizar la interrupción de la carga de trabajo.

Prácticas recomendadas

- OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas
- OPS10-BP02 Implementación de un proceso por alerta
- OPS10-BP03 Priorización de los eventos operativos según el impacto empresarial
- OPS10-BP04 Definición de rutas de escalado
- OPS10-BP05 Definición de un plan de comunicación con los clientes en caso de eventos que afecten al servicio
- OPS10-BP06 Comunicación del estado a través de paneles
- OPS10-BP07 Automatización de las respuestas a eventos

### OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas

La capacidad de administrar eficazmente los eventos, los incidentes y los problemas es clave para mantener el estado y el rendimiento de las cargas de trabajo. Es crucial reconocer y comprender las diferencias entre estos elementos para desarrollar una estrategia eficaz de respuesta y resolución.

Establecer y seguir un proceso bien definido para cada aspecto ayuda a su equipo a administrar de forma rápida y eficaz cualquier desafío operativo que surja.

Resultado deseado: su organización administra eficazmente los eventos, incidentes y problemas operativos a través de procesos bien documentados y almacenados de forma centralizada. Estos procesos se actualizan constantemente para reflejar los cambios, agilizar la gestión y mantener una alta fiabilidad del servicio y el rendimiento de las cargas de trabajo.

Patrones comunes de uso no recomendados:

- Responde a los eventos reactivamente, en lugar de hacerlo proactivamente.
- Se adoptan enfoques incoherentes para diferentes tipos de eventos o incidentes.
- Su organización no analiza los incidentes ni aprende de ellos para evitar que ocurran en el futuro.

Beneficios de establecer esta práctica recomendada:

- Procesos de respuesta simplificados y estandarizados.
- Reducción del impacto de los incidentes en los servicios y los clientes.
- Resolución rápida de problemas.
- Mejora continua de los procesos operativos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La implementación de esta práctica recomendada implica el seguimiento de los eventos de la carga de trabajo. Dispone de procesos para gestionar las incidencias y los problemas. Los procesos se documentan, se comparten y se actualizan con frecuencia. Los problemas se identifican, se priorizan y se solucionan.

Comprensión de los eventos, los incidentes y los problemas

- Eventos: un evento consiste en observar de una acción, un suceso o un cambio de estado. Los eventos pueden planificarse o no y pueden originarse de forma interna o externa en la carga de trabajo.
- Incidentes: los incidentes son eventos que requieren una respuesta, como interrupciones no planificadas o mermas en la calidad del servicio. Representan interrupciones que requieren atención inmediata para restablecer el funcionamiento normal de las cargas de trabajo.

- Problemas: los problemas son las causas subyacentes de uno o más incidentes. Identificar y resolver los problemas implica profundizar en los incidentes para evitar que ocurran en el futuro.

Pasos para la implementación

Events (Eventos

1. Supervisión de los eventos:
   - Implemente la observabilidad y utilice la observabilidad de la carga de trabajo.
   - Las acciones de supervisión hechas por un usuario, un rol o un servicio de AWS se registran como eventos en AWS CloudTrail.
   - Responda a los cambios operativos en sus aplicaciones en tiempo real con Amazon EventBridge.
   - Evalúe, supervise y registre de forma continua los cambios en la configuración de los recursos con AWS Config.
2. Creación de procesos:
   - Desarrolle un proceso para evaluar qué eventos son importantes y requieren supervisión. Esto implica establecer umbrales y parámetros para las actividades normales y anómalas.
   - Determine los criterios por los que un evento pasa a ser un incidente. Por ejemplo, puede basarse en la gravedad, el impacto en los usuarios o la desviación del comportamiento esperado.
   - Revise periódicamente los procesos de supervisión y respuesta a los eventos. Por ejemplo, analice los incidentes pasados o ajuste los umbrales y los mecanismos de alerta.

Incidentes

1. Respuesta a los incidentes:
   - Utilice la información de las herramientas de observabilidad para identificar y responder rápidamente a los incidentes.
   - Implemente el Centro de operaciones de AWS Systems Manager para agregar, organizar y priorizar los elementos e incidentes operativos.
   - Utilice servicios como Amazon CloudWatch y AWS X-Ray para llevar a cabo análisis más detallados y solucionar problemas.
   - Considere la posibilidad de usar AWS Managed Services (AMS) para mejorar la administración de incidentes, aprovechando sus capacidades proactivas, preventivas y de detección. AMS

amplía el soporte operativo con servicios como la supervisión, la detección y respuesta a incidentes y la administración de la seguridad.

- Los clientes de Enterprise Support pueden usar Detección y respuesta a incidentes de AWS, que proporciona supervisión proactiva continua y administración de incidentes para las cargas de trabajo de producción.

2. Creación de un proceso de administración de incidentes:
   - Establezca un proceso estructurado de administración de incidentes, que incluya protocolos de comunicación, pasos para resolver problemas y roles claramente establecidos.
   - Integre la administración de incidentes con herramientas como Amazon Q Developer en las aplicaciones de chat para una respuesta y coordinación eficientes.
   - Clasifique los incidentes por gravedad, con planes de respuesta a incidentes predefinidos para cada categoría.
3. Aprenda y mejore:
   - Lleve a cabo un análisis posterior al incidente para comprender las causas fundamentales y la eficacia de la resolución.
   - Actualice y mejore continuamente los planes de respuesta en función de las revisiones y en la evolución de los procedimientos.
   - Documente y comparta las lecciones aprendidas entre los equipos para mejorar la resiliencia operativa.
   - Los clientes de Enterprise Support pueden solicitar el taller de administración de incidentes a su Technical Account Manager. Este taller guiado pone a prueba su actual plan de respuesta a incidentes y le ayuda a identificar áreas de mejora.

Problemas de

1. Identificación de los problemas:
   - Utilice los datos de incidentes anteriores para identificar patrones periódicos que pueden indicar problemas sistémicos más profundos.
   - Aproveche herramientas como AWS CloudTrail y Amazon CloudWatch para analizar las tendencias y descubrir los problemas subyacentes.
   - Involucre a equipos multifuncionales, incluidas las unidades de operaciones, desarrollo y negocios, para obtener diversas perspectivas sobre las causas raíz.
2. Creación de un proceso de administración de problemas:

- Desarrolle un proceso estructurado para la administración de problemas y céntrese en soluciones a largo plazo en lugar de en soluciones rápidas.
- Incorpore técnicas de análisis de causa raíz (RCA) para investigar y comprender las causas subyacentes de los incidentes.
- Actualice las políticas, los procedimientos y la infraestructura operativos en función de los resultados para evitar que se repitan.

3. Continuación de la mejora:
   - Fomente una cultura de aprendizaje y mejora constantes, y anime a los equipos a identificar y abordar de manera proactiva los posibles problemas.
   - Revise periódicamente los procesos y herramientas de administración de problemas para adaptarlos a la evolución de la empresa y la tecnología.
   - Comparta información y prácticas recomendadas con el resto de la organización para crear un entorno operativo más resiliente y eficiente.
4. Uso de AWS Support:
   - Utilice los recursos de asistencia de AWS, como AWS Trusted Advisor, para obtener orientación proactiva y recomendaciones de optimización.
   - Los clientes de Enterprise Support pueden acceder a programas especializados como AWS Countdown para obtener asistencia durante eventos críticos.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS04-BP02 Implementación de telemetría de aplicaciones
- OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos
- OPS07-BP04 Uso de manuales de estrategias para investigar problemas
- OPS08-BP01 Análisis de las métricas de la carga de trabajo
- OPS11-BP02 Análisis después del incidente

Documentos relacionados:

- AWS Security Incident Response Guide
- AWS Incident Detection and Response
- AWS Cloud Adoption Framework: Operations Perspective - Incident and problem management
- Incident Management in the Age of DevOps and SRE
- PagerDuty - What is Incident Management?

Videos relacionados:

- Top incident response tips from AWS
- AWS re:Invent 2022 - The Amazon Builders' Library: 25 yrs of Amazon operational excellence
- AWS re:Invent 2022 - AWS Incident Detection and Response (SUP201)
- Introducing Incident Manager from AWS Systems Manager

Ejemplos relacionados:

- AWS Proactive Services – Incident Management Workshop
- How to Automate Incident Response with PagerDuty and Administrador de incidentes de AWS Systems Manager
- Engage Incident Responders with the On-Call Schedules in Administrador de incidentes de AWS Systems Manager
- Improve the Visibility and Collaboration during Incident Handling in Administrador de incidentes de AWS Systems Manager
- Incident reports and service requests in AMS

Servicios relacionados:

- Amazon EventBridge

OPS10-BP02 Implementación de un proceso por alerta

Establecer un proceso claro y definido para cada alerta de su sistema es esencial para una administración de incidentes eficaz y eficiente. Esta práctica garantiza que cada alerta genere una respuesta específica y procesable, lo que mejora la fiabilidad y la capacidad de respuesta de sus operaciones.

Resultado deseado: cada alerta inicia un plan de respuesta específico y bien definido. Siempre que sea posible, las respuestas se automatizan, con una propiedad clara y una ruta de escalado definida. Las alertas están vinculadas a una base de conocimientos actualizada para que cualquier operador pueda responder de forma coherente y eficaz. Las respuestas son rápidas y uniformes en todos los ámbitos, lo que mejora la eficiencia y la fiabilidad operativas.

Patrones comunes de uso no recomendados:

- Las alertas no tienen un proceso de respuesta predefinido, lo que lleva a resoluciones improvisadas y tardías.
- La sobrecarga de alertas hace que se pasen por alto alertas importantes.
- Las alertas se gestionan de forma incoherente debido a la falta de propiedad y responsabilidad claras.

Beneficios de establecer esta práctica recomendada:

- Se ha reducido la fatiga de las alertas al generar solo alertas procesables.
- Disminución del tiempo medio de resolución (MTTR) de los problemas operativos.
- Disminución del tiempo medio de investigación (MTTI), lo que ayuda a reducir el MTTR.
- Mejora de la capacidad para escalar las respuestas operativas.
- Mejora de la coherencia y la fiabilidad en la gestión de los eventos operativos.

Por ejemplo, cuenta con un proceso definido para eventos de AWS Health para cuentas críticas, incluidas las alarmas de operaciones, los problemas operativos, los eventos de ciclo de vida planificados (como actualizar las versiones de Amazon EKS antes de que los clústeres se actualicen automáticamente) y ofrece a sus equipos la capacidad para monitorear activamente estos eventos, comunicarse y responder al respecto. Estas acciones lo ayudan a evitar las interrupciones del servicio causadas por cambios en AWS o mitigarlas más rápidamente cuando se producen problemas inesperados.

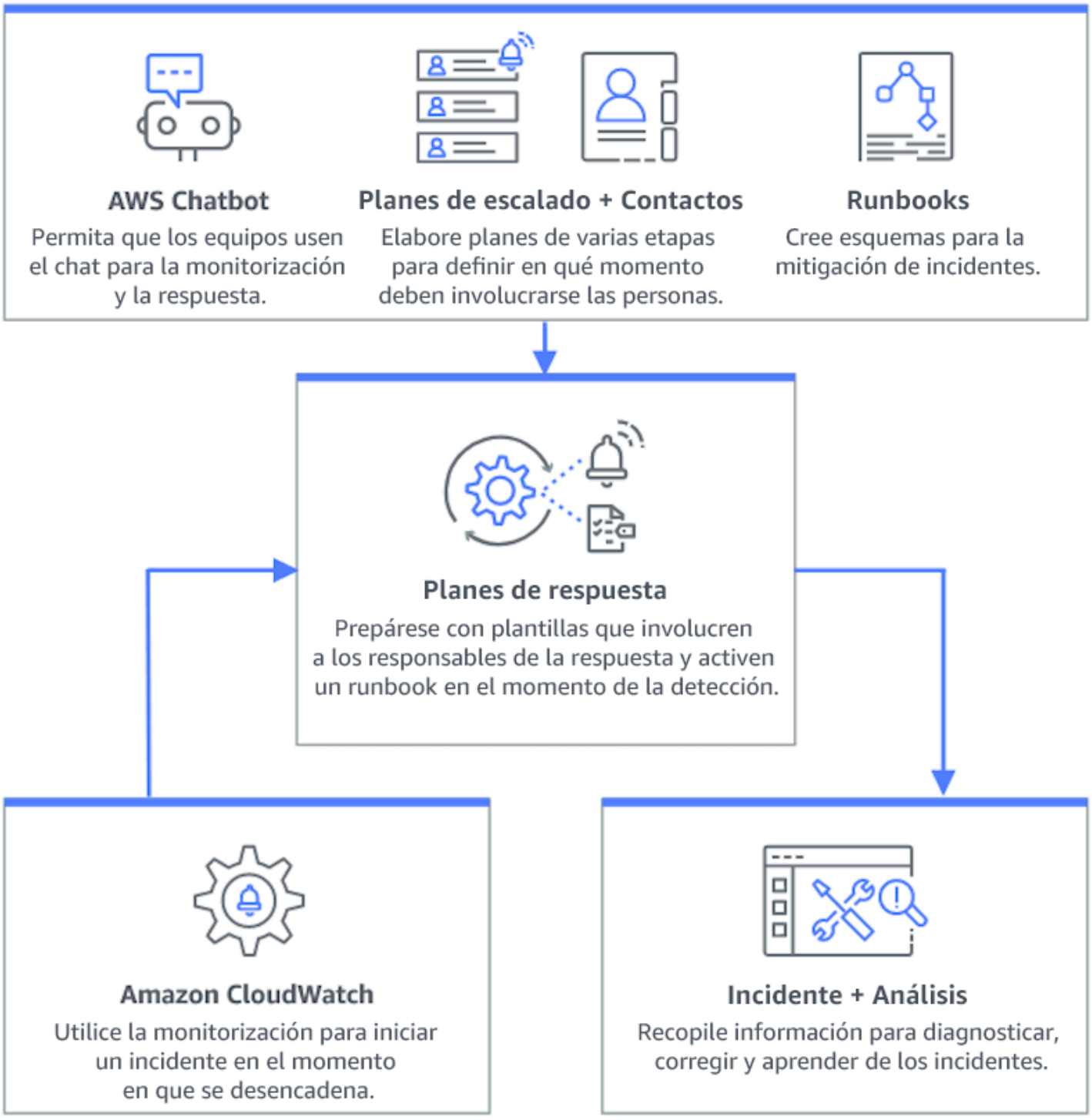
Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Tener un proceso por alerta implica establecer un plan de respuesta claro para cada alerta, automatizar las respuestas siempre que sea posible y perfeccionar continuamente estos procesos en función de los comentarios operativos y los requisitos en evolución.

### Pasos para la implementación

El siguiente diagrama muestra el flujo de trabajo de administración de incidentes en Administrador de incidentes de AWS Systems Manager. Está diseñado para responder rápidamente a los problemas operativos mediante la creación automática de incidentes en respuesta a eventos específicos de Amazon CloudWatch o Amazon EventBridge. Cuando se crea un incidente, ya sea de forma automática o manual, el Administrador de incidentes centraliza la administración del incidente, organiza la información relevante de los recursos de AWS e inicia planes de respuesta predefinidos. Esto incluye ejecutar manuales de procedimientos de Automatización de Systems Manager para tomar medidas inmediatas, así como crear un elemento de trabajo operativo principal en el Centro de operaciones para hacer un seguimiento de las tareas y los análisis relacionados. Este proceso simplificado acelera y coordina la respuesta a los incidentes en todo su entorno de AWS.




1. Uso de alarmas compuestas: cree alarmas compuestas en CloudWatch para agrupar las alarmas relacionadas, reducir el ruido y permitir respuestas más significativas.
2. Manténgase informado con AWS Health: AWS Health es la fuente autorizada de información sobre el estado de los recursos de Nube de AWS. Utilice AWS Health para visualizar y recibir

notificaciones de cualquier evento de servicio actual y de los próximos cambios, como los eventos de ciclo de vida planificados, de forma que pueda tomar medidas para mitigar los impactos.

a. Cree notificaciones de eventos de AWS Health adaptadas al propósito para el correo electrónico y los canales de chat a través de AWS User Notifications e intégrelas mediante programación con las herramientas de supervisión y alerta a través de Amazon EventBridge o la API de AWS Health.

b. Planifique y realice un seguimiento del progreso de los eventos de estado que requieran una acción mediante la integración con herramientas de administración de cambios o ITSM (como Jira o ServiceNow) que ya pueda utilizar a través de Amazon EventBridge o la API de AWS Health.

c. Si usa AWS Organizations, habilite la vista de organización para AWS Health a fin de agregar eventos de AWS Health en todas las cuentas.

3. Integración de las alarmas de Amazon CloudWatch con el Administrador de incidentes: configure las alarmas de CloudWatch para crear incidentes automáticamente en Administrador de incidentes de AWS Systems Manager.
4. Integración de Amazon EventBridge con el Administrador de incidentes: cree reglas de EventBridge para reaccionar ante los eventos y crear incidentes mediante planes de respuesta definidos.
5. Preparación para incidentes en el Administrador de incidentes:
   - Establezca planes de respuesta detallados en el Administrador de incidentes para cada tipo de alerta.
   - Establezca canales de chat mediante Amazon Q Developer en aplicaciones de chat conectadas a los planes de respuesta del Administrador de incidentes, lo que facilita la comunicación en tiempo real durante los incidentes en plataformas como Slack, Microsoft Teams y Amazon Chime.
   - Incorpore manuales de procedimientos de Automatización de Systems Manager en el Administrador de incidentes para impulsar respuestas automatizadas a los incidentes.

Recursos

Prácticas recomendadas relacionadas:

- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS08-BP04 Creación de alertas procesables

Documentos relacionados:

- AWS Cloud Adoption Framework: Operations Perspective - Incident and problem management
- Uso de las alarmas de Amazon CloudWatch
- Setting up Administrador de incidentes de AWS Systems Manager
- Preparing for incidents in Incident Manager

Videos relacionados:

- Top incident response tips from AWS
- re:Invent 2023 | Manage resource lifecycle events at scale with AWS Health

Ejemplos relacionados:

- AWS Workshops - Administrador de incidentes de AWS Systems Manager - Automate incident response to security events

OPS10-BP03 Priorización de los eventos operativos según el impacto empresarial

Responder con prontitud a los eventos operativos es fundamental, pero no todos los eventos son iguales. Cuando se establecen prioridades en función del impacto en la empresa, también se da prioridad a los eventos que pueden tener consecuencias importantes, como la seguridad, las pérdidas financieras, las infracciones de la normativa o los daños a la reputación.

Resultado deseado: las respuestas a los eventos operativos se priorizan en función del posible impacto en las operaciones y los objetivos comerciales. Esto hace que las respuestas sean eficientes y efectivas.

Patrones comunes de uso no recomendados:

- Todos los eventos se tratan con el mismo nivel de urgencia, lo que genera confusión y retrasos a la hora de abordar los problemas críticos.
- No puede distinguir entre eventos de alto y bajo impacto, lo que lleva a una mala asignación de recursos.
- Su organización carece de un marco de priorización claro, lo que deriva en respuestas incongruentes a los eventos operativos.

- Los eventos se priorizan en función del orden en el que se informan, en lugar de su impacto en los resultados empresariales.

Beneficios de establecer esta práctica recomendada:

- Garantiza que las funciones empresariales críticas reciban la atención en primer lugar, lo que minimiza los posibles daños.
- Mejora la asignación de recursos durante varios eventos simultáneos.
- Mejora la capacidad de la organización para mantener la confianza y cumplir con los requisitos reglamentarios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Cuando nos enfrentamos a múltiples eventos operativos, es esencial adoptar un enfoque estructurado para la priorización en función del impacto y la urgencia. Este enfoque le ayuda a tomar decisiones informadas, dirigir los esfuerzos hacia donde más se necesitan y mitigar el riesgo para la continuidad del negocio.

Pasos para la implementación

1. Evaluación del impacto: desarrolle un sistema de clasificación para evaluar la gravedad de los eventos en términos de su posible impacto en las operaciones y los objetivos comerciales. En el siguiente ejemplo se muestran las categorías de impacto:

| Nivel de impacto | Descripción |
| --- | --- |
| Alto | Afecta a muchos empleados o clientes, tiene un alto impacto financiero, o causa graves daños a la reputación o perjuicios. |
| Medio | Afecta a grupos de empleados o clientes, tiene un impacto financiero moderado o un daño a la reputación moderado. |

| Nivel de impacto | Descripción |
| --- | --- |
| Bajo | Afecta al personal o a los clientes individua les, tiene un bajo impacto financiero o un daño reducido a la reputación. |

2. Evaluación de la urgencia: defina los niveles de urgencia para determinar la rapidez con la que un evento necesita una respuesta, teniendo en cuenta factores como la seguridad, las implicaciones financieras y los acuerdos de nivel de servicio (SLA). En el siguiente ejemplo se muestran las categorías de urgencia:

| Nivel de urgencia | Descripción |
| --- | --- |
| Alto | Daños que aumentan exponencialmente, trabajo sensible al tiempo afectado, escalado inminente, o usuarios o grupos VIP afectados. |
| Medio | Los daños aumentan con el tiempo, o afecta a un único usuario VIP o grupo. |
| Bajo | Los daños marginales aumentan con el tiempo, o el trabajo no sensible al tiempo se ve afectado. |

3. Creación de una matriz de priorización:
    - Utilice una matriz para hacer referencias cruzadas del impacto y la urgencia mediante la asignación de niveles de prioridad a diferentes combinaciones.
    - Haga que todos los miembros del equipo responsables de las respuestas a los eventos operativos puedan acceder a la matriz y comprenderla.
    - La siguiente matriz de ejemplo muestra la gravedad del incidente según la urgencia y el impacto:

| Urgencia e impacto | Alto | Medio | Bajo |
| --- | --- | --- | --- |
| Alto | Crítica | Urgente | Alto |
| Medio | Urgente | Alto | Normal |

| Urgencia e impacto | Alto | Medio | Bajo |
| --- | --- | --- | --- |
| Bajo | Alto | Normal | Bajo |

4. Formación y comunicación: forme a los equipos de respuesta sobre la matriz de priorización y la importancia de seguirla durante un evento. Comunique el proceso de priorización a todas las partes interesadas para establecer expectativas claras.
5. Integración con la respuesta a incidentes:
   - Incorpore la matriz de priorización en sus planes y herramientas de respuesta a incidentes.
   - Automatice la clasificación y la priorización de los eventos siempre que sea posible para acelerar los tiempos de respuesta.
   - Los clientes de Enterprise Support pueden aprovechar la Detección y respuesta a incidentes de AWS, que proporciona supervisión proactiva y administración de incidentes ininterrumpidas para las cargas de trabajo de producción.
6. Revisión y adaptación: revise de forma periódica la eficacia del proceso de priorización y haga ajustes en función de las opiniones y los cambios en el entorno empresarial.

Recursos

Prácticas recomendadas relacionadas:

- OPS03-BP03 Fomento de la derivación
- OPS08-BP04 Creación de alertas procesables
- OPS09-BP01 Medición de los objetivos operativos y los KPI con métricas

Documentos relacionados:

- Atlassian - Understanding incident severity levels
- IT Process Map - Checklist Incident Priority

OPS10-BP04 Definición de rutas de escalado

Establezca rutas de escalado claras dentro de sus protocolos de respuesta a incidentes para facilitar una acción oportuna y eficaz. Esto incluye especificar las indicaciones para el escalado, detallar el proceso de escalado y aprobar previamente las acciones para acelerar la toma de decisiones y reducir el tiempo medio de resolución (MTTR).

Resultado deseado: un proceso estructurado y eficiente que eleve los incidentes al personal apropiado, lo que reduce los tiempos de respuesta y el impacto.

Patrones comunes de uso no recomendados:

- La falta de claridad en los procedimientos de recuperación conduce a respuestas improvisadas durante los incidentes críticos.
- La ausencia de permisos y propiedad definidos provoca retrasos cuando se necesita una acción urgente.
- Las partes interesadas y los clientes no reciben información de acuerdo con las expectativas.
- Las decisiones importantes se retrasan.

Beneficios de establecer esta práctica recomendada:

- Respuesta simplificada a los incidentes mediante procedimientos de escalado predefinidos.
- Se ha reducido el tiempo de inactividad con acciones preaprobadas y una propiedad clara.
- Mejora de la asignación de recursos y los ajustes del nivel de soporte según la gravedad del incidente.
- Mejora de la comunicación con las partes interesadas y los clientes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Las rutas de escalado bien definidas son cruciales para una respuesta rápida a los incidentes. Administrador de incidentes de AWS Systems Manager permite establecer planes de escalado estructurados y programas de guardia, que alertan al personal adecuado para que esté preparado para actuar cuando se produzcan incidentes.

Pasos para la implementación

1. Configuración de las indicaciones de escalado: configure alarmas de CloudWatch para crear un incidente en Administrador de incidentes de AWS Systems Manager.
2. Configuración de programas de guardia: cree programas de guardia en el Administrador de incidentes que se ajusten a sus rutas de escalado. Proporcione al personal de guardia los permisos y las herramientas necesarios para actuar con rapidez.
3. Detalle los procedimientos de escalado:

- Determine las condiciones específicas en las que se debe escalar un incidente.
- Cree planes de escalado en el Administrador de incidentes.
- Los canales de escalado deben consistir en un contacto o un programa de guardia.
- Defina las funciones y responsabilidades del equipo en cada nivel de escalado.

4. Aprobación previa de las acciones de mitigación: colabore con los responsables de la toma de decisiones para aprobar previamente las acciones para los escenarios previstos. Utilice los manuales de procedimientos de Automatización de Systems Manager integrados con el Administrador de incidentes para acelerar la resolución de incidentes.
5. Especificación de la propiedad: identifique claramente a los propietarios internos de cada paso de la ruta de escalado.
6. Detalle los escalados de terceros:
   - Documente los acuerdos de nivel de servicio (SLA) de terceros y ajústelos a los objetivos internos.
   - Establezca protocolos claros para la comunicación con los proveedores durante los incidentes.
   - Integre los contactos de los proveedores en las herramientas de administración de incidentes para que se pueda acceder directamente a ellos.
   - Lleve a cabo simulacros periódicos que incluyan situaciones de respuesta de terceros.
   - Mantenga la información de escalado de proveedores bien documentada y accesible.
7. Formación y práctica de los planes de escalado: forme a su equipo en el proceso de escalado y lleve a cabo simulacros o días de juego de respuesta a incidentes con regularidad. Los clientes de Enterprise Support pueden solicitar un taller de administración de incidentes.
8. Continuación de la mejora: revise la eficacia de sus rutas de escalado con regularidad. Actualice sus procesos en función de las lecciones aprendidas a partir de los análisis posteriores a los incidentes y los comentarios continuos.

Nivel de esfuerzo para el plan de implementación: moderado

Recursos

Prácticas recomendadas relacionadas:

- OPS08-BP04 Creación de alertas procesables
- OPS10-BP02 Implementación de un proceso por alerta
- OPS11-BP02 Análisis después del incidente

Documentos relacionados:

- Administrador de incidentes de AWS Systems Manager Escalation Plans
- Working with on-call schedules in Incident Manager
- Creación y administración de manuales de procedimientos
- Temporary elevated access management with AWS IAM Identity Center
- Atlassian - Escalation policies for effective incident management

OPS10-BP05 Definición de un plan de comunicación con los clientes en caso de eventos que afecten al servicio

Es fundamental comunicarse eficazmente durante los eventos que afectan al servicio para mantener la confianza y la transparencia con los clientes. Un plan de comunicación bien definido ayuda a su organización a compartir información de forma rápida y clara, tanto interna como externamente, durante los incidentes.

Resultado deseado:

- Un plan de comunicación sólido que informe eficazmente a los clientes y partes interesadas durante los eventos que afectan al servicio.
- Transparencia en la comunicación para generar confianza y reducir la ansiedad de los clientes.
- Minimizar el impacto de los eventos que afectan el servicio en la experiencia del cliente y las operaciones comerciales.

Patrones comunes de uso no recomendados:

- Una comunicación inadecuada o tardía genera confusión e insatisfacción en los clientes.
- Los mensajes demasiado técnicos o vagos no transmiten el impacto real a los usuarios.
- No existe una estrategia de comunicación predefinida, lo que da como resultado mensajes incoherentes y reactivos.

Beneficios de establecer esta práctica recomendada:

- Mejora de la confianza y la satisfacción de los clientes mediante una comunicación proactiva y clara.

- Se ha reducido la carga de los equipos de asistencia al abordar de forma preventiva las inquietudes de los clientes.
- Capacidad mejorada para administrar los incidentes y recuperarse de ellos de forma eficaz.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La creación de un plan de comunicación integral para los eventos que afectan al servicio implica múltiples facetas, desde la elección de los canales correctos hasta la elaboración del mensaje y el tono. El plan debe ser adaptable, escalable y contemplar diferentes escenarios de interrupción del servicio.

Pasos para la implementación

1. Definición de roles y responsabilidades:
    - Asigne un major incident manager para supervisar las actividades de respuesta a los incidentes.
    - Designe a un communications manager que se encargue de coordinar todas las comunicaciones externas e internas.
    - Incluya al support manager para proporcionar una comunicación congruente a través de los tiques de soporte.
2. Identificación de los canales de comunicación: seleccione canales como el chat del lugar de trabajo, el correo electrónico, los SMS, las redes sociales, las notificaciones dentro de las aplicaciones y las páginas de estado. Estos canales deben ser resilientes y capaces de funcionar de forma independiente durante los eventos que afecten al servicio.
3. Comunicación con los clientes rápida, clara y regular:
    - Elabore plantillas para varios escenarios de deterioro del servicio, haciendo énfasis en la simplicidad y los detalles esenciales. Incluya información sobre el deterioro del servicio, el tiempo de resolución esperado y el impacto.
    - Utilice Amazon Pinpoint para alertar a los clientes mediante notificaciones push, notificaciones dentro de las aplicaciones, correos electrónicos, mensajes de texto, mensajes de voz y mensajes a través de canales personalizados.
    - Utilice Amazon Simple Notification Service (Amazon SNS) para alertar a los suscriptores mediante programación o por correo electrónico, notificaciones push móviles y mensajes de texto.

- Comparta de forma pública el panel de Amazon CloudWatch para comunicar el estado del incidente.
- Fomente la participación en las redes sociales:
  - Supervise activamente las redes sociales para entender la opinión de los clientes.
  - Publique en las plataformas de redes sociales para proporcionar información pública actualizada e implicar a la comunidad.
  - Prepare plantillas para una comunicación clara y coherente en las redes sociales.

4. Coordinación de la comunicación interna: implemente protocolos internos mediante herramientas como Amazon Q Developer en aplicaciones de chat para coordinar a los equipos y facilitar la comunicación. Utilice los paneles de CloudWatch para comunicar el estado.
5. Orquestación de la comunicación con herramientas y servicios dedicados:
   - Utilice Administrador de incidentes de AWS Systems Manager con Amazon Q Developer en aplicaciones de chat para configurar canales de chat dedicados para la comunicación interna y la coordinación en tiempo real durante los incidentes.
   - Utilice manuales de procedimientos de Administrador de incidentes de AWS Systems Manager para automatizar las notificaciones a los clientes a través de Amazon Pinpoint, Amazon SNS o herramientas de terceros, como las plataformas de redes sociales, durante los incidentes.
   - Incorpore flujos de trabajo de aprobación en los manuales de procedimientos para revisar y autorizar, de forma opcional, todas las comunicaciones externas antes de enviarlas.
6. Práctica y mejora:
   - Lleve a cabo formaciones sobre el uso de herramientas y estrategias de comunicación. Permita a los equipos tomar decisiones oportunas durante los incidentes.
   - Ponga a prueba el plan de comunicación mediante simulacros o días de juego. Use estas pruebas para ajustar los mensajes y evaluar la eficacia de los canales.
   - Implemente mecanismos para conocer la opinión de los clientes y evaluar así la eficacia de la comunicación durante los incidentes. Desarrolle continuamente el plan de comunicación en función de los comentarios y las necesidades cambiantes.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos

- OPS10-BP06 Comunicación del estado a través de paneles
- OPS11-BP02 Análisis después del incidente

Documentos relacionados:

- Atlassian - Incident communication best practices
- Atlassian - How to write a good status update
- PagerDuty - A Guide to Incident Communications

Videos relacionados:

- Atlassian - Create your own incident communication plan: Incident templates

Ejemplos relacionados:

- Panel de AWS Health

OPS10-BP06 Comunicación del estado a través de paneles

Utilice los paneles como una herramienta estratégica para transmitir el estado operativo y las métricas clave en tiempo real a diferentes públicos, incluidos los equipos técnicos internos, los líderes y los clientes. Estos paneles ofrecen una representación visual centralizada del estado del sistema y el rendimiento empresarial, lo que mejora la transparencia y la eficiencia de la toma de decisiones.

Resultado deseado:

- Sus paneles proporcionan una visión completa del sistema y de las métricas empresariales relevantes para las diferentes partes interesadas.
- Las partes interesadas pueden acceder de forma proactiva a la información operativa, lo que reduce la necesidad de solicitudes de estado frecuentes.
- La toma de decisiones en tiempo real mejora durante las operaciones normales y los incidentes.

Patrones comunes de uso no recomendados:

- Los ingenieros que se unen a una llamada de administración de incidentes necesitan actualizaciones de estado para ponerse al día.
- Confiar en los informes manuales para la administración, lo que provoca retrasos y posibles imprecisiones.
- Los equipos de operaciones se interrumpen con frecuencia para actualizar el estado durante los incidentes.

Beneficios de establecer esta práctica recomendada:

- Ofrece a las partes interesadas acceso inmediato a información crítica, promoviendo la toma de decisiones informadas.
- Reduce las ineficiencias operativas al minimizar los informes manuales y las consultas frecuentes sobre el estado.
- Aumenta la transparencia y la confianza a través de la visibilidad en tiempo real del rendimiento del sistema y las métricas empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los paneles comunican eficazmente el estado del sistema y las métricas empresariales y se pueden adaptar a las necesidades de los diferentes grupos de audiencia. Las herramientas como los paneles de Amazon CloudWatch y Amazon Quick ayudan a crear paneles interactivos y en tiempo real para la monitorización del sistema y la inteligencia empresarial.

Pasos para la implementación

1. Identificación de las necesidades de las partes interesadas: determine las necesidades de información específicas de los diferentes grupos de audiencia, como los equipos técnicos, los líderes y los clientes.
2. Elija las herramientas adecuadas: seleccione las herramientas adecuadas, como los paneles de Amazon CloudWatch para supervisar el sistema y Amazon Quick para obtener inteligencia empresarial interactiva. AWS Health proporciona una experiencia lista para usar en Panel de AWS Health o puede utilizar eventos de estado en Amazon EventBridge o mediante la API de AWS Health para aumentar los paneles propios.
3. Diseñe paneles eficaces:

- Diseñe paneles para presentar con claridad las métricas y los KPI relevantes, asegurándose de que sean comprensibles y procesables.
- Incorpore vistas a nivel de sistema y empresarial según sea necesario.
- Incluya paneles de alto nivel (para obtener una visión general) y de bajo nivel (para un análisis detallado).
- Integre alarmas automatizadas en los paneles para resaltar los problemas críticos.
- Incluya umbrales de métricas y objetivos importantes en los paneles para poder acceder a esos datos de forma inmediata.

4. Integración de los orígenes de datos:
   - Utilice Amazon CloudWatch para agregar y mostrar métricas de varios servicios de AWS y consultar métricas de otros orígenes de datos, creando una vista unificada de las métricas empresariales y de estado de su sistema.
   - Utilice características como Información de registros de CloudWatch para consultar y visualizar los datos de registro de diferentes aplicaciones y servicios.
   - Utilice los eventos de AWS Health para mantenerse informado sobre el estado operativo y los problemas operativos confirmados de los servicios de AWS mediante la API de AWS Health o los eventos de AWS Health en Amazon EventBridge.
5. Acceso de autoservicio:
   - Comparta paneles de CloudWatch con las partes interesadas pertinentes para acceder a la información de autoservicio mediante características para compartir paneles.
   - Asegúrese de que se pueda acceder fácilmente a los paneles y que incluyan información actualizada en tiempo real.
6. Actualice y ajuste los paneles cada cierto tiempo:
   - Modifique los paneles de forma periódica para alinearlos con las cambiantes necesidades empresariales y las opiniones de las partes interesadas.
   - Revise los paneles cada cierto tiempo para que sigan siendo pertinentes a la hora de transmitir la información necesaria.

Recursos

Prácticas recomendadas relacionadas:

- OPS08-BP05 Creación de paneles

Documentos relacionados:

- La creación de paneles para la visibilidad operativa
- Uso de paneles de Amazon CloudWatch
- Cree paneles flexibles con variables de panel
- Compartir paneles de CloudWatch
- Consulta de métricas de otros orígenes de datos
- Agregue un widget personalizado a un panel de CloudWatch

Ejemplos relacionados:

- One Observability Workshop - Dashboards

OPS10-BP07 Automatización de las respuestas a eventos

La automatización de las respuestas a eventos es clave para una gestión operativa rápida, coherente y sin errores. Cree procesos simplificados y utilice herramientas para administrar y responder automáticamente a los eventos, lo que minimiza las intervenciones manuales y mejora la eficacia operativa.

Resultado deseado:

- Se han reducido los errores humanos y tiempos de resolución más rápidos mediante la automatización.
- Gestión de eventos operativos coherente y fiable.
- Se ha mejorado la eficiencia operativa y la fiabilidad del sistema.

Patrones comunes de uso no recomendados:

- La gestión manual de eventos provoca retrasos y errores.
- La automatización se pasa por alto en las tareas críticas y repetitivas.
- Las tareas manuales y repetitivas provocan saturación de alertas y la omisión de problemas críticos.

Beneficios de establecer esta práctica recomendada:

- Respuestas rápidas a los eventos, lo que reduce el tiempo de inactividad del sistema.
- Operaciones fiables con una gestión de eventos automatizada y coherente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Incorpore la automatización para crear flujos de trabajo operativos eficientes y minimizar las intervenciones manuales.

Pasos para la implementación

1. Identificación de las oportunidades de automatización: determine qué tareas repetitivas pueden automatizarse, como la resolución de problemas, el enriquecimiento de tiques, la administración de la capacidad, el escalado, las implementaciones y las pruebas.
2. Identificación de los avisos de automatización:
   - evalúe y defina las condiciones o métricas específicas que inician las respuestas automatizadas mediante acciones de alarmas de Amazon CloudWatch.
   - Use Amazon EventBridge para responder a eventos en servicios de AWS, cargas de trabajo personalizadas y aplicaciones SaaS.
   - Tenga en cuenta los eventos de inicio, como entradas de registro específicas, umbrales de métricas de rendimiento o cambios de estado en los recursos de AWS.
3. Implementación de la automatización basada en eventos:
   - Utilice los manuales de procedimientos de Automatización de AWS Systems Manager para simplificar las tareas de mantenimiento, implementación y corrección.
   - La creación de incidentes en el Administrador de incidentes recopila y agrega automáticamente detalles sobre los recursos de AWS involucrados en el incidente.
   - Supervise las cuotas de forma proactiva mediante el Monitor de cuotas para AWS.
   - Ajuste automáticamente la capacidad con AWS Auto Scaling para mantener la disponibilidad y el rendimiento.
   - Automatice las canalizaciones de desarrollo con Amazon CodeCatalyst.
   - Haga pruebas de humo o supervise continuamente los puntos de conexión y las API mediante la supervisión sintética.
4. Mitigación de los riesgos mediante la automatización:
   - Implemente respuestas de seguridad automatizadas para abordar los riesgos con rapidez.

- Use AWS Systems Manager State Manager para reducir los cambios en la configuración.
- Corrija los recursos no conformes con Reglas de AWS Config.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- OPS08-BP04 Creación de alertas procesables
- OPS10-BP02 Implementación de un proceso por alerta

Documentos relacionados:

- Using Systems Manager Automation runbooks with Incident Manager
- Creating incidents in Incident Manager
- Service Quotas de AWS
- Monitor resource usage and send notifications when approaching quotas
- AWS Auto Scaling
- What is Amazon CodeCatalyst?
- Uso de las alarmas de Amazon CloudWatch
- Uso de las acciones de alarma de Amazon CloudWatch
- Remediating Noncompliant Resources with Reglas de AWS Config
- Creating metrics from log events using filters
- AWS Systems ManagerState Manager

Videos relacionados:

- Create Automation Runbooks with AWS Systems Manager
- How to automate IT Operations on AWS
- AWS Security Hub CSPM automation rules
- Start your software project fast with Amazon CodeCatalyst blueprints

Ejemplos relacionados:

- Amazon CodeCatalyst Tutorial: Creating a project with the Modern three-tier web application blueprint
- One Observability Workshop
- Respond to incidents using Incident Manager

## Evolución

Pregunta

- OPS 11. ¿Cómo desarrolla las operaciones?

### OPS 11. ¿Cómo desarrolla las operaciones?

Dedique tiempo y recursos a la mejora gradual casi continua, para incrementar la eficacia y la eficiencia de sus operaciones.

Prácticas recomendadas

- OPS11-BP01 Implementación de un proceso de mejora continua
- OPS11-BP02 Análisis después del incidente
- OPS11-BP03 Implementación de bucles de retroalimentación
- OPS11-BP04 Administración de conocimientos
- OPS11-BP05 Definición de los elementos que impulsan la mejora
- OPS11-BP06 Validación de la información
- OPS11-BP07 Revisiones de métricas de operaciones
- OPS11-BP08 Documentación y comunicación de las lecciones aprendidas
- OPS11-BP09 Asignación de tiempo para implementar mejoras

#### OPS11-BP01 Implementación de un proceso de mejora continua

Evalúe su carga de trabajo con respecto a las prácticas recomendadas de arquitectura interna y externa. Lleve a cabo revisiones frecuentes e intencionadas de la carga de trabajo. Priorice las oportunidades de mejora en su cadencia de desarrollo de software.

Resultado deseado:

- Analiza con frecuencia su carga de trabajo con respecto a las prácticas recomendadas de arquitectura.
- Da a las oportunidades de mejora la misma prioridad que a las características de su proceso de desarrollo de software.

Patrones comunes de uso no recomendados:

- No ha revisado la arquitectura de su carga de trabajo desde que se implementó hace varios años.
- Le da menos prioridad a las oportunidades de mejora. En comparación con las nuevas características, estas oportunidades se quedan pendientes.
- No existe un estándar para implementar las modificaciones de las prácticas recomendadas para la organización.

Beneficios de establecer esta práctica recomendada:

- Su carga de trabajo se mantiene actualizada en cuanto a las prácticas recomendadas de arquitectura.
- Desarrolla su carga de trabajo de manera intencionada.
- Puede aprovechar las prácticas recomendadas de la organización para mejorar todas las cargas de trabajo.
- Obtiene ganancias marginales que tienen un efecto acumulativo, lo que impulsa una mayor eficiencia.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Revise con frecuencia la arquitectura de su carga de trabajo. Use prácticas recomendadas internas y externas, evalúe su carga de trabajo e identifique las oportunidades de mejora. Priorice las oportunidades de mejora en su cadencia de desarrollo de software.

Pasos para la implementación

1. Revise periódicamente la arquitectura de su carga de trabajo de producción con una frecuencia acordada. Utilice un estándar de arquitectura documentado que incluya prácticas recomendadas específicas de AWS.

a. Use sus estándares definidos internamente para estas revisiones. Si no dispone de un estándar interno, utilice el Marco de AWS Well-Architected.

b. Utilice AWS Well-Architected Tool para crear un enfoque personalizado de sus prácticas recomendadas internas y llevar a cabo la revisión de la arquitectura.

c. Contacte con su AWS Solution Architect o Technical Account Manager para llevar a cabo una revisión guiada del Marco de Well-Architected de su carga de trabajo.

2. Priorice las oportunidades de mejora identificadas durante la revisión en su proceso de desarrollo de software.

Nivel de esfuerzo para el plan de implementación: bajo. Puede utilizar el Marco de AWS Well-Architected para llevar a cabo la revisión anual de la arquitectura.

Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP02 Análisis después del incidente
- OPS11-BP08 Documentación y comunicación de las lecciones aprendidas
- OPS04 Implementación de la observabilidad

Documentos relacionados:

- AWS Well-Architected Tool - Custom Lenses
- Documento técnico sobre AWS Well-Architected: el proceso de revisión
- Customize Well-Architected Reviews using Custom Lenses and the AWS Well-Architected Tool
- Implementing the AWS Well-Architected Custom Lens lifecycle in your organization

Videos relacionados:

- AWS re:Invent 2023 - Scaling AWS Well-Architected best practices across your organization

Ejemplos relacionados:

- AWS Well-Architected Tool

### OPS11-BP02 Análisis después del incidente

Revise los eventos que afectan a los clientes e identifique los factores que contribuyen a ellos y las medidas preventivas. Use esta información para desarrollar un plan de mitigación que limite o evite la reaparición del problema. Desarrolle procedimientos para proporcionar respuestas rápidas y eficaces. Comunique los factores que han contribuido al problema y las medidas correctivas según corresponda, adaptados al público de destino.

Resultado deseado:

- Ha establecido procesos de administración de incidentes que incluyen análisis después del incidente.
- Tiene planes de observabilidad para recopilar datos sobre los eventos.
- Con estos datos, comprende y recopila las métricas que respaldan su proceso de análisis posterior al incidente.
- Aprende de los incidentes para mejorar los resultados futuros.

Patrones comunes de uso no recomendados:

- Administra un servidor de aplicaciones. Aproximadamente cada 23 horas y 55 minutos finalizan todas las sesiones activas. Ha tratado de identificar lo que no funciona correctamente en el servidor de aplicaciones. Sospecha que podría tratarse de un problema de red, pero no consigue que el equipo de red colabore porque están demasiado ocupados para ayudarle. Carece de un proceso predefinido para obtener asistencia y recopilar la información necesaria para determinar lo que está sucediendo.
- Ha sufrido pérdidas de datos dentro de la carga de trabajo. Es la primera vez que ocurre y la causa no es evidente. Decide que no es importante porque puede volver a crear los datos. La pérdida de datos comienza a producirse con mayor frecuencia, lo que afecta a los clientes. Esto también supone una carga operativa adicional al restaurar los datos perdidos.

Beneficios de establecer esta práctica recomendada:

- Dispone de un proceso predefinido para determinar los componentes, las condiciones, las acciones y los eventos que han contribuido a un incidente le permite identificar oportunidades de mejora.
- Utiliza los datos del análisis posterior al incidente para aplicar mejoras.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Use un proceso para determinar los factores que han contribuido al problema. Revise todos los incidentes que afectan a los clientes. Disponga de un proceso para identificar y documentar los factores que han contribuido al incidente, de manera que se puedan elaborar medidas de mitigación para limitar o prevenir su repetición y se puedan desarrollar procedimientos para dar respuestas rápidas y eficaces. Comunique las causas raíz de los incidentes según corresponda y adapte la comunicación a su público objetivo. Comparta la información obtenida con el resto de la organización.

Pasos para la implementación

1. Recopile métricas como el cambio de implementación o de configuración, la hora de inicio del incidente, la hora de la alarma, la hora de activación, la hora de inicio de la mitigación y la hora de resolución del incidente.
2. Describa los puntos temporales clave en el cronograma para comprender los eventos del incidente.
3. Hágase las siguientes preguntas:
    a. ¿Podría mejorar el tiempo de detección?
    b. ¿Existen actualizaciones de las métricas y alarmas que detectarían el incidente en menos tiempo?
    c. ¿Puede mejorar el tiempo hasta el diagnóstico?
    d. ¿Existen actualizaciones para sus planes de respuesta o planes de escalada que implicarían en menos tiempo a los respondedores correctos?
    e. ¿Puede mejorar el tiempo de mitigación?
    f. ¿Hay pasos del manual de procedimientos o de estrategias que pueda agregar o mejorar?
    g. ¿Puede evitar que ocurran futuros incidentes?
4. Cree listas de verificación y acciones. Haga un seguimiento y cumpla con todas las acciones.

Nivel de esfuerzo para el plan de implementación: medio

Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP01 Implementación de un proceso de mejora continua
- OPS 4: Implementación de la observabilidad

Documentos relacionados:

- Performing a post-incident analysis in Incident Manager
- Operational Readiness Review

OPS11-BP03 Implementación de bucles de retroalimentación

Los bucles de retroalimentación proporcionan información procesable que impulsa la toma de decisiones. Cree bucles de retroalimentación en sus procedimientos y cargas de trabajo. Le servirán para identificar los problemas y las áreas que necesitan mejoras. También validan las inversiones hechas en las mejoras. Estos bucles de retroalimentación son la base para mejorar continuamente la carga de trabajo.

Los bucles de retroalimentación se dividen en dos categorías: retroalimentación inmediata y análisis retrospectivo. La retroalimentación inmediata se obtiene mediante la revisión del rendimiento y los resultados de las actividades operativas. Esta retroalimentación procede de los miembros del equipo, de los clientes o del resultado automático de la actividad. Se recibe retroalimentación inmediata de aspectos como las pruebas A/B y el envío de nuevas características. Es esencial responder rápido a los errores.

El análisis retrospectivo se lleva a cabo periódicamente para obtener retroalimentación de la revisión de resultados operativos y de las métricas a lo largo del tiempo. Estas retrospectivas tienen lugar al final de un sprint, en una cadencia, o después de lanzamientos o eventos importantes. Este tipo de bucle de retroalimentación valida las inversiones en operaciones o la carga de trabajo. Lo ayuda a medir el éxito y valida su estrategia.

Resultado deseado: utiliza la retroalimentación inmediata y el análisis retrospectivo para impulsar las mejoras. Existe un mecanismo para obtener la retroalimentación de los usuarios y de los miembros del equipo. El análisis retrospectivo se utiliza para identificar las tendencias que impulsan las mejoras.

Patrones comunes de uso no recomendados:

- Lanza una nueva característica, pero no tiene forma de recibir la retroalimentación de los clientes sobre ella.

- Después de invertir en mejoras operativas, no lleva a cabo una retrospectiva para validarlas.
- Recopila la retroalimentación de los clientes, pero no la revisa con regularidad.
- Los bucles de retroalimentación dan lugar a propuestas de acción, pero no se incluyen en el proceso de desarrollo del software.
- Los clientes no reciben retroalimentación sobre las mejoras que han propuesto.

Beneficios de establecer esta práctica recomendada:

- Puede hacer un recorrido inverso desde el cliente para impulsar nuevas características.
- La cultura de su organización puede reaccionar más rápidamente a los cambios.
- Las tendencias se utilizan para identificar las oportunidades de mejora.
- Las retrospectivas validan las inversiones hechas en la carga de trabajo y las operaciones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La implementación de esta práctica recomendada implica utilizar tanto la retroalimentación inmediata como el análisis retrospectivo. Estos bucles de retroalimentación impulsan las mejoras. Existen muchos mecanismos para obtener retroalimentación inmediata, como encuestas, sondeos de opinión de los clientes o formularios de retroalimentación. Su organización también utiliza las retrospectivas para identificar las oportunidades de mejora y validar las iniciativas.

Ejemplo de cliente

AnyCompany Retail ha creado un formulario web en el que los clientes pueden dar retroalimentación o informar de sus problemas. Durante el examen semanal, el equipo de desarrollo de software evalúa la retroalimentación de los usuarios. La retroalimentación se utiliza periódicamente para dirigir la evolución de la plataforma. Se lleva a cabo una retrospectiva al final de cada sprint para identificar los elementos que quiere mejorar.

Pasos para la implementación

1. Retroalimentación inmediata
    - Necesita un mecanismo para recibir retroalimentación de los clientes y de los miembros del equipo. Las actividades de sus operaciones también se pueden configurar para ofrecer retroalimentación automática.

- Su organización necesita un proceso para revisar esta retroalimentación, determinar qué hay que mejorar y programar la mejora.
- Los comentarios deben agregarse a su proceso de desarrollo de software.
- A medida que vaya incorporando mejoras, haga un seguimiento del remitente de la retroalimentación.
  - Puede usar el Centro de operaciones de AWS Systems Manager para crear esta mejora como OpsItems y hacer un seguimiento de ellas.

2. Análisis retrospectivo
   - Lleve a cabo retrospectivas al final de un ciclo de desarrollo, con una cadencia determinada o después de un lanzamiento importante.
   - Convoque a las partes implicadas en la carga de trabajo para una reunión retrospectiva.
   - Cree tres columnas en una pizarra u hoja de cálculo: Detener, Iniciar y Mantener.
     - Detener corresponde a lo que quiera que su equipo deje de hacer.
     - Iniciar corresponde a las ideas que quiere empezar a hacer.
     - Mantener corresponde a lo que quiere seguir haciendo.
   - Recorra la sala y recopile la retroalimentación de las partes interesadas.
   - Priorice la retroalimentación. Asigne acciones y partes interesadas a los elementos de los apartados Iniciar o Mantener.
   - Agregue las acciones a su proceso de desarrollo de software y comunique las actualizaciones de estado a las partes interesadas a medida que haga las mejoras.

Nivel de esfuerzo para el plan de implementación: medio. Para implementar esta práctica recomendada, necesita un método para recibir retroalimentación inmediata y analizarla. Además, debe establecer un proceso de análisis retrospectivo.

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP01 Evaluación de las necesidades de los clientes externos: los bucles de retroalimentación son un mecanismo para recopilar las necesidades de los clientes externos.
- OPS01-BP02 Evaluación de las necesidades de los clientes internos: las partes interesadas internas pueden utilizar los bucles de retroalimentación para comunicar las necesidades y los requisitos.

- OPS11-BP02 Análisis después del incidente: los análisis posteriores a los incidentes son una forma importante de análisis retrospectivo que se lleva a cabo después de los incidentes.
- OPS11-BP07 Revisiones de métricas de operaciones: las revisiones de las métricas de las operaciones identifican tendencias y áreas de mejora.

Documentos relacionados:

- 7 Pitfalls to Avoid When Building a CCOE
- Atlassian Team Playbook - Retrospectives
- Email Definitions: Feedback Loops
- Establishing Feedback Loops Based on the AWS Well-Architected Framework Review
- IBM Garage Methodology - Hold a retrospective
- Investopedia – The PDCS Cycle
- Maximizing Developer Effectiveness, por Tim Cochran
- Operations Readiness Reviews (ORR) Whitepaper - Iteration
- ITIL CSI - Continual Service Improvement
- When Toyota met e-commerce: Lean at Amazon

Videos relacionados:

- Building Effective Customer Feedback Loops

Ejemplos relacionados:

- Astuto: herramienta de código abierto para la retroalimentación de los clientes
- Soluciones de AWS: QnABot en AWS
- Fider: una plataforma para organizar la retroalimentación de los clientes

Servicios relacionados:

- AWS Systems Manager OpsCenter

### OPS11-BP04 Administración de conocimientos

La gestión del conocimiento ayuda a los miembros del equipo a encontrar la información necesaria para cumplir con su cometido. En las organizaciones basadas en el aprendizaje, la información se comparte libremente, lo que capacita a los individuos. La información puede detectarse o buscarse. La información es precisa y está actualizada. Existen mecanismos para crear nueva información, actualizar la existente y archivar la obsoleta. Los ejemplos más frecuentes de plataforma de administración del conocimiento es un sistema de administración de contenido, como una wiki.

Resultado deseado:

- Los miembros del equipo tienen acceso a información oportuna y precisa.
- Se puede buscar información.
- Existen mecanismos para agregar, actualizar y archivar la información.

Patrones comunes de uso no recomendados:

- No existe un almacenamiento centralizado de conocimientos. Los miembros del equipo administran sus propias notas en sus máquinas locales.
- Dispone de una wiki autoalojada, pero no de mecanismos para administrar la información, lo que provoca que esta quede obsoleta.
- Alguien identifica información que falta, pero no existe un proceso para solicitar que se agrega a la wiki del equipo. La agregan ellos mismos, pero se saltan un paso clave, lo que provoca una interrupción del servicio.

Beneficios de establecer esta práctica recomendada:

- Los miembros del equipo tienen más poder porque la información se comparte libremente.
- Los nuevos miembros del equipo se incorporan más rápidamente porque la documentación está actualizada y es posible hacer búsquedas en ella.
- La información es oportuna, precisa y procesable.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

### Guía para la implementación

La administración del conocimiento es una faceta importante de las organizaciones de aprendizaje. Para empezar, necesita un repositorio central para almacenar los conocimientos (un ejemplo habitual es una wiki autoalojada). Debe desarrollar procesos para agregar, actualizar y archivar conocimientos. Desarrolle estándares sobre lo que debe documentarse y permita que todos contribuyan.

### Ejemplo de cliente

AnyCompany Retail alberga una wiki interna donde se almacenan todos los conocimientos. Se anima a los miembros del equipo a agregar información a la base de conocimientos mientras hacen sus tareas cotidianas. Cada trimestre, un equipo interfuncional evalúa las páginas menos actualizadas y determina si deben archivarse o actualizarse.

### Pasos para la implementación

1. Empiece por identificar el sistema de administración de contenido en el que se almacenarán los conocimientos. Consiga el acuerdo de las partes interesadas de toda la organización.
   a. Si no dispone de un sistema de administración de contenido, considere la posibilidad de crear una wiki autoalojada o utilizar un repositorio de control de versiones como punto de partida.
2. Elabore manuales de procedimientos para agregar, actualizar y archivar información. Forme a su equipo en estos procesos.
3. Identifique qué conocimientos deben almacenarse en el sistema de administración de contenido. Empiece por las actividades diarias (manuales de procedimientos y manuales de estrategias) que llevan a cabo los miembros del equipo. Colabore con las partes interesadas para priorizar qué conocimientos se agregan.
4. Trabaje periódicamente con las partes interesadas para identificar la información obsoleta y archivarla o actualizarla.

Nivel de esfuerzo para el plan de implementación: medio. Si no dispone de un sistema de administración de contenido, puede crear una wiki autoalojada o un repositorio de documentos controlado por versiones.

### Recursos

Prácticas recomendadas relacionadas:

- OPS11-BP08 Documentación y comunicación de las lecciones aprendidas: la administración del conocimiento facilita el intercambio de información sobre las lecciones aprendidas.

Documentos relacionados:

- Atlassian - Knowledge Management

Ejemplos relacionados:

- DokuWiki
- Gollum
- MediaWiki
- Wiki.js

OPS11-BP05 Definición de los elementos que impulsan la mejora

Identifique los factores que impulsan la mejora para ayudarlo a evaluar y priorizar las oportunidades en función de los datos y los bucles de comentarios. Explore las oportunidades de mejora en sus sistemas y procesos, y automatice cuando corresponda.

Resultado deseado:

- Se hace un seguimiento de los datos en todo el entorno.
- Se correlacionan los eventos y las actividades con los resultados empresariales.
- Puede comparar y contrastar entornos y sistemas.
- Mantiene un historial de actividad detallado de sus implementaciones y resultados.
- Recopila datos para respaldar su postura de seguridad.

Patrones comunes de uso no recomendados:

- Recopila datos de todo su entorno, pero no correlaciona eventos ni actividades.
- Recopila datos detallados de todo su patrimonio y esto aumenta la actividad y los costos de Amazon CloudWatch y AWS CloudTrail. Sin embargo, no utiliza estos datos de manera significativa.
- No tiene en cuenta los resultados empresariales al definir los factores que impulsan la mejora.

- No se miden los efectos de las nuevas características.

Beneficios de establecer esta práctica recomendada:

- Minimiza la repercusión de las motivaciones basadas en eventos o la inversión emocional al determinar criterios de mejora.
- Responde a los eventos empresariales, no solo a los técnicos.
- Mide su entorno para identificar las áreas de mejora.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Conozca los factores que impulsan la mejora: solo debe hacer cambios en un sistema cuando estos producen un resultado deseado.
  - Capacidades deseadas: evalúe las características y capacidades deseadas al evaluar las oportunidades de mejora.
    - Novedades de AWS
  - Problemas inaceptables: evalúe los problemas, errores y vulnerabilidades inaceptables al evaluar las oportunidades de mejora. Haga un seguimiento de las opciones de redimensionamiento y busque oportunidades de optimización.
    - Últimos boletines de seguridad de AWS
    - AWS Trusted Advisor
    - Cloud Intelligence Dashboards
  - Requisitos de cumplimiento: evalúe las actualizaciones y los cambios necesarios para mantener el cumplimiento de la normativa, la política o la asistencia de terceros cuando revise las oportunidades de mejora.
    - AWS Cumplimiento de
    - Programas de conformidad de AWS
    - Novedades sobre conformidad de AWS

Recursos

Prácticas recomendadas relacionadas:

- OPS01 Prioridades de la organización
- OPS02 Relaciones y propiedades
- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS08 Uso de la observabilidad de la carga de trabajo
- OPS09 Descripción del estado operativo
- OPS11-BP03 Implementación de bucles de retroalimentación

Documentos relacionados:

- Amazon Athena
- Quick
- AWS Cumplimiento de
- Novedades sobre conformidad de AWS
- Programas de conformidad de AWS
- AWS Glue
- Últimos boletines de seguridad de AWS
- AWS Trusted Advisor
- Export your log data to Amazon S3
- Novedades de AWS
- Los aspectos imprescindibles de la innovación centrada en el cliente
- Digital Transformation: Hype or a Strategic Necessity?

Videos relacionados

- AWS re:Invent 2023 - Improve operational efficiency and resilience with Soporte (SUP310)

OPS11-BP06 Validación de la información

Revise los resultados de los análisis y las respuestas con equipos multifuncionales y con los propietarios de la empresa. Use estas revisiones para establecer un entendimiento común, identificar repercusiones adicionales y determinar cursos de acción. Ajuste las respuestas cuando corresponda.

Resultados deseados:

- Revisa la información con los propietarios de la empresa de forma regular. Los propietarios de la empresa proporcionan un contexto adicional a la información recién adquirida.
- Revisa la información y solicita comentarios de sus compañeros técnicos y comparte lo que ha aprendido entre los equipos.
- Publica datos e información para que otros equipos técnicos y de la empresa los revisen. Incorpora lo que ha aprendido en las nuevas prácticas de otros departamentos.
- Resuma y revise la nueva información con los líderes sénior. Los líderes sénior utilizan la nueva información para definir la estrategia.

Patrones comunes de uso no recomendados:

- Lanza una nueva característica. Esta característica cambia algunos de los comportamientos de sus clientes. Su observabilidad no tiene en cuenta estos cambios. No cuantifica los beneficios de estos cambios.
- Publica una nueva actualización y descuida la actualización de su CDN. La caché de CDN ya no es compatible con la última versión. Mide el porcentaje de solicitudes con errores. Todos sus usuarios informan de errores HTTP 400 cuando se comunican con los servidores backend. Investiga los errores del cliente y descubre que midió la dimensión incorrecta y ha perdido el tiempo.
- Su acuerdo de nivel de servicio estipula un tiempo de actividad del 99,9 % y su objetivo de punto de recuperación es de cuatro horas. El propietario del servicio sostiene que el sistema no tiene ningún tiempo de inactividad. Implementa una solución de replicación costosa y compleja, lo que supone una pérdida de tiempo y dinero.

Beneficios de establecer esta práctica recomendada:

- Al validar la información con los propietarios de las unidades de negocio y los expertos en la materia, puede establecer un entendimiento común y orientar las mejoras de forma más eficaz.
- Descubre problemas ocultos y los tiene en cuenta en las decisiones futuras.
- Su enfoque pasa de los resultados técnicos a los resultados empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Validación de la información: colabore con los propietarios de la empresa y los expertos en la materia para asegurarse de que exista un entendimiento común y un acuerdo sobre el significado de los datos que ha recopilado. Identifique las preocupaciones adicionales y las repercusiones potenciales, y determine un curso de acción.

Recursos

Prácticas recomendadas relacionadas:

- OPS01-BP06 Evaluación de las compensaciones al administrar los beneficios y los riesgos
- OPS02-BP06 Responsabilidades predefinidas o negociadas entre equipos
- OPS11-BP03 Implementación de bucles de retroalimentación

Documentos relacionados:

- Designing a Cloud Center of Excellence (CCOE)

Videos relacionados:

- Building observability to increase resiliency

OPS11-BP07 Revisiones de métricas de operaciones

Lleve a cabo análisis retrospectivos periódicos de las métricas de operaciones con participantes de diferentes equipos y áreas de la empresa. Use estas revisiones para identificar las oportunidades de mejora, los posibles cursos de acción y para compartir las lecciones aprendidas. Busque oportunidades para mejorar en todos sus entornos (por ejemplo, desarrollo, pruebas y producción).

Resultado deseado:

- Revisa con frecuencia las métricas que afectan a la empresa
- Detecta y revisa las anomalías a través de sus capacidades de observabilidad
- Utiliza los datos para respaldar los resultados y objetivos empresariales

Patrones comunes de uso no recomendados:

- Su periodo de mantenimiento interrumpe una importante promoción de ventas. La empresa no es consciente de que existe un periodo de mantenimiento estándar que podría retrasarse si se producen otros eventos que afecten a la empresa.
- Ha sufrido una interrupción prolongada porque su organización suele utilizar una biblioteca no actualizada en su organización. Desde entonces, ha migrado a una biblioteca compatible. Los demás equipos de su organización no saben que corren un riesgo.
- No revisa con regularidad el cumplimiento de los SLA de los clientes. Tiende a no cumplir los SLA de los clientes. Existen sanciones económicas relacionadas con el incumplimiento de los SLA de los clientes.

Beneficios de establecer esta práctica recomendada:

- Cuando se reúne con regularidad para revisar las métricas, los eventos y los incidentes de las operaciones, mantiene un entendimiento común entre los equipos.
- Su equipo se reúne de forma rutinaria para revisar las métricas y los incidentes, lo que le permite tomar medidas ante los riesgos y reconocer los SLA de los clientes.
- Comparte las lecciones aprendidas, lo que proporciona datos para establecer prioridades y aplica mejoras específicas para los resultados empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Lleve a cabo análisis retrospectivos periódicos de las métricas de operaciones con participantes de diferentes equipos y áreas de la empresa.
- Involucre a las partes interesadas, incluidos los equipos de negocio, desarrollo y operaciones, para confirmar los resultados obtenidos de los comentarios inmediatos y el análisis retrospectivo, y para compartir las lecciones aprendidas.
- Use sus ideas para identificar las oportunidades de mejora y los posibles cursos de acción.

Recursos

Prácticas recomendadas relacionadas:

- OPS08-BP05 Creación de paneles
- OPS09-BP03 Revisión de las métricas de las operaciones y priorización de las mejoras

- OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas

Documentos relacionados:

- Amazon CloudWatch
- Referencia de métricas y dimensiones de Amazon CloudWatch
- Publish custom metrics
- Uso de métricas de Amazon CloudWatch
- Paneles y visualizaciones con CloudWatch

OPS11-BP08 Documentación y comunicación de las lecciones aprendidas

Documente y comparta las lecciones aprendidas de las actividades de operaciones para poder aplicarlas internamente y entre los equipos. Debe compartir lo que sus equipos aprenden para aumentar el beneficio en toda su organización. Comparta información y recursos para evitar errores evitables y facilitar los esfuerzos de desarrollo, y céntrese en ofrecer las características deseadas.

Utilice AWS Identity and Access Management (IAM) para definir permisos que permitan el acceso controlado a los recursos que desea compartir en las cuentas y entre ellas.

Resultado deseado:

- Utiliza repositorios controlados por versión para compartir bibliotecas de aplicaciones, procedimientos en scripts, documentación de procedimientos y otra documentación del sistema.
- Comparte sus estándares de infraestructura como plantillas de AWS CloudFormation controladas por versiones.
- Revisa las lecciones aprendidas en los equipos.

Patrones comunes de uso no recomendados:

- Ha sufrido una interrupción prolongada porque su organización suele utilizar una biblioteca con errores. Desde entonces, ha migrado a una biblioteca fiable. Los demás equipos de su organización no saben que están en peligro. Nadie documenta ni comparte la experiencia con esta biblioteca, y no son conscientes del riesgo que esto supone.

- Ha identificado un caso límite en un microservicio compartido internamente que provoca la caída de las sesiones. Ha actualizado sus llamadas al servicio para evitar este caso límite. Los demás equipos de su organización no saben que corren un riesgo.
- Ha encontrado una forma de reducir significativamente los requisitos de uso de la CPU para uno de sus microservicios. No sabe si otros equipos podrían aprovechar esta técnica.

Beneficios de establecer esta práctica recomendada: comparta las lecciones aprendidas para apoyar la mejora y maximizar los beneficios de la experiencia.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

- Documentación y comunicación de las lecciones que ha aprendido: disponga de procedimientos para documentar las lecciones aprendidas durante la ejecución de las actividades de operaciones y el análisis retrospectivo para que puedan servir a otros equipos.
- Comunicación de los resultados del aprendizaje: disponga de procedimientos para compartir las lecciones aprendidas y los artefactos asociados entre equipos. Por ejemplo, comparta los procedimientos actualizados, la orientación, la gobernanza y las prácticas recomendadas a través de una wiki accesible. Comparta los scripts, el código y las bibliotecas por medio de un repositorio común.
  - Aproveche AWS re:Post Private como servicio de conocimiento para agilizar la colaboración y el intercambio de conocimientos dentro de su organización.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP06 Responsabilidades predefinidas o negociadas entre equipos
- OPS05-BP01 Uso del control de versiones
- OPS05-BP06 Uso compartido de estándares de diseño
- OPS11-BP03 Implementación de bucles de retroalimentación
- OPS11-BP07 Revisiones de métricas de operaciones

Documentos relacionados:

- Aumente la colaboración y comparta de forma segura los conocimientos sobre la nube con AWS re:Post Private
- Reduce project delays with a docs-as-code solution

Videos relacionados:

- AWS re:Invent 2023 - Collaborate within your company and with AWS using AWS re:Post Private
- Soportes You | Exploring the Incident Management Tabletop Exercise

OPS11-BP09 Asignación de tiempo para implementar mejoras

Dedique tiempo y recursos de sus procesos para hacer posibles las mejoras incrementales continuas.

Resultado deseado:

- Crea duplicados temporales de los entornos, lo que reduce el riesgo, el esfuerzo y el costo de la experimentación y las pruebas.
- Estos ambientes duplicados pueden usarse para probar las conclusiones de su análisis, experimentar, así como para desarrollar y probar las mejoras planeadas.
- Lleva a cabo días de juego y utiliza el Servicio de inyección de errores (FIS) para proporcionar los controles y las barreras de protección que los equipos necesitan para ejecutar experimentos en un entorno similar al de producción.

Patrones comunes de uso no recomendados:

- Hay un problema de rendimiento conocido en su servidor de aplicaciones. Se agrega a las tareas pendientes existentes detrás de cada implementación de características programadas. Si el ritmo al que se agregan las características previstas se mantiene constante, el problema del rendimiento nunca se solucionará.
- Para respaldar la mejora continua, aprueba que los administradores y desarrolladores utilicen todo su tiempo extra para seleccionar e implementar las mejoras. Las mejoras no se completan nunca.
- La aceptación operativa está completa y no se vuelven a probar las prácticas operativas.

Beneficios de establecer esta práctica recomendada: al dedicar tiempo y recursos a sus procesos, hará posibles mejoras continuas e incrementales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

- Asignación de tiempo para efectuar mejoras: dedique tiempo y recursos dentro de sus procesos para llevar a cabo mejoras incrementales continuas.
- Aplique cambios para mejorar y evalúe los resultados para determinar el éxito.
- Si los resultados no alcanzan los objetivos y la mejora sigue siendo una prioridad, busque cursos de acción alternativos.
- Simule las cargas de trabajo de producción durante los días de juego y utilice lo aprendido en estas simulaciones para mejorar.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP08 Uso de varios entornos

Videos relacionados:

- AWS re:Invent 2023 - Improve application resilience with AWS Fault Injection Service

# Seguridad

El pilar de seguridad engloba la capacidad de proteger datos, sistemas y activos para sacar partido de las tecnologías de nube con el fin de mejorar su nivel de seguridad. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de seguridad.

Áreas de prácticas recomendadas

- Aspectos básicos de seguridad
- Identity and Access Management
- Detección
- Protección de la infraestructura
- Protección de datos
- Respuesta a incidentes

- Seguridad de las aplicaciones

## Aspectos básicos de seguridad

Pregunta

- SEC 1. ¿Cómo gestiona su carga de trabajo de forma segura?

### SEC 1. ¿Cómo gestiona su carga de trabajo de forma segura?

Para gestionar su carga de trabajo de forma segura, debe aplicar las prácticas recomendadas generales en todas las áreas de seguridad. Tome los requisitos y procesos que ha definido en materia de excelencia operativa en los niveles organizativo y de carga de trabajo, y aplíquelos a todas las áreas. Mantenerse al día con las recomendaciones de AWS y del sector y con la inteligencia sobre amenazas lo ayuda a desarrollar el modelo de amenazas y los objetivos de control. La automatización de los procesos de seguridad, las pruebas y la validación le permite escalar las operaciones de seguridad.

Prácticas recomendadas

- SEC01-BP01 Separación de cargas de trabajo con cuentas
- SEC01-BP02 Protección del usuario raíz y las propiedades de la cuenta
- SEC01-BP03 Identificación y validación de los objetivos de control
- SEC01-BP04 Actualización constante de las amenazas y recomendaciones de seguridad
- SEC01-BP05 Reducción del alcance de la administración de la seguridad
- SEC01-BP06 Automatización de la implementación de controles de seguridad estándares
- SEC01-BP07 Identificación de amenazas y priorización de mitigaciones con un modelo de amenazas
- SEC01-BP08 Evaluación e implementación de nuevos servicios y características de seguridad de forma periódica

#### SEC01-BP01 Separación de cargas de trabajo con cuentas

Establezca barreras de protección y medidas de aislamiento comunes entre los entornos (por ejemplo, producción, desarrollo y pruebas) y las cargas de trabajo mediante una estrategia de varias cuentas. Es muy recomendable que la separación se haga en la cuenta, ya que así se consigue una barrera de aislamiento sólida para gestionar la seguridad, la facturación y el acceso.

Resultado deseado: una estructura de cuentas que aísla las operaciones en la nube, las cargas de trabajo no relacionadas y los entornos en cuentas independientes, lo que aumenta la seguridad en toda la infraestructura de la nube.

Patrones comunes de uso no recomendados:

- Colocar en la misma cuenta varias cargas de trabajo no relacionadas con diferentes niveles de confidencialidad de los datos.
- Definir de manera insuficiente la estructura de la unidad organizativa (OU).

Beneficios de establecer esta práctica recomendada:

- Menor alcance del impacto si se accede inadvertidamente a una carga de trabajo.
- Gobernanza central del acceso a los servicios, recursos y regiones de AWS.
- Mantenimiento de la seguridad de la infraestructura en la nube con políticas y una administración centralizada de los servicios de seguridad.
- Proceso automatizado de creación y mantenimiento de las cuentas.
- Auditoría centralizada de la infraestructura para los requisitos de conformidad y reglamentarios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las Cuentas de AWS proporcionan una barrera de aislamiento de seguridad entre cargas de trabajo o recursos que operan con distintos niveles de confidencialidad. AWS ofrece herramientas para administrar las cargas de trabajo en la nube a escala mediante una estrategia de varias cuentas para aprovechar esta barrera de aislamiento. Para obtener orientación sobre los conceptos, los patrones y la implementación de una estrategia de varias cuentas en AWS, consulte Organizing Your AWS Environment Using Multiple Accounts.

Cuando tenga varias Cuentas de AWS con una administración central, las cuentas deben organizarse en una jerarquía definida por capas de unidades organizativas (OU). Luego, pueden organizarse y aplicarse controles de seguridad a las OU y a las cuentas miembro mediante el establecimiento de controles preventivos uniformes en las cuentas miembros de la organización. Los controles de seguridad se heredan, lo que permite filtrar los permisos disponibles para las cuentas miembros situadas en niveles inferiores de una jerarquía de OU. Un buen diseño aprovecha esta

herencia para reducir el número y la complejidad de las políticas de seguridad necesarias para lograr los controles de seguridad deseados para cada cuenta miembro.

AWS Organizations y AWS Control Tower son dos servicios que puede utilizar para implementar y administrar esta estructura de varias cuentas en el entorno de AWS. AWS Organizations le permite organizar las cuentas en una jerarquía definida por una o más capas de unidades organizativas, y cada unidad organizativa contiene varias cuentas miembro. Las políticas de control de servicio (SCP) permiten al administrador de la organización establecer controles preventivos detallados en las cuentas miembro, y AWS Config se puede utilizar para establecer controles proactivos y de detección en las cuentas miembro. Muchos servicios de AWS se integran en AWS Organizations para proporcionar controles administrativos delegados y llevar a cabo tareas específicas del servicio en todas las cuentas miembro de la organización.

Además de AWS Organizations, AWS Control Tower ofrece una configuración de prácticas recomendadas con un solo clic para un entorno de AWS de varias cuentas con una zona de aterrizaje. La zona de aterrizaje es el punto de entrada al entorno de varias cuentas que se establece por medio de Control Tower. Control Tower ofrece varias ventajas frente a AWS Organizations. Estas son tres ventajas que mejoran la gobernanza de las cuentas:

- Controles de protección de seguridad obligatorios integrados que se aplican automáticamente a las cuentas que se admiten en la organización.
- Controles de protección opcionales que pueden activarse o desactivarse para un conjunto determinado de OU.
- AWS Control Tower Account Factory ofrece una implementación automatizada de cuentas que contienen bases de referencia y opciones de configuración previamente aprobadas dentro de su organización.

Pasos para la implementación

1. Diseño de una estructura de unidades organizativas: una estructura de unidades organizativas bien diseñada reduce la carga administrativa necesaria para crear y mantener las políticas de control de servicios y otros controles de seguridad. La estructura de la unidad organizativa debe estar alineada con las necesidades empresariales, la confidencialidad de los datos y la estructura de las carga de trabajo.
2. Creación de una zona de aterrizaje para el entorno de varias cuentas: una zona de aterrizaje proporciona una base de seguridad e infraestructura coherente a partir de la cual la organización

puede desarrollar, lanzar e implementar cargas de trabajo rápidamente. Puede utilizar una zona de aterrizaje hecha a medida o AWS Control Tower para organizar el entorno.

3. Establecimiento de barreras de protección: implemente barreras de protección de seguridad uniformes para el entorno a través de la zona de aterrizaje. AWS Control Tower proporciona una lista de controles obligatorios y opcionales que se pueden implementar. Los controles obligatorios se implementan automáticamente al implementar Control Tower. Revise la lista de los controles más recomendables y opcionales, e implemente los controles que sean adecuados a sus necesidades.
4. Restricción del acceso a las regiones recién agregadas: en el caso de las nuevas Regiones de AWS, los recursos de IAM, como los usuarios y roles, solo se propagan a las regiones que especifique. Esta acción se puede llevar a cabo a través de la consola cuando se utiliza Control Tower o mediante el ajuste de las políticas de permisos de IAM en AWS Organizations.
5. Consideración de uso AWS CloudFormation StackSets: StackSets le ayuda a implementar recursos que incluyen políticas, roles y grupos de IAM en diferentes regiones y Cuentas de AWS a partir de una plantilla aprobada.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP04 Uso de un proveedor de identidades centralizado

Documentos relacionados:

- AWS Control Tower
- Directivas de auditoría de seguridad de AWS
- Prácticas recomendadas de IAM
- Use CloudFormation StackSets to provision resources across multiple Cuentas de AWS and regions
- Preguntas frecuentes sobre Organizations
- Terminología y conceptos de AWS Organizations
- Best Practices for Service Control Policies in an AWS Organizations Multi-Account Environment
- AWS Account Management Reference Guide
- Organización de su entorno de AWS con varias cuentas

Videos relacionados:

- Enable AWS adoption at scale with automation and governance
- Security Best Practices the Well-Architected Way
- Building and Governing Multiple Accounts using AWS Control Tower
- Enable Control Tower for Existing Organizations

SEC01-BP02 Protección del usuario raíz y las propiedades de la cuenta

El usuario raíz es el usuario con más privilegios de una Cuenta de AWS. Tiene acceso administrativo completo a todos los recursos de la cuenta y, en algunos casos, no se puede limitar con políticas de seguridad. Deshabilitar el acceso programático al usuario raíz, establecer controles apropiados para este usuario y evitar su uso rutinario ayuda a reducir el riesgo de exposición inadvertida de las credenciales raíz y el consiguiente peligro que esto supone para el entorno de la nube.

Resultado deseado: proteger al usuario raíz ayuda a reducir la posibilidad de que se produzcan daños accidentales o intencionados por el mal uso de las credenciales del usuario raíz. Establecer controles de detección también puede servir para alertar al personal adecuado cuando se llevan a cabo acciones con el usuario raíz.

Patrones comunes de uso no recomendados:

- Utilizar el usuario raíz para llevar a cabo tareas que no se encuentran entre las pocas que requieren credenciales de usuario raíz.
- Dejar de comprobar periódicamente los planes de contingencia para verificar el funcionamiento de las infraestructuras críticas, los procesos y el personal durante una emergencia.
- Considerar únicamente el flujo de inicio de sesión típico de la cuenta y olvidarse de considerar o probar métodos alternativos de recuperación de la cuenta.
- No ocuparse de DNS, servidores de correo electrónico y proveedores de telefonía como parte del perímetro crítico de seguridad, ya que estos se utilizan en el flujo de recuperación de la cuenta.

Beneficios de establecer esta práctica recomendada: proteger el acceso al usuario raíz aumenta la confianza de que las acciones de la cuenta están controladas y auditadas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

AWS dispone de muchas herramientas para proteger su cuenta. Sin embargo, dado que algunas de estas medidas no están activadas de forma predeterminada, deberá implementarlas directamente. Considere estas recomendaciones como pasos básicos para proteger la Cuenta de AWS. A medida que vaya implementando estos pasos, es importante que cree un proceso para evaluar y supervisar continuamente los controles de seguridad.

Cuando crea una Cuenta de AWS por primera vez, empieza con una sola identidad que tiene acceso completo a todos los servicios y recursos de AWS de la cuenta. Esta identidad recibe el nombre de usuario raíz de la Cuenta de AWS. Puede iniciar sesión como usuario raíz utilizando la dirección de correo electrónico y contraseña que usó al crear la cuenta. Debido al acceso elevado que se concede al usuario raíz de AWS, debe limitar el uso del usuario raíz de AWS únicamente a las tareas que lo requieran específicamente. Las credenciales de inicio de sesión del usuario raíz deben estar muy bien protegidas y siempre se debe utilizar la autenticación multifactor (MFA) para el usuario raíz de la Cuenta de AWS.

Además del flujo de autenticación normal para iniciar sesión con el usuario raíz mediante un nombre de usuario, una contraseña y un dispositivo de autenticación multifactor (MFA), existen flujos de recuperación de la cuenta para iniciar sesión con el usuario raíz de la Cuenta de AWS que tiene acceso a la dirección de correo electrónico y al número de teléfono asociados a la cuenta. Por lo tanto, también es muy importante proteger la cuenta de correo electrónico del usuario raíz a la que se envía el mensaje de recuperación y el número de teléfono asociado a la cuenta. Tenga en cuenta también las posibles dependencias circulares si la dirección de correo electrónico asociada al usuario raíz está alojada en servidores de correo electrónico o recursos del servicio de nombres de dominio (DNS) de la misma Cuenta de AWS.

Cuando se utiliza AWS Organizations, hay varias Cuentas de AWS y cada una de ellas tiene un usuario raíz. Se designa una cuenta como cuenta de administración y, a continuación, se pueden agregar varias capas de cuentas miembro por debajo de esa cuenta de administración. Priorice la seguridad del usuario raíz de su cuenta de administración y, luego, céntrese en los usuarios raíz de las cuentas miembros. La estrategia para proteger el usuario raíz de la cuenta de administración puede ser diferente de la de los usuarios raíz de las cuentas miembro, y puede colocar controles de seguridad preventivos en los usuarios raíz de las cuentas miembro.

Pasos para la implementación

Se recomienda seguir estos pasos de implementación para establecer controles para el usuario raíz. Cuando corresponda, las recomendaciones se cotejan con la versión 1.4.0 del punto de referencia

de CIS AWS Foundations. Además de estos pasos, consulte las pautas de prácticas recomendadas de AWS para proteger los recursos y la Cuenta de AWS.

Controles preventivos

1. Configure información de contacto precisa para la cuenta.
    a. Esta información se utiliza para el flujo de recuperación de las contraseñas perdidas, el flujo de recuperación de cuentas de los dispositivos MFA perdidos y para comunicaciones críticas relacionadas con la seguridad con su equipo.
    b. Utilice una dirección de correo electrónico alojada en su dominio corporativo (preferiblemente una lista de distribución) como dirección de correo electrónico del usuario raíz. Al utilizar una lista de distribución en lugar de la cuenta de correo electrónico de una persona, se consigue redundancia y continuidad adicionales para acceder a la cuenta raíz durante largos periodos de tiempo.
    c. El número de teléfono que figure en la información de contacto debe ser un teléfono dedicado y seguro para este fin. El número de teléfono no debe figurar en ninguna parte ni compartirse con nadie.
2. No cree claves de acceso para el usuario raíz. Si existen claves de acceso, elimínelas (CIS 1.4).
    a. Elimine cualquier credencial programática de larga duración (claves de acceso y secretas) para el usuario raíz.
    b. Si ya existen claves de acceso del usuario raíz, debe hacer la transición de los procesos que utilizan esas claves para utilizar claves de acceso temporales de un rol de AWS Identity and Access Management (IAM) y, a continuación, eliminar las claves de acceso del usuario raíz.
3. Determine si necesita almacenar las credenciales del usuario raíz.
    a. Si utiliza AWS Organizations para crear nuevas cuentas miembro, la contraseña inicial del usuario raíz de esas nuevas cuentas miembro se establece en un valor aleatorio que no se le revela. Considere la posibilidad de utilizar el flujo de restablecimiento de contraseñas de la cuenta de administración de su organización de AWS para acceder a la cuenta de miembro si es necesario.
    b. Para Cuentas de AWS independientes o la cuenta de administración de AWS, considere la posibilidad de crear y almacenar de forma segura credenciales para el usuario raíz. Utilice MFA para el usuario raíz.
4. Use controles preventivos para los usuarios raíz de las cuentas miembro en entornos de varias cuentas de AWS.

a. Considere la posibilidad de utilizar la barrera de protección preventiva No permitir la creación de claves de acceso raíz para el usuario raíz como barrera preventiva para las cuentas de los miembros.

b. Considere la posibilidad de utilizar la barrera de protección preventiva No permitir la creación de claves de acceso raíz para el usuario raíz para las cuentas de los miembros.

5. Si necesita credenciales para el usuario raíz:

a. Utilice una contraseña compleja.

b. Active la autenticación multifactor (MFA) para el usuario raíz, especialmente para las cuentas de administración de AWS Organizations (pagador) (CIS 1.5).

c. Considere la posibilidad de usar dispositivos MFA físicos para mejorar la resiliencia y la seguridad, ya que los dispositivos de un solo uso pueden reducir las posibilidades de que los dispositivos que contienen los códigos MFA puedan reutilizarse para otros fines. Verifique que los dispositivos MFA físicos que funcionan con baterías se sustituyan periódicamente (CIS 1.6).

- Para configurar la MFA en el usuario raíz, siga las instrucciones a fin de crear un dispositivo MFA virtual o un dispositivo MFA físico.

d. Considere la posibilidad de inscribir varios dispositivos MFA para hacer copias de seguridad. Se permiten hasta 8 dispositivos MFA por cuenta.

- Tenga en cuenta que al inscribir más de un dispositivo MFA para el usuario raíz, se desactiva automáticamente el flujo de recuperación de la cuenta en caso de pérdida del dispositivo MFA.

e. Guarde la contraseña con todas las medidas de seguridad y tenga en cuenta las dependencias circulares si la guarda electrónicamente. No guarde la contraseña de forma que sea necesario acceder a la misma Cuenta de AWS para obtenerla.

6. Opcional: considere la posibilidad de establecer un programa de rotación periódica de contraseñas para el usuario raíz.

- Las prácticas recomendadas de administración de credenciales dependen de los requisitos de las normativas y políticas que tenga. Los usuarios raíz protegidos por MFA no dependen de una contraseña como único factor de autenticación.
- Si se cambia la contraseña del usuario raíz de forma periódica, se reduce el riesgo de que una contraseña expuesta de forma inadvertida se utilice indebidamente.

Controles de detección

- Cree alarmas para detectar el uso de las credenciales del usuario raíz (CIS 1.7). Amazon GuardDuty puede supervisar y alertar sobre el uso de las credenciales de las API del usuario raíz mediante el resultado RootCredentialUsage.
- Evalúe e implemente los controles de detección incluidos en el paquete de conformidad del pilar de seguridad de AWS Well-Architected para AWS Config o, si utiliza AWS Control Tower, los controles más recomendados disponibles en Control Tower.

Guía operativa

- Determine qué persona de la organización debe tener acceso a las credenciales del usuario raíz.
  - Utilice la regla de dos personas para no haya una sola persona que tenga acceso a todas las credenciales y el dispositivo MFA necesarios para obtener acceso de usuario raíz.
  - Compruebe que sea la organización, y no una única persona, quien mantenga un control del número de teléfono y el alias de correo electrónico asociados a la cuenta (que se utilizan para el restablecimiento de la contraseña y el flujo de restablecimiento de MFA).
- Utilice el usuario raíz únicamente de forma excepcional (CIS 1.7).
  - El usuario raíz de AWS no debe utilizarse para las tareas diarias, ni siquiera para las tareas administrativas. Inicie sesión únicamente como usuario raíz para llevar a cabo las tareas de AWS que lo requieran. Todas las demás acciones deben hacerlas otros usuarios que asuman los roles apropiados.
- Compruebe periódicamente que el acceso al usuario raíz funcione con el fin de probar los procedimientos antes de que se produzca una situación de emergencia que requiera el uso de las credenciales del usuario raíz.
- Compruebe periódicamente que la dirección de correo electrónico asociada a la cuenta y las que aparecen en Contactos alternativos funcionen. Supervise las bandejas de entrada de estas direcciones de correo electrónico para comprobar si se reciben notificaciones de seguridad de `<abuse@amazon.com>`. Asegúrese también de que los números de teléfono asociados a la cuenta funcionen.
- Prepare procedimientos de respuesta a incidentes para responder al uso indebido de la cuenta raíz. Consulte AWS Security Incident Response Guide y las prácticas recomendadas en la sección Respuesta ante incidentes del documento técnico sobre el Pilar de seguridad para obtener más información sobre cómo crear una estrategia de respuesta a incidentes adecuada para su Cuenta de AWS.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP01 Separación de cargas de trabajo con cuentas
- SEC02-BP01 Uso de mecanismos de inicio de sesión sólidos
- SEC03-BP02 Concesión de acceso con privilegios mínimos
- SEC03-BP03 Establecimiento de un proceso de acceso de emergencia
- SEC10-BP05 Aprovisionamiento previo del acceso

Documentos relacionados:

- AWS Control Tower
- Directivas de auditoría de seguridad de AWS
- Prácticas recomendadas de IAM
- Amazon GuardDuty – root credential usage alert
- Step-by-step guidance on monitoring for root credential use through CloudTrail
- Tokens de MFA aprobados para su uso con AWS
- Implementación del acceso de emergencia en AWS
- Top 10 security items to improve in your Cuenta de AWS
- What do I do if I notice unauthorized activity in my Cuenta de AWS?

Videos relacionados:

- Enable AWS adoption at scale with automation and governance
- Security Best Practices the Well-Architected Way
- Limiting use of AWS root credentials from AWS re:inforce 2022 – Security best practices with AWS IAM

SEC01-BP03 Identificación y validación de los objetivos de control

En función de sus requisitos de cumplimiento y los riesgos identificados a partir de su modelo de amenazas, extraiga y valide los objetivos de control y los controles que tiene que aplicar a su carga

de trabajo. La validación continua tanto de los objetivos de control como de los controles le ayuda a medir la efectividad de la mitigación de riesgos.

Resultado deseado: los objetivos de control de seguridad de su empresa están bien definidos y alineados con sus requisitos de conformidad. Se implementan y ponen en marcha controles mediante la automatización y las políticas, y se evalúan de forma continua con el fin de determinar su eficacia para lograr sus objetivos. Poner a disposición de los auditores demostraciones de eficacia, tanto en un momento determinado como durante un periodo de tiempo.

Patrones comunes de uso no recomendados:

- Incomprensión por parte de la empresa de los requisitos normativos, las expectativas del mercado y los estándares del sector en cuanto al control de la seguridad.
- Alineación incorrecta de los marcos de ciberseguridad y los objetivos de control con los requisitos de la empresa.
- Ausencia de una correspondencia estrecha y medible entre la implementación de los controles y los objetivos de control.
- Falta de uso de la automatización para informar sobre la eficacia de los controles.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Existen muchos marcos de ciberseguridad comunes que pueden constituir la base de sus objetivos de control de seguridad. Debe tener en cuenta los requisitos normativos, las expectativas del mercado y los estándares del sector para su empresa con el objetivo de determinar qué marcos se adaptan mejor a sus necesidades. Entre los ejemplos, se incluyen AICPA SOC 2, HITRUST, PCI-DSS, ISO 27001 y NIST SP 800-53.

Una vez identificados los objetivos de control, debe analizar cómo los servicios de AWS de los que hace uso le ayudan a conseguir dichos objetivos. Utilice AWS Artifact para buscar la documentación y los informes que se alineen con sus marcos objetivo que describan el alcance de la responsabilidad cubierta por AWS y una guía para el ámbito restante que caiga bajo su responsabilidad. Para obtener más orientación específica sobre los servicios que se ajusten a las diversas declaraciones de control del marco, consulte AWS Customer Compliance Guides.

A medida que defina unos controles que sirvan para lograr sus objetivos, reglamente su aplicación mediante controles preventivos y automatice las mitigaciones mediante controles de detección. Ayude a evitar configuraciones y acciones de recursos no conformes en todo su sistema de AWS

Organizations mediante las políticas de control de servicios (SCP). Implemente reglas en AWS Config para supervisar los recursos que no cumplan con las normas e informar sobre ellos y, a continuación, cambie las reglas a un modelo de cumplimiento una vez que esté seguro de su comportamiento. Para implementar conjuntos de reglas predefinidas y administradas que se ajusten a sus marcos de ciberseguridad, evalúe el uso de estándares de AWS Security Hub CSPM como primera opción. El estándar Prácticas recomendadas de seguridad básicas (FSBP) de AWS y el CIS AWS Foundations Benchmark son buenos puntos de partida y contienen controles alineados con muchos de los objetivos comunes en varios marcos estándares. Cuando Security Hub CSPM no cuente intrínsecamente con las detecciones de control deseadas, puede complementarse con paquetes de conformidad de AWS Config.

Utilice los paquetes para socios de APN recomendados por el equipo de AWS Global Security and Compliance Acceleration (GSCA) para obtener asistencia de asesores de seguridad, agencias consultoras, sistemas de recopilación de pruebas y presentación de informes, auditores y otros servicios complementarios cuando sea necesario.

Pasos para la implementación

1. Valore los marcos de ciberseguridad comunes y alinee sus objetivos de control con los marcos elegidos.
2. Obtenga la documentación pertinente sobre la orientación y las responsabilidades de su marco con AWS Artifact. Determine qué partes del cumplimiento corresponden a AWS según el modelo de responsabilidad compartida y qué partes son de su responsabilidad.
3. Utilice SCP, políticas de recursos, políticas de confianza de roles y otras barreras de protección para evitar configuraciones y acciones de recursos no conformes.
4. Evalúe la implementación de estándares de Security Hub CSPM y paquetes de conformidad de AWS Config que se alineen con sus objetivos de control.

Recursos

Prácticas recomendadas relacionadas:

- SEC03-BP01 Definición de los requisitos de acceso
- SEC04-BP01 Configuración del registro de servicios y aplicaciones
- SEC07-BP01 Comprensión del esquema de clasificación de datos
- OPS01-BP03 Evaluación de los requisitos de gobernanza
- OPS01-BP04 Evaluación de los requisitos de cumplimiento

- PERF01-BP05 Uso de políticas y arquitecturas de referencia
- COST02-BP01 Desarrollo de políticas basadas en los requisitos de su organización

Documentos relacionados:

- AWS Customer Compliance Guides

Herramientas relacionadas:

- AWS Artifact

SEC01-BP04 Actualización constante de las amenazas y recomendaciones de seguridad

Para mantenerse al día de las amenazas y mitigaciones más recientes, supervise las publicaciones de inteligencia sobre amenazas del sector y analice las fuentes de datos en busca de actualizaciones. Evalúe las ofertas de servicios administrados que se actualizan automáticamente en función de los datos de amenazas más recientes.

Resultado deseado: se mantiene al día a medida que las publicaciones del sector se actualizan con las amenazas y recomendaciones más recientes.  Utiliza la automatización para detectar posibles vulnerabilidades y exposiciones a medida que se identifiquen nuevas amenazas. Toma medidas de mitigación para estas amenazas.  Adopta servicios de AWS que se actualicen automáticamente con la información de amenazas más reciente.

Patrones comunes de uso no recomendados:

- No disponer de un mecanismo fiable y repetible para mantenerse al día de la información más reciente sobre amenazas.
- Mantener un inventario manual de su cartera de tecnología, cargas de trabajo y dependencias que requiera una revisión humana para detectar posibles vulnerabilidades y exposiciones.
- No disponer de mecanismos para actualizar sus cargas de trabajo y dependencias a las versiones más recientes disponibles que incluyan mitigaciones de amenazas conocidas.

Beneficios de establecer esta práctica recomendada: el uso de orígenes de inteligencia sobre amenazas para mantenerse al día reduce el riesgo de perderse cambios importantes en el panorama de amenazas que puedan afectar a su empresa.  Utilizar la automatización para analizar, detectar y corregir posibles vulnerabilidades o exposiciones en sus cargas de trabajo y sus dependencias

puede ayudarle a mitigar los riesgos de forma rápida y predecible, en comparación con las alternativas manuales. Esto ayuda a controlar el tiempo y los costos relacionados con la mitigación de vulnerabilidades.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Revise las publicaciones fiables de inteligencia sobre amenazas para mantenerse al día del panorama de amenazas. Consulte la base de conocimiento de MITRE ATT&CK para obtener documentación sobre tácticas, técnicas y procedimientos (TTP) de confrontación conocidos. Consulte la lista de vulnerabilidades y exposiciones comunes (CVE) de MITRE para mantenerse al tanto de las vulnerabilidades conocidas de los productos que utiliza. Conozca los riesgos críticos de las aplicaciones web con el conocido proyecto OWASP Top 10 de Open Worldwide Application Security Project (OWASP).

Manténgase al día de los eventos de seguridad de AWS y las medidas de corrección recomendadas con los boletines de seguridad de AWS para las CVE.

Con el objetivo de reducir el esfuerzo general y los gastos que supone mantenerse al día, plantéese el uso de servicios de AWS que incorporen automáticamente nueva información sobre amenazas con el paso del tiempo. Por ejemplo, Amazon GuardDuty se mantiene al día con la inteligencia de amenazas del sector para detectar comportamientos anómalos y firmas de amenazas en las cuentas. Amazon Inspector mantiene actualizada automáticamente una base de datos de las CVE que utiliza para sus características de análisis continuo. Tanto AWS WAF como AWS Shield Advanced ofrecen grupos de reglas administrados que se actualizan automáticamente a medida que surgen nuevas amenazas.

Revise el pilar de excelencia operativa de Well-Architected para automatizar la administración de flotas y la aplicación de parches.

Pasos para la implementación

- Suscríbase a las actualizaciones de las publicaciones de inteligencia sobre amenazas que resulten pertinentes para su negocio y su sector. Suscríbase a los boletines de seguridad de AWS.
- Considere la posibilidad de adoptar servicios que incorporen automáticamente nuevos conocimientos sobre amenazas, como Amazon GuardDuty y Amazon Inspector.
- Implemente una estrategia de administración de flotas y aplicación de parches acorde con las prácticas recomendadas en el pilar de excelencia operativa de Well-Architected.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP07 Identificación de amenazas y priorización de mitigaciones con un modelo de amenazas
- OPS01-BP05 Evaluación del panorama de amenazas
- OPS11-BP01 Implementación de un proceso de mejora continua

SEC01-BP05 Reducción del alcance de la administración de la seguridad

Determine si puede reducir su alcance de seguridad mediante el uso de servicios de AWS que transfieran la administración de ciertos controles a AWS (servicios administrados). Estos servicios pueden ayudar a reducir las tareas de mantenimiento de la seguridad, como el aprovisionamiento de infraestructuras, la configuración del software, la aplicación de parches o las copias de seguridad.

Resultado deseado: tenga en cuenta el alcance de la administración de la seguridad al seleccionar los servicios de AWS para su carga de trabajo. El costo de los gastos generales de administración y las tareas de mantenimiento (el costo total de propiedad o TCO) se compara con el costo de los servicios que seleccione, además de otras consideraciones relacionadas con el marco de Well-Architected. Incorpora la documentación de control y cumplimiento de AWS en sus procedimientos de evaluación y verificación del control.

Patrones comunes de uso no recomendados:

- Implementar cargas de trabajo sin comprender a fondo el modelo de responsabilidad compartida para los servicios que seleccione.
- Alojar bases de datos y otras tecnologías en máquinas virtuales sin haber evaluado un servicio administrado equivalente.
- No incluir las tareas de administración de la seguridad en el costo total de la propiedad de las tecnologías de host en máquinas virtuales en comparación con las opciones de servicios administrados.

Beneficios de establecer esta práctica recomendada: el uso de servicios administrados puede reducir la carga general que supone administrar los controles de seguridad operativos, lo que puede reducir los riesgos de seguridad y el costo total de la propiedad. El tiempo que de otro modo se dedicaría a determinadas tareas de seguridad puede reinvertirse en tareas que aporten más valor a la empresa.

Los servicios administrados también pueden reducir el alcance de sus requisitos de cumplimiento al trasladar algunos requisitos de control a AWS.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Hay varias formas en las que puede integrar los componentes de la carga de trabajo en AWS. La instalación y ejecución de tecnologías en instancias de Amazon EC2 suele requerir que asuma la mayor parte de la responsabilidad general sobre la seguridad. Para ayudar a reducir la carga de poner en práctica ciertos controles, identifique los servicios administrados de AWS que reduzcan su ámbito de responsabilidad en el modelo de responsabilidad compartida y piense en cómo puede utilizarlos en su arquitectura actual. Entre los ejemplos, se incluye el uso de Amazon Relational Database Service (Amazon RDS) para implementar bases de datos, Amazon Elastic Kubernetes Service (Amazon EKS) o Amazon Elastic Container Service (Amazon ECS) para orquestar contenedores, o el uso de opciones sin servidor. Al desarrollar nuevas aplicaciones, piense en qué servicios pueden ayudar a reducir el tiempo y el costo a la hora de implementar y administrar los controles de seguridad.

Los requisitos de cumplimiento también pueden ser un factor determinante a la hora de seleccionar los servicios. Los servicios administrados pueden trasladar la responsabilidad del cumplimiento de algunos requisitos a AWS. Hable con su equipo de cumplimiento sobre si se sentirían cómodos al auditar los aspectos de los servicios que opera y administra y al aceptar las declaraciones de control en los informes de auditoría de AWS pertinentes. Puede proporcionar los artefactos de auditoría que se encuentran en AWS Artifact a sus auditores o reguladores como prueba de los controles de seguridad de AWS. También puede utilizar la guía de responsabilidad que ofrecen algunos de los artefactos de auditoría de AWS para diseñar la arquitectura, junto con AWS Customer Compliance Guides. Esta guía ayuda a determinar los controles de seguridad adicionales que debe poner en práctica para permitir los casos de uso específicos de su sistema.

Cuando utilice servicios administrados, debe familiarizarse con el proceso de actualización de los recursos a versiones más recientes (por ejemplo, actualizar la versión de una base de datos administrada por Amazon RDS o el tiempo de ejecución de un lenguaje de programación para una función de AWS Lambda). Si bien el servicio administrado puede llevar a cabo esta operación automáticamente, sigue siendo su responsabilidad configurar el momento de la actualización y comprender el impacto en sus operaciones. Herramientas como AWS Health pueden ayudarle a hacer un seguimiento de estas actualizaciones y administrarlas en todos sus entornos.

Pasos para la implementación

1. Evalúe los componentes de la carga de trabajo que puedan sustituirse por un servicio administrado.
   a. Si está migrando una carga de trabajo a AWS, tenga en cuenta la simplificación de la administración (tiempo y gastos) y la reducción del riesgo al evaluar si debe volver a alojar, refactorizar, redefinir la plataforma, reconstruir o reemplazar la carga de trabajo. A veces, la inversión adicional al inicio de una migración puede generar ahorros significativos a largo plazo.
2. Considere la posibilidad de implementar servicios administrados, como Amazon RDS, en lugar de instalar y administrar implementaciones de su tecnología propia.
3. Utilice la guía de responsabilidades de AWS Artifact para determinar los controles de seguridad que debe poner en práctica para la carga de trabajo.
4. Mantenga un inventario de los recursos que se están utilizando y manténgase al día de los nuevos servicios y enfoques para identificar nuevas oportunidades y reducir el alcance.

Recursos

Prácticas recomendadas relacionadas:

- PERF02-BP01 Selección de las mejores opciones computacionales para su carga de trabajo
- PERF03-BP01 Uso de un almacén de datos personalizado que se adapte mejor a los requisitos de acceso y almacenamiento de datos
- SUS05-BP03 Uso de servicios administrados

Documentos relacionados:

- Planned lifecycle events for AWS Health

Herramientas relacionadas:

- AWS Health
- AWS Artifact
- AWS Customer Compliance Guides

Videos relacionados:

- How do I migrate to an Amazon RDS or Aurora MySQL DB instance using AWS DMS?
- AWS re:Invent 2023 - Manage resource lifecycle events at scale with AWS Health

SEC01-BP06 Automatización de la implementación de controles de seguridad estándares

Aplique prácticas modernas de DevOps a medida que desarrolle e implemente controles de seguridad estándar en todos sus entornos de AWS. Defina controles y configuraciones de seguridad estándar mediante plantillas de infraestructura como código (IaC), registre los cambios en un sistema de control de versiones, pruebe los cambios como parte de una canalización de CI/CD y automatice la implementación de los cambios en sus entornos de AWS.

Resultado deseado: las plantillas de IaC capturan los controles de seguridad estandarizados y los envían a un sistema de control de versiones. Existen canalizaciones de CI/CD en lugares en los que se detectan cambios y se automatizan las pruebas y la implementación de sus entornos de AWS. Existen barreras de protección para detectar errores de configuración en las plantillas y alertar sobre ellos antes de proceder a la implementación. Se implementan cargas de trabajo en entornos donde existan controles estándar. Los equipos tienen acceso para implementar configuraciones de servicio aprobadas a través de un mecanismo de autoservicio. Existen estrategias de copia de seguridad y recuperación seguras para controlar las configuraciones, los scripts y los datos relacionados.

Patrones comunes de uso no recomendados:

- Hacer cambios en los controles de seguridad estándar de forma manual, mediante una consola web o una interfaz de línea de comandos.
- Confiar en los equipos de carga de trabajo individuales para implementar manualmente los controles que define un equipo central.
- Confiar en un equipo de seguridad central para implementar controles en el nivel de la carga de trabajo a petición de un equipo de carga de trabajo.
- Permitir que las mismas personas o equipos desarrollen, prueben e implementen scripts de automatización del control de seguridad sin una separación adecuada de funciones ni de controles y contrapesos.

Beneficios de establecer esta práctica recomendada: el uso de plantillas para definir los controles de seguridad estándar permite hacer un seguimiento y comparar los cambios a lo largo del tiempo mediante un sistema de control de versiones. El uso de la automatización para probar e implementar los cambios crea estandarización y previsibilidad, lo que aumenta las posibilidades de que la

implementación se complete correctamente y reduce las tareas manuales repetitivas. Proporcionar un mecanismo de autoservicio para que los equipos de carga de trabajo implementen los servicios y configuraciones aprobados reduce el riesgo de errores de configuración y usos indebidos. Esto también les ayuda a incorporar controles en las primeras etapas del proceso de desarrollo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si sigue las prácticas descritas en SEC01-BP01 Separación de cargas de trabajo con cuentas, terminará teniendo varias Cuentas de AWS para diferentes entornos que administre mediante AWS Organizations. Si bien es posible que cada uno de estos entornos y cargas de trabajo necesite controles de seguridad diferentes, puede estandarizar algunos controles de seguridad en toda la organización. Entre algunos ejemplos de esto se incluyen la integración de proveedores de identidad centralizados, la definición de redes y firewalls y la configuración de ubicaciones estándar para almacenar y analizar los registros. Del mismo modo en que puede utilizar la infraestructura como código (IaC) para aplicar el mismo rigor en el desarrollo del código de aplicación al aprovisionamiento de infraestructuras, también puede utilizar la IaC para definir e implementar los controles de seguridad estándar.

Siempre que sea posible, defina los controles de seguridad de forma declarativa, como en AWS CloudFormation, y almacénelos en un sistema de control de código fuente. Utilice las prácticas de DevOps para automatizar la implementación de los controles para obtener versiones más predecibles, pruebas automatizadas con herramientas como AWS CloudFormation Guard y detectar desviaciones entre los controles implementados y la configuración deseada. Puede utilizar servicios, como AWS CodePipeline, AWS CodeBuild y AWS CodeDeploy, para crear una canalización de CI/CD. Tenga en cuenta las instrucciones de Organizing Your AWS Environment Using Multiple Accounts para configurar estos servicios en sus propias cuentas que sean independientes de otras canalizaciones de implementación.

También puede definir plantillas para estandarizar la definición y la implementación de Cuentas de AWS, servicios y configuraciones. Esta técnica permite que un equipo de seguridad central administre estas definiciones y se las proporcione a los equipos de la carga de trabajo mediante un enfoque de autoservicio. Una forma de lograrlo es mediante Service Catalog, donde puede publicar plantillas como productos que los equipos de la carga de trabajo pueden incorporar a las implementaciones de su propia canalización. Si utiliza AWS Control Tower, hay disponibles algunas plantillas y controles como punto de partida. Control Tower también ofrece la función Account Factory, lo que permite a los equipos de la carga de trabajo crear nuevas Cuentas de AWS con los estándares que defina. Esta función ayuda a eliminar las dependencias de un equipo central para

aprobar y crear nuevas cuentas cuando los equipos de la carga de trabajo las identifiquen como necesarias.  Es posible que necesite estas cuentas para aislar los diferentes componentes de la carga de trabajo en función de motivos como la función que cumplen, la confidencialidad de los datos que se procesan o su comportamiento.

Pasos para la implementación

1. Determine cómo va a almacenar y mantener las plantillas en un sistema de control de versiones.
2. Cree canalizaciones de CI/CD para probar e implementar las plantillas.  Defina pruebas para comprobar si hay errores de configuración y si las plantillas se ajustan a los estándares de su empresa.
3. Cree un catálogo de plantillas estandarizadas para que los equipos de la carga de trabajo implementen Cuentas de AWS y servicios de acuerdo con sus requisitos.
4. Implemente estrategias de copia de seguridad y recuperación seguras para sus configuraciones de control, scripts y datos relacionados.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP01 Uso del control de versiones
- OPS05-BP04 Uso de sistemas de administración de compilación e implementación
- REL08-BP05 Implementación de cambios con automatización
- SUS06-BP01 Adopción de métodos que permitan introducir mejoras en la sostenibilidad rápidamente

Documentos relacionados:

- Organización de su entorno de AWS con varias cuentas

Ejemplos relacionados:

- Automate account creation, and resource provisioning using Service Catalog, AWS Organizations, and AWS Lambda
- Strengthen the DevOps pipeline and protect data with AWS Secrets Manager, AWS KMS, and AWS Gestor de certificados

Herramientas relacionadas:

- AWS CloudFormation Guard
- Landing Zone Accelerator on AWS

SEC01-BP07 Identificación de amenazas y priorización de mitigaciones con un modelo de amenazas

Utilice el modelado de amenazas para identificar y mantener un registro actualizado de las amenazas potenciales y las mitigaciones asociadas para la carga de trabajo. Priorice sus amenazas y adapte sus mitigaciones de controles de seguridad para evitarlas, detectarlas y responder a ellas. Revisite y mantenga todo esto en el contexto de la carga de trabajo y de la evolución del panorama de seguridad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Qué es el modelado de amenazas?

“El modelado de amenazas sirve para identificar, comunicar y comprender las amenazas y las mitigaciones en el contexto de la protección de algo de valor”. – Threat Modeling de Open Web Application Security Project (OWASP

Por qué debería crear un modelo de amenazas?

Los sistemas son complejos y, con el tiempo, se hacen más complejos y potentes aún, por lo que aportan más valor empresarial y aumentan la satisfacción y el compromiso de los clientes. Esto significa que, en las decisiones de diseño de TI, se deben tener en cuenta un número cada vez mayor de casos de uso. Debido a esta complejidad y al número de combinaciones de casos de uso, los enfoques no estructurados suelen resultar ineficaces para encontrar y mitigar las amenazas. En su lugar, se necesita un enfoque sistemático para encontrar las amenazas potenciales para el sistema, pero también para concebir mitigaciones y priorizarlas para asegurarse de que los limitados recursos de la organización tengan el máximo impacto en la mejora de la postura de seguridad general del sistema.

El modelado de amenazas está diseñado para proporcionar este enfoque sistemático, con el objetivo de encontrar y abordar los problemas en las primeras fases del proceso de diseño, cuando las mitigaciones tienen un costo y un esfuerzo relativamente bajos en comparación con las fases posteriores del ciclo de vida. Este enfoque se alinea con el principio del sector relativo al desplazamiento a la izquierda de la seguridad. En última instancia, el modelado de amenazas se

integra en el proceso de administración de riesgos de una organización y ayuda a tomar decisiones sobre qué controles aplicar mediante un enfoque basado en las amenazas.

Cuándo se debe llevar a cabo el modelado de amenazas?

Empiece a modelar las amenazas lo antes posible en el ciclo de vida de la carga de trabajo, ya que así tendrá más flexibilidad para actuar en relación con las amenazas que identifique. Al igual que ocurre con los errores de software, cuanto antes identifique las amenazas, más rentable le resultará abordarlas. Un modelo de amenazas es un documento vivo y debe evolucionar a medida que cambien las cargas de trabajo. Revisite los modelos de amenazas a lo largo del tiempo, especialmente cuando se produzca un cambio importante, un cambio en el panorama de las amenazas o cuando adopte una nueva característica o servicio.

Pasos para la implementación

Cómo podemos llevar a cabo el modelado de amenazas?

Hay muchas formas diferentes de llevar a cabo el modelado de amenazas. Al igual que ocurre con los lenguajes de programación, cada una tiene sus ventajas y sus inconvenientes, por lo que debe elegir la que mejor le convenga. Un enfoque consiste en empezar con Shostack's 4 Question Frame for Threat Modeling, que plantea preguntas abiertas para estructurar el ejercicio de modelado de amenazas:

1. En qué estamos trabajando?

   La finalidad de esta pregunta es ayudarle a comprender y acordar el sistema que está creando y los detalles de ese sistema que son pertinentes para la seguridad. Crear un modelo o diagrama es la forma más popular de responder a esta pregunta, ya que le ayuda a visualizar lo que está creando, por ejemplo, mediante un diagrama de flujo de datos. Anotar las suposiciones y los detalles importantes sobre el sistema también le ayuda a definir el alcance del trabajo. De esta manera, todas las personas que contribuyen al modelo de amenazas pueden centrarse en lo mismo, y evita dar largos rodeos hacia temas que están fuera del alcance (lo que incluye versiones desactualizadas del sistema). Por ejemplo, si crea una aplicación web, probablemente no merezca la pena que modele la secuencia de arranque de confianza del sistema operativo para los clientes del navegador, ya que no tiene capacidad para influir en esto con su diseño.
2. Qué puede salir mal?

   Aquí es donde se identifican las amenazas que afectan al sistema. Las amenazas son acciones o acontecimientos accidentales o intencionados que tienen repercusiones no deseadas y podrían

afectar a la seguridad del sistema. Si no tiene una idea clara de lo que podría salir mal, no podrá hacer nada al respecto.

No existe una lista formal de lo que puede salir mal. La creación de esta lista requiere una lluvia de ideas y la colaboración de todas las personas del equipo y las personas pertinentes involucradas en el ejercicio de modelado de amenazas. Para facilitar la reflexión, puede utilizar un modelo para identificar amenazas, como STRIDE, que sugiere distintas categorías para evaluarlas: suplantación de identidad, manipulación, repudio, divulgación de información, denegación de servicio y elevación de privilegios. Además, para contribuir a la lluvia de ideas, tal vez quiera consultar las listas existentes y buscar inspiración; por ejemplo, en OWASP Top 10, HiTrust Threat Catalog y el catálogo de amenazas propio de su organización.

3. Qué vamos a hacer al respecto?

   Igual que en la pregunta anterior, no existe una lista formal de todas las mitigaciones posibles. En este paso, tenemos las amenazas, los actores y las áreas de mejora identificados en el paso anterior.

   Los asuntos relacionados con la seguridad y la conformidad son una responsabilidad compartida entre el cliente y AWS. Es importante entender que, cuando se pregunta “¿Qué vamos a hacer al respecto?”, también se está preguntando “¿Quién es responsable de hacer algo al respecto?”. Comprender el reparto de responsabilidades entre el cliente y AWS le ayuda a delimitar el modelado de amenazas a las mitigaciones que están bajo su control, que suelen ser una combinación de opciones de configuración de los servicios de AWS y las mitigaciones específicas de su propio sistema.

   En cuanto a la parte de AWS de la responsabilidad compartida, descubrirá que los servicios de AWS forman parte del ámbito de aplicación de muchos programas de cumplimiento. Estos programas le ayudan a conocer los sólidos controles que hay en AWS para mantener la seguridad y la conformidad de la nube. Los informes de auditoría de estos programas están disponibles para que los clientes de AWS los descarguen de AWS Artifact.

   Independientemente de los servicios de AWS que utilice, el cliente siempre tiene una parte de la responsabilidad, y las mitigaciones que se corresponden con estas responsabilidades deben incluirse en el modelo de amenazas. En cuanto a las mitigaciones de los controles de seguridad de los propios servicios de AWS, debe considerar la posibilidad de implementar controles de seguridad en todos los dominios, como los de administración de identidades y accesos (autenticación y autorización), protección de datos (en reposo y en tránsito), seguridad de la infraestructura, registro y supervisión. La documentación de cada servicio de AWS incluye

un capítulo dedicado a la seguridad que proporciona orientación sobre los controles de seguridad que se deben considerar como medidas de mitigación. Y lo que es más importante, considere el código que está escribiendo y sus dependencias, y piense en los controles que podría establecer para hacer frente a esas amenazas. Estos controles pueden ser, por ejemplo, la validación de entradas, la gestión de sesiones y la gestión de límites. Muchas veces, la mayoría de las vulnerabilidades se introducen en el código personalizado, así que céntrese en esta área.

4. Hicimos un buen trabajo?

   El objetivo es que su equipo y su organización mejoren con el tiempo tanto la calidad de los modelos de amenazas como la velocidad a la que los llevan a cabo. Estas mejoras se deben a una combinación de práctica, aprendizaje, enseñanza y revisión. Para profundizar y ponerse manos a la obra, le recomendamos que usted y su equipo completen el curso de formación o el taller sobre modelado de amenazas para constructores. Además, si busca orientación sobre cómo integrar el modelado de amenazas en el ciclo de vida del desarrollo de aplicaciones de su organización, consulte la publicación How to approach threat modeling en el blog de seguridad de AWS.

Threat Composer

Como ayuda y orientación a la hora de modelar las amenazas, considere la posibilidad de utilizar la herramienta Threat Composer, cuyo objetivo es reducir el tiempo que se tarda en generar valor a la hora de crear modelos de amenazas. La herramienta le ayuda a hacer lo siguiente:

- Escribir instrucciones de amenazas útiles alineadas con la gramática de amenazas que funcionen en un flujo de trabajo natural y no lineal
- Generar un modelo de amenazas legible por humanos
- Generar un modelo de amenazas legible por máquina que le permita tratar los modelos de amenazas como código
- Ayudarle a identificar rápidamente las áreas de mejora de la calidad y la cobertura mediante el panel de información

Para obtener más información, visite Threat Composer y cambie al espacio de trabajo de ejemplo definido por el sistema.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP03 Identificación y validación de los objetivos de control
- SEC01-BP04 Actualización constante de las amenazas y recomendaciones de seguridad
- SEC01-BP05 Reducción del alcance de la administración de la seguridad
- SEC01-BP08 Evaluación e implementación de nuevos servicios y características de seguridad de forma periódica

Documentos relacionados:

- How to approach threat modeling (blog de seguridad de AWS)
- NIST: Guide to Data-Centric System Threat modeling

Videos relacionados:

- AWS Summit ANZ 2021 - How to approach threat modelling
- AWS Summit ANZ 2022 - Scaling security – Optimise for fast and secure delivery

Formación relacionada:

- Threat modeling the right way for builders – AWS Skill Builder virtual self-paced training
- Threat modeling the right way for builders – AWS Workshop

Herramientas relacionadas:

- Threat Composer

SEC01-BP08 Evaluación e implementación de nuevos servicios y características de seguridad de forma periódica

Evalúe e implemente servicios y características de seguridad de AWS y socios de AWS que le ayuden a desarrollar la postura de seguridad de su carga de trabajo.

Resultado deseado: cuenta con una práctica estándar que le informa sobre las nuevas características y servicios lanzados por AWS y socios de AWS. Evalúe en qué medida estas nuevas capacidades influyen en el diseño de los controles actuales y nuevos para sus entornos y cargas de trabajo.

Patrones comunes de uso no recomendados:

- No se suscribe a blogs de AWS y fuentes RSS para enterarse rápidamente de las nuevas características y servicios pertinentes.
- Confía en las noticias y actualizaciones sobre los servicios y características de seguridad de fuentes de segunda mano
- No fomenta entre los usuarios de AWS de su organización a mantenerse al día de las últimas actualizaciones

Beneficios de establecer esta práctica recomendada: si está al tanto de los nuevos servicios y características de seguridad, puede tomar decisiones informadas sobre la implementación de controles en cargas de trabajo y entornos de nube. Estos orígenes ayudan a concienciar sobre la evolución del panorama de seguridad y sobre cómo se pueden utilizar los servicios de AWS para protegerse contra las amenazas nuevas y emergentes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

AWS informa a los clientes sobre los nuevos servicios y características de seguridad a través de varios canales:

- AWS Novedades de
- AWS Blog de noticias de
- AWS Blog de seguridad de
- AWS Boletines de seguridad de
- AWS Descripción general de la documentación de

Puede suscribirse a un tema de actualizaciones diarias de características de AWS mediante Amazon Simple Notification Service (Amazon SNS) para obtener un resumen diario completo de las actualizaciones. Algunos servicios de seguridad, como Amazon GuardDuty y AWS Security Hub CSPM, ofrecen sus propios temas de SNS para mantenerse al día sobre los nuevos estándares, resultados y otras actualizaciones de esos servicios en particular.

Los nuevos servicios y características también se anuncian y describen en detalle durante las conferencias, eventos y seminarios web que se llevan a cabo en todo el mundo cada año. Cabe

destacar la conferencia anual de seguridad AWS re:Inforce y la conferencia de carácter más general, AWS re:Invent. Los canales de noticias de AWS mencionados anteriormente comparten estos anuncios de conferencias sobre seguridad y otros servicios, y pueden verse las sesiones temáticas informativas en línea en el canal de eventos de AWS en YouTube.

También puede preguntar a su equipo de Cuenta de AWS sobre las últimas actualizaciones y recomendaciones de los servicios de seguridad. Puede contactar con su equipo a través del formulario de soporte de ventas si no dispone de su información de contacto directo. Del mismo modo, si se suscribió a AWS Enterprise Support, recibirá actualizaciones semanales de su administrador técnico de cuentas (TAM) y podrá programar una reunión de revisión periódica con dicho administrador.

Pasos para la implementación

1. Suscríbase a los distintos blogs y boletines con su lector de RSS favorito o al tema de SNS sobre actualizaciones de características diarias.
2. Evalúe a qué eventos de AWS asistir para conocer de primera mano las nuevas características y servicios.
3. Programe reuniones con su equipo de Cuenta de AWS para resolver cualquier duda sobre la actualización de los servicios y características de seguridad.
4. Plantéese la posibilidad de suscribirse a Enterprise Support para acceder a consultas periódicas con un gerente técnico de cuentas (TAM).

Recursos

Prácticas recomendadas relacionadas:

- PERF01-BP01 Descubrimiento y comprensión de los servicios y las características disponibles en la nube
- COST01-BP07 Seguimiento de la información sobre las nuevas versiones de los servicios

## Identity and Access Management

Preguntas

- SEC 2. ¿Cómo administra la autenticación para personas y máquinas?
- SEC 3. ¿Cómo se gestionan los permisos de las personas y las máquinas?

## SEC 2. ¿Cómo administra la autenticación para personas y máquinas?

Hay dos tipos de identidades que debe administrar al abordar la operación de cargas de trabajo de AWS seguras.

- Identidades humanas: las identidades humanas que requieren acceso a sus entornos y aplicaciones de AWS se pueden clasificar en tres grupos: personal, terceros y usuarios.

  El grupo de personal incluye administradores, desarrolladores y operadores que forman parte de su organización. Necesitan acceso para administrar, crear y operar sus recursos de AWS.

  Los terceros son colaboradores externos, como contratistas, proveedores o socios. Interactúan con sus recursos de AWS como parte de su compromiso con su organización.

  Los usuarios son quienes utilizan sus aplicaciones. Acceden a los recursos de AWS a través de navegadores web, aplicaciones de cliente, aplicaciones móviles o herramientas de línea de comandos interactivas.
- Identidades de máquinas: las aplicaciones de carga de trabajo, las herramientas operativas y los componentes requieren una identidad para hacer solicitudes a los servicios de AWS, por ejemplo, para leer datos. Estas identidades incluyen máquinas que se ejecutan en su entorno de AWS, como instancias de Amazon EC2 o funciones de AWS Lambda. También se podrían administrar identidades de máquina para las partes externas a AWS que necesiten acceso a su entorno de AWS.

Prácticas recomendadas

- SEC02-BP01 Uso de mecanismos de inicio de sesión sólidos
- SEC02-BP02 Uso de credenciales temporales
- SEC02-BP03 Almacenamiento y uso seguros de secretos
- SEC02-BP04 Uso de un proveedor de identidades centralizado
- SEC02-BP05 Auditoría y rotación periódicas de las credenciales
- SEC02-BP06 Uso de grupos y atributos de usuarios

### SEC02-BP01 Uso de mecanismos de inicio de sesión sólidos

Los inicios de sesión (autenticación mediante credenciales de inicio de sesión) pueden presentar riesgos si no se utilizan mecanismos como la autenticación multifactor (MFA), especialmente en

situaciones en las que las credenciales de inicio de sesión se han revelado de forma inadvertida o son fáciles de adivinar. Utilice mecanismos de inicio de sesión sólidos para reducir estos riesgos. Para ello, exija que se cumplan políticas de contraseñas sólidas y se utilice MFA.

Resultado deseado: reduzca los riesgos de acceso no deseado a las credenciales en AWS mediante el uso de mecanismos de inicio de sesión sólidos para los usuarios de AWS Identity and Access Management (IAM), el usuario raíz de la Cuenta de AWS, AWS IAM Identity Center y los proveedores de identidades de terceros. Esto significa exigir que se use MFA, aplicar políticas de contraseñas sólidas y detectar comportamientos de inicio de sesión anómalos.

Patrones comunes de uso no recomendados:

- No aplicar una política de contraseñas segura para sus identidades que incluya contraseñas complejas y MFA.
- Compartir las mismas credenciales entre diferentes usuarios.
- No utilizar controles de detección de inicios de sesión sospechosos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Existen muchas formas en que las identidades humanas puedan iniciar sesión en AWS. Una práctica recomendada de AWS es confiar en un proveedor de identidades centralizado que utilice la federación (federación directa SAML 2.0 entre AWS IAM y el IdP centralizado o usando AWS IAM Identity Center) a la hora de autenticarse en AWS. En ese caso, deberá establecer un proceso de inicio de sesión seguro con su proveedor de identidades o Microsoft Active Directory.

Cuando abre una Cuenta de AWS por primera vez, comienza con un usuario raíz de la Cuenta de AWS. Solo debe usar el usuario raíz de la cuenta para configurar el acceso de los usuarios (y para las tareas que requieren el usuario raíz). Es importante activar la autenticación multifactor (MFA) para el usuario raíz de la cuenta inmediatamente después de abrir la Cuenta de AWS y proteger al usuario raíz según la guía de prácticas recomendadas de AWS.

AWS IAM Identity Center está diseñado para usuarios de la plantilla, y puede crear y administrar las identidades de los usuarios dentro del servicio y proteger el proceso de inicio de sesión con MFA. AWS Cognito, por otro lado, está diseñado para la gestión de la identidad y el acceso de los clientes (CIAM), que proporciona grupos de usuarios y proveedores de identidad para las identidades de los usuarios externos en sus aplicaciones.

Si crea usuarios en el AWS IAM Identity Center, proteja el proceso de inicio de sesión en ese servicio y active la MFA. Para las identidades de los usuarios externos en sus aplicaciones, puede utilizar grupos de usuarios de Amazon Cognito y proteger el proceso de inicio de sesión en ese servicio, o bien utilizar uno de los proveedores de identidades compatibles con los grupos de usuarios de Amazon Cognito.

Además, en el caso de los usuarios del Centro de Identidad de AWS IAM, se puede utilizar Acceso verificado de AWS para proporcionar un nivel de seguridad adicional mediante la verificación de la identidad del usuario y la posición del dispositivo antes de que se les conceda acceso a los recursos de AWS.

Si utiliza usuarios de AWS Identity and Access Management (IAM), debe proteger el proceso de inicio de sesión mediante IAM.

Puede utilizar tanto AWS IAM Identity Center y la federación de IAM directa de forma simultánea para gestionar el acceso a AWS. Puede utilizar la federación de IAM para administrar el acceso a la Consola de administración de AWS y a los servicios e IAM Identity Center para administrar el acceso a aplicaciones empresariales como Quick o Amazon Q Business.

Independientemente del método de inicio de sesión que se utilice, es fundamental aplicar una política de inicio de sesión sólida.

Pasos para la implementación

Estas son recomendaciones generales para un inicio de sesión sólido. Los ajustes reales que configure deben estar establecidos por la política de la empresa o utilizar un estándar como NIST 800-63.

- Require MFA (Requerir MFA): Es práctica recomendada de IAM exigir la MFA para las identidades humanas y las cargas de trabajo. Si se activa MFA, habrá una capa adicional de seguridad que exigirá que los usuarios proporcionen credenciales de inicio de sesión y una contraseña de un solo uso (OTP) o una cadena que se verifica criptográficamente y se genera desde un dispositivo físico.
- Imponga una longitud mínima para la contraseña. Esto es un factor fundamental para la seguridad de la contraseña.
- Imponga una complejidad de las contraseñas para que sean más difíciles de adivinar.
- Permita que los usuarios cambien sus contraseñas.
- Cree identidades individuales en lugar de credenciales compartidas. Si crea identidades individuales, puede dar a cada usuario un conjunto único de credenciales de seguridad. Tener usuarios individuales permite auditar la actividad de cada uno de ellos.

Recomendaciones de IAM Identity Center:

- IAM Identity Center proporciona una política de contraseñas predefinida cuando se utiliza el directorio predeterminado, que establece la longitud, la complejidad y los requisitos de reutilización de las contraseñas.
- Active la MFA y configure los ajustes contextuales o permanentes para la MFA cuando el origen de la identidad sea el directorio predeterminado, AWS Managed Microsoft AD, o AD Connector.
- Permita a los usuarios registrar sus propios dispositivos MFA.

Recomendaciones del directorio de grupos de usuarios de Amazon Cognito:

- Configurar los ajustes de Seguridad de la contraseña.
- Requiera MFA para los usuarios.
- Utilizar la configuración de seguridad avanzada de grupos de usuarios de Amazon Cognito para las características, como la autenticación adaptativa, que puede bloquear los inicios de sesión sospechosos.

Recomendaciones de usuarios de IAM:

- Lo ideal es que utilice IAM Identity Center o la federación directa. Sin embargo, es posible que necesite usuarios de IAM. En ese caso, establezca una política de contraseñas para los usuarios de IAM. Puede usar una política de contraseñas para definir requisitos, tales como la longitud mínima o si deben contener caracteres no alfanuméricos.
- Cree una política de IAM para imponer el inicio de sesión con MFA de modo que los usuarios puedan administrar sus propias contraseñas y dispositivos MFA.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP03 Almacenamiento y uso seguros de secretos
- SEC02-BP04 Uso de un proveedor de identidades centralizado
- SEC03-BP08 Uso compartido de recursos de forma segura en su organización

Documentos relacionados:

- Política de contraseñas del AWS IAM Identity Center
- Política de contraseñas para usuarios de IAM
- Configuración de la contraseña del usuario raíz de la Cuenta de AWS
- Política de contraseñas de Amazon Cognito
- Credenciales de AWS
- Prácticas recomendadas de seguridad de IAM

Videos relacionados:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC02-BP02 Uso de credenciales temporales

Al llevar a cabo cualquier tipo de autenticación, es mejor utilizar credenciales temporales en lugar de credenciales de larga duración para reducir o eliminar riesgos. Por ejemplo, que las credenciales se divulguen, compartan o roben de forma inadvertida.

Resultado deseado: para reducir el riesgo de credenciales a largo plazo, utilice credenciales temporales siempre que sea posible para las identidades humanas y de máquinas. Las credenciales de larga duración entrañan muchos riesgos. Por ejemplo, podrían subirse a repositorios públicos y quedar expuestas. Al utilizar credenciales temporales, reducirá enormemente las posibilidades de que las credenciales se vean comprometidas.

Patrones comunes de uso no recomendados:

- Desarrolladores que utilizan claves de acceso de larga duración de usuarios de IAM en lugar de obtener credenciales temporales de la CLI mediante federación.
- Desarrolladores que incrustan claves de acceso de larga duración en su código y suben ese código a repositorios de Git públicos.
- Desarrolladores que incrustan claves de acceso de larga duración en aplicaciones móviles que luego se ponen a disposición de todo el mundo en las tiendas de aplicaciones.
- Usuarios que comparten claves de acceso de larga duración con otros usuarios, o empleados que abandonan la empresa con claves de acceso de larga duración aún en su poder.
- Utilizar claves de acceso de larga duración para identidades de máquinas cuando podrían utilizarse credenciales temporales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Utilice credenciales de seguridad temporales en lugar de credenciales de larga duración para todas las solicitudes de la API y la AWS CLI. Las solicitudes de la API y la CLI a los servicios de AWS deben, en casi todos los casos, firmarse mediante claves de acceso de AWS. Estas solicitudes pueden firmarse con credenciales temporales o de larga duración. La única vez que debe utilizar credenciales de larga duración, también conocidas como claves de acceso a largo plazo, es si utiliza un usuario de IAM o un usuario raíz de la Cuenta de AWS. Al federarse en AWS o asumir un rol de IAM mediante otros métodos, se generan credenciales temporales. Incluso cuando accede a la Consola de administración de AWS mediante credenciales de inicio de sesión, se generan credenciales temporales para que pueda hacer llamadas a los servicios de AWS. Hay pocas situaciones en las que necesite credenciales de larga duración, y casi todas las tareas se pueden llevar a cabo mediante credenciales temporales.

Evitar el uso de credenciales de larga duración en favor de credenciales temporales debería acompañarse de una estrategia de reducción del uso de usuarios de IAM a favor de la federación y los roles de IAM. Aunque en el pasado se han utilizado usuarios de IAM tanto para identidades humanas como de máquinas, ahora recomendamos no utilizarlos para evitar los riesgos que conlleva el uso de claves de acceso de larga duración.

Pasos para la implementación

Identidades humanas

Para identidades de la plantilla, como las de empleados, administradores, desarrolladores, operadores y clientes:

- Debería basarse en un proveedor de identidades centralizado y exigir que los usuarios humanos utilicen la federación con un proveedor de identidades para acceder a AWS con credenciales temporales. La federación de los usuarios se puede efectuar mediante la federación directa a cada Cuenta de AWS o con AWS IAM Identity Center y el proveedor de identidades que prefiera. La federación tiene una serie de ventajas con respecto a los usuarios de IAM, además de eliminar las credenciales de larga duración. Los usuarios también pueden solicitar credenciales temporales desde la línea de comandos para la federación directa o mediante IAM Identity Center. Esto significa que hay pocos casos de uso que requieran usuarios de IAM o credenciales de larga duración para los usuarios.

Para identidades de terceros:

- Al conceder a terceros, como a proveedores de software como servicio (SaaS), acceso a los recursos en su Cuenta de AWS, puede utilizar roles entre cuentas y políticas basadas en recursos. Además, puede utilizar el flujo de concesión de credenciales de cliente de Amazon Cognito OAuth 2.0 para clientes o socios de SaaS B2B.

Las identidades de usuario que acceden a los recursos de AWS a través de navegadores web, aplicaciones de cliente, aplicaciones móviles o herramientas de línea de comando interactivas:

- Si necesita conceder a las aplicaciones para consumidores o clientes acceso a los recursos de su AWS, puede utilizar grupos de identidades de Amazon Cognito o grupos de usuarios de Amazon Cognito para proporcionar credenciales temporales. Los permisos de las credenciales se controlan mediante los roles de IAM que cree. También puede definir un rol de IAM independiente con permisos limitados para los usuarios invitados que no estén autenticados.

Identidades de máquina

En el caso de las identidades de máquina, puede que necesite utilizar credenciales de larga duración. En estos casos, puede exigir que las cargas de trabajo utilicen credenciales temporales con roles de IAM para acceder a AWS.

- Para Amazon Elastic Compute Cloud (Amazon EC2), puede utilizar roles para Amazon EC2.
- AWS Lambda permite configurar un rol de ejecución de Lambda para conceder al servicio permisos para llevar a cabo acciones de AWS mediante credenciales temporales. Existen muchos otros modelos similares para que los servicios de AWS concedan credenciales temporales con roles de IAM.
- En el caso de los dispositivos de IoT, puede utilizar el proveedor de credenciales de AWS IoT Core para solicitar credenciales temporales.
- Para los sistemas en las instalaciones o los que se ejecutan fuera de AWS que necesitan acceso a los recursos de AWS, puede utilizar IAM Roles Anywhere.

Hay escenarios en los que no se admiten credenciales temporales, pero requieren el uso de credenciales de larga duración. En estas situaciones, audite y rote las credenciales periódicamente y rote las claves de acceso periódicamente. En el caso de las claves de acceso de los usuarios de IAM muy restringidas, tenga en cuenta las siguientes medidas de seguridad adicionales:

- Otorgue permisos muy restringidos:
  - Cumpla con el principio de privilegio mínimo (sea específico en cuanto a las acciones, los recursos y las condiciones).
  - Plantéese la opción de conceder al usuario de IAM solo la operación AssumeRole para un rol específico. En función de la arquitectura local, este enfoque ayuda a aislar y proteger las credenciales de IAM a largo plazo.
- Limite las fuentes de red y las direcciones IP permitidas en la política de confianza de roles de IAM.
- Supervise el uso y configure alertas para detectar permisos no utilizados o mal uso (mediante alarmas y filtros de métricas de AWS CloudWatch Logs).
- Exija el cumplimiento de los límites de los permisos (las políticas de control de servicios [SCP]) y los límites de permisos se complementan entre sí: las SCP son muy generales, mientras que los límites de los permisos son más precisos).
- Implemente un proceso para aprovisionar y almacenar de forma segura (en un almacén local) las credenciales.

Entre las opciones para los escenarios que requieren credenciales a largo plazo también se incluyen:

- Crear su propia API de venta de tokens (mediante Amazon API Gateway).
- En situaciones en las que deba utilizar credenciales de larga duración o para credenciales distintas de las claves de acceso de AWS (como los inicios de sesión en bases de datos), puede utilizar un servicio diseñado para gestionar los secretos, como AWS Secrets Manager. Secrets Manager simplifica la administración, la rotación y el almacenamiento seguro de los secretos cifrados. Muchos servicios de AWS admiten la integración directa con Secrets Manager.
- Para las integraciones multinube, puede utilizar la federación de identidades en función de las credenciales del proveedor de servicios de credenciales (CSP) de origen (consulte AWS STS AssumeRoleWithWebIdentity).

Para obtener más información sobre cómo cambiar las credenciales de larga duración, consulte cómo rotar las claves de acceso.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP03 Almacenamiento y uso seguros de secretos

- SEC02-BP04 Uso de un proveedor de identidades centralizado
- SEC03-BP08 Uso compartido de recursos de forma segura en su organización

Documentos relacionados:

- Credenciales de seguridad temporales
- Credenciales de AWS
- Prácticas recomendadas de seguridad de IAM
- Roles de IAM
- IAM Identity Center
- Federación y proveedores de identidades
- Rotación de las claves de acceso
- Soluciones de socios de seguridad: acceso y control de acceso
- El usuario raíz de la cuenta de AWS
- Access AWS using a Google Cloud Platform native workload identity
- How to access AWS resources from Microsoft Entra ID tenants using AWS Security Token Service

Videos relacionados:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC02-BP03 Almacenamiento y uso seguros de secretos

Una carga de trabajo necesita una capacidad automatizada para demostrar su identidad a bases de datos, recursos y servicios de terceros. Para ello, se utilizan credenciales de acceso secretas, como claves de acceso a API, contraseñas y tokens OAuth. El uso de un servicio creado específicamente para almacenar, administrar y rotar estas credenciales ayuda a reducir la probabilidad de que dichas credenciales se vean comprometidas.

Resultado deseado: implementación de un mecanismo de administración segura de las credenciales de las aplicaciones que logre los siguientes objetivos:

- Identificar qué secretos son necesarios para la carga de trabajo.

- Reducir el número de credenciales de larga duración necesarias y sustituirlas por credenciales de corta duración cuando sea posible.
- Establecer un almacenamiento seguro y una rotación automatizada de las credenciales restantes de larga duración.
- Auditar el acceso a los secretos que existen en la carga de trabajo.
- Supervisar de forma continua para verificar que no se incruste ningún secreto en el código fuente durante el proceso de desarrollo.
- Reduzca la probabilidad de que las credenciales se divulguen por accidente.

Patrones comunes de uso no recomendados:

- No rotar las credenciales.
- Almacenar credenciales a largo plazo en el código fuente o en archivos de configuración.
- Almacenar credenciales en reposo sin cifrar.

Beneficios de establecer esta práctica recomendada:

- Los secretos se almacenan cifrados en reposo y en tránsito.
- El acceso a las credenciales se controla a través de una API (considérela como una máquina expendedora de credenciales).
- El acceso a una credencial (tanto de lectura como de escritura) se audita y registra.
- Separación de preocupaciones: la rotación de credenciales la hace un componente independiente, que puede separarse del resto de la arquitectura.
- Los secretos se distribuyen automáticamente bajo demanda a los componentes de software, y la rotación se produce en una ubicación central.
- El acceso a las credenciales puede controlarse de forma detallada.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

En el pasado, las credenciales que se utilizaban para autenticarse en bases de datos, API de terceros, tokens y otros secretos podían estar incrustadas en el código fuente o en archivos del entorno. AWS proporciona varios mecanismos para almacenar estas credenciales de forma segura, rotarlas automáticamente y auditar su uso.

La mejor manera de abordar la administración de secretos es seguir la norma de eliminar, sustituir y rotar. La credencial más segura es aquella que no se tiene que almacenar, administrar ni manejar. Es posible que haya credenciales que ya no sean necesarias para el funcionamiento de la carga de trabajo y que, por tanto, puedan eliminarse de forma segura.

En el caso de las credenciales que siguen siendo necesarias para el correcto funcionamiento de la carga de trabajo, podría existir la oportunidad de sustituir una credencial de larga duración por una credencial temporal o de corta duración. Por ejemplo, en lugar de codificar rígidamente una clave de acceso secreta de AWS, considere la posibilidad de sustituir esa credencial de larga duración por una credencial temporal a través de roles de IAM.

Es posible que algunos secretos de larga duración no puedan eliminarse ni sustituirse. Estos secretos se pueden almacenar en un servicio, como AWS Secrets Manager, donde se pueden almacenar, administrar y rotar de forma centralizada periódicamente.

Una auditoría del código fuente y de los archivos de configuración de la carga de trabajo puede revelar muchos tipos de credenciales. La siguiente tabla resume las estrategias para manejar los tipos comunes de credenciales:

| Tipo de credenciales | Descripción | Estrategia sugerida |
| --- | --- | --- |
| claves de acceso de IAM | Claves de acceso y secretas de AWS IAM que se utilizan para asumir roles de IAM dentro de una carga de trabajo | Reemplazo: utilice los roles de IAM asignados a las instancias de cómputo (como Amazon EC2 o AWS Lambda) en su lugar. Para garantizar la interoperabilidad con terceros que tengan que acceder a los recursos en la Cuenta de AWS, pregunte si admiten el acceso entre cuentas de AWS. En el caso de aplicaciones móviles, considere la posibilidad de usar credenciales temporales a través de grupos de identidades de Amazon Cognito (identidades federadas). Para las cargas |

| Tipo de credenciales | Descripción | Estrategia sugerida |
| --- | --- | --- |
| | | de trabajo que se ejecutan fuera de AWS, considere IAM Roles Anywhere o Activaciones híbridas de AWS Systems Manager. Para los contenedores, consulte el rol de IAM de la tarea de Amazon ECS o el rol de IAM del nodo de Amazon EKS. |
| Claves de SSH | Las claves privadas de Secure Shell se utilizan para iniciar sesión en las instancias de EC2 de Linux, de forma manual o como parte de un proceso automatizado | Reemplazo: utilice AWS Systems Manager o EC2 Instance Connect para proporcionar acceso mediante programación y humano a las instancias de EC2 mediante roles de IAM. |
| Credenciales de aplicaciones y bases de datos | Contraseñas: cadena de texto sin formato | Rotación: almacene las credenciales en AWS Secrets Manager y establezca una rotación automatizada si es posible. |
| Credenciales de Amazon RDS y Aurora Admin Database | Contraseñas: cadena de texto sin formato | Reemplazo: utilice la integración de Secrets Manager en Amazon RDS o Amazon Aurora. Además, algunos tipos de bases de datos de RDS pueden utilizar roles de IAM en lugar de contraseñas en algunos casos de uso (para obtener más información, consulte Autenticación de bases de datos de IAM). |

| Tipo de credenciales | Descripción | Estrategia sugerida |
| --- | --- | --- |
| Tokens OAuth | Tokens secretos: cadena de texto sin formato | Rotación: almacene los tokens en AWS Secrets Manager y configure la rotación automatizada. |
| Tokens y claves de API | Tokens secretos: cadena de texto sin formato | Rotación: almacénelas en AWS Secrets Manager y establezca una rotación automatizada si es posible. |

Un patrón común de uso no recomendado es incrustar claves de acceso de IAM dentro del código fuente, los archivos de configuración o las aplicaciones móviles. Cuando se necesite una clave de acceso de IAM para comunicarse con un servicio de AWS, utilice credenciales de seguridad temporales (a corto plazo). Estas credenciales a corto plazo se pueden proporcionar mediante roles de IAM para instancias de EC2, roles de ejecución para funciones de Lambda, roles de IAM de Cognito para el acceso de usuarios móviles y políticas de IoT Core para dispositivos de IoT. Al interactuar con terceros, dé prioridad a delegar el acceso a un rol de IAM con el acceso necesario a los recursos de su cuenta en lugar de configurar un usuario de IAM y enviar al tercero la clave de acceso secreta de ese usuario.

Hay muchos casos en los que la carga de trabajo requiere el almacenamiento de los secretos necesarios para interoperar con otros servicios y recursos. AWS Secrets Manager está diseñado específicamente para administrar estas credenciales de forma segura, así como para el almacenamiento, el uso y la rotación de los tokens de API, las contraseñas y otras credenciales.

AWS Secrets Manager proporciona cinco funciones clave para garantizar el almacenamiento y la administración seguros de las credenciales confidenciales: cifrado en reposo, cifrado en tránsito, auditoría exhaustiva, control de acceso detallado y rotación de credenciales ampliable. También son aceptables otros servicios de administración de secretos de socios de AWS o soluciones desarrolladas localmente que proporcionen capacidades y garantías similares.

Cuando recupera un secreto, puede utilizar el componente de almacenamiento en caché del cliente de Secrets Manager para utilizarlo más adelante. Recuperar un secreto almacenado en la memoria caché es más rápido que recuperarlo desde Secrets Manager. Asimismo, dado que la llamada a las

API de Secrets Manager conlleva un coste, el uso de una caché puede reducirlo. Para conocer todas las formas en las que puede recuperar secretos, consulte Obtener secretos.

> **Note**
>
> Algunos lenguajes pueden pedirle que implemente su propio cifrado en memoria para el almacenamiento en caché del cliente.

Pasos para la implementación

1. Identifique las rutas de código que contienen credenciales con codificación rígida mediante herramientas automatizadas, como Amazon CodeGuru.
   a. Utilice Amazon CodeGuru para analizar los repositorios de código. Una vez finalizada la revisión, filtre por Type=Secrets en CodeGuru para encontrar líneas de código que presentan problemas.
2. Identifique las credenciales que pueden eliminarse o sustituirse.
   a. Identifique las credenciales que ya no sean necesarias y márquelas para eliminarlas.
   b. En el caso de las claves secretas de AWS que estén incrustadas en el código fuente, sustitúyalas por roles de IAM asociados a los recursos necesarios. Si parte de la carga de trabajo se encuentra fuera de AWS, pero requiere credenciales de IAM para acceder a los recursos de AWS, considere la posibilidad de usar IAM Roles Anywhere o Activaciones híbridas de AWS Systems Manager.
3. Para otros secretos de terceros de larga duración que requieran el uso de la estrategia de rotación, integre Secrets Manager en el código para recuperar secretos de terceros en tiempo de ejecución.
   a. La consola de CodeGuru puede crear automáticamente un secreto en Secrets Manager con las credenciales detectadas.
   b. Integre la recuperación de secretos desde Secrets Manager en el código de la aplicación.
      i. Las funciones de Lambda sin servidor pueden utilizar una extensión de Lambda independiente del lenguaje.
      ii. Para las instancias o contenedores de EC2, AWS proporciona un ejemplo de código del lado del cliente para recuperar secretos de Secrets Manager en varios lenguajes de programación populares.

4. Revise periódicamente la base de código y vuelva a analizarla para verificar que no se hayan agregado nuevos secretos al código.
   a. Considere la posibilidad de utilizar una herramienta, como git-secrets, para evitar confirmar nuevos secretos en el repositorio del código fuente.
5. Supervisión de la actividad de Secrets Manager para detectar indicios de uso inesperado, acceso inapropiado a secretos o intentos de eliminar secretos.
6. Reduzca la exposición humana a las credenciales. Restrinja el acceso para leer, escribir y modificar credenciales a un rol de IAM dedicado a este fin, y solo proporcione acceso para asumir el rol a un pequeño subconjunto de usuarios operativos.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP02 Uso de credenciales temporales
- SEC02-BP05 Auditoría y rotación periódicas de las credenciales

Documentos relacionados:

- Introducción a AWS Secrets Manager
- Federación y proveedores de identidades
- Amazon CodeGuru presenta el detector de secretos
- Cómo AWS Secrets Manager utiliza AWS Key Management Service
- Cifrado y descifrado de secretos en Secrets Manager
- Entradas del blog de Secrets Manager
- Amazon RDS presenta la integración con AWS Secrets Manager

Videos relacionados:

- Best Practices for Managing, Retrieving, and Rotating Secrets at Scale
- Find Hard-Coded Secrets Using Amazon CodeGuru Secrets Detector
- Securing Secrets for Hybrid Workloads Using AWS Secrets Manager

Talleres relacionados:

- Store, retrieve, and manage sensitive credentials in AWS Secrets Manager
- Activaciones híbridas de AWS Systems Manager

SEC02-BP04 Uso de un proveedor de identidades centralizado

Para las identidades de la plantilla (empleados y contratistas), recurra a un proveedor de identidades que le permita administrar las identidades desde un lugar centralizado. De este modo se facilita la administración del acceso en varias aplicaciones y sistemas, pues crea, asigna, administra, revoca y audita el acceso desde un único lugar.

Resultado deseado: tiene un proveedor de identidades centralizado en el que administra de forma centralizada los usuarios de la plantilla, las políticas de autenticación (como la exigencia de la autenticación multifactor [MFA]) y la autorización de los sistemas y las aplicaciones (como la asignación del acceso en función de la pertenencia o los atributos del grupo del usuario). Los usuarios de la plantilla inician sesión en el proveedor de identidades central y se federan (inicio de sesión único) en aplicaciones internas y externas, lo que elimina la necesidad de que los usuarios recuerden varias credenciales. El proveedor de identidades está integrado en sus sistemas de recursos humanos (RR. HH.) para que los cambios de personal se sincronicen automáticamente con su proveedor de identidades. Por ejemplo, si alguien abandona la organización, puede revocar automáticamente el acceso a las aplicaciones y sistemas federados (incluido AWS). Ha habilitado el registro de auditoría detallado en su proveedor de identidades y supervisa estos registros para detectar comportamientos inusuales de los usuarios.

Patrones comunes de uso no recomendados:

- No utiliza la federación ni el inicio de sesión único. Los usuarios de la plantilla crean cuentas de usuario y credenciales independientes en diversas aplicaciones y sistemas.
- No ha automatizado el ciclo de vida de las identidades de los usuarios de la plantilla, por ejemplo, no ha integrado su proveedor de identidades en sus sistemas de recursos humanos. Cuando un usuario abandona la organización o cambia de rol, se sigue un proceso manual para eliminar o actualizar sus registros en varias aplicaciones y sistemas.

Beneficios de establecer esta práctica recomendada: al usar un proveedor de identidades centralizado, hay un único lugar en el que se administran las identidades y políticas de los usuarios en plantilla, la capacidad de asignar acceso a aplicaciones a los usuarios y grupos y la capacidad de supervisar la actividad de inicio de sesión de los usuarios. Al integrarse en sus sistemas de recursos humanos (RR. HH.), cuando un usuario cambia de rol, estos cambios se sincronizan con el

proveedor de identidades, y las aplicaciones y permisos asignados se actualizan automáticamente. Cuando un usuario abandona la organización, su identidad se inhabilita automáticamente en el proveedor de identidades, lo que revoca su acceso a las aplicaciones y sistemas federados.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Orientaciones para los usuarios de la plantilla que acceden a AWS Es posible que los usuarios de la plantilla, como empleados y contratistas de su organización, tengan que acceder a AWS mediante la Consola de administración de AWS o la AWS Command Line Interface (AWS CLI) para trabajar. Para conceder acceso a AWS, puede federar a los usuarios en plantilla desde su proveedor de identidades centralizado en AWS en dos niveles: federación directa a cada Cuenta de AWS o federación de varias cuentas en su organización de AWS.

Para federar a los usuarios en plantilla directamente con cada Cuenta de AWS, puede utilizar un proveedor de identidades centralizado para federar a AWS Identity and Access Management en esa cuenta. La flexibilidad de IAM le permite habilitar un proveedor de identidades SAML 2.0 u Open ID Connect (OIDC) por separado para cada Cuenta de AWS y utilizar atributos de usuario federado para el control de acceso. Para iniciar sesión en el proveedor de identidades, los usuarios en plantilla utilizarán su navegador web y proporcionarán sus credenciales (como contraseñas y códigos de token de MFA). El proveedor de identidades envía una aserción SAML a su navegador que se envía a la URL de inicio de sesión de la Consola de administración de AWS para permitir que el usuario haga un inicio de sesión único en la Consola de administración de AWS al asumir un rol de IAM. Los usuarios también pueden obtener credenciales temporales de la API de AWS para usarlas en la AWS CLI o los SDK de AWS desde AWS STS si asumen el rol de IAM mediante una aserción de SAML del proveedor de identidades.

Para federar a los usuarios en plantilla con varias cuentas en su organización de AWS, puede utilizar IAM Identity Center de AWS para administrar de forma centralizada el acceso de los usuarios en plantilla a las Cuentas de AWS y a las aplicaciones. Puede habilitar Identity Center para su organización y configurar el origen de las identidades. IAM Identity Center proporciona un directorio predeterminado de orígenes de identidad que puede utilizar para administrar usuarios y grupos. Como alternativa, puede elegir un origen de identidades externo mediante la conexión con el proveedor de identidades externo con SAML 2.0 y el aprovisionamiento automático de usuarios y grupos con SCIM, o mediante la conexión con el directorio de Microsoft AD a través de Directory Service. Una vez configurado un origen de identidades, puede asignar acceso a Cuentas de AWS a usuarios y grupos al definir políticas de privilegios mínimos en los conjuntos de permisos. Los

usuarios en plantilla pueden autenticarse a través de su proveedor de identidades central para iniciar sesión en el portal de acceso de AWS e iniciar sesión de forma única en Cuentas de AWS y en las aplicaciones en la nube que tengan asignadas. Los usuarios pueden configurar la AWS CLI v2 para autenticarse en IAM Identity Center y obtener credenciales para ejecutar comandos de la AWS CLI. Identity Center también permite el acceso mediante inicio de sesión único a aplicaciones de AWS, como Amazon SageMaker AI Studio y portales de AWS IoT Sitewise Monitor.

Tras seguir las instrucciones anteriores, los usuarios en plantilla ya no tendrán que usar grupos y usuarios de IAM para las operaciones normales al administrar las cargas de trabajo en AWS. En cambio, los usuarios y grupos se administran fuera de AWS, y los usuarios pueden acceder a los recursos de AWS como una identidad federada. Las identidades federadas utilizan los grupos definidos por el proveedor de identidades centralizado. Debe identificar y eliminar los grupos de IAM, los usuarios de IAM y las credenciales de usuario de larga duración (contraseñas y claves de acceso) que ya no sean necesarios en sus Cuentas de AWS. Puede encontrar las credenciales sin usar mediante los informes de credenciales de IAM, eliminar los usuarios de IAM correspondientes y eliminar los grupos de IAM. En su organización, puede aplicar una política de control de servicio (SCP) que ayude a evitar la creación de nuevos usuarios y grupos de IAM, y exigir que el acceso a AWS se haga mediante identidades federadas.

> **Note**
>
> Es su responsabilidad gestionar la rotación de los tokens de acceso de SCIM, tal como se describe en la documentación de Aprovisionamiento automático. Además, es responsable de rotar los certificados que respaldan su federación de identidades.

Orientaciones para los usuarios de sus aplicaciones Puede administrar las identidades de los usuarios de sus aplicaciones, como una aplicación móvil, mediante Amazon Cognito como proveedor de identidades centralizado. Amazon Cognito permite la autenticación, autorización y administración de usuarios para sus aplicaciones móviles y web. Amazon Cognito proporciona un almacén de identidades que se escala a millones de usuarios, admite la federación de identidades sociales y empresariales, y ofrece características de seguridad avanzadas para ayudar a proteger a sus usuarios y a su empresa. Puede integrar su aplicación web o móvil personalizada en Amazon Cognito para agregar autenticación de usuarios y control de acceso a sus aplicaciones en cuestión de minutos. Basado en estándares de identidad abiertos, como SAML y Open ID Connect (OIDC), Amazon Cognito es compatible con varias normativas de cumplimiento y se integra en los recursos de desarrollo de frontend y backend.

Pasos para la implementación

Pasos para los usuarios em plantilla que acceden a AWS

- Federe a los usuarios en plantilla en AWS mediante un proveedor de identidades centralizado con uno de los siguientes enfoques:
  - Utilice IAM Identity Center para habilitar el inicio de sesión único en varias Cuentas de AWS de su organización de AWS mediante la federación con su proveedor de identidades.
  - Utilice IAM para conectar su proveedor de identidades directamente a cada Cuenta de AWS, lo que permite un acceso federado y detallado.
- Identifique y elimine los grupos y usuarios de IAM que se sustituyan por identidades federadas.

Pasos para los usuarios de sus aplicaciones

- Utilice Amazon Cognito como proveedor de identidades centralizado para sus aplicaciones.
- Integre sus aplicaciones personalizadas en Amazon Cognito mediante OpenID Connect y OAuth. Puede desarrollar sus aplicaciones personalizadas mediante las bibliotecas de Amplify, que proporcionan interfaces sencillas para integrarse con una variedad de servicios de AWS, como Amazon Cognito, para la autenticación.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP06 Uso de grupos y atributos de usuarios
- SEC03-BP02 Concesión de acceso con privilegios mínimos
- SEC03-BP06 Administración del acceso en función del ciclo de vida

Documentos relacionados:

- Federación de identidades en AWS
- Security best practices in IAM (Prácticas recomendadas de seguridad en IAM)
- Prácticas recomendadas de AWS Identity and Access Management
- Getting started with IAM Identity Center delegated administration
- How to use customer managed policies in IAM Identity Center for advanced use cases
- AWS CLI v2: proveedor de credenciales de IAM Identity Center

Videos relacionados:

- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2018: Mastering Identity at Every Layer of the Cake

Ejemplos relacionados:

- Workshop: Using AWS IAM Identity Center to achieve strong identity management

Herramientas relacionadas:

- Socios con competencia en seguridad de AWS: Identity and Access Management
- saml2aws

SEC02-BP05 Auditoría y rotación periódicas de las credenciales

Audite y rote las credenciales periódicamente para limitar el tiempo que pueden utilizarse para acceder a los recursos. Las credenciales de larga duración entrañan muchos riesgos, pero estos riesgos pueden reducirse con una rotación frecuente.

Resultado deseado: implemente la rotación de credenciales para ayudar a reducir los riesgos asociados al uso de credenciales a largo plazo. Audita regularmente y corrige la no conformidad con las políticas de rotación de credenciales.

Patrones comunes de uso no recomendados:

- No auditar el uso de credenciales.
- Utilizar credenciales de larga duración de forma innecesaria.
- Utilizar credenciales de larga duración y no rotarlas regularmente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Cuando no pueda confiar en credenciales temporales y necesite credenciales de larga duración, audítelas para verificar que los controles definidos, por ejemplo, la autenticación multifactor (MFA), se aplican, se rotan periódicamente y tienen el nivel de acceso adecuado.

Es necesario llevar a cabo una validación periódica, preferiblemente mediante una herramienta automatizada, para verificar que se están aplicando los controles correctos. En el caso de las identidades humanas, debe exigir a los usuarios que cambien sus contraseñas periódicamente y retirar las claves de acceso para sustituirlas por credenciales temporales. Al pasar de usuarios de AWS Identity and Access Management (IAM) a identidades centralizadas, puede generar un informe de credenciales para auditar a los usuarios.

También recomendamos que aplique y supervise una configuración de MFA en su proveedor de identidades. Puede configurar Reglas de AWS Config o utilizar estándares de seguridad de AWS Security Hub CSPM para supervisar si los usuarios han configurado la MFA. Si lo considera oportuno, puede utilizar IAM Roles Anywhere para proporcionar credenciales temporales para identidades de máquinas. En situaciones en las que no sea posible utilizar roles de IAM y credenciales temporales, es necesario llevar a cabo auditorías frecuentes y rotar las claves de acceso.

Pasos para la implementación

- Auditoría periódica de las credenciales: auditar las identidades que están configuradas en el proveedor de identidades e IAM le permite asegurarse de que las únicas identidades que pueden acceder a su carga de trabajo son aquellas que estén autorizadas. Dichas identidades pueden incluir, entre otras, usuarios de IAM, usuarios de AWS IAM Identity Center, usuarios de Active Directory o usuarios de un proveedor de identidades ascendente diferente. Por ejemplo, elimine las personas que abandonen la organización y los roles entre cuentas que ya no sean necesarios. Implante un proceso para auditar periódicamente los permisos a los servicios a los que accede una entidad de IAM. Esto le ayudará a identificar las políticas que debe modificar para eliminar los permisos que no se utilizan. Utilice los informes de credenciales e AWS Identity and Access Management Access Analyzer para auditar las credenciales y los permisos de IAM. Puede usar Amazon CloudWatch para configurar alarmas para llamadas específicas a las API dentro de su entorno de AWS. Además, Amazon GuardDuty puede avisarle de una actividad inesperada, que podría indicar un acceso demasiado permisivo o no intencionado a las credenciales de IAM.
- Rotación periódica de las credenciales: si no puede utilizar credenciales temporales, rote periódicamente las claves de acceso a IAM de larga duración (como máximo cada 90 días). Si se revela una clave de acceso de forma involuntaria sin su conocimiento, esto limita el tiempo durante el que se pueden utilizar las credenciales para acceder a los recursos. Si desea obtener información sobre la rotación de las claves de acceso de los usuarios de IAM, consulte Rotación de las claves de acceso.

- Revisión de los permisos de IAM: para mejorar la seguridad de su Cuenta de AWS, debe revisar y supervisar periódicamente cada una de sus políticas de IAM. Verifique que las políticas sigan el principio del privilegio mínimo.
- Si lo estima oportuno, puede automatizar la creación y las actualizaciones de los recursos de IAM: IAM Identity Center automatiza muchas tareas de IAM, como la administración de roles y políticas. Como alternativa, se puede utilizar AWS CloudFormation para automatizar la implementación de los recursos de IAM, incluidos los roles y las políticas, para reducir la posibilidad de que se produzcan errores humanos, ya que las plantillas se pueden verificar y controlar por versiones.
- Uso de IAM Roles Anywhere para sustituir los usuarios de IAM por identidades de máquinas: IAM Roles Anywhere le permite utilizar roles en áreas que antes no podía utilizar, como en servidores locales. IAM Roles Anywhere utiliza un certificado X.509 de confianza para autenticarse en AWS y recibir credenciales temporales. El uso de IAM Roles Anywhere evita la necesidad de rotar estas credenciales, ya que las credenciales de larga duración ya no se almacenan en el entorno en las instalaciones. Tenga en cuenta que deberá supervisar y rotar el certificado X.509 a medida que se acerque su fecha de vencimiento.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP02 Uso de credenciales temporales
- SEC02-BP03 Almacenamiento y uso seguros de secretos

Documentos relacionados:

- Introducción a AWS Secrets Manager
- Prácticas recomendadas de IAM
- Federación y proveedores de identidades
- Soluciones de socios de seguridad: acceso y control de acceso
- Credenciales de seguridad temporales
- Generación de informes de credenciales para su Cuenta de AWS

Videos relacionados:

- Best Practices for Managing, Retrieving, and Rotating Secrets at Scale

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

### SEC02-BP06 Uso de grupos y atributos de usuarios

Definir los permisos según los grupos y los atributos de usuarios ayuda a reducir la cantidad y la complejidad de las políticas, lo que facilita la aplicación del principio de privilegio mínimo. Puede emplear los grupos de usuarios para administrar los permisos de muchas personas en un solo lugar según la función que desempeñen en su organización. Los atributos, como el departamento, el proyecto o la ubicación, pueden proporcionar un nivel adicional en el ámbito de los permisos si hay personas que hacen una función similar, pero para diferentes subconjuntos de recursos.

Resultado deseado: puede aplicar cambios en los permisos según la función a todos los usuarios que desempeñen esa función. La pertenencia a grupos y los atributos determinan los permisos de los usuarios, lo que reduce la necesidad de administrar los permisos para cada usuario individual. Los grupos y atributos que defina en su proveedor de identidades (IdP) se propagan automáticamente a sus entornos de AWS.

Patrones comunes de uso no recomendados:

- Administrar los permisos de usuarios individuales y duplicarlos para muchos usuarios.
- Definir grupos en un nivel demasiado alto y conceder permisos demasiado amplios.
- Definir grupos en un nivel demasiado detallado, lo que crea duplicación y confusión en torno a la pertenencia a dichos grupos.
- Usar grupos con permisos duplicados en diversos subconjuntos de recursos cuando, en su lugar, se pueden usar atributos.
- No administrar grupos, atributos y pertenencias a grupos con un proveedor de identidades estandarizado integrado con sus entornos de AWS.
- Usar encadenamiento de roles al utilizar las sesiones del AWS IAM Identity Center.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

#### Guía para la implementación

Los permisos de AWS se definen en documentos denominados políticas que están asociados a una entidad principal, como un usuario, grupo, rol o recurso. Puede escalar la gestión de permisos organizando las asignaciones de permisos (grupo, permisos o cuenta) dependiendo de la función del

trabajo, la carga de trabajo y el entorno de SDLC. En el caso de su plantilla, esto le permite definir grupos según la función que desempeñen los usuarios en su organización, en lugar de en virtud de los recursos a los que se accede. Por ejemplo, un grupo de desarrolladores de aplicaciones web puede tener una política adjunta para configurar servicios como Amazon CloudFront dentro de una cuenta de desarrollo. Es posible que un grupo de `AutomationDeveloper` tenga algunos permisos que coincidan con los del grupo de `WebAppDeveloper`. Estos permisos pueden recopilarse en una política independiente y asociarse a ambos grupos, en lugar de permitir que los usuarios de ambas funciones pertenezcan a un grupo de `CloudFrontAccess`.

Además de los grupos, puede utilizar atributos para ampliar el acceso al ámbito. Por ejemplo, puede tener un atributo Proyecto para los usuarios del grupo de `WebAppDeveloper` para limitar el acceso a los recursos específicos de su proyecto. El uso de esta técnica elimina la necesidad de tener diferentes grupos para los desarrolladores de aplicaciones que trabajen en diferentes proyectos si, por lo demás, sus permisos son los mismos. La forma de hacer referencia a los atributos en las políticas de permisos se basa en su origen, ya sea que estén definidos como parte de su protocolo de federación (como SAML, OIDC o SCIM), como aserciones SAML personalizadas o que se hayan configurado en IAM Identity Center.

Pasos para la implementación

1. Establezca dónde definirá los grupos y los atributos:
    a. Al seguir las instrucciones que se indican en SEC02-BP04 Uso de un proveedor de identidades centralizado, puede determinar si tiene que definir grupos y atributos en el proveedor de identidades, en IAM Identity Center o usar grupos de usuarios de IAM en una cuenta específica.
2. Defina grupos:
    a. Determine los grupos según la función y el ámbito del acceso requerido. Puede utilizar una estructura jerárquica o convenciones de nomenclatura para organizar los grupos de forma eficaz.
    b. Si especifica la definición en IAM Identity Center, cree grupos y asocie el nivel de acceso deseado mediante conjuntos de permisos.
    c. Si especifica la definición con un proveedor de identidades externo, determine si el proveedor admite el protocolo SCIM y plantéese la posibilidad de habilitar el aprovisionamiento automático en IAM Identity Center. Esta capacidad sincroniza la creación, la pertenencia y la eliminación de grupos entre su proveedor e IAM Identity Center.
3. Defina los atributos:

a. Si utiliza un proveedor de identidades externo, los protocolos SCIM y SAML 2.0 proporcionan ciertos atributos de forma predeterminada. Los atributos adicionales se pueden definir y transferir mediante aserciones de SAML que utilizan el nombre del atributo `https://aws.amazon.com/SAML/Attributes/PrincipalTag`. Consulte la documentación de su proveedor de identidades a fin de conocer las instrucciones necesarias para definir y configurar atributos personalizados.

b. Si define roles en IAM Identity Center, active la característica de control de acceso basado en atributos (ABAC) y defina los atributos como desee. Debería usar atributos que vayan en consonancia con la estructura o la estrategia de etiquetado de recursos de su organización.

Si necesita encadenar los roles de IAM a partir de los roles de IAM asumidos a través del centro de identidad de IAM, los valores como `source-identity` y `principal-tags` no se propagarán. Para obtener información más detallada, consulte Habilitación y configuración de atributos para el control de acceso.

1. Los permisos de ámbito se basan en grupos y atributos:

   a. Plantéese la posibilidad de incluir condiciones en las políticas de permisos que comparen los atributos de su entidad principal con los atributos de los recursos a los que se accede. Por ejemplo, puede definir una condición para permitir el acceso a un recurso solo si el valor de una clave de condición `PrincipalTag` coincide con el valor de una clave `ResourceTag` del mismo nombre.

   b. Al definir las políticas de la ABAC, siga las instrucciones de las prácticas recomendadas y ejemplos de autorización de la ABAC.

   c. Revise y actualice periódicamente su estructura de grupos y atributos a medida que evolucionen las necesidades de su organización para garantizar una gestión óptima de los permisos.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP04 Uso de un proveedor de identidades centralizado
- SEC03-BP02 Concesión de acceso con privilegios mínimos
- COST02-BP04 Implementación de grupos y roles

Documentos relacionados:

- Prácticas recomendadas de IAM
- Manage Identities in IAM Identity Center
- What Is ABAC for AWS?
- ABAC In IAM Identity Center
- Ejemplos de políticas ABAC

Videos relacionados:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

## SEC 3. ¿Cómo se gestionan los permisos de las personas y las máquinas?

Administre permisos para controlar el acceso a identidades humanas y de máquinas que requieran acceso a AWS y sus cargas de trabajo. Los permisos le permiten controlar a qué puede acceder cada usuario y en qué condiciones. Al establecer permisos para identidades específicas humanas y de máquinas, concede acceso a acciones de servicio específicas en determinados recursos. Asimismo, puede especificar condiciones que deben cumplirse para que se conceda el acceso.

Prácticas recomendadas

- SEC03-BP01 Definición de los requisitos de acceso
- SEC03-BP02 Concesión de acceso con privilegios mínimos
- SEC03-BP03 Establecimiento de un proceso de acceso de emergencia
- SEC03-BP04 Reducción continua de los permisos
- SEC03-BP05 Definición de las barreras de protección de los permisos para una organización
- SEC03-BP06 Administración del acceso en función del ciclo de vida
- SEC03-BP07 Análisis del acceso público y entre cuentas
- SEC03-BP08 Uso compartido de recursos de forma segura en su organización
- SEC03-BP09 Uso compartido seguro de recursos con terceros

SEC03-BP01 Definición de los requisitos de acceso

Los administradores, los usuarios finales u otros componentes deben acceder a cada componente o recurso de la carga de trabajo. Establezca una definición clara de quién o qué debe tener acceso a cada componente y elija el tipo de identidad y el método de autenticación y autorización adecuados.

Patrones comunes de uso no recomendados:

- Codificar de forma rígida o almacenar secretos en la aplicación.
- Conceder permisos personalizados para cada usuario.
- Utilizar credenciales de larga duración.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los administradores, los usuarios finales u otros componentes deben acceder a cada componente o recurso de la carga de trabajo. Establezca una definición clara de quién o qué debe tener acceso a cada componente y elija el tipo de identidad y el método de autenticación y autorización adecuados.

El acceso periódico a las Cuentas de AWS dentro de una organización debe proporcionarse mediante acceso federado o un proveedor de identidades centralizado. También debe centralizar la administración de identidades y asegurarse de que existe una práctica establecida para integrar el acceso de AWS al ciclo de vida de los empleados. Por ejemplo, cuando un empleado cambia a un cargo con un nivel de acceso distinto, su pertenencia al grupo también debe cambiar para reflejar los nuevos requisitos de acceso.

Al definir los requisitos de acceso para las identidades que no son humanas, determine qué aplicaciones y componentes necesitan acceso y cómo se conceden los permisos. El enfoque recomendado es utilizar roles de IAM creados con el modelo de acceso de privilegio mínimo. AWS Las políticas administradas proporcionan políticas de IAM predefinidas que cubren los casos de uso más comunes.

Los servicios de AWS, como AWS Secrets Manager y el Almacén de parámetros de AWS Systems Manager, pueden ayudar a desvincular los secretos de la aplicación o la carga de trabajo de forma segura. En Secrets Manager, puede establecer una rotación automática de las credenciales. Puede utilizar Systems Manager para hacer referencia a los parámetros en los scripts, comandos, documentos SSM, configuración y flujos de trabajo de automatización con el nombre único que especificó al crear el parámetro.

Puede utilizar AWS IAM Roles Anywhere para obtener credenciales de seguridad temporales en IAM para cargas de trabajo que se ejecutan fuera de AWS. Las cargas de trabajo pueden usar las mismas políticas de IAM y roles de IAM que utiliza con las aplicaciones de AWS para acceder a los recursos de AWS.

Siempre que sea posible, se deben preferir las credenciales temporales a corto plazo en lugar de las credenciales estáticas a largo plazo. En aquellas situaciones en las que necesite usuarios con acceso programático y credenciales de larga duración, utilice la información de la última clave de acceso utilizada para rotar y eliminar las claves de acceso.

Los usuarios necesitan acceso programático si desean interactuar con AWS fuera de la Consola de administración de AWS. La forma de conceder el acceso programático depende del tipo de usuario que acceda a AWS.

Para conceder acceso programático a los usuarios, seleccione una de las siguientes opciones.

| ¿Qué usuario necesita acceso programático? | Para | Mediante |
| --- | --- | --- |
| IAM | (Recomendado) Use credenciales de consola como credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS. | Siga las instrucciones de la interfaz que desea utilizar:<br>• Para la AWS CLI, consulte Iniciar sesión para el desarrollo local de AWS en la Guía del usuario de AWS Command Line Interface.<br>• Para los AWS SDK, consulte Iniciar sesión para el desarrollo local de AWS en la Guía de referencia de AWS SDK y herramientas. |
| Identidad del personal<br><br>(Usuarios administrados en el IAM Identity Center) | Utiliza credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS. | Siga las instrucciones de la interfaz que desea utilizar:<br>• Para utilizar la AWS CLI, consulte Configuring the AWS CLI to usa AWS |

| ¿Qué usuario necesita acceso programático? | Para | Mediante |
| --- | --- | --- |
| | | IAM Identity Center en la Guía del usuario de AWS Command Line Interface.<br>• Para usar AWS SDK, las herramientas y las API de AWS, consulta IAM Identity Center authentication en la Guía de referencia del SDK y las herramientas de AWS. |
| IAM | Utiliza credenciales temporales para firmar las solicitudes programáticas a la AWS CLI, los AWS SDK y las API de AWS. | Siguiendo las instrucciones de Uso de credenciales temporales con recursos de AWS de la Guía del usuario de IAM. |

| ¿Qué usuario necesita acceso programático? | Para | Mediante |
| --- | --- | --- |
| IAM | (No recomendado) Utilizar credenciales a largo plazo para firmar las solicitud es programáticas a la AWS CLI, los AWS SDK o las API de AWS. | Siga las instrucciones de la interfaz que desea utilizar:<br>• Para la AWS CLI, consulte Autenticación mediante credenciales de usuario de IAM en la Guía del usuario de AWS Command Line Interface.<br>• Para ver los AWS SDK y las herramientas, consulte Autenticar mediante credenciales a largo plazo en la Guía de referencia de AWS SDK y herramientas.<br>• Para las API de AWS, consulta Administración de claves de acceso para usuarios de IAM en la Guía del usuario de IAM. |

Recursos

Documentos relacionados:

- Control de acceso basado en atributos (ABAC)
- AWS IAM Identity Center
- IAM Roles Anywhere
- AWS Managed policies for IAM Identity Center
- AWS IAM policy conditions
- IAM use cases
- Revisar y eliminar periódicamente usuarios, roles, permisos, políticas y credenciales no utilizados

- Uso de políticas de
- How to control access to AWS resources based on Cuenta de AWS, OU, or organization
- Identify, arrange, and manage secrets easily using enhanced search in AWS Secrets Manager

Videos relacionados:

- Become an IAM Policy Master in 60 Minutes or Less
- Separation of Duties, Least Privilege, Delegation, and CI/CD
- Streamlining identity and access management for innovation

SEC03-BP02 Concesión de acceso con privilegios mínimos

Conceda exclusivamente el acceso que las identidades necesitan para efectuar acciones concretas en recursos específicos en determinadas condiciones. Utilice atributos de grupo y de identidad para configurar dinámicamente los permisos en función de las necesidades en lugar de configurarlos para cada usuario. Por ejemplo, puede conceder acceso a un grupo de desarrolladores para que solamente puedan administrar recursos de su proyecto. De este modo, si un desarrollador abandona el proyecto, su acceso se revoca automáticamente sin cambiar las políticas de acceso subyacentes.

Resultado deseado: los usuarios solo tienen los permisos mínimos necesarios para el trabajo específico que desempeñan. Utiliza Cuentas de AWS independientes para aislar a los desarrolladores de los entornos de producción. Cuando los desarrolladores necesitan acceder a los entornos de producción para realizar tareas específicas, solo se les concede un acceso limitado y controlado durante el tiempo que se necesita para esas tareas. Su acceso a la producción se revoca inmediatamente después de completar el trabajo necesario. Realiza revisiones periódicas de los permisos y los revoca de inmediato cuando ya no son necesarios; por ejemplo, cuando un usuario cambia de rol o deja la organización. Restringe los privilegios de administrador a un grupo pequeño y de confianza para reducir la exposición al riesgo. Concede a las cuentas de máquinas o sistemas los permisos mínimos necesarios para realizar las tareas previstas.

Patrones comunes de uso no recomendados:

- Concede permisos de administrador a los usuarios de forma predeterminada.
- Utiliza la cuenta de usuario raíz para tareas cotidianas.
- Crea políticas demasiado permisivas sin un alcance adecuado.
- Revisa sus permisos con poca frecuencia, lo que provoca un aumento excesivo de los permisos.

- Aísla el entorno o gestiona los permisos basándose exclusivamente en un control de acceso basado en atributos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El principio del privilegio mínimo establece que a las identidades solo se les debe permitir llevar a cabo el conjunto más reducido de acciones necesarias para efectuar una tarea específica. De este modo, se equilibra la facilidad de uso, la eficiencia y la seguridad. Operar según este principio contribuye a limitar el acceso involuntario y a hacer el seguimiento de quién tiene acceso a determinados recursos. Los usuarios y los roles de IAM no tienen permisos de forma predeterminada. De forma predeterminada, el usuario raíz tiene acceso total y debe controlarse y supervisarse rigurosamente, y utilizarse únicamente para las tareas que requieren acceso raíz.

Las políticas de IAM se usan para conceder permisos a roles de IAM o recursos específicos. Por ejemplo, las políticas basadas en la identidad se pueden adjuntar a grupos de IAM, mientras que los buckets de S3 se pueden controlar mediante políticas basadas en recursos.

Al crear una política de IAM, puede especificar las acciones de servicio, los recursos y las condiciones que se deben cumplir para que AWS permita o deniegue el acceso. AWS es compatible con una amplia variedad de condiciones que lo ayudarán a acotar el acceso. Por ejemplo, al usar la clave de condición PrincipalOrgID, puede denegar acciones si el solicitante no forma parte de su organización de AWS.

También puede controlar las solicitudes que hagan los servicios de AWS en su nombre, como que AWS CloudFormation cree una función de AWS Lambda, mediante la clave de condición CalledVia. Puede estratificar los diferentes tipos de políticas para establecer una defensa en profundidad y limitar los permisos generales de sus usuarios. También puede restringir qué permisos se pueden conceder y en qué condiciones. Por ejemplo, puede permitir que sus equipos de carga de trabajo creen sus propias políticas de IAM para los sistemas que creen, pero solo si aplican también un límite de permisos para determinar el número máximo de permisos que el sistema puede recibir.

Pasos para la implementación

- Implementación de políticas de privilegio mínimo: asigne políticas de acceso con privilegio mínimo a los grupos y roles de IAM para reflejar el rol o la función del usuario que haya definido.

- Aísle los entornos de desarrollo y de producción separando las Cuentas de AWS: utilice Cuentas de AWS distintas para los entornos de desarrollo y los de producción y controle el acceso entre ellos mediante políticas de control de servicios, políticas de recursos y políticas de identidad.
- Uso de las API como base de las políticas: una forma de determinar los permisos necesarios consiste en revisar los registros de AWS CloudTrail. Esta revisión le permite crear permisos adaptados a las acciones que el usuario lleva a cabo en AWS. El Analizador de acceso de IAM puede generar automáticamente una política de IAM basada en la actividad de acceso. Puede utilizar el asesor de acceso de IAM en la organización o en la cuenta para hacer un seguimiento de la información a la que se accedió por última vez en relación con una política concreta.
- Puede usar políticas administradas por AWS para funciones de trabajo: cuando empiece a crear políticas de permisos detalladas, puede ser útil usar políticas administradas por AWS para funciones de trabajo comunes, como facturación, administradores de bases de datos y científicos de datos. Estas políticas pueden servir para limitar el acceso que tienen los usuarios al mismo tiempo que se determina cómo implementar las políticas de privilegio mínimo.
- Elimine los permisos innecesarios: detecte y elimine las entidades, credenciales y permisos de IAM no utilizados para cumplir con el principio de privilegios mínimos. Puede utilizar Analizador de acceso de IAM para identificar el acceso externo y el no utilizado, y la generación de políticas de Analizador de acceso de IAM puede ayudar a afinar las políticas de permisos.
- Garantía de que los usuarios tengan un acceso limitado a los entornos de producción: los usuarios solo deben tener acceso a los entornos de producción con un caso de uso válido. Después de que el usuario lleve a cabo las tareas específicas que requieren el acceso a producción, se debe revocar el acceso. La limitación del acceso a los entornos de producción previene los eventos involuntarios que afectan a la producción y reduce el ámbito de las consecuencias del acceso involuntario.
- Plantéese utilizar límites de permisos: un límite de permisos es una característica avanzada que permite utilizar una política administrada para establecer los permisos máximos que una política basada en identidades puede conceder a una entidad de IAM. Un límite de permisos para una entidad le posibilita realizar las acciones que le permitan tanto sus políticas basadas en identidad como sus límites de permisos.
- Limite el acceso mediante el control de acceso basado en atributos y las etiquetas de recursos: el control de acceso basado en atributos (ABAC) que utiliza etiquetas de recursos se puede utilizar para refinar los permisos cuando es compatible. Puede utilizar un modelo ABAC que compare las etiquetas principales con las etiquetas de recursos para refinar el acceso en función de las dimensiones personalizadas que defina. Este método puede simplificar y reducir la cantidad de políticas de permisos en su organización.

  - Se recomienda utilizar ABAC únicamente para el control de accesos cuando tanto las entidades principales como los recursos sean propiedad de su organización de AWS. Las partes externas pueden usar los mismos nombres y valores de etiqueta que su organización para sus propias entidades principales y recursos. Si concede acceso a las entidades principales o los recursos de terceros basándose únicamente en estos pares de nombre-valor, podría conceder permisos que no desea conceder.
- Utilizar políticas de control de servicios para AWS Organizations: las políticas de control de servicios controlan de forma centralizada el máximo de permisos disponibles para las cuentas de los miembros de su organización. Es importante destacar que puede utilizar las políticas de control de servicios para restringir los permisos del usuario raíz en las cuentas de los miembros. Considere también la posibilidad de utilizar AWS Control Tower, que proporciona controles prescriptivos administrados que enriquecen AWS Organizations. También puede definir sus propios controles en Control Tower.
- Establecimiento de una política de ciclo de vida de los usuarios para su organización: las políticas de ciclo de vida de los usuarios definen las tareas que deben llevarse a cabo cuando los usuarios se incorporan a AWS, cuando cambian de rol o ámbito de trabajo, o cuando ya no necesitan acceder a AWS. Realice revisiones de permisos en cada paso del ciclo de vida de un usuario para verificar si son restrictivos de forma correcta y para evitar la acumulación de permisos.
- Establecimiento de una programación periódica para revisar los permisos y eliminar los que no sean necesarios: debe revisar periódicamente el acceso de los usuarios para comprobar que no tengan un acceso demasiado permisivo. AWS Config y el Analizador de acceso de IAM pueden ayudarlo durante las auditorías de permisos.
- Establecimiento de una matriz de roles de trabajo: una matriz de roles de trabajo permite visualizar las distintas funciones y niveles de acceso necesarios en su entorno de AWS. Con una matriz de roles de trabajo, puede definir y separar los permisos según las responsabilidades de usuario en su organización. Utilice grupos en lugar de aplicar los permisos directamente a los usuarios o roles individuales.

Recursos

Documentos relacionados:

- Aplicar permisos de privilegios mínimos
- Límites de permisos para las entidades de IAM
- Techniques for writing least privilege IAM policies

- IAM Access Analyzer makes it easier to implement least privilege permissions by generating IAM policies based on access activity
- Delegate permission management to developers by using IAM permissions boundaries
- Perfeccionar los permisos con la información sobre los últimos accesos
- Tipos de políticas de IAM y cuándo usarlas
- Probar las políticas de IAM con el simulador de política de IAM
- Guardrails in AWS Control Tower
- Zero Trust architectures: An AWS perspective
- How to implement the principle of least privilege with CloudFormation StackSets
- Control de acceso basado en atributos (ABAC)
- Reducción del ámbito de las políticas mediante la consulta de la actividad de los usuarios
- Visualización del acceso a los roles
- Utilizar el etiquetado para organizar el entorno e impulsar la responsabilidad
- AWS Tagging Strategies
- Etiquetado de recursos de AWS

Videos relacionados:

- Next-generation permissions management
- Zero Trust: An AWS perspective

SEC03-BP03 Establecimiento de un proceso de acceso de emergencia

Cree un proceso que permita el acceso de emergencia a sus cargas de trabajo en el caso improbable de que se produzca un problema con su proveedor de identidades centralizado.

Debe diseñar procesos para diferentes modos de error que puedan provocar un evento de emergencia. Por ejemplo, en circunstancias normales, los usuarios de la plantilla se federan en la nube mediante un proveedor de identidades centralizado (SEC02-BP04) para administrar sus cargas de trabajo. Sin embargo, si su proveedor de identidades centralizado no responde o se modifica la configuración de la federación en la nube, es posible que los usuarios de la plantilla no puedan federarse en esta. Un proceso de acceso de emergencia permite a los administradores autorizados acceder a los recursos de la nube a través de medios alternativos (como una forma alternativa de

federación o acceso directo de los usuarios) para solucionar problemas con la configuración de la federación o las cargas de trabajo. El proceso de acceso de emergencia se utiliza hasta que se restablezca el mecanismo de federación normal.

Resultado deseado:

- Ha definido y documentado los modos de error que se consideran una emergencia: tenga en cuenta sus circunstancias normales y los sistemas de los que dependen los usuarios para administrar sus cargas de trabajo. Considere cómo cada una de estas dependencias puede no funcionar y provocar una situación de emergencia. Es posible que las preguntas y las prácticas recomendadas del pilar de fiabilidad resulten útiles para identificar los modos de error y diseñar sistemas más resilientes para minimizar la probabilidad de errores.
- Ha documentado los pasos que se deben seguir para confirmar que el error se trata de un caso de emergencia. Por ejemplo, puede solicitar a sus administradores de identidades que comprueben el estado de sus proveedores de identidades principales y en espera y, si ninguno estuviera disponible, declarar un evento de emergencia por error en el proveedor de identidades.
- Ha definido un proceso de acceso de emergencia concreto para cada tipo de modo de emergencia o error. La especificidad puede reducir la tentación de los usuarios de abusar de un proceso general para todo tipo de emergencias. Los procesos de acceso de emergencia describen las circunstancias en las que se debe utilizar cada proceso y, por otra parte, las situaciones en las que no se debe utilizar el proceso y señala los procesos alternativos que podrían aplicarse.
- Sus procesos están bien documentados con instrucciones detalladas y manuales de estrategias que se pueden seguir de forma rápida y eficiente. Recuerde que un evento de emergencia puede resultar estresante para sus usuarios, ya que pueden estar sometidos a una fuerte presión de plazos, por lo que debe diseñar su proceso de la manera más sencilla posible.

Patrones comunes de uso no recomendados:

- No tiene procesos de acceso de emergencia bien documentados y probados. Cuando los usuarios no están preparados para emergencias, siguen procesos improvisados cuando estas se producen.
- Tener procesos de acceso de emergencia que dependan de los mismos sistemas (como un proveedor de identidades centralizado) que sus mecanismos de acceso normales. Esto significa que el error de un sistema de este tipo podría afectar tanto a sus mecanismos de acceso normales como a los de emergencia y, por lo tanto, repercutir en la capacidad para recuperarse del error.
- Se utilizan los procesos de acceso de emergencia en situaciones que no son de emergencia. Por ejemplo, los usuarios suelen hacer un uso inapropiado de los procesos de acceso de

emergencia, ya que les resulta más fácil hacer cambios directamente que enviarlos a través de una canalización.

- Tener procesos de acceso de emergencia que no generan registros suficientes para auditar los procesos o no supervisar los registros para alertar de un posible uso indebido de los procesos.

Beneficios de establecer esta práctica recomendada:

- Contar con procesos de acceso de emergencia bien documentados y probados puede reducir el tiempo que tardan los usuarios en responder y resolver un evento de emergencia. Esto puede reducir el tiempo de inactividad y aumentar la disponibilidad de los servicios que presta a sus clientes.
- Puede hacer un seguimiento de cada solicitud de acceso de emergencia y detectar intentos no autorizados de uso indebido de los procesos para eventos que no sean de emergencia y alertar sobre estos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Esta sección proporciona guías para crear procesos de acceso de emergencia para varios modos de error relacionados con las cargas de trabajo implementadas en AWS, comenzando con una guía común que se aplica a todos los modos de error y siguiendo con una guía específica basada en el tipo de modo de error.

Guía común para todos los modos de error

Tenga en cuenta lo siguiente al diseñar un proceso de acceso de emergencia para un modo de error:

- Documente las condiciones previas y los supuestos del proceso, es decir, cuándo el proceso debe o no debe aplicarse. Esto ayuda a detallar el modo de error y a documentar los supuestos, como el estado de otros sistemas relacionados. Por ejemplo, el proceso del modo de error 2 da por sentado que el proveedor de identidades está disponible, pero que la configuración activada en AWS se ha modificado o ha caducado.
- Cree de antemano los recursos necesarios para el proceso de acceso de emergencia (SEC10-BP05). Por ejemplo, cree de antemano el acceso de emergencia a la Cuenta de AWS con roles y usuarios de IAM, y los roles de IAM entre cuentas en todas las cuentas de la carga de trabajo. Esto asegurará que estos recursos estén listos y disponibles cuando ocurra una emergencia. Al crear previamente los recursos, no dependerá de las API del plano de control de AWS (que se utilizan

para crear y modificar recursos de AWS) que pueden no estar disponibles en caso de emergencia. Además, al crear previamente los recursos de IAM, no es necesario tener en cuenta los posibles retrasos debidos a una posible coherencia.

- Incluya los procesos de acceso de emergencia como parte de sus planes de administración de incidentes (SEC10-BP02). Documente cómo se hace el seguimiento de los eventos de emergencia y cómo se comunican a otros miembros de su organización, como los equipos de compañeros o la dirección y, cuando corresponda, externamente a sus clientes y socios comerciales.
- Defina el proceso de solicitud de acceso de emergencia en su sistema de flujo de trabajo de solicitudes de servicio existente, si dispone de uno. Por lo general, estos sistemas de flujo de trabajo le permiten crear formularios de entrada para recopilar información sobre la solicitud, hacer un seguimiento de la solicitud en cada etapa del flujo de trabajo y agregar pasos de aprobación automatizados y manuales. Relacione cada solicitud con el correspondiente evento de emergencia registrado en su sistema de administración de incidentes. Disponer de un sistema uniforme para los accesos de emergencia le permite hacer un seguimiento de esas solicitudes en un solo sistema, analizar las tendencias de uso y mejorar sus procesos.
- Compruebe que solo los usuarios autorizados puedan iniciar los procesos de acceso de emergencia y que estos procesos requieran la aprobación de los compañeros del usuario o de la dirección, según corresponda. El proceso de aprobación debe funcionar de manera eficaz, tanto dentro como fuera del horario laboral. Defina cómo admiten las solicitudes de aprobación aprobadores secundarios si los principales no están disponibles y cómo se escalan en la cadena de administración hasta la aprobación.
- Implemente mecanismos sólidos de registro, monitoreo y alerta para el proceso y los mecanismos de acceso de emergencia. Genere registros y eventos de auditoría detallados para los intentos correctos e infructuosos de obtener acceso de emergencia. Correlacione la actividad con los eventos de emergencia en curso de su sistema de administración de incidentes e inicie alertas cuando se produzcan acciones fuera de los periodos de tiempo esperados o cuando se use la cuenta de acceso de emergencia durante las operaciones normales. Solo se debe acceder a la cuenta de acceso de emergencia durante las emergencias, ya que los procedimientos acceso excepcional pueden considerarse una puerta trasera. Intégrela con su herramienta de gestión de eventos e información de seguridad (SIEM) o AWS Security Hub CSPM para informar y auditar todas las actividades durante el periodo de acceso de emergencia. Al volver al funcionamiento normal, cambie automáticamente las credenciales de acceso de emergencia y avise a los equipos pertinentes.
- Pruebe los procesos de acceso de emergencia de manera periódica para verificar que los pasos estén claros y garantizar el nivel de acceso correcto de manera rápida y eficiente. Sus procesos de

acceso de emergencia deben probarse como parte de las simulaciones de respuesta a incidentes (SEC10-BP07) y las pruebas de recuperación de desastres (REL13-BP03).

Modo de error 1: el proveedor de identidades utilizado para federarse en AWS no está disponible

Tal como se describe en SEC02-BP04 Uso de un proveedor de identidades centralizado, recomendamos confiar en un proveedor de identidades centralizado para federar a los usuarios de su plantilla y concederles acceso a las Cuentas de AWS. La federación se puede configurar a varias Cuentas de AWS de su organización de AWS con IAM Identity Center, o bien puede configurar la federación a Cuentas de AWS individuales mediante IAM. En ambos casos, los usuarios de la plantilla se autentican con su proveedor de identidades centralizado antes de que se les redirija a un punto de conexión de inicio de sesión de AWS para el inicio de sesión único.

En el caso poco probable de que su proveedor de identidades centralizado no esté disponible, los usuarios de la plantilla no podrán federarse en las Cuentas de AWS ni administrar sus cargas de trabajo. En este caso de emergencia, puede proporcionar un proceso de acceso de emergencia para que un pequeño grupo de administradores acceda a las Cuentas de AWS con el fin de llevar a cabo tareas cruciales que no puedan esperar a que sus proveedores de identidades centralizados vuelvan a estar disponibles. Por ejemplo, su proveedor de identidades no estará disponible durante 4 horas y, durante ese periodo, necesita modificar los límites superiores de un grupo de Amazon EC2 Auto Scaling en una cuenta de producción para gestionar un aumento inesperado en el tráfico de clientes. Los administradores de emergencias deben seguir el proceso de acceso de emergencia para acceder a la Cuenta de AWS de producción específica y hacer los cambios necesarios.

El proceso de acceso de emergencia se basa en una Cuenta de AWS de acceso de emergencia creada de antemano que se utiliza únicamente para el acceso de emergencia y dispone de recursos de AWS (como los usuarios y roles de IAM) para respaldar el proceso de acceso de emergencia. Durante las operaciones normales, nadie debe acceder a la cuenta de acceso de emergencia y usted debe supervisar y alertar sobre el uso indebido de esta cuenta (para obtener más información, consulte la sección anterior de guía común).

La cuenta de acceso de emergencia tiene roles de IAM de acceso de emergencia con permisos para asumir roles entre cuentas en las Cuentas de AWS que requieran acceso de emergencia. Estos roles de IAM se crean de antemano y se configuran con políticas de confianza que confían en los roles de IAM de la cuenta de emergencia.

El proceso de acceso de emergencia puede utilizar uno de los siguientes enfoques:

- Puede crear previamente un conjunto de usuarios de IAM para sus administradores de emergencia en la cuenta de acceso de emergencia con contraseñas seguras y tokens de MFA asociados. Estos usuarios de IAM tienen permisos para asumir los roles de IAM que, entonces, permiten el acceso entre cuentas a la Cuenta de AWS donde se requiere el acceso de emergencia. Recomendamos crear el menor número posible de usuarios y asignar cada usuario a un único administrador de emergencias. Durante una emergencia, un usuario administrador de emergencias inicia sesión en la cuenta de acceso de emergencia con su contraseña y el código de token de MFA, cambia el rol de IAM de acceso de emergencia en la cuenta de emergencia y, finalmente, cambia el rol de IAM de acceso de emergencia en la cuenta de carga de trabajo para llevar a cabo la acción de cambio de emergencia. La ventaja de este enfoque es que cada usuario de IAM se asigna a un administrador de emergencias y usted puede saber qué usuario inició sesión al revisar los eventos de CloudTrail. La desventaja es que hay que mantener varios usuarios de IAM con sus contraseñas de larga duración y tokens de MFA asociados.
- Puede usar el usuario raíz de Cuenta de AWS de acceso de emergencia para iniciar sesión en la cuenta de acceso de emergencia, asumir el rol de IAM de acceso de emergencia y asumir el rol entre cuentas en la cuenta de carga de trabajo. Recomendamos configurar una contraseña segura y varios tokens de MFA para el usuario raíz. También recomendamos almacenar la contraseña y los tokens de MFA en un almacén de credenciales empresarial seguro que aplique una autenticación y una autorización sólidas. Debe proteger los factores de restablecimiento de la contraseña y el token de MFA. Para ello, establezca la dirección de correo electrónico de la cuenta en una lista de distribución de correo electrónico supervisada por los administradores de seguridad en la nube y el número de teléfono de la cuenta en un número de teléfono compartido también supervisado por los administradores de seguridad. La ventaja de este enfoque es que solo hay que administrar un conjunto de credenciales de usuario raíz. La desventaja es que, dado que se trata de un usuario compartido, es posible que varios administradores inicien sesión como usuario raíz. Debe auditar los eventos de registro del almacén empresarial para identificar qué administrador extrajo la contraseña del usuario raíz.

Modo de error 2: la configuración del proveedor de identidades en AWS se ha modificado o ha caducado

Para permitir que los usuarios de la plantilla se federen en Cuentas de AWS, puede configurar IAM Identity Center con un proveedor de identidades externo o crear un proveedor de identidades de IAM (SEC02-BP04). Por lo general, estos se configuran al importar un documento XML de metadatos de SAML que proporciona el proveedor de identidades. El documento XML de metadatos incluye un

certificado X.509 que corresponde a una clave privada que el proveedor de identidades utiliza para firmar sus aserciones SAML.

Un administrador podría modificar o eliminar estas configuraciones de AWS de forma accidental. En otro escenario, el certificado X.509 importado a AWS podría caducar cuando aún no se ha importado a AWS un nuevo XML de metadatos con un certificado nuevo. Ambas situaciones pueden desbaratar la federación a AWS de los usuarios de la plantilla y provocar una emergencia.

En un caso de emergencia de este tipo, puede proporcionar a sus administradores de identidades acceso a AWS para solucionar los problemas de federación. Por ejemplo, el administrador de identidades utiliza el proceso de acceso de emergencia para iniciar sesión en la Cuenta de AWS de acceso de emergencia, cambia a un rol en la cuenta de administrador del centro de identidades y actualiza la configuración del proveedor de identidades externo al importar el último documento XML de metadatos SAML de su proveedor de identidades para volver a habilitar la federación. Una vez que se corrija la federación, los usuarios de la plantilla seguirán utilizando el proceso operativo normal para federarse en sus cuentas de carga de trabajo.

Puede seguir los enfoques detallados en el modo de error 1 anterior para crear un proceso de acceso de emergencia. Puede conceder permisos con privilegios mínimos a sus administradores de identidades para que accedan únicamente a la cuenta de administrador del centro de identidades y lleven a cabo acciones en el centro de identidades en esa cuenta.

Modo de error 3: interrupción del centro de identidades

En el caso poco probable de que se produzca una interrupción en IAM Identity Center o en una Región de AWS, le recomendamos que establezca una configuración que pueda utilizar para proporcionar acceso temporal a la Consola de administración de AWS.

El proceso de acceso de emergencia utiliza la federación directa desde su proveedor de identidades a IAM en una cuenta de emergencia. Para obtener información detallada sobre el proceso y las consideraciones de diseño, consulte Set up emergency access to the Consola de administración de AWS.

Pasos para la implementación

Pasos comunes para todos los modos de error

- Cree una Cuenta de AWS dedicada a los procesos de acceso de emergencia. Cree de antemano los recursos de IAM necesarios en la cuenta, como roles de IAM o usuarios de IAM, y opcionalmente, proveedores de identidades de IAM. Además, cree de antemano roles de IAM entre cuentas en las Cuentas de AWS de la carga de trabajo con relaciones de confianza

con los roles de IAM correspondientes en la cuenta de acceso de emergencia. Puede usar CloudFormation StackSets con AWS Organizations para crear dichos recursos en las cuentas de los miembros de su organización.

- Cree políticas de control de servicio (SCP) de AWS Organizations para denegar la eliminación y modificación de los roles de IAM entre cuentas en las Cuentas de AWS de miembros.
- Habilite CloudTrail para la Cuenta de AWS de acceso de emergencia y envíe los eventos de ruta a un bucket de S3 central en su Cuenta de AWS de recopilación de registros. Si utiliza AWS Control Tower para configurar y gobernar su entorno multicuenta de AWS, cada cuenta que cree con AWS Control Tower o inscriba en AWS Control Tower tendrá CloudTrail habilitado de forma predeterminada y se enviará a un bucket de S3 en una Cuenta de AWS de archivo de registro dedicada.
- Supervise la actividad de la cuenta de acceso de emergencia mediante la creación de reglas de EventBridge que concuerden con el inicio de sesión de la consola y la actividad de la API por parte de los roles de IAM de emergencia. Envíe notificaciones a su centro de operaciones de seguridad cuando se produzca actividad fuera de un evento de emergencia continuo registrado en su sistema de administración de incidentes.

Pasos adicionales para el modo de error 1: el proveedor de identidades utilizado para federarse en AWS no está disponible y el modo de error 2: la configuración del proveedor de identidades en AWS se ha modificado o ha caducado

- Cree de antemano los recursos en función del mecanismo que elija para el acceso de emergencia:
  - Uso de usuarios de IAM: cree de antemano los usuarios de IAM con contraseñas seguras y los dispositivos MFA asociados.
  - Uso del usuario raíz de la cuenta de emergencia: configure el usuario raíz con una contraseña segura y almacene la contraseña en el almacén de credenciales de su empresa. Asocie varios dispositivos MFA físicos al usuario raíz y almacene los dispositivos en lugares a los que puedan acceder rápidamente los miembros de su equipo de administradores de emergencias.

Pasos adicionales para el modo de error 3: interrupción del centro de identidades

- Tal como se describe en Set up emergency access to the Consola de administración de AWS, en la Cuenta de AWS de acceso de emergencia, cree un proveedor de identidades de IAM para habilitar la federación SAML directa desde su proveedor de identidades.

- Cree grupos de operaciones de emergencia en su IdP sin miembros.
- Cree los roles de IAM correspondientes a los grupos de operaciones de emergencia en la cuenta de acceso de emergencia.

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- SEC02-BP04 Uso de un proveedor de identidades centralizado
- SEC03-BP02 Concesión de acceso con privilegios mínimos
- SEC10-BP02 Desarrollo de planes de administración de incidentes
- SEC10-BP07 Ejecución de simulaciones

Documentos relacionados:

- Set up emergency access to the Consola de administración de AWS
- Concesión de acceso a la a los usuarios federados SAML Consola de administración de AWS
- Break glass access

Videos relacionados:

- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive

Ejemplos relacionados:

- AWS Break Glass Role
- AWS customer playbook framework
- AWS incident response playbook samples

SEC03-BP04 Reducción continua de los permisos

A medida que los equipos determinen qué acceso es necesario, elimine los permisos innecesarios y establezca procesos de revisión para conseguir permisos con privilegios mínimos. Supervise

y elimine continuamente las identidades y los permisos que no se utilicen, tanto para el acceso humano como para el de las máquinas.

Resultado deseado: las políticas de permisos deben cumplir con el principio de privilegio mínimo. A medida que se definan mejor las responsabilidades y los roles del trabajo, debe revisar sus políticas de permisos para eliminar los permisos innecesarios. Este enfoque reduce el alcance del impacto en caso de que las credenciales se expongan de forma inadvertida o se acceda a ellas sin autorización.

Patrones comunes de uso no recomendados:

- Conceder permisos de administrador a los usuarios de forma predeterminada.
- Crear políticas excesivamente permisivas, pero sin todos los privilegios de administrador.
- Mantener políticas de permisos después de que ya no son necesarias.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cuando los equipos y los proyectos están dando sus primeros pasos, utilizar unas políticas de permisos permisivas sirve para fomentar la innovación y la agilidad. Por ejemplo, en un entorno de desarrollo o de pruebas, se puede dar acceso a los desarrolladores a un amplio conjunto de servicios de AWS. Recomendamos que evalúe el acceso continuamente y lo restrinja únicamente a aquellos servicios y acciones de servicio que sean necesarios para llevar a cabo el trabajo actual. Recomendamos llevar a cabo esta evaluación tanto para las identidades humanas como para las de máquina. Las identidades de máquina, que a veces se denominan cuentas del sistema o del servicio, son identidades que dan acceso a AWS a aplicaciones o servidores. Este acceso es especialmente importante en un entorno de producción, donde unos permisos demasiado permisivos pueden tener un impacto enorme y el potencial de exponer los datos de los clientes.

AWS tiene numerosos métodos para ayudar a identificar a los usuarios, roles, permisos y credenciales no utilizados. AWS también puede ayudar a analizar la actividad de acceso de los usuarios y roles de IAM, incluidas las claves de acceso asociadas, y el acceso a recursos de AWS, como los objetos de los buckets de Amazon S3. La generación de políticas de AWS Identity and Access Management Access Analyzer puede ayudarle a crear políticas de permisos restrictivas basadas en los servicios y acciones reales con los que interactúa una entidad principal. El control de acceso basado en atributos (ABAC) puede ayudar a simplificar la administración de permisos, ya que permite proporcionar permisos a los usuarios mediante sus atributos en lugar de asociar las políticas de permisos directamente a cada usuario.

Pasos para la implementación

- Uso de AWS Identity and Access Management Access Analyzer: el Analizador de acceso de IAM le ayuda a identificar los recursos de su organización y sus cuentas, como los buckets de Amazon Simple Storage Service (Amazon S3) o los roles de IAM, que se comparten con una entidad externa.
- Uso de la generación de políticas del Analizador de acceso de IAM: la generación de políticas del Analizador de acceso de IAM le ayuda a crear políticas de permisos detalladas basadas en la actividad de acceso de un usuario o rol de IAM.
- Pruebe los permisos en los entornos inferiores antes de la producción: comience por utilizar los entornos de pruebas y de desarrollo menos críticos a fin de probar los permisos necesarios para diversas funciones de trabajo con IAM Access Analyzer. A continuación, ajuste y valide progresivamente estos permisos en los entornos de pruebas, control de calidad y ensayo antes de aplicarlos a la producción. Los entornos más bajos pueden tener permisos más flexibles al principio, ya que las políticas de control de servicios (SCP) imponen barreras de protección al limitar el número máximo de permisos concedidos.
- Determinación de un marco temporal y una política de uso aceptables para los usuarios y roles de IAM: use la marca de tiempo del último acceso para identificar los usuarios y roles no utilizados y eliminarlos. Revise la información del último acceso a servicios y acciones para identificar y establecer el alcance de los permisos para usuarios y roles específicos. Por ejemplo, puede utilizar la información sobre el último acceso para identificar las acciones específicas de Amazon S3 necesarias para el rol de su aplicación y restringir el acceso únicamente a dichas acciones. Estas características de información sobre el último acceso están disponibles en la Consola de administración de AWS y permiten de manera programática incorporarlas en sus flujos de trabajo de infraestructura y sus herramientas automatizadas.
- Consideración de la posibilidad de registrar eventos de datos en AWS CloudTrail: de manera predeterminada, CloudTrail no registra los eventos de datos, como la actividad de nivel de objeto de Amazon S3 (por ejemplo, `GetObject` y `DeleteObject`) o las actividades de las tablas de Amazon DynamoDB (por ejemplo `PutItem` y `DeleteItem`). Considere la posibilidad de habilitar el registro de estos eventos para determinar qué usuarios y roles necesitan acceder a objetos de Amazon S3 o elementos de tabla de DynamoDB específicos.

Recursos

Documentos relacionados:

- Aplicar permisos de privilegios mínimos
- Revisar y eliminar periódicamente usuarios, roles, permisos, políticas y credenciales no utilizados
- What is AWS CloudTrail?
- Uso de políticas de
- Registro y supervisión en DynamoDB
- Habilitación del registro de eventos de CloudTrail para buckets y objetos de Amazon S
- Generación de informes de credenciales para su Cuenta de AWS

Videos relacionados:

- Become an IAM Policy Master in 60 Minutes or Less
- Separation of Duties, Least Privilege, Delegation, and CI/CD
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive

SEC03-BP05 Definición de las barreras de protección de los permisos para una organización

Utilice las barreras de protección de permisos para reducir el ámbito de los permisos disponibles que se pueden conceder a las entidades principales. La cadena de evaluación de la política de permisos incluye sus barreras de protección para determinar los permisos efectivos de una entidad principal al tomar decisiones de autorización.  Puede definir barreras de protección mediante un enfoque basado en capas. Aplique algunas barreras de protección de manera generalizada en toda la organización y aplique otras de forma específica a las sesiones de acceso temporal.

Resultado deseado: cuenta con un aislamiento claro de los entornos mediante el uso de Cuentas de AWS separadas.  Las políticas de control de servicios (SCP) se utilizan para definir las barreras de protección de permisos en toda la organización. Las barreras de protección más amplias se establecen en los niveles jerárquicos más cercanos a la raíz de la organización, y las más estrictas se establecen más cerca del nivel de las cuentas individuales.

En los casos en los que se pueden utilizar, las políticas de recursos definen las condiciones que debe cumplir una entidad principal para tener acceso a un recurso. Las políticas de recursos también acotan el conjunto de acciones permitidas cuando corresponde. Los límites de permisos se aplican a entidades principales que administran permisos de las cargas de trabajo y delegan la administración de permisos a los propietarios individuales de las cargas de trabajo.

Patrones comunes de uso no recomendados:

- Crear Cuentas de AWS de miembros dentro de una organización de AWS, pero no usar SCP para restringir el uso y los permisos disponibles para sus credenciales raíz.
- Asignar permisos según el principio de privilegio mínimo, pero sin aplicar barreras de protección al conjunto máximo de permisos que se pueden conceder.
- Confiar en el principio de denegación implícita de AWS IAM para restringir los permisos y esperar que las políticas no concedan un permiso explícito no deseado.
- Ejecutar varios entornos de carga de trabajo en la misma Cuenta de AWS y, a continuación, recurrir a mecanismos como las VPC, las etiquetas o las políticas de recursos para hacer cumplir los límites de los permisos.

Beneficios de establecer esta práctica recomendada: las barreras de protección de permisos ayudan a generar confianza en que no se van a conceder permisos no deseados, incluso cuando una política de permisos intente hacerlo.  Esto puede simplificar la definición y la administración de los permisos al reducir el ámbito máximo de los permisos que deben tenerse en cuenta.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Le recomendamos que utilice un enfoque basado en capas para definir las barreras de protección de permisos para su organización. Este enfoque reduce sistemáticamente el conjunto máximo de permisos posibles a medida que se aplican capas adicionales. Esto le ayuda a conceder acceso según el principio de privilegios mínimos, lo que reduce el riesgo de accesos no deseados debidos a una configuración errónea de las políticas.

El primer paso para establecer barreras de protección de permisos es aislar las cargas de trabajo y los entornos en Cuentas de AWS separadas. Las entidades principales de una cuenta no pueden acceder a los recursos de otra cuenta sin un permiso explícito para hacerlo, incluso aunque ambas cuentas se encuentren en la misma organización de AWS o en la misma unidad organizativa (OU). Puede usar las unidades organizativas para agrupar las cuentas que prefiera administrar como una sola unidad.

El siguiente paso consiste en reducir el conjunto máximo de permisos que puede conceder a las entidades principales dentro de las cuentas de los miembros de su organización. Para ello, puede usar las políticas de control de servicios (SCP), que puede aplicar a una unidad organizativa o a una cuenta. Las SCP pueden aplicar controles de acceso comunes, como restringir el acceso a determinadas Regiones de AWS, ayudar a evitar que se eliminen recursos o deshabilitar acciones de servicio potencialmente arriesgadas. Las SCP que se apliquen a la raíz de su organización solo

afectan a las cuentas de los miembros, no a la cuenta de administración. Las SCP solo controlan las entidades principales de su organización. Las SCP no controlan las entidades principales externas a su organización que accedan a sus recursos.

Si está utilizando AWS Control Tower, puede aprovechar sus controles y zonas de aterrizaje como base para sus barreras de protección de permisos y su entorno de múltiples cuentas. Las zonas de aterrizaje proporcionan un entorno básico seguro y preconfigurado con cuentas independientes para diferentes cargas de trabajo y aplicaciones. Las barreras de protección imponen controles obligatorios en materia de seguridad, operaciones y cumplimiento mediante una combinación de políticas de control de servicios (SCP), reglas de AWS Config y otras configuraciones. Sin embargo, cuando se utilizan las barreras de protección y las zonas de aterrizaje de Control Tower junto con los SCP personalizados de las organizaciones, es fundamental seguir las prácticas recomendadas descritas en la documentación de AWS para evitar conflictos y garantizar un gobierno adecuado. Consulte la guía de AWS Control Tower para AWS Organizations a fin de obtener recomendaciones detalladas sobre la administración de los SCP, las cuentas y las unidades organizativas (OU) en un entorno de Control Tower.

Si sigue estas directrices, podrá aprovechar de forma eficaz las barreras de protección, las zonas de aterrizaje y los SCP personalizados de Control Tower, a la vez que mitigará los posibles conflictos y garantizará el gobierno y el control adecuados de su entorno de múltiples cuentas de AWS.

Otro paso consiste en usar políticas de recursos de IAM para determinar el ámbito de las acciones disponibles que se pueden llevar a cabo con respecto a los recursos que controlan, junto con cualquier condición que deba cumplir la entidad principal activa. El ámbito puede ser tan amplio como para permitir todas las acciones siempre que la entidad principal forme parte de su organización (mediante la clave de condición PrincipalOrgID), o tan detallado como para permitir solo acciones específicas para un rol de IAM específico. Puede adoptar un enfoque similar con las condiciones de las políticas de confianza para un rol de IAM. Si una política de confianza de recursos o roles nombra explícitamente una entidad principal en la misma cuenta que el rol o el recurso que controla, esa entidad principal no necesita una política de IAM asociada que otorgue los mismos permisos. Si la entidad principal está en una cuenta diferente a la del recurso, esa entidad principal necesita una política de IAM asociada que otorgue esos permisos.

A menudo, un equipo de carga de trabajo querrá administrar los permisos que requiere su carga de trabajo. Esto podría exigirles la creación de nuevos roles de IAM y políticas de permisos. Puede definir el alcance máximo de los permisos que el equipo puede conceder dentro en un límite de permisos de IAM y asociar este documento a un rol de IAM que el equipo pueda utilizar posteriormente para gestionar sus permisos y roles de IAM. Este método puede proporcionarles

la capacidad de completar su trabajo y, al mismo tiempo, mitigar los riesgos de disponer de acceso administrativo a IAM.

Un paso más detallado consiste en implementar técnicas de administración de acceso privilegiado (PAM) y administración de acceso elevado temporal (TEAM). Un ejemplo de PAM consiste en exigir a las entidades principales que lleven a cabo una autenticación multifactor antes de tomar medidas privilegiadas. Para obtener más información, consulte Configuring MFA-protected API access. TEAM requiere una solución que administre la aprobación y el plazo en el que se permite que una entidad principal tenga acceso de alto nivel. Un enfoque consiste en agregar temporalmente la entidad principal a la política de confianza del rol para un rol de IAM que tenga un acceso de alto nivel. Otro enfoque consiste, en condiciones normales, en reducir los permisos que un rol de IAM concede a una entidad principal mediante una política de sesión y, a continuación, eliminar temporalmente esta restricción durante el periodo de tiempo aprobado. Para obtener más información sobre las soluciones que AWS y determinados socios han validado, consulte Temporary elevated access.

Pasos para la implementación

1. Aísle las cargas de trabajo y los entornos en Cuentas de AWS separadas.
2. Use las SCP para reducir el conjunto máximo de permisos que se pueden conceder a las entidades principales dentro de las cuentas de los miembros de su organización.
   a. Al definir las SCP para reducir el conjunto máximo de permisos que se pueden conceder a las entidades principales dentro de las cuentas de los miembros de su organización, puede elegir entre una estrategia de lista de permitidos o de lista de denegaciones. La estrategia de listas de permisos especifica de forma explícita el acceso permitido y bloquea de forma implícita todos los demás accesos. La estrategia de listas de permisos especifica de forma explícita el acceso permitido y permite de forma predeterminada todos los demás accesos. Ambas estrategias tienen sus ventajas y desventajas, y la elección adecuada depende de los requisitos específicos y del modelo de riesgo de su organización. Para obtener más información, consulte Strategy for using SCPs.
   b. Además, revise los ejemplos de políticas de control de servicios para comprender cómo construir los SCP de manera eficaz.
3. Utilice las políticas de recursos de IAM para acotar y especificar las condiciones para las acciones permitidas en los recursos. Use las condiciones de las políticas de confianza en roles de IAM para crear restricciones a la hora de asumir roles.
4. Asigne límites de permisos de IAM a los roles de IAM que los equipos de carga de trabajo puedan usar para administrar sus propios roles y permisos de IAM en las cargas de trabajo.
5. Evalúe las soluciones de PAM y TEAM en función de sus necesidades.

Recursos

Documentos relacionados:

- Data perimeters on AWS
- Establecimiento de barreras de protección de permisos mediante perímetros de datos
- Lógica de evaluación de políticas

Ejemplos relacionados:

- Service control policy examples

Herramientas relacionadas:

- AWS Solution: Temporary Elevated Access Management
- Validated security partner solutions for TEAM

SEC03-BP06 Administración del acceso en función del ciclo de vida

Supervise y ajuste los permisos otorgados a sus entidades principales (usuarios, cargos y grupos) a lo largo de su ciclo de vida dentro de su organización. Ajuste la pertenencia a grupos a medida que los usuarios cambien de cargo y elimine el acceso cuando un usuario abandone la organización.

Resultado deseado: supervisa y ajusta los permisos a lo largo del ciclo de vida de los directores de la organización, lo que reduce el riesgo de privilegios innecesarios. Concede el acceso pertinente al crear un usuario. Modifica el acceso a medida que cambien las responsabilidades del usuario y elimina el acceso cuando el usuario ya no está activo o ha abandonado la organización. Administra de forma centralizada los cambios en los usuarios, los cargos y los grupos. Utiliza la automatización para propagar los cambios en sus entornos de AWS.

Patrones comunes de uso no recomendados:

- Concede privilegios de acceso excesivos o amplios a las identidades por adelantado, más allá de lo que se requiere inicialmente.
- No revisa ni ajusta los privilegios de acceso, a medida que los cargos y las responsabilidades cambian con el tiempo.
- Deja identidades inactivas o terminadas con privilegios de acceso activos. Esto aumenta el riesgo de acceso no autorizado.

- No aprovecha la automatización para gestionar el ciclo de vida de las identidades.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Administre y ajuste cuidadosamente los privilegios de acceso que otorga a las identidades (como usuarios, cargos y grupos) a lo largo de su ciclo de vida. Este ciclo de vida incluye la fase de incorporación inicial, los cambios continuos en los cargos y las responsabilidades y, en última instancia, la baja o el despido. Gestione el acceso de forma proactiva en función de la etapa del ciclo de vida para mantener el nivel de acceso adecuado. Respete el principio de privilegio mínimo para reducir el riesgo de privilegios de acceso excesivos o innecesarios.

Puede administrar el ciclo de vida de los usuarios de IAM directamente dentro de la Cuenta de AWS, o mediante la federación del proveedor de identidades de su personal con AWS IAM Identity Center. Para los usuarios de IAM, puede crear, modificar y eliminar usuarios y sus permisos asociados dentro de la Cuenta de AWS. En el caso de los usuarios federados, puede utilizar IAM Identity Center para administrar su ciclo de vida mediante la sincronización de la información de usuarios y grupos del proveedor de identidades de su organización mediante el protocolo del sistema de administración de identidades entre dominios (System for Cross-domain Identity Management o SCIM).

El SCIM es un protocolo estándar abierto para el aprovisionamiento y desaprovisionamiento automatizados de identidades de usuario en diferentes sistemas. Al integrar su proveedor de identidades con IAM Identity Center mediante SCIM, puede sincronizar automáticamente la información de los usuarios y los grupos, lo que ayuda a validar que los privilegios de acceso se concedan, modifiquen o revoquen en función de los cambios en la fuente de identidad autorizada de su organización.

A medida que cambien los cargos y responsabilidades de los empleados dentro de su organización, ajuste sus privilegios de acceso en consecuencia. Puede usar los conjuntos de permisos de IAM Identity Center para definir diferentes cargos o responsabilidades laborales y asociarlos a las políticas y los permisos de IAM correspondientes. Cuando cambia el cargo de un empleado, puede actualizar su conjunto de permisos asignado para que refleje sus nuevas responsabilidades. Verifique que tenga el acceso necesario y, al mismo tiempo, cumpla el principio de privilegio mínimo.

Pasos para la implementación

1. Defina y documente un proceso del ciclo de vida de la administración de accesos, incluidos los procedimientos para conceder el acceso inicial, las revisiones periódicas y la baja.

2. Implemente límites de roles, grupos y permisos de IAM para administrar el acceso de manera colectiva y aplicar los niveles de acceso máximos permitidos.
3. Integre con un proveedor de identidades federado (como Microsoft Active Directory, Okta o Ping Identity) como fuente autorizada de la información de usuarios y grupos mediante IAM Identity Center.
4. Utilice el protocolo SCIM para sincronizar la información de usuarios y grupos del proveedor de identidades con el Almacén de identidades de IAM Identity Center.
5. Cree conjuntos de permisos en IAM Identity Center que representen diferentes cargos o responsabilidades laborales dentro de su organización. Defina las políticas y los permisos de IAM adecuados para cada conjunto de permisos.
6. Implemente revisiones del acceso periódicas, medidas de revocación rápida del acceso y mejora continua del proceso del ciclo de vida de la administración accesos.
7. Proporcione formación y concienciación a los empleados sobre las prácticas recomendadas de administración de accesos.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP04 Uso de un proveedor de identidades centralizado

Documentos relacionados:

- Manage your identity source
- Manage identities in IAM Identity Center
- Uso de AWS Identity and Access Management Access Analyzer
- Generación de políticas del Analizador de acceso de IAM

Videos relacionados:

- AWS re:Inforce 2023 - Manage temporary elevated access with AWS IAM Identity Center
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2022 - Harness power of IAM policies & rein in permissions w/Access Analyzer

### SEC03-BP07 Análisis del acceso público y entre cuentas

Supervise continuamente los resultados que pongan en relieve el acceso público y entre cuentas. Reduzca el acceso público y el acceso entre cuentas solo a los recursos que requieran este tipo de acceso.

Resultado deseado: sepa cuáles de los recursos de AWS se comparten y con quién. Supervise y audite continuamente sus recursos compartidos para verificar que solo se compartan con las entidades principales autorizadas.

Patrones comunes de uso no recomendados:

- No mantener un inventario de los recursos compartidos.
- No seguir un proceso para aprobar el acceso público o entre cuentas a los recursos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Si su cuenta pertenece a AWS Organizations, puede conceder acceso a los recursos a toda la organización, a unidades organizativas específicas o a cuentas individuales. Si su cuenta no es miembro de una organización, puede compartir recursos con cuentas individuales. Puede conceder acceso entre cuentas mediante políticas basadas en recursos (por ejemplo, las políticas de buckets de Amazon Simple Storage Service (Amazon S3), o si permite que la entidad principal de otra cuenta asuma un rol de IAM en su cuenta. Cuando utilice políticas de recursos, compruebe que solo se concede acceso a las entidades principales autorizadas. Defina un proceso para aprobar todos los recursos que deban estar disponibles públicamente.

AWS Identity and Access Management Access Analyzer usa la seguridad comprobable para identificar todas las rutas de acceso a un recurso desde fuera de su cuenta. Revisa continuamente las políticas de recursos e informa de los resultados del acceso público y entre cuentas para facilitarle el análisis de un acceso potencialmente amplio. Considere la posibilidad de configurar el Analizador de acceso de IAM con AWS Organizations para verificar que tiene visibilidad en todas sus cuentas. El Analizador de acceso de IAM también permite obtener una vista previa de los resultados antes de implementar los permisos de recursos. Esto le permite validar que sus cambios de política conceden solo el acceso público y entre cuentas previsto a sus recursos. Al diseñar el acceso a varias cuentas, puede utilizar políticas de confianza para controlar en qué casos se puede asumir un rol. Por ejemplo, puede usar la clave de condición `PrincipalOrgId` para denegar un intento de asumir un rol desde fuera de AWS Organizations.

AWS Config puede informar de los recursos que están mal configurados y, mediante comprobaciones de políticas de AWS Config, puede detectar los recursos que tienen configurado el acceso público. Los servicios como AWS Control Tower y AWS Security Hub CSPM simplifican la implementación de controles y barreras de protección en AWS Organizations para identificar y corregir los recursos expuestos públicamente. Por ejemplo, AWS Control Tower tiene una barrera de protección administrada que puede detectar si Cuentas de AWS puede restaurar alguna instantánea de Amazon EBS.

Pasos para la implementación

- Planteamiento de uso de AWS Config para AWS Organizations: AWS Config permite agregar los resultados de varias cuentas de una AWS Organizations cuenta de administrador delegado. Esto proporciona una visión completa y le permite implementar Reglas de AWS Config en todas las cuentas para detectar los recursos de acceso público.
- Configuración de AWS Identity and Access Management Access Analyzer: el Analizador de acceso de IAM lo ayuda a identificar los recursos y cuentas de su organización, como los buckets de Amazon S3 o los roles de IAM que se comparten con una entidad externa.
- Uso de la corrección automática de AWS Config para responder a los cambios en la configuración de acceso público de los buckets de Amazon S3: puede activar automáticamente la configuración de bloqueo de acceso público para los buckets de Amazon S3.
- Implementación de la supervisión y las alertas para identificar si los buckets de Amazon S3 se han convertido en públicos: debe disponer de supervisión y alertas para identificar cuándo está desactivado el bloqueo de acceso público de Amazon S3 y si los buckets de Amazon S3 pasan a ser públicos. Además, si utiliza AWS Organizations, puede crear una política de control de servicios que impida cambios en las políticas de acceso público de Amazon S3. AWS Trusted Advisor comprueba los buckets de Amazon S3 que tienen permisos de acceso abierto. Los permisos del bucket que otorgan, suben o eliminan el acceso para todo el mundo crean posibles vulnerabilidades de seguridad, ya que permiten que cualquiera agregue, modifique o elimine elementos en un bucket. La comprobación de Trusted Advisor examina los permisos explícitos del bucket y las políticas asociadas que podrían anular los permisos del bucket. También puede utilizar AWS Config para supervisar si sus buckets de Amazon S3 tienen acceso público. Para obtener más información, consulte el blog How to Use AWS Config to Monitor for and Respond to Amazon S3 Buckets Allowing Public Access.

Al revisar los controles de acceso de los buckets de Amazon S3, es importante tener en cuenta la naturaleza de los datos almacenados en ellos. Amazon Macie es un servicio diseñado para ayudarlo

a descubrir y proteger datos confidenciales, como la información de identificación personal (PII), la información de salud protegida (PHI) y las credenciales, como las claves privadas o las claves de acceso de AWS.

Recursos

Documentos relacionados:

- Uso de AWS Identity and Access Management Access Analyzer
- AWS Control Tower controls library
- AWS Foundational Security Best Practices standard
- AWS Config Reglas de administradas
- AWS Trusted Advisor check reference
- Monitoring AWS Trusted Advisor check results with Amazon EventBridge
- Managing AWS Config Rules Across All Accounts in Your Organization
- AWS Config y AWS Organizations
- Publicación de la AMI para utilizarla en Amazon EC2

Videos relacionados:

- Best Practices for securing your multi-account environment
- Dive Deep into IAM Access Analyzer

SEC03-BP08 Uso compartido de recursos de forma segura en su organización

A medida que el número de cargas de trabajo va aumentando, es posible que necesite compartir el acceso a los recursos de esas cargas de trabajo o aprovisionar los recursos varias veces entre varias cuentas. Es posible que disponga de constructos para compartimentar el entorno, como, por ejemplo, entornos de desarrollo, pruebas y producción. Sin embargo, disponer de constructos de separación no le impide compartir de forma segura. Al compartir componentes que se solapan, puede reducir la sobrecarga operativa y conseguir una experiencia uniforme sin tener que adivinar qué podría haber pasado por alto al crear el mismo recurso varias veces.

Resultado deseado: minimice el acceso no deseado mediante el uso de métodos seguros para compartir los recursos dentro de su organización y ayudar con su iniciativa de prevención de la pérdida de datos. Reduzca la sobrecarga operativa en comparación con la administración de componentes individuales, reduzca los errores derivados de la creación manual del mismo

componente varias veces y aumentar la escalabilidad de las cargas de trabajo. Puede disminuir el tiempo de resolución en situaciones con varios puntos de fallo y aumentar su confianza a la hora de determinar cuándo un componente ya no es necesario. Para obtener una guía prescriptiva sobre el análisis de los recursos compartidos externamente, consulte SEC03-BP07 Análisis del acceso público y entre cuentas.

Patrones comunes de uso no recomendados:

- Falta de un proceso para supervisar continuamente y alertar automáticamente sobre un uso compartido externo inesperado.
- Falta de una referencia sobre lo que se debe compartir y lo que no.
- Adoptar de manera predeterminada una política muy abierta en lugar de compartir explícitamente cuando es necesario.
- Crear manualmente recursos fundamentales que se solapan cuando es necesario.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Diseñe sus controles y patrones de acceso para que rijan el consumo de recursos compartidos de forma segura y solo con entidades de confianza. Supervise los recursos compartidos y revise el acceso a ellos de forma continua; además, reciba alertas sobre un uso compartido inapropiado o inesperado. Revise Analizar el acceso público y entre cuentas para que le sirva de ayuda para establecer una gobernanza que reduzca el acceso externo únicamente a los recursos que lo requieran, y para establecer un proceso de monitoreo continuo y alertas automáticas.

Servicios de AWS como, por ejemplo, AWS Security Hub CSPM, Amazon GuardDuty y AWS Backup, permiten el uso compartido entre cuentas en AWS Organizations. Estos servicios permiten compartir datos con una cuenta central, acceder a ellos desde una cuenta central o administrar recursos y datos desde una cuenta central. Por ejemplo, AWS Security Hub CSPM puede transferir resultados desde cuentas individuales a una cuenta central en la que podrá verlos todos. AWS Backup puede hacer una copia de seguridad de un recurso y compartirlo entre varias cuentas. Puede usar AWS Resource Access Manager (AWS RAM) para compartir otros recursos comunes, como subredes de VPC y conexiones de puerta de enlace de tránsito, AWS Network Firewall o canalizaciones de Amazon SageMaker AI.

Para restringir su cuenta y compartir solo los recursos de su organización, utilice las políticas de control de servicios (SCP) para impedir el acceso a entidades principales externas. Al compartir

recursos, combine los controles basados en la identidad y los controles de red para crear un perímetro de datos para su organización que ofrezca protección frente al acceso no deseado. Un perímetro de datos es un conjunto de barreras de protección preventivas para ayudar a verificar que solo sus identidades de confianza accedan a los recursos de confianza desde las redes previstas. Estos controles ponen límites apropiados a los recursos que se pueden compartir y evitan que se compartan o expongan recursos que no deberían permitirse. Por ejemplo, como parte de su perímetro de datos, puede utilizar las políticas de punto de conexión de VPC y la condición de `AWS:PrincipalOrgId` para garantizar que las identidades que acceden a sus buckets de Amazon S3 pertenezcan a su organización. Es importante tener en cuenta que los SCP no se aplican a los roles o entidades principales de AWS vinculados al servicio.

Cuando utilice Amazon S3, desactive las ACL de su bucket de Amazon S3 y utilice las políticas de IAM para definir el control de acceso. Para restringir el acceso a un origen de Amazon S3 desde Amazon CloudFront, migre desde la identidad de acceso de origen (OAI) al control de acceso de origen (OAC), que admite características adicionales como el cifrado del servidor con AWS Key Management Service.

En algunos casos, es posible que desee permitir compartir recursos fuera de su organización o conceder a un tercero acceso a sus recursos. Para obtener una guía prescriptiva sobre la administración de permisos para compartir recursos de forma externa, consulte Administración de permisos.

Pasos para la implementación

1. Utilice AWS Organizations: AWS Organizations es un servicio de administración de cuentas que le permite unificar varias Cuentas de AWS en una organización que crea y administra de forma centralizada. Puede agrupar sus cuentas en unidades organizativas (OU) y asociar diferentes políticas a cada OU para ayudarle a satisfacer sus necesidades presupuestarias, de seguridad y de conformidad. También puede controlar cómo los servicios de inteligencia artificial (IA) y machine learning (ML) de AWS pueden recopilar y almacenar datos, y utilizar la administración de varias cuentas de los servicios de AWS integrada con las organizaciones.
2. Integre AWS Organizations con los servicios de AWS: cuando utiliza un servicio de AWS para efectuar tareas en su nombre en las cuentas de los miembros de su organización, AWS Organizations crea un rol vinculado al servicio (SLR) de IAM para ese servicio en cada cuenta miembro. Debe administrar el acceso de confianza mediante la Consola de administración de AWS, las API de AWS o la AWS CLI. Para obtener instrucciones prescriptivas sobre cómo activar el acceso de confianza, consulte Uso de AWS Organizations con otros servicios de AWS y Servicios de AWS que puede usar con las organizaciones.

3. Establezca un perímetro de datos: un perímetro de datos proporciona un límite claro de confianza y propiedad. En AWS, se suele representar como su organización de AWS administrada por AWS Organizations, junto con cualquier red o sistema local que acceda a sus recursos de AWS. El objetivo del perímetro de datos es verificar que se permite el acceso si la identidad es de confianza, el recurso es de confianza y la red es la que se espera. Sin embargo, establecer un perímetro de datos no es una estrategia universal. Evalúe y adopte los objetivos de control descritos en el documento técnico Construir un perímetro en AWS a partir de sus requisitos y modelos de riesgo de seguridad específicos. Debe plantearse detenidamente su postura de riesgo única e implementar los controles perimetrales que se ajusten a sus necesidades de seguridad.
4. Use los recursos compartidos de los servicios de AWS y aplique las restricciones pertinentes: muchos servicios de AWS le permiten compartir recursos con otra cuenta o dirigirse a un recurso de otra cuenta, como Imágenes de máquina de Amazon (AMI) y AWS Resource Access Manager (AWS RAM). Restrinja la API de `ModifyImageAttribute` para especificar las cuentas de confianza con las que compartir la AMI. Especifique la condición `ram:RequestedAllowsExternalPrincipals` cuando se utilice AWS RAM para restringir el uso compartido únicamente a su organización y, de este modo, evitar el acceso desde identidades que no sean de confianza. Para obtener consideraciones e instrucciones prescriptivas, consulte Uso compartido de recursos y objetivos externos.
5. Utilice AWS RAM para compartir de forma segura en una cuenta o con otras Cuentas de AWS: AWS RAM lo ayuda a compartir de forma segura los recursos que ha creado con roles y usuarios de su cuenta y con otras Cuentas de AWS. En un entorno de varias cuentas, AWS RAM le permite crear un recurso una vez y compartirlo con otras cuentas. Este enfoque ayuda a reducir su sobrecarga operativa a la vez que proporciona coherencia, visibilidad y auditabilidad en integraciones con Amazon CloudWatch y AWS CloudTrail, algo que no tiene cuando utiliza el acceso entre cuentas.

   Si tiene recursos que ha compartido anteriormente mediante una política basada en recursos, puede usar la API de `PromoteResourceShareCreatedFromPolicy` o una equivalente para convertir el uso compartido de recursos en un uso compartido de recursos de AWS RAM completo.

   En algunos casos, puede que tenga que dar pasos adicionales para compartir recursos. Por ejemplo, para compartir una instantánea cifrada, debe compartir una clave de AWS KMS.

Recursos

Prácticas recomendadas relacionadas:

- SEC03-BP07 Análisis del acceso público y entre cuentas
- SEC03-BP09 Uso compartido seguro de recursos con terceros
- SEC05-BP01 Creación de capas de red

Documentos relacionados:

- Propietario de bucket que concede permisos entre cuentas para objetos que no le pertenecen
- How to use Trust Policies with IAM
- Building Data Perimeter on AWS
- Cómo utilizar un ID externo al conceder a un tercero el acceso a sus recursos de AWS
- AWS services you can use with AWS Organizations
- Establishing a data perimeter on AWS: Allow only trusted identities to access company data

Videos relacionados:

- Granular Access with AWS Resource Access Manager
- Securing your data perimeter with VPC endpoints
- Establishing a data perimeter on AWS

Herramientas relacionadas:

- Data Perimeter Policy Examples

SEC03-BP09 Uso compartido seguro de recursos con terceros

La seguridad de su entorno en la nube no se limita a su organización. Su organización puede recurrir a terceros para administrar una parte de sus datos. La administración de permisos para el sistema administrado por terceros debe seguir la práctica del acceso justo a tiempo mediante el principio del privilegio mínimo con credenciales temporales. Si colabora estrechamente con un tercero, podrán reducir juntos el alcance del impacto y el riesgo de un acceso no intencionado.

Resultado deseado: evita el uso de credenciales a largo plazo de AWS Identity and Access Management (IAM), como claves de acceso y claves secretas, ya que representan un riesgo para la seguridad si se utilizan indebidamente. En su lugar, utiliza los roles de IAM y las credenciales temporales para mejorar su posición de seguridad y minimizar la sobrecarga operativa que implica la

administración de credenciales a largo plazo. Al conceder acceso a terceros, utiliza un identificador único universal (UUID) como ID externo en la política de confianza de IAM y mantiene bajo su control las políticas de IAM asociadas al rol para garantizar acceso con privilegios mínimos. Para obtener orientaciones prescriptivas sobre el análisis de recursos compartidos externamente, consulte SEC03-BP07 Análisis del acceso público y entre cuentas.

Patrones comunes de uso no recomendados:

- Utilizar la política de confianza de IAM predeterminada sin ninguna condición.
- Utilizar claves de acceso y credenciales de IAM a largo plazo.
- Reutilizar ID externos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Es posible que desee permitir que se compartan recursos fuera de AWS Organizations o conceder a un tercero acceso a su cuenta. Por ejemplo, es posible que un tercero le proporcione una solución de supervisión que necesite acceder a los recursos de su cuenta. En esos casos, cree un rol entre cuentas de IAM con solo los privilegios que necesite el tercero. Defina, además, una política de confianza mediante la condición de ID externo. Cuando utilice un ID externo, usted o el tercero podrán generar un ID único para cada cliente, tercero o tenencia. El ID único no debe controlarlo nadie más que usted después de crearlo. El tercero debe implementar un proceso para relacionar el ID externo con el cliente de una forma segura, auditable y reproducible.

También puede utilizar IAM Roles Anywhere para administrar los roles de IAM para aplicaciones externas a AWS que usan las API de AWS.

Si el tercero ya no necesita acceder a su entorno, elimine el rol. Procure no proporcionar credenciales a largo plazo a terceros. Conozca otros servicios de AWS que permiten el uso compartido, como AWS Well-Architected Tool, que permite compartir una carga de trabajo con otras Cuentas de AWS, y AWS Resource Access Manager, que lo ayuda a compartir de forma segura un recurso de AWS de su propiedad con otras cuentas.

Pasos para la implementación

1. Utilice roles entre cuentas para permitir el acceso a cuentas externas. Los roles entre cuentas reducen la cantidad de información confidencial que almacenan las cuentas externas y los

terceros para atender a sus clientes. Los roles entre cuentas le permiten conceder acceso a los recursos de AWS de su cuenta de forma segura a terceros, como Socios de AWS u otras cuentas de su organización, al tiempo que mantiene la capacidad de administrar y auditar dicho acceso. El tercero podría estar prestando servicio desde una infraestructura híbrida o extrayendo datos a una ubicación externa. IAM Roles Anywhere lo ayuda a permitir que las cargas de trabajo de terceros interactúen de forma segura con sus cargas de trabajo de AWS y a reducir aún más la necesidad de credenciales a largo plazo.

No debe utilizar credenciales a largo plazo ni claves de acceso asociadas a usuarios para proporcionar acceso a cuentas externas. En su lugar, utilice roles entre cuentas para proporcionar el acceso entre cuentas.

2. Actúe con la diligencia debida y garantice el acceso seguro para los proveedores de SaaS de terceros. Al compartir recursos con proveedores de SaaS externos, actúe estrictamente con la diligencia debida para garantizar que ofrezca una estrategia segura y responsable de acceso a sus recursos de AWS. Evalúe su modelo de responsabilidad compartida para comprender qué medidas de seguridad ofrecen y cuáles son de su responsabilidad. Asegúrese de que el proveedor de SaaS cuente con un proceso seguro y auditable de acceder a sus recursos, incluido el uso de identificadores externos y los principios de acceso con privilegios mínimos. El uso de ID externos ayuda a resolver el problema de la sustitución confusa.

   Implemente controles de seguridad para garantizar un acceso seguro y el cumplimiento del principio de privilegios mínimos al conceder acceso a proveedores de SaaS externos. Esto puede incluir el uso de identificadores externos, identificadores únicos universales (UUID) y políticas de confianza de IAM que limitan el acceso únicamente a lo estrictamente necesario. Trabaje en estrecha colaboración con el proveedor de SaaS para establecer mecanismos de acceso seguro, revise periódicamente su acceso a sus recursos de AWS y lleve a cabo auditorías para garantizar el cumplimiento de sus requisitos de seguridad.
3. Declare obsoletas las credenciales a largo plazo proporcionadas por el cliente. Declare obsoleto el uso de credenciales a largo plazo y utilice roles de cuentas cruzadas o IAM Roles Anywhere. Si debe utilizar credenciales a largo plazo, establezca un plan para migrar al acceso basado en roles. Para obtener más información sobre la administración de claves, consulte Administración de identidades. Reúnase también con su equipo de Cuenta de AWS y el tercero para establecer un manual de procedimientos de mitigación de riesgos. Para obtener orientación normativa sobre cómo responder y mitigar el impacto potencial de un incidente de seguridad, consulte Respuesta ante incidentes.
4. Compruebe que la configuración cuente con una guía prescriptiva o que esté automatizada. El ID externo no debe tratarse como un secreto, pero no debe ser un valor fácil de adivinar, como un

número de teléfono, un nombre o un ID de cuenta. Convierta el ID externo en un campo de solo lectura para que no pueda modificarse con el fin de suplantar la configuración.

El ID externo puede generarlo usted o el tercero. Defina un proceso para determinar quién es el responsable de generar el ID. Independientemente de la entidad que cree el ID externo, el tercero aplica la unicidad y los formatos de manera uniforme en todos los clientes.

La política que se cree para el acceso entre cuentas en sus cuentas debe seguir el principio del privilegio mínimo. El tercero debe proporcionarle un documento de políticas de roles o un mecanismo de configuración automatizado que utilice una plantilla de AWS CloudFormation o algo equivalente. Esto reduce la posibilidad de que se produzcan errores asociados a la creación manual de políticas y ofrece un registro de seguimiento auditable. Para obtener más información sobre el uso de una plantilla de AWS CloudFormation para crear roles entre cuentas, consulte Cross-Account Roles.

El tercero debe proporcionar un mecanismo de configuración automatizado y auditable. Sin embargo, debería automatizar la configuración del rol mediante el documento de la política de roles que describe el acceso necesario. Con una plantilla de AWS CloudFormation o un elemento equivalente, debe supervisar los cambios y utilizar la detección de desviaciones como parte de la práctica de auditoría.

5. Tenga en cuenta los cambios. La estructura de su cuenta, su necesidad de utilizar al tercero o la oferta de servicios que este presta pueden cambiar. Debe anticiparse a los cambios y a los fallos, y planificar en consecuencia las personas, los procesos y la tecnología adecuados. Audite de forma periódica el nivel de acceso que proporciona e implemente métodos de detección que lo alerten de cambios inesperados. Monitoree y audite el uso del rol y el almacén de datos de los ID externos. Debe tenerlo todo preparado para revocar el acceso del tercero, de forma temporal o permanente, a causa de cambios o patrones de acceso inesperados. Asimismo, mida el impacto en su operación de revocación, incluido el tiempo que lleva hacerla, las personas implicadas, el costo y el impacto en otros recursos.

   Para obtener una guía prescriptiva sobre los métodos de detección, consulte las prácticas recomendadas de detección.

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP02 Uso de credenciales temporales

- SEC03-BP05 Definición de las barreras de protección de los permisos para una organización
- SEC03-BP06 Administración del acceso en función del ciclo de vida
- SEC03-BP07 Análisis del acceso público y entre cuentas
- SEC04 Detección

Documentos relacionados:

- Propietario de bucket que concede permisos entre cuentas para objetos que no le pertenecen
- How to use trust policies with IAM roles
- Delegación del acceso entre Cuentas de AWS mediante roles de IAM
- ¿Cómo accedo a los recursos de otra Cuenta de AWS mediante IAM?
- Security best practices in IAM (Prácticas recomendadas de seguridad en IAM)
- Lógica de evaluación de políticas entre cuentas
- Cómo utilizar un ID externo al otorgar acceso a los recursos de AWS a terceros
- Collecting Information from AWS CloudFormation Resources Created in External Accounts with Custom Resources
- Securely Using External ID for Accessing AWS Accounts Owned by Others
- Extend IAM roles to workloads outside of IAM with IAM Roles Anywhere

Videos relacionados:

- How do I allow users or roles in a separate Cuenta de AWS access to my Cuenta de AWS?
- AWS re:Invent 2018: Become an IAM Policy Master in 60 Minutes or Less
- AWS Knowledge Center Live: IAM Best Practices and Design Decisions

Ejemplos relacionados:

- Configurar el acceso entre cuentas a Amazon DynamoDB
- AWS STS Network Query Tool

## Detección

Pregunta

- SEC 4. ¿Cómo se detectan e investigan los eventos de seguridad?

## SEC 4. ¿Cómo se detectan e investigan los eventos de seguridad?

Capture y analice los eventos a partir de registros y métricas para obtener una mejor visibilidad. Actúe ante los eventos de seguridad y las posibles amenazas para proteger las cargas de trabajo.

Prácticas recomendadas

- SEC04-BP01 Configuración del registro de servicios y aplicaciones
- SEC04-BP02 Recopilación de registros, resultados y métricas en ubicaciones estandarizadas
- SEC04-BP03 Correlación y enriquecimiento de las alertas de seguridad
- SEC04-BP04 Inicio de correcciones para recursos no conformes

### SEC04-BP01 Configuración del registro de servicios y aplicaciones

Retenga los registros de eventos de seguridad de servicios y aplicaciones. Se trata de un principio fundamental de seguridad en casos de uso de auditoría, investigación y uso operativo, y un requisito de seguridad común basado en las normas, políticas y procedimientos de gobernanza, riesgo y cumplimiento (GRC).

Resultado deseado: una organización debe ser capaz de recuperar de manera fiable y uniforme los registros de eventos de seguridad de los servicios y aplicaciones de AWS en el momento oportuno cuando sea necesario llevar a cabo algún proceso o cumplir una obligación interna, como una respuesta a un incidente de seguridad. Considere la posibilidad de centralizar los registros para obtener mejores resultados operativos.

Patrones comunes de uso no recomendados:

- Almacenar los registros de forma indefinida o eliminarlos demasiado pronto.
- Todo el mundo puede acceder a los registros.
- Depender por completo de procesos manuales para la gobernanza y el uso de los registros.
- Almacenar todos y cada uno de los tipos de registros por si fueran necesarios.
- Comprobar la integridad de los registros solo cuando es necesario.

Beneficios de establecer esta práctica recomendada: implemente un mecanismo de análisis de la causa raíz (RCA) para los incidentes de seguridad y una fuente de pruebas para cumplir sus obligaciones de gobernanza, riesgo y conformidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Durante una investigación de seguridad u otros casos de uso basados en sus requisitos, necesita poder revisar los registros correspondientes para registrar y comprender todo el alcance y la cronología del incidente. También necesita los registros para generar alertas que indican que se han producido determinadas acciones de interés. Es fundamental seleccionar, activar, almacenar y configurar mecanismos de consulta y recuperación, así como de alerta.

Pasos para la implementación

- Seleccione y utilice los orígenes de registros. Antes de una investigación de seguridad, necesita obtener los registros relevantes para reconstruir de forma retroactiva la actividad que se ha producido en una Cuenta de AWS. Seleccione las fuentes de registros relevantes para sus cargas de trabajo.

  Los criterios de selección de las fuentes de registros deben basarse en los casos de uso que requiera su negocio. Establezca un registro de seguimiento para cada Cuenta de AWS mediante AWS CloudTrail o un registro de seguimiento de AWS Organizations y, para ello, configure un bucket de Amazon S3.

  AWS CloudTrail es un servicio de registro que rastrea las llamadas a la API que se hacen en una Cuenta de AWS y captura la actividad de los servicios de AWS. Está activado de forma predeterminada con una retención de 90 días de los eventos de administración que se pueden recuperar a través del historial de eventos de CloudTrail mediante la Consola de administración de AWS, la AWS CLI o un SDK de AWS. Para prolongar la retención y la visibilidad de los eventos de datos, cree un registro de seguimiento de CloudTrail y asócielo a un bucket de Amazon S3 y, de forma opcional, a un grupo de registros de Amazon CloudWatch. Como alternativa, puede crear un CloudTrail Lake, que conserva los registros de CloudTrail durante un máximo de siete años y proporciona un servicio de consultas basado en SQL.

  AWS recomienda que los clientes que utilicen una VPC activen los registros de tráfico de red y DNS mediante los registros de flujo de VPC y los registros de consultas de Amazon Route 53 Resolver, respectivamente, y los transmitan a un bucket de Amazon S3 o a un grupo de registros de CloudWatch. Puede crear un registro de flujo de VPC para una VPC, una subred o una interfaz

de red. En el caso de los registros de flujo de VPC, puede elegir cómo y dónde utilizar los registros de flujo para reducir costos.

Los registros de AWS CloudTrail, los registros de flujo de VPC y los registros de consulta de Route 53 Resolver son los orígenes de registros básicos que facilitan las investigaciones de seguridad en AWS. También puede usar Amazon Security Lake para recopilar, normalizar y almacenar estos datos de registro en formato Apache Parquet y Open Cybersecurity Schema Framework (OCSF), que está listo para su consulta. Security Lake también admite otros registros de AWS y registros de orígenes de terceros.

Los servicios de AWS pueden generar registros que no capturan las fuentes de registros básicas, como los registros de Elastic Load Balancing, los registros de AWS WAF, los registros del registrador de AWS Config, los resultados de Amazon GuardDuty, los registros de auditoría de Amazon Elastic Kubernetes Service (Amazon EKS) y los registros del sistema operativo y las aplicaciones de las instancias de Amazon EC2. Para obtener una lista completa de las opciones de registro y supervisión, consulte el Apéndice A: Definiciones de las capacidades de la nube: registro y eventos de la Guía de respuesta ante incidentes de seguridad de AWS.

- Investigación de las capacidades de registro de cada servicio y aplicación de AWS: cada servicio y aplicación de AWS le ofrece opciones para el almacenamiento de registros, cada una de las cuales tiene sus propias capacidades de retención y ciclo de vida. Los dos servicios de almacenamiento de registros más comunes son Amazon Simple Storage Service (Amazon S3) y Amazon CloudWatch. Para periodos de retención largos, se recomienda utilizar Amazon S3 por su rentabilidad y la flexibilidad de sus ciclos de vida. Si la opción de registro principal es Registros de Amazon CloudWatch, quizá debería considerar la posibilidad de archivar los registros a los que se accede con menos frecuencia en Amazon S3.
- Selección del almacenamiento de registros: la elección del almacenamiento de registros suele estar relacionada con la herramienta de consulta que utilice, las capacidades de retención, la familiaridad y el costo. Las principales opciones para el almacenamiento de registros son un bucket de Amazon S3 o un grupo de registro de CloudWatch.

  Un bucket de Amazon S3 es un almacenamiento rentable y duradero que tiene una política de ciclo de vida opcional. Los registros almacenados en buckets de Amazon S3 pueden consultarse a través de servicios como Amazon Athena.

  Un grupo de registro de CloudWatch ofrece un almacenamiento duradero y una utilidad de consulta integrada a través de Información de registros de CloudWatch.

- Identificación de la retención de registros adecuada: cuando utilice un bucket de Amazon S3 o un grupo de registros de CloudWatch para almacenar registros, debe establecer ciclos de vida adecuados para cada fuente de registros a fin de optimizar los costos de almacenamiento y recuperación. Por lo general, los clientes tienen entre tres meses y un año de registros disponibles para su consulta, con un periodo de retención de hasta siete años. La elección de la disponibilidad y el periodo de retención debe ajustarse a sus requisitos de seguridad y a una combinación de requisitos legales, reglamentarios y empresariales.
- Uso del registro para cada servicio y aplicación de AWS con las políticas de retención y ciclo de vida adecuadas: para cada servicio o aplicación de AWS de su organización, busque la guía de configuración de registros específica:
  - Configuración de registros de seguimiento de AWS CloudTrail
  - Configuración de registros de flujo de VPC
  - Configuración de exportaciones de resultados de Amazon GuardDuty
  - Configuración de grabaciones de AWS Config
  - Configuración del tráfico de ACL web de AWS WAF
  - Configuración de registros del tráfico de red de AWS Network Firewall
  - Configuración de registros de acceso de Elastic Load Balancing
  - Configuración de registros de consultas de Amazon Route 53 Resolver
  - Configuración de registros de Amazon RDS
  - Configuración de registros de plano de control de Amazon EKS
  - Configuración del agente de Amazon CloudWatch para instancias de Amazon EC2 y servidores en las instalaciones
- Selección e implementación de mecanismos de consulta para los registros: para las consultas de registro, puede utilizar Información de registros de CloudWatch para los datos almacenados en los grupos de registros de CloudWatch, y Amazon Athena y Amazon OpenSearch Service para los datos almacenados en Amazon S3. También puede utilizar herramientas de consulta de terceros, como un servicio de administración de eventos e información de seguridad (SIEM).

  En el proceso de selección de una herramienta de consulta de registros, se deben tener en cuenta los aspectos relacionados con las personas, los procesos y la tecnología de sus operaciones de seguridad. Seleccione una herramienta que cumpla los requisitos operativos, empresariales y de seguridad, y que sea accesible y pueda mantenerse a largo plazo. Tenga en cuenta que las herramientas de consulta de registros funcionan de forma óptima cuando el número de registros

a analizar se mantiene dentro de los límites de la herramienta. No es raro disponer de varias herramientas de consulta debido a limitaciones técnicas o de costos.

Por ejemplo, podría utilizar una herramienta de administración de eventos e información de seguridad (SIEM) de terceros para hacer consultas en los últimos 90 días de datos, pero utilizar Athena para efectuar consultas anteriores a esos 90 días debido al costo de la ingestión de registros de un SIEM. Independientemente de cuál sea la implementación, compruebe que su enfoque permite reducir al mínimo el número de herramientas necesarias para maximizar la eficiencia operativa, especialmente durante la investigación de un evento de seguridad.

- Uso de registros para las alertas: AWS proporciona alertas a través de varios servicios de seguridad:
  - AWS Config supervisa y registra las configuraciones de los recursos de AWS y permite automatizar la evaluación y la corrección con las configuraciones deseadas.
  - Amazon GuardDuty es un servicio de detección de amenazas que supervisa continuamente cualquier actividad malintencionada y comportamiento no autorizado para proteger sus cargas de trabajo y sus Cuentas de AWS. GuardDuty ingiere, agrega y analiza la información de las fuentes, como los eventos de administración y datos de AWS CloudTrail, los registros de DNS, los registros de flujo de VPC y los registros de auditoría de Amazon EKS. GuardDuty extrae flujos de datos independientes directamente de CloudTrail, los registros de flujo de VPC, los registros de consultas de DNS y Amazon EKS. No es necesario que administre las políticas de los buckets de Amazon S3 ni que modifique la forma en que recopila y almacena los registros. Aun así, es recomendable que retenga estos registros para sus propios fines de investigación y conformidad.
  - AWS Security Hub CSPM proporciona un único lugar en el que se agregan, organizan y priorizan las alertas de seguridad o los resultados de varios servicios de AWS y productos opcionales de terceros para ofrecerle una vista completa de las alertas de seguridad y los estados de conformidad.

  También puede utilizar motores de generación de alertas personalizados para alertas de seguridad que no cubran estos servicios o para alertas específicas relevantes para su entorno. Para obtener información sobre cómo crear estas alertas y detecciones, consulte la sección Detección en la Guía de respuesta ante incidentes de seguridad de AWS.

Recursos

Prácticas recomendadas relacionadas:

- SEC04-BP02 Recopilación de registros, resultados y métricas en ubicaciones estandarizadas
- SEC07-BP04 Definición de la administración escalable del ciclo de vida de los datos
- SEC10-BP06 Implementación de las herramientas con anticipación

Documentos relacionados:

- AWS Security Incident Response Guide
- Cómo comenzar a utilizar Amazon Security Lake
- Getting started: Amazon CloudWatch Logs

Videos relacionados:

- AWS re:Invent 2022 - Introducing Amazon Security Lake

Ejemplos relacionados:

- Assisted Log Enabler for AWS
- AWS Security Hub CSPM Findings Historical Export

SEC04-BP02 Recopilación de registros, resultados y métricas en ubicaciones estandarizadas

Los equipos de seguridad se basan en los registros y los resultados para analizar aquellos eventos que podrían indicar una actividad no autorizada o cambios no intencionados. Para agilizar este análisis, recopile los registros de seguridad y los resultados en ubicaciones estandarizadas.  Esto hace que los puntos de datos de interés estén disponibles para su correlación y puede simplificar la integración de herramientas.

Resultado deseado: cuenta con un enfoque estandarizado para recopilar, analizar y visualizar los datos de registro, los resultados y las métricas. Los equipos de seguridad pueden correlacionar, analizar y visualizar de manera eficiente los datos de seguridad en sistemas dispares para descubrir posibles eventos de seguridad e identificar anomalías. Dispone de sistemas de gestión de información y eventos de seguridad (SIEM) u otros mecanismos integrados para consultar y analizar los datos de registro con el fin de responder, rastrear y escalar los eventos de seguridad sin demora.

Patrones comunes de uso no recomendados:

- Los equipos tienen y administran de forma independiente registros y recopilaciones de métricas que no se ajustan a la estrategia de registro de la organización.
- Los equipos no tienen controles de acceso adecuados para restringir la visibilidad y la alteración de los datos recopilados.
- Los equipos no gestionan sus registros, resultados y métricas de seguridad como parte de su política de clasificación de datos.
- Los equipos no tienen en cuenta los requisitos de soberanía y localización de los datos al configurar las recopilaciones de datos.

Beneficios de establecer esta práctica recomendada: una solución de registro estandarizada para recopilar y consultar los datos y eventos de registro mejora los conocimientos obtenidos a partir de la información que contienen. La configuración de un ciclo de vida automatizado para los datos de registro recopilados puede reducir los costos derivados del almacenamiento de registros. Puede crear un control de acceso detallado para la información de registro recopilada de acuerdo con el nivel de confidencialidad de los datos y los patrones de acceso que necesiten sus equipos. Puede integrar herramientas para correlacionar, visualizar y obtener información a partir de los datos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

El aumento del uso de AWS dentro de una organización se traduce en un número cada vez mayor de cargas de trabajo y entornos distribuidos. A medida que cada una de estas cargas de trabajo y entornos genera datos sobre la actividad que contienen, la captura y el almacenamiento de estos datos de forma local supone un desafío para las operaciones de seguridad. Los equipos de seguridad utilizan herramientas como los sistemas de gestión de información y eventos de seguridad (SIEM) para recopilar datos de fuentes distribuidas y llevar a cabo tareas de correlación, análisis y elaboración de flujos de trabajo de respuesta. Esto requiere administrar un conjunto complejo de permisos para acceder a los diversos orígenes de datos y una sobrecarga adicional para operar los procesos de extracción, transformación y carga (ETL).

Para superar estos desafíos, puede agregar todos los orígenes pertinentes de datos de registro de seguridad en una cuenta de archivo de registro, tal como se describe en Organizing Your AWS Environment Using Multiple Accounts. Esto incluye todos los datos relacionados con la seguridad de la carga de trabajo y los registros que generan los servicios de AWS, como AWS CloudTrail, AWS WAF, Elastic Load Balancing y Amazon Route 53. La recopilación de estos datos en ubicaciones estandarizadas de una Cuenta de AWS separada que cuente con los permisos multicuenta

adecuados tiene varios beneficios. Esta práctica ayuda a evitar la manipulación de registros en cargas de trabajo y entornos comprometidos, proporciona un punto de integración único para herramientas adicionales y ofrece un modelo más simplificado para configurar el ciclo de vida y la retención de datos.  Evalúe los impactos de la soberanía de los datos, los alcances de cumplimiento y otras normativas para determinar si se requieren varias ubicaciones de almacenamiento y periodos de retención de datos de seguridad.

Para facilitar la recopilación y estandarización de registros y resultados, valore la posibilidad de usar Amazon Security Lake en su cuenta de archivo de registro. Puede configurar Security Lake para que ingiera automáticamente datos de orígenes comunes como CloudTrail, Route 53, Amazon EKS y VPC Flow Logs. También puede configurar AWS Security Hub CSPM como origen de datos en Security Lake, lo que le permite correlacionar los resultados de otros servicios de AWS, como Amazon GuardDuty y Amazon Inspector, con sus datos de registro.  Del mismo modo, puede usar integraciones de orígenes de datos de terceros o configurar orígenes de datos personalizados. Todas las integraciones estandarizan sus datos en el formato Open Cybersecurity Schema Framework (OCSF) y se almacenan en buckets de Amazon S3 como archivos Parquet, lo que elimina la necesidad de procesamiento de extracción, transformación y carga (ETL).

El almacenamiento de los datos de seguridad en ubicaciones estandarizadas proporciona capacidades de análisis avanzadas. AWS recomienda implementar herramientas de análisis de seguridad que estén activas en un entorno de AWS en una cuenta de Security Tooling independiente de su cuenta de archivo de registros. Este enfoque le permite implementar controles exhaustivos para proteger la integridad y la disponibilidad de los registros y el proceso de administración de registros diferentes de las herramientas que se emplean para acceder a ellos.  Considere la posibilidad de usar servicios como Amazon Athena para ejecutar consultas bajo demanda que correlacionen varios orígenes de datos. También puede integrar herramientas de visualización como Quick. Cada vez hay más soluciones basadas en inteligencia artificial disponibles y pueden llevar a cabo funciones como traducir los resultados en resúmenes legibles por humanos o la interacción en lenguaje natural. Estas soluciones suelen integrarse más fácilmente al tener una ubicación de almacenamiento de datos estandarizada para efectuar consultas.

Pasos para la implementación

1. Cree las cuentas de archivo de registros y Security Tooling
    a. Mediante AWS Organizations, cree las cuentas de archivo de registros y Security Tooling en una unidad organizativa de seguridad. Si usa AWS Control Tower para administrar su organización, las cuentas de archivo de registros y Security Tooling se crean automáticamente.

Configure los roles y permisos para acceder a estas cuentas y administrarlas según sea necesario.

2. Configure las ubicaciones de datos de seguridad estandarizadas
   a. Determine su estrategia para crear ubicaciones de datos de seguridad estandarizadas. Puede hacerlo mediante opciones como los enfoques de arquitectura de lagos de datos comunes, productos de datos de terceros o Amazon Security Lake. AWS recomienda recopilar los datos de seguridad de Regiones de AWS que haya elegido para sus cuentas, incluso aunque no estén en uso activamente.
3. Configure la publicación de orígenes de datos en sus ubicaciones estandarizadas
   a. Identifique los orígenes de sus datos de seguridad y configúrelos para que se publiquen en sus ubicaciones estandarizadas. Evalúe las opciones para exportar automáticamente los datos en el formato deseado, en lugar de aquellas que requieran desarrollar los procesos de ETL. Con Amazon Security Lake, podrá recopilar datos de orígenes compatibles con AWS y sistemas integrados de terceros.
4. Configure las herramientas para acceder a sus ubicaciones estandarizadas
   a. Configure herramientas como Amazon Athena, Quick o soluciones de terceros para disponer del acceso necesario a las ubicaciones estandarizadas. Configure estas herramientas para que funcionen desde la cuenta de Security Tooling con acceso de lectura multicuenta a la cuenta de Log Archive, cuando corresponda. Cree suscriptores en Amazon Security Lake para que estas herramientas puedan acceder a sus datos.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP01 Separación de cargas de trabajo con cuentas
- SEC07-BP04 Definición de la administración escalable del ciclo de vida de los datos
- SEC08-BP04 Aplicación del control de acceso
- OPS08-BP02 Análisis de los registros de la carga de trabajo

Documentos relacionados:

- AWS Whitepapers: Organizing Your AWS Environment Using Multiple Accounts
- AWS Prescriptive Guidance: AWS Security Reference Architecture (AWS SRA)
- AWS Prescriptive Guidance: Logging and monitoring guide for application owners

Ejemplos relacionados:

- Agregación, búsqueda y visualización de datos de registro procedentes de fuentes distribuidas con Amazon Athena y Quick.
- Cómo visualizar los resultados de Amazon Security Lake con Quick
- Generate AI powered insights for Amazon Security Lake using Amazon SageMaker AI Studio and Amazon Bedrock
- Identify cybersecurity anomalies in your Amazon Security Lake data using Amazon SageMaker AI
- Ingest, transform, and deliver events published by Amazon Security Lake to Amazon OpenSearch Service
- Simplify AWS CloudTrail log analysis with natural language query generation in CloudTrail Lake

Herramientas relacionadas:

- Amazon Security Lake
- Integraciones de socios de Amazon Security Lake
- Open Cybersecurity Schema Framework (OCSF)
- Amazon Athena
- Quick
- Amazon Bedrock

SEC04-BP03 Correlación y enriquecimiento de las alertas de seguridad

La actividad inesperada puede generar múltiples alertas de seguridad de diferentes fuentes, lo que requiere una mayor correlación y enriquecimiento para comprender el contexto completo. Implemente la correlación y el enriquecimiento automatizados de las alertas de seguridad para ayudar a lograr una identificación y una respuesta a los incidentes más precisas.

Resultado deseado: los mecanismos automatizados correlacionan los datos y enriquecen dichos datos con información adicional a medida que la actividad genere diferentes alertas en sus cargas de trabajo y entornos. Este preprocesamiento ofrece una comprensión más detallada del evento, lo que ayuda a los investigadores a determinar la gravedad del evento y si constituye un incidente que requiere una respuesta formal. Este proceso reduce la carga para los equipos de supervisión e investigación.

Patrones comunes de uso no recomendados:

- Existen grupos de personas distintos que investigan los resultados y alertas generados por los diferentes sistemas, a menos que los requisitos de separación de funciones exijan lo contrario.
- Canalizar en la organización todos los datos de alertas y resultados de seguridad a ubicaciones estándar, pero con la necesidad de que los investigadores lleven a cabo una correlación y un enriquecimiento manuales.
- Confiar únicamente en la inteligencia de los sistemas de detección de amenazas para informar sobre los resultados y establecer el nivel de gravedad.

Beneficios de establecer esta práctica recomendada: la correlación y el enriquecimiento automatizados de las alertas ayudan a reducir la carga cognitiva general y la preparación manual de los datos que requieren los investigadores. Esta práctica puede reducir el tiempo necesario para determinar si el evento representa un incidente e iniciar una respuesta formal. El contexto adicional también ayuda a evaluar con precisión la verdadera gravedad de un evento, ya que puede ser mayor o menor de lo que sugiere una alerta.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Las alertas de seguridad pueden provenir de muchos orígenes diferentes en AWS, entre los que se encuentran:

- Servicios como Amazon GuardDuty, AWS Security Hub CSPM, Amazon Macie, Amazon Inspector, AWS Config, AWS Identity and Access Management Access Analyzer y Analizador de acceso a la red
- Alertas del análisis automatizado de los registros de aplicaciones, infraestructuras y servicios de AWS, como las de Security Analytics para Amazon OpenSearch Service.
- Alarmas que responden a cambios en su actividad de facturación procedentes de orígenes como Amazon CloudWatch, Amazon EventBridge o AWS Budgets.
- Orígenes de terceros, como orígenes de inteligencia sobre amenazas y Soluciones de socios de seguridad de la AWS Partner Network.
- Contactos de AWS Trust & Safety u otros orígenes, como clientes o empleados internos.
- Utilice el Catálogo de técnicas de amenazas (TTC) de AWS para ayudar a identificar y correlacionar el comportamiento de los actores de amenazas mediante la identificación del indicador de compromiso (IoC). El TTC es una extensión del marco MITRE ATT&CK, que clasifica

todos los comportamientos y técnicas de los actores de amenazas conocidos y observados dirigidos a los recursos de AWS.

En su forma más básica, las alertas contienen información sobre quién (la entidad principal o la identidad) está haciendo qué (la acción efectuada) a qué (los recursos afectados). Para cada uno de estos orígenes, identifique si hay formas de crear asignaciones entre identificadores para estas identidades, acciones y recursos como base para llevar a cabo la correlación. Esto puede consistir en integrar las fuentes de las alertas con una herramienta de administración de eventos e información de seguridad (SIEM) para llevar a cabo una correlación automática en su nombre, crear sus propios procesos y canalizaciones de datos, o una combinación de ambas estrategias.

Un ejemplo de servicio que puede efectuar la correlación es Amazon Detective. Detective ingiere continuamente alertas de diversos orígenes de AWS y de terceros, y utiliza diferentes formas de inteligencia para crear un gráfico visual de sus relaciones con el fin de asistir en las investigaciones.

Si bien el nivel de gravedad inicial de una alerta ayuda a establecer prioridades, el contexto en el que se haya producido la alerta determina su verdadero nivel de gravedad. Por ejemplo, Amazon GuardDuty puede avisar de que una instancia de Amazon EC2 de su carga de trabajo está consultando un nombre de dominio inesperado. GuardDuty puede asignar una gravedad baja a esta alerta. Sin embargo, la correlación automatizada con otras actividades en el momento de la alerta podría revelar que varios cientos de instancias de EC2 se han implementado con la misma identidad, lo que aumenta los costos operativos generales. En este caso, este contexto de eventos correlacionados justificaría una nueva alerta de seguridad y el nivel de gravedad se ajustaría en alto, lo que aceleraría las acciones futuras.

Pasos para la implementación

1. Identifique los orígenes de la información de alertas de seguridad. Comprenda en qué medida las alertas de estos sistemas representan la identidad, la acción y los recursos para determinar dónde se puede establecer una correlación.
2. Establezca un mecanismo para capturar las alertas de diferentes orígenes. Para ello, considere la posibilidad de utilizar servicios como Security Hub CSPM, EventBridge y CloudWatch.
3. Identifique los orígenes para la correlación y el enriquecimiento de los datos. Entre las fuentes de ejemplo se incluyen AWS CloudTrail, los registros de flujo de VPC, los registros de Route 53 Resolver y los registros de infraestructura y aplicaciones. Algunos o todos estos registros pueden consumirse mediante una única integración con Amazon Security Lake.

4. Integre sus alertas con sus fuentes de correlación y enriquecimiento de datos para crear contextos de eventos de seguridad más detallados y establecer el nivel de gravedad.
   a. Amazon Detective, las herramientas de SIEM u otras soluciones de terceros pueden llevar a cabo un cierto nivel de ingesta, correlación y enriquecimiento automáticamente.
   b. También puede usar los servicios de AWS para crear sus propias soluciones. Por ejemplo, puede invocar una función de AWS Lambda para ejecutar una consulta de Amazon Athena a AWS CloudTrail o Amazon Security Lake y publicar los resultados en EventBridge.

Recursos

Prácticas recomendadas relacionadas:

- SEC10-BP03 Preparación de las capacidades forenses
- OPS08-BP04 Creación de alertas procesables
- REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)

Documentos relacionados:

- AWS Security Incident Response Guide

Ejemplos relacionados:

- How to enrich AWS Security Hub CSPM findings with account metadata

Herramientas relacionadas:

- Amazon Detective
- Amazon EventBridge
- AWS Lambda
- Amazon Athena

SEC04-BP04 Inicio de correcciones para recursos no conformes

Sus controles de detección pueden alertarle sobre la presencia de recursos no conformes con sus requisitos de configuración. Puede iniciar soluciones definidas mediante programación, de forma

manual o automática, para corregir estos recursos y ayudar a minimizar los posibles impactos. Al definir las correcciones mediante programación, puede tomar medidas rápidas y coherentes.

Si bien la automatización puede mejorar las operaciones de seguridad, debe implementarla y administrarla con cuidado. Establezca mecanismos de supervisión y control adecuados para verificar que las respuestas automatizadas sean eficaces, precisas y estén alineadas con las políticas de la organización y la propensión al riesgo.

Resultado deseado: defina los estándares de configuración de los recursos junto con los pasos para corregir las situaciones en las que se detecte que los recursos no cumplen los requisitos. Cuando sea posible, defina las medidas de corrección mediante programación para que puedan iniciarse de forma manual o automática. Dispone de sistemas de detección para identificar los recursos disconformes y publica alertas en herramientas centralizadas y supervisadas por su personal de seguridad. Usa estas herramientas para ejecutar las correcciones programáticas, de forma manual o automática. Dispone de mecanismos de supervisión y control adecuados en las correcciones automáticas para regular su uso.

Patrones comunes de uso no recomendados:

- Implementar la automatización, pero no verificar ni validar minuciosamente las acciones de corrección. Esto puede tener consecuencias imprevistas, como obstaculizar las operaciones empresariales legítimas o provocar inestabilidad en el sistema.
- Mejorar los tiempos de respuesta y los procedimientos mediante la automatización, pero sin contar con la supervisión y los mecanismos adecuados que permitan la intervención y la decisión de un humano en los casos necesarios.
- Confiar únicamente en las correcciones, en lugar de incluirlas como parte de un programa más amplio de respuesta y recuperación ante incidentes.

Beneficios de establecer esta práctica recomendada: las correcciones automáticas pueden responder a los errores de configuración con mayor rapidez que los procesos manuales, lo que ayuda a minimizar los posibles impactos empresariales, así como a reducir las oportunidades de usos no previstos. Cuando define las correcciones mediante programación, se aplican de forma coherente, lo que reduce el riesgo de error humano. La automatización también puede gestionar un mayor volumen de alertas simultáneamente, lo que resulta particularmente importante en entornos que funcionan a gran escala.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Tal como se describe en SEC01-BP03 Identificación y validación de los objetivos de control, servicios como AWS Config y AWS Security Hub CSPM pueden ayudarlo a supervisar la configuración de los recursos de sus cuentas para garantizar el cumplimiento de sus requisitos. Cuando se detectan recursos no conformes, servicios como AWS Security Hub CSPM pueden ayudar a enrutar las alertas de manera adecuada y a corregirlas. Estas soluciones proporcionan un punto central en el que los investigadores de seguridad puedan supervisar los problemas y tomar medidas correctivas.

Además de AWS Security Hub CSPM, AWS presentó Security Hub Advanced. Este servicio, anunciado en re:Invent 2025, transforma la forma en que las organizaciones priorizan sus problemas de seguridad más críticos y responden a gran escala para proteger sus entornos de nube. Security Hub mejorado ahora utiliza análisis avanzados para correlacionar, enriquecer y priorizar automáticamente las señales de seguridad en todo su entorno de nube. Security Hub se integra perfectamente con Amazon GuardDuty, Amazon Inspector, Amazon Macie y AWS Security Hub CSPM. Los resultados correlacionados en Security Hub pueden dar lugar a un resultado totalmente nuevo, denominado resultado de exposición, que incluye una ruta de ataque supuesta basada en las vulnerabilidades encontradas en cada recurso.

Si bien algunas situaciones de recursos disconformes son únicas y requieren de juicio humano para corregirlas, otras situaciones requieren una respuesta estándar que se puede definir mediante programación. Por ejemplo, una respuesta estándar ante un error de configuración de un grupo de seguridad de VPC podría consistir en eliminar las reglas no permitidas y notificárselo al propietario. Las respuestas se pueden definir en funciones de AWS Lambda, documentos de Automatización de AWS Systems Manager o mediante otros entornos de código que prefiera. Asegúrese de que el entorno pueda autenticarse en AWS con un rol de IAM que tenga la cantidad mínima de permisos necesaria para tomar medidas correctivas.

Una vez que haya definido la corrección deseada, podrá determinar el medio que prefiera para iniciarla. AWS Config puede iniciar las correcciones. Si utiliza Security Hub CSPM, puede hacerlo mediante acciones personalizadas, que publican la información de búsqueda en Amazon EventBridge. A continuación, una regla de EventBridge puede iniciar la corrección. Puede configurar las correcciones a través de Security Hub CSPM para que se ejecuten de forma automática o manual.

En el caso de corrección mediante programación, le recomendamos que disponga de registros y auditorías exhaustivos de las medidas adoptadas, así como de sus resultados. Revise y analice estos registros para evaluar la eficacia de los procesos automatizados e identificar las áreas

de mejora. Capture los registros en Registros de Amazon CloudWatch y los resultados de las correcciones como notas de resultados en Security Hub CSPM.

Como punto de partida, considere la posibilidad de utilizar la Respuesta de seguridad automatizada en AWS, que cuenta con soluciones de corrección prediseñadas para resolver los errores de configuración de seguridad más comunes.

Pasos para la implementación

1. Analice y priorice las alertas.
    a. Unifique las alertas de seguridad de varios servicios de AWS en Security Hub CSPM para obtener visibilidad, priorización y corrección centralizadas.
2. Desarrolle medidas de corrección.
    a. Utilice servicios como Systems Manager y AWS Lambda para ejecutar correcciones programáticas.
3. Configure cómo se inician las correcciones.
    a. Con Systems Manager, defina acciones personalizadas para publicar los resultados en EventBridge. Configure estas acciones para que se inicien manual o automáticamente.
    b. También puede usar Amazon Simple Notification Service (SNS) para enviar notificaciones y alertas a las partes interesadas pertinentes (como el equipo de seguridad o los equipos de respuesta a incidentes) para que intervengan manualmente o escalen el problema si es necesario.
4. Revise y analice los registros de corrección para comprobar su eficacia y mejora.
    a. Envíe la salida del registro a Registros de CloudWatch. Capture los resultados como notas de resultados en Security Hub CSPM.

Recursos

Prácticas recomendadas relacionadas:

- SEC06-BP03 Reducción de la administración manual y el acceso interactivo

Documentos relacionados:

- AWS Security Incident Response Guide - Detection

Ejemplos relacionados:

- Respuesta de seguridad automatizada en AWS
- Monitor EC2 instance key pairs using AWS Config
- Create AWS Config custom rules by using AWS CloudFormation Guard policies
- Automatically remediate unencrypted Amazon RDS DB instances and clusters

Herramientas relacionadas:

- AWS Systems Manager Automation
- Respuesta de seguridad automatizada en AWS

## Protección de la infraestructura

Preguntas

- SEC 5. ¿Cómo protege los recursos de su red?
- SEC 6. ¿Cómo protege sus recursos de computación?

### SEC 5. ¿Cómo protege los recursos de su red?

Cualquier carga de trabajo que tenga forma de conexión de red, ya sea internet o una red privada, requiere varias capas de defensa para protegerse de amenazas internas y externas basadas en la red.

Prácticas recomendadas

- SEC05-BP01 Creación de capas de red
- SEC05-BP02 Control del flujo de tráfico dentro de las capas de red
- SEC05-BP03 Implementación de una protección basada en la inspección
- SEC05-BP04 Automatización de la protección de la red

#### SEC05-BP01 Creación de capas de red

Segmente la topología de red en diferentes capas en función de las agrupaciones lógicas de los componentes de la carga de trabajo según sus requisitos de acceso y la confidencialidad de los datos. Distinga entre los componentes que requieren acceso entrante desde Internet, como los

puntos de conexión web públicos, y aquellos que solo necesitan acceso interno, como las bases de datos.

Resultado deseado: incorporar las capas de su red en un enfoque de seguridad integral de defensa en profundidad que complemente la estrategia de autenticación y autorización de identidad de sus cargas de trabajo. Dispone de las capas de acuerdo con los requisitos de acceso y confidencialidad de los datos, con los mecanismos de control y flujo de tráfico adecuados.

Patrones comunes de uso no recomendados:

- Crea todos los recursos en una única VPC o subred.
- Desarrolla las capas de red sin tener en cuenta los requisitos de confidencialidad de los datos, el comportamiento de los componentes o la funcionalidad.
- Utiliza las VPC y las subredes de forma predeterminada para todas las consideraciones sobre las capas de red y no tener en cuenta la forma en que los servicios administrados de AWS influyen en la topología.

Beneficios de establecer esta práctica recomendada: establecer las capas de red es el primer paso para restringir las rutas innecesarias a través de la red, sobre todo las que conducen a sistemas y datos críticos. Esto dificulta que los actores no autorizados accedan a su red y a los recursos adicionales que contiene. Las capas de red diferenciadas reducen de forma ventajosa el alcance del análisis de los sistemas de inspección, como la detección de intrusos o la prevención del malware. Esto reduce la posibilidad de que se produzcan falsos positivos y disminuye la sobrecarga de procesamiento innecesaria.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Al diseñar una arquitectura de carga de trabajo, es habitual separar los componentes en diferentes capas en función de su responsabilidad. Por ejemplo, una aplicación web puede tener una capa de presentación, una capa de aplicación y una capa de datos. Es posible adoptar un enfoque similar al diseñar su topología de la red. Los controles de red subyacentes pueden ayudarle a hacer cumplir los requisitos de acceso a los datos de su carga de trabajo. Por ejemplo, en una arquitectura de aplicaciones web de tres niveles, puede almacenar los archivos de la capa de presentación estática en Amazon S3 y distribuirlos desde una red de entrega de contenido (CDN), como Amazon CloudFront. La capa de aplicaciones puede tener puntos de conexión públicos a los que presta servicio un equilibrador de carga de aplicación (ALB) en una subred pública de Amazon VPC (similar

a una zona desmilitarizada o DMZ), con servicios de backend implementados en subredes privadas. La capa de datos, en la que se alojan recursos como bases de datos y sistemas de archivos compartidos, puede encontrarse en subredes privadas diferentes de aquellas en las que están los recursos de la capa de aplicación. Puede implementar controles dentro de cada uno de estos límites de capa (CDN, subred pública, subred privada) para que solo los atraviese el tráfico autorizado.

Debe tener también en cuenta el nivel de confidencialidad de los datos que se vayan a procesar, de forma similar a cuando se modelan las capas de red en función del propósito funcional de los componentes de la carga de trabajo. Según el ejemplo de la aplicación web, si bien todos los servicios de carga de trabajo podrían encontrarse en la capa de aplicación, los distintos servicios podrían procesar datos con niveles de confidencialidad diferentes. En este caso, puede ser conveniente dividir la capa de aplicación con varias subredes privadas, distintas VPC en la misma Cuenta de AWS o incluso distintas VPC en Cuentas de AWS diferentes para cada nivel de confidencialidad de los datos, en función de los requisitos de control.

Otra cuestión que debe plantearse para las capas de red es la coherencia del comportamiento de los componentes de la carga de trabajo. En ese mismo ejemplo, es posible que en la capa de aplicación haya servicios que acepten entradas de usuarios finales o integraciones de sistemas externos que planteen intrínsecamente más riesgos que las entradas procedentes de otros servicios. Entre algunos ejemplos de esta situación podemos citar la carga de archivos, la ejecución de scripts de código, el análisis de correo electrónico, etc. Al ubicar estos servicios en su propia capa de red se contribuye a crear un límite de aislamiento más sólido en torno a ellos, y esto permite evitar que su comportamiento peculiar genere alertas por falsos positivos en los sistemas de inspección.

Como parte del diseño, tenga en cuenta cómo el uso de AWS Managed Services influye en la topología de la red. Descubra de qué manera servicios como Amazon VPC Lattice pueden ayudar a facilitar la interoperabilidad de los componentes de la carga de trabajo entre las capas de la red. Al usar AWS Lambda, implemente en sus subredes de VPC, a menos que existan motivos específicos para no hacerlo. Determine en qué casos pueden los puntos de conexión de VPC y AWS PrivateLink simplificar el cumplimiento de las políticas de seguridad que limitan el acceso a las puertas de enlace de Internet.

Pasos para la implementación

1. Revise la arquitectura de su carga de trabajo. Agrupe de forma lógica los componentes y servicios según las funciones que cumplen, la confidencialidad de los datos que procesan y su comportamiento.

2. Para aquellos componentes que respondan a solicitudes de Internet, plantéese la posibilidad de usar equilibradores de carga u otros proxies para proporcionar puntos de conexión públicos. Valore la posibilidad de cambiar los controles de seguridad mediante el uso de servicios administrados, como CloudFront, Amazon API Gateway, Elastic Load Balancing y AWS Amplify para alojar puntos de conexión públicos.
3. Para los componentes que se ejecutan en entornos de computación, como instancias de Amazon EC2, contenedores de AWS Fargate o funciones de Lambda, lleve a cabo la implementación en subredes privadas en función de sus grupos del primer paso.
4. Para los servicios de AWS totalmente administrados, como Amazon DynamoDB, Amazon Kinesis o Amazon SQS, considere la posibilidad de utilizar puntos de conexión de VPC como valores predeterminados para el acceso a través de direcciones IP privadas.

Recursos

Prácticas recomendadas relacionadas:

- REL02 Planificación de la topología de la red
- PERF04-BP01 Comprensión del efecto de las redes en el rendimiento

Videos relacionados:

- AWS re:Invent 2023 - AWS networking foundations

Ejemplos relacionados:

- Ejemplos de VPC
- Acceder a las aplicaciones en contenedores de forma privada en Amazon ECS mediante AWS Fargate, un AWS PrivateLink y un equilibrador de carga de red
- Serve static content in an Amazon S3 bucket through a VPC by using Amazon CloudFront

SEC05-BP02 Control del flujo de tráfico dentro de las capas de red

Dentro de las capas de su red, utilice una mayor segmentación para restringir el tráfico únicamente a los flujos necesarios para cada carga de trabajo. En primer lugar, céntrese en controlar el tráfico entre Internet u otros sistemas externos a una carga de trabajo y su entorno (tráfico norte-sur). A continuación, observe los flujos entre los diferentes componentes y sistemas (tráfico de este a oeste).

Resultado deseado: permite que solo los flujos de red necesarios para que los componentes de sus cargas de trabajo se comuniquen entre sí, con sus clientes y con cualquier otro servicio del que dependan. Tiene en cuenta en su diseño cuestiones como la entrada y salida públicas en comparación con las privadas, la clasificación de datos, las normativas regionales y los requisitos de protocolo. Siempre que sea posible, prefiere los flujos punto a punto en lugar de la interconexión de redes como parte del diseño con el principio de privilegios mínimos.

Patrones comunes de uso no recomendados:

- Adoptar un enfoque de seguridad de red basado en el perímetro y controlar solamente el flujo de tráfico dentro de los límites de las capas de red.
- Dar por sentado que todo el tráfico dentro de una capa de red está autenticado y autorizado.
- Aplicar controles para el tráfico de entrada o de salida, pero no para ambos.
- Confiar solo en los componentes de la carga de trabajo y los controles de red para autenticar y autorizar el tráfico.

Beneficios de establecer esta práctica recomendada: esta práctica ayuda a reducir el riesgo de movimientos no autorizados dentro de la red y agrega un nivel adicional de autorización a sus cargas de trabajo. Al controlar el flujo de tráfico, puede restringir el alcance del impacto de un incidente de seguridad y acelerar la detección y la respuesta.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Si bien las capas de red ayudan a establecer límites en torno a los componentes de la carga de trabajo similares en cuanto a función, nivel de confidencialidad de los datos y comportamiento, puede crear un nivel de control del tráfico mucho más detallado mediante el uso de técnicas para segmentar aún más los componentes de estas capas según el principio de privilegio mínimo. En AWS, las capas de red se definen principalmente mediante subredes según los rangos de direcciones IP dentro de una Amazon VPC. Las capas también se pueden definir mediante diferentes VPC, por ejemplo, para agrupar entornos de microservicios por dominio empresarial. Cuando use varias VPC, intervenga en el enrutamiento mediante una AWS Transit Gateway. Si bien esto permite controlar el tráfico en el nivel de capa 4 (direcciones IP e intervalos de puertos) mediante grupos de seguridad y tablas de enrutamiento, puede obtener un mayor control mediante servicios adicionales como AWS PrivateLink, Amazon Route 53 Resolver DNS Firewall, AWS Network Firewall y AWS WAF.

Determine y haga un inventario de los requisitos de flujo de datos y comunicación de sus cargas de trabajo que incluya las entidades que inician la conexión, los puertos, los protocolos y las capas de red. Evalúe los protocolos disponibles para establecer conexiones y transmitir datos con el objetivo de seleccionar los que cumplan sus requisitos de protección (por ejemplo, HTTPS en lugar de HTTP). Capture estos requisitos tanto en los límites de sus redes como dentro de cada capa. Una vez identificados estos requisitos, explore las opciones para permitir solamente el tráfico requerido en cada punto de conexión. Un buen punto de partida es utilizar grupos de seguridad dentro de la VPC, ya que se pueden asociar a recursos que utilizan una interfaz de red elástica (ENI), como las instancias de Amazon EC2, las tareas de Amazon ECS, los pods de Amazon EKS o las bases de datos de Amazon RDS. A diferencia de un firewall de capa 4, un grupo de seguridad puede tener una regla que permita el tráfico de otro grupo de seguridad mediante su identificador, lo que reduce al mínimo las actualizaciones a medida que los recursos del grupo cambian con el tiempo. También puede filtrar el tráfico con reglas entrantes y salientes mediante grupos de seguridad.

Cuando el tráfico fluye entre las VPC, es habitual utilizar el emparejamiento de VPC para el enrutamiento sencillo o AWS Transit Gateway para el enrutamiento complejo. Con estos enfoques, se facilitan los flujos de tráfico entre el rango de direcciones IP de las redes de origen y destino. Sin embargo, si su carga de trabajo solo requiere flujos de tráfico entre componentes específicos de diferentes VPC, considere la posibilidad de utilizar una conexión punto a punto mediante AWS PrivateLink. Para ello, identifique qué servicio debe actuar como productor y cuál debe actuar como consumidor. Implemente un equilibrador de carga compatible para el productor, active PrivateLink en consecuencia y, a continuación, acepte una solicitud de conexión del consumidor. A continuación, se asignará al servicio del productor una dirección IP privada de la VPC del consumidor que el consumidor podrá usar para efectuar solicitudes posteriores. Este enfoque reduce la necesidad de emparejar las redes. Incluya los costos del procesamiento de datos y el equilibrio de carga como parte de la evaluación de PrivateLink.

Si bien los grupos de seguridad y PrivateLink ayudan a controlar el flujo entre los componentes de sus cargas de trabajo, otro aspecto importante que debe tener en cuenta es cómo controlar los dominios de DNS a los que pueden acceder sus recursos (si los hay). En función de la configuración de DHCP de sus VPC, puede optar por dos servicios de AWS para este fin. La mayoría de los clientes utilizan el servicio de DNS predeterminado de Route 53 Resolver (también denominado servidor DNS de Amazon o AmazonProvideDDNS) disponible para las VPC en la dirección +2 de su rango de CIDR. Con este enfoque, puede crear reglas de firewall de DNS y asociarlas a su VPC para determinar qué acciones tomar para las listas de dominios que proporcione.

Si no usa el servicio Route 53 Resolver o si desea complementarlo con capacidades de inspección y control de flujo más profundas que vayan más allá del filtrado de dominios, considere la posibilidad

de implementar un AWS Network Firewall. Este servicio inspecciona los paquetes individuales mediante reglas sin estado o con estado para determinar si se debe denegar o permitir el tráfico. Puede adoptar un enfoque similar para filtrar el tráfico web entrante a sus puntos de enlace públicos con AWS WAF. Para obtener más información sobre estos servicios, consulte SEC05-BP03 Implementación de una protección basada en la inspección.

Pasos para la implementación

1. Identifique los flujos de datos necesarios entre los componentes de sus cargas de trabajo.
2. Aplique múltiples controles con un enfoque de defensa en profundidad tanto para el tráfico entrante como para el saliente, incluido el uso de grupos de seguridad y tablas de enrutamiento.
3. Utilice firewalls para definir un control detallado del tráfico de red entrante, saliente y que pase por sus VPC, como el Firewall DNS de Route 53 Resolver, AWS Network Firewall y AWS WAF. Considere la posibilidad de usar AWS Firewall Manager para configurar y administrar de forma centralizada las reglas de firewall en toda la organización.

Recursos

Prácticas recomendadas relacionadas:

- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- SEC09-BP02 Aplicación del cifrado en tránsito

Documentos relacionados:

- Prácticas recomendadas de seguridad de la VPC
- AWS Network Optimization Tips
- Guidance for Network Security on AWS
- Secure your VPC's outbound network traffic in the Nube de AWS

Herramientas relacionadas:

- AWS Firewall Manager

Videos relacionados:

- AWS Transit Gateway reference architectures for many VPCs

- Application Acceleration and Protection with Amazon CloudFront, AWS WAF, and AWS Shield
- AWS re:Inforce 2023: Firewalls and where to put them

SEC05-BP03 Implementación de una protección basada en la inspección

Configure puntos de inspección del tráfico entre las capas de la red para asegurarse de que los datos en tránsito coincidan con las categorías y los patrones esperados. Analice los flujos de tráfico, los metadatos y los patrones para ayudar a identificar y detectar los eventos y responder a ellos de manera más eficaz.

Resultado deseado: se inspecciona y se autoriza el tráfico que atraviesa las capas de la red. Basa las decisiones de permiso o denegación en reglas explícitas, información sobre amenazas y desviaciones de los comportamientos de referencia. Las protecciones son más estrictas a medida que el tráfico se acerca a los datos confidenciales.

Patrones comunes de uso no recomendados:

- Confiar únicamente en las reglas de firewall basadas en puertos y protocolos. No aprovechar los sistemas inteligentes.
- Crear reglas de firewall sobre la base de patrones de amenazas actuales específicos sujetos a cambios.
- Inspeccionar únicamente el tráfico cuando fluye de subredes privadas a públicas, o de subredes públicas a Internet.
- No tener una visión básica del tráfico de la red para comparar las anomalías de comportamiento.

Beneficios de establecer esta práctica recomendada: los sistemas de inspección permiten crear reglas inteligentes, como permitir o denegar el tráfico únicamente cuando existan ciertas condiciones en los datos del tráfico. Beneficiarse de los conjuntos de reglas administradas de AWS y nuestros socios, sobre la base de la inteligencia de amenazas más reciente, a medida que el panorama de amenazas cambia con el tiempo. Esto reduce los gastos administrativos que supone mantener las reglas e investigar los indicadores de situaciones de riesgo, lo que reduce la posibilidad de que se produzcan falsos positivos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Controle minuciosamente el tráfico de red con y sin estado mediante AWS Network Firewall, otros firewalls y otros sistemas de prevención de intrusiones (IPS) en AWS Marketplace, que puede implementar detrás de un equilibrador de carga de puerta de enlace (GWLB). AWS Network Firewall es compatible con las especificaciones de IPS de código abierto compatibles con Suricata para proteger su carga de trabajo.

Tanto AWS Network Firewall como las soluciones de otros proveedores que utilizan un GWLB admiten diferentes modelos de implementación de inspecciones en línea. Por ejemplo, puede llevar a cabo la inspección en cada VPC, centralizarla en una VPC de inspección o implementarla en un modelo híbrido en el que el tráfico este-oeste fluya a través de una VPC de inspección y las entradas a Internet se inspeccionen en cada VPC. Otra consideración es si la solución admite desempaquetar la seguridad de la capa de transporte (TLS), lo que permite una inspección profunda de los paquetes en busca de flujos de tráfico iniciados en cualquier dirección. Para obtener más información y detalles en profundidad sobre estas configuraciones, consulte la guía AWS Network Firewall Best Practice.

Si utiliza soluciones que inspeccionan fuera de banda, como el análisis pcap de datos de paquetes de las interfaces de red que funcionan en modo promiscuo, puede configurar el reflejo del tráfico de la VPC. El tráfico reflejado se incluye en el ancho de banda disponible de sus interfaces y está sujeto a los mismos cargos por transferencia de datos que el tráfico no reflejado. Puede comprobar si hay versiones virtuales de estos dispositivos disponibles en AWS Marketplace, que podrían admitir la implementación en línea detrás de un GWLB.

En el caso de los componentes que llevan a cabo transacciones con protocolos basados en HTTP, proteja su aplicación ante las amenazas comunes con un firewall de aplicaciones web (WAF). AWS WAF es un firewall de aplicaciones web que le permite supervisar y bloquear las solicitudes HTTP(S) que coincidan con sus reglas configurables antes de enviarlas a Amazon API Gateway, Amazon CloudFront, AWS AppSync o un equilibrador de carga de aplicación. Piense en la posibilidad de llevar a cabo una inspección de paquetes en profundidad cuando evalúe la implementación del firewall de su aplicación web, ya que algunos requieren que finalice la seguridad de la capa de transporte (TLS) antes de la inspección del tráfico. Para empezar con AWS WAF, puede usar Reglas administradas de AWS junto con sus propias integraciones de socios o utilizar las existentes.

Puede administrar de forma centralizada AWS WAF, AWS Shield Advanced, AWS Network Firewall y los grupos de seguridad de Amazon VPC en toda su organización de AWS con AWS Firewall Manager.

Pasos para la implementación

1. Determine si puede aplicar las reglas de inspección de manera amplia (por ejemplo, mediante una VPC de inspección) o si necesita un enfoque más detallado por cada VPC.
2. Para soluciones de inspección en línea:
   a. Si utiliza AWS Network Firewall, cree reglas, políticas de firewall y el propio firewall. Una vez configurado esto, puede redirigir el tráfico al punto de conexión del firewall para facilitar la inspección.
   b. Si utiliza un dispositivo de terceros con un equilibrador de carga de puerta de enlace (GWLB), implemente y configure su dispositivo en una o más zonas de disponibilidad. A continuación, cree su GWLB, el servicio de punto de conexión y el punto de conexión, y configure el enrutamiento para su tráfico.
3. Para soluciones de inspección fuera de banda:
   1. Active el reflejo del tráfico de VPC en las interfaces en las que se deba reflejar el tráfico entrante y saliente. Puede usar reglas de Amazon EventBridge para invocar una función de AWS Lambda con el fin de activar el reflejo del tráfico en las interfaces cuando se creen nuevos recursos. Dirija las sesiones de reflejo del tráfico al equilibrador de carga de red situado frente al dispositivo que procese el tráfico.
4. Para soluciones de tráfico web entrante:
   a. Para configurar AWS WAF, comience por configurar una lista de control de acceso web (ACL web). La ACL web es una colección de reglas con una acción predeterminada procesada en serie (PERMITIR o DENEGAR) que define la forma en que gestiona el tráfico el WAF. Puede crear sus propias reglas y grupos o usar grupos de reglas administradas de AWS en su ACL web.
   b. Una vez configurada la ACL web, asocie la ACL web a un recurso de AWS (como un equilibrador de carga de aplicación, API Gateway, una API de REST o una distribución de CloudFront) para comenzar a proteger el tráfico web.

Recursos

Documentos relacionados:

- What is Traffic Mirroring?
- Implementing inline traffic inspection using third-party security appliances
- AWS Network Firewall example architectures with routing

- Centralized inspection architecture with AWS Gateway Load Balancer and AWS Transit Gateway

Ejemplos relacionados:

- Prácticas recomendadas para implementar el Equilibrador de carga de puerta de enlace
- TLS inspection configuration for encrypted egress traffic and AWS Network Firewall

Herramientas relacionadas:

- AWS Marketplace IDS/IPS

SEC05-BP04 Automatización de la protección de la red

Automatice la implementación de las protecciones de su red mediante prácticas de DevOps, como la infraestructura como código (IaC) y las canalizaciones de CI/CD. Estas prácticas pueden ayudarle a hacer un seguimiento de los cambios en las protecciones de la red a través de un sistema de control de versiones, a reducir el tiempo necesario para implementar los cambios y a detectar si las protecciones de la red se desvían de la configuración deseada.

Resultado deseado: defina las protecciones de red con plantillas y confirmarlas en un sistema de control de versiones. Las canalizaciones automatizadas se inician cuando se hacen nuevos cambios que sirvan para organizar sus pruebas e implementación. Dispone de comprobaciones de políticas y otras pruebas estáticas para validar los cambios antes de la implementación. Implementa los cambios en un entorno provisional para validar que los controles funcionen según lo esperado. También implementa en sus entornos de producción automáticamente una vez que se aprueben los controles.

Patrones comunes de uso no recomendados:

- Confiar en que los equipos de cada carga de trabajo definan individualmente toda su pila de red, sus protecciones y sus automatizaciones. No publicar los aspectos estándares de la pila de red y las protecciones de forma centralizada para que los consuman los equipos de carga de trabajo.
- Confiar en un equipo de red central para definir todos los aspectos de la red, las protecciones y las automatizaciones. No delegar los aspectos específicos de la carga de trabajo de la pila de red y las protecciones al equipo de esa carga de trabajo.
- Lograr el equilibrio adecuado entre la centralización y la delegación entre un equipo de red y los equipos de las cargas de trabajo, pero no aplicar estándares de prueba e implementación

uniformes en todas las plantillas de IaC y las canalizaciones de CI/CD. No recoger las configuraciones requeridas en herramientas que comprueben si las plantillas se ajustan a dichas configuraciones.

Beneficios de establecer esta práctica recomendada: el uso de plantillas para definir las protecciones de la red le permite hacer un seguimiento y comparar los cambios a lo largo del tiempo con un sistema de control de versiones. El uso de la automatización para probar e implementar los cambios crea estandarización y previsibilidad, lo que aumenta las posibilidades de que la implementación tenga éxito y reduce las configuraciones manuales repetitivas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Varios controles de protección de la red que se describen en SEC05-BP02 Control del flujo de tráfico dentro de las capas de red y SEC05-BP03 Implementación de una protección basada en la inspección incluyen sistemas de reglas administradas que pueden actualizarse automáticamente en función de la información sobre amenazas más reciente. Entre los ejemplos de protección de los puntos de enlace web se incluyen las reglas administradas de AWS WAF y la mitigación de DDoS automática en la capa de aplicaciones de AWS Shield Advanced. Utilice los grupos de reglas administradas de AWS Network Firewall para mantenerse al día de las listas de dominios de baja reputación y las firmas de amenazas.

Más allá de las reglas administradas, le recomendamos que utilice prácticas de DevOps para automatizar la implementación de los recursos de red, las protecciones y las reglas que especifique. Puede plasmar estas definiciones en AWS CloudFormation u otra herramienta de infraestructura como código (IaC) de su elección, confirmarlas en un sistema de control de versiones e implementarlas mediante canalizaciones de CI/CD. Utilice este enfoque para obtener los beneficios tradicionales de DevOps para administrar los controles de red, como versiones más predecibles, pruebas automatizadas con herramientas como AWS CloudFormation Guard y detección de desviaciones entre el entorno implementado y la configuración deseada.

En función de las decisiones que haya tomado como parte del proceso descrito en SEC05-BP01 Creación de capas de red, es posible que tenga un enfoque de administración centralizado para la creación de VPC dedicadas a los flujos de entrada, salida e inspección. Tal y como se describe en AWS Security Reference Architecture (AWS SRA), puede definir estas VPC en una cuenta de infraestructura de red dedicada. Puede utilizar técnicas similares para definir de forma centralizada las VPC que utilizan sus cargas de trabajo en otras cuentas, sus grupos de seguridad, las

implementaciones de AWS Network Firewall, las reglas de Route 53 Resolver y las configuraciones de firewall DNS, además de otros recursos de red. Puede compartir estos recursos con sus demás cuentas mediante AWS Resource Access Manager. Con este enfoque, puede simplificar las pruebas automatizadas y la implementación de los controles de red en la cuenta de red, y solo tendrá un destino que administrar. Puede hacerlo en un modelo híbrido, en el que implementa y comparte ciertos controles de forma centralizada y delega otros controles a los equipos de carga de trabajo individuales y a sus respectivas cuentas.

Pasos para la implementación

1. Determine qué aspectos de la red y qué protecciones se definen de forma centralizada y cuáles pueden mantener sus equipos de carga de trabajo.
2. Cree entornos para probar e implementar cambios en su red y sus protecciones. Por ejemplo, utilice una cuenta de pruebas de red y una cuenta de producción de red.
3. Determine cómo va a almacenar y mantener las plantillas en un sistema de control de versiones. Almacene las plantillas centrales en un repositorio distinto de los repositorios de carga de trabajo; las plantillas de carga de trabajo se pueden almacenar en repositorios específicos para esa carga de trabajo.
4. Cree canalizaciones de CI/CD para probar e implementar plantillas. Defina pruebas para comprobar si hay errores de configuración y si las plantillas se ajustan a los estándares de su empresa.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP06 Automatización de la implementación de controles de seguridad estándares

Documentos relacionados:

- AWS Security Reference Architecture - Network account

Ejemplos relacionados:

- AWS Deployment Pipeline Reference Architecture
- NetDevSecOps to modernize AWS networking deployments

- Integrating AWS CloudFormation security tests with AWS Security Hub CSPM and AWS CodeBuild reports

Herramientas relacionadas:

- AWS CloudFormation
- AWS CloudFormation Guard
- cfn_nag

## SEC 6. ¿Cómo protege sus recursos de computación?

Los recursos de computación de su carga de trabajo requieren varios niveles de defensa para protegerse de las amenazas externas e internas. Entre los recursos de computación se incluyen instancias de EC2, contenedores, funciones de AWS Lambda, servicios de bases de datos, dispositivos IoT, etc.

Prácticas recomendadas

- SEC06-BP01 Administración de las vulnerabilidades
- SEC06-BP02 Aprovisionamiento de computación a partir de imágenes reforzadas
- SEC06-BP03 Reducción de la administración manual y el acceso interactivo
- SEC06-BP04 Validación de la integridad del software
- SEC06-BP05 Automatización de la protección de computación

### SEC06-BP01 Administración de las vulnerabilidades

Analice con frecuencia su código, sus dependencias y su infraestructura en busca de vulnerabilidades, y aplique parches para solucionarlas, para ayudarle a protegerse contra las nuevas amenazas.

Resultado deseado: tiene una solución que analiza continuamente su carga de trabajo en busca de vulnerabilidades de software, posibles defectos y exposición no intencionada a la red. Ha establecido procesos y procedimientos para identificar, priorizar y corregir estas vulnerabilidades en función de los criterios de evaluación de riesgos. Además, ha implementado la administración automática de parches para sus instancias de computación. Su programa de gestión de vulnerabilidades está integrado en su ciclo de vida de desarrollo de software, con soluciones para escanear el código fuente durante la canalización de CI/CD.

Patrones comunes de uso no recomendados:

- No disponer de un programa de administración de vulnerabilidades.
- Aplicar parches en el sistema sin tener en cuenta la gravedad o la forma de evitar riesgos.
- Utilizar software que haya superado la fecha de fin de vida útil (EOL) de su proveedor.
- Implementar código en producción antes de analizarlo en busca de problemas de seguridad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La gestión de vulnerabilidades es un aspecto clave para mantener un entorno de nube seguro y sólido. Conlleva un proceso integral que incluye escaneos de seguridad, identificación y priorización de problemas y operaciones de parches para resolver las vulnerabilidades identificadas. La automatización desempeña un papel fundamental en este proceso, ya que facilita el análisis continuo de las cargas de trabajo para detectar posibles problemas y una exposición no intencionada a la red, así como las iniciativas de corrección.

El modelo de responsabilidad compartida de AWS es un concepto fundamental que sustenta la gestión de vulnerabilidades. Según este modelo, AWS es responsable de proteger la infraestructura subyacente, incluido el equipo, el software, las redes y las instalaciones que ejecutan los servicios de AWS. Por el contrario, es su responsabilidad proteger sus datos, las configuraciones de seguridad y las tareas de administración asociadas a servicios como las instancias de Amazon EC2 y los objetos de Amazon S3.

AWS ofrece numerosos servicios para apoyar los programas de administración de vulnerabilidades. Amazon Inspector analiza continuamente las cargas de trabajo de AWS para detectar vulnerabilidades de software y accesos no deseados a la red, mientras que AWSSystems Manager Patch Manager ayuda a administrar los parches en todas las instancias de Amazon EC2. Estos servicios se pueden integrar con AWS Security Hub CSPM, un servicio de administración de la postura de seguridad en la nube que automatiza los controles de seguridad de AWS, centraliza las alertas de seguridad y proporciona una visión completa de la postura de seguridad de una organización. Además, Amazon CodeGuru Security utiliza el análisis estático del código para identificar posibles problemas en las aplicaciones Java y Python durante la fase de desarrollo.

Al incorporar prácticas de gestión de vulnerabilidades en el ciclo de vida del desarrollo del software, puede abordar las vulnerabilidades de forma proactiva antes de que se introduzcan en los entornos

de producción, lo que reduce el riesgo de eventos de seguridad y minimiza el posible impacto de las vulnerabilidades.

Pasos para la implementación

1. Comprenda el modelo de responsabilidad compartida: revise el modelo de responsabilidad compartida de AWS para comprender sus responsabilidades a la hora de proteger sus cargas de trabajo y los datos en la nube. AWS es responsable de proteger la infraestructura en la nube subyacente, mientras que es su responsabilidad proteger las aplicaciones, los datos y los servicios que utiliza.
2. Implemente el análisis de vulnerabilidades: configure un servicio de análisis de vulnerabilidades, como Amazon Inspector, para que analice automáticamente sus instancias de computación (por ejemplo, máquinas virtuales, contenedores o funciones sin servidor) en busca de vulnerabilidades de software, posibles defectos y exposición no intencionada a la red.
3. Establezca procesos de gestión de vulnerabilidades: defina procesos y procedimientos para identificar, priorizar y corregir las vulnerabilidades. Aquí se puede incluir la configuración de programas periódicos de escaneo de vulnerabilidades, el establecimiento de criterios de evaluación de riesgos y la definición de plazos de corrección en función de la gravedad de la vulnerabilidad.
4. Configure la administración de parches: utilice un servicio de administración de parches para automatizar el proceso de aplicación de parches a sus instancias de computación, tanto para los sistemas operativos como para las aplicaciones. Puede configurar el servicio para que analice las instancias en busca de parches ausentes e instalarlos automáticamente de manera planificada. Si lo considera oportuno, puede usar AWS Systems Manager Patch Manager para proporcionar esta funcionalidad.
5. Configure la protección contra el malware: implemente mecanismos para detectar software malicioso en su entorno. Por ejemplo, puede usar herramientas como Amazon GuardDuty para analizar, detectar y alertar sobre el malware en los volúmenes de EC2 y EBS. GuardDuty también puede escanear los objetos recién cargados en Amazon S3 para detectar posibles virus o malware y tomar medidas para aislarlos antes de que se incorporen a los procesos posteriores.
6. Integre el análisis de vulnerabilidades en las canalizaciones de CI/CD: si utiliza una canalización de CI/CD para la implementación de sus aplicaciones, integre herramientas de análisis de vulnerabilidades en su canalización. Herramientas como Amazon CodeGuru Security y las opciones de código abierto pueden analizar el código fuente, las dependencias y los artefactos en busca de posibles problemas de seguridad.

7. Configure un servicio de monitoreo de seguridad: configure un servicio de monitoreo de seguridad, por ejemplo, AWS Security Hub CSPM, para obtener una perspectiva global del estado de su seguridad en varios servicios en la nube. El servicio debe recopilar los resultados de seguridad de diversas fuentes y presentarlos en un formato estandarizado para facilitar la priorización y la corrección.
8. Implemente pruebas de penetración de aplicaciones web: si su aplicación es una aplicación web y su organización tiene personal capacitado o puede contratar asistencia externa, podría implementar pruebas de penetración de aplicaciones web para identificar posibles vulnerabilidades en su aplicación.
9. Automatice con la infraestructura como código: utilice herramientas de infraestructura como código (IaC), por ejemplo AWS CloudFormation, para automatizar la implementación y la configuración de sus recursos, incluidos los servicios de seguridad mencionados anteriormente. Esta práctica ayuda a crear una arquitectura de recursos más coherente y estandarizada en múltiples cuentas y entornos.
10. Supervise y mejore continuamente: supervise continuamente la eficacia de su programa de gestión de vulnerabilidades y realice las mejoras necesarias. Revise los resultados de seguridad, evalúe la eficacia de sus esfuerzos de corrección y ajuste sus procesos y herramientas en consecuencia.

Recursos

Documentos relacionados:

- AWS Systems Manager
- Security Overview of AWS Lambda
- Amazon CodeGuru
- Improved, Automated Vulnerability Management for Cloud Workloads with a New Amazon Inspector
- Automate vulnerability management and remediation in AWS using Amazon Inspector and AWS Systems Manager – Part 1

Videos relacionados:

- Securing Serverless and Container Services
- Security best practices for the Amazon EC2 instance metadata service

### SEC06-BP02 Aprovisionamiento de computación a partir de imágenes reforzadas

Ofrezca menos oportunidades de acceso no deseado a sus entornos en tiempo de ejecución mediante la implementación desde imágenes reforzadas. Adquiera las dependencias de tiempo de ejecución (como imágenes de contenedores y bibliotecas de aplicaciones) únicamente de registros fiables y verifique sus firmas. Cree sus propios registros privados para almacenar imágenes y bibliotecas confiables para usarlas en sus procesos de desarrollo e implementación.

Resultado deseado: sus recursos informáticos se aprovisionan a partir de imágenes de referencia reforzadas. Recupera dependencias externas, como imágenes de contenedores y bibliotecas de aplicaciones, solamente de registros fiables y verifica sus firmas. Las almacena en registros privados para su consulta en los procesos de desarrollo e implementación. Escanea y actualiza las imágenes y las dependencias con regularidad para ayudar a protegerse contra cualquier vulnerabilidad recién descubierta.

Patrones comunes de uso no recomendados:

- Adquirir imágenes y bibliotecas de registros fiables, pero no verificar su firma ni analizar las vulnerabilidades antes de utilizarlas.
- Reforzar las imágenes, pero no probarlas con regularidad para detectar nuevas vulnerabilidades ni actualizarlas a la versión más reciente.
- Instalar o no eliminar paquetes de software que no sean necesarios durante el ciclo de vida esperado de la imagen.
- Confiar únicamente en los parches para mantener actualizados los recursos de computación de producción. El uso de parches por sí solo puede seguir haciendo que los recursos de computación diverjan del estándar reforzado con el paso del tiempo. También es posible que el uso de parches no elimine el malware que un actor de amenazas pueda haber instalado durante un evento de seguridad.

Beneficios de establecer esta práctica recomendada: el refuerzo de las imágenes ayuda a reducir la cantidad de rutas disponibles en el entorno de tiempo de ejecución que pueden permitir el acceso no deseado a usuarios o servicios no autorizados. También puede reducir el alcance del impacto en caso de que se produzca un acceso no deseado.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para reforzar sus sistemas, comience con las versiones más recientes de los sistemas operativos, las imágenes de contenedores y las bibliotecas de aplicaciones. Aplique parches para solucionar los problemas conocidos. Reduzca al mínimo el sistema eliminando las aplicaciones, los servicios, los controladores de dispositivos, los usuarios predeterminados y otras credenciales que no sean necesarios. Lleve a cabo cualquier otra acción necesaria, como deshabilitar los puertos para crear un entorno que solo tenga los recursos y las capacidades que necesiten sus cargas de trabajo. A partir de ahí, puede instalar el software, los agentes u otros procesos que necesite para objetivos como la supervisión de la carga de trabajo o la administración de vulnerabilidades.

Puede reducir la carga que supone reforzar los sistemas utilizando la orientación que proporcionan orígenes fiables, como el Center for Internet Security (CIS) y las Security Technical Implementation Guides (STIG) de la Defense Information Systems Agency (DISA). Le recomendamos que comience con una imagen de máquina de Amazon (AMI) publicada por AWS o un socio de la APN y que utilice el Generador de imágenes de AWS EC2 para automatizar la configuración de acuerdo con una combinación adecuada de controles del CIS y las STIG.

Si bien existen imágenes reforzadas y recetas del Generador de imágenes de EC2 disponibles que aplican las recomendaciones del CIS o de las STIG de la DISA, es posible que su configuración impida que su software se ejecute correctamente. En esta situación, puede partir de una imagen de base no reforzada, instalar el software y, a continuación, aplicar los controles del CIS de forma gradual para comprobar su impacto. Para cualquier control del CIS que impida la ejecución del software, compruebe si puede implementar las recomendaciones de refuerzo más detalladas prescritas por la DISA. Lleve un registro de los diferentes controles del CIS y de las configuraciones especificadas en las STIG de la DISA que puede aplicar correctamente. Utilícelos para definir consecuentemente sus recetas de refuerzo de imágenes en el Generador de imágenes de EC2.

Para las cargas de trabajo en contenedores, hay imágenes reforzadas de Docker disponibles en el repositorio público de Amazon Elastic Container Registry (ECR). Puede usar el Generador de imágenes de EC2 para reforzar las imágenes de contenedor junto con las AMI.

Al igual que los sistemas operativos y las imágenes de contenedor, puede obtener paquetes de código (o bibliotecas) de repositorios públicos mediante herramientas como pip, npm, Maven y NuGet. Le recomendamos que administre los paquetes de código mediante la integración de repositorios privados, como los que hay en AWS CodeArtifact, con repositorios públicos de confianza. Esta integración puede ocuparse de la recuperación, el almacenamiento y la actualización de los paquetes. Durante los procesos de desarrollo de aplicaciones, se puede obtener y probar la versión más reciente de estos paquetes junto con la aplicación, mediante técnicas como el análisis

de composición de software (SCA), las pruebas de seguridad de aplicaciones estáticas (SAST) y las pruebas dinámicas de seguridad de aplicaciones (DAST).

Para las cargas de trabajo sin servidor que utilicen AWS Lambda, simplifique la administración de las dependencias de los paquetes mediante capas de Lambda. Use las capas de Lambda para configurar un conjunto de dependencias estándares que se compartan entre diferentes funciones en un archivo independiente. Puede crear y mantener capas según su propio proceso de creación, que proporciona un método centralizado de mantener al día sus funciones.

Pasos para la implementación

- Refuerce los sistemas operativos. Utilice imágenes de base de orígenes fiables para crear sus AMI reforzadas. Use el Generador de imágenes de EC2 para ayudar a personalizar el software instalado en las imágenes.
- Refuerce los recursos en contenedores. Configure los recursos en contenedores para cumplir con las prácticas recomendadas de seguridad. Cuando utilice contenedores, implemente el análisis de imágenes de ECR en su canalización de compilación y aplíquelo de forma frecuente a su repositorio de imágenes para buscar CVE en sus contenedores.
- Cuando utilice la implementación sin servidor con AWS Lambda, utilice capas de Lambda para dividir el código de función de la aplicación y las bibliotecas dependientes compartidas. Configure la firma de código para Lambda para asegurarse de que solo se ejecute código fiable en sus funciones de Lambda.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP05 Administración de parches

Videos relacionados:

- Deep dive into AWS Lambda security

Ejemplos relacionados:

- Quickly build STIG-compliant AMI using EC2 Image Builder
- Building better container images

- Using Lambda layers to simplify your development process
- Develop & Deploy AWS Lambda Layers using Serverless Framework
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST and DAST tools

SEC06-BP03 Reducción de la administración manual y el acceso interactivo

Utilice la automatización para hacer tareas de implementación, configuración, mantenimiento e investigación siempre que sea posible. Plantéese el acceso manual a los recursos de computación en casos de procedimientos de emergencia o en entornos seguros (entornos de pruebas) cuando la automatización no esté disponible.

Resultado deseado: los scripts programáticos y los documentos de automatización (manuales de procedimientos) capturan las acciones autorizadas en sus recursos de computación. Estos manuales de procedimientos se inician de forma automática, mediante sistemas de detección de cambios, o manualmente, cuando se requiere una intervención humana. El acceso directo a los recursos de computación solo está disponible en situaciones de emergencia cuando la automatización no está disponible. Todas las actividades manuales se registran y se incorporan a un proceso de revisión para mejorar continuamente las capacidades de automatización.

Patrones comunes de uso no recomendados:

- Acceso interactivo a instancias de Amazon EC2 con protocolos como SSH o RDP.
- Mantener inicios de sesión de los usuarios individuales, como `/etc/passwd` o los usuarios locales de Windows.
- Compartir una contraseña o una clave privada para acceder a una instancia entre varios usuarios.
- Instalar el software y crear o actualizar los archivos de configuración manualmente.
- Actualizar o parchear el software manualmente.
- Iniciar sesión en una instancia para solucionar problemas.

Beneficios de establecer esta práctica recomendada: utilizar la automatización para llevar a cabo acciones le ayuda a reducir el riesgo operativo de los cambios no deseados y los errores de configuración. Eliminar el uso de Secure Shell (SSH) y Remote Desktop Protocol (RDP) para el acceso interactivo reduce el alcance del acceso a los recursos de computación. Todo esto elimina una ruta que se utiliza habitualmente para llevar a cabo acciones no autorizadas. Al reflejar las

tareas de administración de recursos de computación en documentos de automatización y scripts programáticos, se proporciona un mecanismo para definir y auditar todo el alcance de las actividades autorizadas con un alto nivel de detalle.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Iniciar sesión en una instancia es un enfoque clásico de la administración del sistema. Tras instalar el sistema operativo del servidor, los usuarios suelen iniciar sesión manualmente para configurar el sistema e instalar el software deseado. A lo largo de la vida útil del servidor, los usuarios pueden iniciar sesión para llevar a cabo actualizaciones de software, aplicar parches, cambiar configuraciones y solucionar problemas.

Sin embargo, el acceso manual plantea una serie de riesgos. Requiere un servidor que escuche las solicitudes, como un servicio SSH o RDP, que pueden proporcionar una ruta potencial para el acceso no autorizado. También aumenta el riesgo de errores humanos asociados con los pasos manuales. Todo esto puede provocar incidentes relacionados con la carga de trabajo, corrupción o destrucción de datos u otros problemas de seguridad. El acceso humano también requiere protecciones contra el uso compartido de credenciales, lo que genera una sobrecarga de administración adicional.

Para mitigar estos riesgos, puede implementar una solución de acceso remoto basada en agentes, como AWS Systems Manager. AWS Systems Manager El agente (SSM Agent) inicia un canal cifrado y, por lo tanto, no depende de la escucha de solicitudes iniciadas externamente. Tenga en cuenta la posibilidad de configurar el SSM Agent para establecer este canal en un punto de conexión de VPC.

Systems Manager le aporta un control detallado sobre cómo puede interactuar con las instancias administradas. Es el cliente quien define las automatizaciones que se ejecutarán, quién puede ejecutarlas y cuándo pueden ejecutarse. Systems Manager puede aplicar parches, instalar software y hacer cambios en la configuración sin acceso interactivo a la instancia. Systems Manager también puede proporcionar acceso a un intérprete de comandos remoto y registrar todos los comandos invocados y sus resultados durante la sesión en registros y en Amazon S3. AWS CloudTrail registra las invocaciones de las API de Systems Manager para su inspección.

Pasos para la implementación

1. Instale el AWS Systems Manager Agent (SSM Agent) en sus instancias de Amazon EC2. Compruebe si el SSM Agent está incluido y se ha iniciado automáticamente como parte de la configuración básica de la AMI.

2. Compruebe que los roles de IAM asociados a sus perfiles de instancia de EC2 incluyan la política de IAM administrada `AmazonSSMManagedInstanceCore`.
3. Inhabilite SSH, RDP y otros servicios de acceso remoto que se ejecuten en sus instancias. Para hacerlo, puede ejecutar scripts configurados en la sección de datos de usuario de sus plantillas de lanzamiento o crear AMI personalizadas con herramientas como el Generador de imágenes de EC2.
4. Compruebe que las reglas de ingreso de grupos de seguridad aplicables a sus instancias de EC2 no permitan el acceso al puerto 22/tcp (SSH) o al puerto 3389/tcp (RDP). Implemente la detección y las alertas en grupos de seguridad mal configurados mediante servicios como AWS Config.
5. Defina las automatizaciones, los manuales de procedimientos y los comandos de ejecución adecuados en Systems Manager. Utilice políticas de IAM para definir quién puede llevar a cabo estas acciones y las condiciones en las que están permitidas. Pruebe estas automatizaciones minuciosamente en un entorno que no sea de producción. Invoque estas automatizaciones cuando sea necesario, en lugar de acceder a la instancia de forma interactiva.
6. Use AWS Systems Manager Session Manager para proporcionar acceso interactivo a las instancias cuando sea necesario. Active el registro de actividad de la sesión para mantener un registro de auditoría en Registros de Amazon CloudWatch o Amazon S3.

Recursos

Prácticas recomendadas relacionadas:

- REL08-BP04 Implementación mediante una infraestructura inmutable

Ejemplos relacionados:

- Replacing SSH access to reduce management and security overhead with AWS Systems Manager

Herramientas relacionadas:

- AWS Systems Manager

Videos relacionados:

- Controlling User Session Access to Instances in AWS Systems Manager Session Manager

### SEC06-BP04 Validación de la integridad del software

Utilice la verificación criptográfica para validar la integridad de los artefactos de software (incluidas las imágenes) que utiliza su carga de trabajo. Firme criptográficamente su software como protección contra los cambios no autorizados que se ejecuten en sus entornos de computación.

Resultado deseado: todos los artefactos se obtienen de orígenes confiables. Se validan los certificados del sitio web del proveedor. Los artefactos descargados se verifican criptográficamente mediante sus firmas. Sus entornos de computación firman y verifican criptográficamente su propio software.

Patrones comunes de uso no recomendados:

- Confiar en los sitios web de proveedores acreditados para obtener artefactos de software, pero ignorar los avisos de caducidad de los certificados. Continuar con las descargas sin confirmar que los certificados son válidos.
- Validar los certificados de los sitios web de los proveedores, pero no verificar criptográficamente los artefactos descargados de estos sitios web.
- Confiar únicamente en resúmenes o hashes para validar la integridad del software. Los hashes establecen que los artefactos no se han modificado con respecto a la versión original, pero no validan su fuente.
- No firmar su propio software, código o bibliotecas, aunque solo los utilice en sus propias implementaciones.

Beneficios de establecer esta práctica recomendada: la validación de la integridad de los artefactos de los que depende su carga de trabajo ayuda a evitar que el malware acceda a sus entornos de computación. La firma de su software ayuda a protegerse contra la ejecución no autorizada en sus entornos informáticos. Proteja su cadena de suministro de software mediante la firma y verificación del código.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

#### Guía para la implementación

Las imágenes del sistema operativo, las imágenes de contenedores y los artefactos de código suelen distribuirse con comprobaciones de integridad disponibles, por ejemplo, mediante un resumen o un hash. Esto permite a los clientes verificar la integridad calculando su propio hash de la carga útil y validando que sea el mismo que el publicado. Si bien estas comprobaciones ayudan a verificar que

la carga útil no se ha manipulado, no validan que provenga de la fuente original (su procedencia). La verificación de la procedencia requiere un certificado emitido por una autoridad de confianza para firmar digitalmente el artefacto.

Si utiliza un software o artefactos descargados en su carga de trabajo, compruebe si el proveedor proporciona una clave pública para la verificación de la firma digital. Estos son algunos ejemplos de cómo AWS proporciona una clave pública e instrucciones de verificación para el software que publicamos:

- EC2 Image Builder: Verify the signature of the AWSTOE installation download
- AWS Systems Manager: Verifying the signature of SSM Agent
- Amazon CloudWatch: Verifying the signature of the CloudWatch agent package

Incorpore la verificación de firmas digitales en los procesos que utilice para obtener y reforzar las imágenes, como se explica en SEC06-BP02 Aprovisionamiento de computación a partir de imágenes reforzadas.

Puede usar AWS Signer para administrar la verificación de firmas, así como su propio ciclo de vida de firma de código para su propio software y artefactos. Tanto AWS Lambda como Amazon Elastic Container Registry proporcionan integraciones con Signer para verificar las firmas de su código y sus imágenes. Con los ejemplos de la sección Recursos, puede incorporar Signer a sus procesos de integración y entrega continuas (CI/CD) para automatizar la verificación de las firmas y la firma de su propio código e imágenes.

Recursos

Documentos relacionados:

- Cryptographic Signing for Containers
- Best Practices to help secure your container image build pipeline by using AWS Signer
- Announcing Container Image Signing with AWS Signer and Amazon EKS
- Configuring code signing for AWS Lambda
- Best practices and advanced patterns for Lambda code signing
- Code signing using AWS Gestor de certificados Private CA and AWS Key Management Service asymmetric keys

Ejemplos relacionados:

- Automate Lambda code signing with Amazon CodeCatalyst and AWS Signer
- Signing and Validating OCI Artifacts with AWS Signer

Herramientas relacionadas:

- AWS Lambda
- AWS Signer
- AWS Gestor de certificados
- AWS Key Management Service
- AWS CodeArtifact

SEC06-BP05 Automatización de la protección de computación

Automatice las operaciones de protección de computación para reducir la necesidad de intervención humana. Utilice el análisis automatizado para detectar posibles problemas en sus recursos de computación y use respuestas programáticas automatizadas u operaciones de administración de flotas para solucionarlos.  Incorpore la automatización en sus procesos de CI/CD para implementar cargas de trabajo fiables con dependencias actualizadas.

Resultado deseado: los sistemas automatizados llevan a cabo todos los escaneos y parches de los recursos de computación. Use la verificación automática para comprobar que las imágenes y dependencias del software provengan de orígenes fiables y que no se hayan manipulado. Las cargas de trabajo se comprueban automáticamente para determinar si las dependencias están actualizadas y se firman para establecer la fiabilidad en los entornos de computación de AWS.  Las correcciones automatizadas se inician cuando se detectan recursos no conformes con los requisitos.

Patrones comunes de uso no recomendados:

- Seguir la práctica de una infraestructura inmutable sin contar con una solución para la instalación de parches de emergencia o la sustitución de sistemas de producción.
- Usar la automatización para corregir los recursos mal configurados sin contar con un mecanismo de anulación manual.  Es posible que surjan situaciones en las que necesite ajustar los requisitos y tenga que suspender las automatizaciones hasta haber hecho estos cambios.

Beneficios de establecer esta práctica recomendada: la automatización puede reducir el riesgo de accesos y usos no autorizados de sus recursos de computación.  Ayuda a evitar que lleguen

configuraciones incorrectas a los entornos de producción y a detectarla y corregirlas en caso de que se produzcan. La automatización también contribuye a detectar el acceso y el uso no autorizados de los recursos informáticos para reducir el tiempo de respuesta. Esto, a su vez, puede reducir el alcance general del impacto del problema.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Puede aplicar las automatizaciones descritas en las prácticas del pilar de seguridad para proteger sus recursos de computación. En SEC06-BP01 Administración de las vulnerabilidades se describe cómo puede utilizar Amazon Inspector tanto en sus canalizaciones de CI/CD como para analizar continuamente sus entornos en tiempo de ejecución en busca de vulnerabilidades y exposiciones comunes (CVE) conocidas. Puede usar AWS Systems Manager para aplicar parches o volver a implementar imágenes nuevas mediante manuales de procedimientos automatizados para mantener su flota de computación actualizada con el software y las bibliotecas más recientes. Utilice estas técnicas para reducir la necesidad de recurrir a procesos manuales y el acceso interactivo a sus recursos informáticos. Consulte SEC06-BP03 Reducción de la administración manual y el acceso interactivo para obtener más información.

La automatización también desempeña un papel en la implementación de cargas de trabajo confiables, tal como se describe en SEC06-BP02 Aprovisionamiento de computación a partir de imágenes reforzadas y SEC06-BP04 Validación de la integridad del software. Puede usar servicios como el Generador de imágenes de EC2, AWS Signer, AWS CodeArtifact y Amazon Elastic Container Registry (ECR) para descargar, verificar, construir y almacenar dependencias de código e imágenes reforzadas y aprobadas. Junto con Inspector, cada uno de estos elementos puede desempeñar un papel en su proceso de CI/CD, de modo que su carga de trabajo llegue al entorno de producción solo cuando se confirme que sus dependencias están actualizadas y provienen de orígenes fiables. La carga de trabajo también está firmada para que los entornos de computación de AWS, como AWS Lambda y Amazon Elastic Kubernetes Service (EKS), puedan verificar que no se ha manipulado antes de permitir su ejecución.

Además de estos controles preventivos, también puede utilizar la automatización en sus controles de detección para sus recursos de computación. Por ejemplo, AWS Security Hub CSPM ofrece el estándar NIST 800-53 Rev. 5, que incluye comprobaciones como esta: [EC2.8] las instancias de EC2 deben usar la versión 2 del servicio de metadatos de instancia (IMDSv2). El IMDSv2 utiliza técnicas de autenticación de sesión y bloquea las solicitudes que contienen un encabezado HTTP X-Forwarded-For y un TTL de red de 1 para detener el tráfico que se origina en fuentes externas para recuperar información sobre la instancia de EC2. Esta comprobación en Security Hub CSPM puede

detectar si las instancias de EC2 utilizan IMDSv1 e iniciar una corrección automatizada. Obtenga más información sobre la detección y las correcciones automatizadas en SEC04-BP04 Inicio de correcciones para recursos no conformes.

Pasos para la implementación

1. Automatice la creación de AMI seguras, compatibles y reforzadas con el Generador de imágenes de EC2. Puede producir imágenes que incorporen los controles de los estándares comparativos del Center for Internet Security (CIS) o de la Security Technical Implementation Guide (STIG) a partir de imágenes base de AWS e imágenes de socios de APN.
2. Automatice la administración de la configuración. Aplique y valide configuraciones seguras en sus recursos de computación de forma automática mediante el uso de un servicio o herramienta de gestión de configuraciones.
    a. Administración automatizada de la configuración con AWS Config
    b. Administración automatizada de la posición de seguridad y cumplimiento mediante AWS Security Hub CSPM
3. Automatice la aplicación de parches o el reemplazo de instancias de Amazon Elastic Compute Cloud (Amazon EC2). AWS Systems Manager Patch Manager automatiza el proceso de aplicación de parches a instancias administradas con actualizaciones de seguridad y de otro tipo. Puede utilizar Patch Manager para aplicar parches a los sistemas operativos y a las aplicaciones.
    a. AWS Systems Manager Patch Manager
4. Automatice el análisis de los recursos de computación en busca de vulnerabilidades y exposiciones comunes (CVE) e incorpore soluciones de análisis de seguridad en su proceso de desarrollo.
    a. Amazon Inspector
    b. ECR Image Scanning
5. Plantéese el uso de Amazon GuardDuty para detectar amenazas y malware de forma automática con el fin de proteger los recursos de computación. GuardDuty también puede identificar posibles problemas cuando se invoca una función de AWS Lambda en su entorno de AWS.
    a. Amazon GuardDuty
6. Tenga en cuenta las soluciones de socios de AWS. AWS Los socios ofrecen cientos de productos destacados que son equivalentes o idénticos a los controles que ya utiliza en sus entornos en las instalaciones o que pueden integrarse con ellos. Estos productos complementan a los servicios de AWS existentes y le permiten implementar una arquitectura de seguridad integral, así como disfrutar de una experiencia más fluida tanto en la nube como en los entornos en las instalaciones.

a. Seguridad de la infraestructura

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP06 Automatización de la implementación de controles de seguridad estándares

Documentos relacionados:

- Get the full benefits of IMDSv2 and disable IMDSv1 across your AWS infrastructure

Videos relacionados:

- Security best practices for the Amazon EC2 instance metadata service

## Protección de datos

Preguntas

- SEC 7. ¿Cómo clasifica sus datos?
- SEC 8. ¿Cómo protege sus datos en reposo?
- SEC 9. ¿Cómo protege sus datos en tránsito?

### SEC 7. ¿Cómo clasifica sus datos?

La clasificación proporciona una forma de categorizar los datos, basada en el nivel de importancia y la confidencialidad, para ayudarlo a determinar los controles de protección y de conservación adecuados.

Prácticas recomendadas

- SEC07-BP01 Comprensión del esquema de clasificación de datos
- SEC07-BP02 Aplicación de controles de protección de datos según la confidencialidad de los datos
- SEC07-BP03 Automatización de la identificación y la clasificación
- SEC07-BP04 Definición de la administración escalable del ciclo de vida de los datos

### SEC07-BP01 Comprensión del esquema de clasificación de datos

Comprenda la clasificación de los datos que procesa su carga de trabajo, los requisitos de su tratamiento, los procesos empresariales asociados, dónde se almacenan los datos y quién es su propietario. Su esquema de clasificación y tratamiento de datos debe tener en cuenta los requisitos legales y de cumplimiento aplicables a su carga de trabajo y los controles de datos necesarios. La comprensión de los datos es el primer paso en el proceso de clasificación de los datos.

Resultado deseado: se comprenden y se documentan adecuadamente los tipos de datos presentes en su carga de trabajo. Dispone de controles adecuados para proteger los datos confidenciales en función de su clasificación. Estos controles determinan cuestiones como quién puede acceder a los datos y con qué propósito, dónde se almacenan los datos, la política de cifrado de esos datos y la forma en que se administran las claves de cifrado, el ciclo de vida de los datos y sus requisitos de retención, los procesos de destrucción pertinentes, los procesos de copia de seguridad y recuperación existentes y la auditoría del acceso.

Patrones comunes de uso no recomendados:

- No contar con una política formal de clasificación de datos para definir los niveles de confidencialidad de los datos y sus requisitos de tratamiento
- No comprender bien los niveles de confidencialidad de los datos dentro de su carga de trabajo y no reflejar esta información en la documentación de la arquitectura y las operaciones
- No aplicar los controles adecuados en torno a sus datos en función de su confidencialidad y sus requisitos, tal como se describe en su política de clasificación y tratamiento de datos
- No proporcionar comentarios sobre los requisitos de clasificación y tratamiento de datos a los propietarios de las políticas.

Beneficios de establecer esta práctica recomendada: esta práctica elimina la ambigüedad en torno al tratamiento adecuado de los datos dentro de su carga de trabajo. La aplicación de una política formal que defina los niveles de confidencialidad de los datos en su organización y las protecciones que requieren puede ayudarle a cumplir la normativa legal y otras acreditaciones y certificaciones de ciberseguridad. Los propietarios de las cargas de trabajo tienen la confianza de saber dónde se almacenan los datos confidenciales y qué controles de protección existen. Plasmar esta información en la documentación ayuda a los nuevos miembros del equipo a comprenderla mejor y a atenerse a estos controles desde el principio de su incorporación. Estas prácticas también pueden contribuir a reducir los costos al dimensionar correctamente los controles para cada tipo de datos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Al diseñar una carga de trabajo, es posible que se plantee diversas formas de proteger los datos confidenciales de manera intuitiva. Por ejemplo, en una aplicación con varios inquilinos, resulta intuitivo pensar que los datos de cada inquilino son confidenciales y establecer protecciones para que un inquilino no pueda acceder a los datos de otro. Del mismo modo, puede diseñar controles de acceso de forma intuitiva para que únicamente los administradores puedan modificar los datos y que otros usuarios solo tengan acceso de nivel de lectura o no tengan ningún tipo de acceso.

Al definir y plasmar estos niveles de confidencialidad de los datos en la política, junto con sus requisitos de protección de datos, puede identificar formalmente qué datos residen en su carga de trabajo. A continuación, puede determinar si cuenta con los controles correctos, si los controles se pueden auditar y qué respuestas son apropiadas si se determina que se está produciendo un tratamiento de los datos incorrecto.

Para ayudar a identificar dónde se encuentran los datos confidenciales dentro de su carga de trabajo, puede utilizar un catálogo de datos. Un catálogo de datos es una base de datos que mapea los datos de su organización, su ubicación, nivel de confidencialidad y los controles establecidos para proteger los datos. Además, tiene la opción de utilizar etiquetas de recursos si están disponibles. Por ejemplo, puede aplicar una etiqueta que tenga una clave de etiqueta de `Classification` y un valor de etiqueta de `PHI` para la información de salud protegida (PHI), y otra etiqueta que tenga una clave de etiqueta de `Sensitivity` y un valor de etiqueta de `High`. Servicios como AWS Config se pueden usar a continuación para supervisar estos recursos en busca de cambios y alertar si se modifican de modo que dejen de cumplir los requisitos de protección (por ejemplo, al cambiar la configuración de cifrado). Puede capturar la definición estándar de las claves de sus etiquetas y los valores aceptables mediante políticas de etiquetas, una característica de AWS Organizations. No se recomienda que la clave o el valor de la etiqueta contengan datos privados o confidenciales.

Pasos para la implementación

1. Comprenda el esquema de clasificación de datos y los requisitos de protección de su organización.
2. Identifique los tipos de datos confidenciales que procesan sus cargas de trabajo.
3. Recopile los datos en un catálogo de datos que proporcione una vista única de dónde residen los datos en la organización y su nivel de confidencialidad.
4. Plantéese el uso del etiquetado de recursos y datos (si está disponible) para etiquetar los datos con su nivel de confidencialidad y otros metadatos operativos que puedan contribuir a la supervisión y la respuesta a los incidentes.

a. Se pueden utilizar políticas de etiquetas de AWS Organizations para hacer cumplir los estándares de etiquetado.

Recursos

Prácticas recomendadas relacionadas:

- SUS04-BP01 Implementación de una política de clasificación de datos

Documentos relacionados:

- Data Classification whitepaper
- Best Practices for Tagging AWS Resources

Ejemplos relacionados:

- AWS Organizations Tag Policy Syntax and Examples

Herramientas relacionadas

- Editor de etiquetas de AWS

SEC07-BP02 Aplicación de controles de protección de datos según la confidencialidad de los datos

Aplique controles de protección de datos que proporcionen un nivel de control adecuado para cada clase de datos definida en su política de clasificación. Esta práctica puede permitirle proteger los datos confidenciales ante el acceso y el uso no autorizados y, al mismo tiempo, preservar la disponibilidad y el uso de los datos.

Resultado deseado: cuenta con una política de clasificación que defina los diferentes niveles de confidencialidad de los datos de su organización. Para cada uno de estos niveles de confidencialidad, haber publicado directrices claras para los servicios y ubicaciones de almacenamiento y gestión aprobados, así como su configuración requerida. Implementa los controles para cada nivel de acuerdo con el nivel de protección requerido y sus costos asociados. Dispone de medidas de supervisión y generación de alertas para detectar si hay datos en ubicaciones no autorizadas, si se están procesando en entornos no autorizados, si hay actores

no autorizados que accedan a ellos o si la configuración de los servicios relacionados deja de ser conforme a las normas.

Patrones comunes de uso no recomendados:

- Aplicar el mismo nivel de controles de protección a todos los datos. Esto puede provocar un aprovisionamiento excesivo de controles de seguridad para datos con un bajo nivel de confidencialidad o una protección insuficiente de los datos altamente confidenciales.
- No implicar a las partes interesadas pertinentes de los equipos de seguridad, de cumplimiento y de empresa al definir los controles de protección de datos.
- Pasar por alto los gastos operativos y los costos asociados con la implementación y el mantenimiento de los controles de protección de datos.
- No llevar a cabo revisiones periódicas del control de protección de datos para mantener la alineación con las políticas de clasificación.
- No tener un inventario completo de dónde se encuentran los datos en reposo y en tránsito.

Beneficios de establecer esta práctica recomendada: al alinear los controles con el nivel de clasificación de los datos, la organización puede invertir en niveles de control más altos cuando sea necesario. Esto podría suponer un aumento de los recursos destinados a proteger, supervisar, medir, corregir y notificar.  Cuando resulte pertinente tener menos controles, puede mejorar la accesibilidad y la integridad de los datos para sus empleados, clientes o miembros.  Este enfoque ofrece a su organización la máxima flexibilidad en el uso de los datos y, al mismo tiempo, sigue cumpliendo los requisitos de protección de datos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La implementación de controles de protección de datos según los niveles de confidencialidad de los datos implica varios pasos clave. En primer lugar, identifique los diferentes niveles de confidencialidad de los datos dentro de su arquitectura de cargas de trabajo (por ejemplo: públicos, internos, confidenciales y restringidos) y valore dónde almacenar y procesar estos datos. A continuación, defina los límites de aislamiento en torno a los datos en función de su nivel de confidencialidad. Le recomendamos que separe los datos en diferentes Cuentas de AWS y utilice políticas de control de servicios (SCP) para restringir los servicios y las acciones permitidos para cada nivel de confidencialidad de los datos. De esta manera, puede crear límites de aislamiento bien definidos y hacer cumplir el principio de privilegios mínimos.

Después de definir los límites de aislamiento, implemente los controles de protección adecuados en función de los niveles de confidencialidad de los datos. Consulte las prácticas recomendadas para proteger los datos en reposo y proteger los datos en tránsito para implementar los controles pertinentes, como el cifrado, los controles de acceso y la auditoría. Plantéese técnicas como la tokenización o la anonimización para reducir el nivel de confidencialidad de sus datos. Simplifique la aplicación de políticas de datos coherentes en toda su empresa con un sistema centralizado de tokenización y destokenización.

Supervise y pruebe continuamente la eficacia de los controles implementados. Revise y actualice periódicamente el esquema de clasificación de datos, las evaluaciones de riesgos y los controles de protección a medida que evolucionen el panorama de datos y las amenazas de su organización. Alinee los controles de protección de datos implementados con los reglamentos, estándares y requisitos legales pertinentes de su sector. Además, ofrezca recursos de concienciación y formación sobre seguridad para ayudar a los empleados a comprender el esquema de clasificación de datos y sus responsabilidades en cuanto al tratamiento y la protección de los datos confidenciales.

Pasos para la implementación

1. Identifique los niveles de clasificación y confidencialidad de los datos dentro de su carga de trabajo.
2. Defina límites de aislamiento para cada nivel y determine una estrategia de cumplimiento.
3. Evalúe los controles definidos para regular el acceso, el cifrado, la auditoría, la retención y el resto de requisitos que exija su política de clasificación de datos.
4. Evalúe las opciones para reducir el nivel de confidencialidad de los datos cuando corresponda, como el uso de la tokenización o la anonimización.
5. Verifique sus controles mediante pruebas y medidas de supervisión automatizadas de los recursos configurados.

Recursos

Prácticas recomendadas relacionadas:

- PERF03-BP01 Uso de un almacén de datos personalizado que se adapte mejor a los requisitos de acceso y almacenamiento de datos
- COST04-BP05 Aplicación de políticas de retención de datos

Documentos relacionados:

- Data Classification whitepaper
- Prácticas recomendadas para la seguridad, la identidad y el cumplimiento
- Prácticas recomendadas de AWS KMS
- Encryption best practices and features for AWS services

Ejemplos relacionados:

- Building a serverless tokenization solution to mask sensitive data
- How to use tokenization to improve data security and reduce audit scope

Herramientas relacionadas:

- AWS Key Management Service (AWS KMS)
- AWS CloudHSM
- AWS Organizations

SEC07-BP03 Automatización de la identificación y la clasificación

La automatización de la identificación y clasificación de datos puede ayudarle a implementar los controles correctos. El uso de la automatización para aumentar la determinación manual reduce el riesgo de errores humanos y exposiciones.

Resultado deseado: puede verificar si dispone de los controles adecuados en función de su política de clasificación y gestión. Las herramientas y los servicios automatizados le ayudan a identificar y clasificar el nivel de confidencialidad de sus datos.  La automatización también le ayuda a supervisar continuamente sus entornos para detectar y alertar si los datos se almacenan o gestionen de manera no autorizada, de modo que se puedan tomar medidas correctivas rápidamente.

Patrones comunes de uso no recomendados:

- Confiar únicamente en procesos manuales para la identificación y clasificación de datos, que pueden ser propensos a errores y requerir mucho tiempo.  Esto puede provocar una clasificación de datos ineficiente e incoherente, especialmente a medida que aumentan los volúmenes de datos.
- No disponer de mecanismos para rastrear y administrar los activos de datos en toda la organización.

- Pasar por alto la necesidad de supervisar y clasificar continuamente los datos a medida que circulan y evolucionan dentro de la organización.

Beneficios de establecer esta práctica recomendada: la automatización de la identificación y la clasificación de datos puede provocar una aplicación más coherente y precisa de los controles de protección de datos, lo que reduce el riesgo de errores humanos. La automatización también puede proporcionar visibilidad sobre el acceso y la circulación de datos confidenciales, lo que le ayuda a detectar las manipulaciones no autorizadas y a tomar medidas correctivas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si bien es habitual recurrir a las decisiones humanas para clasificar los datos durante las fases iniciales del diseño de una carga de trabajo, plantéese la posibilidad de contar con sistemas que automaticen la identificación y la clasificación de los datos de prueba como control preventivo. Por ejemplo, a los desarrolladores se les puede proporcionar una herramienta o un servicio para analizar datos representativos y determinar su confidencialidad. En AWS, puede cargar conjuntos de datos en Amazon S3 y analizarlos con Amazon Macie, Amazon Comprehend o Amazon Comprehend Medical. Del mismo modo, piense en la posibilidad de incluir el análisis de datos como parte de las pruebas unitarias y de integración para detectar en qué puntos no se espera que haya datos confidenciales. Las alertas sobre la presencia de datos confidenciales en esta etapa pueden poner de manifiesto las brechas en las protecciones antes de la implementación en producción. Otras funciones, como la detección de datos confidenciales en AWS Glue, Amazon SNS y Amazon CloudWatch, también se pueden utilizar para detectar datos personales identificables y tomar medidas de mitigación. En el caso de las herramientas o servicios automatizados, comprenda cómo definen los datos confidenciales y compleméntelos con otras soluciones humanas o automatizadas para cubrir las carencias existentes, según sea necesario.

Como control de detección, utilice la supervisión continua de sus entornos para detectar si se están almacenando datos confidenciales de manera no conforme a las normas. Esto puede ayudar a detectar situaciones como la introducción de datos confidenciales en archivos de registro o la copia de este tipo de información en un entorno de análisis de datos sin la debida anonimización o edición. Los datos que se almacenan en Amazon S3 se pueden supervisar continuamente con Amazon Macie para detectar la presencia de datos confidenciales.

Pasos para la implementación

1. Revise el esquema de clasificación de datos de su organización que se describe en SEC07-BP01.

   a. Si comprende el esquema de clasificación de datos de su organización, podrá establecer procesos precisos para la identificación y clasificación automatizadas que se ajusten a las políticas de la empresa.

2. Analice sus entornos inicialmente para llevar a cabo una identificación y una clasificación automatizadas.

   a. El análisis completo inicial de sus datos puede contribuir a obtener un conocimiento detallado de dónde se encuentran los datos confidenciales en sus entornos. Si inicialmente no se requiere un análisis completo o no se puede completar por adelantado debido al costo, evalúe si las técnicas de muestreo de datos son adecuadas para lograr sus resultados. Por ejemplo, se puede configurar Amazon Macie para llevar a cabo una operación amplia y automatizada de detección de datos confidenciales en los buckets de S3.  Esta capacidad utiliza técnicas de muestreo para llevar a cabo un análisis preliminar de dónde se encuentran los datos confidenciales de forma asequible.  A continuación, se puede hacer un análisis en mayor profundidad de los buckets de S3 mediante un trabajo de detección de datos confidenciales. También se pueden exportar otros almacenes de datos a S3 para escanearlos con Macie.

   b. Establezca el control de acceso que se define en SEC07-BP02 para los recursos de almacenamiento de datos identificados en el análisis.

3. Configure análisis continuos de sus entornos.

   a. La capacidad automatizada de detección de datos confidenciales de Macie se puede utilizar para llevar a cabo análisis continuos de sus entornos.  Los buckets de S3 conocidos autorizados para almacenar datos confidenciales se pueden excluir mediante el uso de una lista de permitidos en Macie.

4. Incorpore la identificación y la clasificación en sus procesos de desarrollo y prueba.

   a. Identifique las herramientas que los desarrolladores pueden usar para analizar los datos en busca de información confidencial mientras se están desarrollando las cargas de trabajo.  Utilice estas herramientas como parte de las pruebas de integración para recibir alertas cuando la presencia de datos confidenciales sea inesperada y evitar así continuar con la implementación.

5. Implemente un sistema o manual de procedimientos para tomar medidas cuando se encuentren datos confidenciales en ubicaciones no autorizadas.

   a. Restrinja el acceso a los datos mediante la corrección automática. Por ejemplo, puede mover estos datos a un bucket de S3 con acceso restringido o etiquetar el objeto si utiliza el control de acceso basado en atributos (ABAC). Además, considere la posibilidad de enmascarar los datos cuando se detecten.

b. Avise a sus equipos de protección de datos y respuesta a incidentes para que investiguen la causa raíz del incidente. Cualquier aprendizaje que identifique puede ayudar a prevenir futuros incidentes.

Recursos

Documentos relacionados:

- AWS Glue: Detect and process sensitive data
- Using managed data identifiers in Amazon SNS
- Amazon CloudWatch Logs: Help protect sensitive log data with masking

Ejemplos relacionados:

- Enabling data classification for Amazon RDS database with Macie
- Detecting sensitive data in DynamoDB with Macie

Herramientas relacionadas:

- Amazon Macie
- Amazon Comprehend
- Amazon Comprehend Medical
- AWS Glue

SEC07-BP04 Definición de la administración escalable del ciclo de vida de los datos

Comprenda los requisitos del ciclo de vida de sus datos en relación con sus diferentes niveles de clasificación y gestión de datos.  Entre ellos pueden estar la forma de gestionar los datos cuando entran por primera vez en su entorno, la manera de transformarlos y las reglas para su destrucción. Tenga en cuenta factores como los periodos de retención, el acceso, la auditoría y el seguimiento de la procedencia.

Resultado deseado: clasifica los datos lo más cerca posible del punto y la hora de la ingestión. Cuando la clasificación de datos requiera el enmascaramiento, la tokenización u otros procesos que reduzcan el nivel de confidencialidad, llevar a cabo estas acciones lo más cerca posible del punto y el momento de su ingesta.

Elimina los datos de acuerdo con su política cuando ya no resulte apropiado conservarlos en función de su clasificación.

Patrones comunes de uso no recomendados:

- Implementar un enfoque único para la administración del ciclo de vida de los datos, sin tener en cuenta los diferentes niveles de confidencialidad y los requisitos de acceso.
- Plantearse la administración del ciclo de vida únicamente desde la perspectiva de los datos utilizables o de los datos para los que existan copias de seguridad, pero no desde ambas perspectivas.
- Dar por sentado que los datos que han llegado a la carga de trabajo son válidos, sin determinar su valor o procedencia.
- Confiar en la durabilidad de los datos como alternativa a hacer copias de seguridad y protegerlos.
- Retener los datos más allá de su plazo de utilidad y del periodo de retención requerido.

Beneficios de establecer esta práctica recomendada: una estrategia de administración del ciclo de vida de los datos bien definida y escalable ayuda a mantener el cumplimiento normativo, mejora la seguridad de los datos, optimiza los costos de almacenamiento y mejora la eficiencia en el acceso a los datos y el intercambio de estos, mientras se mantienen los controles pertinentes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los datos de una carga de trabajo suelen ser dinámicos. La forma que adoptan al entrar en el entorno de una carga de trabajo puede ser diferente a la que adoptan cuando se almacenan o se usan en la lógica empresarial, los informes, el análisis o el machine learning. Además, su valor puede cambiar con el tiempo. Algunos datos son de naturaleza temporal y pierden valor con el paso del tiempo. Tenga en cuenta cómo afectan estos cambios de los datos a la evaluación planteada según su esquema de clasificación de datos y los controles asociados. Siempre que sea posible, utilice un mecanismo de ciclo de vida automatizado, como las políticas de ciclo de vida de Amazon S3 y Amazon Data Lifecycle Manager, para configurar los procesos de retención, archivado y caducidad de datos. Para los datos almacenados en DynamoDB, puede utilizar la característica Tiempo de vida (TTL) para definir una marca de tiempo de caducidad por artículo.

Distinga entre los datos que están disponibles para su uso y aquellos almacenados en copias de seguridad. Plantéese la posibilidad de utilizar AWS Backup para automatizar la copia de seguridad de los datos en todos los servicios de AWS. Las instantáneas de Amazon EBS ofrecen

una forma de copiar un volumen de EBS y almacenarlo mediante las funciones de S3, como el ciclo de vida, la protección de datos y el acceso a los mecanismos de protección. Dos de estos mecanismos son Bloqueo de objetos de S3 y Bloqueo de almacenes de AWS Backup, que pueden proporcionarle medidas de seguridad y control adicionales sobre sus copias de seguridad. Administre una separación clara de las funciones y el acceso a las copias de seguridad. Aísle las copias de seguridad de cuenta para mantener la separación del entorno afectado durante un evento.

Otro aspecto de la administración del ciclo de vida consiste en registrar el historial de los datos a medida que avanzan en la carga de trabajo, lo que se denomina seguimiento de la procedencia de los datos. Esto puede ofrecerle la confianza de saber de dónde provienen los datos, qué transformaciones se han hecho, qué propietario o proceso ha hecho esos cambios y cuándo los ha hecho.  Disponer de este historial ayuda a solucionar problemas y a llevar a cabo investigaciones durante posibles eventos de seguridad.  Por ejemplo, puede registrar los metadatos sobre las transformaciones en una tabla de Amazon DynamoDB.  Dentro de un lago de datos, puede guardar copias de los datos transformados en diferentes buckets de S3 para cada etapa de la canalización de datos. Almacene la información del esquema y la marca de tiempo en un AWS Glue Data Catalog.  Independientemente de cuál sea su solución, tenga en cuenta los requisitos de los usuarios finales a la hora de determinar las herramientas adecuadas que necesita para informar sobre la procedencia de sus datos.  Esto le ayudará a determinar la mejor manera de rastrear su procedencia.

Pasos para la implementación

1. Analice los tipos de datos, los niveles de confidencialidad y los requisitos de acceso de la carga de trabajo para clasificar los datos y definir las estrategias de administración del ciclo de vida adecuadas.
2. Diseñe e implemente políticas de retención de datos y procesos de destrucción automatizados que se ajusten a los requisitos legales, normativos y organizativos.
3. Establezca procesos y medidas de automatización para la supervisión, la auditoría y el ajuste continuos de las estrategias, los controles y las políticas de administración del ciclo de vida de los datos a medida que evolucionen los requisitos y las normativas de la carga de trabajo.
   a. Detecte los recursos que no tienen activada la administración automatizada del ciclo de vida con AWS Config.

Recursos

Prácticas recomendadas relacionadas:

- COST04-BP05 Aplicación de políticas de retención de datos

- SUS04-BP03 Uso de políticas para administrar el ciclo de vida de los conjuntos de datos

Documentos relacionados:

- Data Classification Whitepaper
- AWS Blueprint for Ransomware Defense
- DevOps Guidance: Improve traceability with data provenance tracking

Ejemplos relacionados:

- How to protect sensitive data for its entire lifecycle in AWS
- Build data lineage for data lakes using AWS Glue, Amazon Neptune, and Spline

Herramientas relacionadas:

- AWS Backup
- Administrador de vida útil de datos de Amazon
- AWS Identity and Access Management Access Analyzer

## SEC 8. ¿Cómo protege sus datos en reposo?

Proteja sus datos en reposo mediante la implementación de varios controles para reducir el riesgo de acceso no autorizado o mala gestión.

Prácticas recomendadas

- SEC08-BP01 Implementación de una administración de claves segura
- SEC08-BP02 Aplicación del cifrado en reposo
- SEC08-BP03 Automatización de la protección de los datos en reposo
- SEC08-BP04 Aplicación del control de acceso

### SEC08-BP01 Implementación de una administración de claves segura

La administración segura de claves incluye el almacenamiento, la rotación, el control de acceso y la supervisión del material de claves necesario para proteger los datos en reposo para su carga de trabajo.

Resultado deseado: tiene un mecanismo de administración de claves escalable, repetible y automatizado. El mecanismo hace cumplir el acceso con privilegios mínimos al material de claves y proporciona el equilibrio correcto entre la disponibilidad, la confidencialidad y la integridad de las claves. Supervisa el acceso a las claves y, si es necesario rotar el material de claves, las rota mediante un proceso automatizado. No permite que los operadores humanos accedan al material de claves.

Patrones comunes de uso no recomendados:

- Acceso humano a material de claves no cifrado.
- Crear algoritmos criptográficos personalizados.
- Permisos demasiado amplios para acceder a material de claves.

Beneficios de establecer esta práctica recomendada: al establecer un mecanismo de administración de claves seguro para su carga de trabajo, puede ayudar a proteger su contenido contra el acceso no autorizado. Además, es posible que esté sujeto a requisitos reglamentarios de cifrado de datos. Una solución de administración de claves eficaz puede proporcionar mecanismos técnicos alineados con esas regulaciones para proteger el material de claves.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El cifrado de los datos en reposo es un control de seguridad fundamental. Para implementar este control, su carga de trabajo necesita un mecanismo que almacene y administre de forma segura el material de claves utilizado para cifrar los datos en reposo.

AWS ofrece AWS Key Management Service (AWS KMS) para proporcionar un almacenamiento duradero, seguro y redundante para las claves de AWS KMS. Muchos servicios de AWS se integran con AWS KMS para respaldar el cifrado de sus datos. AWS KMS utiliza módulos de seguridad de hardware validados por la norma FIPS 140-3 de nivel 3 para proteger sus claves. No hay ningún mecanismo para exportar claves de AWS KMS en texto sin formato.

Si se implementan cargas de trabajo mediante una estrategia de cuentas múltiples, se considera una debe mantener las claves de AWS KMS en la misma cuenta que la carga de trabajo que las utiliza. En este modelo distribuido, la responsabilidad de administrar las claves de AWS KMS recae sobre su equipo. En otros casos de uso, su organización puede optar por almacenar las claves de AWS KMS en una cuenta centralizada. Esta estructura centralizada requiere políticas adicionales

para habilitar el acceso entre cuentas necesario para que la cuenta de carga de trabajo acceda a las claves almacenadas en la cuenta centralizada, pero puede ser más aplicable en casos de uso en los que una sola clave se comparte entre varias Cuentas de AWS.

Independientemente de dónde se almacene el material de claves, el acceso a la clave debe controlarse estrictamente mediante el uso de políticas de claves y políticas de IAM. Las políticas de claves son la forma principal de controlar el acceso a una clave de AWS KMS. Además, las concesiones de claves de AWS KMS pueden proporcionar acceso a servicios de AWS para cifrar y descifrar datos en su nombre. Revise las orientaciones de control de acceso a sus claves de AWS KMS.

Debería supervisar el uso de claves de cifrado para detectar patrones de acceso inusuales. Las operaciones hechas con claves administradas por AWS y claves administradas por el cliente almacenadas en AWS KMS pueden registrarse en AWS CloudTrail y deben revisarse periódicamente. Preste especial atención al monitoreo de los eventos de destrucción de claves. Para mitigar la destrucción accidental o malintencionada de material de claves, los eventos de destrucción de claves no eliminan el material de claves inmediatamente. Los intentos de eliminar claves de AWS KMS están sujetos a un periodo de espera predeterminado de 30 días y un mínimo de 7 días, lo que da tiempo a los administradores para revisar estas acciones y anular la solicitud si es necesario.

La mayoría de los servicios de AWS utilizan AWS KMS de forma transparente; su único requisito es decidir si desea utilizar una clave administrada por AWS o por el cliente. Si la carga de trabajo requiere el uso directo de AWS KMS para cifrar o descifrar datos, se recomienda utilizar cifrado de sobre para proteger los datos. La SDK de cifrado de AWS puede proporcionar a sus aplicaciones elementos básicos de cifrado del cliente para implementar el cifrado de sobre e integrarse con AWS KMS.

Pasos para la implementación

1. Determine las opciones de administración de claves apropiadas (administradas por AWS o administradas por el cliente) para la clave.
   a. Para facilitar el uso, AWS ofrece claves propias de AWS y administradas por AWS para la mayoría de los servicios, que proporcionan la capacidad de cifrado en reposo sin la necesidad de administrar el material de claves o las políticas de claves.
   b. Si utiliza claves administradas por el cliente, considere el almacén de claves predeterminado para ofrecer el mejor equilibrio entre agilidad, seguridad, soberanía de datos y disponibilidad. Otros casos de uso podrían exigir el uso de almacenes de claves personalizados con AWS CloudHSM o el almacén de clave externo.

2. Revise la lista de servicios que utiliza para su carga de trabajo para comprender cómo AWS KMS se integra con el servicio. Por ejemplo, las instancias de EC2 pueden usar volúmenes de EBS cifrados, que verifican que las instantáneas de Amazon EBS creadas a partir de esos volúmenes también estén cifradas mediante una clave administrada por el cliente y mitigan la divulgación accidental de datos de instantáneas no cifrados.
    a. How AWS services use AWS KMS
    b. Para obtener información detallada sobre las opciones de cifrado que ofrece un servicio de AWS, consulte el tema de cifrado en reposo en la guía del usuario o en la guía para desarrolladores del servicio.
3. Implemente AWS KMS: AWS KMS le permite crear y administrar fácilmente las claves y controlar el uso del cifrado en una gran variedad de servicios de AWS y en sus aplicaciones.
    a. Introducción: AWS Key Management Service (AWS KMS)
    b. Revise las prácticas recomendadas de control de acceso a sus claves de AWS KMS.
4. Puede usar el SDK de cifrado de AWS: utilice el SDK de cifrado de AWS con la integración de AWS KMS cuando la aplicación necesite cifrar datos del cliente.
    a. SDK de cifrado de AWS
5. Habilite el Analizador de acceso de IAM para revisar y notificar automáticamente si hay políticas de claves de AWS KMS demasiado amplias.
    a. Plantéese utilizar comprobaciones de políticas personalizadas para comprobar que una actualización de la política de recursos no concede acceso público a las claves de KMS.
6. Habilite Security Hub CSPM para recibir notificaciones si hay políticas de claves mal configuradas, claves programadas para su eliminación o claves sin la rotación automática habilitada.
7. Determine el nivel de registro adecuado para sus claves de AWS KMS. Como las llamadas a AWS KMS, incluidos los eventos de solo lectura, se registran, los registros de CloudTrail asociados con AWS KMS pueden resultar voluminosos.
    a. Algunas organizaciones prefieren dividir la actividad de registro de AWS KMS en una ruta distinta. Para obtener más detalles, consulte la sección de registro de llamadas a la API de AWS KMS con CloudTrail de la guía para desarrolladores de AWS KMS.

Recursos

Documentos relacionados:

- AWS Key Management Service

- AWS cryptographic services and tools
- Protecting Amazon S3 Data Using Encryption
- Cifrado de sobre
- Digital sovereignty pledge
- Demystifying AWS KMS key operations, bring your own key, custom key store, and ciphertext portability
- AWS Key Management Service cryptographic details

Videos relacionados:

- How Encryption Works in AWS
- Securing Your Block Storage on AWS
- AWS data protection: Using locks, keys, signatures, and certificates

Ejemplos relacionados:

- Implement advanced access control mechanisms using AWS KMS

SEC08-BP02 Aplicación del cifrado en reposo

Cifre los datos privados en reposo para mantener la confidencialidad y proporcionar una capa adicional de protección contra la divulgación o exfiltración de datos no deseada. El cifrado protege los datos para que no sea posible leerlos ni acceder a ellos sin antes descifrarlos. Haga un inventario y lleve un control de los datos no cifrados para mitigar los riesgos asociados a la exposición de los datos.

Resultado deseado: tiene mecanismos que cifran los datos privados de forma predeterminada cuando estén en reposo. Estos mecanismos ayudan a mantener la confidencialidad de los datos y proporcionan una capa adicional de protección contra la divulgación o exfiltración involuntaria de datos. Mantiene un inventario de datos no cifrados y comprende los controles que existen para protegerlos.

Patrones comunes de uso no recomendados:

- No utilizar configuraciones para que el cifrado se haga de forma predeterminada.
- Proporcionar un acceso demasiado permisivo a las claves de descifrado.

- No supervisar el uso de las claves de cifrado y descifrado.
- Almacenar datos sin cifrar.
- Utilizar la misma clave de cifrado para todos los datos, independientemente de su uso, tipos y clasificación.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Asigne claves de cifrado a clasificaciones de datos en sus cargas de trabajo. Este enfoque ayuda a proteger contra un acceso excesivamente permisivo si utiliza una única clave de cifrado, o muy pocas, para sus datos (consulte SEC07-BP01 Comprensión del esquema de clasificación de datos).

AWS Key Management Service (AWS KMS) se integra con muchos servicios de AWS para facilitar el cifrado de sus datos en reposo. Por ejemplo, en Amazon Elastic Compute Cloud (Amazon EC2) puede establecer el cifrado predeterminado en cuentas para que todos los volúmenes nuevos de EBS se cifren automáticamente. Cuando utilice AWS KMS, tenga en cuenta hasta qué punto es necesario restringir los datos. Las claves de AWS KMS predeterminadas y controladas por el servicio son administradas y utilizadas por AWS en su nombre. En el caso de los datos confidenciales que requieren un acceso detallado a la clave de cifrado subyacente, considere la posibilidad de usar claves administradas por el cliente (CMK). Tiene el control total sobre las CMK, incluida la rotación y la administración del acceso mediante el uso de políticas de claves.

Además, servicios como Amazon Simple Storage Service (Amazon S3) ahora cifran todos los objetos nuevos de forma predeterminada. Esta implementación proporciona una seguridad mejorada sin afectar al rendimiento.

Otros servicios, como Amazon Elastic Compute Cloud (Amazon EC2) o Amazon Elastic File System (Amazon EFS) admiten ajustes para el cifrado predeterminado. También puede utilizar Reglas de AWS Config para comprobar automáticamente que está utilizando cifrado para volúmenes de Amazon Elastic Block Store (Amazon EBS), instancias de Amazon Relational Database Service (Amazon RDS), buckets de Amazon S3 y otros servicios de su organización.

AWS también proporciona opciones para el cifrado del cliente, lo que le permite cifrar los datos antes de subirlos a la nube. El AWS Encryption SDK proporciona una forma de cifrar sus datos mediante el cifrado de sobre. Proporciona la clave de encapsulado y AWS Encryption SDK genera una clave de datos única para cada objeto de datos que cifra. Considere la posibilidad de utilizar AWS CloudHSM si necesita un módulo de seguridad de hardware (HSM) administrado de un solo

inquilino. AWS CloudHSM le permite generar, importar y administrar claves criptográficas en un HSM validado por FIPS 140-2 nivel 3. Entre los casos de uso de AWS CloudHSM, se incluye la protección de claves privadas para la emisión de una autoridad de certificación (CA) y la habilitación del cifrado de datos transparente (TDE) para bases de datos Oracle. El SDK de cliente de AWS CloudHSM proporciona software que le permite cifrar datos del cliente mediante claves almacenadas dentro de AWS CloudHSM antes de subir sus datos a AWS. El Cliente de encriptación de Amazon DynamoDB también le permite cifrar y firmar elementos antes de subirlos a una tabla de DynamoDB.

Pasos para la implementación

- Configuración del cifrado predeterminado para los nuevos volúmenes de Amazon EBS: especifique que desea que todos los volúmenes de Amazon EBS recién creados se creen de forma cifrada, con la opción de utilizar la clave predeterminada que proporciona AWS o una clave que cree.
- Configuración de imágenes de máquina de Amazon (AMI) cifradas: al copiar una AMI existente con cifrado habilitado, se cifran automáticamente las instantáneas y los volúmenes raíz.
- Configuración del cifrado de Amazon RDS: configure el cifrado para sus clústeres de base de datos e instantáneas en reposo de Amazon RDS mediante la opción de cifrado.
- Creación y configuración de claves de AWS KMS con políticas que limiten el acceso a las entidades principales adecuadas para cada clasificación de datos: por ejemplo, cree una clave de AWS KMS para cifrar los datos de producción y otra distinta para cifrar los datos de desarrollo o de prueba. También puede proporcionar acceso a la clave a otras Cuentas de AWS. Considere la posibilidad de tener cuentas diferentes para sus entornos de desarrollo y de producción. Si en su entorno de producción es necesario descifrar artefactos en la cuenta de desarrollo, puede editar la política de CMK que se utiliza para cifrar los artefactos de desarrollo para otorgar a la cuenta de producción la capacidad de descifrar dichos artefactos. Después, el entorno de producción puede ingerir los datos descifrados para usarlos en producción.
- Configuración del cifrado en servicios de AWS adicionales: para otros servicios de AWS que utilice, revise la documentación de seguridad de ese servicio para determinar las opciones de cifrado del servicio.

Recursos

Documentos relacionados:

- AWS Crypto Tools
- AWS Encryption SDK
- AWS KMS Cryptographic Details Whitepaper

- AWS Key Management Service
- AWS cryptographic services and tools
- Amazon EBS Encryption
- Default encryption for Amazon EBS volumes
- Encrypting Amazon RDS Resources
- ¿Cómo puedo habilitar el cifrado predeterminado para un bucket de Amazon S3?
- Protecting Amazon S3 Data Using Encryption

Videos relacionados:

- How Encryption Works in AWS
- Securing Your Block Storage on AWS

SEC08-BP03 Automatización de la protección de los datos en reposo

Utilice la automatización para validar y aplicar los controles de datos en reposo. Utilice el análisis automatizado para detectar errores de configuración de sus soluciones de almacenamiento de datos y, en la medida de lo posible, aplique las correcciones mediante una respuesta programática automatizada. Incorpore la automatización en sus procesos de CI/CD para detectar errores de configuración del almacenamiento de datos antes de que se implementen en producción.

Resultado deseado: los sistemas automatizados analizan y supervisan las ubicaciones de almacenamiento de datos para detectar los errores de configuración de los controles, el acceso no autorizado y el uso inesperado. La detección de ubicaciones de almacenamiento mal configuradas inicia las correcciones automatizadas. Los procesos automatizados crean copias de seguridad de los datos y almacenan copias inmutables fuera del entorno original.

Patrones comunes de uso no recomendados:

- No tener en cuenta las opciones de habilitar el cifrado de forma predeterminada, cuando sea posible.
- No tener en cuenta los eventos de seguridad, además de los eventos operativos, al formular una estrategia automatizada de copias de seguridad y recuperación.
- No aplicar la configuración de acceso público a los servicios de almacenamiento.
- No supervisar ni auditar los controles para proteger los datos en reposo.

Beneficios de establecer esta práctica recomendada: la automatización ayuda a prevenir el riesgo de configurar erróneamente las ubicaciones de almacenamiento de datos. También ayuda a evitar que los errores de configuración lleguen a los entornos de producción. Esta práctica recomendada también ayuda a detectar y corregir errores de configuración si se producen.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La automatización es una noción común a todas las prácticas de protección de los datos en reposo. SEC01-BP06 Automatización de la implementación de controles de seguridad estándares describe cómo puede plasmar la configuración de sus recursos en plantillas de infraestructura como código (IaC), por ejemplo, conAWS CloudFormation. Estas plantillas están confirmadas con un sistema de control de versiones y se utilizan para implementar recursos en AWS a través de una canalización de CI/CD. Estas técnicas también se aplican a la automatización de la configuración de sus soluciones de almacenamiento de datos, como la configuración de cifrado en los buckets de Amazon S3.

Puede comprobar la configuración que defina en sus plantillas de IaC para determinar si hay errores de configuración en sus canalizaciones de CI/CD mediante las reglas en AWS CloudFormation Guard. Puede supervisar los ajustes que aún no estén disponibles en CloudFormation u otras herramientas de IaC para detectar errores de configuración con AWS Config. Las alertas que Config genera por errores de configuración se pueden corregir automáticamente, tal como se describe en SEC04-BP04 Inicio de correcciones para recursos no conformes.

El uso de la automatización como parte de su estrategia de administración de permisos también es un componente integral de las protecciones de datos automatizadas. SEC03-BP02 Concesión de acceso con privilegios mínimos y SEC03-BP04 Reducción continua de los permisos describen la configuración de políticas de acceso con privilegios mínimos bajo la supervisión continua del AWS Identity and Access Management Access Analyzer para generar resultados cuando puedan reducirse los permisos. Además de la automatización de los permisos de supervisión, puede configurar Amazon GuardDuty para detectar comportamientos anómalos en el acceso a los datos de sus volúmenes de EBS (a través de una instancia de EC2), buckets de S3 y bases de datos de Amazon Relational Database Service compatibles.

La automatización también desempeña un papel a la hora de detectar cuándo se almacenan datos confidenciales en ubicaciones no autorizadas. SEC07-BP03 Automatización de la identificación y la clasificación describe cómo Amazon Macie puede supervisar sus buckets de S3 para detectar datos confidenciales inesperados y generar alertas que puedan iniciar una respuesta automática.

Siga las prácticas de REL09 Copia de seguridad de los datos para desarrollar una estrategia automatizada de copia de seguridad y recuperación de datos. La copia de seguridad y la recuperación de datos son tan importantes para la recuperación de los eventos de seguridad como para los eventos operativos.

Pasos para la implementación

1. Capture la configuración de almacenamiento de datos en plantillas de IaC.  Utilice comprobaciones automatizadas en sus canalizaciones de CI/CD para detectar errores de configuración.
    a. Para CloudFormation puede usar sus plantillas de IaC, y CloudFormation Guard para comprobar si hay errores de configuración en las plantillas.
    b. Utilice AWS Config para ejecutar reglas en un modo de evaluación proactiva. Use esta configuración para comprobar la conformidad de un recurso como uno de los pasos del proceso de CI/CD antes de crearlo.
2. Supervise los recursos en busca de errores de configuración de almacenamiento de datos.
    a. Configure AWS Config para supervisar los recursos de almacenamiento de datos con el fin de detectar cambios en las configuraciones de control y para generar alertas que invoquen acciones correctivas cuando se detecte un error de configuración.
    b. Consulte SEC04-BP04 Inicio de correcciones para recursos no conformes para obtener más información sobre las correcciones automatizadas.
3. Supervise y reduzca los permisos de acceso a los datos de forma continua mediante la automatización.
    a. El Analizador de acceso de IAM puede ejecutarse de forma continua para generar alertas cuando los permisos puedan reducirse.
4. Supervise los comportamientos anómalos de acceso a los datos y emita alertas si detecta alguno.
    a. GuardDuty vigila tanto las firmas de amenazas conocidas como las desviaciones de los comportamientos de acceso básicos para los recursos de almacenamiento de datos, como los volúmenes de EBS, los buckets de S3 y las bases de datos de RDS.
5. Supervise los datos confidenciales que se almacenan en ubicaciones inesperadas y emita alertas si detecta algún caso.
    a. Use Amazon Macie para analizar continuamente sus buckets de S3 en busca de datos confidenciales.
6. Automatice las copias de seguridad seguras y cifradas de sus datos.

a. AWS Backup es un servicio administrado que crea copias de seguridad de diferentes orígenes de datos en AWS. La Recuperación de desastres elástica le permite copiar cargas de trabajo completas del servidor y mantener una protección de datos continua con un objetivo de punto de recuperación (RPO) medido en segundos. Puede configurar ambos servicios para que funcionen en conjunto con el fin de automatizar la creación de copias de seguridad de datos y su almacenamiento en ubicaciones de conmutación por error. Esto puede ayudar a mantener sus datos disponibles cuando se vean afectados por eventos operativos o de seguridad.

Recursos

Prácticas recomendadas relacionadas:

- SEC01-BP06 Automatización de la implementación de controles de seguridad estándares
- SEC03-BP02 Concesión de acceso con privilegios mínimos
- SEC03-BP04 Reducción continua de los permisos
- SEC04-BP04 Inicio de correcciones para recursos no conformes
- SEC07-BP03 Automatización de la identificación y la clasificación
- REL09-BP02 Protección y cifrado de copias de seguridad
- REL09-BP03 Copias de seguridad automáticas de los datos

Documentos relacionados:

- AWS Prescriptive Guidance: Automatically encrypt existing and new Amazon EBS volumes
- Ransomware Risk Management on AWS Using the NIST Cyber Security Framework (CSF)

Ejemplos relacionados:

- How to use AWS Config proactive rules and AWS CloudFormation Hooks to prevent creation of noncompliant cloud resources
- Automate and centrally manage data protection for Amazon S3 with AWS Backup
- AWS re:Invent 2023 - Implement proactive data protection using Amazon EBS snapshots
- AWS re:Invent 2022 - Build and automate for resilience with modern data protection

Herramientas relacionadas:

- AWS CloudFormation Guard
- AWS CloudFormation Guard Rules Registry
- IAM Access Analyzer
- Amazon Macie
- AWS Backup
- Recuperación de desastres elástica

SEC08-BP04 Aplicación del control de acceso

Para ayudarlo a proteger sus datos en reposo, aplique el control de acceso mediante mecanismos como el aislamiento y el control de versiones. Aplique controles de acceso condicional y de privilegios mínimos. Impida que se conceda acceso público a sus datos.

Resultado deseado: verifica que solo los usuarios autorizados puedan acceder a los datos en función de su necesidad de utilizarlos. Protege sus datos con copias de seguridad periódicas y el control de versiones para evitar que se modifiquen o eliminen de forma intencionada o involuntaria. Aísla los datos críticos de otros datos para proteger la confidencialidad y la integridad.

Patrones comunes de uso no recomendados:

- Almacenar juntos datos con diferentes requisitos de confidencialidad o clasificación.
- Utilizar permisos demasiado permisivos en las claves de descifrado.
- Clasificar incorrectamente los datos.
- No conservar copias de seguridad detalladas de los datos importantes.
- Proporcionar acceso persistente a los datos de producción.
- No auditar el acceso a los datos ni revisar periódicamente los permisos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La protección de los datos en reposo es importante para mantener la integridad, la confidencialidad y el cumplimiento de los requisitos normativos. Puede implementar varios controles para lograrlo, como el control de acceso, el aislamiento, el acceso condicional y el control de versiones.

Puede aplicar el control de acceso con el principio de privilegios mínimos, que proporciona a los usuarios y los servicios únicamente los permisos necesarios para realizar sus tareas. Esto incluye el

acceso a las claves de cifrado. Revise sus políticas de AWS Key Management Service (AWS KMS) para comprobar que el nivel de acceso que concede es el adecuado y que se aplican las condiciones pertinentes.

Puede separar los datos en función de diferentes niveles de clasificación utilizando distintas Cuentas de AWS para cada nivel y administrar estas cuentas utilizando AWS Organizations. Este aislamiento puede ayudar a evitar el acceso no autorizado y minimizar el riesgo de exposición de los datos.

Revise con regularidad el nivel de acceso concedido en las políticas de buckets de Amazon S3. Evite el uso de buckets de lectura o escritura pública a menos que sea absolutamente necesario. Plantéese utilizar AWS Config para detectar buckets disponibles para el público y Amazon CloudFront para ofrecer contenido de Amazon S3. Verifique que los buckets que no deben permitir el acceso público estén configurados correctamente para impedirlo.

Implemente mecanismos de control de versiones y bloqueo de objetos para los datos críticos almacenados en Amazon S3. El control de versiones de Amazon S3 conserva las versiones anteriores de los objetos para recuperar los datos en caso de que se eliminen o sobrescriban accidentalmente. Bloqueo de objetos de Amazon S3 proporciona un control de acceso obligatorio para los objetos, lo que impide que se eliminen o sobrescriban, incluso por parte del usuario raíz, hasta que caduque el bloqueo. Además, Amazon Glacier Vault Lock ofrece una característica similar para los archivos almacenados en Amazon Glacier.

Pasos para la implementación

1. Aplique el control de acceso con el principio del privilegio mínimo:
   - Revise los permisos de acceso concedidos a los usuarios y servicios y compruebe que solo tienen los permisos necesarios para realizar sus tareas.
   - Revise el acceso a las claves de cifrado consultando las políticas de AWS Key Management Service (AWS KMS).
2. Separe los datos en función de los diferentes niveles de clasificación:
   - Utilice distintas Cuentas de AWS para cada nivel de clasificación de datos.
   - Administre estas cuentas mediante AWS Organizations.
3. Revise los permisos de buckets y objetos de Amazon S3:
   - Revise con regularidad el nivel de acceso concedido en las políticas de buckets de Amazon S3.
   - Evite el uso de buckets de lectura o escritura pública a menos que sea absolutamente necesario.
   - Valore la posibilidad de utilizar AWS Config para detectar buckets con disponibilidad pública.

- Use Amazon CloudFront para ofrecer contenido de Amazon S3.
- Verifique que los buckets que no deben permitir el acceso público estén configurados correctamente para impedirlo.
- Puede aplicar el mismo proceso de revisión a las bases de datos y a cualquier otro origen de datos que utilice la autenticación de IAM, como SQS o almacenes de datos de terceros.

4. Use AWS IAM Access Analyzer:
   - Puede configurar Analizador de acceso de IAM de AWS para analizar buckets de Amazon S3 y generar resultados cuando una política de S3 concede acceso a una entidad externa.
5. Implemente mecanismos de control de versiones y bloqueo de objetos:
   - Utilice el control de versiones de Amazon S3 para conservar las versiones anteriores de los objetos, lo que permite recuperarlos en caso de eliminaciones o sobrescrituras accidentales.
   - Utilice Bloqueo de objetos de Amazon S3 para añadir un control de acceso obligatorio para los objetos, con lo que ni siquiera el usuario raíz podrá eliminarlos o sobrescribirlos hasta que caduque el bloqueo.
   - Utilice Amazon Glacier Vault Lock para los archivos almacenados en Amazon Glacier.
6. Utilice el inventario de Amazon S:
   - Puede usar el inventario de Amazon S3 para auditar e informar sobre el estado de replicación y cifrado de sus objetos de S3.
7. Revise los permisos de uso compartido de Amazon EBS y AMI:
   - Revise los permisos de uso compartido de Amazon EBS y Uso compartido de AMI para comprobar que las imágenes y los volúmenes se compartan con Cuentas de AWS externas a su carga de trabajo.
8. Revise periódicamente los recursos compartidos de AWS Resource Access Manager:
   - Puede utilizar AWS Resource Access Manager para compartir recursos, como las políticas de AWS Network Firewall, las reglas de Amazon Route 53 Resolver y las subredes, dentro de sus Amazon VPC.
   - Audite periódicamente los recursos compartidos y deje de compartir los recursos que ya no sea necesario compartir.

Recursos

Prácticas recomendadas relacionadas:

- SEC03-BP01 Definición de los requisitos de acceso

- SEC03-BP02 Concesión de acceso con privilegios mínimos

Documentos relacionados:

- AWS KMS Cryptographic Details Whitepaper
- Introducción a la administración de permisos de acceso para los recursos de Amazon S
- Overview of managing access to your AWS KMS resources
- Reglas de AWS Config
- Amazon S3 + Amazon CloudFront: A Match Made in the Cloud
- Usar el control de versiones en buckets de S
- Usar Bloqueo de objetos de Amazon S
- Sharing an Amazon EBS Snapshot
- AMI compartidas
- Hosting a single-page application on Amazon S3
- AWS Claves de condición globales de
- Building a Data Perimeter on AWS

Videos relacionados:

- Securing Your Block Storage on AWS

## SEC 9. ¿Cómo protege sus datos en tránsito?

Proteja sus datos en tránsito mediante la implementación de varios controles para reducir el riesgo de acceso no autorizado o pérdida.

Prácticas recomendadas

- SEC09-BP01 Implementación de la administración segura de claves y certificados
- SEC09-BP02 Aplicación del cifrado en tránsito
- SEC09-BP03 Autenticación de las comunicaciones de red

### SEC09-BP01 Implementación de la administración segura de claves y certificados

Los certificados de seguridad de la capa de transporte (TLS) se utilizan para proteger las comunicaciones de red y establecer la identidad de los sitios web, los recursos y las cargas de trabajo a través de Internet, así como de las redes privadas.

Resultado deseado: un sistema de administración de certificados seguro que puede aprovisionar, implementar, almacenar y renovar certificados en una infraestructura de clave pública (PKI). Un mecanismo seguro de administración de claves y certificados evita que se divulgue el material de claves privadas del certificado y renueva automáticamente el certificado de forma periódica. También se integra con otros servicios para proporcionar comunicaciones de red e identidad seguras para los recursos de la máquina dentro de su carga de trabajo. Las identidades humanas nunca deben tener acceso al material de claves.

Patrones comunes de uso no recomendados:

- Seguir pasos manuales durante los procesos de implementación o renovación del certificado.
- No prestar suficiente atención a la jerarquía de la autoridad de certificación (CA) al diseñar una CA privada.
- Usar certificados autofirmados para recursos públicos.

Beneficios de establecer esta práctica recomendada:

- Simplificar la administración de certificados mediante la implementación y la renovación automatizadas
- Fomentar el cifrado de los datos en tránsito mediante certificados TLS
- Aumentar la seguridad y auditabilidad de las medidas de certificación adoptadas por la autoridad de certificación
- Organizar las tareas de administración en los diferentes capas de la jerarquía de CA

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

#### Guía para la implementación

Las cargas de trabajo modernas hacen un uso extensivo de las comunicaciones de red cifradas mediante protocolos PKI como TLS. La administración de certificados de PKI puede ser compleja, pero el aprovisionamiento, la implementación y la renovación automatizados de los certificados pueden reducir la fricción asociada con la administración de certificados.

AWS proporciona dos servicios para administrar los certificados de PKI de uso general: AWS Gestor de certificados y AWS Autoridad de certificación privada (AWS Privada CA). ACM es el servicio principal que los clientes utilizan para aprovisionar, administrar e implementar certificados para su uso tanto en cargas de trabajo de AWS tanto públicas como privadas. ACM emite certificados privados mediante AWS Privada CA y se integra con muchos otros servicios administrados de AWS para proporcionar certificados TLS seguros para las cargas de trabajo. ACM también puede emitir certificados de confianza pública de los servicios de confianza de Amazon. Los certificados públicos de ACM se pueden usar en cargas de trabajo públicas, ya que los navegadores y sistemas operativos modernos confían en estos certificados de forma predeterminada.

AWS Privada CA le permite establecer su propia autoridad de certificación raíz o subordinada y emitir certificados TLS a través de una API. Puede usar este tipo de certificados en situaciones en las que controla y administra la cadena de confianza en el lado del cliente de la conexión TLS. Además de los casos de uso de TLS, AWS Privada CA se puede utilizar para emitir certificados para pods de Kubernetes, atestaciones de productos de dispositivos Matter, firma de código y otros casos de uso con una plantilla personalizada. También puede utilizar IAM Roles Anywhere para proporcionar credenciales temporales de IAM a las cargas de trabajo en las instalaciones a las que se les hayan emitido certificados X.509 firmados por su CA privada.

Además de ACM y AWS Privada CA, AWS IoT Core proporciona soporte especializado para el aprovisionamiento, la administración y la implementación de certificados de PKI en dispositivos de IoT. AWS IoT Core proporciona mecanismos especializados para incorporar dispositivos de IoT en su infraestructura de clave pública a escala.

Algunos servicios de AWS, como Amazon API Gateway y Elastic Load Balancing, ofrecen sus propias capacidades de uso de certificados para proteger las conexiones de las aplicaciones. Por ejemplo, tanto API Gateway como Application Load Balancer (ALB) admiten el TLS mutuo (mTLS) mediante certificados de cliente que se crean y exportan mediante la Consola de administración de AWS, CLI o las API.

Consideraciones para establecer una jerarquía de CA privada

Si tiene que establecer una CA privada, es importante prestar especial atención para diseñar correctamente la jerarquía de CA desde el principio. Se recomienda implementar cada nivel de jerarquía de CA en Cuentas de AWS independientes al crear una jerarquía de CA privada. Este paso deliberado reduce el área de superficie de cada nivel de la jerarquía de CA, lo que facilita la detección de anomalías en los datos de registro de CloudTrail y reduce el alcance del acceso o el impacto si se produce un acceso no autorizado a una de las cuentas. La CA raíz debe residir

en su propia cuenta independiente y solo debe usarse para emitir uno o más certificados de CA intermedios.

A continuación, cree una o más CA intermedias en cuentas independientes de la cuenta de la CA raíz para emitir certificados para los usuarios finales, los dispositivos u otras cargas de trabajo. Por último, emita certificados desde su CA raíz a las CA intermedias, que a su vez emitirán certificados para sus usuarios finales o dispositivos. Para obtener más información sobre la planificación de la implementación de la CA y el diseño de la jerarquía de las CA, incluida la planificación de la resiliencia, la replicación entre regiones, el uso compartido de las CA en toda la organización y mucho más, consulte Planificación de la implementación de AWS Privada CA.

Pasos para la implementación

1. Determine los servicios de AWS pertinentes que necesita para su caso de uso:
    - Muchos casos de uso pueden utilizar la infraestructura de clave pública existente de AWS mediante AWS Gestor de certificados. ACM se puede usar para implementar certificados TLS para servidores web, equilibradores de carga u otros usos para certificados de confianza pública.
    - Considere AWS Privada CA cuando necesite establecer su propia jerarquía de autoridades de certificación privadas o necesite acceder a certificados exportables. ACM se puede utilizar entonces para emitir muchos tipos de certificados de entidad final mediante la AWS Privada CA.
    - Para los casos de uso en los que los certificados se deben aprovisionar a escala para dispositivos de Internet de las cosas (IoT) integrados, considere AWS IoT Core.
    - Puede utilizar la funcionalidad mTLS nativa en servicios como Amazon API Gateway o Application Load Balancer.
2. Implemente la renovación automática de certificados siempre que sea posible:
    - Utilice la renovación administrada de ACM para los certificados emitidos por ACM junto con los servicios administrados de AWS integrados.
3. Establezca registros y registros de auditoría:
    - Habilite los registros de CloudTrail para hacer un seguimiento del acceso a las cuentas que tienen autoridades de certificación. Considere configurar la validación de integridad del archivo de registro en CloudTrail para verificar la autenticidad de los datos de registro.
    - Genere y revise periódicamente informes de auditoría que enumeren los certificados que su CA privada ha emitido o revocado. Estos informes se pueden exportar a un bucket de S3.
    - Al implementar una CA privada, también tendrá que establecer un bucket de S3 para almacenar la lista de revocación de certificados (CRL). Para obtener instrucciones sobre cómo configurar

este bucket de S3 en función de los requisitos de su carga de trabajo, consulte Planificación de una lista de revocación de certificados (CRL).

Recursos

Prácticas recomendadas relacionadas:

- SEC02-BP02 Uso de credenciales temporales
- SEC08-BP01 Implementación de una administración de claves segura
- SEC09-BP03 Autenticación de las comunicaciones de red

Documentos relacionados:

- How to host and manage an entire private certificate infrastructure in AWS
- How to secure an enterprise scale ACM Private CA hierarchy for automotive and manufacturing
- Private CA best practices
- How to use AWS RAM to share your ACM Private CA cross-account

Videos relacionados:

- Activating AWS Gestor de certificados Private CA (workshop)

Ejemplos relacionados:

- Private CA workshop
- IOT Device Management Workshop (incluido el aprovisionamiento de dispositivos)

Herramientas relacionadas:

- Plugin to Kubernetes cert-manager to use AWS Privada CA

SEC09-BP02 Aplicación del cifrado en tránsito

Aplique los requisitos de cifrado definidos en función de las políticas, las obligaciones reglamentarias y las normas de su organización para ayudarle a cumplir los requisitos organizativos, legales y de cumplimiento. Utilice únicamente protocolos con cifrado cuando transmita datos confidenciales

fuera de su nube privada virtual (VPC). El cifrado ayuda a mantener la confidencialidad de los datos incluso cuando transitan por redes que no son de confianza.

Resultado deseado: el tráfico de red entre sus recursos e Internet debe cifrarse para mitigar el acceso no autorizado a los datos. Cifra el tráfico de red en su entorno de AWS interno de acuerdo con sus requisitos de seguridad. Cifra todos los datos en tránsito mediante protocolos TLS seguros y conjuntos de cifrado.

Patrones comunes de uso no recomendados:

- Utilizar versiones de SSL, TLS y componentes del conjunto de cifrado obsoletos (por ejemplo, SSL v3.0, claves RSA de 1024 bits y cifrado RC4).
- Permitir tráfico no cifrado (HTTP) hacia o desde recursos destinados al público.
- No supervisar y sustituir los certificados X.509 antes de que caduquen.
- Utilizar certificados X.509 autofirmados para TLS.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los servicios de AWS facilitan puntos de conexión HTTPS con TLS para la comunicación, lo que proporciona cifrado en tránsito al comunicarse con las API de AWS. Los protocolos no seguros, como HTTP, se pueden auditar y bloquear en una nube privada virtual (VPC) mediante el uso de grupos de seguridad. Las solicitudes HTTP también se pueden redirigir automáticamente a HTTPS en Amazon CloudFront o en un Equilibrador de carga de aplicación. Puede utilizar una política de buckets de Amazon Simple Storage Service (Amazon S3) para restringir la capacidad de cargar objetos a través de HTTP, lo que impone el uso de HTTPS para cargar objetos en sus buckets. Dispone de un control total sobre los recursos de computación para implementar el cifrado en tránsito en los servicios. También puede usar la conectividad de VPN en la VPC desde una red externa o AWS Direct Connect para facilitar el cifrado de tráfico. Compruebe que sus clientes hagan llamadas a las API de AWS mediante al menos TLS 1.2, ya que AWS va a dejar de utilizar versiones anteriores de TLS en febrero de 2024. Se recomienda utilizar TLS 1.3. Si tiene requisitos especiales para el cifrado en tránsito, puede encontrar soluciones de terceros disponibles en AWS Marketplace.

Pasos para la implementación

- Aplicación del cifrado en tránsito: los requisitos de cifrado definidos deben basarse en los últimos estándares y prácticas recomendadas, y solo permitir protocolos seguros. Por ejemplo, configure

un grupo de seguridad para permitir solamente el protocolo HTTPS a una instancia del equilibrador de carga de aplicaciones o una instancia de Amazon EC2.

- Configuración de protocolos seguros en los servicios de periferia: configure HTTPS con Amazon CloudFront y utilice un perfil de seguridad apropiado para su posición de seguridad y su caso de uso.
- Uso de una VPN para la conectividad externa: considere la posibilidad de utilizar una VPN IPsec para proteger las conexiones de punto a punto o de red a red para ofrecer tanto privacidad como integridad de los datos.
- Configuración de protocolos seguros en los equilibradores de carga: seleccione una política de seguridad que proporcione los conjuntos de cifrado más seguros que admitan los clientes que se conectarán al oyente. Cree un oyente HTTPS para su equilibrador de carga de aplicación.
- Configuración de protocolos seguros en Amazon Redshift: configure su clúster para que requiera una conexión de capa de sockets seguros (SSL) o de seguridad de la capa de transporte (TLS).
- Configuración de protocolos seguros: revise la documentación del servicio de AWS para determinar las capacidades de cifrado en tránsito.
- Configuración del acceso seguro al cargar en los buckets de Amazon S3: utilice los controles de políticas de buckets de Amazon S3 para aplicar el acceso seguro a los datos.
- Consideración de uso de AWS Gestor de certificados: ACM le permite aprovisionar, administrar e implementar certificados TLS públicos para utilizarlos con los servicios de AWS.
- Consideración de uso de AWS Autoridad de certificación privada para las necesidades de PKI privadas: AWS Privada CA le permite crear jerarquías de autoridades de certificación (CA) privadas para emitir certificados X.509 de entidad final que pueden utilizarse para crear canales TLS cifrados.

Recursos

Documentos relacionados:

- Uso de HTTPS con CloudFront
- Conectar la VPC a redes remotas mediante AWS Virtual Private Network
- Create an HTTPS listener for your Application Load Balancer
- Tutorial: Configure SSL/TLS on Amazon Linux 2
- Uso de SSL/TLS para cifrar una conexión a una instancia o clúster de base de datos
- Configuración de las opciones de seguridad para las conexiones

SEC09-BP03 Autenticación de las comunicaciones de red

Verifique la identidad de las comunicaciones mediante el uso de protocolos que admiten la autenticación, como la seguridad de la capa de transporte (TLS) o IPsec.

Diseñe su carga de trabajo para utilizar protocolos de red seguros y autenticados siempre que haya una comunicación entre servicios, aplicaciones o usuarios. El uso de protocolos de red que admiten autenticación y autorización proporciona un mayor control sobre los flujos de red y reduce la repercusión del acceso no autorizado.

Resultado deseado: una carga de trabajo con flujos de tráfico entre servicios bien definidos en el plano de datos y en el plano de control. Los flujos de tráfico utilizan protocolos de red autenticados y cifrados cuando es técnicamente posible.

Patrones comunes de uso no recomendados:

- Tener tráfico no cifrado o no autenticado en la carga de trabajo.
- Reutilizar credenciales de autenticación para varios usuarios o entidades.
- Confiar únicamente en los controles de red como mecanismo de control de acceso.
- Crear un mecanismo de autenticación personalizado en lugar de confiar en los mecanismos de autenticación estándar del sector.
- Tener un tráfico excesivamente permisivo entre los componentes del servicio u otros recursos de la VPC.

Beneficios de establecer esta práctica recomendada:

- Limita el alcance de la repercusión del acceso no autorizado a una parte de la carga de trabajo.
- Proporciona un nivel de garantía mayor de que las acciones solo las llevan a cabo entidades autenticadas.
- Mejora el desacoplamiento de los servicios al definir claramente las interfaces de transferencia de datos previstas y obligar a usarlas.
- Mejora la supervisión, el registro y la respuesta a los incidentes mediante la atribución de solicitudes y unas interfaces de comunicación bien definidas.
- Proporciona una defensa en profundidad para las cargas de trabajo al combinar los controles de red con los controles de autenticación y autorización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Los patrones de tráfico de red de la carga de trabajo se pueden clasificar en dos categorías:

- El tráfico este-oeste representa los flujos de tráfico entre los servicios que constituyen una carga de trabajo.
- El tráfico norte-sur representa los flujos de tráfico entre su carga de trabajo y los consumidores.

Aunque es una práctica común cifrar el tráfico norte-sur, es menos común proteger el tráfico este-oeste mediante protocolos autenticados. Las prácticas de seguridad modernas recomiendan que el diseño de red por sí solo no garantice una relación de confianza entre dos entidades. Cuando dos servicios pueden residir dentro de un límite de red común, sigue siendo una buena práctica recomendada cifrar, autenticar y autorizar las comunicaciones entre esos servicios.

Por ejemplo, las API del servicio de AWS utilizan el protocolo de firma AWS Signature Version 4 (SigV4) para autenticar a la persona que llama, independientemente de la red en la que se origine la solicitud. Esta autenticación garantiza que las API de AWS puedan verificar la identidad que solicitó la acción y, a continuación, esa identidad se pueda combinar con políticas para tomar una decisión de autorización que determine si la acción debe permitirse o no.

Servicios como Amazon VPC Lattice y Amazon API Gateway le permiten usar el mismo protocolo de firma SigV4 para incorporar autenticación y autorización al tráfico este-oeste en sus propias cargas de trabajo. Si los recursos fuera de su entorno de AWS necesitan comunicarse con servicios que requieren autenticación y autorización basadas en Sigv4, puede usar AWS Identity and Access Management (IAM) Roles Anywhere en el recurso que no es de AWS para adquirir credenciales de AWS temporales. Estas credenciales se pueden usar para firmar solicitudes de los servicios que utilizan SigV4 para autorizar el acceso.

Otro mecanismo común para autenticar el tráfico este-oeste es la autenticación mutua de TLS (mTLS). Muchas aplicaciones de internet de las cosas (IoT), aplicaciones de empresa a empresa y microservicios utilizan mTLS para validar la identidad de ambos lados de una comunicación TLS mediante el uso de certificados X.509 del lado del cliente y del lado del servidor. Estos certificados puede emitirlos AWS Autoridad de certificación privada (AWS Privada CA). Puede utilizar servicios como Amazon API Gateway para proporcionar autenticación mTLS para la comunicación entre cargas de trabajo o dentro de ellas. El Equilibrador de carga de aplicación también admite mTLS para cargas de trabajo internas o externas. Aunque mTLS proporciona información de autenticación para ambos lados de una comunicación TLS, no tiene un mecanismo de autorización.

Por último, OAuth 2.0 y OpenID Connect (OIDC) son dos protocolos que se suelen utilizar para controlar el acceso de los usuarios a los servicios, pero ahora también se están popularizando para el tráfico de servicio a servicio. API Gateway proporciona un autorizador de token web JSON (JWT) que permite a las cargas de trabajo restringir el acceso a las rutas de la API mediante JWT emitidas por proveedores de identidad OIDC u OAuth 2.0. Los ámbitos OAuth2 pueden utilizarse como fuente para tomar las decisiones de autorización básicas, pero las comprobaciones de autorizaciones siguen teniendo que implementarse en la capa de aplicación, y los ámbitos OAuth2 por sí solos no pueden satisfacer necesidades de autorización más complejas.

Pasos para la implementación

- Definición y documentación de los flujos de red de su carga de trabajo: el primer paso para implementar una estrategia de defensa en profundidad es definir los flujos de tráfico de la carga de trabajo.
  - Cree un diagrama de flujo de datos en el que se defina claramente cómo se transmiten los datos entre los diferentes servicios que componen su carga de trabajo. Este diagrama es el primer paso para imponer esos flujos a través de canales de red autentificados.
  - Instrumente su carga de trabajo en las fases de desarrollo y prueba para validar que el diagrama de flujo de datos refleje con precisión el comportamiento de la carga de trabajo en tiempo de ejecución.
  - Un diagrama de flujo de datos también puede ser útil cuando se lleva a cabo un ejercicio de modelado de amenazas, como se describe en SEC01-BP07 Identificación de amenazas y priorización de mitigaciones con un modelo de amenazas.
- Establecimiento de controles de red: considere la posibilidad de usar las capacidades de AWS para establecer controles de red que se ajusten a sus flujos de datos. Aunque los límites de la red no deberían ser el único control de seguridad, estos proporcionan una capa en la estrategia de defensa en profundidad para proteger su carga de trabajo.
  - Use grupos de seguridad para establecer, definir y restringir los flujos de datos entre los recursos.
  - Considere la posibilidad de usar AWS PrivateLink para comunicarse con servicios de AWS y de terceros compatibles con AWS PrivateLink. Los datos que se envían a través de un punto de conexión de la interfaz de AWS PrivateLink permanecen en la estructura de red de AWS y no atraviesan la Internet pública.
- Implementación de autenticación y autorización en todos los servicios de su carga de trabajo: elija el conjunto de servicios de AWS más adecuado para proporcionar flujos de tráfico autenticados y cifrados en su carga de trabajo.

  - Considere la posibilidad de usar Amazon VPC Lattice para proteger la comunicación de servicio a servicio. VPC Lattice puede usar la autenticación SigV4 combinada con políticas de autenticación para controlar el acceso de un servicio a otro.
  - Para la comunicación de servicio a servicio mediante mTLS, puede usar API Gateway o Equilibrador de carga de aplicación. AWS Privada CA se puede usar para establecer una jerarquía de CA privada capaz de emitir certificados para su uso con los mTLS.
  - Al hacer la integración con servicios que utilizan OAuth 2.0 u OIDC, considere la posibilidad de usar API Gateway con el autorizador JWT.
  - Para la comunicación entre la carga de trabajo y los dispositivos de IoT, considere la posibilidad de usar AWS IoT Core, que ofrece varias opciones para el cifrado y la autenticación del tráfico de red.
- Supervisión del acceso no autorizado: supervise continuamente los canales de comunicación no deseados, las entidades principales no autorizadas que intentan acceder a los recursos protegidos y otros patrones de acceso inadecuados.
  - Si utiliza VPC Lattice para administrar el acceso a sus servicios, piense en la posibilidad de habilitar y supervisar registros de acceso de VPC Lattice. Estos registros de acceso incluyen información sobre la entidad solicitante, información de red, incluida la VPC de origen y destino, y los metadatos de la solicitud.
  - Considere la posibilidad de habilitar registros de flujo de VPC para capturar los metadatos de los flujos de red y revisarlos periódicamente para detectar anomalías.
  - Consulte la AWS Security Incident Response Guide y la sección Respuesta ante incidentes del pilar de seguridad del Marco de AWS Well-Architected para obtener más información sobre la planificación, la simulación y la respuesta a los incidentes de seguridad.

Recursos

Prácticas recomendadas relacionadas:

- SEC03-BP07 Análisis del acceso público y entre cuentas
- SEC02-BP02 Uso de credenciales temporales
- SEC01-BP07 Identificación de amenazas y priorización de mitigaciones con un modelo de amenazas

Documentos relacionados:

- Evaluating access control methods to secure Amazon API Gateway APIs
- Configuring mutual TLS authentication for a REST API
- How to secure API Gateway HTTP endpoints with JWT authorizer
- Authorizing direct calls to AWS services using AWS IoT Core credential provider
- AWS Security Incident Response Guide

Videos relacionados:

- AWS re:invent 2022: Introducing VPC Lattice
- AWS re:invent 2020: Serverless API authentication for HTTP APIs on AWS

Ejemplos relacionados:

- Amazon VPC Lattice Workshop
- Taller “Zero-Trust Episode 1 – The Phantom Service Perimeter”

## Respuesta a incidentes

Pregunta

- SEC 10. ¿Cómo anticipa y se recupera de los incidentes y cómo responde ante ellos?

### SEC 10. ¿Cómo anticipa y se recupera de los incidentes y cómo responde ante ellos?

Incluso con controles eficaces de detección y prevención, la organización debería continuar implementando mecanismos para responder ante incidentes de seguridad y mitigar su posible impacto. Su preparación afecta considerablemente a la capacidad de los equipos de operar de forma eficaz durante un incidente, de aislar, contener y hacer una investigación forense de los problemas y de restaurar operaciones a un estado conocido correcto. La preparación de las herramientas y el acceso en previsión de un incidente de seguridad, así como la práctica periódica de la respuesta ante incidentes durante simulacros, lo ayudan a asegurarse de que podrá recuperarse con una interrupción mínima en el negocio.

Prácticas recomendadas

- SEC10-BP01 Identificación del personal clave y los recursos externos
- SEC10-BP02 Desarrollo de planes de administración de incidentes

- SEC10-BP03 Preparación de las capacidades forenses
- SEC10-BP04 Desarrollo y prueba de manuales de estrategias de respuesta a incidentes de seguridad
- SEC10-BP05 Aprovisionamiento previo del acceso
- SEC10-BP06 Implementación de las herramientas con anticipación
- SEC10-BP07 Ejecución de simulaciones
- SEC10-BP08 Establecimiento de un marco de trabajo para aprender de los incidentes

SEC10-BP01 Identificación del personal clave y los recursos externos

Identifique las obligaciones legales, el personal y los recursos internos y externos que puedan ayudar a su organización a responder ante un incidente.

Resultado deseado: cuenta con una lista del personal clave, su información de contacto y las funciones que desempeñan al responder a un evento de seguridad. Revisa esta información con regularidad y la actualiza para reflejar los cambios de personal desde la perspectiva de las herramientas internas y externas. Al documentar esta información, tener en cuenta a todos los vendedores y proveedores de servicios externos, incluidos los socios de seguridad, los proveedores de nube y las aplicaciones de software como servicio (SaaS). Durante un evento de seguridad, disponer de personal con el nivel adecuado de responsabilidad, contexto y acceso para poder responder y recuperarse.

Patrones comunes de uso no recomendados:

- No mantener una lista actualizada del personal clave con información de contacto, sus cargos y responsabilidades al responder a los eventos de seguridad.
- Dar por sentada una comprensión general de las personas, las dependencias, la infraestructura y las soluciones pertinentes a la hora de responder a un evento y recuperarse de él.
- No contar con un repositorio de documentos o conocimientos relacionados con la infraestructura clave o el diseño de aplicaciones.
- No disponer de procesos de incorporación adecuados para que los nuevos empleados contribuyan de manera eficaz a la respuesta a un evento de seguridad, como llevar a cabo simulacros de eventos.
- No disponer de una ruta de escalado para los casos en los que el personal clave no esté disponible temporalmente o no responda durante los eventos de seguridad.

Beneficios de establecer esta práctica recomendada: esta práctica reduce el tiempo de clasificación y respuesta que se dedica a identificar al personal adecuado y sus funciones durante un evento. También disminuye al mínimo la pérdida de tiempo durante un evento, ya que mantiene una lista actualizada del personal clave y sus cargos, de modo que pueda recurrir a las personas adecuadas para la clasificación y la recuperación de un evento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Identificación del personal clave de la organización: mantenga una lista de contactos del personal de su organización al que necesitaría involucrar. Revise y actualice periódicamente esta información en caso de que se produzcan cambios de personal, como cambios organizativos, ascensos y cambios en el equipo. Esto es especialmente importante para los puestos clave, como los administradores de incidentes, el personal de respuesta a incidentes y el líder de comunicaciones.

- Administrador de incidentes: los administradores de incidentes tienen la autoridad general durante la respuesta al evento.
- Personal de respuesta a incidentes: el personal de respuesta a incidentes es el responsable de las actividades de investigación y corrección. Estas personas pueden variar según el tipo de evento, pero suelen ser desarrolladores y miembros de los equipos de operaciones responsables de la aplicación afectada.
- Líder de comunicaciones: el líder de comunicaciones es responsable de las comunicaciones internas y externas, especialmente las destinadas a agencias públicas, organismos reguladores y clientes.
- Proceso de incorporación: forme e incorpore periódicamente a nuevos empleados a fin de que adquieran las habilidades y los conocimientos necesarios para contribuir de manera eficaz a los esfuerzos de respuesta a incidentes. Incorpore simulaciones y ejercicios prácticos como parte del proceso de incorporación para facilitar su preparación.
- Expertos en la materia (SME): en el caso de equipos distribuidos y autónomos, le recomendamos que identifique un SME para las cargas de trabajo críticas. Ofrecen información sobre el funcionamiento y la clasificación de datos de las cargas de trabajo críticas relacionadas con el evento.

Formato de tabla de ejemplo:

```
| Role | Name | Contact Information | Responsibilities |
```

```
1 | ——- | ——- | ——- | ——- |
2 | Incident Manager | Jane Doe| jane.doe@example.com | Overall authority during
 response |
3 | Incident Responder | John Smith | john.smith@example.com | Investigation and
 remediation |
4 | Communications Lead | Emily Johnson | emily.johnson@example.com | Internal and
 external communications |
5 | Communications Lead | Michael Brown | michael.brown@example.com | Insights on
 critical workloads |
```

Plantéese el uso de la característica Administrador de incidentes de AWS Systems Manager para determinar los contactos clave, definir un plan de respuesta, automatizar horarios de guardia y crear planes de escalado. Automatice y rote a todo el personal según un horario de guardias, de modo que la responsabilidad de la carga de trabajo se comparta entre los responsables de esta. Esto fomenta las prácticas recomendadas, como la creación de métricas y registros relevantes, y también la definición de los umbrales de alarma pertinentes para la carga de trabajo.

Identificación de los socios externos: las empresas utilizan herramientas creadas por proveedores de software independientes (ISV), socios y subcontratistas con el fin de desarrollar soluciones diferenciadoras para sus clientes. Implique al personal clave de estos colectivos que pueda ayudarle a responder y recuperarse de un incidente. Le recomendamos que se registre en el nivel adecuado de Soporte para poder acceder rápidamente a expertos en la materia de AWS a través de un caso de soporte. Plantéese la posibilidad de establecer acuerdos similares con todos los proveedores de soluciones críticas para las cargas de trabajo. Algunos eventos de seguridad requieren que las empresas que coticen en bolsa notifiquen el evento y sus impactos a los organismos públicos y entidades normativas pertinentes. Mantenga y actualice la información de contacto de los departamentos pertinentes y las personas responsables.

Pasos para la implementación

1. Configure una solución de administración de incidentes.
   a. Piense en implementar el Administrador de incidentes en su cuenta de Security Tooling.
2. Defina los contactos en su solución de administración de incidentes.
   a. Defina al menos dos tipos de canales de contacto para cada contacto (como SMS, teléfono o correo electrónico) para garantizar la accesibilidad durante un incidente.
3. Defina un plan de respuesta.
   a. Identifique los contactos más apropiados para interactuar durante un incidente. Defina planes de escalado alineados con los cargos del personal al que se va a recurrir, en lugar de con

los contactos individuales. Considere la posibilidad de incluir contactos que puedan ser responsables de informar a entidades externas, incluso aunque no participen directamente en la resolución del incidente.

Recursos

Prácticas recomendadas relacionadas:

- OPS02-BP03 Actividades operativas con propietarios identificados responsables de su rendimiento

Documentos relacionados:

- AWS Security Incident Response Guide

Ejemplos relacionados:

- AWS customer playbook framework
- Prepare for and respond to security incidents in your AWS environment

Herramientas relacionadas:

- AWS Systems Manager Incident Manager

Videos relacionados:

- Amazon's approach to security during development

SEC10-BP02 Desarrollo de planes de administración de incidentes

El primer documento que se desarrolla para la respuesta a incidentes es el plan de respuesta a incidentes. El plan de respuesta a incidentes está diseñado para ser la base de su programa y estrategia de respuesta a incidentes.

Beneficios de establecer esta práctica recomendada: desarrollar procesos de respuesta a incidentes exhaustivos y claramente definidos es clave para que el programa de respuesta a incidentes sea satisfactorio y escalable. Cuando se produce un evento de seguridad, tener unos pasos y flujos de trabajo claros puede ayudarle a responder a tiempo. Es posible que ya tenga procesos de respuesta

a incidentes. Independientemente de su estado actual, es importante actualizar, iterar y probar sus procesos de respuesta a incidentes con regularidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Un plan de administración de incidentes es fundamental para responder y mitigar el impacto potencial de los incidentes de seguridad, así como de cara a la recuperación. Un plan de administración de incidentes es un proceso estructurado para identificar y solucionar los incidentes de seguridad y responder a ellos en el momento oportuno.

La nube tiene muchos de los mismos roles y requisitos operativos que se encuentran en un entorno en las instalaciones. A la hora de crear un plan de administración de incidentes, es importante tener en cuenta las estrategias de respuesta y recuperación que mejor se ajusten al resultado empresarial y a los requisitos de conformidad. Por ejemplo, si trabaja con cargas de trabajo en AWS que cumplen con la normativa FedRAMP en Estados Unidos, siga las recomendaciones de la Guía de administración de seguridad informática NIST SP 800-61. Del mismo modo, cuando opere con cargas de trabajo que almacenan información de identificación personal (PII), plantéese cómo proteger y responder a los problemas relacionados con la residencia y el uso de datos.

Al crear un plan de administración de incidentes para sus cargas de trabajo en AWS, comience con el Modelo de responsabilidad compartida de AWS para crear un enfoque de defensa en profundidad para la respuesta a los incidentes. En este modelo, AWS administra la seguridad de la nube y el cliente es responsable de la seguridad en la nube. Esto significa que retiene el control y es responsable de los controles de seguridad que decida implementar. La Guía de respuesta ante incidentes de seguridad de AWS expone en detalle los conceptos clave y las orientaciones básicas para crear un plan de administración de incidentes centrado en la nube.

Un plan eficaz de administración de incidentes debe iterarse continuamente, lo que le permite mantenerse al día con su objetivo de operaciones en la nube. Considere la posibilidad de utilizar los planes de implementación que se detallan a continuación cuando cree y haga evolucionar su plan de administración de incidentes.

Pasos para la implementación

1. Defina las funciones y responsabilidades dentro de su organización para gestionar los eventos de seguridad. Aquí debería incluir a los representantes de varios departamentos, entre los que se incluyen:

- Recursos humanos (RR. HH.)
- Equipo ejecutivo
- Departamento legal
- Propietarios y desarrolladores de aplicaciones (expertos en la materia o SME)

2. Describa con claridad quién es responsable, encargado, consultado e informado (RACI) durante un incidente. Cree un diagrama RACI para facilitar una comunicación rápida y directa, y describa claramente el liderazgo en las diferentes etapas de un evento.
3. Incluya a los propietarios y desarrolladores de aplicaciones (SME) durante un incidente, ya que pueden proporcionar información y contexto que resultan valiosos para ayudar a medir el impacto. Entable relación con estos SME y practique con ellos escenarios de respuesta a incidentes antes de que se produzca un incidente real.
4. Incluya a socios de confianza o expertos externos en el proceso de investigación o de respuesta, ya que pueden ofrecer una mayor experiencia y amplitud de miras.
5. Acompase sus planes y funciones de administración de incidentes a cualquier normativa o requisito de cumplimiento local por los que se rija su organización.
6. Practique y pruebe sus planes de respuesta a incidentes con regularidad e incluya a todos los roles y responsabilidades definidos. Esto ayuda a agilizar el proceso y a verificar que se cuenta con una respuesta coordinada y eficiente a los incidentes de seguridad.
7. Revise y actualice los roles, las responsabilidades y el diagrama RACI periódicamente o a medida que cambien la estructura organizativa o los requisitos.

Información sobre los equipos de asistencia y respuesta de AWS

- AWS Support
  - Soporte ofrece una serie de planes que proporcionan acceso a herramientas y conocimientos que contribuyen al éxito y la salud operativa de sus soluciones de AWS. Si necesita asistencia técnica y más recursos para planificar, implementar y optimizar su entorno de AWS, puede seleccionar el plan de asistencia que mejor se adapte a su caso de uso de AWS.
  - Piense en el Centro de soporte de Consola de administración de AWS (es necesario iniciar sesión) como punto de contacto central para obtener asistencia en caso de problemas que afecten a sus recursos de AWS. El acceso a Soporte está controlado por AWS Identity and Access Management. Para obtener más información sobre el acceso a las características de Soporte, consulte Introducción a Soporte.
- AWS Equipo de respuesta a incidentes de clientes (CIRT) de)

  - El equipo de respuesta a incidentes de clientes (CIRT) de AWS es un equipo global de AWS especializado que ofrece asistencia a los clientes las 24 horas del día y los 7 días de la semana durante eventos de seguridad activos en el lado del cliente del Modelo de responsabilidad compartida de AWS.
  - Cuando el CIRT de AWS le ofrece asistencia, le ayuda en la clasificación y la recuperación de un evento de seguridad activo en AWS. Puede ayudarle a analizar la causa raíz mediante el uso de registros de servicio de AWS y ofrecerle recomendaciones para la recuperación. También puede proporcionar recomendaciones de seguridad y prácticas recomendadas para ayudarle a evitar eventos de seguridad en el futuro.
  - Los clientes de AWS pueden interactuar con el CIRT de AWS a través de un caso de Soporte.
- Respuesta frente a incidencias de seguridad de AWS
  - Anunciado en re:Invent 2024, Respuesta frente a incidencias de seguridad de AWS es un servicio administrado de respuesta a incidentes de seguridad que utiliza tanto tecnología moderna de clasificación como intervención humana. El servicio recopila todos los resultados de GuardDuty y cualquier resultado de terceros enviado para su clasificación, con el fin de alertar al cliente solo sobre los resultados que requieren una investigación. El servicio también proporciona un portal para enviar casos reactivos en caso de que el cliente detecte un incidente de seguridad y recibir asistencia del equipo avanzado de respuesta a incidentes de AWS.
- Asistencia en respuestas a DDoS
  - AWS ofrece AWS Shield, que ofrece un servicio administrado de protección contra ataques de denegación de servicio distribuidos (DDoS) que protege las aplicaciones web que se ejecutan en AWS. Shield proporciona una mitigación en línea automática y detección siempre activa que puede minimizar el tiempo de inactividad y la latencia de la aplicación, por lo que no es necesario disponer de Soporte para beneficiarse de la protección DDoS. Hay dos capas de Shield: AWS Shield Standard y AWS Shield Advanced. Para conocer las diferencias entre estos dos niveles, consulte la documentación de características de Shield.
- AWS Managed Services (AMS)
  - AWS Managed Services (AMS) proporciona una administración continua de su infraestructura de AWS para que pueda centrarse en sus aplicaciones. Mediante la implementación de prácticas recomendadas para mantener su infraestructura, AMS le ayuda a reducir la carga y el riesgo operativos. AMS automatiza actividades comunes, como solicitudes de cambios, supervisión, administración de parches, seguridad y servicios de copia de seguridad, y ofrece servicios de ciclo de vida completo para aprovisionar, ejecutar y brindar soporte a su infraestructura.

- AMS asume la responsabilidad de implementar un conjunto de controles de detección de seguridad y proporciona una primera línea de respuesta a las alertas las 24 horas del día y los 7 días de la semana. Cuando se inicia una alerta, AMS sigue un conjunto estándar de guías automáticas y manuales para verificar una respuesta coherente. Estas guías de estrategias se comparten con los clientes de AMS durante la incorporación para que puedan desarrollar y coordinar una respuesta con AMS.

Desarrollo del plan de respuesta a incidentes

El plan de respuesta a incidentes está diseñado para ser la base de su programa y estrategia de respuesta a incidentes. El plan de respuesta a incidentes debe figurar en un documento formal. Un plan de respuesta a incidentes suele incluir las siguientes secciones:

- Descripción general del equipo de respuesta a incidentes: describe los objetivos y las funciones del equipo de respuesta a incidentes.
- Funciones y responsabilidades: enumera las partes interesadas de la respuesta a los incidentes y detalla sus funciones cuando se produce un incidente.
- Un plan de comunicación: detalla la información de contacto y cómo se comunica durante un incidente.
- Métodos de comunicación auxiliares: se recomienda tener un método de comunicación auxiliar fuera de banda para informar de los incidentes. Un ejemplo de una aplicación que proporciona un canal de comunicaciones fuera de banda seguro es AWS Wickr.
- Fases de la respuesta a un incidente y medidas que tomar: se enumeran las fases de la respuesta a un incidente (por ejemplo, detección, análisis, erradicación, contención y recuperación), incluidas las medidas de alto nivel que se deben tomar en esas fases.
- Definiciones de gravedad y priorización del incidente: detalla cómo clasificar la gravedad de un incidente, cómo priorizar el incidente y, a continuación, cómo las definiciones de gravedad afectan a los procedimientos de escalamiento.

Aunque estas secciones son comunes en empresas de diferentes tamaños y de diferentes sectores, el plan de respuesta a incidentes de cada organización es único. Debe elaborar un plan de respuesta a incidentes que mejor se adapte a su organización.

Recursos

Prácticas recomendadas relacionadas:

- SEC04 Detección

Documentos relacionados:

- AWS Security Incident Response Guide
- NIST: guía de administración de incidentes de seguridad informática

SEC10-BP03 Preparación de las capacidades forenses

Antes de que se produzca un incidente de seguridad, considere la posibilidad de desarrollar capacidades forenses que lo ayuden a investigar los eventos de seguridad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Los conceptos de la ciencia forense tradicional que se utiliza en el entorno en las instalaciones también son aplicables a AWS. Para obtener información sobre cómo comenzar a desarrollar capacidades forenses en la Nube de AWS, consulte Forensic investigation environment strategies in the Nube de AWS.

Una vez que haya configurado la estructura del entorno y la Cuenta de AWS para el análisis forense, defina las tecnologías necesarias para ejecutar de forma eficaz unas metodologías sólidas desde el punto de vista forense en las cuatro fases:

- Recopilación: recopile registros de AWS pertinentes, como los registros de AWS CloudTrail, AWS Config, de flujo de VPC y de nivel de host. Siempre que sea posible, recopile instantáneas, copias de seguridad y volcados de memoria de los recursos de AWS afectados.
- Examen: examine los datos recopilados mediante la extracción y la evaluación de la información importante.
- Análisis: analice los datos recopilados para comprender el incidente y sacar conclusiones.
- Informes: presente la información resultante de la fase de análisis.

Pasos para la implementación

Preparación del entorno forense

AWS Organizations le permite administrar y gestionar un entorno de AWS de forma centralizada a medida que aumentan y se escalan los recursos de AWS. Una organización de AWS se encarga

de agrupar las cuentas de Cuentas de AWS para que pueda administrarlas como una sola unidad. Puede utilizar unidades organizativas para agrupar las cuentas que desee administrar como una sola unidad.

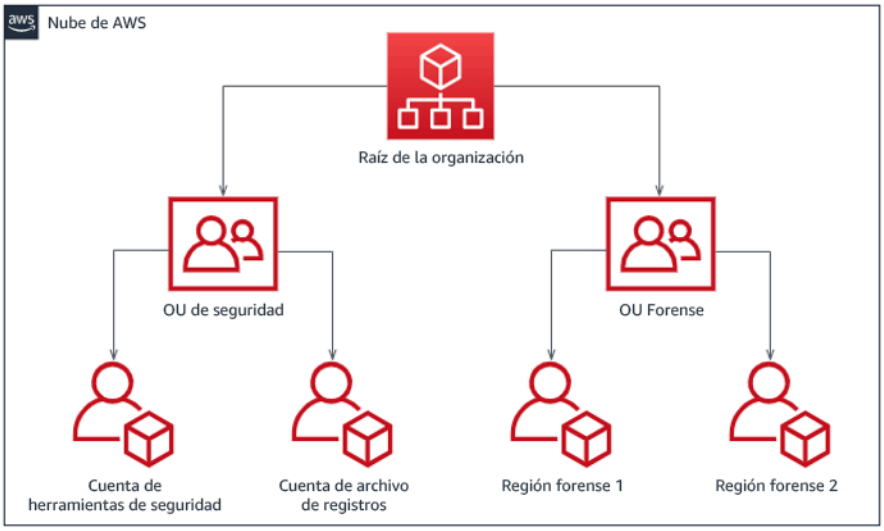
Para la respuesta a incidentes, es útil contar con una estructura de Cuenta de AWS que respalde las funciones de respuesta ante incidentes, lo que incluye una OU de seguridad y una OU forense. Dentro de la unidad organizativa de seguridad, debe tener cuentas para:

- Archivo de registros: agregue los registros en una Cuenta de AWS de archivo de registros con permisos limitados.
- Herramientas de seguridad: centralice los servicios de seguridad en una Cuenta de AWS de herramientas de seguridad. Esta cuenta funciona como un administrador delegado de los servicios de seguridad.

Dentro de la unidad organizativa forense, tiene la opción de implementar una o varias cuentas forenses diferentes para cada una de las regiones en las que opera, en función de lo que le venga mejor a su modelo empresarial y operativo. Si crea una cuenta forense para cada región, puede impedir que se creen recursos de AWS fuera de esa región y reducir el riesgo de que esos recursos se copien en una región no deseada. Por ejemplo, si solo opera en la región Este de EE. UU. (Norte de Virginia) (`us-east-1`) y Oeste de EE. UU. (Oregón) (`us-west-2`), tendría dos cuentas en la unidad organizativa forense: una para `us-east-1` y otra para `us-west-2`.

Puede crear una Cuenta de AWS forense para varias regiones. Debe tener cuidado al copiar los recursos de AWS en esa cuenta y asegurarse de que cumple los requisitos de soberanía de datos. Dado que aprovisionar nuevas cuentas lleva tiempo, es imperativo crear e instrumentar las cuentas forenses mucho antes de que se produzca un incidente para que los responsables puedan estar preparados y utilizarlas eficazmente en su respuesta.

En el siguiente diagrama, se muestra un ejemplo de una estructura de cuentas que incluye una unidad organizativa forense con cuentas forenses para cada región:




Estructura de cuentas por región para la respuesta a incidentes

Captura de copias de seguridad e instantáneas

Crear copias de seguridad de los principales sistemas y bases de datos es fundamental para poder recuperarse de un incidente de seguridad y para fines forenses. Con las copias de seguridad, puede restaurar los sistemas a su estado seguro anterior. En AWS, puede crear instantáneas de diversos recursos. Las instantáneas le proporcionan copias de seguridad puntuales de esos recursos. Hay muchos servicios de AWS que pueden ayudarle con la copia de seguridad y la recuperación. Para obtener más detalles sobre estos servicios y enfoques de copia de seguridad y recuperación, consulte la Backup and Recovery Prescriptive Guidance y Use backups to recover from security incidents.

Es esencial que las copias de seguridad estén bien protegidas, especialmente en ciertas situaciones, como el ransomware. Para obtener información sobre cómo proteger las copias de seguridad, consulte Top 10 security best practices for securing backups in AWS. Además de proteger las copias de seguridad, debe probar periódicamente los procesos de copia de seguridad y restauración para comprobar que la tecnología y los procesos que tiene implementados funcionan según lo previsto.

Automatización de los análisis forenses

Durante un evento de seguridad, es necesario que el equipo de respuesta a incidentes pueda recopilar y analizar las pruebas rápidamente y, al mismo tiempo, mantener la precisión durante todo el tiempo que rodee al evento (por ejemplo, capturar registros relacionados con un evento o recurso específico, o recopilar un volcado de memoria de una instancia de Amazon EC2). Para el equipo de respuesta a incidentes, resulta difícil y lleva mucho tiempo recopilar manualmente

las pruebas pertinentes, especialmente en una gran cantidad de instancias y cuentas. Además, la recopilación manual puede ser más propensa a errores humanos. Por estas razones, debe desarrollar e implementar la automatización del análisis forense en la medida que sea posible.

AWS ofrece una serie de recursos de automatización para el análisis forense, que se enumeran en la sección de recursos. Estos recursos son ejemplos de patrones forenses que hemos desarrollado y que los clientes han implementado. Aunque pueden resultar útiles como arquitectura de referencia al empezar, valore la posibilidad de modificarlos o crear nuevos patrones de automatización forense en función del entorno, los requisitos, las herramientas y los procesos forenses.

Recursos

Documentos relacionados:

- AWS Security Incident Response Guide - Develop Forensics Capabilities
- AWS Security Incident Response Guide - Forensics Resources
- Forensic investigation environment strategies in the Nube de AWS
- How to automate forensic disk collection in AWS
- AWS Prescriptive Guidance - Automate incident response and forensics

Videos relacionados:

- Automating Incident Response and Forensics

Ejemplos relacionados:

- Automated Incident Response and Forensics Framework
- Automated Forensics Orchestrator for Amazon EC2

SEC10-BP04 Desarrollo y prueba de manuales de estrategias de respuesta a incidentes de seguridad

Una parte esencial de la preparación de los procesos de respuesta a incidentes consiste en desarrollar manuales de estrategias. Los manuales de estrategias de respuesta a incidentes ofrecen directrices y pasos prescriptivos que deben seguirse cuando se produce un evento de seguridad. Contar con una estructura y unos pasos claros simplifica la respuesta y reduce la probabilidad de que se produzcan errores humanos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Deben crearse guías estratégicas para escenarios de incidentes, como, por ejemplo:

- Incidentes esperados: deben crearse manuales de estrategias para los incidentes que anticipe. Esto puede incluir amenazas como la denegación de servicio (DoS), el ransomware y las amenazas de las credenciales.
- Alertas o resultados de seguridad conocidos: deben crearse manuales de estrategias para abordar las alertas y los resultados de seguridad conocidos, como los resultados de Amazon GuardDuty. Cuando reciba un resultado de GuardDuty, el manual debería incluir medidas claras para evitar que la alerta se gestione mal o se ignore. Para obtener más información y orientación sobre las soluciones, consulte Corrección de problemas de seguridad detectados por GuardDuty.

Las guías estratégicas deben incluir los pasos técnicos que los analistas de seguridad deben completar para investigar y responder adecuadamente a un posible incidente de seguridad.

El equipo de respuesta a incidentes del cliente (CIRT) de AWS ha publicado un repositorio de GitHub que contiene manuales de estrategias de respuesta ante incidentes organizados por recurso, tipo y situación de amenaza. Estos manuales de estrategias se pueden adaptar para alinearlos con sus procedimientos de respuesta a incidentes existentes o servir como base para desarrollar otros nuevos.

Pasos para la implementación

Algunos de los elementos que deben incluirse en un manual de estrategias son los siguientes:

- Descripción general de la guía estratégica: ¿qué escenario de riesgo o incidente se aborda en este manual de estrategias? ¿Cuál es el objetivo del manual de estrategias?
- Requisitos previos: ¿qué registros, mecanismos de detección y herramientas automatizadas se necesitan en el escenario de este incidente? ¿Cuál es la notificación esperada?
- Información sobre la comunicación y la información de escalado: ¿quiénes participan y cuál es su información de contacto? ¿Cuáles son las responsabilidades de cada una de las partes interesadas?
- Medidas de respuesta: en las diferentes fases de respuesta a un incidente, ¿qué medidas tácticas se deben tomar? ¿Qué consultas deben ejecutar los analistas? ¿Qué código debe ejecutarse para lograr el resultado deseado?

  - Detección: ¿cómo se va a detectar el incidente?
  - Análisis: ¿cómo se va a determinar el alcance del impacto?
  - Contención: ¿cómo se va a aislar el incidente para limitar el alcance?
  - Erradicación: ¿cómo se va a eliminar la amenaza del entorno?
  - Recuperación: ¿cómo se va a conseguir que el sistema o recurso afectado vuelva a ser productivo?
- Resultados esperados: después de ejecutar las consultas y el código, ¿cuál es el resultado esperado de la guía estratégica?

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- SEC10-BP02 Desarrollo de planes de administración de incidentes

Documentos relacionados:

- Framework for Incident Response Playbooks
- Develop your own Incident Response Playbooks
- Incident Response Playbook Samples
- Building an AWS incident response runbook using Jupyter playbooks and CloudTrail Lake

SEC10-BP05 Aprovisionamiento previo del acceso

Verifique que haya aprovisionado previamente el acceso correcto a los equipos de intervención de incidentes en AWS para reducir el tiempo necesario de investigación hasta la recuperación.

Patrones comunes de uso no recomendados:

- Usar la cuenta raíz para la respuesta ante incidentes.
- Alterar las cuentas existentes.
- Manipular los permisos de IAM directamente al proporcionar un aumento puntual de los privilegios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

AWS recomienda reducir o eliminar la dependencia de credenciales de larga duración siempre que sea posible, en favor de credenciales temporales y mecanismos de aumento puntual de los privilegios. Las credenciales de larga duración están expuestas a riesgos de seguridad y aumentan la carga operativa. Para la mayoría de las tareas de administración, así como para las tareas de respuesta ante incidentes, le recomendamos que implemente la federación de identidades junto con el escalado temporal para el acceso administrativo. En este modelo, un usuario solicita el aumento a un nivel superior de privilegios (como un rol de respuesta ante incidentes) y, siempre que el usuario reúna los requisitos para el aumento, se envía una solicitud a un aprobador. Si se aprueba la solicitud, el usuario recibe un conjunto de credenciales de AWS temporales que puede utilizar para completar sus tareas. Una vez que caducan estas credenciales, el usuario debe enviar una nueva solicitud de aumento.

Recomendamos el uso del escalado temporal de privilegios en la mayoría de las situaciones de respuesta ante incidentes. La forma correcta de hacerlo es utilizar AWS Security Token Service y las políticas de sesión para limitar el acceso.

Hay situaciones en las que las identidades federadas no están disponibles; por ejemplo:

- Interrupción relacionada con un proveedor de identidades (IdP) comprometido.
- Una configuración deficiente o un error humano provocan la ruptura del sistema de administración de acceso federado.
- Actividad maliciosa como un evento de denegación de servicio distribuido (DDoS) o un sistema no disponible.

En los casos anteriores, debe haber un acceso de emergencia break glass configurado para permitir la investigación y la reparación oportuna de los incidentes. Se recomienda utilizar un usuario, grupo o rol con los permisos adecuados para llevar a cabo tareas y acceder a los recursos de AWS. Utilice el usuario raíz únicamente para llevar a cabo tareas que requieran credenciales de usuario raíz. Para verificar que los equipos de intervención de incidentes disponen del nivel correcto de acceso a AWS y otros sistemas pertinentes, recomendamos el aprovisionamiento previo de cuentas exclusivas. Las cuentas requieren un acceso con privilegios y se deben controlar y supervisar de forma estricta. Las cuentas deben crearse con el menor número de privilegios requeridos para llevar a cabo las tareas necesarias y el nivel de acceso debe basarse en las guías de estrategias creadas como parte del plan de administración de incidentes.

La práctica recomendada es crear usuarios y roles personalizados y exclusivos. El hecho de escalar temporalmente el acceso de los usuarios o de los roles mediante la incorporación de políticas de IAM provoca que no esté claro qué acceso tenían los usuarios durante el incidente y se corre el riesgo de que los privilegios escalados no se revoquen.

Es importante eliminar tantas dependencias como sea posible para verificar que se puede acceder en el mayor número posible de escenarios de error. Como medida de apoyo, cree una guía de estrategias para verificar que los usuarios de respuesta ante incidentes se crean como usuarios en una cuenta de seguridad exclusiva y no se administran a través de una federación existente o una solución de inicio de sesión único (SSO). Cada miembro del equipo de intervención debe tener su propia cuenta con nombre. La configuración de la cuenta debe aplicar una política de contraseñas seguras y la autenticación multifactor (MFA). Si las guías de estrategias de respuesta ante incidentes solo requieren acceso a la Consola de administración de AWS, el usuario no debería tener configuradas las claves de acceso y se le debería prohibir explícitamente la creación de claves de acceso. Esto se puede configurar con políticas de IAM o políticas de control de servicios (SCP) como se menciona en las prácticas recomendadas de seguridad de AWS para SCP de AWS Organizations. Los usuarios solo deben tener el privilegio de poder asumir roles de respuesta ante incidentes en otras cuentas.

Durante un incidente, podría ser necesario conceder acceso a otras personas internas o externas para respaldar las actividades de investigación, reparación o recuperación. En este caso, utilice el mecanismo de guía de estrategias mencionado anteriormente. Debe haber un proceso para verificar que cualquier acceso adicional se revoque inmediatamente después de que finalice el incidente.

Para verificar que el uso de los roles de respuesta ante incidentes se puede supervisar y auditar de forma adecuada, es esencial que las cuentas de IAM creadas para este fin no se compartan con otras personas y que el Usuario raíz de la cuenta de AWS no se utilice a menos que se requiera para tareas específicas. Si el usuario raíz es necesario (por ejemplo, no está disponible el acceso de IAM a una cuenta específica), utilice un proceso aparte con una guía de estrategias disponible para verificar la disponibilidad de las credenciales de inicio de sesión y el token MFA del usuario raíz.

Para configurar las políticas de IAM para los roles de respuesta ante incidentes, considere la posibilidad de utilizar el Analizador de acceso de IAM para generar políticas basadas en los registros de AWS CloudTrail. Para ello, conceda acceso de administrador al rol de respuesta ante incidentes en una cuenta que no sea de producción y ejecute las guías de estrategias. Una vez completado, se puede crear una política que únicamente permita las acciones hechas. Esta política se puede aplicar a los roles de respuesta ante incidentes en todas las cuentas. Es recomendable crear una política de IAM independiente para cada manual de estrategias a fin de facilitar la administración y la

auditoría. Entre los ejemplos de manuales de estrategias se podrían incluir planes de respuesta para ransomware, vulneraciones de datos, pérdida de acceso a la producción y otras situaciones.

Utilice las cuentas de respuesta ante incidentes para asumir los roles de IAM dedicados de respuesta ante incidentes en otras Cuentas de AWS. Estos roles se deben configurar para que solo puedan asumirlos los usuarios de la cuenta de seguridad. La relación de confianza debe requerir que la entidad principal de llamada se haya autenticado mediante MFA. Los roles deben utilizar políticas de IAM de ámbito estricto para controlar el acceso. Asegúrese de que todas las solicitudes de `AssumeRole` para estos roles estén registradas en CloudTrail y se haya alertado de ellas y que se registre cualquier acción hecha con estos roles.

Se recomienda que tanto las cuentas de IAM como los roles de IAM tengan nombres claros para poder encontrarlos fácilmente en los registros de CloudTrail. Un ejemplo sería asignar a las cuentas de IAM el nombre *<USER_ID>*`-BREAK-GLASS` y a los roles de IAM `BREAK-GLASS-ROLE`.

CloudTrail se utiliza para registrar la actividad de la API en sus cuentas de AWS y debe usarse para configurar alertas sobre el uso de las funciones de respuesta ante incidentes. Consulte la publicación del blog sobre la configuración de alertas cuando se utilizan claves de usuario raíz. Las instrucciones se pueden modificar para configurar la métrica de Amazon CloudWatch de filtro a filtro en los eventos de `AssumeRole` relacionados con el rol de IAM de respuesta ante incidentes:

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =
 "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !
= "AwsServiceEvent" }
```

Como es probable que los roles de respuesta ante incidentes tengan un nivel de acceso alto, es importante que estas alertas lleguen a un grupo amplio y se actúe con rapidez.

Durante un incidente, es posible que un miembro del equipo de intervención necesite acceder a sistemas que no están directamente protegidos por IAM. Puede tratarse de instancias de Amazon Elastic Compute Cloud, bases de datos de Amazon Relational Database Service o plataformas de software como servicio (SaaS). Se recomienda encarecidamente que, en lugar de utilizar protocolos nativos como SSH o RDP, AWS Systems Manager Session Manager se utilice para todos los accesos administrativos a las instancias de Amazon EC2. Este acceso se puede controlar mediante IAM, que es seguro y está auditado. También es posible automatizar partes de sus guías de estrategias mediante documentos de ejecución de comandos de AWS Systems Manager, lo que puede reducir los errores de los usuarios y mejorar el tiempo de recuperación. Para el acceso a las bases de datos y a las herramientas de terceros, recomendamos almacenar las credenciales de

acceso en AWS Secrets Manager y conceder el acceso a los roles de equipos de intervención ante incidentes.

Por último, la gestión de las cuentas de IAM de respuesta ante incidentes debe agregarse a sus procesos de incorporación, traslado y abandono, así como revisarse y probarse periódicamente para comprobar que solo se permite el acceso previsto.

Recursos

Documentos relacionados:

- Managing temporary elevated access to your AWS environment
- AWS Security Incident Response Guide
- AWS Elastic Disaster Recovery
- Administrador de incidentes de AWS Systems Manager
- Configuración de una política de contraseñas de la cuenta para usuarios de IAM
- Uso de autenticación multifactor (MFA) en AWS
- Configuring Cross-Account Access with MFA
- Using IAM Access Analyzer to generate IAM policies
- Best Practices for AWS Organizations Service Control Policies in a Multi-Account Environment
- How to Receive Notifications When Your AWS Account's Root Access Keys Are Used
- Create fine-grained session permissions using IAM managed policies
- Break glass access

Videos relacionados:

- Automating Incident Response and Forensics in AWS
- DIY guide to runbooks, incident reports, and incident response
- Prepare for and respond to security incidents in your AWS environment

SEC10-BP06 Implementación de las herramientas con anticipación

Asegúrese de que el personal de seguridad implementa las herramientas correctas con anticipación para reducir el plazo de investigación hasta conseguir la recuperación.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para automatizar las funciones de operaciones y de respuesta de seguridad, puede utilizar un completo conjunto de API y herramientas de AWS. Puede automatizar totalmente las funcionalidades de administración de identidades, seguridad de red, protección de datos y supervisión, y hacer que estén disponibles a través de métodos de desarrollo de software populares que ya tenga establecidos. Al crear procesos de automatización de seguridad, el sistema podrá supervisar, revisar e iniciar una respuesta, y no necesitará empleados que supervisen el nivel de seguridad y reaccionen manualmente a los eventos.

Si los equipos de intervención de incidentes siguen respondiendo a alertas de la misma forma, corren el riesgo de fatigarse por el excesivo número de alertas. Con el paso del tiempo, el equipo puede llegar a no reaccionar ante las alertas e incluso cometer errores durante la gestión de situaciones habituales o pasar por alto alertas inusuales. La automatización ayuda a evitar este problema con funciones que procesan alertas repetitivas y habituales, dejando a las personas que gestionen los incidentes extraordinarios y delicados. La integración de sistemas de detección de anomalías, como Amazon GuardDuty, AWS CloudTrail Insights y Detección de anomalías de Amazon CloudWatch, puede reducir la carga de alertas comunes basadas en umbrales.

Puede mejorar los procesos manuales automatizando los pasos del proceso mediante programación. Después de definir el patrón de solución de un evento, puede descomponer dicho patrón en una lógica procesable y escribir el código que ejecute dicha lógica. A continuación, los equipos de intervención pueden ejecutar ese código para solucionar el problema. Con el paso del tiempo, puede automatizar cada vez más pasos y, en última instancia, gestionar automáticamente todas las clases de incidentes comunes.

Durante una investigación de seguridad, necesitará poder revisar los registros correspondientes para registrar y comprender todo el alcance y la cronología del incidente. También necesita los registros para generar alertas que indican que se han producido determinadas acciones de interés. Es fundamental seleccionar, habilitar, almacenar y configurar mecanismos de consulta y recuperación, así como de alerta. Además, una forma eficaz de proporcionar herramientas para buscar datos de registro es usar Amazon Detective.

AWS tiene a su disposición más de 200 servicios en la nube y miles de características. Le recomendamos que revise los servicios que pueden respaldar y simplificar su estrategia de respuesta a incidentes.

Además de los registros, debe desarrollar e implementar una estrategia de etiquetado. El etiquetado puede ayudarle a proporcionar contexto en relación con el propósito de un recurso de AWS. El etiquetado también se puede utilizar en la automatización.

Pasos para la implementación

Selección y configuración de registros de análisis y alertas

Consulte la siguiente documentación sobre la configuración de registros para la respuesta a incidentes:

- Logging strategies for security incident response
- SEC04-BP01 Configuración del registro de servicios y aplicaciones

Activación de los servicios de seguridad para respaldar la detección y la respuesta

AWS ofrece funcionalidades de detección, prevención y respuesta, y se pueden utilizar otros servicios para diseñar soluciones de seguridad personalizadas. Para obtener una lista de los servicios más relevantes para la respuesta a incidentes de seguridad, consulte Definiciones de capacidades en la nube y la página de inicio de Respuesta a un incidente de seguridad.

Desarrollo e implementación de una estrategia de etiquetado

Puede resultar difícil obtener información contextual sobre el caso de uso empresarial y las partes interesadas internas pertinentes en relación con un recurso de AWS. Una forma de hacerlo es mediante etiquetas, que asignan metadatos a los recursos de AWS y se componen de una clave y un valor definidos por el usuario. Puede crear etiquetas para clasificar los recursos en función de su propósito, propietario, entorno, tipo de datos procesados y otros criterios de su elección.

Una estrategia de etiquetado coherente puede acelerar los tiempos de respuesta y minimizar el tiempo que se invierte en el contexto de la organización al permitirle identificar y discernir rápidamente la información contextual sobre un recurso de AWS. Las etiquetas también pueden servir como un mecanismo para iniciar automatizaciones de respuesta. Para obtener más información sobre qué etiquetar, consulte Tagging your AWS resources. Primero tendrá que definir las etiquetas que desea implementar en toda la organización. Después, implementará y hará cumplir la estrategia de etiquetado. Para obtener más información sobre la implementación y el cumplimiento, consulte Implement AWS resource tagging strategy using AWS Tag Policies and Service Control Policies (SCPs).

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- SEC04-BP01 Configuración del registro de servicios y aplicaciones
- SEC04-BP02 Recopilación de registros, resultados y métricas en ubicaciones estandarizadas

Documentos relacionados:

- Logging strategies for security incident response
- Definiciones de las capacidades de la nube de respuesta ante incidentes

Ejemplos relacionados:

- Threat Detection and Response with Amazon GuardDuty and Amazon Detective
- Security Hub Workshop
- Vulnerability Management with Amazon Inspector

SEC10-BP07 Ejecución de simulaciones

Las organizaciones crecen y evolucionan con el tiempo, pero también las amenazas, por lo que es importante que revise continuamente sus capacidades de respuesta a los incidentes. Ejecutar simulaciones (también conocidas como días de juego) es un buen método para llevar a cabo esta evaluación. En las simulaciones, se utilizan escenarios de eventos de seguridad reales diseñados para imitar las tácticas, técnicas y procedimientos (TTP) del actor de una amenaza y permiten a la organización probar y evaluar sus capacidades de respuesta a los incidentes respondiendo a estos simulacros de ataques cibernéticos tal y como podría ocurrir en la realidad.

Beneficios de establecer esta práctica recomendada: las simulaciones ofrecen una serie de beneficios:

- Comprobar si se está preparado para un ataque cibernético y mejorar la confianza de los equipos de respuesta a los incidentes.
- Probar la precisión y la eficiencia de las herramientas y los flujos de trabajo.
- Perfeccionar los métodos de comunicación y escalamiento en consonancia con su plan de respuesta a incidentes.

- Ofrecer la oportunidad de responder a vectores menos comunes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Hay tres tipos principales de simulaciones:

- Ejercicios prácticos: el enfoque de los ejercicios prácticos consiste en llevar a cabo una sesión de debate en la que participen las diversas partes interesadas en la respuesta a los incidentes para practicar las funciones y responsabilidades y utilizar las herramientas de comunicación y los manuales de estrategia establecidos. Por lo general, este ejercicio se puede hacer durante un día completo en un lugar virtual o físico, o bien en una combinación de ambos. Como se trata de un debate, el ejercicio de simulación se centra en los procesos, las personas y la colaboración. La tecnología forma parte integral del debate, pero en este tipo de ejercicio no se hace un uso real de las herramientas o los guiones de respuesta a incidentes.
- Ejercicios del equipo morado: los ejercicios del equipo morado aumentan el nivel de colaboración entre las personas que se encargan de la respuesta a los incidentes (equipo azul) y los actores de las amenazas simuladas (equipo rojo). El equipo azul está compuesto por miembros del centro de operaciones de seguridad (SOC), pero también puede incluir a otras partes interesadas que participarían durante un ataque cibernético real. El equipo rojo está compuesto por un equipo de pruebas de penetración o partes interesadas clave que cuentan con formación en seguridad ofensiva. El equipo rojo trabaja en colaboración con los facilitadores del ejercicio para diseñar un escenario que sea preciso y factible. Durante los ejercicios del equipo morado, la atención se centra en los mecanismos de detección, las herramientas y los procedimientos operativos estándar (SOP) que facilitan las iniciativas de respuesta a los incidentes.
- Ejercicios del equipo rojo: durante un ejercicio del equipo rojo, el atacante (equipo rojo) hace una simulación para lograr un determinado objetivo o conjunto de objetivos desde un ámbito predeterminado. Los defensores (equipo azul) no conocen necesariamente el ámbito y la duración del ejercicio; de esta manera, se consigue una evaluación más realista de cómo responderían ante un incidente real. Dado que los ejercicios de equipo rojo pueden ser pruebas invasivas, tenga cuidado e implemente controles para verificar que el ejercicio no produzca un daño real en su entorno.

Considere la posibilidad de llevar a cabo simulaciones de ataques cibernéticos con regularidad. Cada tipo de ejercicio puede aportar ventajas únicas para los participantes y la organización en su conjunto, por lo que puede optar por empezar con tipos de simulaciones menos complejos (como

los ejercicios prácticos) y pasar luego a los más complejos (ejercicios de equipo rojo). El tipo de simulación se debe elegir en función de su nivel de madurez en seguridad, sus recursos y los resultados deseados. Es posible que algunos clientes opten por no llevar a cabo los ejercicios de equipo rojo por su complejidad y su costo.

Pasos para la implementación

Independientemente del tipo de simulación que elija, las simulaciones suelen tener estos pasos de implementación:

1. Definición de los elementos básicos del ejercicio: defina el escenario de simulación y los objetivos de la simulación. Ambos deben contar con la aceptación de los directivos.
2. Identificación de las principales partes interesadas: como mínimo, en un ejercicio debe haber facilitadores y participantes. En función del escenario, podrían participar otras partes interesadas, como los directivos del departamento legal, de comunicaciones o ejecutivo.
3. Creación y prueba del escenario: es posible que sea necesario redefinir el escenario a medida que se crea si algunos elementos específicos no son factibles. Se espera que, al final de esta etapa, haya un escenario definitivo.
4. Facilitación de la simulación: el tipo de simulación determina la forma de llevarla a cabo (un escenario en papel o un escenario simulado muy técnico). Los facilitadores deben adaptar sus tácticas de facilitación a los objetivos del ejercicio y, siempre que sea posible, involucrar a todos los participantes del ejercicio para obtener la mayor ventaja.
5. Desarrollo del informe posterior a la acción (AAR): identifique las áreas que funcionaron bien, las que pueden mejorar y las posibles carencias. El AAR debe medir la eficacia de la simulación, así como la respuesta del equipo al evento simulado, de modo que se pueda seguir su progreso a lo largo del tiempo con futuras simulaciones.

Recursos

Documentos relacionados:

- Guía de respuestas ante incidentes de AWS

Videos relacionados:

- AWS GameDay - Security Edition
- Running effective security incident response simulations

SEC10-BP08 Establecimiento de un marco de trabajo para aprender de los incidentes

La implementación de un marco de trabajo sobre las lecciones aprendidas y una funcionalidad de análisis de la causa raíz no solo puede ayudar a mejorar las capacidades de respuesta a los incidentes, sino también a evitar que el incidente se repita. Al aprender de cada incidente, puede ayudar a evitar que se repitan los mismos errores, exposiciones o configuraciones incorrectas, lo que no solo mejorará el nivel de seguridad, sino también minimizará el tiempo que se pierde en situaciones evitables.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Es importante implementar un marco de trabajo sobre las lecciones aprendidas que establezca y logre, al más alto nivel, los puntos siguientes:

- ¿Cuándo se imparte una lección aprendida?
- ¿Qué implica el proceso de lecciones aprendidas?
- ¿Cómo se lleva a cabo una lección aprendida?
- ¿Quién participa en el proceso y cómo?
- ¿Cómo se van a identificar las áreas de mejora?
- ¿Cómo se va a garantizar que las mejoras se supervisan e implementan de manera efectiva?

El marco no debe centrarse en las personas ni en buscar culpables, sino en mejorar las herramientas y los procesos.

Pasos para la implementación

Además de los resultados generales enumerados anteriormente, es importante asegurarse de que se hacen las preguntas correctas para obtener el máximo valor del proceso (información que conduzca a mejoras viables). Considere la posibilidad de usar estas preguntas para fomentar el debate sobre las lecciones aprendidas:

- ¿Cuál fue el incidente?
- ¿Cuándo se identificó por primera vez el incidente?
- ¿Cómo se identificó?
- ¿Qué sistemas alertaron sobre la actividad?

- ¿Qué sistemas, servicios y datos estaban involucrados?
- ¿Qué ocurrió exactamente?
- ¿Qué funcionó correctamente?
- ¿Qué no funcionó correctamente?
- ¿Qué procesos o procedimientos fallaron o no se lograron escalar para responder al incidente?
- ¿Qué se puede mejorar en las siguientes áreas?:
  - People
    - ¿Las personas a las que había que contactar estaban realmente disponibles y la lista de contactos estaba actualizada?
    - ¿A las personas les faltaba formación o capacidades necesarias para responder e investigar el incidente de manera eficaz?
    - ¿Los recursos adecuados estaban listos y disponibles?
  - Proceso
    - ¿Se siguieron los procesos y los procedimientos?
    - ¿Los procesos y procedimientos para este (tipo de) incidente estaban documentados y disponibles?
    - ¿Faltaba algún proceso y procedimiento necesario?
    - ¿Los encargados de responder al incidente pudieron acceder oportunamente a la información necesaria para responder al problema?
  - Tecnología
    - ¿Los sistemas de alerta existentes identificaron la actividad y alertaron sobre ella eficazmente?
    - ¿Cómo podríamos haber reducido el tiempo de detección en un 50 %?
    - ¿Es necesario mejorar las alertas existentes o crear nuevas alertas para este (tipo de) incidente?
    - ¿Las herramientas existentes permitían investigar (buscar o analizar) el incidente de forma eficaz?
    - ¿Qué se puede hacer para poder identificar antes este (tipo de) incidente?
    - ¿Qué se puede hacer para ayudar a evitar que este (tipo de) incidente vuelva a ocurrir?
    - ¿Quién es el responsable del plan de mejora y cómo comprobará que se ha implementado?
    - ¿Qué plazos hay para implementar y probar otros procesos y controles preventivos o de supervisión?

Esta lista no incluye todas las posibilidades. Solo pretende servir como punto de partida para identificar cuáles son las necesidades de la organización y la empresa, y cómo se pueden analizar para aprender lo mejor posible de los incidentes y aumentar continuamente el nivel de seguridad. Lo más importante es empezar incorporando las lecciones aprendidas como un componente estándar del proceso de respuesta a incidentes, la documentación y las expectativas de las partes interesadas.

Recursos

Documentos relacionados:

- AWS Security Incident Response Guide - Establish a framework for learning from incidents
- NCSC CAF guidance - Lessons learned

## Seguridad de las aplicaciones

Pregunta

- SEC 11. ¿Cómo incorpora y valida las propiedades de seguridad de las aplicaciones durante el ciclo de vida de diseño, desarrollo e implementación?

### SEC 11. ¿Cómo incorpora y valida las propiedades de seguridad de las aplicaciones durante el ciclo de vida de diseño, desarrollo e implementación?

La capacitación de los usuarios, las pruebas mediante automatización, el conocimiento de las dependencias y la validación de las propiedades de seguridad de herramientas y aplicaciones contribuyen a reducir la probabilidad de que se produzcan problemas de seguridad en las cargas de trabajo de producción.

Prácticas recomendadas

- SEC11-BP01 Formación en seguridad de las aplicaciones
- SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento
- SEC11-BP03 Pruebas de penetración periódicas
- SEC11-BP04 Revisiones del código de conducta
- SEC11-BP05 Centralización de servicios para paquetes y dependencias

- SEC11-BP06 Implementación de software mediante programación
- SEC11-BP07 Evaluación periódica de las propiedades de seguridad de las canalizaciones
- SEC11-BP08 Creación de un programa que integre la propiedad de la seguridad en los equipos de la carga de trabajo

### SEC11-BP01 Formación en seguridad de las aplicaciones

Forme a su equipo en prácticas de desarrollo y funcionamiento seguros, lo que los ayudará a crear software seguro y de alta calidad. Esta práctica ayuda a su equipo a prevenir, detectar y solucionar los problemas de seguridad en una fase temprana del ciclo de vida del desarrollo. Puede plantearse impartir formación sobre el modelado de amenazas, las prácticas de codificación segura y el uso de servicios para una configuración y funcionamiento seguros. Proporcione a su equipo acceso a la formación mediante recursos de autoservicio y recopile periódicamente sus comentarios para mejorar continuamente.

Resultado deseado: dota a su equipo de los conocimientos y las habilidades necesarios para diseñar y crear software teniendo en cuenta la seguridad desde el principio. Gracias a la formación sobre modelos de amenazas y prácticas de desarrollo seguro, su equipo tiene un profundo conocimiento de los posibles riesgos de seguridad y de cómo mitigarlos durante el ciclo de vida del desarrollo del software (SDLC). Este enfoque proactivo de la seguridad forma parte de la cultura de su equipo, por lo que podrá identificar y solucionar los posibles problemas de seguridad desde el principio. Como resultado, su equipo ofrece software y características seguras y de alta calidad de manera más eficiente, lo que acelera el plazo general de entrega. Su organización tiene una cultura de seguridad colaborativa e inclusiva, en la que todos los creadores comparten la propiedad de la seguridad.

Patrones comunes de uso no recomendados:

- Espera a una revisión de seguridad para estudiar las propiedades de seguridad de un sistema.
- Deja todas las decisiones de seguridad en manos del equipo de seguridad.
- No comunica claramente cómo se relacionan las decisiones tomadas en el SDLC con las expectativas o políticas generales de seguridad de la organización.
- Interviene demasiado tarde en el proceso de revisión de la seguridad.

Beneficios de establecer esta práctica recomendada:

- Entender mejor los requisitos de la organización en materia de seguridad en una fase temprana del ciclo de desarrollo.

- Poder identificar y corregir más rápidamente los posibles problemas de seguridad, lo que se traduce en una entrega más rápida de las características.
- Mejora de la calidad del software y los sistemas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para desarrollar software seguro y de calidad, forme a su equipo en prácticas habituales de desarrollo y funcionamiento seguros de las aplicaciones. Esta práctica puede ayudar a su equipo a prevenir, detectar y solucionar los problemas de seguridad en una fase temprana del ciclo de vida del desarrollo, lo que puede acelerar sus plazos de entrega.

Para que esta práctica dé resultado, considere la posibilidad de formar a su equipo en el modelado de amenazas mediante recursos de AWS, como el taller de modelado de amenazas. El modelado de amenazas puede ayudar a su equipo a comprender los posibles riesgos de seguridad y a diseñar sistemas teniendo en cuenta la seguridad desde el principio. Además, puede proporcionar acceso a formación de Formación de AWS and Certification, del sector o de los socios de AWS en prácticas de desarrollo seguras. Para obtener más información sobre un enfoque integral de diseño, desarrollo, protección y funcionamiento eficiente a escala, consulte la Guía de AWS DevOps.

Defina y comunique con claridad el proceso de revisión de la seguridad de su organización y describa las responsabilidades de su equipo, el equipo de seguridad y otras partes interesadas. Publique guías de autoservicio, ejemplos de código y plantillas con las que demuestre cómo cumplir sus requisitos de seguridad. Puede utilizar servicios de AWS como AWS CloudFormation, Constructos de Kit de desarrollo de la nube de AWS (AWS CDK) (AWS CDK) y Service Catalog para proporcionar configuraciones seguras y aprobadas previamente, y reducir la necesidad de configuraciones personalizadas.

Obtenga periódicamente comentarios de su equipo sobre su experiencia con el proceso de formación y revisión de la seguridad, y utilícelos para mejorar continuamente. Lleve a cabo días de juego o campañas de detección de errores para identificar y abordar los problemas de seguridad y, al mismo tiempo, mejorar las habilidades de su equipo.

Pasos para la implementación

1. Identifique las necesidades de formación: evalúe el nivel actual de habilidades y las brechas de conocimiento de su equipo en relación con las prácticas de desarrollo seguras mediante encuestas, revisiones de código o conversaciones con los miembros del equipo.

2. Planifique la formación: en función de las necesidades identificadas, cree un plan de formación que abarque temas relevantes, como el modelado de amenazas, las prácticas de codificación segura, las pruebas de seguridad y las prácticas de implementación segura. Emplee recursos como el taller de modelado de amenazas, Formación de AWS and Certification, y los programas de formación del sector o de los socios de AWS.
3. Programe e imparta formación: programe sesiones de formación o talleres periódicos para su equipo. Pueden ser impartidos por un instructor o a su propio ritmo, según las preferencias y la disponibilidad de su equipo. Haga hincapié en los ejercicios de aplicación de lo aprendido y en los ejemplos prácticos para reforzar el aprendizaje.
4. Defina un proceso de revisión de la seguridad: colabore con su equipo de seguridad y otras partes interesadas para definir claramente el proceso de revisión de la seguridad de las aplicaciones. Documente las responsabilidades de cada equipo o persona que participa en el proceso, incluidos el equipo de desarrollo, el equipo de seguridad y otras partes interesadas relevantes.
5. Cree recursos de autoservicio: elabore guías de autoservicio, ejemplos de código y plantillas que demuestren cómo cumplir los requisitos de seguridad de su organización. Puede usar servicios de AWS como CloudFormation, Constructos de AWS CDK y Service Catalog para proporcionar configuraciones seguras previamente aprobadas y reducir la necesidad de configuraciones personalizadas.
6. Comuníquese y socialice: comunique de manera eficaz a su equipo el proceso de revisión de la seguridad y los recursos de autoservicio disponibles. Lleve a cabo sesiones de formación o talleres para que se familiaricen con estos recursos y compruebe que entienden cómo usarlos.
7. Obtenga comentarios y mejore: obtenga periódicamente comentarios de su equipo sobre su experiencia con el proceso de formación y revisión de la seguridad. Utilice estos comentarios para identificar las áreas de mejora y perfeccionar continuamente los materiales de formación, los recursos de autoservicio y el proceso de revisión de la seguridad.
8. Realice ejercicios de seguridad: organice jornadas de juego o campañas de detección de errores para identificar y abordar los problemas de seguridad de las aplicaciones. Estos ejercicios no solo ayudan a descubrir posibles vulnerabilidades, sino que también sirven como oportunidades prácticas de aprendizaje para su equipo, ya que mejoran sus habilidades de desarrollo y funcionamiento seguros.
9. Siga aprendiendo y mejorando: anime a su equipo a mantenerse al día con las últimas prácticas, herramientas y técnicas de desarrollo seguro. Revise y actualice periódicamente los materiales y recursos de formación para reflejar la evolución del panorama de la seguridad y las prácticas recomendadas.

Recursos

Prácticas recomendadas relacionadas:

- SEC11-BP08 Creación de un programa que integre la propiedad de la seguridad en los equipos de la carga de trabajo

Documentos relacionados:

- Formación de AWS and Certification
- How to think about cloud security governance
- How to approach threat modeling
- Accelerating training – The AWS Skills Guild
- AWS Sagas de DevOps

Videos relacionados:

- Proactive security: Considerations and approaches

Ejemplos relacionados:

- Workshop on threat modeling
- Industry awareness for developers

Servicios relacionados:

- AWS CloudFormation
- Kit de desarrollo de la nube de AWS (AWS CDK) (AWS CDK) Constructs
- Service Catalog

SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento

Automatice las pruebas de las propiedades de seguridad a lo largo del ciclo de vida de desarrollo y lanzamiento. La automatización facilita la identificación coherente y repetible de posibles problemas en el software antes de su lanzamiento, lo que reduce el riesgo de problemas de seguridad en el software suministrado.

Resultado deseado: el objetivo de las pruebas automatizadas es proporcionar una forma programática de detectar posibles problemas de forma temprana y frecuente a lo largo del ciclo de vida del desarrollo. Al automatizar las pruebas de regresión, puede volver a ejecutar pruebas funcionales y no funcionales para verificar que el software probado previamente siga funcionando como se esperaba después de un cambio. Cuando se definen pruebas unitarias de seguridad para detectar errores de configuración habituales, como autenticación dañada o ausente, es posible identificar y solucionar estos problemas en una fase temprana del proceso de desarrollo.

La automatización de pruebas utiliza casos de prueba creados específicamente para la validación de aplicaciones, basados en los requisitos de la aplicación y la funcionalidad deseada. El resultado de las pruebas automatizadas se basa en la comparación de los resultados de las pruebas generados con los resultados esperados, lo que agiliza el ciclo de vida de las pruebas. Las metodologías de pruebas como las pruebas de regresión y los conjuntos de pruebas unitarias son las más adecuadas para la automatización. La automatización de las pruebas de las propiedades de seguridad permite a los creadores recibir información automatizada sin tener que esperar a una revisión de seguridad. Las pruebas automatizadas en forma de análisis de código estático o dinámico permiten aumentar la calidad del código y contribuyen a detectar posibles problemas de software en una fase temprana del ciclo de vida de desarrollo.

Patrones comunes de uso no recomendados:

- No comunicar los casos de prueba y los resultados de las pruebas automatizadas.
- Llevar a cabo las pruebas automatizadas solo justo antes del lanzamiento.
- Automatizar casos de prueba con requisitos que cambian con frecuencia.
- No proporcionar orientación sobre cómo abordar los resultados de las pruebas de seguridad.

Beneficios de establecer esta práctica recomendada:

- Se ha reducido la dependencia de las personas que evalúan las propiedades de seguridad de los sistemas.
- La obtención de resultados coherentes en numerosos flujos de trabajo mejora la coherencia general.
- Menos probabilidades de que se introduzcan problemas de seguridad en el software de producción.
- Reducción del intervalo de tiempo entre la detección y la corrección gracias a la detección temprana de los problemas de software.

- Mayor visibilidad del comportamiento sistémico o repetido en numerosos flujos de trabajo, que puede servir para impulsar mejoras en toda la organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

A medida que crea el software, adopte diversos mecanismos de prueba de software para asegurarse de probar tanto los requisitos funcionales, basados en la lógica empresarial, como los requisitos no funcionales, que se centran en la fiabilidad, el rendimiento y la seguridad de su aplicación.

Las pruebas de seguridad de aplicaciones estáticas (SAST) analizan el código fuente para revelar patrones de seguridad anómalos y proporcionan indicios de código propenso a errores. Las pruebas SAST se basan en datos estáticos, como la documentación (especificación de requisitos, documentación de diseño y especificaciones de diseño) y el código fuente de la aplicación, con objeto de encontrar una serie de problemas de seguridad conocidos. Los analizadores de código estático pueden ayudar a agilizar el análisis de grandes volúmenes de código. El NIST Quality Group ofrece una comparación de analizadores de seguridad del código fuente, que incluye herramientas de código abierto para analizadores de código de bytes y analizadores de código binario.

Complemente las pruebas estáticas con metodologías de pruebas de seguridad de análisis dinámico (DAST), que efectúan pruebas de la aplicación en ejecución a fin de identificar comportamiento potencialmente inesperado. Las pruebas dinámicas pueden utilizarse para detectar problemas potenciales que no son evidentes mediante el análisis estático. Las pruebas en las etapas de repositorio de código, compilación y canalización le permiten comprobar si existen diferentes tipos de problemas potenciales que podrían introducirse en el código. Amazon Q Developer proporciona recomendaciones de código, incluido el análisis de seguridad, en el IDE del creador. Amazon CodeGuru Security puede identificar problemas cruciales, problemas de seguridad y errores difíciles de detectar durante el desarrollo de la aplicación, y proporciona recomendaciones para mejorar la calidad del código. La extracción de la lista de materiales del software (SBOM) también le permite extraer un registro formal que contiene los detalles y las relaciones de los distintos componentes utilizados en la creación del software. Esto le permite informar sobre la gestión de vulnerabilidades e identificar rápidamente las dependencias del software o los componentes y los riesgos de la cadena de suministro.

El taller Security for Developers usa herramientas para desarrolladores de AWS, como AWS CodeBuild, AWS CodeCommit y AWS CodePipeline, para la automatización de canalizaciones de lanzamiento que incluye las metodologías de prueba SAST y DAST.

A medida que avance en el SDLC, establezca un proceso iterativo que incorpore revisiones periódicas de las aplicaciones con su equipo de seguridad. Los comentarios recopilados en estas revisiones de seguridad deben abordarse y validarse como parte de la revisión de la preparación para el lanzamiento. Estas revisiones establecen una sólida postura de seguridad de la aplicación y proporcionan a los desarrolladores información práctica para afrontar posibles problemas.

Pasos para la implementación

- Implemente herramientas coherentes de IDE, revisión de código y CI/CD que incluyan pruebas de seguridad.
- Considere en qué momento del SDLC es apropiado bloquear las canalizaciones en lugar de limitarse a notificar a los creadores que es necesario solucionar los problemas.
- Automated Security Helper (ASH) es un ejemplo de herramienta de análisis de seguridad de código abierto.
- La ejecución de pruebas o el análisis de código mediante herramientas automatizadas, como Amazon Q Developer integrado con los IDE de los desarrolladores y el Amazon CodeGuru Security para escanear el código al confirmar, ayuda a los creadores a obtener información en el momento adecuado.
- Si usa AWS Lambda para la compilación, puede usar Amazon Inspector para analizar el código de la aplicación en sus funciones.
- Cuando se incluyen pruebas automatizadas en las canalizaciones de CI/CD, es preciso utilizar un sistema de tickets para hacer un seguimiento de la notificación y corrección de problemas de software.
- En el caso de las pruebas de seguridad que puedan generar resultados, la vinculación a orientaciones para la corrección ayuda a los creadores a mejorar la calidad del código.
- Analice periódicamente los resultados de las herramientas automatizadas para dar prioridad a la siguiente automatización, la formación de los creadores o la campaña de concienciación.
- Para extraer la SBOM como parte de las canalizaciones de CI/CD, utilice el generador de SBOM de Amazon Inspector para obtener SBOM de archivos, imágenes de contenedores, directorios, sistemas locales y binarios compilados de Go y Rust en el formato SBOM de CycloneDX.

Recursos

Prácticas recomendadas relacionadas:

- DevOps Guidance: DL.CR.3 Establish clear completion criteria for code tasks

Documentos relacionados:

- Entrega continua e implementación continua

- AWS Socios con competencias en DevOps de
- AWS Socios con competencia en seguridad de para la seguridad de aplicaciones
- Choosing a Well-Architected CI/CD approach
- Secrets detection in Amazon CodeGuru Security
- Amazon CodeGuru Security Detection Library
- Accelerate deployments on AWS with effective governance
- Automatización de implementaciones seguras y sin intervención de AWS
- How Amazon CodeGuru Security helps you effectively balance security and velocity

Videos relacionados:

- Hands-off: Automating continuous delivery pipelines at Amazon
- Automating cross-account CI/CD pipelines
- The Software Development Process at Amazon
- Testing software and systems at Amazon

Ejemplos relacionados:

- Industry awareness for developers
- Automated Security Helper (ASH)
- AWS CodePipeline Governance - Github

SEC11-BP03 Pruebas de penetración periódicas

Lleve a cabo pruebas de penetración periódicas de su software. Este mecanismo ayuda a identificar posibles problemas de software que no pueden detectarse mediante pruebas automatizadas o una revisión manual del código. También puede ayudarle a comprender la eficacia de sus controles de detección. Las pruebas de penetración deben tratar de determinar si se puede hacer que el software lleve a cabo operaciones inesperadas, como exponer datos que deberían estar protegidos o conceder permisos más amplios de lo esperado.

Resultado deseado: las pruebas de penetración se utilizan para detectar, corregir y validar las propiedades de seguridad de la aplicación. Las pruebas de penetración periódicas y programadas deben formar parte del ciclo de vida de desarrollo de software (SDLC). Los resultados de las pruebas de penetración deben resolverse antes del lanzamiento del software. Debe analizar los resultados de las pruebas de penetración para identificar si hay problemas que podrían detectarse mediante la automatización. El uso de un proceso de pruebas de penetración periódicas y repetibles que incluya un mecanismo de retroalimentación activo ayuda a orientar a los creadores y mejora la calidad del software.

Patrones comunes de uso no recomendados:

- Hacer pruebas de penetración solo para problemas de seguridad conocidos o frecuentes.
- Hacer pruebas de penetración de aplicaciones sin herramientas ni bibliotecas de terceros dependientes.
- Hacer pruebas de penetración solo para problemas de seguridad de paquete, sin evaluar la lógica empresarial implementada.

Beneficios de establecer esta práctica recomendada:

- Mayor confianza en las propiedades de seguridad del software antes de su lanzamiento.
- Oportunidad de identificar los patrones de aplicación preferidos, lo que conduce a una mayor calidad del software.
- Un ciclo de retroalimentación que identifica en una fase más temprana del ciclo de desarrollo dónde la automatización o la formación adicional podrían mejorar las propiedades de seguridad del software.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las pruebas de penetración son un ejercicio estructurado de pruebas de seguridad en el que se ejecutan escenarios planificados de infracción de la seguridad para detectar, remediar y validar los controles de seguridad. Las pruebas de penetración comienzan con el reconocimiento, durante el cual se recopilan datos basados en el diseño actual de la aplicación y sus dependencias. Luego, se elabora y ejecuta una lista seleccionada de escenarios de pruebas de seguridad. El objetivo principal de estas pruebas es descubrir problemas de seguridad en la aplicación, que podrían aprovecharse para obtener acceso no deseado a su entorno o acceso no autorizado a los datos. Debe llevar a

cabo pruebas de penetración cuando lance nuevas características, o siempre que la aplicación haya sufrido cambios importantes en su funcionamiento o implementación técnica.

Debe identificar la etapa más apropiada del ciclo de vida de desarrollo en el que llevar a cabo las pruebas de penetración. Estas pruebas deben hacerse lo bastante tarde como para que la funcionalidad del sistema se aproxime al estado de lanzamiento previsto, pero con tiempo suficiente para solucionar cualquier problema.

Pasos para la implementación

- Disponga de un proceso estructurado para determinar el alcance de las pruebas de penetración; basar este proceso en el modelo de amenazas es una buena forma de mantener el contexto.
- Identifique la etapa más apropiada del ciclo de desarrollo en el que llevar a cabo las pruebas de penetración. Debería ser cuando se espera un cambio mínimo en la aplicación, pero con tiempo suficiente para llevar a cabo la corrección.
- Forme a sus creadores sobre qué esperar de los resultados de las pruebas de penetración y cómo obtener información sobre la corrección.
- Utilice herramientas para acelerar el proceso de las pruebas de penetración mediante la automatización de pruebas habituales o repetibles.
- Analice los resultados de las pruebas de penetración con vistas a identificar problemas de seguridad sistémicos y utilice estos datos para efectuar pruebas automatizadas adicionales y para la formación continua de los creadores.

Recursos

Prácticas recomendadas relacionadas:

- SEC11-BP01 Formación en seguridad de las aplicaciones
- SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento

Documentos relacionados:

- Las pruebas de penetración de AWS proporcionan una guía detallada sobres las pruebas de penetración en AWS
- Accelerate deployments on AWS with effective governance
- AWS Socios con competencia en seguridad de

- Modernize your penetration testing architecture on AWS Fargate
- AWS Fault injection Simulator

Ejemplos relacionados:

- Automate API testing with AWS CodePipeline (GitHub)
- Automated security helper (GitHub)

SEC11-BP04 Revisiones del código de conducta

Implemente revisiones de código para ayudar a verificar la calidad y la seguridad del software que se está desarrollando. En las revisiones del código, otros miembros del equipo, además del autor del código original, revisan el código para detectar posibles problemas o vulnerabilidades y comprobar que cumpla con los estándares y las prácticas recomendadas de codificación. Este proceso ayuda a detectar errores, incoherencias y fallos de seguridad que el desarrollador original podría haber pasado por alto. Use herramientas automatizadas para que lo ayuden con las revisiones del código.

Resultado deseado: incluye revisiones de código durante el desarrollo para aumentar la calidad del software que se está escribiendo. Mejora las habilidades de los miembros del equipo con menos experiencia a través de lo aprendido durante la revisión del código. Identifica las oportunidades de automatización y apoya el proceso de revisión del código mediante herramientas y pruebas automatizadas.

Patrones comunes de uso no recomendados:

- No revisa el código antes de la implementación.
- El código lo escribe y lo revisa la misma persona.
- No utiliza la automatización como ayuda con las revisiones del código o para su organización.
- No forma a los creadores en seguridad de las aplicaciones antes de que revisen el código.

Beneficios de establecer esta práctica recomendada:

- Mayor calidad del código.
- Mayor coherencia en el desarrollo del código gracias a la reutilización de estrategias comunes.
- Reducción del número de problemas revelados durante las pruebas de penetración y etapas posteriores.

- Mejora de la transferencia de conocimientos dentro del equipo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Las revisiones de código ayudan a verificar la calidad y la seguridad del software durante su desarrollo. En las revisiones del código, un miembro del equipo que no es el autor del código original revisa el código para detectar posibles problemas o vulnerabilidades y comprobar que cumpla con los estándares y las prácticas recomendadas de codificación. Este proceso ayuda a detectar errores, incoherencias y fallos de seguridad que el desarrollador original podría haber pasado por alto.

Puede usar Amazon CodeGuru Security como ayuda para realizar revisiones de código automatizadas. CodeGuru Security utiliza el machine learning y el razonamiento automatizado para analizar su código e identificar posibles vulnerabilidades de seguridad y problemas de codificación. Integre las revisiones automatizadas del código con los repositorios de código y las canalizaciones de implementación continua/implementación continua (CI/CD) existentes.

Pasos para la implementación

1. Establezca un proceso de revisión del código:
    - Defina cuándo deben realizarse las revisiones del código. Por ejemplo, antes de fusionar el código en la rama principal o antes de implementarlo en producción.
    - Determine quién debe participar en el proceso de revisión del código, como los miembros del equipo, los desarrolladores sénior y los expertos en seguridad.
    - Decida la metodología de revisión del código, incluidos el proceso y las herramientas que se utilizarán.
2. Configure las herramientas de revisión de código:
    - Evalúe y seleccione herramientas de revisión de código que se adapten a las necesidades de su equipo, como GitHub Pull Requests o CodeGuru Security.
    - Integre las herramientas elegidas con sus repositorios de código y canalizaciones de CI/CD existentes.
    - Configure las herramientas para aplicar los requisitos de revisión del código, como el número mínimo de revisores y las reglas de aprobación.
3. Defina una lista de verificación y directrices para la revisión del código:

- Cree una lista de verificación o pautas para la revisión del código que describa lo que debe revisarse. Tenga en cuenta factores como la calidad del código, las vulnerabilidades de seguridad, el cumplimiento de los estándares de codificación y el rendimiento.
- Comparta la lista de verificación o las directrices con el equipo de desarrollo y compruebe que todos entiendan lo que se espera.

4. Forme a los desarrolladores en las prácticas recomendadas de revisión de código:
   - Ofrezca a su equipo formación sobre cómo llevar a cabo revisiones de código eficaces.
   - Informe a su equipo sobre los principios de seguridad de las aplicaciones y las vulnerabilidades más comunes que deben tenerse en cuenta durante las revisiones.
   - Fomente el intercambio de conocimientos y combine las sesiones de programación para mejorar la competencia a los miembros del equipo con menos experiencia.
5. Implemente el proceso de revisión del código:
   - Integre el paso de revisión del código en su flujo de trabajo de desarrollo, como crear una solicitud de extracción y asignar revisores.
   - Exija que los cambios de código se sometan a revisión antes de fusionarlos o implementarlos.
   - Fomente una comunicación abierta y comentarios constructivos durante el proceso de revisión.
6. Supervise y mejore:
   - Revise periódicamente la eficacia del proceso de revisión del código y recopile los comentarios del equipo.
   - Identifique oportunidades de automatización o mejoras en las herramientas para agilizar el proceso de revisión del código.
   - Actualice y perfeccione continuamente la lista de verificación o las directrices de revisión del código en función de lo aprendido y las prácticas recomendadas del sector.
7. Fomente una cultura de revisión del código:
   - Haga hincapié en la importancia de las revisiones del código para mantener la calidad y la seguridad del código.
   - Celebre los éxitos y los aprendizajes del proceso de revisión del código.
   - Fomente un entorno colaborativo y de apoyo en el que los desarrolladores se sientan cómodos dando y recibiendo comentarios.

Recursos

Prácticas recomendadas relacionadas:

- SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento

Documentos relacionados:

- DevOps Guidance: DL.CR.2 Perform peer review for code changes
- About pull requests in GitHub

Ejemplos relacionados:

- Automate code reviews with Amazon CodeGuru Security
- Automating detection of security vulnerabilities and bugs in CI/CD pipelines using Amazon CodeGuru Security CLI

Videos relacionados:

- Continuous improvement of code quality with Amazon CodeGuru Security

SEC11-BP05 Centralización de servicios para paquetes y dependencias

Proporcione servicios centralizados para que sus equipos obtengan paquetes de software y otras dependencias. De este modo, se podrán validar los paquetes antes de incluirlos en el software que escriba y se dispondrá de un origen de datos para el análisis del software que se utiliza en su organización.

Resultado deseado: crea la carga de trabajo a partir de paquetes de software externos además del código que programe. Esto permite implementar más fácilmente funcionalidades de uso reiterado, como un analizador JSON o una biblioteca de cifrado. Centraliza los orígenes de estos paquetes y dependencias para que su equipo de seguridad los valide antes de utilizarlos. Utiliza este planteamiento junto con los flujos de pruebas manuales y automatizadas para aumentar la confianza en la calidad del software que desarrolla.

Patrones comunes de uso no recomendados:

- Obtiene paquetes de repositorios arbitrarios de Internet.
- No prueba los nuevos paquetes antes de ponerlos a disposición de los creadores.

Beneficios de establecer esta práctica recomendada:

- Mejor comprensión de los paquetes que se utilizan en el software que se crea.
- Poder notificar a los equipos de carga de trabajo cuándo es necesario actualizar un paquete en función de la comprensión de quién utiliza qué.
- Reducción del riesgo de que se incluya en el software un paquete con problemas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Proporcione servicios centralizados para paquetes y dependencias de una manera que resulte sencilla de consumir a los creadores. Los servicios centralizados pueden ser lógicamente centrales en lugar de implementarse como un sistema monolítico. Este método le permite proporcionar servicios de una manera que satisfaga las necesidades de los creadores. Debe implementar una forma eficaz de agregar paquetes al repositorio cuando se produzcan actualizaciones o surjan nuevos requisitos. Los servicios de AWS como AWS CodeArtifact o las soluciones similares de socios de AWS son una forma de ofrecer esta capacidad.

Pasos para la implementación

- Implemente un servicio de repositorio lógicamente centralizado que esté disponible en todos los entornos en los que se desarrolla software.
- Incluya el acceso al repositorio como parte del proceso de aprovisionamiento de cuentas de Cuenta de AWS.
- Consolide la automatización para probar paquetes antes de que se publiquen en un repositorio.
- Mantenga métricas de los paquetes, lenguajes y equipos más utilizados y con mayor cantidad de cambios.
- Proporcione un mecanismo automatizado para que los equipos de creación soliciten nuevos paquetes y proporcionen comentarios.
- Analice periódicamente los paquetes del repositorio para identificar la posible repercusión de los problemas que se acaban de detectar.

Recursos

Prácticas recomendadas relacionadas:

- SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento

Documentos relacionados:

- DevOps Guidance: DL.CS.2 Sign code artifacts after each build
- Supply chain Levels for Software Artifacts (SLSA)

Ejemplos relacionados:

- Accelerate deployments on AWS with effective governance
- Tighten your package security with CodeArtifact Package Origin Control toolkit
- Multi Region Package Publishing Pipeline (GitHub)
- Publishing Node.js Modules on AWS CodeArtifact using AWS CodePipeline (GitHub)
- AWS CDK Java CodeArtifact Pipeline Sample (GitHub)
- Distribute private .NET NuGet packages with AWS CodeArtifact (GitHub)

Videos relacionados:

- Proactive security: Considerations and approaches
- The AWS Philosophy of Security (re:Invent 2017)
- When security, safety, and urgency all matter: Handling Log4Shell

SEC11-BP06 Implementación de software mediante programación

Siempre que sea posible, lleve a cabo las implementaciones de software mediante programación. Con este enfoque se reduce la probabilidad de que se produzca un error en la implementación o de que surja un problema inesperado debido a un error humano.

Resultado deseado: la versión de la carga de trabajo que se prueba es la versión que implementa, y la implementación se realiza de forma coherente en todo momento. Externaliza la configuración de su carga de trabajo, lo que ayuda a realizar la implementación en diferentes entornos sin cambios. Emplea la firma criptográfica de los paquetes de software para verificar que no ha cambiado nada entre entornos.

Patrones comunes de uso no recomendados:

- Implementar manualmente el software en producción.
- Hacer cambios manualmente en el software para adaptarlo a distintos entornos.

Beneficios de establecer esta práctica recomendada:

- Mayor confianza en el proceso de lanzamiento de software.
- Reducción del riesgo de que un cambio erróneo afecte a las funciones de la empresa.
- Aumento de la cadencia de lanzamiento debido al menor riesgo del cambio.
- Capacidad de reversión automática en caso de imprevistos durante la implementación.
- Capacidad para demostrar criptográficamente que el software probado es el software implementado.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para mantener una infraestructura de aplicaciones sólida y fiable, aplique prácticas de implementación seguras y automatizadas. Esta práctica implica eliminar el acceso humano persistente de los entornos de producción, utilizar herramientas de CI/CD para las implementaciones y externalizar los datos de configuración específicos del entorno. Si utiliza esta estrategia, puede mejorar la seguridad, reducir el riesgo de errores humanos y agilizar el proceso de implementación.

Puede crear su estructura de Cuenta de AWS para eliminar el acceso humano persistente de los entornos de producción. Esta práctica minimiza el riesgo de cambios no autorizados o modificaciones accidentales, lo que mejora la integridad de los sistemas de producción. En lugar del acceso humano directo, puede utilizar herramientas de CI/CD como AWS CodeBuild y AWS CodePipeline para realizar implementaciones. Puede utilizar estos servicios para automatizar los procesos de creación, prueba e implementación, lo que reduce la intervención manual y aumenta la coherencia.

Para mejorar aún más la seguridad y la trazabilidad, puede firmar los paquetes de aplicaciones después de probarlos y validar estas firmas durante la implementación. Para ello, utilice herramientas criptográficas como AWS Signer o AWS Key Management Service (AWS KMS). Al firmar y verificar los paquetes, puede asegurarse de implementar solo código autorizado y validado en sus entornos.

Asimismo, su equipo puede diseñar la carga de trabajo para que los datos de configuración específicos del entorno se obtengan de una fuente externa, como el Almacén de parámetros de AWS Systems Manager. Esta práctica separa el código de la aplicación de los datos de configuración,

lo que ayuda a administrar y actualizar las configuraciones de forma independiente sin modificar el código de la aplicación en sí.

Para agilizar el aprovisionamiento y la administración de la infraestructura, puede plantearse utilizar herramientas de infraestructura como código (IaC), por ejemplo, AWS CloudFormation o AWS CDK. Puede usar estas herramientas para definir su infraestructura como código, lo que mejora la coherencia y la repetibilidad de las implementaciones en diferentes entornos.

Plantéese usar las implementaciones canario para validar la implementación correcta de su software. Las implementaciones canario suponen implementar cambios en un subconjunto de instancias o usuarios antes de implementarlas en todo el entorno de producción. Luego, puede monitorear el impacto de los cambios y revertirlos si es necesario, lo que minimiza el riesgo de que los problemas se generalicen.

Siga las recomendaciones descritas en el documento técnico Organizing Your AWS Environment Using Multiple Accounts. Este documento técnico proporciona orientación sobre cómo separar los entornos (como los de desarrollo, ensayo y producción) en distintas Cuentas de AWS, lo que mejora aún más la seguridad y el aislamiento.

Pasos para la implementación

1. Configure la estructura de la Cuenta de AWS:
   - Siga las instrucciones del documento técnico Organizing Your AWS Environment Using Multiple Accounts para crear Cuentas de AWS separadas para diferentes entornos (por ejemplo, de desarrollo, ensayo y producción).
   - Configure los controles de acceso y los permisos adecuados para cada cuenta a fin de restringir el acceso humano directo a los entornos de producción.
2. Implemente un proceso de CI/CD:
   - Configure una canalización de CI/CD utilizando servicios como AWS CodeBuild y AWS CodePipeline.
   - Configure la canalización para compilar, probar e implementar automáticamente el código de la aplicación en los entornos respectivos.
   - Integre los repositorios de código con la canalización de CI/CD para el control de versiones y la administración del código.
3. Firme y verifique los paquetes de aplicaciones:
   - Utilice AWS Signer o AWS Key Management Service (AWS KMS) para firmar los paquetes de aplicaciones después de haberlos probado y validado.

- Configure el proceso de implementación para verificar las firmas de los paquetes de aplicaciones antes de implementarlos en los entornos de destino.

4. Externalice los datos de configuración:
   - Guarde los datos de configuración específicos del entorno en el almacén de parámetros de AWS Systems Manager.
   - Modifique el código de la aplicación para recuperar los datos de configuración del almacén de parámetros durante la implementación o el tiempo de ejecución.
5. Implemente Infraestructura como código (IaC):
   - Utilice herramientas de IaC, como AWS CloudFormation o AWS CDK, para definir y administrar su infraestructura como código.
   - Cree plantillas de CloudFormation o scripts de CDK para aprovisionar y configurar los recursos de AWS necesarios para su aplicación.
   - Integre el IaC con la canalización de CI/CD para implementar automáticamente los cambios en la infraestructura junto con los cambios en el código de las aplicaciones.
6. Despliegue implementaciones canario:
   - Configure su proceso de implementación para que sea compatible con las implementaciones canario, en las que los cambios se implementan en un subconjunto de instancias o usuarios antes de implementarlos en todo el entorno de producción.
   - Utilice servicios como AWS CodeDeploy o AWSECS para gestionar las implementaciones canario y monitorear el impacto de los cambios.
   - Implemente mecanismos de reversión para volver a la versión estable anterior si se detectan problemas durante la implementación canario.
7. Monitoree y audite:
   - Configure mecanismos de supervisión y registro para llevar un seguimiento de las implementaciones, el rendimiento de las aplicaciones y los cambios en la infraestructura.
   - Utilice servicios como Amazon CloudWatch y AWS CloudTrail para recopilar y analizar registros y métricas.
   - Implemente comprobaciones de auditoría y conformidad para verificar el cumplimiento de las prácticas recomendadas de seguridad y los requisitos normativos.
8. Mejore continuamente:
   - Revise y actualice periódicamente las prácticas de implementación e incorpore los comentarios y las lecciones aprendidas de las implementaciones anteriores.

- Automatice la mayor parte posible del proceso de implementación para reducir la intervención manual y los posibles errores humanos.
- Colabore con equipos multifuncionales (por ejemplo, de operaciones o seguridad) para adaptar y mejorar continuamente las prácticas de implementación.

Si sigue estos pasos, puede desplegar prácticas de implementación seguras y automatizadas en su entorno de AWS, lo que mejora la seguridad, reduce el riesgo de errores humanos y agiliza el proceso de implementación.

Recursos

Prácticas recomendadas relacionadas:

- SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento
- DL.CI.2 Trigger builds automatically upon source code modifications

Documentos relacionados:

- Accelerate deployments on AWS with effective governance
- Automatización de implementaciones seguras y sin intervención
- Code signing using AWS Certificate Manager Private CA and AWS Key Management Service asymmetric keys
- Code Signing, a Trust and Integrity Control for AWS Lambda

Videos relacionados:

- Hands-off: Automating continuous delivery pipelines at Amazon

Ejemplos relacionados:

- Blue/Green deployments with AWS Fargate

### SEC11-BP07 Evaluación periódica de las propiedades de seguridad de las canalizaciones

Aplique los principios del pilar de seguridad de Well-Architected a sus canalizaciones y preste especial atención a la separación de permisos. Evalúe periódicamente las propiedades de seguridad de su infraestructura de canalización. La administración eficaz de la seguridad de las canalizaciones le permite garantizar la seguridad del software que pasa por ellas.

Resultado deseado: las canalizaciones utilizadas para crear e implementar el software siguen las mismas prácticas recomendadas que cualquier otra carga de trabajo de su entorno. Los equipos que las utilizan no pueden editar las pruebas que implementa en las canalizaciones. Concede a las canalizaciones solo los permisos necesarios para las implementaciones que se están llevando a cabo mediante credenciales temporales. Implementa medidas de seguridad para evitar que las canalizaciones se desplieguen en los entornos incorrectos. Configura las canalizaciones en el estado de emisión para que se pueda validar la integridad de los entornos de compilación.

Patrones comunes de uso no recomendados:

- Pruebas de seguridad que los creadores pueden omitir.
- Permisos demasiado amplios para las canalizaciones de implementación.
- Canalizaciones no configuradas para validar entradas.
- No revisar periódicamente los permisos asociados a la infraestructura de CI/CD.
- Uso de credenciales a largo plazo o codificadas.

Beneficios de establecer esta práctica recomendada:

- Mayor confianza en la integridad del software que se crea e implementa a través de las canalizaciones.
- Capacidad para detener una implementación cuando hay actividades sospechosas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las canalizaciones de implementación son un componente fundamental del ciclo de vida de desarrollo de software y deben seguir los mismos principios y prácticas de seguridad que cualquier otra carga de trabajo de su entorno. Esto incluye implementar controles de acceso adecuados, validar las entradas y revisar y auditar periódicamente los permisos asociados a la infraestructura de CI/CD.

Compruebe que los equipos responsables de crear e implementar las aplicaciones no tengan la capacidad de editar o eludir las pruebas y los controles de seguridad implementados en los procesos. Esta separación de preocupaciones ayuda a mantener la integridad de los procesos de creación e implementación.

Como punto de partida, si lo considera oportuno puede utilizar la arquitectura de referencia de canalizaciones de implementación de AWS. Esta arquitectura de referencia proporciona una base segura y escalable sobre la que construir las canalizaciones de CI/CD de AWS.

Además, puede utilizar servicios como AWS Identity and Access Management Access Analyzer para generar políticas de IAM con privilegios mínimos para los permisos de la canalización y como un paso más en la canalización para verificar los permisos de la carga de trabajo. Esto ayuda a comprobar que las canalizaciones y las cargas de trabajo solo tienen los permisos necesarios para sus funciones específicas, lo que reduce el riesgo de que se produzcan accesos o acciones no autorizados.

Pasos para la implementación

- Comience con la arquitectura de referencia de canalizaciones de implementación de AWS.
- Considere la posibilidad de utilizar el Analizador de acceso de AWS IAM para generar mediante programación políticas de IAM con privilegio mínimo para las canalizaciones.
- Integre las canalizaciones con la supervisión y las alertas para recibir notificaciones de actividades inesperadas o anómalas. En el caso de los servicios administrados de AWS, Amazon EventBridge le permite dirigir los datos a destinos como AWS Lambda o Amazon Simple Notification Service (Amazon SNS).

Recursos

Documentos relacionados:

- AWS Deployment Pipelines Reference Architecture
- Supervisión de AWS CodePipeline
- Prácticas recomendadas de seguridad para AWS CodePipeline

Ejemplos relacionados:

- DevOps monitoring dashboard (GitHub)

### SEC11-BP08 Creación de un programa que integre la propiedad de la seguridad en los equipos de la carga de trabajo

Elabore un programa o un mecanismo que permita a los equipos de creadores tomar decisiones de seguridad sobre el software que crean. Aun así, su equipo de seguridad debe validar estas decisiones durante una revisión, pero integrar la propiedad de la seguridad en los equipos de creadores permite crear cargas de trabajo más rápidas y seguras. Este mecanismo también fomenta una cultura de propiedad que repercute positivamente en el funcionamiento de los sistemas que se crean.

Resultado deseado: ha integrado la propiedad de la seguridad y la toma de decisiones en sus equipos. Ha formado a sus equipos sobre cómo pensar en la seguridad o los ha ampliado con personal de seguridad integrado o asociado. Como resultado, sus equipos toman decisiones de seguridad de mayor calidad en una fase más temprana del ciclo de desarrollo.

Patrones comunes de uso no recomendados:

- Dejar todas las decisiones del diseño de la seguridad en manos del equipo de seguridad.
- No hacer frente a los requisitos de seguridad con suficiente antelación en el proceso de desarrollo.
- No obtener comentarios de los creadores y del personal de seguridad sobre el funcionamiento del programa.

Beneficios de establecer esta práctica recomendada:

- Reducción del tiempo necesario para completar las revisiones de seguridad.
- Reducción de los problemas de seguridad que solo se detectan en la fase de revisión de la seguridad.
- Mejora de la calidad general del software que se programa.
- Oportunidad de identificar y comprender problemas sistémicos o áreas de mejora de alto valor.
- Reducción de la cantidad de tareas que es necesario repetir debido a los resultados de la revisión de seguridad.
- Mejora de la percepción de la función de seguridad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

### Guía para la implementación

Comience con las directrices que se proporcionan en SEC11-BP01 Formación en seguridad de las aplicaciones. A continuación, identifique el modelo operativo para el programa que crea que puede funcionar mejor para su organización. Los dos modelos principales son formar a los desarrolladores o integrar al personal de seguridad en los equipos de creadores. Una vez que haya decidido el enfoque inicial, deberá llevar a cabo una prueba piloto con uno o un pequeño grupo de equipos de carga de trabajo para comprobar que el modelo funciona en su organización. El apoyo de los líderes de los departamentos de creación y seguridad de la organización contribuye a la implantación y al éxito del programa. A medida que cree este programa, es importante elegir métricas que sirvan para mostrar el valor del programa. Aprender de cómo AWS ha tratado este problema es una buena experiencia de aprendizaje. Esta práctica recomendada se centra en gran medida en la cultura y el cambio organizativo. Las herramientas que emplee deben apoyar la colaboración entre las comunidades de creadores y de seguridad.

### Pasos para la implementación

- Empiece por formar a los desarrolladores en la seguridad para las aplicaciones.
- Cree una comunidad y un programa de incorporación para educar a los desarrolladores.
- Elija un nombre para el programa. Los más utilizados son Guardians, Champions o Advocates.
- Identifique el modelo que se va a utilizar: formar a los desarrolladores, incorporar ingenieros de seguridad o tener roles de seguridad afines.
- Identifique a los patrocinadores del proyecto entre los encargados de la seguridad, los desarrolladores y, quizá, otros grupos pertinentes.
- Haga un seguimiento del número de personas que participan en el programa, el tiempo necesario para las revisiones y los comentarios de los desarrolladores y el personal de seguridad. Utilice estas métricas para hacer mejoras.

### Recursos

Prácticas recomendadas relacionadas:

- SEC11-BP01 Formación en seguridad de las aplicaciones
- SEC11-BP02 Automatización de las pruebas a lo largo del ciclo de vida de desarrollo y lanzamiento

Documentos relacionados:

- How to approach threat modeling
- How to think about cloud security governance
- How AWS built the Security Guardians program, a mechanism to distribute security ownership
- How to build a Security Guardians program to distribute security ownership

Videos relacionados:

- Proactive security: Considerations and approaches
- AppSec tooling and culture tips from AWS and Toyota Motor North America

# Fiabilidad

El pilar de fiabilidad abarca la capacidad de una carga de trabajo para llevar a cabo su función prevista de forma correcta y coherente cuando se espera que lo haga. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de fiabilidad.

Áreas de prácticas recomendadas

- Principios básicos
- Arquitectura de la carga de trabajo
- Administración de cambios
- Administración de errores

## Principios básicos

Preguntas

- REL 1. ¿Cómo administra las Service Quotas y las restricciones?
- REL 2. ¿Cómo planifica la topología de la red?

### REL 1. ¿Cómo administra las Service Quotas y las restricciones?

Para las arquitecturas de carga de trabajo basadas en la nube, existen Service Quotas (que también se denominan límites de servicio). Estas cuotas existen para evitar el aprovisionamiento accidental de más recursos de los que se necesitan y para limitar las tasas de solicitudes en las operaciones de la API a fin de proteger los servicios contra el abuso. También hay limitaciones de recursos, por

ejemplo, la velocidad a la que se pueden introducir bits por un cable de fibra óptica o la cantidad de almacenamiento en un disco físico.

Prácticas recomendadas

- REL01-BP01 Conocimiento de las cuotas y restricciones del servicio
- REL01-BP02 Administración de cuotas de servicio en cuentas y regiones
- REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura
- REL01-BP04 Supervisión y administración de cuotas
- REL01-BP05 Automatización de la administración de cuotas
- REL01-BP06 Garantía de que exista una diferencia suficiente entre las cuotas actuales y el uso máximo para permitir la conmutación por error

REL01-BP01 Conocimiento de las cuotas y restricciones del servicio

Conozca las cuotas predeterminadas y administre las solicitudes de aumento de cuota para su arquitectura de carga de trabajo. Sepa qué restricciones de recursos en la nube, como el disco o la red, pueden causar impacto.

Resultado deseado: los clientes pueden evitar la degradación o la interrupción de sus Cuentas de AWS mediante la implementación de las directrices adecuadas para supervisar las métricas clave, las revisiones de la infraestructura y los pasos de corrección de la automatización para comprobar que no se alcancen las cuotas ni las restricciones de los servicios que puedan provocar un deterioro o interrupción de los mismos.

Patrones comunes de uso no recomendados:

- Implementar una carga de trabajo sin conocer las cuotas estrictas o flexibles y sus límites para los servicios utilizados.
- Implementar una carga de trabajo de reemplazo sin analizar ni volver a configurar las cuotas necesarias o sin contactar previamente con el servicio de asistencia.
- Suponer que los servicios en la nube no tienen límites y que los servicios pueden utilizarse sin tener en cuenta tarifas, límites, recuentos o cantidades.
- Suponer que las cuotas se incrementarán automáticamente.
- Desconocer el proceso y la cronología de las solicitudes de cuotas.

- Suponer que la cuota de servicio predeterminada en la nube es la misma para todos los servicios en diferentes regiones.
- Suponer que se pueden incumplir las restricciones del servicio y que los sistemas se escalarán automáticamente o agregarán un aumento del límite más allá de las restricciones del recurso.
- No probar la aplicación en picos de tráfico para estresar el uso de sus recursos.
- Aprovisionar el recurso sin analizar el tamaño de recurso requerido.
- Sobreaprovisionar la capacidad mediante la elección de tipos de recursos que van mucho más allá de las necesidades reales o de los picos previstos.
- No evaluar las necesidades de capacidad para nuevos niveles de tráfico antes de que se produzca un nuevo evento con un cliente o de implementar una nueva tecnología.

Beneficios de establecer esta práctica recomendada: la supervisión y la administración automatizada de las cuotas de servicio y las limitaciones de recursos pueden reducir los fallos de forma proactiva. Los cambios en los patrones de tráfico para el servicio de un cliente pueden provocar una interrupción o un deterioro si no se siguen las prácticas recomendadas. Con la supervisión y la administración de estos valores en todas las regiones y en todas las cuentas, las aplicaciones pueden tener una mayor resiliencia ante acontecimientos adversos o imprevistos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Service Quotas es un servicio de AWS que le ayuda a administrar sus cuotas de más de 250 servicios de AWS desde una sola ubicación. Además de consultar los valores de las cuotas, también puede solicitar y hacer un seguimiento de los aumentos de las cuotas desde la consola de Service Quotas o mediante el AWS SDK. AWS Trusted Advisor ofrece una comprobación de las cuotas que muestra su uso y las cuotas para ciertos aspectos de algunos servicios. Las cuotas de servicio predeterminadas por servicio también se encuentran en la documentación de AWS de cada servicio correspondiente (por ejemplo, consulte Cuotas de Amazon VPC).

Algunos límites de servicio, como los límites de velocidad en las API limitadas, se establecen en Amazon API Gateway mediante la configuración de un plan de uso. Algunos límites que se establecen como configuración en sus servicios respectivos incluyen las IOPS aprovisionadas, el almacenamiento asignado en Amazon RDS y las asignaciones de volumen de Amazon EBS. Amazon Elastic Compute Cloud tiene su propio panel de límites de servicio que puede ayudarle a administrar su instancia, Amazon Elastic Block Store y los límites de direcciones IP elásticas. Si tiene

un caso de uso en el que las cuotas de servicio repercuten en el rendimiento de su aplicación y no se ajustan a sus necesidades, contacte con Soporte para ver si existen mitigaciones.

Las cuotas de servicio pueden ser específicas de una región o también pueden tener carácter global. El uso de un servicio de AWS que alcance su cuota no actuará del modo previsto en un uso normal y puede provocar la interrupción o el deterioro del servicio. Por ejemplo, una cuota de servicio limita el número de instancias de Amazon EC2 de DL utilizadas en una región. Ese límite se puede alcanzar durante un evento de escalado de tráfico mediante grupos de escalado automático (ASG).

Las cuotas de servicio de cada cuenta deben evaluarse periódicamente para determinar cuáles podrían ser los límites de servicio adecuados para esa cuenta. Estas cuotas de servicio existen como barreras de protección operativas para evitar aprovisionar por accidente más recursos de los necesarios. También sirven para limitar las tasas de solicitudes en las operaciones de API para proteger los servicios del abuso.

Las restricciones de servicio son diferentes de las cuotas de servicio. Las restricciones de servicio representan los límites de un recurso concreto, tal y como los define ese tipo de recurso. Pueden ser la capacidad de almacenamiento (por ejemplo, gp2 tiene un límite de tamaño de 1 GB - 16 TB) o el rendimiento del disco. Es esencial que las restricciones de un tipo de recurso se diseñen y evalúen constantemente para detectar un uso que pueda alcanzar su límite. Si se alcanza una restricción de forma inesperada, las aplicaciones o servicios de la cuenta pueden deteriorarse o interrumpirse.

Si existe un caso de uso en el que las cuotas de servicio repercuten en el rendimiento de una aplicación y no pueden ajustarse a las necesidades requeridas, contacte con Soporte para ver si existen mitigaciones. Para obtener más información sobre el ajuste de las cuotas fijas, consulte REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura.

Hay una serie de servicios y herramientas de AWS con las que se puede supervisar y administrar Service Quotas. El servicio y las herramientas se deben utilizar para proporcionar comprobaciones automatizadas o manuales de los niveles de cuota.

- AWS Trusted Advisor ofrece una comprobación de cuotas de servicio que muestra su uso y las cuotas para ciertos aspectos de algunos servicios. Puede ayudar a identificar los servicios que están cerca de la cuota.
- Consola de administración de AWS proporciona métodos para mostrar los valores de las cuotas de los servicios, administrar, solicitar nuevas cuotas, supervisar el estado de las solicitudes de cuotas y mostrar el historial de cuotas.
- AWS CLI y los CDK ofrecen métodos programáticos para administrar y supervisar automáticamente los niveles de cuota de servicio y el uso.

Pasos para la implementación

Para Service Quotas:

- Revise AWS Service Quotas.
- Para conocer sus cuotas de servicio existentes, determine los servicios (como Analizador de acceso de IAM) que se utilizan. Hay aproximadamente 250 servicios de AWS controlados por cuotas de servicio. A continuación, determine el nombre específico de la cuota de servicio que se podría utilizar en cada cuenta y región. Hay aproximadamente 3000 nombres de cuotas de servicio por región.
- Amplíe este análisis de cuotas con AWS Config para encontrar todos los recursos de AWS que utiliza en suCuentas de AWS.
- Utilice los datos de AWS CloudFormation para determinar los recursos de AWS que utiliza. Observe los recursos que se crearon en la Consola de administración de AWS o con el comando `list-stack-resources` de la AWS CLI. También puede ver los recursos configurados para implementarlos en la propia plantilla.
- Consulte el código de implementación para determinar todos los servicios que necesita su carga de trabajo.
- Determine las cuotas de servicio que se aplican. Use la información accesible mediante programación de Trusted Advisor y Service Quotas.
- Establezca un método de supervisión automatizado (consulte REL01-BP02 Administración de cuotas de servicio en cuentas y regiones y REL01-BP04 Supervisión y administración de cuotas) para alertar e informar si las cuotas de servicios están cerca o han alcanzado su límite.
- Establezca un método automatizado y programático para comprobar si se ha modificado una cuota de servicio en una región, pero no en otras regiones de la misma cuenta (consulte REL01-BP02 Administración de cuotas de servicio en cuentas y regiones y REL01-BP04 Supervisión y administración de cuotas).
- Automatice el análisis de los registros y las métricas de las aplicaciones para determinar si existen errores de cuotas o restricciones de servicio. Si se producen estos errores, envíe alertas al sistema de supervisión.
- Establezca procedimientos de ingeniería para calcular el cambio requerido en la cuota (consulte REL01-BP05 Automatización de la administración de cuotas) una vez que se haya identificado que se requieren cuotas más altas para servicios específicos.

- Cree un flujo de trabajo de aprovisionamiento y aprobación para solicitar cambios en la cuota de servicio. Debería incluir un flujo de trabajo de excepciones en caso de denegación de la solicitud o de aprobación parcial.
- Cree un método de ingeniería para efectuar una revisión de las cuotas de servicio previa al aprovisionamiento y el uso de nuevos servicios de AWS antes de implementarlos en entornos de producción o de carga (por ejemplo, una cuenta de pruebas de carga).

Para las restricciones de servicio:

- Establezca métodos de supervisión y medición para alertar de la lectura de los recursos próximos a sus restricciones. Use CloudWatch de manera correspondiente para las métricas o la supervisión de registros.
- Establezca umbrales de alerta para cada recurso con una restricción significativa para la aplicación o el sistema.
- Cree procedimientos de administración de flujos de trabajo e infraestructuras para cambiar el tipo de recurso si la restricción está próxima a su uso. Como práctica recomendada, este flujo de trabajo debe incluir pruebas de carga para verificar que el nuevo tipo sea el tipo de recurso correcto con las nuevas restricciones.
- Migre el recurso identificado al nuevo tipo de recurso recomendado, mediante los procedimientos y los procesos existentes.

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP02 Administración de cuotas de servicio en cuentas y regiones
- REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura
- REL01-BP04 Supervisión y administración de cuotas
- REL01-BP05 Automatización de la administración de cuotas
- REL01-BP06 Garantía de que exista una diferencia suficiente entre las cuotas actuales y el uso máximo para permitir la conmutación por error
- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

- REL11-BP03 Automatización de la reparación en todas las capas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Documentos relacionados:

- AWS Well-Architected Framework's Reliability Pillar: Availability
- AWS Service Quotas (conocido anteriormente como límites del servicio)
- AWS Trusted Advisor Best Practice Checks (consulte la sección Service Limits)
- Monitor de cuotas para AWS en respuestas de AWS
- Cuotas de servicio de Amazon EC
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guía del usuario de Service Quotas
- Monitor de cuotas para AWS
- AWS Fault Isolation Boundaries
- Disponibilidad con redundancia
- AWS para datos
- ¿Qué es la integración continua?
- ¿Qué es la entrega continua?
- Socio de APN: socios que pueden ayudar con la administración de la configuración
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower

Videos relacionados:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo

Herramientas relacionadas:

- Revisor de Amazon CodeGuru
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP02 Administración de cuotas de servicio en cuentas y regiones

Si utiliza múltiples cuentas o regiones, solicite las cuotas pertinentes en todos los entornos en los que se ejecutan sus cargas de trabajo de producción.

Resultado deseado: los servicios y las aplicaciones no deberían verse afectados por el agotamiento de la cuota de servicio de las configuraciones que abarquen cuentas o regiones o que tengan diseños de resiliencia que utilicen la conmutación por error de zona, región o cuenta.

Patrones comunes de uso no recomendados:

- Permitir que aumente el uso de recursos en una región aislada sin ningún mecanismo para mantener la capacidad en las demás.
- Configurar manualmente todas las cuotas de forma independiente en regiones aisladas.
- No considerar el efecto de las arquitecturas de resiliencia (activa o pasiva) en las futuras necesidades de cuota durante un deterioro de la región no principal.
- No evaluar las cuotas periódicamente ni hacer los cambios necesarios en cada región y cuenta donde se ejecuta la carga de trabajo.
- No utilizar las plantillas de solicitud de cuotas para solicitar aumentos en varias regiones y cuentas.
- No actualizar las cuotas de servicio por pensar erróneamente que el aumento de las cuotas tiene implicaciones de costo, como las solicitudes de reserva de computación.

Beneficios de establecer esta práctica recomendada: comprobar que puede gestionar la carga actual en regiones o cuentas secundarias en caso de que los servicios regionales dejen de estar disponibles. Esto puede reducir el número de errores o los niveles de deterioro que se producen durante la pérdida de una región.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El seguimiento de las cuotas de servicio se lleva a cabo por cuenta. A no ser que se especifique lo contrario, cada cuota es específica de una Región de AWS. Además de los entornos de producción, administre también las cuotas en todos los entornos que no sean de producción aplicables, de modo que las pruebas y el desarrollo no se vean limitados. El mantenimiento de un elevado nivel de resiliencia requiere que las cuotas de servicio se evalúen continuamente (ya sea de forma automatizada o manual).

Dado que cada vez hay más cargas de trabajo en todas las regiones debido a la implementación de diseños que utilizan enfoques Activo/Activo, Activo/Pasivo (caliente), Activo/Pasivo (frío) y Activo/Pasivo (luz piloto), es esencial comprender los niveles de cuota de todas las regiones y cuentas. Los patrones de tráfico anteriores no siempre son un buen indicador de si la cuota de servicio está configurada correctamente.

Igualmente importante es el hecho de que el límite de nombres de cuota de servicio no es siempre el mismo para todas las regiones. En una región, el valor podría ser cinco, y en otra, diez. La administración de estas cuotas debe abarcar los mismos servicios, cuentas y regiones para proporcionar una resiliencia coherente bajo carga.

Concilie todas las diferencias de cuota de servicio entre las distintas regiones (región activa o región pasiva) y cree procesos para conciliar continuamente estas diferencias. Los planes de prueba de las conmutaciones por error pasivas de las regiones en muy pocas ocasiones se escalan a la capacidad activa máxima, lo que significa que los ejercicios del día de juego o de mesa pueden no encontrar diferencias en las cuotas de servicio entre las regiones y tampoco mantener los límites correctos.

Es muy importante hacer un seguimiento y evaluar la variación de las cuotas de servicio, es decir, la condición en la que se modifican los límites de las cuotas de servicio para una cuota determinada en una región y no en todas las regiones. Debe considerarse la posibilidad de cambiar la cuota en las regiones con tráfico o con posibilidad de tener tráfico.

- Seleccione las cuentas y regiones que correspondan según sus requisitos de servicio, latencia, normativos y de recuperación de desastres (DR).

- Identifique las cuotas de servicio en todas las cuentas, regiones y zonas de disponibilidad pertinentes. Los límites se determinan por cuenta y región. Estos valores deben compararse para detectar diferencias.

Pasos para la implementación

- Revise los valores de Service Quotas que podrían haber superado el nivel de riesgo de uso. AWS Trusted Advisor proporciona alertas si superan los umbrales del 80 % y el 90 %.
- Revise los valores de las cuotas de servicio en cualquier región pasiva (en un diseño Activo/Pasivo). Verifique que la carga se ejecutará correctamente en las regiones secundarias si se produce un error en la región principal.
- Automatice la evaluación de si se ha producido alguna desviación de la cuota de servicio entre regiones de la misma cuenta y actúe en consecuencia para modificar los límites.
- Si las unidades organizativas (UO) del cliente están estructuradas de la forma admitida, las plantillas de cuotas de servicio se deberán actualizar para reflejar los cambios en las cuotas que deban aplicarse a varias regiones y cuentas.
  - Cree una plantilla y asocie regiones al cambio de cuota.
  - Revise todas las plantillas de cuota de servicio existentes por si fuera necesario hacer algún cambio (región, límites y cuentas).

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP01 Conocimiento de las cuotas y restricciones del servicio
- REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura
- REL01-BP04 Supervisión y administración de cuotas
- REL01-BP05 Automatización de la administración de cuotas
- REL01-BP06 Garantía de que exista una diferencia suficiente entre las cuotas actuales y el uso máximo para permitir la conmutación por error
- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

- REL11-BP03 Automatización de la reparación en todas las capas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Documentos relacionados:

- AWS Well-Architected Framework's Reliability Pillar: Availability
- AWS Service Quotas (conocido anteriormente como límites del servicio)
- AWS Trusted Advisor Best Practice Checks (consulte la sección Service Limits)
- Monitor de cuotas para AWS en respuestas de AWS
- Cuotas de servicio de Amazon EC
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guía del usuario de Service Quotas
- Monitor de cuotas para AWS
- AWS Fault Isolation Boundaries
- Disponibilidad con redundancia
- AWS para datos
- ¿Qué es la integración continua?
- ¿Qué es la entrega continua?
- Socio de APN: socios que pueden ayudar con la administración de la configuración
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower

Videos relacionados:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo

Servicios relacionados:

- Revisor de Amazon CodeGuru
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura

Conozca las cuotas de servicio inalterables, las restricciones de servicio y los límites de recursos físicos. Diseñe arquitecturas para aplicaciones y servicios que eviten que estos límites afecten a la fiabilidad.

Algunos ejemplos son el ancho de banda de la red, el tamaño de la carga útil de la invocación de funciones sin servidor, la tasa de ráfagas de aceleración para una puerta de enlace de API y las conexiones simultáneas de usuarios a una base de datos.

Resultado deseado: la aplicación o el servicio funciona según lo esperado en condiciones de tráfico normal y alto. Se han diseñado para funcionar dentro de las limitaciones fijadas para ese recurso o cuotas de servicio.

Patrones comunes de uso no recomendados:

- Elegir un diseño que utilice el recurso de un servicio, sin saber que existen restricciones que provocarán que este diseño produzca un error en el escalado.
- Efectuar una evaluación comparativa que no es realista y que alcanzará las cuotas fijas del servicio durante la evaluación. Por ejemplo, ejecutar pruebas con un límite de ráfagas, pero durante un periodo prolongado.

- Elegir un diseño que no pueda escalarse ni modificarse si se van a superar las cuotas de servicio fijas. Por ejemplo, un tamaño de carga útil de SQS de 256 KB.
- No se diseña ni se implementa la observabilidad para supervisar y avisar sobre umbrales de cuotas de servicio que podrían estar en riesgo durante eventos de alto tráfico.

Beneficios de establecer esta práctica recomendada: comprobar que la aplicación se ejecute en todos los niveles de carga de servicios proyectados sin interrupciones ni deterioro.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

A diferencia de las cuotas de servicio o recursos flexibles que pueden sustituirse por unidades de mayor capacidad, las cuotas fijas de los servicios de AWS no pueden modificarse. Esto significa que todos estos tipos de servicios de AWS deben evaluarse para detectar posibles límites estrictos de capacidad cuando se utilizan en el diseño de una aplicación.

Los límites estrictos se muestran en la consola de Service Quotas. Si las columnas muestran `ADJUSTABLE = No`, el servicio tiene un límite estricto. Los límites estrictos también se muestran en algunas páginas de configuración de recursos. Por ejemplo, Lambda tiene límites estrictos específicos que no se pueden ajustar.

Por ejemplo, cuando se diseña una aplicación python para que se ejecute en una función de Lambda, es necesario evaluar la aplicación para determinar si existe alguna posibilidad de que Lambda se ejecute durante más de 15 minutos. Si el código puede ejecutarse durante más tiempo de este límite de cuota de servicio, deben considerarse tecnologías o diseños alternativos. Si se alcanza el límite después de la implementación en producción, la aplicación sufrirá degradación e interrupciones hasta que pueda remediarse. A diferencia de las cuotas flexibles, no existe ningún método para cambiar estos límites, ni siquiera en caso de eventos de emergencia de gravedad 1.

Una vez que la aplicación se ha implementado en un entorno de pruebas, se deben utilizar estrategias para averiguar si se puede alcanzar algún límite estricto. Las pruebas de estrés, las pruebas de carga y las pruebas de caos deben formar parte del plan de pruebas de introducción.

Pasos para la implementación

- Revise la lista completa de servicios de AWS que podrían utilizarse en la fase de diseño de la aplicación.

- Revise los límites de cuotas flexibles y estrictos para todos estos servicios. No todos los límites se muestran en la consola de Service Quotas. Algunos servicios describen estos límites en ubicaciones alternativas.
- Al diseñar su aplicación, revise los impulsores empresariales y tecnológicos de la carga de trabajo, como los resultados empresariales, el caso de uso, los sistemas dependientes, los objetivos de disponibilidad y los objetos de recuperación de desastres. Permita que sus impulsores empresariales y tecnológicos guíen el proceso para identificar el sistema distribuido adecuado para su carga de trabajo.
- Analice la carga de servicio en todas las regiones y cuentas. Muchos límites estrictos se basan en la región para los servicios. Sin embargo, algunos límites se basan en las cuentas.
- Analice el uso de recursos de las arquitecturas de resistencia durante un error zonal y un error regional. En la progresión de los diseños multirregionales que utilizan enfoques activo/activo, activo/pasivo: en caliente, activo/pasivo: en frío y activo/pasivo: luz piloto, estos casos de error provocarán un mayor uso. Esto crea un caso de uso potencial para alcanzar límites estrictos.

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP01 Conocimiento de las cuotas y restricciones del servicio
- REL01-BP02 Administración de cuotas de servicio en cuentas y regiones
- REL01-BP04 Supervisión y administración de cuotas
- REL01-BP05 Automatización de la administración de cuotas
- REL01-BP06 Garantía de que exista una diferencia suficiente entre las cuotas actuales y el uso máximo para permitir la conmutación por error
- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores
- REL11-BP03 Automatización de la reparación en todas las capas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Documentos relacionados:

- AWS Well-Architected Framework’s Reliability Pillar: Availability

- AWS Service Quotas (conocido anteriormente como límites del servicio)
- AWS Trusted Advisor Best Practice Checks (consulte la sección Service Limits)
- Monitor de cuotas para AWS en respuestas de AWS
- Cuotas de servicio de Amazon EC
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guía del usuario de Service Quotas
- Monitor de cuotas para AWS
- AWS Fault Isolation Boundaries
- Disponibilidad con redundancia
- AWS para datos
- Qué es la integración continua?
- Qué es la entrega continua?
- Socio de APN: socios que pueden ayudar con la administración de la configuración
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower
- Actions, resources, and condition keys for Service Quotas

Videos relacionados:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small

Herramientas relacionadas:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP04 Supervisión y administración de cuotas

Evalúe el uso potencial y aumente las cuotas pertinentemente, lo que permitirá un crecimiento planificado del uso.

Resultado deseado: se implementaron sistemas activos y automatizados que administran y supervisan. Estas soluciones operativas garantizan que los umbrales de uso de las cuotas estén a punto de alcanzarse. Esto se solucionaría de forma proactiva mediante cambios solicitados en las cuotas.

Patrones comunes de uso no recomendados:

- No configurar la supervisión para comprobar el umbral de cuota de servicio.
- No configurar la supervisión de los límites estrictos, aunque esos valores no puedan modificarse.
- Suponer que el tiempo necesario para solicitar y asegurar un cambio de cuota flexible es inmediato o de corta duración.
- Configurar alarmas de aproximación para cuotas de servicio sin contar con ningún proceso para responder a una alerta.
- Configurar alarmas solo para servicios compatibles con AWS Service Quotas y no supervisar ningún otro servicio de AWS.
- No considerar la administración de cuotas para diseños de resiliencia multirregional, como los métodos activo/activo, activo/pasivo: en caliente, activo/pasivo: en frío y activo/pasivo: luz piloto.
- No evaluar las diferencias de cuota entre regiones.

- No evaluar las necesidades de cada región para una solicitud específica de aumento de cuota.
- No utilizar plantillas para la gestión de cuotas en varias regiones.

Beneficios de establecer esta práctica recomendada: el seguimiento automático de AWS Service Quotas y la supervisión del uso en función de dichas cuotas le permitirá comprobar cuándo se acerca al límite de una cuota. También puede utilizar estos datos de supervisión para ayudar a limitar cualquier degradación debida al agotamiento de cuotas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para los servicios admitidos, puede supervisar las cuotas por medio de la configuración de varios servicios diferentes que pueden evaluar y enviar alertas o alarmas. Esto puede ayudar a supervisar el uso y alertarle de que se aproxima a las cuotas. Estas alarmas se pueden invocar desde AWS Config, funciones de Lambda, Amazon CloudWatch o AWS Trusted Advisor. También puede usar filtros de métricas en Registros de CloudWatch para buscar y extraer patrones en registros de modo que pueda determinar si el uso se aproxima a los umbrales de las cuotas.

Pasos para la implementación

Para la supervisión:

- Capture el consumo de recursos actual (por ejemplo, buckets o instancias). Utilice las operaciones de la API de servicio, como la API `DescribeInstances` de Amazon EC2, para recopilar el consumo actual de recursos.
- Capture las cuotas actuales que son esenciales y aplicables a los servicios que utilizan lo siguiente:
  - Service Quotas de AWS
  - AWS Trusted Advisor
  - Documentación de AWS
  - Páginas específicas de los servicios de AWS
  - AWS Command Line Interface (AWS CLI)
  - Kit de desarrollo de la nube de AWS (AWS CDK)
- Utilice AWS Service Quotas, un servicio de AWS que le ayuda a administrar sus cuotas de más de 250 servicios de AWS desde una sola ubicación.

- Utilice los límites de servicio de Trusted Advisor para supervisar sus límites de servicio actuales en diversos umbrales.
- Utilice el historial de cuotas de servicio (consola o AWS CLI) para comprobar los aumentos regionales.
- Compare los cambios de cuota de servicio en cada región y cada cuenta para crear equivalencias, si es necesario.

Para la administración:

- Automatizada: configure una regla personalizada de AWS Config para analizar las cuotas de servicio en todas las regiones y comparar las diferencias.
- Automatizada: configure una función de Lambda programada para analizar las cuotas de servicio en todas las regiones y comparar las diferencias.
- Manual: analice la cuota de servicios a través de la AWS CLI, la API o la consola de AWS para analizar las cuotas de servicio en todas las regiones y comparar las diferencias. Informe de las diferencias.
- Si se identifican diferencias en las cuotas entre las regiones, solicite un cambio de cuota, si es necesario.
- Revise el resultado de todas las solicitudes.

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP01 Conocimiento de las cuotas y restricciones del servicio
- REL01-BP02 Administración de cuotas de servicio en cuentas y regiones
- REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura
- REL01-BP05 Automatización de la administración de cuotas
- REL01-BP06 Garantía de que exista una diferencia suficiente entre las cuotas actuales y el uso máximo para permitir la conmutación por error
- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

- REL11-BP03 Automatización de la reparación en todas las capas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Documentos relacionados:

- AWS Well-Architected Framework's Reliability Pillar: Availability
- AWS Service Quotas (conocido anteriormente como límites del servicio)
- AWS Trusted Advisor Best Practice Checks (consulte la sección Service Limits)
- Monitor de cuotas para AWS en respuestas de AWS
- Cuotas de servicio de Amazon EC2
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guía del usuario de Service Quotas
- Monitor de cuotas para AWS
- AWS Fault Isolation Boundaries
- Disponibilidad con redundancia
- AWS para datos
- ¿Qué es la integración continua?
- ¿Qué es la entrega continua?
- Socio de APN: socios que pueden ayudar con la administración de la configuración
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower
- Actions, resources, and condition keys for Service Quotas

Videos relacionados:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas

- AWS IAM Quotas Demo
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small

Herramientas relacionadas:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP05 Automatización de la administración de cuotas

Las cuotas de servicio, que también se denominan límites en el servicio de AWS, establecen los valores máximos de recursos para la Cuenta de AWS. Cada servicio de AWS define un conjunto de cuotas y sus valores predeterminados. Para proporcionar a su carga de trabajo acceso a todos los recursos que necesita, tal vez deba aumentar los valores de las cuotas de servicio.

El aumento del consumo de recursos de AWS por parte de la carga de trabajo puede poner en peligro la estabilidad de la carga de trabajo y afectar a la experiencia del usuario si se superan las cuotas. Implemente herramientas que lo avisen cuando su carga de trabajo se acerque a los límites y considere la posibilidad de crear solicitudes de aumento de cuota automáticamente.

Resultado deseado: las cuotas están configuradas adecuadamente para las cargas de trabajo que se ejecutan en cada región de Cuenta de AWS.

Patrones comunes de uso no recomendados:

- No tiene en cuenta las cuotas ni las ajusta de forma adecuada para cumplir con los requisitos de carga de trabajo.

- Realiza un seguimiento de las cuotas y el uso mediante métodos que pueden quedar obsoletos, como hojas de cálculo.
- Los límites de servicio solo se actualizan de forma periódica.
- Su organización carece de procesos operativos para revisar las cuotas existentes y solicitar aumentos de las cuotas de servicio cuando sea necesario.

Beneficios de establecer esta práctica recomendada:

- Mayor resiliencia de la carga de trabajo: evita los errores causados por superar las cuotas de recursos de AWS.
- Recuperación ante desastres simplificada: puede reutilizar los mecanismos automatizados de gestión de cuotas creados en la región principal durante la configuración de recuperación ante desastres en otra Región de AWS.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Consulte las cuotas actuales y realice un seguimiento del consumo continuo de cuotas a través de mecanismos como la consola de AWS Service Quotas, AWS Command Line Interface (AWS CLI) y los SDK de AWS. También puede integrar las bases de datos de gestión de la configuración (CMDB) y los sistemas de gestión de servicios de TI (ITSM) con las API de cuota de servicio de AWS.

Genere alertas automatizadas si el uso de la cuota alcanza los umbrales definidos y defina un proceso para enviar las solicitudes de aumento de cuota cuando reciba alertas. Si la carga de trabajo subyacente es fundamental para su empresa, puede automatizar las solicitudes de aumento de cuota, pero extreme las precauciones al realizar las pruebas de automatización para evitar el riesgo de que se tomen medidas descontroladas, como un ciclo de retroalimentación sobre el crecimiento.

Los aumentos de cuota más pequeños suelen aprobarse automáticamente. Es posible que el soporte técnico de AWS deba procesar manualmente las solicitudes de cuotas más altas, y su revisión y procesamiento pueden tardar más tiempo. Disponga de más tiempo para procesar varias solicitudes o solicitudes de grandes aumentos.

Pasos para la implementación

- Implemente un monitoreo automatizado de las cuotas de servicio y emita alertas si el aumento del uso de los recursos de su carga de trabajo se acerca a los límites de su cuota. Por ejemplo,

el Supervisor de cuotas para AWS puede proporcionar un monitoreo automatizado de las cuotas de servicio. Esta herramienta se integra con AWS Organizations y se implementa mediante CloudFormation StackSets para que las nuevas cuentas se monitoreen automáticamente al crearlas.

- Utilice características como las plantillas de solicitud de Service Quotas o AWS Control Tower para simplificar la configuración de Service Quotas para cuentas nuevas.
- Cree paneles de control del uso actual de sus cuotas de servicio en todas las regiones de Cuentas de AWS y consúltelos según sea necesario para no superar sus cuotas. El panel organizativo de Trusted Advisor (TAO), que forma parte de los paneles de inteligencia en la nube, puede ayudarlo a empezar rápidamente con dicho panel.
- Lleve un seguimiento de las solicitudes de aumento de límites de servicio. Con Consolidated Insights from Multiple Accounts (CIMA) puede ver las solicitudes de toda su organización.
- Pruebe la generación de alertas y cualquier automatización de solicitudes de aumento de cuotas estableciendo umbrales de cuota más bajos en las cuentas que no son de producción. No realice estas pruebas en una cuenta de producción.

Recursos

Prácticas recomendadas relacionadas:

- OPS10-BP07 Automatización de las respuestas a eventos

Documentos relacionados:

- Socio de APN: socios que pueden ayudar con la administración de la configuración
- AWS Marketplace: productos de CMDB que ayudan a hacer un seguimiento de los límites
- AWS Service Quotas (conocido anteriormente como límites del servicio)
- AWS Trusted Advisor Best Practice Checks (consulte la sección Service Limits)
- Monitor de cuotas para AWS. Solución de AWS
- What is Service Quotas?
- ¿Qué son las plantillas de solicitud de Service Quotas?

Videos relacionados:

- AWS Live re:Inforce 2019 - Service Quotas

- Automating Service Limit Increases and Enterprise Support with AWS Control Tower

Herramientas relacionadas:

- Monitor de cuotas para AWS

REL01-BP06 Garantía de que exista una diferencia suficiente entre las cuotas actuales y el uso máximo para permitir la conmutación por error

En este artículo se explica cómo mantener el espacio entre la cuota de recursos y su uso, y cómo puede beneficiar a su organización. Cuando termine de usar un recurso, es posible que la cuota de uso siga teniendo en cuenta ese recurso. Esto puede provocar un fallo del recurso o que sea inaccesible. Para prevenir el fallo del recurso, compruebe que sus cuotas cubran el solapamiento de los recursos inaccesibles y sus sustitutos. A la hora de calcular esta brecha, tenga en cuenta casos de uso tales como los errores de red, los errores de la zona de disponibilidad o los errores regionales.

Resultado deseado: los errores pequeños o grandes en los recursos o en su accesibilidad pueden cubrirse dentro de los umbrales de servicio actuales. En la planificación de recursos se tienen en cuenta los errores de zona, de red o, incluso, regionales.

Patrones comunes de uso no recomendados:

- Se establecen cuotas de servicio sobre la base de las necesidades actuales sin tener en cuenta los casos de conmutación por error.
- No se tienen en cuenta los principios de estabilidad estática al calcular la cuota máxima de un servicio.
- No se tiene en cuenta el potencial de recursos inaccesibles al calcular la cuota total necesaria para cada región.
- No se tienen en cuenta los límites de aislamiento de errores del servicio de AWS para algunos servicios y sus posibles patrones de uso anómalos.

Beneficios de establecer esta práctica recomendada: cuando los eventos de interrupción del servicio afecten a la disponibilidad de las aplicaciones, utilice la nube para implementar estrategias que le permitan recuperarse de estos eventos. Un ejemplo de estrategia es crear recursos adicionales para reemplazar los recursos inaccesibles y adaptarse a las condiciones de conmutación por error sin agotar el límite de servicio.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Al evaluar el límite de cuota, considere los casos de conmutación por error que podrían producirse debido a algún deterioro. Considere los siguientes casos de conmutación por error.

- Una VPC interrumpida o inaccesible.
- Una subred inaccesible.
- Una zona de disponibilidad degradada que afecta a la accesibilidad de los recursos.
- Rutas de red o puntos de entrada y salida bloqueados o modificados.
- Una región degradada que afecta a la accesibilidad de los recursos.
- Un subconjunto de recursos afectados por un fallo en una región o zona de disponibilidad.

La decisión de utilizar la conmutación por error es única para cada situación, ya que el efecto empresarial puede variar drásticamente. Planifique la capacidad de los recursos en la ubicación de conmutación por error y las cuotas de los recursos antes de decidir llevar a cabo la conmutación por error de una aplicación o un servicio.

Tenga en cuenta los picos de actividad superiores a los normales al revisar las cuotas de cada servicio. Estos picos pueden estar relacionados con recursos que son inaccesibles a causa de la red o los permisos, pero que siguen activos. Los recursos activos no finalizados cuentan para el límite de cuota de servicio.

Pasos para la implementación

- Deje espacio entre la cuota de servicio y el uso máximo para permitir la conmutación por error o una pérdida de accesibilidad.
- Determine sus cuotas de servicio. Tenga en cuenta los patrones de implementación típicos, los requisitos de disponibilidad y el crecimiento del consumo.
- Solicite aumentos de la cuota si fuera necesario. Prevea un tiempo de espera para la solicitud de aumento de cuota.
- Determine sus requisitos de fiabilidad (también conocidos como número de nueves).
- Comprenda los posibles escenarios de error, como la pérdida de un componente, una zona de disponibilidad o una región.
- Establezca su metodología de implementación (por ejemplo, canario, azul/verde, rojo/negro o continua).

- Incluya un búfer adecuado en el límite de cuota actual. Un ejemplo de búfer podría ser del 15 %.
- Incluya cálculos de estabilidad estática (zonal y regional) cuando proceda.
- Planifique el crecimiento del consumo y supervise sus tendencias de consumo.
- Considere la repercusión de la estabilidad estática para las cargas de trabajo más críticas. Evalúe los recursos conforme a un sistema estáticamente estable en todas las regiones y zonas de disponibilidad.
- Considere el uso de reservas de capacidad bajo demanda para programar la capacidad antes de que se produzca una conmutación por error. Es una estrategia útil para implementar las programaciones comerciales críticas a fin de reducir los riesgos potenciales de obtener la cantidad y el tipo correctos de recursos durante la conmutación por error.

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP01 Conocimiento de las cuotas y restricciones del servicio
- REL01-BP02 Administración de cuotas de servicio en cuentas y regiones
- REL01-BP03 Adaptación de las cuotas de servicio fijas y las restricciones a través de la arquitectura
- REL01-BP04 Supervisión y administración de cuotas
- REL01-BP05 Automatización de la administración de cuotas
- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores
- REL11-BP03 Automatización de la reparación en todas las capas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Documentos relacionados:

- AWS Well-Architected Framework’s Reliability Pillar: Availability
- AWS Service Quotas (conocido anteriormente como límites del servicio)
- AWS Trusted Advisor Best Practice Checks (consulte la sección Service Limits)
- Monitor de cuotas para AWS en respuestas de AWS

- Cuotas de servicio de Amazon EC2
- What is Service Quotas?
- How to Request Quota Increase
- Service endpoints and quotas
- Guía del usuario de Service Quotas
- Monitor de cuotas para AWS
- AWS Fault Isolation Boundaries
- Disponibilidad con redundancia
- AWS para datos
- ¿Qué es la integración continua?
- ¿Qué es la entrega continua?
- Socio de APN: socios que pueden ayudar con la administración de la configuración
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS
- Managing and monitoring API throttling in your workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower
- Actions, resources, and condition keys for Service Quotas

Videos relacionados:

- AWS Live re:Inforce 2019 - Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas
- AWS IAM Quotas Demo
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small

Herramientas relacionadas:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch

- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

## REL 2. ¿Cómo planifica la topología de la red?

Las cargas de trabajo suelen existir en varios entornos. Estos incluyen varios entornos de nube (tanto de acceso público como privados) y, posiblemente, la infraestructura de su centro de datos existente. Los planes deben incluir consideraciones de la red, como la conectividad dentro de los sistemas y entre ellos, la administración de las direcciones IP públicas, la administración de las direcciones IP privadas y la resolución de nombres de dominio.

Prácticas recomendadas

- REL02-BP01 Uso de conectividad de red de alta disponibilidad para los puntos de conexión públicos de la carga de trabajo
- REL02-BP02 Aprovisionamiento de conectividad redundante entre las redes privadas en la nube y los entornos en las instalaciones
- REL02-BP03 Garantía de que la asignación de subredes IP tenga en cuenta la expansión y la disponibilidad
- REL02-BP04 Preferencia de topologías radiales (“hub-and-spoke”) frente a una conexión en malla de varios a varios
- REL02-BP05 Aplicación de intervalos de direcciones IP privadas que no se superponen en todos los espacios de direcciones privadas en los que están conectados

### REL02-BP01 Uso de conectividad de red de alta disponibilidad para los puntos de conexión públicos de la carga de trabajo

La creación de una conectividad de red de alta disponibilidad para los puntos de conexión públicos de las cargas de trabajo puede ayudarle a reducir el tiempo de inactividad debido a la pérdida de conectividad y mejorar la disponibilidad y el SLA de su carga de trabajo. Para conseguirlo, use DNS,

redes de entrega de contenido (CDN), puertas de enlace de API, un equilibrador de carga o proxies inversos altamente disponibles.

Resultado deseado: es fundamental planificar, construir y poner en funcionamiento una conectividad de red de alta disponibilidad para sus puntos de conexión públicos. Si la carga de trabajo resulta inaccesible debido a una pérdida de conectividad, incluso si la carga de trabajo está en funcionamiento y disponible, los clientes verán su sistema como caído. Al combinar una conectividad de red de alta disponibilidad y resistente para los puntos de conexión públicos de la carga de trabajo, junto con una arquitectura resistente para la propia carga de trabajo, puede proporcionar la mejor disponibilidad y nivel de servicio posibles a sus clientes.

AWS Global Accelerator, Amazon CloudFront, Amazon API Gateway, las URL de funciones de AWS Lambda, las API de AWS AppSync y Elastic Load Balancing (ELB) ofrecen puntos de conexión públicos de alta disponibilidad. Amazon Route 53 proporciona un servicio de DNS de alta disponibilidad para la resolución de nombres de dominio a fin de comprobar que las direcciones de los puntos de conexión públicos se puedan resolver.

También puede evaluar las aplicaciones de software de AWS Marketplace que proporcionen equilibrio de carga o uso de proxies.

Patrones comunes de uso no recomendados:

- Diseñar una carga de trabajo de alta disponibilidad sin planificar el DNS y la conectividad de red para alta disponibilidad.
- Usar direcciones de internet públicas en instancias o contenedores individuales y administrar la conectividad a ellas a con DNS.
- Usar direcciones IP en lugar de nombres de dominio para localizar los servicios.
- No hacer pruebas de escenarios en que se pierda la conectividad con sus puntos de conexión públicos.
- No analizar las necesidades de rendimiento de la red y los patrones de distribución.
- No hacer pruebas ni planificar escenarios en los que la conectividad de la red de internet a sus puntos de conexión públicos de la carga de trabajo puede ser interrumpida.
- Proporcionar contenido (como páginas web, activos estáticos o archivos multimedia) a una gran área geográfica y no usar una red de entrega de contenido.
- No planificar en caso de que se produzcan ataques de denegación de servicio distribuido (DDoS). Los ataques DDoS corren el riesgo de cerrar el tráfico legítimo y reducir la disponibilidad para sus usuarios.

Beneficios de establecer esta práctica recomendada: diseñar una conectividad de red flexible y de alta disponibilidad garantiza que su carga de trabajo sea accesible y esté disponible para los usuarios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El enrutamiento del tráfico es el núcleo de la creación de una conectividad de red de alta disponibilidad para sus puntos de conexión públicos. Para verificar que el tráfico puede llegar a los puntos de conexión, el DNS debe ser capaz de resolver los nombres de dominio en sus direcciones IP correspondientes. Use un sistema de nombres de dominio (DNS) escalable y de alta disponibilidad, como Amazon Route 53, para administrar los registros de DNS de su dominio. También puede utilizar las comprobaciones de estado proporcionadas por Amazon Route 53. Las comprobaciones de estado verifican que la aplicación sea accesible, esté disponible y funcione; se pueden configurar de manera que imiten el comportamiento de su usuario, como la solicitud de una página web o una URL concreta. En caso de error, Amazon Route 53 responde a las solicitudes de resolución de DNS y dirige el tráfico únicamente a los puntos de conexión en buen estado. También puede plantearse el uso de las capacidades de DNS geográfico y enrutamiento basado en la latencia que ofrece Amazon Route 53.

Para comprobar que la carga de trabajo sea de alta disponibilidad, use Elastic Load Balancing (ELB). Amazon Route 53 se puede utilizar para dirigir el tráfico a ELB, que distribuye el tráfico a las instancias de computación de destino. También puede utilizar Amazon API Gateway junto con AWS Lambda para una solución sin servidor. Los clientes también pueden ejecutar cargas de trabajo en varias Regiones de AWS. Con un patrón activo/activo de varios sitios, la carga de trabajo puede atender el tráfico de varias regiones. Con un patrón activo/pasivo de varios sitios, la carga de trabajo atiende el tráfico de la región activa, mientras que los datos se replican en la región secundaria y se activan en caso de que se produzca un error en la región principal. Las comprobaciones de estado de Route 53 se pueden utilizar para controlar la conmutación por error de DNS desde cualquier punto de conexión de una región principal a un punto de conexión de una región secundaria, lo que permite comprobar que los usuarios tengan acceso a la carga de trabajo y que esta esté disponible para ellos.

Amazon CloudFront proporciona una API sencilla para distribuir contenido con baja latencia y altas velocidades de transferencia de datos, ya que atiende las solicitudes mediante una red de ubicaciones periféricas en todo el mundo. Las redes de entrega de contenido (CDN) atienden a los clientes al proporcionarles contenido ubicado o almacenado en caché en una ubicación cercana

al usuario. Esto también mejora la disponibilidad de la aplicación, ya que la carga de contenido se traslada de los servidores a las ubicaciones periféricas de CloudFront. Las ubicaciones periféricas y las cachés periféricas regionales mantienen copias en caché de su contenido cerca de sus usuarios, lo que permite una recuperación rápida y aumenta la accesibilidad y la disponibilidad de su carga de trabajo.

Para cargas de trabajo con usuarios distribuidos geográficamente, AWS Global Accelerator ayuda a mejorar la disponibilidad y el rendimiento de las aplicaciones. AWS Global Accelerator proporciona direcciones IP estáticas anycast que sirven como punto de entrada fijo a su aplicación alojada en una o más Regiones de AWS. Esto permite que el tráfico entre en la red global de AWS lo más cerca posible de sus usuarios, lo que mejora la accesibilidad y disponibilidad de su carga de trabajo. AWS Global Accelerator también supervisa el estado de los puntos de conexión de su aplicación mediante comprobaciones de estado de TCP, HTTP y HTTPS. Cualquier cambio en el estado o la configuración de sus puntos de conexión permite el redireccionamiento del tráfico de usuario a puntos de conexión en buen estado que ofrezcan el mejor rendimiento y disponibilidad a los usuarios. Además, AWS Global Accelerator cuenta con un diseño de aislamiento de errores que utiliza dos direcciones IPv4 estáticas atendidas por zonas de red independientes que aumentan la disponibilidad de las aplicaciones.

Para ayudar a proteger a los clientes de los ataques DDoS, AWS proporciona AWS Shield Standard. Shield Estándar se activa automáticamente y protege de los ataques comunes a la infraestructura (capas 3 y 4), como las inundaciones SYN/UDP y los ataques de reflexión, para respaldar la alta disponibilidad de sus aplicaciones en AWS. Para obtener protecciones adicionales contra ataques más sofisticados y grandes (como inundaciones UDP) y ataques de agotamiento de estado (como inundaciones de TCP SYN), y para ayudar a proteger sus aplicaciones que se ejecutan en Amazon Elastic Compute Cloud (Amazon EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator y Route 53, puede considerar el uso de AWS Shield Advanced. Para la protección contra ataques en la capa de aplicación como HTTP POST o inundaciones GET, utilice AWS WAF. AWS WAF puede utilizar condiciones de direcciones IP, encabezados HTTP, cuerpos HTTP, cadenas de URI, inyección de código SQL y scripting entre sitios para determinar si se debe bloquear o permitir una solicitud.

Pasos para la implementación

1. Configuración de un DNS de alta disponibilidad: Amazon Route 53 es un servicio web de sistema de nombres de dominio (DNS) escalable y de alta disponibilidad. Route 53 conecta las solicitudes de los usuarios con las aplicaciones de Internet que se ejecutan en AWS o en las instalaciones. Para obtener más información, consulte Configuring Amazon Route 53 as your DNS service.

2. Configuración de comprobaciones de estado: cuando utilice Route 53, verifique que solo se puedan resolver los destinos en buen estado. Comience por la creación de comprobaciones de estado de Amazon Route 53 y la configuración de la conmutación por error de DNS. Es importante tener en cuenta los siguientes aspectos a la hora de configurar las comprobaciones de estado:
    a. How Amazon Route 53 determines whether a health check is healthy
    b. Creating, updating, and deleting health checks
    c. Monitoring health check status and getting notifications
    d. Best practices for Amazon Route 53 DNS
3. Conecte su servicio de DNS a los puntos de conexión.
    a. Cuando utilice Elastic Load Balancing como objetivo para el tráfico, cree un registro de alias con Amazon Route 53 que apunte al punto de conexión regional del equilibrador de carga. Durante la creación del registro de alias, establezca la opción de evaluación de estado del destino en Sí.
    b. Utilice Route 53 para dirigir el tráfico a API Gateway para las cargas de trabajo sin servidor o API privadas.
4. Decida la red de entrega de contenido.
    a. Para entregar contenido con ubicaciones periféricas más cercanas al usuario, antes debe entender cómo entrega contenido CloudFront.
    b. Comience con una distribución sencilla de CloudFront. CloudFront sabrá entonces desde dónde desea que se entregue el contenido, así como los detalles sobre cómo hacer el seguimiento y administrar la entrega de contenido. Es importante comprender y tener en cuenta los siguientes aspectos al configurar la distribución de CloudFront:
        i. Cómo funciona el almacenamiento en caché con ubicaciones periférica de CloudFront
        ii. Incremento de la proporción de solicitudes que se atienden directamente desde las cachés de CloudFront (tasa de aciertos de caché)
        iii. Uso de Origin Shield de Amazon CloudFront
        iv. Optimización de alta disponibilidad con conmutación por error de origen de CloudFront
5. Configuración de la protección de la capa de aplicación: AWS WAF le ayuda a protegerse contra ataques web y bots habituales que pueden afectar a la disponibilidad, comprometer la seguridad o consumir demasiados recursos. Para obtener una comprensión más profunda, consulte How AWS WAF works y, cuándo lo tenga todo listo para implementar protecciones contra las inundaciones HTTP, POST y GET de la capa de aplicaciones, consulte Getting started with AWS WAF. También

puede usar AWS WAF con CloudFront. Consulte la documentación sobre cómo funciona AWS WAF con las características de Amazon CloudFront.

6. Configuración de protección DDoS adicional: de forma predeterminada, todos los clientes de AWS reciben protección frente a los ataques DDoS más habituales y frecuentes de la capa de red y transporte dirigidos a su sitio web o aplicación con AWS Shield Standard y sin ningún cargo adicional. Para obtener una protección adicional de las aplicaciones con acceso a Internet que se ejecutan en Amazon EC2, Elastic Load Balancing, Amazon CloudFront, AWS Global Accelerator y Amazon Route 53, puede considerar AWS Shield Advanced y ver ejemplos de arquitecturas resistentes a DDoS. Para proteger su carga de trabajo y sus puntos de conexión públicos de los ataques DDoS, consulte Getting started with AWS Shield Advanced.

Recursos

Prácticas recomendadas relacionadas:

- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP04 Confianza en el plano de datos y no en el plano de control durante la recuperación
- REL11-BP06 Envío de notificaciones cuando los eventos afecten a la disponibilidad

Documentos relacionados:

- Socio de APN: socios que pueden ayudarle a planificar sus redes
- AWS Marketplace para la infraestructura de red
- Qué es AWS Global Accelerator?
- ¿Qué es Amazon CloudFront?
- What is Amazon Route 53?
- Qué es Elastic Load Balancing?
- Network Connectivity capability - Establishing Your Cloud Foundations
- ¿Qué es Amazon API Gateway?
- What are AWS WAF, AWS Shield, and AWS Firewall Manager?
- What is Amazon Application Recovery Controller?
- Configuración de las comprobaciones de estado personalizadas para la conmutación por error de DNS

Videos relacionados:

- AWS re:Invent 2022 - Improve performance and availability with AWS Global Accelerator
- AWS re:Invent 2020: Global traffic management with Amazon Route 53
- AWS re:Invent 2022 - Operating highly available Multi-AZ applications
- AWS re:Invent 2022 - Dive deep on AWS networking infrastructure
- AWS re:Invent 2022 - Building resilient networks

Ejemplos relacionados:

- Disaster Recovery with Amazon Application Recovery Controller (ARC)
- AWS Global Accelerator Workshop

REL02-BP02 Aprovisionamiento de conectividad redundante entre las redes privadas en la nube y los entornos en las instalaciones

Implemente la redundancia en las conexiones entre redes privadas en la nube y entornos en las instalaciones para lograr la resiliencia de la conectividad. Esto se puede lograr mediante la implementación de dos o más enlaces y rutas de tráfico, lo que permite mantener la conectividad en el caso de que se produzcan errores en la red.

Patrones comunes de uso no recomendados:

- Depende de una única conexión de red, lo que crea un único punto de error.
- Utiliza únicamente un túnel de VPN o varios túneles que terminan en la misma zona de disponibilidad.
- Confía en un único proveedor de servicios de Internet (ISP) para la conectividad de VPN, lo que puede provocar un fallo total de la conexión durante las interrupciones del ISP.
- No implementar protocolos de enrutamiento dinámico como BGP, que son fundamentales para redirigir el tráfico durante las interrupciones de la red.
- Ignora las limitaciones de ancho de banda de los túneles de VPN y sobrestima sus capacidades de copia de seguridad.

Beneficios de establecer esta práctica recomendada: al implementar la conectividad redundante entre el entorno en la nube y su entorno corporativo o en las instalaciones, puede garantizar que los servicios dependientes entre los dos entornos se puedan comunicar con fiabilidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Al utilizar AWS Direct Connect para conectar la red en las instalaciones a AWS, puede lograr la máxima resiliencia de la red (SLA del 99,99 %) mediante el uso de conexiones independientes que terminan en distintos dispositivos en más de una ubicación en las instalaciones y en más de una ubicación de AWS Direct Connect. Esta topología ofrece resiliencia frente a los errores de los dispositivos, los problemas de conectividad y las interrupciones totales de la conexión producidas en la ubicación. Como alternativa, puede lograr una alta resiliencia (SLA del 99,9 %) mediante el uso de dos conexiones individuales a varias ubicaciones (cada ubicación en las instalaciones conectada a una única ubicación de Direct Connect). Este enfoque protege frente a las interrupciones de conectividad provocadas por cortes en la fibra o errores en los dispositivos y ayuda a mitigar los fallos totales de la conexión producidos en la ubicación. El Kit de herramientas de resiliencia de Direct Connect puede ayudarle a diseñar su topología de AWS Direct Connect.

También puede plantearse la posibilidad de utilizar AWS Site-to-Site VPN que finaliza en una AWS Direct Connect como una opción de conexión alternativa y rentable en el caso de que surjan problemas con la conexión principal de AWS Transit Gateway. Esta configuración permite el enrutamiento de múltiples rutas de igual costo (ECMP) a través de varios túneles de VPN, lo que permite un rendimiento de hasta 50 Gbps, aunque cada túnel de VPN tenga un límite de 1,25 Gbps. Sin embargo, es importante tener en cuenta que AWS Direct Connect sigue siendo la opción más eficaz para reducir al mínimo las interrupciones producidas en la red y proporcionar una conectividad estable.

Al utilizar VPN a través de Internet para conectar el entorno de nube al centro de datos en las instalaciones, configure dos túneles de VPN como parte de una única conexión de Site-to-Site VPN. Cada túnel debe terminar en una zona de disponibilidad diferente para lograr una alta disponibilidad y usar hardware redundante con el fin de evitar errores en los dispositivos en las instalaciones. Asimismo, tenga en cuenta la posibilidad de establecer varias conexiones a Internet de varios proveedores de servicios de Internet (ISP) en su ubicación en las instalaciones para evitar una interrupción total de la conectividad de la VPN debido a una única interrupción del ISP. La selección de ISP con enrutamientos e infraestructuras diversos, especialmente aquellos con rutas físicas independientes a los puntos de conexión de AWS, proporciona una alta disponibilidad de la conectividad.

Además de la redundancia física con varias conexiones de AWS Direct Connect y varios túneles de VPN (o una combinación de ambos), también es fundamental implementar el enrutamiento dinámico del protocolo de puerta de enlace fronteriza (BGP). El BGP dinámico permite redirigir

automáticamente el tráfico de una ruta a otra en función de las condiciones de la red en tiempo real y las políticas configuradas. Este comportamiento dinámico es especialmente beneficioso para mantener la disponibilidad de la red y la continuidad del servicio en caso de que se produzcan errores de enlace o en la red. Selecciona rápidamente rutas alternativas, lo que mejora la resiliencia y la fiabilidad de la red.

Pasos para la implementación

- Establezca una conectividad de alta disponibilidad entre AWS y el entorno en las instalaciones.
  - Use varias conexiones de AWS Direct Connect o túneles de VPN entre redes privadas implementadas por separado.
  - Use varias ubicaciones de Direct Connect para contar con alta disponibilidad.
  - Si utiliza varias Regiones de AWS, cree redundancia en al menos dos de ellas.
- Use AWS Transit Gateway, cuando sea posible, para finalizar su conexión VPN.
- Evalúe los dispositivos de AWS Marketplace para finalizar las VPN o ampliar su SD-WAN a AWS. Si utiliza dispositivos de AWS Marketplace, implemente instancias redundantes para obtener alta disponibilidad en diferentes zonas de disponibilidad.
- Proporcione una conexión redundante al entorno en las instalaciones.
  - Es posible que necesite conexiones redundantes a varias Regiones de AWS para cubrir sus necesidades de disponibilidad.
  - Use el Kit de herramientas de resiliencia de Direct Connect para comenzar.

Recursos

Documentos relacionados:

- Recomendaciones sobre resiliencia de AWS Direct Connect
- Using Redundant Site-to-Site VPN Connections to Provide Failover
- Routing policies and BGP communities
- Active/Active and Active/Passive Configurations in AWS Direct Connect
- Socio de APN: socios que pueden ayudarle a planificar sus redes
- AWS Marketplace para la infraestructura de red
- Documento técnico sobre las opciones de conectividad a la nube virtual privada de Amazon
- Creación de una infraestructura de red de AWS multiVPC escalable y segura

- Using redundant Site-to-Site VPN connections to provide failover
- Using the Direct Connect Resiliency Toolkit to get started
- Puntos de conexión de VPC y servicios de punto de conexión de VPC (AWS PrivateLink)
- Qué es Amazon VPC?
- What is a transit gateway?
- Qué es AWS Site-to-Site VPN?
- Uso de puertas de enlace de Direct Connect

Videos relacionados:

- AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs

REL02-BP03 Garantía de que la asignación de subredes IP tenga en cuenta la expansión y la disponibilidad

Los intervalos de direcciones IP de Amazon VPC deben ser lo suficientemente amplios como para dar cabida a los requisitos de las cargas de trabajo, como la posible expansión futura y la asignación de direcciones IP a las subredes de las zonas de disponibilidad. Esto incluye equilibradores de carga, instancias de EC2 y aplicaciones basadas en contenedores.

Cuando planifica la topología de su red, el primer paso es definir el espacio de la dirección IP. Se deben asignar rangos de direcciones IP privadas para cada VPC (según las directrices de la RFC 1918). Facilite los siguientes requisitos como parte de este proceso:

- Permita los espacios de direcciones IP para más de una VPC por región.
- En una VPC, deje espacio para varias subredes para poder cubrir varias zonas de disponibilidad.
- Considere la posibilidad de dejar siempre un espacio de bloque de CIDR sin usar en una VPC para posibles expansiones futuras.
- Asegúrese de que haya espacio de direcciones IP suficiente como para satisfacer las necesidades de flotas transitorias de instancias de Amazon EC2 que podría usar, como flotas de spot para el machine learning, clústeres de Amazon EMR o clústeres de Amazon Redshift. Se debe prestar una atención similar a los clústeres de Kubernetes, como Amazon Elastic Kubernetes Service (Amazon EKS), ya que a cada pod de Kubernetes se le asigna una dirección enrutable desde el bloque de CIDR de la VPC de forma predeterminada.

- Tenga en cuenta que las primeras cuatro direcciones IP y la última dirección IP de cada bloque CIDR de subred están reservadas y no están disponibles para que las use.
- Tenga en cuenta que el bloque de CIDR de la VPC inicial asignado a su VPC no debe cambiar ni eliminarse, pero puede agregar bloques de CIDR que no se solapen a la VPC. Los CIDR IPv4 de subred no se pueden cambiar; sin embargo, los CIDR IPv6 sí.
- El bloque de CIDR de VPC más grande posible es un /16 y el más pequeño es un /28.
- Tenga en cuenta otras redes conectadas (VPC, en las instalaciones u otros proveedores de nube) y asegúrese de que el espacio de direcciones IP no se superponga. Para obtener más información, consulte REL02-BP05 Aplicación de intervalos de direcciones IP privadas que no se superponen en todos los espacios de direcciones privadas en los que están conectados.

Resultado deseado: una subred IP escalable puede ayudarle a adaptarse al crecimiento futuro y evitar desperdicios innecesarios.

Patrones comunes de uso no recomendados:

- No tener en cuenta el crecimiento futuro, lo que hace que los bloques de CIDR sean demasiado pequeños y se tengan que reconfigurar, lo que puede provocar tiempos de inactividad a su vez.
- Calcular incorrectamente cuántas direcciones IP puede usar un equilibrador de carga elástico.
- Implementar muchos equilibradores de carga de tráfico intenso en las mismas subredes
- Usar mecanismos de escalado automatizados sin supervisar el consumo de direcciones IP.
- Definir rangos de CIDR excesivamente grandes que superen con creces las expectativas de crecimiento futuro, lo que puede generar dificultades para conectarse con otras redes con rangos de direcciones superpuestos.

Beneficios de establecer esta práctica recomendada: de esta forma, se asegurará de dar cabida al crecimiento de sus cargas de trabajo y seguir proporcionando disponibilidad al escalar verticalmente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Planificar su red para que se adapte al crecimiento, el cumplimiento normativo y la integración con otros. El crecimiento se puede subestimar, la conformidad normativa puede variar y las adquisiciones y conexiones de redes privadas pueden ser difíciles de llevar a cabo sin una planificación adecuada.

- Seleccione las regiones y Cuentas de AWS que correspondan según sus requisitos normativos y de servicio, latencia y recuperación de desastres (DR).
- Identifique sus necesidades para implementaciones regionales de VPC.
- Identifique el tamaño de las VPC.
  - Determine si va a implementar la conectividad de varias VPC.
    - What Is a Transit Gateway?
    - Single Region Multi-VPC Connectivity
  - Determine si necesita redes divididas conforme a los requisitos normativos.
  - Cree VPC con bloques de CIDR del tamaño adecuado para adaptarse a sus necesidades actuales y futuras.
    - Si desconoce sus proyecciones de crecimiento, es posible que desee optar por bloques de CIDR más grandes para no tener que volver a configurarlos en el futuro
  - Considere la posibilidad de utilizar las direcciones IPv6 para las subredes como parte de una VPC de doble pila. IPv6 es ideal en subredes privadas que contengan flotas de instancias o contenedores efímeros que, de otro modo, requerirían un gran número de direcciones IPv4.

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- REL02-BP05 Aplicación de intervalos de direcciones IP privadas que no se superponen en todos los espacios de direcciones privadas en los que están conectados

Documentos relacionados:

- Socio de APN: socios que pueden ayudarle a planificar sus redes
- AWS Marketplace para la infraestructura de red
- Documento técnico sobre las opciones de conectividad a la nube virtual privada de Amazon
- Conectividad a la red de varios centros de datos HA
- Single Region Multi-VPC Connectivity
- Qué es Amazon VPC?
- IPv6 en AWS
- IPv6 on reference architectures

- Amazon Elastic Kubernetes Service launches IPv6 support
- Recommendations for your VPC - Classic Load Balancers
- Subredes de zona de disponibilidad - Equilibrador de carga de aplicación
- Zonas de disponibilidad - Equilibradores de carga de red

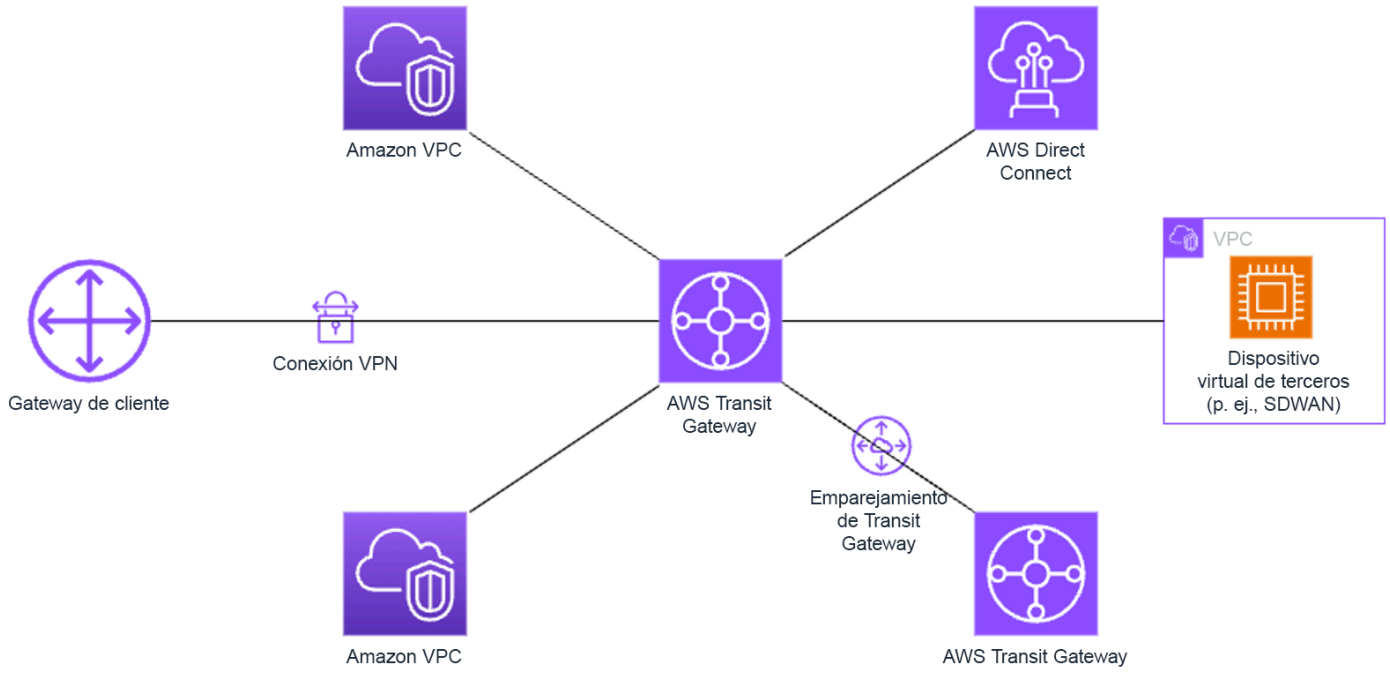
Videos relacionados:

- AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC (NET303)
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs (NET406-R1)
- AWS re:Invent 2023: AWS Ready for what's next? Designing networks for growth and flexibility (NET310)

REL02-BP04 Preferencia de topologías radiales (“hub-and-spoke”) frente a una conexión en malla de varios a varios

Al conectar varias redes privadas, como nubes privadas virtuales (VPC) y redes en las instalaciones, opte por una topología radial (“hub-and-spoke”) en lugar de una en malla. A diferencia de las topologías en malla, en las que cada red se conecta directamente a las demás y aumenta la complejidad y la sobrecarga de administración, la arquitectura radial (“hub-and-spoke”) centraliza las conexiones a través de un único hub. Esta centralización simplifica la estructura de la red y mejora su operatividad, escalabilidad y control.

AWS Transit Gateway es un servicio administrado, escalable y de alta disponibilidad diseñado para la construcción de redes radiales (“hub-and-spoke”) en AWS. Sirve como centro de la red que proporciona una segmentación de la red, un enrutamiento centralizado y una conexión simplificada a los entornos en las instalaciones y en la nube. La siguiente figura muestra cómo puede utilizar AWS Transit Gateway para crear su topología radial (“hub-and-spoke”).




Resultado deseado: ha conectado las nubes privadas virtuales (VPC) y las redes en las instalaciones a través de un concentrador central. Las conexiones de emparejamiento se configuran a través del hub, que actúa como un enrutador en la nube altamente escalable. El enrutamiento se simplifica porque no es necesario trabajar con relaciones de interconexión complejas. El tráfico entre redes está cifrado y tiene la capacidad de aislar las redes.

Patrones comunes de uso no recomendados:

- Crea reglas de interconexión de redes complejas.
- Proporciona rutas entre redes que no deberían comunicarse entre sí (por ejemplo, cargas de trabajo independientes que no tienen interdependencias).
- La gobernanza de la instancia central es ineficaz.

Beneficios de establecer esta práctica recomendada: a medida que aumenta la cantidad de redes conectadas, la administración y la expansión de la conectividad en malla se vuelven cada vez más desafiantes. Una arquitectura de malla presenta desafíos adicionales, como componentes de infraestructura adicionales, requisitos de configuración y cuestiones de implementación. La malla también introduce una sobrecarga adicional para administrar y monitorear el plano de datos y los componentes del plano de control. Debe pensar en cómo proporcionar una alta disponibilidad de la

arquitectura de malla, cómo monitorear el estado y el rendimiento de la malla y cómo gestionar las actualizaciones de los componentes de la malla.

Por otro lado, un modelo radial establece un enrutamiento de tráfico centralizado en varias redes. Simplifica la estrategia de administración y monitoreo del plano de datos y los componentes del plano de control.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cree una cuenta de servicios de red si no hay ninguna. Coloque el hub en la cuenta de servicios de red de la organización. Este enfoque permite que los ingenieros de redes administren el hub de forma centralizada.

El hub del modelo radial funciona como un enrutador virtual del flujo de tráfico entre las nubes privadas virtuales (VPC) y las redes en las instalaciones. Esta estrategia reduce la complejidad de la red y facilita la resolución de problemas de red.

Tenga en cuenta el diseño de su red, incluidas las VPC, AWS Direct Connect y las conexiones VPN de sitio a sitio que desee interconectar.

Plantéese utilizar una subred independiente para cada archivo asociado a la VPC de la puerta de enlace de tránsito. En cada subred, utilice un CIDR pequeño, (por ejemplo /28), a fin de tener más espacio de direcciones para los recursos de computación. Además, cree una única ACL de red y asóciela a todas las subredes relacionadas con el hub. Mantenga abierta la ACL de red tanto en las direcciones de entrada como de salida.

Diseñe e implemente sus tablas de enrutamiento de manera que las rutas se establezcan solo entre redes que deban comunicarse. Omita rutas entre redes que no deberían comunicarse entre sí (por ejemplo, cargas de trabajo independientes que no tienen interdependencias).

Pasos para la implementación

1. Planifique su red. Determine qué redes desea conectar y compruebe que no compartan rangos de CIDR superpuestos.
2. Cree una AWS Transit Gateway y asocie sus VPC.
3. Si es necesario, cree conexiones VPN o puertas de enlace de Direct Connect y asócielas a la instancia de Transit Gateway.

4. Defina cómo se dirige el tráfico entre las VPC conectadas y otras conexiones mediante la configuración de las tablas de enrutamiento de Transit Gateway.
5. Utilice Amazon CloudWatch para supervisar y ajustar las configuraciones según sea necesario para optimizar el rendimiento y los costos.

Recursos

Prácticas recomendadas relacionadas:

- REL02-BP03 Garantía de que la asignación de subredes IP tenga en cuenta la expansión y la disponibilidad
- REL02-BP05 Aplicación de intervalos de direcciones IP privadas que no se superponen en todos los espacios de direcciones privadas en los que están conectados

Documentos relacionados:

- What Is a Transit Gateway?
- Prácticas recomendadas de diseño de una puerta de enlace de tránsito
- Creación de una infraestructura de red de AWS multiVPC escalable y segura
- Building a global network using AWS Transit Gateway Inter-Region peering
- Opciones de conectividad de Amazon Virtual Private Cloud
- Socio de APN: socios que pueden ayudarle a planificar sus redes
- AWS Marketplace para la infraestructura de red

Videos relacionados:

- AWS re:Invent 2023 - AWS networking foundations
- AWS re:Invent 2023 - Advanced VPC designs and new capabilities

Talleres relacionados:

- AWS Transit Gateway Workshop

REL02-BP05 Aplicación de intervalos de direcciones IP privadas que no se superponen en todos los espacios de direcciones privadas en los que están conectados

Los intervalos de direcciones IP de cada VPC no deben superponerse si se emparejan o conectan mediante Transit Gateway o VPN. Evite conflictos de direcciones IP entre una VPC y los entornos en las instalaciones o con otros proveedores de servicios en la nube que utilice. También debe tener una forma de asignar intervalos de direcciones IP privadas cuando sea necesario. Un sistema Administrador de direcciones IP (IPAM) puede ayudar en esta automatización.

Resultado deseado:

- No hay conflictos de intervalo de direcciones IP entre VPC, entornos en las instalaciones u otros proveedores de servicios en la nube.
- La administración adecuada de las direcciones IP permite escalar de forma más sencilla la infraestructura de red para adaptarse al crecimiento y los cambios en los requisitos de la red.

Patrones comunes de uso no recomendados:

- Usar el mismo intervalo de direcciones IP en la VPC que en el entorno en las instalaciones, en la red corporativa o en otros proveedores de servicios en la nube
- No controlar los intervalos de direcciones IP de las VPC usadas para implementar sus cargas de trabajo.
- Confiar en los procesos manuales de administración de direcciones IP, como, por ejemplo, las hojas de cálculo.
- Sobredimensionar o infradimensionar bloques de CIDR, lo que provoca un derroche en el uso de direcciones IP o un espacio insuficiente para las direcciones de la carga de trabajo.

Beneficios de establecer esta práctica recomendada: la planificación activa de la red garantizará que no tenga varias instancias de la misma dirección IP en las redes interconectadas. Con esto evitará que se produzcan problemas de enrutamiento en las partes de la carga de trabajo que usan las diferentes aplicaciones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Utilice un IPAM, como el Administrador de direcciones IP de Amazon VPC, para supervisar y administrar el uso del CIDR. En AWS Marketplace, también hay disponibles varios IPAM. Evalúe su

potencial de uso en AWS, agregue intervalos de CIDR a las VPC existentes y cree VPC para permitir un crecimiento planificado del uso.

Pasos para la implementación

- Capture el consumo actual de CIDR (por ejemplo, VPC y subredes).
  - Use operaciones de la API de servicio para recopilar el consumo actual de CIDR.
  - Utilice el Administrador de direcciones IP de Amazon VPC para detectar recursos.
- Registre el uso actual de la subred.
  - Use operaciones de la API de servicio para recopilar las subredes por VPC en cada región.
  - Utilice el Administrador de direcciones IP de Amazon VPC para detectar recursos.
- Registre el uso actual.
- Determine si creó intervalos de direcciones IP superpuestos.
- Calcule la capacidad de reserva.
- Identifique los intervalos de direcciones IP superpuestos. Puede migrar a un nuevo intervalo de direcciones o considerar la posibilidad de utilizar técnicas como una puerta de enlace NAT privada o AWS PrivateLink si necesita conectar los intervalos superpuestos.

Recursos

Prácticas recomendadas relacionadas:

- Protección de redes

Documentos relacionados:

- Socio de APN: socios que pueden ayudarle a planificar sus redes
- AWS Marketplace para la infraestructura de red
- Documento técnico sobre las opciones de conectividad a la nube virtual privada de Amazon
- Conectividad a la red de varios centros de datos HA
- Connecting Networks with Overlapping IP Ranges
- Qué es Amazon VPC?
- ¿Qué es IPAM?

Videos relacionados:

- AWS re:Invent 2023 - Advanced VPC designs and new capabilities
- AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs
- AWS re:Invent 2023 - Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2021 - {New Launch} Manage your IP addresses at scale on AWS

## Arquitectura de la carga de trabajo

Preguntas

- REL 3. ¿Cómo diseña la arquitectura de servicio de su carga de trabajo?
- REL 4. ¿Cómo diseña las interacciones en un sistema distribuido para evitar los errores?
- REL 5. ¿Cómo se diseñan las interacciones en un sistema distribuido para mitigar o resistir los errores?

### REL 3. ¿Cómo diseña la arquitectura de servicio de su carga de trabajo?

Desarrolle cargas de trabajo escalables y fiables mediante una arquitectura orientada a servicios (SOA) o una arquitectura de microservicios. La arquitectura orientada a servicios (SOA) es hacer que los componentes de software se puedan reutilizar mediante interfaces de servicio. La arquitectura de microservicios va más allá, para hacer que los componentes sean más pequeños y sencillos.

Prácticas recomendadas

- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL03-BP02 Desarrollo de servicios centrados en funcionalidades y dominios empresariales específicos
- REL03-BP03 Disposición de contratos de servicio por cada API

#### REL03-BP01 Elección de cómo segmentar su carga de trabajo

La segmentación de la carga de trabajo es importante a la hora de determinar los requisitos de resiliencia de su aplicación. La arquitectura monolítica debe evitarse siempre que sea posible. En su lugar, considere detenidamente qué componentes de la aplicación pueden dividirse en microservicios. Según los requisitos de su aplicación, esto puede terminar siendo una combinación de una arquitectura orientada a servicios (SOA) con microservicios cuando sea posible. Las

cargas de trabajo que son capaces de no tener estado son más capaces de implementarse como microservicios.

Resultado deseado: las cargas de trabajo se deben admitir, ser escalables y tener el acoplamiento más débil que sea posible.

A la hora de elegir cómo segmentar la carga de trabajo, hay que sopesar las ventajas frente a las complejidades. Lo que puede ser adecuado para un nuevo producto encaminado a su primer lanzamiento es diferente a lo que necesita una carga de trabajo creada para escalarse desde el principio. Al refactorizar un monolito existente, tendrá que considerar en qué medida soportará la aplicación una descomposición hacia la falta de estado. Dividir los servicios en partes más pequeñas permite que equipos pequeños y bien definidos los desarrollen y administren. No obstante, los servicios más pequeños pueden introducir complejidades que incluyen un aumento de la latencia, una depuración más compleja y un mayor lastre operativo.

Patrones comunes de uso no recomendados:

- La Estrella de la muerte de microservicios es una situación en la que los componentes atómicos son tan interdependientes que el error de uno de ellos provoca un error mucho mayor, lo que hace que los componentes sean tan rígidos y frágiles como un monolito.

Beneficios de establecer esta práctica:

- Los segmentos más específicos conducen a una mayor agilidad, flexibilidad organizativa y escalabilidad.
- Reducción del impacto de las interrupciones del servicio.
- Los componentes de la aplicación pueden tener diferentes requisitos de disponibilidad, que pueden soportarse mediante una segmentación más atómica.
- Responsabilidades bien definidas para los equipos que apoyan la carga de trabajo.

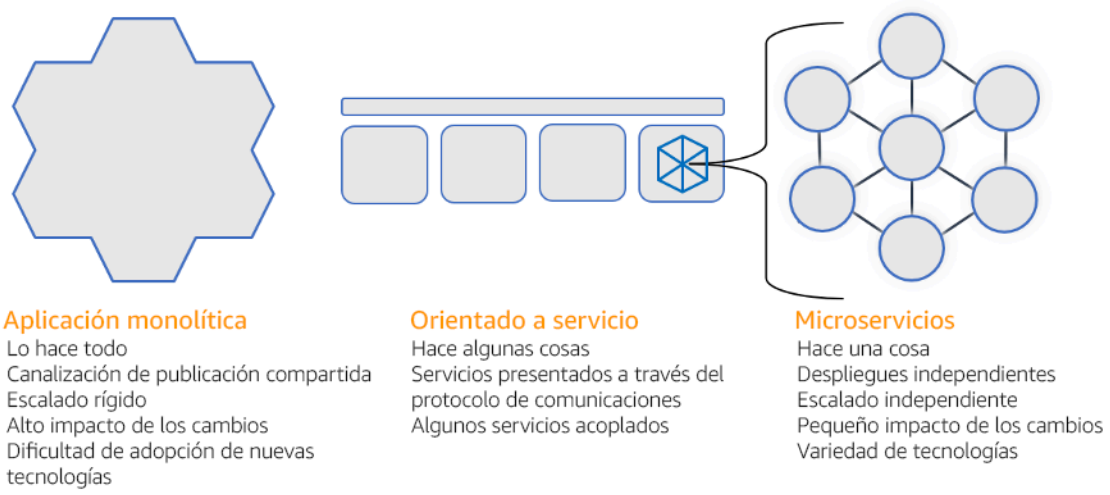
Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Seleccione el tipo de arquitectura en función de cómo va a segmentar su carga de trabajo. Seleccione una SOA o una arquitectura de microservicios (o, en algunos casos raros, una arquitectura monolítica). Incluso si decide empezar con una arquitectura monolítica, debe asegurarse de que sea modular y de que pueda evolucionar hacia SOA o microservicios de forma definitiva,

a medida que su producto escala con la adopción por parte de los usuarios. La SOA y los microservicios ofrecen respectivamente una segmentación más pequeña, lo que resulta preferible como arquitectura moderna escalable y fiable, pero existen compensaciones a tener en cuenta, especialmente al implementar una arquitectura de microservicios.

Una compensación principal es que se dispone de una arquitectura de computación distribuida que puede dificultar el cumplimiento de los requisitos de latencia del usuario y existe una complejidad adicional en la depuración y el rastreo de las interacciones del usuario. Puede utilizar AWS X-Ray para ayudarle a resolver este problema. Otro efecto que hay que tener en cuenta es el aumento de la complejidad operativa a medida que aumenta el número de aplicaciones que se administran, lo que requiere la implementación de componentes con varias independencias.




Arquitecturas monolíticas, orientadas al servicio y de microservicios

Pasos para la implementación

- Determine la arquitectura adecuada para refactorizar o desarrollar su aplicación. La SOA y los microservicios ofrecen respectivamente una segmentación más pequeña, lo que resulta preferible como arquitectura moderna escalable y fiable. La SOA puede ofrecer un término intermedio ideal para conseguir una segmentación más pequeña y, a la vez, evitar algunas de las complejidades de los microservicios. Para obtener más información, consulte Microservice Trade-Offs.
- Si su carga de trabajo lo admite y su organización puede permitírselo, debería usar una arquitectura de microservicios para conseguir la mejor agilidad y fiabilidad. Para obtener más información, consulte Implementing Microservices on AWS.
- Considere la posibilidad de seguir el patrón del higo estrangulador para refactorizar un monolito en componentes más pequeños. Esto implica reemplazar gradualmente componentes específicos

de la aplicación por nuevas aplicaciones y servicios. AWS Migration Hub Refactor Spaces actúa como punto de partida para la refactorización incremental. Para obtener más información, consulte Seamlessly migrate on-premises legacy workloads using a strangler pattern.

- La implementación de microservicios puede necesitar de un mecanismo de detección de servicios que permita que estos servicios distribuidos se comuniquen entre sí. AWS App Mesh se puede utilizar con arquitecturas orientadas a los servicios para proporcionar una detección y un acceso fiables a estos. AWS Cloud Map también se puede utilizar para la detección dinámica de servicios basada en DNS.
- Si va a migrar de un entorno monolítico a SOA, Amazon MQ puede ayudarle a cerrar la brecha, en calidad de bus de servicio, a la hora de rediseñar aplicaciones heredadas en la nube.
- Para los monolitos existentes con una única base de datos compartida, elija cómo reorganizar los datos en segmentos más pequeños. Puede ser por unidad de negocio, patrón de acceso o estructura de datos. En este punto del proceso de refactorización, debe elegir entre una base de datos de tipo relacional o no relacional (NoSQL). Para obtener más información, consulte From SQL to NoSQL.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- REL03-BP02 Desarrollo de servicios centrados en funcionalidades y dominios empresariales específicos

Documentos relacionados:

- Amazon API Gateway: Configuring a REST API Using OpenAPI
- ¿Qué es la arquitectura orientada a servicios (SOA)?
- Bounded Context (a central pattern in Domain-Driven Design)
- Implementing Microservices on AWS
- Microservice Trade-Offs
- Microservices - a definition of this new architectural term
- Microservicios en AWS
- Qué es AWS App Mesh?

Ejemplos relacionados:

- Iterative App Modernization Workshop

Videos relacionados:

- Delivering Excellence with Microservices on AWS

REL03-BP02 Desarrollo de servicios centrados en funcionalidades y dominios empresariales específicos

La arquitectura orientada a servicios (SOA) define servicios con funciones bien delineadas y determinadas por necesidades empresariales. Los microservicios utilizan modelos de dominio y contextos delimitados para trazar los límites de los servicios en los límites del contexto empresarial. Centrarse en los dominios y las funcionalidades empresariales ayuda a los equipos a definir requisitos de fiabilidad independientes para sus servicios. Los contextos delimitados aíslan y encapsulan la lógica empresarial, lo que permite a los equipos mejorar la forma en que gestionan los errores.

Resultado deseado: los ingenieros y las partes interesadas de la empresa definen conjuntamente los contextos delimitados y los utilizan para diseñar sistemas como servicios que cumplan funciones empresariales específicas. Estos equipos utilizan prácticas establecidas, como las tormentas de eventos, para definir los requisitos. Las nuevas aplicaciones se diseñan como límites bien definidos de servicios y con acoplamiento débil. Los monolitos existentes se descomponen en contextos delimitados y los diseños de sistemas avanzan hacia arquitecturas SOA o de microservicios. Cuando los monolitos se refactorizan, se aplican enfoques establecidos, como contextos burbuja y patrones de descomposición de monolitos.

Los servicios orientados al dominio se ejecutan como uno o más procesos que no comparten el estado. Responden de forma independiente a las fluctuaciones de la demanda y gestionan los escenarios de error en función de los requisitos específicos del dominio.

Patrones comunes de uso no recomendados:

- Se forman equipos en torno a dominios técnicos específicos, como la interfaz de usuario y la experiencia de usuario, el middleware o la base de datos, en lugar de formarse en torno a dominios empresariales específicos.

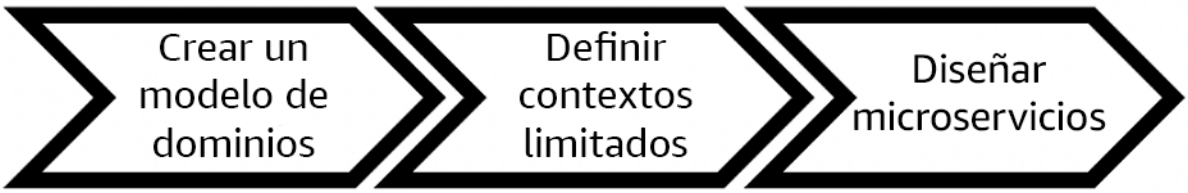
- Las aplicaciones abarcan las responsabilidades del dominio. Los servicios que abarcan contextos delimitados pueden ser más difíciles de mantener, exigen más pruebas y requieren la participación de equipos de varios dominios en las actualizaciones del software.
- Las dependencias de dominio, como las bibliotecas de entidades de dominio, se comparten entre los servicios, de modo que los cambios en un dominio de servicio requieren cambios en otros dominios de servicio
- Los contratos de servicio y la lógica empresarial no expresan las entidades en un lenguaje de dominio común y coherente, lo que genera capas de traducción que complican los sistemas e incrementan los esfuerzos de depuración.

Beneficios de establecer esta práctica recomendada: las aplicaciones se diseñan como servicios independientes delimitados por dominios empresariales y utilizan un lenguaje empresarial común. Los servicios se pueden probar e implementar de forma independiente. Los servicios cumplen los requisitos de resiliencia específicos del dominio implementado.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El enfoque de decisiones impulsadas por dominio (DDD) es el enfoque fundamental para diseñar y crear software en torno a los dominios empresariales. Resulta útil trabajar con un marco existente a la hora de crear servicios centrados en dominios empresariales. Si trabaja con aplicaciones monolíticas existentes, puede utilizar patrones de descomposición que ofrecen técnicas establecidas para modernizar las aplicaciones y convertirlas en servicios.




Diseño basado en el dominio

Pasos para la implementación

- Los equipos pueden organizar talleres de tormentas de eventos para identificar rápidamente eventos, comandos, agregados y dominios en un formato de notas adhesivas más ligero.

- Cuando las entidades y funciones del dominio se formen en un contexto de dominio, puede dividir el dominio en servicios mediante un contexto delimitado, en el que se agrupan las entidades que comparten características y atributos similares. Si el modelo está dividido en contextos, tendrá una plantilla para limitar los microservicios.
  - Por ejemplo, las entidades del sitio web de Amazon.com podrían incluir el empaquetado, la entrega, la programación, el precio, el descuento y la divisa.
  - El empaquetado, la entrega y la programación se agrupan en el contexto del envío, mientras que el precio, el descuento y la divisa se agrupan en el contexto de los precios.
- La descomposición de los monolitos en microservicios describe los patrones para refactorizar los microservicios. El uso de patrones de descomposición por capacidad empresarial, subdominio o transacción se ajusta bien a los enfoques basados en dominios.
- Técnicas tácticas como el contexto burbuja, que permiten introducir DDD en aplicaciones existentes o heredadas sin necesidad de reescrituras iniciales ni confirmaciones completas de las DDD. En un enfoque con contexto burbuja, se establece un pequeño contexto delimitado mediante una capa de asignación y coordinación de servicios (capa anticorrupción), que protege el modelo de dominio recién definido de influencias externas.

Después de que los equipos analicen el dominio y definan las entidades y los contratos de servicio, podrán utilizar los servicios de AWS para implementar su diseño basado en dominio como servicios basados en la nube.

- Para comenzar el desarrollo, defina pruebas en las que se utilicen las reglas empresariales de su dominio. El desarrollo basado en pruebas (TDD) y el desarrollo basado en comportamiento (BDD) ayudan a los equipos a mantener los servicios centrados en resolver problemas empresariales.
- Seleccione los servicios de AWS que mejor se adapten a los requisitos del dominio empresarial y a la arquitectura de microservicios:
  - AWS sin servidor permite a su equipo centrarse en una lógica de dominio específica en lugar de administrar servidores e infraestructuras.
  - Los contenedores en AWS simplifican la administración de su infraestructura para que pueda centrarse en los requisitos de su dominio.
  - Las bases de datos personalizadas le ayudan a adaptar los requisitos de su dominio al tipo de base de datos más adecuado.
- La creación de arquitecturas hexagonales en AWS describe un marco para integrar la lógica empresarial en los servicios que funcionan de manera inversa desde un dominio empresarial para cumplir los requisitos funcionales y, a continuación, asociar los adaptadores de integración.

Los patrones que separan los detalles de la interfaz de la lógica empresarial con los servicios de AWS ayudan a los equipos a centrarse en la funcionalidad del dominio y a mejorar la calidad del software.

Recursos

Prácticas recomendadas relacionadas:

- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL03-BP03 Disposición de contratos de servicio por cada API

Documentos relacionados:

- Microservicios de AWS
- Implementing Microservices on AWS
- How to break a Monolith into Microservices
- Getting Started with DDD when Surrounded by Legacy Systems
- Domain-Driven Design: Tackling Complexity in the Heart of Software
- Building hexagonal architectures on AWS
- Decomposing monoliths into microservices
- Event Storming
- Messages Between Bounded Contexts
- Microservices
- Desarrollo guiado por pruebas
- Desarrollo guiado por comportamiento

Ejemplos relacionados:

- Designing Cloud Native Microservices on AWS (from DDD/EventStormingWorkshop)

Herramientas relacionadas:

- Bases de datos en la nube de Nube de AWS
- Sin servidor en AWS

- Contenedores en AWS

REL03-BP03 Disposición de contratos de servicio por cada API

Los contratos de servicio son acuerdos documentados entre los productores y los consumidores de las API que se encuentran en una definición de API legible por máquina. Una estrategia de control de versiones permite a los clientes seguir usando la API existente y migrar sus aplicaciones a la nueva API cuando estén listas. La implementación del productor puede efectuarse en cualquier momento, siempre y cuando se cumpla el contrato. Los equipos del servicio pueden usar la pila tecnológica que prefieran para cumplir el contrato de la API.

Resultado deseado: las aplicaciones creadas con arquitecturas orientadas a servicios o de microservicios pueden funcionar de forma independiente y, al mismo tiempo, tener integrada una dependencia de la versión en tiempo de ejecución. Los cambios implementados en un consumidor o productor de API no interrumpen la estabilidad del sistema general cuando ambas partes utilizan el mismo contrato de API. Los componentes que se comunican a través de las API de servicio pueden llevar a cabo lanzamientos funcionales independientes, actualizar las dependencias en tiempo de ejecución o efectuar conmutaciones por error a un sitio de recuperación de desastres (DR) con poco o ningún impacto entre sí. Además, los servicios discretos pueden escalarse de forma independiente y absorber la demanda de recursos sin que sea necesario que otros servicios se escalen al unísono.

Patrones comunes de uso no recomendados:

- Crear API de servicio sin esquemas estrictamente asignados. Como consecuencia, las API no se pueden usar para generar enlaces de API y las cargas útiles no se pueden validar mediante programación.
- No adoptar una estrategia de control de versiones, lo que obliga a los usuarios de la API a actualizarla y lanzarla; de lo contrario, fallará cuando los contratos de servicio evolucionen.
- Mensajes de error que filtran detalles de la implementación del servicio subyacente en lugar de describir los errores de integración en el contexto y el lenguaje del dominio.
- No utilizar contratos de API para desarrollar casos de prueba ni simulaciones de implementaciones de API para probar de forma independiente los componentes del servicio.

Beneficios de establecer esta práctica recomendada: los sistemas distribuidos que constan de componentes que se comunican a través de contratos de servicio de API pueden mejorar la fiabilidad. Los desarrolladores pueden detectar posibles problemas al principio del proceso de desarrollo mediante la comprobación de tipos durante la compilación para comprobar que las

solicitudes y las respuestas cumplan el contrato de la API y que los campos obligatorios estén presentes. Los contratos de la API proporcionan una interfaz clara y autodocumentada para las API y mejoran la interoperabilidad entre diferentes sistemas y lenguajes de programación.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Una vez que hayan identificado los dominios empresariales y determinado la segmentación de la carga de trabajo, podrá desarrollar las API de sus servicios. Primero, defina contratos de servicio legibles por máquina para las API y, a continuación, implemente una estrategia de control de versiones de API. Cuando lo tenga todo preparado para integrar servicios a través de protocolos comunes, como REST, GraphQL o eventos asíncronos, podrá incorporar servicios de AWS a su arquitectura para integrar sus componentes con contratos de API estrictamente asignados.

Servicios de AWS para contratos de API de servicios

Incorpore servicios de AWS como Amazon API Gateway, AWS AppSync y Amazon EventBridge a su arquitectura para utilizar los contratos de servicios de API en su aplicación. Amazon API Gateway le ayuda a integrarse directamente con servicios de AWS nativos y otros servicios web. API Gateway admite el control de versiones y la especificación de OpenAPI. AWS AppSync es un punto de conexión de GraphQL administrado que se configura mediante la definición de un esquema de GraphQL para definir una interfaz de servicio para consultas, mutaciones y suscripciones. Amazon EventBridge utiliza esquemas de eventos para definir eventos y generar enlaces de código para sus eventos.

Pasos para la implementación

- Primero, defina un contrato para su API. En un contrato, se expresan las capacidades de una API y se definen objetos y campos de datos estrictamente asignados para la entrada y la salida de la API.
- Cuando configure las API en API Gateway, puede importar y exportar las especificaciones de OpenAPI para sus puntos de conexión.
  - La importación de una definición de OpenAPI simplifica la creación de su API y se puede integrar con la infraestructura de AWS, como herramientas de código (por ejemplo, AWS Serverless Application Model y Kit de desarrollo de la nube de AWS (AWS CDK)).
  - La exportación de una definición de API simplifica la integración con las herramientas de prueba de API y proporciona a los consumidores de servicios una especificación de la integración.

- Puede definir y administrar las API de GraphQL con AWS AppSync mediante la definición de un archivo de esquema de GraphQL para generar su interfaz de contrato y simplificar la interacción con modelos de REST complejos, múltiples tablas de bases de datos o servicios heredados.
- Los proyectos de AWS Amplify que están integrados con AWS AppSync generan archivos de consulta de JavaScript estrictamente asignados para usarlos en su aplicación, así como una biblioteca de clientes de AWS AppSync GraphQL para tablas de Amazon DynamoDB.
- Cuando se consumen eventos de servicio de Amazon EventBridge, los eventos se ajustan a esquemas que ya existen en el registro de esquemas o que se definen con la especificación de OpenAPI. Si tiene un esquema definido en el registro, también puede generar enlaces de clientes desde el contrato de esquema para integrar el código con los eventos.
- Amplíe la API o lleve a cabo un control de versiones. Ampliar una API es la opción más sencilla cuando se agregan campos que se pueden configurar con campos opcionales o valores predeterminados para los campos obligatorios.
  - Los contratos basados en JSON para protocolos como REST y GraphQL pueden ser una buena opción para la ampliación del contrato.
  - Los contratos basados en XML para protocolos como SOAP deben probarse con los consumidores de servicios para determinar la viabilidad de la ampliación del contrato.
- Al llevar a cabo el control de versiones de una API, considere la posibilidad de implementar un control de versiones por proxy en el que se utilice una fachada para admitir las versiones, de modo que la lógica se pueda mantener en una única base de código.
  - Con API Gateway, puede usar asignaciones de solicitud y respuesta para simplificar la absorción de los cambios en los contratos mediante el establecimiento de una fachada que proporcione valores predeterminados para los campos nuevos o para quitar los campos eliminados de una solicitud o respuesta. Con este enfoque, el servicio subyacente puede mantener una única base de código.

Recursos

Prácticas recomendadas relacionadas:

- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL03-BP02 Desarrollo de servicios centrados en funcionalidades y dominios empresariales específicos
- REL04-BP02 Implementación de dependencias con acoplamiento débil
- REL05-BP03 Control y limitación de las llamadas de reintento

- REL05-BP05 Definición de los tiempos de espera del cliente

Documentos relacionados:

- ¿Qué es una interfaz de programación de aplicaciones (API)?
- Implementing Microservices on AWS
- Microservice Trade-Offs
- Microservices - a definition of this new architectural term
- Microservicios en AWS
- Working with API Gateway extensions to OpenAPI
- OpenAPI-Specification
- GraphQL: Schemas and Types
- Amazon EventBridge code bindings

Ejemplos relacionados:

- Amazon API Gateway: Configuring a REST API Using OpenAPI
- Amazon API Gateway to Amazon DynamoDB CRUD application using OpenAPI
- Modern application integration patterns in a serverless age: API Gateway Service Integration
- Implementing header-based API Gateway versioning with Amazon CloudFront
- AWS AppSync: Building a client application

Videos relacionados:

- Using OpenAPI in AWS SAM to manage API Gateway

Herramientas relacionadas:

- Amazon API Gateway
- AWS AppSync
- Amazon EventBridge

## REL 4. ¿Cómo diseña las interacciones en un sistema distribuido para evitar los errores?

Los sistemas distribuidos se basan en las redes de comunicaciones para interconectar componentes, como servidores o servicios. Su carga de trabajo debe funcionar de manera fiable a pesar de la pérdida de datos o la latencia en estas redes. Los componentes del sistema distribuido deben funcionar de manera que no afecten negativamente a otros componentes o a la carga de trabajo. Estas prácticas recomendadas previenen los fallos y mejoran el tiempo medio entre errores (MTBD).

Prácticas recomendadas

- REL04-BP01 Identificación del tipo de sistemas distribuidos de los que depende
- REL04-BP02 Implementación de dependencias con acoplamiento débil
- REL04-BP03 Trabajo constante
- REL04-BP04 Cómo hacer idempotentes las operaciones de mutación

### REL04-BP01 Identificación del tipo de sistemas distribuidos de los que depende

Los sistemas distribuidos pueden ser síncronos, asíncronos o por lotes. Los sistemas síncronos deben procesar las solicitudes lo más rápido posible y comunicarse entre sí mediante llamadas síncronas de solicitud y respuesta mediante protocolos HTTP/S, REST o de llamada a procedimiento remoto (RPC). Los sistemas asíncronos se comunican entre sí mediante el intercambio de datos de forma asíncrona a través de un servicio intermediario sin acoplar sistemas individuales. Los sistemas por lotes reciben un gran volumen de datos de entrada, ejecutan procesos de datos automatizados sin intervención humana y generan datos de salida.

Resultado deseado: diseñe una carga de trabajo que interactúe eficazmente con las dependencias síncronas, asíncronas y por lotes.

Patrones comunes de uso no recomendados:

- La carga de trabajo espera indefinidamente una respuesta de sus dependencias, lo que podría provocar que se agote el tiempo de espera de los clientes de la carga de trabajo sin saber si su solicitud se ha recibido.
- La carga de trabajo utiliza una cadena de sistemas dependientes que se llaman entre sí de forma síncrona. Para ello, cada sistema debe estar disponible y procesar correctamente una solicitud para que toda la cadena pueda tener éxito, lo que se traduce en un comportamiento y una disponibilidad general potencialmente frágiles.

- La carga de trabajo se comunica con sus dependencias de forma asíncrona y se basa en la entrega garantizada de mensajes exactamente una vez, aunque aún es posible que se reciban mensajes duplicados.
- La carga de trabajo no utiliza herramientas adecuadas de programación por lotes y permite la ejecución simultánea del mismo trabajo por lotes.

Beneficios de establecer esta práctica recomendada: es habitual que una carga de trabajo determinada implemente uno o más estilos de comunicación entre los sistemas síncronos, asíncronos o por lotes. Esta práctica recomendada le ayuda a identificar las diferentes ventajas y desventajas asociadas a cada estilo de comunicación para que su carga de trabajo pueda tolerar las interrupciones en cualquiera de sus dependencias.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las siguientes secciones contienen una guía de implementación general y específica para cada tipo de dependencia.

General guidance

- Asegúrese de que los objetivos de nivel de servicio (SLO) de rendimiento y fiabilidad que ofrecen sus dependencias cumplan los requisitos de rendimiento y fiabilidad de su carga de trabajo.
- Utilice los servicios de observabilidad de AWS para supervisar los tiempos de respuesta y las tasas de error y asegurarse de que su dependencia presta el servicio a los niveles que necesita su carga de trabajo.
- Identifique los posibles desafíos a los que puede enfrentarse su carga de trabajo al comunicarse con sus dependencias. Los sistemas distribuidos se enfrentan a una amplia gama de desafíos que pueden aumentar la complejidad de la arquitectura, la carga operativa y el costo. Entre los desafíos comunes, se incluyen la latencia, las interrupciones de la red, la pérdida de datos, el escalado y el retardo en la replicación de datos.
- Implemente un sistema sólido de gestión y registro de errores para ayudarle a solucionar problemas cuando su dependencia experimente problemas.

Dependencia síncrona

En las comunicaciones síncronas, la carga de trabajo envía una solicitud a su dependencia y bloquea la operación en espera de una respuesta. Cuando la dependencia recibe la solicitud, intenta

gestionarla lo antes posible y envía una respuesta a su carga de trabajo. Un problema importante de la comunicación síncrona es que provoca un acoplamiento temporal, por lo que la carga de trabajo y sus dependencias deben estar disponibles al mismo tiempo. Cuando la carga de trabajo necesite comunicarse de forma síncrona con sus dependencias, tenga en cuenta lo siguiente:

- La carga de trabajo no debe depender de varias dependencias síncronas para llevar a cabo una sola función. Esta cadena de dependencias aumenta la fragilidad general, porque todas las dependencias de la ruta deben estar disponibles para que la solicitud se complete correctamente.
- Cuando una dependencia no esté en buen estado o no esté disponible, determine sus estrategias de gestión de errores y reintentos. Evite utilizar un comportamiento bimodal. El comportamiento bimodal se produce cuando la carga de trabajo presenta un comportamiento diferente en los modos normal y de error. Para obtener más información sobre el comportamiento bimodal, consulte REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal.
- Tenga en cuenta que responder rápido a los errores es mejor que hacer esperar a la carga de trabajo. Por ejemplo, la Guía para desarrolladores de AWS Lambda describe cómo gestionar los reintentos y los errores al invocar funciones de Lambda.
- Establezca tiempos de espera cuando la carga de trabajo llame a su dependencia. Esta técnica evita esperar demasiado o indefinidamente una respuesta. Para un análisis útil sobre este tema, consulte Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications.
- Minimice la cantidad de llamadas que se hacen desde la carga de trabajo a su dependencia para atender una sola solicitud. Si hay una conversación demasiado intensa entre ellos, aumenta el acoplamiento y la latencia.

### Dependencia asíncrona

Para desvincular temporalmente la carga de trabajo de su dependencia, deben comunicarse de forma asíncrona. Con un enfoque asíncrono, la carga de trabajo puede continuar con cualquier otro procesamiento sin tener que esperar a que su dependencia o cadena de dependencias envíe una respuesta.

Cuando la carga de trabajo necesite comunicarse de forma asíncrona con su dependencia, tenga en cuenta lo siguiente:

- Determine si va a utilizar la mensajería o la transmisión de eventos en función de su caso de uso y sus requisitos. La mensajería permite que la carga de trabajo se comunique con su dependencia mediante el envío y la recepción de mensajes a través de un agente de mensajes. La transmisión

de eventos permite que su carga de trabajo y su dependencia utilicen un servicio de transmisión para publicar y suscribirse a eventos. Estas transmisiones se distribuyen como flujos continuos de datos, que deben procesarse lo antes posible.

- La mensajería y la transmisión de eventos gestionan los mensajes de manera diferente, por lo que debe decidir si compensan en función de lo siguiente:
  - Prioridad de mensajes: los agentes de mensajes pueden procesar los mensajes de alta prioridad antes que los de prioridad normal. En la transmisión de eventos, todos los mensajes tienen la misma prioridad.
  - Consumo de mensajes: los agentes de mensajes se aseguran de que los consumidores reciban el mensaje. Los consumidores de la transmisión de eventos deben llevar un registro del último mensaje que leyeron.
  - Orden de los mensajes: con la mensajería, no se garantiza la recepción de los mensajes en el orden exacto en que se envíen, a menos que se utilice el enfoque de “el primero en entrar es el primero en salir” (FIFO). La transmisión de eventos siempre mantiene el orden en que se produjeron los datos.
  - Eliminación de mensajes: en el caso de la mensajería, el consumidor debe eliminar el mensaje después de procesarlo. El servicio de transmisión de eventos agrega el mensaje a una transmisión y permanece allí hasta que venza el periodo de retención. Esta política de eliminación hace que la transmisión de eventos sea adecuada para volver a reproducir mensajes.
- Defina la forma en que la carga de trabajo sabe cuándo su dependencia ha terminado el trabajo. Por ejemplo, cuando la carga de trabajo invoca una función de Lambda de forma asíncrona, Lambda pone la solicitud en una cola y devuelve una respuesta de operación correcta sin información adicional. Cuando finalice el procesamiento, la función de Lambda puede enviar el resultado a un destino, que se puede configurar con base en el éxito o el error.
- Aumente la carga de trabajo para gestionar los mensajes duplicados mediante la idempotencia. La idempotencia significa que los resultados de la carga de trabajo no cambian aunque esta se genere más de una vez para el mismo mensaje. Es importante señalar que los servicios de mensajería o transmisión volverán a entregar un mensaje si se produce un error en la red o si no se ha recibido un acuse de recibo.
- Si la carga de trabajo no recibe una respuesta de su dependencia, debe volver a enviar la solicitud. Considere la posibilidad de limitar el número de reintentos para conservar los recursos de CPU, memoria y red de la carga de trabajo para gestionar otras solicitudes. La documentación de AWS Lambda muestra cómo gestionar los errores en la invocación asíncrona.

- Utilice las herramientas de observabilidad, depuración y rastreo adecuadas para administrar y utilizar la comunicación asíncrona de la carga de trabajo con su dependencia. Puede utilizar Amazon CloudWatch para supervisar los servicios de mensajería y transmisión de eventos. También puede instrumentar su carga de trabajo con AWS X-Ray para obtener información rápidamente que le permita solucionar problemas.

### Dependencia por lotes

Los sistemas por lotes toman los datos de entrada, inician una serie de trabajos para procesarlos y producen algunos datos de salida, sin intervención manual. Según el tamaño de los datos, los trabajos pueden durar desde unos minutos hasta, en algunos casos, varios días. Cuando la carga de trabajo se comunique con su dependencia por lotes, tenga en cuenta lo siguiente:

- Defina el intervalo de tiempo en el que la carga de trabajo debe ejecutar el trabajo por lotes. La carga de trabajo puede configurar un patrón de recurrencia para invocar un sistema por lotes como, por ejemplo, cada hora o al final de cada mes.
- Determine la ubicación de la entrada de datos y la salida de los datos procesados. Elija un servicio de almacenamiento, como Amazon Simple Storage Service (Amazon S3), Amazon Elastic File System (Amazon EFS) y Amazon FSx para Lustre, que permita que su carga de trabajo lea y escriba archivos a escala.
- Si su carga de trabajo necesita invocar varios trabajos por lotes, puede utilizar AWS Step Functions para simplificar la orquestación de los trabajos por lotes que se ejecutan en AWS o en las instalaciones. Este proyecto de ejemplo demuestra la orquestación de trabajos por lotes mediante Step Functions, AWS Batch y Lambda.
- Supervise los trabajos por lotes para detectar anomalías, como que un trabajo tarde más de lo debido en completarse. Puede utilizar herramientas como Información de contenedores de CloudWatch para supervisar entornos y trabajos de AWS Batch. En este caso, su carga de trabajo impediría el inicio del siguiente trabajo e informaría al personal correspondiente de la excepción.

### Recursos

Documentos relacionados:

- Operaciones de Nube de AWS: supervisión y observabilidad
- Amazon Builders' Library: los desafíos de los sistemas distribuidos
- REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal

- Guía para desarrolladores de AWS Lambda: control de errores y reintentos automáticos en AWS Lambda
- Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications
- Mensajería de AWS
- ¿Qué son los datos de streaming?
- Guía para desarrolladores de AWS Lambda: invocación asíncrona
- Preguntas frecuentes sobre Amazon Simple Queue Service: colas FIFO
- Amazon Kinesis Data Streams Developer Guide: Handling Duplicate Records
- Amazon Simple Queue Service Developer Guide: Available CloudWatch metrics for Amazon SQS
- Amazon Kinesis Data Streams Developer Guide: Monitoring the Amazon Kinesis Data Streams Service with Amazon CloudWatch
- AWS X-Ray Developer Guide: AWS X-Ray concepts
- Ejemplos de AWS en GitHub: aplicación de AWS Step Functions Complex Orchestrator
- AWS Batch User Guide: AWS Batch CloudWatch Container Insights

Videos relacionados:

- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS (COP310)

Herramientas relacionadas:

- Amazon CloudWatch
- Registros de Amazon CloudWatch
- AWS X-Ray
- Amazon Simple Storage Service (Amazon S3)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx para Lustre
- AWS Step Functions
- AWS Batch

REL04-BP02 Implementación de dependencias con acoplamiento débil

Las dependencias, como los sistemas de colas, los sistemas de transmisión, los flujos de trabajo y los equilibradores de carga, tienen un acoplamiento débil. El acoplamiento débil ayuda a aislar el comportamiento de un componente de otros componentes que dependen de él, lo que aumenta la resiliencia y la agilidad.

El desacoplamiento de las dependencias, como los sistemas de colas, los sistemas de transmisión y los flujos de trabajo, ayuda a minimizar el impacto de los cambios o los errores en un sistema. Esta separación aísla el comportamiento de un componente para que no afecte a otros que dependan de él, lo que mejora la resiliencia y la agilidad.

En sistemas de acoplamiento ajustado, los cambios en un componente pueden requerir cambios en otros componentes que dependan de él, lo que reduce el rendimiento de todos los componentes. El acoplamiento débil elimina esta dependencia, de forma que los componentes dependientes solo necesitan conocer la interfaz publicada y con control de versiones. La implementación de un acoplamiento débil entre las dependencias aísla un error en una de ellas para que no afecte a otra.

El acoplamiento débil permite modificar el código o agregar características a un componente y, al mismo tiempo, minimizar el riesgo para otros componentes que dependan de él. También permite una resiliencia granular de los componentes, lo que permite escalar horizontalmente o incluso cambiar la implementación subyacente de la dependencia.

Para mejorar aún más la resiliencia mediante el acoplamiento débil, haga que las interacciones entre componentes sean asincrónicas siempre que sea posible. Este modelo es adecuado para cualquier interacción que no necesite una respuesta inmediata y en la que baste con el reconocimiento de que una solicitud se ha registrado. Consta de un componente que genera eventos y de otro que los consume. Ambos componentes no se integran mediante una interacción directa de punto a punto, sino que normalmente emplean una capa de almacenamiento duradera intermedia, como una cola de Amazon SQS o una plataforma de restringa de datos como Amazon Kinesis o AWS Step Functions.

Figura 4: Las dependencias, como los sistemas de colas y los balanceadores de carga, tienen un acoplamiento débil

Las colas de Amazon SQS y AWS Step Functions son solo dos formas de agregar una capa intermedia para el acoplamiento débil. Las arquitecturas basadas en eventos también se pueden crear en la Nube de AWS con Amazon EventBridge, que puede separar a los clientes (productores

de eventos) de los servicios de los que dependen (consumidores de eventos). Amazon Simple Notification Service (Amazon SNS) es una solución eficaz para cuando sean necesarios mensajes de alto rendimiento, de tipo push y de varios a varios. Con el uso de temas de Amazon SNS, los sistemas de su publicador pueden repartir mensajes por una gran cantidad de puntos de conexión de suscriptores para procesarlos en paralelo.

Aunque las colas ofrecen varias ventajas, en la mayoría de sistemas en tiempo real estricto, las solicitudes que superan un umbral temporal (que suele ser de segundos) se consideran obsoletas (el cliente ha desistido y ya no espera una respuesta), por lo que no se procesan. De esta manera, se pueden procesar las solicitudes más recientes (y probablemente aún válidas) en su lugar.

Resultado deseado: la implementación de dependencias con un acoplamiento débil permite minimizar la superficie de posibles errores a nivel del componente, lo que ayuda a diagnosticar y resolver problemas. También simplifica los ciclos de desarrollo, lo que permite a los equipos implementar cambios a nivel modular sin que eso afecte al rendimiento de otros componentes que dependan de él. Este enfoque ofrece la capacidad de escalar horizontalmente a nivel de componente en función de los recursos que sean necesarios, así como de utilizar un componente que contribuye a ahorrar costos.

Patrones comunes de uso no recomendados:

- Implementar una carga de trabajo monolítica.
- Invocar directamente las API entre capas de la carga de trabajo sin la capacidad de conmutar por error ni procesar de manera asíncrona la solicitud.
- Utilizar un acoplamiento ajustado con datos compartidos. Los sistemas de acoplamiento débil no deben compartir datos a través de bases de datos compartidas u otras formas de almacenamiento de datos de acoplamiento ajustado, que pueden reintroducir el acoplamiento ajustado y dificultar la escalabilidad.
- Ignorar la contrapresión. La carga de trabajo debe tener la capacidad de ralentizar o detener los datos entrantes cuando un componente no pueda procesarlos al mismo ritmo.

Beneficios de establecer esta práctica recomendada: el acoplamiento débil ayuda a aislar el comportamiento de un componente de otros que dependen de él, lo que aumenta la resiliencia y la agilidad. Un error en un componente está aislado de los demás componentes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Implemente dependencias con acoplamiento débil. Existen varias soluciones que permiten crear aplicaciones con un acoplamiento débil. Entre ellas, se incluyen servicios para implementar colas totalmente administradas, flujos de trabajo automatizados, reacción a eventos y API, entre otras, que pueden ayudar a aislar el comportamiento de los componentes de otros componentes y, por lo tanto, aumentar la resiliencia y la agilidad.

- Creación de arquitecturas basadas en eventos: Amazon EventBridge le ayuda a crear arquitecturas impulsadas por eventos distribuidas y acopladas de forma débil.
- Implementación de colas en sistemas distribuidos: puede utilizar Amazon Simple Queue Service (Amazon SQS) para integrar y desacoplar sistemas distribuidos.
- Colocación en contenedores de los componentes como microservicios: los microservicios permiten a los equipos crear aplicaciones compuestas por pequeños componentes independientes que se comunican a través de API bien definidas. Amazon Elastic Container Service (Amazon ECS) y Amazon Elastic Kubernetes Service (Amazon EKS) le pueden ayudar a comenzar con el uso de contenedores.
- Administración de los flujos de trabajo con Step Functions: Step Functions le ayuda a coordinar varios servicios de AWS en flujos de trabajo flexibles.
- Uso de las arquitecturas de mensajería de publicación y suscripción (pub/sub): Amazon Simple Notification Service (Amazon SNS) proporciona la entrega de mensajes de los publicadores a los suscriptores (también conocidos como productores y consumidores).

Pasos para la implementación

- Los componentes de una arquitectura basada en eventos se inician mediante eventos. Los eventos son acciones que ocurren en un sistema, como cuando un usuario agrega un artículo a una cesta. Cuando una acción se lleva a cabo correctamente, se genera un evento que activa el siguiente componente del sistema.
  - Building Event-driven Applications with Amazon EventBridge
  - AWS re:Invent 2022 - Designing Event-Driven Integrations using Amazon EventBridge
- Los sistemas de mensajería distribuida tienen tres partes principales que deben implementarse para una arquitectura basada en colas. Incluyen los componentes del sistema distribuido, la cola que se usa para el desacoplamiento (distribuida en servidores de Amazon SQS servers) y los mensajes de la cola. Un sistema típico tiene productores que inician el mensaje en la cola y el

consumidor que recibe el mensaje de la cola. La cola almacena los mensajes en varios servidores de Amazon SQS para garantizar la redundancia.

  - Basic Amazon SQS architecture
  - Send Messages Between Distributed Applications with Amazon Simple Queue Service
- Los microservicios, cuando se utilizan bien, facilitan el mantenimiento y aumentan la escalabilidad, ya que los componentes de acoplamiento débil los administran equipos independientes. También permiten aislar los comportamientos en un solo componente en caso de que se hagan cambios.
  - Implementing Microservices on AWS
  - Let's Architect! Architecting microservices with containers
- Con AWS Step Functions, puede crear aplicaciones distribuidas, automatizar procesos y orquestar microservicios, entre otras cosas. La orquestación de varios componentes en un flujo de trabajo automatizado le permite desacoplar las dependencias de su aplicación.
  - Cree un flujo de trabajo sin servidor con y AWS Step FunctionsAWS Lambda
  - Introducción a AWS Step Functions

Recursos

Documentos relacionados:

- Amazon EC2: Ensuring Idempotency
- Amazon Builders' Library: Desafíos de los sistemas distribuidos
- Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- Qué es Amazon EventBridge?
- What Is Amazon Simple Queue Service?
- Break up with your monolith
- Orchestrate Queue-based Microservices with AWS Step Functions and Amazon SQS
- Basic Amazon SQS architecture
- Queue-Based Architecture

Videos relacionados:

- AWS New York Summit 2019: Intro to Event-driven Architectures and Amazon EventBridge (MAD205)

- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes loose coupling, constant work, static stability)
- AWS re:Invent 2019: Moving to event-driven architectures (SVS308)
- AWS re:Invent 2019: Scalable serverless event-driven applications using Amazon SQS and Lambda
- AWS re:Invent 2022 - Designing event-driven integrations using Amazon EventBridge
- AWS re:Invent 2017: Elastic Load Balancing Deep Dive and Best Practices

REL04-BP03 Trabajo constante

Los sistemas pueden producir error cuando hay cambios rápidos grandes en la carga. Por ejemplo, si la carga de trabajo está llevando a cabo una comprobación de estado que supervisa el estado de miles de servidores, debería enviar siempre una carga del mismo tamaño (una instantánea completa del estado actual). Si no hay errores en ningún servidor, o hay errores en todos ellos, el sistema de comprobación de estado estará haciendo un trabajo constante sin rápidos cambios de gran tamaño.

Por ejemplo, si el sistema de comprobación de estado supervisa 100 000 servidores, la carga contenida en él es nominal con un porcentaje de errores del servidor normalmente bajo. Sin embargo, si un evento importante deja a la mitad de esos servidores en mal estado, el sistema de comprobación de estado se sobrecargaría al intentar actualizar los sistemas de notificación y comunicar el estado a sus clientes. Por ello, el sistema de comprobación de estado debería enviar cada vez la instantánea completa del estado actual. 100 000 estados de servidor, cada uno representado por un bit, solo constituiría una carga de 12,5 KB. Si no hay errores en ningún servidor, o hay errores en todos ellos, el sistema de comprobación de estado estará haciendo un trabajo constante y los cambios rápidos de gran tamaño no pondrán en peligro la estabilidad del sistema. En realidad, así es como Amazon Route 53 gestiona las comprobaciones de estado de los puntos de conexión (como las direcciones IP) para determinar cómo se enruta a los usuarios finales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

- Trabaje constantemente para que los sistemas no tengan errores cuando haya cambios grandes y rápidos en la carga.
- Implemente dependencias con acoplamiento débil. Las dependencias, como los sistemas de colas, los sistemas de transmisión, los flujos de trabajo y los equilibradores de carga, tienen un

acoplamiento débil. El acoplamiento débil ayuda a aislar el comportamiento de un componente de otros componentes que dependen de él, lo que aumenta la resiliencia y la agilidad.

- Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes constant work)
  - En el ejemplo de una sistema de comprobación de estado que supervisa 100 000 servidores, diseñe las cargas de trabajo de forma que los tamaños de la carga útil sean iguales independientemente del número de éxitos o fracasos.

Recursos

Documentos relacionados:

- Amazon EC2: Ensuring Idempotency
- Amazon Builders' Library: Desafíos de los sistemas distribuidos
- Amazon Builders' Library: Reliability, constant work, and a good cup of coffee

Videos relacionados:

- AWS New York Summit 2019: Intro to Event-driven Architectures and Amazon EventBridge (MAD205)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes constant work)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes loose coupling, constant work, static stability)
- AWS re:Invent 2019: Moving to event-driven architectures (SVS308)

REL04-BP04 Cómo hacer idempotentes las operaciones de mutación

Un servicio idempotente promete que cada solicitud se procesará una y solo una vez, de tal forma que hacer varias solicitudes idénticas tiene el mismo efecto que hacer una sola solicitud. De este modo, un cliente lo tiene más fácil para implementar los reintentos sin la preocupación de que una solicitud se procese varias veces por error. Para ello, los clientes pueden emitir solicitudes de API con un token de idempotencia, que se utiliza siempre que se repite la solicitud. Una API de servicio

idempotente usa el token para devolver una respuesta idéntica a la que se devolvió por primera vez cuando se completó la solicitud, incluso aunque haya cambiado el estado subyacente del sistema.

En un sistema distribuido, es relativamente fácil llevar a cabo una acción una vez como máximo (el cliente solo hace una solicitud) o al menos una vez (sigue haciendo la solicitud hasta que el cliente obtiene una confirmación del éxito). Es más difícil garantizar que una acción se realice exactamente una vez, de modo que hacer varias solicitudes idénticas tenga el mismo efecto que llevar a cabo una sola solicitud. Con el uso de tokens de idempotencia en las API, los servicios pueden recibir una solicitud de migración una o más veces sin necesidad de crear registros duplicados ni efectos secundarios.

Resultado deseado: un enfoque coherente, bien documentado y ampliamente adoptado para garantizar la idempotencia de todos los componentes y servicios.

Patrones comunes de uso no recomendados:

- Aplica la idempotencia de forma indiscriminada, incluso cuando no es necesaria.
- Introduce una lógica demasiado compleja para implementar la idempotencia.
- Usa las marcas de tiempo como claves para la idempotencia. Esto puede provocar imprecisiones debido al sesgo de reloj o a que varios clientes utilicen las mismas marcas de tiempo para aplicar los cambios.
- Almacena cargas útiles completas para la idempotencia. Con este enfoque, se guardan las cargas útiles de datos completas de cada solicitud y se sobrescriben en cada nueva solicitud. Esto puede reducir el rendimiento y afectar a la escalabilidad.
- Genera claves de forma incoherente en todos los servicios. Sin claves coherentes, es posible que los servicios no reconozcan las solicitudes duplicadas, lo que se traduce en resultados imprevistos.

Beneficios de establecer esta práctica recomendada:

- Mayor escalabilidad: el sistema puede gestionar los reintentos y las solicitudes duplicadas sin tener que realizar una lógica adicional o una compleja gestión del estado.
- Fiabilidad mejorada: la idempotencia ayuda a los servicios a gestionar varias solicitudes idénticas de manera coherente, lo que reduce el riesgo de efectos secundarios no deseados o registros duplicados. Esto es especialmente importante en los sistemas distribuidos, donde se producen fallos de red y reintentos con frecuencia.

- Mejora de la coherencia de datos: dado que la misma solicitud produce la misma respuesta, la idempotencia ayuda a mantener la coherencia de datos en todos los sistemas distribuidos. Esto es esencial para mantener la integridad de las transacciones y las operaciones.
- Gestión de errores: los tokens de idempotencia simplifican la gestión de errores. Si un cliente no recibe una respuesta debido a un problema, puede reenviar la solicitud de forma segura con el mismo token de idempotencia.
- Transparencia operativa: la idempotencia permite una mejor supervisión y registro. Los servicios pueden registrar las solicitudes con sus tokens de idempotencia, lo que facilita el rastreo y la depuración de los problemas.
- Contrato de API simplificado: puede simplificar el contrato entre los sistemas del cliente y del servidor y reducir la preocupación por posibles errores en el procesamiento de los datos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

En un sistema distribuido, es relativamente fácil llevar a cabo una acción una vez como máximo (el cliente solo hace una solicitud) o al menos una vez (sigue haciendo la solicitud hasta que el cliente obtiene una confirmación del funcionamiento correcto). Sin embargo, es difícil implementar un comportamiento que se dé una sola vez. Para lograrlo, sus clientes deben generar y proporcionar un token de idempotencia para cada solicitud.

Mediante el uso de fichas de idempotencia, un servicio puede distinguir entre solicitudes nuevas y solicitudes repetidas. Cuando un servicio recibe una solicitud con un token de idempotencia, comprueba si el token ya se ha utilizado. Si se ha utilizado el token, el servicio recupera y devuelve la respuesta almacenada. Si el token es nuevo, el servicio procesa la solicitud, almacena la respuesta junto con el token y, a continuación, devuelve la respuesta. Este mecanismo hace que todas las respuestas sean idempotentes, lo que mejora la fiabilidad y la coherencia del sistema distribuido.

La idempotencia también es un comportamiento importante de las arquitecturas basadas en eventos. Estas arquitecturas suelen estar respaldadas por una cola de mensajes como Amazon SQS, Amazon MQ, Amazon Kinesis Streams o Amazon Managed Streaming para Apache Kafka (MSK). En algunas circunstancias, un mensaje que se ha publicado solo una vez puede entregarse accidentalmente más de una vez. Cuando un publicador genera e incluye símbolos de idempotencia en los mensajes, solicita que al procesar cualquier mensaje duplicado recibido no se repita ninguna acción para el mismo mensaje. Los consumidores deben llevar un registro de cada token recibido e ignorar los mensajes que contengan tokens duplicados.

Los servicios y los consumidores también deberían transferir el token de idempotencia recibido a cualquier servicio posterior al que este llame. Todos los servicios posteriores de la cadena de procesamiento son igualmente responsables de garantizar que la idempotencia se implemente para evitar el efecto secundario de procesar un mensaje más de una vez.

Pasos para la implementación

1. Identifique las operaciones idempotentes

   Determine qué operaciones requieren idempotencia. Por lo general, incluyen los métodos HTTP POST, PUT y DELETE y las operaciones de inserción, actualización o eliminación de bases de datos. Las operaciones que no cambian de estado, como las consultas de solo lectura, no suelen requerir idempotencia, a menos que tengan efectos secundarios.

2. Use identificadores únicos

   Incluye un token único en cada solicitud de operación idempotente que envíe el remitente, ya sea directamente en la solicitud o como parte de sus metadatos (por ejemplo, un encabezado HTTP). Esto permite al receptor reconocer y gestionar las solicitudes u operaciones duplicadas. Los identificadores que se utilizan habitualmente para los tokens incluyen los identificadores únicos universales (UUID) y los identificadores únicos clasificables por K (KSUID).

3. Rastree y gestione el estado

   Mantenga el estado de cada operación o solicitud de su carga de trabajo. Esto se puede lograr almacenando el token de idempotencia y el estado correspondiente (como pendiente, completado o fallido) en una base de datos, caché u otro almacén persistente. Esta información de estado permite a la carga de trabajo identificar y gestionar las solicitudes u operaciones duplicadas.

   Mantenga la coherencia y la atomicidad mediante el uso de los mecanismos de control de simultaneidad adecuados, si es necesario, como bloqueos, transacciones o controles de simultaneidad optimistas. Esto incluye el proceso de registrar el token idempotente y ejecutar todas las operaciones de mutación asociadas con la atención de la solicitud. Esto ayuda a prevenir las condiciones de carrera y verifica que las operaciones idempotentes se ejecuten correctamente.

   Elimine periódicamente los tokens de idempotencia antiguos del almacén de datos para gestionar el almacenamiento y el rendimiento. Si su sistema de almacenamiento lo admite, plantéese utilizar marcas de tiempo de caducidad para los datos (conocidas como tiempo de vida o valores TTL). La probabilidad de que se reutilicen los tokens de idempotencia disminuye con el tiempo.

Las opciones de almacenamiento de AWS más comunes que se suelen utilizar para almacenar los tokens de idempotencia y el estado relacionado incluyen:

- Amazon DynamoDB: DynamoDB es un servicio de base de datos NoSQL que proporciona un rendimiento de baja latencia y alta disponibilidad, lo que lo hace ideal para el almacenamiento de datos relacionados con la idempotencia. El modelo de datos de documentos y valores clave de DynamoDB permite almacenar y recuperar de forma eficiente los símbolos de idempotencia y la información de estado asociada. DynamoDB también puede hacer que los tokens de idempotencia caduquen automáticamente si la aplicación establece un valor TTL al insertarlos.
- Amazon ElastiCache: ElastiCache puede almacenar tokens de idempotencia con alto rendimiento, baja latencia y bajo coste. Tanto ElastiCache (Redis) como ElastiCache (Memcached) también pueden hacer que los tokens de idempotencia caduquen automáticamente si la aplicación establece un valor TTL al insertarlos.
- Amazon Relational Database Service (RDS): puede utilizar Amazon RDS para almacenar los tokens de idempotencia y la información de estado relacionada, especialmente si su aplicación ya utiliza una base de datos relacional para otros fines.
- Amazon Simple Storage Service (S3): Amazon S3 es un servicio de almacenamiento de objetos duradero y altamente escalable que se puede utilizar para almacenar tokens de idempotencia y metadatos relacionados. Las capacidades de control de versiones de S3 pueden resultar particularmente útiles para mantener el estado de las operaciones idempotentes. La elección del servicio de almacenamiento suele depender de factores como el volumen de datos relacionados con la idempotencia, las características de rendimiento requeridas, la necesidad de durabilidad y disponibilidad y la forma en que el mecanismo de idempotencia se integra en la arquitectura de la carga de trabajo general.

4. Implemente operaciones idempotentes

   Diseñe sus componentes de API y de carga de trabajo para que sean idempotentes. Incorpore controles de idempotencia en los componentes de su carga de trabajo. Antes de procesar una solicitud o realizar una operación, compruebe si el identificador único ya se ha procesado. Si es así, devuelva el resultado anterior en lugar de volver a ejecutar la operación. Por ejemplo, si un cliente envía una solicitud para crear un usuario, compruebe si ya existe un usuario con el mismo identificador único. Si el usuario existe, debería devolver la información del usuario existente en lugar de crear una nueva. Del mismo modo, si un consumidor de la cola recibe un mensaje con un token de idempotencia duplicado, debe ignorar el mensaje.

Cree conjuntos de pruebas integrales que validen la idempotencia de las solicitudes. Deben cubrir una amplia gama de escenarios, como las solicitudes correctas, las fallidas y las duplicadas.

Si su carga de trabajo aprovecha las funciones de AWS Lambda, puede usar Powertools para AWS Lambda. Powertools para AWS Lambda es un kit de herramientas para desarrolladores para implementar prácticas recomendadas sin servidor y aumentar la velocidad de los desarrolladores cuando trabaja con funciones de AWS Lambda. En concreto, proporciona una utilidad para convertir las funciones de Lambda en operaciones idempotentes que se pueden volver a intentar de forma segura.

5. Comunique la idempotencia con claridad

   Documente su API y los componentes de la carga de trabajo para comunicar claramente la naturaleza idempotente de las operaciones. Esto ayuda a los clientes a entender el comportamiento esperado y cómo interactuar con su carga de trabajo de forma fiable.

6. Monitoree y audite:

   Implemente mecanismos de supervisión y auditoría para detectar cualquier problema relacionado con la idempotencia de las respuestas, como las variaciones inesperadas de las respuestas o el exceso de gestión de solicitudes duplicadas. Esto puede ayudarlo a detectar e investigar cualquier problema o comportamiento inesperado en su carga de trabajo.

Recursos

Prácticas recomendadas relacionadas:

- REL05-BP03 Control y limitación de las llamadas de reintento
- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)
- REL08-BP02 Integración de las pruebas funcionales como parte de la implementación

Documentos relacionados:

- Amazon Builders' Library: Making retries safe with idempotent APIs
- Amazon Builders' Library: Desafíos de los sistemas distribuidos
- Amazon Builders' Library: Reliability, constant work, and a good cup of coffee
- Amazon Elastic Container Service: Ensuring idempotency

- ¿Cómo puedo hacer que mi función de Lambda sea idempotente?
- Ensuring idempotency in Amazon EC2 API requests

Videos relacionados:

- Building Distributed Applications with Event-driven Architecture. Charlas técnicas en línea de AWS
- AWS re:Invent 2023 - Building next-generation applications with event-driven architecture
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2018 - Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes loose coupling, constant work, static stability)
- AWS re:Invent 2019 - Moving to event-driven architectures (SVS308)

Herramientas relacionadas:

- Idempotencia con AWS Lambda Powertools (Java)
- Idempotencia con AWS Lambda Powertools (Python)
- Página de GitHub de AWS Lambda Powertools

## REL 5. ¿Cómo se diseñan las interacciones en un sistema distribuido para mitigar o resistir los errores?

Los sistemas distribuidos dependen de las redes de comunicaciones para interconectar componentes, como servidores o servicios. Su carga de trabajo debe funcionar de manera fiable aunque se pierdan datos o haya latencia en estas redes. Los componentes del sistema distribuido deben funcionar de manera que no afecten negativamente a otros componentes o a la carga de trabajo. Estas prácticas recomendadas permiten que las cargas de trabajo toleren el estrés o los errores, se recuperen más rápidamente de ellos y mitiguen el impacto de dichos errores. El resultado es un tiempo medio de recuperación (MTTR) mejor.

Prácticas recomendadas

- REL05-BP01 Implementación de una degradación estable para transformar las dependencias estrictas en flexibles
- REL05-BP02 Limitación de las solicitudes

- REL05-BP03 Control y limitación de las llamadas de reintento
- REL05-BP04 Respuesta rápida a los errores y limitación de las colas
- REL05-BP05 Definición de los tiempos de espera del cliente
- REL05-BP06 Creación de sistemas sin estado cuando sea posible
- REL05-BP07 Implementación de recursos de emergencia

REL05-BP01 Implementación de una degradación estable para transformar las dependencias estrictas en flexibles

Los componentes de la aplicación deben seguir desempeñando su función principal incluso si las dependencias dejan de estar disponibles. Es posible que proporcionen datos ligeramente obsoletos, datos alternativos o incluso ningún dato. Esto garantiza que los errores localizados solo impidan lo mínimo del funcionamiento general del sistema y, al mismo tiempo, se obtenga el valor empresarial central.

Resultado deseado: cuando las dependencias de un componente no están en buen estado, el propio componente puede seguir funcionando, aunque con capacidad mermada. Los modos de errores de los componentes deben considerarse parte del funcionamiento normal. Los flujos de trabajo deben diseñarse de tal manera que dichos errores no produzcan un fallo total o, al menos, lleven a estados predecibles y recuperables.

Patrones comunes de uso no recomendados:

- No identificar la funcionalidad empresarial principal necesaria. No probar que los componentes funcionen, incluso durante los errores de dependencia.
- No proporcionar datos en caso de error o cuando solo una de las múltiples dependencias no está disponible y aún se pueden devolver resultados parciales.
- Crear un estado incoherente cuando una transacción falla parcialmente.
- No tener una forma alternativa de acceder a un almacén de parámetros central.
- Invalidar o vaciar un estado local como resultado de un fallo de actualización sin tener en cuenta las consecuencias.

Beneficios de establecer esta práctica recomendada: la degradación gradual mejora la disponibilidad del sistema en su conjunto y mantiene la funcionalidad de las funciones más importantes incluso cuando hay errores.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La implementación de una degradación gradual ayuda a minimizar el impacto de los errores de dependencia en la función de los componentes. Lo ideal sería que un componente detectara los errores de dependencia y siguiese funcionando de una forma que afectara lo menos posible a otros componentes o clientes.

Diseñar una arquitectura que permita una degradación gradual implica considerar los posibles modos de errores durante el diseño de las dependencias. Para cada modo de error, disponga de una forma de ofrecer la mayoría o, al menos la funcionalidad más crítica del componente, a las personas que llaman o a los clientes. Estos factores pueden convertirse en requisitos adicionales que se pueden probar y verificar. Lo ideal es que un componente pueda desempeñar su función principal de manera aceptable incluso cuando falla una o varias dependencias.

Se trata tanto de una cuestión empresarial como técnica. Todos los requisitos empresariales son importantes y deben cumplirse si es posible. Sin embargo, es lógico preguntarse qué debe suceder cuando no se puedan cumplir todos. Se puede diseñar un sistema para que esté disponible y sea coherente, pero en circunstancias en las que haya que eliminar un requisito, ¿cuál es más importante? En el caso del procesamiento de pagos, puede ser la coherencia. En una aplicación en tiempo real, puede ser la disponibilidad. En el caso de un sitio web orientado al cliente, la respuesta dependería de las expectativas del cliente.

Lo que esto significa depende de los requisitos del componente y de lo que deba considerarse su función principal. Por ejemplo:

- Un sitio web de comercio electrónico podría mostrar en su página de inicio los datos de varios sistemas diferentes, como las recomendaciones personalizadas, los productos mejor clasificados y el estado de los pedidos de los clientes. Cuando un sistema anterior falla, sigue siendo lógico mostrar todo lo demás en lugar de mostrar una página de error al cliente.
- Un componente que lleva a cabo escrituras por lotes puede seguir procesando un lote si se produce un error en una de las operaciones individuales. Implementar un mecanismo de reintento debería ser sencillo. Para hacerlo, se puede devolver a la persona que llama información sobre qué operaciones se han hecho correctamente, cuáles han fallado y por qué han fallado, o colocar las solicitudes que han fallado en una cola de mensajes fallidos para implementar reintentos asíncronos. También se debe registrar la información sobre las operaciones que han fallado.
- Un sistema que procese las transacciones debe verificar que se ejecuten todas o ninguna de las actualizaciones individuales. En el caso de las transacciones distribuidas, se puede usar el

patrón Saga para revertir operaciones anteriores en caso de que falle una operación posterior de la misma transacción. En este caso, la función principal es mantener la coherencia.

- Los sistemas en los que el tiempo es crítico deberían contar con la capacidad de gestionar de manera oportuna las dependencias que no respondan. En estos casos, se puede utilizar el patrón del disyuntor. Cuando se agota el tiempo de espera de las respuestas de una dependencia, el sistema puede cambiar a un estado cerrado en el que no se hacen llamadas adicionales.
- Una aplicación puede leer parámetros de un almacén de parámetros. Puede resultar útil crear imágenes de contenedores con un conjunto predeterminado de parámetros y utilizarlos en caso de que ese almacén de parámetros no esté disponible.

Tenga en cuenta que las soluciones que se adopten en caso de fallo de un componente deben probarse y ser significativamente más sencillas que la solución principal. En general, se debe evitar el uso de estrategias alternativas.

Pasos para la implementación

Identifique las dependencias externas e internas. Considere qué tipos de errores pueden producirse en ellas. Piense en formas de minimizar el impacto negativo en los sistemas anteriores y posteriores y en los clientes durante esos errores.

A continuación, tenemos una lista de dependencias y la descripción de cómo degradar correctamente cuando fallan:

1. Errores parciales de dependencias: un componente puede hacer varias solicitudes a los sistemas posteriores, ya sean varias solicitudes a un sistema o una sola solicitud destinada a varios sistemas. En función del contexto empresarial, es posible que haya diferentes formas apropiadas de gestionar este problema (para obtener más información, consulte los ejemplos anteriores en la Guía de implementación).
2. Un sistema descendente no puede procesar las solicitudes debido a la alta carga: si las solicitudes a un sistema descendente fallan constantemente, no tiene sentido volver a intentarlo. Esto puede suponer una carga adicional para un sistema ya sobrecargado y dificultar la recuperación. Aquí se puede utilizar el patrón de disyuntor, que supervisa las llamadas que fallaron al enviarlas a un sistema posterior. Si falla un gran número de llamadas, dejará de enviar más solicitudes al sistema posterior y solo permitirá ocasionalmente el paso de las llamadas para comprobar si el sistema posterior vuelve a estar disponible.
3. Un almacén de parámetros no está disponible: para transformar un almacén de parámetros, se puede utilizar el almacenamiento en caché de dependencia flexible o los valores predeterminados

en buen estado que se incluyen en las imágenes de contenedores o máquinas. Tenga en cuenta que estos valores predeterminados deben mantenerse actualizados e incluirse en los conjuntos de pruebas.

4. Un servicio de supervisión u otra dependencia no funcional no está disponible: si un componente no puede enviar registros, métricas o rastros de forma intermitente a un servicio de monitorización central, suele ser mejor seguir ejecutando las funciones empresariales como de costumbre. No registrar ni subir métricas de forma silenciosa durante mucho tiempo no suele ser aceptable. Además, algunos casos de uso pueden requerir entradas de auditoría completas para satisfacer los requisitos de cumplimiento.
5. Es posible que una instancia principal de una base de datos relacional no esté disponible: Amazon Relational Database Service, como casi todas las bases de datos relacionales, solo puede tener una instancia de escritura principal. Esto crea un único punto de error para las cargas de trabajo de escritura y dificulta el escalamiento. Este problema se puede mitigar parcialmente mediante el uso de una configuración Multi-AZ para lograr alta disponibilidad o de Amazon Aurora sin servidor para mejorar el escalado. Cuando los requisitos de disponibilidad son muy altos, podría ser conveniente no utilizar en absoluto el escritor principal. Para consultas de solo lectura, se pueden utilizar réplicas de lectura, que proporcionan redundancia y capacidad de escalado horizontal, no solo vertical. Las escrituras se pueden almacenar en búfer, por ejemplo, en una cola de Amazon Simple Queue Service, de modo que las solicitudes de escritura de los clientes puedan seguir aceptándose incluso si la principal no está disponible temporalmente.

Recursos

Documentos relacionados:

- Amazon API Gateway: Throttle API Requests for Better Throughput
- CircuitBreaker (summarizes Circuit Breaker from “Release It!” book)
- Error Retries and Exponential Backoff in AWS
- Michael Nygard “Release It! Design and Deploy Production-Ready Software”
- Amazon Builders' Library: Evitar los planes alternativos en los sistemas distribuidos
- Amazon Builders' Library: Cómo evitar demoras de colas insuperables
- Amazon Builders' Library: Desafíos y estrategias del almacenamiento en caché
- Amazon Builders' Library: Tiempos de espera, reintentos y retardo con fluctuación

Videos relacionados:

- Retry, backoff, and jitter: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)

REL05-BP02 Limitación de las solicitudes

Limite las solicitudes para mitigar el agotamiento de los recursos debido a aumentos inesperados de la demanda. Las solicitudes por debajo de los índices de limitación se procesan, pero las que superan el límite definido se rechazan y se envía un mensaje que indica que la solicitud no se ha procesado a causa de la limitación.

Resultado deseado: la limitación de las solicitudes mitiga los grandes picos de volumen, ya sea debido a un aumento repentino del tráfico de clientes, a ataques por desbordamiento o a tormentas de reintentos, lo que permite que las cargas de trabajo sigan procesando de manera normal el volumen de solicitudes admitido.

Patrones comunes de uso no recomendados:

- Las limitaciones de puntos de conexión de la API no se implementan o se mantienen en los valores predeterminados sin tener en cuenta los volúmenes esperados.
- Los puntos de conexión de la API no se someten a pruebas de carga ni se prueban las limitaciones.
- Los índices de solicitudes se limitan sin tener en cuenta el tamaño o la complejidad de las solicitudes.
- Los índices o el tamaño máximos de las solicitudes se prueban, pero por separado.
- Los recursos no se aprovisionan con los mismos límites establecidos en las pruebas.
- No se han configurado ni considerado planes de uso para los consumidores de API de aplicación a aplicación (A2A).
- Los consumidores de cola que escalan horizontalmente no tienen configurado un valor máximo de simultaneidad.
- No se ha implementado la limitación de índices por dirección IP.

Beneficios de establecer esta práctica recomendada: las cargas de trabajo que establecen límites pueden funcionar con normalidad y procesar correctamente la carga de solicitudes aceptada en caso de que se produzcan picos de volumen inesperados. Los picos repentinos o sostenidos de solicitudes a las API y las colas se limitan y no agotan los recursos de procesamiento de solicitudes. Hay límites de índices que limitan a solicitantes individuales para que un gran volumen de tráfico

desde una sola dirección IP o un único consumidor de API no agote los recursos y afecte a otros consumidores.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los servicios deben diseñarse para procesar una capacidad de solicitudes conocida; esta capacidad se puede establecer mediante pruebas de carga. Si los índices de llegada de solicitudes superan los límites, se emite la respuesta correspondiente que indica que la solicitud no se ha procesado a causa de las limitaciones. Esto permite al consumidor gestionar el error y volver a intentarlo más tarde.

Cuando su servicio requiera la implementación de limitaciones, considere la posibilidad de implementar el algoritmo del bucket de tokens, en el que un token se refiere a una solicitud. Los tokens se recargan a un índice de limitación por segundo y se vacían de forma asíncrona a un ritmo de un token por solicitud.




El algoritmo del bucket de tokens.

Amazon API Gateway implementa el algoritmo del bucket de tokens de acuerdo con los límites de la cuenta y la región, y se puede configurar por cliente con planes de uso. Además, Amazon Simple Queue Service (Amazon SQS) y Amazon Kinesis pueden almacenar las solicitudes en búfer para reducir la tasa de solicitudes y permitir tasas de limitación más altas para las solicitudes que se pueden atender. Por último, puede implementar una limitación de velocidad con AWS WAF para limitar los consumidores de API específicos que generan una carga inusualmente alta.

Pasos para la implementación

Puede configurar API Gateway con límites de limitación para sus API y devolver errores `429 Too Many Requests` cuando se superen los límites. Puede utilizar AWS WAF con sus puntos de conexión de AWS AppSync y API Gateway para habilitar la limitación de índices por dirección IP. Además, si su sistema tolera el procesamiento asíncrono, puede colocar los mensajes en una cola o secuencia para acelerar las respuestas a los clientes del servicio, lo que le permite ampliar los índices de limitación más altos.

Con el procesamiento asíncrono, cuando haya configurado Amazon SQS como origen de eventos de AWS Lambda, podrá configurar la máxima simultaneidad para evitar que las altas tasas de eventos consuman la cuota de ejecución simultánea de la cuenta disponible necesaria para otros servicios de su carga de trabajo o cuenta.

Si bien API Gateway proporciona una implementación administrada del bucket de tokens, en los casos en que no pueda usar API Gateway, puede utilizar las implementaciones de código abierto específicas de cada lenguaje (consulte los ejemplos relacionados en Recursos) del bucket de tokens para sus servicios.

- Comprenda y configure los límites de limitación de API Gateway en la cuenta por región, API por etapa y clave de API por nivel de plan de uso.
- Aplique reglas de limitación de tasas de AWS WAF a API Gateway y a los puntos de conexión de AWS AppSync para protegerse contra las inundaciones y bloquear las IP maliciosas. Las reglas de limitación de índices también se pueden configurar en las claves de API de AWS AppSync para los consumidores de A2A.
- Analice si necesita un control de limitación superior a la limitación de índices para las API de AWS AppSync y, de ser así, configure una API Gateway enfrente de su punto de conexión de AWS AppSync.
- Cuando las colas de Amazon SQS se configuran como activadores para los consumidores de colas de Lambda, defina la simultaneidad máxima en un valor que procese lo suficiente como para cumplir sus objetivos de nivel de servicio, pero que no consuma los límites de simultaneidad que afecten a otras funciones de Lambda. Considere la posibilidad de configurar la simultaneidad reservada en otras funciones de Lambda de la misma cuenta y región cuando consuma colas con Lambda.
- Utilice API Gateway con integraciones de servicios nativos para Amazon SQS o Kinesis para almacenar en búfer las solicitudes.

- Si no puede utilizar API Gateway, consulte las bibliotecas específicas del lenguaje para implementar el algoritmo del bucket de tokens para su carga de trabajo. Consulte la sección de ejemplos e investigue por su cuenta para encontrar una biblioteca adecuada.
- Pruebe los límites que tiene pensado establecer o que va a permitir que se aumenten, y documente los límites probados.
- No aumente los límites por encima de lo que establezca en las pruebas. Cuando aumente un límite, antes de aplicar ese aumento, compruebe que los recursos aprovisionados sean equivalentes o superiores a los de las situaciones de prueba.

Recursos

Prácticas recomendadas relacionadas:

- REL04-BP03 Trabajo constante
- REL05-BP03 Control y limitación de las llamadas de reintento

Documentos relacionados:

- Amazon API Gateway: Throttle API Requests for Better Throughput
- AWS WAF: Rate-based rule statement
- Introducing maximum concurrency of AWS Lambda when using Amazon SQS as an event source
- AWS Lambda: Maximum Concurrency

Ejemplos relacionados:

- The three most important AWS WAF rate-based rules
- Java Bucket4j
- Python token-bucket
- Node token-bucket
- .NET System Threading Rate Limiting

Videos relacionados:

- Implementing GraphQL API security best practices with AWS AppSync

Herramientas relacionadas:

- Amazon API Gateway
- AWS AppSync
- Amazon SQS
- Amazon Kinesis
- AWS WAF
- Sala de espera virtual en AWS

REL05-BP03 Control y limitación de las llamadas de reintento

Utilice un retroceso exponencial para reintentar las solicitudes a intervalos progresivamente más largos entre cada reintento. Introduzca una fluctuación entre reintentos para aleatorizar los intervalos de reintentos. Limite el número máximo de reintentos.

Resultado deseado: entre los componentes típicos de un sistema de software distribuido se incluyen servidores, equilibradores de carga, bases de datos y servidores DNS. Durante el funcionamiento normal, estos componentes pueden responder a las solicitudes con errores temporales o limitados, y también con errores que serían persistentes independientemente de los reintentos. Cuando los clientes hacen solicitudes a los servicios, esas solicitudes consumen recursos, como memoria, subprocesos, conexiones, puertos o cualquier otro recurso limitado. Controlar y limitar los reintentos es una estrategia para liberar y minimizar el consumo de recursos, de modo que los componentes del sistema sometidos a presión no se sobrecarguen.

Cuando se agota el tiempo de espera de las solicitudes del cliente o se reciben respuestas de error, deben determinar si deben volver a intentarlo o no. Si lo vuelven a intentar, lo hacen con un retroceso exponencial con fluctuaciones y un valor de reintento máximo. Como resultado, los servicios y procesos de backend tienen menos carga y más tiempo para recuperarse automáticamente, lo que se traduce en una recuperación más rápida y una tramitación satisfactoria de las solicitudes.

Patrones comunes de uso no recomendados:

- Implementar los reintentos sin agregar valores de retroceso exponencial, fluctuación y reintentos máximos. El retroceso y la fluctuación ayudan a evitar picos de tráfico artificiales debidos a reintentos coordinados involuntariamente a intervalos comunes.
- Implementar reintentos sin probar sus efectos o asumir que los reintentos ya están integrados en un SDK sin probar los escenarios de reintento.

- No entender los códigos de error publicados de las dependencias, lo que lleva a volver a intentar todos los errores, incluidos los que tienen una causa clara que indica una falta de permisos, un error de configuración u otro problema que es de esperar que no se pueda resolver sin una intervención manual.
- No utilizar prácticas de observabilidad, como supervisión y alertas en caso de errores de servicio repetidos, para conocer problemas subyacentes y poder solucionarlos.
- Desarrollar mecanismos de reintento personalizados cuando son suficientes las capacidades de reintento integradas o de terceros.
- Reintentar en varias capas de la pila de aplicaciones de una forma que se acumulen, lo que consume aún más recursos en una tormenta de reintentos. Asegúrese de entender cómo afectan estos errores a las dependencias en las que se basa y, a continuación, implemente los reintentos en un solo nivel.
- Reintentar llamadas de servicio que no son idempotentes, lo que provoca efectos secundarios inesperados, como resultados duplicados.

Beneficios de establecer esta práctica recomendada: los reintentos ayudan a los clientes a obtener los resultados deseados cuando las solicitudes fallan, pero también consumen más tiempo del servidor para obtener las respuestas satisfactorias que desean. Cuando los errores son poco frecuentes o transitorios, los reintentos funcionan bien. Cuando los errores se deben a una sobrecarga de recursos, los reintentos pueden empeorar las cosas. Agregar un retroceso exponencial con fluctuaciones para los reintentos de los clientes permite que los servidores se recuperen cuando los errores se deben a una sobrecarga de recursos. La fluctuación evita que haya picos de solicitudes y el retroceso disminuye el escalamiento de la carga provocado por la adición de reintentos a la carga normal de solicitudes. Por último, es importante configurar un número de reintentos máximo o un tiempo transcurrido máximo para evitar que se acumulen tareas pendientes que generen errores metaestables.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Controle y limite las llamadas de reintento. Use el retroceso exponencial para los reintentos tras intervalos cada vez más largos. Introduzca una fluctuación para aleatorizar los intervalos de reintento y limite el número máximo de reintentos.

Algunos AWS SDK implementan los reintentos y el retroceso exponencial de forma predeterminada. Utilice estas implementaciones de AWS integradas cuando corresponda en su carga de trabajo.

Implemente una lógica similar en su carga de trabajo cuando llame a servicios que sean idempotentes y en los que los reintentos mejoren la disponibilidad de sus clientes. Decida cuáles son los tiempos de espera y cuándo dejar de reintentar según su caso de uso. Cree y ejecute situaciones de prueba para esos casos de uso de reintentos.

Pasos para la implementación

- Determine la capa óptima de la pila de aplicaciones para implementar los reintentos de los servicios de los que depende su aplicación.
- Tenga en cuenta que los SDK existentes implementan estrategias de reintento probadas con retroceso exponencial y fluctuaciones para el lenguaje que elija, y dé preferencia a estas estrategias en lugar de escribir sus propias implementaciones de reintentos.
- Verifique que los servicios sean idempotentes antes de implementar los reintentos. Una vez implementados, asegúrese de que se prueben y se utilicen regularmente en producción.
- Al llamar a las API del servicio de AWS, utilice los AWS SDK y AWS CLI y comprenda las opciones de configuración de reintentos. Determine si los valores predeterminados funcionan para su caso de uso, pruébelos y ajústelos según sea necesario.

Recursos

Prácticas recomendadas relacionadas:

- REL04-BP04 Cómo hacer idempotentes las operaciones de mutación
- REL05-BP02 Limitación de las solicitudes
- REL05-BP04 Respuesta rápida a los errores y limitación de las colas
- REL05-BP05 Definición de los tiempos de espera del cliente
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

Documentos relacionados:

- Error Retries and Exponential Backoff in AWS
- Amazon Builders' Library: Tiempos de espera, reintentos y retardo con fluctuación
- Exponential Backoff and Jitter
- Making retries safe with idempotent APIs

Ejemplos relacionados:

- Spring Retry
- Resilience4j Retry

Videos relacionados:

- Retry, backoff, and jitter: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)

Herramientas relacionadas:

- AWS SDKs and Tools: Retry behavior
- AWS Command Line Interface: Reintentos de AWS CLI

REL05-BP04 Respuesta rápida a los errores y limitación de las colas

Cuando un servicio no pueda responder correctamente a una solicitud, responda rápido a los errores. Esto permite que se liberen los recursos asociados a una solicitud y que un servicio se recupere cuando se le agotan los recursos. La respuesta rápida a los errores es un patrón de diseño de software bien establecido que se puede utilizar para conseguir cargas de trabajo enormemente fiables en la nube. Las colas también son un patrón de integración empresarial bien establecido que puede suavizar la carga y permitir a los clientes liberar recursos cuando se pueda tolerar el procesamiento asíncrono. Cuando un servicio puede responder correctamente en condiciones normales, pero falla cuando el índice de solicitudes es demasiado alto, utilice una cola para almacenar en búfer las solicitudes. Sin embargo, no permita que se acumulen largas colas de tareas pendientes, ya que eso podría hacer que se procesaran solicitudes obsoletas a las que un cliente ya ha renunciado.

Resultado deseado: cuando los sistemas sufren contención de recursos, tiempos de espera, excepciones o errores grises que hacen que los objetivos de nivel de servicio sean inalcanzables, las estrategias de respuesta rápida a los errores permiten recuperar el sistema más rápido. Los sistemas que deben absorber los picos de tráfico y pueden adaptarse al procesamiento asíncrono pueden mejorar la fiabilidad al permitir a los clientes liberar rápidamente las solicitudes mediante el uso de colas para almacenar en búfer las solicitudes a los servicios de backend. Cuando las solicitudes a las colas se almacenan en búfer, se implementan estrategias de administración de colas para evitar retrasos insuperables.

Patrones comunes de uso no recomendados:

- Implementar colas de mensajes, pero no configurar colas de mensajes fallidos (DLQ) ni alarmas en los volúmenes de DLQ para detectar cuándo está fallando un sistema.
- No medir la antigüedad de los mensajes de una cola, que es una medida de la latencia para saber cuándo los usuarios de la cola sufren retrasos o producen errores que dan lugar a reintentos.
- No borrar los mensajes pendientes de una cola cuando no sirve de nada procesar esos mensajes si la empresa ya no necesita hacerlo.
- Configurar colas de primero en entrar/primero en salir (FIFO) cuando las colas de último en entrar, primero en salir (LIFO) responderían mejor a las necesidades de los clientes, por ejemplo, cuando no se requieren pedidos estrictos y el procesamiento pendiente retrasa todas las solicitudes nuevas y urgentes, lo que hace que se infrinjan los niveles de servicio de todos los clientes.
- Exponer las colas internas a los clientes en lugar de exponer las API que administran la entrada de trabajo y colocan las solicitudes en colas internas.
- Combinar demasiados tipos de solicitudes de trabajo en una sola cola puede agravar las condiciones de las tareas pendientes al distribuir la demanda de recursos entre los tipos de solicitudes.
- Procesar solicitudes complejas y simples en la misma cola, a pesar de necesitar diferentes niveles de supervisión, tiempos de espera y asignaciones de recursos.
- No validar las entradas ni utilizar afirmaciones para implementar mecanismos de respuesta rápida a los errores en el software que envíen las excepciones a componentes de nivel superior que puedan gestionar los errores con facilidad.
- No eliminar los recursos que fallan del enrutamiento de solicitudes, especialmente cuando los errores grises emiten tanto éxitos como errores debido a bloqueos y reinicios, errores de dependencia intermitentes, una reducción de la capacidad o la pérdida de paquetes de red.

Beneficios de establecer esta práctica recomendada: los sistemas que responden rápido a los errores son más fáciles de depurar y corregir y, a menudo, revelan problemas de codificación y configuración antes de que las versiones se publiquen en producción. Los sistemas que incorporan estrategias de puesta en cola eficaces tienen una mayor resiliencia y fiabilidad a los picos de tráfico y a las condiciones de errores intermitentes del sistema.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las estrategias de respuesta rápida a los errores pueden codificarse en soluciones de software y también configurarse en la infraestructura. Además de la respuesta rápida a los errores, las colas son una técnica arquitectónica sencilla, pero potente, para desacoplar los componentes del sistema sin problemas de carga. Amazon CloudWatch proporciona capacidades para supervisar los errores y alertar en caso de que existan. Una vez que se sabe que un sistema está fallando, se pueden invocar estrategias de mitigación, como el alejamiento de los recursos deteriorados. Cuando los sistemas implementan colas con Amazon SQS y otras tecnologías de cola para facilitar la carga, deben considerar cómo administrar los atrasos en las colas, así como los errores en el consumo de mensajes.

Pasos para la implementación

- Implemente afirmaciones programáticas o métricas específicas en su software y utilícelas para alertar explícitamente sobre problemas del sistema. Amazon CloudWatch le ayuda a crear métricas y alarmas basadas en el patrón de registro de la aplicación y la instrumentación del SDK.
- Utilice métricas y alarmas de CloudWatch para alejarse de los recursos deteriorados que aumentan la latencia del procesamiento o que no procesan las solicitudes de forma reiterada.
- Utilice el procesamiento asíncrono diseñando API que acepten solicitudes y las anexen a las colas internas mediante Amazon SQS y, a continuación, respondan al cliente que produce los mensajes con un mensaje de éxito, de modo que el cliente pueda liberar recursos y continuar con otras tareas mientras los consumidores de la cola del backend procesan las solicitudes.
- Para medir y supervisar la latencia de procesamiento de las colas, genere una métrica de CloudWatch cada vez que se retire un mensaje de una cola mediante la comparación en ese momento con la marca de tiempo del mensaje.
- Cuando los errores impidan procesar correctamente los mensajes o los picos de tráfico en los volúmenes que no se pueden procesar dentro de los acuerdos de nivel de servicio, aparte el tráfico antiguo o excesivo y colóquelo en una cola secundaria. Esto permite procesar de forma prioritaria los trabajos nuevos y dejar los antiguos para cuando haya capacidad disponible. Esta técnica es una aproximación al procesamiento LIFO y permite que el sistema procese normalmente todos los trabajos nuevos.
- Utilice colas de mensajes fallidos o redireccione las colas para sacar de la lista de espera los mensajes que no se puedan procesar y colocarlos en una ubicación que pueda investigarse y resolverse más adelante

- Vuelva a intentarlo o, cuando sea tolerable, elimine los mensajes antiguos. Para ello, compárelos en ese momento con la marca de tiempo del mensaje y descarte los mensajes que ya no sean pertinentes para el cliente que los ha solicitado.

Recursos

Prácticas recomendadas relacionadas:

- REL04-BP02 Implementación de dependencias con acoplamiento débil
- REL05-BP02 Limitación de las solicitudes
- REL05-BP03 Control y limitación de las llamadas de reintento
- REL06-BP02 Definición y cálculo de métricas (agregación)
- REL06-BP07 Supervisión del seguimiento de las solicitudes de principio a fin en todo el sistema

Documentos relacionados:

- Cómo evitar demoras de colas insuperables
- Fail Fast
- ¿Cómo puedo evitar que se acumulen mensajes en mi cola de Amazon SQS?
- Elastic Load Balancing: Zonal Shift
- Amazon Application Recovery Controller: Routing control for traffic failover

Ejemplos relacionados:

- Enterprise Integration Patterns: Dead Letter Channel

Videos relacionados:

- AWS re:Invent 2022 - Operating highly available Multi-AZ applications

Herramientas relacionadas:

- Amazon SQS
- Amazon MQ
- AWS IoT Core

- Amazon CloudWatch

REL05-BP05 Definición de los tiempos de espera del cliente

Defina tiempos de espera adecuados para las conexiones y las solicitudes, verifíquelos sistemáticamente y no use los valores predeterminados, ya que no tienen en cuenta las características específicas de la carga de trabajo.

Resultado deseado: en los tiempos de espera de los clientes, se debe tener en cuenta el costo para el cliente, el servidor y la carga de trabajo asociados a la espera de las solicitudes que tardan un tiempo anormal en completarse. Dado que no es posible conocer la causa exacta de ningún tiempo de espera, los clientes deben utilizar el conocimiento de los servicios para fijar expectativas sobre las causas probables y los tiempos de espera adecuados

El tiempo de espera de las conexiones del cliente se agota en función de los valores configurados. Cuando el tiempo de espera se agota, los clientes toman la decisión de dar marcha atrás y volver a intentarlo o abrir un disyuntor. Estos patrones evitan que se emitan solicitudes que puedan agravar una condición de error subyacente.

Patrones comunes de uso no recomendados:

- No estar al tanto de los tiempos de espera del sistema o de los tiempos de espera predeterminados.
- No estar al tanto del tiempo normal de finalización de las solicitudes.
- No conocer las posibles causas por las que las solicitudes tardan un tiempo anormalmente largo en completarse ni los costos para el rendimiento del cliente, el servicio o la carga de trabajo asociados a la espera a que se completen.
- No conocer la probabilidad de que la red deteriorada haga que una solicitud falle solo una vez que se haya agotado el tiempo de espera, ni de los costos que supone para el rendimiento del cliente y la carga de trabajo no utilizar un tiempo de espera más corto.
- No probar escenarios de tiempo de espera tanto para las conexiones como para las solicitudes.
- Definir tiempos de espera demasiado altos, lo que puede provocar tiempos de espera prolongados y aumentar el uso de los recursos.
- Definir tiempos de espera demasiado bajos, lo que provoca errores artificiales.
- Pasar por alto los patrones para solucionar los errores de tiempo de espera de las llamadas remotas, como disyuntores y reintentos.

- No considerar la posibilidad de supervisar los índices de errores de las llamadas de servicio, los objetivos de nivel de servicio referentes a la latencia y los valores atípicos de latencia. Estas métricas pueden proporcionar información sobre tiempos de espera agresivos o permisivos

Beneficios de establecer esta práctica recomendada: los tiempos de espera de las llamadas remotas están configurados y los sistemas están diseñados para gestionar los tiempos de espera correctamente, de modo que los recursos se conserven cuando las llamadas remotas responden con una lentitud anormal y los clientes del servicio gestionan correctamente los errores de tiempo de espera.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Defina un tiempo de espera de conexión y un tiempo de espera de solicitud en cualquier llamada de dependencia del servicio y, normalmente, en todas las llamadas de los procesos. Muchos marcos integran capacidades de tiempo de espera, pero tenga cuidado, ya que algunos tienen valores predeterminados que son infinitos o superiores a lo aceptable para sus objetivos de servicio. Un valor demasiado alto reduce la utilidad del tiempo de espera porque se siguen consumiendo recursos mientras el cliente espera a que transcurra el tiempo de espera. Un valor demasiado bajo puede generar un aumento del tráfico en el backend y un aumento de la latencia debido a que las solicitudes hacen demasiados reintentos. En algunos casos, esto puede producir una interrupción completa si se reintentan todas las solicitudes.

Tenga en cuenta lo siguiente al determinar las estrategias de tiempo de espera:

- Las solicitudes pueden tardar más de lo normal en procesarse debido a su contenido, a deficiencias en un servicio de destino o a un error en la partición de la red.
- Las solicitudes con contenido anormalmente caro podrían consumir recursos innecesarios del servidor y del cliente. En este caso, si se agota el tiempo de espera de estas solicitudes y no se vuelven a intentar, se pueden conservar los recursos. Los servicios también deberían protegerse del contenido anormalmente caro con restricciones y tiempos de espera del lado del servidor.
- Se puede agotar el tiempo de espera y volver a intentar las solicitudes que tarden un tiempo anormalmente largo debido a una interrupción del servicio. Se deben tener en cuenta los costos del servicio de la solicitud y el reintento, pero si la causa es una deficiencia localizada, es probable que el reintento no sea caro y reduzca el consumo de recursos del cliente. El tiempo de espera también puede liberar recursos del servidor según la naturaleza de la deficiencia.

- Se puede agotar el tiempo de espera y volver a intentar las solicitudes que tarden mucho en completarse porque la red no ha podido entregar la solicitud o la respuesta. Como la solicitud o la respuesta no se han entregado, el resultado habría sido un error independientemente del tiempo de espera. En este caso, el tiempo de espera no liberará los recursos del servidor, pero sí liberará los recursos del cliente y mejorará el rendimiento de la carga de trabajo.

Aproveche patrones de diseño bien establecidos, como los reintentos y los disyuntores, para gestionar los tiempos de espera correctamente y ofrecer enfoques de respuesta rápida a los errores. AWS Los SDK y AWS CLI permiten configurar los tiempos de espera de conexión y solicitud y los reintentos con retrocesos y fluctuaciones exponenciales. Las funciones de AWS Lambda permiten configurar los tiempos de espera y, con AWS Step Functions, se pueden crear disyuntores con poco código que aprovechen las integraciones predefinidas con servicios de AWS y SDK. AWS App Mesh Envoy incluye capacidades de tiempo de espera y de disyuntor.

Pasos para la implementación

- Configure tiempos de espera en las llamadas de servicio remotas y aproveche las características integradas de tiempo de espera del lenguaje o las bibliotecas de tiempo de espera de código abierto.
- Cuando su carga de trabajo haga llamadas con un AWS SDK, consulte la documentación para ver la configuración del tiempo de espera específica de cada lenguaje.
  - Python
  - PHP
  - .NET
  - Ruby
  - Java
  - Go
  - Node.js
  - C++
- Cuando utilice AWS SDK o comandos de la AWS CLI en su carga de trabajo, defina los valores predeterminados de tiempo de espera mediante la configuración de los valores predeterminados de configuración de AWS para `connectTimeoutInMillis` y `tlsNegotiationTimeoutInMillis`.
- Aplique las opciones de línea de comandos `cli-connect-timeout` y `cli-read-timeout` para controlar comandos únicos de la AWS CLI para los servicios de AWS.

- Supervise las llamadas de servicio remotas para comprobar si hay tiempos de espera y configure alarmas en caso de errores persistentes para poder gestionar los escenarios de error de forma proactiva.
- Implemente métricas de CloudWatch y la detección de anomalías de CloudWatch en los índices de error de las llamadas, los objetivos de nivel de servicio en lo que se refiere a la latencia y los valores atípicos de latencia para proporcionar información sobre la administración de tiempos de espera demasiado agresivos o permisivos.
- Configure los tiempos de espera en las funciones de Lambda.
- Los clientes de API Gateway deben implementar sus propios reintentos al gestionar los tiempos de espera. API Gateway admite un tiempo de espera de integración de 50 milisegundos a 29 segundos para las integraciones posteriores y no lo vuelve a intentar cuando la integración solicita el tiempo de espera.
- Implemente el patrón de disyuntor para que no se hagan llamadas remotas cuando se agote el tiempo de espera. Abra el circuito para evitar llamadas fallidas y ciérrelo cuando las llamadas respondan con normalidad.
- Para las cargas de trabajo basadas en contenedores, consulte las funciones de App Mesh Envoy para aprovechar los tiempos de espera y los disyuntores integrados.
- Utilice AWS Step Functions para crear disyuntores de poco código para las llamadas de servicio remotas, especialmente cuando se utilizan AWS SDK nativos e integraciones de Step Functions compatibles para simplificar la carga de trabajo.

Recursos

Prácticas recomendadas relacionadas:

- REL05-BP03 Control y limitación de las llamadas de reintento
- REL05-BP04 Respuesta rápida a los errores y limitación de las colas
- REL06-BP07 Supervisión del seguimiento de las solicitudes de principio a fin en todo el sistema

Documentos relacionados:

- AWS SDK: Retries and Timeouts
- Amazon Builders' Library: Tiempos de espera, reintentos y retardo con fluctuación
- Cuotas de Amazon API Gateway y notas importantes
- AWS Command Line Interface: Command line options

- AWS SDK for Java 2.x: Configure API Timeouts
- AWS Botocore using the config object and Config Reference
- AWS SDK para .NET: Retries and Timeouts
- AWS Lambda: Configuración de funciones de Lambda

Ejemplos relacionados:

- Using the circuit breaker pattern with AWS Step Functions and Amazon DynamoDB
- Martin Fowler: CircuitBreaker

Herramientas relacionadas:

- AWS SDK
- AWS Lambda
- Amazon SQS
- AWS Step Functions
- AWS Command Line Interface

REL05-BP06 Creación de sistemas sin estado cuando sea posible

Los sistemas deben o bien no requerir estado o bien descargar el estado, de forma que entre solicitudes de clientes distintos no haya dependencia en los datos almacenados localmente en disco y en memoria. Esto permite reemplazar los servidores a voluntad sin que la disponibilidad resulte afectada.

Cuando los usuarios o los servicios interactúan con una aplicación, suelen llevar a cabo una serie de interacciones que constituyen una sesión. Una sesión es un dato único para los usuarios que persiste entre las solicitudes mientras utilizan la aplicación. Una aplicación sin estado es aquella que no necesita conocer las interacciones anteriores y no almacena la información de la sesión.

Una vez se ha diseñado para no tener estado, puede utilizar servicios de computación sin servidor, como AWS Lambda o AWS Fargate.

Además del reemplazo del servidor, otro beneficio de las aplicaciones sin estado es que pueden escalar horizontalmente porque cualquiera de los recursos de computación disponibles (como las instancias de EC2 y las funciones de AWS Lambda) puede dar servicio a cualquier solicitud.

Beneficios de establecer esta práctica recomendada: los sistemas que se han diseñado para no tener estado se adaptan mejor al escalado horizontal, lo que permite agregar o eliminar capacidad en función de la fluctuación del tráfico y la demanda. También son intrínsecamente resistentes a los errores y proporcionan flexibilidad y agilidad en el desarrollo de aplicaciones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cree aplicaciones sin estado. Las aplicaciones sin estado permiten el escalado horizontal y toleran el error de un nodo individual. Analice y comprenda los componentes de la aplicación que mantienen el estado dentro de la arquitectura. Esto le ayuda a evaluar el posible impacto de la transición a un diseño sin estado. Una arquitectura sin estado desacopla los datos del usuario y descarga los datos de la sesión. Esto proporciona la flexibilidad para escalar cada componente de forma independiente para cumplir con las diferentes demandas de carga de trabajo y optimizar el uso de los recursos.

Pasos para la implementación

- Identifique y comprenda los componentes con estado de la aplicación.
- Para desacoplar los datos, separe y administre los datos de usuario de la lógica principal de la aplicación.
  - Amazon Cognito puede desacoplar los datos de usuario del código de la aplicación mediante características, tales como grupos de identidades, grupos de usuarios y Amazon Cognito Sync.
  - Para usar AWS Secrets Manager a fin de desacoplar los datos de usuario, almacene los secretos en una ubicación segura y centralizada. Esto significa que el código de la aplicación no necesita almacenar secretos, lo que la hace más segura.
  - Plantéese utilizar Amazon S3 para almacenar datos no estructurados y de gran volumen, como imágenes y documentos. La aplicación puede recuperar estos datos cuando sea necesario, lo que elimina la necesidad de almacenarlos en la memoria.
  - Utilice Amazon DynamoDB para almacenar información, como, por ejemplo, perfiles de usuario. La aplicación puede consultar estos datos prácticamente en tiempo real.
- Descargue los datos de la sesión en una base de datos, caché o archivos externos.
  - Amazon ElastiCache, Amazon DynamoDB, Amazon Elastic File System (Amazon EFS) y Amazon MemoryDB son ejemplos de servicios de AWS que puede usar para descargar datos de sesión.
- Diseñe una arquitectura sin estado después de identificar qué datos de estado y de usuario deben conservarse con la solución de almacenamiento que elija.

Recursos

Prácticas recomendadas relacionadas:

- REL11-BP03 Automatización de la reparación en todas las capas

Documentos relacionados:

- Amazon Builders' Library: Evitar los planes alternativos en los sistemas distribuidos
- Amazon Builders' Library: Cómo evitar demoras de colas insuperables
- Amazon Builders' Library: Desafíos y estrategias del almacenamiento en caché
- Prácticas recomendadas para el nivel web sin estado en AWS

REL05-BP07 Implementación de recursos de emergencia

Los recursos de emergencia son procesos rápidos que pueden mitigar el impacto en la disponibilidad de la carga de trabajo.

Los recursos de emergencia desactivan, limitan o cambian el comportamiento de componentes o dependencias mediante mecanismos conocidos y probados. Esto puede aliviar las deficiencias de la carga de trabajo causadas por el agotamiento de los recursos debido a los aumentos inesperados de la demanda y reducir el impacto de los fallos en los componentes no críticos de la carga de trabajo.

Resultado deseado: al implementar recursos de emergencia, puede establecer procesos que se sabe que son buenos para mantener la disponibilidad de los componentes críticos de su carga de trabajo. La carga de trabajo debe degradarse de forma estable y seguir llevando a cabo sus funciones críticas para la empresa durante la activación de un recurso de emergencia. Para obtener más información sobre la degradación estable, consulte REL05-BP01 Implementación de una degradación estable para transformar las dependencias estrictas en flexibles.

Patrones comunes de uso no recomendados:

- El fallo de las dependencias no críticas repercute en la disponibilidad de su carga de trabajo principal.
- No probar o verificar el comportamiento de los componentes críticos durante el deterioro de los componentes no críticos.
- No definir criterios claros y deterministas para la activación o desactivación de un recurso de emergencia.

Beneficios de establecer esta práctica recomendada: la implementación de recursos de emergencia puede mejorar la disponibilidad de los componentes críticos de su carga de trabajo al proporcionar a sus solucionadores procesos establecidos para responder a picos inesperados de demanda o fallos de dependencias no críticas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Identifique los componentes críticos de su carga de trabajo.
- Diseñe y cree los componentes críticos de su carga de trabajo para que resistan los fallos de los componentes no críticos.
- Haga pruebas para validar el comportamiento de sus componentes críticos durante el fallo de los componentes no críticos.
- Defina y supervise las métricas o los factores desencadenantes relevantes para iniciar los procedimientos de recursos de emergencia.
- Defina los procedimientos (manuales o automáticos) que componen el recurso de emergencia.

Pasos para la implementación

- Identifique los componentes críticos para la empresa en su carga de trabajo.
  - Cada componente técnico de su carga de trabajo debe asignarse a su función empresarial relevante y clasificarse como crítico o no crítico. Para ver ejemplos de funciones críticas y no críticas de Amazon, consulte Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second.
  - Se trata de una decisión tanto técnica como empresarial, y varía según la organización y la carga de trabajo.
- Diseñe y cree los componentes críticos de su carga de trabajo para que resistan los fallos de los componentes no críticos.
  - Durante el análisis de dependencias, tenga en cuenta todos los modos de fallo potenciales y verifique que sus mecanismos de recursos de emergencia proporcionan la funcionalidad crítica a los componentes descendentes.
- Haga pruebas para validar el comportamiento de sus componentes críticos durante la activación de sus recursos de emergencia.
  - Evite el comportamiento bimodal. Para obtener más información, consulte REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal.

- Defina, supervise y alerte sobre las métricas relevantes para iniciar el procedimiento del recurso de emergencia.
  - Encontrar las métricas adecuadas para supervisar depende de su carga de trabajo. Algunos ejemplos de métricas son la latencia o el número de solicitudes fallidas a una dependencia.
- Defina los procedimientos manuales o automáticos que componen el recurso de emergencia.
  - Esto puede incluir mecanismos como el desbordamiento de carga, la limitación de solicitudes o la implementación de una degradación estable.

Recursos

Prácticas recomendadas relacionadas:

- REL05-BP01 Implementación de una degradación estable para transformar las dependencias estrictas en flexibles
- REL05-BP02 Limitación de las solicitudes
- REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal

Documentos relacionados:

- Automatización de implementaciones seguras y sin intervención
- Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second

Videos relacionados:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability

## Administración de cambios

Preguntas

- REL 6. ¿Cómo se supervisan los recursos de la carga de trabajo?
- REL 7. ¿Cómo diseña su carga de trabajo para adaptarla a los cambios de la demanda?
- REL 8. ¿Cómo implementa el cambio?

## REL 6. ¿Cómo se supervisan los recursos de la carga de trabajo?

Los registros y las métricas son herramientas poderosas para obtener información sobre el estado de su carga de trabajo. Puede configurar su carga de trabajo para supervisar los registros y las métricas y enviar notificaciones cuando se superen los umbrales o se produzcan eventos significativos. La supervisión permite que su carga de trabajo reconozca cuándo se cruzan umbrales de bajo rendimiento o se producen errores, para que pueda recuperarse de los errores de forma automática una vez recibida una respuesta.

Prácticas recomendadas

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP02 Definición y cálculo de métricas (agregación)
- REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)
- REL06-BP04 Automatización de las respuestas (procesamiento y alarmas en tiempo real)
- REL06-BP05 Análisis de registros
- REL06-BP06 Revisiones frecuentes
- REL06-BP07 Supervisión del seguimiento de las solicitudes de principio a fin en todo el sistema

### REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)

Supervise los componentes de la carga de trabajo con Amazon CloudWatch o herramientas de terceros. Supervise los servicios de AWS con el panel de AWS Health.

Debe supervisar todos los componentes de su carga de trabajo, incluidos los niveles del frontend, la lógica empresarial y el almacenamiento. Defina métricas claves, describa cómo extraerlas de los registros (si fuera necesario) y establezca umbrales para desencadenar los eventos de alarma correspondientes. Asegúrese de que las métricas sean pertinentes para los indicadores clave de rendimiento (KPI) de su carga de trabajo, y utilice métricas y registros para identificar signos de advertencia tempranos de degradación del servicio. Por ejemplo, una métrica relacionada con los resultados empresariales como el número de pedidos procesado satisfactoriamente por minuto, puede indicar problemas con la carga de trabajo más rápido que una métrica técnica, como el uso de la CPU. Utilice el panel de AWS Health para obtener una vista personalizada sobre el rendimiento y la disponibilidad de los servicios de AWS subyacentes a sus recursos de AWS.

La supervisión en la nube ofrece nuevas oportunidades. La mayoría de proveedores en la nube han desarrollado enlaces personalizables y pueden proporcionar conocimientos para ayudarle a

supervisar varias capas de su carga de trabajo. Los servicios de AWS como Amazon CloudWatch aplican algoritmos estadísticos y de machine learning para analizar continuamente las métricas de los sistemas y aplicaciones, determinar las bases de referencia normales y hacer aflorar anomalías con una intervención mínima del usuario. Los algoritmos de detección de anomalías dan cuenta de la estacionalidad y los cambios de tendencia de las métricas.

AWS pone a disposición una gran cantidad de información de supervisión y registro para el consumo que se puede usar para definir métricas específicas de la carga de trabajo, procesos de cambio en la demanda y adoptar técnicas de machine learning independientemente de los conocimientos sobre ML.

Además, puede supervisar todos sus puntos de conexión externos para asegurarse de que sean independientes de su implementación base. Esta supervisión activa se puede llevar a cabo con transacciones sintéticas (a las que a veces se denomina canarios de usuario, y que no deben confundirse con las implementaciones canario), que ejecutan periódicamente varias tareas comunes que se ajustan a las acciones que hacen los clientes de la carga de trabajo. Mantenga una duración breve para estas tareas y asegúrese de no sobrecargar sus cargas de trabajo durante las pruebas. Amazon CloudWatch Synthetics le permite crear canarios sintéticos para supervisar sus puntos de conexión y API. También puede combinar los nodos de cliente del canario sintético con la consola de AWS X-Ray para detectar qué canarios sintéticos están teniendo problemas de errores, fallos o limitaciones para el periodo de tiempo seleccionado.

Resultado deseado:

Recopila y usa métricas esenciales de todos los componentes de la carga de trabajo para garantizar la fiabilidad de la carga de trabajo y una experiencia de usuario óptima. Detectar que una carga de trabajo no está logrando resultados empresariales le permite declarar rápidamente un desastre y recuperarse de un incidente.

Patrones comunes de uso no recomendados:

- Supervisar solamente las interfaces externas con su carga de trabajo.
- No generar métricas específicas de una carga de trabajo y basarse solamente en las métricas que proporcionan los servicios de AWS que usa su carga de trabajo.
- Usar exclusivamente métricas técnicas en su carga de trabajo y no supervisar las métricas relacionadas con KPI no técnicos a los que contribuye la carga de trabajo.
- Confiar en el tráfico de producción y las comprobaciones de estado sencillas para supervisar y evaluar el estado de las cargas de trabajo.

Beneficios de establecer esta práctica recomendada: la supervisión de todos los niveles de la carga de trabajo le permite prever y resolver los problemas rápidamente en los componentes de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

1. Active el registro cuando esté disponible. La supervisión de los datos debe obtenerse a partir de todos los componentes de las cargas de trabajo. Active métodos de registro adicionales, como los registros de acceso de S3, y permita que su carga de trabajo registre datos específicos de la carga de trabajo. Recopile métricas para los promedios de CPU, E/S de red y E/S de disco de servicios como Amazon ECS, Amazon EKS, Amazon EC2, Elastic Load Balancing, AWS Auto Scaling y Amazon EMR. Consulte Servicios de AWS que publican métricas de CloudWatch para consultar una lista de servicios de AWS que publican métricas en CloudWatch.
2. Revise todas las métricas predeterminadas y explore las carencias en cuanto a recopilación de datos. Todos los servicios generan métricas predeterminadas. La recopilación de métricas predeterminadas le permite comprender mejor las dependencias entre los componentes de la carga de trabajo, y cómo la fiabilidad y el rendimiento de los componentes afectan a la carga de trabajo. También puede crear y publicar sus propias métricas en CloudWatch mediante la AWS CLI o una API.
3. Evalúe todas las métricas para decidir sobre cuáles alertar en cada servicio de AWS en su carga de trabajo. Puede decidir seleccionar un subconjunto de métricas que tenga un impacto importante en la fiabilidad de la carga de trabajo. Al centrarse en las métricas y umbrales críticos, podrá refinar el número de alertas de emergencia y contribuir a reducir al mínimo los falsos positivos.
4. Defina las alertas y los procesos de recuperación para su carga de trabajo una vez que se active la alerta. La definición de alertas le permite notificar, escalar y seguir los pasos necesarios rápidamente para recuperarse de un incidente y cumplir el objetivo de tiempo de recuperación (RTO) prescrito. Puede usar alarmas de Amazon CloudWatch para invocar flujos de trabajo automatizados e iniciar procedimientos de recuperación basados en los umbrales definidos.
5. Explore el uso de transacciones sintéticas para recopilar datos relevantes sobre el estado de las cargas de trabajo. La supervisión sintética sigue las mismas rutas y lleva a cabo las mismas acciones que un cliente, lo que le permite verificar continuamente su experiencia de usuario incluso si no tiene tráfico de cliente en sus cargas de trabajo. Al usar transacciones sintéticas, puede detectar los problemas antes de que lo hagan los clientes.

Recursos

Prácticas recomendadas relacionadas:

- REL11-BP03 Automatización de la reparación en todas las capas

Documentos relacionados:

- Getting started with your AWS Health Dashboard – Your account health
- Servicios de AWS que publican métricas de CloudWatch
- Access Logs for Your Network Load Balancer
- Access logs for your application load balancer
- Uso de Registros de Amazon CloudWatch con AWS Lambda
- Registro de acceso al servidor de Simple Storage Service (Amazon S
- Enable Access Logs for Your Classic Load Balancer
- Exporting log data to Amazon S3
- Instalación del agente de CloudWatch en una instancia de Amazon EC2
- Publicar métricas personalizadas
- Uso de paneles de Amazon CloudWatch
- Uso de métricas de Amazon CloudWatch
- Uso de canarios (Amazon CloudWatch Synthetics)
- What are Amazon CloudWatch Logs?

Guías del usuario:

- Creating a trail
- Supervisión de memoria y métricas del disco para las instancias de Linux de Amazon EC2
- Uso de Registros de CloudWatch con instancias de contenedor
- registros de flujo de VPC
- What is Amazon DevOps Guru?
- Qué es AWS X-Ray?

Blogs relacionados:

- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray

Ejemplos relacionados:

- Amazon Builders' Library: Instrumentación de los sistemas distribuidos para obtener visibilidad operativa
- Observability workshop

REL06-BP02 Definición y cálculo de métricas (agregación)

Recopile métricas y registros de los componentes de su carga de trabajo y calcule las métricas agregadas relevantes a partir de ellos. Estas métricas proporcionan una observabilidad amplia y profunda de su carga de trabajo y pueden mejorar significativamente su posición de resiliencia.

La observabilidad es algo más que recopilar métricas de los componentes de la carga de trabajo a fin de poder verlas y alertar sobre ellas. Se trata de obtener una visión de conjunto de cómo se comporta la carga de trabajo. Esta información de comportamiento proviene de todos los componentes de sus cargas de trabajo, que incluyen los servicios en la nube de los que dependen, los registros bien elaborados y las métricas. Estos datos le permiten supervisar el comportamiento de la carga de trabajo en su conjunto, así como comprender la interacción de cada componente con cada unidad de trabajo con un nivel de detalle preciso.

Resultado deseado:

- Recopila registros de los componentes de su carga de trabajo y las dependencias de los servicios de AWS y los publica en una ubicación central donde se puede acceder a ellos y procesarlos fácilmente.
- Sus registros contienen marcas de tiempo precisas y de alta fidelidad.
- Sus registros contienen información relevante sobre el contexto de procesamiento, como un identificador de rastreo, un identificador de usuario o de cuenta y una dirección IP remota.
- Crea métricas agregadas a partir de sus registros, que representan el comportamiento de su carga de trabajo desde una perspectiva de alto nivel.
- Puede consultar sus registros agregados para obtener información detallada y relevante sobre su carga de trabajo e identificar problemas reales y potenciales.

Patrones comunes de uso no recomendados:

- No recopila registros ni métricas relevantes de las instancias de computación en las que se ejecutan sus cargas de trabajo ni de los servicios en la nube que utilizan.
- Pasa por alto la recopilación de registros y métricas relacionados con los indicadores clave de rendimiento (KPI) de su empresa.
- Analiza la telemetría relacionada con la carga de trabajo de forma aislada, sin agregación ni correlación.
- Permite que las métricas y los registros caduquen demasiado rápido, lo que dificulta el análisis de tendencias y la identificación de problemas recurrentes.

Ventajas de establecer estas prácticas recomendadas: puede detectar más anomalías y correlacionar eventos y métricas entre los distintos componentes de su carga de trabajo. Puede crear información estratégica a partir de los componentes de su carga de trabajo en función de la información contenida en los registros que, por lo general, no está disponible únicamente en las métricas. Puede determinar las causas de los errores con mayor rapidez consultando sus registros a escala.

Nivel de exposición al riesgo si no se establecen estas prácticas recomendadas: alto

Guía para la implementación

Identifique las fuentes de datos de telemetría que son relevantes para sus cargas de trabajo y sus componentes. Estos datos provienen no solo de los componentes que publican métricas, como el sistema operativo (SO), y los tiempos de ejecución de las aplicaciones, como Java, sino también de los registros de aplicaciones y servicios en la nube. Por ejemplo, los servidores web suelen registrar cada solicitud con información detallada, como la marca de tiempo, la latencia del procesamiento, el ID de usuario, la dirección IP remota, la ruta y la cadena de consulta. El nivel de detalle de estos registros lo ayuda a realizar consultas detalladas y a generar métricas que de otro modo no estarían disponibles.

Recopile las métricas y los registros mediante las herramientas y los procesos adecuados. Un agente, como el Agente de Amazon CloudWatch, puede recopilar los registros generados por las aplicaciones que se ejecutan en una instancia de Amazon EC2 y publicarlos en un servicio de almacenamiento central, como Registros de Amazon CloudWatch. Los servicios de cómputo gestionados por AWS, como AWS Lambda y Amazon Elastic Container Service, publican automáticamente los registros en CloudWatch Logs. Habilite la recopilación de registros para los servicios de almacenamiento y procesamiento de AWS que utilizan sus cargas de trabajo, como Amazon CloudFront, Amazon S3, Elastic Load Balancing y Amazon API Gateway.

Mejore sus datos de telemetría con dimensiones que lo ayuden a ver los patrones de comportamiento con mayor claridad y a aislar los problemas correlacionados en grupos de componentes relacionados. Una vez agregados, puede observar el comportamiento de los componentes con un nivel de detalle más preciso, detectar los fallos correlacionados y tomar las medidas correctivas adecuadas. Algunos ejemplos de dimensiones útiles son la zona de disponibilidad, el ID de instancia de EC2 y la tarea de contenedor o ID de pod.

Una vez recopiladas las métricas y los registros, puede escribir consultas y generar métricas agregadas a partir de ellas que proporcionen información útil sobre el comportamiento normal y anómalo. Por ejemplo, puede utilizar Información de registros de Amazon CloudWatch para obtener métricas personalizadas de los registros de las aplicaciones, Información de métricas de Amazon CloudWatch para consultar las métricas a escala, Información de contenedores de Amazon CloudWatch para recopilar, agregar y resumir las métricas y los registros de sus aplicaciones y microservicios en contenedores, o Lambda Insights de Amazon CloudWatch si utiliza funciones de AWS Lambda. Para crear una métrica de tasa de error agregada, puede sumar un contador cada vez que encuentre una respuesta o mensaje de error en los registros de sus componentes, o bien calcular el valor agregado de una métrica de tasa de error existente. Puede utilizar estos datos para generar histogramas que muestren el comportamiento final, como las solicitudes o los procesos con peor rendimiento. También puede escanear estos datos en tiempo real para detectar patrones anómalos mediante soluciones como la detección de anomalías de CloudWatch Logs. Esta información estratégica se puede colocar en los paneles de control para mantenerlos organizados de acuerdo con sus necesidades y preferencias.

La consulta de los registros puede ayudarlo a comprender cómo gestionaron las solicitudes específicas los componentes de la carga de trabajo y a revelar los patrones de las solicitudes u otro contexto que repercuta en la resiliencia de la carga de trabajo. Puede resultar útil investigar y preparar las consultas con antelación, en función de sus conocimientos sobre el comportamiento de sus aplicaciones y otros componentes, de forma que pueda ejecutarlas más fácilmente según sea necesario. Por ejemplo, con Información de registros de CloudWatch, puede buscar y analizar de forma interactiva los datos de registro almacenados en Registros de Amazon CloudWatch. También puede usar Amazon Athena para consultar registros de varias fuentes, incluidos muchos servicios de AWS, a escala de petabytes.

Al definir una política de retención de registros, tenga en cuenta el valor de los registros históricos. Los registros históricos pueden ayudar a identificar los patrones de uso y comportamiento a largo plazo, las regresiones y las mejoras en el rendimiento de la carga de trabajo. Los registros eliminados permanentemente no se pueden analizar más adelante. Sin embargo, el valor de los registros históricos tiende a disminuir durante largos periodos de tiempo. Elija una política que

equilibre sus necesidades según corresponda y que cumpla con todos los requisitos legales o contractuales a los que pueda estar sujeto.

Pasos para la implementación

1. Elija mecanismos de recopilación, almacenamiento, análisis y visualización para sus datos de observabilidad.
2. Instale y configure los recopiladores de métricas y registros en los componentes correspondientes de su carga de trabajo (por ejemplo, en las instancias de Amazon EC2 y en los contenedores sidecar). Configure estos recopiladores para que se reinicien automáticamente si se detienen inesperadamente. Habilite el almacenamiento en búfer de disco o memoria para los recopiladores para que los errores de publicación temporales no afecten a sus aplicaciones ni provoquen la pérdida de datos.
3. Habilite el inicio de sesión en los servicios de AWS que utiliza como parte de sus cargas de trabajo y, si es necesario, reenvíe esos registros al servicio de almacenamiento que haya seleccionado. Consulte las guías de usuario o desarrollador de los servicios correspondientes para obtener instrucciones detalladas.
4. Defina las métricas operativas relevantes para sus cargas de trabajo en función de sus datos de telemetría. Podrían basarse en métricas directas emitidas por los componentes de la carga de trabajo, que pueden incluir métricas relacionadas con los KPI empresariales, o en los resultados de cálculos agregados, como sumas, tasas, percentiles o histogramas. Calcule estas métricas con su analizador de registros y colóquelas en los paneles según corresponda.
5. Prepare las consultas de registro adecuadas para analizar los componentes de la carga de trabajo, las solicitudes o el comportamiento de las transacciones, según sea necesario.
6. Defina y habilite una política de retención de registros para los registros de sus componentes. Elimine periódicamente los registros cuando sean más antiguos de lo que permite la política.

Recursos

Prácticas recomendadas relacionadas:

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)
- REL06-BP04 Automatización de las respuestas (procesamiento y alarmas en tiempo real)
- REL06-BP05 Análisis de registros
- REL06-BP06 Revisiones frecuentes

- REL06-BP07 Supervisión del seguimiento de las solicitudes de principio a fin en todo el sistema

Documentación relacionada:

- Funcionamiento de Amazon CloudWatch.
- Amazon Managed Prometheus
- Amazon Managed Grafana
- Analyzing Log Data with CloudWatch Logs Insights
- Lambda Insights de Amazon CloudWatch
- Información de contenedores de Amazon CloudWatch
- Consulta de métricas de CloudWatch con Información de métricas de CloudWatch
- AWS Distro para OpenTelemetry
- Amazon CloudWatch Logs Insights Sample Queries
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- Searching and Filtering Log Data
- Sending Logs Directly to Amazon S3
- Amazon Builders' Library: Instrumentación de los sistemas distribuidos para obtener visibilidad operativa

Talleres relacionados:

- One Observability Workshop

Herramientas relacionadas:

- AWS Distro para OpenTelemetry (GitHub)

REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)

Cuando las organizaciones detectan posibles problemas, envían notificaciones y alertas en tiempo real al personal y los sistemas correspondientes para poder responder de manera rápida y eficaz a estos problemas.

Resultado deseado: es posible responder rápidamente a los eventos operativos con la configuración de las alarmas correspondientes en función de las métricas del servicio y la aplicación. Cuando se

superan los umbrales de alarma, se avisa al personal y a los sistemas adecuados para que puedan abordar los problemas subyacentes.

Patrones comunes de uso no recomendados:

- Las alarmas están configuradas con un umbral excesivamente alto, lo que impide que se envíen notificaciones vitales.
- Las alarmas están configuradas con un umbral demasiado bajo, lo que provoca inacción en las alertas importantes por el ruido que genera el exceso de notificaciones.
- Las alarmas y los umbrales no se actualizan cuando hay cambios de uso.
- En el caso de las alarmas que se abordan mejor con acciones automatizadas, en lugar de generar dichas acciones, se envían notificaciones al personal, lo que provoca un exceso de notificaciones.

Beneficios de establecer esta práctica recomendada: enviar notificaciones y alertas en tiempo real al personal y a los sistemas adecuados permite detectar problemas de forma temprana y responder rápidamente a los incidentes operativos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las cargas de trabajo deben estar equipadas con sistemas de procesamiento y generación de alarmas en tiempo real que permitan mejorar la capacidad de detección de problemas que podrían afectar a la disponibilidad de la aplicación y actúen como desencadenantes de una respuesta automatizada. Las organizaciones pueden llevar a cabo el procesamiento y generar alarmas en tiempo real mediante la creación de alertas con métricas definidas para recibir notificaciones siempre que ocurran eventos importantes o una métrica supere un umbral.

Amazon CloudWatch le permite crear alarmas de métricas y compuestas mediante alarmas de CloudWatch en función del umbral estático, la detección de anomalías y otros criterios. Para obtener más información sobre los tipos de alarmas que puede configurar con CloudWatch, consulte la sección de alarmas de la documentación de CloudWatch.

Puede crear vistas personalizadas de las métricas y alertas de los recursos de AWS para sus equipos mediante los paneles de CloudWatch. Las páginas de inicio personalizables de la consola de CloudWatch le permiten supervisar los recursos a través de una única vista de las diferentes regiones.

Las alarmas pueden llevar a cabo una o más acción, como enviar una notificación a un tema de Amazon SNS, ejecutar una acción de Amazon EC2 o una acción de Amazon EC2 Auto Scaling o crear un OpsItem o incidente en AWS Systems Manager.

Amazon CloudWatch usa Amazon SNS para enviar notificaciones cuando la alarma cambia de estado, lo que permite que los editores (productores) envíen mensajes a los suscriptores (consumidores). Para obtener más información sobre la configuración de las notificaciones de Amazon SNS, consulte Configuring Amazon SNS.

CloudWatch envía eventos a EventBridge cada vez que se crea, actualiza o elimina una alarma de CloudWatch o cambia su estado. Puede usar EventBridge con estos eventos para crear reglas que lleven a cabo acciones, como avisarle cada vez que cambie el estado de una alarma o que activen eventos en la cuenta de forma automática mediante la automatización de Systems Manager.

Manténgase informado con AWS Health. AWS Health es la fuente autorizada de información sobre el estado de los recursos de Nube de AWS. Use AWS Health para recibir notificaciones de cualquier evento de servicio confirmado, de modo que pueda tomar rápidamente medidas para mitigar cualquier impacto. Cree notificaciones de eventos de AWS Health adecuados para su propósito para los canales de correo electrónico y chat a través de AWS User Notifications e intégrelas mediante programación con las herramientas de supervisión y alertas a través de Amazon EventBridge. Si usa AWS Organizations, agregue eventos de AWS Health entre cuentas.

Cuándo debe utilizar EventBridge o Amazon SNS?

Tanto EventBridge como Amazon SNS se pueden utilizar para desarrollar aplicaciones basadas en eventos, así que la elección de uno u otro dependerá de sus necesidades específicas.

Se recomienda Amazon EventBridge si desea crear una aplicación que reaccione a los eventos de sus propias aplicaciones, aplicaciones SaaS y servicios de AWS. EventBridge es el único servicio basado en eventos que se integra directamente con socios de SaaS externos. EventBridge también ingiere automáticamente eventos de más de 200 servicios de AWS sin que los desarrolladores tengan que crear ningún recurso en su cuenta.

EventBridge utiliza una estructura definida basada en JSON para los eventos y le ayuda a crear reglas que se aplican a todo el cuerpo del evento para seleccionar los eventos que se van a reenviar a un destino. Actualmente, EventBridge admite más de 20 servicios de AWS como destino, incluidos AWS Lambda, Amazon SQS, Amazon SNS, Amazon Kinesis Data Streams y Amazon Data Firehose.

Se recomienda usar Amazon SNS con aplicaciones que necesiten una gran distribución (miles o millones de puntos de conexión). Un patrón común que vemos con frecuencia es que los clientes

usan Amazon SNS como destino de la regla para filtrar los eventos que necesitan y distribuirlos a diversos puntos de conexión.

Los mensajes no están estructurados y pueden estar en el formato que desee. Amazon SNS admite el reenvío de mensajes a seis tipos diferentes de destinos, incluidos Lambda, Amazon SQS, puntos de conexión HTTP/S, SMS, notificaciones push y correo electrónico. La latencia habitual de Amazon SNS es inferior a 30 milisegundos. Hay un gran número de servicios de AWS que envían mensajes de Amazon SNS si se configuran para ello (hay más de 30, incluidos Amazon EC2, Amazon S3 y Amazon RDS).

Pasos para la implementación

1. Cree una alarma con las alarmas de Amazon CloudWatch.
    a. Las alarmas de métricas supervisan una única métrica de CloudWatch o una expresión que depende de las métricas de CloudWatch. La alarma inicia una o varias acciones en función del valor de la métrica o de la expresión en comparación con un umbral durante varios intervalos de tiempo. La acción puede ser el envío de una notificación a un tema de Amazon SNS, la ejecución de una acción de Amazon EC2 o una acción de Amazon EC2 Auto Scaling o la creación de un OpsItem o incidente en AWS Systems Manager.
    b. Una alarma compuesta es una expresión de regla que tiene en cuenta las condiciones de otras alarmas que se han creado. La alarma compuesta solo entra en estado de alarma si se cumplen todas las condiciones de la regla. Las alarmas especificadas en la expresión de la regla de una alarma compuesta pueden ser alarmas de métricas y otras alarmas compuestas. Las alarmas compuestas pueden enviar notificaciones de Amazon SNS cuando cambian de estado y pueden crear OpsItems de Systems Manager o incidentes cuando entran en estado de alarma, pero no pueden llevar a cabo acciones de Amazon EC2 ni acciones de escalado automático.
2. Configure las notificaciones de Amazon SNS. Al crear una alarma de CloudWatch, puede incluir un tema de Amazon SNS para enviar una notificación cuando la alarma cambie de estado.
3. Cree reglas en EventBridge que coincidan con las alarmas de CloudWatch especificadas. Cada regla admite varios destinos, incluidas las funciones de Lambda. Por ejemplo, puede definir una alarma que se inicie cuando el espacio disponible en disco se esté agotando, lo que desencadenará una función de Lambda mediante una regla de EventBridge para limpiar el espacio. Para obtener más información sobre los objetivos de EventBridge, consulta los objetivos de EventBridge.

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP02 Definición y cálculo de métricas (agregación)
- REL12-BP01 Uso de manuales de estrategias para investigar los errores

Documentos relacionados:

- Amazon CloudWatch
- CloudWatch Logs insights
- Uso de las alarmas de Amazon CloudWatch
- Uso de paneles de Amazon CloudWatch
- Uso de métricas de Amazon CloudWatch
- Configuración de notificaciones de Amazon SNS
- Uso de la detección de anomalías de CloudWatch
- CloudWatch Logs data protection
- Amazon EventBridge
- Amazon Simple Notification Service

Videos relacionados:

- Videos sobre observabilidad de reinvent
- AWS re:Invent 2022 - Observability best practices at Amazon

Ejemplos relacionados:

- One Observability Workshop
- Amazon EventBridge to AWS Lambda with feedback control by Amazon CloudWatch Alarms

REL06-BP04 Automatización de las respuestas (procesamiento y alarmas en tiempo real)

Use la automatización para actuar cuando se detecte un evento, por ejemplo, para sustituir componentes defectuosos.

El procesamiento automatizado de las alarmas en tiempo real se implementa para que los sistemas puedan tomar medidas correctivas rápidas e intentar evitar fallos o que el servicio se degrade cuando se activan las alarmas. Entre las respuestas automatizadas a las alarmas, se podría incluir la sustitución de los componentes que fallan, el ajuste de la capacidad de computación, el redireccionamiento del tráfico a hosts, zonas de disponibilidad u otras regiones en buen estado y la notificación a los operadores.

Resultado deseado: se identifican las alarmas en tiempo real y se configura el procesamiento automatizado de las alarmas para invocar las acciones apropiadas que se necesitan para mantener los objetivos de nivel de servicio y los acuerdos de nivel de servicio (SLA). La automatización puede abarcar desde actividades de autorreparación de componentes individuales hasta la conmutación por error de todo el sitio.

Patrones comunes de uso no recomendados:

- No tener un inventario o catálogo claros de las principales alarmas en tiempo real.
- No tener respuestas automatizadas en las alarmas críticas (por ejemplo, cuando los recursos de computación están a punto de agotarse, se produce un escalado automático).
- Acciones de respuesta a alarmas contradictorias.
- No tener procedimientos operativos estándar (SOP) que los operadores puedan seguir cuando reciben notificaciones de alerta.
- No supervisar los cambios de configuración, ya que los cambios de configuración no detectados pueden provocar un tiempo de inactividad en las cargas de trabajo.
- No tener una estrategia para deshacer los cambios de configuración no deseados.

Beneficios de establecer esta práctica recomendada: la automatización del procesamiento de alarmas puede mejorar la resiliencia del sistema. El sistema aplica las medidas correctivas automáticamente, lo que reduce las actividades manuales que dan lugar a intervenciones humanas que son más susceptibles a errores. Las operaciones de carga de trabajo cumplen los objetivos de disponibilidad y reducen la interrupción del servicio.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para administrar eficazmente las alertas y automatizar su respuesta, clasifique las alertas en función de su importancia y repercusión, documente los procedimientos de respuesta y planifique las respuestas antes de clasificar las tareas.

Identifique las tareas que requieren medidas específicas (suelen detallarse en los manuales de procedimientos) y examine todos los manuales de procedimientos y manuales de estrategias para determinar qué tareas se pueden automatizar. Si se pueden definir acciones, estas suelen poderse automatizar. Si las acciones no se pueden automatizar, documente los pasos manuales en un SOP y forme a los operadores sobre ellos. Analice continuamente los procesos manuales en busca de oportunidades de automatización en las que pueda establecer y mantener un plan para automatizar las respuestas a las alertas.

Pasos para la implementación

1. Creación de un inventario de alarmas: para obtener una lista de todas las alarmas, puede utilizar la AWS CLI con el comando de Amazon CloudWatch `describe-alarms`. Según el número de alarmas que haya configurado, puede que tenga que utilizar la paginación para recuperar un subconjunto de alarmas para cada llamada o, si lo prefiere, puede utilizar el AWS SDK para obtener las alarmas mediante una llamada a la API.
2. Documentación de todas las acciones de la alarma: actualice un manual de procedimientos con todas las alarmas y sus acciones, independientemente de si son manuales o automatizadas. AWS Systems Managerproporciona manuales de procedimientos predefinidos. Para obtener información acerca de los manuales de procedimientos, consulte Trabajar con manuales de procedimientos. Para obtener información acerca de cómo ver el contenido del manual de procedimiento, consulte View runbook content.
3. Configuración y administración de acciones de la alarma: para cualquiera de las alarmas que requieran una acción, especifique la acción automatizada mediante el SDK de CloudWatch. Por ejemplo, puede cambiar el estado de sus instancias de Amazon EC2 automáticamente en función de una alarma de CloudWatch. Para ello, cree y habilite acciones en una alarma o deshabilite acciones en una alarma.

   También se puede utilizar Amazon EventBridge para responder automáticamente a los eventos del sistema, como los problemas de disponibilidad de las aplicaciones o los cambios en los recursos. Puede crear reglas para indicar qué eventos le resultan de interés, así como qué acciones se van a realizar cuando un evento cumpla una de las reglas. Entre las acciones que se pueden iniciar automáticamente, se incluye invocar una función de AWS Lambda, invocar el `Run Command` de Amazon EC2, transmitir el evento a Amazon Kinesis Data Streams y ver cómo se automatiza Amazon EC2 con EventBridge.
4. Procedimientos operativos estándar (SOP): en función de los componentes que tenga su aplicación, AWS Resilience Hub recomienda varias plantillas de SOP. Puede utilizar estos SOP para documentar todos los procesos que debe seguir un operador en caso de que se genere una

alerta. También puede crear un SOP basado en recomendaciones de Resilience Hub cuando necesite una aplicación Resilience Hub con una política de resiliencia asociada, así como una evaluación de resiliencia histórica en relación con esa aplicación. Las recomendaciones para su SOP provienen de la evaluación de resiliencia.

Resilience Hub funciona con Systems Manager para automatizar los pasos de sus SOP al proporcionar una serie de documentos SSM que puede utilizar como base para esos SOP. Por ejemplo, Resilience Hub puede recomendar un SOP para agregar espacio en disco en un documento de automatización de SSM existente.

5. Acciones automatizadas con Amazon DevOps Guru: puede utilizar Amazon DevOps Guru para supervisar automáticamente los recursos de la aplicación en busca de un comportamiento anómalo y ofrecer recomendaciones específicas para reducir el tiempo de identificación y resolución de problemas. Con DevOps Guru, puede supervisar secuencias de datos operativos casi en tiempo real desde múltiples orígenes, como métricas de Amazon CloudWatch, AWS Config, AWS CloudFormation y AWS X-Ray. También puede utilizar DevOps Guru para crear automáticamente OpsItems en OpsCenter y enviar eventos a EventBridge para una automatización adicional.

Recursos

Prácticas recomendadas relacionadas:

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP02 Definición y cálculo de métricas (agregación)
- REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)
- REL08-BP01 Uso de manuales de procedimientos para actividades estándar como la implementación

Documentos relacionados:

- AWS Systems Manager Automation
- Creating an EventBridge Rule That Triggers on an Event from an AWS Resource
- One Observability Workshop
- Amazon Builders' Library: Instrumentación de los sistemas distribuidos para obtener visibilidad operativa
- What is Amazon DevOps Guru?

- Working with Automation Documents (Playbooks)

Videos relacionados:

- AWS re:Invent 2022 - Observability best practices at Amazon
- AWS re:Invent 2020: Automate anything with AWS Systems Manager
- Introduction to AWS Resilience Hub
- Create Custom Ticket Systems for Amazon DevOps Guru Notifications
- Enable Multi-Account Insight Aggregation with Amazon DevOps Guru

Ejemplos relacionados:

- Amazon CloudWatch and Systems Manager Workshop

REL06-BP05 Análisis de registros

Recopile archivos de registros e historiales de métricas y analícelos para identificar tendencias e información sobre las cargas de trabajo.

Información de registros de Amazon CloudWatch admite un lenguaje de consultas sencillo, pero potente, que se puede utilizar para analizar datos de registro. Registros de Amazon CloudWatch también admite suscripciones que permiten que los datos fluyan sin problemas a Amazon S3, donde puede usarlo o usar Amazon Athena para consultar los datos. También es compatible con consultas en una gran variedad de formatos. Consulte Formatos de SerDes y datos compatibles en la Guía del usuario de Amazon Athena para obtener más información. Para los análisis de conjuntos de archivos de registro enormes, puede ejecutar un clúster de Amazon EMR para ejecutar análisis en la escala de los petabytes.

Hay una serie de herramientas proporcionadas por socios de AWS y terceros que permiten la agregación, procesamiento, almacenamiento y análisis. Entre estas herramientas se incluyen New Relic, Splunk, Loggly, Logstash, CloudHealth y Nagios. Sin embargo, la generación fuera de los registros del sistema y las aplicaciones es exclusiva de cada proveedor de la nube y, a menudo, exclusiva de cada servicio.

Una parte del proceso de supervisión que a menudo se pasa por alto es la administración de datos. Necesita determinar los requisitos de retención para supervisar los datos y, luego, aplicar las políticas del ciclo de vida correspondientemente. Amazon S3 admite la gestión del ciclo de vida en

el nivel de bucket de S3. Esta administración del ciclo de vida se puede aplicar de manera diferente a diferentes rutas en el bucket. Hacia el final del ciclo de vida, puede llevar a cabo la transición de datos a Amazon Glacier para el almacenamiento a largo plazo y vencimiento, una vez alcanzado el final del periodo de retención. La clase de almacenamiento S3 Intelligent-Tiering se ha diseñado para optimizar los costos de almacenamiento mediante el desplazamiento automático de los datos a la capa de acceso de almacenamiento más rentable, sin que afecte al rendimiento ni se produzca sobrecarga operativa.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Información de registros de CloudWatch le permite buscar y analizar de forma interactiva los datos de registro en Registros de Amazon CloudWatch.
  - Analyzing Log Data with CloudWatch Logs Insights
  - Amazon CloudWatch Logs Insights Sample Queries
- Use Registros de Amazon CloudWatch para enviar registros a Amazon S3, donde puede usar Amazon Athena para consultar los datos.
  - ¿Cómo analizo mis registros de acceso al servidor de Amazon S3 mediante Athena?
    - Cree una política de ciclo de vida de S3 para su bucket de registros de acceso al servidor. Configure la política de ciclo de vida para que se eliminen periódicamente los archivos de registros. Esto reduce la cantidad de datos que Athena analiza para cada consulta.
      - ¿Cómo creo una política de ciclo de vida para un bucket de S3?

Recursos

Documentos relacionados:

- Amazon CloudWatch Logs Insights Sample Queries
- Analyzing Log Data with CloudWatch Logs Insights
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- ¿Cómo creo una política de ciclo de vida para un bucket de S3?
- ¿Cómo analizo mis registros de acceso al servidor de Amazon S3 mediante Athena?
- One Observability Workshop
- Amazon Builders' Library: Instrumentación de los sistemas distribuidos para obtener visibilidad operativa

### REL06-BP06 Revisiones frecuentes

Revise frecuentemente cómo está implementada la supervisión de cargas de trabajo y actualícela a medida que su carga de trabajo y su arquitectura evolucionen. Las auditorías periódicas de su monitorización ayudan a reducir el riesgo de que los indicadores de problemas ignoren o se pasen por alto y, además, ayudan a que su carga de trabajo cumpla sus objetivos de disponibilidad.

Un monitoreo eficaz se basa en métricas empresariales clave, que evolucionan a medida que cambian las prioridades empresariales. Su proceso de revisión del monitoreo debe hacer hincapié en los indicadores de nivel de servicio (SLI) e incorporar información de su infraestructura, aplicaciones, clientes y usuarios.

Resultado deseado: cuenta con una estrategia de monitoreo eficaz que se revisa y actualiza periódicamente, así como después de cualquier evento o cambio significativo. Verifica que los indicadores clave del estado de las aplicaciones sigan siendo relevantes a medida que evolucionan su carga de trabajo y sus requisitos empresariales.

Patrones comunes de uso no recomendados:

- Recopila solo métricas predeterminadas.
- Establece una estrategia de monitoreo, pero nunca la revisa.
- No habla sobre el monitoreo cuando se implementan cambios importantes.
- Confía en métricas anticuadas para determinar el estado de la carga de trabajo.
- La carga de trabajo de sus equipos de operaciones es excesiva debido a las alertas de falsos positivos por la obsolescencia de las métricas y los umbrales.
- No tiene capacidad de observación de los componentes de la aplicación que no se monitorean.
- Se centra únicamente en las métricas técnicas de bajo nivel y excluye las métricas empresariales en su supervisión.

Beneficios de establecer esta mejor práctica: si revisa periódicamente su supervisión, puede anticipar los posibles problemas y comprobar que es capaz de detectarlos. También le permite descubrir puntos ciegos que podría haber pasado por alto durante las revisiones anteriores, lo que mejora aún más su capacidad para detectar problemas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Revise las métricas y el alcance del monitoreo durante el proceso de revisión de la preparación operativa (ORR). Realice revisiones periódicas de la preparación operativa siguiendo un cronograma coherente para evaluar si hay alguna brecha entre su carga de trabajo actual y la supervisión que ha configurado. Establezca una cadencia regular en las revisiones de rendimiento operativo y el intercambio de conocimientos para mejorar su capacidad de lograr un mayor rendimiento de sus equipos operativos. Compruebe si los umbrales de alerta existentes siguen siendo adecuados y compruebe si hay situaciones en las que los equipos operativos reciban alertas de falsos positivos o no supervisen los aspectos de la aplicación que deben supervisarse.

El Marco de análisis de la resiliencia proporciona una guía útil que puede ayudarlo a gestionar el proceso. El marco se centra en identificar los posibles modos de fallo y los controles preventivos y correctivos que puede utilizar para mitigar su impacto. Este conocimiento puede ayudar a identificar las métricas y los eventos correctos para monitorear y alertar sobre ellos.

Pasos para la implementación

1. Programe y lleve a cabo revisiones periódicas de los paneles de cargas de trabajo. Puede tener diferentes cadencias para el alcance de la inspección.
2. Inspeccione las tendencias en las métricas. Compare los valores de las métricas con los valores históricos para saber si hay tendencias que puedan indicar que algo necesita ser investigado. Algunos ejemplos son un aumento de la latencia, una reducción de la función empresarial principal y un aumento de las respuestas a los errores.
3. Compruebe si hay valores atípicos y anomalías en sus métricas, que pueden ocultarse mediante promedios o medianas. Examine los valores más altos y más bajos durante el periodo de tiempo e investigue las causas de las observaciones que exceden con creces los límites normales. Durante la eliminación de estas causas, podrá ajustar los límites métricos esperados en función de la mejora de la coherencia del rendimiento de sus cargas de trabajo.
4. Busque cambios bruscos en el comportamiento. Un cambio inmediato en la cantidad o en la dirección de una métrica podría indicar que se ha producido un cambio en la aplicación o factores externos que podrían necesitar la inclusión de métricas adicionales para su seguimiento.
5. Compruebe si la estrategia de supervisión actual sigue siendo relevante para la aplicación. Basándose en un análisis de incidentes anteriores (o en el marco de análisis de la resiliencia), evalúe si hay aspectos adicionales de la aplicación que deban incorporarse al ámbito de la supervisión.

6. Revise sus métricas de monitoreo de usuarios reales (RUM) para determinar si hay brechas en la cobertura de las funcionalidades de la aplicación.
7. Revise su proceso de administración de cambios. Actualice sus procedimientos si es necesario para incluir un paso de análisis de supervisión que deba realizarse antes de aprobar un cambio.
8. Implemente la supervisión y la revisión como parte de sus procesos de revisión de la preparación operativa y corrección de errores.

Recursos

Prácticas recomendadas relacionadas:

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP02 Definición y cálculo de métricas (agregación)
- REL06-BP07 Supervisión del seguimiento de las solicitudes de principio a fin en todo el sistema
- REL12-BP02 Análisis después del incidente
- REL12-BP06 Planificación periódica de días de juego

Documentos relacionados:

- Why you should develop a correction of error (COE)
- Uso de paneles de Amazon CloudWatch
- La creación de paneles para la visibilidad operativa
- Advanced Multi-AZ Resilience Patterns - Gray failures
- Amazon CloudWatch Logs Insights Sample Queries
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- One Observability Workshop
- Amazon Builders' Library: Instrumentación de los sistemas distribuidos para obtener visibilidad operativa
- Uso de paneles de Amazon CloudWatch
- AWS Observability Best Practices
- Resilience analysis framework
- Marco de análisis de resiliencia: observabilidad
- Operational Readiness Review - ORR

REL06-BP07 Supervisión del seguimiento de las solicitudes de principio a fin en todo el sistema

Haga un seguimiento de las solicitudes a medida que se procesan a través de los componentes del servicio para que los equipos de producto puedan analizar y depurar los problemas con mayor facilidad y mejorar el rendimiento.

Resultado deseado: las cargas de trabajo con un seguimiento exhaustivo de todos los componentes son fáciles de depurar, lo que mejora el tiempo medio de recuperación (MTTR) de los errores y la latencia al simplificar la detección de la causa raíz. El rastreo integral reduce el tiempo necesario para descubrir los componentes afectados y analizar detalladamente las causas raíz de los errores o la latencia.

Patrones comunes de uso no recomendados:

- El rastreo se utiliza para algunos componentes, pero no para todos. Por ejemplo, si no se rastreara AWS Lambda, es posible que los equipos no entendieran con claridad la latencia que producen los arranques en frío en una carga de trabajo con picos.
- Los canarios sintéticos o la supervisión de usuarios reales (RUM) no tienen configurado el rastreo. Sin valores controlados ni RUM, la telemetría de interacción con el cliente se omite del análisis del rastreo, lo que da lugar a un perfil de rendimiento incompleto.
- Las cargas de trabajo híbridas incluyen herramientas de rastreo nativas en la nube y de terceros, pero no se han tomado medidas para integrar por completo una única solución de rastreo. En función de la solución de rastreo elegida, se deben utilizar SDK de rastreo nativos en la nube para instrumentar componentes que no sean nativos en la nube o se deben configurar herramientas de terceros para ingerir la telemetría de rastreo nativa en la nube.

Beneficios de establecer esta práctica recomendada: cuando los equipos de desarrollo reciben alertas sobre los problemas, ven una imagen completa de las interacciones entre los componentes del sistema, incluida la correlación componente por componente con el registro, el rendimiento y los errores. Dado que el rastreo facilita la identificación visual de las causas raíz, se dedica menos tiempo a investigar estas causas. Los equipos que conocen bien las interacciones de los componentes toman decisiones mejores y más rápidas a la hora de resolver problemas. Las decisiones, como cuándo invocar una conmutación por error de recuperación de desastres (DR) o cuál es la mejor forma de implementar las estrategias de autorreparación, se pueden mejorar mediante el análisis de los rastros de los sistemas y, en última instancia, puede mejorar la satisfacción del cliente con sus servicios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los equipos que utilizan aplicaciones distribuidas pueden utilizar herramientas de rastreo para establecer un identificador de correlación, recopilar rastros de las solicitudes y crear mapas de servicio de los componentes conectados. Todos los componentes de la aplicación deben incluirse en los rastros de solicitudes, como las puertas de enlace de middleware, los buses de eventos y los clientes del servicio, los componentes de computación y el almacenamiento, incluidos los almacenes de valores clave y las bases de datos. Incluya canarios sintéticos y la supervisión de usuarios reales en su configuración de rastreo integral para medir las interacciones y la latencia de los clientes remotos, de modo que pueda evaluar con precisión el rendimiento de sus sistemas en función de sus acuerdos y objetivos de nivel de servicio.

Puede utilizar AWS X-Ray y los servicios de instrumentación de supervisión de aplicaciones de Amazon CloudWatch para ofrecer una visión completa de las solicitudes a medida que pasan por la aplicación. X-Ray recopila la telemetría de las aplicaciones y le permite visualizarla y filtrarla entre cargas útiles, funciones, rastros, servicios y API, y se puede activar para los componentes del sistema sin código o con poco código. La supervisión de aplicaciones de CloudWatch incluye ServiceLens para integrar sus rastros con métricas, registros y alarmas. La supervisión de aplicaciones de CloudWatch también incluye elementos sintéticos para supervisar los puntos de conexión y las API, así como la supervisión de usuarios reales para instrumentar los clientes de sus aplicaciones web.

Pasos para la implementación

- Use AWS X-Ray en todos los servicios nativos compatibles como Amazon S3, AWS Lambda y Amazon API Gateway. Estos servicios de AWS habilitan X-Ray con conmutadores de configuración que utilizan la infraestructura como código, AWS SDK o la Consola de administración de AWS.
- Aplicaciones de instrumento AWS Distro para OpenTelemetry y X-Ray o agentes recopiladores externos.
- Consulte la guía para desarrolladores de AWS X-Ray para obtener información sobre la implementación de lenguajes de programación específicos. En estas secciones de la documentación, se detalla cómo instrumentar las solicitudes HTTP, las consultas SQL y otros procesos específicos del lenguaje de programación de su aplicación.
- Utilice el rastreo de X-Ray para canarios de Amazon CloudWatch Synthetics y Amazon CloudWatch RUM para analizar la ruta de solicitud desde su cliente de usuario final hasta su infraestructura posterior de AWS.

- Configure métricas y alarmas de CloudWatch en función del estado de los recursos y la telemetría de canarios para que los equipos reciban alertas de los problemas rápidamente y, a continuación, puedan analizar en profundidad los rastros y los mapas de servicio con ServiceLens.
- Habilite la integración de X-Ray para herramientas de rastreo de terceros como Datadog, New Relic o Dynatrace si utiliza herramientas de terceros para su solución de rastreo principal.

Recursos

Prácticas recomendadas relacionadas:

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

Documentos relacionados:

- Qué es AWS X-Ray?
- Amazon CloudWatch: Application Monitoring
- Debugging with Amazon CloudWatch Synthetics and AWS X-Ray
- Amazon Builders' Library: Instrumentación de los sistemas distribuidos para obtener visibilidad operativa
- Integrating AWS X-Ray with other AWS services
- AWS Distro para OpenTelemetry y AWS X-Ray
- Amazon CloudWatch: Using synthetic monitoring
- Amazon CloudWatch: Use CloudWatch RUM
- Set up Amazon CloudWatch synthetics canary and Amazon CloudWatch alarm
- Availability and Beyond: Understanding and Improving the Resilience of Distributed Systems on AWS

Ejemplos relacionados:

- One Observability Workshop

Videos relacionados:

- AWS re:Invent 2022 - How to monitor applications across multiple accounts

- How to Monitor your AWS Applications

Herramientas relacionadas:

- AWS X-Ray
- Amazon CloudWatch
- Amazon Route 53

## REL 7. ¿Cómo diseña su carga de trabajo para adaptarla a los cambios de la demanda?

Una carga de trabajo escalable proporciona elasticidad para agregar o eliminar recursos automáticamente, de modo que se ajusten perfectamente a la demanda actual en cualquier momento dado.

Prácticas recomendadas

- REL07-BP01 Uso de la automatización al obtener o escalar recursos
- REL07-BP02 Obtención de recursos tras detectar un impedimento en una carga de trabajo
- REL07-BP03 Obtención de recursos tras detectar que se necesitan más recursos para una carga de trabajo
- REL07-BP04 Pruebas en su carga de trabajo

### REL07-BP01 Uso de la automatización al obtener o escalar recursos

La fiabilidad en la nube se basa en aspectos clave como la definición programática, el aprovisionamiento y la administración de la infraestructura y los recursos. La automatización lo ayuda a optimizar el aprovisionamiento de recursos, facilitar las implementaciones coherentes y seguras y escalar los recursos en toda su infraestructura.

Resultado deseado: gestione su infraestructura como código (IaC). Define y mantiene su código de infraestructura en los sistemas de control de versiones (VCS). Puede delegar el aprovisionamiento de los recursos de AWS en mecanismos automatizados y aprovechar los servicios gestionados, como los grupos Equilibrador de carga de aplicación (ALB), Equilibrador de carga de red (NLB) y de escalado automático. Los recursos se aprovisionan mediante canalizaciones de integración y entrega continuas (CI/CD) para que los cambios de código inicien automáticamente las actualizaciones de los recursos, incluidas las actualizaciones de las configuraciones de escalado automático.

Patrones comunes de uso no recomendados:

- Los recursos se despliegan manualmente mediante la línea de comandos o la Consola de administración de AWS (también denominado click-ops).
- Combina estrechamente los componentes o los recursos de la aplicación y, como resultado, crea arquitecturas inflexibles.
- Implementa políticas de escalado inflexibles que no se adaptan a los cambiantes requisitos empresariales, a los patrones de tráfico o a los nuevos tipos de recursos.
- La capacidad se estima manualmente para satisfacer la demanda prevista.

Ventajas de establecer esta mejor práctica: la infraestructura como código (IaC) permite definir la infraestructura mediante programación. Esto ayuda a gestionar los cambios en la infraestructura durante el mismo ciclo de vida de desarrollo de software que los cambios en las aplicaciones, lo que promueve la coherencia y la repetibilidad y reduce el riesgo de realizar tareas manuales propensas a errores. Puede optimizar aún más el proceso de aprovisionamiento y actualización de los recursos mediante la implementación de la IaC con canalizaciones de entrega automatizadas. Puede implementar actualizaciones de infraestructura de manera fiable y eficiente sin necesidad de intervención manual. Esta agilidad es especialmente importante a la hora de escalar los recursos para satisfacer las demandas fluctuantes.

Puede lograr un escalado de recursos dinámico y automatizado junto con la IaC y las canalizaciones de entrega. Al monitorear las métricas clave y aplicar políticas de escalado predefinidas, el escalado automático puede aprovisionar o desaprovisionar recursos automáticamente según sea necesario, lo que mejora el rendimiento y la rentabilidad. Esto reduce la posibilidad de que se produzcan errores manuales o demoras en respuesta a los cambios en los requisitos de las aplicaciones o la carga de trabajo.

La combinación de IaC, canalizaciones de entrega automatizadas y escalado automático ayuda a las organizaciones a aprovisionar, actualizar y escalar sus entornos con confianza. Esta automatización es esencial para mantener una infraestructura en la nube con capacidad de respuesta, resiliente y gestionada de manera eficiente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para configurar la automatización con canalizaciones de CI/CD e infraestructura como código (IaC) para su arquitectura de AWS, elija un sistema de control de versiones como Git para almacenar las

plantillas y la configuración de IaC. Estas plantillas se pueden escribir con herramientas como AWS CloudFormation. Para empezar, defina los componentes de su infraestructura (como las VPC de AWS, los grupos de Amazon EC2 Auto Scaling EC2 y las bases de datos de Amazon RDS) en estas plantillas.

A continuación, integre estas plantillas de IaC con una canalización de CI/CD para automatizar el proceso de implementación. AWS CodePipeline le ofrece una solución nativa de AWS perfecta, o puede utilizar otras soluciones de CI/CD de terceros. Cree una canalización que se active cuando se produzcan cambios en su repositorio de control de versiones. Configure la canalización para que incluya etapas que analicen y validen sus plantillas de IaC, implementen la infraestructura en un entorno provisional, ejecuten pruebas automatizadas y, por último, las implementen en la producción. Incorpore los pasos de aprobación cuando sea necesario para mantener el control de los cambios. Esta canalización automatizada no solo acelera la implementación, sino que también facilita la coherencia y la fiabilidad en todos los entornos.

Configure el escalado automático de recursos como instancias de Amazon EC2, tareas de Amazon ECS y réplicas de bases de datos en su IaC para proporcionar escalado horizontal y escalado vertical automáticos según sea necesario. Esta estrategia mejora la disponibilidad y el rendimiento de las aplicaciones y optimiza los costes mediante el ajuste dinámico de los recursos en función de la demanda. Para obtener una lista de los recursos compatibles, consulte Amazon EC2 Auto Scaling y AWS Auto Scaling.

Pasos para la implementación

1. Cree y utilice un repositorio de código fuente para almacenar el código que controla la configuración de su infraestructura. Confirme los cambios en este repositorio para reflejar cualquier cambio en curso que desee realizar.
2. Seleccione una solución de infraestructura como código, por ejemplo, AWS CloudFormation, para mantener su infraestructura actualizada y detectar inconsistencias (desviaciones) con respecto al estado deseado.
3. Integre la plataforma IaC con la canalización de CI/CD para automatizar las implementaciones.
4. Determine y recopile las métricas adecuadas para el escalado automático de los recursos.
5. Configure el escalado automático de los recursos mediante políticas de escalado horizontal y vertical adecuadas para los componentes de la carga de trabajo. Plantéese la opción de utilizar el escalado programado para obtener patrones de uso predecibles.
6. Supervise las implementaciones para detectar errores y regresiones. Implemente mecanismos de reversión en su plataforma de CI/CD para revertir los cambios si es necesario.

Recursos

Documentos relacionados:

- AWS Auto Scaling: How Scaling Plans Work
- AWS Marketplace: productos que pueden usarse con escalado automático
- Administración automática de la capacidad de rendimiento con el escalado automático de DynamoDB
- Usar un equilibrador de carga con un grupo de escalado automático
- ¿Qué es AWS Global Accelerator?
- What Is Amazon EC2 Auto Scaling?
- Qué es AWS Auto Scaling?
- Qué es Amazon CloudFront?
- What is Amazon Route 53?
- Qué es Elastic Load Balancing?
- Qué es un equilibrador de carga de red?
- Qué es un equilibrador de carga de aplicación?
- Integrating Jenkins with AWS CodeBuild and AWS CodeDeploy
- Creating a four stage pipeline with AWS CodePipeline

Videos relacionados:

- Back to Basics: Deploy Your Code to Amazon EC2
- AWS Supports You | Starting Your Infrastructure as Code Solution Using AWS CloudFormation Templates
- Streamline Your Software Release Process Using AWS CodePipeline
- Monitor AWS Resources Using Amazon CloudWatch Dashboards
- Create Cross Account & Cross Region CloudWatch Dashboards | Amazon Web Services

REL07-BP02 Obtención de recursos tras detectar un impedimento en una carga de trabajo

Escale recursos de forma retroactiva cuando sea necesario si la disponibilidad se ve afectada para restaurar la disponibilidad de la carga de trabajo.

Primero debe configurar las comprobaciones de estado y los criterios de dichas comprobaciones para indicar cuándo se ve afectada la disponibilidad por falta de recursos. A continuación, notifique al personal pertinente para que escale manualmente el recurso o inicie la automatización, a fin de que el escalado se lleve a cabo de forma automática.

La escala puede ajustarse manualmente para su carga de trabajo (por ejemplo, se puede cambiar el número de instancias de EC2 en un grupo de escalado automático o se puede modificar el rendimiento de una tabla de DynamoDB mediante la Consola de administración de AWS o la AWS CLI). Sin embargo, la automatización debe usarse siempre que sea posible (consulte Usar la automatización al obtener o escalar recursos).

Resultado deseado: se inician las actividades de escalado (de forma automática o manual) para restablecer la disponibilidad al detectar un error o una experiencia del cliente degradada.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Implemente la observabilidad y la supervisión en todos los componentes de su carga de trabajo para supervisar la experiencia del cliente y detectar errores. Defina los procedimientos, manuales o automatizados, que escalan los recursos necesarios. Para obtener más información, consulte REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores.

Pasos para la implementación

- Defina los procedimientos, manuales o automatizados, que escalan los recursos requeridos.
  - Los procedimientos de escalado dependen de cómo estén diseñados los distintos componentes de la carga de trabajo.
  - Estos procedimientos también varían según la tecnología subyacente que se utilice.
    - Los componentes que utilizan AWS Auto Scaling pueden usar planes de escalado para configurar un conjunto de instrucciones para escalar los recursos. Si trabaja con AWS CloudFormation o agrega etiquetas a recursos de AWS, puede configurar planes de escalamiento para diferentes conjuntos de recursos por aplicación. Auto Scaling proporciona recomendaciones de estrategias de escalado personalizadas según cada recurso. Tras crear el plan de escalado, Auto Scaling combina el escalado dinámico y los métodos de escalado predictivos para ayudarle en su estrategia de escalado. Para obtener más información, consulte Cómo funcionan los planes de escalado.
    - Amazon EC2 Auto Scaling verifica que tenga el número correcto de instancias de Amazon EC2 disponibles para gestionar la carga de la aplicación. Crea colecciones de instancias EC2,

denominadas grupo de escalado automático. Puede especificar el número mínimo y máximo de instancias en cada grupo de Auto Scaling y Amazon EC2 Auto Scaling garantiza que su grupo nunca tenga un tamaño por encima o por debajo de este límite. Para obtener más información, consulte ¿Qué es Amazon EC2 Auto Scaling?

- El escalado automático de Amazon DynamoDB usa el servicio Auto Scaling de aplicaciones de para ajustar de manera dinámica y automática la capacidad de rendimiento aprovisionada en respuesta a los patrones de tráfico reales. Esto permite a una tabla o índice secundario global incrementar su capacidad de lectura y escritura aprovisionada para abastecer incrementos repentinos del tráfico sin limitaciones. Para obtener más información, consulte Administración automática de la capacidad de rendimiento con el escalado automático de DynamoDB.

Recursos

Prácticas recomendadas relacionadas:

- REL07-BP01 Uso de la automatización al obtener o escalar recursos
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

Documentos relacionados:

- AWS Auto Scaling: How Scaling Plans Work
- Administración automática de la capacidad de rendimiento con el escalado automático de DynamoDB
- What Is Amazon EC2 Auto Scaling?

REL07-BP03 Obtención de recursos tras detectar que se necesitan más recursos para una carga de trabajo

Una de las características más valiosas de la computación en la nube es la capacidad de aprovisionar recursos de forma dinámica.

En los entornos informáticos tradicionales en las instalaciones, debe identificar y aprovisionar suficiente capacidad con antelación para atender los picos de demanda. Esto supone un problema porque resulta caro y supone un riesgo para la disponibilidad si se subestiman las necesidades de capacidad máxima de la carga de trabajo.

En la nube, no tiene que realizar esta acción. En su lugar, puede aprovisionar la capacidad de cómputo, la base de datos y otros recursos según sea necesario para satisfacer la demanda actual y prevista. Las soluciones automatizadas, como Amazon EC2 Auto Scaling y el escalamiento automático de la aplicación, pueden poner los recursos en línea automáticamente en función de las métricas que especifique. Esto puede facilitar el proceso de escalado y hacerlo más predecible, así como mejorar la fiabilidad de su carga de trabajo al garantizar que haya recursos disponibles suficientes en todo momento.

Resultado deseado: configura el escalado automático de los recursos informáticos y otros recursos para satisfacer la demanda. Sus políticas de escalado ofrecen suficiente margen de maniobra para poder atender ráfagas de tráfico y, al mismo tiempo, poner en funcionamiento recursos adicionales.

Patrones comunes de uso no recomendados:

- Aprovisiona un número fijo de recursos escalables.
- Elige una métrica de escalado que no se correlaciona con la demanda real.
- No proporciona suficiente margen de maniobra en sus planes de escalado para adaptarse a las ráfagas de demanda.
- Sus políticas de escalado añaden capacidad demasiado tarde, lo que provoca el agotamiento de la capacidad y la degradación del servicio, a la vez que se ponen en línea recursos adicionales.
- No configura correctamente los recuentos de recursos mínimos y máximos, lo que provoca errores de escalado.

Ventajas de establecer esta práctica recomendada: disponer de recursos suficientes para satisfacer la demanda actual es fundamental para ofrecer una alta disponibilidad de la carga de trabajo y cumplir los objetivos de nivel de servicio (SLO) definidos. El escalado automático le permite proporcionar la cantidad adecuada de recursos informáticos, de bases de datos y de otro tipo que su carga de trabajo necesita para satisfacer la demanda actual y prevista. No es necesario determinar las necesidades de capacidad máxima ni asignar recursos de forma estática para atenderlas. En cambio, a medida que aumenta la demanda, puede asignar más recursos para adaptarla y, una vez que la demanda disminuya, puede desactivar los recursos para reducir los costes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

En primer lugar, determine si el componente de la carga de trabajo es adecuado para el escalado automático. Estos componentes se denominan escalables horizontalmente porque proporcionan

los mismos recursos y se comportan de forma idéntica. Entre los ejemplos de componentes escalables horizontalmente se incluyen las instancias de EC2 que están configuradas de forma similar, las tareas de Amazon Elastic Container Service (ECS) y los pods que se ejecutan en Amazon Elastic Kubernetes Service (EKS). Estos recursos de cómputo suelen estar ubicados detrás de un equilibrador de carga y se denominan réplicas.

Otros recursos replicados pueden incluir réplicas de lectura de bases de datos, tablas de Amazon DynamoDB y clústeres de Amazon ElastiCache (Redis OSS). Para obtener una lista completa de los recursos admitidos, consulte los servicios de AWS que puede utilizar con el escalamiento automático de la aplicación.

En el caso de las arquitecturas basadas en contenedores, es posible que necesite escalar de dos maneras diferentes. En primer lugar, tal vez necesite escalar los contenedores que proporcionan servicios escalables horizontalmente. En segundo lugar, puede que necesite escalar los recursos informáticos para dejar espacio para nuevos contenedores. Existen diferentes mecanismos de escalado automático para cada capa. Para escalar las tareas de ECS, puede usar el escalamiento automático de la aplicación. Para escalar los pods de Kubernetes, puede usar el escalador automático de pods horizontal (HPA) o el escalado automático controlado por eventos de Kubernetes (KEDA). Para escalar los recursos de computación, puede usar los proveedores de capacidad para ECS o, para Kubernetes, puede usar Karpenter o el Cluster Autoscaler.

A continuación, seleccione cómo realizará el escalado automático. Hay tres opciones principales: el escalado basado en métricas, el escalado programado y el escalado predictivo.

Escalado basado en métricas

El escalado basado en métricas aprovisiona los recursos en función del valor de una o más métricas de escalado. Una métrica de escalado es aquella que corresponde a la demanda de su carga de trabajo. Una buena forma de determinar las métricas de escalado adecuadas es realizar pruebas de carga en un entorno que no sea de producción. Durante las pruebas de carga, mantenga fija la cantidad de recursos escalables y aumente lentamente la demanda (por ejemplo, el rendimiento, la simultaneidad o los usuarios simulados). A continuación, busque métricas que aumenten (o disminuyan) a medida que aumente la demanda y, a la inversa, que disminuyan (o aumenten) a medida que la demanda disminuya. Las métricas de escalado típicas incluyen el uso de la CPU, la profundidad de las colas de trabajo (como una cola de Amazon SQS), el número de usuarios activos y el rendimiento de la red.

> **Note**
>
> AWS ha observado que, en la mayoría de las aplicaciones, la utilización de la memoria aumenta a medida que la aplicación se calienta y, después, alcanza un valor estable. Cuando la demanda disminuye, la utilización de la memoria suele mantenerse elevada en lugar de disminuir en consecuencia. Como la utilización de la memoria no se corresponde con la demanda en ambas direcciones (es decir, aumentando o disminuyendo según la demanda), piénseselo bien antes de seleccionar esta métrica para el escalado automático.

El escalado basado en métricas es una operación latente. Las métricas de utilización pueden tardar varios minutos en propagarse a los mecanismos de escalado automático y, por lo general, estos mecanismos esperan una señal clara de aumento de la demanda antes de reaccionar. Luego, a medida que el escalador automático crea nuevos recursos, es posible que tarde más tiempo en funcionar por completo. Por este motivo, es importante no fijar los objetivos de las métricas de escalado demasiado cerca de la plena utilización (por ejemplo, un 90 % de utilización de la CPU). Si lo hace, se corre el riesgo de agotar la capacidad de recursos existente antes de que se pueda poner en línea capacidad adicional. Los objetivos típicos de utilización de los recursos pueden oscilar entre el 50 % y el 70 % para lograr una disponibilidad óptima, en función de los patrones de demanda y del tiempo necesario para aprovisionar recursos adicionales.

Escalado programado

El escalado programado aprovisiona o elimina recursos según el calendario o la hora del día. Se usa con frecuencia para cargas de trabajo que tienen una demanda predecible, como los picos de uso durante el horario laboral de lunes a viernes o en eventos de ventas. Tanto Amazon EC2 Auto Scaling como el escalamiento automático de la aplicación admiten el escalado programado. El escalador cron de KEDA admite el escalado programado de los pods de Kubernetes.

Escalado predictivo

El escalado predictivo utiliza machine learning para escalar automáticamente los recursos en función de la demanda prevista. Analiza el valor histórico de la métrica de utilización que le proporcione y predice continuamente su valor futuro. A continuación, el valor previsto se utiliza para escalar el recurso hacia arriba o hacia abajo. Amazon EC2 Auto Scaling puede realizar un escalado predictivo.

Pasos para la implementación

1. En primer lugar, determine si el componente de la carga de trabajo es adecuado para el escalado automático.
2. Determine qué tipo de mecanismo de escalado es el más adecuado para la carga de trabajo: escalado basado en métricas, escalado programado o escalado predictivo.
3. Seleccione el mecanismo de escalado automático adecuado para el componente. Para instancias Amazon EC2, utilice Amazon EC2 Auto Scaling. Para otros servicios de AWS, utilice el escalamiento automático de la aplicación. Para los pods de Kubernetes (como los que se ejecutan en un clúster de Amazon EKS), puede usar el escalador automático de pods horizontal (HPA) o el escalado automático controlado por eventos de Kubernetes (KEDA). Para los nodos de Kubernetes o EKS, puede usar Karpenter y Cluster Auto Scaler (CAS).
4. Para el escalado con métricas o programado, realice pruebas de carga para determinar las métricas de escalado y los valores objetivo adecuados para su carga de trabajo. Para el escalado programado, determine la cantidad de recursos necesarios en las fechas y horas que seleccione. Determine el número máximo de recursos necesarios para atender los picos de tráfico previstos.
5. Configure el escalador automático en función de la información recopilada anteriormente. Consulte la documentación del servicio de escalado automático para obtener información más detallada. Compruebe que los límites de escalado máximo y mínimo estén configurados correctamente.
6. Compruebe que la configuración de escalado funciona según lo esperado. Realice las pruebas de carga en un entorno que no sea de producción, observe cómo reacciona el sistema y haga los ajustes pertinentes. Cuando habilite el escalado automático en producción, configure las alarmas adecuadas para que le notifiquen cualquier comportamiento inesperado.

Recursos

Documentos relacionados:

- What Is Amazon EC2 Auto Scaling?
- Orientación normativa de AWS: aplicaciones de prueba de carga
- AWS Marketplace: productos que pueden usarse con escalado automático
- Administración automática de la capacidad de rendimiento con el escalado automático de DynamoDB
- Predictive Scaling for EC2, Powered by Machine Learning
- Scheduled Scaling for Amazon EC2 Auto Scaling

- Telling Stories About Little's Law

REL07-BP04 Pruebas en su carga de trabajo

Adopte una metodología de prueba de carga para medir si la actividad de escalado satisface los requisitos de la carga de trabajo.

Es importante llevar a cabo pruebas de carga sostenidas. Las pruebas de carga deben descubrir el punto de ruptura y probar el rendimiento de su carga de trabajo. AWS facilita la creación de entornos de prueba temporales que modelan la escala de su carga de trabajo de producción. En la nube, puede crear un entorno de prueba a escala de producción, completar sus pruebas y desmantelar los recursos. Debido a que solo paga por el entorno de prueba cuando se ejecuta, puede simular su entorno real por una fracción del costo de las pruebas en las instalaciones.

Las pruebas de carga en producción también deben considerarse como parte de los días de juegos en los que se estresa el sistema de producción, durante las horas de menor uso por parte de los clientes, con todo el personal a mano para interpretar los resultados y abordar cualquier problema que surja.

Patrones comunes de uso no recomendados:

- Hacer pruebas de carga en implementaciones que no tienen la misma configuración que su producción.
- Hacer pruebas de carga solo en elementos individuales de su carga de trabajo y no en toda la carga.
- Hacer pruebas de carga con un subconjunto de solicitudes y no con un conjunto representativo de solicitudes reales.
- Hacer pruebas de carga con un pequeño factor de seguridad por encima de la carga prevista.

Beneficios de establecer esta práctica recomendada: sabrá qué componentes de su arquitectura presentan errores bajo carga y podrá identificar qué métricas se deben vigilar para indicar que se está acercando a esa carga a tiempo para solucionar el problema, con lo que se evitará el impacto de ese error.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

- Haga pruebas de carga para identificar qué aspecto de su carga de trabajo indica que debe agregar o eliminar capacidad. Las pruebas de carga deben tener un tráfico representativo similar al que se recibe en producción. Aumente la carga mientras vigila las métricas que ha instrumentado para determinar qué métrica indica cuándo debe agregar o eliminar recursos.
  - Pruebas de carga distribuida en AWS: simular miles de usuarios conectados
    - Identifique la combinación de solicitudes. Es posible que tenga una combinación variada de solicitudes, por lo que deberá tener en cuenta diversos periodos de tiempo a la hora de identificar la combinación de tráfico.
    - Implemente un controlador de carga. Puede utilizar software de código personalizado, de código abierto o comercial para implementar un controlador de carga.
    - Haga la prueba de carga inicialmente con una capacidad pequeña. Ve algunos efectos inmediatos al pasar la carga a una capacidad menor, posiblemente tan pequeña como una instancia o un contenedor.
    - Lleve a cabo una prueba de carga con una capacidad mayor. Los efectos serán diferentes en una carga distribuida, por lo que debe efectuar las pruebas en un entorno lo más parecido al del producto.

Recursos

Documentos relacionados:

- Pruebas de carga distribuida en AWS: simular miles de usuarios conectados
- Aplicaciones de pruebas de carga

Videos relacionados:

- AWS Summit ANZ 2023: Accelerate with confidence through AWS Distributed Load Testing

## REL 8. ¿Cómo implementa el cambio?

Los cambios controlados son necesarios para implementar nuevas funcionalidades y comprobar que las cargas de trabajo y el entorno operativo ejecuten software conocido y que puedan recibir revisiones o reemplazos de manera predecible. Si estos cambios no se controlan, es difícil predecir su efecto o abordar los problemas que surjan a causa de ellos.

Prácticas recomendadas

- REL08-BP01 Uso de manuales de procedimientos para actividades estándar como la implementación
- REL08-BP02 Integración de las pruebas funcionales como parte de la implementación
- REL08-BP03 Integración de las pruebas de resiliencia como parte de la implementación
- REL08-BP04 Implementación mediante una infraestructura inmutable
- REL08-BP05 Implementación de cambios con automatización

REL08-BP01 Uso de manuales de procedimientos para actividades estándar como la implementación

Los manuales de procedimientos son procedimientos predefinidos para obtener resultados concretos. Use manuales de procedimientos para llevar a cabo actividades estándar manuales o automáticas. Algunos ejemplos incluyen implementar una carga de trabajo, aplicarle un parche a dicha carga de trabajo o hacer modificaciones de DNS.

Por ejemplo, establezca procesos para garantizar la seguridad de la reversión durante las implementaciones. Tener la garantía de poder dar marcha atrás en una implementación sin disrupciones para sus clientes es esencial a la hora de hacer que un servicio sea fiable.

Para los procedimientos del manual de procedimientos, comience con un proceso manual válido y efectivo, impleméntelo en código e invóquelo para que se ejecute automáticamente cuando corresponda.

Incluso en el caso de cargas de trabajo sofisticadas y altamente automatizadas, los manuales de procedimientos siguen siendo útiles para ejecutar los días de juego o para cumplir con los rigurosos requisitos de informes y auditoría.

Tenga en cuenta que los manuales de estrategias se usan en respuesta a incidentes específicos y los manuales de procedimientos se usan para conseguir resultados determinados. A menudo, los manuales de procedimientos son para actividades rutinarias, mientras que los manuales de estrategias se utilizan para responder a eventos no rutinarios.

Patrones comunes de uso no recomendados:

- Hacer cambios imprevistos en la configuración en producción.
- Omitir pasos del plan para que la implementación sea más rápida, lo que da lugar a una implementación errónea.

- Hacer cambios sin probar la revocación del cambio.

Beneficios de establecer esta práctica recomendada: la planificación eficaz de los cambios aumenta su capacidad de efectuar correctamente el cambio, ya que sabrá qué sistemas se verán afectados. Validar el cambio en los entornos de prueba aumenta la confianza.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

- Para obtener respuestas sistematizadas e inmediatas a eventos conocidos, documente los procedimientos en manuales de procedimientos.
- Use el principio de infraestructura como código para definir la infraestructura. Si usa AWS CloudFormation (o un tercero de confianza) para definir su infraestructura, puede utilizar software de control de versiones para crear versiones y hacer un seguimiento de los cambios.
  - Use AWS CloudFormation (o un proveedor tercero de confianza) para definir su infraestructura.
    - Qué es AWS CloudFormation?
  - Cree plantillas singulares y desacopladas mediante buenos principios de diseño de software.
    - Determine los permisos, las plantillas y las partes responsables de su implementación.
      - Control de acceso con AWS Identity and Access Management
    - Utilice un sistema de administración de código fuente alojado basado en una tecnología popular como Git para almacenar el código fuente y la configuración de infraestructura como código (IaC).

Recursos

Documentos relacionados:

- Socio de APN: socios que pueden ayudarle a crear soluciones de implementación automatizadas
- AWS Marketplace: productos que pueden usarse para automatizar sus implementaciones
- Qué es AWS CloudFormation?

Ejemplos relacionados:

- Automating operations with Playbooks and Runbooks

REL08-BP02 Integración de las pruebas funcionales como parte de la implementación

Utilice técnicas como las pruebas unitarias y las pruebas de integración que validen la funcionalidad requerida.

Las pruebas unitarias son un proceso en el que se prueba la unidad funcional de código más pequeña para validar su comportamiento. Las pruebas de integración sirven para validar que cada característica de la aplicación funcione de acuerdo con los requisitos del software. Mientras que las pruebas unitarias se centran en probar parte de una aplicación de forma aislada, las pruebas de integración tienen en cuenta los efectos secundarios (por ejemplo, el efecto de la modificación de los datos mediante una operación de mutación). En cualquier caso, las pruebas se deben integrar en una canalización de implementación y, si no se cumplen los criterios de éxito, la canalización se detiene o se revierte. Estas pruebas se llevan a cabo en un entorno de preproducción, que se lleva a cabo antes de la producción en la canalización.

Los mejores resultados se obtienen cuando estas pruebas se ejecutan automáticamente como parte de las acciones de creación e implementación. Por ejemplo, con AWS CodePipeline, los desarrolladores confirman los cambios en un repositorio de origen, donde CodePipeline detecta automáticamente los cambios. Se crea la aplicación y se ejecutan las pruebas unitarias. Una vez completadas las pruebas unitarias, el código compilado se implementa en servidores de ensayo para su comprobación. En el servidor de ensayo, CodePipeline ejecuta pruebas adicionales, como pruebas de integración o carga. Una vez completadas correctamente estas pruebas, CodePipeline implementa el código probado y aprobado en instancias de producción.

Resultado deseado: utiliza la automatización para realizar pruebas unitarias y de integración a fin de validar que el código se comporta según lo previsto. Estas pruebas se integran en el proceso de implementación y, en caso de fallo de una prueba, se interrumpe la implementación.

Patrones comunes de uso no recomendados:

- Durante el proceso de implementación, se ignoran u omiten los errores y planes de las pruebas para acelerar el cronograma de implementación.
- Las pruebas se hacen manualmente fuera de la canalización de implementación.
- Omite los pasos de prueba de la automatización mediante flujos de trabajo de emergencia manuales.
- Las pruebas automatizadas se ejecutan en un entorno que no se parece mucho al entorno de producción.

- Crea un conjunto de pruebas que no es lo suficientemente flexible y que es difícil de mantener, actualizar o escalar a medida que la aplicación evoluciona.

Beneficios de establecer esta práctica recomendada: las pruebas automatizadas durante el proceso de implementación detectan los problemas de forma temprana, lo que reduce el riesgo de que se lance a producción con errores o un comportamiento inesperado. Las pruebas unitarias validan que el código se comporte como se desea y que se cumplan los contratos de la API. Las pruebas de integración validan que el sistema funcione de acuerdo con los requisitos especificados. Estos tipos de pruebas verifican de manera sistemática el orden de funcionamiento previsto de los componentes, como las interfaces de usuario, las API, las bases de datos y el código fuente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Adopte una estrategia de desarrollo basada en pruebas (TDD) para desarrollar software que incluya casos de prueba para especificar y validar el código. Para empezar, cree casos de prueba para cada función. Si la prueba falla, escriba un código nuevo para superarla. Este enfoque ayuda a validar el resultado esperado de cada función. Ejecute pruebas unitarias y valide que se aprueben antes de enviar el código a un repositorio de código fuente.

Implemente pruebas unitarias o de integración como parte de las etapas de compilación, prueba e implementación de la canalización de CI/CD. Automatice las pruebas e inicie las pruebas automáticamente cada vez que haya una nueva versión de la aplicación lista para su implementación. Si no se satisfacen los criterios de éxito, la canalización se detiene o se revierte.

Si la aplicación es una aplicación web o móvil, realice pruebas de integración automatizadas en varios navegadores de escritorio o dispositivos reales. Este enfoque es particularmente útil para validar la compatibilidad y la funcionalidad de las aplicaciones móviles en una amplia gama de dispositivos.

Pasos para la implementación

1. Escriba pruebas unitarias antes de desarrollar código funcional (desarrollo basado en pruebas o TDD). Establezca pautas de código para que escribir y ejecutar pruebas unitarias sea un requisito de codificación no funcional.
2. Cree un conjunto de pruebas de integración automatizadas que cubran las funcionalidades comprobables identificadas. Estas pruebas deben simular las interacciones de los usuarios y validar los resultados esperados.

3. Cree el entorno de pruebas necesario para ejecutar las pruebas de integración. Esto puede incluir entornos de preparación o preproducción que imiten fielmente el entorno de producción.
4. Configuración de las etapas de origen, compilación, prueba e implementación mediante la consola de AWS CodePipeline o la AWS Command Line Interface (CLI).
5. Implemente la aplicación una vez que se haya creado y probado el código. AWS CodeDeploy puede implementarlo en sus entornos de puesta en escena (pruebas) y producción. Estos entornos pueden incluir instancias de Amazon EC2, funciones de AWS Lambda o servidores en las instalaciones. Se debe usar el mismo mecanismo de implementación para implementar la aplicación en todos los entornos.
6. Monitoree el progreso de la canalización y del estado de cada etapa. Utilice controles de calidad para bloquear la canalización en función del estado de las pruebas. También puede recibir notificaciones cuando se produzca un fallo en una etapa de canalización o cuando se haya completado la canalización.
7. Supervise continuamente los resultados de las pruebas y busque patrones, regresiones o áreas que requieran más atención. Utilice esta información para mejorar el conjunto de pruebas, identificar las áreas de la aplicación que necesitan pruebas más sólidas y optimizar el proceso de implementación.

Recursos

Prácticas recomendadas relacionadas:

- REL07-BP04 Pruebas en su carga de trabajo
- REL08-BP03 Integración de las pruebas de resiliencia como parte de la implementación
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Documentos relacionados:

- Guía prescriptiva de AWS: automatización de pruebas
- Integración y entrega continuas
- Indicators for functional testing
- Monitorear canalizaciones
- Use AWS CodePipeline with AWS CodeBuild to test code and run builds
- AWS Device Farm

REL08-BP03 Integración de las pruebas de resiliencia como parte de la implementación

Para integrar las pruebas de resiliencia, cree fallos en el sistema de forma intencionada para medir su capacidad en caso de situaciones disruptivas. Las pruebas de resiliencia son diferentes de las pruebas unitarias y funcionales que suelen estar integradas en los ciclos de implementación, ya que se centran en la identificación de fallos imprevistos en el sistema. Aunque es seguro comenzar con la integración de pruebas de resiliencia en la fase de preproducción, fíjese el objetivo de implementar estas pruebas en producción como parte de sus días de juegos.

Resultado deseado: las pruebas de resiliencia ayudan a generar confianza en la capacidad del sistema para resistir la degradación en la producción. Los experimentos identifican los puntos débiles que podrían provocar fallos, lo que le ayuda a mejorar su sistema para mitigar de manera automática y eficiente los fallos y la degradación.

Patrones comunes de uso no recomendados:

- Falta de observabilidad y supervisión en los procesos de implementación
- Confianza en las personas para resolver fallos del sistema
- Mecanismos de análisis de mala calidad
- Centrarse en los problemas conocidos de un sistema y falta de experimentación para identificar cualquier elemento desconocido
- Identificación de fallos, pero sin ninguna resolución
- Falta de documentación de los resultados y manuales de procedimientos

Beneficios de establecer prácticas recomendadas: las pruebas de resiliencia integradas en las implementaciones ayudan a identificar problemas desconocidos en el sistema que, de otro modo, pasarían desapercibidos y que pueden provocar tiempos de inactividad en la producción. La identificación de estos elementos desconocidos en un sistema le ayuda a documentar los resultados, integrar las pruebas en su proceso de CI/CD y crear manuales de procedimientos, lo que simplifica la mitigación mediante mecanismos eficientes y repetibles.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los formularios de pruebas de resiliencia más comunes que se pueden integrar en las implementaciones de su sistema son la recuperación de desastres y la ingeniería del caos.

- Incluya actualizaciones en sus planes de recuperación de desastres y procedimientos operativos estándar (SOP) en cualquier implementación importante.
- Integre las pruebas de fiabilidad en sus canalizaciones de implementación automatizadas. Estos servicios, como AWS Resilience Hub, se pueden integrar en su canalización de CI/CD para establecer evaluaciones de resiliencia continuas que se evalúen automáticamente como parte de cada implementación.
- Defina sus aplicaciones en AWS Resilience Hub. Las evaluaciones de resiliencia generan fragmentos de código que le ayudan a crear procedimientos de recuperación como documentos de AWS Systems Manager para sus aplicaciones y proporcionan una lista de monitores y alarmas recomendados de Amazon CloudWatch.
- Una vez actualizados los planes de recuperación de desastres y los procedimientos operativos estándar, lleve a cabo las pruebas de recuperación de desastres para verificar que sean efectivos. Las pruebas de recuperación de desastres le ayudan a determinar si puede restaurar el sistema después de un evento y recuperar el funcionamiento normal. Puede simular varias estrategias de recuperación de desastres e identificar si su planificación es suficiente para cumplir sus requisitos de tiempo de actividad. Las estrategias comunes de recuperación de desastres incluyen copias de seguridad y restauración, el enfoque de luz piloto, la espera en frío, la espera semiactiva, la espera activa y la estrategia activa-activa, y todas difieren en costo y complejidad. Antes de llevar a cabo las pruebas de recuperación de desastres, le recomendamos que defina su objetivo de tiempo de recuperación (RTO) y objetivo de punto de recuperación (RPO) para simplificar la elección de la estrategia que desee simular. AWS ofrece herramientas de recuperación de desastres, como AWS Elastic Disaster Recovery, que le ayudarán a empezar con la planificación y las pruebas.
- Los experimentos de ingeniería del caos introducen disrupciones en el sistema, como cortes de la red y fallos del servicio. Al ejecutar simulaciones con fallos controlados, puede descubrir las vulnerabilidades de su sistema y, al mismo tiempo, contener la repercusión de los fallos inyectados. Al igual que las demás estrategias, ejecute simulaciones de fallas controladas en entornos que no sean de producción utilizando servicios como AWS Fault Injection Service para ganar confianza antes de implementarlos en producción.

Recursos

Documentos relacionados:

- Experiment with failure using resilience testing to build recovery preparedness
- Continually assessing application resilience with AWS Resilience Hub and AWS CodePipeline
- Disaster recovery (DR) architecture on AWS, part 1: Strategies for recovery in the cloud

- Verify the resilience of your workloads using Chaos Engineering
- Principios de la ingeniería del caos
- Chaos Engineering Workshop

Videos relacionados:

- AWS re:Invent 2020: Testing Resilience using Chaos Engineering
- Improve Application Resilience with AWS Fault Injection Service
- Prepare & Protect Your Applications From Disruption With AWS Resilience Hub

REL08-BP04 Implementación mediante una infraestructura inmutable

La infraestructura inmutable es un modelo que exige que no haya actualizaciones, parches de seguridad ni cambios de configuración en las cargas de trabajo de producción. Cuando es necesario aplicar un cambio, la arquitectura se integra en una nueva infraestructura y se implementa en producción.

Utilice una estrategia de implementación de infraestructura inmutable para aumentar la fiabilidad, la coherencia y la reproducibilidad de las implementaciones de sus cargas de trabajo.

Resultado deseado: con una infraestructura inmutable, no se permiten modificaciones in situ para ejecutar los recursos de infraestructura dentro de una carga de trabajo. En su lugar, cuando es necesario hacer un cambio, se implementa en paralelo un nuevo conjunto de recursos de infraestructura actualizados que contienen todos los cambios que es necesario aplicar en los recursos existentes. Esta implementación se valida automáticamente y, si se efectúa correctamente, el tráfico se desplaza gradualmente al nuevo conjunto de recursos.

Esta estrategia de implementación se aplica a las actualizaciones de software, las revisiones de seguridad, los cambios de infraestructura, las actualizaciones de la configuración y las actualizaciones de las aplicaciones, entre otros.

Patrones comunes de uso no recomendados:

- Implementar cambios in situ en los recursos de infraestructura en ejecución.

Beneficios de establecer esta práctica recomendada:

- Mayor coherencia entre los entornos: dado que no hay diferencias en los recursos de infraestructura entre los entornos, la coherencia aumenta y las pruebas se simplifican.
- Reducción de las desviaciones de la configuración: al sustituir los recursos de infraestructura por una configuración conocida y controlada por versiones, la infraestructura se define en un estado conocido, probado y fiable, lo que evita desviaciones de la configuración.
- Implementaciones atómicas fiables: las implementaciones se completan correctamente o no cambia nada, lo que aumenta la coherencia y la fiabilidad del proceso de implementación.
- Implementaciones simplificadas: las implementaciones se simplifican porque no tienen que admitir actualizaciones. Las actualizaciones son simplemente nuevas implementaciones.
- Implementaciones más seguras con procesos de restauración y recuperación rápidos: las implementaciones son más seguras porque no se modifica la versión operativa anterior. Puede restaurarla si se detecta algún error.
- Mejora de la posición de seguridad: al no permitir cambios en la infraestructura, se pueden deshabilitar los mecanismos de acceso remoto (como SSH). Esto reduce el vector de ataque y mejora la posición de seguridad de su organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Automation

A la hora de definir una estrategia de implementación de infraestructura inmutable, se recomienda utilizar la automatización en la medida de lo posible para aumentar la reproducibilidad y minimizar la posibilidad de que se cometan errores humanos. Para obtener más información, consulte REL08-BP05 Implementación de cambios con automatización y Automatización de implementaciones seguras y sin intervención.

Con la infraestructura como código (IaC), los pasos de aprovisionamiento, orquestación e implementación de la infraestructura se definen de forma programática, descriptiva y declarativa y se almacenan en un sistema de control de código fuente. El uso de la infraestructura como código simplifica la automatización de la implementación de la infraestructura y ayuda a lograr la inmutabilidad de la infraestructura.

Patrones de implementación

Cuando es necesario hacer un cambio en la carga de trabajo, la estrategia inmutable de implementación de la infraestructura requiere la implementación de un nuevo conjunto de recursos

de infraestructura que incluya todos los cambios necesarios. Es importante que este nuevo conjunto de recursos siga un patrón de implementación que minimice la repercusión en los usuarios. Hay dos estrategias principales para esta implementación:

Implementación canario: práctica que consiste en dirigir a un número reducido de clientes a la nueva versión, que normalmente se ejecuta en una instancia de servicio único (canario). A continuación, puede analizar en profundidad los errores o los cambios en el comportamiento que se hayan generado. Puede eliminar el tráfico del canario si encuentra problemas críticos y enviar a los usuarios de vuelta a la versión anterior. Si la implementación se lleva a cabo correctamente, puede continuar implementando a la velocidad deseada y, al mismo tiempo, supervisar los cambios para detectar errores, hasta que esté completamente implementada. AWS CodeDeploy se puede configurar con una configuración de implementación que permita una implementación canario.

Implementación azul/verde: similar a la implementación canario, excepto que se implementa en paralelo una flota completa de la aplicación. Alterne sus implementaciones en las dos pilas (azul y verde). Una vez más, puede enviar tráfico a la nueva versión y volver a la versión anterior si observa problemas con la implementación. Por lo general, todo el tráfico se conmuta a la vez, pero también puede utilizar fracciones del tráfico en cada versión para acelerar la adopción de la nueva versión mediante las capacidades de enrutamiento de DNS ponderado de Amazon Route 53. AWS CodeDeploy y AWS Elastic Beanstalk se pueden establecer con una configuración de implementación que permita una implementación azul/verde.

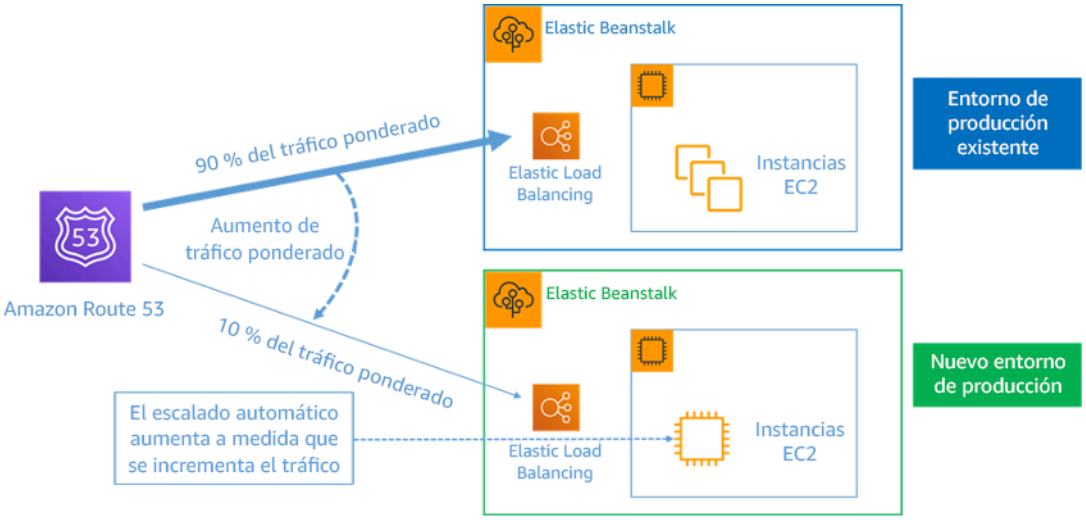



Figura 8: implementación azul/verde con AWS Elastic Beanstalk y Amazon Route 53

Detección de desviaciones

La desviación se define como cualquier cambio que provoque que un recurso de infraestructura tenga un estado o una configuración diferentes a los esperados. Cualquier tipo de cambio de

configuración no administrado va en contra de la noción de infraestructura inmutable y debe detectarse y remediarse para que la infraestructura inmutable se implemente correctamente.

Pasos para la implementación

- No permita que se hagan modificaciones in situ de los recursos de la infraestructura en ejecución.
  - Puede usar AWS Identity and Access Management (IAM) para especificar quién o qué puede acceder a los servicios y recursos de AWS, administrar de forma centralizada los permisos detallados y analizar el acceso para refinar los permisos en AWS.
- Automatice la implementación de los recursos de la infraestructura para aumentar la reproducibilidad y minimizar la posibilidad de que se cometan errores humanos.
  - Como se describe en el documento técnico de introducción a DevOps en AWS, la automatización es la piedra angular de los servicios de AWS y es compatible internamente con todos los servicios, características y ofertas.
  - Preprocesar la Imagen de máquina de Amazon (AMI) puede acelerar el tiempo de lanzamiento. El Generador de imágenes de EC2 es un servicio totalmente administrado de AWS que le ayuda a automatizar la creación, el mantenimiento, la validación, el uso compartido y la implementación de AMI personalizadas, seguras y actualizadas para Linux o Windows.
  - Estos son algunos de los servicios que permiten la automatización:
    - AWS Elastic Beanstalk es un servicio para implementar y escalar rápidamente aplicaciones web desarrolladas con Java, .NET, PHP, Node.js, Python, Ruby, Go y Docker en servidores conocidos como, por ejemplo, Apache, NGINX, Passenger e IIS.
    - AWS Proton ayuda a los equipos de plataformas a conectar y coordinar todas las herramientas que sus equipos de desarrollo necesitan para el aprovisionamiento de infraestructuras, la implementación de código, la supervisión y las actualizaciones. AWS Proton permite una infraestructura automatizada, como el aprovisionamiento de código y la implementación de aplicaciones basadas en contenedores y sin servidor.
  - Usar la infraestructura como código facilita la automatización de su implementación y ayuda a lograr que sea inmutable. AWS proporciona servicios que permiten crear, implementar y mantener la infraestructura de forma programática, descriptiva y declarativa.
    - AWS CloudFormation ayuda a los desarrolladores a crear recursos de AWS de manera ordenada y predecible. Los recursos se escriben en archivos de texto en formato JSON o YAML. Las plantillas requieren una sintaxis y una estructura específicas que dependen de los tipos de recursos que se crean y administran. Los recursos se crean en JSON o YAML con cualquier editor de código, se registran en un sistema de control de versiones

y, a continuación, CloudFormation crea los servicios especificados de una forma segura y repetible.

  - AWS Serverless Application Model (AWS SAM) es un marco de código abierto que se puede utilizar para crear aplicaciones sin servidor en AWS. AWS SAM se integra con otros servicios de AWS y es una extensión de CloudFormation.
  - Kit de desarrollo de la nube de AWS (AWS CDK) es un marco de desarrollo de software de código abierto para modelar y aprovisionar los recursos de sus aplicaciones en la nube mediante lenguajes de programación conocidos. Puede usar AWS CDK para modelar la infraestructura de las aplicaciones mediante TypeScript, Python, Java y .NET. AWS CDK utiliza CloudFormation en segundo plano para aprovisionar recursos de una forma segura y repetible.
  - API de control de nube de AWS presenta un conjunto común de API de creación, lectura, actualización, eliminación y enumeración (CRUDL) que ayudan a los desarrolladores a administrar su infraestructura en la nube de forma sencilla y coherente. Las API comunes de la API de control en la nube permiten a los desarrolladores administrar de manera uniforme el ciclo de vida de los servicios de AWS y de terceros.
- Aplique patrones de implementación que tengan la mínima repercusión en los usuarios.
  - Implementaciones canario:
    - Configuración de una implementación de un lanzamiento canario de API Gateway
    - Create a pipeline with canary deployments for Amazon ECS using AWS App Mesh
  - Implementaciones azul/verde: en el documento técnico sobre implementaciones azul/verde en AWS se describen técnicas de ejemplo para implementar estrategias de implementación azul/verde.
- Detecte desviaciones de la configuración o el estado. Para obtener más detalles, consulte Detectar cambios de configuración no administrados en pilas y recursos con detección de derivación.

Recursos

Prácticas recomendadas relacionadas:

- REL08-BP05 Implementación de cambios con automatización

Documentos relacionados:

- Automatización de implementaciones seguras y sin intervención

- Leveraging AWS CloudFormation to create an immutable infrastructure at Nubank
- Infraestructura como código
- Implementing an alarm to automatically detect drift in AWS CloudFormation stacks

Videos relacionados:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability

REL08-BP05 Implementación de cambios con automatización

Las implementaciones y la aplicación de revisiones se automatizan para eliminar su impacto negativo.

Los cambios en los sistemas de producción son una de las mayores áreas de riesgo para muchas organizaciones. Consideramos que las implementaciones son un problema de primer orden que se debe resolver junto con los problemas empresariales que aborda nuestro software. Hoy en día, esto significa usar automatización en las operaciones siempre que resulte práctico, incluidas las pruebas y la implementación de cambios, la incorporación o eliminación de capacidad y la migración de datos.

Resultado deseado: incorpore la seguridad de la implementación automatizada en el proceso de lanzamiento con extensas pruebas de preproducción, reversiones automáticas e implementaciones de producción escalonadas. Esta automatización minimiza la posible repercusión en producción causada por implementaciones fallidas. Además, los desarrolladores ya no tienen que vigilar activamente las implementaciones en producción.

Patrones comunes de uso no recomendados:

- Los cambios se hacen de forma manual.
- Se salta los pasos de la automatización mediante flujos de trabajo de emergencia manuales.
- No sigue los planes y procesos establecidos en favor de plazos más rápidos.
- Lleva a cabo implementaciones de seguimiento rápidas sin dejar tiempo de incorporación.

Beneficios de establecer esta práctica recomendada: al utilizar la automatización para implementar todos los cambios, se elimina la posibilidad de que se produzcan errores humanos y se ofrece la posibilidad de llevar a cabo pruebas antes de cambiar de producción. Al llevar a cabo este proceso antes de la fase de producción, se verifica que los planes estén completos. Además, con la reversión

automática al proceso de lanzamiento, se pueden identificar los problemas de producción y devolver la carga de trabajo a su estado operativo anterior.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Automatice su canalización de implementación. Las canalizaciones de implementación le permiten invocar pruebas automatizadas, detectar anomalías y detener la canalización en un paso determinado antes de la implementación en producción o revertir automáticamente un cambio. Una parte integral de esto es la adopción de la cultura de integración continua e implementación/entrega continua (CI/CD), en la que la confirmación o el cambio de código pasan por varias etapas automatizadas, desde las etapas de creación y prueba hasta la implementación en entornos de producción.

Aunque la sabiduría convencional sugiere que mantenga a las personas informadas sobre los procedimientos operativos más difíciles, le recomendamos que automatice los procedimientos más difíciles por esa misma razón.

Pasos para la implementación

Siga estos pasos de automatización de las implementaciones para eliminar las operaciones manuales:

- Configure un repositorio de código para almacenar su código de forma segura: utilice un sistema de administración de código fuente alojado basado en una tecnología popular como Git para almacenar el código fuente y la configuración de infraestructura como código (IaC).
- Configuración de un servicio de integración continua para compilar el código fuente, ejecución de pruebas y creación de artefactos de implementación: para configurar un proyecto de compilación con este fin, consulte Getting started with AWS CodeBuild using the console.
- Configuración de un servicio de implementación que automatice las implementaciones de aplicaciones y gestione la complejidad de las actualizaciones de las aplicaciones sin depender de implementaciones manuales propensas a errores: AWS CodeDeploy automatiza las implementaciones de software en múltiples servicios de computación, como Amazon EC2, AWS Fargate, AWS Lambda y sus servidores en las instalaciones. Para configurar estos pasos, consulte Getting started with CodeDeploy.
- Configuración de un servicio de entrega continua que automatice las canalizaciones de lanzamiento para lograr actualizaciones de infraestructuras y aplicaciones más rápidas y fiables:

considere usar AWS CodePipeline para automatizar las canalizaciones de lanzamiento. Para obtener más información, consulte CodePipeline tutorials.

Recursos

Prácticas recomendadas relacionadas:

- OPS05-BP04 Uso de sistemas de administración de compilación e implementación
- OPS05-BP10 Automatización completa de la integración y la implementación
- OPS06-BP02 Implementaciones de prueba
- OPS06-BP04 Automatización de las pruebas y la reversión

Documentos relacionados:

- Continuous Delivery of Nested AWS CloudFormation Stacks Using AWS CodePipeline
- Socio de APN: socios que pueden ayudarle a crear soluciones de implementación automatizadas
- AWS Marketplace: productos que pueden usarse para automatizar sus implementaciones
- Automate chat messages with webhooks.
- Amazon Builders' Library: Asegurar la seguridad en las restauraciones durante las implementaciones
- Amazon Builders' Library: Evolución más rápida con la entrega continua
- Qué es AWS CodePipeline?
- What Is CodeDeploy?
- AWS Systems Manager Patch Manager
- What is Amazon SES?
- What is Amazon Simple Notification Service?

Videos relacionados:

- AWS Summit 2019: CI/CD on AWS

## Administración de errores

Preguntas

- REL 9. ¿Cómo se hace una copia de seguridad de los datos?
- REL 10. ¿Cómo utiliza el aislamiento de errores para proteger su carga de trabajo?
- REL 11. ¿Cómo diseña su carga de trabajo para que soporte los errores de los componentes?
- REL 12. ¿Cómo pone a prueba la fiabilidad?
- REL 13. ¿Cómo planifica la recuperación de desastres (DR)?


## REL 9. ¿Cómo se hace una copia de seguridad de los datos?

Realice copias de seguridad de los datos, las aplicaciones y la configuración para cumplir con los requisitos de objetivos de tiempo de recuperación (RTO) y objetivos de punto de recuperación (RPO).

Prácticas recomendadas

- REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes
- REL09-BP02 Protección y cifrado de copias de seguridad
- REL09-BP03 Copias de seguridad automáticas de los datos
- REL09-BP04 Recuperación periódica de los datos para verificar la integridad de la copia de seguridad y los procesos

REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes

Comprenda y use las funciones de copia de seguridad de los servicios y recursos de datos usados por la carga de trabajo. La mayoría de los servicios ofrecen capacidades para crear copias de seguridad de los datos de la carga de trabajo.

Resultado deseado: se han identificado y clasificado los orígenes de datos según su criticidad. A continuación, establezca una estrategia de recuperación de datos basada en el RPO. Esta estrategia supone crear una copia de seguridad de estos orígenes de datos o tener la capacidad de reproducir datos desde otros orígenes. En el caso de pérdida de datos, la estrategia implementada permite recuperar o reproducir los datos dentro de los RPO y RTO definidos.

Fase de madurez de la nube: básica

Patrones comunes de uso no recomendados:

- No ser consciente de todos los orígenes de datos de la carga de trabajo y su nivel de gravedad.
- No crear copias de seguridad de los orígenes de datos críticos.
- Crear copias de seguridad solamente de algunos orígenes de datos sin usar la criticidad como criterio.
- RPO sin definir o una frecuencia de copias de seguridad que no puede ajustarse al RPO.
- No evaluar si una copia de seguridad es necesaria o si se pueden reproducir datos desde otros orígenes.

Beneficios de establecer esta práctica recomendada: identificar los lugares en los que se necesitan copias de seguridad e implementar un mecanismo para crearlas, o poder reproducir los datos desde un origen externo, mejora la capacidad de restaurar y recuperar los datos durante una interrupción.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Todos los almacenes de datos de AWS ofrecen capacidades de copia de seguridad. En los servicios como Amazon RDS y Amazon DynamoDB también se pueden hacer copias de seguridad automatizadas, lo que facilita la recuperación en un momento dado (PITR). De este modo, podrá restaurar una copia de seguridad a cualquier momento hasta cinco minutos (o menos) antes del momento actual. Muchos servicios de AWS ofrecen la posibilidad de copiar las copias de seguridad a otra Región de AWS. AWS Backup es una herramienta que le permite centralizar y automatizar la protección de datos en todos los servicios de AWS. AWS Elastic Disaster Recovery le permite copiar cargas de trabajo completas del servidor y mantener una protección continua de los datos en las instalaciones, entre zonas de disponibilidad o entre regiones, con un objetivo de punto de recuperación (RPO) medido en segundos.

Amazon S3 puede usarse como destino de las copias de seguridad para los orígenes de datos autoadministrados y administrados por AWS. Los servicios de AWS como Amazon EBS, Amazon RDS y Amazon DynamoDB tienen capacidades integradas para crear copias de seguridad. También se puede usar software de copias de seguridad de terceros.

Se pueden hacer copias de seguridad de los datos en las instalaciones en la Nube de AWS con AWS Storage Gateway o AWS DataSync. Los buckets de Amazon S3 se pueden utilizar para almacenar estos datos en AWS. Amazon S3 ofrece varios niveles de almacenamiento, como Amazon Glacier o Amazon Glacier Deep Archive, para reducir el costo del almacenamiento de datos.

Es posible que pueda satisfacer las necesidades de recuperación de datos mediante la reproducción de los datos desde otros orígenes. Por ejemplo, los nodos de réplica de Amazon ElastiCache o las réplicas de lectura de Amazon RDS podrían usarse para reproducir datos si se pierde el elemento principal. En aquellos casos en que los orígenes como este se puedan utilizar para cumplir su objetivo de punto de recuperación (RPO) y objetivo de tiempo de recuperación (RTO), es posible que no necesite ninguna copia de seguridad. Otro ejemplo: si trabaja con Amazon EMR, puede que no sea necesario hacer copias de seguridad del almacén de datos de HDFS, siempre y cuando pueda reproducir los datos en Amazon EMR desde Amazon S3.

Al seleccionar una estrategia de copia de seguridad, piense en el tiempo que se necesita para recuperar los datos. El tiempo necesario para recuperar datos depende del tipo de copia de seguridad (en el caso de una estrategia de copia de seguridad) o de la complejidad del mecanismo de reproducción de datos. Este tiempo debería ajustarse al RTO de la carga de trabajo.

Pasos para la implementación

1. Identifique todos los orígenes de datos de la carga de trabajo. Los datos se pueden almacenar en varios recursos, como bases de datos, volúmenes, sistemas de archivos, sistemas de registro y almacenamiento de objetos. Consulte la sección Recursos para encontrar documentos relacionados sobre los diferentes servicios de AWS en los que se almacenan los datos y la capacidad de copia de seguridad que ofrecen estos servicios.
2. Clasifique los orígenes de datos según su criticidad. Los distintos conjuntos de datos tendrán diferentes niveles de criticidad para una carga de trabajo y, por tanto, distintos requisitos de resiliencia. Por ejemplo, algunos datos podrían ser críticos y requerir un RPO cercano a cero, mientras que otros datos podrían ser menos críticos y tolerar un RPO más alto y cierta pérdida de datos. Del mismo modo, los distintos conjuntos de datos podrían tener también diferentes requisitos en cuanto al RTO.
3. Uso de AWS o servicios de terceros para crear copias de seguridad de los datos. AWS Backup es un servicio administrado que permite crear copias de seguridad de diversos orígenes de datos en AWS. AWS Elastic Disaster Recovery administra la replicación automatizada de datos en menos de un segundo a una Región de AWS. La mayoría de los servicios de AWS también disponen de capacidades nativas para crear copias de seguridad. AWS Marketplace tiene muchas soluciones que ofrecen también estas capacidades. Consulte la sección Recursos que aparece a continuación para obtener información sobre cómo crear copias de seguridad de los datos de distintos servicios de AWS.
4. Establezca un mecanismo de reproducción de datos para los datos que no tengan copia de seguridad. Puede decidir no crear una copia de seguridad de datos que puedan reproducirse

desde otros orígenes y por distintos motivos. Podría darse una situación en la que sea más barato reproducir datos de orígenes cuando sea necesario en lugar de crear una copia de seguridad, ya que podría existir un costo asociado con el almacenamiento de copias de seguridad. Otro ejemplo es cuando la restauración desde una copia de seguridad tarda más tiempo que la reproducción de los datos desde el origen, lo que implica un incumplimiento del RTO. En tales situaciones, sopese los pros y los contras y establezca un proceso bien definido sobre cómo se pueden reproducir los datos desde estos orígenes cuando sea necesaria una recuperación de los datos. Por ejemplo, si ha cargado datos desde Amazon S3 en un almacenamiento de datos (como Amazon Redshift) o un clúster de MapReduce (como Amazon EMR) para analizar dichos datos, esto podría ser un ejemplo de datos que se pueden reproducir desde otros orígenes. Siempre y cuando los resultados de estos análisis se almacenen en algún lugar o sean reproducibles, no sufriría ninguna pérdida de datos por un error en el almacenamiento de datos o el clúster de MapReduce. Otros ejemplos que se pueden reproducir desde el origen son las cachés (como Amazon ElastiCache) o las réplicas de lectura de RDS.

5. Establezca una cadencia para hacer copias de seguridad de los datos. La creación de copias de seguridad de orígenes de datos es un proceso periódico y la frecuencia debería depender del RPO.

Nivel de esfuerzo para el plan de implementación: moderado

Recursos

Prácticas recomendadas relacionadas:

REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos

REL13-BP02 Uso de estrategias de recuperación definidas para cumplir los objetivos de recuperación

Documentos relacionados:

- Qué es AWS Backup?
- What is AWS DataSync?
- What is Volume Gateway?
- Socio de APN: socios que pueden ayudar con la copia de seguridad
- AWS Marketplace: productos que pueden usarse para la copia de seguridad
- Instantáneas de Amazon EBS

- Backing Up Amazon EFS
- Backing up Amazon FSx for Windows File Server
- Backup and Restore for ElastiCache for Redis
- Creating a DB Cluster Snapshot in Neptune
- Creación de una instantánea de base de datos
- Creating an EventBridge Rule That Triggers on a Schedule
- Replicación entre regiones con Amazon S3
- AWS Backup de EFS a EFS
- Exporting Log Data to Amazon S3
- Administración del ciclo de vida del almacenamiento
- Copia de seguridad y restauración bajo demanda para DynamoDB
- Recuperación a un momento dado en DynamoDB
- Working with Amazon OpenSearch Service Index Snapshots
- What is AWS Elastic Disaster Recovery?

Videos relacionados:

- AWS re:Invent 2021 - Backup, disaster recovery, and ransomware protection with AWS
- AWS Backup Demo: Cross-Account and Cross-Region Backup
- AWS re:Invent 2019: Deep dive on AWS Backup, ft. Rackspace (STG341)

REL09-BP02 Protección y cifrado de copias de seguridad

Controle y detecte el acceso a las copias de seguridad con autenticación y autorización. Evite que la integridad de los datos de las copias de seguridad se vea comprometida (y detecte los casos en los que así sea) mediante el cifrado.

Implemente controles de seguridad para evitar el acceso no autorizado a los datos de copia de seguridad. Cifre las copias de seguridad para proteger la confidencialidad y la integridad de los datos.

Patrones comunes de uso no recomendados:

- Tener el mismo acceso a las automatizaciones de las copias de seguridad y restauración que a los datos.

- No cifrar las copias de seguridad.
- No se implementa la inmutabilidad como protección contra la eliminación o la manipulación.
- Se usa el mismo dominio de seguridad para los sistemas de producción y de copia de seguridad.
- No se valida la integridad de las copias de seguridad mediante pruebas periódicas.

Beneficios de establecer esta práctica recomendada:

- La protección de las copias de seguridad impide que se manipulen los datos y el cifrado de los datos impide el acceso a esos datos si se exponen de forma accidental.
- Protección mejorada contra el ransomware y otras ciberamenazas que afectan a la infraestructura de copia de seguridad.
- Reducción del tiempo de recuperación tras un ciberincidente mediante procesos de recuperación validados.
- Capacidades de continuidad empresarial mejoradas durante los incidentes de seguridad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Controle y detecte el acceso a las copias de seguridad con autenticación y autorización, como AWS Identity and Access Management (IAM). Evite que la integridad de los datos de las copias de seguridad se vea comprometida (y detecte los casos en los que así sea) mediante el cifrado.

Amazon S3 admite varios métodos de cifrado de los datos en reposo. Con el cifrado del servidor, Amazon S3 acepta sus objetos como datos sin cifrar y después los cifra a medida que se almacenan. Con el cifrado del cliente, la aplicación de la carga de trabajo es la responsable de cifrar los datos antes de que se envíen a Amazon S3. Ambos métodos le permiten utilizar AWS Key Management Service (AWS KMS) para crear y almacenar la clave de los datos, o puede facilitar la suya propia, de la que será responsable. Con AWS KMS, puede establecer políticas con IAM sobre quién puede acceder a sus claves de datos y datos descifrados y quién no.

Para Amazon RDS, si ha decidido cifrar las bases de datos, sus copias de seguridad también estarán cifradas. Las copias de seguridad de DynamoDB siempre están cifradas. Al usar AWS Elastic Disaster Recovery, todos los datos en tránsito y en reposo están cifrados. Con la Recuperación de desastres elástica, los datos en reposo se pueden cifrar con la clave de cifrado de volumen predeterminada de Amazon EBS o una clave personalizada administrada por el cliente.

### Consideraciones sobre la resiliencia cibernética

Para mejorar la seguridad de las copias de seguridad frente a las ciberamenazas, considere la posibilidad de implementar estos controles adicionales además del cifrado:

- Implemente la inmutabilidad mediante AWS Backup Vault Lock o el bloqueo de objetos de Amazon S3 para evitar que los datos de copia de seguridad se alteren o eliminen durante su periodo de retención, protegiéndolos contra el ransomware y la eliminación malintencionada.
- Establezca un aislamiento lógico entre los entornos de producción y de copia de seguridad con un almacén aislado lógicamente de AWS Backup para los sistemas críticos, lo que crea una separación que ayuda a evitar que ambos entornos se vean comprometidos simultáneamente.
- Valide la integridad de las copias de seguridad con regularidad mediante pruebas de restauración de AWS Backup para comprobar que las copias de seguridad no están dañadas y que se pueden restaurar correctamente tras un ciberincidente.
- Implemente la aprobación de varias partes para las operaciones de recuperación críticas mediante la aprobación de varias partes de AWS Backup para evitar intentos de recuperación no autorizados o malintencionados, exigiendo la autorización de varios aprobadores designados.

### Pasos para la implementación

1. Use el cifrado en cada uno de los almacenes de datos. Si los datos de origen están cifrados, la copia de seguridad también estará cifrada.
   - Utilice el cifrado en Amazon RDS. Puede configurar el cifrado en reposo mediante AWS Key Management Service al crear una instancia de RDS.
   - Use el cifrado en volúmenes de Amazon EBS. Puede configurar el cifrado predeterminado o especificar una clave única al crear los volúmenes.
   - Use el cifrado de Amazon DynamoDB requerido. DynamoDB cifra todos los datos en reposo. Puede utilizar una clave de AWS propiedad de AWS KMS o una clave de KMS administrada por AWS, siempre que especifique una clave que esté almacenada en su cuenta.
   - Cifre los datos almacenados en Amazon EFS. Configure el cifrado cuando cree el sistema de archivos.
   - Configure el cifrado en las regiones de origen y destino. Puede configurar el cifrado en reposo en Amazon S3 con las claves almacenadas en KMS, pero las claves son específicas de la región. Puede especificar las claves de destino cuando configure la replicación.

- Elija si desea utilizar el cifrado predeterminado o utilizar el cifrado personalizado de Amazon EBS para la Recuperación de desastres elástica. Esta opción cifrará sus datos replicados en reposo en los discos de la subred de la zona de preparación y en los discos replicados.

2. Implemente permisos de privilegio mínimo para acceder a las copias de seguridad. Siga las prácticas recomendadas para limitar el acceso a las copias de seguridad, instantáneas y réplicas de acuerdo con las prácticas recomendadas de seguridad.
3. Configure la inmutabilidad de las copias de seguridad críticas. En el caso de los datos críticos, implemente AWS Backup Vault Lock o el bloqueo de objetos de S3 para evitar que se eliminen o modifiquen durante el periodo de retención especificado. Para conocer los detalles de implementación, consulte AWS Backup Vault Lock.
4. Cree una separación lógica para los entornos de copia de seguridad. Implemente un almacén aislado lógicamente de AWS Backup para los sistemas críticos que requieren una protección mejorada contra las ciberamenazas. Para obtener una guía de implementación, consulte Building cyber resiliency with AWS Backup logically air-gapped vault.
5. Implemente procesos de validación de copias de seguridad. Configure las pruebas de restauración de AWS Backup para comprobar que las copias de seguridad no están dañadas y que se pueden restaurar correctamente tras un ciberincidente. Para obtener más información, consulte Validate recovery readiness with AWS Backup restore testing.
6. Configure la aprobación de varias partes para las operaciones de recuperación confidenciales. En el caso de los sistemas críticos, implemente la aprobación de varias partes de AWS Backup para solicitar la autorización de varios aprobadores designados antes de que pueda continuar con la recuperación. Para obtener información detallada sobre la implementación, consulte Improve recovery resilience with AWS Backup support for Multi-party approval.

Recursos

Documentos relacionados:

- AWS Marketplace: productos que pueden usarse para la copia de seguridad
- Amazon EBS Encryption
- Amazon S3: protección de los datos mediante el cifrado
- Configuración adicional de CRR: replicación de objetos creados con el cifrado del servidor (SSE) usando las claves de cifrado almacenadas en AWS KMS
- Cifrado en reposo en DynamoDB
- Encrypting Amazon RDS Resources

- Encrypting Data and Metadata in Amazon EFS
- Encryption for Backups in AWS
- Administración de tablas de cifrado
- Pilar de seguridad: Marco de AWS Well-Architected
- What is AWS Elastic Disaster Recovery?
- FSISEC11: ¿Cómo se protege contra el ransomware?
- Ransomware Risk Management on AWS Using the NIST Cyber Security Framework
- Building cyber resiliency with AWS Backup logically air-gapped vault
- Validate recovery readiness with AWS Backup restore testing
- Improve recovery resilience with AWS Backup support for Multi-party approval

Ejemplos relacionados:

- Implementing Bi-Directional Cross-Region Replication (CRR) for Amazon S3

REL09-BP03 Copias de seguridad automáticas de los datos

Configure las copias de seguridad para que se creen automáticamente con arreglo a un calendario periódico determinado por el objetivo de punto de recuperación (RPO) o cuando se produzcan cambios en el conjunto de datos. En el caso de los conjuntos de datos críticos con requisitos de pérdida de datos bajos, es necesario crear una copia de seguridad automática con frecuencia, mientras que en el de los datos menos críticos para los que resultan aceptables ciertas pérdidas, las copias de seguridad pueden ser menos frecuentes.

Resultado deseado: un proceso automatizado que crea copias de seguridad de los orígenes de datos con una cadencia establecida.

Patrones comunes de uso no recomendados:

- Hacer las copias de seguridad manualmente.
- Usar recursos que tengan la función de copia de seguridad, pero no incluir la copia de seguridad en la automatización.

Beneficios de establecer esta práctica recomendada: al automatizar las copias de seguridad, se comprueba que se hagan con regularidad en función del RPO y, si no se hacen, se avisa al usuario.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

AWS Backup se puede usar para crear copias de seguridad automatizadas de los datos de diversos orígenes de datos de AWS. Es posible crear copias de seguridad de las instancias de Amazon RDS casi de forma continua cada cinco minutos, y de los objetos de Amazon S3 cada quince minutos, lo que facilita una recuperación en un momento dado (PITR) a un punto específico del historial de copias de seguridad. Para otros orígenes de datos de AWS, como los volúmenes de Amazon EBS, las tablas de Amazon DynamoDB o los sistemas de archivos de Amazon FSx, AWS Backup puede ejecutar una copia de seguridad automatizada con una frecuencia que puede llegar a ser de una hora. Estos servicios también ofrecen capacidades de copia de seguridad nativas. Los servicios de AWS que ofrecen copias de seguridad automatizadas con recuperación en un momento dado son Amazon DynamoDB, Amazon RDS y Amazon Keyspaces (para Apache Cassandra), que se pueden restaurar a un momento específico del historial de copias de seguridad. La mayoría del resto de servicios de almacenamiento de datos de AWS ofrecen la capacidad de programar copias de seguridad periódicas, con una frecuencia que puede llegar a ser de una hora.

Amazon RDS y Amazon DynamoDB ofrecen copias de seguridad continuas con recuperación en un momento dado. El control de versiones de Amazon S3, una vez activado, es automático. Se puede utilizar Amazon Data Lifecycle Manager para automatizar la creación, retención y eliminación de instantáneas de Amazon EBS. También puede automatizar la creación, copia, desuso y anulación de registro de imágenes de máquina de Amazon (AMI) basadas en Amazon EBS y sus instantáneas de Amazon EBS subyacentes.

AWS Elastic Disaster Recovery proporciona replicación continua en el nivel de bloque desde el entorno de origen (en las instalaciones o AWS) a la región de recuperación de destino. El servicio crea y administra automáticamente instantáneas de Amazon EBS de un momento dado.

Para obtener una vista centralizada de la automatización y el historial de sus copias de seguridad, AWS Backup proporciona una solución de copia de seguridad totalmente administrada y basada en políticas. Centraliza y automatiza la copia de seguridad de datos entre varios servicios de AWS en la nube y en el entorno en las instalaciones con AWS Storage Gateway.

De forma adicional al control de versiones, Amazon S3 incluye también replicación. Todo el bucket de S3 se puede replicar automáticamente en otro bucket de la misma Región de AWS o una diferente.

Pasos para la implementación

1. Identifique los orígenes de datos de los que se están haciendo copias de seguridad manualmente. Para obtener más información, consulte REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes.
2. Determine el RPO de la carga de trabajo. Para obtener más información, consulte REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos.
3. Use una solución de copia de seguridad automatizada o un servicio administrado. AWS Backup es un servicio totalmente administrado que facilita la centralización y automatización de la protección de datos en todos los servicios de AWS, en la nube y en las instalaciones. Mediante planes de copia de seguridad en AWS Backup, cree reglas que definan de qué recursos se debe hacer copia de seguridad y con qué frecuencia deben crearse. Esta frecuencia debe determinarla el RPO establecido en el paso 2. Para obtener orientación práctica sobre cómo crear copias de seguridad automatizadas mediante AWS Backup, consulte Testing Backup and Restore of Data. La mayoría de los servicios de AWS que almacenan datos ofrecen capacidades de copia de seguridad nativas. Por ejemplo, se puede utilizar RDS para hacer copias de seguridad automatizadas con recuperación en un momento dado (PITR).
4. En el caso de los orígenes de datos que no sean compatibles con una solución de copia de seguridad automatizada o un servicio administrado, como los orígenes de datos en las instalaciones o las colas de mensajes, considere la posibilidad de utilizar una solución de terceros de confianza para crear copias de seguridad automatizadas. Como alternativa, puede crear una automatización que se encargue de esto con la AWS CLI o algún SDK. Puede usar funciones de AWS Lambda o AWS Step Functions para definir la lógica implicada en la creación de una copia de seguridad de datos y utilizar Amazon EventBridge para invocarla con una frecuencia basada en el RPO.

Nivel de esfuerzo para el plan de implementación: bajo

Recursos

Documentos relacionados:

- Socio de APN: socios que pueden ayudar con la copia de seguridad
- AWS Marketplace: productos que pueden usarse para la copia de seguridad
- Creating an EventBridge Rule That Triggers on a Schedule
- ¿Qué es AWS Backup?
- ¿Qué es AWS Step Functions?

- What is AWS Elastic Disaster Recovery?

Videos relacionados:

- AWS re:Invent 2019: Deep dive on AWS Backup, ft. Rackspace (STG341)

REL09-BP04 Recuperación periódica de los datos para verificar la integridad de la copia de seguridad y los procesos

Valide que su implementación del proceso de copia de seguridad cumpla con los objetivos de tiempo de recuperación (RTO) y los objetivos de punto de recuperación (RPO) mediante una prueba de recuperación.

Resultado deseado: los datos de las copias de seguridad se recuperan periódicamente mediante mecanismos bien definidos para verificar que la recuperación sea posible dentro del objetivo de tiempo de recuperación (RTO) establecido para la carga de trabajo. Verifique que la restauración a partir de una copia de seguridad dé como resultado un recurso que contenga los datos originales sin que ninguno de ellos resulte dañado o inaccesible, y una pérdida de datos coherente con el objetivo de punto de recuperación (RPO).

Patrones comunes de uso no recomendados:

- Restaurar una copia de seguridad, pero no consultar ni recuperar ningún dato para comprobar que la restauración sea posible.
- Suponer que existe una copia de seguridad.
- Suponer que la copia de seguridad de un sistema está plenamente operativa y que es posible recuperar datos de ella.
- Suponer que el tiempo de restauración o recuperación de datos de una copia de seguridad entra dentro del RTO para la carga de trabajo.
- Suponer que los datos que contiene la copia de seguridad están dentro del RPO para la carga de trabajo
- Restaurar cuando sea necesario, sin usar un manual de procedimientos, o fuera de un procedimiento automatizado.

Beneficios de establecer esta práctica recomendada: al probar la recuperación de las copias de seguridad, se comprueba que los datos se pueden restaurar cuando sea necesario sin preocuparse

de que falten datos o estén dañados, de que la restauración y la recuperación sean posibles dentro del RTO de la carga de trabajo y que cualquier pérdida de datos recaiga dentro del RPO correspondiente a la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La comprobación de la capacidad de copia de seguridad y restauración aumenta la confianza en la capacidad de llevar a cabo estas acciones durante una interrupción. Restaure periódicamente las copias de seguridad en una nueva ubicación y lleve a cabo pruebas para verificar la integridad de los datos. Algunas de las pruebas habituales que deberían efectuarse consisten en comprobar que todos los datos estén disponibles, no estén dañados, sean accesibles y si la pérdida de datos (si la hay) se ajusta al RPO de la carga de trabajo. Estas pruebas también pueden ayudar a determinar si los mecanismos de recuperación son lo suficientemente rápidos como para tener capacidad para el RTO de la carga de trabajo.

Con AWS, puede crear un entorno de pruebas y restaurar sus copias de seguridad para evaluar las capacidades en cuanto al RTO y al RPO y llevar a cabo pruebas sobre el contenido y la integridad de los datos.

Además, Amazon RDS y Amazon DynamoDB permiten la recuperación en un momento dado (PITR). Mediante la copia de seguridad continua, puede restaurar su conjunto de datos al estado en el que se encontraba en una fecha y hora específicas.

Si todos los datos están disponibles, no están dañados, son accesibles y la pérdida de datos (si la hay) se ajusta al RPO de la carga de trabajo. Estas pruebas también pueden ayudar a determinar si los mecanismos de recuperación son lo suficientemente rápidos como para tener capacidad para el RTO de la carga de trabajo.

AWS Elastic Disaster Recovery ofrece instantáneas de recuperación en un momento dado continuas de volúmenes de Amazon EBS. A medida que se replican los servidores de origen, los estados de un momento dado se cronifican en el tiempo en función de la política configurada. La Recuperación de desastres elástica ayuda a verificar la integridad de estas instantáneas mediante el lanzamiento de instancias con fines de prueba y simulacro sin redirigir el tráfico.

Pasos para la implementación

1. Identifique los orígenes de datos de los que se están haciendo copias de seguridad actualmente y dónde se almacenan dichas copias de seguridad. Para obtener una guía para la implementación,

consulte REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes.

2. Establezca los criterios de validación de datos para cada origen de datos. Los diferentes tipos de datos tendrán distintas propiedades, lo que podría requerir diferentes mecanismos de validación. Considere cómo se podrían validar estos datos antes de contar con la confianza suficiente para usarlos en producción. Algunas formas habituales de validar los datos son usar las propiedades de datos y copias de seguridad como el tipo de datos, el formato, la suma de comprobación, el tamaño o una combinación de ellas con lógica de validación personalizada. Por ejemplo, podría tratarse de una comparación de los valores de las sumas de comprobación entre el recurso restaurado y el origen de datos en el momento en que se creó la copia de seguridad.
3. Establezca el RTO y el RPO para restaurar los datos según su importancia. Para obtener una guía para la implementación, consulte REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos.
4. Evalúe su capacidad de recuperación. Revise su estrategia de copia de seguridad y restauración para comprender si se ajusta a su RTO y RPO y ajuste la estrategia según sea necesario. Con AWS Resilience Hub, puede evaluar la carga de trabajo. La evaluación compara la configuración de su aplicación con la política de resiliencia y notifica si se pueden cumplir los objetivos de RTO y RPO.
5. Ejecute una restauración de prueba con los procesos actualmente establecidos que se utilizan en la producción para la restauración de datos. Estos procesos dependen de cómo se haya creado la copia de seguridad del origen de datos, el formato y la ubicación del almacenamiento de la copia de seguridad, o de si los datos se reproducen desde otros orígenes. Por ejemplo, si utiliza un servicio administrado, como AWS Backup, podría ser tan sencillo como restaurar la copia de seguridad en un nuevo recurso. Si usó AWS Elastic Disaster Recovery, puede iniciar un simulacro de recuperación.
6. Valide la recuperación de datos del recurso restaurado en función de los criterios que estableció previamente para la validación de los datos. ¿Los datos restaurados y recuperados contienen el registro o elemento más reciente en el momento de la copia de seguridad? ¿Estos datos se ajustan al RPO de la carga de trabajo?
7. Calcule el tiempo necesario para la restauración y la recuperación y compárelo con el RTO establecido. ¿Este proceso se ajusta al RTO de la carga de trabajo? Por ejemplo, compare las marcas de tiempo del momento en que se inició el proceso de restauración y de cuando se completó la validación de la recuperación para calcular cuánto tarda este proceso. Todas las llamadas a la API de AWS llevan una marca de tiempo y esta información está disponible en AWS CloudTrail. Aunque esta información puede proporcionar detalles sobre cuándo se inició

el proceso de restauración, la marca de tiempo final del momento en que finalizó la validación debería quedar registrada mediante su lógica de validación. Si se utiliza un proceso automatizado, se pueden utilizar servicios como Amazon DynamoDB para almacenar esta información. Además, muchos servicios de AWS proporcionan un historial de eventos que facilita información con marcas de tiempo cuando ocurren determinadas acciones. En AWS Backup, las acciones de copia de seguridad y restauración se denominan trabajos, y estos trabajos contienen información sobre la marca de tiempo como parte de sus metadatos, que se puede utilizar para medir el tiempo necesario para la restauración y la recuperación.

8. Notifique a las partes interesadas si se produce un error en la validación de los datos o si el tiempo necesario para la restauración y la recuperación supera el RTO establecido para la carga de trabajo. Al implementar la automatización para que lo haga, como en este laboratorio, se pueden utilizar servicios como Amazon Simple Notification Service (Amazon SNS) para enviar notificaciones push, como correos electrónicos o SMS, a las partes interesadas. Estos mensajes también se pueden publicar en aplicaciones de mensajería como Amazon Chime, Slack o Microsoft Teams, o se pueden usar para crear tareas como OpsItems con el Centro de operaciones de AWS Systems Manager.
9. Automatice este proceso para que se ejecute periódicamente. Por ejemplo, se pueden usar servicios como AWS Lambda o una máquina de estados en AWS Step Functions para automatizar los procesos de restauración y recuperación, mientras que se puede usar Amazon EventBridge para invocar este flujo de trabajo de automatización periódicamente como se muestra en el siguiente diagrama de arquitectura. Obtenga información sobre cómo automatizar la validación de recuperación de datos con AWS Backup. Además, en este laboratorio de Well-Architected se ofrece una experiencia práctica sobre una forma de ejecutar la automatización de varios de los pasos que se describen aquí.

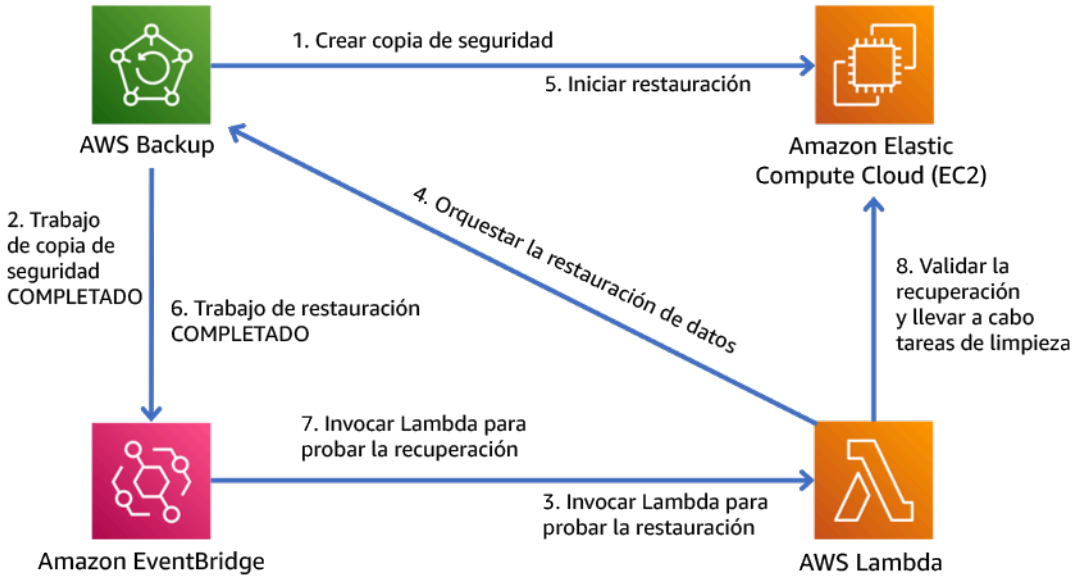



Figura 9. Proceso de copia de seguridad y restauración automatizado

Nivel de esfuerzo para el plan de implementación: de moderado a alto, según la complejidad de los criterios de validación.

Recursos

Documentos relacionados:

- Automate data recovery validation with AWS Backup
- Socio de APN: socios que pueden ayudar con la copia de seguridad
- AWS Marketplace: productos que pueden usarse para la copia de seguridad
- Creating an EventBridge Rule That Triggers on a Schedule
- Copia de seguridad y restauración bajo demanda para DynamoDB
- Qué es AWS Backup?
- Qué es AWS Step Functions?
- Qué es AWS Elastic Disaster Recovery
- AWS Elastic Disaster Recovery

## REL 10. ¿Cómo utiliza el aislamiento de errores para proteger su carga de trabajo?

El aislamiento de fallos limita el impacto de un fallo en un componente o sistema a un límite definido. Mediante un aislamiento adecuado, los componentes que se encuentran fuera del límite no se ven

afectados por el error. Hacer que la carga de trabajo supere varios límites de aislamiento de fallos puede hacer que sea más resistente a los fallos.

Prácticas recomendadas

- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL10-BP02 Automatización de la recuperación de los componentes restringidos a una sola ubicación
- REL10-BP03 Uso de arquitecturas herméticas para limitar el alcance del impacto

REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones

Distribuya los datos y los recursos de la carga de trabajo entre varias zonas de disponibilidad o, si es necesario, entre varias Regiones de AWS.

Un principio fundamental para el diseño de servicios en AWS es evitar puntos únicos de error, incluida la infraestructura física subyacente. AWS proporciona recursos y servicios de computación en la nube a nivel internacional en múltiples ubicaciones geográficas denominadas regiones. Cada región es independiente física y lógicamente y consta de tres o más zonas de disponibilidad (AZ). Las zonas de disponibilidad están geográficamente cerca unas de otras, pero están físicamente separadas y aisladas. Cuando distribuye sus cargas de trabajo entre las zonas y regiones de disponibilidad, mitiga el riesgo de amenazas como incendios, inundaciones, desastres relacionados con el clima, terremotos y errores humanos.

Cree una estrategia de ubicación para ofrecer una alta disponibilidad que sea adecuada para las cargas de trabajo.

Resultado deseado: las cargas de trabajo de producción se distribuyen entre varias zonas de disponibilidad (AZ) o regiones para lograr la tolerancia a los errores y una alta disponibilidad.

Patrones comunes de uso no recomendados:

- La carga de trabajo de producción solo existe en una sola zona de disponibilidad.
- Se implementa una arquitectura multirregional cuando una arquitectura Multi-AZ podría cumplir los requisitos empresariales.
- Las implementaciones o los datos se desincronizan, lo que provoca cambios en la configuración o una replicación insuficiente de los datos.
- No se tienen en cuenta las dependencias entre los componentes de la aplicación si los requisitos de resiliencia y de múltiples ubicaciones difieren entre esos componentes.

Beneficios de establecer esta práctica recomendada:

- Su carga de trabajo es más resistente a los incidentes, como los cortes de energía o de control ambiental, los desastres naturales, los fallos del servicio previo o los problemas de red que afectan a una zona de disponibilidad o a toda una región.
- Puede acceder a un inventario más amplio de instancias de Amazon EC2 y reducir la probabilidad de que se produzcan InsufentCapacityExceptions (ICE) al lanzar tipos de instancias de EC2 específicos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Implemente y opere todas las cargas de trabajo de producción en al menos dos zonas de disponibilidad (AZ) en una región.

Uso de varias zonas de disponibilidad

Las zonas de disponibilidad son ubicaciones de alojamiento de recursos que están separadas físicamente entre sí para evitar fallos correlacionados debidos a riesgos como incendios, inundaciones y tornados. Cada zona de disponibilidad tiene una infraestructura física independiente, que incluye conexiones eléctricas de servicios públicos, fuentes de energía de emergencia, servicios mecánicos y conectividad de red. Esta disposición limita los errores en cualquiera de estos componentes únicamente a la zona de disponibilidad afectada. Por ejemplo, si un incidente en toda la zona de disponibilidad hace que las instancias de EC2 no estén disponibles en la zona de disponibilidad afectada, las instancias de la otra zona de disponibilidad seguirán disponibles.

A pesar de estar separadas físicamente, las zonas de disponibilidad de la misma Región de AWS están lo suficientemente cerca como para proporcionar redes de alto rendimiento y baja latencia (milisegundos de un solo dígito). Puede replicar los datos de forma sincrónica entre las zonas de disponibilidad para la mayoría de las cargas de trabajo sin que ello afecte significativamente a la experiencia del usuario. Esto significa que puede utilizar las zonas de disponibilidad en una configuración activa/activa o activa/en espera.

Toda la computación asociada a su carga de trabajo debe distribuirse entre varias zonas de disponibilidad. Esto incluye instancias de Amazon EC2, tareas de AWS Fargate y funciones de AWS Lambda conectadas a VPC. Los servicios de computación de AWS, incluidos EC2 Auto Scaling, Amazon Elastic Container Service (ECS) y Amazon Elastic Kubernetes Service (EKS), le

proporcionan formas de lanzar y administrar el procesamiento en todas las zonas de disponibilidad. Configúrelos para reemplazar automáticamente el procesamiento según sea necesario en una zona de disponibilidad diferente para mantener la disponibilidad. Para dirigir el tráfico a las zonas de disponibilidad disponibles, coloque un equilibrador de cargas delante del equipo, como un Equilibrador de carga de aplicación o un Equilibrador de carga de red. Los equilibradores de carga de AWS pueden redirigir el tráfico a las instancias disponibles en caso de que la zona de disponibilidad se vea afectada.

También debe replicar los datos de su carga de trabajo y ponerlos a disposición en varias zonas de disponibilidad. Algunos servicios de datos gestionados de AWS, como Amazon S3, Amazon Elastic File Service (EFS), Amazon Aurora, Amazon DynamoDB, Amazon Simple Queue Service (SQS) y Amazon Kinesis Data Streams, replican los datos en varias zonas de disponibilidad de forma predeterminada y son robustos frente al deterioro de la zona de disponibilidad. Con otros servicios de datos gestionados de AWS, como Amazon Relational Database Service (RDS), Amazon Redshift y Amazon ElastiCache, debe habilitar la replicación Multi-AZ. Una vez activados, estos servicios detectan automáticamente una alteración en la zona de disponibilidad, redirigen las solicitudes a una zona de disponibilidad disponible y vuelven a replicar los datos según sea necesario tras la recuperación sin la intervención del cliente. Familiarícese con la guía del usuario de cada servicio de datos gestionado de AWS que utilice para comprender las funciones, comportamientos y operaciones en zonas de disponibilidad múltiples.

Si utiliza un almacenamiento autogestionado, como los volúmenes de Amazon Elastic Block Store (EBS) o el almacenamiento de instancias de Amazon EC2, debe gestionar usted mismo la replicación en zonas de disponibilidad múltiples (Multi-AZ).

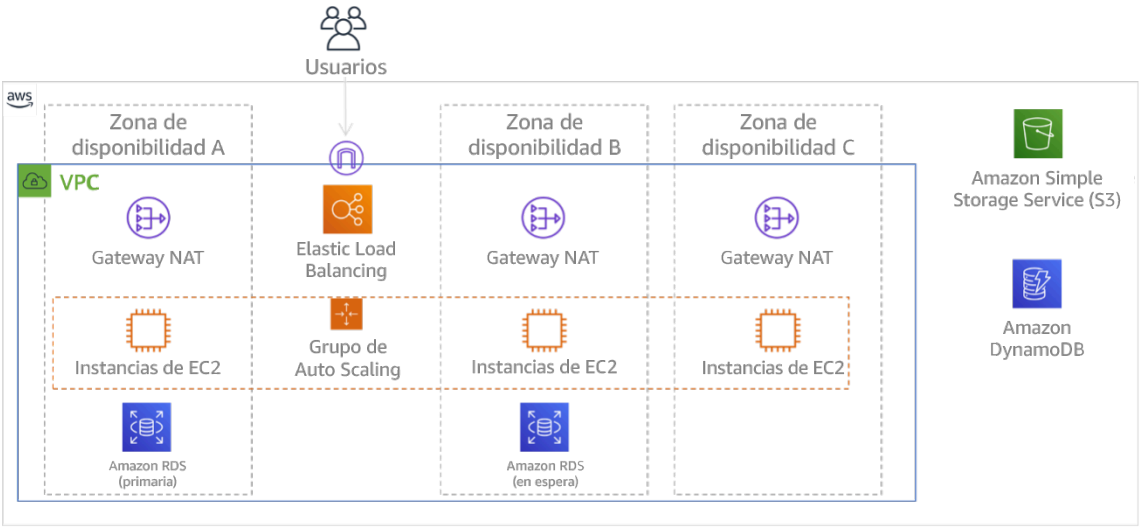


Figura 9: Arquitectura de varios niveles implementada en tres zonas de disponibilidad. Tenga en cuenta que Amazon S3 y Amazon DynamoDB siempre se implementan automáticamente en varias zonas de disponibilidad (Multi-AZ). El ELB también se implementa en las tres zonas.

Uso de múltiples Regiones de AWS

Si tiene cargas de trabajo que requieren una capacidad de recuperación extrema (como infraestructuras críticas, aplicaciones relacionadas con la salud o servicios con requisitos de disponibilidad estrictos u obligatorios por parte del cliente), es posible que necesite una disponibilidad adicional más allá de lo que puede ofrecer una sola Región de AWS. En este caso, debe implementar y operar su carga de trabajo en al menos dos Regiones de AWS (siempre que sus requisitos de residencia de datos lo permitan).

Las Regiones de AWS se encuentran en diferentes regiones geográficas de todo el mundo y en varios continentes. Las Regiones de AWS tienen una separación física y un aislamiento aún mayores que las zonas de disponibilidad por sí solas. Los servicios de AWS, con pocas excepciones, aprovechan este diseño para funcionar de forma totalmente independiente entre diferentes regiones (también conocidos como servicios regionales). El fallo de un servicio en una Región de AWS está diseñado para no afectar al servicio en una región diferente.

Cuando utilice su carga de trabajo en varias regiones, debe tener en cuenta requisitos adicionales. Como los recursos de las distintas regiones están separados y son independientes los unos de los otros, debe duplicar los componentes de la carga de trabajo en cada región. Esto incluye la infraestructura básica, como las VPC, además de los servicios de computación y de datos.

NOTA: Cuando se plantee un diseño multirregional, compruebe que su carga de trabajo sea capaz de ejecutarse en una sola región. Si crea dependencias entre regiones en las que un componente de una región depende de servicios o componentes de una región diferente, puede aumentar el riesgo de fallos y reducir considerablemente su nivel de fiabilidad.

Para facilitar las implementaciones multirregionales y mantener la coherencia, los AWS CloudFormation StackSets pueden replicar toda su infraestructura de AWS en varias regiones. AWS CloudFormation también puede detectar cambios en la configuración e informarle cuando los recursos de AWS de una región no estén sincronizados. Muchos servicios de AWS ofrecen replicación multirregional para activos de carga de trabajo importantes. Por ejemplo, Generador de imágenes de EC2 puede publicar sus imágenes de máquina (AMI) de EC2 después de cada compilación en cada región que utilice. Amazon Elastic Container Registry (ECR) puede replicar las imágenes del contenedor en las regiones seleccionadas.

También debe replicar los datos en cada una de las regiones que elija. Muchos servicios de datos gestionados de AWS ofrecen capacidad de replicación entre regiones, incluidos Amazon S3, Amazon DynamoDB, Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon ElastiCache, y Amazon EFS. Las tablas globales de Amazon DynamoDB aceptan escrituras en cualquier región compatible y replicarán los datos en todas las demás regiones configuradas. Con otros servicios, debe designar una región principal para las escrituras, ya que otras regiones contienen réplicas de solo lectura. Para cada servicio de datos gestionado de AWS que utilice su carga de trabajo, consulte la guía del usuario y la guía para desarrolladores para comprender sus capacidades y limitaciones en varias regiones. Preste especial atención a dónde deben dirigirse las escrituras, a las capacidades y limitaciones de las transacciones, a cómo se lleva a cabo la replicación y a cómo supervisar la sincronización entre regiones.

AWS también ofrece la posibilidad de dirigir el tráfico de solicitudes a sus implementaciones regionales con gran flexibilidad. Por ejemplo, puede configurar sus registros de DNS con Amazon Route 53 para dirigir el tráfico a la región disponible más cercana al usuario. Como alternativa, puede configurar sus registros de DNS en una configuración activa/en espera, en la que designe una región como principal y recurra a una réplica regional solo si la región principal deja de estar en buen estado. Puede configurar las comprobaciones de estado de Route 53 para detectar puntos de conexión en mal estado y realizar una conmutación por error automática y, además, utilizar Amazon Application Recovery Controller (ARC) para proporcionar un control de enrutamiento de alta disponibilidad para redireccionar el tráfico manualmente según sea necesario.

Incluso si decide no operar en varias regiones por motivos de alta disponibilidad, considere la posibilidad de hacerlo en varias regiones como parte de su estrategia de recuperación ante desastres (DR). Si es posible, replique los datos y los componentes de la infraestructura de su carga de trabajo en una configuración de espera activa o piloto ligero en una región secundaria. En este diseño, se replica la infraestructura de referencia de la región principal, como las VPC, los grupos de escalado automático, los orquestadores de contenedores y otros componentes, pero se configuran los componentes de tamaño variable de la región en espera (como el número de instancias de EC2 y réplicas de bases de datos) para que tengan un tamaño operativo mínimo. También puede organizar la replicación continua de los datos desde la región principal a la región en espera. Si se produce un incidente, puede escalar horizontalmente o aumentar los recursos de la región en espera y, después, convertirla en la región principal.

Pasos para la implementación

1. Coordínese con las partes interesadas de la empresa y los expertos en residencia de datos para determinar qué Regiones de AWS se pueden utilizar para alojar sus recursos y datos.

2. Contacte con las partes interesadas tanto a nivel técnico como empresarial para evaluar su carga de trabajo y determinar si sus necesidades de resiliencia pueden satisfacerse mediante un método Multi-AZ (una sola Región de AWS) o si requieren un enfoque multirregional (si se permiten varias regiones). El uso de varias regiones puede proporcionar una mayor disponibilidad, pero también puede suponer una complejidad y un coste adicionales. Tenga en cuenta los siguientes factores en la evaluación:
    a. Objetivos empresariales y requisitos del cliente: ¿cuánto tiempo de inactividad se permite en caso de que se produzca un incidente que afecte a la carga de trabajo en una zona o región de disponibilidad? Evalúe los objetivos de puntos de recuperación, tal como se describe en REL13-BP01. Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos.
    b. Requisitos de recuperación ante desastres (DR): ¿Contra qué tipo de posible desastre quiere asegurarse? Plantéese la posibilidad de que se produzcan pérdidas de datos o una falta de disponibilidad prolongada en diferentes ámbitos de impacto, desde una sola zona de disponibilidad hasta una región completa. Si replica los datos y los recursos en todas las zonas de disponibilidad y una sola zona de disponibilidad sufre un error prolongado, puede recuperar el servicio en otra zona de disponibilidad. Si replica los datos y los recursos en todas las regiones, puede recuperar el servicio en otra región.
3. Implemente los recursos informáticos en varias zonas de disponibilidad.
    a. En la VPC, cree varias subredes en diferentes zonas de disponibilidad. Configure cada una de ellas para que sea lo suficientemente grande como para albergar los recursos necesarios para atender la carga de trabajo, incluso durante un incidente. Para obtener más información, consulte REL02-BP03 Garantía de que la asignación de subredes IP tenga en cuenta la expansión y la disponibilidad.
    b. Si utiliza instancias de Amazon EC2, utilice EC2 Auto Scaling para gestionar las instancias. Especifique las subredes que haya elegido en el paso anterior al crear los grupos de escalado automático.
    c. Si utiliza la computación de AWS Fargate para Amazon ECS o Amazon EKS, seleccione las subredes que haya elegido en el primer paso al crear un servicio de ECS, lanzar una tarea de ECS o crear un perfil de Fargate para EKS.
    d. Si utiliza funciones de AWS Lambda que deben ejecutarse en su VPC, seleccione las subredes que haya elegido en el primer paso al crear la función de Lambda. Para cualquier función que no tenga una configuración de VPC, AWS Lambda administra la disponibilidad automáticamente por usted.

   e. Coloque los directores de tráfico, como los equilibradores de carga, al frente de sus recursos informáticos. Si el equilibrio de carga entre zonas está activado, los equilibradores de carga de aplicaciones de AWS y los equilibradores de carga de red detectan si no es posible contactar con los destinos, como las instancias de EC2 y los contenedores, debido al deterioro de la zona de disponibilidad y redirigen el tráfico hacia los destinos que se encuentran en las zonas de disponibilidad en buen estado. Si inhabilita el equilibrio de carga entre zonas, utilice Amazon Application Recovery Controller (ARC) para proporcionar la capacidad de cambio de zona. Si utiliza un equilibrador de carga de terceros o ha implementado sus propios equilibradores de carga, configúrelos con múltiples interfaces en diferentes zonas de disponibilidad.

4. Replique los datos de la carga de trabajo en varias zonas de disponibilidad.

   a. Si utiliza un servicio de datos gestionado de AWS, como Amazon RDS, Amazon ElastiCache o Amazon FSx, consulte detenidamente su guía del usuario para conocer sus capacidades de replicación y resiliencia de datos. Habilite la replicación entre zonas de disponibilidad y la conmutación por error si es necesario.

   b. Si utiliza servicios de almacenamiento gestionados de AWS, como Amazon S3, Amazon EFS y Amazon FSx, evite utilizar configuraciones Single-AZ o One Zone para datos que requieran una gran durabilidad. Utilice una configuración Multi-AZ para estos servicios. Consulte la guía del usuario del servicio correspondiente para determinar si la replicación Multi-AZ está habilitada de forma predeterminada o si debe habilitarla.

   c. Si ejecuta una base de datos, una cola u otro servicio de almacenamiento autogestionado, organice la replicación en zonas de disponibilidad múltiples según las instrucciones o las prácticas recomendadas de la aplicación. Familiarícese con los procedimientos de conmutación por error de su aplicación.

5. Configure su servicio de DNS para detectar el deterioro de la zona de disponibilidad y redirigir el tráfico a una zona de disponibilidad en buen estado. Amazon Route 53, en combinación con Elastic Load Balancers, puede hacerlo automáticamente. Route 53 también se puede configurar con registros de conmutación por error que utilizan comprobaciones de estado para responder a las consultas únicamente con direcciones IP en buen estado. En el caso de cualquier registro de DNS que se utilice para la conmutación por error, especifique un valor de tiempo de vida (TTL) corto (por ejemplo, 60 segundos o menos) para evitar que el almacenamiento en caché de los registros impida la recuperación (los registros de alias de Route 53 le proporcionan los TTL adecuados).

Pasos adicionales cuando se utilizan varias Regiones de AWS

1. Replique todo el código del sistema operativo (SO) y de la aplicación que utilice su carga de trabajo en las regiones seleccionadas. Replique las imágenes de máquina de Amazon (AMI) que utilizan sus instancias de EC2 si es necesario mediante soluciones como Generador de imágenes de Amazon EC2. Replique las imágenes de contenedores almacenadas en los registros mediante soluciones como la replicación entre regiones de Amazon ECR. Habilite la replicación regional para cualquier bucket de Amazon S3 que se utilice para almacenar los recursos de la aplicación.
2. Implemente sus recursos informáticos y metadatos de configuración (como los parámetros almacenados en el almacén de parámetros de AWS Systems Manager) en varias regiones. Utilice los mismos procedimientos descritos en los pasos anteriores, pero replique la configuración para cada región que utilice para su carga de trabajo. Utilice la infraestructura como soluciones de código, tales como AWS CloudFormation, para reproducir uniformemente las configuraciones entre las regiones. Si utiliza una región secundaria en una configuración piloto sencilla para la recuperación ante desastres, puede reducir la cantidad de recursos informáticos a un valor mínimo para ahorrar costes, con el consiguiente aumento del tiempo de recuperación.
3. Replique los datos de la región principal en las regiones secundarias.
    a. Las tablas globales de Amazon DynamoDB proporcionan réplicas globales de sus datos que se pueden escribir desde cualquier región compatible. Con otros servicios de datos gestionados por AWS, como Amazon RDS, Amazon Aurora y Amazon Elasticache, se designa una región principal (lectura/escritura) y regiones de réplica (solo lectura). Consulte las guías de usuario y del desarrollador de los servicios respectivos para obtener más información sobre la replicación regional.
    b. Si ejecuta una base de datos autogestionada, organice la replicación en varias regiones siguiendo las instrucciones o las prácticas recomendadas de la aplicación. Familiarícese con los procedimientos de conmutación por error de su aplicación.
    c. Si la carga de trabajo utiliza AWS EventBridge, tal vez tenga que reenviar los eventos seleccionados de su región principal a las regiones secundarias. Para ello, especifique los buses de eventos de sus regiones secundarias como destinos de los eventos coincidentes en su región principal.
4. Piense si quiere utilizar claves de cifrado idénticas en todas las regiones y en qué medida. Un enfoque típico que equilibra la seguridad y la facilidad de uso consiste en utilizar claves de ámbito regional para los datos y la autenticación regionales y locales y utilizar claves de ámbito internacional para cifrar los datos que se replican entre distintas regiones. AWS Key Management Service (KMS) admite claves de varias regiones para distribuir y proteger de forma segura las claves compartidas entre las regiones.

5. Puede usar AWS Global Accelerator para mejorar la disponibilidad de la aplicación, que dirige el tráfico a regiones que contengan puntos de conexión en buen estado.

Recursos

Prácticas recomendadas relacionadas:

- REL02-BP03 Garantía de que la asignación de subredes IP tenga en cuenta la expansión y la disponibilidad
- REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal
- REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos

Documentos relacionados:

- AWS Infraestructura global de
- Documento técnico: AWS Fault Isolation Boundaries
- Resiliencia en Amazon EC2 Auto Scaling
- Amazon EC2 Auto Scaling: Ejemplo: Distribuir instancias entre zonas de disponibilidad
- Cómo funciona Generador de Imágenes de EC2
- Cómo coloca Amazon ECS las tareas en las instancias de contenedor (incluido Fargate)
- Resiliencia en AWS Lambda
- Amazon S3: información general de la replicación de objetos
- Replicación de imágenes privadas en Amazon ECR
- Tablas globales: replicación en varias regiones para DynamoDB
- Amazon Elasticache para Redis OSS: replicación en Regiones de AWS con almacenes de datos globales
- Resiliencia en Amazon RDS
- Uso de bases de datos globales de Amazon Aurora
- AWS Guía para desarrolladores de Global Accelerator
- Multi-Region keys in AWS KMS
- Amazon Route 53: configuración de la recuperación ante errores a nivel de DNS

- Guía para desarrolladores del Controlador de recuperación de aplicaciones de Amazon (ARC)
- Sending and receiving Amazon EventBridge events between Regiones de AWS
- Serie de blogs Creating a Multi-Region Application with AWS Services
- Arquitectura de recuperación de desastres (DR) en AWS, parte I: estrategias de recuperación en la nube
- Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby

Videos relacionados:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications
- AWS re:Invent 2019: Innovation and operation of the AWS global network infrastructure

REL10-BP02 Automatización de la recuperación de los componentes restringidos a una sola ubicación

Si los componentes de la carga de trabajo solo se pueden ejecutar en una zona de disponibilidad o en el centro de datos en las instalaciones, implemente la capacidad de volver a crear la carga de trabajo de acuerdo con los objetivos de recuperación definidos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si la práctica recomendada de implementar la carga de trabajo en varias ubicaciones no es posible por limitaciones tecnológicas, debe implementar una ruta alternativa hacia la resiliencia. Debe automatizar la capacidad de recrear la infraestructura necesaria, volver a implementar las aplicaciones y volver a crear los datos necesarios para estos casos.

Por ejemplo, Amazon EMR lanza todos los nodos de un clúster determinado en la misma zona de disponibilidad, porque la ejecución de un clúster en la misma zona mejora el rendimiento de los flujos de trabajo, ya que ofrece una velocidad de acceso a los datos más alta. Si este componente resulta necesario para la resiliencia de la carga de trabajo, debe tener una forma de volver a implementar el clúster y sus datos. Además, para Amazon EMR, debería aprovisionar la redundancia de formas diferentes al uso de varias zonas de disponibilidad. Puede aprovisionar varios nodos. Con el sistema de archivos de EMR (EMRFS), los datos de EMR se pueden almacenar en Amazon S3, que a su vez se puede replicar en varias zonas de disponibilidad o Regiones de AWS.

De modo similar, en el caso de Amazon Redshift, el clúster se aprovisiona de forma predeterminada en una zona de disponibilidad seleccionada al azar dentro de la Región de AWS que haya seleccionado. Todos los nodos del clúster se aprovisionan en la misma zona.

Para cargas de trabajo basadas en servidores con estado implementadas en un centro de datos en las instalaciones, puede utilizar AWS Elastic Disaster Recovery para proteger sus cargas de trabajo en AWS. Si ya se aloja en AWS, puede usar la Recuperación de desastres elástica para proteger la carga de trabajo en una región o zona de disponibilidad alternativa. La Recuperación de desastres elástica utiliza la replicación continua en el nivel de bloque en un espacio de almacenamiento ligero para proporcionar una recuperación rápida y fiable de las aplicaciones en las instalaciones y basadas en la nube.

Pasos para la implementación

1. Implemente la autorrecuperación. Implemente sus instancias o contenedores con escalado automático siempre que sea posible. Si no puede usar el escalado automático, utilice la recuperación automática para instancias de EC2 o implemente la automatización de autorrecuperación basada en eventos de ciclo de vida del contenedor de Amazon EC2 o ECS.
   - Utilice grupos de Amazon EC2 Auto Scaling para instancias y cargas de trabajo de contenedor que no tengan requisitos de una sola dirección IP de instancia, dirección IP privada, dirección IP elástica y metadatos de instancia.
     - Los datos de usuario de la plantilla de lanzamiento se pueden usar para implementar una automatización que pueda solucionar la mayoría de las cargas de trabajo.
   - Utilice la recuperación automática de instancias de Amazon EC2 para cargas de trabajo que requieran una única dirección ID de instancia, dirección IP privada, dirección IP elástica y metadatos de instancia.
     - La recuperación automática enviará alertas de estado de recuperación a un tema de SNS cuando se detecte un error en la instancia.
   - Utilice los eventos del ciclo de vida de la instancia de Amazon EC2 o los eventos de Amazon ECS para automatizar la autorrecuperación cuando no se pueda utilizar el escalado automático ni la recuperación de EC2.
     - Utilice los eventos para invocar la automatización que reparará su componente de acuerdo con la lógica de proceso que necesita.
   - Proteja las cargas de trabajo con estado que están limitadas a una única ubicación con AWS Elastic Disaster Recovery.

Recursos

Documentos relacionados:

- Eventos de Amazon ECS
- Enlaces de ciclo de vida de Amazon EC2 Auto Scaling
- Recuperación de instancias.
- Escalado automático de su servicio
- What Is Amazon EC2 Auto Scaling?
- AWS Elastic Disaster Recovery

REL10-BP03 Uso de arquitecturas herméticas para limitar el alcance del impacto

La implementación de arquitecturas herméticas (también conocidas como arquitecturas basadas en celdas) restringe el efecto del fallo dentro de una carga de trabajo a un número limitado de componentes.

Resultado deseado: una arquitectura basada en celdas utiliza varias instancias aisladas de una carga de trabajo, donde cada instancia se conoce como celda. Cada celda es independiente, no comparte estado con otras celdas y gestiona un subconjunto de las solicitudes de la carga de trabajo global. Esto reduce la posible repercusión de un error, como una actualización de software incorrecta, en una celda individual y en las solicitudes que está procesando. Si una carga de trabajo utiliza 10 celdas para atender 100 solicitudes cuando se produce un error, el 90 % del total de las solicitudes no se verá afectado por el error.

Patrones comunes de uso no recomendados:

- Permitir que las celdas crezcan sin límites.
- Aplicar actualizaciones o implementaciones de código a todas las celdas al mismo tiempo.
- Compartir estado o componentes entre celdas (a excepción de la capa de enrutador).
- Agregar lógica compleja de negocio o de enrutamiento a la capa de enrutador.
- No minimizar las interacciones entre celdas.

Beneficios de establecer esta práctica recomendada: con las arquitecturas basadas en celdas, muchos tipos de fallos comunes se encuentran dentro de la propia celda, lo que proporciona un

aislamiento adicional de los fallos. Estos límites de los errores pueden favorecer la resiliencia frente a tipos de errores que, de otro modo, serían difíciles de contener, como implementaciones de código fallidas o solicitudes dañadas o que invocan un modo de error específico (también conocidas como solicitudes de píldora venenosa).

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

En un barco, los mamparos garantizan que una brecha en el casco quede contenida en una sola sección del casco. En los sistemas complejos, este modelo de contención suele imitarse para permitir el aislamiento de errores. Los límites aislados de los errores restringen el efecto de un error en una carga de trabajo a un número limitado de componentes. Los componentes que se encuentran fuera del límite no se ven afectados por el error. Al usar múltiples límites aislados de errores, puede limitar el impacto en su carga de trabajo. En AWS, los clientes pueden utilizar varias zonas y regiones de disponibilidad para proporcionar aislamiento de errores, pero el concepto de aislamiento de errores también puede extenderse a la arquitectura de su carga de trabajo.

La carga de trabajo global se divide en celdas mediante una clave de partición. Esta clave tiene que alinearse con la corriente del servicio o con la forma natural en que la carga de trabajo de un servicio puede subdividirse con mínimas interacciones entre celdas. Algunos ejemplos de claves de partición son el ID de cliente, el ID de recurso o cualquier otro parámetro fácilmente accesible en la mayoría de las llamadas a la API. Una capa de enrutador de celdas distribuye las solicitudes a celdas individuales en función de la clave de partición y presenta un único punto de conexión a los clientes.

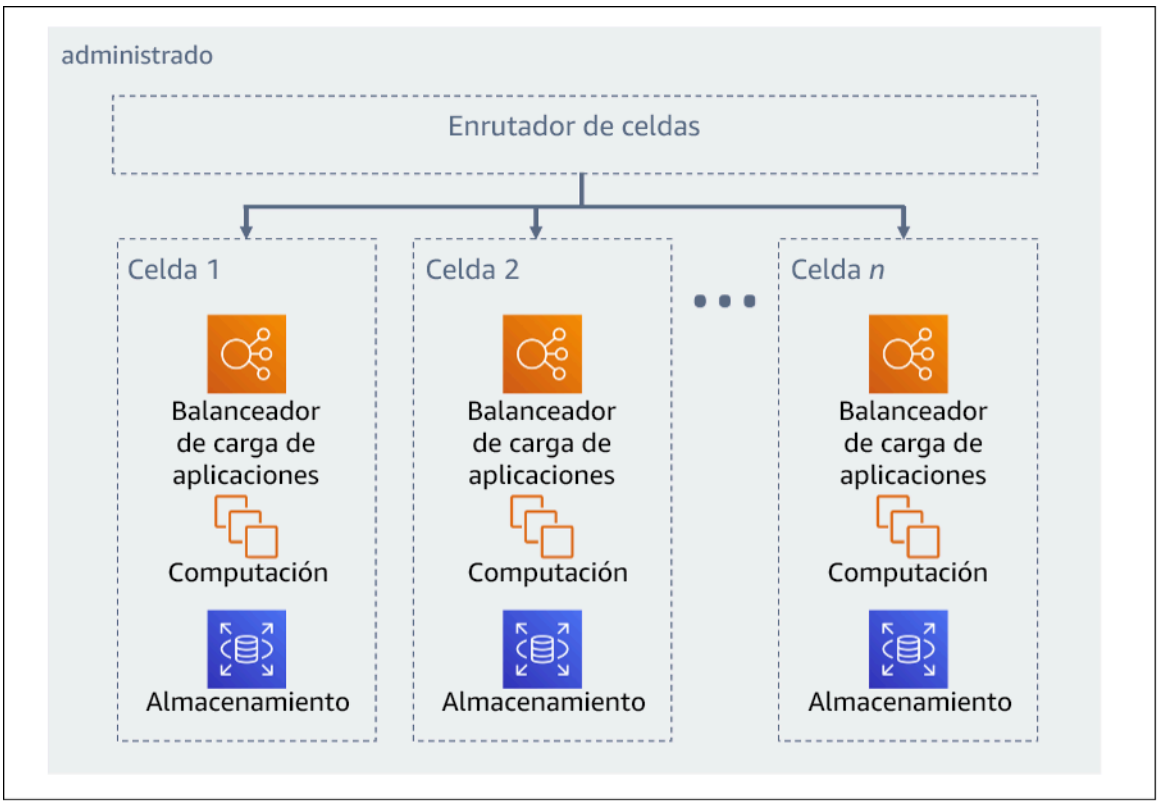



Figura 11: Arquitectura basada en celdas

Pasos para la implementación

Al diseñar una arquitectura basada en celdas, hay que tener en cuenta varias consideraciones de diseño:

1. Clave de partición: se debe tener especial cuidado al elegir la clave de partición.
    - Debe alinearse con la corriente del servicio o con la forma natural en que la carga de trabajo de un servicio puede subdividirse con mínimas interacciones entre celdas. Algunos ejemplos son `customer ID` o `resource ID`.
    - La clave de partición debe estar disponible en todas las solicitudes, ya sea de modo directo o de una manera que se pueda inferir con facilidad de forma determinista por otros parámetros.
2. Asignación persistente de celdas: los servicios ascendentes solo deberían interactuar con una única celda durante el ciclo de vida de sus recursos.
    - Según la carga de trabajo, puede ser necesaria una estrategia de migración de celda para migrar datos de una celda a otra. Un posible escenario en el que puede ser precisa una migración de celda es si un usuario o recurso concreto de la carga de trabajo crece demasiado y requiere una celda dedicada.

- Las celdas no deben compartir estados ni componentes entre ellas.
- En consecuencia, las interacciones entre celdas deben evitarse o mantenerse al mínimo, ya que dichas interacciones crean dependencias entre las celdas y, por lo tanto, disminuyen las ventajas en el aislamiento de errores.

3. Capa de enrutador: la capa de enrutador es un componente compartido entre las celdas y, por lo tanto, no puede seguir la misma estrategia de compartimentación que las celdas.
   - Se recomienda que la capa de enrutador distribuya las solicitudes a las celdas individuales mediante un algoritmo de asignación de particiones de una manera eficiente a nivel computacional, como la combinación de funciones hash criptográficas y aritmética modular para asignar claves de partición a las celdas.
   - Para evitar impactos multicelda, la capa de enrutador debe ser lo más simple y escalable horizontalmente posible, lo que requiere evitar una lógica de negocio compleja dentro de esta capa. Esto tiene la ventaja agregada de facilitar la comprensión de su comportamiento esperado en todo momento, lo que permite una comprobabilidad exhaustiva. Como explica Colm MacCárthaigh en Reliability, constant work, and a good cup of coffee, los diseños simples y los patrones de trabajo constantes producen sistemas fiables y reducen la antifragilidad.
4. Tamaño de la celda: las celdas deben tener un tamaño máximo y no debe permitirse que lo superen.
   - Para determinar el tamaño máximo, se deben llevar a cabo pruebas exhaustivas hasta que se alcancen puntos de ruptura y se establezcan márgenes de funcionamiento seguros. Para obtener más detalles sobre cómo implementar prácticas de prueba, consulte REL07-BP04 Pruebas en su carga de trabajo
   - La carga de trabajo global crecerá a medida que se agreguen celdas adicionales, lo que permite escalar la carga de trabajo con los aumentos de la demanda.
5. Estrategias de varias zonas de disponibilidad o de varias regiones: se deben aprovechar las diversas capas de resiliencia para protegerse contra diferentes dominios de error.
   - Para obtener resiliencia, debe utilizar un enfoque que cree capas de defensa. Una capa protege de las interrupciones más pequeñas y frecuentes mediante la creación de una arquitectura de alta disponibilidad con múltiples AZ. Otra capa de defensa está pensada para proteger de eventos poco frecuentes, como las catástrofes naturales generalizadas y las interrupciones a nivel regional. Esta segunda capa implica la arquitectura de su aplicación para que abarque múltiples Regiones de AWS. Implementar una estrategia multirregión para la carga de trabajo ayuda a protegerla de catástrofes naturales generalizadas que afecten a una región geográfica amplia de un país o de errores técnicos de alcance regional. Tenga en cuenta que implementar

una arquitectura multirregional puede ser significativamente complejo y no suele ser necesario para la mayoría de las cargas de trabajo. Para obtener más información, consulte REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones.

6. Implementación de código: debería preferirse una estrategia de implementación de código escalonada en lugar de implementar cambios de código en todas las celdas al mismo tiempo.
   - Esto ayuda a minimizar posibles errores en numerosas celdas provocados por una implementación incorrecta o a un error humano. Para obtener más detalles, consulte Automatización de implementaciones seguras y sin intervención.

Recursos

Prácticas recomendadas relacionadas:

- REL07-BP04 Pruebas en su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones

Documentos relacionados:

- Reliability, constant work, and a good cup of coffee
- AWS and Compartmentalization
- Aislamiento de las cargas de trabajo a través de la fragmentación aleatoria
- Automatización de implementaciones seguras y sin intervención

Videos relacionados:

- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small
- AWS re:Invent 2018: How AWS Minimizes the Blast Radius of Failures (ARC338)
- Shuffle-sharding: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)
- AWS Summit ANZ 2021 - Everything fails, all the time: Designing for resilience

## REL 11. ¿Cómo diseña su carga de trabajo para que soporte los errores de los componentes?

Las cargas de trabajo con un requisito de alta disponibilidad y un tiempo de recuperación (MTTR) bajo deben diseñarse para que sean resilientes.

Prácticas recomendadas

- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores
- REL11-BP02 Conmutación por error a recursos en buen estado
- REL11-BP03 Automatización de la reparación en todas las capas
- REL11-BP04 Confianza en el plano de datos y no en el plano de control durante la recuperación
- REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal
- REL11-BP06 Envío de notificaciones cuando los eventos afecten a la disponibilidad
- REL11-BP07 Diseño de su producto para cumplir objetivos de disponibilidad y acuerdos de nivel de servicio (SLA) de tiempo de actividad

### REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

Supervise continuamente el estado de las cargas de trabajo para que usted y los sistemas automatizados sepan cuándo se produce degradaciones o errores en cuanto ocurran. Supervise los indicadores clave de rendimiento (KPI) en función del valor empresarial.

Todos los mecanismos de recuperación y corrección deben comenzar por la capacidad de detectar problemas rápidamente. Los fallos técnicos deberían detectarse en primer lugar para poder resolverse. Sin embargo, la disponibilidad se basa en la capacidad de su carga de trabajo para ofrecer valor empresarial, de modo que los indicadores clave de rendimiento (KPI) que midan esto tienen que formar parte de su estrategia de detección y corrección.

Resultado deseado: los componentes esenciales de una carga de trabajo se supervisan de forma independiente para detectar los errores en el momento y el lugar en que se producen y alertar sobre ellos.

Patrones comunes de uso no recomendados:

- No se han configurado alarmas, por lo que las interrupciones se producen sin notificación.
- Existen alarmas, pero en umbrales que no proporcionan el tiempo necesario para reaccionar.

- No se recopilan métricas con la suficiente regularidad para satisfacer el objetivo de tiempo de recuperación (RTO).
- Solo se supervisan activamente las interfaces de la carga de trabajo orientadas a los clientes.
- Solo se recopilan métricas técnicas, no métricas de funciones empresariales.
- No hay métricas que midan la experiencia del usuario con la carga de trabajo.
- Se crean demasiadas supervisiones.

Beneficios de establecer esta práctica recomendada: una supervisión adecuada de todas las capas le permite reducir el tiempo de recuperación al reducirse el tiempo de detección.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Identifique todas las cargas de trabajo que se revisarán para su supervisión. Una vez que haya identificado todos los componentes de la carga de trabajo que deberán supervisarse, tendrá que determinar el intervalo de supervisión. El intervalo de supervisión tendrá un impacto directo en la rapidez con la que se puede iniciar la recuperación en función del tiempo que se tarde en detectar un error. El tiempo medio de detección (MTTD) es el tiempo transcurrido entre la aparición de un error y el inicio de las operaciones de reparación. La lista de servicios debe ser amplia y completa.

La supervisión debe cubrir todas las capas de la pila de aplicaciones, incluidas la aplicación, la plataforma, la infraestructura y la red.

Su estrategia de supervisión debe considerar el impacto de los errores grises. Para obtener más información sobre los errores grises, consulte Gray failures en el documento técnico Advanced Multi-AZ Resilience Patterns.

Pasos para la implementación

- El intervalo de supervisión depende de la rapidez con la que deba recuperarse. El tiempo de recuperación depende del tiempo que tarde la recuperación, por lo que debe determinar la frecuencia de recopilación teniendo en cuenta este tiempo y el objetivo de tiempo de recuperación (RTO).
- Configure la supervisión detallada de los componentes y los servicios administrados.
  - Determine si son necesarios la supervisión detallada de las instancias de EC2 y el escalado automático. La supervisión detallada proporciona métricas en intervalos de un minuto y la supervisión predeterminada proporciona métricas en intervalos de cinco minutos.

  - Determine si se necesita la supervisión mejorada de RDS. La supervisión mejorada usa un agente en las instancias de RDS para obtener información útil sobre los diferentes procesos o subprocesos.
  - Determine los requisitos de supervisión de los componentes sin servidor cruciales para Lambda, API Gateway, Amazon EKS, Amazon ECS y todos los tipos de equilibradores de carga.
  - Determine los requisitos de supervisión de los componentes de almacenamiento para Amazon S3, Amazon FSx, Amazon EFS y Amazon EBS.
- Cree métricas personalizadas para medir los indicadores clave de rendimiento (KPI) del negocio. Las cargas de trabajo implementan funciones empresariales clave, que deben usarse como KPI para ayudar a identificar cuándo se produce un problema indirecto.
- Supervise la experiencia del usuario para detectar errores mediante canarios del usuario. Las pruebas de transacciones sintéticas (también denominadas “pruebas canario”, que no deben confundirse con las implementaciones canario) que puedan ejecutar y simular el comportamiento de los clientes son uno de los procesos de prueba más importantes. Ejecute estas pruebas constantemente en los puntos de conexión de las cargas de trabajo desde distintas ubicaciones remotas.
- Cree métricas personalizadas que controlen la experiencia del usuario. Si puede instrumentar la experiencia del cliente, puede determinar cuándo se degrada la experiencia del cliente.
- Defina alarmas para detectar cuándo alguna parte de la carga de trabajo no funciona correctamente y para indicar cuándo escalar automáticamente los recursos. Las alarmas pueden mostrarse visualmente en paneles, enviar alertas a través de Amazon SNS o por correo electrónico y trabajar con escalado automático para escalar o reducir verticalmente los recursos de la carga de trabajo.
- Cree paneles para visualizar las métricas. Se pueden usar paneles para visualizar las tendencias, los valores atípicos y otros indicadores de problemas potenciales, o para proporcionar una indicación de problemas que tal vez le convenga investigar.
- Cree una supervisión de rastreo distribuida para sus servicios. Con la supervisión distribuida, podrá saber cómo se comporta su aplicación y sus servicios subyacentes para identificar y resolver la causa raíz de los problemas y errores de rendimiento.
- Cree paneles de sistemas de supervisión (mediante CloudWatch o X-Ray) y recopilaciones de datos en una región y una cuenta independientes.
- Manténgase informado sobre las degradaciones del servicio con AWS Health. Cree notificaciones de eventos de AWS Health adecuados para su propósito para los canales de correo electrónico y

chat a través de AWS User Notifications e intégrelas mediante programación con las herramientas de supervisión y alertas a través de Amazon EventBridge.

Recursos

Prácticas recomendadas relacionadas:

- Availability Definition
- REL11-BP06 Envío de notificaciones cuando los eventos afecten a la disponibilidad

Documentos relacionados:

- Amazon CloudWatch Synthetics le permite crear canarios de usuario
- Activar o desactivar el monitoreo detallado para las instancias
- Supervisión mejorada
- Monitoring Your Auto Scaling Groups and Instances Using Amazon CloudWatch
- Publicar métricas personalizadas
- Uso de las alarmas de Amazon CloudWatch
- Uso de paneles de CloudWatch
- Uso de paneles de CloudWatch entre regiones y entre cuentas
- Uso del rastreo de X-Ray entre regiones y entre cuentas
- Understanding availability

Videos relacionados:

- Mitigating gray failures

Ejemplos relacionados:

- One Observability Workshop: Explore X-Ray

Herramientas relacionadas:

- CloudWatch
- CloudWatch X-Ray

REL11-BP02 Conmutación por error a recursos en buen estado

Si un recurso fallara, los recursos en buen estado deberían seguir atendiendo las solicitudes. Para problemas de ubicación (como zonas de disponibilidad o Región de AWS), asegúrese de que disponga de sistemas para conmutar por error a recursos en buen estado en ubicaciones sin problemas.

Al diseñar un servicio, distribuya la carga entre los recursos, las zonas de disponibilidad o las regiones. De esta manera, el error de un recurso individual o el deterioro puede mitigarse al desplazar el tráfico a los recursos restantes en buen estado. Tenga en cuenta cómo se descubren los servicios y cómo se enruta a ellos en caso de que se produzca un error.

Tenga en cuenta la recuperación de errores al diseñar sus servicios. En AWS, diseñamos servicios para minimizar el tiempo de recuperación de los errores y el impacto en los datos. Nuestros servicios utilizan principalmente almacenes de datos que confirman las solicitudes solo después de que se almacenen de forma duradera en varias réplicas en una región. Se han diseñado para utilizar el aislamiento basado en celdas y el aislamiento de errores que proporcionan las zonas de disponibilidad. Utilizamos ampliamente la automatización en nuestros procedimientos operativos. También optimizamos nuestra funcionalidad de reemplazo y reinicio para recuperarnos rápidamente de las interrupciones.

Los patrones y diseños que permiten la conmutación por error varían para cada servicio de plataforma de AWS. Muchos servicios administrados nativos de AWS son zonas de disponibilidad múltiples de forma nativa (como Lambda o API Gateway). Otros servicios de AWS (como EC2 y EKS) requieren diseños específicos de las prácticas recomendadas para admitir la conmutación por error de los recursos o el almacenamiento de datos en las AZ.

La supervisión debe configurarse para que compruebe que el recurso de conmutación por error esté en buen estado, hacer un seguimiento del progreso de los recursos de conmutación por error y supervisar la recuperación de los procesos empresariales.

Resultado deseado: los sistemas son capaces de utilizar nuevos recursos de forma automática o manual para recuperarse de la degradación.

Patrones comunes de uso no recomendados:

- La planificación de errores no forma parte de la fase de planificación y diseño.
- No se establecen el RTO ni el RPO.
- Supervisión insuficiente para detectar recursos defectuosos.

- Aislamiento adecuado de los dominios de error.
- No se considera la conmutación por error multirregional.
- La detección de errores es demasiado sensible o agresiva a la hora de decidir efectuar una conmutación por error.
- No probar ni validar el diseño de la conmutación por error.
- Llevar a cabo la automatización de la recuperación automática, pero no notificar que se necesitaba una reparación.
- Ausencia de un periodo de amortiguación para evitar que la conmutación por error se lleve a cabo demasiado pronto.

Beneficios de establecer esta práctica recomendada: con una degradación uniforme y una recuperación rápida, puede crear sistemas más resilientes que mantengan la fiabilidad cuando se producen errores.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los servicios de AWS, como Elastic Load Balancing y Amazon EC2 Auto Scaling, ayudan a distribuir la carga entre los recursos y las zonas de disponibilidad. Por lo tanto, el error de un recurso individual (como una instancia de EC2) o el deterioro de una zona de disponibilidad puede mitigarse si se desplaza el tráfico a los recursos restantes en buen estado.

Para las cargas de trabajo multirregionales, los diseños son más complicados. Por ejemplo, las réplicas de lectura entre regiones le permiten implementar sus datos en varias Regiones de AWS. Sin embargo, la conmutación por error sigue siendo necesaria para convertir la réplica de lectura en principal y, a continuación, dirigir el tráfico al nuevo punto de conexión. Amazon Route 53, el Controlador de recuperación de aplicaciones de (ARC), Amazon CloudFront y AWS Global Accelerator pueden ayudar a enrutar el tráfico en las Regiones de AWS.

Los servicios de AWS, como Amazon S3, Lambda, API Gateway, Amazon SQS, Amazon SNS, Amazon SES, Amazon Pinpoint, Amazon ECR, AWS Gestor de certificados, EventBridge o Amazon DynamoDB, se implementan automáticamente en varias zonas de disponibilidad mediante AWS. En caso de error, estos servicios de AWS dirigen automáticamente el tráfico a ubicaciones en buen estado. Los datos se almacenan de forma redundante en varias zonas de disponibilidad y siguen estando disponibles.

Para Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon EKS o Amazon ECS, Multi-AZ es una opción de configuración. AWS puede dirigir el tráfico a la instancia en buen estado si se inicia la conmutación por error. Esta acción de conmutación por error puede llevarla a cabo AWS o según lo requiera el cliente

Para las instancias de Amazon EC2, Amazon Redshift, las tareas de Amazon ECS o los pods de Amazon EKS, elige en qué zonas de disponibilidad deben implementarse. En algunos diseños, Elastic Load Balancing proporciona la solución para detectar instancias en zonas que no están en buen estado y dirigir el tráfico a las que sí están en buen estado. Elastic Load Balancing también puede dirigir el tráfico a componentes de su centro de datos en las instalaciones.

En cuanto a la conmutación por error del tráfico multirregional, el reenrutamiento puede utilizar Amazon Route 53, el Controlador de recuperación de aplicaciones de Amazon, AWS Global Accelerator, DNS privado de Route 53 para VPC o CloudFront para proporcionar una forma de definir dominios de Internet y asignar políticas de enrutamiento, incluidas comprobaciones de estado, para enrutar el tráfico a regiones en buen estado. AWS Global Accelerator proporciona direcciones IP estáticas que actúan como punto de entrada fijo a su aplicación; a continuación, se enrutan a los puntos de conexión de las Regiones de AWS que elija, mediante la red global de AWS en lugar de Internet para mejorar el rendimiento y la fiabilidad.

Pasos para la implementación

- Cree diseños de conmutación por error para todas las aplicaciones y servicios pertinentes. Aísle cada componente de la arquitectura y cree diseños de conmutación por error que satisfagan el RTO y el RPO de cada componente.
- Configure entornos inferiores (como los de desarrollo o prueba) con todos los servicios que sean necesarios para tener un plan de conmutación por error. Implemente las soluciones mediante la infraestructura como código (IaC) para garantizar la repetibilidad.
- Configure un sitio de recuperación, como una segunda región, para implementar y probar los diseños de conmutación por error. Si fuera necesario, los recursos para las pruebas se pueden configurar temporalmente para limitar los costos adicionales.
- Determine qué planes de conmutación por error se automatizan mediante AWS, cuáles pueden automatizarse mediante un proceso de DevOps y cuáles pueden ser manuales. Documente y mida el RTO y el RPO de cada servicio.
- Cree un manual de estrategias de conmutación por error e incluya todos los pasos de la conmutación por error de cada recurso, aplicación y servicio.

- Cree un manual de estrategias de conmutación por recuperación e incluya todos los pasos de la conmutación por recuperación (con plazos) de cada recurso, aplicación y servicio.
- Cree un plan para iniciar y ensayar el manual de estrategias. Utilice simulaciones y pruebas de caos para poner a prueba los pasos del manual de estrategias y la automatización.
- Para problemas de ubicación (como zonas de disponibilidad o Región de AWS), asegúrese de que disponga de sistemas para conmutar por error a recursos en buen estado en ubicaciones sin problemas. Compruebe la cuota, los niveles de escalado automático y los recursos en ejecución antes de llevar a cabo la prueba de conmutación por error.

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- REL13: Plan para DR
- REL10: Uso del aislamiento de errores para proteger la carga de trabajo

Documentos relacionados:

- Setting RTO and RPO Targets
- Failover using Route 53 Weighted routing
- Disaster Recovery with Amazon Application Recovery Controller
- EC2 with autoscaling
- EC2 Deployments - Multi-AZ
- ECS Deployments - Multi-AZ
- Switch traffic using Amazon Application Recovery Controller
- Lambda con un equilibrador de carga de aplicación y conmutación por error
- ACM Replication and Failover
- Parameter Store Replication and Failover
- ECR cross region replication and Failover
- Secrets manager cross region replication configuration
- Enable cross region replication for EFS and Failover
- EFS Cross Region Replication and Failover
- Conmutación por error de red

- S3 Endpoint failover using MRAP
- Creación de replicación entre regiones para S3
- Guidance for Cross Region Failover and Graceful Failback on AWS
- Failover using multi-region global accelerator
- Failover with DRS

Ejemplos relacionados:

- Disaster Recovery on AWS
- Elastic Disaster Recovery on AWS

REL11-BP03 Automatización de la reparación en todas las capas

Cuando se detecte un error, utilice las funciones automatizadas para tomar medidas correctivas. Las degradaciones pueden repararse automáticamente a través de mecanismos de servicio interno o requerir que los recursos se reinicien o eliminen a través de medidas de corrección.

Para las aplicaciones autoadministradas y la reparación entre regiones, los diseños de recuperación y los procesos de reparación automatizados se pueden extraer de las prácticas recomendadas existentes.

La capacidad de reiniciar o eliminar un recurso es una herramienta importante para corregir los errores. Una práctica recomendada es convertir los servicios en servicios sin estado siempre que sea posible. Esto evita la pérdida de datos o disponibilidad tras el reinicio del recurso. En la nube, puede (y generalmente debería) sustituir todo el recurso (por ejemplo, la instancia de computación o la función sin servidor) como parte del reinicio. El reinicio en sí es una forma sencilla y fiable de recuperarse de un error. En las cargas de trabajo ocurren muchos tipos de errores diferentes. Los errores pueden ocurrir en el hardware, el software, las comunicaciones y el funcionamiento.

El reinicio o el reintento también se aplican a las solicitudes de red. Se aplica el mismo enfoque de recuperación tanto a un tiempo de espera de la red como a un error en la dependencia, en el que la dependencia devuelve un error. Ambos eventos tienen un efecto similar en el sistema, por lo que, en lugar de intentar convertir cada uno en un caso especial, se aplicaría una estrategia similar de reintento con retroceso exponencial y fluctuación. La capacidad de reiniciar es un mecanismo de recuperación que aparece en la computación orientada a la recuperación y en las arquitecturas de clústeres de alta disponibilidad.

Resultado deseado: se llevan a cabo medidas automatizadas para corregir la detección de un error.

Patrones comunes de uso no recomendados:

- Aprovisionar recursos sin escalado automático.
- Implementar las aplicaciones en instancias o contenedores individualmente.
- Implementar aplicaciones que no se pueden implementar en varias ubicaciones sin usar la recuperación automática.
- Reparar manualmente las aplicaciones que el escalado automático y la recuperación automática no pueden reparar.
- No hay automatización de las bases de datos de conmutación por error.
- Carencia de métodos automatizados para redirigir el tráfico a nuevos puntos de conexión.
- No hay replicación del almacenamiento.

Beneficios de establecer esta práctica recomendada: la reparación automática puede reducir el tiempo medio de recuperación y mejorar la disponibilidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los diseños de Amazon EKS u otros servicios de Kubernetes deben incluir conjuntos de réplicas o con estado mínimo y máximo y el tamaño mínimo del clúster y los grupos de nodos. Estos mecanismos proporcionan una cantidad mínima de recursos de procesamiento disponibles de forma continua y, al mismo tiempo, corrigen automáticamente cualquier error mediante el plano de control de Kubernetes.

Los patrones de diseño a los que se accede a través de un equilibrador de carga mediante clústeres de computación deben utilizar los grupos de escalado automático. Elastic Load Balancing (ELB) distribuye automáticamente el tráfico de aplicaciones entrante entre varios destinos y dispositivos virtuales en una o más zonas de disponibilidad (AZ).

Los diseños basados en computación en clúster que no utilizan el equilibrio de carga deben tener un diseño de tamaño que dé cabida a la pérdida de al menos un nodo. Esto permitirá que el servicio siga funcionando con una capacidad potencialmente reducida mientras recupera un nuevo nodo. Algunos servicios son Mongo, el Acelerador de DynamoDB, Amazon Redshift, Amazon EMR, Cassandra, Kafka, MSK-EC2, Couchbase, ELK y Amazon OpenSearch Service. Muchos de estos

servicios se pueden diseñar con características adicionales de autorreparación. Algunas tecnologías de clústeres deben generar una alerta tras la pérdida de un nodo, lo que desencadena un flujo de trabajo automático o manual para recrear un nuevo nodo. Este flujo de trabajo se puede automatizar con AWS Systems Manager para corregir los problemas rápidamente.

Se puede usar Amazon EventBridge para supervisar y filtrar los eventos, como las alarmas de CloudWatch o cambios en el estado en otros servicios de AWS. En función de la información del evento, se puede invocar a AWS Lambda, Automatización de Systems Manager u otros destinos para ejecutar una lógica de corrección personalizada en la carga de trabajo. Amazon EC2 Auto Scaling se puede configurar para comprobar el estado de la instancia de EC2. Si el estado de la instancia es cualquier otro estado distinto de En ejecución o si el estado del sistema es Dañado, Amazon EC2 Auto Scaling considera que la instancia tiene un estado incorrecto y lanza una instancia de reemplazo. Para sustituciones a gran escala (como la pérdida de toda una zona de disponibilidad), se prefiere la estabilidad estática para la alta disponibilidad.

Pasos para la implementación

- Use grupos de escalado automático para implementar niveles en una carga de trabajo. El escalado automático puede llevar a cabo una reparación automática de aplicaciones sin estado y agregar o eliminar capacidad.
- En el caso de las instancias de computación indicadas anteriormente, utilice el equilibrio de carga y elija el tipo de equilibrador de carga adecuado.
- Considere la reparación para Amazon RDS. Con las instancias en espera, defina la configuración de la conmutación por error automática en la instancia en espera. Para la réplica de lectura de Amazon RDS, se requiere un flujo de trabajo automatizado para convertir una réplica de lectura en principal.
- Implemente la recuperación automática en instancias de EC2 que tengan aplicaciones implementadas que no se puedan implementar en varias ubicaciones y puedan tolerar el reinicio tras un error. La recuperación automática se puede usar para reemplazar hardware defectuoso y reiniciar la instancia cuando la aplicación no se puede implementar en varias ubicaciones. Los metadatos de la instancia y las direcciones IP asociadas se conservan, así como los volúmenes de EBS y los puntos de montaje en Amazon Elastic File System o File Systems para Lustre y Windows. Con AWS OpsWorks, puede configurar la reparación automática de las instancias de EC2 en el nivel de capa.
- Implemente la recuperación automática mediante AWS Step Functions y AWS Lambda cuando no pueda usar el escalado automático ni la recuperación automática o cuando la recuperación automática produzca un error. Cuando no pueda usar el escalado automático ni la recuperación

automática, o esta produzca un error, puede automatizar la reparación con AWS Step Functions y AWS Lambda.

- Se puede usar Amazon EventBridge para supervisar y filtrar los eventos, como las alarmas de CloudWatch o cambios en el estado en otros servicios de AWS. En función de la información del evento, se puede invocar AWS Lambda (u otros destinos) para ejecutar una lógica de corrección personalizada en su carga de trabajo.

Recursos

Prácticas recomendadas relacionadas:

- Availability Definition
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

Documentos relacionados:

- Funcionamiento de AWS Auto Scaling
- Recuperación automática de Amazon EC2
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- What is Amazon FSx for Lustre?
- What is Amazon FSx for Windows File Server?
- AWS OpsWorks: Using Auto Healing to Replace Failed Instances
- Qué es AWS Step Functions?
- Qué es AWS Lambda?
- Qué es Amazon EventBridge?
- Using Amazon CloudWatch Alarms
- Amazon RDS Failover
- SSM - Systems Manager Automation
- Resilient Architecture Best Practices

Videos relacionados:

- Automatically Provision and Scale OpenSearch Service

- Amazon RDS Failover Automatically

Ejemplos relacionados:

- Amazon RDS Failover Workshop

Herramientas relacionadas:

- CloudWatch
- CloudWatch X-Ray

REL11-BP04 Confianza en el plano de datos y no en el plano de control durante la recuperación

Los planos de control proporcionan las API administrativas que se utilizan para crear, leer y describir, actualizar, eliminar y enumerar los recursos (CRUDL), mientras que los planos de datos gestionan el tráfico de servicio diario. Al implementar respuestas de recuperación o mitigación a eventos que puedan afectar a la resiliencia, céntrese en utilizar un número mínimo de operaciones del plano de control para recuperar, reescalar, restaurar, reparar o conmutar por error el servicio. La acción del plano de datos debe reemplazar cualquier actividad durante estos eventos de degradación.

Por ejemplo, las siguientes son todas las acciones del plano de control: lanzar una nueva instancia de computación, crear almacenamiento en bloques y describir los servicios de colas. Al lanzar instancias de computación, el plano de control debe hacer varias tareas, como encontrar un host físico con capacidad, asignar interfaces de red, preparar los volúmenes de almacenamiento en bloques locales, generar credenciales y agregar reglas de seguridad. Los planos de control suelen tener una orquestación complicada.

Resultado deseado: cuando un recurso entra en un estado deteriorado, el sistema es capaz de recuperarse automática o manualmente al cambiar el tráfico de recursos deteriorados a recursos en buen estado.

Patrones comunes de uso no recomendados:

- Depender de cambiar los registros de DNS para redirigir el tráfico.
- Depender de las operaciones de escalado del plano de control para reemplazar los componentes dañados debido a que no se han aprovisionado suficientes recursos.
- Confiar en amplias acciones del plano de control de varios servicios y varias API para corregir cualquier categoría de deterioro.

Beneficios de establecer esta práctica recomendada: el aumento de la tasa de éxito de la corrección automatizada puede reducir el tiempo medio de recuperación y mejorar la disponibilidad de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio (para determinados tipos de degradaciones del servicio, los planos de control se ven afectados). Si se depende del uso extensivo del plano de control para la corrección, se puede aumentar el tiempo de recuperación (RTO) y el tiempo medio de recuperación (MTTR).

Guía para la implementación

Para limitar las acciones del plano de datos, evalúe cada servicio para determinar qué acciones son necesarias para restablecer el servicio.

Use el Controlador de recuperación de aplicaciones de Amazon para cambiar el tráfico de DNS. Estas características supervisan continuamente la capacidad de la aplicación de recuperarse de los errores y le permiten controlar la recuperación de la aplicación en las distintas Regiones de AWS, zonas de disponibilidad y en las instalaciones.

Las políticas de enrutamiento de Route 53 utilizan el plano de control, por lo que no debe confiar en él para la recuperación. Los planos de datos de Route 53 responden a consultas de DNS y llevan a cabo y evalúan comprobaciones de estado. Están distribuidos por todo el mundo y están diseñados para cumplir con un acuerdo de nivel de servicio (SLA) con una disponibilidad del 100 %.

Las API de administración de Route 53 y las consolas en las que se crean, actualizan y eliminan recursos de Route 53 se ejecutan en planos de control diseñados para dar prioridad a la sólida coherencia y durabilidad que necesita al administrar DNS. Para conseguirlo, los planos de control se encuentran en una única región: Este de EE. UU. (Norte de Virginia). Aunque ambos sistemas se han diseñado para ser muy fiables, los planos de control no están incluidos en el SLA. Podría haber eventos poco frecuentes en los que el diseño resiliente del plano de datos permita mantener la disponibilidad mientras que los planos de control no lo permitan. Con los mecanismos de recuperación de desastres y conmutación por error, utilice las funciones del plano de datos para proporcionar la mejor fiabilidad posible.

Diseñe su infraestructura informática para que sea estable desde el punto de vista estático a fin de evitar el uso del plano de control durante un incidente. Por ejemplo, si utiliza instancias de Amazon EC2, evite aprovisionar nuevas instancias manualmente o dar instrucciones a los grupos de escalado automático para que agreguen instancias en respuesta. Para obtener los niveles más altos de resiliencia, aprovisione suficiente capacidad en el clúster utilizado para la conmutación por error. Si

este umbral de capacidad debe limitarse, establezca limitaciones en todo el sistema de principio a fin para limitar de forma segura el tráfico total que llega al conjunto limitado de recursos.

En el caso de los servicios como Amazon DynamoDB, Amazon API Gateway, los equilibradores de carga y sin servidor de AWS Lambda, el uso de esos servicios utiliza el plano de datos. Sin embargo, la creación de nuevas funciones, equilibradores de carga, puertas de enlace de API o tablas de DynamoDB es una acción del plano de control y debe completarse antes de la degradación como preparación para un evento y ensayo de las acciones de conmutación por error. En el caso de Amazon RDS, las acciones del plano de datos permiten el acceso a los datos.

Para obtener más información sobre planos de datos, planos de control y cómo AWS crea servicios para cumplir los objetivos de alta disponibilidad, consulte Estabilidad estática con zonas de disponibilidad.

Comprenda qué operaciones están en el plano de datos y cuáles están en el plano de control.

Pasos para la implementación

Para cada carga de trabajo que deba restaurarse después de un evento de degradación, evalúe el manual de procedimientos de conmutación por error, el diseño de alta disponibilidad, el diseño de reparación automática o el plan de restauración de recursos de alta disponibilidad. Identifique cada acción que pueda considerarse una acción del plano de control.

Considere cambiar la acción de control por una acción del plano de datos:

- Escalado automático (plano de control) a los recursos de Amazon EC2 preescalados (plano de datos)
- Escalado de instancias de Amazon EC2 (plano de control) a escalado de AWS Lambda (plano de datos)
- Evalúe cualquier diseño con Kubernetes y la índole de las acciones del plano de control. Agregar pods es una acción del plano de datos en Kubernetes. Las acciones deben limitarse a agregar pods y no a agregar nodos. Usar nodos sobreaprovisionados es el método preferido para limitar las acciones del plano de control

Tenga en cuenta los enfoques alternativos que permiten que las acciones del plano de datos afecten a la misma corrección.

- Cambio de registro de Route 53 (plano de control) o Controlador de recuperación de aplicaciones de Amazon (plano de datos)

- Comprobaciones de estado de Route 53 para obtener actualizaciones más automatizadas

Considere algunos servicios en una región secundaria, en caso de que el servicio sea crítico, para permitir más acciones del plano de control y el plano de datos en una región no afectada.

- Amazon EC2 Auto Scaling o Amazon EKS en una región principal en comparación con Amazon EC2 Auto Scaling o Amazon EKS en una región secundaria y enrutamiento del tráfico a una región secundaria (acción del plano de control)
- Hacer réplicas de lectura en la región principal secundaria o intentar la misma acción en la región principal (acción del plano de control)

Recursos

Prácticas recomendadas relacionadas:

- Availability Definition
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores

Documentos relacionados:

- Socio de APN: socios que pueden ayudar con la automatización de su tolerancia a errores
- AWS Marketplace: productos que pueden usarse para tolerancia a errores
- Amazon Builders' Library: Avoiding overload in distributed systems by putting the smaller service in control
- API de Amazon DynamoDB (plano de control y plano de datos)
- Ejecuciones de AWS Lambda (divididas entre el plano de control y el plano de datos)
- Plano de datos de AWS Elemental MediaStore
- Building highly resilient applications using Amazon Application Recovery Controller, Part 1: Single-Region stack
- Building highly resilient applications using Amazon Application Recovery Controller, Part 2: Multi-Region stack
- Creating Disaster Recovery Mechanisms Using Amazon Route 53
- What is Amazon Application Recovery Controller
- Kubernetes Control Plane and data plane

Videos relacionados:

- Back to Basics - Using Static Stability
- Building resilient multi-site workloads using AWS global services

Ejemplos relacionados:

- Introducing Amazon Application Recovery Controller
- Amazon Builders' Library: Avoiding overload in distributed systems by putting the smaller service in control
- Building highly resilient applications using Amazon Application Recovery Controller, Part 1: Single-Region stack
- Building highly resilient applications using Amazon Application Recovery Controller, Part 2: Multi-Region stack
- Estabilidad estática con zonas de disponibilidad

Herramientas relacionadas:

- Amazon CloudWatch
- AWS X-Ray

REL11-BP05 Uso de la estabilidad estática para evitar el comportamiento bimodal

Las cargas de trabajo deben ser estáticamente estables y funcionar solo en un único modo normal. El comportamiento bimodal se produce cuando la carga de trabajo presenta un comportamiento diferente en los modos normal y de error.

Por ejemplo, puede intentar recuperarse de un error en una zona de disponibilidad mediante el lanzamiento de nuevas instancias en una zona de disponibilidad distinta. Esto puede dar como resultado una respuesta bimodal durante un modo de error. En lugar de ello, debe crear cargas de trabajo que sean estables estáticamente y funcionen en un solo modo. En este ejemplo, esas instancias deben haberse aprovisionado en la segunda zona de disponibilidad antes del error. Este diseño de estabilidad estática verifica que la carga de trabajo solo funcione en un único modo.

Resultado deseado: las cargas de trabajo no muestran un comportamiento bimodal durante los modos normal y de error.

Patrones comunes de uso no recomendados:

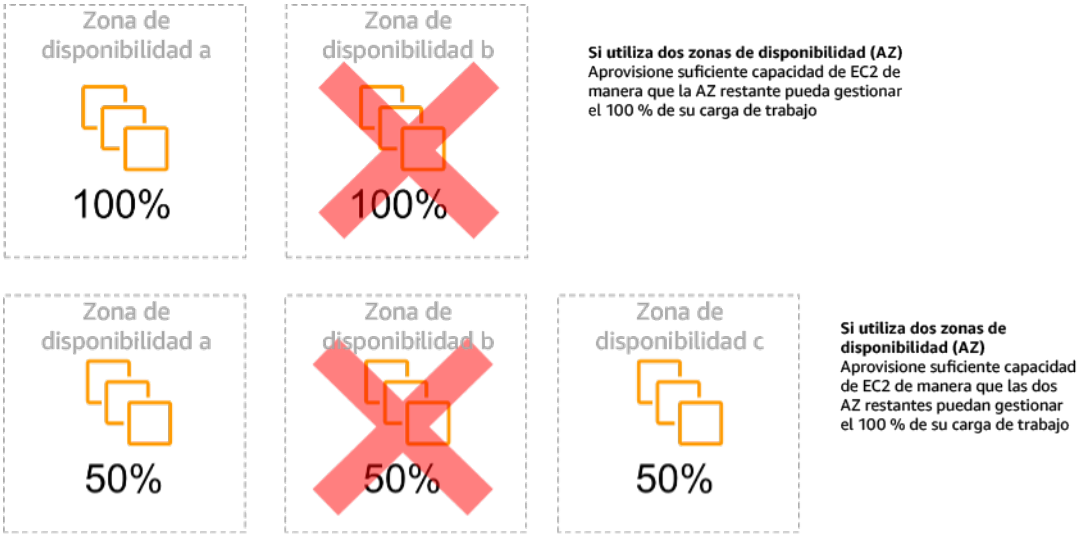
- Suponer que los recursos siempre se pueden aprovisionar independientemente del alcance del error.
- Intentar adquirir recursos de forma dinámica durante un error.
- No aprovisionar los recursos adecuados en todas las zonas o regiones hasta que se produzca un error.
- Considerar diseños estáticos estables solo para recursos de computación.

Beneficios de establecer esta práctica recomendada: las cargas de trabajo que se ejecutan con diseños estáticamente estables pueden tener resultados predecibles durante eventos normales y de error.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

El comportamiento bimodal ocurre cuando la carga de trabajo exhibe diferentes comportamientos en los modos normal y de error (como confiar en el lanzamiento de nuevas instancias si se produce un error en una zona de disponibilidad). Un ejemplo de comportamiento bimodal ocurre cuando los diseños de Amazon EC2 estables aprovisionan suficientes instancias en cada zona de disponibilidad para gestionar la carga de trabajo si se eliminara una zona de disponibilidad. Se comprobaría el estado de Elastic Load Balancing o Amazon Route 53 para desviar una carga de las instancias dañadas. Una vez desviado el tráfico, use AWS Auto Scaling para sustituir de manera asíncrona las instancias de la zona con errores y lanzarlas en las zonas en buen estado. La estabilidad estática para implementaciones de computación (como instancias de EC2 o contenedores) da como resultado la máxima fiabilidad.




Estabilidad estática de las instancias de EC2 entre zonas de disponibilidad

Esto debe ponderarse en comparación con el costo de este modelo y el valor empresarial de mantener la carga de trabajo en todos los casos de resiliencia. Es menos costoso aprovisionar menos capacidad de computación y confiar en el lanzamiento de nuevas instancias en caso de error, pero en el caso de errores a gran escala (como un deterioro regional o de zona de disponibilidad), este enfoque es menos eficaz porque se basa tanto en un plano operativo como en la disponibilidad de recursos suficientes en las zonas o regiones no afectadas.

Su solución también debe ponderar la fiabilidad en comparación con los costos necesarios para la carga de trabajo. Las arquitecturas de estabilidad estática se aplican a una variedad de arquitecturas, incluidas las instancias de computación distribuidas en las zonas de disponibilidad, los diseños de réplicas de lectura de bases de datos, los diseños de clústeres de Kubernetes (Amazon EKS) y las arquitecturas de conmutación por error multirregional.

También es posible implementar un diseño más estable desde el punto de vista estático mediante el uso de más recursos en cada zona. Al agregar más zonas, reduce la cantidad de procesamiento adicional que necesita para la estabilidad estática.

Un ejemplo de comportamiento bimodal sería un tiempo de espera de la red que podría provocar que un sistema intente actualizar el estado de configuración de todo el sistema. Se agregaría una carga inesperada a otro componente, lo que podría hacer que se produzca un error y desencadene otras consecuencias inesperadas. Este bucle de retroalimentación negativa afecta a la disponibilidad de su carga de trabajo. En lugar de ello, puede crear cargas de trabajo que sean estables estáticamente y funcionen en un solo modo. Un diseño estáticamente estable haría un trabajo constante y actualizaría continuamente el estado de configuración a una cadencia establecida. Cuando una

llamada genera un error, la carga de trabajo utiliza el valor previamente almacenado en caché e inicia una alarma.

Otro ejemplo de comportamiento bimodal es permitir que los clientes omitan la caché de la carga de trabajo si se produce un error. Esto podría parecer una solución para satisfacer las necesidades del cliente, pero puede cambiar notablemente la demanda de la carga de trabajo y es probable que produzca errores.

Evalúe las cargas de trabajo críticas para determinar cuáles requieren este tipo de diseño de resiliencia. Se debe revisar cada componente de la aplicación en las cargas que se consideren cruciales. Algunos tipos de servicios que requieren evaluaciones de estabilidad estática son:

- Computación: Amazon EC2, EKS-EC2, ECS-EC2, EMR-EC2
- Bases de datos: Amazon Redshift, Amazon RDS, Amazon Aurora
- Almacenamiento: Amazon S3 (zona única), Amazon EFS (montajes), Amazon FSx (montajes)
- Equilibradores de carga: según diseños determinados

Pasos para la implementación

- Cree cargas de trabajo que sean estables estáticamente y funcionen en un solo modo. En este caso, aprovisione suficientes instancias en cada región o zona de disponibilidad para gestionar la capacidad de la carga de trabajo en caso de que se eliminara una región o zona de disponibilidad. Puede usar una variedad de servicios para el enrutamiento a recursos en buen estado, como:
  - Cross Region DNS Routing
  - MRAP Amazon S3 MultiRegion Routing
  - AWS Global Accelerator
  - Controlador de recuperación de aplicaciones de Amazon
- Configure las réplicas de lectura de base de datos de modo que tengan en cuenta la pérdida de una única instancia principal o una réplica de lectura. Si las réplicas de lectura atienden el tráfico, la cantidad en cada zona de disponibilidad y cada región debe ser igual a la necesidad general en caso de que se produzca un error en la zona o región.
- Configure los datos cruciales en el almacenamiento de Amazon S3 que está diseñado para ser estáticamente estable para los datos almacenados en caso de que se produzca un error en la zona de disponibilidad. Si se utiliza la clase de almacenamiento Amazon S3 One Zone-IA, no debe considerarse estable desde el punto de vista estático, ya que la pérdida de esa zona minimiza el acceso a los datos almacenados.

- Los equilibradores de carga a veces están configurados incorrectamente o por diseño para prestar servicio a una zona de disponibilidad específica. En este caso, el diseño estáticamente estable podría consistir en distribuir una carga de trabajo entre varias zonas de disponibilidad en un diseño más complejo. El diseño original se puede utilizar para reducir el tráfico entre zonas por motivos de seguridad, latencia o costo.

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- Availability Definition
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores
- REL11-BP04 Confianza en el plano de datos y no en el plano de control durante la recuperación

Documentos relacionados:

- Minimizing Dependencies in a Disaster Recovery Plan
- Amazon Builders' Library: estabilidad estática con zonas de disponibilidad
- Fault Isolation Boundaries
- Estabilidad estática con zonas de disponibilidad
- RDS multizona
- Minimizing Dependencies in a Disaster Recovery Plan
- Cross Region DNS Routing
- MRAP Amazon S3 MultiRegion Routing
- AWS Global Accelerator
- Controlador de recuperación de aplicaciones de Amazon
- Amazon S3 de zona única
- Cross Zone Load Balancing

Videos relacionados:

- Static stability in AWS: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328)

REL11-BP06 Envío de notificaciones cuando los eventos afecten a la disponibilidad

Se envían notificaciones cuando se detecta que se han superado los umbrales, incluso si el evento que causó el problema se ha resuelto automáticamente.

La corrección automática permite que la carga de trabajo sea fiable. Sin embargo, también puede ocultar problemas subyacentes que deberían abordarse. Implemente una supervisión y unos eventos adecuados para poder detectar patrones de problemas, incluidos los que pueden abordarse mediante corrección automática, para que pueda resolver los problemas de la causa fundamental.

Los sistemas resilientes están diseñados para que los eventos de degradación se comuniquen inmediatamente a los equipos correspondientes. Estas notificaciones deben enviarse a través de uno o varios canales de comunicación.

Resultado deseado: las alertas se envían inmediatamente a los equipos de operaciones cuando se superan los umbrales, como las tasas de error, la latencia u otras métricas cruciales de los indicadores clave de rendimiento (KPI), para que estos problemas se resuelvan lo antes posible y se evite o minimice el impacto en los usuarios.

Patrones comunes de uso no recomendados:

- Enviar demasiadas alarmas.
- Enviar alarmas que no son procesables.
- Establecer umbrales de alarma demasiado altos (muy sensibles) o demasiado bajos (poco sensibles).
- No enviar alarmas para dependencias externas.
- No considerar los errores grises al diseñar la supervisión y las alarmas.
- Llevar a cabo la automatización de la reparación, pero sin notificar al equipo adecuado que se necesita una reparación.

Beneficios de establecer esta práctica recomendada: las notificaciones de recuperación permiten que los equipos operativos y empresariales estén al tanto de las degradaciones del servicio para que puedan reaccionar de inmediato y minimizar tanto el tiempo medio de detección (MTTD) como el tiempo medio de reparación (MTTR). Las notificaciones de los eventos de recuperación también garantizan que no se ignoren problemas que ocurren con poca frecuencia.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio. Si no se implementan los mecanismos adecuados de supervisión y notificación de eventos, es posible que no se detecten

patrones de problemas, incluidos los que pueden abordarse mediante la corrección automática. El equipo solo descubrirá la degradación del sistema cuando los usuarios se pongan en contacto con el servicio de atención al cliente o por casualidad.

Guía para la implementación

Al definir una estrategia de supervisión, la activación de una alarma es un evento frecuente. Es probable que este evento contenga un identificador de la alarma, el estado de la alarma (como `IN ALARM` o `OK`) y los detalles de lo que la desencadenó. En muchos casos, se debe detectar el evento de alarma y enviar una notificación por correo electrónico. Este es un ejemplo de una acción en una alarma. La notificación de alarmas es fundamental en la observabilidad, ya que informa a las personas adecuadas de que existe un problema. Sin embargo, cuando la acción sobre los eventos madura en su solución de observabilidad, puede solucionar el problema automáticamente sin necesidad de intervención humana.

Una vez que se hayan establecido las alarmas de supervisión de los KPI, se deben enviar alertas a los equipos correspondientes cuando se superen los umbrales. Esas alertas también se pueden usar para activar procesos automatizados que intentarán corregir la degradación.

Para una supervisión de umbrales más compleja, se deben considerar las alarmas compuestas. Las alarmas compuestas utilizan una serie de alarmas de supervisión de KPI para crear una alerta basada en la lógica empresarial operativa. Las alarmas de CloudWatch se pueden configurar para enviar correos electrónicos o para registrar incidentes en sistemas de seguimiento de incidentes de terceros mediante la integración con Amazon SNS o Amazon EventBridge.

Pasos para la implementación

Cree varios tipos de alarmas en función de la forma en que se supervisan las cargas de trabajo, como, por ejemplo:

- Las alarmas de las aplicaciones se utilizan para detectar cuando alguna parte de la carga de trabajo no funciona correctamente.
- Las alarmas de la infraestructura indican cuándo escalar los recursos. Las alarmas se pueden mostrar visualmente en paneles, enviar alertas a través de Amazon SNS o por correo electrónico y trabajar con el escalado automático para reducir o escalar horizontalmente los recursos de la carga de trabajo.
- Se pueden crear alarmas estáticas sencillas para supervisar cuando una métrica supera un umbral estático durante un número específico de periodos de evaluación.

- Las alarmas compuestas pueden abarcar alarmas complejas de numerosos orígenes.
- Una vez creada la alarma, cree los eventos de notificación adecuados. Puede invocar directamente una API de Amazon SNS para enviar notificaciones y vincular cualquier automatización para su corrección o comunicación.
- Manténgase informado sobre las degradaciones del servicio con AWS Health. Cree notificaciones de eventos de AWS Health adecuados para su propósito para los canales de correo electrónico y chat a través de AWS User Notifications e intégrelas mediante programación con las herramientas de supervisión y alertas a través de Amazon EventBridge.

Recursos

Prácticas recomendadas de Well-Architected relacionadas:

- Availability Definition

Documentos relacionados:

- Cree una alarma de CloudWatch basada en un umbral estático
- Qué es Amazon EventBridge?
- What is Amazon Simple Notification Service?
- Publicar métricas personalizadas
- Uso de las alarmas de Amazon CloudWatch
- Configuración de alarmas compuestas de CloudWatch
- What's new in AWS Observability at re:Invent 2022

Herramientas relacionadas:

- CloudWatch
- CloudWatch X-Ray

REL11-BP07 Diseño de su producto para cumplir objetivos de disponibilidad y acuerdos de nivel de servicio (SLA) de tiempo de actividad

Diseñe el producto para que cumpla con los objetivos de disponibilidad y los acuerdos de nivel de servicio (SLA) de tiempo de actividad. Si publica o acuerda en privado objetivos de disponibilidad

o SLA de tiempo de actividad, verifique que su arquitectura y procesos operativos están diseñados para admitirlos.

Resultado deseado: cada aplicación tiene un objetivo definido de disponibilidad y un SLA para las métricas de rendimiento, que se pueden supervisar y mantener para cumplir con los resultados comerciales.

Patrones comunes de uso no recomendados:

- Diseño e implementación de cargas de trabajo sin establecer acuerdos de nivel de servicio.
- Las métricas de los SLA se fijan en niveles demasiado altos sin justificación ni requisitos empresariales.
- Establecimiento de SLA sin tener en cuenta las dependencias y sus SLA subyacentes.
- Los diseños de aplicaciones se crean sin tener en cuenta el Modelo de responsabilidad compartida para la resiliencia.

Beneficios de establecer esta práctica recomendada: diseñar aplicaciones en función de los objetivos clave de resiliencia ayuda a cumplir los objetivos empresariales y las expectativas de los clientes. Estos objetivos contribuyen a impulsar el proceso de diseño de aplicaciones que evalúa diferentes tecnologías y tiene en cuenta varios compromisos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

El diseño de aplicaciones debe tener en cuenta una serie de requisitos derivados de objetivos empresariales, operativos y financieros. Dentro de los requisitos operativos, las cargas de trabajo deben tener objetivos concretos de métricas de resiliencia para que se puedan supervisar y respaldar adecuadamente. Las métricas de resiliencia no deben establecerse ni derivarse después de implementar la carga de trabajo. En cambio, deben definirse durante la fase de diseño y ayudar a orientar diversas decisiones y compensaciones.

- Cada carga de trabajo debe tener su propio conjunto de métricas de resiliencia. Esas métricas pueden ser diferentes de las de otras aplicaciones empresariales.
- Reducir las dependencias puede tener un impacto positivo en la disponibilidad. Cada carga de trabajo debe considerar sus dependencias y sus SLA. En general, seleccione dependencias con objetivos de disponibilidad iguales o superiores a los objetivos de su carga de trabajo.

- Siempre que sea posible, examine diseños de acoplamiento débil para que la carga de trabajo pueda funcionar correctamente a pesar del deterioro de las dependencias.
- Reduzca las dependencias del plano de control, especialmente durante la recuperación o una degradación. Evalúe diseños que sean estáticamente estables para las cargas de trabajo críticas. Utilice el ahorro de recursos para aumentar la disponibilidad de esas dependencias en una carga de trabajo.
- La observabilidad y la instrumentación son fundamentales para alcanzar los SLA al reducir el tiempo medio de detección (MTTD) y el tiempo medio de reparación (MTTR).
- Los tres factores que se utilizan para mejorar la disponibilidad en los sistemas distribuidos son una menor frecuencia de errores (MTBF más largo), tiempos de detección de errores más cortos (MTTD más corto) y tiempos de reparación más cortos (MTTR más corto).
- Establecer y cumplir las métricas de resiliencia para una carga de trabajo es una parte imprescindible de todo diseño eficaz. Estos diseños deben tener en cuenta las compensaciones de la complejidad del diseño, las dependencias de los servicios, el rendimiento, la escalabilidad y los costos.

Pasos para la implementación

- Revise y documente el diseño de la carga de trabajo con las siguientes preguntas en mente:
  - ¿Dónde se utilizan los planos de control en la carga de trabajo?
  - ¿Cómo implementa la carga de trabajo la tolerancia a errores?
  - ¿Cuáles son los patrones de diseño para el escalado, el escalado automático, la redundancia y los componentes de alta disponibilidad?
  - ¿Cuáles son los requisitos de coherencia y disponibilidad de los datos?
  - ¿Se tiene en cuenta el ahorro de recursos o la estabilidad estática de los recursos?
  - ¿Cuáles son las dependencias de los servicios?
- Defina las métricas de los SLA en función de la arquitectura de la carga de trabajo mientras trabaja con las partes interesadas. Considere los SLA de todas las dependencias que utiliza la carga de trabajo.
- Una vez establecido el objetivo del SLA, optimice la arquitectura para que cumpla el SLA.
- Una vez establecido el diseño que cumplirá el SLA, implemente cambios operativos, automatización de procesos y manuales de procedimientos que también se centren en reducir el MTTD y el MTTR.
- Una vez implementado, supervise y cree informes del SLA.

Recursos

Prácticas recomendadas relacionadas:

- REL03-BP01 Elección de cómo segmentar su carga de trabajo
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores
- REL11-BP03 Automatización de la reparación en todas las capas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos
- REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos
- Comprender el estado de la carga de trabajo

Documentos relacionados:

- Disponibilidad con redundancia
- Pilar de fiabilidad: disponibilidad
- Measuring availability
- AWS Fault Isolation Boundaries
- Modelo de responsabilidad compartida para la resiliencia
- Estabilidad estática con zonas de disponibilidad
- Acuerdos de nivel de servicios (SLA) de AWS
- Guidance for Cell-based Architecture on AWS
- AWS infrastructure
- Documento técnico Advanced Multi-AZ Resilience Patterns

Servicios relacionados:

- Amazon CloudWatch
- AWS Config
- AWS Trusted Advisor

## REL 12. ¿Cómo pone a prueba la fiabilidad?

Una vez diseñada la carga de trabajo para que sea resiliente al estrés de producción, las pruebas son la única forma de comprobar que funcionará según lo previsto y proporcionará la resiliencia esperada.

Prácticas recomendadas

- REL12-BP01 Uso de manuales de estrategias para investigar los errores
- REL12-BP02 Análisis después del incidente
- REL12-BP03 Requisitos de escalado y rendimiento de las pruebas
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos
- REL12-BP05 Planificación periódica de días de juego

### REL12-BP01 Uso de manuales de estrategias para investigar los errores

Permite obtener respuestas sistemáticas e inmediatas a escenarios de error que no se comprendan bien mediante la documentación del proceso de investigación en guías de estrategias. Los manuales de estrategias son pasos predefinidos hechos para identificar los factores que contribuyen a un escenario de error. Los resultados de cualquier paso del proceso se utilizan para determinar los siguientes pasos, hasta que el problema se identifique o escale.

El manual de estrategias es una planificación proactiva que debe hacer para poder tomar medidas reactivas de manera efectiva. Cuando se produzcan situaciones de error que no estén contempladas en el manual de estrategias, aborde primero el problema (resuelva la crisis). A continuación, repase los pasos que ha seguido para solucionar el problema y utilícelos para agregar una nueva entrada al manual de estrategias.

Tenga en cuenta que los manuales de estrategias se usan en respuesta a incidentes específicos, mientras que los manuales de procedimientos se usan para conseguir resultados específicos. A menudo, los manuales de procedimientos se utilizan para actividades rutinarias, mientras que los manuales de estrategias se utilizan para responder a eventos no rutinarios.

Patrones comunes de uso no recomendados:

- Planificar la implementación de una carga de trabajo sin conocer los procesos para diagnosticar los problemas o responder a los incidentes.
- Decisiones no planificadas acerca de qué sistemas se recopilan registros y métricas cuando se investiga un evento.

- No retener las métricas y los eventos el tiempo suficiente para poder recuperar los datos.

Beneficios de establecer esta práctica recomendada: la captura de esta información en manuales de estrategias garantiza que el proceso pueda seguirse sistemáticamente. La creación de manuales de estrategias limita la introducción de errores de la actividad manual. La automatización de manuales de estrategias reduce el tiempo para responder a un evento, ya que se elimina el requisito de intervención de un miembro del equipo o se dispone de información adicional al inicio de su intervención.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

- Use manuales de estrategias para identificar problemas. Los manuales de estrategias son procesos documentados para investigar problemas. Permita las respuestas sistemáticas e inmediatas a escenarios de error mediante la documentación de los procesos en manuales de estrategias. Los manuales de estrategias deben contener la información y las instrucciones necesarias para que alguien con la formación adecuada reúna la información correspondiente, identifique las posibles fuentes de error, aísle los errores y determine los factores que han contribuido al problema (hacer un análisis después del incidente).
  - Implemente manuales de estrategias como código. Efectúe sus operaciones como código mediante scripts de sus manuales de estrategias para garantizar la sistematicidad y reducir los errores provocados por los procesos manuales. Los manuales de estrategias pueden constar de varios scripts que representen los diferentes pasos que podrían ser necesarios para identificar los factores que contribuyen a un problema. Se pueden invocar o llevar a cabo actividades del manual de procedimientos como parte de las actividades de un manual de estrategias, o se puede solicitar la ejecución de un manual de estrategias en respuesta a eventos identificados.
    - Automatice sus manuales operativos con AWS Systems Manager
    - AWS Systems Manager Run Command
    - AWS Systems Manager Automation
    - ¿Qué es AWS Lambda?
    - ¿Qué es Amazon EventBridge?
    - Uso de las alarmas de Amazon CloudWatch

Recursos

Documentos relacionados:

- AWS Systems Manager Automation
- AWS Systems Manager Run Command
- Automatice sus manuales operativos con AWS Systems Manager
- Uso de las alarmas de Amazon CloudWatch
- Uso de canarios (Amazon CloudWatch Synthetics)
- ¿Qué es Amazon EventBridge?
- ¿Qué es AWS Lambda?

Ejemplos relacionados:

- Automating operations with Playbooks and Runbooks

REL12-BP02 Análisis después del incidente

Revise los eventos que afectan a los clientes e identifique los factores que contribuyen al evento y las medidas preventivas. Use esta información para desarrollar un plan de mitigación que limite o evite la reaparición del problema. Desarrolle procedimientos para proporcionar respuestas rápidas y eficaces. Comunique los factores que han contribuido al problema y las medidas correctivas según corresponda, adaptados al público de destino. Disponga de un método para comunicar estas causas a otros usuarios según sea necesario.

Evalúe por qué las pruebas existentes no han detectado el problema. Agregue pruebas para este caso si no hay pruebas ya establecidas.

Resultado deseado: sus equipos tienen un enfoque coherente y consensuado para gestionar el análisis posterior al incidente. Un mecanismo es el proceso de corrección de errores (COE). El proceso de COE ayuda a los equipos a identificar, comprender y abordar las causas fundamentales de los incidentes, a la vez que crea mecanismos y barreras de protección para limitar la probabilidad de que se repita el mismo incidente.

Patrones comunes de uso no recomendados:

- Buscar los factores que han contribuido al problema, pero no seguir investigando si existen otros problemas potenciales o enfoques que mitigar.

- Identificar solo los errores humanos y no proporcionar ninguna formación o automatización que pueda evitar estos errores.
- Concentrarse en determinar la culpa en lugar de en conocer la causa fundamental, lo que da lugar a una cultura de miedo y obstaculiza la comunicación abierta.
- No intercambiar información, lo que hace que los resultados del análisis de incidentes los conozca solo un grupo pequeño y evita que otros se beneficien de las lecciones aprendidas.
- No existe ningún mecanismo para capturar el conocimiento institucional, por lo que se pierde información valiosa al no preservar las lecciones aprendidas en forma de actualizaciones de las prácticas recomendadas y, por lo tanto, se repiten incidentes con la misma causa fundamental o una similar

Beneficios de establecer esta práctica recomendada: hacer análisis después de un incidente y compartir los resultados permite que el riesgo se mitigue en otras cargas de trabajo si estas tienen implementadas los mismos factores que han contribuido al problema, y permite también implementar la mitigación o la recuperación automatizada antes de que se produzca un incidente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Un buen análisis posterior a un incidente ofrece oportunidades de proponer soluciones comunes para problemas con patrones de arquitectura que se utilizan en otros lugares de los sistemas.

Un aspecto clave del proceso de COE es documentar y resolver los problemas. Es recomendable definir una forma estandarizada de documentar las causas fundamentales críticas y garantizar que estas se revisan y solucionan. Asigne una responsabilidad clara al proceso de análisis posterior al incidente. Nombre un equipo o persona responsable que supervise las investigaciones y el seguimiento de los incidentes.

Fomente una cultura que se centre en el aprendizaje y la mejora en lugar de en la culpa. Haga hincapié en que el objetivo es prevenir futuros incidentes, no penalizar a las personas.

Desarrolle procedimientos bien definidos para llevar a cabo análisis posteriores a los incidentes. En estos procedimientos, se deben describir los pasos que hay que seguir, la información que se va a recopilar y las preguntas clave que se abordarán durante el análisis. Investigue los incidentes a fondo y vaya más allá de las causas inmediatas para identificar las causas fundamentales y los factores que contribuyen a ellos. Utilice técnicas como los cinco porqués para profundizar en los problemas subyacentes.

Mantenga un repositorio de las lecciones aprendidas de los análisis de incidentes. Este conocimiento institucional puede servir como referencia para incidentes futuros e iniciativas de prevención. Comparta los resultados y la información de los análisis posteriores a los incidentes y considere la posibilidad de celebrar reuniones de revisión de invitación abierta después de los incidentes para analizar las lecciones aprendidas.

Pasos para la implementación

- Al hacer un análisis posterior al incidente, asegúrese de que en el proceso no se culpe a nadie. Esto permite que las personas involucradas en el incidente se muestren imparciales con respecto a las medidas correctivas propuestas, además de fomentar una autoevaluación honesta y la colaboración entre los equipos.
- Defina una forma estandarizada de documentar los problemas críticos. Un ejemplo de estructura para dicho documento es el siguiente:
  - ¿Qué ha pasado?
  - ¿Cómo ha afectado a los clientes y a la empresa?
  - ¿Cuál ha sido la causa fundamental?
  - ¿Qué datos tiene para corroborarlo?
    - Por ejemplo, métricas y gráficos
  - ¿Qué pilares básicos estuvieron implicados, con especial atención a la seguridad?
    - Al diseñar cargas de trabajo, se hacen concesiones entre pilares según el contexto del negocio. Estas decisiones de negocio puede impulsar sus prioridades de diseño. Podría dar preferencia a reducir el costo a expensas de la fiabilidad en el desarrollo de entornos o, para soluciones críticas, podría optimizar la fiabilidad con un aumento de los costos. La seguridad siempre es la tarea primordial, ya que sus clientes deben estar protegidos.
  - ¿Qué lecciones aprendió?
  - ¿Qué medidas correctivas va a tomar?
    - Elementos de acción
    - Elementos relacionados
- Cree procedimientos operativos estándar bien definidos para llevar a cabo análisis posteriores a los incidentes.
- Configure un proceso estandarizado de notificación de incidentes. Documente todos los incidentes de manera exhaustiva, incluido el informe inicial del incidente, los registros, las comunicaciones y las medidas tomadas durante el incidente.

- Recuerde que un incidente no requiere una interrupción. Podría tratarse de un cuasi incidente o de un sistema que funciona de una forma inesperada, pero que cumple su función.
- Mejore continuamente su proceso de análisis posterior a un incidente en función de los comentarios y las lecciones aprendidas.
- Registre los resultados clave en un sistema de administración del conocimiento y considere cualquier patrón que deba agregarse a las guías para desarrolladores o a las listas de verificación previas a la implementación.

Recursos

Documentos relacionados:

- Why you should develop a correction of error (COE)

Videos relacionados:

- Amazon's approach to failing successfully
- AWS re:Invent 2021 - Amazon Builders' Library: Operational Excellence at Amazon

REL12-BP03 Requisitos de escalado y rendimiento de las pruebas

Utilice técnicas, como las pruebas de carga, para validar que la carga de trabajo satisface los requisitos de escalado y rendimiento.

En la nube, puede crear un entorno de pruebas a escala de producción bajo demanda para la carga de trabajo. En lugar de confiar en un entorno de pruebas reducido, lo que podría provocar predicciones inexactas de los comportamientos de producción, puede utilizar la nube para proporcionar un entorno de pruebas que refleje fielmente el entorno de producción esperado. Este entorno lo ayuda a realizar pruebas en una simulación más precisa de las condiciones reales a las que se enfrenta su aplicación.

Además de las pruebas de rendimiento que intente llevar a cabo, asegúrese de validar que los recursos básicos, la configuración de escalado, las cuotas de servicio y el diseño de resiliencia funcionen según lo esperado bajo carga. Este enfoque holístico verifica que su aplicación pueda escalarse y funcionar de manera confiable según sea necesario, incluso en las condiciones más exigentes.

Resultado deseado: su carga de trabajo mantiene el comportamiento esperado incluso cuando está sujeta a picos de carga. Aborda de forma proactiva cualquier problema relacionado con el rendimiento que pueda surgir a medida que la aplicación crece y evoluciona.

Patrones comunes de uso no recomendados:

- Utiliza entornos de prueba que no coinciden estrechamente con el entorno de producción.
- Considera las pruebas de carga como una actividad independiente y única, y no como una parte integrada del proceso de integración continua (CI) de la implementación.
- No se definen requisitos de rendimiento claros y mensurables, como el tiempo de respuesta, el rendimiento y los objetivos de escalabilidad.
- Realiza pruebas con escenarios de carga poco realistas o insuficientes, y no realiza pruebas para detectar picos de carga, picos repentinos y cargas elevadas sostenidas.
- No se pone a prueba la carga de trabajo usando límites de carga esperados excesivos.
- Utilizas herramientas de pruebas de carga y elaboración de perfiles de rendimiento inadecuadas.
- Carece de sistemas integrales de supervisión y alerta para realizar un seguimiento de las métricas de rendimiento y detectar anomalías.

Beneficios de establecer esta práctica recomendada:

- Las pruebas de carga lo ayudan a identificar posibles obstáculos en el rendimiento de su sistema antes de que entre en producción. Al simular el tráfico y las cargas de trabajo en la producción, puede identificar las áreas en las que el sistema puede tener dificultades para gestionar la carga, como los tiempos de respuesta lentos, la escasez de recursos o los fallos del sistema.
- Probar el sistema en distintas condiciones de carga lo ayudará a comprender mejor los requisitos de recursos necesarios para respaldar su carga de trabajo. Esta información puede ayudarlo a tomar decisiones fundamentadas sobre la asignación de recursos y a evitar el aprovisionamiento excesivo o insuficiente de los recursos.
- Para identificar los posibles puntos de fallo, puede observar el rendimiento de su carga de trabajo en condiciones de carga elevada. Esta información lo ayuda a mejorar la fiabilidad y la resiliencia de su carga de trabajo mediante la implementación de mecanismos de tolerancia a fallos, estrategias de conmutación por error y medidas de redundancia, según corresponda.
- Puede identificar y abordar los problemas de rendimiento anticipadamente y evitar así el elevado coste de las interrupciones del sistema, la lentitud de los tiempos de respuesta y la insatisfacción de los usuarios.

- Los datos de rendimiento detallados y la información de creación de perfiles recopilados durante las pruebas pueden ayudarlo a solucionar los problemas relacionados con el rendimiento que puedan surgir en la producción. Esto puede acelerar la respuesta y la resolución de los incidentes, lo que reduce el impacto en los usuarios y en las operaciones de la organización.
- En algunos sectores, las pruebas de rendimiento proactivas pueden ayudar a que su carga de trabajo cumpla con los estándares de conformidad, lo que reduce el riesgo de sanciones o problemas legales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El primer paso es definir una estrategia de pruebas integral que abarque todos los aspectos de los requisitos de escalado y rendimiento. Para empezar, defina claramente los objetivos de nivel de servicio (SLO) de su carga de trabajo en función de las necesidades de su empresa, como el rendimiento, el histograma de latencia y la tasa de errores. A continuación, diseñe un conjunto de pruebas que puedan simular varios escenarios de carga, desde un uso medio hasta picos repentinos y picos de carga sostenidos, y compruebe que el comportamiento de la carga de trabajo cumpla con sus SLO. Estas pruebas deben automatizarse e integrarse en la canalización de integración e implementación continuas para detectar las regresiones de rendimiento al principio del proceso de desarrollo.

Para probar de forma eficaz el escalado y el rendimiento, invierta en las herramientas y la infraestructura adecuadas. Esto incluye herramientas de pruebas de carga que pueden generar un tráfico de usuarios realista, herramientas de creación de perfiles de rendimiento para identificar los cuellos de botella y soluciones de supervisión para realizar un seguimiento de las métricas clave. Por otro lado, es muy importante comprobar que sus entornos de prueba se ajusten perfectamente al entorno de producción en términos de infraestructura y condiciones ambientales para que los resultados de las pruebas sean lo más precisos posible. Para facilitar la replicación y el escalado fiables de configuraciones similares a las de producción, utilice la infraestructura como código y aplicaciones basadas en contenedores.

Las pruebas de escalado y rendimiento son un proceso continuo, no una actividad que se realiza una sola vez. Implemente la supervisión y las alertas de manera exhaustiva para realizar un seguimiento del rendimiento de la aplicación en producción y utilice estos datos para perfeccionar continuamente sus estrategias de prueba y sus esfuerzos de optimización. Analice periódicamente los datos de rendimiento para identificar los problemas que surjan, pruebe nuevas estrategias de escalado e implemente optimizaciones para mejorar la eficiencia y la fiabilidad de la aplicación. Si adopta un

enfoque iterativo y aprende constantemente de los datos de producción, puede comprobar que su aplicación puede adaptarse a las demandas variables de los usuarios y mantener la resiliencia y un rendimiento óptimo a lo largo del tiempo.

Pasos para la implementación

1. Establezca requisitos de rendimiento claros y mensurables, como el tiempo de respuesta, el rendimiento y los objetivos de escalabilidad. Estos requisitos deben basarse en los patrones de uso de la carga de trabajo, las expectativas de los usuarios y las necesidades empresariales.
2. Seleccione y configure una herramienta de pruebas de carga que pueda imitar con precisión los patrones de carga y el comportamiento de los usuarios en su entorno de producción.
3. Configure un entorno de pruebas que se ajuste perfectamente al entorno de producción, incluidas las condiciones de la infraestructura y el entorno, para mejorar la precisión de los resultados de las pruebas.
4. Cree un conjunto de pruebas que abarque una amplia gama de escenarios, desde patrones de uso promedio hasta picos de carga, picos rápidos y cargas elevadas sostenidas. Integre las pruebas en la canalización de integración e implementación continuas para detectar las regresiones de rendimiento al principio del proceso de desarrollo.
5. Realice pruebas de carga para simular el tráfico de usuarios del mundo real y comprender cómo se comporta su aplicación en diferentes condiciones de carga. Para someter la aplicación a una prueba de estrés, supere la carga esperada y observe su comportamiento, como la degradación del tiempo de respuesta, el agotamiento de los recursos o los fallos del sistema. Esto ayuda a identificar el punto de ruptura de su aplicación y a determinar las estrategias de escalado. Evalúe la escalabilidad de su carga de trabajo aumentando gradualmente la carga y mida el impacto en el rendimiento para identificar los límites de escalado y planificar las necesidades de capacidad futuras.
6. Implemente la supervisión y las alertas de manera exhaustiva para realizar un seguimiento de las métricas de rendimiento, detectar anomalías e iniciar acciones de escalado o notificaciones cuando se superen los umbrales.
7. Supervise y analice continuamente los datos de rendimiento para identificar las áreas de mejora. Repita sus estrategias de prueba y sus esfuerzos de optimización.

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP04 Supervisión y administración de cuotas

- REL06-BP01 Supervisión de todos los componentes de la carga de trabajo (generación)
- REL06-BP03 Envío de notificaciones (procesamiento y alarmas en tiempo real)

Documentos relacionados:

- Aplicaciones de pruebas de carga
- Pruebas de carga distribuidas en AWS
- Supervisión del rendimiento de las aplicaciones
- Amazon EC2 Testing Policy

Ejemplos relacionados:

- Pruebas de carga distribuidas en AWS (GitHub)

Herramientas relacionadas:

- Generador de perfiles de Amazon CodeGuru
- Amazon CloudWatch RUM
- Apache JMeter
- K6
- Vegeta
- Hey
- ab
- wrk
- Pruebas de carga distribuidas en AWS

REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos

Haga experimentos de caos con regularidad en entornos que estén en producción o lo más cerca posible de ella para entender cómo responde su sistema a condiciones adversas.

Resultado deseado:

La resiliencia de la carga de trabajo se verifica regularmente aplicando la ingeniería del caos en forma de experimentos de inyección de errores o inyección de carga inesperada, además

de las pruebas de resiliencia que validan el comportamiento esperado conocido de su carga de trabajo durante un evento. Combine la ingeniería del caos y las pruebas de resiliencia para tener la seguridad de que su carga de trabajo puede sobrevivir a los errores de los componentes y puede recuperarse de las interrupciones inesperadas con un impacto mínimo o nulo.

Patrones comunes de uso no recomendados:

- Diseñar para lograr la resiliencia, pero no verificar cómo funciona la carga de trabajo en su conjunto cuando se producen errores.
- No experimentar nunca en condiciones reales y con la carga prevista.
- No tratar los experimentos como código ni mantenerlos durante el ciclo de desarrollo.
- No ejecutar experimentos de caos tanto como parte de su canalización de CI/CD, así como fuera de las implementaciones.
- No utilizar los análisis posteriores a los incidentes a la hora de determinar los errores con los que experimentar.

Beneficios de establecer esta práctica recomendada: inyectar errores para verificar la resiliencia de la carga de trabajo permite tener seguridad de que los procedimientos de recuperación de su diseño resiliente funcionarán en caso de un error real.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La ingeniería del caos proporciona a sus equipos capacidades para inyectar continuamente interrupciones del mundo real (simulaciones) de forma controlada según el proveedor de servicios, infraestructura, carga de trabajo y componentes, con un impacto mínimo o nulo para sus clientes. Permite que sus equipos aprendan de los errores y observen, midan y mejoren la resiliencia de las cargas de trabajo, además de validar que las alertas se activen y que los equipos reciban notificaciones en caso de algún evento.

Cuando se hace de forma continua, la ingeniería del caos puede poner de manifiesto deficiencias en las cargas de trabajo que, si no se abordan, podrían afectar negativamente a la disponibilidad y al funcionamiento.

> **Note**
>
> La ingeniería del caos es la disciplina que consiste en experimentar en un sistema para generar confianza en la capacidad del sistema de resistir condiciones adversas en producción. – Principios de la ingeniería del caos

Si un sistema puede soportar estas interrupciones, el experimento del caos debe mantenerse como una prueba de regresión automatizada. De este modo, los experimentos de caos deben hacerse como parte de su ciclo de vida de desarrollo de sistemas (SDLC) y como parte de su canalización de CI/CD.

Para garantizar que la carga de trabajo puede sobrevivir a los errores de los componentes, inyecte eventos reales como parte de los experimentos. Por ejemplo, experimente con la pérdida de instancias de Amazon EC2 o la conmutación por error de la instancia primaria de la base de datos de Amazon RDS y verifique que la carga de trabajo no se ve afectada (o solo en un mínimo). Utilice una combinación de errores de componentes para simular los eventos que puede causar una interrupción en una zona de disponibilidad.

Para los errores a nivel de aplicación (como las caídas), se puede empezar con factores de estrés como el agotamiento de la memoria y la CPU.

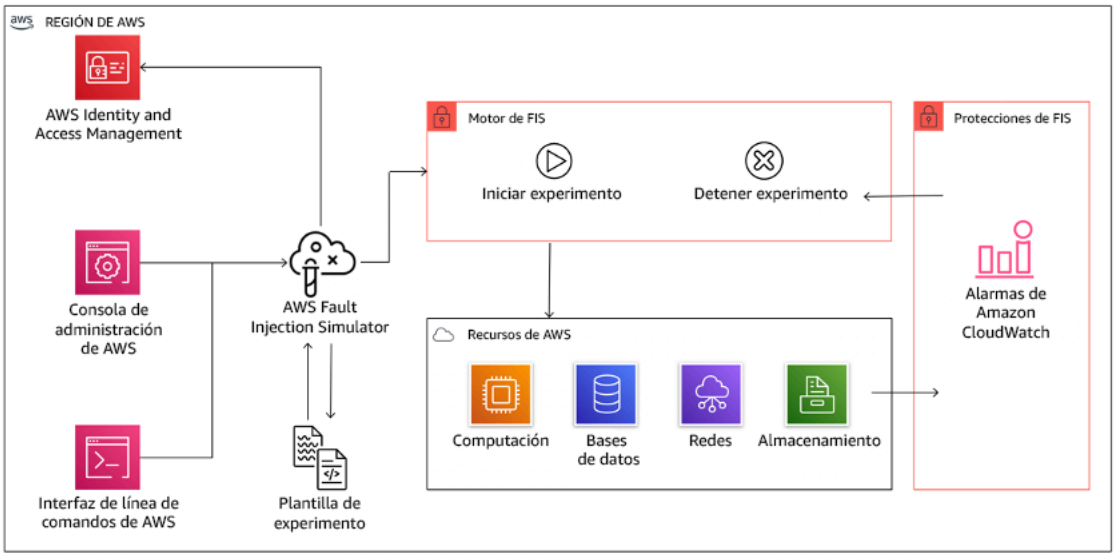
Para validar los mecanismos de respaldo y de conmutación por error para las dependencias externas debido a interrupciones intermitentes de la red, sus componentes deben simular un evento de este tipo mediante el bloqueo del acceso a los proveedores de terceros durante una duración especificada que puede durar desde segundos hasta horas.

Otros modos de degradación podrían provocar una funcionalidad reducida y respuestas lentas, lo que a menudo da como resultado una interrupción de los servicios. Los orígenes comunes de esta degradación son una mayor latencia en los servicios críticos y una comunicación de red poco fiable (paquetes omitidos). Los experimentos con estos errores, como, por ejemplo, efectos de red como la latencia, los mensajes perdidos y los errores de DNS, podrían incluir la incapacidad de resolver un nombre, alcanzar el servicio DNS o establecer conexiones con servicios dependientes.

Herramientas de ingeniería del caos:

AWS Fault Injection Service (AWS FIS) es un servicio completamente administrado para hacer experimentos de inyección de errores que puede utilizarse como parte de su canalización de CD. AWS FIS es una buena opción para usar durante los días de simulación de ingeniería del caos.

Admite la introducción simultánea de errores en distintos tipos de recursos, tales como Amazon EC2, Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) y Amazon RDS. Estos errores incluyen la terminación de los recursos, forzado de conmutación por error, estrés de CPU o memoria, limitación, latencia y pérdida de paquetes. Al estar integrado con las alarmas de Amazon CloudWatch, puede configurar las condiciones de detención como barreras de protección para revertir un experimento si provoca un impacto inesperado.




AWS Fault Injection Service se integra con recursos de AWS para permitirle ejecutar experimentos de inyección de errores para las cargas de trabajo.

También hay varias opciones de terceros para los experimentos de inyección de errores. Entre estas se incluyen herramientas de código abierto como Chaos Toolkit, Chaos Mesh y Litmus Chaos, así como opciones comerciales como Gremlin. Para ampliar el número de errores que se pueden inyectar en AWS, AWS FIS se integra con Chaos Mesh y Litmus Chaos, lo que le permite coordinar los flujos de trabajo de inyección de errores entre varias herramientas. Por ejemplo, puede ejecutar una prueba de estrés en la CPU de un pod mediante errores de Chaos Mesh o Litmus, además de terminar un porcentaje seleccionado al azar de nodos del clúster mediante acciones de error de AWS FIS.

Pasos para la implementación

1. Determine qué errores se van a utilizar en los experimentos.

   Evalúe el diseño de su carga de trabajo para la resiliencia. Estos diseños (creados mediante las prácticas recomendadas del Marco de Well-Architected) tienen en cuenta los riesgos en función de las dependencias críticas, los eventos pasados, los problemas conocidos y los requisitos de

cumplimiento. Enumere cada elemento del diseño destinado a mantener la resiliencia y los errores que pretende mitigar. Para obtener más información sobre la creación de dichas listas, consulte el documento técnico sobre revisión de la preparación operativa, que guía sobre cómo crear un proceso para evitar que se repitan incidentes anteriores. El proceso de análisis de modos de error y efectos (FMEA) le proporciona un marco para hacer un análisis de los errores de componentes y cómo afectan a la carga de trabajo. Adrian Cockcroft describe con más detalle el FMEA en Failure Modes and Continuous Resilience.

2. Asigne una prioridad a cada error.

   Comience con una categorización amplia, como alta, media o baja. Para evaluar la prioridad, hay que tener en cuenta la frecuencia del error y su impacto en la carga de trabajo global.

   Al considerar la frecuencia de un determinado error, analice los datos anteriores de esta carga de trabajo cuando estén disponibles. Si no están disponibles, utilice datos de otras cargas de trabajo que se ejecuten en un entorno similar.

   Cuando se considera el impacto de un error determinado, cuanto mayor sea el alcance del error, generalmente mayor será el impacto. También hay que tener en cuenta el diseño y la finalidad de la carga de trabajo. Por ejemplo, la capacidad de acceder a los almacenes de datos de origen es fundamental para una carga de trabajo que haga transformaciones y análisis de datos. En este caso, se daría prioridad a los experimentos de errores de acceso, así como al acceso limitado y a la inserción de latencia.

   Los análisis posteriores a los incidentes son un buen origen de datos para comprender tanto la frecuencia como el impacto de los modos de error.

   Utilice la prioridad asignada para determinar con qué errores experimentar primero y en qué orden desarrollar nuevos experimentos de inyección de errores.

3. En cada experimento que haga, siga la ingeniería del caos y el volante de resiliencia continua en la figura siguiente.




Volante de ingeniería del caos y de resiliencia continua, con el método científico de Adrian Hornsby.

a. Defina el estado estable como un resultado medible de una carga de trabajo que indica un comportamiento normal.

   Su carga de trabajo exhibe un estado estable si está operando de manera fiable y como se espera. Por tanto, valide que su carga de trabajo tenga un buen estado antes de definir el estado estable. El estado estable no significa necesariamente que no haya impacto en la carga de trabajo cuando se produce un error, ya que un cierto porcentaje en los errores podría estar dentro de los límites aceptables. El estado estable es la línea de base que observará durante el experimento, que pondrá de manifiesto las anomalías si su hipótesis definida en el siguiente paso no resulta de la forma esperada.

Por ejemplo, un estado estable de un sistema de pagos puede definirse como el procesamiento de 300 TPS con una tasa de éxito del 99 % y un tiempo de ida y vuelta de 500 ms.

b. Formule una hipótesis sobre cómo reaccionará la carga de trabajo ante el error.

Una buena hipótesis se basa en cómo se espera que la carga de trabajo mitigue el error para mantener el estado estable. La hipótesis establece que, dado el error de un tipo específico, el sistema o la carga de trabajo continuará en estado estable, porque la carga de trabajo se diseñó con mitigaciones específicas. En la hipótesis deben especificarse el tipo específico de error y las mitigaciones.

Se puede utilizar la siguiente plantilla para la hipótesis (pero también se acepta otra redacción):

> **Note**
>
> Si se produce un *error específico*, el *nombre de la carga de trabajo describirá los controles de mitigación* para mantener el *impacto de las métricas empresariales o técnicas*.

Por ejemplo:

- Si el 20 % de los nodos del grupo de nodos de Amazon EKS se eliminan, la API de creación de transacciones sigue sirviendo el percentil 99 de peticiones en menos de 100 ms (estado estable). Los nodos de Amazon EKS se recuperarán en cinco minutos y los pods se programarán y procesarán el tráfico en ocho minutos tras el inicio del experimento. Las alertas se activan en tres minutos.
- Si se produce un error de instancia de Amazon EC2, la comprobación de estado de Elastic Load Balancing del sistema de pedidos hará que Elastic Load Balancing solo envíe solicitudes a las instancias en buen estado restantes mientras Amazon EC2 Auto Scaling sustituye la instancia con error, manteniendo un aumento inferior al 0,01 % en los errores del servidor (5xx) (estado estable).
- Si se produce un error en la instancia de la base de datos principal de Amazon RDS, la carga de trabajo de recopilación de datos de la cadena de suministro se conmutará por error y se conectará a la instancia de la base de datos de Amazon RDS en espera para mantener menos de 1 minuto de errores de lectura o escritura en la base de datos (estado estable).

c. Inyecte el error para hacer el experimento.

Un experimento debe ser de manera predeterminada a prueba de errores y tolerado por la carga de trabajo. Si sabe que se producirá un error en la carga de trabajo, no haga el experimento. Debe utilizarse la ingeniería del caos para encontrar incógnitas conocidas o incógnitas desconocidas. Las incógnitas desconocidas son aspectos que conoce, pero que no comprende del todo, y las incógnitas desconocidas son aspectos que no conoce ni comprende del todo. Experimentar con una carga de trabajo que se sabe que está descompuesta no proporcionará información nueva. El experimento se debe planificar con cuidado, tener un alcance claro del impacto y proporcionar un mecanismo de retroceso que pueda aplicarse en caso de turbulencias inesperadas. Si su diligencia demuestra que la carga de trabajo debe sobrevivir al experimento, siga adelante. Hay varias opciones para inyectar los errores. Para las cargas de trabajo en AWS, AWS FIS proporciona diversas simulaciones de errores predefinidas que se denominan acciones. También puede definir acciones personalizadas que se ejecuten en AWS FIS, mediante documentos de AWS Systems Manager.

Desaconsejamos el uso de scripts personalizados para los experimentos de caos, a menos que los scripts tengan la capacidad de entender el estado actual de la carga de trabajo, sean capaces de emitir registros y proporcionen mecanismos para retrocesos y condiciones de parada cuando sea posible.

Un marco o conjunto de herramientas eficaz que apoye la ingeniería del caos debe hacer el seguimiento del estado actual de un experimento, emitir registros y proporcionar mecanismos de reversión para apoyar la ejecución controlada de un experimento. Comience con un servicio establecido como AWS FIS que permite hacer experimentos con un alcance claramente definido y mecanismos de seguridad que reviertan el experimento si este introduce turbulencias inesperadas. Para obtener información sobre una variedad más amplia de experimentos mediante AWS FIS, consulte también el laboratorio Resilient and Well-Architected Apps with Chaos Engineering. Además, AWS Resilience Hub analizará la carga de trabajo y creará experimentos que puede elegir para implementar y ejecutar en AWS FIS.

> **Note**
>
> Para cada experimento, comprenda claramente el alcance y su impacto. Recomendamos que los errores se simulen primero en un entorno de no producción antes de ejecutarlos en producción.

Cuando sea posible, los experimentos deben ejecutarse en un entorno de producción en condiciones reales, mediante implementaciones canario que permitan implementar un sistema de control y uno experimental. Ejecutar los experimentos durante las horas de menor actividad es una buena práctica para mitigar el impacto potencial cuando se experimenta por primera vez en producción. Además, si utilizar el tráfico real del cliente supone demasiado riesgo, puede hacer experimentos con tráfico sintético en la infraestructura de producción en las implementaciones de control y experimentales. Cuando no sea posible utilizar la producción, ejecute los experimentos en entornos de preproducción que sean lo más parecidos posible a la producción.

Debe establecer y supervisar las barreras de protección para garantizar que el experimento no afecte al tráfico de producción o a otros sistemas más allá de los límites aceptables. Establezca condiciones de parada para detener un experimento si alcanza un umbral en una métrica de barrera de protección que usted defina. Esto debe incluir las métricas para el estado estable de la carga de trabajo, así como la métrica en comparación con los componentes en los que está inyectando el error. Un monitor sintético (también conocido como canario de usuario) es una métrica que normalmente debe incluir como proxy de usuario. Las condiciones de parada para AWS FIS se admiten como parte de la plantilla del experimento, lo que permite hasta cinco condiciones de parada por plantilla.

Uno de los principios del caos es minimizar el alcance del experimento y su impacto:

Aunque hay que tener en cuenta algún impacto negativo a corto plazo, es responsabilidad y obligación del ingeniero del caos garantizar que las consecuencias de los experimentos se minimicen y contengan.

Un método para verificar el alcance y el impacto potencial es hacer el experimento primero en un entorno que no sea de producción, y verificar que los umbrales para las condiciones de parada se activan como se espera durante un experimento y la observabilidad está disponible para detectar una excepción, en lugar de experimentar directamente en producción.

Cuando se hagan experimentos de inyección de errores, verifique que todas las partes responsables estén bien informadas. Comuníquese con los equipos adecuados, como los equipos de operaciones, los equipos de fiabilidad del servicio y el servicio de atención al cliente, para informarles de cuándo se llevarán a cabo los experimentos y qué pueden esperar. Proporcione herramientas de comunicación a estos equipos para que informen a quienes dirigen el experimento si observan algún efecto adverso.

Debe restablecer la carga de trabajo y sus sistemas subyacentes al estado original de funcionalidad comprobada. A menudo, el diseño resistente de la carga de trabajo se autorrepara. No obstante, algunos diseños de errores o experimentos con errores pueden dejar la carga de trabajo en un estado de error inesperado. Al final del experimento, debe ser consciente de ello y restablecer la carga de trabajo y los sistemas. Con AWS FIS puede establecer una configuración de reversión (también llamada acción posterior) en los parámetros de la acción. Una acción posterior devuelve el destino al estado en el que se encontraba antes de que se ejecutara la acción. Ya sean automatizadas (como con AWS FIS) o manuales, estas acciones posteriores deben formar parte de una guía de estrategias que describa cómo detectar y gestionar los errores.

d. Verifique la hipótesis.

   Principios de la ingeniería del caos ofrece estas directrices sobre cómo verificar el estado estable de su carga de trabajo:

   Céntrese en los resultados medibles de un sistema, más que en sus atributos internos. Las mediciones de esos resultados durante un corto periodo constituyen una aproximación al estado estable del sistema. El rendimiento global del sistema, las tasas de error y los percentiles de latencia podrían ser métricas de interés que representen el comportamiento del estado estable. Al centrarse en los patrones de comportamiento sistémico durante los experimentos, la ingeniería del caos verifica que el sistema funcione, en lugar de intentar validar su funcionamiento.

   En nuestros dos ejemplos anteriores, incluimos las métricas de estado estable de menos del 0,01 % de aumento de errores del servidor (5xx) y menos de un minuto de errores de lectura y escritura en la base de datos.

   Los errores 5xx son una buena métrica porque son una consecuencia del modo de error que experimentará directamente un cliente de la carga de trabajo. La medición de los errores de la base de datos es buena como consecuencia directa del error, pero también debe complementarse con una medición del impacto en el cliente, como las solicitudes con errores de los clientes o los errores que aparecen en el cliente. Además, incluya un monitor sintético (también conocido como canario de usuario) en cualquier API o URI al que acceda directamente el cliente de su carga de trabajo.

e. Mejora del diseño de la carga de trabajo para la resiliencia.

Si el estado estable no se mantuvo, investigue cómo se puede mejorar el diseño de la carga de trabajo para mitigar el error y aplique las prácticas recomendadas del pilar de fiabilidad de AWS Well-Architected. Puede encontrar orientación y recursos adicionales en la AWS Builders' Library, que contiene artículos sobre cómo mejorar las comprobaciones de estado o utilizar los reintentos con retroceso en el código de la aplicación, entre otros temas.

Una vez aplicados estos cambios, vuelva a hacer el experimento (mostrado por la línea de puntos en el volante de ingeniería del caos) para determinar su eficacia. Si el paso de verificación indica que la hipótesis es cierta, entonces la carga de trabajo estará en estado estable y el ciclo continúa.

4. Lleve a cabo experimentos periódicos.

   Un experimento de caos es un ciclo y los experimentos deben hacerse regularmente como parte de la ingeniería del caos. Después de que una carga de trabajo cumpla con la hipótesis del experimento, este debe automatizarse para ejecutarse continuamente como parte de la regresión de su canalización de CI/CD. Para obtener información sobre cómo hacerlo, consulte este blog sobre cómo hacer experimentos de AWS FIS con AWS CodePipeline. Este laboratorio sobre experimentos recurrentes de AWS FIS en una canalización de CI/CD le permite trabajar de forma práctica.

   Los experimentos de inyección de errores también forman parte de los días de simulación (consulte REL12-BP05 Planificación periódica de días de juego). En los días de simulación se simula un error o evento para verificar los sistemas, los procesos y las respuestas de los equipos. El objetivo es llevar a cabo las acciones que llevaría a cabo el equipo si se produjera un evento excepcional.

5. Capture y almacene los resultados de los experimentos.

   Los resultados de los experimentos de inyección de errores deben capturarse y persistir. Incluya todos los datos necesarios (como el tiempo, la carga de trabajo y las condiciones) para poder analizar posteriormente los resultados y las tendencias del experimento. Algunos ejemplos de resultados pueden ser capturas de pantalla de paneles, volcados en CSV de la base de datos de métricas o un registro escrito a mano de los eventos y las observaciones del experimento. El registro de experimentos con AWS FIS puede ser parte de esta captura de datos.

Recursos

Prácticas recomendadas relacionadas:

- REL08-BP03 Integración de las pruebas de resiliencia como parte de la implementación
- REL13-BP03 Prueba de la implementación de recuperación de desastres para validarla

Documentos relacionados:

- Qué es AWS Fault Injection Service?
- Qué es AWS Resilience Hub?
- Principios de la ingeniería del caos
- Chaos Engineering: Planning your first experiment
- Resilience Engineering: Learning to Embrace Failure
- Chaos Engineering stories
- Evitar los planes alternativos en los sistemas distribuidos
- Canary Deployment for Chaos Experiments

Videos relacionados:

- AWS re:Invent 2020: Testing resiliency using chaos engineering (ARC316)
- AWS re:Invent 2019: Improving resiliency with chaos engineering (DOP309-R1)
- AWS re:Invent 2019: Performing chaos engineering in a serverless world (CMY301)

Herramientas relacionadas:

- AWS Fault Injection Service
- AWS Marketplace: Gremlin Chaos Engineering Platform
- Chaos Toolkit
- Chaos Mesh
- Litmus

REL12-BP05 Planificación periódica de días de juego

Organice días de juego para poner en práctica con regularidad sus procedimientos de respuesta a los eventos y deficiencias que afecten a la carga de trabajo. Incluya a los mismos equipos que se encargarían de gestionar los escenarios de producción. Estos ejercicios ayudan a aplicar medidas

para evitar que los eventos de producción afecten a los usuarios. Si practica sus procedimientos de respuesta en condiciones realistas, podrá identificar y abordar cualquier laguna o punto débil antes de que se produzca un suceso real.

En los días de juego, se simulan eventos en entornos de producción para probar los sistemas, los procesos y las respuestas de los equipos. El objetivo es poner en práctica las acciones que llevaría a cabo el equipo si se produjera realmente un evento. Estos ejercicios lo ayudan a comprender dónde se pueden efectuar mejoras y a desarrollar la experiencia de la organización a la hora de gestionar eventos y deficiencias. Deben hacerse periódicamente, para que el equipo desarrolle hábitos sobre cómo responder.

Los días de juego preparan a los equipos para que puedan abordar los eventos de producción con mayor confianza. Los equipos que tienen mucha experiencia son más capaces de detectar y responder rápidamente a varios escenarios. Esto se traduce en una postura de preparación y resiliencia significativamente mejorada.

Resultado deseado: organiza los días de juego de resiliencia de forma constante y programada. Estos días de juego se integran armoniosamente en la actividad empresarial. Su organización ha creado una cultura de preparación y, cuando se producen problemas de producción, sus equipos están bien preparados para responder con eficacia, resolver los problemas de manera eficiente y mitigar las repercusiones para los clientes.

Patrones comunes de uso no recomendados:

- Documenta los procedimientos, pero no los llega a poner en práctica.
- En los ejercicios de prueba se excluye a los responsables de la toma de decisiones empresariales.
- Organiza un día de juego, pero no informa a todas las partes interesadas pertinentes.
- Se centra únicamente en los fallos técnicos, pero no incluye a las partes interesadas de la empresa.
- No incorpora las lecciones aprendidas de los días de juego en sus procesos de recuperación.
- Culpa a los equipos de los fallos o los errores.

Beneficios de establecer esta práctica recomendada:

- Mejorar las habilidades de respuesta: durante los días de juego, los equipos practican sus tareas y ponen a prueba sus mecanismos de comunicación durante los eventos simulados, lo que permite una respuesta más coordinada y eficiente en situaciones de producción.

- Identificar y abordar las dependencias: los entornos complejos suelen conllevar dependencias intrincadas entre varios sistemas, servicios y componentes. Los días de juego pueden ayudar a identificar y abordar estas dependencias, y a comprobar que los manuales de procedimientos abordan sus sistemas y servicios críticos debidamente, así como que puedan ampliarse o recuperarse de forma oportuna.
- Fomentar una cultura de resiliencia: los días de juego pueden ayudar a cultivar una mentalidad de resiliencia en una organización. Al implicar a equipos multidisciplinarios y a las partes interesadas, estos ejercicios fomentan la concienciación, la colaboración y una comprensión compartida de la importancia de la resiliencia en toda la organización.
- Mejorar y adaptarse continuamente: los días de juego practicados con regularidad ayudan a evaluar y adaptar continuamente las estrategias de resiliencia, de modo que sigan siendo relevantes y eficaces aunque cambien las circunstancias.
- Aumentar la confianza en el sistema: los días de juego ejecutados correctamente pueden ayudar a generar confianza en la capacidad del sistema para resistir las interrupciones y recuperarse de ellas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Una vez que haya diseñado e implementado las medidas de resiliencia necesarias, organice un día de juego para comprobar que todo funciona según lo previsto durante la fase de producción. Un día de juego, sobre todo el primero, debe incluir a todos los miembros del equipo, y se debe informar con antelación a todas las partes interesadas y participantes sobre la fecha, la hora y los escenarios simulados.

Durante el día de juego, los equipos participantes simulan varios eventos y posibles escenarios de acuerdo con los procedimientos prescritos. Los participantes monitorean y evalúan de cerca el impacto de estos eventos simulados. Si el sistema funciona según lo diseñado, los mecanismos automatizados de detección, escalado y recuperación automática deberían activarse y tener un impacto mínimo o nulo en los usuarios. Si el equipo observa algún impacto negativo, anula la prueba y soluciona los problemas identificados, ya sea mediante medios automatizados o mediante una intervención manual documentada en los manuales aplicables.

Para mejorar continuamente la resiliencia, es fundamental documentar e incorporar las lecciones aprendidas. Este proceso es un ciclo de retroalimentación que recopila de forma sistemática la

información obtenida durante los días de juego y la utiliza para mejorar los sistemas, los procesos y las capacidades de los equipos.

Para ayudarlo a reproducir situaciones reales en las que los componentes o servicios del sistema podrían fallar inesperadamente, utilice la simulación de fallos como ejercicio para un día de juego. Los equipos pueden probar la resiliencia y la tolerancia a los fallos de sus sistemas y simular sus procesos de respuesta y recuperación ante incidentes en un entorno controlado.

En AWS, los días de juego se pueden llevar a cabo con réplicas de su entorno de producción utilizando la infraestructura como código. Mediante este proceso, puede realizar las pruebas en un entorno seguro que se parece mucho a su entorno de producción. Puede utilizar AWS Fault Injection Service para crear diferentes escenarios de error. Use servicios como Amazon CloudWatch y AWS X-Ray para monitorear el comportamiento del sistema durante los días de juego. Use AWS Systems Manager para administrar y ejecutar guías de estrategia, y utilice AWS Step Functions para organizar los flujos de trabajo recurrentes del día del juego.

Pasos para la implementación

- Establezca un programa para los días de juego: desarrolle un programa estructurado que defina la frecuencia, el alcance y los objetivos de los días de juego. Incluya a las principales partes interesadas y a los expertos en la materia en la planificación y ejecución de estos ejercicios.
- Prepare el día de juego:
  1. Identifique los principales servicios críticos para la empresa en los que se centrará el día de juego. Catalogue y mapee las personas, los procesos y las tecnologías que respaldan esos servicios.
  2. Establezca la agenda para el día de juego y prepare a los equipos implicados para que participen en el evento. Prepare sus servicios de automatización para simular los escenarios planificados y ejecutar los procesos de recuperación adecuados. Los servicios de AWS, como AWS Fault Injection Service, AWS Step Functions y AWSSystems Manager pueden ayudarlo a automatizar varios aspectos de los días de juego, como la inyección de errores y el inicio de las acciones de recuperación.
- Ejecute su simulación: el día de juego, ejecute el escenario planificado. Observe y documente cómo reaccionan las personas, los procesos y las tecnologías ante el evento simulado.
- Realice revisiones después del ejercicio: después del día de juego, realice una sesión retrospectiva para repasar las lecciones aprendidas. Identifique las áreas de mejora y cualquier acción necesaria para mejorar la resiliencia operativa. Documente sus resultados y realice

un seguimiento de los cambios necesarios para mejorar sus estrategias de resiliencia y su preparación para llevarlas a cabo.

Recursos

Prácticas recomendadas relacionadas:

- REL12-BP01 Uso de manuales de estrategias para investigar los errores
- REL12-BP04 Pruebas de resiliencia mediante ingeniería del caos
- OPS04-BP01 Identificación de los indicadores clave de rendimiento
- OPS07-BP03 Uso de manuales de procedimientos para llevar a cabo los procedimientos
- OPS10-BP01 Uso de un proceso para la administración de eventos, incidentes y problemas

Documentos relacionados:

- ¿Qué es AWS GameDay?

Videos relacionados:

- AWS re:Invent 2023 - Practice like you play: How Amazon scales resilience to new heights

Ejemplos relacionados:

- AWS Workshop - Navigate the storm: Unleashing controlled chaos for resilient systems
- Build Your Own Game Day to Support Operational Resilience

## REL 13. ¿Cómo planifica la recuperación de desastres (DR)?

Disponer de copias de seguridad y de componentes de cargas de trabajo redundantes es el principio de su estrategia de DR. El RTO y el RPO son los objetivos de restauración de las cargas de trabajo. Estos se definen en función de las necesidades del negocio. Implemente una estrategia para satisfacer estos objetivos teniendo en cuenta las ubicaciones y la función de los recursos de las cargas de trabajo y los datos. La probabilidad de una interrupción y el costo de recuperación son también factores clave que ayudan a conocer el valor empresarial de proporcionar recuperación de desastres para una carga de trabajo.

Prácticas recomendadas

- REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos
- REL13-BP02 Uso de estrategias de recuperación definidas para cumplir los objetivos de recuperación
- REL13-BP03 Prueba de la implementación de recuperación de desastres para validarla
- REL13-BP04 Administración de la desviación de la configuración en el sitio o región de DR
- REL13-BP05 Automatización de la recuperación

REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos

Los errores pueden afectar a su empresa de varias maneras. En primer lugar, los errores pueden provocar la interrupción del servicio (tiempo de inactividad). En segundo lugar, pueden provocar la pérdida de datos, su incoherencia o su obsolescencia. Para orientar la forma de responder y recuperarse ante los errores, defina un objetivo de tiempo de recuperación (RTO) y un objetivo de punto de recuperación (RPO) para cada carga de trabajo. El objetivo de tiempo de recuperación (RTO) es el tiempo máximo aceptable entre la interrupción del servicio y su restablecimiento. El objetivo de punto de recuperación (RPO) es el tiempo máximo aceptable tras el último punto de recuperación de datos.

Resultado deseado: cada carga de trabajo tiene un RTO y un RPO designados en función de las circunstancias técnicas y de repercusión para la empresa.

Patrones comunes de uso no recomendados:

- No ha designado objetivos de recuperación.
- Selecciona objetivos de recuperación arbitrarios.
- Selecciona objetivos de recuperación demasiado permisivos que no cumplen los objetivos de la empresa.
- No ha evaluado las consecuencias del tiempo de inactividad y la pérdida de datos.
- Selecciona objetivos de recuperación poco realistas, como un tiempo de recuperación nulo o una pérdida de datos nula, que la configuración de la carga de trabajo tal vez no pueda alcanzar.
- Selecciona objetivos de recuperación más estrictos que los objetivos empresariales reales. Esto obliga a hacer implementaciones de recuperación más costosas y complejas que lo que necesita la carga de trabajo.

- Selecciona objetivos de recuperación incompatibles con los de una carga de trabajo dependiente.
- No tiene en cuenta los requisitos normativos y de cumplimiento.

Beneficios de establecer esta práctica recomendada: al definir los RTO y los RPO para sus cargas de trabajo, establece objetivos de recuperación claros y medibles en función de las necesidades de su empresa. Una vez que haya establecido esos objetivos, puede crear planes de recuperación ante desastres (DR) para alcanzar dichos objetivos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Elabore una matriz o una hoja de trabajo que le sirva de guía para planificar la recuperación ante desastres. En su matriz, cree diferentes categorías o niveles de carga de trabajo en función de cómo afecten a su empresa (de nivel crítico, alto, medio y bajo, por ejemplo) y los RTO y RPO asociados a los que desee asignar cada uno de ellos. En la siguiente matriz se muestra un ejemplo (tenga en cuenta que los valores de RTO y RPO pueden diferir) que puede seguir:

| Matriz de recuperación de desastres | | | | | | |
|---|---|---|---|---|---|---|
| | | Objetivo de punto de recuperación | | | | |
| | | < 1 minuto | < 1 hora | < 6 horas | < 1 día | Más de 1 día |
| Objetivo de tiempo de recuperación | < 10 minutos | Crítico | Crítico | Alto | Medio | Medio |
| | < 2 horas | Crítico | Alto | Medio | Medio | Bajo |
| | < 8 horas | Alto | Medio | Medio | Bajo | Bajo |
| | < 24 horas | Medio | Medio | Bajo | Bajo | Bajo |
| | Más de 24 horas | Medio | Bajo | Bajo | Bajo | Bajo |

Ejemplo de matriz de recuperación ante desastres

Para cada carga de trabajo, debe comprender como afectan el tiempo de inactividad y la pérdida de datos a su empresa. Por lo general, el impacto aumenta con el tiempo de inactividad y la pérdida de datos, pero la forma del impacto puede variar según el tipo de carga de trabajo. Por ejemplo, un tiempo de inactividad de hasta una hora puede tener un impacto mínimo, pero después ese impacto podría aumentar rápidamente. Puede afectar de muchas maneras; por ejemplo, a nivel financiero (como la pérdida de ingresos), operativo (como la falta de nóminas o la disminución de

la productividad) o normativo, así como en la reputación (incluida la pérdida de la confianza de los clientes). Una vez completado, asigne la carga de trabajo al nivel adecuado.

Tenga en cuenta las siguientes preguntas al analizar el impacto de un error:

1. ¿Durante cuánto tiempo se puede desactivar la carga de trabajo como máximo antes de que afecte gravemente a la empresa?
2. ¿En qué medida afecta a la empresa una interrupción de la carga de trabajo y de qué modo? Tenga en cuenta todos los tipos de impacto, a nivel financiero, operativo, normativo y en la reputación.
3. ¿Cuántos datos se pueden perder como máximo antes de que la empresa se vea gravemente afectada?
4. ¿Se pueden volver a crear los datos perdidos a partir de otras fuentes (también conocidos como datos derivados)? Si es así, incluya también los RPO de todos los datos de origen utilizados para volver a crea los datos de la carga de trabajo.
5. ¿Cuáles son los objetivos de recuperación y las expectativas de disponibilidad de las cargas de trabajo de las que depende (en sentido descendente)? Los objetivos de su carga de trabajo deben poder alcanzarse gracias a las capacidades de recuperación de sus dependencias posteriores. Puede usar soluciones alternativas o mitigaciones de las dependencias posteriores que puedan mejorar la capacidad de recuperación de esta carga de trabajo.
6. ¿Cuáles son los objetivos de recuperación y las expectativas de disponibilidad de las cargas de trabajo de las que depende (en sentido ascendente)? Los objetivos de la carga de trabajo anterior pueden requerir que esta carga de trabajo tenga capacidades de recuperación más estrictas de lo que parece a primera vista.
7. ¿Existen diferentes objetivos de recuperación en función del tipo de incidente? Por ejemplo, puede haber diferentes RTO y RPO en función de que el incidente afecte a una zona de disponibilidad o a toda una región.
8. ¿Cambian sus objetivos de recuperación durante determinados eventos o épocas del año? Por ejemplo, es posible que tenga diferentes RTO y RPO en función de las temporadas de compras navideñas, los eventos deportivos, las rebajas especiales y los lanzamientos de nuevos productos.
9. ¿Cómo se acompasan los objetivos de recuperación con las posibles estrategias de recuperación ante desastres de su línea de negocio y de su organización?
10. ¿Hay que tener en cuenta las ramificaciones legales o contractuales? Por ejemplo, ¿tiene una obligación contractual de prestar un servicio con un RTO o RPO determinado? ¿A qué sanciones podría enfrentarse por no cumplirlas?

11¿Tiene obligación de mantener la integridad de los datos para cumplir con los requisitos normativos o de conformidad?

La siguiente hoja de trabajo puede ayudarlo a evaluar cada carga de trabajo. Puede modificar esta hoja de trabajo para adaptarla a sus necesidades específicas, por ejemplo, agregar otras preguntas.

| Paso 2: Preguntas principales | ¿Se aplica a la carga de trabajo? | RTO de carga de trabajo | RPO de carga de trabajo | Ajuste de RTO | Ajuste de RPO | Instrucciones |
|---|---|---|---|---|---|---|
| [1] tiempo máximo que puede estar inoperativa la carga de trabajo | | | | | | medido en tiempo desde el inicio de la interrupción hasta la recuperación |
| [2] cantidad máxima de datos que se pueden perder | | | | | | medido en tiempo desde el último conjunto de datos restaurable correcto conocido |
| [3a] dependencias upstream | | | | | | introduzca los objetivos de recuperación upstream más estrictos |
| [3b] dependencias downstream | | | | | | introduzca los objetivos de recuperación downstream menos estrictos |
| [3a] dependencias upstream reconciliadas | | | | | | Si el valor upstream es menor que los valores actuales y el valor downstream es mayor, |
| [3b] dependencias downstream reconciliadas | | | | | | trabaje con las dependencias para reconciliarlas e introducir aquí los valores reconciliados |
| [3] dependencias | | | | | | reduzca los valores para ajustarse a las dependencias upstream o auméntelos en función de las capacidades de dependencias downstream |
| | | | | | | |
| **Paso 2: Preguntas adicionales** | | | | | | Indique si se aplica la pregunta. Si no se aplica, omítala |
| RTO/RPO base | | | | | | Traslade los valores de RTO y RPO de arriba aquí |
| [4] tipo de interrupción | [ ]S / [ ]N | | | | | Introduzca los objetivos de recuperación del tipo de evento con los requisitos más estrictos |
| [5] objetivos específicos basados en el tiempo | [ ]S / [ ]N | | | | | Introduzca los objetivos de recuperación de tiempo con los requisitos más estrictos |
| [6] clientes afectados | [ ]S / [ ]N | | | | | Realice un gráfico de los clientes afectados en función del tiempo de inactividad o de los datos perdidos. Utilícelo para introducir los RTO y RPO máximos permisibles en función del impacto sobre los clientes |
| [7] impacto reputacional | [ ]S / [ ]N | | | | | Trabaje con la empresa para determinar los RTO y RPO máximos en función del impacto sobre la reputación |
| [8] impacto operativo | [ ]S / [ ]N | | | | | Introduzca los RTO y RPO máximos en función del impacto operativo |
| [9] alineación organizativa | [ ]S / [ ]N | | | | | Introduzca los RTO y RPO máximos para cargas de trabajo de este tipo según los requisitos organizativos y de LOB |
| [10] obligaciones contractuales | [ ]S / [ ]N | | | | | Introduzca los RTO y RPO máximos en función de las obligaciones contractuales |
| [11] conformidad normativa | [ ]S / [ ]N | | | | | Introduzca los RTO y RPO máximos en función de la conformidad normativa pertinente |
| objetivo basado en preguntas adicionales | | | | | | Tome el valor mínimo (valor más estricto) de entre Q 4-11 e introdúzcalo aquí |
| objetivo ajustado | | | | | | Si no se puede dar cabida a los objetivos de la línea anterior, colabore con los interesados para transigir en los límites e introduzca aquí un nuevo valor mínimo |
| | | | | | | |
| RTO/RPO ajustados | | | | | | Introduzca los valores base de RPO/RTO o el objetivo ajustado, el menor de los valores |
| | | | | | | |
| **Paso 3** | | | | | | |
| Mapa a categoría o nivel predefinidos | | | | | | Ajuste ambos valores a la baja (lo más estricto) para alinearlos con el nivel definido más cercano |

Hoja de trabajo

Pasos para la implementación

1. Identifique cuáles son las partes interesadas de la empresa y los equipos técnicos responsables de cada carga de trabajo y contacte con ellos.
2. Cree categorías o niveles de importancia crítica del impacto de la carga de trabajo en su organización. Entre las categorías de ejemplo se incluyen las siguientes: crítico, alto, medio y bajo. Para cada categoría, elija un RTO y un RPO que reflejen los objetivos y requisitos de su empresa.
3. Asigne una de las categorías de impacto que ha creado en el paso anterior a cada carga de trabajo. Para decidir cómo se asigna una carga de trabajo a una categoría, tenga en cuenta la importancia de la carga de trabajo para la empresa y el impacto de la interrupción o la pérdida de

datos. Las preguntas anteriores pueden guiarle. De este modo, obtendrá un RTO y un RPO para cada carga de trabajo.

4. Tenga en cuenta el RTO y el RPO para cada carga de trabajo determinada en el paso anterior. Incluya a los equipos técnicos y empresariales de la carga de trabajo para determinar si se deben ajustar los objetivos. Por ejemplo, las partes interesadas de la empresa podrían determinar que se requieren objetivos más estrictos. Si lo prefiere, los equipos técnicos podrían determinar si los objetivos deberían modificarse para que se puedan alcanzar con los recursos disponibles y las limitaciones tecnológicas.

Recursos

Prácticas recomendadas relacionadas:

- REL09-BP04 Recuperación periódica de los datos para verificar la integridad de la copia de seguridad y los procesos
- REL12-BP01 Uso de manuales de estrategias para investigar los errores
- REL13-BP02 Uso de estrategias de recuperación definidas para cumplir los objetivos de recuperación
- REL13-BP03 Prueba de la implementación de recuperación ante desastres para validarla

Documentos relacionados:

- AWS Architecture Blog: Disaster Recovery Series
- Recuperación de desastres de cargas de trabajo en AWS: recuperación en la nube (documento técnico de AWS)
- Administrar las políticas de resiliencia con AWS Resilience Hub
- Socio de APN: socios que pueden ayudar con la recuperación ante desastres
- AWS Marketplace: productos que pueden usarse para la recuperación ante desastres

Videos relacionados:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications
- Disaster Recovery of Workloads on AWS

### REL13-BP02 Uso de estrategias de recuperación definidas para cumplir los objetivos de recuperación

Defina una estrategia de recuperación de desastres (DR) que se ajuste a los objetivos de recuperación de su carga de trabajo. Elija una estrategia como copia de seguridad y restauración, estado de espera (activa/pasiva) o activa/activa.

Resultado deseado: para cada carga de trabajo, hay una estrategia de DR definida e implementada que permite que la carga de trabajo alcance los objetivos de DR. Las estrategias de DR entre cargas de trabajo emplean patrones reutilizables (como las estrategias descritas anteriormente).

Patrones comunes de uso no recomendados:

- Implementar procedimientos de recuperación incoherentes para cargas de trabajo con objetivos de DR similares.
- Dejar la estrategia de DR para implementarla ad hoc cuando se produzca un desastre.
- No tener un plan de recuperación de desastres.
- Depender de las operaciones del plano de control durante la recuperación.

Beneficios de establecer esta práctica recomendada:

- El uso de estrategias de recuperación definidas le permite emplear herramientas y procedimientos de prueba comunes.
- El uso de estrategias de recuperación definidas mejora el intercambio de conocimiento entre equipos y la implementación de la DR en las cargas de trabajo que se encuentran bajo su responsabilidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto. Sin una estrategia de DR planificada, implementada y probada, es poco probable que consiga sus objetivos de recuperación en caso de desastre.

Guía para la implementación

Una estrategia de DR depende de la capacidad de poner en marcha su carga de trabajo en un sitio de recuperación si su ubicación principal deja de estar disponible para la ejecución de dicha carga de trabajo. Los objetivos de recuperación más comunes son el RTO y el RPO, como explicamos en REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos.

Una estrategia de DR en varias zonas de disponibilidad (AZ) en una única Región de AWS puede ofrecer mitigación contra eventos de desastres como incendios, inundaciones y cortes de suministro eléctrico considerables. Si es necesario implementar medidas de protección contra un evento poco probable que evite que su carga de trabajo pueda ejecutarse en una Región de AWS determinada, puede seguir una estrategia de DR que abarque múltiples regiones.

A la hora de diseñar una estrategia de DR en varias regiones, debe elegir una de las siguientes estrategias. Se enumeran en orden ascendente de costo y complejidad, y en orden descendente de RTO y RPO. La región de recuperación se refiere a una Región de AWS distinta de la principal que se utiliza para la carga de trabajo.




Figura 17: Estrategias de recuperación de desastres (DR)

- Copia de seguridad y restauración (RPO en horas, RTO en 24 horas o menos): haga una copia de seguridad de los datos y aplicaciones en la región de recuperación. El uso de copias de seguridad automatizadas o continuas permitirá la recuperación en un momento dado (PITR), lo que puede reducir el RPO a hasta 5 minutos en algunos casos. En caso de desastre, implementará la infraestructura (mediante la infraestructura como código para reducir el RTO), implementará el código y restaurará los datos desde la copia de seguridad para recuperarse del desastre en la región de recuperación.
- Luz piloto (RPO en minutos, RTO en decenas de minutos): aprovisione una copia de la infraestructura de carga de trabajo principal en la región de recuperación. Replique los datos en la región de recuperación y cree allí copias de seguridad de estos. Los recursos necesarios para permitir la replicación y copia de seguridad de los datos, como el almacenamiento de bases de

datos y objetos, están siempre disponibles. Otros elementos, como los servidores de aplicaciones o la computación sin servidor, no se implementan, pero pueden crearse cuando sea necesario con la configuración y el código de aplicación pertinentes.

- Espera semiactiva (RPO en segundos, RTO en minutos): mantenga una versión reducida, pero completamente funcional de la carga de trabajo que se ejecute siempre en la región de recuperación. Los sistemas críticos se duplican en su totalidad y siempre están activos, pero con una flota reducida. Los datos se replican y están activos en la región de recuperación. Cuando llegue el momento de la recuperación, el sistema se ampliará rápidamente para asumir la carga de producción. Cuanto mayor sea la escala de la espera semiactiva, menor será el RTO y la fiabilidad del plano de control. Cuando se amplía por completo, esto se conoce como espera activa.
- Activo-activo en varias regiones (varios sitios) (RPO cercano a cero, RTO potencialmente nulo): la carga de trabajo se implementa en varias Regiones de AWS y atiende de manera activa el tráfico procedente de estas. Esta estrategia requiere que sincronice datos entre regiones. Los posibles conflictos causados por escrituras en el mismo registro en dos réplicas regionales diferentes deben evitarse o gestionarse, lo que puede resultar complejo. La replicación de datos es útil para la sincronización de datos y es una protección ante algunos tipos de desastres, pero no ante el daño o la destrucción de datos, a no ser que su solución incluya también opciones para una recuperación en un momento dado.

> **Note**
>
> La diferencia entre la luz piloto y la espera semiactiva a veces puede ser difícil de comprender. Ambos métodos incluyen un entorno en su región de recuperación con copias de los activos de su región principal. La distinción es que la luz piloto no puede procesar solicitudes sin tomar primero medidas adicionales, mientras que la espera semiactiva puede gestionar el tráfico (a niveles de capacidad reducidos) inmediatamente. La luz piloto exige que active servidores, posiblemente que implemente infraestructura adicional (no principal) y que escale verticalmente, mientras que la espera semiactiva solo requiere que escale verticalmente (ya está todo implementado y en ejecución). Elija una de estas opciones en función de sus necesidades de RTO y RPO.
>
> Cuando el costo sea una preocupación y desee alcanzar unos objetivos de RPO y RTO similares a los definidos en la estrategia de espera semiactiva, podría plantearse soluciones nativas en la nube, como AWS Elastic Disaster Recovery, que adoptan el enfoque de luz piloto y ofrecen objetivos de RPO y RTO mejorados.

Pasos para la implementación

1. Determine una estrategia de recuperación de desastres que satisfaga los requisitos de recuperación de esta carga de trabajo.

   La selección de una estrategia de DR requiere alcanzar un punto de equilibrio entre la reducción del tiempo de inactividad y la pérdida de datos (RTO y RPO) y los costos y la complejidad de implementar la estrategia. Debe evitar implementar una estrategia que sea más exigente de lo necesario, ya que esto supone costos innecesarios.

   Por ejemplo, en el siguiente diagrama, la empresa ha determinado su RTO máximo permisible y el límite de gasto en su estrategia de restauración del servicio. Dados los objetivos de la empresa, las estrategias de DR de luz piloto o espera semiactiva satisfarán tanto el RTO como los criterios de costo.

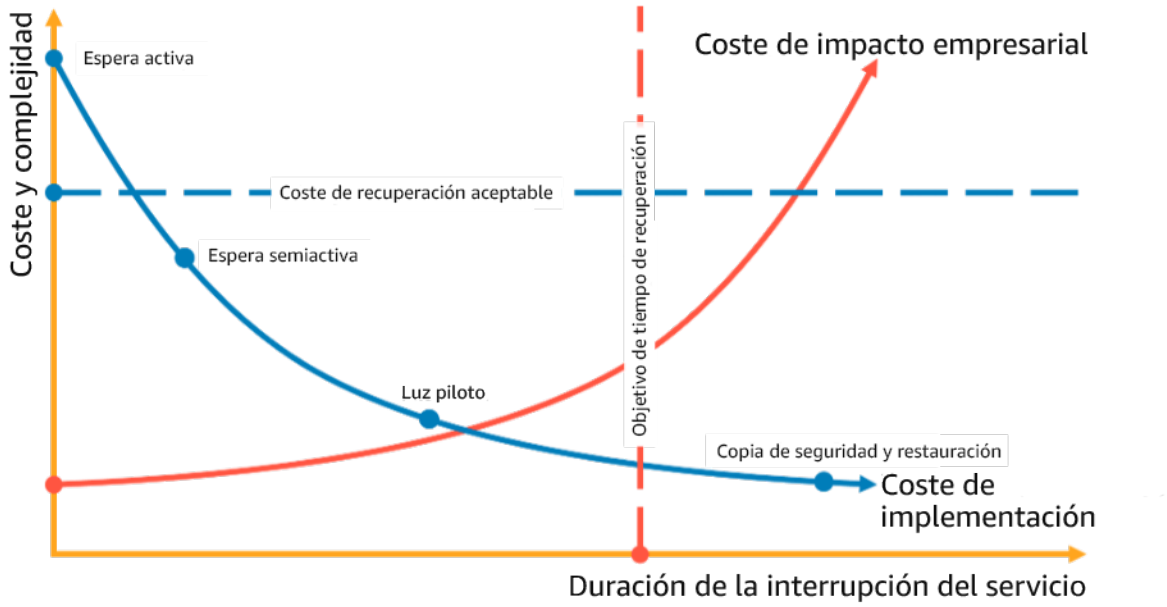



   Figura 18: Selección de una estrategia de DR en función del RTO y costo

   Para más información, consulte el Plan de continuidad del negocio (BCP).

2. Revise los patrones de cómo se puede implementar la estrategia de DR seleccionada.

   Este paso implica comprender cómo implementará la estrategia seleccionada. Las estrategias se explican mediante las Regiones de AWS para determinar un sitio principal y otro de recuperación. Sin embargo, también puede decidir utilizar zonas de disponibilidad en una única región como estrategia de DR, que utiliza los elementos de varias de estas estrategias.

En los siguientes pasos, puede aplicar la estrategia a su carga de trabajo específica.

Copia de seguridad y restauración

La copia de seguridad y restauración son la estrategia menos compleja de implementar, pero requerirán más tiempo y esfuerzo para restaurar la carga de trabajo, lo que generará un RTO y un RPO más altos. Se recomienda hacer siempre copias de seguridad de los datos y copiarlas en otro sitio (por ejemplo, otra Región de AWS).

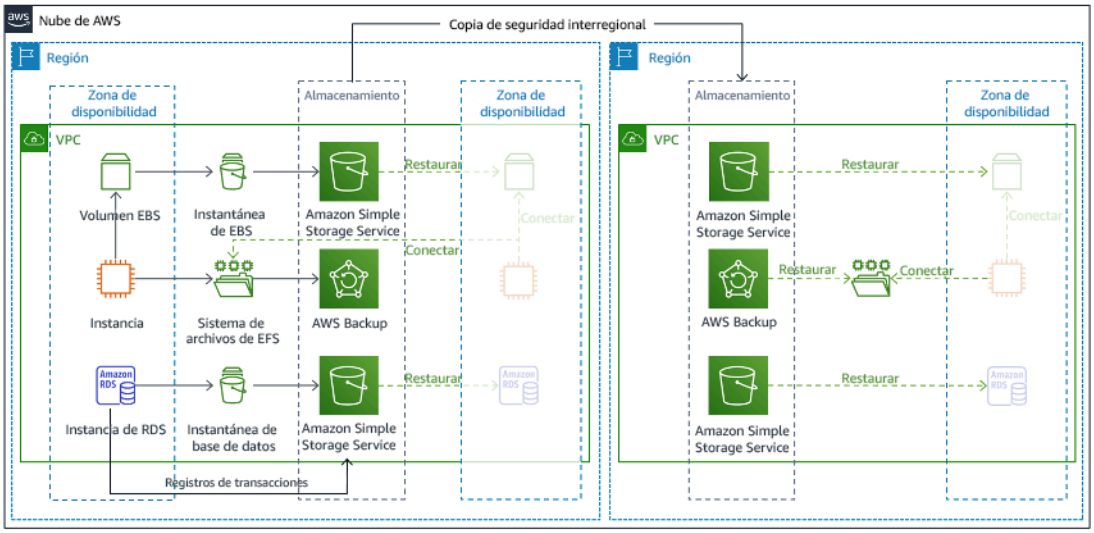



Figura 19: Arquitectura de copia de seguridad y restauración

Para obtener más información sobre esta estrategia, consulte Disaster Recovery (DR) Architecture on AWS, Part II: Backup and Restore with Rapid Recovery.

Luz piloto

Con el enfoque de luz piloto, se replican los datos de la región principal a la región de recuperación. Los recursos principales utilizados para la infraestructura de la carga de trabajo se implementan en la región de recuperación; sin embargo, se siguen necesitando recursos adicionales y las dependencias pertinentes para que esta pila sea funcional. Por ejemplo, en la figura 20 no se implementan instancias de computación.

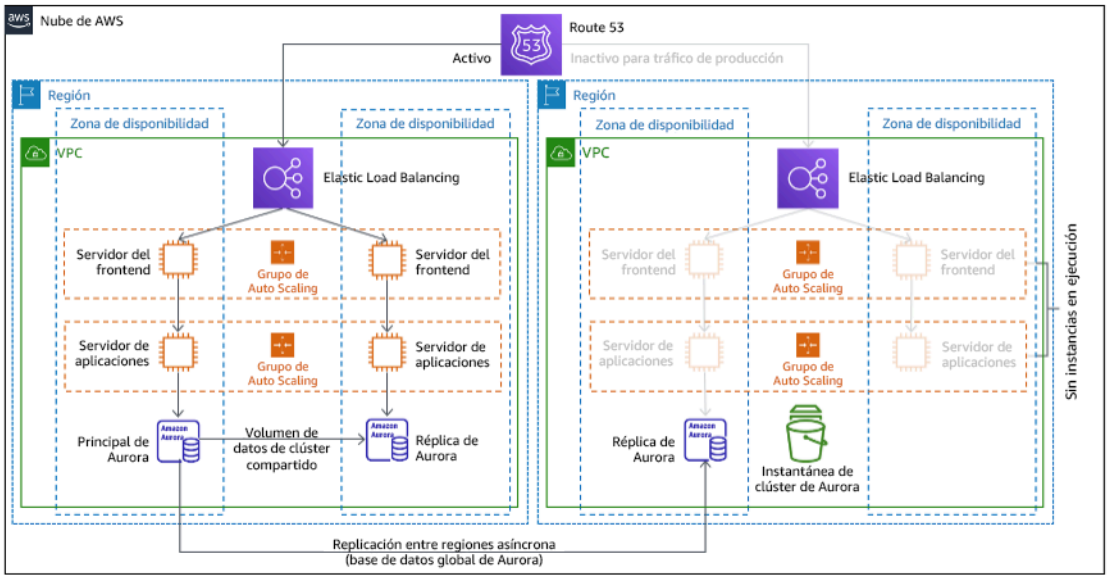



Figura 20: Arquitectura de luz piloto

Para obtener más información sobre esta estrategia, consulte Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby.

Espera semiactiva

El enfoque de espera semiactiva implica garantizar que haya una copia reducida, pero completamente funcional, del entorno de producción en otra región. Este enfoque extiende el concepto de luz piloto y reduce el tiempo de recuperación, ya que su carga de trabajo tiene disponibilidad permanente en otra región. Si la región de recuperación se implementa al máximo de su capacidad, esto se conoce como modo de espera activa.

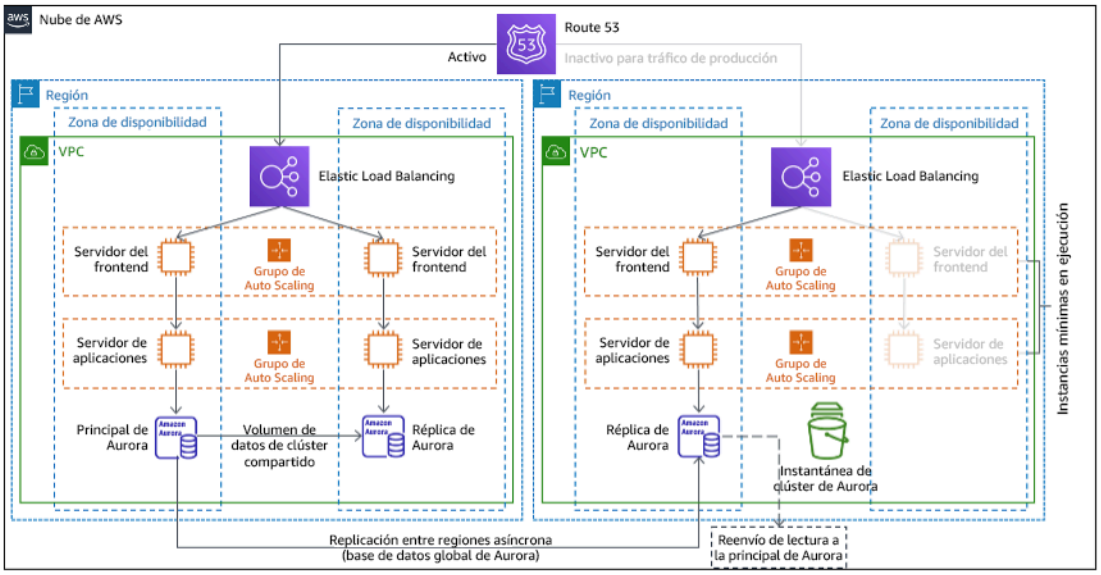



Figura 21: Arquitectura de espera semiactiva

El uso de los enfoques de espera semiactiva o luz piloto requiere escalar verticalmente los recursos en la región de recuperación. Para comprobar que la capacidad esté disponible cuando sea necesaria, considere la posibilidad de utilizar reservas de capacidad para las instancias de EC2. Si se utiliza AWS Lambda, la simultaneidad aprovisionada puede proporcionar entornos de tiempo de ejecución para que estén preparados para responder a las invocaciones de la función.

Para obtener más información sobre esta estrategia, consulte Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby.

Activa-activa multisitio

Puede ejecutar la carga de trabajo de forma simultánea en varias regiones como parte de una estrategia activa-activa multisitio. La estrategia activa-activa multisitio suministra tráfico desde todas las regiones en las que se implementa. Los clientes podrían seleccionar esta estrategia por motivos ajenos a la DR. Se puede utilizar para aumentar la disponibilidad o cuando se implementa una carga de trabajo para una audiencia global (para colocar el punto de conexión más cerca de los usuarios o para implementar pilas localizadas para la audiencia de esa región). Como estrategia de DR, si la carga de trabajo no es compatible en una de las Regiones de AWS en la que se implemente, esa región se evacúa y las regiones restantes se utilizar para mantener la disponibilidad. La estrategia activa-activa multisitio es la más compleja de las estrategias de DR a nivel operativo y solo se debe seleccionar cuando los requisitos empresariales lo exijan.

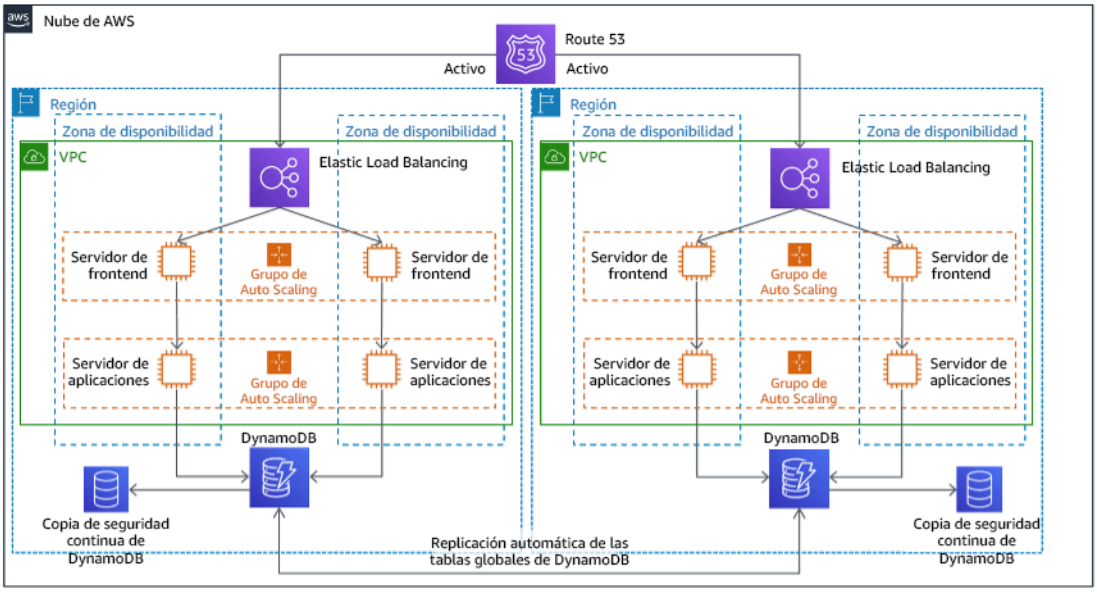



Figura 22: Arquitectura activa-activa multisitio

Para obtener más información sobre esta estrategia, consulte Disaster Recovery (DR) Architecture on AWS, Part IV: Multi-site Active/Active.

AWS Elastic Disaster Recovery

Si está considerando la posibilidad de adoptar la estrategia de luz piloto o espera semiactiva en caso de recuperación de desastres, AWS Elastic Disaster Recovery podría proporcionar un enfoque alternativo con mejores beneficios. La Recuperación de desastres elástica puede ofrecer un objetivo de RPO y RTO similar al de los sistemas de espera semiactiva, pero mantiene el enfoque de bajo costo de luz piloto. La Recuperación de desastres elástica replica los datos de la región principal a la región de recuperación y utiliza una protección de datos continua para lograr un RPO medido en segundos y un RTO que se puede medir en minutos. En la región de recuperación se implementan solo los recursos necesarios para replicar los datos, lo que mantiene los costos bajos, de forma similar a la estrategia de luz piloto. Al utilizar Recuperación de desastres elástica, el servicio coordina y organiza la recuperación de los recursos de computación cuando se inicia como parte de una conmutación por error o un simulacro.

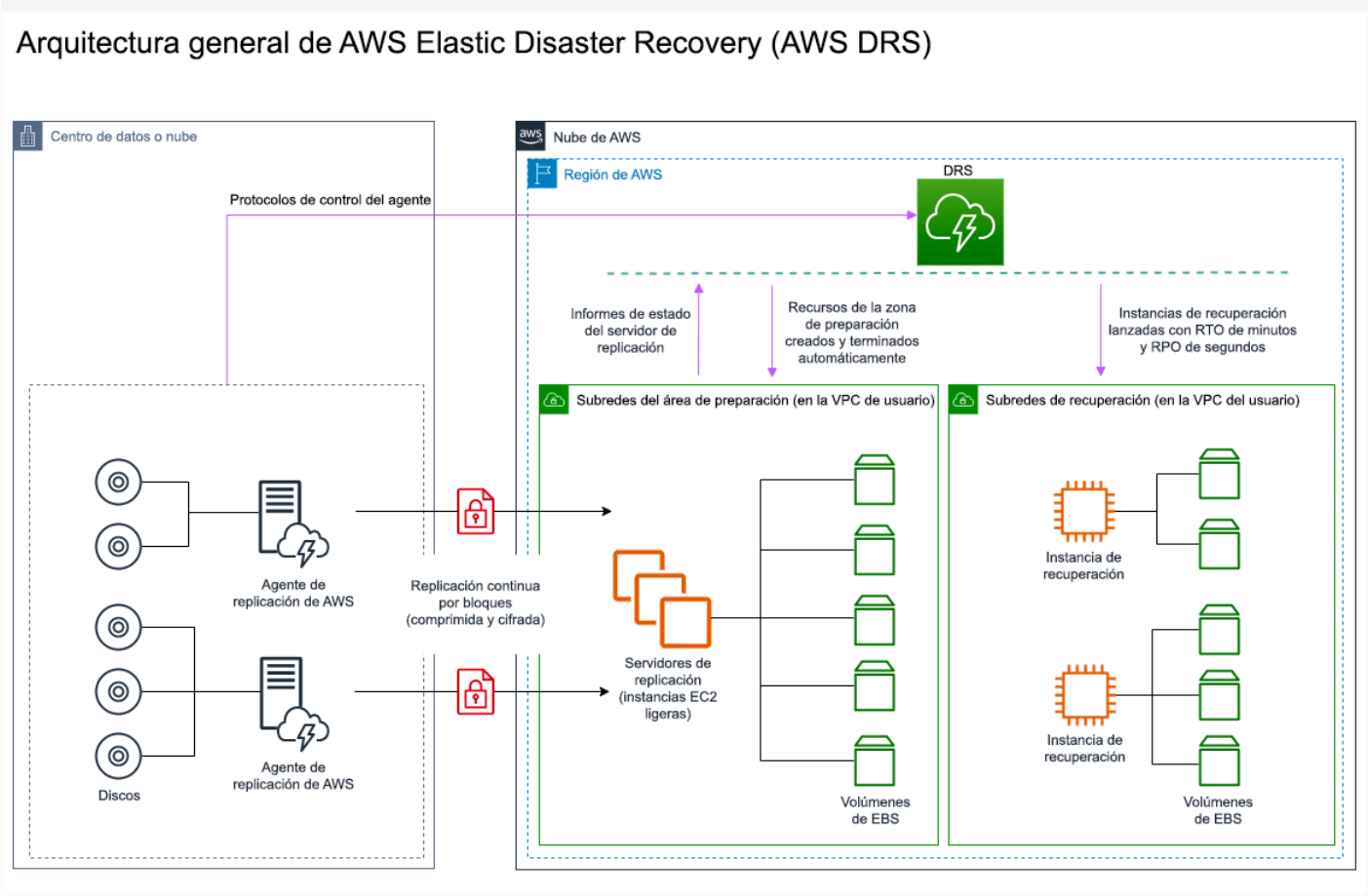



Figura 23: Arquitectura de AWS Elastic Disaster Recovery

Prácticas adicionales para la protección de los datos

Con todas las estrategias, también debe mitigar los posibles desastres de datos. La replicación de datos continua le brindará protección ante algunos tipos de desastres, pero no ante el daño o la destrucción de datos, a no ser que su estrategia incluya también control de versiones para una recuperación en un momento dado. También debe hacer una copia de seguridad de los datos replicados en el sitio de recuperación para crear copias de seguridad en un momento dado además de las réplicas.

Uso de varias zonas de disponibilidad (AZ) en una sola Región de AWS

Al usar varias AZ en una única región, la implementación de DR utiliza varios elementos de las estrategias anteriores. Primero, debe crear una arquitectura de alta disponibilidad con varias AZ, como se muestra en la figura 23. Esta arquitectura utiliza un enfoque activo-activo multisitio, ya que las instancias de Amazon EC2 y el equilibrador de carga elástico tienen recursos

implementados en varias zonas de disponibilidad y gestionan las solicitudes de forma activa. La arquitectura también muestra el modo de espera activa, donde, si se produce un error en la instancia principal de Amazon RDS (o se produce un error en la propia AZ), la instancia en espera pasa a ser principal.

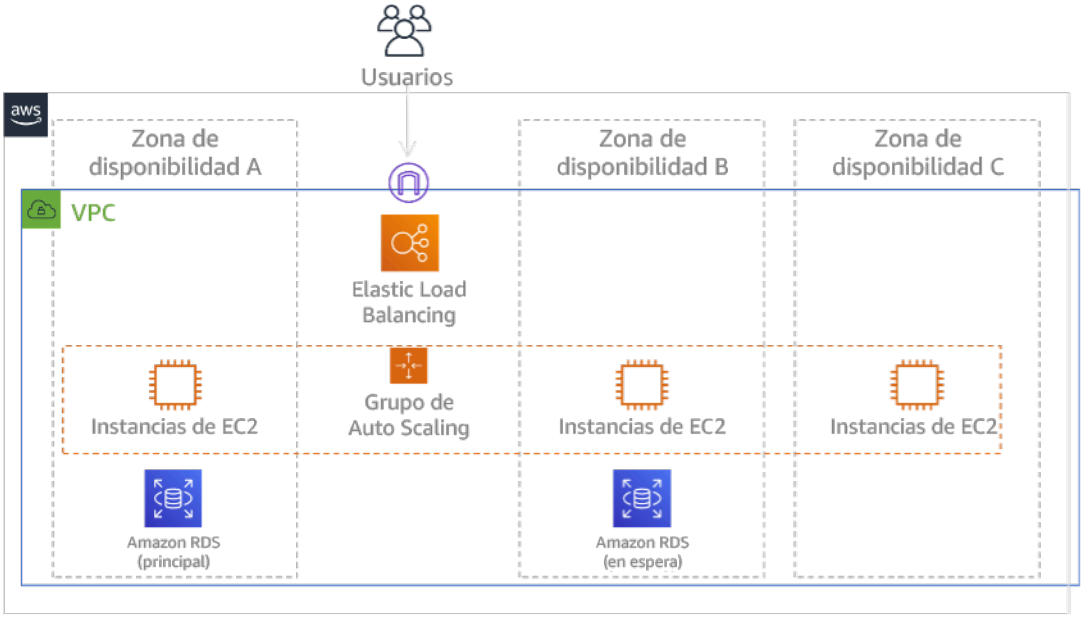



Figura 24: Arquitectura multi-AZ

Además de esta arquitectura de alta disponibilidad, debe agregar copias de seguridad con todos los datos necesarios para ejecutar la carga de trabajo. Esto es especialmente importante para los datos que están restringidos a una sola zona, como los volúmenes de Amazon EBS o los clústeres de Amazon Redshift. Si se produce un error en una AZ, tendrá que restaurar estos datos en otra AZ. Siempre que sea posible, deberá copiar las copias de seguridad de los datos en otra Región de AWS como capa de protección adicional.

Un enfoque alternativo menos común para la DR de una sola región de varias zonas de disponibilidad se ilustra en la entrada del blog Building highly resilient applications using Amazon Application Recovery Controller, Part 1: Single-Region stack. Aquí, la estrategia es mantener la máxima cantidad posible de aislamiento entre las AZ, tal y como funcionan las regiones. Al utilizar esta estrategia alternativa, puede elegir un enfoque activo-activo o activo-pasivo.

> **Note**
>
> Algunas cargas de trabajo tienen requisitos normativos de residencia de datos. Si esto se aplica a su carga de trabajo en una ubicación que actualmente tenga solo una Región

de AWS, el enfoque multirregión no se adaptará a sus necesidades empresariales. Las estrategias de multi-AZ ofrecen una buena protección contra la mayoría de los desastres.

3. Evalúe los recursos de la carga de trabajo y cuál será su configuración en la región de recuperación antes de la conmutación por error (durante el funcionamiento normal).

   Para la infraestructura y los recursos de AWS, utilice la infraestructura como código, por ejemplo, AWS CloudFormation o herramientas de terceros, como Hashicorp Terraform. Para efectuar la implementación en varias cuentas y regiones en una sola operación, puede utilizar AWS CloudFormation StackSets. En el caso de las estrategias activa-activa multisitio o espera activa, la infraestructura implementada en la región de recuperación tiene los mismos recursos que la región principal. En las estrategias de luz piloto y espera semiactiva, la infraestructura implementada requerirá acciones adicionales para prepararse para la producción. Con los parámetros y la lógica condicional de CloudFormation, puede controlar si una pila implementada está activa o en espera con una sola plantilla. Al utilizar la Recuperación de desastres elástica, el servicio replicará y orquestará la restauración de las configuraciones de las aplicaciones y los recursos de computación.

   Todas las estrategias de DR requieren que se haga una copia de seguridad de los orígenes de datos en la Región de AWS y, a continuación, que esas copias de seguridad se copien en la región de recuperación. AWS Backup proporciona una vista centralizada en la que se pueden configurar, programar y supervisar las copias de seguridad de estos recursos. Para los enfoques de luz piloto, espera semiactiva y activo-activo multisitio, también debe replicar los datos de la región principal en los recursos de datos de la región de recuperación, como las instancias de bases de datos de Amazon Relational Database Service (Amazon RDS) o las tablas de Amazon DynamoDB. De esta forma, estos recursos de datos estarán activos y preparados para responder a solicitudes en la región de recuperación.

   Para más información sobre el funcionamiento de los servicios de AWS en las regiones, consulte esta serie de blogs en Creating a Multi-Region Application with AWS Services.

4. Determine e implemente cómo preparará la región de recuperación para la conmutación por error cuando sea necesario (durante un desastre).

   En el caso de la opción activa-activa multisitio, la conmutación por error implica evacuar una región y recurrir a las regiones activas restantes. En general, esas regiones están listas para aceptar tráfico. En las estrategias de luz piloto y espera semiactiva, las acciones de recuperación tendrán que implementar los recursos faltantes, como las instancias de EC2 en la figura 20, además de otros recursos faltantes.

En todas las estrategias anteriores, es posible que tenga que promover instancias de solo lectura de bases de datos para que se conviertan en la instancia de lectura y escritura principal.

En copias de seguridad y restauración, la restauración de datos desde una copia de seguridad crea recursos para esos datos, como volúmenes de EBS, instancias de bases de datos de RDS y tablas de DynamoDB. También tiene que restaurar la infraestructura e implementar el código. Puede utilizar AWS Backup para restaurar datos en la región de recuperación. Consulte REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes para obtener más detalles. Volver a crear la infraestructura incluye la creación de recursos, como instancias de EC2, además de Amazon Virtual Private Cloud (Amazon VPC), las subredes y los grupos de seguridad necesarios. Puede automatizar gran parte del proceso de restauración. Consulte esta entrada de blog para obtener más información.

5. Determine e implemente cómo redirigirá el tráfico a la conmutación por error cuando sea necesario (durante un desastre).

   Esta operación de conmutación por error se puede iniciar automática o manualmente. La conmutación por error iniciada automáticamente en función de comprobaciones de estado o alarmas se debe utilizar con cuidado, ya que una conmutación por error innecesaria (falsa alarma) supone ciertos inconvenientes, como la falta de disponibilidad y la pérdida de datos. Por tanto, la conmutación por error iniciada manualmente es la que se suele utilizar. En este caso, debe seguir automatizando los pasos de la conmutación por error, de modo que la iniciación manual sea como pulsar un botón.

   Hay varias opciones de administración del tráfico que tener en cuenta al utilizar servicios de AWS. Una opción es utilizar Amazon Route 53. Al utilizar Amazon Route 53, puede asociar varios puntos de conexión de IP en una o varias Regiones de AWS a un nombre de dominio de Route 53. Para implementar la conmutación por error iniciada manualmente, puede utilizar el Controlador de recuperación de aplicaciones de Amazon, que proporciona una API de plano de datos de alta disponibilidad para redirigir el tráfico a la región de recuperación. Al implementar la conmutación por error, utilice las operaciones del plano de datos y evite las del plano de control, como se describe en REL11-BP04 Confianza en el plano de datos y no en el plano de control durante la recuperación.

   Para más información sobre esta y otras opciones, consulte esta sección del documento técnico sobre recuperación de desastres.

6. Diseñe un plan sobre cómo se recuperará su carga de trabajo.

La conmutación por recuperación se produce cuando se devuelve la operación de una carga de trabajo a la región principal una vez disminuido un desastre. Por lo general, el aprovisionamiento de la infraestructura y el código en la región principal sigue los mismos pasos que se utilizaron inicialmente y se basa en la infraestructura como código y en las canalizaciones de implementación del código. El reto que plantea la conmutación por recuperación es restaurar los almacenes de datos y garantizar que sean coherentes con la región de recuperación en funcionamiento.

En el estado de conmutación por error, las bases de datos en la región de recuperación están activas y tienen los datos actualizados. El objetivo es volver a efectuar la sincronización desde la región de recuperación a la región principal, lo que garantiza que está actualizada.

Algunos servicios de AWS harán esto automáticamente. Si utiliza las tablas globales de Amazon DynamoDB, aunque la tabla de la región principal no esté disponible, cuando vuelva a estar en línea, DynamoDB reanudará la propagación de las escrituras pendientes. Si utiliza la Base de datos global de Amazon Aurora y la conmutación por error planificada administrada, se mantiene la topología de reproducción existente de la base de datos global de Aurora. Por tanto, la instancia de lectura y escritura anterior en la región principal se convertirá en una réplica y recibirá actualizaciones desde la región de recuperación.

En los casos en los que esto no se haga automáticamente, tendrá que restablecer la base de datos en la región principal como una réplica de la base de datos en la región de recuperación. En muchos casos, esto supondrá eliminar la antigua base de datos principal y crear nuevas réplicas.

Tras una conmutación por error, si puede seguir operando en su región de recuperación, considere la posibilidad de convertir esta región en la nueva región principal. Seguiría completando los pasos anteriores para hacer que la antigua región principal fuera una región de recuperación. Algunas organizaciones llevan a cabo una rotación programada y cambian sus regiones principal y de recuperación periódicamente (por ejemplo, cada tres meses).

Todos los pasos necesarios para la conmutación por error y conmutación por recuperación deben mantenerse en una guía de estrategias disponible para todos los miembros del equipo que se revise periódicamente.

Al utilizar la Recuperación de desastres elástica, el servicio ayudará a organizar y automatizar el proceso de conmutación por recuperación. Para obtener más información, consulte Performing a failback.

Nivel de esfuerzo para el plan de implementación: alto

Recursos

Prácticas recomendadas relacionadas:

- the section called “REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes”
- the section called “REL11-BP04 Confianza en el plano de datos y no en el plano de control durante la recuperación”
- the section called “REL13-BP01 Definición de objetivos de recuperación para el tiempo de inactividad y la pérdida de datos”

Documentos relacionados:

- AWS Architecture Blog: Disaster Recovery Series
- Recuperación de desastres de cargas de trabajo en AWS: recuperación en la nube (documento técnico de AWS)
- Opciones de recuperación de desastres en la nube
- Build a serverless multi-region, active-active backend solution in an hour
- Multi-region serverless backend — reloaded
- RDS: replicación de una réplica de lectura entre regiones
- Route 53: Configuring DNS Failover
- S3: replicación entre regiones
- Qué es AWS Backup?
- What is Amazon Application Recovery Controller?
- AWS Elastic Disaster Recovery
- HashiCorp Terraform: Get Started - AWS
- Socio de APN: socios que pueden ayudar con la recuperación de desastres
- AWS Marketplace: productos que pueden usarse para la recuperación ante desastres

Videos relacionados:

- Disaster Recovery of Workloads on AWS
- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2)
- Get Started with AWS Elastic Disaster Recovery | Amazon Web Services

REL13-BP03 Prueba de la implementación de recuperación de desastres para validarla

Compruebe periódicamente la conmutación por error de su sitio de recuperación para verificar que funcione adecuadamente y que se cumplan el RTO y el RPO.

Patrones comunes de uso no recomendados:

- No llevar a cabo nunca conmutaciones por error en producción.

Beneficios de establecer esta práctica recomendada: las pruebas periódicas del plan de recuperación de desastres verifican que el plan funcione cuando llegue el momento y que su equipo sepa cómo llevar a cabo la estrategia.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Un patrón que debe evitarse es el desarrollo de rutas de recuperación que se pongan en práctica pocas veces. Por ejemplo, puede tener un almacén de datos secundario que se utilice para consultas de solo lectura. Cuando escribe en un almacén de datos y se produce un error del almacén principal, es posible que quiera conmutar por error al almacén de datos secundario. Si no se prueba frecuentemente esta conmutación por error, es posible que sus suposiciones sobre las capacidades del almacén de datos secundario sean incorrectas. Es posible que la capacidad del almacén de datos secundario, que quizás fuera suficiente cuando se probó por última vez, ya no pueda tolerar la carga en esta situación. Nuestra experiencia ha demostrado que la única forma de recuperación de errores que funciona es aquella que se prueba constantemente. Por ello, es mejor tener un número reducido de rutas de recuperación. Puede establecer patrones de recuperación y probarlos con frecuencia. Si tiene una ruta de recuperación compleja o crítica, todavía debe llevar a efecto ese error en producción periódicamente para asegurarse de que la ruta funcione. En el ejemplo que acabamos de comentar, se debe conmutar por error al modo de espera con regularidad, sin importar si es necesario.

Pasos para la implementación

1. Diseñe sus cargas de trabajo para que se puedan recuperar. Pruebe regularmente sus rutas de recuperación. La computación orientada a la recuperación identifica las características de los sistemas que mejoran la recuperación: aislamiento y redundancia, capacidad en todo el sistema para revertir los cambios, capacidad para supervisar y determinar el estado, capacidad para proporcionar diagnósticos, recuperación automatizada, diseño modular y capacidad para reiniciar. Ponga en práctica la ruta de recuperación para verificar que pueda llevar a cabo la recuperación en el tiempo especificado para el estado especificado. Use sus manuales de procedimientos durante esta recuperación para documentar los problemas y encontrar soluciones para estos antes de la próxima prueba.
2. En el caso de las cargas de trabajo basadas en Amazon EC2, utilice AWS Elastic Disaster Recovery para implementar y lanzar instancias de simulacro para la estrategia de DR. AWS Elastic Disaster Recovery ofrece la posibilidad de ejecutar simulacros de forma eficiente, lo que ayuda a prepararse para un evento de conmutación por error. También puede lanzar con frecuencia sus instancias mediante Recuperación de desastres elástica para hacer pruebas y simulacros sin redirigir el tráfico.

Recursos

Documentos relacionados:

- Socio de APN: socios que pueden ayudar con la recuperación de desastres
- AWS Architecture Blog: Disaster Recovery Series
- AWS Marketplace: productos que pueden usarse para la recuperación de desastres
- AWS Elastic Disaster Recovery
- Recuperación de desastres de cargas de trabajo en AWS: recuperación en la nube (documento técnico de AWS)
- AWS Elastic Disaster Recovery Preparing for Failover
- The Berkeley/Stanford recovery-oriented computing project
- What is AWS Fault Injection Simulator?

Videos relacionados:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications
- AWS re:Invent 2019: Backup-and-restore and disaster-recovery solutions with AWS

REL13-BP04 Administración de la desviación de la configuración en el sitio o región de DR

Para llevar a cabo un procedimiento de recuperación ante desastres (DR) correctamente, su carga de trabajo debe poder reanudar las operaciones normales de manera oportuna sin pérdida relevante de funcionalidad o datos una vez que el entorno de DR se haya puesto en funcionamiento. Para lograr este objetivo, es esencial mantener una infraestructura, datos y configuraciones consistentes entre el entorno de DR y el entorno principal.

Resultado deseado: la configuración y los datos de su sitio de recuperación ante desastres son iguales a los del sitio principal, lo que facilita una recuperación rápida y completa cuando es necesario.

Patrones comunes de uso no recomendados:

- No se actualizan las ubicaciones de recuperación cuando se realizan cambios en las ubicaciones principales, lo que se traduce en configuraciones desactualizadas que podrían dificultar los esfuerzos de recuperación.
- No tiene en cuenta las posibles limitaciones, como las diferencias de los servicios entre las ubicaciones principales y de recuperación, que pueden provocar fallos inesperados durante la conmutación por error.
- Confía en los procesos manuales para actualizar y sincronizar el entorno de recuperación ante desastres, lo que aumenta el riesgo de errores humanos e incoherencias.
- No se detectan desviaciones en la configuración, lo que genera una falsa sensación de que el sitio de recuperación ante desastres está preparado antes de que se produzca un incidente.

Beneficios de establecer esta mejor práctica: la coherencia entre el entorno de recuperación ante desastres y el entorno principal mejora considerablemente las probabilidades de una recuperación satisfactoria tras un incidente y reduce el riesgo de que se produzca un error en el procedimiento de recuperación.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Un enfoque integral de la administración de la configuración y la preparación para la conmutación por error puede ayudarlo a comprobar que el sitio de recuperación ante desastres se actualiza constantemente y está preparado para asumir el control en caso de que se produzca un fallo en el sitio principal.

Para lograr la coherencia entre su entorno principal y el de recuperación ante desastres (DR), compruebe que sus canales de entrega distribuyen las aplicaciones tanto en el sitio principal como en el de recuperación ante desastres. Despliegue los cambios en los sitios de recuperación ante desastres después de un periodo de evaluación adecuado (también conocido como despliegues escalonados) para detectar problemas en el sitio principal y detener la implementación antes de que se propaguen. Implemente la supervisión para detectar desviaciones en la configuración y lleve un seguimiento de los cambios y el cumplimiento en todos sus entornos. Realice correcciones automatizadas en el sitio de recuperación ante desastres para mantener la máxima coherencia y que esté listo para funcionar en caso de que se produzca un incidente.

Pasos para la implementación

1. Valide que la región de recuperación ante desastres contenga los servicios y las características de AWS necesarios para ejecutar correctamente el plan de recuperación ante desastres.
2. Utilice infraestructura como código (IaC). Mantenga la precisión de sus plantillas de configuración de aplicaciones e infraestructura de producción y aplíquelas periódicamente a su entorno de recuperación ante desastres. AWS CloudFormationpuede detectar una desviación entre lo que especifican las plantillas de CloudFormation y lo que realmente se implementa.
3. Configure las canalizaciones de CI/CD para implementar aplicaciones y actualizaciones de infraestructura en todos los entornos, incluidos los sitios principales y de recuperación ante desastres. Las soluciones de CI/CD, como AWS CodePipeline, pueden automatizar el proceso de implementación, lo que reduce el riesgo de cambios en la configuración.
4. Distribuya las implementaciones entre el entorno principal y el de recuperación ante desastres. Este enfoque permite implementar y probar las actualizaciones inicialmente en el entorno principal, lo que aísla los problemas en el sitio principal antes de que se propaguen al sitio de DR. Así se evita que los defectos se transmitan simultáneamente a la planta de producción y al centro de recuperación ante desastres y mantiene la integridad del entorno de recuperación ante desastres.
5. Supervise continuamente las configuraciones de los recursos en los entornos principal y de DR. Soluciones como AWS Config pueden ayudar a garantizar el cumplimiento de la configuración y detectar desviaciones, lo que ayuda a mantener la coherencia de las configuraciones en todos los entornos.
6. Implemente mecanismos de alerta para rastrear y notificar cualquier cambio en la configuración o interrupción o retraso en la replicación de datos.
7. Automatice la corrección de las desviaciones de configuración detectadas.

8. Programe auditorías y comprobaciones de conformidad periódicas para verificar la alineación continua entre las configuraciones principal y de recuperación ante desastres. Las revisiones periódicas ayudan a mantener el cumplimiento de las normas definidas e identificar cualquier discrepancia que deba abordarse.
9. Compruebe si hay discrepancias en la capacidad aprovisionada, las cuotas de servicio, los límites de aceleración y las discrepancias de configuración y versión de AWS.

Recursos

Prácticas recomendadas relacionadas:

- REL01-BP01 Conocimiento de las cuotas y restricciones del servicio
- REL01-BP02 Administración de cuotas de servicio en cuentas y regiones
- REL01-BP04 Supervisión y administración de cuotas
- REL13-BP03 Prueba de la implementación de recuperación ante desastres para validarla

Documentos relacionados:

- Remediating Noncompliant AWS Resources by Reglas de AWS Config
- AWS Systems Manager Automation
- AWS CloudFormation: Detección de cambios de configuración no administrados en pilas y recursos
- AWS CloudFormation: Detección de desviaciones en una pila de CloudFormation completa
- AWS Systems Manager Automation
- Recuperación de desastres de cargas de trabajo en AWS: recuperación en la nube (documento técnico de AWS)
- How do I implement an Infrastructure Configuration Management solution on AWS?
- Remediating Noncompliant AWS Resources by Reglas de AWS Config

Videos relacionados:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2)

Ejemplos relacionados:

- CloudFormation Registro
- Monitor de cuotas para AWS
- Implemente la corrección automática de desviaciones para AWS CloudFormation mediante Amazon CloudWatch y AWS Lambda
- AWS Architecture Blog: Disaster Recovery Series
- AWS Marketplace: productos que pueden usarse para la recuperación ante desastres
- Automatización de implementaciones seguras y sin intervención

REL13-BP05 Automatización de la recuperación

Implemente mecanismos de recuperación comprobados y automatizados que sean fiables, observables y reproducibles para reducir el riesgo y el impacto empresarial de los fallos.

Resultado deseado: ha implementado un flujo de trabajo de automatización bien documentado, estandarizado y probado exhaustivamente para los procesos de recuperación. La automatización de la recuperación corrige automáticamente los problemas menores que suponen un bajo riesgo de pérdida de datos o de falta de disponibilidad. Puede invocar rápidamente a los procesos de recuperación en caso de incidentes graves, observar el comportamiento de corrección mientras están en funcionamiento y finalizar los procesos si observa situaciones peligrosas o fallos.

Patrones comunes de uso no recomendados:

- Como parte de su plan de recuperación, depende de los componentes o mecanismos que se encuentran en un estado defectuoso o degradado.
- Los procesos de recuperación requieren una intervención manual, como el acceso a la consola (también conocido como operaciones de clic).
- Los procedimientos de recuperación se inician automáticamente en situaciones que presentan un alto riesgo de pérdida de datos o de falta de disponibilidad.
- No incluye un mecanismo para interrumpir un procedimiento de recuperación (similar a un sistema Andon o a un botón rojo de parada de emergencia) que no funciona o que plantea riesgos adicionales.

Beneficios de establecer esta práctica recomendada:

- Mayor fiabilidad, previsibilidad y consistencia de las operaciones de recuperación.

- Capacidad para cumplir objetivos de recuperación más estrictos, como el objetivo de tiempo de recuperación (RTO) y el objetivo de punto de recuperación (RPO).
- Menor probabilidad de fallos en la recuperación durante un incidente.
- Reducción del riesgo de fallos asociados a los procesos de recuperación manual que son propensos a errores humanos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para implementar la recuperación automatizada, necesita un enfoque integral que utilice los servicios de AWS y las prácticas recomendadas. Para empezar, identifique los componentes críticos y los posibles puntos de fallo de su carga de trabajo. Desarrolle procesos automatizados que puedan recuperar sus cargas de trabajo y datos en caso de fallos sin intervención humana.

Desarrolle la automatización de la recuperación mediante los principios de la infraestructura como código (IaC). Esto hace que su entorno de recuperación sea coherente con el entorno de origen y permita controlar las versiones de sus procesos de recuperación. Para organizar flujos de trabajo de recuperación complejos, puede usar soluciones como AWSSystems Manager Automation o AWS Step Functions.

La automatización de los procesos de recuperación ofrece beneficios importantes y puede ayudar a alcanzar el objetivo de tiempo de recuperación (RTO) y el objetivo de punto de recuperación (RPO) con mayor facilidad. Sin embargo, pueden encontrarse con situaciones inesperadas que tal vez provoquen fallos o creen nuevos riesgos por su parte, como un mayor tiempo de inactividad y la pérdida de datos. Para mitigar este riesgo, ofrezca la posibilidad de detener rápidamente una automatización de la recuperación en curso. Una vez detenida, puede investigar y tomar medidas correctivas.

En el caso de las cargas de trabajo compatibles, puede probar soluciones como la recuperación elástica ante desastres de AWS (AWS DRS) para proporcionar una conmutación por error automatizada. AWS DRS replica continuamente las máquinas (incluido el sistema operativo, la configuración del estado del sistema, las bases de datos, las aplicaciones y los archivos) en un área provisional de su cuenta de Cuenta de AWS de destino y en la región preferida. Si se produce un incidente, AWS DRS automatiza la conversión de los servidores replicados en cargas de trabajo totalmente aprovisionadas en la región de recuperación en AWS.

El mantenimiento y la mejora de la recuperación automática son un proceso continuo. Pruebe y perfeccione continuamente sus procedimientos de recuperación en función de las lecciones

aprendidas y manténgase al tanto de los nuevos servicios y características de AWS que pueden mejorar sus capacidades de recuperación.

Pasos para la implementación

1. Planifique una recuperación automatizada
    a. Realice una revisión exhaustiva de la arquitectura, los componentes y las dependencias de su carga de trabajo para identificar y planificar los mecanismos de recuperación automatizados. Clasifique las dependencias de su carga de trabajo en dependencias estrictas y flexibles. Las dependencias estrictas son imprescindibles y sin ellas la carga de trabajo no puede funcionar; además, no se puede ofrecer ninguna alternativa. Las dependencias flexibles son aquellas que suele utilizar la carga de trabajo, pero que pueden sustituirse por sistemas o procesos alternativos de manera temporal o que pueden gestionarse mediante una degradación estable.
    b. Establezca procesos para identificar y recuperar los datos dañados o ausentes.
    c. Defina los pasos para confirmar un estado estable recuperado una vez finalizadas las acciones de recuperación.
    d. Considere cualquier acción necesaria para que el sistema recuperado esté listo para funcionar plenamente, como precalentar y llenar las cachés.
    e. Considere los problemas que podrían surgir durante el proceso de recuperación y cómo detectarlos y solucionarlos.
    f. Plantéese situaciones en las que no se pueda acceder al sitio principal y a su plano de control. Compruebe que las acciones de recuperación se puedan realizar de forma independiente sin depender del sitio principal. Puede usar soluciones como el controlador de recuperación de aplicaciones de Amazon (ARC) para redirigir el tráfico sin necesidad de mutar manualmente los registros de DNS.
2. Desarrolle un proceso de recuperación automatizado
    a. Implemente mecanismos automatizados de detección de fallos y conmutación por error para una recuperación sin intervención manual. Cree paneles de control, como Amazon CloudWatch, para informar sobre el progreso y el estado de los procedimientos de recuperación automatizados. Incluya procedimientos para validar una recuperación correcta. Proporcione un mecanismo para interrumpir una recuperación en curso.
    b. Cree guías de estrategia como un proceso alternativo para los errores que no permitan una recuperación automática y tenga en cuenta su plan de recuperación ante desastres.
    c. Pruebe los procesos de recuperación tal y como se describe en REL13-BP03.
3. Prepárese para la recuperación

a. Evalúe el estado de su sitio de recuperación e implemente los componentes críticos en él con antelación. Para obtener más información, consulte REL13-BP04.
b. Defina funciones, responsabilidades y procesos de toma de decisiones claros para las operaciones de recuperación, con la participación de las partes interesadas y los equipos pertinentes de toda la organización.
c. Defina las condiciones para iniciar los procesos de recuperación.
d. Cree un plan para revertir el proceso de recuperación y vuelva a su sitio principal si es necesario o después de considerarlo seguro.

Recursos

Prácticas recomendadas relacionadas:

- REL07-BP01 Uso de la automatización al obtener o escalar recursos
- REL11-BP01 Supervisión de todos los componentes de la carga de trabajo para detectar errores
- REL13-BP02 Uso de estrategias de recuperación definidas para cumplir los objetivos de recuperación
- REL13-BP03 Prueba de la implementación de recuperación ante desastres para validarla
- REL13-BP04 Administración de la desviación de la configuración en el sitio o región de DR

Documentos relacionados:

- AWS Architecture Blog: Disaster Recovery Series
- Recuperación de desastres de cargas de trabajo en AWS: recuperación en la nube (documento técnico de AWS)
- Orchestrate Disaster Recovery Automation using Amazon Route 53 ARC and AWS Step Functions
- Build AWS Systems Manager Automation runbooks using AWS CDK
- AWS Marketplace: productos que pueden usarse para la recuperación ante desastres
- AWS Systems Manager Automation
- AWS Elastic Disaster Recovery
- Using Elastic Disaster Recovery for Failover and Failback
- AWS Elastic Disaster Recovery Resources
- Socio de APN: socios que pueden ayudar con la recuperación ante desastres

Videos relacionados:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2)
- AWS re:Invent 2022: AWS On Air ft. AWS Failback for AWS Elastic Disaster Recovery

# Eficiencia del rendimiento

El pilar de eficiencia del rendimiento incluye la capacidad para utilizar los recursos de la nube de forma eficaz a fin de que satisfagan los requisitos de rendimiento y para mantener dicha eficacia a medida que la demanda cambia y las tecnologías evolucionan. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de eficiencia del rendimiento.

Áreas de prácticas recomendadas

- Selección de la arquitectura
- Computación y hardware
- Administración de datos
- Redes y entrega de contenido
- Proceso y cultura

## Selección de la arquitectura

Preguntas

- PERF 1. ¿Cómo selecciona los recursos y la arquitectura en la nube adecuados para su carga de trabajo?

### PERF 1. ¿Cómo selecciona los recursos y la arquitectura en la nube adecuados para su carga de trabajo?

La solución óptima para una carga de trabajo concreta varía y las soluciones suelen combinar varios enfoques. Las cargas de trabajo de Well-Architected utilizan varias soluciones y admiten diferentes características para mejorar el rendimiento.

Prácticas recomendadas

- PERF01-BP01 Descubrimiento y comprensión de los servicios y las características disponibles en la nube
- PERF01-BP02 Uso de las recomendaciones del proveedor de servicios en la nube o de un socio adecuado para conocer los modelos de arquitectura y las prácticas recomendadas
- PERF01-BP03 Contemplación de los costos en las decisiones sobre arquitectura
- PERF01-BP04 Evaluación del efecto de las decisiones en los clientes y en la eficiencia de la arquitectura
- PERF01-BP05 Uso de políticas y arquitecturas de referencia
- PERF01-BP06 Uso de pruebas comparativas para tomar decisiones arquitectónicas
- PERF01-BP07 Uso de un enfoque basado en los datos en sus decisiones arquitectónicas

PERF01-BP01 Descubrimiento y comprensión de los servicios y las características disponibles en la nube

Investigue continuamente los servicios y configuraciones disponibles que pueden ayudarle a tomar mejores decisiones arquitectónicas y a mejorar la eficiencia del rendimiento de la arquitectura de su carga de trabajo.

Patrones comunes de uso no recomendados:

- Utiliza la nube como un centro de datos coubicado.
- Después de migrar a la nube, no moderniza la aplicación.
- Utiliza un único tipo de almacenamiento para todo lo que necesita conservar.
- Utiliza los tipos de instancia que más se ajustan a sus estándares actuales, pero son más grandes cuando es necesario.
- Implementa y administra tecnologías que están disponibles como servicios administrados.

Beneficios de establecer esta práctica recomendada: al explorar nuevos servicios y configuraciones, es posible que pueda mejorar considerablemente el rendimiento, reducir los costos y optimizar el esfuerzo necesario para mantener la carga de trabajo. También podrá reducir el tiempo de amortización de los productos habilitados para la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

AWS lanza nuevos servicios y características de forma continua que pueden mejorar el rendimiento y reducir el costo de las cargas de trabajo en la nube. Para mantener un rendimiento eficaz en la nube, es crucial estar al tanto de estos nuevos servicios y características. Modernizar la arquitectura de la carga de trabajo también le ayudará a acelerar la productividad, a impulsar la innovación y a descubrir más oportunidades de crecimiento.

Pasos para la implementación

- Haga un inventario del software y la arquitectura de su carga de trabajo para los servicios relacionados. Decida la categoría de productos sobre la que desea obtener más información.
- Explore las ofertas de AWS para identificar y conocer los servicios y las opciones de configuración pertinentes que pueden ayudarlo a mejorar el rendimiento y a reducir los costos y la complejidad operativa.
  - Amazon Web Services Cloud
  - AWS Academy
  - Novedades de AWS
  - Blog de AWS
  - AWS Skill Builder de
  - Eventos y seminarios web de AWS
  - Formación de AWS and Certifications
  - Canal de YouTube de AWS
  - AWS Workshops
  - AWS Communities
- Use Amazon Q para obtener información y consejos pertinentes sobre los servicios.
- Utilice entornos de pruebas (que no sean de producción) para aprender y experimentar con los nuevos servicios sin incurrir en costos extraordinarios.
- Obtenga información continua sobre los nuevos servicios y características de la nube.

Recursos

Documentos relacionados:

- Overview of Amazon Web Services

- Características de Amazon EC2
- Aprenda paso a paso con un plan de aprendizaje para socios de AWS
- Capacitación y certificación de AWS
- My learning path to become an AWS solutions architect
- AWS Centro de arquitectura de
- AWS Partner Network
- AWS Biblioteca de soluciones de
- Centro de conocimiento de AWS
- Cree aplicaciones modernas en AWS

Videos relacionados:

- AWS re:Invent 2023 - What's new with Amazon EC2
- AWS re:Invent 2022 - Reduce your operational and infrastructure costs with Amazon ECS
- AWS re:Invent 2023 - Build with the efficiency, agility & innovation of the cloud with AWS
- AWS re:Invent 2022 - Deploy ML models for inference at high performance and low cost
- This is my Architecture

Ejemplos relacionados:

- AWS Ejemplos del
- AWS Ejemplos del SDK

PERF01-BP02 Uso de las recomendaciones del proveedor de servicios en la nube o de un socio adecuado para conocer los modelos de arquitectura y las prácticas recomendadas

Utilice los recursos corporativos de la nube, como la documentación, los arquitectos de soluciones, los servicios profesionales o los socios adecuados, para que le sirvan de guía en sus decisiones arquitectónicas. Estos recursos le ayudarán a revisar y mejorar su arquitectura para obtener un rendimiento óptimo.

Patrones comunes de uso no recomendados:

- Utiliza AWS como un proveedor de servicios en la nube al uso.
- Utiliza los servicios de AWS de una manera para la que no se diseñaron.

- Sigue todas las directrices sin tener en cuenta su contexto empresarial.

Beneficios de establecer esta práctica recomendada: seguir las directrices de un proveedor de servicios en la nube o de un socio adecuado puede ayudarle a tomar las decisiones sobre arquitectura correctas para su carga de trabajo y a ganar confianza en sus decisiones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

AWS ofrece un gran número de directrices, documentación y recursos que pueden ayudarle a crear y administrar cargas de trabajo en la nube de forma eficiente. La documentación de AWS contiene ejemplos de código, tutoriales y explicaciones detalladas de los servicios. Además de la documentación, AWS ofrece programas de formación y certificación, arquitectos de soluciones y servicios profesionales que pueden ayudar a los clientes a explorar diferentes aspectos de los servicios en la nube y a implementar una arquitectura en la nube eficiente en AWS.

Aproveche estos recursos para obtener valiosos conocimientos y prácticas recomendadas, ahorrar tiempo y lograr mejores resultados en la Nube de AWS.

Pasos para la implementación

- Revise la documentación y las directrices de AWS y siga las prácticas recomendadas. Estos recursos pueden ayudarle a elegir y configurar los servicios de manera eficaz y a lograr un mejor rendimiento.
  - Documentación de AWS (como guías de usuario y documentos técnicos)
  - Blog de AWS
  - Formación de AWS and Certifications
  - Canal de YouTube de AWS
- Únase a los eventos de los socios de AWS (como los AWS Global Summits, AWS re:Invent, grupos de usuarios y talleres) para aprender de la mano de expertos de AWS las prácticas recomendadas acerca de cómo usar los servicios de AWS.
  - Aprenda paso a paso con un plan de aprendizaje para socios de AWS
  - Eventos y seminarios web de AWS
  - AWS Workshops
  - AWS Communities

- Contacte con AWS cuando necesite más ayuda o información sobre un producto. AWS Los Solutions Architects y AWS Professional Services proporcionan orientación para la implementación de soluciones. AWS Los socios ponen a su disposición el conocimiento experto de AWS para ayudarle a mejorar la agilidad y la innovación para su empresa.
- Use Soporte si necesita asistencia técnica para usar un servicio de forma eficaz. Nuestros planes de asistencia están diseñados para ofrecerle la combinación perfecta de herramientas junto con el acceso a conocimientos especializados para que pueda tener éxito con AWS mientras optimiza el rendimiento, administra los riesgos y mantiene los costos bajo control.

Recursos

Documentos relacionados:

- AWS Centro de arquitectura de
- AWS Partner Network
- Biblioteca de soluciones de AWS
- Centro de conocimiento de AWS
- AWS Enterprise Support

Videos relacionados:

- This is my Architecture
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2023 - Implementing distributed design patterns on AWS
- AWS re:Invent 2023 - Application architecture as code

Ejemplos relacionados:

- AWS Ejemplos del
- Ejemplos del AWS SDK
- AWS Analytics Reference Architecture

PERF01-BP03 Contemplación de los costos en las decisiones sobre arquitectura

Tenga en cuenta los costos en sus decisiones arquitectónicas para mejorar el uso de los recursos y la eficiencia del rendimiento de su carga de trabajo en la nube. Si conoce las implicaciones financieras de su carga de trabajo en la nube, es más probable que aproveche los recursos de forma eficiente y reduzca las prácticas innecesarias.

Patrones comunes de uso no recomendados:

- Solo utiliza una familia de instancias.
- No contempla la posibilidad de utilizar soluciones con licencia en lugar de soluciones de código abierto.
- No tiene políticas definidas sobre el ciclo de vida del almacenamiento.
- No revisa los nuevos servicios y características de la Nube de AWS.
- Solo utiliza el almacenamiento de bloques.

Beneficios de establecer esta práctica recomendada: si tiene en cuenta los costos a la hora de tomar decisiones, tendrá la oportunidad de utilizar recursos más eficientes y explorar otras inversiones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si optimiza las cargas de trabajo con arreglo a los costos, puede mejorar el uso de los recursos y evitar pérdidas en una carga de trabajo en la nube. Por lo general, al contemplar los costos en las decisiones de arquitectura, los componentes de la carga de trabajo se dimensionan correctamente y se favorece la elasticidad, lo que se traduce en una mejora de la eficiencia del rendimiento de las cargas de trabajo en la nube.

Pasos para la implementación

- Establezca objetivos de costos, como los límites presupuestarios de la carga de trabajo en la nube.
- Identifique los componentes clave (como las instancias y el almacenamiento) que influyen en los costos de su carga de trabajo. Puede usar Calculadora de precios de AWS y AWS Cost Explorer para identificar los principales factores que influyen en los costos de su carga de trabajo.
- Consulte los modelos de precios en la nube, como instancias bajo demanda, instancias reservadas, Savings Plans e instancias de spot.

- Utilice las prácticas recomendadas de optimización de costos de Well-Architected para optimizar estos componentes clave en términos de costos.
- Supervise y analice los costos de forma continua para identificar oportunidades que le permitan optimizar los gastos de su carga de trabajo.
  - Use AWS Budgets para recibir alertas sobre costos inaceptables.
  - Use AWS Compute Optimizer o AWS Trusted Advisor para obtener recomendaciones sobre la optimización de costos.
  - Use la Detección de anomalías en los costos de AWS para detectar automáticamente las anomalías en los costos y analizar la causa raíz.

Recursos

Documentos relacionados:

- What is AWS Billing and Cost Management?
- Optimización de costos con AWS
- Choosing an AWS cost management strategy
- A Beginner's Guide to AWS Cost Management
- A Detailed Overview of the Cost Intelligence Dashboard
- Centro de arquitectura de AWS
- Biblioteca de soluciones de AWS
- Centro de conocimiento de AWS

Videos relacionados:

- This is my Architecture
- AWS re:Invent 2023 - What's new with AWS cost optimization
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2023 - AWS storage cost-optimization best practices
- AWS re:Invent 2023 - Optimize costs in your multi-account environments

Ejemplos relacionados:

- Código de demostración de AWS Compute Optimizer

- Cost Optimization Workshop
- Cloud Financial Management Technical Implementation Playbooks
- Startup optimization: Tuning application performance for maximum efficiency
- Serverless Optimization Workshop (Performance and Cost)
- Scaling cost effective architectures

PERF01-BP04 Evaluación del efecto de las decisiones en los clientes y en la eficiencia de la arquitectura

Cuando evalúe las mejoras relacionadas con el rendimiento, debe determinar qué decisiones afectarán a sus clientes y a la eficiencia de la carga de trabajo. Por ejemplo, si el uso de un almacén de datos clave-valor mejora el rendimiento del sistema, es importante analizar cómo la naturaleza eventualmente consistente de este cambio afectaría a los clientes.

Patrones comunes de uso no recomendados:

- Da por hecho que habría que implementar todas las ventajas relacionadas con el rendimiento, aunque esta implementación tenga repercusiones.
- Solo evalúa los cambios en las cargas de trabajo cuando un problema de rendimiento ha alcanzado un punto crítico.

Beneficios de establecer esta práctica recomendada: al evaluar las mejoras potenciales relacionadas con el rendimiento, debe decidir si las compensaciones que exigen los cambios son aceptables de acuerdo con los requisitos de la carga de trabajo. En algunos casos, es posible que tenga que implementar controles adicionales para contrarrestar estas repercusiones.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Identifique las áreas críticas de la arquitectura en términos de cómo afectan al rendimiento y a los clientes. Determine cómo puede hacer mejoras, qué repercusiones tienen esas mejoras y cómo afectan al sistema y a la experiencia del usuario. Por ejemplo, la implementación de datos en caché puede mejorar drásticamente el rendimiento, pero requiere una estrategia clara sobre cómo y cuándo actualizar o invalidar los datos en caché para evitar un comportamiento incorrecto del sistema.

Pasos para la implementación

- Comprenda los requisitos de la carga de trabajo y los SLA.
- Defina claramente los factores de la evaluación. Estos factores pueden estar relacionados con los costos, la fiabilidad, la seguridad y el rendimiento de su carga de trabajo.
- Seleccione una arquitectura y unos servicios que puedan satisfacer sus necesidades.
- Lleve a cabo experimentos y pruebas de conceptos (POC) para analizar las repercusiones y el impacto que pueden tener en los clientes y en la eficiencia de la arquitectura. Por lo general, las cargas de trabajo seguras, de alto rendimiento y de alta disponibilidad consumen más recursos de la nube, aunque proporcionan una mejor experiencia al cliente. Comprenda las compensaciones de la complejidad, el rendimiento y el costo de su carga de trabajo. Por lo general, priorizar dos de los factores se produce a expensas del tercero.

Recursos

Documentos relacionados:

- Amazon Builders' Library
- KPI de Quick
- Amazon CloudWatch RUM
- Documentación de X-Ray
- Understand resiliency patterns and trade-offs to architect efficiently in the cloud

Videos relacionados:

- Optimize applications through Amazon CloudWatch RUM
- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems

Ejemplos relacionados:

- Measure page load time with Amazon CloudWatch Synthetics
- Cliente web de Amazon CloudWatch RUM

PERF01-BP05 Uso de políticas y arquitecturas de referencia

Cuando elija los servicios y las configuraciones, utilice políticas internas y arquitecturas de referencia existentes para ser más eficiente al diseñar e implementar su carga de trabajo.

Patrones comunes de uso no recomendados:

- Permite usar una gran variedad de tecnologías, lo que puede incidir en los gastos generales de administración de la empresa.

Beneficios de establecer esta práctica recomendada: establecer una política para la elección de la arquitectura, la tecnología y el proveedor permite tomar decisiones de forma rápida.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Contar con políticas internas para seleccionar los recursos y la arquitectura proporciona estándares y pautas que pueden seguirse al tomar decisiones sobre arquitectura. Estas directrices agilizan el proceso de toma de decisiones a la hora de elegir el servicio de nube correcto y pueden ayudar a mejorar la eficiencia del rendimiento. Implemente la carga de trabajo a través de políticas o arquitecturas de referencia. Integre los servicios en su implementación en la nube y, a continuación, utilice las pruebas de rendimiento para asegurarse de que puede seguir cumpliendo los requisitos establecidos.

Pasos para la implementación

- Conozca al detalle los requisitos de su carga de trabajo en la nube.
- Consulte políticas internas y externas para identificar las más relevantes.
- Utilice las arquitecturas de referencia adecuadas que le ofrece AWS o las prácticas recomendadas por el sector.
- Cree un conjunto coherente de políticas, estándares, arquitecturas de referencia y pautas prescriptivas para situaciones comunes. De este modo, sus equipos podrán avanzar más rápido. Adapte los activos a su sector, si procede.
- Coteje estas políticas y arquitecturas de referencia con su carga de trabajo en entornos de pruebas.
- Manténgase al tanto de los estándares sectoriales y las actualizaciones de AWS para asegurarse de que las políticas y las arquitecturas de referencia le ayudan a optimizar su carga de trabajo en la nube.

Recursos

Documentos relacionados:

- Centro de arquitectura de AWS
- AWS Partner Network
- Biblioteca de soluciones de AWS
- Centro de conocimiento de AWS
- AWS Architecture Blog

Videos relacionados:

- This is my Architecture
- AWS re:Invent 2022 - Accelerate value for your business with SAP & AWS reference architecture

Ejemplos relacionados:

- Ejemplos del AWS
- Ejemplos del AWS SDK

PERF01-BP06 Uso de pruebas comparativas para tomar decisiones arquitectónicas

Mida el rendimiento de una carga de trabajo existente para entender cómo rinde en la nube y fundamentar sus decisiones sobre arquitectura en esos datos.

Patrones comunes de uso no recomendados:

- Utiliza pruebas comparativas de uso común que no son indicativas de las características concretas de su carga de trabajo.
- La única referencia que tiene en cuenta son los comentarios y las percepciones de los clientes.

Beneficios de establecer esta práctica recomendada: el estudio comparativo de su implementación actual le permite medir las mejoras del rendimiento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Utilice la evaluación comparativa con pruebas sintéticas para evaluar el rendimiento de los componentes de su carga de trabajo. Las pruebas comparativas suelen ser más rápidas de configurar que las pruebas de carga y se utilizan para evaluar la tecnología de un componente concreto. Estas pruebas comparativas suelen usarse al comienzo de un nuevo proyecto, cuando aún no se tiene una solución completa para hacer una prueba de carga.

Puede crear sus propias pruebas comparativas personalizadas, o bien usar un estándar del sector, como TPC-DS, para comparar sus cargas de trabajo. Las pruebas comparativas sectoriales son útiles cuando se comparan entornos. Los puntos de referencia personalizados son útiles para encontrar tipos específicos de operaciones que espera llevar a cabo en su arquitectura.

Con las pruebas comparativas, es importante llevar a cabo los preparativos necesarios en el entorno de prueba para asegurarse de que los resultados obtenidos son válidos. Ejecute la misma comparativa muchas veces para asegurarse de que detecta cualquier variación que haya podido surgir con el tiempo.

Como las pruebas comparativas por lo general se ejecutan más rápido que las pruebas de carga, pueden usarse antes en la canalización de implementación y proporcionan información de una forma más rápida sobre las desviaciones del rendimiento. Al evaluar un cambio importante en un componente o servicio, puede resultar más rápido usar una prueba comparativa para determinar si el esfuerzo que conlleva el cambio es justificable. Es importante usar pruebas de carga junto con las pruebas comparativas, ya que las pruebas de carga le informan del rendimiento de la carga de trabajo en producción.

Pasos para la implementación

- Planificación y definición:
  - Defina los objetivos, la base de referencia, los escenarios de prueba, las métricas (como la utilización de la CPU, la latencia o el rendimiento) y los KPI para el punto de referencia.
  - Céntrese en los requisitos de los usuarios en lo que respecta a la experiencia de usuario y factores como el tiempo de respuesta y la accesibilidad.
  - Identifique una herramienta de pruebas comparativas que sea adecuada para su carga de trabajo. Puede usar los servicios de AWS (como Amazon CloudWatch) o una herramienta de terceros que sea compatible con su carga de trabajo.
- Configuración e instrumentación:
  - Configure el entorno y los recursos.

    - Implemente la supervisión y el registro para recopilar los resultados de las pruebas.
- Comparación y supervisión:
    - Haga las pruebas comparativas y supervise las métricas durante la prueba.
- Análisis y documentación:
    - Documente el proceso de evaluación comparativa y los resultados.
    - Analice los resultados para identificar los cuellos de botella, las tendencias y las áreas de mejora.
    - Utilice los resultados de las pruebas para tomar decisiones arquitectónicas y ajustar la carga de trabajo. Para ello, puede ser necesario cambiar los servicios o adoptar nuevas características.
- Optimizar y repetir:
    - Ajuste las configuraciones y asignaciones de los recursos en función de los puntos de referencia.
    - Vuelva a probar la carga de trabajo después del ajuste para validar las mejoras.
    - Documente la información obtenida y repita el proceso para identificar otras áreas de mejora.

Recursos

Documentos relacionados:

- AWS Centro de arquitectura de
- AWS Partner Network
- AWS Biblioteca de soluciones de
- Centro de conocimiento de AWS
- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Genomics workflows, Part 5: automated benchmarking
- Benchmark and optimize endpoint deployment in Amazon SageMaker AI JumpStart

Videos relacionados:

- AWS re:Invent 2023 - Benchmarking AWS Lambda cold starts
- Benchmarking stateful services in the cloud
- This is my Architecture

- Optimize applications through Amazon CloudWatch RUM
- Demo of Amazon CloudWatch Synthetics

Ejemplos relacionados:

- AWS Ejemplos del
- Ejemplos del AWS SDK
- Pruebas de carga distribuidas
- Measure page load time with Amazon CloudWatch Synthetics
- Cliente web de Amazon CloudWatch RUM

PERF01-BP07 Uso de un enfoque basado en los datos en sus decisiones arquitectónicas

Defina un enfoque claro basado en los datos para utilizarlo cuando tome decisiones sobre arquitectura y asegurarse de que se utilizan los servicios y las configuraciones en la nube correctos para satisfacer las necesidades específicas de su empresa.

Patrones comunes de uso no recomendados:

- Presupone que la arquitectura actual es estática y no debe actualizarse con el tiempo.
- Las decisiones arquitectónicas que toma se basan en conjeturas y suposiciones.
- Se introducen cambios en la arquitectura a lo largo del tiempo sin justificación.

Beneficios de establecer una práctica recomendada: al contar con un enfoque bien definido y aplicarlo a la hora de optar por las opciones arquitectónicas, se utilizan los datos para influir en el diseño de la carga de trabajo y tomar decisiones fundamentadas a lo largo del tiempo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para seleccionar los recursos y los servicios de su arquitectura, aproveche la experiencia y los conocimientos sobre la nube del personal interno o utilice recursos externos, como los casos de uso publicados o los documentos técnicos. Debe contar con un proceso bien definido que contribuya a probar y comparar los servicios que podrían utilizarse en su carga de trabajo.

La lista de tareas pendientes para las cargas de trabajo críticas no solo debe incluir casos de usuario que brinden una funcionalidad relevante para la empresa y los usuarios, sino también casos

técnicos que conformen un plan arquitectónico para la carga de trabajo. Este plan se nutre de nuevos avances en tecnología y nuevos servicios, que se incorporan con arreglo a los datos y de forma justificada. Esto garantiza que la arquitectura siempre está preparada para el futuro y no se queda anquilosada.

Pasos para la implementación

- Hable con las principales partes interesadas para definir los requisitos de la carga de trabajo, incluidas las consideraciones de rendimiento, disponibilidad y costos. Tenga en cuenta factores como la cantidad de usuarios y el modo de uso de la carga de trabajo.
- Cree un plan de arquitectura o una lista de tareas pendientes relacionadas con la tecnología que tengan la misma prioridad que las tareas pendientes relacionadas con la funcionalidad.
- Evalúe y valore los diferentes servicios en la nube (para obtener más información, consulte PERF01-BP01 Descubrimiento y comprensión de los servicios y las características disponibles en la nube).
- Analice diferentes patrones arquitectónicos, como los microservicios o la computación sin servidor, que se ajusten a sus requisitos de rendimiento (para obtener más información, consulte PERF01-BP02 Uso de las recomendaciones del proveedor de servicios en la nube o de un socio adecuado para conocer los modelos de arquitectura y las prácticas recomendadas).

- Consulte otros equipos, diagramas de arquitectura y recursos, como arquitectos de soluciones de AWS, Centro de arquitectura de AWS y AWS Partner Network, para poder elegir la arquitectura adecuada para su carga de trabajo.

- Defina métricas, como el rendimiento y el tiempo de respuesta, que puedan ser de ayuda a la hora de evaluar el rendimiento de su carga de trabajo.
- Pruebe y utilice las métricas definidas para validar el rendimiento de la arquitectura seleccionada.
- Mantenga un control continuo y haga los ajustes necesarios para garantizar el rendimiento óptimo de su arquitectura.
- Documente la arquitectura seleccionada y las decisiones adoptadas de forma que sirvan de referencia para futuras actualizaciones y formaciones.
- Revise y actualice continuamente el enfoque de selección de arquitectura con arreglo a los nuevos conocimientos, las nuevas tecnologías y las métricas que indiquen un cambio necesario o un problema en el enfoque actual.

Recursos

Documentos relacionados:

- Biblioteca de soluciones de AWS
- Centro de conocimiento de AWS
- Architectural Patterns to Build End-to-End Data Driven Applications on AWS

Videos relacionados:

- This is my Architecture
- AWS re:Invent 2021 - Data-driven enterprise: Going from vision to value
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results

Ejemplos relacionados:

- AWS Ejemplos del
- Ejemplos del AWS SDK

## Computación y hardware

Preguntas

- PERF 2. ¿Cómo selecciona y utiliza los recursos de computación en su carga de trabajo?

### PERF 2. ¿Cómo selecciona y utiliza los recursos de computación en su carga de trabajo?

La elección óptima de computación para una carga de trabajo concreta puede variar en función del diseño de la aplicación, los patrones de uso y los ajustes de configuración. Las arquitecturas pueden usar diferentes opciones de computación para varios componentes y admiten diferentes características para mejorar el rendimiento. No seleccionar la opción de computación correcta para una arquitectura puede disminuir la eficiencia del rendimiento.

Prácticas recomendadas

- PERF02-BP01 Selección de las mejores opciones computacionales para su carga de trabajo
- PERF02-BP02 Comprensión de las opciones de configuración y las características de computación disponibles
- PERF02-BP03 Recopilación de métricas relacionadas con la computación
- PERF02-BP04 Configuración y dimensionamiento correcto de los recursos de computación
- PERF02-BP05 Escalado de los recursos de computación de forma dinámica
- PERF02-BP06 Uso de aceleradores de computación optimizados basados en hardware

PERF02-BP01 Selección de las mejores opciones computacionales para su carga de trabajo

Si selecciona la opción computacional más adecuada para su carga de trabajo, podrá mejorar el rendimiento, reducir los costos de infraestructura innecesarios y aligerar los esfuerzos operativos necesarios para mantener esa carga de trabajo.

Patrones comunes de uso no recomendados:

- Utiliza la misma opción computacional que en el entorno en las instalaciones.
- No tiene información suficiente sobre las opciones de computación, las características y las soluciones de la nube, y cómo estas podrían mejorar el rendimiento de computación.
- Ha aprovisionado en exceso una opción de computación existente para cumplir los requisitos de escalado o rendimiento cuando una opción de computación alternativa se ajustaría con mayor precisión a las características de la carga de trabajo.

Beneficios de establecer una práctica recomendada: al identificar los requisitos de computación y evaluarlos con arreglo a las opciones disponibles, puede hacer que su carga de trabajo sea más eficiente en términos de recursos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para optimizar las cargas de trabajo en la nube y lograr un rendimiento eficiente, es importante seleccionar las opciones de computación más adecuadas para su caso de uso y los requisitos de rendimiento. AWS ofrece una variedad de opciones de computación que se adaptan a diferentes cargas de trabajo en la nube. Por ejemplo, puede usar Amazon EC2 para lanzar y administrar servidores virtuales, AWS Lambda para poner en marcha código sin tener que aprovisionar

o administrar servidores, Amazon ECS o Amazon EKS para poner en marcha y administrar contenedores, o AWS Batch para procesar grandes volúmenes de datos en paralelo. En función de sus necesidades de computación y escalado, debe elegir y configurar la solución computacional que sea óptima para su caso. También puede considerar la posibilidad de usar diferentes tipos de soluciones computacionales en una misma carga de trabajo, ya que cada una de ellas tiene sus propias ventajas e inconvenientes.

Los siguientes pasos le permitirán seleccionar las opciones computacionales adecuadas que se adaptan a las características de su carga de trabajo y a los requisitos de rendimiento.

Pasos para la implementación

- Comprenda cuáles son los requisitos computacionales de su carga de trabajo. Algunos de los principales requisitos son las necesidades de procesamiento, los patrones de tráfico, los patrones de acceso a los datos, las necesidades de escalado y los requisitos de latencia.
- Obtenga información sobre los diferentes servicios de computación de AWS para su carga de trabajo. Para obtener más información, consulte PERF01-BP01 Descubrimiento y comprensión de los servicios y las características disponibles en la nube. Estas son algunas de las principales opciones de computación de AWS, sus características y casos de uso comunes:

| Servicio de AWS | Características clave | Casos de uso comunes |
|---|---|---|
| Amazon Elastic Compute Cloud (Amazon EC2) | Cuenta con una opción dedicada para hardware, requisitos de licencia, una amplia selección de distintas familias de instancias, tipos de procesadores y acelerado res de computación. | Migraciones mediante lift-and-shift, aplicación monolític a, entornos híbridos, aplicacio nes empresariales |
| Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) | Implementación sencilla, entornos coherentes, escalable | Microservicios, entornos híbridos |
| AWS Lambda | Servicio de computación sin servidor que pone en marcha código como respuesta | Microservicios, aplicaciones basadas en eventos |

| Servicio de AWS | Características clave | Casos de uso comunes |
| --- | --- | --- |
| | a eventos y administra automáticamente los recursos de computación subyacentes. | |
| AWS Batch | Aprovisiona y escala de manera eficiente y dinámica Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) y recursos de computación de AWS Fargate, con la opción de usar instancias de spot o bajo demanda en función de los requisitos de su trabajo | HPC, entrenamiento de modelos de ML |
| Amazon Lightsail | Aplicación de Linux y Windows preconfigurada para poner en marcha cargas de trabajo pequeñas | Aplicaciones web simples, sitio web personalizado |

- Calcule el costo (por ejemplo, el costo por hora o la transferencia de datos) y los gastos generales de administración (como la aplicación de parches y el escalado) asociados a cada opción de computación.
- Lleve a cabo experimentos y pruebas comparativas en un entorno que no sea de producción para identificar qué opción de computación puede satisfacer mejor los requisitos de su carga de trabajo.
- Una vez que haya probado e identificado su nueva solución de computación, planifique la migración y valide sus métricas de rendimiento.
- Utilice las herramientas de supervisión de AWS, como Amazon CloudWatch, y los servicios de optimización, como AWS Compute Optimizer, para optimizar continuamente los recursos de computación en función de los patrones de uso reales.

## Recursos

Documentos relacionados:

- Computación en la nube con AWS
- Tipos de instancias de Amazon EC
- Contenedores de Amazon EKS: nodos de trabajo de Amazon EKS
- Contenedores de Amazon ECS: instancias de contenedor de Amazon ECS
- Funciones: configuración de funciones de Lambda
- Prescriptive Guidance for Containers
- Prescriptive Guidance for Serverless

Videos relacionados:

- AWS re:Invent 2023 - AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 - New Amazon Elastic Compute Cloud generative AI capabilities in AMS
- AWS re:Invent 2023 - What's new with Amazon Elastic Compute Cloud
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2021 - Powering next-gen Amazon Elastic Compute Cloud: Deep dive on the Nitro System
- AWS re:Invent 2019 - Optimize performance and cost for your AWS compute
- AWS re:Invent 2019 - Amazon Elastic Compute Cloud foundations
- AWS re:Invent 2022 - Deploy ML models for inference at high performance and low cost
- AWS re:Invent 2019 - Optimize performance and cost for your AWS compute
- Amazon EC2 foundations
- Implementación de modelos de ML para realizar inferencias con un alto rendimiento y un bajo costo

Ejemplos relacionados:

- Migrating the Web application to containers
- Run a Serverless Hello World
- Taller de Amazon EKS
- Amazon EC2 Workshop
- Efficient and Resilient Workloads with Amazon Elastic Compute Cloud Auto Scaling
- Migrating to AWS Graviton with Container Services

PERF02-BP02 Comprensión de las opciones de configuración y las características de computación disponibles

Conozca las opciones de configuración y las características disponibles para su servicio de computación, lo que le permitirá aprovisionar la cantidad de recursos adecuada y conseguir un rendimiento más eficiente.

Patrones comunes de uso no recomendados:

- No evalúan las opciones de computación ni las familias de instancias disponibles con arreglo a las características de la carga de trabajo.
- Aprovisiona un exceso de recursos de computación para satisfacer los picos de demanda.

Beneficios de establecer esta práctica recomendada: familiarícese con las configuraciones y las características computacionales de AWS para utilizar una solución computacional optimizada que se ajuste a las características y necesidades de su carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cada solución de computación tiene disponibles configuraciones y características únicas que admiten diferentes características y requisitos de la carga de trabajo. Descubra cómo estas opciones complementan su carga de trabajo y determine qué opciones de configuración son mejores para su caso. Algunas de estas opciones pueden ser, por ejemplo, la familia de instancias, el tamaño, las características (GPU, E/S, etc.), la capacidad de ampliación, los tiempos de espera, los tamaños de funciones, las instancias de contenedor y la simultaneidad. Si su carga de trabajo ha estado utilizando la misma opción de computación durante más de cuatro semanas y prevé que las características seguirán siendo las mismas en el futuro, puede utilizar AWS Compute Optimizer para comprobar si su opción computacional actual es apropiada para las cargas de trabajo en cuanto a CPU y memoria.

Pasos para la implementación

- Sepa cuáles son los requisitos de la carga de trabajo (como los requisitos de CPU, la memoria y la latencia).
- Consulte la documentación y las prácticas recomendadas de AWS para obtener información sobre las opciones de configuración recomendadas que pueden ayudar a mejorar el rendimiento

computacional. Estas son algunas de las principales opciones de configuración que debe tener en cuenta:

| Opción de configuración | Ejemplos |
| --- | --- |
| Tipo de instancia | • Las instancias optimizadas para la computación son ideales para las cargas de trabajo que requieren una relación entre vCPU y memoria más alta.<br>• Las instancias optimizadas para la memoria ofrecen grandes cantidades de memoria para admitir cargas de trabajo que hacen un uso intensivo de la memoria.<br>• Las instancias optimizadas para el almacenamiento están diseñadas para cargas de trabajo que requieren un alto acceso secuencial de lectura y escritura (IOPS) al almacenamiento local. |

| Opción de configuración | Ejemplos |
| --- | --- |
| Modelo de precios | • Las instancias bajo demanda le permiten utilizar la capacidad de computación por horas o por segundos sin compromis o a largo plazo. Estas instancias son adecuadas para ampliar la capacidad por encima de las necesidades de rendimiento estándar.<br>• Los Savings Plans ofrecen un ahorro significativo en comparación con las instancias bajo demanda a cambio del compromiso de utilizar una cantidad específica de capacidad de computación durante un período de uno o tres años.<br>• Las instancias de spot le permiten aprovechar la capacidad de las instancias que no se utilizan en cargas de trabajo sin estado y tolerantes a errores con descuento . |
| escalado automático | Use la configuración de escalado automático para ajustar los recursos de computación a los patrones de tráfico. |
| Ajuste del tamaño | • Use Compute Optimizer para obtener recomendaciones con tecnología de machine learning sobre qué configuración de computación se ajusta mejor a sus características de computación.<br>• Use AWS Lambda Power Tuning para seleccionar la mejor configuración para su función de Lambda. |

| Opción de configuración | Ejemplos |
| --- | --- |
| Aceleradores de computación basados en hardware | • Las instancias de computación acelerada ponen en marcha funciones de diversos tipos, por ejemplo, de procesamiento de gráficos o de búsqueda de patrones de datos, de manera más eficiente que las alternativas basadas en CPU.<br>• Para las cargas de trabajo de machine learning, utilice hardware personalizado específico para su carga de trabajo, como AWS Trainium, AWS Inferentia y Amazon EC2 DL1 |

Recursos

Documentos relacionados:

- Computación en la nube con AWS
- Tipos de instancias de Amazon EC
- Control de los estados del procesador de la instancia de Amazon EC2 Linux
- Contenedores de Amazon EKS: nodos de trabajo de Amazon EKS
- Contenedores de Amazon ECS: instancias de contenedor de Amazon ECS
- Funciones: configuración de funciones de Lambda

Videos relacionados:

- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in Consola de administración de AWS
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

- AWS re:Invent 2022 – Optimizing Amazon EKS for performance and cost on AWS

Ejemplos relacionados:

- Código de demostración de Compute Optimizer
- Amazon EC2 spot instances workshop
- Efficient and Resilient Workloads with Amazon EC2 AWS Auto Scaling
- Graviton developer workshop
- AWS for Microsoft workloads immersion day
- AWS for Linux workloads immersion day
- Código de demostración de AWS Compute Optimizer
- Taller de Amazon EKS

PERF02-BP03 Recopilación de métricas relacionadas con la computación

Registre y supervise las métricas relacionadas con los recursos de computación para comprender mejor el rendimiento de los recursos de computación y mejorar su rendimiento y su uso.

Patrones comunes de uso no recomendados:

- Solo se utiliza la búsqueda manual de métricas en los archivos de registro.
- Solo utiliza las métricas predeterminadas registradas en el software de supervisión seleccionado.
- Solo se revisan las métricas cuando hay un problema.

Beneficios de establecer esta práctica recomendada: recopilar métricas relacionadas con el rendimiento le permitirá ajustar el rendimiento de las aplicaciones a los requisitos empresariales para garantizar que cumple con las necesidades de su carga de trabajo. También puede ser de ayuda para mejorar continuamente el rendimiento y el uso de los recursos en su carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Las cargas de trabajo en la nube pueden generar grandes volúmenes de datos, como métricas, registros y eventos. En Nube de AWS, la recopilación de métricas es un paso crucial para mejorar

la seguridad, la rentabilidad, el rendimiento y la sostenibilidad. AWS ofrece una amplia variedad de métricas relacionadas con el rendimiento a través de servicios de supervisión como Amazon CloudWatch para proporcionarle información valiosa. Las métricas como la utilización de la CPU, la utilización de la memoria, las operaciones de E/S del disco y la entrada y salida de la red pueden proporcionar información sobre los niveles de uso o los cuellos de botella del rendimiento. Utilice estas métricas como parte de un enfoque basado en datos para ajustar y optimizar activamente los recursos de su carga de trabajo.  En un supuesto ideal, debería recopilar todas las métricas relacionadas con sus recursos de computación en una única plataforma que tuviera políticas de retención implementadas para satisfacer los objetivos operativos y financieros.

Pasos para la implementación

- Identifique qué métricas relacionadas con el rendimiento son relevantes para su carga de trabajo. Debe recopilar métricas sobre el uso de los recursos y la forma en que funciona su carga de trabajo en la nube (por ejemplo, el tiempo de respuesta y el rendimiento).
  - Amazon EC2 default metrics
  - Amazon ECS default metrics
  - Amazon EKS default metrics
  - Lambda default metrics
  - Amazon EC2 memory and disk metrics
- Elija y configure la solución de registro y supervisión adecuada para su carga de trabajo.
  - AWS native Observability
  - AWS Distro para OpenTelemetry
  - Servicio administrado por Amazon para Prometheus
- Defina el filtro y la agregación que se necesitan para las métricas en función de los requisitos de su carga de trabajo.
  - Quantify custom application metrics with Amazon CloudWatch Logs and metric filters
  - Collect custom metrics with Amazon CloudWatch strategic tagging
- Configure políticas de retención de datos para que las métricas se ajusten a los objetivos operativos y de seguridad.
  - Retención de datos predeterminada para las métricas de CloudWatch
  - Retención de datos predeterminada para Registros de CloudWatch
- Si es necesario, cree alarmas y notificaciones para sus métricas, lo que le ayudará a responder de manera proactiva a los problemas relacionados con el rendimiento.

  - Create alarms for custom metrics using Amazon CloudWatch anomaly detection
  - Create metrics and alarms for specific web pages with Amazon CloudWatch RUM
- Utilice la automatización para implementar los agentes de agregación de métricas y registros.
  - AWS Systems Manager automation
  - OpenTelemetry Collector

Recursos

Documentos relacionados:

- Monitoreo y observabilidad
- Best practices: implementing observability with AWS
- Documentación de Amazon CloudWatch
- Recopilación de métricas y registros de instancias de Amazon EC2 y en los servidores en las instalaciones con el agente de CloudWatch
- Uso de Registros de Amazon CloudWatch con AWS Lambda
- Uso de Registros de CloudWatch con instancias de contenedor
- Publish custom metrics
- AWS Answers: Registro centralizado
- Servicios de AWS que publican métricas de CloudWatch
- Monitoring Amazon EKS on AWS Fargate

Videos relacionados:

- AWS re:Invent 2023 – [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 – Implementing application observability
- AWS re:Invent 2023 – Building an effective observability strategy
- AWS re:Invent 2023 – Seamless observability with AWS Distro for OpenTelemetry
- Application Performance Management on AWS

Ejemplos relacionados:

- AWS for Linux Workloads Immersion Day- Amazon CloudWatch

- Monitoring Amazon ECS clusters and containers
- Monitoring with Amazon CloudWatch dashboards
- Taller de Amazon EKS

PERF02-BP04 Configuración y dimensionamiento correcto de los recursos de computación

Configure y dimensione correctamente los recursos de computación para que se ajusten a los requisitos de rendimiento de su carga de trabajo y evitar la infrautilización o el uso excesivo de recursos.

Patrones comunes de uso no recomendados:

- Ignora los requisitos de rendimiento de la carga de trabajo, lo que genera una falta o un exceso de aprovisionamiento de recursos de computación.
- Solo elige la instancia más grande o más pequeña disponible para todas las cargas de trabajo.
- Solo usa una familia de instancias para facilitar la administración.
- No tiene en cuenta las recomendaciones de AWS Cost Explorer o Compute Optimizer para ajustar el tamaño.
- No somete a nuevas evaluaciones a la carga de trabajo para determinar la idoneidad de nuevos tipos de instancias.
- Solo certifica una pequeña cantidad de configuraciones de instancias para su organización.

Beneficios de establecer esta práctica recomendada: el dimensionamiento correcto de los recursos de computación garantiza un funcionamiento óptimo en la nube al evitar que se produzca un exceso o falta de aprovisionamiento de recursos. El dimensionamiento adecuado de los recursos computacionales generalmente se traduce en un mayor rendimiento y una mejor experiencia del cliente, al tiempo que se reducen los costos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Un dimensionamiento correcto permite a las organizaciones gestionar la infraestructura en la nube de manera eficiente y rentable, al tiempo que abordan sus necesidades empresariales. Un aprovisionamiento excesivo de los recursos en la nube puede generar costos adicionales, mientras que un aprovisionamiento insuficiente puede provocar un rendimiento deficiente y una experiencia

negativa para el cliente. AWS proporciona herramientas como AWS Compute Optimizer y AWS Trusted Advisor, que utilizan datos históricos para ofrecer recomendaciones sobre el tamaño adecuado de sus recursos de computación.

Pasos para la implementación

- Elija el tipo de instancia que mejor se adapte a sus necesidades:
  - ¿Cómo elijo el tipo de instancia de Amazon EC2 apropiado para mi carga de trabajo?
  - Selección de tipo de instancia basada en atributos para la Flota de Amazon EC2
  - Create an Auto Scaling group using attribute-based instance type selection
  - Optimizing your Kubernetes compute costs with Karpenter consolidation
- Analice las distintas características de rendimiento de su carga de trabajo y la relación que tienen con el uso de memoria, redes y CPU. Use estos datos para elegir recursos que encajen bien con el perfil de la carga de trabajo y los objetivos de rendimiento.
- Controle el uso de los recursos con las herramientas de supervisión de AWS, como Amazon CloudWatch.
- Seleccione la configuración correcta para cada recurso de computación.
  - En el caso de cargas de trabajo efímeras, evalúe las métricas de Amazon CloudWatch de la instancia, como `CPUUtilization`, para identificar si la instancia está infrautilizada o sobreutilizada.
  - En las cargas de trabajo estables, consulte regularmente las herramientas de dimensionamiento de AWS, como AWS Compute Optimizer y AWS Trusted Advisor, para identificar oportunidades de optimizar y dimensionar correctamente el recurso de computación.
- Pruebe los cambios de configuración en un entorno que no sea de producción antes de implementarlos en un entorno activo.
- Revalúe continuamente las nuevas ofertas de computación y compárelas con las necesidades de la carga de trabajo.

Recursos

Documentos relacionados:

- Computación en la nube con AWS
- Tipos de instancias de Amazon EC
- Contenedores de Amazon ECS: instancias de contenedor de Amazon ECS

- Contenedores de Amazon EKS: nodos de trabajo de Amazon EKS
- Funciones: configuración de funciones de Lambda
- Control de los estados del procesador de la instancia de Amazon EC2

Videos relacionados:

- Amazon EC2 foundations
- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in Consola de administración de AWS
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

Ejemplos relacionados:

- Código de demostración de AWS Compute Optimizer
- Taller de Amazon EKS
- Right-sizing recommendations

PERF02-BP05 Escalado de los recursos de computación de forma dinámica

Utilice la elasticidad de la nube para aumentar o reducir sus recursos computacionales de forma dinámica de forma que se ajusten a sus necesidades, lo que evitará un aprovisionamiento de capacidad excesivo o insuficiente para su carga de trabajo.

Patrones comunes de uso no recomendados:

- Reacciona a las alarmas mediante el aumento manual de la capacidad.
- Utiliza las mismas directrices de dimensionamiento (por lo general, una infraestructura estática) que en el entorno en las instalaciones.
- Dejar la capacidad aumentada después de un evento de ajuste de escala en lugar de volver a desescalar verticalmente.

Beneficios de establecer esta práctica recomendada: configurar y probar la elasticidad de los recursos de computación puede ser útil para ahorrar dinero, mantener los puntos de referencia de rendimiento y mejorar la fiabilidad a medida que cambia el tráfico.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

AWS le ofrece la flexibilidad necesaria para aumentar o reducir los recursos de forma dinámica a través de una gran variedad de mecanismos de escalado que se ajustan a los cambios de demanda. Junto con las métricas relacionadas con la computación, el escalado dinámico permite que las cargas de trabajo respondan automáticamente a los cambios y utilicen el conjunto óptimo de recursos de computación para lograr su objetivo.

Puede usar distintos enfoques para hacer que el suministro de recursos coincida con la demanda.

- Enfoque de seguimiento del objetivo: supervise la métrica de escalado y aumente o reduzca de forma automática la capacidad en función de sus necesidades.
- Escalado predictivo: reduzca horizontalmente de antemano según las tendencias diarias y semanales previstas.
- Enfoque basado en una programación: establezca su propia programación de escalado según los cambios de carga predecibles.
- Escalado de servicio: elija servicios (como los servicios sin servidor) diseñados para escalar automáticamente.

Debe asegurarse de que las implementaciones de la carga de trabajo puedan manejar eventos de escalado vertical y reducción vertical.

Pasos para la implementación

- Las instancias de computación, los contenedores y las funciones proporcionan mecanismos que favorecen la elasticidad, ya sea en combinación con funciones de escalado automático o como características del servicio. Estos son algunos ejemplos de mecanismos de escalado automático:

| Mecanismo de escalado automático | Dónde se usa |
| --- | --- |
| Amazon EC2 Auto Scaling | Para garantizar que cuenta con la cantidad correcta de instancias de Amazon EC2 |

| Mecanismo de escalado automático | Dónde se usa |
| --- | --- |
| | disponibles para controlar la carga de usuarios de su aplicación. |
| Aplicación de escalado automático | Para escalar automáticamente los recursos de servicios de AWS específicos más allá de Amazon EC2, como las funciones de AWS Lambda o los servicios de Amazon Elastic Container Service (Amazon ECS). |
| Kubernetes Cluster Autoscaler/Karpenter | Para escalar automáticamente clústeres de Kubernetes. |

- Normalmente, se habla del escalado en relación con los servicios de computación, como las instancias de Amazon EC2 o las funciones de AWS Lambda. No olvide que también debe tener en cuenta la configuración de otros servicios no computacionales, como AWS Glue, para adaptarse a la demanda.
- Asegúrese de que las métricas de escalado se ajusten a las características de la carga de trabajo que se implementa. Si está implementando una aplicación de transcodificación de vídeo, se espera un uso del 100 % de la CPU y no debería ser su métrica principal. En su lugar, utilice la profundidad de la cola de trabajos de transcodificación. Si es necesario, puede utilizar una métrica personalizada para su política de escalado. Para elegir las métricas adecuadas, tenga en cuenta las siguientes directrices para Amazon EC2:
  - La métrica debe ser una métrica de utilización válida y describir el grado de ocupación de una instancia.
  - El valor de la métrica debe aumentar o disminuir proporcionalmente al número de instancias del grupo de escalado automático.
- Asegúrese de usar el escalado dinámico en lugar del escalado manual para su grupo de escalado automático. También le recomendamos que utilice políticas de escalado de seguimiento objetivo en su escalado dinámico.
- Compruebe que las implementaciones de la carga de trabajo puedan gestionar ambos eventos de escalado (escalado vertical y reducción vertical). Por ejemplo, puede usar el historial de actividad para verificar la actividad de escalado de un grupo de escalado automático.
- Evalúe los patrones predecibles de su carga de trabajo y escale de forma proactiva para anticiparse a los cambios previstos y planeados en la demanda. Con el escalado predictivo, puede

eliminar la necesidad de aprovisionar capacidad en exceso. Para más información, consulte Predictive Scaling with Amazon EC2 Auto Scaling.

Recursos

Documentos relacionados:

- Computación en la nube con AWS
- Tipos de instancias de Amazon EC
- Contenedores de Amazon ECS: instancias de contenedor de Amazon ECS
- Contenedores de Amazon EKS: nodos de trabajo de Amazon EKS
- Funciones: configuración de funciones de Lambda
- Control de los estados del procesador de la instancia de Amazon EC2
- Deep Dive on Amazon ECS Cluster Auto Scaling
- Introducing Karpenter – An Open-Source High-Performance Kubernetes Cluster Autoscaler

Videos relacionados:

- AWS re:Invent 2023 – AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS Management Console
- AWS re:Invent 2023 – What's new with Amazon EC2
- AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System
- AWS re:Invent 2019 – Amazon EC2 foundations

Ejemplos relacionados:

- Ejemplos de grupos de Amazon EC2 Auto Scaling
- Taller de Amazon EKS
- Scale your Amazon EKS workloads by running on IPv6

PERF02-BP06 Uso de aceleradores de computación optimizados basados en hardware

Use aceleradores de hardware para llevar a cabo ciertas funciones de manera más eficiente que con las alternativas basadas en CPU.

Patrones comunes de uso no recomendados:

- En su carga de trabajo, no ha comparado una instancia de uso general con una instancia personalizada que pueda ofrecer mayor rendimiento y costos más reducidos.
- Utiliza aceleradores de computación basados en hardware para tareas en las que pueda ser más eficiente utilizar alternativas basadas en CPU.
- No supervisa el uso de GPU.

Beneficios de establecer esta práctica recomendada: al utilizar aceleradores basados en hardware, como unidades de procesamiento gráfico (GPU) y matrices de puertas programables en campo (FPGA), puede poner en marcha determinadas funciones de procesamiento de manera más eficiente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Las instancias de computación acelerada proporcionan acceso a aceleradores de computación basados en hardware, como las GPU y las FPGA. Estos aceleradores de hardware llevan a cabo ciertas funciones, como el procesamiento gráfico o la concordancia de patrones de datos, de forma más eficiente que las alternativas basadas en CPU. Muchas cargas de trabajo aceleradas, como el renderizado, la transcodificación y el machine learning, son muy variables en cuanto al uso de recursos. Ejecute este hardware solo durante el tiempo que sea necesario y retírelo mediante automatización cuando no se requiera para mejorar la eficiencia del rendimiento general.

Pasos para la implementación

- Identifique qué instancias de computación acelerada pueden satisfacer sus requisitos.
- Para las cargas de trabajo de machine learning, utilice hardware personalizado específico para la carga de trabajo, como AWS Trainium, AWS Inferentia y Amazon EC2 DL1. AWS Las instancias de Inferentia, como las instancias Inf2, ofrecen hasta un 50 % más de rendimiento por vatio que las instancias de Amazon EC2 comparables.
- Recopile las métricas de uso de las instancias de computación acelerada. Por ejemplo, puede usar el agente de CloudWatch para recopilar métricas como `utilization_gpu` y

`utilization_memory` para sus GPU, como se muestra en Recopilación de métricas de GPU NVIDIA con Amazon CloudWatch.

- Optimice el código, el funcionamiento de la red y la configuración de los aceleradores de hardware para asegurarse de que se aprovecha al máximo el hardware subyacente.
  - Optimización de las configuraciones de GPU
  - GPU Monitoring and Optimization in the Deep Learning AMI
  - Optimizing I/O for GPU performance tuning of deep learning training in Amazon SageMaker AI
- Utilice las bibliotecas de alto rendimiento y los controladores de GPU más recientes.
- Use la automatización para liberar instancias de GPU cuando no se estén usando.

Recursos

Documentos relacionados:

- Uso de GPU en Amazon Elastic Container Service
- instancias de GPU
- Instances with AWS Trainium
- Instances with AWS Inferentia
- Let's Architect! Architecting with custom chips and accelerators

- Computación acelerada
- Amazon EC2 VT1 Instances
- ¿Cómo elijo el tipo de instancia de Amazon EC2 apropiado para mi carga de trabajo?
- Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker AI

Videos relacionados:

- AWS re:Invent 2021 - How to select Amazon Elastic Compute Cloud GPU instances for deep learning
- AWS re:Invent 2022 - [NEW LAUNCH!] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances
- AWS re:Invent 2022 - Accelerate deep learning and innovate faster with AWS Trainium

- AWS re:Invent 2022 - Deep learning on AWS with NVIDIA: From training to deployment

Ejemplos relacionados:

- Amazon SageMaker AI y NVIDIA GPU Cloud (NGC)
- Uso de SageMaker AI con Trainium e Inferentia para optimizar las cargas de trabajo de aprendizaje profundo, entrenamiento e inferencia
- Optimización de modelos de NLP con instancias Inf1 de Amazon Elastic Compute Cloud en Amazon SageMaker AI

## Administración de datos

Preguntas

- PERF 3. ¿Cómo almacena, administra y accede a los datos de su carga de trabajo?

### PERF 3. ¿Cómo almacena, administra y accede a los datos de su carga de trabajo?

La solución de administración de datos óptima para un sistema concreto varía según el tipo de datos (bloque, archivo u objeto), patrones de acceso (aleatorio o secuencial), rendimiento requerido, frecuencia de acceso (en línea, fuera de línea, archivo), frecuencia de actualización (WORM, dinámica), y restricciones de disponibilidad y durabilidad. Las cargas de trabajo de Well-Architected utilizan almacenes de datos diseñados específicamente que admiten diferentes características para mejorar el rendimiento.

Prácticas recomendadas

- PERF03-BP01 Uso de un almacén de datos personalizado que se adapte mejor a los requisitos de acceso y almacenamiento de datos
- PERF03-BP02 Evaluación de las opciones de configuración disponibles
- PERF03-BP03 Recopilación y registro de las métricas de rendimiento del almacén de datos
- PERF03-BP04 Implementación de estrategias para mejorar el rendimiento de las consultas en el almacén de datos
- PERF03-BP05 Implementación de patrones de acceso a datos que utilicen el almacenamiento en caché

PERF03-BP01 Uso de un almacén de datos personalizado que se adapte mejor a los requisitos de acceso y almacenamiento de datos

Debe saber cuáles son las características de los datos (por ejemplo, si se pueden compartir, su tamaño, los patrones de acceso, la latencia, el rendimiento y su persistencia) para seleccionar los almacenes de datos personalizados acordes a su carga de trabajo (almacenamiento o base de datos).

Patrones comunes de uso no recomendados:

- Utiliza exclusivamente un almacén de datos porque la experiencia y los conocimientos internos se limitan a un tipo concreto de solución de base de datos.
- Presupone que todas las cargas de trabajo tienen unos requisitos similares en relación con el almacenamiento de datos y el acceso a la información.
- No ha implementado un catálogo de datos para inventariar sus activos de datos.

Beneficios de establecer esta práctica recomendada: comprender las características y los requisitos de los datos le permite determinar la tecnología de almacenamiento más eficiente y funcional para las necesidades de su carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Al seleccionar e implementar el almacenamiento de datos, asegúrese de que las características de consulta, escalado y almacenamiento se ajusten a los requisitos de datos de la carga de trabajo. AWS ofrece un gran número de tecnologías de almacenamiento y bases de datos, como el almacenamiento en bloques, el almacenamiento de objetos, el almacenamiento en streaming, los sistemas de archivos, las bases de datos relacionales, las bases de datos de clave-valor, las bases de datos de documentos, las bases de datos en memoria, las bases de datos de grafos, las bases de datos de series temporales y las bases de datos de libro mayor. Cada solución de administración de datos tiene opciones y configuraciones a su disposición que se ajustan a los casos de uso y a los modelos de datos. Si conoce las características y los requisitos de los datos, puede dejar atrás la tecnología de almacenamiento monolítica y los enfoques restrictivos de “una misma cosa vale para todo”, y centrarse en gestionar correctamente los datos.

Pasos para la implementación

- Haga un inventario de los distintos tipos de datos que existen en su carga de trabajo.

- Estudie y documente las características y los requisitos de los datos, como:
  - Tipo de datos (no estructurados, semiestructurados o relacionales)
  - Volumen y crecimiento de los datos
  - Durabilidad de los datos: persistentes, efímeros o transitorios
  - Requisitos de ACID (atomicidad, consistencia, aislamiento, durabilidad)
  - Patrones de acceso a los datos (lectura o escritura intensivas)
  - Latencia
  - Rendimiento
  - IOPS (operaciones de entrada/salida por segundo)
  - Periodo de retención de datos
- Obtenga información sobre los diferentes almacenes de datos (servicios de almacenamiento y base de datos) disponibles en AWS para su carga de trabajo que se ajustan a las características de los datos, tal y como se describe en PERF01-BP01 Descubrimiento y comprensión de los servicios y las características disponibles en la nube. Estos son algunos ejemplos de tecnologías de almacenamiento de AWS y sus principales características:

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Almacenamiento de objetos | Amazon S3 | Escalabilidad ilimitada, alta disponibilidad y múltiples opciones de accesibilidad. La transferencia y el acceso a objetos dentro y fuera de Amazon S3 puede utilizar un servicio, como Aceleración de transferencias o Puntos de acceso, para respaldar su ubicación, sus necesidades de seguridad y sus patrones de acceso. |
| Almacenamiento de archivos | de Amazon Glacier | Diseñado para archivar datos. |

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Almacenamiento en streaming | Amazon Kinesis<br><br>Amazon Managed Streaming para Apache Kafka (Amazon MSK) | Ingesta y almacenamiento eficientes de datos de streaming. |
| Sistema de archivos compartidos | Amazon Elastic File System (Amazon EFS) | Sistema de archivos montable al que pueden acceder varios tipos de soluciones de computación. |
| Sistema de archivos compartidos | Amazon FSx | Se basa en las últimas soluciones de computación de AWS para admitir cuatro sistemas de archivos de uso común: NetApp ONTAP, OpenZFS, Windows File Server y Lustre. La latencia, rendimiento e IOPS de Amazon FSx varían según el sistema de archivos y deben tenerse en cuenta a la hora de seleccionar el sistema de archivos adecuado para sus necesidades de carga de trabajo. |

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Almacenamiento en bloque | Amazon Elastic Block Store (Amazon EBS) | Servicio de almacenamiento en bloque de alto rendimiento, escalable y fácil de usar diseñado para Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS incluye almacenamiento respaldado por SSD para cargas de trabajo transaccionales y de IOPS intensivas, así como almacenamiento respaldado por HDD para cargas de trabajo de rendimiento intensivo. |
| Base de datos relacional | Amazon Aurora, Amazon RDS, Amazon Redshift. | Se han diseñado para respaldar las transacciones ACID (atomicidad, coherencia, aislamiento, durabilidad) y mantener la integridad referencial y una sólida coherencia de datos. Muchas aplicaciones tradicionales, la planificación de recursos empresariales (ERP), la administración de relaciones con los clientes (CRM) y el comercio electrónico utilizan bases de datos relacionales para almacenar sus datos. |

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Base de datos de clave-valor | Amazon DynamoDB | Optimizada para patrones de acceso comunes, normalmente para almacenar y recuperar grandes volúmenes de datos. Las aplicaciones web con mucho tráfico, los sistemas de comercio electrónico y las aplicaciones de juegos son casos de uso típicos para las bases de datos de clave-valor. |
| Base de datos de documentos | Amazon DocumentDB | Diseñada para almacenar datos semiestructurados como documentos tipo JSON. Estas bases de datos ayudan a los desarrolladores a crear y actualizar de forma rápida aplicaciones como la administración de contenido, catálogos y perfiles de usuario. |

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Base de datos en memoria | Amazon ElastiCache, Amazon MemoryDB para Redis | Se utilizan para aplicaciones que requieren acceso a los datos en tiempo real, menor latencia y mayor rendimiento. Puede usar bases de datos en memoria para el almacenamiento en caché de aplicaciones, la administración de sesiones, las tablas de clasificación de juegos, el almacén de características de ML de baja latencia, el sistema de mensajería de microservicios y un mecanismo de streaming de alto rendimiento. |
| Base de datos de gráficos | Amazon Neptune | Se utiliza para aplicaciones que deben navegar y consultar millones de relaciones entre conjuntos de datos de grafos con un alto grado de conexión y con una latencia de milisegundos a gran escala. Muchas empresas utilizan las bases de datos de gráficos para detección de fraude, redes sociales y motores de recomendaciones. |

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Base de datos de serie temporal | Amazon Timestream | Se usa para recopilar, sintetizar y obtener informaci ón de forma eficaz a partir de datos que cambian con el tiempo. Las aplicaciones de IoT, DevOps y telemetrí a industrial pueden utilizar bases de datos de serie temporal. |
| Columna ancha | Amazon Keyspaces (para Apache Cassandra) | Utiliza tablas, filas y columnas, pero, a diferenci a de una base de datos relacional, los nombres y el formato de las columnas pueden variar de una fila a otra en la misma tabla. Por lo general, un almacén de columnas anchas está en aplicaciones industria les a gran escala para el mantenimiento de equipos, la administración de flotas y la optimización de rutas. |

| Tipo | AWS Servicios de | Características clave |
| --- | --- | --- |
| Libro mayor | Amazon Quantum Ledger Database (Amazon QLDB) | Proporciona una autoridad centralizada y de confianza para mantener un registro de transacciones escalable , inmutable y verificable criptográficamente para cada aplicación. Las bases de datos de libro mayor se utilizan para sistemas de registro, la cadena de suministro, registros e incluso transacciones bancarias. |

- Si está creando una plataforma de datos, aproveche la arquitectura de datos moderna en AWS para integrar su lago de datos, su almacenamiento de datos y sus almacenes de datos personalizados.
- Las principales preguntas que debe hacerse al elegir un almacén de datos para su carga de trabajo son las siguientes:

| Pregunta | Aspectos que deben tenerse en cuenta |
| --- | --- |
| ¿Cómo se estructuran los datos? | • Si los datos no están estructurados, considere un almacén de objetos como Amazon S3 o una base de datos NoSQL como Amazon DocumentDB<br>• Para los datos de clave-valor, considere DynamoDB, Amazon ElastiCache (Redis OSS) o Amazon MemoryDB |
| ¿Qué nivel de integridad referencial se requiere? | • Para las restricciones de claves externas, las bases de datos relacionales como Amazon RDS y Aurora pueden proporcionar este nivel de integridad. |

| Pregunta | Aspectos que deben tenerse en cuenta |
| --- | --- |
| | • Normalmente, en un modelo de datos NoSQL, los datos se desnormalizarían en un documento o una colección de documentos en lugar de combinarse en diferentes documentos o tablas, lo que permitiría recuperarlos en una única solicitud. |
| ¿Se requiere el cumplimiento de ACID (atomicidad, coherencia, aislamiento, durabilidad)? | • Si se requiere cumplir las propiedades ACID asociadas a las bases de datos relacionales, considere la posibilidad de usar una base de datos relacional como Amazon RDS y Aurora.<br>• Si se requiere una coherencia sólida para la base de datos NoSQL, puede utilizar lecturas altamente coherentes con DynamoDB. |
| ¿Cómo cambiarán los requisitos de almacenamiento con el tiempo? ¿Cómo afecta esto a la escalabilidad? | • Las bases de datos sin servidor, como DynamoDB y Amazon Quantum Ledger Database (Amazon QLDB), se escalarán de forma dinámica.<br>• Las bases de datos relacionales tienen límites máximos de almacenamiento aprovisionado y, a menudo, cuando alcanzan estos límites, es necesario hacer particiones horizontales a través de diversos mecanismos, como el particionamiento. |

| Pregunta | Aspectos que deben tenerse en cuenta |
| --- | --- |
| ¿Cuál es la proporción de consultas de lectura en relación con las de escritura? ¿Es probable que el almacenamiento en caché mejore el rendimiento? | • Las cargas de trabajo de lectura intensiva pueden beneficiarse de una capa de almacenamiento en caché, como ElastiCache o DAX si la base de datos es de DynamoDB.<br>• Las lecturas también pueden descargarse en réplicas de lectura con bases de datos relacionales, como Amazon RDS. |
| ¿Tiene mayor prioridad el almacenamiento y la modificación (OLTP, procesamiento de transacciones en línea) o la recuperación y la elaboración de informes (OLAP, procesamiento analítico en línea)? | • Para el procesamiento transaccional de lecturas de alto rendimiento sin hacer cambios, considere la posibilidad de usar una base de datos NoSQL, como DynamoDB.<br>• En el caso de los patrones de lectura complejos y de alto rendimiento (como una combinación) que tienen coherencia, use Amazon RDS.<br>• Para las consultas analíticas, considere utilizar una base de datos en columnas como Amazon Redshift o exportar los datos a Amazon S3 y llevar a cabo análisis con Athena o Amazon Quick. |

| Pregunta | Aspectos que deben tenerse en cuenta |
| --- | --- |
| ¿Qué nivel de durabilidad requieren los datos? | • Aurora replica los datos automáticamente en tres zonas de disponibilidad de una región, lo que significa que los datos tendrán una gran durabilidad y menos posibilidades de sufrir pérdidas.<br>• DynamoDB se replica automáticamente en varias zonas de disponibilidad, lo que proporciona una elevada disponibilidad y durabilidad de los datos.<br>• Amazon S3 proporciona un nivel de durabilidad de once nueves. Muchos servicios de bases de datos, como Amazon RDS y DynamoDB, permiten exportar datos a Amazon S3 para retenerlos y archivarlos durante largos periodos de tiempo. |
| ¿Existe el deseo de evitar los motores de bases de datos comerciales o los costos de licencia? | • Considere la posibilidad de utilizar motores de código abierto como PostgreSQL y MySQL en Amazon RDS o Aurora.<br>• Use AWS Database Migration Service y AWS Schema Conversion Tool para llevar a cabo migraciones de los motores de bases de datos comerciales a los de código abierto. |
| ¿Cuál es la expectativa operativa de la base de datos? ¿El cambio a los servicios administr ados es una preocupación principal? | • Si usa Amazon RDS en lugar de Amazon EC2 y utiliza DynamoDB o Amazon DocumentDB en lugar de alojar una base de datos NoSQL en sus propios sistemas, puede reducir los costos operativos. |

| Pregunta | Aspectos que deben tenerse en cuenta |
| --- | --- |
| ¿Cómo se accede actualmente a la base de datos? ¿Se trata solo del acceso a la aplicación, o hay usuarios de inteligencia empresarial (BI) y otras aplicaciones comerciales conectadas? | • Si tiene dependencias en herramientas externas, es posible que deba mantener la compatibilidad con las bases de datos que admiten. Amazon RDS es totalmente compatible con las diferentes versiones de motores que admite, como Microsoft SQL Server, Oracle, MySQL y PostgreSQL. |

- Lleve a cabo experimentos y pruebas comparativas en un entorno que no sea de producción para identificar qué almacén de datos se ajusta a los requisitos de su carga de trabajo.

Recursos

Documentos relacionados:

- Tipos de volúmenes de Amazon EBS
- Opciones de almacenamiento para sus instancias de Amazon EC2
- Amazon EFS: Amazon EFS Performance
- Amazon FSx for Lustre Performance
- Amazon FSx for Windows File Server Performance
- Amazon Glacier: documentación de Amazon Glacier
- Amazon S3: consideraciones de la tasa de solicitudes y del rendimiento
- Almacenamiento en la nube en AWS
- Amazon EBS I/O Characteristics
- Bases de datos en la nube de AWS
- AWS Database Caching
- DynamoDB Accelerator
- Prácticas recomendadas de Amazon Aurora
- Rendimiento de Amazon Redshift
- Amazon Athena top 10 performance tips
- Amazon Redshift Spectrum best practices
- Prácticas recomendadas para Amazon DynamoDB

- Choose between Amazon EC2 and Amazon RDS
- Best Practices for Implementing Amazon ElastiCache

Videos relacionados:

- AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient
- AWS re:Invent 2023: Optimizing storage price and performance with Amazon Simple Storage Service
- AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service
- AWS re:Invent 2022: Building modern data architectures on AWS
- AWS re:Invent 2022: Building data mesh architectures on AWS
- AWS re:Invent 2023: Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023: Advanced data modeling with Amazon DynamoDB
- AWS re:Invent 2022: Modernize apps with purpose-built databases
- Amazon DynamoDB deep dive: Advanced design patterns

Ejemplos relacionados:

- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS
- Ejemplos de Amazon S
- Optimize Data Pattern using Amazon Redshift Data Sharing
- Database Migrations
- MS SQL Server - AWS Database Migration Service (AWS DMS) Replication Demo
- Database Modernization Hands On Workshop
- Amazon Neptune Samples

### PERF03-BP02 Evaluación de las opciones de configuración disponibles

Estudie y evalúe las diversas características y opciones de configuración disponibles para sus almacenes de datos a fin de optimizar el espacio de almacenamiento y el rendimiento de su carga de trabajo.

Patrones comunes de uso no recomendados:

- Solo utiliza un tipo de almacenamiento, como, por ejemplo, Amazon EBS, para todas las cargas de trabajo.
- Utiliza IOPS aprovisionadas en todas las cargas de trabajo sin efectuar pruebas reales con todos los niveles de almacenamiento.
- No conoce las opciones de configuración de la solución de administración de datos que ha elegido.
- La única opción que contempla es aumentar el tamaño de las instancias, sin valorar otras opciones de configuración disponibles.
- No lleva a cabo pruebas en las características de escalado de su almacén de datos.

Beneficios de establecer esta práctica recomendada: si explora y experimenta con las configuraciones de almacenamiento de datos, puede reducir el costo de la infraestructura, mejorar el rendimiento y reducir el esfuerzo necesario para mantener sus cargas de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

En una carga de trabajo, puede haber uno o varios almacenamientos de datos que se utilicen en función de los requisitos de almacenamiento y acceso. Para optimizar los costos y la eficiencia del rendimiento, debe evaluar los patrones de acceso a los datos y determinar cuáles son las configuraciones de almacenamiento de datos adecuadas. Cuando explore las opciones de almacenamiento de datos, tenga en cuenta diversos aspectos, como las opciones de almacenamiento, la memoria, los recursos de computación, la réplica de lectura, los requisitos de coherencia, la agrupación de conexiones y las opciones de almacenamiento en caché. Pruebe estas diferentes opciones de configuración para mejorar las métricas de eficiencia del rendimiento.

Pasos para la implementación

- Estudie las configuraciones actuales (como el tipo de instancia, el tamaño de almacenamiento o la versión del motor de base de datos) de su almacén de datos.

- Consulte la documentación y las prácticas recomendadas de AWS para obtener información sobre las opciones de configuración recomendadas que pueden ser de ayuda para mejorar el rendimiento de su almacén de datos. Las principales opciones de almacenamiento de datos que debe tener en cuenta son las siguientes:

| Opción de configuración | Ejemplos |
| --- | --- |
| Descarga de lecturas (como réplicas de lectura y almacenamiento en caché) | • En el caso de las tablas de DynamoDB, puede descargar las lecturas con DAX para el almacenamiento en caché.<br>• Puede crear un clúster de Amazon ElastiCache (Redis OSS) y configurar la aplicación para que lea primero la memoria caché y, si el elemento solicitado no está presente, recurra a la base de datos.<br>• Las bases de datos relacionales, como Amazon RDS y Aurora, y las bases de datos NoSQL aprovisionadas, como Neptune y Amazon DocumentDB, permiten agregar réplicas de lectura para descargar las partes de lectura de la carga de trabajo.<br>• Las bases de datos sin servidor, como DynamoDB, se escalarán automátic amente. Asegúrese de que tenga suficient es unidades de capacidad de lectura (RCU) aprovisionadas para gestionar la carga de trabajo. |

| Opción de configuración | Ejemplos |
| --- | --- |
| Escalado de escrituras (como la fragmentación de claves de partición o la introducción de una cola) | • En el caso de las bases de datos relacionales, puede aumentar el tamaño de la instancia para acomodar una mayor carga de trabajo o aumentar las IOPS aprovisionadas para mejorar el rendimiento del almacenamiento subyacente.<br>• También puede introducir una cola delante de la base de datos en lugar de escribir directamente en la base de datos. Este patrón permite desacoplar la ingesta de la base de datos y controlar el caudal para que la base de datos no se vea desbordada.<br>• Si agrupa las solicitudes de escritura en lugar de crear muchas transacciones de corta duración, puede mejorar el rendimiento de las bases de datos relacionales con un gran volumen de operaciones de escritura.<br>• Las bases de datos sin servidor, como DynamoDB, pueden escalar el rendimiento de escritura automáticamente o al ajustar las unidades de capacidad de escritura (WCU) aprovisionadas en función del modo de capacidad.<br>• Puede tener problemas con las particiones activas si alcanza los límites de rendimiento de una clave de partición determinada. Esto puede mitigarse si se elige una clave de partición distribuida de manera más uniforme o se particiona la escritura en función de la clave de partición. |

| Opción de configuración | Ejemplos |
| --- | --- |
| Políticas para administrar el ciclo de vida de los conjuntos de datos | • Puede usar Amazon S3 Lifecycle para administrar los objetos a lo largo de su ciclo de vida. Si sus patrones de acceso son desconocidos, cambiantes o impredecibles, puede utilizar Amazon S3 Intelligent-Tiering, que supervisa los patrones de acceso y traslada automáticamente los objetos de una capa a otra. Puede aprovechar las métricas de Lente de almacenamiento de Amazon S3 para identificar las oportunidades de optimización y las brechas en la administración del ciclo de vida.<br>• La administración del ciclo de vida de Amazon EFS administra automáticamente el almacenamiento económico de los archivos para sus sistemas de archivos. |
| Administración y agrupación de conexiones | • Amazon RDS Proxy puede utilizarse con Amazon RDS y Aurora para administrar conexiones a la base de datos.<br>• Las bases de datos sin servidor, como DynamoDB, no tienen conexiones asociadas, pero tienen en cuenta la capacidad aprovisionada y las políticas de escalado automático para hacer frente a los picos de carga. |

- Lleve a cabo experimentos y pruebas comparativas en un entorno que no sea de producción para identificar qué opción de computación se ajusta a los requisitos de la carga de trabajo.
- Una vez hecho esto, planifique la migración y valide las métricas de rendimiento.
- Use las herramientas de supervisión (como Amazon CloudWatch) y optimización (como Lente de almacenamiento de Amazon S3) de AWS para optimizar continuamente el almacén de datos utilizando patrones de uso del mundo real.

Recursos

Documentos relacionados:

- Almacenamiento en la nube en AWS
- Tipos de volúmenes de Amazon EBS
- Opciones de almacenamiento para sus instancias de Amazon EC
- Amazon EFS: Amazon EFS Performance
- Amazon FSx for Lustre Performance
- Amazon FSx for Windows File Server Performance
- Amazon Glacier: documentación de Amazon Glacier
- Amazon S3: consideraciones de la tasa de solicitudes y del rendimiento
- Amazon EBS I/O Characteristics
- Bases de datos en la nube de AWS
- AWS Database Caching
- DynamoDB Accelerator
- Prácticas recomendadas de Amazon Aurora
- Rendimiento de Amazon Redshift
- Amazon Athena top 10 performance tips
- Amazon Redshift Spectrum best practices
- Prácticas recomendadas para Amazon DynamoDB

Videos relacionados:

- AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient
- AWS re:Invent 2023: Optimize storage price and performance with Amazon Simple Storage Service
- AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service
- AWS re:Invent 2023: What's new with AWS file storage
- AWS re:Invent 2023: Dive deep into Amazon DynamoDB

Ejemplos relacionados:

- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Amazon EBS Autoscale
- Ejemplos de Amazon S
- Ejemplos de Amazon DynamoDB
- AWS Database migration samples
- Database Modernization Workshop
- Working with parameters on your Amazon RDS for Postgress DB

PERF03-BP03 Recopilación y registro de las métricas de rendimiento del almacén de datos

Supervise y registre las métricas de rendimiento relevantes del almacén de datos para saber cómo funcionan las soluciones de administración de datos. Estas métricas pueden ser de ayuda para optimizar el almacén de datos, garantizar que se cumplen los requisitos de la carga de trabajo y proporcionar una visión general clara del rendimiento de la carga de trabajo.

Patrones comunes de uso no recomendados:

- Solo se utiliza la búsqueda manual de métricas en los archivos de registro.
- Solo publica métricas en las herramientas internas que su equipo utiliza y no tiene una imagen completa de su carga de trabajo.
- Solo se utilizan las métricas predeterminadas registradas por el software de supervisión seleccionado.
- Solo se revisan las métricas cuando hay un problema.
- Solo se supervisan las métricas del sistema y no se captura las métricas de acceso o de uso de datos.

Beneficios de establecer esta práctica recomendada: instaurar una base de referencia de rendimiento le permite comprender el comportamiento habitual y los requisitos de las cargas de trabajo. Los patrones anómalos pueden identificarse y depurarse más rápidamente, lo que mejora el rendimiento y la fiabilidad del almacén de datos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para supervisar el rendimiento de sus almacenes de trabajo, debe registrar diversas métricas de rendimiento a lo largo del tiempo. De este modo, podrá detectar anomalías y medir el rendimiento con respecto a las métricas de la empresa para asegurarse de que se están satisfaciendo las necesidades de su carga de trabajo.

Las métricas deben incluir tanto el sistema subyacente que da servicio al almacén de datos como las métricas de la base de datos. Las métricas del sistema subyacente podrían ser el uso de la CPU, la memoria, el almacenamiento en disco disponible, las operaciones de E/S del disco, la proporción de aciertos de la caché y las métricas de entrada y salida de la red, mientras que las métricas del almacén de datos podrían ser las transacciones por segundo, las consultas principales, las tasas medias de consultas, los tiempos de respuesta, el uso de índices, los bloqueos de tablas, los tiempos de espera de las consultas y el número de conexiones abiertas. Estos datos son cruciales para entender cómo funciona la carga de trabajo y cómo se utiliza la solución de administración de datos. Utilice estas métricas como parte de un enfoque basado en datos para ajustar y optimizar los recursos de la carga de trabajo.

Use herramientas, bibliotecas y sistemas que registren las medidas de rendimiento relacionadas con el rendimiento de la base de datos.

Pasos para la implementación

- Identifique las métricas de rendimiento clave del almacén de datos que desee supervisar.
  - Métricas y dimensiones de Amazon S3
  - Supervisión de métricas en una instancia de Amazon RDS
  - Monitoreo de la carga de base de datos con Performance Insights en Amazon RDS
  - Descripción general de la supervisión mejorada
  - Dimensiones y métricas de DynamoDB
  - Supervisión de DynamoDB Accelerator
  - Supervisión de Amazon MemoryDB con Amazon CloudWatch
  - Qué métricas debo monitorear?
  - Supervisión del rendimiento de clústeres de Amazon Redshift
  - Dimensiones y métricas de Timestream
  - Métricas de Amazon CloudWatch para Amazon Aurora
  - Supervisión de Amazon Keyspaces (para Apache Cassandra)

  - Supervisión de recursos de Amazon Neptune
- Use una solución de registro y supervisión aprobada para recopilar estas métricas. Amazon CloudWatch puede recopilar métricas entre los recursos de su arquitectura. También puede recopilar y publicar métricas del cliente para negocios de superficie o métricas derivadas. Utilice CloudWatch o soluciones de terceros para establecer alarmas que indiquen cuándo se superan los umbrales.
- Compruebe si la supervisión del almacén de datos puede beneficiarse de una solución de machine learning que detecte anomalías de rendimiento.
  - Amazon DevOps Guru para Amazon RDS brinda visibilidad sobre los problemas de rendimiento y recomienda acciones correctivas.
- Configure la retención de datos de la solución de supervisión y registro para que se ajuste a sus objetivos operativos y de seguridad.
  - Retención de datos predeterminada para las métricas de CloudWatch
  - Retención de datos predeterminada para Registros de CloudWatch

Recursos

Documentos relacionados:

- AWS Database Caching
- Amazon Athena top 10 performance tips
- Prácticas recomendadas con Amazon Aurora
- DynamoDB Accelerator
- Prácticas recomendadas para Amazon DynamoDB
- Amazon Redshift Spectrum best practices
- Rendimiento de Amazon Redshift
- Bases de datos en la nube con AWS
- Información de rendimiento de Amazon RDS

Videos relacionados:

- AWS re:Invent 2022 - Performance monitoring with Amazon RDS and Aurora, featuring Autodesk
- Database Performance Monitoring and Tuning with Amazon DevOps Guru for Amazon RDS

- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - Dive deep into Amazon DynamoDB
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - Dive deep into Amazon DynamoDB
- Best Practices for Monitoring Redis Workloads on Amazon ElastiCache

Ejemplos relacionados:

- AWS Dataset Ingestion Metrics Collection Framework
- Amazon RDS Monitoring Workshop
- AWS Purpose Built Databases Workshop

PERF03-BP04 Implementación de estrategias para mejorar el rendimiento de las consultas en el almacén de datos

Implemente estrategias que permitan optimizar los datos y mejorar las consultas para aumentar la escalabilidad y conseguir un rendimiento eficiente para su carga de trabajo.

Patrones comunes de uso no recomendados:

- No divide en particiones los datos en su almacén de datos.
- Almacena los datos en un solo formato en su almacén de datos.
- No utiliza índices en su almacén de datos.

Beneficios de establecer esta práctica recomendada: al optimizar el rendimiento de los datos y las consultas, se consigue una mayor eficiencia, una reducción de los costos y una mejor experiencia de usuario.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La optimización de los datos y el ajuste de las consultas son aspectos fundamentales en la eficiencia del rendimiento de un almacén de datos, ya que afectan al rendimiento y a la capacidad de respuesta de toda la carga de trabajo en la nube. Las consultas que no están optimizadas pueden aumentar el

uso de recursos y generar cuellos de botella, lo que reduce la eficiencia general de los almacenes de datos.

La optimización de datos incluye diversas técnicas que garantizan la eficiencia del almacenamiento de datos y su acceso. Esto también ayuda a mejorar el rendimiento de las consultas en un almacén de datos. Algunas de las estrategias clave son la partición, la compresión y la desnormalización de los datos, lo que ayuda a optimizarlos tanto a la hora de almacenarlos como de acceder a ellos.

Pasos para la implementación

- Estudie y analice las consultas de datos críticos que se llevan a cabo en el almacén de datos.
- Identifique las consultas de procesamiento lento del almacén de datos y utilice planes de consulta para conocer su estado actual.
  - Análisis del plan de consulta en Amazon Redshift
  - Uso de EXPLAIN y EXPLAIN ANALYZE en Athena
- Implemente estrategias para mejorar el rendimiento de las consultas. Algunas de las estrategias clave son:
  - Usar un formato de archivo de columnas (como Parquet u ORC).
  - Comprimir los datos en el almacén de datos para reducir el espacio de almacenamiento y la operación de E/S.
  - Crear particiones de datos para dividir la información en partes más pequeñas y reducir el tiempo de análisis de los datos.
    - Partición de datos en Athena
    - Particiones y distribución de datos
  - Indexar los datos de las columnas más frecuentes de la consulta.
  - Utilizar vistas materializadas para consultas frecuentes.
    - Understanding materialized views
    - Creación de vistas materializadas en Amazon Redshift
  - Elegir la operación de unión correcta para la consulta. Cuando una dos tablas, especifique la tabla mayor en el lado izquierdo de la unión y la tabla menor en el lado derecho de la unión.
  - Usar una solución de almacenamiento en caché distribuida para mejorar la latencia y reducir la cantidad de operaciones de E/S de la base de datos.
  - Llevar a cabo un mantenimiento periódico, como vacío, reindexación y ejecución de estadísticas.
- Experimente y pruebe estrategias en un entorno que no sea de producción.

Recursos

Documentos relacionados:

- Prácticas recomendadas de Amazon Aurora
- Rendimiento de Amazon Redshift
- Amazon Athena top 10 performance tips
- AWS Database Caching
- Best Practices for Implementing Amazon ElastiCache
- Partición de datos en Athena

Videos relacionados:

- AWS re:Invent 2023 - AWS storage cost-optimization best practices
- AWS re:Invent 2022 - Performance monitoring with Amazon RDS and Aurora, featuring Autodesk
- Optimize Amazon Athena Queries with New Query Analysis Tools

Ejemplos relacionados:

- AWS Purpose Built Databases Workshop

PERF03-BP05 Implementación de patrones de acceso a datos que utilicen el almacenamiento en caché

Implemente patrones de acceso que puedan beneficiarse del almacenamiento en caché de los datos para lograr una recuperación rápida de los datos a los que se accede con frecuencia.

Patrones comunes de uso no recomendados:

- Almacena en caché datos que cambian con frecuencia.
- Confía en los datos en caché como si estuvieran almacenados de forma duradera y siempre disponibles.
- No tiene en cuenta la coherencia de los datos en caché.
- No supervisa la eficiencia de su implementación de almacenamiento en caché.

Beneficios de establecer esta práctica recomendada: el almacenamiento de datos en una memoria caché puede mejorar la latencia de lectura, el rendimiento de lectura, la experiencia del usuario y la eficiencia general, además de reducir los costos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Una memoria caché es un componente de software o hardware destinado a almacenar datos para que las futuras solicitudes de los mismos se puedan atender de manera más rápida o eficiente. Los datos almacenados en una memoria caché pueden reconstruirse si se pierden mediante la repetición de un cálculo anterior o mediante la recuperación de otro almacén de datos.

El almacenamiento en caché de los datos puede ser una de las estrategias más eficaces para mejorar el rendimiento general de la aplicación y reducir la carga sobre los orígenes de datos principales subyacentes. Los datos se pueden almacenar en caché en varios niveles de la aplicación, por ejemplo, dentro de la aplicación mediante llamadas remotas, lo que se conoce como almacenamiento en caché del cliente, o mediante un servicio secundario rápido para almacenar los datos, conocido como almacenamiento en caché remoto.

Almacenamiento en caché del cliente

Con el almacenamiento en caché del cliente, cada cliente (una aplicación o servicio que consulta el almacén de datos del backend) puede almacenar los resultados de sus consultas únicas de forma local durante un período de tiempo determinado. Esto puede reducir el número de solicitudes a través de la red a un almacén de datos al comprobar primero la memoria caché del cliente local. Si no hay resultados presentes, la aplicación puede consultar el almacén de datos y almacenar esos resultados localmente. Este patrón permite a cada cliente almacenar los datos en la ubicación más cercana posible (el propio cliente), lo que tiene como resultado la latencia más baja posible. Los clientes también pueden seguir atendiendo algunas consultas cuando el almacén de datos del backend no esté disponible, lo que aumenta la disponibilidad de todo el sistema.

Una desventaja de este enfoque es que, cuando hay varios clientes implicados, pueden almacenar los mismos datos en caché localmente, lo que se traduce en un uso duplicado del almacenamiento y en una incoherencia de los datos entre esos clientes. Un cliente puede almacenar en caché los resultados de una consulta y, un minuto después, otro cliente puede ejecutar la misma consulta y obtener un resultado diferente.

Almacenamiento remoto en caché

Para resolver el problema de la duplicación de datos entre clientes, se puede utilizar un servicio externo rápido, o una caché remota, para almacenar los datos consultados. En lugar de comprobar un almacén de datos local, cada cliente comprobará la memoria caché remota antes de consultar el almacén de datos del backend. Esta estrategia facilita respuestas más coherentes entre los clientes, una mayor eficiencia en los datos almacenados y un mayor volumen de datos en caché, ya que el espacio de almacenamiento se escala independientemente de los clientes.

La desventaja de una memoria caché remota es que es posible que todo el sistema tenga una latencia mayor, ya que se requiere un salto de red adicional para comprobar la memoria caché remota. A fin de mejorar la latencia, es posible utilizar el almacenamiento en caché del lado del cliente junto con el almacenamiento en caché remoto para el almacenamiento en caché de varios niveles.

Pasos para la implementación

- Identifique las bases de datos, las API y los servicios de red que podrían beneficiarse del almacenamiento en caché. Los servicios que tienen cargas de trabajo de lectura pesadas, tienen una alta relación de lectura y escritura o son caros de escalar son candidatos para el almacenamiento en caché.
  - Database Caching
  - Habilitación del almacenamiento en caché de la API para mejorar la capacidad de respuesta
- Identifique el tipo de estrategia de almacenamiento en caché adecuada que mejor se adapte a su patrón de acceso.
  - Estrategias de almacenamiento en caché
  - AWS Caching Solutions
- Siga las prácticas recomendadas de almacenamiento en caché para su almacén de datos.
- Configure una estrategia de invalidación de caché, como un tiempo de vida (TTL), para todos los datos que equilibre la actualización de los datos y reduzca la presión sobre el almacén de datos de backend.
- Habilite características como reintentos de conexión automáticos, retroceso exponencial, tiempos de espera del lado del cliente y agrupación de conexiones en el cliente, si están disponibles, ya que pueden mejorar el rendimiento y la fiabilidad.
  - Best practices: Redis clients and Amazon ElastiCache (Redis OSS)
- Supervise la tasa de aciertos de caché con un objetivo del 80 % o superior. Los valores más bajos pueden indicar un tamaño de caché insuficiente o un patrón de acceso que no se beneficia del almacenamiento en caché.

  - Which metrics should I monitor?
  - Best practices for monitoring Redis workloads on Amazon ElastiCache
  - Monitoring best practices with Amazon ElastiCache (Redis OSS) using Amazon CloudWatch
- Implemente la replicación de datos para descargar las lecturas en varias instancias y mejorar el rendimiento y la disponibilidad de la lectura de datos.

Recursos

Documentos relacionados:

- Using the Amazon ElastiCache Well-Architected Lens
- Monitoring best practices with Amazon ElastiCache (Redis OSS) using Amazon CloudWatch
- Qué métricas debo monitorear?
- Documento técnico Performance at Scale with Amazon ElastiCache
- Desafíos y estrategias del almacenamiento en caché

Videos relacionados:

- Amazon ElastiCache Learning Path
- Design for success with Amazon ElastiCache best practices
- AWS re:Invent 2020 - Design for success with Amazon ElastiCache best practices
- AWS re:Invent 2023 - [LAUNCH] Introducing Amazon ElastiCache Serverless
- AWS re:Invent 2022 - 5 great ways to reimagine your data layer with Redis
- AWS re:Invent 2021 - Deep dive on Amazon ElastiCache (Redis OSS)

Ejemplos relacionados:

- Boosting MySQL database performance with Amazon ElastiCache (Redis OSS)

## Redes y entrega de contenido

Preguntas

- PERF 4. ¿Cómo selecciona y configura los recursos de red en su carga de trabajo?

## PERF 4. ¿Cómo selecciona y configura los recursos de red en su carga de trabajo?

La solución de redes óptima para una carga de trabajo varía según los requisitos de latencia, rendimiento, fluctuaciones y ancho de banda. Las limitaciones físicas, como los recursos de usuario o en las instalaciones, determinan las opciones de ubicación. Estas limitaciones pueden compensarse con las ubicaciones periféricas o la ubicación de los recursos.

Prácticas recomendadas

- PERF04-BP01 Comprensión del efecto de las redes en el rendimiento
- PERF04-BP02 Evaluación de las características de las redes disponibles
- PERF04-BP03 Elección de la conectividad o VPN dedicadas adecuadas para la carga de trabajo
- PERF04-BP04 Uso del equilibrio de carga para distribuir el tráfico entre varios recursos
- PERF04-BP05 Elección de los protocolos de red para mejorar el rendimiento
- PERF04-BP06 Elección de la ubicación de la carga de trabajo en función de los requisitos de la red
- PERF04-BP07 Optimización de la configuración de red según las métricas

### PERF04-BP01 Comprensión del efecto de las redes en el rendimiento

Analice y comprenda cómo las decisiones relacionadas con la red afectan a su carga de trabajo para ofrecer un rendimiento eficiente y una mejor experiencia de usuario.

Patrones comunes de uso no recomendados:

- Todo el tráfico fluye a través de sus centros de datos existentes.
- Enruta todo el tráfico a través de firewalls centrales en lugar de utilizar herramientas de seguridad de red nativas en la nube.
- Aprovisiona conexiones de AWS Direct Connect sin comprender los requisitos de uso reales.
- No tiene en cuenta las características de la carga de trabajo ni la sobrecarga de cifrado al definir sus soluciones de redes.
- Utiliza conceptos y estrategias en las instalaciones para las soluciones de redes en la nube.

Beneficios de establecer esta práctica recomendada: comprender el impacto de las redes en el rendimiento de la carga de trabajo lo ayuda a identificar posibles cuellos de botella, mejorar la experiencia del usuario, aumentar la fiabilidad y reducir el mantenimiento operativo a medida que cambia la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La red es responsable de la conectividad entre los componentes de las aplicaciones, los servicios en la nube, las redes periféricas y los datos en las instalaciones, por lo que puede tener un gran impacto en el rendimiento de las cargas de trabajo. Además del rendimiento de la carga de trabajo, la experiencia del usuario también puede verse afectada por la latencia de la red, el ancho de banda, los protocolos, la ubicación, la congestión de la red, las fluctuaciones, el rendimiento y las reglas de enrutamiento.

Disponga de una lista documentada de los requisitos de redes de la carga de trabajo, incluida la latencia, el tamaño de los paquetes, las reglas de enrutamiento, los protocolos y los patrones de tráfico que admiten. Examine las soluciones de red disponibles e identifique qué servicio se ajusta a las características de red de su carga de trabajo. Las redes basadas en la nube se pueden reconstruir rápidamente, de modo que hacer evolucionar su arquitectura de red con el tiempo resulta necesario para mantener la eficiencia del rendimiento.

Pasos para la implementación:

- Defina y documente los requisitos de rendimiento de la red e incluya métricas como la latencia de red, el ancho de banda, los protocolos, las ubicaciones, los patrones de tráfico (picos y frecuencia), el rendimiento, el cifrado, la inspección y las reglas de enrutamiento.
- Obtenga información sobre los principales servicios de red de AWS, como las VPC, AWS Direct Connect, Elastic Load Balancing (ELB) y Amazon Route 53.
- Capture las siguientes características clave de la red:

| Características | Herramientas y métricas |
| --- | --- |
| Características fundamentales de las redes | • Registros de flujo de VPC<br>• Registros de flujo de AWS Transit Gateway<br>• AWS Transit Gateway metrics<br>• AWS PrivateLink metrics |
| Características de las redes de aplicaciones | • Elastic Fabric Adapter<br>• AWS App Mesh metrics<br>• Métricas de Amazon API Gateway |

| Características | Herramientas y métricas |
| --- | --- |
| Características de las redes de periferia | • Métricas de Amazon CloudFront<br>• Amazon Route 53 metrics<br>• AWS Global Accelerator metrics |
| Características de las redes híbridas | • Direct Connect metrics<br>• AWS Site-to-Site VPN metrics<br>• AWS Client VPN metrics<br>• Nube de AWS WAN metrics |
| Características de las redes de seguridad | • AWS Shield, AWS WAF, and AWS Network Firewall metrics |
| Características de rastreo | • AWS X-Ray<br>• VPC Reachability Analyzer<br>• Analizador de acceso a la red<br>• Amazon Inspector<br>• Amazon CloudWatch RUM |

- Comparación y prueba del rendimiento de la red:
  - Lleve a cabo pruebas comparativas del rendimiento de la red, ya que algunos factores pueden afectar al rendimiento de red de Amazon EC2 cuando las instancias están en la misma VPC. Mida el ancho de banda de la red entre las instancias Linux de Amazon EC2 en la misma VPC.
  - Haga pruebas de carga para experimentar con soluciones y opciones de redes.

Recursos

Documentos relacionados:

- Equilibrador de carga de aplicación
- Redes de EC2 mejoradas en Linux
- Redes de EC2 mejoradas en Windows
- Grupos de ubicación de EC2
- Habilitación de las redes mejoradas con Elastic Network Adapter (ENA) en las instancias Linux

- Equilibrador de carga de red de
- Productos de redes con AWS
- Transit Gateway
- Transitioning to latency-based routing in Amazon Route 53
- VPC Endpoints

Videos relacionados:

- AWS re:Invent 2023 - AWS networking foundations
- AWS re:Invent 2023 - What can networking do for your application?
- AWS re:Invent 2023 - Advanced VPC designs and new capabilities
- AWS re:Invent 2023 - A developer's guide to cloud networking
- AWS re:Invent 2019 - Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2019 - Optimizing Network Performance for Amazon EC2 Instances
- AWS Summit Online - Improve Global Network Performance for Applications
- AWS re:Invent 2020 - Networking best practices and tips with the Well-Architected Framework
- AWS re:Invent 2020 - AWS networking best practices in large-scale migrations

Ejemplos relacionados:

- AWS Transit Gateway and Scalable Security Solutions
- Talleres de redes de AWS
- Hands-on Network Firewall Workshop
- Observing and Diagnosing your Network on AWS
- Finding and addressing Network Misconfigurations on AWS

PERF04-BP02 Evaluación de las características de las redes disponibles

Evalúe las características de la red en la nube que pueden aumentar el rendimiento. Mida el impacto de estas características a través de pruebas, métricas y análisis. Por ejemplo, aproveche las características de red que están disponibles para reducir la latencia, la distancia de la red o las fluctuaciones.

Patrones comunes de uso no recomendados:

- Se mantiene dentro de una región porque es allí donde se encuentra físicamente su sede.
- Utiliza firewalls en lugar de grupos de seguridad para filtrar el tráfico.
- Se infringe la TLS para inspeccionar el tráfico en lugar de confiar en grupos de seguridad, políticas de puntos de conexión y otras funciones nativas en la nube.
- Solo utiliza la segmentación basada en subredes en lugar de grupos de seguridad.

Beneficios de establecer esta práctica recomendada: evaluar todas las características y opciones del servicio puede aumentar el rendimiento de su carga de trabajo, disminuir el esfuerzo necesario para mantener su carga de trabajo y aumentar su posición de seguridad general. Puede utilizar la estructura global de AWS para ofrecer una experiencia de red óptima a sus clientes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

AWS ofrece servicios como AWS Global Accelerator y Amazon CloudFront que pueden ayudar a mejorar el rendimiento de la red, mientras que la mayoría de los servicios de AWS incluyen características de producto (como la función Aceleración de transferencias de Amazon S3) para optimizar el tráfico de red.

Revise qué opciones de configuración relacionadas con la red tiene a su disposición y cómo podrían afectar a su carga de trabajo. La optimización del rendimiento depende de comprender cómo interactúan estas opciones con su arquitectura y el impacto que tendrán tanto en el rendimiento medido como en la experiencia del usuario.

Pasos para la implementación

- Cree una lista de componentes de la carga de trabajo.
  - Considere la posibilidad de usar WAN en la Nube de AWS para diseñar, administrar y supervisar la red de su organización al crear una red global unificada.
  - Supervise sus redes globales y principales con las métricas de Registros de Amazon CloudWatch. Use Amazon CloudWatch RUM, que proporciona información para ayudar a identificar, comprender y mejorar la experiencia digital de los usuarios.
  - Consulte la latencia de red agregada entre Regiones de AWS y las zonas de disponibilidad, así como dentro de cada zona de disponibilidad, mediante AWS Network Manager para obtener información sobre la relación entre el rendimiento de su aplicación y el rendimiento de la red de AWS subyacente.

  - Utilice una herramienta de base de datos de administración de la configuración (CMDB) existente o un servicio como AWS Config para crear un inventario de su carga de trabajo y de su configuración.
- Si se trata de una carga de trabajo existente, identifique y documente el punto de referencia para sus métricas de rendimiento, y céntrese en los cuellos de botella y las áreas que debe mejorar. Las métricas de red relacionadas con el rendimiento variarán según la carga de trabajo en función de los requisitos empresariales y las características de la carga de trabajo. Para empezar, podría ser importante revisar estas métricas para su carga de trabajo: ancho de banda, latencia, pérdida de paquetes, fluctuación y retransmisiones.
- Si se trata de una carga de trabajo nueva, ejecute pruebas de carga para identificar los cuellos de botella en el rendimiento.
- Para los cuellos de botella en el rendimiento que identifique, revise las opciones de configuración de sus soluciones para identificar las oportunidades de mejora del rendimiento. Eche un vistazo a las siguientes opciones y características de red clave:

| Oportunidad de mejora | Solución |
| --- | --- |
| Rutas de red | Utilice el Analizador de acceso a la red para identificar rutas o rutas. |
| Protocolos de red | Consulte PERF04-BP05 Elección de los protocolos de red para mejorar el rendimiento |
| Topología de la red | Evalúe las ventajas y desventajas operativa s de usar el emparejamiento de VPC frente a usar AWS Transit Gateway al conectar varias cuentas. AWS Transit Gateway simplifica la forma de interconectar todas sus VPC, que pueden abarcar miles de Cuentas de AWS y sus redes en las instalaciones. Comparta su AWS Transit Gateway entre varias cuentas mediante AWS Resource Access Manager.<br><br>Consulte PERF04-BP03 Elección de la conectividad o VPN dedicadas adecuadas para la carga de trabajo |

| Oportunidad de mejora | Solución |
| --- | --- |
| Servicios de red | AWS Global Accelerator es un servicio de redes que mejora el rendimiento del tráfico de los usuarios hasta un 60 % al utilizar la infraestructura de red global de AWS.<br><br>Amazon CloudFront puede mejorar el rendimiento de la carga de trabajo, la entrega de contenido y la latencia a nivel mundial.<br><br>Use Lambda@Edge para ejecutar funciones que personalicen el contenido que CloudFront ofrece más cerca de los usuarios, reduzcan la latencia y mejoren el rendimiento.<br><br>Amazon Route 53 ofrece opciones de enrutamiento basado en la latencia, enrutamiento de geolocalización, enrutamiento de geoproximidad y enrutamiento basado en IP para ayudarle a mejorar el rendimiento de su carga de trabajo para un público global. Para identificar qué opción de enrutamiento optimizaría el rendimiento de su carga de trabajo, revise su tráfico y la ubicación de los usuarios cuando la carga de trabajo se distribuya globalmente. |

| Oportunidad de mejora | Solución |
| --- | --- |
| Características de los recursos de almacenamiento | La Aceleración de transferencias de Amazon S3 es una característica que permite que los usuarios externos se beneficien de las optimizaciones de redes de CloudFront para cargar datos en Amazon S3. Esto mejora la capacidad de transferir grandes cantidades de datos desde ubicaciones remotas que no tienen conectividad dedicada a la Nube de AWS.<br><br>Los puntos de acceso multirregionales de Amazon S3 replican el contenido en varias regiones y simplifica la carga de trabajo al proporcionar un punto de acceso. Cuando se utiliza un punto de acceso multirregión, se pueden solicitar o escribir datos en Amazon S3 con el servicio que identifica el bucket de menor latencia. |

| Oportunidad de mejora | Solución |
| --- | --- |
| Características de recursos de computación | Las interfaces de red elásticas (ENI) utilizadas por las instancias de Amazon EC2, los contenedores y las funciones de Lambda están limitadas por el flujo. Revise sus grupos de ubicación para optimizar el rendimiento de sus redes de EC2. Para evitar un cuello de botella por cada flujo, diseñe su aplicación para que utilice varios flujos. Para supervisar y obtener visibilidad de las métricas de red relacionadas con la computación, utilice métricas de CloudWatch y ethtool. El comando `ethtool` se incluye en el controlador de ENA y expone métricas adicionales relacionadas con la red que pueden publicarse como una métrica personalizada en CloudWatch.<br><br>Los Amazon Elastic Network Adapters (ENA) entregan una mayor optimización al proporcionar un mayor rendimiento para las instancias de un grupo con ubicación en clúster.<br><br>Elastic Fabric Adapter (EFA) es una interfaz de red para instancias de Amazon EC2 que le permite ejecutar aplicaciones que requieren altos niveles de comunicaciones entre nodos a escala en AWS.<br><br>Las instancias optimizadas para Amazon EBS utilizan una pila de configuración optimizada y ofrecen capacidad dedicada adicional para aumentar las operaciones de E/S de Amazon EBS. |

Recursos

Documentos relacionados:

- Equilibrador de carga de aplicación de
- Redes de EC2 mejoradas en Linux
- Redes de EC2 mejoradas en Windows
- Grupos de ubicación de EC2
- Habilitación de las redes mejoradas con Elastic Network Adapter (ENA) en las instancias Linux
- Equilibrador de carga de red de
- Productos de redes con AWS
- Transitioning to Latency-Based Routing in Amazon Route 53
- VPC Endpoints
- registros de flujo de VPC

Videos relacionados:

- AWS re:Invent 2023 – Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2023 – Advanced VPC designs and new capabilities
- AWS re:Invent 2023 – A developer's guide to cloud networking
- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure
- AWS re:Invent 2019 – Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2018 – Optimizing Network Performance for Amazon EC2 Instances
- AWS Global Accelerator

Ejemplos relacionados:

- AWS Transit Gateway and Scalable Security Solutions
- Talleres de redes de AWS
- Observing and diagnosing your network
- Finding and addressing network misconfigurations on AWS

PERF04-BP03 Elección de la conectividad o VPN dedicadas adecuadas para la carga de trabajo

Cuando se requiera conectividad híbrida para conectar los recursos en las instalaciones y de la nube, aprovisione el ancho de banda adecuado para satisfacer sus requisitos de rendimiento. Calcule los requisitos de ancho de banda y de latencia para la carga de trabajo híbrida. Estas cifras determinarán los requisitos de tamaño.

Patrones comunes de uso no recomendados:

- Solo evalúa las soluciones de VPN para los requisitos de cifrado de su red.
- No evalúa las opciones de conectividad redundante o de respaldo.
- No identifica todos los requisitos de la carga de trabajo (necesidades de cifrado, protocolo, ancho de banda y tráfico).

Beneficios de establecer esta práctica recomendada: la selección y configuración de las soluciones de conectividad adecuadas aumentará la fiabilidad de su carga de trabajo y maximizará el rendimiento. Si identifica los requisitos de la carga de trabajo, planifica con antelación y evalúa las soluciones híbridas, puede minimizar los costosos cambios en la red física y los gastos operativos, a la vez que acelera el tiempo de rentabilización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Desarrolle una arquitectura de red híbrida en función de los requisitos de ancho de banda. Direct Connect le permite conectar su red en las instalaciones de forma privada con AWS. Es conveniente cuando se necesita un gran ancho de banda y baja latencia con un rendimiento uniforme. Una conexión VPN establece una conexión segura a través de Internet. Se usa cuando solo se requiere una conexión temporal, cuando el costo es un factor o como alternativa mientras se espera que se establezca una conectividad de red física resiliente durante el uso de Direct Connect.

Si los requisitos de ancho de banda son elevados, podría considerar la posibilidad de utilizar varios servicios de Direct Connect o VPN. Es posible equilibrar la carga del tráfico entre los servicios, aunque no recomendamos equilibrar la carga entre Direct Connect y una VPN debido a las diferencias de latencia y ancho de banda.

Pasos para la implementación

- Calcule los requisitos de ancho de banda y de latencia de sus aplicaciones actuales.

    - En el caso de cargas de trabajo existentes que se trasladan a AWS, utilice los datos de sus sistemas internos de supervisión de red.
    - En el caso de cargas de trabajo nuevas o existentes para las que no disponga de datos de supervisión, consulte con los propietarios del producto para determinar las métricas de rendimiento adecuadas y ofrecer una buena experiencia de usuario.
- Seleccione una conexión dedicada o VPN como opción de conectividad. En función de todos los requisitos de la carga de trabajo (necesidades de cifrado, ancho de banda y tráfico), puede elegir AWS Direct Connect o Site-to-Site VPN (o ambas). El siguiente diagrama puede ayudarle a elegir el tipo de conexión adecuado.
    - AWS Direct Connect ofrece conectividad dedicada al entorno de AWS, desde 50 Mbps hasta 100 Gbps, mediante conexiones dedicadas o conexiones alojadas. Esto le ofrece un ancho de banda aprovisionado y una latencia administrada y controlada, a fin de que su carga de trabajo pueda conectarse de manera eficiente a otros entornos. Mediante el uso de socios de AWS Direct Connect, puede disponer de conectividad de extremo a extremo desde varios entornos, lo que proporciona una red ampliada con un rendimiento coherente. AWS ofrece un ancho de banda de conexión directa escalable mediante 100 Gbps nativos, un grupo de agregación de enlaces (LAG) o varias rutas de igual costo (ECMP) con BGP.
    - AWS Site-to-Site VPN proporciona un servicio de VPN administrado compatible con la seguridad del protocolo de Internet (IPsec). Cuando se crea una conexión VPN, cada conexión VPN incluye dos túneles para ofrecer una alta disponibilidad.
- Siga la documentación de AWS para elegir la opción de conectividad adecuada:
    - Si decide usar Direct Connect, seleccione el ancho de banda adecuado para su conectividad.
    - Si utiliza una conexión de AWS Site-to-Site VPN en varias ubicaciones para conectarse a una Región de AWS, utilice una conexión de VPN Site-to-Site acelerada para tener la oportunidad de mejorar el rendimiento de la red.
    - Si el diseño de su red consta de una conexión VPN IPsec a través de AWS Direct Connect, considere la posibilidad de utilizar una VPN con IP privada para mejorar la seguridad y lograr la segmentación. AWS La VPN con IP privada de Site-to-Site se implementa sobre la interfaz virtual (VIF) de tránsito.
    - AWS Direct Connect SiteLink permite crear conexiones redundantes y de baja latencia entre sus centros de datos de todo el mundo mediante el envío de datos a través de la ruta más corta entre las ubicaciones de AWS Direct Connect, pasando por alto Regiones de AWS.

- Valide la configuración de la conectividad antes de la implementación en producción. Lleve a cabo pruebas de seguridad y rendimiento para asegurarse de que cumpla los requisitos de ancho de banda, fiabilidad, latencia y cumplimiento.
- Supervise periódicamente el rendimiento y el uso de la conectividad y optimícelo si es necesario.

Diagrama de flujo de rendimiento determinístico

Recursos

Documentos relacionados:

- Productos de redes con AWS
- AWS Transit Gateway
- VPC Endpoints
- Creación de una infraestructura de red de AWS multiVPC escalable y segura
- Client VPN

Videos relacionados:

- AWS re:Invent 2023 – Building hybrid network connectivity with AWS
- AWS re:Invent 2023 – Secure remote connectivity to AWS
- AWS re:Invent 2022 – Optimizing performance with Amazon CloudFront
- AWS re:Invent 2019 – Connectivity to AWS and hybrid AWS network architectures
- AWS re:Invent 2020 – AWS Transit Gateway Connect

Ejemplos relacionados:

- AWS Transit Gateway and Scalable Security Solutions
- AWS Talleres de redes de

PERF04-BP04 Uso del equilibrio de carga para distribuir el tráfico entre varios recursos

Distribuya el tráfico entre varios recursos o servicios para que su carga de trabajo aproveche la elasticidad que ofrece la nube. También puede utilizar el equilibrio de carga para descargar la terminación del cifrado con el objetivo de mejorar el rendimiento, la fiabilidad y administrar y dirigir el tráfico de manera eficaz.

Patrones comunes de uso no recomendados:

- No tiene en cuenta los requisitos de la carga de trabajo al elegir el tipo de equilibrador de carga.
- No aprovecha las características del equilibrador de carga para optimizar el rendimiento.
- La carga de trabajo se expone directamente a Internet sin un equilibrador de carga.
- Enruta todo el tráfico de Internet a través de los equilibradores de carga existentes.
- Utiliza el equilibrio de carga TCP genérico y hace que cada nodo de computación gestione el cifrado SSL.

Beneficios de establecer esta práctica recomendada: un equilibrador de carga gestiona la carga variable del tráfico de la aplicación en una única zona de disponibilidad o en varias zonas de disponibilidad y facilita una alta disponibilidad, un escalado automático y un mejor uso de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los equilibradores de carga actúan como punto de entrada de la carga de trabajo y, a partir de ahí, distribuyen el tráfico a los destinos de backend, como instancias de computación o contenedores, para mejorar el uso.

La elección del tipo de equilibrador de carga adecuado es el primer paso para optimizar su arquitectura. Comience por enumerar las características de su carga de trabajo, como el protocolo (por ejemplo, TCP, HTTP, TLS o WebSockets), el tipo de destino (como instancias, contenedores o sin servidor), los requisitos de la aplicación (como conexiones de larga duración, autenticación de usuarios o permanencia) y la ubicación (como región, zona local, Outpost o aislamiento de zona).

AWS proporciona varios modelos para que sus aplicaciones utilicen el equilibrio de carga. El equilibrador de carga de aplicación es el más adecuado para el equilibrio de carga del tráfico de HTTP y HTTPS y entrega un direccionamiento de solicitudes avanzado enfocado a la entrega de arquitecturas de aplicaciones modernas, incluidos los microservicios y los contenedores.

El equilibrador de carga de red es el más adecuado para el equilibrio de carga del tráfico de TCP donde se necesite un rendimiento extremo. Es capaz de gestionar millones de solicitudes por segundo a la vez que mantiene latencias ultrabajas y está optimizado para manejar patrones de tráfico repentinos y volátiles.

Elastic Load Balancing proporciona administración de certificados y descifrado SSL/TLS integrados, lo que le permite la flexibilidad de administrar de forma centralizada la configuración SSL del equilibrador de carga y descargar el trabajo intensivo de la CPU de su carga de trabajo.

Una vez elegido el equilibrador de carga adecuado, puede empezar a utilizar sus características para reducir el esfuerzo que debe hacer su backend para atender al tráfico.

Por ejemplo, al utilizar tanto el equilibrador de carga de aplicación (ALB) como el equilibrador de carga de red (NLB), puede llevar a cabo la descarga de cifrado SSL/TLS, lo que da la oportunidad de evitar que sus destinos completen el establecimiento de comunicación TLS, que consume mucha CPU, y también para mejorar la administración de certificados.

Cuando configura la descarga SSL/TLS en el equilibrador de carga, este se ocupa del cifrado del tráfico desde y hacia los clientes, al tiempo que entrega el tráfico sin cifrar a sus backends, lo que libera recursos de backend y mejora el tiempo de respuesta para los clientes.

El equilibrador de carga de aplicación también puede atender el tráfico HTTP/2 sin necesidad de soporte en sus destinos. Esta simple decisión puede mejorar el tiempo de respuesta de su aplicación, ya que HTTP/2 utiliza las conexiones TCP de forma más eficiente.

Los requisitos de latencia de la carga de trabajo deben tenerse en cuenta a la hora de definir la arquitectura. Por ejemplo, si tiene una aplicación sensible a la latencia, puede decidir utilizar el equilibrador de carga de red, que ofrece latencias extremadamente bajas. Como alternativa, puede decidir acercar su carga de trabajo a sus clientes con el equilibrador de carga de aplicación en las zonas locales de AWS o incluso AWS Outposts.

Otra consideración para las cargas de trabajo sensibles a la latencia es el equilibrio de carga entre zonas. Con el equilibrio de carga entre zonas, cada nodo del equilibrador de carga distribuye el tráfico entre los destinos registrados en todas las zonas de disponibilidad permitidas.

Utilice el escalado automático integrado con su equilibrador de carga. Uno de los aspectos clave de un sistema con un rendimiento eficiente tiene que ver con el redimensionamiento correcto de sus recursos de backend. Para ello, puede utilizar las integraciones del equilibrador de carga para los recursos de destino de backend. Mediante la integración del equilibrador de carga con los grupos

de escalado automático, los destinos se agregarán o eliminarán del equilibrador de carga según sea necesario y en respuesta al tráfico entrante. Los equilibradores de carga también pueden integrarse con Amazon ECS y Amazon EKS para cargas de trabajo en contenedores.

- Amazon ECS: equilibrador de carga del servicio
- Equilibrador de carga de aplicaciones en Amazon EKS
- Equilibrio de carga de red en Amazon EKS

Pasos para la implementación

- Defina sus requisitos de equilibrio de carga, incluidos el volumen de tráfico, la disponibilidad y la escalabilidad de las aplicaciones.
- Elija el tipo de equilibrador de carga adecuado para su aplicación.
  - Use el equilibrador de carga de aplicación para cargas de trabajo HTTP/HTTPS.
  - Utilice el equilibrador de carga de red para cargas de trabajo distintas de HTTP que se ejecuten en TCP o UDP.
  - Utilice una combinación de ambos (ALB como objetivo de NLB) si desea aprovechar las características de ambos productos. Por ejemplo, puede hacerlo si desea utilizar las IP estáticas del equilibrador de carga de red junto con el enrutamiento basado en encabezado HTTP del equilibrador de carga de aplicación, o si desea exponer su carga de trabajo HTTP a un AWS PrivateLink.
  - Para obtener una comparación completa de los equilibradores de carga, consulte la comparación de productos de ELB.
- Utilice la descarga SSL/TLS si es posible.
  - Configure los oyentes HTTPS/TLS con un equilibrador de carga de aplicación y un equilibrador de carga de red integrados con AWS Gestor de certificados.
  - Tenga en cuenta que algunas cargas de trabajo pueden requerir cifrado de extremo a extremo por motivos de conformidad. En este caso, es un requisito permitir el cifrado en los destinos.
  - Para conocer las prácticas recomendadas de seguridad, consulte SEC09-BP02 Aplicación del cifrado en tránsito.
- Seleccione el algoritmo de enrutamiento adecuado (solo para el ALB).
  - El algoritmo de enrutamiento puede marcar la diferencia en el grado de utilización de sus destinos de backend y, por lo tanto, en su repercusión en el rendimiento. Por ejemplo, el equilibrador de carga de aplicación ofrece dos opciones para los algoritmos de enrutamiento:

    - Solicitudes menos pendientes: utilícelas para lograr una mejor distribución de la carga a sus destinos de backend para los casos en que las solicitudes de la aplicación varíen en complejidad o los destinos varíen en capacidad de procesamiento.
    - Patrón rotativo: utilícelo cuando las solicitudes y los destinos sean similares, o si necesita distribuir las solicitudes equitativamente entre los destinos.
- Considere el aislamiento entre zonas o de zonas.
    - Desactive el aislamiento entre zonas (aislamiento de zonas) para mejorar la latencia y los dominios de error de zona. Está desactivado de forma predeterminada en el equilibrador de carga de red y en el equilibrador de carga de aplicación lo puede desactivar por grupo de destino.
    - Active el aislamiento entre zonas para aumentar la disponibilidad y flexibilidad. Está activado de forma predeterminada para el equilibrador de carga de aplicación y en el equilibrador de carga de red lo puede activar por grupo de destino.
- Active la conexión persistente HTTP para sus cargas de trabajo HTTP (solo ALB). Con esta característica, el equilibrador de carga puede reutilizar las conexiones de backend hasta que expire el tiempo de espera activo, lo que mejora el tiempo de solicitud y respuesta HTTP, además de reducir la utilización de recursos en los destinos de backend. Para obtener más información sobre cómo hacer esto para Apache y Nginx, consulte ¿Cuál es la configuración óptima para usar Apache o NGINX como servidor de backend para ELB?.
- Active la supervisión de su equilibrador de carga.
    - Active los registros de acceso del equilibrador de carga de aplicación y el equilibrador de carga de red.
    - Los principales campos a tener en cuenta para el equilibrador de carga de aplicación son `request_processing_time`, `request_processing_time` y `response_processing_time`.
    - Los principales campos a tener en cuenta para el equilibrador de carga de red son `connection_time` y `tls_handshake_time`.
    - Esté preparado para consultar los registros cuando los necesite. Puede usar Amazon Athena para consultar tanto los registros del equilibrador de carga de aplicación como los registros del equilibrador de carga de red.
    - Cree alarmas para las métricas relacionadas con el rendimiento, como `TargetResponseTime` para el ALB.

Recursos

Documentos relacionados:

- Comparación de productos de Elastic Load Balancing
- Infraestructura global de AWS
- Improving Performance and Reducing Cost Using Availability Zone Affinity
- Step by step for Log Analysis with Amazon Athena
- Consulta de los registros del Equilibrador de carga de aplicación
- Monitor your Application Load Balancers
- Monitor your Network Load Balancer
- Use Elastic Load Balancing to distribute traffic across the instances in your Auto Scaling group

Videos relacionados:

- AWS re:Invent 2023: What can networking do for your application?
- AWS re:Inforce 20: How to use Elastic Load Balancing to enhance your security posture at scale
- AWS re:Invent 2018: Elastic Load Balancing: Deep Dive and Best Practices
- AWS re:Invent 2021 - How to choose the right load balancer for your AWS workloads
- AWS re:Invent 2019: Get the most from Elastic Load Balancing for different workloads

Ejemplos relacionados:

- Gateway Load Balancer
- CDK and CloudFormation samples for Log Analysis with Amazon Athena

PERF04-BP05 Elección de los protocolos de red para mejorar el rendimiento

Tome decisiones sobre los protocolos de comunicación entre sistemas y redes en función del impacto en el rendimiento de la carga de trabajo.

Existe una relación entre la latencia y el ancho de banda para lograr el rendimiento. Si la transferencia de archivos utiliza el protocolo de control de transmisión (TCP), las latencias más altas probablemente reducirán el rendimiento general. Existen enfoques para solucionar esto con el

ajuste de TCP y protocolos de transferencia optimizados, pero una solución es utilizar el protocolo de datagramas de usuario (UDP).

Patrones comunes de uso no recomendados:

- Utiliza TCP para todas las cargas de trabajo, independientemente de los requisitos de rendimiento.

Beneficios de establecer esta práctica recomendada: verificar que se utiliza un protocolo adecuado para la comunicación entre los usuarios y los componentes de la carga de trabajo ayuda a mejorar la experiencia general del usuario para sus aplicaciones. Por ejemplo, UDP sin conexión permite una alta velocidad, pero no ofrece retransmisión ni alta fiabilidad. TCP es un protocolo con todas las características, pero requiere una mayor sobrecarga para procesar los paquetes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si tiene la capacidad de elegir diferentes protocolos para su aplicación y tiene experiencia en esta área, optimice la aplicación y la experiencia del usuario final mediante un protocolo diferente. Tenga en cuenta que este enfoque presenta una dificultad significativa y solo debe intentarse si primero ha optimizado su aplicación de otras maneras.

Una consideración primordial para mejorar el rendimiento de la carga de trabajo es comprender los requisitos de latencia y rendimiento, y luego elegir protocolos de red que optimicen el rendimiento.

Cuándo considerar el uso de TCP

TCP proporciona una entrega de datos fiable, y se puede utilizar para la comunicación entre los componentes de la carga de trabajo cuando la fiabilidad y la entrega garantizada de datos es importante. Muchas aplicaciones basadas en web dependen de protocolos basados en TCP, como HTTP y HTTPS, con el fin de abrir sockets TCP para la comunicación entre componentes de la aplicación. La transferencia de datos de correo electrónico y archivos son aplicaciones habituales que también utilizan TCP, ya que es un mecanismo de transferencia sencillo y fiable entre los componentes de la aplicación. El uso de TLS con TCP puede agregar cierta sobrecarga a la comunicación, lo que puede provocar un aumento de la latencia y una reducción del rendimiento, pero tiene la ventaja de seguridad. La sobrecarga proviene principalmente de la sobrecarga agregada del proceso de establecimiento de comunicación, que puede tardar varias idas y vueltas en completarse. Una vez completado el proceso, la sobrecarga de cifrado y descifrado de datos es relativamente pequeña.

Cuándo considerar el uso de UDP

UDP es un protocolo sin conexión y, por tanto, adecuado para aplicaciones que necesitan una transmisión rápida y eficiente, como datos de registro, supervisión y VoIP. Además, considere el uso de UDP si tiene componentes de carga de trabajo que responden a pequeñas consultas de un gran número de clientes, a fin de garantizar un rendimiento óptimo de la carga de trabajo. La seguridad de la capa de transporte de datagramas (DTLS) es el equivalente UDP de la seguridad de la capa de transporte (TLS). Cuando se utiliza DTLS con UDP, la sobrecarga proviene del cifrado y descifrado de los datos, ya que el proceso de establecimiento de comunicación se simplifica. DTLS también agrega una pequeña cantidad de sobrecarga a los paquetes UDP, ya que incluye campos adicionales para indicar los parámetros de seguridad y detectar manipulaciones.

Cuándo considerar el uso de SRD

Scalable reliable datagram (SRD) es un protocolo de transporte de red optimizado para cargas de trabajo de alto rendimiento debido a su capacidad para equilibrar la carga de tráfico a través de numerosas rutas y recuperarse rápidamente de las caídas de paquetes o errores de enlace. Por lo tanto, es mejor utilizar SRD para cargas de trabajo de computación de alto rendimiento (HPC) que exigen un alto rendimiento y una comunicación de baja latencia entre nodos de computación. Puede que esto incluya tareas de procesamiento paralelo como simulación, modelado y análisis de datos que impliquen una gran cantidad de transferencia de datos entre nodos.

Pasos para la implementación

- Use los servicios de AWS Global Accelerator y AWS Transfer Family para mejorar el rendimiento de sus aplicaciones de transferencia de archivos en línea. El servicio AWS Global Accelerator le ayuda a conseguir una latencia menor entre sus dispositivos cliente y su carga de trabajo en AWS. Con AWS Transfer Family, puede utilizar protocolos basados en TCP como el protocolo de transferencia de archivos de shell seguro (SFTP) y el protocolo de transferencia de archivos sobre SSL (FTPS) para escalar y administrar de forma segura las transferencias de archivos a los servicios de almacenamiento de AWS.
- Utilice la latencia de la red para determinar si TCP es adecuado para la comunicación entre los componentes de la carga de trabajo. Si la latencia de la red entre la aplicación cliente y el servidor es alta, la comunicación TCP de tres vías puede tardar un tiempo, lo que afectará a la capacidad de respuesta de la aplicación. Para medir la latencia de la red pueden utilizarse métricas, como el tiempo hasta el primer byte (TTFB) y el tiempo de ida y vuelta (RTT). Si la carga de trabajo sirve contenido dinámico a los usuarios, considere la posibilidad de usar Amazon CloudFront, que

establece una conexión persistente con cada origen para el contenido dinámico para eliminar el tiempo de configuración de la conexión que, de otro modo, ralentizaría cada solicitud del cliente.

- El uso de TLS con TCP o UDP puede aumentar la latencia y reducir el rendimiento de la carga de trabajo debido al impacto del cifrado y el descifrado. Para estas cargas de trabajo, considere la posibilidad de usar la descarga de SSL/TLS en Elastic Load Balancing para mejorar el rendimiento de la carga de trabajo al permitir que el equilibrador de carga gestione el proceso de cifrado y descifrado SSL/TLS, en lugar de que lo hagan las instancias de backend. Esto puede ayudar a reducir el uso de la CPU en las instancias backend, lo que puede mejorar el rendimiento y aumentar la capacidad.
- Use el equilibrador de carga de red (NBT) para implementar servicios que dependan del protocolo UDP, como autenticación y autorización, registro, DNS, IoT y streaming multimedia, para mejorar el rendimiento y la fiabilidad de su carga de trabajo. El equilibrador de carga de red distribuye el tráfico UDP entrante entre varios destinos, lo que le permite escalar su carga de trabajo horizontalmente, aumentar la capacidad y reducir la sobrecarga de un único destino.
- Para sus cargas de trabajo de computación de alto rendimiento (HPC), considere la posibilidad de utilizar la funcionalidad Elastic Network Adapter (ENA) Express, que utiliza el protocolo SRD para mejorar el rendimiento de la red al proporcionar un ancho de banda de flujo único más alto (25 Gbps) y una latencia de cola más baja (percentil 99,9) para el tráfico de red entre las instancias de EC2.
- Utilice el equilibrador de carga de aplicación (ALB) para enrutar y equilibrar la carga del tráfico gRPC (llamadas a procedimientos remotos) entre componentes de carga de trabajo o entre clientes y servicios gRPC. gRPC utiliza el protocolo HTTP/2 basado en TCP para el transporte y proporciona ventajas de rendimiento como una huella de red más ligera, compresión, serialización binaria eficiente, compatibilidad con numerosos idiomas y streaming bidireccional.

Recursos

Documentos relacionados:

- How to route UDP traffic into Kubernetes
- Equilibrador de carga de aplicación
- Redes de EC2 mejoradas en Linux
- Redes de EC2 mejoradas en Windows
- Grupos de ubicación de EC
- Habilitación de las redes mejoradas con Elastic Network Adapter (ENA) en las instancias Linux

- Equilibrador de carga de red de
- Productos de redes con AWS
- Transitioning to Latency-Based Routing in Amazon Route 53
- VPC Endpoints

Videos relacionados:

- AWS re:Invent 2022 – Scaling network performance on next-gen Amazon Elastic Compute Cloud instances
- AWS re:Invent 2022 – Application networking foundations

Ejemplos relacionados:

- AWS Transit Gateway and Scalable Security Solutions
- Talleres de redes de AWS

PERF04-BP06 Elección de la ubicación de la carga de trabajo en función de los requisitos de la red

Evalúe las opciones de colocación de recursos para reducir la latencia de la red y mejorar el rendimiento, lo que proporcionará una experiencia de usuario óptima al reducir los tiempos de carga de las páginas y de transferencia de datos.

Patrones comunes de uso no recomendados:

- Consolida todos los recursos de la carga de trabajo en una ubicación geográfica.
- Ha elegido la región más cercana a su ubicación, pero no al usuario final de la carga de trabajo.

Beneficios de establecer esta práctica recomendada: la experiencia del usuario se ve muy afectada por la latencia entre el usuario y la aplicación. Al utilizar las Regiones de AWS adecuadas y la red global privada de AWS, puede reducir la latencia y ofrecer una mejor experiencia a los usuarios remotos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los recursos, como las instancias de Amazon EC2, se colocan en zonas de disponibilidad dentro de Regiones de AWS, zonas locales de AWS, AWS Outposts o zonas de AWS Wavelength. La selección de esta ubicación influye en la latencia y el rendimiento de la red desde una ubicación de usuario. Los servicios de periferia como Amazon CloudFront y AWS Global Accelerator también se pueden utilizar para mejorar el rendimiento de la red al almacenar contenido en caché en ubicaciones periféricas o proporcionar a los usuarios una ruta óptima a la carga de trabajo a través de la red global de AWS.

Amazon EC2 ofrece grupos de ubicación para la creación de redes. Un grupo de ubicación es una agrupación lógica de instancias para reducir la latencia. El uso de grupos de ubicación con tipos de instancias compatibles y un Elastic Network Adapter (ENA) permite que las cargas de trabajo participen en una red de 25 Gbps de baja latencia y fluctuación reducida. Se recomiendan grupos de colocación para cargas de trabajo que aprovechan la baja latencia de red, el alto rendimiento de red o ambos.

Los servicios sensibles a la latencia se prestan en ubicaciones periféricas mediante una red global de AWS, como Amazon CloudFront. Estas ubicaciones periféricas normalmente prestan servicios como red de entrega de contenido (CDN) y sistema de nombres de dominio (DNS). Al tener estos servicios en la periferia, las cargas de trabajo pueden responder con baja latencia a las solicitudes de contenido o de resolución de DNS. Estos servicios pueden ofrecer servicios geográficos como la geolocalización del contenido (que proporciona contenido diferente según la ubicación de los usuarios finales) o el enrutamiento basado en la latencia para dirigir a los usuarios finales hacia la región más cercana (latencia mínima).

Utilice los servicios de periferia para reducir la latencia y permitir el almacenamiento en caché del contenido. Configure correctamente el control de caché para DNS y HTTP/HTTPS a fin de obtener el mayor beneficio de estos enfoques.

Pasos para la implementación

- Capture información sobre el tráfico IP que entra y sale de las interfaces de red.
  - Registro del tráfico de IP con registros de flujo de la VPC
  - How the client IP address is preserved in AWS Global Accelerator
- Analice los patrones de acceso a la red en su carga de trabajo para identificar cómo utilizan los usuarios su aplicación.

  - Utilice herramientas de supervisión, como Amazon CloudWatch y AWS CloudTrail, para recopilar datos sobre las actividades de la red.
  - Analice los datos para identificar el patrón de acceso a la red.
- Seleccione las regiones para la implementación de la carga de trabajo en función de los siguientes elementos clave:
  - Ubicación de los datos: en el caso de las aplicaciones con gran cantidad de datos (como macrodatos y machine learning), el código de la aplicación debe ejecutarse lo más cerca posible de los datos.
  - Ubicación de los usuarios: para las aplicaciones orientadas al usuario, elija una región (o regiones) cercana a los usuarios de su carga de trabajo.
  - Otras restricciones: tenga en cuenta las limitaciones, como el costo y el cumplimiento, tal y como se explica en What to Consider when Selecting a Region for your Workloads.
- Use Zonas locales de AWS para ejecutar cargas de trabajo como la renderización de video. Las zonas locales le permiten beneficiarse de tener recursos de computación y almacenamiento más cerca de los usuarios finales.
- Utilice AWS Outposts para cargas de trabajo que deban seguir en las instalaciones y en las que desee que esa carga de trabajo se ejecute sin problemas con el resto de sus demás cargas de trabajo en AWS.
- Aplicaciones como la transmisión de vídeo en directo de alta resolución, audio de alta fidelidad y realidad aumentada/realidad virtual (RA/RV) requieren una latencia ultrabaja para dispositivos 5G. Para estas aplicaciones, considere la posibilidad de usar AWS Wavelength. AWS Wavelength integra los servicios de computación y almacenamiento de AWS en las redes 5G, proporcionando una infraestructura de computación periférica móvil para desarrollar, implementar y escalar aplicaciones de latencia ultrabaja.
- Utilice almacenamiento en caché local o soluciones de almacenamiento en caché de AWS para los activos de uso frecuente con el fin de mejorar el rendimiento, reducir el movimiento de datos y disminuir el impacto medioambiental.

| Servicio | Cuándo se debe usar |
| --- | --- |
| Amazon CloudFront | Se usa para almacenar en caché el contenido estático como imágenes, scripts y videos, así como el contenido dinámico como respuestas de API y aplicaciones web. |

| Servicio | Cuándo se debe usar |
| --- | --- |
| Amazon ElastiCache | Se usa para almacenar en caché el contenido de las aplicaciones web. |
| DynamoDB Accelerator | Se usa para agregar aceleración en memoria a sus tablas de DynamoDB. |

- Utilice servicios que puedan ayudarle a ejecutar el código más cerca de los usuarios de su carga de trabajo, como los siguientes:

| Servicio | Cuándo se debe usar |
| --- | --- |
| Lambda@Edge | Se usa para las operaciones que utilizan muchos recursos de computación que se inician cuando los objetos no están en la memoria caché. |
| Amazon CloudFront Functions | Se usan para casos de uso sencillos como las manipulaciones de solicitudes o respuesta s HTTP(s) que pueden iniciarse mediante funciones de corta duración. |
| AWS IoT Greengrass | Se usa para ejecutar la computación local, la mensajería y el almacenamiento en caché de datos para los dispositivos conectados. |

- Algunas aplicaciones requieren puntos de entrada fijos o un mayor rendimiento mediante el aumento del rendimiento y la reducción de la fluctuación y de la latencia del primer byte. Esta aplicaciones pueden aprovechar servicios de redes que proporcionen direcciones IP estáticas de anycast y la terminación de TCP en las ubicaciones periféricas. AWS Global Accelerator puede mejorar el rendimiento de las aplicaciones hasta en un 60 % y proporcionar una rápida conmutación por error para arquitecturas multirregionales. AWS Global Accelerator le proporciona direcciones IP estáticas de difusión por proximidad que sirven como punto de entrada fijo para las aplicaciones alojadas en una o más Regiones de AWS. Estas direcciones IP permiten que el tráfico entre en la red global de AWS lo más cerca posible de sus usuarios. AWS Global Accelerator reduce el tiempo de configuración de la conexión inicial al establecer una conexión TCP entre el cliente y la ubicación periférica de AWS más cercana al cliente. Revise el uso

de AWS Global Accelerator para mejorar el rendimiento de sus cargas de trabajo TCP/UDP y proporcionar una rápida conmutación por error para arquitecturas multirregión.

Recursos

Prácticas recomendadas relacionadas:

- COST07-BP02 Implementación de regiones según los costos
- COST08-BP03 Implementación de servicios para reducir los costos de transferencia de datos
- REL10-BP01 Implementación de la carga de trabajo en varias ubicaciones
- REL10-BP02 Selección de las ubicaciones adecuadas para la implementación en varias ubicaciones
- SUS01-BP01 Selección de la región en función de los requisitos empresariales y los objetivos de sostenibilidad
- SUS02-BP04 Optimización de la ubicación geográfica de las cargas de trabajo en función de sus requisitos de red
- SUS04-BP07 Minimización del movimiento de datos entre redes

Documentos relacionados:

- Infraestructura global de AWS
- AWS Local Zones and AWS Outposts, choosing the right technology for your edge workload
- Grupos de ubicación
- Zonas locales de AWS
- AWS Outposts
- AWS Wavelength
- Amazon CloudFront
- AWS Global Accelerator
- AWS Direct Connect
- AWS Site-to-Site VPN
- Amazon Route 53

Videos relacionados:

- AWS Local Zones Explainer Video
- AWS Outposts: Overview and How it Works
- AWS re:Invent 2023 - A migration strategy for edge and on-premises workloads
- AWS re:Invent 2021 - AWS Outposts: Bringing the AWS experience on premises
- AWS re:Invent 2020: AWS Wavelength: Run apps with ultra-low latency at 5G edge
- AWS re:Invent 2022 - AWS Local Zones: Building applications for a distributed edge
- AWS re:Invent 2021 - Building low-latency websites with Amazon CloudFront
- AWS re:Invent 2022 - Improve performance and availability with AWS Global Accelerator
- AWS re:Invent 2022 - Build your global wide area network using AWS
- AWS re:Invent 2020: Global traffic management with Amazon Route 53

Ejemplos relacionados:

- AWS Global Accelerator Custom Routing Workshop
- Handling Rewrites and Redirects using Edge Functions

PERF04-BP07 Optimización de la configuración de red según las métricas

Utilice los datos recogidos y analizados para tomar decisiones informadas sobre la optimización de la configuración de su red.

Patrones comunes de uso no recomendados:

- Supone que todos los problemas de rendimiento están relacionados con las aplicaciones.
- Solo hace pruebas del rendimiento de la red desde una ubicación cercana al punto de implementación de la carga de trabajo.
- Se utilizan configuraciones predeterminadas para todos los servicios de red.
- Se sobreaprovisionan los recursos de red para proporcionar capacidad suficiente.

Beneficios de establecer esta práctica recomendada: la recopilación de las métricas necesarias de su red de AWS y la implementación de herramientas de supervisión de red le permiten comprender el rendimiento y optimizar las configuraciones de la misma.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

La supervisión del tráfico hacia y desde VPC, subredes o interfaces de red es crucial para comprender cómo utilizar los recursos de red de AWS y optimizar las configuraciones de la red. Con las siguientes herramientas de la red de AWS, puede inspeccionar más a fondo la información sobre el uso del tráfico, el acceso a la red y los registros.

Pasos para la implementación

- Identifique las métricas clave de rendimiento, como la latencia o la pérdida de paquetes, que desee recopilar. AWS proporciona varias herramientas que pueden ayudarle a recopilar dichas métricas. Mediante las siguientes herramientas, puede inspeccionar más a fondo la información sobre el uso del tráfico, el acceso a la red y los registros:

| AWSHerramienta | Dónde se usa |
| --- | --- |
| Administrador de direcciones IP de Amazon VPC. | Utilice IPAM para planificar, seguir y supervisa r las direcciones IP para sus cargas de trabajo de AWS y en las instalaciones. Esta es una práctica recomendada para optimizar el uso y la asignación de direcciones IP. |
| Registros de flujo de VPC | Utilice los registros de flujo de la VPC para capturar información detallada sobre el tráfico hacia y desde las interfaces de red en sus VPC. Con los registros de flujo de la VPC, puede diagnosticar reglas de grupos de seguridad excesivamente restrictivas o permisivas y determinar la dirección del tráfico hacia y desde las interfaces de red. |
| Registros de flujo de AWS Transit Gateway | Utilice los registros de flujo AWS Transit Gateway para recoger información sobre el tráfico IP que entra y sale de sus instancias de Transit Gateway. |
| Registro de consultas de DNS | Registre información sobre las consultas de DNS públicas o privadas que recibe Route 53. Con los registros de DNS, puede optimizar las |

| AWSHerramienta | Dónde se usa |
| --- | --- |
| | configuraciones de DNS al conocer el dominio o subdominio que se solicitó o las ubicacion es periféricas de Route 53 que respondieron a las consultas de DNS. |
| Analizador de accesibilidad | El Analizador de accesibilidad le ayuda a analizar y depurar la accesibilidad de la red. El Analizador de accesibilidad es una herramien ta de análisis de configuración que le permite hacer pruebas de conectividad entre un recurso de origen y uno de destino en las VPC. Esta herramienta le ayuda a verificar que la configuración de su red coincida con la conectividad prevista. |
| Analizador de acceso a la red | Puede utilizar el Analizador de acceso a la red para comprobar el acceso de la red a los recursos. Puede utilizar el Analizador de acceso a la red para especificar los requisitos de acceso a la red e identificar posibles rutas de red que no cumplan los requisitos especific ados. Al optimizar su configuración de red correspondiente, puede comprender y verificar el estado de su red y demostrar si su red en AWS cumple con sus requisitos de conformid ad. |

| AWSHerramienta | Dónde se usa |
| --- | --- |
| Amazon CloudWatch | Utilice Amazon CloudWatch y habilite las métricas adecuadas para las opciones de red. Asegúrese de elegir la métrica de red adecuada para su carga de trabajo. Por ejemplo, puede activar métricas para el uso de direcciones de red VPC, la puerta de enlace de NAT de VPC, AWS Transit Gateway, túneles de VPN, AWS Network Firewall, Elastic Load Balancing y AWS Direct Connect. La supervisión continua de las métricas es una práctica recomendada para observar y comprender el estado y el uso de su red, lo que le ayuda a optimizar la configuración de esta con base en sus observaciones. |
| AWS Network Manager | Con AWS Network Manager, puede supervisar el rendimiento histórico y en tiempo real de la red global de AWS con fines operativos y de planificación. El Administrador de redes proporciona una latencia de red agregada entre las Regiones de AWS y las zonas de disponibilidad y dentro de cada zona de disponibilidad, lo que le permite comprender mejor la relación entre el rendimiento de las aplicaciones y el rendimiento de la red de AWS subyacente. |
| Amazon CloudWatch RUM | Use Amazon CloudWatch RUM para recopilar las métricas que le proporcionan información que le ayuda a identificar, comprender y mejorar la experiencia del usuario. |

- Identifique los principales interlocutores y patrones de tráfico de las aplicaciones mediante VPC y Registros de flujo de AWS Transit Gateway.

- Evalúe y optimice su arquitectura de red actual, incluidas las VPC, las subredes y el enrutamiento. Por ejemplo, puede evaluar cómo diferentes emparejamientos de VPC o AWS Transit Gateway pueden ayudarle a mejorar las redes de su arquitectura.
- Evalúe las rutas de enrutamiento de su red para verificar que siempre se utilice la ruta más corta entre los destinos. El Analizador de acceso a la red puede ayudarle a hacer esto.

Recursos

Documentos relacionados:

- Registro de consultas de DNS públicas
- ¿Qué es IPAM?
- ¿Qué es Analizador de accesibilidad?
- ¿Qué es Analizador de acceso a la red?
- Métricas de CloudWatch para sus VPC
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format
- Supervisión de la red global con métricas de Amazon CloudWatch
- Supervisar de forma continua el tráfico y los recursos de red

Videos relacionados:

- AWS re:Invent 2023 – A developer's guide to cloud networking
- AWS re:Invent 2023 – Ready for what's next? Designing networks for growth and flexibility
- AWS re:Invent 2023 – Advanced VPC designs and new capabilities
- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure
- AWS re:Invent 2020 – Networking best practices and tips with the AWS Well-Architected Framework
- AWS re:Invent 2020 – Monitoring and troubleshooting network traffic

Ejemplos relacionados:

- Talleres de redes de AWS
- Supervisión de la red de AWS

- Observing and diagnosing your network on AWS
- Finding and addressing network misconfigurations on AWS

## Proceso y cultura

Preguntas

- PERF 5. ¿Cómo contribuyen las prácticas y la cultura de su organización a la eficiencia del rendimiento en su carga de trabajo?

### PERF 5. ¿Cómo contribuyen las prácticas y la cultura de su organización a la eficiencia del rendimiento en su carga de trabajo?

Al diseñar cargas de trabajo, hay principios y prácticas que puede adoptar con el fin de ayudarle a ejecutar mejor cargas de trabajo en la nube eficientes y de alto rendimiento. Para adoptar una cultura que fomente la eficiencia del rendimiento de las cargas de trabajo en la nube, tenga en cuenta estos principios y prácticas clave:

Prácticas recomendadas

- PERF05-BP01 Establecimiento de indicadores clave de rendimiento (KPI) para medir el estado y el rendimiento de la carga de trabajo
- PERF05-BP02 Uso de soluciones de supervisión para saber en qué áreas es más crítico el rendimiento
- PERF05-BP03 Definición de un proceso para mejorar el rendimiento de la carga de trabajo
- PERF05-BP04 Pruebas de carga de la carga de trabajo
- PERF05-BP05 Uso de la automatización para solucionar de forma proactiva los problemas relacionados con el rendimiento
- PERF05-BP06 Mantenimiento de la carga de trabajo y los servicios actualizados
- PERF05-BP07 Revisión de las métricas a intervalos regulares

PERF05-BP01 Establecimiento de indicadores clave de rendimiento (KPI) para medir el estado y el rendimiento de la carga de trabajo

Identifique los KPI que miden de forma cuantitativa y cualitativa el rendimiento de la carga de trabajo. Los KPI ayudan a medir el estado y el rendimiento de una carga de trabajo en relación con un objetivo empresarial.

Patrones comunes de uso no recomendados:

- Supervisa únicamente las métricas del nivel del sistema para obtener información sobre su carga de trabajo sin comprender el impacto empresarial de dichas métricas.
- Presupone que los KPI ya se publican y comparten como datos de métricas estándar.
- No tiene definido un KPI cuantitativo (que se pueda medir).
- Los KPI no se corresponden con los objetivos o estrategias empresariales.

Beneficios de establecer esta práctica recomendada: identificar los KPI específicos que representan el estado y el rendimiento de la carga de trabajo ayuda a alinear a los equipos con sus prioridades y a definir unos resultados empresariales satisfactorios. Al compartir estas métricas con todos los departamentos, se obtiene información y se fomenta un enfoque coherente en relación con los umbrales, las expectativas y las repercusiones empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los KPI ayudan a las empresas y a los equipos de ingeniería a organizarse en función de la medición de los objetivos y estrategias. Además, indican cómo estos factores se combinan para producir resultados empresariales. Por ejemplo, en una carga de trabajo de un sitio web, el tiempo de carga de la página se podría usar como indicativo del rendimiento general. Esta métrica sería uno de los múltiples puntos de datos que miden la experiencia del usuario. Además de identificar los umbrales de los tiempos de carga de la página, debería documentar el resultado previsto o el riesgo empresarial si no se cumple el ideal de rendimiento. Si una página tarda en cargarse, los usuarios finales se ven directamente afectados, se reduce su valoración de la experiencia y se pueden perder clientes. Cuando defina los umbrales de KPI, combine tanto las referencias del sector como las expectativas de los usuarios finales. Por ejemplo, si la referencia sectorial actual es que una página web se cargue en dos segundos, pero los usuarios esperan que tarde solamente un segundo, debería tener en cuenta estos dos puntos de datos al establecer el KPI.

El equipo debe evaluar los KPI de su carga de trabajo por medio de datos granulares en tiempo real y datos históricos como referencia. Además, debe crear paneles en los que se hagan cálculos de métricas sobre los datos de los KPI para obtener información sobre las operaciones y la utilización. Los KPI se deben documentar e incluir umbrales que respalden los objetivos y las estrategias de la empresa, además de asignarse a las métricas que se supervisen. Los KPI deberían revisitarse siempre que cambien los objetivos empresariales, las estrategias o los requisitos del usuario final.

Pasos para la implementación

- Identificación de las partes interesadas: identifique y documente las partes interesadas clave de la empresa, como los equipos de desarrollo y operación.
- Definición de los objetivos: trabaje con estas partes interesadas para definir y documentar los objetivos de su carga de trabajo. Tenga en cuenta los aspectos esenciales de desempeño de las cargas de trabajo, como, por ejemplo, el rendimiento, el tiempo de respuesta y el costo, así como los objetivos empresariales, como, por ejemplo, la satisfacción del usuario.
- Revisión de las prácticas recomendadas del sector: revise las prácticas sectoriales recomendadas para identificar los KPI relevantes que se ajusten a los objetivos de su carga de trabajo.
- Identificación de las métricas: identifique las métricas que estén alineadas con los objetivos de su carga de trabajo y que puedan ayudarle a medir el rendimiento y los objetivos empresariales. Establezca los KPI en función de estas métricas. Las métricas de ejemplo son medidas como el tiempo promedio de respuesta o el número de usuarios simultáneos.
- Definición y documentación de los KPI: utilice las prácticas recomendadas del sector y los objetivos de su carga de trabajo para establecer los objetivos del KPI de su carga de trabajo. Utilice esta información para establecer los umbrales de gravedad o el nivel de alarma de los KPI. Identifique y documente el riesgo y el impacto del incumplimiento de los KPI.
- Implementación de la supervisión: utilice herramientas de supervisión como Amazon CloudWatch o AWS Config para recopilar métricas y medir los KPI.
- Comunicación de los KPI de forma visual: utilice herramientas de panel como Amazon Quick para visualizar y comunicar los KPI a las partes interesadas.
- Análisis y optimización: revise y analice periódicamente las métricas para identificar las áreas de la carga de trabajo que deben mejorarse. Colabore con las partes interesadas para implementar estas mejoras.
- Revisita y refinamiento: revise periódicamente las métricas y los KPI para evaluar su eficacia, especialmente cuando cambien los objetivos empresariales o el rendimiento de la carga de trabajo.

Recursos

Documentos relacionados:

- Documentación de CloudWatch
- Supervisión, registro y rendimiento de los AWS Partner

- Herramientas de observabilidad de AWS
- La importancia de los indicadores clave de rendimiento (KPI) para las migraciones a gran escala a la nube
- How to track your cost optimization KPIs with the KPI Dashboard
- Documentación de X-Ray
- Uso de paneles de Amazon CloudWatch
- KPI de Quick

Videos relacionados:

- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2023 - Manage resource lifecycle events at scale with AWS Health
- AWS re:Invent 2023 - Performance & efficiency at Pinterest: Optimizing the latest instances
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2023 - Scaling on AWS for the first 10 million users
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- Creating an Effective Metrics Strategy for Your Business | AWS Events

Ejemplos relacionados:

- Creación de un panel con Quick

PERF05-BP02 Uso de soluciones de supervisión para saber en qué áreas es más crítico el rendimiento

Comprenda y detecte las áreas en las que un aumento de rendimiento de la carga de trabajo tendrá un impacto positivo en la eficiencia o en la experiencia del cliente. Por ejemplo, un sitio web que tenga una gran interacción del cliente se beneficiaría de utilizar servicios en la periferia para acercar la entrega de contenido a los clientes.

Patrones comunes de uso no recomendados:

- Supone que las métricas de computación estándares como el uso de CPU o la presión sobre la memoria son suficientes para detectar problemas de rendimiento.
- Solo se utilizan las métricas predeterminadas registradas por el software de supervisión seleccionado.
- Solo se revisan las métricas cuando hay un problema.

Ventajas de establecer esta práctica recomendada: el conocimiento de las áreas críticas de rendimiento ayuda a los propietarios de la carga de trabajo a supervisar los KPI y a priorizar las mejoras de alto impacto.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Configure el seguimiento de extremo a extremo para identificar los patrones de tráfico, la latencia y las áreas esenciales de rendimiento. Supervise los patrones de acceso a los datos para detectar consultas lentas o datos fragmentados y particionados de forma deficiente. Identifique las áreas restringidas de la carga de trabajo mediante pruebas de carga o supervisión.

Para aumentar la eficiencia del rendimiento, comprenda su arquitectura, patrones de tráfico y patrones de acceso a los datos e identificar sus tiempos de latencia y procesamiento. Identifique los posibles cuellos de botella que puedan afectar a la experiencia del cliente a medida que aumenta la carga de trabajo. Al identificar esas áreas, fíjese en qué solución podría implementar para acabar con los problemas de rendimiento.

Pasos para la implementación

- Configure la supervisión de extremo a extremo para capturar todos los componentes y métricas de la carga de trabajo. A continuación, se muestran algunos ejemplos de soluciones de supervisión de AWS.

| Servicio | Dónde se usa |
| --- | --- |
| Amazon CloudWatch Real-User Monitoring (RUM) | Para capturar las métricas de rendimiento de las aplicaciones a partir de las sesiones reales de los usuarios en el cliente y del frontend. |
| AWS X-Ray | Para hacer un seguimiento del tráfico a través de las capas de la aplicación e identific |

| Servicio | Dónde se usa |
| --- | --- |
| | ar la latencia entre los componentes y las dependencias. Utilice los mapas de servicios de X-Ray para ver las relaciones y la latencia entre los componentes de la carga de trabajo. |
| Información sobre rendimiento de Amazon Relational Database Service | Para ver las métricas de rendimiento de la base de datos e identificar las mejoras de rendimiento. |
| Amazon RDS Enhanced Monitoring | Para ver las métricas de rendimiento del sistema operativo de la base de datos. |
| Amazon DevOps Guru | Para detectar patrones operativos anormales de forma que pueda identificar los problemas operativos antes de que afecten a sus clientes. |

- Lleve a cabo pruebas para generar métricas, identificar patrones de tráfico, cuellos de botella y áreas críticas de rendimiento. Estos son algunos ejemplos de cómo se hacen las pruebas:
  - Configure Canarios sintéticos de CloudWatch para imitar las actividades de los usuarios en el navegador mediante programación con expresiones de frecuencia o tareas cron de Linux y generar métricas coherentes a lo largo del tiempo.
  - Use la solución Pruebas de carga distribuidas de AWS para generar picos de tráfico o probar la carga de trabajo con la tasa de crecimiento prevista.
- Evalúe las métricas y la telemetría para identificar sus áreas fundamentales de rendimiento. Revise estas áreas con su equipo con el fin de analizar la supervisión y las soluciones para evitar los cuellos de botella.
- Experimente con las mejoras de rendimiento y mida los cambios con datos. Por ejemplo, puede usar CloudWatch Evidently para probar nuevas mejoras y los impactos en el rendimiento de su carga de trabajo.

Recursos

Documentos relacionados:

- What's new in AWS Observability at re:Invent 2023

- Amazon Builders' Library
- Documentación de X-Ray
- Amazon CloudWatch RUM
- Amazon DevOps Guru

Videos relacionados:

- AWS re:Invent 2023 - [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 - Implementing application observability
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2022 - The Amazon Builders' Library: 25 years of Amazon operational excellence
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- Visual Monitoring of Applications with Amazon CloudWatch Synthetics

Ejemplos relacionados:

- Measure page load time with Amazon CloudWatch Synthetics
- Cliente web de Amazon CloudWatch RUM
- SDK de X-Ray para Python
- Pruebas de carga distribuidas en AWS

PERF05-BP03 Definición de un proceso para mejorar el rendimiento de la carga de trabajo

Defina un proceso para evaluar nuevos servicios, patrones de diseño, tipos de recursos y configuraciones a medida que estén disponibles. Por ejemplo, ejecute las pruebas de rendimiento existentes en las nuevas ofertas de instancias a fin de determinar su capacidad para mejorar su carga de trabajo.

Patrones comunes de uso no recomendados:

- Presupone que la arquitectura actual es estática y no se va a actualizar con el tiempo.
- Incorpora cambios en la arquitectura a lo largo del tiempo sin justificación basada en métricas.

Beneficios de establecer esta práctica recomendada: al definir el proceso para hacer cambios en la arquitectura, puede utilizar los datos recopilados para influir en el diseño de la carga de trabajo a lo largo del tiempo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

El rendimiento de su carga de trabajo tiene algunas limitaciones clave. Documéntelos para que sepa qué tipos de innovación pueden mejorar el rendimiento de su carga de trabajo. Utilice esta información cuando conozca nuevos servicios o tecnologías a medida que estén disponibles para identificar formas de mitigar las limitaciones o cuellos de botella.

Identifique las principales restricciones en el rendimiento de su carga de trabajo. Documente las restricciones de rendimiento de la carga de trabajo para que sepa los tipos de innovación que puedan mejorarlo.

Pasos para la implementación

- Identificación de los KPI: identifique los KPI de rendimiento de su carga de trabajo tal como se describe en PERF05-BP01 Establecimiento de indicadores clave de rendimiento (KPI) para medir el estado y el rendimiento de la carga de trabajo para basar su carga de trabajo.
- Implementación de la supervisión: utilice herramientas de observabilidad de AWS para recopilar métricas de rendimiento y medir los KPI.
- Análisis: haga un análisis exhaustivo para identificar las áreas de la carga de trabajo (como la configuración y el código de la aplicación) que tienen un rendimiento inferior, tal y como se describe en PERF05-BP02 Uso de soluciones de supervisión para saber en qué áreas es más crítico el rendimiento. Utilice sus herramientas de análisis y rendimiento para identificar las estrategias de mejora del rendimiento.
- Validación de las mejoras: utilice entornos de pruebas o de preproducción para validar la eficacia de la estrategia.
- Implementación de cambios: implemente los cambios en la producción y supervise continuamente el rendimiento de la carga de trabajo. Documente las mejoras y comunique los cambios a las partes interesadas.
- Revisita y ajuste: revise periódicamente su proceso de mejora del rendimiento para identificar las áreas que se puedan optimizar.

Recursos

Documentos relacionados:

- AWS Blog de
- Novedades de AWS
- AWS Skill Builder de

Videos relacionados:

- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2023 - Optimize cost and performance and track progress toward mitigation
- AWS re:Invent 2022 - AWS optimization: Actionable steps for immediate results
- AWS re:Invent 2022 - Optimize your AWS workloads with best-practice guidance

Ejemplos relacionados:

- AWS GitHub

PERF05-BP04 Pruebas de carga de la carga de trabajo

Haga una prueba de carga en su carga de trabajo para comprobar que puede gestionar la carga de producción e identificar cualquier cuello de botella en el rendimiento.

Patrones comunes de uso no recomendados:

- Hace pruebas de partes individuales de su carga de trabajo, pero no de la carga completa.
- Hace pruebas de carga en una infraestructura que no es la misma que su entorno de producción.
- Solo hace pruebas de carga hasta su carga prevista y no más allá, para ayudar a prever dónde puede tener problemas en el futuro.
- Hace pruebas de carga sin consultar la Política de pruebas de Amazon EC2 ni presentar un formulario de envío de eventos simulados. Esto hace que la prueba no se ponga en marcha, ya que parece un evento de denegación de servicio.

Beneficios de establecer esta práctica recomendada: calcular el rendimiento en una prueba de carga le mostrará qué áreas se verán afectadas a medida que aumente la carga. De este modo, podrá anticipar los cambios necesarios antes de que afecten a la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Las pruebas de carga en la nube son un proceso que permite medir el rendimiento de la carga de trabajo en la nube bajo condiciones realistas y con la carga de usuarios esperada. Este proceso implica el aprovisionamiento de un entorno de nube similar al de producción, el uso de herramientas de pruebas de carga para generar la carga y el análisis de métricas para evaluar la capacidad de la carga de trabajo a la hora de gestionar una carga realista. Las pruebas de carga deben ponerse en marcha con versiones sintéticas o saneadas de los datos de producción (debe eliminarse la información confidencial o de identificación). Haga automáticamente pruebas de carga en la canalización de entrega y compare los resultados con los KPI y los umbrales predefinidos. Este proceso le permitirá seguir alcanzando el rendimiento requerido.

Pasos para la implementación

- Definición de los objetivos de la prueba: identifique los aspectos de desempeño de su carga de trabajo que desea evaluar, como el rendimiento y el tiempo de respuesta.
- Selección de una herramienta para hacer la prueba: elija y configure la herramienta para hacer la prueba de carga que se ajuste a su carga de trabajo.
- Configuración del entorno: configure el entorno de prueba en función de su entorno de producción. Puede usar los servicios de AWS para poner en marcha entornos a escala de producción y poner a prueba su arquitectura.
- Implementación de la supervisión: utilice herramientas de supervisión como Amazon CloudWatch para recopilar métricas de todos los recursos de su arquitectura. También puede recopilar y publicar métricas personalizadas.
- Definición de escenarios: defina los escenarios y los parámetros de las pruebas de carga (como la duración de la prueba y el número de usuarios).
- Pruebas de carga: lleve a cabo escenarios de prueba a escala. Utilice la Nube de AWS para probar la carga de trabajo y detectar las áreas en las que el escalado no se hace correctamente o no se produce de forma lineal. Por ejemplo, utilice instancias de spot para generar cargas a bajo costo y descubrir obstáculos antes que se experimenten en la producción.

- Análisis de los resultados de las pruebas: analice los resultados para identificar los cuellos de botella del rendimiento y las áreas en las que se pueden mejorar.
- Documentación y comunicación de los resultados: documente e informe sobre los resultados y recomendaciones. Comparta esta información con las partes interesadas para que puedan tomar decisiones fundamentadas con respecto a las estrategias de optimización del rendimiento.
- Repetición continua: las pruebas de carga deben hacerse con periodicidad, especialmente después de un cambio o actualización del sistema.

Recursos

Documentos relacionados:

- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Pruebas de carga distribuidas en AWS

Videos relacionados:

- AWS Summit ANZ 2023: Accelerate with confidence through AWS Distributed Load Testing
- AWS re:Invent 2022 - Scaling on AWS for your first 10 million users
- Solving with AWS Solutions: Distributed Load Testing

- AWS re:Invent 2021 - Optimize applications through end user insights with Amazon CloudWatch RUM
- Demo of Amazon CloudWatch Synthetics

Ejemplos relacionados:

- Pruebas de carga distribuidas en AWS

PERF05-BP05 Uso de la automatización para solucionar de forma proactiva los problemas relacionados con el rendimiento

Utilice los indicadores clave de rendimiento (KPI), junto con los sistemas de supervisión y alerta, para abordar de forma proactiva los problemas relacionados con el rendimiento.

Patrones comunes de uso no recomendados:

- Únicamente permite que el personal de operaciones pueda llevar a cabo cambios operativos en la carga de trabajo.
- Permite que todas las alarmas se filtren al equipo de operaciones sin medidas de corrección proactivas.

Beneficios de establecer esta práctica recomendada: la corrección proactiva de las acciones de alarma permite al personal de asistencia centrarse en aquellos elementos que no son accionables automáticamente. De este modo, el personal de operaciones podrá gestionar todas las alarmas sin sentirse abrumado y concentrarse exclusivamente en las críticas.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Use alarmas para activar acciones automatizadas y corregir los problemas siempre que sea posible. Escale la alarma a aquellos capaces de responder cuando no se pueda recurrir a la respuesta automatizada. Por ejemplo, podría tener un sistema capaz de predecir los valores esperados de los indicadores clave de rendimiento (KPI) y emitir alarmas cuando se sobrepasen ciertos umbrales, o una herramienta que pudiera detener o revertir automáticamente las implementaciones si los KPI están fuera de los valores esperados.

Implemente procesos que informen el rendimiento cuando la carga de trabajo esté en marcha. Cree paneles de supervisión y establezca normas de referencia sobre las expectativas del rendimiento para determinar si la carga de trabajo funciona de manera óptima.

Pasos para la implementación

- Identificación del flujo de trabajo de corrección: identifique y estudie si el problema de rendimiento puede solucionarse automáticamente. Utilice soluciones de supervisión de AWS, como Amazon CloudWatch o AWS X-Ray, que le permitan entender mejor la causa raíz del problema.
- Definición de un proceso de automatización: cree un plan y un proceso de corrección paso a paso que pueda utilizar para solucionar el problema automáticamente.
- Configure el evento de inicio: configure el evento para iniciar automáticamente el proceso de corrección. Por ejemplo, puede definir un activador que reinicie automáticamente una instancia cuando se alcance un determinado umbral de uso de la CPU.

- Automatización de la corrección: utilice los servicios y las tecnologías de AWS para automatizar el proceso de corrección. Por ejemplo, Automatización de AWS Systems Manager proporciona una forma segura y escalable para automatizar el proceso de corrección. Asegúrese de usar la lógica de autorrecuperación para revertir los cambios si el problema no se soluciona correctamente.
- Prueba del flujo de trabajo: pruebe el proceso de corrección automatizado en un entorno de preproducción.
- Implementación del flujo de trabajo: implemente la corrección automática en el entorno de producción.
- Elaboración de un manual de estrategias: elabore y documente un manual de estrategias que describa los pasos del plan de corrección, incluidos los eventos de inicio, la lógica de corrección y las medidas adoptadas. Asegúrese de que las partes interesadas reciban formación para que puedan responder de manera eficaz a los eventos de corrección automatizada.
- Revisión y perfeccionamiento: evalúe periódicamente la eficacia del flujo de trabajo de corrección automatizado. Ajuste los eventos de inicio y la lógica de corrección si es necesario.

Recursos

Documentos relacionados:

- Documentación de CloudWatch
- Socios de AWS Partner Network de supervisión, registro y rendimiento
- Documentación de X-Ray
- Using Alarms and Alarm Actions in CloudWatch
- Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services
- Automate your Amazon Redshift performance tuning with automatic table optimization

Videos relacionados:

- AWS re:Invent 2023 - Strategies for automated scaling, remediation, and smart self-healing
- AWS re:Invent 2023 - [LAUNCH] Application monitoring for modern workloads
- AWS re:Invent 2023 - Implementing application observability
- AWS re:Invent 2021 - Intelligently automating cloud operations
- AWS re:Invent 2022 - Setting up controls at scale in your AWS environment

- AWS re:Invent 2022 - Automating patch management and compliance using AWS
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS
- AWS re:Invent 2021 -{New Launch} Automatically detect and resolve issues with Amazon DevOps Guru
- AWS re:Invent 2023 - Centralize your operations

Ejemplos relacionados:

- CloudWatch Logs Customize Alarms

PERF05-BP06 Mantenimiento de la carga de trabajo y los servicios actualizados

Manténgase al tanto de los nuevos servicios y características de la nube para adoptar características eficientes, resolver problemas y mejorar la eficiencia general del rendimiento de la carga de trabajo.

Patrones comunes de uso no recomendados:

- Asume que su arquitectura actual es estática y no se actualizará con el tiempo.
- No dispone de sistemas ni de una cadencia regular para evaluar si los programas y paquetes actualizados son compatibles con la carga de trabajo.

Beneficios de establecer esta práctica recomendada: al establecer un proceso que le permita estar al tanto de los nuevos servicios y ofertas, puede adoptar nuevas características y funcionalidades, resolver problemas y mejorar el rendimiento de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Evalúe mecanismos para mejorar el rendimiento a medida que disponga de nuevos servicios, patrones de diseño y características de productos. Determine cuáles de ellas podrían mejorar el rendimiento o aumentar la eficiencia de la carga de trabajo mediante una evaluación, un debate interno o un análisis externo. Defina un proceso para evaluar las actualizaciones, las nuevas características y servicios pertinentes para su carga de trabajo. Por ejemplo, cree una prueba de concepto que utilice nuevas tecnologías o consulte a un grupo interno. Cuando pruebe nuevas ideas

o servicios, haga pruebas de rendimiento para medir el impacto que tienen en el rendimiento de la carga de trabajo.

Pasos para la implementación

- Inventario de la carga de trabajo: haga un inventario del software y la arquitectura de su carga de trabajo e identifique los componentes que deben actualizarse.
- Identificación de los orígenes de actualización: identifique las noticias y los orígenes de actualización relacionados con los componentes de su carga de trabajo. Por ejemplo, puede suscribirse al blog de novedades de AWS para ver los productos que se adapten a su componente de carga de trabajo. Puede suscribirse a la fuente RSS o administrar sus suscripciones de correo electrónico.
- Definición de un calendario de actualización: establezca un calendario para evaluar nuevos servicios y características con su carga de trabajo.
  - Puede usar Inventario de AWS Systems Manager para recopilar los metadatos del sistema operativo (SO), las aplicaciones y los metadatos de instancias de sus instancias de Amazon EC2 y comprender rápidamente qué instancias están poniendo en marcha el software y las configuraciones requeridas por su política de software, así como las instancias que deben actualizarse.
- Evaluación de la nueva actualización: entienda cómo actualizar los componentes de su carga de trabajo. Aproveche la agilidad de la nube para probar rápidamente cómo las nuevas características pueden mejorar la eficiencia del rendimiento de la carga de trabajo.
- Uso de la automatización: utilice la automatización del proceso de actualización a fin de reducir el nivel de esfuerzo para implementar nuevas funciones y limitar los errores causados por los procesos manuales.
  - Puede usar CI/CD para actualizar automáticamente las AMI, las imágenes de contenedor y otros artefactos relacionados con la aplicación en la nube.
  - Puede utilizar herramientas como AWS Systems Manager Patch Manager para automatizar el proceso de actualizaciones del sistema y programar la actividad mediante Ventanas de mantenimiento de AWS Systems Manager.
- Documentación del proceso: documente su proceso para evaluar las actualizaciones y los nuevos servicios. Proporcione a los propietarios el tiempo y el espacio necesarios para investigar, probar, experimentar y validar las actualizaciones y los nuevos servicios. Consulte los requisitos empresariales documentados y los KPI para ayudar a priorizar qué actualización tendrá un impacto empresarial positivo.

Recursos

Documentos relacionados:

- Blog de AWS
- Novedades de AWS
- Implementing up-to-date images with automated EC2 Image Builder pipelines

Videos relacionados:

- AWS re:Inforce 2022 - Automating patch management and compliance using AWS
- All Things Patch: AWS Systems Manager | AWS Events

Ejemplos relacionados:

- Inventory and Patch Management
- One Observability Workshop

PERF05-BP07 Revisión de las métricas a intervalos regulares

Revise qué métricas se recopilan durante el mantenimiento rutinario o en respuesta a eventos o incidentes. Utilice estas revisiones para determinar qué métricas son esenciales para abordar los problemas y cuáles otras, en caso de que se les haga un seguimiento, podrían ayudar a identificar, abordar o prevenir problemas.

Patrones comunes de uso no recomendados:

- Permite que las métricas se mantengan en un estado de alarma durante un periodo de tiempo prolongado.
- Crea alarmas que un sistema de automatización no puede accionar.

Beneficios de establecer esta práctica recomendada: revise continuamente las métricas que se recopilan para verificar que puedan identificar, abordar o prevenir problemas correctamente. Las métricas también pueden quedarse obsoletas si deja que permanezcan en un estado de alarma durante mucho tiempo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Mejore continuamente la recopilación y la supervisión de métricas. Como parte de la respuesta a incidentes o sucesos, evalúe qué métricas fueron útiles para abordar el problema y cuáles podrían haber ayudado, pero no se les da seguimiento actualmente. Utilice este método para mejorar la calidad de las métricas que recopila, de modo que pueda prevenir o resolver incidentes en el futuro con mayor rapidez.

Como parte de la respuesta a incidentes o sucesos, evalúe qué métricas fueron útiles para abordar el problema y cuáles podrían haber ayudado, pero no se les da seguimiento actualmente. Utilícelo para mejorar la calidad de la métrica que recopila, de modo que pueda prevenir o resolver más rápidamente incidentes futuros.

Pasos para la implementación

- Definición de las métricas: defina las métricas de rendimiento críticas para supervisar que estén adaptadas al objetivo de su carga de trabajo. Esto incluye métricas como el tiempo de respuesta y la utilización de los recursos.
- Establecimiento de bases de referencia: establezca una base de referencia y el valor que desee para cada métrica. La base de referencia debe proporcionar puntos de referencia para identificar desviaciones o anomalías.
- Configuración de una cadencia: establezca una cadencia (como semanal o mensual) para revisar las métricas críticas.
- Identificación de los problemas de rendimiento: durante cada revisión, evalúe las tendencias y la desviación de los valores de la base de referencia. Busque cualquier cuello de botella o anomalía en el rendimiento. Lleve a cabo un análisis exhaustivo de la causa raíz de los problemas identificados para conocer qué los provoca.
- Identificación de las acciones correctivas: utilice su análisis para identificar las acciones correctivas. Entre dichas medidas se pueden incluir el ajuste de parámetros, la corrección de errores y el escalado de los recursos.
- Documentación de los resultados: documente sus resultados, incluidos los problemas identificados, las causas raíz y las acciones correctivas.
- Iteración y mejora: evalúe y mejore continuamente el proceso de revisión de las métricas. Aplique lo que ha aprendido de la revisión anterior para mejorar el proceso con el tiempo.

Recursos

Documentos relacionados:

- Documentación de CloudWatch
- Recopilación de métricas y registros de instancias de Amazon EC2 y en los servidores en las instalaciones con el agente de CloudWatch
- Consulte sus métricas con Información de métricas de CloudWatch
- Socios de AWS Partner Network de supervisión, registro y rendimiento
- Documentación de X-Ray

Videos relacionados:

- AWS re:Invent 2022 - Setting up controls at scale in your AWS environment
- AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance
- AWS re:Invent 2023 - Building an effective observability strategy
- AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS
- AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS

Ejemplos relacionados:

- Creación de un panel con Quick
- CloudWatch Dashboards

# Optimización de costos

El pilar de optimización de costos incluye la capacidad de ejecutar sistemas para ofrecer valor empresarial al precio más bajo posible. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de optimización de costos.

Áreas de prácticas recomendadas

- Práctica de administración financiera en la nube
- Conocimiento del gasto y del uso
- Recursos rentables

- Administración de la demanda y suministro de recursos
- Optimización a lo largo del tiempo

## Práctica de administración financiera en la nube

Pregunta

- COST 1. ¿Cómo implementa la administración financiera en la nube?

### COST 1. ¿Cómo implementa la administración financiera en la nube?

Implementar la administración financiera en la nube ayuda a las empresas a obtener valor empresarial y éxito financiero al optimizar su costo y uso, y al escalar en AWS.

Prácticas recomendadas

- COST01-BP01: establecimiento de la responsabilidad de la optimización de costos
- COST01-BP02 Establecimiento de la colaboración entre los departamentos de finanzas y tecnología
- COST01-BP03 Establecimiento de presupuestos y previsiones de la nube
- COST01-BP04 Implementación de conciencia de costos en los procesos organizativos
- COST01-BP05 Creación de informes y notificaciones sobre la optimización de costos
- COST01-BP06 Supervisión proactiva de los costos
- COST01-BP07 Seguimiento de la información sobre las nuevas versiones de los servicios
- COST01-BP08 Creación de una cultura sensibilizada con los costos
- COST01-BP09: cuantificación del valor empresarial de la optimización de costos

#### COST01-BP01: establecimiento de la responsabilidad de la optimización de costos

Cree un equipo (Oficina de negocios en la nube, Centro de excelencia en la nube o equipo de FinOps) que se encargue de establecer y afianzar el concienciación sobre los costos en toda la organización. El responsable de la optimización de costos puede ser una persona o un equipo (requiere representantes de los equipos financieros, tecnológicos y empresariales) que comprenda toda la organización y las finanzas en la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Esta es la introducción a un equipo de Oficina de negocios en la nube (CBO) o Centro de excelencia en la nube (CCOE) que se encargue de establecer y afianzar una cultura de concienciación sobre los costos en la computación en la nube. Esta función puede ser una persona existente, un equipo de la organización o un nuevo equipo formado por partes interesadas clave de los equipos de Finanzas, Tecnología y Organización de la organización.

Esta función (la persona o el equipo) prioriza y dedica el porcentaje necesario de su tiempo a actividades de administración y optimización de costos. En una organización pequeña, es posible que esta función dedique menos tiempo a ello, si lo comparamos con una función a tiempo completo de una empresa grande.

Esta función debe tener carácter multidisciplinar, es decir, debe tener experiencia en administración de proyectos, ciencia de datos, análisis financiero y desarrollo de software o infraestructura. Para mejorar la eficiencia de la carga de trabajo, puede ejecutar optimizaciones de costos dentro de tres propietarios diferentes:

- Centralizado: a través de equipos designados, como el equipo de FinOps, el equipo de Administración financiera en la nube (CFM), la oficina de negocios en la nube (CBO) o el Centro de excelencia en la nube (CCoE), los clientes pueden diseñar e implementar mecanismos de gobernanza e impulsar las prácticas recomendadas en toda la empresa.
- Descentralizado: se influye en los equipos de Tecnología para que haga optimizaciones de costos.
- Híbrido: una combinación de equipos centralizados y descentralizados que pueden trabajar de forma conjunta para optimizar los costos.

La función se evalúa según su capacidad de ejecutar y alcanzar los objetivos de optimización de costos (por ejemplo, las métricas de eficiencia de las cargas de trabajo).

Debe conseguir el patrocinio de los ejecutivos para esta función, lo cual es un factor clave para el éxito. El patrocinador es considerado el campeón del consumo rentable de la nube y proporciona apoyo al equipo para garantizar que las actividades de optimización de costos se traten según el nivel de prioridad definido por la organización. De lo contrario, se podrán ignorar las directrices y no se dará prioridad a las oportunidades de ahorro. De forma conjunta, el patrocinador y el equipo garantizan que su organización haga un consumo eficiente de la nube y ofrezca valor empresarial.

Si tiene el plan de asistencia Business, Enterprise-On-Ramp o Enterprise y necesita ayuda para crear este equipo o función, contacte con los expertos de administración financiera en la nube (CFM) a través de su equipo de Cuentas.

Pasos para la implementación

- Definición de los miembros clave: todas las partes pertinentes de la organización deben contribuir y estar interesadas en la administración de costos. En general, los equipos de las organizaciones constan de equipos de Finanzas, propietarios de aplicaciones o productos, y equipos administrativos y técnicos (DevOps). Algunos tienen dedicación completa (técnicos y financieros) mientras que otros participan periódicamente, según sea necesario. Las personas o los equipos encargadas de la CFM necesitan el siguiente conjunto de habilidades:
  - Desarrollo de software: en el caso de que se creen scripts y automatización.
  - Conocimientos de ingeniería de infraestructuras: para implementar scripts, automatizar procesos y entender cómo se aprovisionan los servicios o recursos.
  - Perspicacia en las operaciones: la CFM consiste en operar en la nube de forma eficiente midiendo, supervisando, modificando, planificando y escalando el uso eficiente de la nube.
- Definición de los objetivos y las métricas: esta función debe proporcionar valor a la organización de distintas maneras. Estos objetivos se definen y evolucionan de forma continua a medida que evoluciona la organización. Estas son las actividades habituales: crear y ejecutar programas educativos sobre optimización de costos en la organización, desarrollar estándares para toda la organización (como la supervisión y la creación de informes de optimización de costos) y establecer objetivos de carga de trabajo sobre la optimización. Esta función también debe informar regularmente a la organización sobre la capacidad de optimizar costos de la organización.

  Puede definir indicadores clave de rendimiento (KPI) basados en el valor o el costo. Cuando se definen los KPI, se puede calcular el costo previsto en términos de eficiencia y el resultado empresarial esperado. Los KPI basados en el valor vinculan las métricas de costo y uso a los impulsores del valor empresarial y ayudan a racionalizar los cambios en el gasto de AWS. El primer paso para derivar los KPI basados en el valor es trabajar juntos, entre organizaciones, para seleccionar y acordar un conjunto estándar de KPI.
- Establecimiento de una cadencia periódica: el grupo (equipos de Finanzas, Tecnología y Negocios) debe reunirse de manera regular para revisar sus objetivos y métricas. Una cadencia típica implica revisar el estado de la organización, los programas que se ejecutan actualmente y las métricas generales financieras y de optimización. Después, se debe informar sobre las cargas de trabajo clave con mayor detalle.

Durante estas revisiones periódicas, se puede revisar la eficiencia de la carga de trabajo (costo) y los resultados empresariales. Por ejemplo, un aumento del 20 % en el costo de una carga de trabajo puede coincidir con un mayor uso por parte del cliente. En este caso, este aumento del 20 % de los costos puede interpretarse como una inversión. Estas reuniones de cadencia periódica pueden ayudar a los equipos a identificar los KPI de valor que proporcionan significado a toda la organización.

Recursos

Documentos relacionados:

- Blog de CCoE de AWS
- Creating Cloud Business Office
- CCOE - Cloud Center of Excellence

Videos relacionados:

- Vanguard CCOE Success Story

Ejemplos relacionados:

- Using a Cloud Center of Excellence (CCOE) to Transform the Entire Enterprise
- Building a CCOE to transform the entire enterprise
- 7 Pitfalls to Avoid When Building CCOE

COST01-BP02 Establecimiento de la colaboración entre los departamentos de finanzas y tecnología

Debe implicar a los equipos de finanzas y tecnología en las conversaciones sobre costos y uso en todas las etapas del traspaso a la nube. Los equipos deben reunirse y tratar regularmente temas como los objetivos organizativos, el estado actual de los costos y el uso, y las prácticas contables y financieras.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

## Guía para la implementación

Los equipos tecnológicos innovan más rápido en la nube gracias a que los ciclos de aprobación, adquisición e implementación de la infraestructura son más cortos. Esto puede implicar un ajuste para las organizaciones financieras acostumbradas a tanto a lanzar procesos que requieren mucho tiempo y consumen muchos recursos para obtener e implementar capital en entornos de centros de datos y en las instalaciones, como a asignar los costos solo al aprobar el proyecto.

Desde el punto de vista de las organizaciones financieras y de adquisiciones, el proceso de presupuestos de capital, las solicitudes de capital, las aprobaciones, las adquisiciones y la instalación de la infraestructura física es un proceso que se ha aprendido y estandarizado durante décadas:

- Los equipos de Ingeniería o TI suelen ser los solicitantes.
- Varios equipos de Finanzas actúan como aprobadores y compradores.
- Los equipos de Operaciones montan, apilan y entregan la infraestructura lista para usar.




Con la adopción de la nube, la adquisición y el consumo de infraestructura dejan de estar a expensas de una cadena de dependencias. En el modelo de la nube, los equipos de tecnología y productos ya no son solo creadores, sino operadores y propietarios de sus productos, responsables de la mayoría de las actividades históricamente asociadas a los equipos de finanzas y operaciones, incluidas las adquisiciones y la implementación.

Todo lo que se necesita para aprovisionar recursos en la nube es una cuenta y el conjunto de permisos adecuado. Esto es también lo que reduce el riesgo de la TI y de las finanzas, lo cual significa que, con unos pocos clics o llamadas a la API, los equipos pueden eliminar los recursos inactivos o innecesarios en la nube. Esto también permite a los equipos de tecnología innovar más rápidamente gracias a la agilidad y la capacidad de poner en marcha experimentos

y luego desmantelarlos. Aunque la naturaleza variable del consumo de la nube puede afectar a la previsibilidad desde el punto de vista de los presupuestos y las previsiones de capital, la nube ofrece a las organizaciones la posibilidad de reducir el costo del exceso de aprovisionamiento, así como el costo de oportunidad asociado al subaprovisionamiento conservador.




Establezca una asociación entre los partes interesadas clave de Finanzas y Tecnología para lograr un entendimiento común de los objetivos organizativos y desarrollar mecanismos para obtener éxito financiero en el modelo de gasto variable de la computación en la nube. Los equipos pertinentes de su organización deben estar presentes en las conversaciones sobre costos y uso en todas las etapas del traspaso a la nube, incluidos los siguientes:

- Líderes de finanzas: los directores financieros, responsables financieros, planificadores financieros, analistas empresariales y responsables de adquisición, abastecimiento y cuentas por pagar deben entender el modelo de consumo de la nube, las opciones de compra y el proceso de facturación mensual. El departamento de Finanzas debe asociarse con los equipos de Tecnología

para crear y compartir una historia de valor de TI y, así, ayudar a los equipos de Negocios a comprender cómo el gasto en tecnología está vinculado a los resultados empresariales. De esta manera, los gastos en tecnología no se consideran costos, sino inversiones. Dado que hay diferencias fundamentales entre la nube (por ejemplo, la velocidad del cambio en el uso, los precios de pago por uso, los precios por niveles, los modelos de precios y la información detallada sobre la facturación y el uso) y las operaciones en las instalaciones, resulta esencial que el equipo de Finanzas comprenda de qué manera puede afectar el uso de la nube a aspectos empresariales como los procesos de adquisición, el seguimiento de incentivos, la asignación de costos y los estados financieros.

- Líderes de tecnología: los líderes de tecnología (incluidos los propietarios de aplicaciones y productos) deben conocer los requisitos financieros (por ejemplo, las limitaciones presupuestarias) y los requisitos empresariales (por ejemplo, los acuerdos de nivel de servicio). Esto permite que la carga de trabajo se implemente para lograr los objetivos empresariales deseados.

La asociación entre los departamentos de Finanzas y Tecnología aporta los siguientes beneficios:

- Los equipos de finanzas y tecnología tienen visibilidad casi en tiempo real de los costos y el uso.
- Los equipos de finanzas y tecnología establecen un procedimiento operativo estándar para gestionar la variación del gasto en la nube.
- Las partes interesadas de Finanzas actúan como asesores estratégicos en cuanto a cómo se utiliza el capital para comprar descuentos por compromiso de compra (por ejemplo, instancias reservadas o Savings Plans de AWS) y cómo se utiliza la nube para hacer crecer la organización.
- Las cuentas por pagar y los procesos de adquisición existentes también se usan con la nube.
- Los equipos de Finanzas y Tecnología colaboran a la hora de prever los costos y el uso de AWS en el futuro para adaptar y diseñar los presupuestos organizativos.
- Mejor comunicación dentro de la organización al compartir el mismo lenguaje y tener un conocimiento común de los conceptos financieros.

Otras partes interesadas de su organización que deberían estar implicadas en las discusiones sobre costos y uso son:

- Propietarios de unidades de negocio: los propietarios de unidades de negocio deben comprender el modelo de negocio de la nube para poder establecer directrices para las unidades de negocio y la empresa entera. Este conocimiento de la nube resulta esencial para llevar a cabo previsiones

de crecimiento y de uso de las cargas de trabajo, pero también para valorar diferentes opciones de compra a más largo plazo (como instancias reservadas o Savings Plans).

- Equipo de Ingeniería: establecer una asociación entre los equipos de Finanzas y Tecnología es esencial para crear una cultura sensibilizada con los costos que anime a los ingenieros a actuar en la administración financiera en la nube (CFM). Uno de los problemas habituales de los profesionales de la CFM o las operaciones financieras y de los equipos de Finanzas es conseguir que los ingenieros entiendan todo el negocio en la nube, sigan las prácticas recomendadas y adopten las medidas recomendadas.
- Terceros: si en su organización participan terceros (por ejemplo, consultores o herramientas), asegúrese de que también sigan sus objetivos empresariales y que lo demuestren a través de sus modelos de compromiso y el retorno de la inversión (ROI). Por lo general, los terceros contribuyen a la generación de informes y al análisis de las cargas de trabajo que administran y también aportan análisis de costos de cualquier carga de trabajo que diseñan.

Para implementar la CFM y tener éxito, se necesita colaboración entre los equipos de Finanzas, Tecnología y Negocios y un cambio en la forma en que se comunica y evalúa el gasto en la nube en toda la organización. Incluya a los equipos de Ingeniería para que puedan formar parte de estas conversaciones sobre costos y uso en todas las etapas y anímelos a seguir las prácticas recomendadas y a adoptar las medidas acordadas de forma apropiada.

Pasos para la implementación

- Definición de los miembros clave: compruebe que todos los miembros pertinentes de sus equipos de Finanzas y Tecnología participen en la asociación. Los miembros de Finanzas pertinentes serán aquellos que interactúen con la factura de la nube. Suelen ser los directores financieros, responsables financieros, planificadores financieros, analistas empresariales, responsables de adquisiciones y responsables de abastecimiento. Los miembros de Tecnología suelen ser los propietarios de las aplicaciones y los productos y los gerentes y representantes técnicos de todos los equipos que crean en la nube. Otros miembros pueden ser los propietarios de las unidades de negocio, como el departamento de Marketing, que influyen en el uso de los productos, y terceros como consultores, para ajustarse a los objetivos y mecanismos y para asistir en la generación de informes.
- Definición de los temas de análisis: defina los temas comunes de todos los equipos o que requieran una compresión compartida. Haga un seguimiento del costo desde el momento en que se genera hasta que se paga la factura. Tome nota de todos los miembros implicados y de los procesos organizativos que deben aplicarse. Comprenda cada paso o proceso por el que pasa

y la información asociada, como los modelos de precios disponibles, los precios por niveles, los modelos de descuento, la creación de presupuestos y los requisitos financieros.

- Establecimiento de una cadencia periódica: para crear una asociación entre los equipos de Finanzas y Tecnología, establezca una cadencia de comunicación periódica para crear y mantener la coherencia. El grupo debe reunirse de forma periódica para tratar sus objetivos y métricas. Una cadencia típica implica revisar el estado de la organización, los programas que se ejecutan actualmente y las métricas generales financieras y de optimización. Después, se debe informar sobre las cargas de trabajo clave con mayor detalle.

Recursos

Documentos relacionados:

- AWS Blog de noticias de

COST01-BP03 Establecimiento de presupuestos y previsiones de la nube

Ajuste los procesos de presupuestos y previsión de la organización para que sean compatibles con la naturaleza altamente variable de los costos y el uso de la nube. Los procesos deben ser dinámicos y usar algoritmos basados en tendencias o el motor principal de la empresa, o en una combinación de ambos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

En las configuraciones tradicionales de TI en las instalaciones, los clientes suelen enfrentarse al reto que supone la planificación de unos costos fijos que solo cambian en ocasiones, normalmente con la compra de nuevos servicios y hardware de TI para satisfacer los picos de demanda. En cambio, Nube de AWS adopta un enfoque diferente, en el que los clientes pagan por los recursos que utilizan de acuerdo con sus necesidades empresariales y de TI reales. En el entorno de la nube, la demanda puede fluctuar cada mes, cada día o incluso cada hora.

El uso de la nube aporta eficacia, rapidez y agilidad, lo que se traduce en un patrón de costo y uso muy variable. Los costos pueden disminuir o a veces aumentar en respuesta a la mayor eficacia de las cargas de trabajo o a la implementación de nuevas cargas de trabajo y características. Cuando las cargas de trabajo se escalan para atender a una base de clientes en expansión, el uso y los costos de la nube aumentan en consecuencia debido a la mayor accesibilidad de los recursos. Esta

flexibilidad de los servicios en la nube se extiende a los costos y las previsiones, lo que da lugar a un cierto grado de elasticidad.

Es esencial ajustarse muy bien a estos impulsores de la demanda y necesidades empresariales tan cambiantes e intentar que la planificación sea lo más precisa posible. Los procesos presupuestarios tradicionales de la organización deben adaptarse a esta variabilidad.

Considere la posibilidad de crear modelos de costos a la hora de pronosticar el costo de las nuevas cargas de trabajo. Los modelos de costos le permiten disponer de información de referencia sobre los costos previstos de la nube, lo que lo ayuda a llevar a cabo análisis sobre el costo total de propiedad (TCO), el retorno de la inversión (ROI) y otros análisis financieros, fijar objetivos y expectativas con las partes interesadas e identificar oportunidades para optimizar los costos.

Su organización debe conocer las definiciones de costos y las agrupaciones aceptadas. El nivel de detalle de las previsiones puede variar en función de la estructura y los flujos de trabajo internos de su organización. Seleccione una granularidad que se adapte a sus requisitos específicos y a la configuración de su organización. Es importante entender a qué nivel se hace la previsión:

- AWS Organizations o cuenta de administración: la cuenta de administración es la que usa para crear AWS Organizations. De forma predeterminada, las organizaciones tienen una cuenta de administración.
- Cuenta de miembro o vinculada: una cuenta de Organizations es una Cuenta de AWS estándar que contiene sus recursos de AWS y las identidades que pueden acceder a esos recursos.
- Entorno: un entorno es un conjunto de recursos de AWS que ejecutan una versión de la aplicación. Se puede crear un entorno con varias cuentas vinculadas o de miembro.
- Proyecto: un proyecto es una combinación de objetivos o tareas establecidos que deben lograrse en un periodo fijo. Es importante tener en cuenta el ciclo de vida del proyecto durante la previsión.
- Servicios de AWS: grupos o categorías, como servicios de computación o almacenamiento, en los que puede agrupar servicios de AWS para su previsión.
- Agrupación personalizada: puede crear grupos personalizados en función de las necesidades de su organización, como unidades de negocio, centros de costos, equipos, etiquetas de asignación de costos, categorías de costos, cuentas vinculadas o una combinación de estas necesidades.

Identifique los impulsores empresariales que pueden repercutir en sus costos de uso y haga previsiones para cada uno de ellos por separado para calcular el uso esperado con antelación. Algunos de los impulsores podrían estar relacionados con los equipos de TI y de Productos de la

organización. Los líderes de ventas, marketing y negocios conocen otros impulsores empresariales, como los eventos de marketing, las promociones, las expansiones geográficas, las fusiones y las adquisiciones, por lo que es importante colaborar con ellos y tenerlos en cuenta también.

Puede usar AWS Cost Explorer para previsiones basadas en tendencias en un intervalo de tiempo futuro definido en función de su gasto anterior. El motor de previsión de AWS Cost Explorer segmenta los datos históricos en función de los tipos de cargo (por ejemplo, instancias reservadas) y utiliza una combinación de machine learning y modelos basados en reglas para predecir el gasto en todos los tipos de cargo individualmente.

Una vez que haya establecido el proceso de previsión y creado los modelos, puede usar AWS Budgets para establecer presupuestos personalizados de forma pormenorizada especificando el periodo de tiempo, la periodicidad o el importe (fijo o variable) y agregando filtros como el servicio, la Región de AWS y las etiquetas. Por lo general, el presupuesto se prepara para un solo año y no cambia, por lo que todas las partes involucradas deben ajustarse a él estrictamente. Por el contrario, las previsiones son más flexibles, ya que permiten reajustes a lo largo del año y proporcionan proyecciones dinámicas durante un periodo de uno, dos o tres años. La elaboración de presupuestos y de previsiones desempeña un papel crucial a la hora de establecer las expectativas financieras entre las diversas partes interesadas de los equipos de Tecnología y Negocios. La precisión de las previsiones y de su implementación también impone responsabilidades a las partes interesadas, que son directamente responsables de los costos de aprovisionamiento en primer lugar, y también puede aumentar su concienciación general de los costos.

Para mantenerse informado sobre el cumplimiento de los presupuestos existentes, puede crear y programar el envío regular de informes de AWS Budgets por correo electrónico a usted y a otras partes interesadas. También puede crear alertas de AWS Budgets basadas en los costos reales, que son de naturaleza reactiva, o en los costos previstos, lo que permite tener tiempo para implementar mitigaciones contra posibles sobrecostos. Puede recibir alertas cuando el costo o el uso reales superen un nivel determinado o si se prevé que superen el importe presupuestado.

Ajuste los procesos de elaboración de presupuestos y previsiones existentes para que sean más dinámicos, ya sea utilizando un algoritmo basado en las tendencias (que usa los costos históricos como entradas) o algoritmos basados en impulsores (por ejemplo, lanzamientos de productos nuevos, una expansión regional o nuevos entornos para cargas de trabajo), ya que son ideales para un entorno de gasto dinámico y variable. Una vez que haya determinado su previsión basada en las tendencias con el Explorador de costos o cualquier otra herramienta, use Calculadora de precios de AWS para estimar su caso de uso de AWS y los costos futuros en función del uso esperado (tráfico, solicitudes por segundo o instancias de Amazon EC2 necesarias).

Haga un seguimiento de la precisión de esa previsión, ya que los presupuestos deben establecerse en función de estos cálculos y estimaciones de la previsión. Supervise la precisión y la eficacia de las previsiones de costos de la nube integrados. Compare periódicamente los gastos reales con su previsión y haga los ajustes necesarios para mejorar la precisión de la previsión. Haga un seguimiento de la desviación de las previsiones y lleve a cabo un análisis de la causa raíz de la desviación notificada para tomar medidas y ajustar las previsiones.

Como se indica en COST01-BP02 Establecimiento de la colaboración entre los departamentos de finanzas y tecnología, es importante contar con asociaciones y cadencias entre los departamentos de TI y Finanzas y otras partes interesadas para garantizar que todos utilicen las mismas herramientas o procesos en aras de la coherencia. En los casos en que los presupuestos deban cambiar, aumente los puntos de contacto regulares para reaccionar a esos cambios más rápidamente.

Pasos para la implementación

- Defina el lenguaje de costos de la organización: cree un lenguaje común de costos de AWS dentro de la organización con múltiples dimensiones y agrupaciones. Asegúrese de que las partes interesadas comprenden la granularidad de las previsiones, los modelos de precios y el nivel de sus previsiones de costos.
- Analice las previsiones basadas en las tendencias: utilice herramientas de previsión basadas en tendencias, como AWS Cost Explorer y Amazon Forecast. Analice su costo de uso en múltiples dimensiones (por ejemplo, servicio, cuentas, etiquetas y categorías de costos).
- Análisis de las previsiones basadas en impulsores: identifique el efecto de los impulsores empresariales en el uso de la nube y haga previsiones para cada uno de ellos por separado para calcular el costo de uso esperado con antelación. Trabaje en estrecha colaboración con los propietarios de las unidades de negocio y las partes interesadas para conocer la repercusión en los nuevos impulsores y calcular los cambios de costos esperados a la hora de definir presupuestos precisos.
- Actualice los procesos existentes de elaboración de presupuestos y previsiones: en función de los métodos de previsión adoptados, como los basados en tendencias, los basados en impulsores empresariales o una combinación, defina sus procesos de elaboración de presupuestos y previsiones. Los presupuestos deben estar bien calculados, ser realistas y basarse en sus previsiones.
- Configure alertas y notificaciones: utilice las alertas de AWS Budgets y la detección de anomalías de costos para recibir alertas y notificaciones.

- Revisiones periódicas con las principales partes interesadas: por ejemplo, las partes interesadas en TI, finanzas, equipos de plataforma y otras áreas de la empresa deben alinearse con los cambios en la dirección y el uso de la empresa.

Recursos

Documentos relacionados:

- AWS Cost Explorer
- AWS Cost and Usage Report
- Forecasting with Cost Explorer
- Previsiones de Quick
- AWS Budgets

Videos relacionados:

- How can I use AWS Budgets to track my spending and usage
- AWS Cost Optimization Series: AWS Budgets

Ejemplos relacionados:

- Understand and build driver-based forecasting
- How to establish and drive a forecasting culture
- How to improve your cloud cost forecasting
- Using the right tools for your cloud cost forecasting

COST01-BP04 Implementación de conciencia de costos en los procesos organizativos

Implemente la conciencia de costos, cree transparencia y responsabilidad de los costos en los procesos nuevos y existentes que afecten al uso, y aproveche los procesos existentes para tomar conciencia de los costos. Implemente la conciencia de costos en la capacitación del personal.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La conciencia de costos debe implementarse en los procesos organizativos nuevos y existentes. Es una de las capacidades fundamentales, requisito previo para otras prácticas recomendadas. Se recomienda reutilizar y modificar los procesos existentes cuando sea posible, lo que minimiza el impacto en la agilidad y la velocidad. Informe de los costos de la nube a los equipos de Tecnología y a los responsables de la toma de decisiones de los equipos de Negocios y Finanzas para concienciar sobre los costos, y establezca indicadores clave de rendimiento (KPI) de eficiencia para las partes interesadas de finanzas y negocios. Las siguientes recomendaciones ayudarán a implementar la conciencia de costos en su carga de trabajo:

- Verifique que la administración de los cambios incluya la medición de los costos para cuantificar el efecto financiero de sus cambios. Esto ayuda a abordar de forma proactiva las preocupaciones relacionadas con los costos y destacar el ahorro de costos.
- Verifique que la optimización de costos sea un componente central de sus capacidades operativas. Por ejemplo, puede aprovechar los procesos existentes de administración de incidentes para investigar e identificar las causas raíz de las anomalías de los costos y el uso o costos excesivos.
- Acelere el ahorro de costos y la materialización del valor de negocio a través de la automatización o las herramientas. Al pensar en el costo de implementación, enmarque la conversación para que incluya un componente de retorno de la inversión (ROI) para justificar la inversión de tiempo o dinero.
- Asigne los costos de la nube mediante la aplicación de devoluciones o reembolsos de los gastos en la nube, incluidos los gastos en las opciones de compra basadas en el compromiso, los servicios compartidos y las compras en el mercado para impulsar el consumo de la nube teniendo siempre presentes los costos.
- Amplíe los programas de capacitación y desarrollo existentes para que incluyan la conciencia de costos en toda la organización. Se recomienda incluir capacitación y certificaciones continuas. Con ello logrará tener una organización capaz de autoadministrar los costos y el uso.
- Aproveche las herramientas nativas gratuitas de AWS como AWS Cost Anomaly Detection, AWS Budgets e informes de AWS Budgets.

Cuando las organizaciones adoptan sistemáticamente prácticas de administración financiera en la nube (CFM), esos comportamientos se arraigan en la forma de trabajar y la toma de decisiones. El resultado es una cultura que tiene más en cuenta los costos, desde los desarrolladores que diseñan una nueva aplicación nacida en la nube hasta los administradores financieros que analizan el retorno de estas nuevas inversiones en la nube.

Pasos para la implementación

- Identifique los procesos organizativos relevantes: cada unidad organizativa debe revisar sus procesos e identificar los procesos que afecten a los costos y el uso. Cualquier proceso que conlleve la creación o terminación de un recurso debe incluirse en la revisión. Debe buscar procesos que ayuden a tomar conciencia de los costos en su negocio, como la administración de incidentes y la capacitación.
- Establezca una cultura autosuficiente en materia de costos: asegúrese de que todas las partes interesadas pertinentes se alineen con la causa del cambio y el impacto como costo para que entiendan el costo de la nube. Esto permitirá a su organización establecer una cultura de innovación autosuficiente y sensibilizada con los costos.
- Actualice los procesos teniendo en cuenta los costos: cada proceso se modifica para adaptarse a los costos. El proceso puede requerir controles previos adicionales, como evaluar el impacto del costo, o controles posteriores que validen que se han producido los cambios esperados en el costo y el uso. La asistencia a procesos tales como la capacitación y la administración de incidentes puede ampliarse para incluir elementos de costo y uso.

Para obtener ayuda, contacte con los expertos de CFM a través de su equipo de Cuentas o explore los recursos y documentos relacionados a continuación.

Recursos

Documentos relacionados:

- Administración financiera en la nube de AWS

Ejemplos relacionados:

- Strategy for Efficient Cloud Cost Management
- Cost Control Blog Series #3: How to Handle Cost Shock
- A Beginner’s Guide to AWS Cost Management

COST01-BP05 Creación de informes y notificaciones sobre la optimización de costos

Establezca presupuestos para la nube y configure mecanismos para detectar anomalías en el uso. Configure las herramientas relacionadas para que proporcionen alertas de costo y uso respecto a objetivos predefinidos y reciba notificaciones cuando el uso supere esos objetivos. Organice

reuniones periódicas para analizar la rentabilidad de las cargas de trabajo y promover la conciencia de costos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Debe informar regularmente sobre la optimización de costos y el uso en su organización. Puede implementar sesiones dedicadas a examinar el rendimiento de costos o incluir la optimización de costos en los ciclos de preparación de informes operativos habituales para sus cargas de trabajo. Utilice los servicios y las herramientas para supervisar el rendimiento de los costos con regularidad e implementar oportunidades de ahorro de costos.

Consulte sus costo y uso con varios filtros y grado de detalle con AWS Cost Explorer, que proporciona paneles e informes, como costos por servicio o por cuenta, costos diarios o costos del marketplace. Haga un seguimiento de la evolución del costo y el uso según los presupuestos creados con AWS Budgets Reports.

Utilice AWS Budgets para establecer presupuestos personalizados para hacer un seguimiento de sus costos y uso, y responder rápidamente a las alertas recibidas por correo electrónico o las notificaciones del Amazon Simple Notification Service (Amazon SNS) si supera su límite. Establezca su período presupuestario preferido para que sea diario, mensual, trimestral o anual, y cree límites presupuestarios específicos para mantenerse informado sobre cómo los costos y el uso reales o previstos avanzan hacia el límite de su presupuesto. También puede configurar alertas y acciones con respecto a esas alertas para que se ejecuten automáticamente o mediante un proceso de aprobación cuando se supere un objetivo presupuestario.

Implemente notificaciones sobre el costo y el uso para garantizar que los cambios en el costo y el uso se puedan corregir rápidamente en caso de que sean inesperados. AWS Cost Anomaly Detection le permite reducir los costos imprevistos y mejorar el control sin ralentizar la innovación. AWS Cost Anomaly Detection identifica los gastos anómalos y las causas raíz, lo que ayuda a reducir el riesgo de sorpresas en la facturación. Con tres sencillos pasos, puede crear su propio monitor contextualizado y recibir alertas cuando se detecte cualquier gasto anómalo.

También puede usar Quick con datos de AWS Cost and Usage Report (CUR) para proporcionar informes altamente personalizados con datos más detallados. Quick le permite programar informes y recibir correos electrónicos de informes de costos periódicos para un historial de costos y uso u oportunidades de ahorro de costos. Consulte nuestra solución Cost Intelligence Dashboard (CID) basada en Quick, que ofrece visibilidad avanzada.

Use AWS Trusted Advisor, que proporciona orientación para verificar si los recursos aprovisionados están en consonancia con las prácticas recomendadas de AWS para la optimización de costos.

Compare sus recomendaciones de Savings Plans a través de gráficos visuales con sus costos y uso detallados. En los gráficos por hora se muestra el gasto bajo demanda junto con el compromiso de Savings Plans recomendado, lo que proporciona información sobre los ahorros, la cobertura de Savings Plans y la utilización de Savings Plans estimados. Esto ayuda a las organizaciones a comprender cómo sus Savings Plans se aplican a cada hora de gasto sin tener que invertir tiempo y recursos en la creación de modelos para analizar el gasto.

Crear periódicamente informes que contengan un resumen de los Savings Plans, las instancias reservadas y recomendaciones de tamaño adecuado de Amazon EC2 desde AWS Cost Explorer para empezar a reducir el costo asociado a las cargas de trabajo en estado estable, los recursos inactivos y los infrautilizados. Identifique y recupere el gasto asociado a los residuos de la nube para los recursos que se implementan. Los residuos de la nube se producen cuando se crean recursos de tamaño incorrecto o se observan patrones de uso diferentes a los esperados. Siga las prácticas recomendadas de AWS para reducir los residuos o pida a su socio y equipo de Cuentas que lo ayuden a optimizar y ahorrar en relación con los costos de la nube.

Genere informes con regularidad para mejorar las opciones de compra de sus recursos y reducir los costos unitarios de sus cargas de trabajo. Las opciones de compra, como los Savings Plans, las instancias reservadas o las instancias de spot de Amazon EC2, ofrecen el mayor ahorro de costos para las cargas de trabajo con tolerancia a errores y permiten a las partes interesadas (propietarios de la empresa y equipos financieros y técnicos) formar parte de estas conversaciones de compromiso.

Comparta los informes que contengan oportunidades o anuncios de nuevas versiones que puedan ayudarlo a reducir el costo total de propiedad (TCO) de la nube. Adopte nuevos servicios, regiones, características, soluciones o nuevas formas de lograr una mayor reducción de costos.

Pasos para la implementación

- Configuración de AWS Budgets: configure AWS Budgets en todas las cuentas de su carga de trabajo. Establezca un presupuesto para el gasto general de la cuenta y un presupuesto para la carga de trabajo con etiquetas.
  - Well-Architected Labs: Cost and Governance Usage
- Informe de la optimización de costos: defina un ciclo habitual para tratar y analizar la eficiencia de la carga de trabajo. Utilice las métricas establecidas y notifique las métricas alcanzadas y el costo

para alcanzarlas. Identifique y corrija las tendencias negativas, así como las tendencias positivas que puede promover en su organización. La preparación de informes debe incluir a representantes de los equipos y propietarios de las aplicaciones, de las finanzas y los principales responsables de la toma de decisiones con respecto al gasto en la nube.

Recursos

Documentos relacionados:

- AWS Cost Explorer
- AWS Trusted Advisor
- AWS Budgets
- AWS Cost and Usage Report
- AWS Budgets Prácticas recomendadas de
- Análisis de Amazon S3

Ejemplos relacionados:

- Key ways to start optimizing your AWS cloud costs

COST01-BP06 Supervisión proactiva de los costos

Implemente herramientas y paneles para supervisar los costos de la carga de trabajo de forma proactiva. Revise periódicamente los costos con herramientas configuradas o listas para usar, no se limite a mirar los costos y las categorías cuando reciba las notificaciones. Supervisar y analizar los costos de forma proactiva ayuda a identificar las tendencias positivas y permite promoverlas en toda la organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Se recomienda supervisar los costos y el uso de forma proactiva en la organización, no solo cuando surjan anomalías o excepciones. Disponer de unos paneles muy visibles en la oficina o en el entorno de trabajo permite a las personas clave tener acceso a la información que necesitan y transmite la idea de que la organización se centra en la optimización de costos. Los paneles visibles permiten promover de forma activa los resultados de éxito e implementarlos en toda la organización.

Cree una rutina diaria o frecuente para utilizar AWS Cost Explorer o cualquier otro panel, como Amazon Quick, para ver los costos y analizarlos de forma proactiva. Analice el uso y los costos de los servicios de AWS por cuenta de AWS, carga de trabajo o servicio específico de AWS con agrupaciones y filtros, y valide si son los esperados o no. Utilice la granularidad por hora y recurso y las etiquetas para filtrar e identificar los costos incurridos de los principales recursos. También puede crear sus propios informes con Cost Intelligence Dashboard, una solución de Amazon Quick creada por arquitectos de soluciones de AWS, y comparar sus presupuestos con el costo y el uso reales.

Pasos para la implementación

- Informe de la optimización de costos: defina un ciclo habitual para tratar y analizar la eficiencia de la carga de trabajo. Utilice las métricas establecidas y notifique las métricas alcanzadas y el costo para alcanzarlas. Identifique y corrija las tendencias negativas e identifique las tendencias positivas que quiere promover en su organización. La administración de informes debe implicar a los representantes de los equipos de Aplicaciones y los propietarios, así como de los departamentos de Finanzas y de Dirección.
- Creación y activación de presupuestos de AWS Budgets de granularidad diaria del costo y el uso a fin de tomar medidas oportunas para evitar posibles sobrecostos: AWS Budgets le permite configurar las notificaciones de alerta para mantenerse informado en caso de que alguno de sus tipos de presupuesto supere los umbrales preconfigurados. La mejor manera de aprovechar AWS Budgets es establecer los costos y el uso previstos como límites, de modo que todo lo que supere los presupuestos se considere un gasto excesivo.
- Creación de AWS Cost Anomaly Detection para la supervisión de costos: AWS Cost Anomaly Detection utiliza tecnología avanzada de machine learning para identificar los gastos anómalos y las causas que los originan para que pueda adoptar medidas rápidamente. Le permite configurar monitores de costos que definen los segmentos de gastos que desea evaluar (por ejemplo, servicios individuales de AWS, cuentas de miembro, etiquetas de asignación de costos y categorías de costos) y le permite establecer cuándo, dónde y cómo recibir sus notificaciones de alerta. Para cada monitor, adjunte varias suscripciones de alerta para los propietarios de negocios y los equipos de tecnología, que incluyan un nombre, un umbral de impacto de costos y la frecuencia de las alertas (alertas individuales, resumen diario, resumen semanal) para cada suscripción.
- Use AWS Cost Explorer o integre de sus datos de AWS Cost and Usage Report (CUR) con los paneles de Amazon Quick para visualizar los costos de su organización: AWS Cost Explorer cuenta con una interfaz fácil de usar que le permite visualizar, comprender y administrar los costos y el uso de AWS a lo largo del tiempo. Cost Intelligence Dashboard es un panel personalizable y

accesible para ayudar a crear la base de su propia herramienta de administración y optimización de costos.

Recursos

Documentos relacionados:

- AWS Budgets
- AWS Cost Explorer
- Daily Cost and Usage Budgets
- AWS Cost Anomaly Detection

Ejemplos relacionados:

- AWS Cost Anomaly Detection Alert with Slack

COST01-BP07 Seguimiento de la información sobre las nuevas versiones de los servicios

Consulte regularmente con expertos o socios de AWS qué servicios y características proporcionan un costo inferior. Revise los blogs de AWS y otras fuentes de información.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

AWS está agregando constantemente nuevas capacidades para que pueda aprovechar las últimas tecnologías para experimentar e innovar más rápidamente. Puede implementar nuevos servicios y características de AWS para incrementar la rentabilidad de su carga de trabajo. Consulte periódicamente Administración de costos de AWS, el blog de novedades de AWS, el blog de administración de costos de AWS y Novedades de AWS para obtener información sobre las nuevas versiones de servicios y características. En las publicaciones de novedades se ofrece un breve resumen de todos los anuncios de servicios, características y ampliación de regiones de AWS a medida que se publican.

Pasos para la implementación

- Suscripción a los blogs: vaya a las páginas de los blogs de AWS y suscríbase al blog de novedades y a otros blogs pertinentes. Puede registrarse en la página de preferencias de comunicación con su dirección de correo electrónico.

- Suscripción a las novedades de AWS: consulte periódicamente el blog de novedades de AWS y Novedades de AWS para obtener información sobre los nuevos lanzamientos de servicios y características. Suscríbase a la fuente RSS o con su correo electrónico para seguir los anuncios y lanzamientos.
- Seguimiento de las reducciones de precios de AWS: la reducción periódica de los precios de todos nuestros servicios ha sido una forma habitual de AWS de trasladar a nuestros clientes las eficiencias económicas obtenidas gracias a nuestra escala. A 20 de septiembre de 2023, AWS ha reducido los precios 134 veces desde 2006. Si tiene alguna decisión comercial pendiente por cuestiones de precio, puede volver a revisarla después de las reducciones de precios y las nuevas integraciones de servicios. Puede obtener información sobre las iniciativas de reducción de precios anteriores, incluidas las instancias de Amazon Elastic Compute Cloud (Amazon EC2), en la categoría de reducción de precios del blog de novedades de AWS.
- Eventos y reuniones de AWS: asista a la cumbre local de AWS y a cualquier reunión local con otras organizaciones de su zona. Si no puede asistir en persona, intente asistir a los eventos virtuales para conocer mejor a los expertos de AWS y los casos empresariales de otros clientes.
- Reunión con su equipo de Cuentas: programe una cadencia regular con su equipo de cuentas, reúnase con él y trate sobre las tendencias del sector y los servicios de AWS. Hable con el administrador de cuentas, el arquitecto de soluciones y el equipo de Asistencia.

Recursos

Documentos relacionados:

- Administración de costos de AWS
- Novedades de AWS
- Blog de noticias de AWS

Ejemplos relacionados:

- Amazon EC2 – 15 Years of Optimizing and Saving Your IT Costs
- AWS News Blog - Price Reduction

### COST01-BP08 Creación de una cultura sensibilizada con los costos

Implemente cambios o programas en la organización para crear una cultura sensibilizada con los costos. Se recomienda empezar discretamente, y a medida que crezcan las capacidades y el uso de la nube por parte de la empresa, implementar programas grandes y de gran alcance.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

#### Guía para la implementación

Afianzar una cultura de conciencia de los costos permite mejorar la optimización de costos y la administración financiera en la nube (operaciones financieras, centro de excelencia en la nube, equipos de operaciones en la nube, etc.) a través de las prácticas recomendadas llevadas a cabo de forma orgánica y descentralizada en toda la organización. La conciencia de costos permite obtener grandes niveles de capacidad en la organización con un esfuerzo mínimo, en comparación con un enfoque centralizado y descendente.

La conciencia de costos en la computación en la nube, especialmente para los principales impulsores de costos en la computación en la nube, permite a los equipos comprender los resultados esperados de cualquier cambio en la perspectiva de los costos. Los equipos que acceden a los entornos de la nube deben conocer los modelos de precios y la diferencia entre los centros de datos tradicionales en las instalaciones y la computación en la nube.

La principal ventaja de una cultura sensibilizada con los costos es que los equipos de Tecnología los optimizan de forma proactiva y continua (por ejemplo, se consideran un requisito no funcional a la hora de diseñar nuevas cargas de trabajo o de hacer cambios en las existentes) en lugar de llevar a cabo optimizaciones de costos reactivas según sea necesario.

Aplicar unos pequeños cambios en la cultura puede tener un gran impacto en la eficiencia de las cargas de trabajo actuales y futuras. Ejemplos:

- Dar visibilidad y sensibilizar a los equipos de ingeniería para que comprendan lo que hacen y su impacto en términos de costos.
- Ludificar los costos y el uso en toda la organización. Esto se puede hacer con un panel visible para todo el personal o mediante un informe que compare los costos y el uso normalizados de los diferentes equipos (por ejemplo, costo por carga de trabajo y costo por transacción).
- Reconocer la rentabilidad. Premie los logros voluntarios o espontáneos de optimización de costos de forma pública o privada y aprenda de los errores para no repetirlos en el futuro.

- Crear requisitos organizativos descendentes para que las cargas de trabajo se ejecuten con presupuestos predefinidos.
- Cuestionar los requisitos empresariales de los cambios y el impacto de los costos de los cambios solicitados en la infraestructura de la arquitectura o la configuración de la carga de trabajo para asegurarse de que se paga solo lo que se necesita.
- Asegurarse de que el planificador del cambio es consciente de los cambios previstos que tienen un impacto en los costos y que las partes interesadas los confirmen para obtener resultados empresariales de forma rentable.

Pasos para la implementación

- Informe de los costos de la nube a los equipos de Tecnología: para aumentar la conciencia de costos y establecer KPI de eficiencia para las partes interesadas de las finanzas y la empresa.
- Informe a las partes interesadas o a los miembros del equipo sobre los cambios previstos: cree un punto en el orden del día para debatir los cambios previstos y el impacto del costo-beneficio en la carga de trabajo durante las reuniones semanales sobre cambios.
- Reunión con el equipo de Cuentas: establezca una cadencia de reuniones regular con su equipo de cuentas, y trate las tendencias del sector y los servicios de AWS. Hable con el administrador de cuentas, el arquitecto y el equipo de Asistencia.
- Comparta historias de éxito: comparta historias de éxito sobre la reducción de costos para cualquier carga de trabajo, Cuenta de AWS u organización para crear una actitud positiva y un estímulo en torno a la optimización de costos.
- Formación asegúrese de que los equipos técnicos o los miembros del equipo reciban formación para conocer los costos de los recursos en Nube de AWS.
- Eventos y reuniones de AWS:  asista a las cumbres locales de AWS y a cualquier reunión local con otras organizaciones de su zona.
- Suscripción a los blogs: vaya a las páginas de los blogs de AWS y suscríbase al blog de novedades y a otros blogs pertinentes para seguir los nuevos lanzamientos, implementaciones, ejemplos y cambios que comparte AWS.

Recursos

Documentos relacionados:

- Blog de AWS

- Administración de costos de AWS
- Blog de noticias de AWS

Ejemplos relacionados:

- Administración financiera en la nube de AWS

COST01-BP09: cuantificación del valor empresarial de la optimización de costos

Cuantificar el valor empresarial de la optimización de costos le permite comprender todos los beneficios para su organización. Dado que la optimización de costos es una inversión necesaria, cuantificar el valor empresarial le permite explicar el retorno de la inversión a las partes interesadas. Cuantificar el valor empresarial lo puede ayudar a lograr mayor aceptación de las partes interesadas para hacer inversiones futuras en optimización de costos y, además, le proporciona un marco para medir los resultados de las actividades de optimización de costos de la organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cuantificar el valor empresarial implica determinar los beneficios que las empresas obtienen de las acciones y decisiones que toman. El valor empresarial puede ser tangible (por ejemplo, una reducción de los gastos o un aumento de los beneficios) o intangible (por ejemplo, la mejora de la reputación de la marca o el aumento de la satisfacción del cliente).

Para cuantificar el valor empresarial que se produce con la optimización de costos, hay que determinar cuánto valor o beneficio obtiene de sus esfuerzos por gastar de manera más eficiente. Por ejemplo, si una empresa gasta 100 000 USD en implementar una carga de trabajo en AWS y, posteriormente, la optimiza, el nuevo costo pasa a ser de solo 80 000 USD sin sacrificar la calidad ni el rendimiento. En este escenario, el valor empresarial cuantificado de la optimización de costos sería un ahorro de 20 000 USD. No obstante, más allá del ahorro, la empresa también podría cuantificar el valor en términos de tiempos de entrega más rápidos, una mayor satisfacción del cliente u otras métricas que se deriven de los esfuerzos de optimización de costos. Las partes interesadas deben tomar decisiones sobre el valor potencial de la optimización de costos, el costo de optimizar la carga de trabajo y el valor de retorno.

Además de informar sobre los ahorros de la optimización de costos, se recomienda cuantificar el valor adicional conseguido. Los beneficios de la optimización de costos se suelen cuantificar en

términos de menos costos por resultado empresarial. Por ejemplo, puede cuantificar los ahorros de costos de Amazon Elastic Compute Cloud (Amazon EC2) al comprar Savings Plans, lo que reduce los costos y mantiene los niveles de producción de la carga de trabajo. Puede cuantificar reducciones de costos en el gasto de AWS cuando se eliminan las instancias de Amazon EC2 inactivas o cuando se eliminan volúmenes de Amazon Elastic Block Store (Amazon EBS) no asociados.

Sin embargo, la optimización de costos tiene muchos más beneficios, aparte de reducir o evitar costos. Plantéese capturar más datos para medir las mejoras en la eficiencia y el valor empresarial.

Pasos para la implementación

- Evaluación de los beneficios empresariales: este es el proceso de analizar y ajustar los costos de la Nube de AWS de manera que se maximice el beneficio recibido por cada dólar gastado. En lugar de centrarse en la reducción de costos sin valor empresarial, considere los beneficios empresariales y el retorno de la inversión de la optimización de costos, lo que puede aportar más valor al dinero que gasta. Se trata de gastar con prudencia y llevar a cabo inversiones y gastos en las áreas que tengan el mejor rendimiento.
- Análisis de los costos de AWS de las previsiones: las previsiones ayudan a las partes interesadas de finanzas a establecer expectativas con otras partes interesadas internas y externas de la organización y pueden mejorar la previsibilidad financiera de la organización. AWS Cost Explorer se puede utilizar para hacer una previsión de los costos y el uso.

Recursos

Documentos relacionados:

- Fundamentos económicos de la Nube de AWS
- Blog de AWS
- Administración de costos de AWS
- Blog de noticias de AWS
- Documento técnico Pilar de fiabilidad: Well-Architected
- Explorador de costos de AWS

Videos relacionados:

- Unlock Business Value with Windows on AWS

Ejemplos relacionados:

- Measuring and Maximizing the Business Value of Customer 360
- The Business Value of Adopting Amazon Web Services Managed Databases
- The Business Value of Amazon Web Services for Independent Software Vendors
- Business Value of Cloud Modernization
- The Business Value of Migration to Amazon Web Services

## Conocimiento del gasto y del uso

Preguntas

- COST 2. ¿Cómo controla el uso?
- COST 3. ¿Cómo supervisa sus costos y su uso?
- COST 4. ¿Cómo retira los recursos?

### COST 2. ¿Cómo controla el uso?

Establezca políticas y mecanismos para comprobar que se incurre en costos apropiados mientras se alcanzan los objetivos. Al emplear un enfoque basado en evaluar la situación, puede innovar sin gastar de más.

Prácticas recomendadas

- COST02-BP01 Desarrollo de políticas basadas en los requisitos de su organización
- COST02-BP02 Implementación de objetivos y metas
- COST02-BP03 Implementación de una estructura de cuentas
- COST02-BP04 Implementación de grupos y roles
- COST02-BP05 Implementación de controles de costos
- COST02-BP06 Seguimiento del ciclo de vida de los proyectos

#### COST02-BP01 Desarrollo de políticas basadas en los requisitos de su organización

Desarrolle políticas que definan la forma en que su organización administra los recursos e inspecciónelas periódicamente. Las políticas deben abarcar los aspectos de costo de los recursos y las cargas de trabajo, como su creación, modificación y retirada durante la vida útil del recurso.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Es fundamental comprender los costos y lo que impulsa a su organización para administrar eficazmente el costo y el uso e identificar las oportunidades de reducción de costos. Las organizaciones suelen operar múltiples cargas de trabajo que ejecutan varios equipos. Estos equipos pueden estar en diferentes unidades organizativas, cada una con su propio flujo de ingresos. La capacidad de atribuir los costos de los recursos a las cargas de trabajo, a la organización individual o a los propietarios de los productos impulsa un comportamiento de uso eficiente y contribuye a reducir los residuos. La monitorización precisa de los costos y el uso le ayuda a comprender el nivel de optimización de una carga de trabajo, así como la rentabilidad de las unidades y los productos de la organización. Este conocimiento permite tomar decisiones más fundamentadas sobre dónde asignar los recursos dentro de la organización. El conocimiento del uso en todos los niveles de la organización es clave para impulsar el cambio, ya que el cambio en el uso impulsa los cambios en el costo. Considere la posibilidad de adoptar un enfoque multifacético para conocer su uso y sus gastos.

El primer paso para aplicar la gobernanza es utilizar los requisitos de su organización para desarrollar políticas de uso de la nube. Estas políticas definen cómo su organización utiliza la nube y cómo se administran los recursos. Las políticas deben tratar todos los aspectos de los recursos y las cargas de trabajo que tienen que ver con el costo o el uso, como su creación, modificación y retirada durante la vida útil del recurso. Verifique que se siguen e implementan las políticas y los procedimientos ante cualquier cambio en un entorno en la nube. Durante las reuniones de administración de cambios de TI, formule preguntas para averiguar el impacto en los costos de los cambios previstos, tanto si aumentan como si disminuyen, la justificación empresarial y el resultado esperado.

Las políticas deben ser sencillas para que se comprendan fácilmente y puedan implementarse con eficacia en toda la organización. Las políticas también deben ser fáciles de seguir e interpretar (para que se usen) y específicas (para que no haya malinterpretaciones entre los equipos). Además, deben inspeccionarse periódicamente (igual que nuestros mecanismos) y actualizarse a medida que cambien las condiciones empresariales o las prioridades de los clientes, ya que esto podría hacer que la política quedara obsoleta.

Empiece con políticas amplias y generales, como la región geográfica que se usará o las horas del día en las que deben ejecutarse los recursos. Mejore gradualmente las políticas para las distintas unidades organizativas y cargas de trabajo. Entre las políticas más comunes se incluyen los servicios y las características que pueden utilizarse (por ejemplo, el almacenamiento de menor rendimiento en

los entornos de prueba o de desarrollo), los tipos de recursos que pueden utilizar los distintos grupos (por ejemplo, el mayor tamaño de recurso en una cuenta de desarrollo es el medio) y durante cuánto tiempo estarán en uso estos recursos (temporalmente, a corto plazo o durante un periodo de tiempo específico).

Ejemplo de política

A continuación, tenemos un ejemplo de política que puede utilizar para crear sus propias políticas de gobernanza en la nube que se centren en la optimización de costos. Asegúrese de ajustar la política en función de los requisitos de su organización y de las peticiones de las partes interesadas.

- Nombre de la política: defina un nombre claro, como “Política de optimización de recursos y reducción de costos”.
- Propósito: explique por qué se debe utilizar esta política y cuál es el resultado esperado. El objetivo de esta política es verificar que se requiere un costo mínimo para implementar y ejecutar la carga de trabajo deseada con el fin de cumplir los requisitos empresariales.
- Ámbito: defina claramente quién debe usar esta política y cuándo debe usarse; por ejemplo, podría indicar que el equipo X de DevOps debe usar esta política en los clientes de la región us-east para el entorno X (de producción o no producción).

Declaración de la política

1. Seleccione us-east-1 o varias regiones de us-east en función del entorno y los requisitos empresariales de su carga de trabajo (desarrollo, pruebas de aceptación de los usuarios, preproducción o producción).
2. Programe instancias de Amazon EC2 y Amazon RDS para que se ejecuten entre las seis de la mañana y las ocho de la tarde (hora estándar del este [EST]).
3. Detenga todas las instancias de Amazon EC2 no utilizadas después de ocho horas y las instancias de Amazon RDS no utilizadas después de 24 horas de inactividad.
4. Termine todas las instancias de Amazon EC2 no utilizadas después de 24 horas de inactividad en entornos que no sean de producción. Recuérdele al propietario de la instancia de Amazon EC2 (basándose en las etiquetas) que revise las instancias de Amazon EC2 detenidas en producción e infórmele de que sus instancias de Amazon EC2 se terminarán en un plazo de 72 horas si no están en uso.
5. Utilice una familia y un tamaño de instancias genéricos, como m5.large, y luego cambie el tamaño de la instancia en función del uso de la CPU y la memoria mediante AWS Compute Optimizer.

6. Priorice el uso del escalado automático para ajustar dinámicamente la cantidad de instancias en ejecución en función del tráfico.
7. Utilice instancias de spot para cargas de trabajo no críticas.
8. Revise los requisitos de capacidad para confirmar Savings Plans o instancias reservadas para cargas de trabajo predecibles e informe al equipo de Administración financiera en la nube.
9. Utilice políticas de ciclo de vida de Amazon S3 para mover los datos a los que se accede con poca frecuencia a niveles de almacenamiento más económicos. Si no se ha definido ninguna política de retención, utilice Amazon S3 Intelligent-Tiering para mover los objetos al nivel de archivado automáticamente.
10. Supervise el uso de los recursos y configure alarmas para activar eventos de escalado mediante Amazon CloudWatch.
11. Para cada Cuenta de AWS, utilice AWS Budgets para establecer presupuestos de costos y uso para su cuenta en función del centro de costos y las unidades de negocios.
12. Si usa AWS Budgets para establecer presupuestos de costos y uso para su cuenta, puede resultarle más fácil controlar sus gastos y evitar facturas inesperadas, lo que te permitirá controlar mejor sus costos.

Procedimiento: proporcione procedimientos detallados para implementar esta política o consulte otros documentos en los que se describa cómo implementar cada declaración de la política. En esta sección, se deben proporcionar instrucciones paso a paso para cumplir los requisitos de la política.

Para implementar esta política, puede utilizar diversas herramientas o reglas de AWS Config de terceros para comprobar si se cumple la declaración de la política y activar medidas de corrección automatizadas mediante funciones de AWS Lambda. También puede utilizar AWS Organizations para hacer cumplir la política. Además, debe revisar periódicamente el uso de sus recursos y ajustar la política según sea necesario para comprobar que sigue satisfaciendo las necesidades de su empresa.

Pasos para la implementación

- Reunión con las partes interesadas: para desarrollar políticas, pida a las partes interesadas (oficinas de la empresa en la nube, ingenieros o responsables de la toma de decisiones funcionales para la aplicación de las políticas) de su organización que especifiquen sus requisitos y los documenten. Adopte un enfoque iterativo; para ello, empiece con un enfoque amplio y vaya reduciendo hasta llegar a las unidades más pequeñas en cada paso. Entre los miembros del equipo se encuentran los que tienen un interés directo en la carga de trabajo, como las unidades

organizativas o los propietarios de las aplicaciones, además de los grupos de asistencia, como los equipos de Seguridad y Finanzas.

- Obtención de la confirmación: asegúrese de que los equipos se ponen de acuerdo en las políticas sobre quién puede acceder e implementar en la Nube de AWS. Asegúrese de que siguen las políticas de su organización y confirme que sus creaciones de recursos se ajustan a las políticas y procedimientos acordados.
- Creación de sesiones de formación de incorporación: pida a los nuevos miembros de la organización que completen los cursos de capacitación de incorporación para concienciar de los costos y que conozcan los requisitos de la organización. Es posible que asuman políticas diferentes debido a su experiencia anterior o que no piensen en ellas en absoluto.
- Definición de ubicaciones para la carga de trabajo: defina dónde opera su carga de trabajo, incluido el país y la zona dentro del país. Esta información se utiliza para la asignación de Regiones de AWS y zonas de disponibilidad.
- Definición y agrupación de los servicios y los recursos: defina los servicios que requieren las cargas de trabajo. Para cada servicio, especifique los tipos, el tamaño y el número de recursos necesarios. Defina grupos para los recursos por función, como servidores de aplicaciones o almacenamiento de bases de datos. Los recursos pueden pertenecer a varios grupos.
- Definición y agrupación de los usuarios por función: defina a los usuarios que interactúan con la carga de trabajo; para ello, céntrese en lo que hacen y en cómo utilizan la carga de trabajo, no en quiénes son o en su puesto en la organización. Agrupe usuarios o funciones similares. Puede utilizar las políticas administradas por AWS como guía.
- Definición de las acciones: mediante las ubicaciones, los recursos y los usuarios identificados anteriormente, defina las acciones que requiere cada uno de ellos para lograr los resultados de la carga de trabajo a lo largo de su vida útil (desarrollo, funcionamiento y retirada). Identifique las acciones en función de los grupos, y no de los elementos individuales de los grupos, en cada ubicación. Empiece a grandes rasgos con la lectura o la escritura y, después, vaya reduciendo hasta llegar a las acciones específicas para cada servicio.
- Definición del periodo de revisión: las cargas de trabajo y los requisitos organizativos pueden cambiar con el tiempo. Defina el calendario de revisión de la carga de trabajo para asegurarse de que se mantiene alineado con las prioridades organizativas.
- Documentación de las políticas: verifique que las políticas que se han definido sean accesibles tal y como lo requiere su organización. Estas políticas se utilizan para implementar, mantener y auditar el acceso de sus entornos.

Recursos

Documentos relacionados:

- Change Management in the Cloud
- Políticas administradas de AWS para funciones de trabajo
- Estrategia de facturación de varias cuentas de AWS
- Actions, Resources, and Condition Keys for AWS Services
- Administración y gobernanza en AWS
- Control access to Regiones de AWS using IAM policies
- Regiones y zonas de disponibilidad de la infraestructura global

Videos relacionados:

- AWS Management and Governance at Scale

COST02-BP02 Implementación de objetivos y metas

Implemente objetivos de costos y uso para la carga de trabajo. Los objetivos son una guía de resultados previstos para la organización. Las metas proporcionan resultados medibles específicos que se deben alcanzar para las cargas de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Desarrolle objetivos y metas de costos y uso para su organización. Para una organización en crecimiento en AWS, es importante establecer objetivos de optimización de costos y hacer un seguimiento de ellos. Estos objetivos o indicadores clave de rendimiento(KPI) pueden incluir aspectos como el porcentaje del gasto bajo demanda o la adopción de ciertos servicios optimizados, como instancias de AWS Graviton o tipos de volúmenes gp3 de EBS. Establezca objetivos cuantificables y factibles que lo ayuden a medir las mejoras de la eficiencia, lo cual es importante para las operaciones empresariales. Los objetivos son una guía sobre los resultados esperados para su organización.

Las metas proporcionan resultados cuantificables específicos que se deben alcanzar. En resumen, un objetivo es la dirección en la que quiere ir y la meta es hasta dónde ir en esa dirección y cuándo

debe lograrse ese objetivo (mediante una orientación específica, cuantificable, asignable, realista y oportuna, o SMART, por sus siglas en inglés). Un ejemplo de un objetivo es que el uso de la plataforma debería incrementarse de forma significativa con tan solo un ligero incremento (no lineal) del costo. Un ejemplo de meta es un incremento del 20 % del uso de la plataforma con un incremento de menos del 5 % de los costos. Otro objetivo común es que las cargas de trabajo deben ser más eficientes cada seis meses. La meta correspondiente sería que las métricas de costo por empresa disminuyan un 5 % cada seis meses. Use las métricas correctas y establezca KPI calculados para su organización. Puede empezar con KPI básicos e ir evolucionando en función de las necesidades de la empresa.

Un objetivo de la optimización de costos es aumentar la eficiencia de la carga de trabajo, lo que supone reducir el costo por resultado empresarial de la carga de trabajo con el tiempo. Implemente este objetivo para todas las cargas de trabajo y fíjese una meta, como un aumento del cinco por ciento de la eficacia cada seis meses a un año. En la nube, puede lograrlo estableciendo capacidades en la optimización de costos, así como con el lanzamiento de nuevos servicios y características.

Las metas son los puntos de referencia cuantificables que desea alcanzar para cumplir sus objetivos y los puntos de referencia le permiten comparar sus resultados reales con una meta. Establezca puntos de referencia con KPI para el costo unitario de los servicios de computación (por ejemplo, adopción de Spot, adopción de Graviton, últimos tipos de instancias y cobertura bajo demanda), los servicios de almacenamiento (por ejemplo, adopción de EBS GP3, instantáneas obsoletas de EBS y almacenamiento de Amazon S3 Standard) o el uso de los servicios de bases de datos (por ejemplo, motores de código abierto de RDS, adopción de Graviton y cobertura bajo demanda). Estos puntos de referencia y KPI pueden ayudarlo a verificar que utiliza los servicios de AWS de la manera más rentable.

En la siguiente tabla se proporciona una lista de métricas de AWS estándar como referencia. Cada organización puede tener diferentes valores meta para estos KPI.

| Categoría | KPI (%) | Descripción |
| --- | --- | --- |
| Computación | Cobertura de uso de EC2 | Instancias de EC2 (en costo u horas) que utilizan Savings Plans + instancias reservadas + spot en comparación con el total (en costo u horas) de las instancias de EC2 |

| Categoría | KPI (%) | Descripción |
| --- | --- | --- |
| Computación | Cómputo del uso de Savings Plans o instancias reservadas | Horas utilizadas de Savings Plans o instancias reservadas en comparación con el total de horas disponibles de Savings Plans o instancias reservadas |
| Computación | Costo de EC2/hora | Costo de EC2 dividido por el número de instancias de EC2 que se ejecutan en esa hora |
| Computación | Costo de vCPU | Costo por vCPU para todas las instancias |
| Computación | Última generación de instancias | Porcentaje de instancias en Graviton (u otros tipos de instancias de generación moderna) |
| Base de datos | Cobertura de RDS | Instancias de RDS (en costo u horas) que utilizan instancias reservadas en comparación con el total (en costo u horas) de las instancias de RDS |
| Base de datos | Utilización de RDS | Horas utilizadas de instancias reservadas en comparación con el total de horas disponibles de instancias reservadas |
| Base de datos | Tiempo de actividad de RDS | El costo de RDS dividido por el número de instancias de RDS que se ejecuten en esa hora |

| Categoría | KPI (%) | Descripción |
| --- | --- | --- |
| Base de datos | Última generación de instancias | Porcentaje de instancias en Graviton (u otros tipos modernos de instancias) |
| Almacenamiento | Uso del almacenamiento | Costo de almacenamiento optimizado (por ejemplo, Glacier, archivo profundo o acceso poco frecuente) dividido por el costo total de almacenamiento |
| Etiquetado | Recursos sin etiquetar | Explorador de costos:<br><br>1. Filtre créditos, descuentos, impuestos, reembolsos y mercados, y copie el último costo mensual.<br><br>2. Seleccione Mostrar solo los recursos sin etiquetar en el Explorador de costos.<br><br>3. Divida la cantidad de recursos sin etiquetar por el costo mensual. |

Utilizando esta tabla, incluya los valores meta o de referencia, que deberán calcularse en función de los objetivos de la organización. Debe medir determinadas métricas de su negocio y conocer el resultado empresarial de esa carga de trabajo para definir unos indicadores clave de rendimiento (KPI) precisos y realistas. Cuando evalúe las métricas de rendimiento dentro de una organización, distinga entre diferentes tipos de métricas que sirven para propósitos distintos. Estas métricas miden principalmente el rendimiento y la eficiencia de la infraestructura técnica en lugar de la repercusión global en el negocio directamente. Por ejemplo, podrían hacer un seguimiento de los tiempos de respuesta del servidor, la latencia de la red o el tiempo de actividad del sistema. Estas métricas son cruciales para evaluar cómo admite la infraestructura las operaciones técnicas de la

organización. Sin embargo, no proporcionan información directa de objetivos empresariales más amplios, como la satisfacción del cliente, el crecimiento de los ingresos o la cuota de mercado. Para tener un conocimiento global del rendimiento empresarial, complemente estas métricas de eficiencia con métricas empresariales estratégicas que estén directamente relacionadas con los resultados empresariales.

Establezca una visibilidad casi en tiempo real de los KPI y las oportunidades de ahorro relacionadas y haga un seguimiento del progreso a lo largo del tiempo. Para empezar a definir los objetivos de los KPI y hacer un seguimiento de ellos, recomendamos utilizar el panel de KPI de Cloud Intelligence Dashboards (CID). En función de los datos del Informe de costos y uso (CUR), el panel de KPI proporciona una serie de KPI de optimización de costos recomendados con la capacidad de establecer objetivos personalizados y hacer un seguimiento de su progreso a lo largo del tiempo.

Si dispone de otras soluciones para establecer los objetivos de los KPI y hacer un seguimiento de ellos, asegúrese de que todas las partes interesadas de la administración financiera en la nube de su organización adopten esos métodos.

Pasos para la implementación

- Defina los niveles de uso esperados: para empezar, céntrese en los niveles de uso. Contacte con los propietarios de aplicaciones, los equipos de Marketing y otros equipos importantes de la empresa para conocer los niveles de uso esperados de la carga de trabajo. ¿Cómo podría cambiar la demanda de los clientes con el tiempo y qué podría cambiar debido a los incrementos de temporada o a las campañas de marketing?
- Defina los recursos y los costos de las cargas de trabajo: una vez definidos los niveles de uso, se deben cuantificar los cambios en los recursos de las cargas de trabajo necesarios para ajustarse a dichos niveles de uso. Es posible que tenga que incrementar el tamaño o el número de recursos para un componente de la carga de trabajo, incrementar la transferencia de datos o cambiar los componentes de las cargas de trabajo por un servicio distinto en un nivel determinado. Especifique los costos en cada uno de estos puntos importantes y prediga el cambio en el costo cuando se produzca un cambio en el uso.
- Defina los objetivos empresariales: debe combinar el resultado de los cambios previstos en el uso y los costos con los cambios previstos en la tecnología, o cualquier programa que esté ejecutando, y establecer objetivos para la carga de trabajo. Los objetivos deben centrarse en el uso y los costos, además de en la relación entre ambos. Los objetivos deben ser sencillos y de alto nivel. Además, deben ayudar a otras personas a saber lo que espera la empresa en cuanto a los resultados (por ejemplo, asegurarse de que los recursos sin usar estén por debajo de un determinado nivel de costo). No tiene que definir objetivos para cada tipo de recurso no utilizado ni

definir costos que provoquen pérdidas en los objetivos y las metas. Verifique que haya programas organizativos (por ejemplo, desarrollo de capacidades a través de cursos de formación) si se prevén cambios en los costos sin cambios en el uso.

- Definición de objetivos: debe especificar un objetivo cuantificable para cada de uno de los objetivos definidos. Si el objetivo es incrementar la eficiencia de la carga de trabajo, la meta debería cuantificar la mejora (normalmente en resultados empresariales por cada dólar gastado) y cuándo debe producirse. Por ejemplo, podría fijar un objetivo para minimizar el desperdicio debido a un exceso de aprovisionamiento. Con este objetivo, su meta puede ser que el desperdicio debido a un exceso de aprovisionamiento de computación en el primer nivel de las cargas de trabajo de producción no supere el 10 % del costo de computación del nivel. Además, una segunda meta podría ser que el desperdicio debido a un exceso de aprovisionamiento de computación en el segundo nivel de las cargas de trabajo de producción no supere el 5 % del costo de computación del nivel.

Recursos

Documentos relacionados:

- Políticas administradas de AWS para funciones de trabajo
- Estrategia de facturación de varias cuentas de AWS
- Control access to Regiones de AWS using IAM policies
- S.M.A.R.T. Goals
- How to track your cost optimization KPIs with the CID KPI Dashboard

Videos relacionados:

- Well-Architected Labs: Goals and Targets (Level 100)

Ejemplos relacionados:

- What is a unit metric?
- Selecting a unit metric to support your business
- Unit metrics in practice – lessons learned
- How unit metrics help create alignment between business functions

COST02-BP03 Implementación de una estructura de cuentas

Implemente una estructura de cuentas adaptada a su organización. Esto le ayudará a asignar y administrar los costos en toda la organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

AWS Organizations le permite crear varias Cuentas de AWS que pueden ayudarle a controlar de forma centralizada su entorno a medida que escala sus cargas de trabajo en AWS. Puede modelar su jerarquía organizativa si agrupa las Cuentas de AWS en una estructura de unidades organizativas (OU) y crea varias Cuentas de AWS en cada OU. Para crear una estructura de cuentas, primero debe decidir cuál de sus Cuentas de AWS será la de administración. Después de eso, puede crear nuevas Cuentas de AWS o seleccionar cuentas existentes como cuentas de miembro en función de la estructura de cuentas que haya diseñado según las prácticas recomendadas de cuentas de administración y las prácticas recomendadas de cuentas de miembro.

Se aconseja tener siempre al menos una cuenta de administración con una cuenta de miembro vinculada, sin importar el tamaño de la organización o su uso. Los recursos de las cargas de trabajo deberían estar solo en las cuentas de miembro y no se debería crear ningún recurso en la cuenta de administración. En cuanto a la pregunta sobre la cantidad de Cuentas de AWS que se debe tener, no existe una sola respuesta correcta para todas las situaciones. Primero debe evaluar sus modelos operativos y de costos, tanto actuales como futuros, para asegurarse de que la estructura de sus Cuentas de AWS refleje los objetivos de su organización. Algunas empresas crean varias Cuentas de AWS por motivos empresariales, por ejemplo:

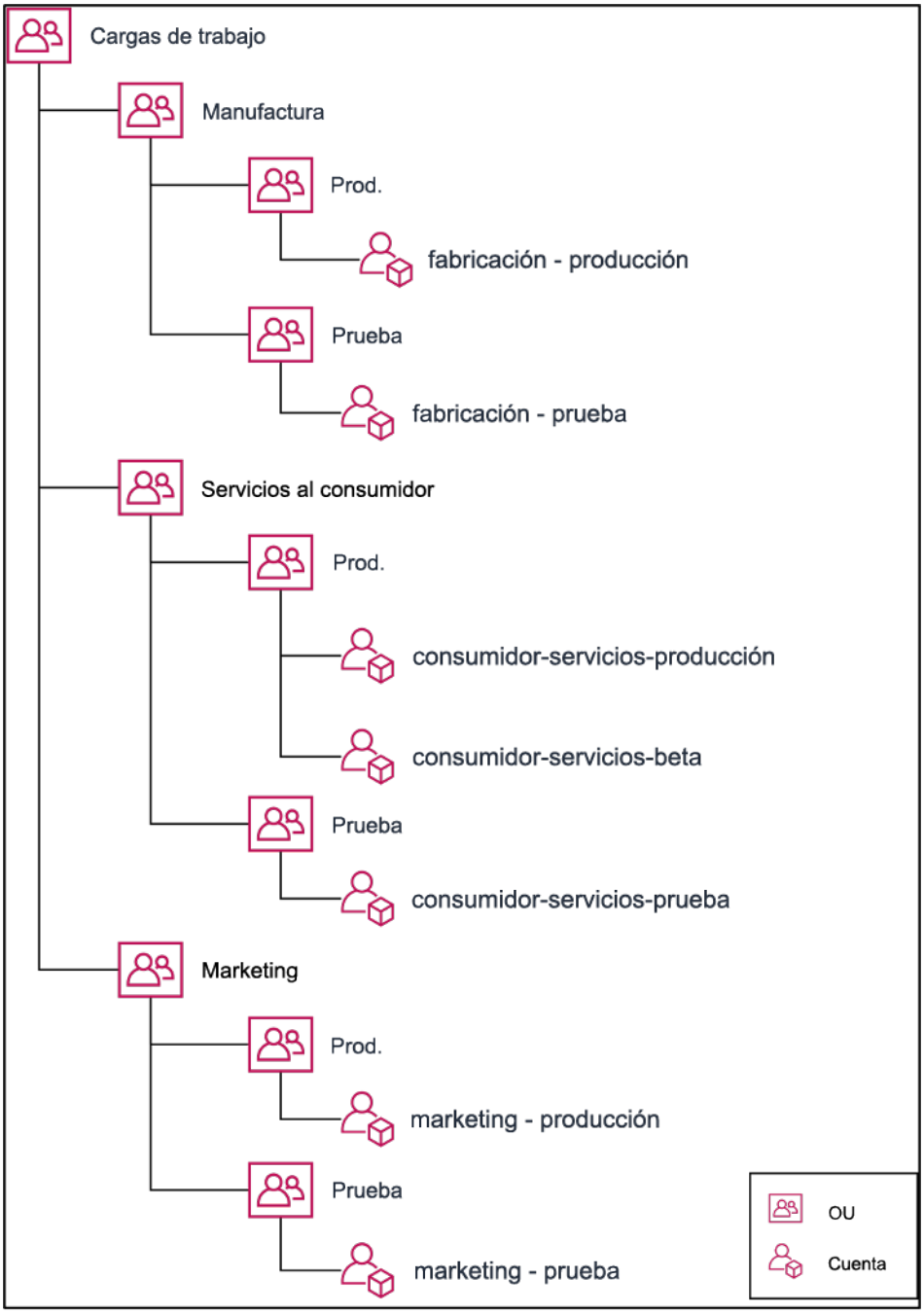
- Se requiere aislamiento administrativo o fiscal y de facturación entre unidades organizativas, centros de costos o cargas de trabajo específicas.
- Los límites de servicio de AWS están configurados para ser específicos para cargas de trabajo concretas.
- Existe un requisito de aislamiento y separación entre cargas de trabajo y recursos.

En AWS Organizations, la facturación unificada crea la estructura entre una o más cuentas de miembro y la cuenta de administración. Las cuentas de miembro le permiten aislar y distinguir los costos y el uso por grupos. Una práctica común es tener cuentas de miembro independientes para cada unidad organizativa (como finanzas, marketing y ventas), para cada ciclo de vida del entorno

(como desarrollo, prueba y producción) o para cada carga de trabajo (carga de trabajo a, b y c) y luego agregar dichas cuentas vinculadas mediante la facturación unificada.

La facturación unificada le permite unificar el pago de varias Cuentas de AWS de miembro en una sola cuenta de administración y proporcionar a la vez visibilidad de la actividad de cada cuenta vinculada. A medida que se agregan costos y uso a la cuenta de administración, puede maximizar los descuentos de volumen de servicio y el uso de los descuentos por compromiso (Savings Plans e instancias reservadas) para obtener los mayores descuentos.

En el siguiente diagrama se muestra cómo puede utilizar AWS Organizations con unidades organizativas (OU) para agrupar varias cuentas y colocar múltiples Cuentas de AWS en cada OU. Se recomienda utilizar OU para diversos casos de uso y cargas de trabajo, lo que proporciona patrones para organizar las cuentas.




Ejemplo de agrupación de varias Cuentas de AWS en unidades organizativas.

AWS Control Tower puede configurar rápidamente varias cuentas de AWS, lo que garantiza que la gobernanza esté alineada con los requisitos de la organización.

Pasos para la implementación

- Definición de los requisitos de separación: los requisitos de separación son una combinación de múltiples factores, como la seguridad, la fiabilidad y los modelos financieros. Ocúpese de cada

factor por orden y especifique si la carga de trabajo o el entorno de la carga de trabajo deberían separarse de otras cargas de trabajo. La seguridad promueve la adhesión a los requisitos de acceso y datos. La fiabilidad administra los límites de tal forma que los entornos y las cargas de trabajo no afecten a los demás. Consulte periódicamente los pilares de seguridad y fiabilidad del Marco de Well-Architected y siga las prácticas recomendadas. Los componentes financieros crean una separación financiera estricta (centro de costo diferente, propietarios de la carga de trabajo y responsabilidad). Algunos ejemplos comunes de separación son la ejecución de cargas de trabajo de producción y prueba en cuentas separadas o el uso de una cuenta separada para que la factura y los datos de facturación se puedan proporcionar a las unidades de negocios o departamentos individuales de la organización o parte interesada propietaria de la cuenta.

- Definición de los requisitos de agrupación: los requisitos de agrupación no anulan los requisitos de separación, pero se utilizan para ayudar en la administración. Agrupe entornos o cargas de trabajo similares que no requieran separación. Un ejemplo es agrupar múltiples entornos de prueba o desarrollo de una o varias cargas de trabajo.
- Definición de la estructura de cuentas: use estas separaciones y agrupaciones, especifique una cuenta para cada grupo y asegúrese de que se cumplan los requisitos de separación. Estas cuentas son sus cuentas de miembro o vinculadas. Al agrupar estas cuentas de miembro en una única cuenta de administración o de pagador, combina el uso, lo que le permite disfrutar de descuentos de mayor volumen en todas las cuentas y le proporciona una sola factura para todas las cuentas. Es posible separar los datos de facturación y proporcionar a cada cuenta de miembro una vista individual de sus datos de facturación. Si una cuenta de miembro no debe tener los datos de facturación o de uso visibles para las demás cuentas, o si se requiere una factura distinta de AWS, defina múltiples cuentas de administración o de pagador. En este caso, cada cuenta de miembro tiene su propia cuenta de administración o de pagador. Los recursos deberían colocarse siempre en las cuentas de miembro o vinculadas. Las cuentas de administración o de pagador solo deben usarse para tareas de administración.

Recursos

Documentos relacionados:

- Uso de etiquetas de asignación de costos
- Políticas administradas de AWS para funciones de trabajo
- Estrategia de facturación de varias cuentas de AWS
- Control access to Regiones de AWS using IAM policies
- AWS Control Tower

- AWS Organizations
- Prácticas recomendadas de cuentas de administración y cuentas de miembro
- Organización de su entorno de AWS con varias cuentas
- Turning on shared reserved instances and Savings Plans discounts
- Consolidated billing
- Consolidated billing

Ejemplos relacionados:

- Splitting the CUR and Sharing Access

Videos relacionados:

- Introducing AWS Organizations
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations

Ejemplos relacionados:

- Defining an AWS Multi-Account Strategy for telecommunications companies
- Best Practices for Optimizing Cuentas de AWS
- Best Practices for Organizational Units with AWS Organizations

COST02-BP04 Implementación de grupos y roles

Implemente grupos y roles que se ajusten a sus políticas y controle quién puede crear, modificar o retirar instancias y recursos en cada grupo. Por ejemplo, implemente grupos de desarrollo, de pruebas y de producción. Esto se aplica a los servicios de AWS y a las soluciones de terceros.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Los roles y los grupos de usuarios son elementos fundamentales en el diseño y la implementación de sistemas seguros y eficaces. Los roles y los grupos ayudan a las organizaciones a equilibrar la necesidad de control con el requisito de flexibilidad y productividad, lo que facilita, en última instancia, los objetivos de la organización y las necesidades de los usuarios. Como se recomienda

en la sección de Administración de identidad y acceso del pilar de seguridad del Marco de AWS Well-Architected, necesita contar con una sólida gestión de identidades y permisos para proporcionar acceso a los recursos correctos a las personas adecuadas en las condiciones adecuadas. Los usuarios reciben solo el acceso necesario para completar sus tareas. Esto minimiza el riesgo asociado con el acceso no autorizado o el uso indebido.

Después de desarrollar las políticas, puede crear grupos lógicos y roles de usuario dentro de la organización. Esto le permite asignar permisos, controlar el uso y ayudar a implementar mecanismos de control de acceso sólidos, lo que evita el acceso no autorizado a la información confidencial. Empiece con grupos de personas generales. Esto suele corresponderse con unidades organizativas y roles de trabajos (por ejemplo, un administrador de sistemas del departamento de TI, responsable financiero o analistas empresariales). Los grupos permiten clasificar a las personas que llevan a cabo tareas similares y necesitan accesos similares. Los roles definen lo que debe hacer un grupo. Es más fácil administrar permisos de grupos y roles que permisos de usuarios individuales. Los roles y los grupos asignan permisos de manera uniforme y sistemática a todos los usuarios, lo que evita errores e incoherencias.

Cuando cambia el rol de un usuario, los administradores pueden ajustar el acceso por rol o grupo, en lugar de volver a configurar cuentas de usuarios individuales. Por ejemplo, un administrador de sistemas de TI requiere acceso para crear todos los recursos, pero un miembro del equipo de Análisis solo lo necesita para crear recursos de análisis.

Pasos para la implementación

- Implementación de grupos: use los grupos de usuarios definidos en sus políticas organizativas e implemente los grupos correspondientes, si es necesario. Para conocer las prácticas recomendadas sobre los usuarios, los grupos y la autenticación, consulte el pilar de seguridad del Marco de AWS Well-Architected.
- Implementación de roles y políticas: use las acciones definidas en sus políticas organizativas y cree los roles y las políticas de acceso necesarios. Para conocer las prácticas recomendadas sobre funciones y políticas, consulte el pilar de seguridad del Marco de AWS Well-Architected.

Recursos

Documentos relacionados:

- Políticas administradas de AWS para funciones de trabajo
- Estrategia de facturación de varias cuentas de AWS

- Pilar de seguridad: Marco de AWS Well-Architected
- AWS Identity and Access Management (IAM)
- Políticas de AWS Identity and Access Management

Videos relacionados:

- Why use Identity and Access Management

Ejemplos relacionados:

- Control access to Regiones de AWS using IAM policies
- Starting your Cloud Financial Management journey: Cloud cost operations

COST02-BP05 Implementación de controles de costos

Implemente controles basados en las políticas de la organización y en los grupos y los roles definidos. De este modo, se certifica que los costos solo se producen según los requisitos de la organización, como controlar el acceso a regiones o tipos de recursos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Un primer paso común en la implementación de los controles de costos es establecer notificaciones cuando se producen eventos de costos o de uso fuera de las políticas. Puede actuar con rapidez y verificar si es necesaria una acción correctiva, sin restringir ni afectar negativamente a las cargas de trabajo o la nueva actividad. Una vez que conozca los límites de la carga de trabajo y del entorno, podrá aplicar la gobernanza. AWS Budgets le permite configurar notificaciones y definir presupuestos mensuales para sus costos, uso y descuentos por compromiso de AWS (Savings Plans e instancias reservadas). Puede crear presupuestos en un nivel de costo agregado (por ejemplo, todos los costos) o en un nivel más detallado en el que incluya solo dimensiones específicas como, por ejemplo, cuentas vinculadas, servicios, etiquetas o zonas de disponibilidad.

Una vez que haya establecido sus límites presupuestarios con AWS Budgets, use AWS Cost Anomaly Detection para reducir los costos imprevistos. AWS Cost Anomaly Detection es un servicio de administración de costos que utiliza machine learning para supervisar de forma continua el costo y el uso y, así, detectar gastos inusuales. Le ayuda a identificar los gastos anómalos y las causas

que los originan para que pueda adoptar medidas rápidamente. En primer lugar, cree un monitor de costos en AWS Cost Anomaly Detection, y, a continuación, elija su preferencia de alerta mediante el establecimiento de un umbral en dólares (como una alerta sobre anomalías con un impacto superior a 1000 USD). Una vez que reciba las alertas, podrá analizar la causa raíz que provoca la anomalía y el impacto en los costos. También puede supervisar y hacer sus propios análisis de anomalías en AWS Cost Explorer.

Aplique las políticas de gobernanza en AWS mediante AWS Identity and Access Management y políticas de control de servicios (SCP) de AWS Organizations. IAM le permite administrar el acceso a los servicios y recursos de AWS de forma segura. Mediante IAM, puede controlar quién puede crear o administrar los recursos de AWS, el tipo de recursos que se pueden crear y dónde se pueden crear. Esto minimiza la posibilidad de que se creen recursos fuera de la política definida. Utilice los roles y grupos creados anteriormente y asigne políticas de IAM para aplicar el uso correcto. La SCP ofrece un control centralizado de los permisos máximos disponibles para todas las cuentas de su organización, lo que mantiene sus cuentas según las directrices de control de acceso. Las SCP están disponibles solo en una organización que tenga todas las características activadas. Puede configurar las SCP para denegar o permitir acciones en las cuentas de los miembros de forma predeterminada. Para obtener más información sobre la implementación de la administración del acceso, consulte el documento técnico Pilar de seguridad: Marco de Well-Architected.

También se puede implementar la gobernanza mediante la administración de cuotas de servicio de AWS. Si garantiza que las cuotas de servicio se configuran con los gastos generales mínimos y se mantienen correctamente, puede minimizar la creación de recursos que no necesite su organización. Para lograrlo, debe conocer la velocidad con la que pueden cambiar sus requisitos, comprender los proyectos en curso (tanto la creación como la retirada de recursos) y tener en cuenta la rapidez con la que se pueden implementar los cambios de cuota. Las cuotas de servicio se pueden utilizar para aumentar las cuotas cuando sea necesario.

Pasos para la implementación

- Implementación de notificaciones sobre los gastos: utilice las políticas definidas de su organización y cree AWS Budgets para recibir notificaciones cuando los gastos estén fuera de sus políticas. Configure varios presupuestos de costos, uno para cada cuenta, que le notifiquen el gasto global de la cuenta. Configure presupuestos de costos adicionales en cada cuenta para unidades más pequeñas en ella. Estas unidades varían en función de la estructura de la cuenta. Algunos ejemplos comunes son las Regiones de AWS, las cargas de trabajo (mediante etiquetas) o los servicios de AWS. Configure una lista de distribución de correo electrónico como destinatario de las notificaciones y no una cuenta de correo electrónico individual. Puede configurar un

presupuesto real en caso de que se supere una cantidad o utilizar un presupuesto previsto para notificar el uso previsto. También puede preconfigurar acciones presupuestarias de AWS que pueden aplicar políticas de IAM o SCP específicas, o detener las instancias de Amazon EC2 y Amazon RDS de destino. Las acciones presupuestarias se pueden ejecutar automáticamente o requerir la aprobación del flujo de trabajo.

- Implementación de notificaciones sobre los gastos anómalos: use AWS Cost Anomaly Detection para reducir los costos imprevistos de su organización y analizar la causa fundamental de los posibles gastos anómalos. Una vez que haya creado el monitor de costos para identificar los gastos inusuales con el detalle que especifique y haya configurado las notificaciones en AWS Cost Anomaly Detection, le enviará una alerta cuando se detecten gastos inusuales. Esto le permitirá analizar la causa raíz detrás de la anomalía y comprender el impacto en el costo. Utilice las categorías de costos de AWS durante la configuración de AWS Cost Anomaly Detection para identificar qué equipo de proyecto o de unidad de negocio puede analizar la causa raíz del costo inesperado y tomar las medidas necesarias a tiempo.
- Implementación de controles de uso: mediante las políticas definidas de su organización, implemente políticas y roles de IAM para especificar qué acciones pueden hacer los usuarios y cuáles no. En una política de AWS pueden incluirse múltiples políticas de la organización. De la misma manera en que ha definido las políticas, empiece de manera amplia y, a continuación, aplique controles más detallados en cada paso. Los límites de servicio son también un control eficaz del uso. Implemente los límites de servicio correctos en todas las cuentas.

Recursos

Documentos relacionados:

- Políticas administradas de AWS para funciones de trabajo
- Estrategia de facturación de varias cuentas de AWS
- Control access to Regiones de AWS using IAM policies
- AWS Budgets
- AWS Cost Anomaly Detection
- Controle los costos de AWS

Videos relacionados:

- How can I use AWS Budgets to track my spending and usage

Ejemplos relacionados:

- Ejemplos de políticas de administración de acceso de IAM
- Example service control policies
- AWS Acciones de Budgets
- Creación de una política de IAM para controlar el acceso a los recursos de Amazon EC2 mediante etiquetas
- Restricción del acceso de IAM Identity a recursos específicos de Amazon EC2
- Slack integrations for Cost Anomaly Detection using Amazon Q Developer in chat applications

COST02-BP06 Seguimiento del ciclo de vida de los proyectos

Controle, mida y audite el ciclo de vida de los proyectos, equipos y entornos para evitar el uso y el pago de recursos innecesarios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Al hacer un seguimiento eficaz del ciclo de vida del proyecto, las organizaciones pueden controlar mejor los costos al planificar, administrar y optimizar los recursos. La información que se obtiene con el seguimiento es muy valiosa para tomar decisiones fundamentadas que contribuyen a la rentabilidad y al éxito general del proyecto.

El seguimiento de todo el ciclo de vida de la carga de trabajo le ayuda a comprender cuándo dejan de ser necesarias las cargas de trabajo o sus componentes. Puede que parezca que se usan las cargas de trabajo y los componentes existentes, pero cuando AWS publica nuevos servicios o características, estos pueden retirarse o adoptarse. Compruebe las etapas anteriores de las cargas de trabajo. Cuando una carga de trabajo ya no está en producción, los entornos previos se pueden retirar o reducirse en gran medida hasta que se necesiten de nuevo.

Puede etiquetar los recursos con un marco de tiempo o un recordatorio para fijar el momento en que se revisó la carga de trabajo. Por ejemplo, si el entorno de desarrollo se revisó por última vez hace meses, podría ser un buen momento para revisarlo de nuevo para analizar si se pueden adoptar nuevos servicios o si el entorno está en uso. Puede agrupar y etiquetar sus aplicaciones con myApplications activado en AWS para gestionar y hacer un seguimiento de los metadatos, como la criticidad, el entorno, la última revisión y el centro de costos. Puede hacer un seguimiento del ciclo de

vida de su carga de trabajo y supervisar y administrar el costo, el estado, la postura de seguridad y el rendimiento de sus aplicaciones.

AWS proporciona varios servicios de administración y gobernanza que puede usar para hacer un seguimiento del ciclo de vida de la entidad. Puede usar AWS Config o AWS Systems Manager para proporcionar un inventario detallado de sus recursos de AWS y su configuración. Se recomienda efectuar una integración con sus sistemas de administración de proyectos o recursos existentes para hacer un seguimiento de los proyectos y productos activos en su organización. Mediante la combinación del sistema actual con el amplio conjunto de eventos y métricas que ofrece AWS, podrá crear una vista de eventos importantes del ciclo de vida y administrar de forma proactiva los recursos a fin de reducir costos innecesarios.

Al igual que en la administración del ciclo de vida de aplicaciones (ALM), el seguimiento del ciclo de vida de un proyecto debe implicar la colaboración de varios procesos, herramientas y equipos, como el diseño y el desarrollo, las pruebas, la producción, el soporte y la redundancia de la carga de trabajo.

Al supervisar cuidadosamente cada fase del ciclo de vida de un proyecto, las organizaciones obtienen información crucial y mejoran el control, lo que facilita la planificación, implementación y finalización exitosas del proyecto. En esta cuidadosa supervisión, se verifica que los proyectos no solo cumplan los estándares de calidad, sino que se entreguen a tiempo y dentro del presupuesto, lo que fomenta el ahorro de costos.

Para obtener más información sobre la implementación del seguimiento del ciclo de vida de las entidades, consulte el documento técnico del Pilar de excelencia operativa de AWS Well-Architected.

Pasos para la implementación

- Establecimiento del proceso de supervisión del ciclo de vida de los proyectos: el equipo del Centro de excelencia en la nube debe establecer el proceso de supervisión del ciclo de vida de los proyectos. Establezca un enfoque estructurado y sistemático para supervisar las cargas de trabajo a fin de mejorar el control, la visibilidad y el rendimiento de los proyectos. Haga que el proceso de supervisión sea transparente y colaborativo y esté centrado en la mejora continua para maximizar su eficacia y valor.
- Revisiones de la carga de trabajo: según lo definan las políticas de su organización, establezca un ritmo regular para auditar sus proyectos existentes y hacer revisiones de la carga de trabajo. El esfuerzo dedicado a la auditoría debería ser proporcional al riesgo, el valor o el costo aproximados de la organización. Las principales áreas que debería incluir en la auditoría son el riesgo de incidente o interrupción en la organización, el valor o la contribución a la organización (medida en

ingresos o reputación de la marca), el costo de la carga de trabajo (medido como costo total de los recursos y costos operativos) y el uso de la carga de trabajo (medido en número de resultados organizativos por unidad de tiempo). Si estas áreas cambian durante el ciclo de vida, se deberá ajustar la carga de trabajo, por ejemplo, mediante una retirada total o parcial.

Recursos

Documentos relacionados:

- Guidance for Tagging on AWS
- ¿Qué es la administración del ciclo de vida de las aplicaciones (ALM)?
- Políticas administradas de AWS para funciones de trabajo

Ejemplos relacionados:

- Control access to Regiones de AWS using IAM policies

Herramientas relacionadas

- AWS Config
- AWS Systems Manager
- AWS Budgets
- AWS Organizations
- AWS CloudFormation

## COST 3. ¿Cómo supervisa sus costos y su uso?

Establezca políticas y procedimientos para supervisar y asignar adecuadamente sus costos. Esto le permite medir y mejorar la rentabilidad de esta carga de trabajo.

Prácticas recomendadas

- COST03-BP01 Configuración de los orígenes de información detallados
- COST03-BP02 Incorporación de información de la organización a los costos y el uso
- COST03-BP03 Identificación de las categorías de atribución de costos
- COST03-BP04 Establecimiento de métricas de la organización

- COST03-BP05 Configuración de herramientas de administración de facturación y costos
- COST03-BP06 Asignación de costos según las métricas de la carga de trabajo

COST03-BP01 Configuración de los orígenes de información detallados

Configure herramientas de generación de informes y administración de costos para mejorar el análisis y la transparencia de los datos de costos y uso. Configure la carga de trabajo para crear entradas de registro que faciliten el seguimiento y la división de los costos y el uso.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

La información de facturación detallada, como la especificidad por horas en las herramientas de administración de costos, permite a las organizaciones hacer un seguimiento más detallado de su consumo y les ayuda a identificar algunos de los motivos del aumento de costos. Estos orígenes de datos proporcionan la visión más veraz del costo y uso en toda la organización.

Puede usar Exportaciones de datos de AWS para crear exportaciones del AWS Cost and Usage Report (CUR) 2.0. Esta es la nueva forma recomendada de recibir los datos detallados de costo y uso de AWS. Proporciona especificidad de uso diario o por horas, tarifas, costos y atributos de uso de todos los servicios de AWS de pago (la misma información que el CUR), junto con algunas mejoras. Todas las dimensiones posibles están en el CUR, como, por ejemplo, el etiquetado, la ubicación, los atributos de recursos y los ID de cuentas.

Hay tres tipos de exportación según el tipo de exportación que desee crear: una exportación de datos estándar, una exportación a un panel de costos y uso con la integración de Quick o una exportación de datos heredados.

- Exportación de datos estándar: exportación personalizada de una tabla que se suministra a Amazon S3 de forma periódica.
- Panel de costos y uso: exportación e integración a Quick para implementar un panel de costos y uso prediseñado.
- Exportación de datos heredados: exportación del AWS Cost and Usage Report (CUR) heredado.

Puede crear exportaciones de datos con las siguientes personalizaciones:

- Inclusión de los ID de recurso

- División de los datos de asignación de costos
- Especificidad por horas
- Control de versiones
- Tipo de compresión y formato de archivo

En el caso de cargas de trabajo que ejecuten contenedores en Amazon ECS o Amazon EKS, habilite los datos de asignación de costos divididos para asignar los costos de los contenedores a unidades de negocio y equipos individuales, en función del modo en que las cargas de trabajo de los contenedores consumen los recursos compartidos de computación y memoria. Los datos de asignación de costos divididos también ingresan datos de costos y uso de los nuevos recursos de contenedor en el AWS Cost and Usage Report. Los datos de asignación de costos divididos se calculan mediante el cálculo del costo de los servicios y tareas individuales de ECS que se ejecutan en el clúster.

Un panel de costos y uso exporta la tabla del panel de costos y uso a un bucket de S3 de forma periódica e implementa un panel de costos y uso prediseñado en Quick. Use esta opción si desea implementar rápidamente un panel de sus datos de costos y uso sin que se pueda personalizar.

Si lo desea, puede seguir exportando CUR en modo heredado, donde puede integrar otros servicios de procesamiento, como AWS Glue para preparar los datos para el análisis y ejecutar análisis de datos con Amazon Athena mediante SQL para consultar los datos.

Pasos para la implementación

- Creación de exportaciones de datos: cree exportaciones personalizadas con los datos que desee y controle el esquema de sus exportaciones. Cree exportaciones de datos de administración de facturación y costos mediante SQL básico y visualice dichos datos mediante la integración con Quick. También puede exportar los datos en modo estándar para analizarlos con otras herramientas de procesamiento, como, por ejemplo, Amazon Athena.
- Configuración del informe de costo y uso: configure al menos un informe de costo y uso con la consola de facturación. Configure un informe detallado por horas que incluya todos los identificadores y los ID de recurso. También puede crear otros informes con distintos niveles de especificidad para proporcionar información resumida de nivel superior.
- Configuración de la especificidad horaria en el Explorador de costos: para acceder a los datos de costo y uso con especificidad de los últimos 14 días, plantéese habilitar los datos por hora y recursos en la consola de facturación.

- Configuración de los registros de la aplicación: verifique que su aplicación registre cada resultado empresarial que ofrece para que se pueda hacer el seguimiento y la medición. Asegúrese de que la especificidad de estos datos sea, como mínimo, por horas, para que coincidan con los datos de costo y uso. Para obtener más información sobre el registro y la supervisión, consulte el pilar de excelencia operativa de Well-Architected.

Recursos

Documentos relacionados:

- Exportaciones de datos de AWS
- AWS Glue
- Quick
- Administración financiera en la nube con AWS
- Etiquetado de recursos de AWS
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

Ejemplos relacionados:

- AWS Account Setup
- Data Exports for AWS Billing and Cost Management
- AWS Cost Explorer Common Use Cases

COST03-BP02 Incorporación de información de la organización a los costos y el uso

Defina un esquema de etiquetado basado en su organización, los atributos de carga de trabajo y las categorías de asignación de costos para poder filtrar y buscar recursos o supervisar el costo y el uso en las herramientas de administración de costos. Implemente un etiquetado coherente en todos los recursos, siempre que sea posible, por finalidad, equipo, entorno u otros criterios pertinentes para su empresa.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Implemente el etiquetado en AWS para agregar información de la organización a sus recursos, que luego se agregará a su información de costo y uso. Una etiqueta es un par clave-valor: la clave está definida y debe ser única en toda la organización, mientras que el valor es único para un grupo de recursos. Un ejemplo de par clave-valor es una clave `Environment`, con un valor `Production`. Todos los recursos del entorno de producción tendrán este par clave-valor. El etiquetado le permite categorizar y controlar los costos con información de la organización relevante y útil. Puede aplicar etiquetas que representen categorías de la organización (como centros de costos, nombres de aplicaciones, proyectos o propietarios) e identificar cargas de trabajo y características de cargas de trabajo (por ejemplo, de prueba o producción) para clasificar sus costos y uso en toda la organización.

Cuando aplica etiquetas a sus recursos de AWS (como instancias Amazon Elastic Compute Cloud o buckets de Amazon Simple Storage Service) y las activa, AWS agrega esta información a los informes de uso y costos. Puede ejecutar informes y análisis en recursos con etiquetas o sin ellas para permitir un mayor cumplimiento de las políticas de administración de costos internos y garantizar una atribución precisa.

Crear e implementar un estándar de etiquetado de AWS en las cuentas de su organización le ayudará a administrar y controlar sus entornos de AWS de manera coherente y uniforme. Utilice las políticas de etiquetas en AWS Organizations para definir reglas sobre cómo usar las etiquetas en los recursos de AWS de sus cuentas en AWS Organizations. Las políticas de etiquetas le permiten adoptar un enfoque estandarizado para los recursos de etiquetado de AWS.

El Editor de etiquetas de AWS le permite agregar, eliminar y administrar etiquetas de varios recursos. Con el editor de etiquetas, busque los recursos que desee etiquetar y, a continuación, administre las etiquetas de los recursos que aparecen en los resultados de la búsqueda.

Categorías de costos de AWS le permite asignar un significado organizativo a sus costos, sin necesidad de etiquetar los recursos. Puede asignar la información de costos y uso a estructuras organizativas internas únicas. Debe definir reglas de categorías para asignar y categorizar los costos mediante dimensiones de facturación, como cuentas y etiquetas. Esto proporciona otro nivel de capacidad de administración, además del etiquetado. También puede asignar cuentas específicas y etiquetas a varios proyectos.

Pasos para la implementación

- Definición de un esquema de etiquetado: reúna a todas las partes interesadas de su empresa para definir un esquema. En general, son personas con cargos técnicos, financieros o administrativos.

Defina una lista de etiquetas que deben tener todos los recursos, así como una lista de las etiquetas que deberían tener los recursos. Compruebe que los nombres y los valores de las etiquetas sean coherentes en toda la organización.

- Etiquete recursos: se las categorías de atributos de costos definidas para colocar etiquetas en todos los recursos en las cargas de trabajo según las categorías. Use herramientas como la CLI, el Editor de etiquetas o AWS Systems Manager para incrementar la eficiencia.
- Implementación de Categorías de costos de AWS: puede crear Categorías de costos sin implementar el etiquetado. Las categorías de costos usan las dimensiones de costos y uso existentes. Cree reglas de categorías a partir de su esquema e impleméntelas en las categorías de costos.
- Etiquetado automático: para comprobar que mantiene altos niveles de etiquetado en todos los recursos, automatice el etiquetado para que los recursos reciban etiquetas automáticamente en cuanto se creen. Utilice servicios como AWS CloudFormation para verificar que los recursos estén etiquetados al crearlos. También puede crear una solución personalizada para etiquetar automáticamente con funciones de Lambda o usar un microservicio personalizado que escanee la carga de trabajo periódicamente y elimine cualquier recurso que no tenga etiqueta, lo que es ideal para los entornos de prueba y de desarrollo.
- Supervisión e informes mediante el etiquetado: para garantizar que mantiene altos niveles de etiquetado en toda la organización, reporte y supervise las etiquetas de sus cargas de trabajo. Puede usar el AWS Cost Explorer para ver el costo de los recursos etiquetados o sin etiquetar, o bien usar servicios como el Editor de etiquetas. Revise periódicamente el número de recursos no etiquetados y agregue etiquetas hasta alcanzar el nivel de etiquetado que desee.

Recursos

Documentos relacionados:

- Tagging Best Practices
- AWS CloudFormation Resource Tag
- AWS Cost Categories
- Etiquetado de recursos de AWS
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

Videos relacionados:

- How can I tag my AWS resources to divide up my bill by cost center or project
- Tagging AWS Resources

COST03-BP03 Identificación de las categorías de atribución de costos

Identifique las categorías de la organización como las unidades empresariales, los departamentos o los proyectos que podrían utilizarse para asignar los costos dentro de su organización a las entidades consumidoras internas. Utilice esas categorías para imponer la responsabilidad del gasto, concienciar sobre los costos y fomentar comportamientos de consumo eficaces.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El proceso de categorización de los costos es crucial en la elaboración de presupuestos, la contabilidad, los informes financieros, la toma de decisiones, las evaluaciones comparativas y la administración de proyectos. Al clasificar y categorizar los gastos, los equipos pueden comprender mejor los tipos de costos en los que incurrirán durante su traspaso a la nube, lo que les ayuda a tomar decisiones fundamentadas y a administrar los presupuestos de manera eficaz.

La responsabilidad de los gastos en la nube es un buen incentivo para conseguir una administración disciplinada de la demanda y los costos. Como resultado, las organizaciones que destinan la mayor parte de su gasto en la nube a unidades empresariales o equipos que consumen recursos ahorran mucho más en costos en la nube. Además, la asignación del gasto en la nube ayuda a las organizaciones a adoptar más prácticas recomendadas de gobernanza en la nube centralizada.

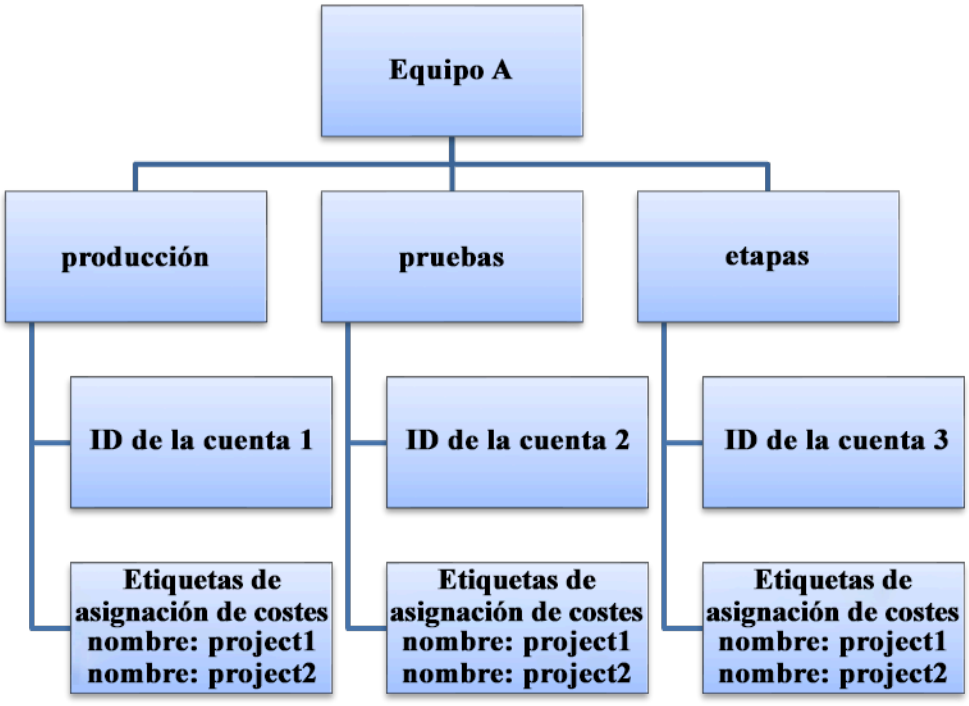
Colabore con el equipo financiero y otras partes interesadas pertinentes para comprender los requisitos sobre cómo deben asignarse los costos en la organización durante las llamadas de cadencia periódicas. Los costos de las cargas de trabajo deben asignarse a todo el ciclo de vida, como las fases de desarrollo, pruebas, producción y retirada. Debe saber qué costos de la organización proceden de la formación, el desarrollo del personal y la creación de ideas. Puede ser útil para asignar correctamente las cuentas que se usan para los presupuestos de formación y desarrollo, en lugar de presupuestos genéricos de costos de TI.

Tras definir las categorías de atribución de costos con las partes interesadas de la organización, utilice Categorías de costos de AWS para agrupar la información de costos y uso en categorías significativas en Nube de AWS, como el costo de un proyecto específico o Cuentas de AWS

de departamentos o unidades de negocio. Puede crear categorías personalizadas y asignar su información de costos y uso a estas categorías en función de las reglas que defina mediante varias dimensiones, tales como: cuenta, etiqueta, servicio o tipo de cargo. Tras configurar las categorías de costos, puede ver la información de costos y uso por estas categorías, lo que permite a la organización tomar mejores decisiones estratégicas y de compra. Estas categorías también se pueden ver en el AWS Cost Explorer, AWS Budgets y el AWS Cost and Usage Report.

Por ejemplo, cree categorías de costos para sus unidades empresariales (equipo de DevOps) y, en cada categoría, cree varias reglas (para cada subcategoría) con múltiples dimensiones (Cuentas de AWS, etiquetas de asignación de costos, servicios o tipo de cargo) basadas en las agrupaciones definidas. Con Cost Categories, puede organizar los costos usando un motor basado en reglas. Las reglas que configura organizan los costos en categorías. En estas reglas, puede llevar a cabo el filtrado mediante varias dimensiones para cada categoría, como Cuentas de AWS, servicios de AWS o tipos de cargos específicos. Puede utilizar estas categorías en los múltiples productos de la consola de Administración de costos y Administración de facturación y costos de AWS. Esto incluye AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report y AWS Cost Anomaly Detection.

En el siguiente diagrama se muestra, a modo de ejemplo, cómo agrupar la información de costos y uso de la organización si tiene varios equipos (categoría de costos) con varios entornos (reglas) y cada entorno tiene varios recursos o activos (dimensiones).




Organigrama de costos y uso

También puede crear agrupaciones de costos mediante categorías de costos. Después de crear las categorías de costos (deje que transcurran hasta 24 horas desde la creación de una categoría de costos para que sus registros de uso se actualicen con valores), estas aparecen en AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report y AWS Cost Anomaly Detection. En AWS Cost Explorer y AWS Budgets, una categoría de costo aparece como una dimensión de facturación adicional. Puede utilizarla para aplicar filtros para el valor de la categoría de costos específico o para efectuar agrupaciones por categorías de costos.

Pasos para la implementación

- Definición de las categorías de la organización: reúnase con las partes interesadas internas y las unidades empresariales para definir las categorías que reflejen la estructura y los requisitos de su organización. Estas categorías deben reflejar directamente la estructura de las categorías financieras existentes, como unidad empresarial, presupuestaria, centro de costos o departamento. Consulte los resultados de la nube para su empresa, como la formación o la educación, pues también son categorías de la organización.
- Definición de las categorías funcionales: reúnase con las partes interesadas internas y las unidades empresariales para definir las categorías que reflejen las funciones de su empresa. Pueden ser los nombres de las aplicaciones o las cargas de trabajo y el tipo de entorno, como producción, pruebas o desarrollo.
- Definición de categorías de costos de AWS: cree categorías de costos para organizar la información de costos y uso mediante Categorías de costos de AWS y asigne el costo y el uso de AWS a categorías significativas. Se pueden asignar varias categorías a un recurso y un recurso puede estar en muchas categorías distintas, por lo que se recomienda definir tantas categorías como sea necesario para que pueda administrar sus costos en la estructura de categorías con Categorías de costos de AWS.

Recursos

Documentos relacionados:

- Etiquetado de recursos de AWS
- Uso de etiquetas de asignación de costos de
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

- AWS Cost Categories
- Administración de costos con AWS Cost Categories
- Creating cost categories
- Tagging cost categories
- Splitting charges within cost categories
- AWS Características de las categorías de costos de

Ejemplos relacionados:

- Organize your cost and usage data with AWS Cost Categories
- Administración de costos con AWS Cost Categories

COST03-BP04 Establecimiento de métricas de la organización

Establezca las métricas de la organización necesarias para esta carga de trabajo. Algunos ejemplos de métricas de cargas de trabajo son los informes de clientes producidos o las páginas web que se entregan a los clientes.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Comprenda cómo se mide el rendimiento de su carga de trabajo en relación con el éxito empresarial. Cada carga de trabajo suele tener un pequeño conjunto de resultados principales que indican el rendimiento. Si tiene una carga de trabajo compleja con muchos componentes, puede priorizar la lista o definir las métricas de cada componente y hacer un seguimiento de ellas. Colabore con sus equipos para entender qué métricas debe utilizar. Esta unidad se usará para comprender la eficiencia de la carga de trabajo o el costo de cada resultado empresarial.

Pasos para la implementación

- Definición de los resultados de las cargas de trabajo: reúnase con las partes interesadas de la empresa y defina los resultados de las cargas de trabajo. Son una medida principal del uso de los clientes y deben ser métricas empresariales y no técnicas. Debe haber un pequeño número de métricas generales (menos de cinco) por carga de trabajo. Si la carga de trabajo produce varios resultados para diferentes casos de uso, agrúpelos en una sola métrica.

- Definición de los resultados de los componentes de las cargas de trabajo: de manera opcional, si tiene una carga de trabajo grande y compleja, o puede dividir fácilmente su carga de trabajo en componentes (como microservicios) con entradas y salidas bien definidas, establezca métricas para cada componente. El esfuerzo debe reflejar el valor y el costo del componente. Empiece por los componentes más grandes y continúe con los más pequeños.

Recursos

Documentos relacionados:

- Etiquetado de recursos de AWS
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

COST03-BP05 Configuración de herramientas de administración de facturación y costos

Configure las herramientas de administración de costos que cumplan las políticas de su organización para administrar y optimizar el gasto en la nube. Esto incluye servicios, herramientas y recursos para organizar los datos de costos y uso y hacer un seguimiento de ellos, mejorar el control mediante una facturación consolidada y permisos de acceso, mejorar la planificación mediante presupuestos y previsiones, recibir notificaciones o alertas y reducir los costos con optimizaciones de recursos y precios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para establecer una responsabilidad sólida, analice primero la estrategia de cuentas como parte de la estrategia de asignación de costos. Si lo hace bien, es posible que no necesite nada más. En caso contrario, puede haber desconocimiento y problemas adicionales.

Para fomentar la responsabilidad del gasto en la nube, conceda a los usuarios acceso a herramientas que les proporcionen visibilidad de sus costos y uso. AWS recomienda configurar todas las cargas de trabajo y equipos para los siguientes fines:

- Organización: establezca su base de referencia de asignación de costos y gobernanza con su propia estrategia de etiquetado y taxonomía. Cree varias cuentas de AWS con herramientas

como AWS Control Tower o AWS Organization. Etiquete los recursos que admite AWS y clasifíquelos de una forma fácil de entender en función de la estructura de su organización (unidades empresariales, departamentos o proyectos). Etiquete los nombres de las cuentas para centros de costos específicos y asígnelos a Categorías de costos de AWS para agrupar las cuentas de las unidades empresariales con sus centros de costos, de modo que el propietario de la unidad empresarial pueda ver el consumo de varias cuentas en un solo lugar.

- Acceso: haga un seguimiento de la información de facturación de toda la organización en la facturación unificada. Verifique que las partes interesadas y los propietarios empresariales adecuados tengan acceso a ella.
- Control: cree mecanismos de gobernanza efectivos con las barreras de protección adecuadas para evitar escenarios inesperados cuando se utilicen las políticas de control de servicio (SCP), las políticas de etiquetas, las políticas de IAM y las alertas de presupuestos. Por ejemplo, puede permitir que los equipos creen recursos específicos solo en las regiones que prefieran mediante mecanismos de control eficaces e impedir la creación de recursos sin una etiqueta específica (como el centro de costos).
- Estado actual: configure un panel que muestre los niveles actuales de costo y uso. El panel debe estar disponible en un lugar muy visible del entorno de trabajo, de forma similar a un panel de operaciones. Puede exportar los datos y utilizar el Panel de costos y uso del Centro de optimización de costos de AWS o cualquier producto compatible para conseguir esta visibilidad. Es posible que tenga que crear diferentes paneles para diferentes perfiles. Por ejemplo, el panel de administrador podría ser diferente al panel de ingeniero.
- Notificaciones: envíe notificaciones cuando el costo o el uso superen los límites definidos y se produzcan anomalías con AWS Budgets o la Detección de anomalías en los costos de AWS.
- Informes: resuma toda la información sobre costos y uso. Aumente la concienciación y la responsabilidad de su gasto en la nube con datos de costos detallados y atribuibles. Cree informes que sean relevantes para el equipo que los consume y que contengan recomendaciones.
- Seguimiento: muestre el costo y uso actuales con respecto a los objetivos o las metas configurados.
- Análisis: permita a los miembros del equipo ejecutar análisis detallados y personalizados con diferentes filtros (recurso, cuenta, etiqueta, etc.) con distintos filtros (recurso, cuenta, etiqueta, etc.).
- Inspección: manténgase al día de las oportunidades de implementación de recursos y optimización de costos. Reciba notificaciones mediante Amazon CloudWatch, Amazon SNS o Amazon SES para las implementaciones de recursos en la organización. Revise las recomendaciones de optimización de costos con AWS Trusted Advisor o AWS Compute Optimizer.

- Informes de tendencias: muestre la variabilidad del costo y uso durante el periodo requerido con el nivel de detalle necesario.
- Previsiones: muestre los costos futuros estimados y calcule el uso de sus recursos y el gasto con paneles de previsión que puede crear manualmente.

Puede utilizar el Centro de optimización de costos de AWS para conocer las posibles oportunidades de ahorro de costos consolidadas desde una ubicación centralizada y crear exportaciones de datos para su integración con Amazon Athena. También puede usar el Centro de optimización de costos de AWS para implementar el Panel de costos y uso que utiliza Quick para ejecutar un análisis de costos interactivo y un intercambio seguro de información sobre los costos.

Si no tiene las habilidades o el ancho de banda esenciales en su organización, puede trabajar con AWS ProServ, AWS Managed Services (AMS) o socios de AWS. También puede usar herramientas de terceros, pero asegúrese de validar la propuesta de valor.

Pasos para la implementación

- Concesión del acceso a las herramientas por equipo: configure sus cuentas y cree grupos que tengan acceso a los informes de costos y uso necesarios para su consumo y utilice AWS Identity and Access Management para controlar el acceso a herramientas como el AWS Cost Explorer. Estos grupos deben incluir representantes de todos los equipos que posean o administren una aplicación. De este modo, se certifica que cada equipo tiene acceso a su información de costos y uso para hacer el seguimiento de su consumo.
- Organización de las etiquetas y categorías de costos: organice sus costos entre equipos, unidades empresariales, aplicaciones, entornos y proyectos. Use etiquetas de recursos para organizar los costos por etiquetas de asignación de costos. Cree categorías de costos en función de las dimensiones mediante etiquetas, cuentas, servicios, etc., para asignar los costos.
- Configuración de AWS Budgets: configure AWS Budgets en todas las cuentas para sus cargas de trabajo. Establezca presupuestos para el gasto general de la cuenta y presupuestos para las cargas de trabajo con etiquetas y categorías de costos. Configure las notificaciones en AWS Budgets para recibir alertas cuando supere los importes presupuestados o cuando los costos estimados superen sus presupuestos.
- Configuración de la Detección de anomalías en los costos de AWS: use la Detección de anomalías en los costos de AWS para sus cuentas, servicios básicos o categorías de costos que haya creado para supervisar el costo y el uso, y detectar gastos fuera de lo habitual. Puede recibir las alertas individualmente en informes agregados y en un correo electrónico o un tema de Amazon SNS

que le permita analizar y determinar la causa raíz de la anomalía, e identificar el factor que está provocando el aumento de los costos.

- Uso de herramientas de análisis de costos: configure AWS Cost Explorer para su carga de trabajo y cuentas para visualizar los datos de costos y hacer un análisis posterior. Cree un panel para la carga de trabajo que haga un seguimiento del gasto general, las métricas clave de uso de la carga de trabajo y la previsión de los costos futuros a partir de sus datos históricos de costos.
- Utilice herramientas de análisis para ahorrar costos: utilice Centro de optimización de costos de AWS para identificar las oportunidades de ahorro con recomendaciones personalizadas, como eliminar los recursos no utilizados, ajustar el tamaño, Savings Plans, hacer reservas y hacer recomendaciones para optimizar el cálculo.
- Configure herramientas avanzadas: si lo desea, puede crear imágenes para facilitar el análisis interactivo y el intercambio de información sobre los costos. Con la exportación de datos del Centro de optimización de costos de AWS, puede crear un Panel de costos y uso con tecnología de Quick para la organización que aporte detalles y grado de detalle adicional. También puede implementar funciones de análisis avanzadas mediante el uso de exportaciones de datos en Amazon Athena para consultas avanzadas y crear paneles en Quick. Trabaje con socios de AWS para adoptar soluciones de administración en la nube para la supervisión y la optimización consolidadas de las facturas en la nube.

Recursos

Documentos relacionados:

- What is Administración de facturación y costos de AWS and Cost Management?
- Establecer su entorno de acuerdo a las prácticas recomendadas de AWS
- Best Practices for Tagging AWS Resources
- Etiquetado de los recursos de AWS
- AWS Cost Categories
- Analyzing your costs with AWS Budgets
- Analyzing your costs with AWS Cost Explorer
- What is AWS Data Exports?

Videos relacionados:

- Deploying Cloud Intelligence Dashboards

- Get Alerts on any FinOps or Cost Optimization Metric or KPI

Ejemplos relacionados:

- Panel de costos y uso con tecnología de Quick
- AWS Cost and Usage Governance Workshop

COST03-BP06 Asignación de costos según las métricas de la carga de trabajo

Asigne los costos de la carga de trabajo según las métricas de uso o los resultados empresariales para medir la eficiencia de los costos. Implemente un proceso para analizar los datos de costos y uso con servicios de análisis que pueden proporcionar información y capacidad de recuperación.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

La optimización de costos significa aportar resultados empresariales al menor precio, lo que solo se puede conseguir asignando los costos de la carga de trabajo por métricas de carga de trabajo (medidas por eficiencia de la carga de trabajo). Supervise las métricas de la carga de trabajo definidas mediante archivos de registro u otro tipo de supervisión de la aplicación. Combine estos datos con los costos de la carga de trabajo, que pueden obtenerse al consultar los costos que tienen un valor de etiqueta o identificador de cuenta específicos. Lleve a cabo este análisis en el nivel de hora. Por lo general, su eficiencia cambia si tiene componentes de costos estáticos (por ejemplo, una base de datos backend que se ejecuta permanentemente) con un índice de solicitudes variable (por ejemplo, el uso alcanza su punto máximo entre las nueve de la mañana y las cinco de la tarde, pero hay pocas solicitudes por la noche). Comprender la relación entre los costos variables y fijos le ayuda a centrar sus actividades de optimización.

Crear métricas de carga de trabajo para recursos compartidos puede ser difícil en comparación con recursos como las aplicaciones en contenedores de Amazon Elastic Container Service (Amazon ECS) y Amazon API Gateway. Sin embargo, hay ciertas formas de clasificar el uso y hacer un seguimiento de los costos. Si necesita hacer un seguimiento de los recursos compartidos de Amazon ECS y AWS Batch, puede habilitar los datos de asignación de costos divididos en AWS Cost Explorer. Al dividir los datos de asignación de costos, puede comprender y optimizar el costo y el uso de sus aplicaciones en contenedores y volver a asignar los costos de las aplicaciones a entidades empresariales individuales en función de cómo se consumen los recursos compartidos de computación y memoria.

Pasos para la implementación

- Asignación de costos a métricas de cargas de trabajo: use las métricas definidas y las etiquetas configuradas, y cree una métrica que combine el resultado de la carga de trabajo y el costo de la carga de trabajo. Utilice servicios de análisis como Amazon Athena y Amazon Quick para crear un panel de eficiencia para la carga de trabajo general y cualquier componente.

Recursos

Documentos relacionados:

- Etiquetado de recursos de AWS
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

Ejemplos relacionados:

- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

## COST 4. ¿Cómo retira los recursos?

Implemente el control de cambios y la administración de recursos desde el inicio del proyecto hasta su finalización. Esto garantiza el cierre o la terminación de recursos no utilizados para reducir el desperdicio.

Prácticas recomendadas

- COST04-BP01 Seguimiento de los recursos a lo largo de su ciclo de vida
- COST04-BP02 Implementación de un proceso de retirada
- COST04-BP03 Retirada de recursos
- COST04-BP04 Retirada automática de los recursos
- COST04-BP05 Aplicación de políticas de retención de datos

COST04-BP01 Seguimiento de los recursos a lo largo de su ciclo de vida

Defina e implemente un método para hacer un seguimiento de los recursos y sus asociaciones con los sistemas a lo largo de su ciclo de vida. Puede usar etiquetas para identificar la carga de trabajo o la función del recurso.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Retire los recursos de la carga de trabajo que ya no necesite. Por ejemplo, después de hacer pruebas, los recursos empleados se pueden eliminar. El seguimiento de los recursos con etiquetas (y la ejecución de informes en dichas etiquetas) puede ayudarle a identificar los elementos que se deben eliminar, ya que no estarán en uso o caducará su licencia. Usar etiquetas es una forma efectiva de hacer un seguimiento de los recursos. Se puede etiquetar el recurso con su función o una fecha conocida en la que se puede retirar. Puede ejecutar informes en estas etiquetas. Los valores de ejemplo para el etiquetado de características son `feature-X testing` para identificar la finalidad del recurso en términos del ciclo de vida de la carga de trabajo. Otro ejemplo es usar `LifeSpan` o `TTL` para los recursos, como el nombre y el valor de la clave de la etiqueta que se va a eliminar, para definir el periodo o el tiempo específico de retirada.

Pasos para la implementación

- Implementación de un esquema de etiquetado: implemente un esquema de etiquetado que identifique la carga de trabajo a la que pertenece el recurso y verifique que todos los recursos de la carga de trabajo estén etiquetados en consecuencia. El etiquetado le ayuda a categorizar los recursos por finalidad, equipo, entorno u otros criterios pertinentes para su empresa. Para obtener más información sobre los casos de uso, las estrategias y las técnicas de etiquetado, consulte Prácticas recomendadas para el etiquetado de los recursos de AWS.
- Implementación de la supervisión del rendimiento o la producción de la carga de trabajo: implemente la supervisión o las alarmas del rendimiento de la carga de trabajo para que se inicien en las solicitudes de entrada o en las terminaciones de salida. Configúrela para que proporcione notificaciones cuando las solicitudes de carga de trabajo o los resultados lleguen a cero, lo que significa que ya no se usan los recursos de la carga de trabajo. Incorpore un factor de tiempo si la carga de trabajo baja a cero de forma periódica en condiciones normales. Para obtener más información sobre los recursos no utilizados o infrautilizados, consulte AWS Trusted Advisor Cost Optimization checks.
- Agrupación de recursos de AWS: cree grupos de recursos para sus recursos de AWS. Puede utilizar Grupos de recursos de AWS para organizar y administrar los recursos de AWS que se

encuentran en la misma Región de AWS. Puede agregar etiquetas a la mayoría de sus recursos como ayuda para identificarlos y clasificarlos en su organización. Utilice el editor de etiquetas para agregar etiquetas a los recursos compatibles de forma masiva. Considere la posibilidad de utilizar AWS Service Catalog para crear, administrar y distribuir carteras de productos aprobados para los usuarios finales y administrar el ciclo de vida del producto.

Recursos

Documentos relacionados:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS Trusted Advisor Cost Optimization Checks
- Etiquetado de recursos de AWS
- Publicar métricas personalizadas

Videos relacionados:

- How to optimize costs using AWS Trusted Advisor

Ejemplos relacionados:

- ¿Cómo organizo mis recursos de AWS?
- Optimización de costos con AWS Trusted Advisor

COST04-BP02 Implementación de un proceso de retirada

Implemente un proceso para identificar y retirar los recursos sin usar.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Implemente un proceso estandarizado en toda la organización para identificar y eliminar los recursos que no se utilizan. El proceso debería definir la frecuencia con que se hacen las búsquedas y los procesos para retirar el recurso para verificar que se cumplan todos los requisitos de la organización.

Pasos para la implementación

- Creación e implementación de un proceso de retirada: trabaje con los desarrolladores y propietarios de las cargas de trabajo para diseñar un proceso de retirada de la carga de trabajo y sus recursos. El proceso debería incluir un método para verificar si se usa la carga de trabajo y también si se usa cada recurso de la carga de trabajo. Detalle los pasos necesarios para retirar el recurso del servicio a la vez que garantiza el cumplimiento de cualquier requisito normativo. Se debe incluir cualquier recurso asociado, como licencias o almacenamiento asociado. Notifique a los propietarios de las cargas de trabajo que se ha iniciado el proceso de retirada.

  Siga estos pasos de retirada como guía sobre lo que se debe comprobar como parte del proceso:

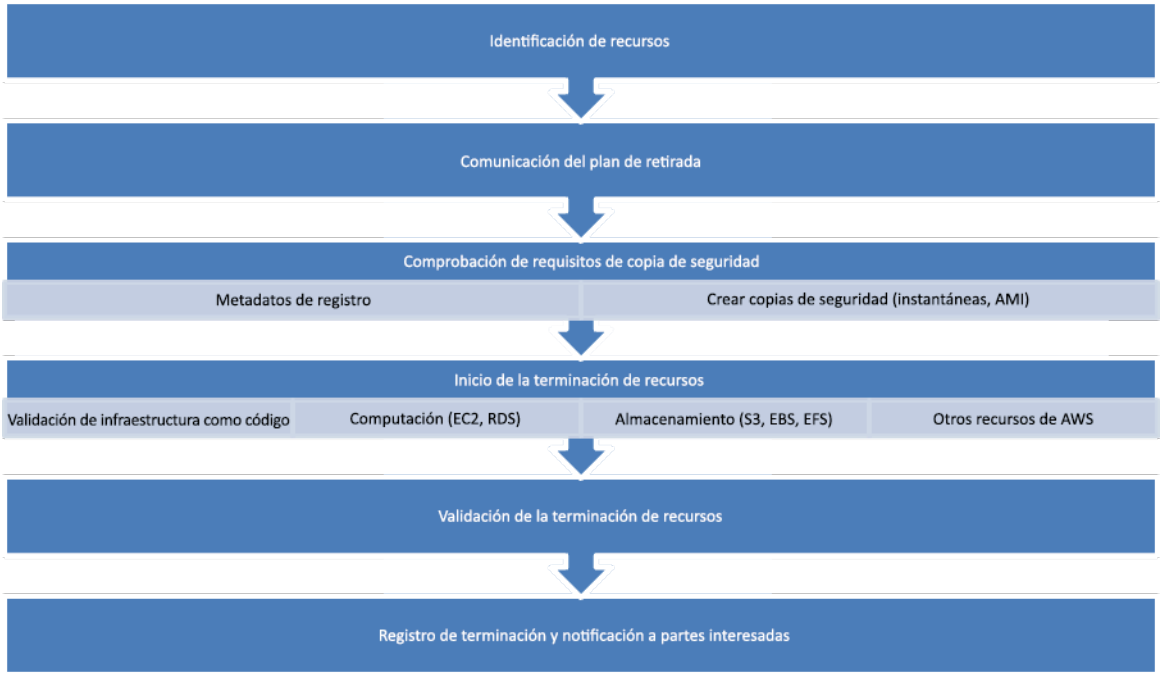
  - Identificación de los recursos que se van a retirar: identifique los recursos que son aptos para la retirada de su Nube de AWS. Registre toda la información necesaria y programe la retirada. En su cronología, asegúrese de tener en cuenta si surgen (y cuándo surgen) problemas inesperados durante el proceso.
  - Coordinación y comunicación: trabaje con los propietarios de la carga de trabajo para confirmar el recurso que se va a retirar
  - Registro de los metadatos y creación de copias de seguridad: registre los metadatos (como las IP públicas, la región, la AZ, la VPC, la subred y los grupos de seguridad) y cree copias de seguridad (como las instantáneas de Amazon Elastic Block Store o la captura de AMI, la exportación de claves y la exportación de certificados) si es necesario para los recursos del entorno de producción o si son recursos críticos.
  - Validación de la infraestructura como código: determine si los recursos se implementaron con CloudFormation, Terraform, Kit de desarrollo de la nube de AWS (AWS CDK) o cualquier otra herramienta de implementación de infraestructura como código para poder volver a implementarlos si es necesario.
  - Prevención del acceso: aplique controles restrictivos durante un periodo de tiempo para evitar el uso de los recursos y, al mismo tiempo, determinar si el recurso es necesario. Verifique que el entorno del recurso se pueda revertir a su estado original si es necesario.
  - Ejecución del proceso de desmantelamiento interno: siga las tareas administrativas y el proceso de retirada de su organización, como eliminar el recurso del dominio de la organización, eliminar el registro de DNS y eliminar el recurso de la herramienta de administración de la configuración, la herramienta de supervisión, la herramienta de automatización y las herramientas de seguridad.

  Si el recurso es una instancia de Amazon EC2, consulte la siguiente lista. Para obtener más información, consulte ¿Cómo puedo eliminar o terminar mis recursos de Amazon EC2?

- Detenga o termine todos sus equilibradores de carga e instancias de Amazon EC2. Las instancias de Amazon EC2 son visibles en la consola por poco tiempo después de su terminación. No se facturan las instancias que no se encuentran en estado de ejecución.
- Elimine la infraestructura de escalado automático.
- Libere todos los host dedicados.
- Elimine todos los volúmenes de Amazon EBS y las instantáneas de Amazon EBS.
- Libere todas las direcciones IP elásticas.
- Anule el registro de todas las imágenes de máquina de Amazon (AMI).
- Termine todos los entornos de AWS Elastic Beanstalk.

Si el recurso es un objeto en el almacenamiento de Amazon Glacier y si elimina un archivo antes de cumplir la duración de almacenamiento mínima, se le cobrará una tarifa prorrateada por eliminación anticipada. La duración de almacenamiento mínima de Amazon Glacier depende de la clase de almacenamiento utilizada. Para obtener un resumen de la duración mínima de almacenamiento para cada clase de almacenamiento, consulte Rendimiento en las clases de almacenamiento de Amazon S3. Para obtener información sobre cómo se calculan las tarifas por eliminación anticipada, consulte Precios de Amazon S3.

En el sencillo diagrama de flujo del proceso de retirada que figura a continuación se describen las etapas de retirada. Antes de retirar los recursos, verifique que la organización no use los que ha identificado para su retirada.




Flujo de retirada de recursos.

Recursos

Documentos relacionados:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

Videos relacionados:

- Delete CloudFormation stack but retain some resources
- Find out which user launched Amazon EC2 instance

Ejemplos relacionados:

- ¿Cómo puedo eliminar o terminar mis recursos de Amazon EC2?
- ¿Cómo puedo saber qué usuario ha iniciado una instancia de Amazon EC2 en mi cuenta?

### COST04-BP03 Retirada de recursos

Retire los recursos que algunos eventos inician, como las auditorías periódicas o los cambios en el uso. La retirada se suele llevar a cabo periódicamente y es manual o automática.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

#### Guía para la implementación

La frecuencia y el esfuerzo dedicados a buscar recursos que no se utilizan deberían reflejar el ahorro potencial, de manera que una cuenta con pocos costos debería analizarse con menos frecuencia que una cuenta con costos mayores. Las búsquedas y los eventos de retirada pueden iniciarse por cambios de estado de la carga de trabajo, como el fin de la vida útil de un producto o su reemplazo. También pueden iniciarse por eventos externos, como cambios en las condiciones de mercado o la finalización de un producto.

#### Pasos para la implementación

- Recursos de retirada: se trata de la fase de amortización de los recursos de AWS que ya no se necesitan o que tienen un contrato de licencia que está a punto de finalizar. Complete todas las comprobaciones finales hechas antes de pasar a la fase de eliminación y retirada de recursos para evitar interrupciones no deseadas, como la captura de instantáneas o la creación de copias de seguridad. Use el proceso de retirada para retirar los recursos identificados como no utilizados.

#### Recursos

Documentos relacionados:

- AWS Auto Scaling
- AWS Trusted Advisor

### COST04-BP04 Retirada automática de los recursos

Diseñe su carga de trabajo para que gestione de manera sencilla la finalización de recursos a medida que identifica y retira recursos que no son críticos, recursos innecesarios o recursos con poco uso.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Use la automatización para reducir o eliminar los costos asociados al proceso de retirada. El diseño de la carga de trabajo para que aplique procesos de retirada automáticos reducirá los costos generales de la carga de trabajo durante su vida. Puede utilizar Amazon EC2 Auto Scaling o Escalado automático de aplicaciones para llevar a cabo el proceso de retirada. También puede implementar código personalizado mediante la API o el SDK para retirar automáticamente los recursos de la carga de trabajo.

Las aplicaciones modernas se crean primero sin servidor, una estrategia que prioriza la adopción de servicios sin servidor. AWS desarrolló servicios sin servidor para los tres niveles de su pila: computación, integración y almacenes de datos. El uso de la arquitectura sin servidor le permitirá ahorrar costos durante periodos de poco tráfico, con escalado y desescalado verticales de forma automática.

Pasos para la implementación

- Implementación de Amazon EC2 Auto Scaling o Escalado automático de aplicaciones: para los recursos compatibles, configúrelos con Amazon EC2 Auto Scaling o Escalado automático de aplicaciones. Estos servicios pueden ayudarle a optimizar el uso y la rentabilidad a la hora de consumir servicios de AWS. Cuando baje la demanda, estos servicios eliminarán automáticamente cualquier exceso de capacidad de recursos para evitar un gasto excesivo.
- Configuración de CloudWatch para finalizar instancias: las instancias se pueden configurar para que finalicen mediante alarmas de CloudWatch. Use las métricas del proceso de retirada para implementar una alarma con una acción de Amazon Elastic Compute Cloud. Verifique la operación en un entorno que no sea de producción antes de la implementación.
- Implementación del código dentro de la carga de trabajo: puede usar el AWS SDK o la AWS CLI para retirar los recursos de la carga de trabajo. Implemente código en la aplicación que se integre con AWS y finalice o elimine recursos que ya no se usan.
- Uso de servicios sin servidor: priorice la creación de arquitecturas sin servidor y una arquitectura basada en eventos en AWS para crear y ejecutar sus aplicaciones. AWS ofrece varios servicios de tecnología sin servidor que, de forma inherente, optimizan automáticamente el uso de los recursos y automatizan la retirada (reducción y escalado horizontales). Con las aplicaciones sin servidor, el uso de los recursos se optimiza automáticamente y nunca pagará por un exceso de aprovisionamiento.

Recursos

Documentos relacionados:

- Amazon EC2 Auto Scaling
- Getting Started with Amazon EC2 Auto Scaling
- Aplicación de escalado automático
- AWS Trusted Advisor
- Sin servidor en AWS
- Creación de alarmas para parar, terminar, reiniciar o recuperar una instancia
- Agregación de acciones de terminación a las alarmas de Amazon CloudWatch

Ejemplos relacionados:

- Programación de la eliminación automática de las pilas de AWS CloudFormation

COST04-BP05 Aplicación de políticas de retención de datos

Defina políticas de retención de datos en los recursos admitidos para gestionar la eliminación de objetos según los requisitos de su organización. Identifique y elimine los recursos y objetos innecesarios o huérfanos que ya no sean necesarios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Utilice las políticas de retención de datos y de ciclo de vida para reducir los costos asociados al proceso de retirada y los costos de almacenamiento de los recursos identificados. Definir sus políticas de retención de datos y de ciclo de vida para llevar a cabo la migración y eliminación automatizadas de clases de almacenamiento reducirá los costos generales de almacenamiento durante su vida útil. Puede utilizar Amazon Data Lifecycle Manager para automatizar la creación y eliminación de instantáneas de Amazon Elastic Block Store e imágenes de máquina de Amazon (AMI) basadas en Amazon EBS, y utilizar Amazon S3 Intelligent-Tiering o una configuración del ciclo de vida de Amazon S3 para administrar el ciclo de vida de sus objetos de Amazon S3. También puede implementar código personalizado mediante la API o el SDK para crear políticas de ciclo de vida y reglas de políticas para que los objetos se eliminen automáticamente.

Pasos para la implementación

- Uso de Amazon Data Lifecycle Manager: utilice políticas de ciclo de vida en Amazon Data Lifecycle Manager para automatizar la eliminación de las instantáneas de Amazon EBS y las AMI basadas en Amazon EBS.
- Configuración del ciclo de vida en un bucket: utilice la configuración del ciclo de vida de Amazon S3 en un bucket para definir las acciones que Amazon S3 debe llevar a cabo durante el ciclo de vida de un objeto, así como la eliminación al final del ciclo de vida del objeto, en función de los requisitos de su empresa.

Recursos

Documentos relacionados:

- AWS Trusted Advisor
- Administrador de vida útil de datos de Amazon
- Configuración de un ciclo de vida en un bucket de Amazon S3

Videos relacionados:

- Automate Amazon EBS Snapshots with Amazon Data Lifecycle Manager
- ¿Cómo puedo vaciar un bucket de Amazon S3 mediante una regla de configuración del ciclo de vida?

Ejemplos relacionados:

- ¿Cómo puedo vaciar un bucket de Amazon S3 mediante una regla de configuración del ciclo de vida?

## Recursos rentables

Preguntas

- COST 5. ¿Cómo evalúa el costo cuando selecciona servicios?
- COST 6. ¿Cómo cumple los objetivos de costos cuando selecciona el tipo, el tamaño y el número de recursos?
- COST 7. ¿Cómo utiliza los modelos de fijación de precios para reducir los costos?
- COST 8. ¿Cómo planifica los gastos de transferencia de datos?

## COST 5. ¿Cómo evalúa el costo cuando selecciona servicios?

Amazon EC2, Amazon EBS y Amazon S3 son servicios de AWS básicos. Los servicios administrados, como Amazon RDS y Amazon DynamoDB, son servicios de AWS de nivel superior o de aplicación. Al seleccionar los bloques de creación y los servicios administrados apropiados, puede optimizar esta carga de trabajo para el costo. Por ejemplo, al usar servicios administrados, puede reducir o eliminar gran parte de sus gastos administrativos y operativos, lo que le permite trabajar en aplicaciones y actividades relacionadas con el negocio.

Prácticas recomendadas

- COST05-BP01 Identificación de los requisitos de la organización en relación con el costo
- COST05-BP02 Análisis de todos los componentes de la carga de trabajo
- COST05-BP03 Ejecución de un análisis exhaustivo de cada componente
- COST05-BP04 Selección de software con licencias rentables
- COST05-BP05 Selección de los componentes de la carga de trabajo para optimizar los costos de acuerdo con las prioridades de la organización
- COST05-BP06 Análisis de costos para diferentes usos a lo largo del tiempo

### COST05-BP01 Identificación de los requisitos de la organización en relación con el costo

Trabaje con los miembros del equipo para definir el equilibrio entre la optimización de costos y otros pilares, como el rendimiento y la fiabilidad, de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

En la mayoría de las organizaciones, el departamento de tecnología de la información (TI) está compuesto por varios equipos pequeños, cada uno con su propia agenda y área de enfoque, que refleja las especialidades y habilidades de los miembros de su equipo. Debe conocer los objetivos, prioridades y metas generales de su organización y cómo cada departamento o proyecto contribuye a estos objetivos. La clasificación de todos los recursos esenciales, incluidos el personal, el equipo, la tecnología, los materiales y los servicios externos, es crucial para lograr los objetivos de la organización y una planificación del presupuesto exhaustiva. La adopción de este enfoque sistemático para la identificación y comprensión de los costos es fundamental para establecer un plan de costos realista y sólido para la organización.

A la hora de seleccionar los servicios para su carga de trabajo, es fundamental que entienda las prioridades de su organización. Cree un equilibrio entre la optimización de costos y otros pilares del Marco de AWS Well-Architected, como el rendimiento y la fiabilidad. Este proceso debe llevarse a cabo de manera sistemática y regular para reflejar los cambios en los objetivos de la organización, las condiciones del mercado y la dinámica operativa. Una carga de trabajo totalmente optimizada en cuanto a costos es la solución que más se ajusta a los requisitos de su organización, no necesariamente la de menor costo. Reúnase con todos los equipos de su organización (por ejemplo, de productos, empresarial, técnico y financiero) para recopilar información. Evalúe el impacto de las compensaciones que se hacen entre intereses opuestos o enfoques alternativos para ayudar a tomar decisiones fundamentadas a la hora de determinar dónde centrar los esfuerzos o elegir una vía de acción.

Por ejemplo, comercializar más rápido las nuevas características puede primar sobre la optimización de los costos, o se podría elegir una base de datos relacional para los datos no relacionales para simplificar el esfuerzo de migración de un sistema en lugar de migrar a una base de datos optimizada para su tipo de datos y actualizar su aplicación.

Pasos para la implementación

- Identificación de los requerimientos para costos de su organización: reúnase con los miembros de los equipos de su organización, incluidos los de administración de productos, propietarios de aplicaciones, equipos de desarrollo y operativos, departamentos de administración y roles en finanzas. Priorice los pilares de Well-Architected para esta carga de trabajo y sus componentes. El resultado debería ser una lista ordenada de los pilares. También puede agregar una ponderación a cada pilar para indicar cuánto enfoque adicional tiene, o las similitudes de un enfoque entre dos pilares.
- Corrección de la deuda técnica y documentación: durante la revisión de la carga de trabajo, corrija la deuda técnica. Documente una tarea pendiente para revisitar la carga de trabajo en el futuro, con el objetivo de refactorizarla o rediseñarla para optimizarla aún más. Es esencial comunicar claramente a otras partes interesadas las compensaciones que se han hecho.

Recursos

Prácticas recomendadas relacionadas:

- REL11-BP07 Diseño de su producto para cumplir objetivos de disponibilidad y acuerdos de nivel de servicio (SLA) de tiempo de actividad
- OPS01-BP06 Evaluación de las compensaciones

Documentos relacionados:

- Calculadora del costo total de propiedad (TCO) de AWS
- Clases de almacenamiento de Amazon S3
- Productos de la nube

COST05-BP02 Análisis de todos los componentes de la carga de trabajo

Asegúrese de que se analice cada componente de la carga de trabajo, independientemente del tamaño o del costo actuales. El esfuerzo de revisión debería reflejar el beneficio potencial, como los costos actuales y previstos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Los componentes de la carga de trabajo, que están diseñados para ofrecer valor empresarial a la organización, pueden abarcar varios servicios. Para cada componente, se pueden elegir servicios específicos de Nube de AWS para abordar las necesidades empresariales. Esta selección podría estar influenciada por factores como la familiaridad con estos servicios o la experiencia previa con ellos.

Después de identificar los requisitos de su organización, tal como se menciona en COST05-BP01 Identificación de los requisitos de la organización en relación con el costo y haga un análisis exhaustivo de todos los componentes de su carga de trabajo. Analice cada componente teniendo en cuenta los costos y tamaños actuales y previstos. Compare el costo del análisis con cualquier posible ahorro en la carga de trabajo a lo largo de su ciclo de vida. El esfuerzo que se dedique a analizar todos los componentes de esta carga de trabajo debe compensar los posibles ahorros o mejoras que se tiene previsto conseguir con la optimización de ese componente específico. Por ejemplo, si el costo del recurso propuesto es de 10 USD al mes y, según las cargas previstas, no superaría los 15 USD al mes, dedicar un día de esfuerzo a reducir los costos un 50 % (5 USD al mes) no debería superar el beneficio potencial durante la vida del sistema. Utilice una estimación basada en datos más eficiente y rápida para conseguir el mejor resultado global para este componente.

Las cargas de trabajo pueden cambiar con el tiempo y el conjunto adecuado de servicios podría no ser óptimo si la arquitectura o el uso de la carga de trabajo cambia. En el análisis para seleccionar los servicios, se deben incluir estados de carga de trabajo actuales y futuros y niveles de uso. Implementar un servicio para un estado o uso de la carga de trabajo futura puede reducir los costos globales al reducir o eliminar el esfuerzo requerido para hacer cambios en el futuro. Por ejemplo,

es posible que sea adecuado utilizar EMR sin servidor en un principio. Sin embargo, a medida que aumenta el consumo de ese servicio, la transición a EMR en EC2 podría reducir los costos de ese componente de la carga de trabajo.

AWS Cost Explorer y el AWS Cost and Usage Report (CUR) pueden analizar el costo de una prueba de concepto (PoC) o un entorno en ejecución. También puede utilizar Calculadora de precios de AWS para estimar los costos de la carga de trabajo.

Escriba el flujo de trabajo que deben seguir los equipos técnicos para revisar sus cargas de trabajo. Procure que este flujo de trabajo sea sencillo, pero que abarque todos los pasos necesarios para asegurarse de que los equipos conozcan cada componente de la carga de trabajo y sus precios. Luego, su organización puede seguir y personalizar este flujo de trabajo en función de las necesidades específicas de cada equipo.

1. Enumeración de cada servicio que se use para la carga de trabajo: este es un buen punto de partida. Identifique todos los servicios que se están utilizando actualmente y el origen de los costos.
2. Comprensión de cómo funcionan los precios de esos servicios: comprenda el modelo de precios de cada servicio. Los distintos servicios de AWS tienen diferentes modelos de precios en función de factores como el volumen de uso, la transferencia de datos y los precios de características específicas.
3. Concentración en los servicios que conllevan costos de la carga de trabajo inesperados y que no se ajustan al uso y los resultados empresariales esperados: identifique los valores atípicos o los servicios en los que el costo no sea proporcional al valor o al uso mediante AWS Cost Explorer o AWS Cost and Usage Report. Es importante correlacionar los costos con los resultados empresariales para priorizar los esfuerzos de optimización.
4. AWS Cost Explorer, Registros de CloudWatch, VPC Flow Logs y Lente de almacenamiento de Amazon S3 para comprender la causa principal de esos altos costos: estas herramientas son fundamentales para diagnosticar los altos costos. Cada servicio ofrece una perspectiva diferente para ver y analizar el uso y los costos. Por ejemplo, el Explorador de costos ayuda a determinar las tendencias generales de los costos, Registros de CloudWatch proporciona información operativa, Registros de flujos de VPC muestran el tráfico IP y la Lente de almacenamiento de Amazon S3 es útil para analizar el almacenamiento.
5. Uso de AWS Budgets para establecer presupuestos para una determinada cantidad de servicios o cuentas: establecer presupuestos es una forma proactiva de administrar los costos. Utilice AWS Budgets para establecer umbrales presupuestarios personalizados y recibir alertas cuando los costos superen esos umbrales.

6. Configuración de las alarmas de Amazon CloudWatch para enviar alertas de facturación y uso: configure la supervisión y las alertas para las métricas de costo y uso. Las alarmas de CloudWatch pueden avisarle cuando se alcanzan ciertos umbrales, lo que mejora el tiempo de respuesta a la intervención.

Consiga una mejora y un ahorro financiero importantes a lo largo del tiempo mediante una revisión estratégica de todos los componentes de la carga de trabajo e independientemente de sus atributos actuales. El esfuerzo invertido en este proceso de revisión debe ser deliberado y deben estudiarse cuidadosamente las ventajas que podrían conseguirse.

Pasos para la implementación

- Enumeración de los componentes de la carga de trabajo: cree una lista de los componentes de la carga de trabajo. Utilice esta lista para comprobar que se hayan analizado todos los componentes. El esfuerzo que le dedique debería reflejar la importancia de la carga de trabajo, tal como definen las prioridades de la organización. Agrupar los recursos mejora la eficiencia funcional (por ejemplo, el almacenamiento de la base de datos de producción si hay varias bases de datos).
- Priorización de la lista de componentes: tome la lista de componentes y priorícela por orden de esfuerzo. En general, se ordena por el costo del componente, es decir, de más caro a menos caro, o por la importancia definida en las prioridades de la organización.
- Análisis: para cada componente de la lista, revise las opciones y los servicios disponibles y elija la opción que mejor se adapte a las prioridades de su organización.

Recursos

Documentos relacionados:

- Calculadora de precios de AWS
- AWS Cost Explorer
- Clases de almacenamiento de Amazon S3
- Productos en la Nube de AWS

Videos relacionados:

- AWS Cost Optimization Series: CloudWatch

COST05-BP03 Ejecución de un análisis exhaustivo de cada componente

Consulte el costo total que supone para la organización cada componente. Calcule el costo total de propiedad teniendo en cuenta el costo de las operaciones y la administración, sobre todo cuando utilice servicios administrados por el proveedor de servicios en la nube. El esfuerzo de revisión debe reflejar los posibles beneficios (por ejemplo, el tiempo empleado en analizar es proporcional al costo de los componentes).

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Considere el ahorro de tiempo que permitirá a su equipo centrarse en la retirada de la deuda técnica, la innovación, las características que agregan valor y la creación de lo que diferencia a la empresa. Por ejemplo, puede que necesite migrar mediante lift-and-shift (también conocido como “volver a alojar”) sus bases de datos de su entorno en las instalaciones a la nube lo más rápidamente posible y optimizarlas más tarde. Merece la pena explorar el ahorro que puede suponer el uso de servicios administrados en AWS que puedan eliminar o reducir los costos de las licencias. Los servicios administrados en AWS eliminan la carga operativa y administrativa del mantenimiento de un servicio, como la aplicación de parches o la actualización del sistema operativo, y le permiten centrarse en la innovación y la empresa.

Dado que los servicios administrados operan a la escala de la nube, pueden ofrecer un costo menor por transacción o servicio. Puede llevar a cabo optimizaciones potenciales para obtener alguna ventaja tangible, sin cambiar la arquitectura principal de la aplicación. Por ejemplo, es posible que desee reducir la cantidad de tiempo que dedica a administrar instancias de bases de datos mediante la migración a una plataforma de base de datos como servicio, como Amazon Relational Database Service (Amazon RDS), o mediante la migración de su aplicación a una plataforma totalmente administrada, como AWS Elastic Beanstalk.

Normalmente, los servicios administrados tienen atributos que puede configurar para garantizar una capacidad suficiente. Debe configurar y supervisar estos atributos para que su exceso de capacidad se mantenga al mínimo y el rendimiento se maximice. Puede modificar los atributos de AWS Managed Services mediante la Consola de administración de AWS o las API y los SDK de AWS para adaptar las necesidades de recursos a la demanda cambiante. Por ejemplo, puede aumentar o disminuir la cantidad de nodos en un clúster de Amazon EMR (o en un clúster de Amazon Redshift) para reducir o escalar horizontalmente.

También puede empaquetar varias instancias en un recurso de AWS para activar un uso de mayor densidad. Por ejemplo, puede aprovisionar varias bases de datos pequeñas en una sola instancia

de base de datos de Amazon Relational Database Service (Amazon RDS). A medida que aumenta el uso, puede migrar una de las bases de datos a una instancia de base de datos de Amazon RDS dedicada mediante un proceso de restauración y una instantánea.

Cuando aprovisione cargas de trabajo mediante servicios administrados, debe conocer los requisitos para ajustar la capacidad del servicio. Estos requisitos suelen ser tiempo, esfuerzo y cualquier impacto en el funcionamiento normal de la carga de trabajo. El recurso aprovisionado debe dejar tiempo para que se produzca cualquier cambio, por lo que debe aprovisionar la sobrecarga necesaria para permitirlo. El esfuerzo continuo requerido para modificar los servicios se puede reducir a prácticamente cero mediante el uso de las API y los SDK que se integran con el sistema y las herramientas de supervisión, como Amazon CloudWatch.

Amazon RDS, Amazon Redshift y Amazon ElastiCache proporcionan un servicio de bases de datos administrado. Amazon Athena, Amazon EMR y Amazon OpenSearch Service proporcionan un servicio de análisis administrado.

AMS es un servicio que utiliza la infraestructura de AWS en nombre de los socios y clientes de la empresa. Proporciona un entorno seguro y conforme a las normativas en el que puede implementar sus cargas de trabajo. AMS utiliza modelos operativos de nube empresarial con automatización para permitirle satisfacer los requisitos de su organización, trasladarse a la nube más rápidamente y reducir los costos de administración continua.

Pasos para la implementación

- Análisis exhaustivo: mediante la lista de componentes, examine cada uno de ellos de mayor a menor prioridad. En el caso de los componentes con mayor prioridad y más costosos, lleve a cabo un análisis adicional y evalúe todas las opciones disponibles y su impacto a largo plazo. En el caso de los componentes con menor prioridad, evalúe si los cambios en el uso modificarían la prioridad del componente y, a continuación, haga un análisis del esfuerzo adecuado.
- Comparación de los recursos administrados y no administrados: considere el costo operativo de los recursos que administra y compárelos con los recursos administrados de AWS. Por ejemplo, revise sus bases de datos que se ejecutan en instancias de Amazon EC2 y compárelas con las opciones de Amazon RDS (un servicio administrado de AWS) o compare Amazon EMR con la ejecución de Apache Spark en Amazon EC2. Cuando cambie de una carga de trabajo autoadministrada a una completamente administrada por AWS, investigue cuidadosamente sus opciones. Los tres factores más importantes que tener en cuenta son el tipo de servicio administrado que desea utilizar, el proceso que utilizará para migrar los datos y comprender el modelo de responsabilidad compartida de AWS.

Recursos

Documentos relacionados:

- AWS Calculadora del costo total de propiedad (TCO) de
- Clases de almacenamiento de Amazon S3
- Productos en la Nube de AWS
- Modelo de responsabilidad compartida de AWS

Videos relacionados:

- Why move to a managed database?
- What is Amazon EMR and how can I use it for processing data?

Ejemplos relacionados:

- Why to move to a managed database
- Consolidate data from identical SQL Server databases into a single Amazon RDS for SQL Server database using AWS DMS
- Deliver data at scale to Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Migrate an ASP.NET web application to AWS Elastic Beanstalk

COST05-BP04 Selección de software con licencias rentables

El software de código abierto elimina los costos de licencias de software, lo que puede repercutir enormemente en los costos de las cargas de trabajo. Si se requiere software con licencia, evite licencias vinculadas a atributos arbitrarios como las CPU y busque licencias vinculadas a los resultados. El costo de estas licencias está más vinculado al beneficio que aportan.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

El código abierto se originó en el contexto del desarrollo de software e indica que el software cumple ciertos criterios para su distribución gratuita. El software de código abierto consta de código fuente que cualquiera puede inspeccionar, modificar y mejorar. En función de los requisitos empresariales,

la habilidad de los ingenieros, el uso previsto u otras dependencias tecnológicas, las organizaciones pueden considerar la posibilidad de utilizar software de código abierto en AWS para minimizar los costos de sus licencias. En otras palabras, el costo de las licencias de software se puede reducir mediante el uso de software de código abierto. Esto puede repercutir de forma significativa en los costos de la carga de trabajo a medida que esta aumente.

Determine los beneficios del software con licencia teniendo en cuenta el costo total para optimizar su carga de trabajo. Haga simulaciones de los cambios en las licencias y estudie cómo afectaría a los costos de la carga de trabajo. Si un proveedor cambia el costo de la licencia de la base de datos, investigue cómo afecta eso a la eficiencia general de la carga de trabajo. Consulte el historial de anuncios de precios de sus proveedores para ver las tendencias en los cambios de las licencias en sus productos. Los costos de licencia también pueden variar sin tener en cuenta el rendimiento o el uso, como las licencias que varían según el hardware (licencias vinculadas a la CPU). Estas licencias deberían evitarse porque sus costos pueden incrementarse rápidamente sin que haya unos resultados correspondientes.

Por ejemplo, utilizar una instancia de Amazon EC2 en us-east-1 con un sistema operativo Linux le permite reducir los costos en aproximadamente un 45 %, en comparación con la ejecución de otra instancia de Amazon EC2 que se ejecute en Windows.

Calculadora de precios de AWS ofrece una forma integral de comparar los costos de varios recursos con diferentes opciones de licencia, como las instancias de Amazon RDS y los diferentes motores de bases de datos. Además, AWS Cost Explorer proporciona una perspectiva muy valiosa de los costos de las cargas de trabajo existentes, especialmente aquellas que vienen con diferentes licencias. Para la administración de licencias, AWS License Manager ofrece un método simplificado para supervisar y administrar las licencias de software. Los clientes pueden implementar y poner en funcionamiento su software de código abierto preferido en Nube de AWS.

Pasos para la implementación

- Análisis de las opciones de licencia: revise las condiciones de licencia del software disponible. Busque versiones de código abierto que dispongan de las funciones requeridas y si los beneficios del software con licencia superan su costo. Si las condiciones son favorables, el costo del software se compensa con el beneficio que aporta.
- Análisis del proveedor de software: revise el historial de cambios en los precios y las licencias del proveedor. Busque cambios que no se alineen con los resultados, tales como términos punitivos si se ejecuta hardware o se trabaja con plataformas de proveedores específicos. Además, fíjese en cómo hacen las auditorías y las sanciones que se podrían aplicar.

Recursos

Documentos relacionados:

- Open Source at AWS
- Calculadora del costo total de propiedad (TCO) de AWS
- Clases de almacenamiento de Amazon S
- Productos de la nube

Ejemplos relacionados:

- Open Source Blogs
- AWS Open Source Blogs
- Evaluación de optimización y licencias

COST05-BP05 Selección de los componentes de la carga de trabajo para optimizar los costos de acuerdo con las prioridades de la organización

Tenga en cuenta el costo al seleccionar los componentes de su carga de trabajo. Esto incluye el uso de servicios administrados y por aplicación o de una arquitectura sin servidor, de contenedores o basada en eventos para reducir el costo global. Minimice los costos de licencia con software de código abierto, software que no tenga costos de licencia o alternativas para reducir el costo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Tenga en cuenta el costo de los servicios y las opciones a la hora de seleccionar los componentes. Esto incluye el uso de servicios administrados y a nivel de aplicación, como Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Simple Notification Service (Amazon SNS) y Amazon Simple Email Service (Amazon SES) para reducir los costos organizativos generales.

Utilice contenedores y tecnología sin servidor para la computación, como AWS Lambda y Amazon Simple Storage Service (Amazon S3) para sitios web estáticos. Si es posible, coloque la aplicación en contenedores y utilice servicios de contenedores administrados de AWS, como Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

Minimice los costos de licencia con software de código abierto o software que no tenga costos de licencia (por ejemplo, Amazon Linux para cargas de trabajo de computación o migre bases de datos a Amazon Aurora).

Puede utilizar servicios sin servidor o por aplicación, como Lambda, Amazon Simple Queue Service (Amazon SQS), Amazon SNS y Amazon SES. Estos servicios eliminan la necesidad de administrar un recurso y proporcionan la función de ejecución de código, servicios de colas y entrega de mensajes. La otra ventaja es que reducen horizontalmente el rendimiento y el costo de acuerdo con el uso, por lo que permiten la asignación y atribución de costos de forma eficiente.

El uso de una arquitectura basada en eventos también es posible con servicios sin servidor. Las arquitecturas basadas en eventos se basan en la inserción, por lo que todo sucede bajo demanda a medida que el evento se presenta en el enrutador. De esta forma, no pagará por un sondeo continuo para comprobar si hay algún evento. Esto se traduce en un menor consumo de ancho de banda de la red, un menor uso de la CPU, una menor capacidad inactiva de la flota y menos establecimientos de protocolo de enlace SSL/TLS.

Para obtener más información sobre la tecnología sin servidor, consulte el documento técnico sobre lentes de Well-Architected Serverless Application.

Pasos para la implementación

- Seleccione cada servicio para optimizar el costo: se la lista de prioridades y el análisis para seleccionar la opción que se adapte mejor a las prioridades de la organización. En lugar de aumentar la capacidad para satisfacer la demanda, considere otras opciones que puedan ofrecerle un mejor rendimiento con un costo menor. Por ejemplo, si debe revisar el tráfico previsto para sus bases de datos en AWS, considere la posibilidad de aumentar el tamaño de la instancia o de utilizar servicios de Amazon ElastiCache (Redis o Memcached) a fin de proporcionar mecanismos de caché para sus bases de datos.
- Evaluación de la arquitectura basada en eventos: el uso de una arquitectura sin servidor también le permite crear una arquitectura basada en eventos para aplicaciones distribuidas basadas en microservicios, lo que le ayuda a crear soluciones escalables, resilientes, ágiles y rentables.

Recursos

Documentos relacionados:

- Calculadora del costo total de propiedad (TCO) de AWS
- AWS sin servidor

- ¿Qué es la arquitectura basada en eventos (EDA)?
- Clases de almacenamiento de Amazon S3
- Productos de la nube
- Amazon ElastiCache (Redis OSS)

Ejemplos relacionados:

- Getting started with event-driven architecture
- Arquitectura basada en eventos
- How Statsig runs 100x more cost-effectively using Amazon ElastiCache (Redis OSS)
- Prácticas recomendadas para trabajar con funciones de AWS Lambda

COST05-BP06 Análisis de costos para diferentes usos a lo largo del tiempo

Las cargas de trabajo pueden cambiar con el tiempo. Algunos servicios o características son más rentables en diferentes niveles de uso. Al analizar cada componente a lo largo del tiempo, así como el uso previsto, la carga de trabajo se mantiene rentable durante su vida útil.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

A medida que AWS lanza nuevos servicios y características, los servicios óptimos para su carga de trabajo también pueden cambiar. El esfuerzo necesario debe reflejar las ventajas potenciales. La frecuencia de revisión de la carga de trabajo depende de los requisitos de la organización. Si se trata de una carga de trabajo con un costo importante, implementar nuevos servicios antes maximizará el ahorro, por lo que llevar a cabo la revisión con mayor frecuencia puede ser de gran ayuda. Otro aspecto inicial que revisar es el cambio en los patrones de uso. Unos cambios significativos en el uso pueden indicar que unos servicios alternativos serían óptimos.

Si necesita trasladar datos a la Nube de AWS, puede seleccionar cualquier amplia variedad de servicios que ofrece AWS y herramientas de socios para ayudarle a migrar sus conjuntos de datos, ya sean archivos, bases de datos, imágenes de máquinas, volúmenes de bloques o, incluso, copias de seguridad en cinta. Por ejemplo, para trasladar una gran cantidad de datos con destino y origen en AWS o procesar datos en la periferia, puede utilizar uno de los dispositivos personalizados de AWS para trasladar de forma rentable petabytes de datos fuera de línea. Otro ejemplo: para tasas de

transferencia de datos más elevadas, un servicio de conexión directa puede resultar más barato que una VPN que proporcione la coherencia de conectividad necesaria para su empresa.

Revise su actividad de escalado basándose en el análisis de costos para diferentes usos a lo largo del tiempo. Analice el resultado para ver si la política de escalado puede ajustarse para agregar instancias con varios tipos de instancia y opciones de compra. Revise la configuración para ver si es posible reducir el mínimo para atender las solicitudes de los usuarios, pero con una flota de menor tamaño, y agregue más recursos para satisfacer la elevada demanda prevista.

Para efectuar un análisis de costos para diferentes usos a lo largo del tiempo, converse con las partes interesadas de su organización y utilice la característica de previsión de AWS Cost Explorer para predecir el impacto potencial de los cambios en el servicio. Supervise los desencadenadores del nivel de uso mediante AWS Budgets, alarmas de facturación de CloudWatch y AWS Cost Anomaly Detection para identificar y lanzar antes los servicios más rentables.

Pasos para la implementación

- Definición de los patrones de uso previstos: trabaje con su organización, como los propietarios de marketing y de productos, para documentar cuáles serán los patrones de uso esperados y previstos para la carga de trabajo. Hable con las partes interesadas de la empresa sobre el aumento de costo y uso, tanto históricos como previstos, y asegúrese de que el aumento se ajusta a los requisitos de la empresa. Identifique los días naturales, las semanas o los meses en los que espera que más usuarios utilicen sus recursos de AWS, lo que indica que debe aumentar la capacidad de los recursos existentes o adoptar servicios adicionales para reducir el costo y aumentar el rendimiento.
- Análisis de costos según el uso previsto: use los patrones de uso definidos para llevar a cabo un análisis en cada uno de estos puntos. El esfuerzo de análisis debería reflejar el resultado potencial. Por ejemplo, si el cambio de uso es grande, debería hacerse un análisis exhaustivo para verificar los costos y los cambios. En otras palabras, cuando el costo aumenta, el uso también debería aumentar para la empresa.

Recursos

Documentos relacionados:

- Calculadora del costo total de propiedad (TCO) de AWS
- Clases de almacenamiento de Amazon S3
- Productos de la nube

- Amazon EC2 Auto Scaling
- Migración de datos a la nube
- AWS Snow Family

Videos relacionados:

- AWS OpsHub for Snow Family

## COST 6. ¿Cómo cumple los objetivos de costos cuando selecciona el tipo, el tamaño y el número de recursos?

Compruebe que elija el tamaño y el número de recursos apropiados para la tarea en cuestión. Al seleccionar el tipo, el tamaño y el número más rentables, minimiza el desperdicio.

Prácticas recomendadas

- COST06-BP01 Modelado de costos
- COST06-BP02 Selección del tipo, tamaño y número de recursos en función de los datos
- COST06-BP03 Selección automática del tipo, tamaño y número de recursos en función de las métricas
- COST06-BP04 Consideración del uso de los recursos compartidos

### COST06-BP01 Modelado de costos

Identifique los requisitos de la organización (como las necesidades empresariales y los compromisos existentes) y lleve a cabo un modelado de costos (costos generales) de la carga de trabajo y de cada uno de sus componentes. Lleve a cabo actividades de referencia para la carga de trabajo bajo diferentes cargas previstas y compare los costos. El esfuerzo para llevar a cabo el modelado debería reflejar la ventaja potencial. Por ejemplo, el tiempo dedicado debe ser proporcional al costo del componente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Lleve a cabo la creación de modelos de costos para la carga de trabajo y cada uno de sus componentes para comprender el equilibrio entre los recursos. Busque el tamaño adecuado para

cada recurso de la carga de trabajo según un determinado nivel de rendimiento. Comprender las consideraciones de costos puede servir de base al caso empresarial de su organización y al proceso de toma de decisiones cuando se evalúen los resultados de obtención de valor para la implementación planificada de la carga de trabajo.

Lleve a cabo actividades de referencia para la carga de trabajo bajo diferentes cargas previstas y compare los costos. El esfuerzo de creación de modelos debe reflejar los posibles beneficios; por ejemplo, el tiempo dedicado es proporcional al costo de los componentes o al ahorro previsto. Para conocer las mejores prácticas, consulte la sección de revisión del pilar de eficiencia de rendimiento del Marco de AWS Well-Architected.

Por ejemplo, para crear modelos de costos para una carga de trabajo compuesta por recursos informáticos, AWS Compute Optimizer puede ayudar a modelar los costos de las cargas de trabajo en ejecución. Proporciona recomendaciones de tamaño ideal para los recursos de computación en función del historial de uso. Asegúrese de que se implementen agentes de CloudWatch en las instancias de Amazon EC2 para recopilar métricas de memoria que le ayuden con recomendaciones más precisas en AWS Compute Optimizer. Se trata del origen de datos ideal para los recursos de computación porque es un servicio gratuito que usa el machine learning para llevar a cabo numerosas recomendaciones en función de los niveles de riesgo.

Hay varios servicios que puede utilizar con registros personalizados como orígenes de datos para el dimensionamiento correcto de las operaciones de otros servicios y componentes de la carga de trabajo como AWS Trusted Advisor, Amazon CloudWatch y Registros de Amazon CloudWatch. AWS Trusted Advisor comprueba los recursos y marca los que se utilizan poco, lo que puede ayudarle a dimensionar correctamente sus recursos y a crear modelos de costos.

Estas son recomendaciones de datos y métricas de creación de modelos de costos:

- La supervisión debe reflejar fielmente la experiencia del usuario. Seleccione la granularidad correcta del periodo y elija cuidadosamente el percentil 99 o el percentil máximo en lugar del promedio.
- Seleccione el nivel de detalle correcto para el periodo de análisis necesario a fin de cubrir cualquier ciclo de carga de trabajo. Por ejemplo, si se lleva a cabo un análisis de dos semanas, es posible que esté pasando por alto un ciclo mensual de alto uso, lo que podría generar un aprovisionamiento insuficiente.
- Elija los servicios de AWS adecuados para la carga de trabajo prevista; para ello, tenga en cuenta sus compromisos existentes, los modelos de precios seleccionados para otras cargas de trabajo y la capacidad de innovar más rápidamente y centrarse en el valor empresarial principal.

Pasos para la implementación

- Modelado de costos: implemente la carga de trabajo o una prueba de concepto en una cuenta separada con los tipos y tamaños de recurso específicos de la prueba. Ejecute la carga de trabajo con los datos de la prueba y registre los resultados de la salida, así como los datos de costos del momento en que se ejecutó la prueba. Después, vuelva a implementar la carga de trabajo o cambie los tipos y tamaños de recurso y vuelva a ejecutar la prueba. Incluya las tarifas de licencia de cualquier producto que pueda utilizar con estos recursos y los costos estimados de las operaciones (mano de obra o ingenieros) para implementar y administrar estos recursos durante la creación del modelado de costos. Considere el modelado de costos para un periodo (por hora, por día, por mes, por año o por trienio).

Recursos

Documentos relacionados:

- AWS Auto Scaling
- Identificación de oportunidades para ajustar el tamaño
- Características de Amazon CloudWatch
- Cost Optimization: Amazon EC2 Right Sizing
- AWS Compute Optimizer
- Calculadora de precios de AWS

Ejemplos relacionados:

- Creación de modelos de costos basado en datos
- ¿Cómo calculo el costo de las configuraciones planificadas de recursos de AWS?
- Choose the right AWS tools

COST06-BP02 Selección del tipo, tamaño y número de recursos en función de los datos

Seleccione el tamaño o tipo de recurso en función de los datos sobre las características de la carga de trabajo y de los recursos. Por ejemplo, computación, memoria, rendimiento o uso intensivo de escritura. Para llevar a cabo esta selección, suele utilizarse una versión anterior (en las instalaciones) de la carga de trabajo, la documentación u otras fuentes de información sobre la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Amazon EC2 ofrece una amplia selección de tipos de instancias con diferentes niveles de capacidad de CPU, memoria, almacenamiento y red para adaptarse a diferentes casos de uso. Estos tipos de instancias ofrecen diferentes combinaciones de capacidades de CPU, memoria, almacenamiento y red, lo que le proporciona versatilidad a la hora de seleccionar la combinación de recursos adecuada para sus proyectos. Cada tipo de instancia se ofrece en varios tamaños, de modo que puede ajustar sus recursos en función de las demandas de su carga de trabajo. Para determinar qué tipo de instancia necesita, recopile datos sobre los requisitos del sistema de la aplicación o el software que tiene pensado ejecutar en su instancia. Estos datos deben incluir lo siguiente:

- Sistema operativo
- Número de núcleos de CPU
- Núcleos de GPU
- Cantidad de memoria del sistema (RAM)
- Tipo y espacio de almacenamiento
- Requisitos de ancho de banda de la red

Identifique el propósito de los requisitos de computación y qué instancia se necesita y, a continuación, examine las distintas familias de instancias de Amazon EC2. Amazon ofrece las siguientes familias de tipos de instancias:

- Uso general
- Computación optimizada
- Optimizada para memoria
- Optimización de almacenamiento
- Computación acelerada
- Optimizadas para HPC

Para obtener información más profunda sobre los propósitos y casos de uso específicos que puede cumplir una familia de instancias concreta de Amazon EC2, consulte Tipos de instancias de AWS.

La recopilación de requisitos del sistema es fundamental para seleccionar la familia de instancias y el tipo de instancia específicos que mejor se ajusten a sus necesidades. Los nombres de los tipos de

instancias están compuestos por el nombre de la familia y el tamaño de la instancia. Por ejemplo, la instancia t2.micro pertenece a la familia T2 y tiene el tamaño micro.

Seleccione el tamaño o el tipo de recurso en función de las características de la carga de trabajo y de los recursos (por ejemplo: computación, memoria, rendimiento o uso intensivo de escritura). Para llevar a cabo esta selección, suele utilizarse la creación de modelos de costos, una versión anterior de la carga de trabajo (por ejemplo, una versión en las instalaciones), mediante documentación u otras fuentes de información sobre la carga de trabajo (documentos técnicos o soluciones publicadas). El uso de calculadoras de precios o herramientas de administración de costos de AWS puede ayudar a tomar decisiones informadas sobre los tipos, tamaños y configuraciones de las instancias.

Pasos para la implementación

- Selección de los recursos en función de los datos: utilice los datos de modelado de costos para seleccionar el nivel de uso previsto de la carga de trabajo y elija el tipo y el tamaño de los recursos especificados. En función de los datos del modelado de costos, determine el número de CPU virtuales, la memoria total (GiB), el volumen del almacén de instancias local (GB), los volúmenes de Amazon EBS y el nivel de rendimiento de la red, teniendo en cuenta la velocidad de transferencia de datos necesaria para la instancia. Haga siempre selecciones basadas en análisis detallados y datos precisos para optimizar el rendimiento al tiempo que administra los costos de forma eficaz.

Recursos

Documentos relacionados:

- AWS Tipos de instancias
- AWS Auto Scaling
- Características de Amazon CloudWatch
- Optimización de costos: Ajuste correcto del tamaño de EC2

Videos relacionados:

- Selecting the right Amazon EC2 instance for your workloads
- Right size your service

Ejemplos relacionados:

- It just got easier to discover and compare Amazon EC2 instance types

COST06-BP03 Selección automática del tipo, tamaño y número de recursos en función de las métricas

Use métricas de la carga de trabajo actual para seleccionar el tamaño y tipo correctos para optimizar el costo. Aprovisione de forma adecuada el rendimiento, el tamaño y el almacenamiento para los servicios de computación, almacenamiento, datos y redes. Esto puede hacerse con un bucle de retroalimentación, como el escalado automático, o mediante un código personalizado en la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Cree un bucle de retroalimentación en la carga de trabajo que use métricas activas de la carga de trabajo en ejecución para hacer cambios en dicha carga de trabajo. Puede utilizar un servicio gestionado, por ejemplo AWS Auto Scaling, que configure para que lleve a cabo las operaciones de dimensionamiento adecuadas. AWS también proporciona API, SDK y funciones que permiten modificar los recursos con un esfuerzo mínimo. Puede programar una carga de trabajo para que detenga e inicie una instancia de Amazon EC2 a fin de poder hacer un cambio en el tamaño o el tipo de instancia. Esto permite obtener el tamaño adecuado y, además, permite eliminar casi todo el costo operativo necesario para hacer el cambio.

Algunos servicios de AWS incluyen una selección automática de tipos o tamaños, como Amazon Simple Storage Service Intelligent-Tiering. Amazon S3 Intelligent-Tiering mueve automáticamente los datos entre dos niveles de acceso (frecuente y poco frecuente) en función de sus patrones de uso.

Pasos para la implementación

- Configuración de las métricas de la carga de trabajo para aumentar la observabilidad: capture las métricas clave de la carga de trabajo. Estas métricas son indicativas de la experiencia del cliente, como el resultado de la carga de trabajo, y alinean las diferencias que hay entre los tipos y los tamaños de los recursos, como la CPU y el uso de memoria. En el caso del recurso de computación, analice los datos de rendimiento para determinar el tamaño adecuado de sus instancias de Amazon EC2. Identifique las instancias inactivas y las infrautilizadas. Las métricas clave que hay que tener en cuenta son el uso de la CPU y de la memoria (por ejemplo, el 40 % de

uso de la CPU el 90 % del tiempo, como se explica en Rightsizing with AWS Compute Optimizer and Memory Utilization Enabled). Identifique las instancias con un uso de CPU y de memoria máximos inferiores al 40 % durante un periodo de cuatro semanas. Estas son las instancias que hay que dimensionar correctamente para reducir costos. Para los recursos de almacenamiento como Amazon S3, puede utilizar la Lente de almacenamiento de Amazon S3, que le permite ver 28 métricas de varias categorías por bucket y 14 días de datos históricos en el panel de control de forma predeterminada. Puede filtrar su panel de Lente de almacenamiento de Amazon S3 por resumen y optimización de costos o eventos para analizar métricas específicas.

- Consulta de las recomendaciones de ajuste de tamaño: utilice las recomendaciones de ajuste de tamaño de AWS Compute Optimizer y la herramienta de ajuste de tamaño de Amazon EC2 de la consola de gestión de costos, o revise el tamaño correcto de sus recursos para hacer ajustes en su carga de trabajo de AWS Trusted Advisor. Es importante utilizar las herramientas adecuadas a la hora de dimensionar los distintos recursos y seguir las pautas de dimensionamiento correctas, ya sea una instancia de Amazon EC2, clases de almacenamiento de AWS o tipos de instancias de Amazon RDS. En el caso de los recursos de almacenamiento, puede utilizar la Lente de almacenamiento de Amazon S3, que le ofrece visibilidad sobre el uso del almacenamiento de objetos, las tendencias de actividad y le proporciona recomendaciones prácticas para optimizar los costos y aplicar las prácticas recomendadas de protección de datos. Con las recomendaciones contextuales que la Lente de almacenamiento de Amazon S3 obtiene del análisis de las métricas de su organización, puede tomar medidas inmediatas para optimizar su almacenamiento.
- Selección automática del tipo y el tamaño del recurso en función de las métricas: utilice las métricas de la carga de trabajo para seleccionar los recursos de la carga de trabajo de forma manual o automática. En el caso de los recursos de computación, configurar AWS Auto Scaling o implementar el código en su aplicación puede reducir el esfuerzo necesario si deben hacerse cambios frecuentes, y así podrá implementar cambios potenciales antes que con el proceso manual. Puede lanzar y escalar automáticamente una flota de instancias en diferido e instancias de spot en un solo grupo de Auto Scaling. Además de los descuentos relacionados con las instancias de spot, puede utilizar instancias reservadas o un Savings Plan para conseguir mejores precios de los habituales en las instancias en diferido. Todos estos factores combinados le ayudarán a optimizar el ahorro de costos de las instancias de Amazon EC2 y a determinar la escala y el rendimiento que desea para su aplicación. También puede utilizar una estrategia de selección de tipo de instancia basada en atributos (ABS) en los grupos de escalado automático (ASG), lo que le permite expresar sus requisitos de instancia como un conjunto de atributos, como, por ejemplo, vCPU, memoria y almacenamiento. Puede utilizar automáticamente los tipos de instancia de nueva generación cuando se lancen y acceder a una gama más amplia de capacidad con las instancias de spot de Amazon EC2. Flota de Amazon EC2 y Amazon EC2 Auto Scaling

seleccionan y lanzan instancias que se ajusten a los atributos especificados, por lo que no es necesario elegir manualmente los tipos de instancia. En cuanto a los recursos de almacenamiento, puede utilizar las características Amazon S3 Intelligent-Tiering y Amazon EFS Infrequent Access, que le permiten seleccionar automáticamente las clases de almacenamiento que ofrecen ahorros automáticos en los costos de almacenamiento cuando cambian los patrones de acceso a los datos, sin que ello afecte al rendimiento ni a la sobrecarga operativa.

Recursos

Documentos relacionados:

- AWS Auto Scaling
- Tamaño correcto de AWS
- AWS Compute Optimizer
- Características de Amazon CloudWatch
- Configuración inicial de CloudWatch
- Publicar métricas personalizadas de CloudWatch
- Getting Started with Amazon EC2 Auto Scaling
- Amazon S3 Storage Lens
- Amazon S3 Intelligent-Tiering
- Amazon EFS Infrequent Access
- Launch an Amazon EC2 Instance Using the SDK

Videos relacionados:

- Right Size Your Services

Ejemplos relacionados:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive scaling with Amazon EC2 Auto Scaling
- Optimice los costos y obtenga visibilidad sobre el uso gracias a la Lente de almacenamiento de Amazon S3

### COST06-BP04 Consideración del uso de los recursos compartidos

En el caso de los servicios ya implementados en la organización para varias unidades de negocio, plantéese la posibilidad de usar los recursos compartidos para aumentar su uso y reducir el costo total de propiedad (TCO). El uso de recursos compartidos puede ser una opción rentable para centralizar la administración y los costos mediante el uso de las soluciones existentes, el uso compartido de componentes o ambas opciones. Administre funciones comunes, como la supervisión, las copias de seguridad y la conectividad, ya sea dentro de los límites de una cuenta o en una cuenta dedicada. También puede reducir los costos mediante la implementación de la estandarización y la reducción de la duplicación y la complejidad.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

#### Guía para la implementación

Cuando varias cargas de trabajo provoquen la misma función, utilice las soluciones existentes y los componentes compartidos para mejorar la administración y optimizar los costos. Considere la posibilidad de utilizar los recursos existentes (especialmente los compartidos), como, por ejemplo, los servidores de bases de datos o los servicios de directorio que no sean de producción, para reducir los costos de la nube. Para ello, siga las directrices indicadas en las prácticas recomendadas de seguridad y las normativas de la organización. Para generar valor y lograr un nivel de eficiencia de forma óptima, es fundamental volver a asignar los costos (mediante el análisis y los reintegros) a las áreas pertinentes de la empresa que impulsan el uso de recursos.

El análisis se refiere a los informes que desglosan los costos de la nube en categorías atribuibles, como consumidores, unidades de negocio, cuentas de contabilidad general u otras entidades responsables. El objetivo del análisis es mostrar a los equipos, las unidades de negocio o las personas el costo de los recursos de nube que se utilizan.

El reintegro consiste en asignar los gastos del servicio central a las unidades de costo en función de una estrategia adecuada para un proceso de gestión financiera específico. En el caso de los clientes, el reintegro cobra el costo generado desde una cuenta de servicios compartidos a diferentes categorías de costos financieros adecuadas para un proceso de generación de informes de clientes. Al establecer mecanismos de reintegro, puede informar de los costos generados por diferentes unidades de negocio, productos y equipos.

Las cargas de trabajo se pueden clasificar en esenciales y no esenciales. Según esta clasificación, utilice recursos compartidos con configuraciones generales para cargas de trabajo menos esenciales. Para optimizar aún más los costos, reserve servidores dedicados exclusivamente para cargas de trabajo esenciales. Comparta recursos o aprovisiónelos en varias cuentas para

administrarlos de manera eficaz. Incluso con distintos entornos de desarrollo, pruebas y producción, es posible compartir de forma segura sin que se vea afectada la estructura organizativa.

Para mejorar sus conocimientos y optimizar el costo y el uso de las aplicaciones en contenedores, utilice datos de asignación de costos divididos que le ayudan a asignar los costos a entidades empresariales individuales en función de cómo la aplicación consume los recursos de computación y memoria compartidos. Los datos de asignación de costos divididos le ayudan a lograr análisis y reintegros por tareas en cargas de trabajo del contenedor que se ejecutan en Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Kubernetes Service (Amazon EKS).

En el caso de arquitecturas distribuidas, cree una VPC de servicios compartidos, que ofrezca acceso centralizado a los servicios compartidos que requieren las cargas de trabajo en cada una de las VPC. En estos servicios compartidos se pueden incluir recursos como servicios de directorio o puntos de conexión de VPC. Para reducir los costos generales y administrativos, comparta los recursos desde una ubicación central en lugar de crearlos en cada VPC.

Al utilizar recursos compartidos, puede ahorrar en costos operativos, sacar el máximo partido de la utilización de los recursos y mejorar la coherencia. En un diseño de varias cuentas, puede alojar algunos servicios de AWS de forma centralizada y acceder a ellos mediante varias aplicaciones y cuentas en un centro para ahorrar costos. Puede usar AWS Resource Access Manager (AWS RAM) para compartir otros recursos comunes, como subredes de VPC y adjuntos de AWS Transit Gateway, AWS Network Firewall o canalizaciones de Amazon SageMaker AI. En un entorno de varias cuentas, utilice AWS RAM para crear un recurso una vez y compartirlo con otras cuentas.

Las organizaciones deben etiquetar los costos compartidos de forma eficaz y verificar que no tengan una parte considerable de sus costos sin etiquetar o sin asignar. Si no se asignan los costos compartidos de forma eficaz y nadie se hace responsable de la administración de estos costos, los costos compartidos de la nube pueden aumentar vertiginosamente. Debe saber dónde se han producido costos en el nivel de recursos, carga de trabajo, equipo u organización, ya que podrá comprender mejor el valor entregado en el nivel aplicable en comparación con los resultados empresariales logrados. En definitiva, las organizaciones sacan partido de los ahorros de costos como resultado del uso compartido de la infraestructura en la nube. Fomente la asignación de costos en los recursos compartidos de la nube para optimizar el gasto en la nube.

Pasos para la implementación

- Evaluación de los recursos existentes: revise las cargas de trabajo existentes que utilizan servicios similares para su carga de trabajo. En función de los componentes de la carga de trabajo, tenga en cuenta las plataformas existentes si la lógica empresarial o los requisitos técnicos lo permiten.

- Uso compartido de recursos de AWS RAM y restricción en consecuencia: utilice AWS RAM para compartir recursos con otras cuentas de AWS de su organización. Al compartir recursos, no necesita duplicarlos en varias cuentas, lo que reduce al mínimo la carga operativa del mantenimiento de los recursos. Este proceso también le ayuda a compartir de forma segura los recursos que ha creado con roles y usuarios de su cuenta y con otras Cuentas de AWS.
- Etiquetado de recursos: etiquete los recursos que sean candidatos para la elaboración de informes de costos y clasifíquelos dentro de las categorías de costos. Active estas etiquetas de recursos relacionados con los costos para la asignación de costos con el fin de proporcionar visibilidad del uso de los recursos de AWS. Céntrese en crear un nivel adecuado de especificidad con respecto a la visibilidad de los costos y el uso, e influya en los comportamientos de consumo de la nube mediante los informes de asignación de costos y el seguimiento de los KPI.

Recursos

Prácticas recomendadas relacionadas:

- SEC03-BP08 Uso compartido de recursos de forma segura en su organización

Documentos relacionados:

- What is AWS Resource Access Manager?
- Servicios de AWS que se pueden utilizar con AWS Organizations
- Recursos de AWS que se pueden compartir
- AWS Cost and Usage (CUR) Queries

Videos relacionados:

- AWS Resource Access Manager - granular access control with managed permissions
- How to design your AWS cost allocation strategy
- AWS Cost Categories

Ejemplos relacionados:

- How-to chargeback shared services: An AWS Transit Gateway example
- How to build a chargeback/showback model for Savings Plans using the CUR

- Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture
- Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data
- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

## COST 7. ¿Cómo utiliza los modelos de fijación de precios para reducir los costos?

Use el modelo de fijación de precios más apropiado para sus recursos a fin de minimizar los gastos.

Prácticas recomendadas

- COST07-BP01 Análisis de los modelos de precios
- COST07-BP02 Elección de regiones según el costo
- COST07-BP03 Selección de acuerdos de terceros con condiciones rentables
- COST07-BP04 Implementación de modelos de precios para todos los componentes de la carga de trabajo
- COST07-BP05 Análisis de modelos de precios en el nivel de la cuenta de administración

### COST07-BP01 Análisis de los modelos de precios

Analice cada componente de la carga de trabajo. Determine si el componente y los recursos se ejecutarán durante periodos extensos (por descuentos por compromiso) o periodos dinámicos y de corta ejecución (para spot o bajo demanda). Haga un análisis de la carga de trabajo mediante las recomendaciones de las herramientas de administración de costos y aplique las reglas empresariales a dichas recomendaciones para conseguir un alto rendimiento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

AWS tiene varios modelos de precios que le permiten pagar sus recursos de la manera más rentable que se adapte a las necesidades de su organización y en función del producto. Determine con sus equipos el modelo de precios más apropiado. Su modelo de precios suele constar de una combinación de varias opciones, según lo determine su disponibilidad

Las instancias bajo demanda le permiten pagar la capacidad de computación o de la base de datos por hora o por segundo (60 segundos como mínimo) en función de las instancias que ejecute, sin necesidad de asumir compromisos a largo plazo ni de hacer pagos iniciales.

Savings Plans es un modelo de precios flexible que ofrece precios económicos por el uso de Amazon EC2, Lambda y AWS Fargate a cambio de comprometerse a una cantidad constante de uso (medida en USD/hora) durante un plazo de uno o tres años.

Las instancias de spot son un mecanismo de precios de Amazon EC2 que le permite solicitar capacidad de computación sobrante a una tarifa horaria reducida (hasta un 90 % menos sobre el precio bajo demanda) sin compromiso previo.

Las instancias reservadas le ofrecen hasta un 75 % de descuento si paga por adelantado la capacidad. Para obtener más información, consulte Optimización de costos con reservas.

Puede incluir Savings Plans para los recursos asociados a los entornos de producción, calidad y desarrollo. Como alternativa, dado que los recursos de entorno aislado solo se activan cuando es necesario, puede elegir un modelo bajo demanda para los recursos de ese entorno. Utilice las instancias de spot de Amazon para reducir los costos de Amazon EC2 o utilice Savings Plans para computación para reducir los costos de Amazon EC2, Fargate y Lambda. La herramienta de recomendaciones de AWS Cost Explorer ofrece oportunidades de descuentos por compromiso con los Savings Plans.

Si ha adquirido instancias reservadas para Amazon EC2 en el pasado o ha establecido prácticas de asignación de costos en su organización, puede seguir utilizando las instancias reservadas de Amazon EC2 por el momento. Sin embargo, recomendamos elaborar una estrategia para usar Savings Plans en el futuro como mecanismo más flexible de ahorro de costos. Puede actualizar las recomendaciones de Savings Plans (SP) en AWS Cost Management para generar nuevas recomendaciones de Savings Plans en cualquier momento. Use las instancias reservadas para reducir los costos de Amazon RDS, Amazon Redshift, Amazon ElastiCache y Amazon OpenSearch Service. Los Savings Plans y las instancias reservadas están disponibles en tres modalidades de pago: puede abonarse el total por adelantado, abonarse parte por adelantado y no abonarse nada por adelantado. Utilice las recomendaciones de compra de instancias reservadas y Savings Plans de AWS Cost Explorer.

Para buscar oportunidades para cargas de trabajo de spot, use una vista por hora del uso general y busque periodos regulares de uso cambiante o de elasticidad. Puede utilizar instancias de spot para diversas aplicaciones flexibles y tolerantes a errores. Algunos ejemplos son los servidores web sin estado, los puntos de conexión de API, las aplicaciones de macrodatos y análisis, las cargas de trabajo en contenedores, CI/CD y otras cargas de trabajo flexibles.

Analice sus instancias de Amazon EC2 y Amazon RDS si pueden desactivarse cuando no las utilice (fuera de horario laboral y en fines de semana). Este enfoque le permitirá reducir costos en un

70 % o más con respecto a su uso ininterrumpido. Si tiene clústeres de Amazon Redshift que solo deben estar disponibles en momentos concretos, puede pausar el clúster y reanudarlo más tarde. Cuando se detiene el clúster de Amazon Redshift o la instancia de Amazon EC2 y Amazon RDS, la facturación de computación se detiene y solo se aplica el cargo por almacenamiento.

Tenga en cuenta que las reservas de capacidad bajo demanda (ODCR) no son un descuento en el precio. Las reservas de capacidad se cobrarán según la tarifa bajo demanda equivalente, independientemente de que ejecute instancias en la capacidad reservada o no. Deben tenerse en cuenta cuando necesite proporcionar suficiente capacidad para los recursos que tiene previsto ejecutar. Las ODCR no tienen por qué estar vinculadas a compromisos a largo plazo, ya que pueden cancelarse cuando ya no las necesite, pero también pueden beneficiarse de los descuentos que ofrecen los Savings Plans o las instancias reservadas.

Pasos para la implementación

- Análisis de la elasticidad de la carga de trabajo: mediante el grado de detalle por horas del Explorador de costos o un panel personalizado, analice la elasticidad de la carga de trabajo. Busque cambios regulares en el número de instancias que se están ejecutando. Las instancias de corta duración son candidatas para las instancias o la flota de spot.
  - Well-Architected Lab: Cost Explorer
  - Well-Architected Lab: Cost Visualization
- Revisión de los contratos de precios existentes: revise los contratos o compromisos actuales para determinar las necesidades a largo plazo. Analice lo que tiene actualmente y en qué medida se utilizan esos compromisos. Aproveche los descuentos contractuales o los acuerdos empresariales preexistentes. Los contratos Enterprise ofrecen a los clientes la opción de personalizar los acuerdos que mejor se adapten a sus necesidades. En el caso de compromisos a largo plazo, considere los descuentos por precios reservados, las instancias reservadas o Savings Plans para el tipo de instancia específico, la familia de instancias, la Región de AWS y las zonas de disponibilidad.
- Análisis de descuento por compromiso: utilice el Explorador de costos en su cuenta y revise las recomendaciones de Savings Plans e instancias reservadas. Para comprobar que está implementando las recomendaciones correctas con los descuentos y el riesgo necesarios, siga los laboratorios de Well-Architected.

Recursos

Documentos relacionados:

- Accessing Reserved Instance recommendations
- Opciones de compra de instancias
- AWS Enterprise

Videos relacionados:

- Save up to 90% and run production workloads on Spot

Ejemplos relacionados:

- Well-Architected Lab: Cost Explorer
- Well-Architected Lab: Cost Visualization
- Well-Architected Lab: Pricing Models

COST07-BP02 Elección de regiones según el costo

Los precios de los recursos pueden variar según la región. Identifique las diferencias regionales de costos y efectúe la implementación solo en las regiones con costos más elevados para cumplir los requisitos de latencia, residencia de datos y soberanía de los datos. Si tiene en cuenta el costo de la región, podrá pagar el precio global más bajo por esta carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La infraestructura de la Nube de AWS es global, está alojada en múltiples ubicaciones en todo el mundo y se basa en Regiones de AWS, zonas de disponibilidad, zonas locales, AWS Outposts y zonas de Wavelength. Una región es una ubicación física en el mundo y cada región es un área geográfica independiente en la que AWS tiene varias zonas de disponibilidad. Las zonas de disponibilidad, que son varias ubicaciones aisladas en cada región, constan de uno o varios centros de datos separados, cada uno de ellos con alimentación, redes y conectividad redundantes.

Cada Región de AWS opera según las condiciones del mercado local y el precio de los recursos es distinto en cada región debido a las diferencias del costo del terreno, la fibra, la electricidad y los impuestos, por ejemplo. Elija una región específica en la que desee aplicar un componente de su solución o la solución completa a fin de poder ejecutar al precio más bajo posible a nivel mundial. Utilice la Calculadora de AWS para calcular los costos de su carga de trabajo en varias regiones

mediante la búsqueda de servicios por tipo de ubicación (región, zona de Wavelength y zona local) y región.

Al diseñar soluciones, una práctica recomendada es intentar colocar los recursos de computación más cerca de los usuarios a fin de brindar una latencia más baja y una soberanía de datos sólida. Seleccione la ubicación geográfica en función de los requisitos de su empresa, privacidad de datos, rendimiento y seguridad. En el caso de aplicaciones con usuarios finales en todo el mundo, utilice varias ubicaciones.

Recurra a las regiones que ofrecen precios más bajos por los servicios de AWS para implementar sus cargas de trabajo si no tiene obligaciones en materia de privacidad de datos, seguridad y requisitos de empresa. Por ejemplo, si su región predeterminada es Asia-Pacífico (Sídney) (`ap-southwest-2`) y si no existen restricciones (privacidad de los datos o seguridad, por ejemplo) para utilizar otras regiones, implementar instancias de Amazon EC2 no críticas (desarrollo y pruebas) en la región Este de EE. UU. (Norte de Virginia) (`us-east-1`) tendrá menos costos.

| | *Cumplimiento* | *Latencia* | *Coste* | *Servicios/características* |
|---|---|---|---|---|
| *Región 1* | ✓ | 15 ms | $$ | ✓ |
| *Región 2* | ✓ | 20 ms | $$$ | X |
| *Región 3* | ✓ | 80 ms | $ | ✓ |
| *Región 4* | ✓ | 15 ms | $$ | ✓ |
| *Región 5* | ✓ | 20 ms | $$$ | X |
| *Región 6* | ✓ | 15 ms | $ | ✓ |
| *Región 7* | ✓ | 80 ms | $ | ✓ |
| *Región 8* | ✓ | 15 ms | $ | X |

Tabla matricial de características de las regiones

En la tabla matricial anterior se nos muestra que la Región 6 es la mejor opción para este escenario específico, porque la latencia es baja en comparación con otras regiones, el servicio está disponible y es la región menos cara.

Pasos para la implementación

- Revisión de los precios de la Región de AWS: analice los costos de la carga de trabajo de la región actual. A partir de los costos más elevados por servicio y tipo de uso, calcule los costos en otras regiones que estén disponibles. Si el ahorro previsto supera el costo de trasladar el componente o la carga de trabajo, migre a la nueva región.
- Revisión de los requisitos para las implementaciones de varias regiones: analice los requisitos y las obligaciones de su empresa (privacidad de los datos, seguridad o rendimiento) para averiguar si existe alguna restricción que le impida utilizar varias regiones. Si no hay obligaciones que restrinjan el uso de una sola región, utilice varias.
- Revisión de la transferencia de datos requerida: tenga en cuenta los costos de transferencia de datos al seleccionar las regiones. Mantenga sus datos cerca de su cliente y de los recursos. Seleccione Regiones de AWS menos costosas donde fluyan los datos y donde la transferencia de datos sea mínima. En función de los requisitos empresariales en materia de transferencia de datos, puede utilizar Amazon CloudFront, AWS PrivateLink, AWS Direct Connect y AWS Virtual Private Network para reducir los costos de red, y mejorar el rendimiento y la seguridad.

Recursos

Documentos relacionados:

- Accessing Reserved Instance recommendations
- Precios de Amazon EC2
- Opciones de compra de instancias
- Tabla de regiones

Videos relacionados:

- Save up to 90% and run production workloads on Spot

Ejemplos relacionados:

- Overview of Data Transfer Costs for Common Architectures
- Cost Considerations for Global Deployments
- What to Consider when Selecting a Region for your Workloads

### COST07-BP03 Selección de acuerdos de terceros con condiciones rentables

Los acuerdos y condiciones rentables garantizan que el costo de estos servicios vaya a la par de los beneficios que proporcionan. Seleccione acuerdos y precios que se escalen cuando proporcionen beneficios adicionales a la organización.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

#### Guía para la implementación

Hay varios productos en el mercado que pueden ayudarlo a administrar los costos en sus entornos de nube. Puede que haya algunas diferencias en lo que se refiere a las características que dependen de los requisitos del cliente. Por ejemplo, puede que algunos se centren en la gobernanza de costos o la visibilidad de costos y otros en la optimización de costos. Un factor clave para que la optimización de los costos y la gobernanza sean efectivas es utilizar la herramienta adecuada con las características necesarias y el modelo de precios correcto. Estos productos tienen diferentes modelos de precios. Algunos cobran un porcentaje determinado de la factura mensual, mientras que otros cobran un porcentaje del ahorro que se consigue. Lo ideal es que pague solo lo que necesita.

Al utilizar soluciones o servicios de terceros en la nube, es importante que las estructuras de precios se ajusten a los resultados deseados. Los precios deben ir a la par de los resultados y el valor que aportan. Un ejemplo de ello es el software que se lleva una parte del ahorro que proporciona: cuanto más ahorra (resultado), más cobra. Los acuerdos de licencias en los que paga más a medida que aumentan sus gastos no siempre le convienen para optimizar costos. Sin embargo, si el proveedor ofrece ventajas claras en todas las partes de su factura, este aumento de tarifa podría estar justificado.

Por ejemplo, una solución que proporciona recomendaciones para Amazon EC2 y cobra un porcentaje de toda la factura podría ser más cara si usa otros servicios que no generan ningún beneficio. Otro ejemplo es un servicio administrado que se cobra a un porcentaje del costo de los recursos que se administran. Un mayor tamaño de la instancia no tiene por qué requerir un mayor esfuerzo de administración, aunque sí se podría cobrar más. A fin de impulsar la eficiencia, asegúrese de que, en estos acuerdos de precios del servicio, se incluya un programa o características de optimización de costos en su servicio.

Los clientes podrían encontrar en el mercado estos productos más avanzados o fáciles de usar. Debe considerar el costo de estos productos y pensar en los posibles resultados de optimización de costos a largo plazo.

Pasos para la implementación

- Análisis de los acuerdos y condiciones de terceros: revise los precios de los acuerdos de terceros. Haga modelados de los diferentes niveles de uso y tenga en cuenta nuevos costos, como el uso de nuevos servicios o incrementos en los servicios actuales debido al crecimiento de la carga de trabajo. Decida si los costos adicionales proporcionan los beneficios necesarios para su empresa.

Recursos

Documentos relacionados:

- Accessing Reserved Instance recommendations
- Opciones de compra de instancias

Videos relacionados:

- Save up to 90% and run production workloads on Spot

COST07-BP04 Implementación de modelos de precios para todos los componentes de la carga de trabajo

Al ejecutar recursos de forma permanente se debe utilizar la capacidad reservada como los Savings Plans o las instancias reservadas. La capacidad a corto plazo se configura con instancias o una flota de spot. Las instancias bajo demanda solo se usan para cargas de trabajo a corto plazo que no se pueden interrumpir y que no se ejecutan lo suficiente como para tener capacidad reservada, es decir, de un 25 a un 75 % del periodo, según el tipo de recurso.

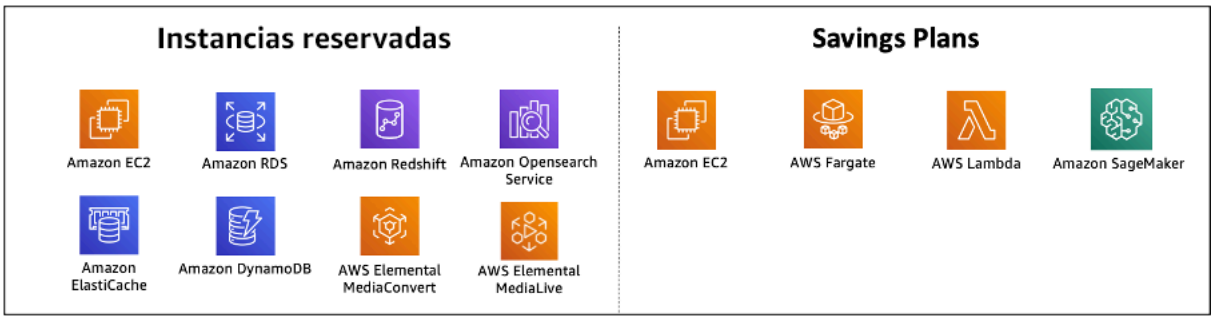
Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Para mejorar la rentabilidad, AWS proporciona varias recomendaciones de compromiso basadas en el uso pasado. Estas recomendaciones pueden servirle para saber lo que puede ahorrar y cómo se utilizará el compromiso. Puede utilizar estos servicios como instancias bajo demanda o spot, o comprometerse por un periodo de tiempo determinado y reducir sus costos bajo demanda con instancias reservadas y Savings Plans (SP). Necesita conocer, no solo los componentes de cada carga de trabajo y los múltiples servicios de AWS, sino también los descuentos por compromiso, las opciones de compra y las instancias de spot de estos servicios para optimizar su carga de trabajo.

Tenga en cuenta los requisitos de los componentes de su carga de trabajo e infórmese de los diferentes modelos de precios de estos servicios. Defina el requisito de disponibilidad de estos componentes. Determine si hay varios recursos independientes que ejecuten la función en la carga de trabajo y cuáles son los requisitos de la carga de trabajo a lo largo del tiempo. Compare el costo de los recursos con el modelo de precios bajo demanda predeterminado y otros modelos aplicables. Tenga en cuenta cualquier cambio potencial en los recursos o en los componentes de la carga de trabajo.

Veamos, por ejemplo, esta arquitectura de aplicaciones web en AWS. Este ejemplo de carga de trabajo consta de varios servicios de AWS, como Amazon Route 53, AWS WAF, Amazon CloudFront, instancias de Amazon EC2, instancias de Amazon RDS, equilibradores de carga, almacenamiento de Amazon S3 y Amazon Elastic File System (Amazon EFS). Debe revisar cada uno de estos servicios e identificar las posibles oportunidades de ahorro de costos con diferentes modelos de precios. Algunos de ellos podrían ser elegibles para RI o SP, mientras que otros podrían estar disponibles solo bajo demanda. Como se muestra en la siguiente imagen, algunos de los servicios de AWS pueden comprometerse mediante instancias reservadas o Savings Plans.




Servicios de AWS comprometidos mediante instancias reservadas y Savings Plans

Pasos para la implementación

- Implementación de modelos de precios: con los resultados de sus análisis, compre Savings Plans o instancias reservadas o implemente instancias de spot. Si se trata de su primera compra de compromiso, elija las cinco o diez mejores recomendaciones de la lista y, a continuación, supervise y analice los resultados durante uno o dos meses. La AWS Cost Management Console le guiará a lo largo del proceso. Revise las recomendaciones de instancias reservadas o Savings Plans desde la consola, personalice las recomendaciones (tipo, pago y plazo), revise el compromiso por hora (por ejemplo, 20 USD por hora) y, a continuación, agréguelas a la cesta. Los descuentos se aplican automáticamente al uso elegible. Compre una pequeña cantidad de descuentos por compromiso en ciclos regulares (por ejemplo, cada 2 semanas o mensualmente). Implemente instancias de spot para las cargas de trabajo que se puedan interrumpir o no tengan estado. Por

último, seleccione instancias de Amazon EC2 bajo demanda y asigne recursos para los requisitos restantes.

- Ciclo de revisión de la carga de trabajo: implemente un ciclo de revisión de la carga de trabajo que analice específicamente la cobertura del modelo de precios. Cuando la carga de trabajo tenga la cobertura requerida, compre descuentos por compromiso adicionales parcialmente (cada pocos meses) o a medida que cambie el uso en la organización.

Recursos

Documentos relacionados:

- Understanding your Savings Plans recommendations
- Accessing Reserved Instance recommendations
- Cómo adquirir instancias reservadas
- Opciones de compra de instancias
- Spot Instances
- Reservation models for other AWS services
- Savings Plans Supported Services

Videos relacionados:

- Save up to 90% and run production workloads on Spot

Ejemplos relacionados:

- ¿Qué debo tener en cuenta antes de comprar un Savings Plan?
- ¿Cómo puedo usar el Explorador de costos para analizar mis gastos y mi consumo?

COST07-BP05 Análisis de modelos de precios en el nivel de la cuenta de administración

Consulte las herramientas de facturación y administración de costos y vea los descuentos recomendados con compromisos y reservas para hacer análisis periódicos en el nivel de la cuenta de administración.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

La creación periódica de modelos de costos lo ayuda a implementar oportunidades de optimización en múltiples cargas de trabajo. Por ejemplo, si varias cargas de trabajo usan instancias bajo demanda en un nivel agregado, el riesgo de cambio es menor e implementar un descuento basado en el compromiso puede tener un costo general inferior. Se recomienda hacer análisis en ciclos regulares de dos semanas a un mes. De este modo, podrá hacer compras de ajustes pequeños para que sus modelos de precios puedan seguir evolucionando a medida que cambien sus cargas de trabajo y sus componentes.

Utilice el AWS Cost Explorer para buscar descuentos por compromiso en su cuenta de administración. Las recomendaciones en el nivel de cuenta de administración se calculan teniendo en cuenta el uso en todas las cuentas de su organización de AWS que tengan activadas las instancias reservadas o Savings Plans (SP). También se calculan cuando se activa la opción de compartir descuentos para recomendar un compromiso que maximice los ahorros en todas las cuentas.

Aunque la compra en el nivel de cuenta de administración permite conseguir el máximo ahorro en muchos casos, puede haber situaciones en las que podría plantearse la posibilidad de comprar SP en el nivel de la cuenta vinculada cuando, por ejemplo, desee que los descuentos se apliquen primero al uso de esa cuenta vinculada en particular. Las recomendaciones para las cuentas de miembros se calculan en el nivel de cuenta individual para maximizar el ahorro de cada cuenta. Si su cuenta es propietaria de compromisos de instancias reservadas y Savings Plans, se aplicarán en este orden:

1. Instancia reservada de zona
2. Instancia reservada estándar
3. Instancia reservada convertible
4. Savings Plans para instancias
5. Savings Plans para computación

Si compra un plan de Savings Plans en el nivel de cuenta de administración, el ahorro se aplicará en función del porcentaje de descuento más alto al más bajo. Los Savings Plans de nivel de cuenta de administración examinan todas las cuentas vinculadas y aplican el ahorro allí donde el descuento sea más alto. Si desea restringir dónde se aplica el ahorro, puede comprar un plan de Savings Plans en el nivel de cuenta vinculada para que, cada vez que esa cuenta utilice servicios de computación que cumplan los requisitos, el descuento se aplique primero allí. Cuando la cuenta no utilice servicios

de computación que cumplan los requisitos, el descuento se compartirá entre las demás cuentas vinculadas de la misma cuenta de administración. La opción de compartir descuentos está activada de forma predeterminada, pero se puede desactivar si es necesario.

En una familia de facturación consolidada, los Savings Plans se aplican primero al uso de la cuenta del propietario y, después, al uso de otras cuentas. Esto ocurre solo si tiene activado el uso compartido. Sus Savings Plans se aplican primero a su porcentaje de ahorro más alto. Si hay varios usos con porcentajes de ahorro iguales, los Savings Plans se aplican al primer uso con la tarifa más baja de Savings Plans. Los Savings Plans seguirán aplicándose hasta que no queden más usos o hasta que se agote su compromiso. El uso restante se cobra según la tarifa bajo demanda. Puede actualizar las recomendaciones de Savings Plans (SP) en Administración de costos de AWS para generar nuevas recomendaciones de Savings Plans en cualquier momento.

Tras analizar la flexibilidad de las instancias, puede comprometerse siguiendo las recomendaciones. Cree modelos de costos al analizar los costos de la carga de trabajo a corto plazo con posibles opciones de recursos diferentes, además de analizar los modelos de precios de AWS y su alineación con los requisitos empresariales para averiguar el costo total de propiedad y las oportunidades de optimización de costos.

Pasos para la implementación

Análisis de descuento por compromiso: utilice el Explorador de costos en su cuenta y consulte las recomendaciones de Savings Plans e instancias reservadas. Asegúrese de que entiende las recomendaciones de Savings Plans y calcule el gasto y el ahorro mensual. Revise las recomendaciones en el nivel de cuenta de administración que se calculan teniendo en cuenta el uso en todas las cuentas de miembro de su organización de AWS que tengan activadas las instancias reservadas o el reparto de descuentos de Savings Plans para obtener el máximo ahorro en todas las cuentas. Puede verificar que ha implementado las recomendaciones correctas con los descuentos y riesgos necesarios si sigue los laboratorios de Well-Architected.

Recursos

Documentos relacionados:

- ¿Cómo funcionan los precios de AWS?
- Opciones de compra de instancias
- Saving Plan Overview
- Saving Plan recommendations

- Accessing Reserved Instance recommendations
- Understanding your Saving Plans recommendation
- How Savings Plans apply to your AWS usage
- Savings Plans con facturación unificada
- Turning on shared reserved instances and Savings Plans discounts

Videos relacionados:

- Save up to 90% and run production workloads on Spot

Ejemplos relacionados:

- Qué debo tener en cuenta antes de comprar un Savings Plan?
- How can I use rolling Savings Plans to reduce commitment risk?
- When to Use Spot Instances

## COST 8. ¿Cómo planifica los gastos de transferencia de datos?

Compruebe que planifique y supervise los cargos de transferencia de datos para que pueda tomar decisiones en cuanto al diseño y minimizar los costos. Un cambio de diseño pequeño, pero efectivo, puede reducir drásticamente sus costos operativos con el tiempo.

Prácticas recomendadas

- COST08-BP01 Modelado de transferencia de datos
- COST08-BP02 Selección de componentes para optimizar el costo de la transferencia de datos
- COST08-BP03 Implementación de servicios para reducir los costos de transferencia de datos

### COST08-BP01 Modelado de transferencia de datos

Reúna los requisitos de la organización y haga un modelado de transferencia de datos de la carga de trabajo y de cada uno de sus componentes. Se identifica el punto de costo más bajo para los requisitos de transferencia de datos actuales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Cuando se diseña una solución en la nube, las tarifas de transferencia de datos suelen olvidarse. Esto se debe a la costumbre de diseñar la arquitectura utilizando centros de datos en las instalaciones o a la falta de conocimientos. Los cargos por transferencia de datos en AWS vienen determinados por el origen, el destino y el volumen del tráfico. Si se tienen en cuenta estas tarifas durante la fase de diseño, es posible ahorrar costos. Saber dónde se produce la transferencia de datos en su carga de trabajo, el costo de la transferencia y su beneficio asociado es muy importante para calcular con precisión el costo total de propiedad (TCO). De este modo, podrá tomar una decisión informada para modificar o aceptar la decisión arquitectónica. Por ejemplo, podría tener una configuración de la zona de disponibilidad múltiple donde replicar los datos entre las zonas de disponibilidad.

Después, modele los componentes de los servicios que transfieren los datos en su carga de trabajo y determine que se trata de un costo aceptable (similar al de pagar por la computación y el almacenamiento en la zona de disponibilidad) para lograr la fiabilidad y resiliencia requeridas. Debe modelar los costos según los distintos niveles de uso. El uso de la carga de trabajo puede cambiar con el tiempo y algunos servicios podrían ser más rentables en diferentes niveles.

Al modelar la transferencia de datos, piense en la cantidad de datos que se ingieren y de dónde provienen esos datos. Además, considere cuántos datos se procesan y cuánta capacidad de almacenamiento o computación se necesita. Durante el modelado, siga las prácticas recomendadas de redes para la arquitectura de la carga de trabajo con el fin de optimizar los costos potenciales de la transferencia de datos.

Calculadora de precios de AWS puede ayudarlo a conocer los costos estimados de servicios de AWS específicos y la transferencia de datos esperada. Si ya tiene una carga de trabajo en ejecución (con fines de prueba o en un entorno de preproducción), utilice el AWS Cost Explorer o el AWS Cost and Usage Report (CUR) para comprender y modelar sus costos de transferencia de datos. Configure una prueba de concepto (POC) o pruebe su carga de trabajo y ejecute una prueba con una carga simulada realista. Puede modelar sus costos con distintas demandas de carga de trabajo.

Pasos para la implementación

- Identificación de los requisitos: ¿Cuál es el objetivo principal y los requisitos empresariales para la transferencia de datos planificada entre el origen y el destino? ¿Cuál es el resultado empresarial que se espera al final? Recopile los requisitos empresariales y defina los resultados esperados.

- Identificación del origen y el destino: ¿Cuál es el origen y el destino de los datos para la transferencia de datos, por ejemplo, dentro de Regiones de AWS a los servicios de AWS o hacia Internet?
  - Data transfer within an Región de AWS
  - Data transfer between Regiones de AWS
  - Data transfer out to the internet
- Identificación de las clasificaciones de datos: ¿Cuál es la clasificación de datos para esta transferencia de datos? ¿Qué tipo de datos son? ¿Qué tamaño tienen los datos? ¿Con qué frecuencia se deben transferir los datos? ¿Los datos son confidenciales?
- Identificación de los servicios o herramientas de AWS que se van a utilizar: ¿Qué servicios de AWS se utilizan para esta transferencia de datos? ¿Es posible utilizar un servicio ya aprovisionado para otra carga de trabajo?
- Cálculo de los costos de transferencia de datos: utilice Precios de AWS, el modelo de transferencia de datos que creó anteriormente, para calcular los costos de transferencia de datos para la carga de trabajo. Calcule los costos de transferencia de datos en distintos niveles de uso para los incrementos y las reducciones del uso de la carga de trabajo. Si hay múltiples opciones para la arquitectura de la carga de trabajo, calcule el costo de cada opción para compararlas.
- Enlace de los costos con los resultados: especifique el resultado obtenido por la carga de trabajo para cada costo de transferencia de datos incurrido. Si es una transferencia entre componentes, puede deberse a un desacoplamiento y, si es entre zonas de disponibilidad, puede deberse a la redundancia.
- Creación de un modelado de transferencia de datos: después de recopilar toda la información, cree un modelo de transferencia de datos base conceptual para múltiples casos de uso y diferentes cargas de trabajo.

Recursos

Documentos relacionados:

- AWS caching solutions
- AWS Precios de
- Precios de Amazon EC
- Precios de Amazon VPC
- Understanding data transfer charges

Videos relacionados:

- Monitoring and Optimizing Your Data Transfer Costs
- S3 Transfer Acceleration

Ejemplos relacionados:

- Overview of Data Transfer Costs for Common Architectures
- AWS Prescriptive Guidance for Networking

COST08-BP02 Selección de componentes para optimizar el costo de la transferencia de datos

Se seleccionan todos los componentes y se diseña la arquitectura para reducir los costos de transferencia de datos. Incluye el uso de componentes como la optimización de la red de área extendida (WAN) y las configuraciones de varias zonas de disponibilidad (AZ).

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La arquitectura para la transferencia de datos minimiza los costos de transferencia de datos. Para ello, es posible que deba usar redes de entrega de contenido para colocar los datos más cerca de los usuarios, o bien enlaces de red dedicados entre sus instalaciones y AWS. También puede usar la optimización de WAN y de las aplicaciones para reducir la cantidad de datos que se transfieren entre los componentes.

Al transferir datos a la Nube de AWS, o dentro de ella, es muy importante conocer el destino en función de los diversos casos de uso, la naturaleza de los datos y los recursos de red disponibles a fin de seleccionar los servicios de AWS correctos para optimizar la transferencia de datos. AWS ofrece una amplia gama de servicios de transferencia de datos diseñados para satisfacer diversos requisitos de migración de datos. Seleccione las opciones de almacenamiento de datos y transferencia de datos adecuadas en función de las necesidades empresariales de su organización.

Al planificar o revisar la arquitectura de la carga de trabajo, tenga en cuenta lo siguiente:

- Uso de los puntos de conexión de VPC en AWS: los puntos de conexión de VPC habilitan conexiones privadas entre la VPC y los servicios de AWS admitidos. Esto le permite evitar el uso de la red pública de Internet, que puede dar lugar a costos de transferencia de datos.

- Uso de una puerta de enlace de NAT: utilice una puerta de enlace de NAT para que las instancias de una subred privada puedan conectarse a Internet o a servicios fuera de la VPC. Compruebe si los recursos que hay detrás de la puerta de enlace de NAT que envían la mayor cantidad de tráfico se encuentran en la misma zona de disponibilidad que la puerta de enlace de NAT. Si no lo están, cree nuevas puertas de enlace de NAT en la misma zona de disponibilidad que el recurso para reducir los cargos por transferencia de datos entre AZ.
- Uso de AWS Direct Connect: Direct Connect evita la red pública de Internet y establece una conexión privada y directa entre la red en las instalaciones y AWS. Esto puede resultar más rentable y coherente que la transferencia de grandes volúmenes de datos a través de Internet.
- Imposibilidad de transferir datos a través de las fronteras regionales: las transferencias de datos entre Regiones de AWS (de una región a otra) suelen conllevar gastos. La vía multirregional debería ser una decisión muy meditada. Para obtener más información, consulte Multi-Region scenarios.
- Supervisión de la transferencia de datos: utilice los registros de flujo de Amazon CloudWatch y VPC para recopilar detalles sobre la transferencia de datos y el uso de la red. Analice la información de tráfico de red de sus VPC, como la dirección IP o el rango, que va y viene de las interfaces de red.
- Análisis del uso de la red: utilice herramientas de medición e informes, como el AWS Cost Explorer, CUDOS Dashboards o CloudWatch, para comprender el costo de transferencia de datos de su carga de trabajo.

Pasos para la implementación

- Selección de componentes para la transferencia de datos: use el modelo de transferencia de datos que se explica en COST08-BP01 Modelado de transferencia de datos para centrarse en dónde se encuentran los mayores costos de transferencia de datos o dónde estarían si cambia el uso de la carga de trabajo. Busque arquitecturas alternativas o componentes adicionales que eliminen o reduzcan la necesidad de transferir datos (o que reduzcan su costo).

Recursos

Prácticas recomendadas relacionadas:

- COST08-BP01 Modelado de transferencia de datos
- COST08-BP03 Implementación de servicios para reducir los costos de transferencia de datos

Documentos relacionados:

- Migración de datos a la nube
- AWS caching solutions
- Deliver content faster with Amazon CloudFront

Ejemplos relacionados:

- Overview of Data Transfer Costs for Common Architectures
- AWS Network Optimization Tips
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format

COST08-BP03 Implementación de servicios para reducir los costos de transferencia de datos

Implemente servicios para reducir la transferencia de datos. Por ejemplo, utilice ubicaciones periféricas o redes de entrega de contenido (CDN) para ofrecer contenido a los usuarios finales, cree capas de almacenamiento en caché delante de sus servidores de aplicaciones o bases de datos y utilice conexiones de red dedicadas en lugar de VPN para conectarse a la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Existen varios servicios de AWS que pueden ayudarlo a optimizar el uso de la transferencia de datos de la red. Según los componentes de la carga de trabajo, el tipo y la arquitectura de la nube, estos servicios pueden ayudar a comprimir, almacenar en caché y compartir y distribuir su tráfico en la nube.

- Amazon CloudFront es una red global de entrega de contenido que brinda datos con baja latencia y altas velocidades de transferencia. Almacena en caché los datos en ubicaciones periféricas en todo el mundo, lo que reduce la carga de sus recursos. Mediante CloudFront, puede reducir el esfuerzo administrativo que supone entregar contenido a una cantidad grande de usuarios a nivel mundial y hacerlo con una latencia mínima. El Paquete de promociones en seguridad puede ayudarlo a ahorrar hasta un 30 % en el uso de CloudFront si planea un aumento de este con el tiempo.

- AWS Direct Connect permite establecer una conexión de red dedicada con AWS. Esto puede reducir los costos de red, aumentar el ancho de banda y brindar una experiencia de red más coherente que las conexiones basadas en Internet.
- Site-to-Site VPN le permite establecer una conexión segura y privada entre su red privada y la red global de AWS. Es ideal para oficinas pequeñas o socios empresariales porque proporciona una conectividad simplificada, además de ser un servicio elástico y completamente administrado.
- Los puntos de conexión de VPC permiten la conexión entre los servicios de AWS a través de la red privada y se pueden usar para reducir los costos de la transferencia de datos pública y los costos de la puerta de enlace de NAT. Los puntos de conexión de VPC de puerta de enlace no tienen cargos por hora y son compatibles con Amazon S3 y Amazon DynamoDB. Los puntos de conexión de VPC de interfaz están proporcionados por AWS PrivateLink y tienen una tarifa por hora y un costo de uso por GB.
- Las puertas de enlace de NAT ofrecen escalado y administración integrados para reducir los costos, a diferencia de una instancia de NAT independiente. Coloque las puertas de enlace de NAT en las mismas zonas de disponibilidad que las instancias de alto tráfico y plantéese usar puntos de conexión de VPC para las instancias que necesiten acceder a Amazon DynamoDB o Amazon S3 a fin de reducir los costos de transferencia y procesamiento de datos.
- Utilice dispositivos de AWS Snow Family con recursos informáticos para recopilar y procesar datos en la periferia. Los dispositivos de AWS Snow Family (Snowball Edge, Snowball Edge y Snowmobile) le permiten mover petabytes de datos a la Nube de AWS, una opción rentable y sin conexión a Internet.

Pasos para la implementación

- Implementación de los servicios: seleccione los servicios de red de AWS aplicables en función del tipo de carga de trabajo del servicio mediante el modelado de transferencia de datos y la revisión de registros de flujo de VPC. Observe dónde están los mayores costos y los mayores flujos de volumen. Revise los servicios de AWS y evalúe si existe un servicio que reduzca o elimine la transferencia, especialmente en relación con la entrega de contenido y las redes. Busque también servicios de almacenamiento en caché donde haya acceso repetido a los datos o grandes cantidades de datos.

Recursos

Documentos relacionados:

- AWS Direct Connect
- Explore nuestros productos de AWS
- AWS caching solutions
- Amazon CloudFront
- AWS Snow Family
- Paquete de promociones en seguridad de Amazon CloudFront

Videos relacionados:

- Monitoring and Optimizing Your Data Transfer Costs
- AWS Cost Optimization Series: CloudFront
- How can I reduce data transfer charges for my NAT gateway?

Ejemplos relacionados:

- How-to chargeback shared services: An AWS Transit Gateway example
- Understand AWS data transfer details in depth from cost and usage report using Athena query and QuickSight
- Overview of Data Transfer Costs for Common Architectures
- Using AWS Cost Explorer to analyze data transfer costs
- Cost-Optimizing your AWS architectures by utilizing Amazon CloudFront features
- How can I reduce data transfer charges for my NAT gateway?

## Administración de la demanda y suministro de recursos

Pregunta

- COST 9. ¿Cómo administra la demanda y aprovisiona los recursos?

### COST 9. ¿Cómo administra la demanda y aprovisiona los recursos?

Para que la carga de trabajo tenga un gasto y un rendimiento equilibrados, compruebe que se utiliza todo aquello en lo que invierte y evite desaprovechar significativamente las instancias. Una métrica de uso sesgada en cualquier dirección tiene un efecto adverso en su organización, ya sea

en los costos operativos (rendimiento degradado debido al sobreúso) o en los gastos de AWS desperdiciados (debido al sobreaprovisionamiento).

Prácticas recomendadas

- COST09-BP01 Análisis de la demanda de la carga de trabajo
- COST09-BP02 Implementación de un búfer o una limitación para administrar la demanda
- COST09-BP03 Suministro dinámico de recursos

COST09-BP01 Análisis de la demanda de la carga de trabajo

Analice la demanda de la carga de trabajo a lo largo del tiempo. Compruebe que el análisis cubra las tendencias estacionales y represente con precisión las condiciones de servicio durante toda la vida útil de la carga de trabajo. El análisis debe reflejar los posibles beneficios; por ejemplo, el tiempo empleado es proporcional al costo de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

El análisis de la demanda de la carga de trabajo para la computación en la nube implica comprender los patrones y las características de las tareas de computación que se inician en el entorno de la nube. Este análisis ayuda a los usuarios a optimizar la asignación de recursos, administrar los costos y verificar que el rendimiento cumpla con los niveles requeridos.

Debe conocer los requisitos de la carga de trabajo. Los requisitos de su organización deben indicar los tiempos de respuesta de la carga de trabajo frente a las solicitudes. El tiempo de respuesta se puede usar para determinar si la demanda está administrada o si el suministro de recursos debe cambiar para adaptarse a la demanda.

El análisis debe incluir la previsibilidad y la repetibilidad de la demanda, el ritmo y la cantidad de cambio en la demanda. Haga el análisis a lo largo de un periodo suficiente de tiempo para que incorpore variantes estacionales, como un procesamiento de final de mes o los picos de las vacaciones.

La actividad de análisis debe reflejar los posibles beneficios de la implementación del escalado. Consulte el costo total previsto del componente y cualquier incremento o descenso del uso, así como el costo durante el periodo de vida de la carga de trabajo.

Estos son algunos aspectos clave que se deben tener en cuenta al hacer un análisis de la demanda de la carga de trabajo para la computación en la nube:

1. Métricas de uso y rendimiento de los recursos: analice cómo se utilizan los recursos de AWS a lo largo del tiempo. Determine los patrones de uso en las horas punta y fuera de las horas punta para optimizar la asignación de recursos y las estrategias de escalado. Supervise las métricas de rendimiento, como los tiempos de respuesta, la latencia, el rendimiento y las tasas de error. Estas métricas ayudan a evaluar el estado general y la eficiencia de la infraestructura de la nube.
2. Comportamiento del escalado de usuarios y aplicaciones: comprenda el comportamiento de los usuarios y cómo afecta a la demanda de carga de trabajo. El análisis de los patrones del tráfico de usuarios ayuda a mejorar la entrega de contenido y la capacidad de respuesta de las aplicaciones. Analice cómo se escalan las cargas de trabajo a medida que aumenta la demanda. Determine si los parámetros de escalado automático están configurados de forma correcta y eficaz para gestionar las fluctuaciones de carga.
3. Tipos de cargas de trabajo: identifique los diferentes tipos de cargas de trabajo que se ejecutan en la nube, como el procesamiento por lotes, el procesamiento de datos en tiempo real, las aplicaciones web, las bases de datos o el machine learning. Cada tipo de carga de trabajo puede tener requisitos de recursos y perfiles de rendimiento diferentes.
4. Acuerdos de nivel de servicio (SLA): compare el rendimiento real con los SLA para garantizar el cumplimiento e identificar las áreas que necesitan mejoras.

Puede utilizar Amazon CloudWatch para recopilar métricas y hacer un seguimiento de ellas, supervisar archivos de registros, establecer alarmas y reaccionar automáticamente a los cambios en sus recursos AWS. También puede utilizar Amazon CloudWatch para obtener visibilidad de todo el sistema sobre la utilización de recursos, el rendimiento de las aplicaciones y el estado de funcionamiento.

Con AWS Trusted Advisor, puede aprovisionar sus recursos conforme a las prácticas recomendadas para mejorar el rendimiento y la fiabilidad del sistema, aumentar la seguridad y buscar oportunidades para ahorrar dinero. También puede desactivar las instancias que no son de producción y utilizar Amazon CloudWatch y Auto Scaling para adaptarlas a los aumentos o reducciones de la demanda.

Por último, puede utilizar AWS Cost Explorer o Quick con el archivo de AWS Cost and Usage Report (CUR) o los registros de la aplicación para hacer un análisis avanzado de la demanda de la carga de trabajo.

En general, un análisis integral de la demanda de la carga de trabajo permite a las organizaciones tomar decisiones informadas sobre el aprovisionamiento, el escalado y la optimización de los recursos, lo que se traduce en un mejor rendimiento, rentabilidad y satisfacción de los usuarios.

Pasos para la implementación

- Análisis de los datos de la carga de trabajo existente: analice los datos de la carga de trabajo existente, las versiones anteriores de la carga de trabajo o los patrones de uso previstos. Utilice Amazon CloudWatch, los archivos de registro y los datos de supervisión para obtener información sobre cómo se utilizó la carga de trabajo. Analice un ciclo completo de la carga de trabajo y recopile datos de los cambios estacionales, como los eventos de final de mes o de final de año. El esfuerzo reflejado en este análisis debe mostrar las características de la carga de trabajo. Debe ponerse mayor empeño en las cargas de trabajo de mayor valor con mayores cambios en la demanda. Debe ponerse menor empeño en las cargas de trabajo de menor valor con menores cambios en la demanda.
- Previsión de los factores externos: reúnase con miembros de equipos de toda la organización que puedan influir en la demanda de la carga de trabajo o cambiarla. Estos equipos suelen ser los de Ventas, Marketing o Desarrollo empresarial. Colabore con ellos para conocer los ciclos en los que operan y si hay eventos especiales que puedan cambiar la demanda de la carga de trabajo. Haga una previsión de la demanda de la carga de trabajo con estos datos.

Recursos

Documentos relacionados:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling
- AWS Instance Scheduler
- Getting started with Amazon SQS
- AWS Cost Explorer
- Quick

Ejemplos relacionados:

- Monitor, Track and Analyze for cost optimization
- Searching and analyzing logs in CloudWatch

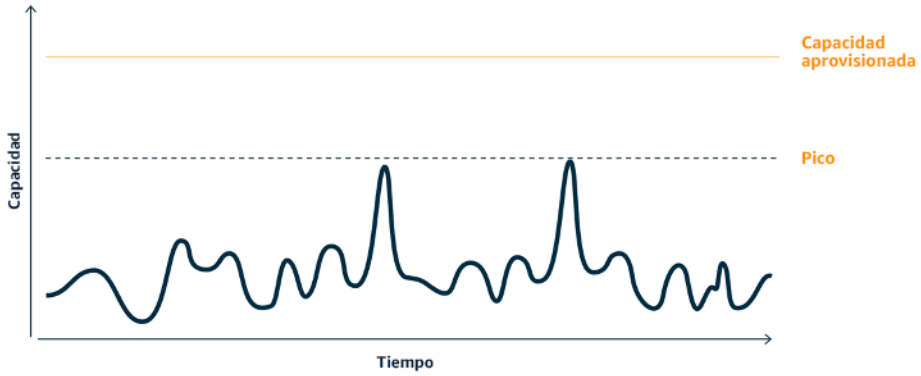
COST09-BP02 Implementación de un búfer o una limitación para administrar la demanda

El almacenamiento en búfer y la limitación modifican la demanda de la carga de trabajo y suavizan los picos. Implemente limitaciones cuando sus clientes lleven a cabo reintentos. Implemente el almacenamiento en búfer para almacenar la solicitud y aplazar el procesamiento para más adelante. Verifique que las limitaciones y los búferes se hayan diseñado de tal manera que los clientes reciban una respuesta en el tiempo requerido.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La implementación de un búfer o limitación es crucial en la computación en la nube para administrar la demanda y reducir la capacidad aprovisionada necesaria para la carga de trabajo. Para conseguir un rendimiento óptimo, es esencial evaluar la demanda total, incluidos los picos, el ritmo de cambio en las solicitudes y el tiempo de respuesta necesario. Cuando los clientes tienen la posibilidad de reenviar sus solicitudes, es práctico aplicar la limitación. Por el contrario, para los clientes que carecen de funcionalidades de reintento, lo ideal es implementar una solución de búfer. Estos búferes agilizan la afluencia de solicitudes y optimizan la interacción de las aplicaciones con diferentes velocidades operativas.




Curva de demanda con dos picos distintos que requieren una elevada capacidad aprovisionada

Supongamos que tenemos una carga de trabajo con la curva de demanda que se muestra en la imagen anterior. Esta carga de trabajo tiene dos picos y, para gestionarlos, se aprovisiona la capacidad de recursos que muestra la línea naranja. Los recursos y la energía utilizados para esta carga de trabajo no están indicados en el área situada debajo de la curva de demanda, sino en el área situada debajo de la línea de capacidad aprovisionada, ya que esta capacidad es la que se necesita para gestionar esos dos picos. El aplanamiento de la curva de demanda de la carga de trabajo puede ayudarle a reducir la capacidad aprovisionada para una carga de trabajo y a reducir su

impacto medioambiental. Para suavizar el pico, considere la posibilidad de implementar una solución de limitación o almacenamiento en búfer.

Vamos a profundizar en las limitaciones y el almacenamiento en búfer para entenderlos mejor.

Limitación: si la fuente de la demanda tiene capacidad de reintento, puede implementar la limitación. La limitación le dice al origen que, si no puede atender la solicitud en ese momento, debe intentarlo más tarde. El origen espera un tiempo y vuelve a intentar la solicitud. Implementar una limitación tiene la ventaja de que se limita la cantidad máxima de recursos y costos de la carga de trabajo. En AWS, puede utilizar Amazon API Gateway para implementar la limitación.

Basado en búfer: un enfoque basado en búfer utiliza productores (componentes que envían mensajes a la cola), consumidores (componentes que reciben los mensajes de la cola) y una cola (que contiene los mensajes) para almacenar los mensajes. De este modo, los consumidores pueden leer y procesar los mensajes, lo que permite que dichos mensajes se ejecuten a la velocidad que cumpla con los requisitos empresariales de los consumidores. Al utilizar una metodología centrada en los búferes, los mensajes de los productores se alojan en colas o secuencias, listos para que los consumidores accedan a ellos a un ritmo que se ajuste a sus demandas operativas.

En AWS, puede elegir entre varios servicios para implementar un enfoque basado en almacenamiento en búfer. Amazon Simple Queue Service (Amazon SQS) es un servicio administrado que incorpora colas que permiten a un solo consumidor leer mensajes individuales. Amazon Kinesis ofrece una secuencia que permite que muchos consumidores lean los mismos mensajes.

El almacenamiento en búfer y las limitaciones pueden suavizar cualquier pico al modificar la demanda de la carga de trabajo. Utilice limitaciones cuando los clientes vuelvan a intentar efectuar acciones y utilice el almacenamiento en búfer para retener la solicitud y procesarla más adelante. Al trabajar con un enfoque basado en búfer, diseñe su carga de trabajo para atender la solicitud en el tiempo requerido y verifique que pueda gestionar las solicitudes duplicadas. Analice la demanda general, la tasa de cambio y el tiempo de respuesta requerido para dimensionar correctamente la limitación o el búfer requeridos.

Pasos para la implementación

- Analice los requisitos del cliente: analice las solicitudes del cliente para determinar si puede llevar a cabo reintentos. Para los clientes que no puedan llevarlos a cabo, deberán implementarse búferes. Analice la demanda general, el ritmo de cambio y el tiempo de respuesta requerido para determinar el tamaño de la limitación o del búfer requeridos.

- Implemente un búfer o una limitación: implemente un búfer o una limitación en la carga de trabajo. Una cola, como Amazon Simple Queue Service (Amazon SQS), puede proporcionar un búfer a los componentes de la carga de trabajo. Amazon API Gateway puede proporcionar limitación a los componentes de la carga de trabajo.

Recursos

Prácticas recomendadas relacionadas:

- SUS02-BP06 Implementación del almacenamiento en búfer o la limitación para aplanar la curva de demanda
- REL05-BP02 Limitación de las solicitudes

Documentos relacionados:

- AWS Auto Scaling
- AWS Instance Scheduler
- Amazon API Gateway
- Amazon Simple Queue Service
- Getting started with Amazon SQS
- Amazon Kinesis

Videos relacionados:

- Choosing the Right Messaging Service for Your Distributed App

Ejemplos relacionados:

- Managing and monitoring API throttling in your workloads
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway
- Application integration Using Queues and Messages

COST09-BP03 Suministro dinámico de recursos

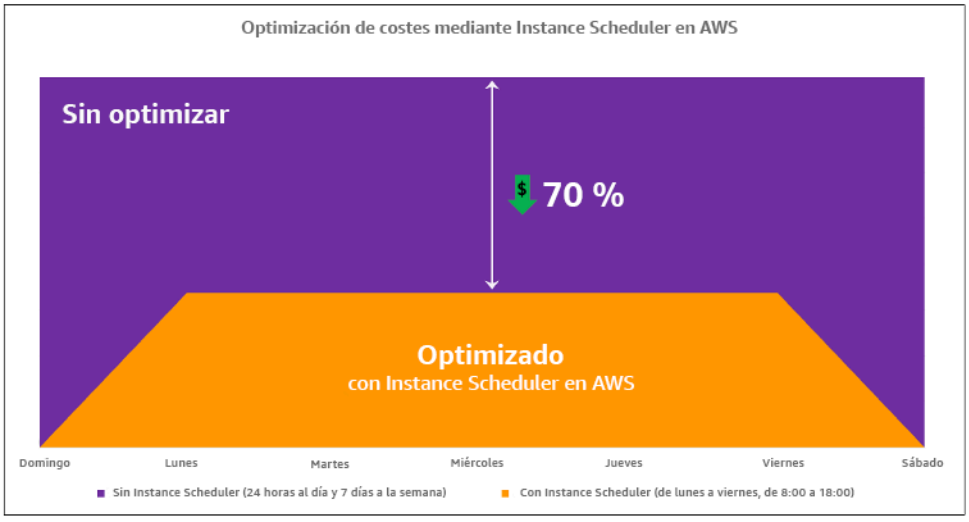
Los recursos se aprovisionan de manera planificada. Esto puede basarse en la demanda (por ejemplo, mediante el escalado automático) o en el tiempo, donde la demanda es predecible y los recursos se proporcionan en función del tiempo. Estos métodos conllevan la menor cantidad de aprovisionamiento excesivo o insuficiente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Hay varias formas en que los clientes de AWS pueden aumentar los recursos disponibles para sus aplicaciones y suministrar recursos para satisfacer la demanda. Una de estas opciones consiste en utilizar AWS Instance Scheduler, que automatiza el inicio y la detención de instancias de Amazon Elastic Compute Cloud (Amazon EC2) y de Amazon Relational Database Service (Amazon RDS). La otra opción es usar AWS Auto Scaling, que le permite escalar automáticamente los recursos de computación en función de la demanda de la aplicación o servicio. Suministrar recursos en función de la demanda le permitirá pagar únicamente por los recursos que utilice y reducir los costos, ya que solo lanza los recursos cuando se necesitan y los cancela cuando no.

AWS Instance Scheduler le permite configurar la detención y el inicio de sus instancias de Amazon EC2 y Amazon RDS en momentos definidos para que pueda satisfacer la demanda de los mismos recursos según un patrón temporal coherente; por ejemplo, que todos los días los usuarios accedan a las ocho de la mañana a instancias de Amazon EC2 que no se necesitan después de las seis de la tarde. Esta solución contribuye a reducir los costos operativos, ya que se detienen los recursos que no están en uso y se ponen en marcha cuando se necesitan.




Optimización de costos con AWS Instance Scheduler.

También puede configurar fácilmente los horarios de sus instancias de Amazon EC2 en todas sus cuentas y regiones con una interfaz de usuario (IU) sencilla mediante la configuración rápida de AWS Systems Manager. Puede programar instancias de Amazon EC2 o Amazon RDS con AWS Instance Scheduler y detener e iniciar las instancias existentes. Sin embargo, no puede detener ni iniciar instancias que formen parte de su grupo de escalado automático (ASG) ni que administren servicios como Amazon Redshift o Amazon OpenSearch Service. Los grupos de escalado automático tienen su propia programación para las instancias del grupo y estas instancias se crean.

AWS Auto Scaling lo ayuda a ajustar la capacidad para mantener un rendimiento predecible y estable al menor costo posible para satisfacer los cambios en la demanda. Se trata de un servicio gratuito y totalmente administrado para escalar la capacidad de su aplicación que se integra con las instancias de Amazon EC2 y flotas de spot, Amazon ECS, Amazon DynamoDB y Amazon Aurora. El escalado automático detecta los recursos automáticamente, lo que lo ayuda a buscar recursos en su carga de trabajo que se pueden configurar. Además, tiene estrategias de escalado integradas para optimizar el rendimiento y los costos, o un equilibrio entre ambos, y proporciona escalado predictivo para ayudar en los picos que se producen periódicamente.

Hay varias opciones de escalado disponibles para escalar su grupo de escalado automático:

- Mantener los niveles de instancia actuales en todo momento
- Escalar manualmente
- Escalado según una programación
- Escalado basado en la demanda
- Usar el escalado predictivo

Existen diferentes políticas de escalado automático. Se pueden clasificar en políticas de escalado dinámicas y programadas. Las políticas dinámicas son escalados manuales o dinámicos, que pueden ser programados o predictivos. Puede utilizar políticas de escalado para un escalado dinámico, programado y predictivo. También puede usar métricas y alarmas de Amazon CloudWatch para activar eventos de escalado de su carga de trabajo. Le recomendamos que utilice plantillas de lanzamiento que le permiten acceder a las últimas características y mejoras. No todas las funciones de escalado automático están disponibles cuando se utilizan configuraciones de lanzamiento. Por ejemplo, no puede crear un grupo de Auto Scaling que lance instancias de spot y bajo demanda o que especifique varios tipos de instancia. Debe utilizar una plantilla de lanzamiento para configurar estas características. Cuando utilice plantillas de lanzamiento, le recomendamos que haga un control de versiones en cada una de ellas. Con el control de versiones de plantillas de lanzamiento, puede

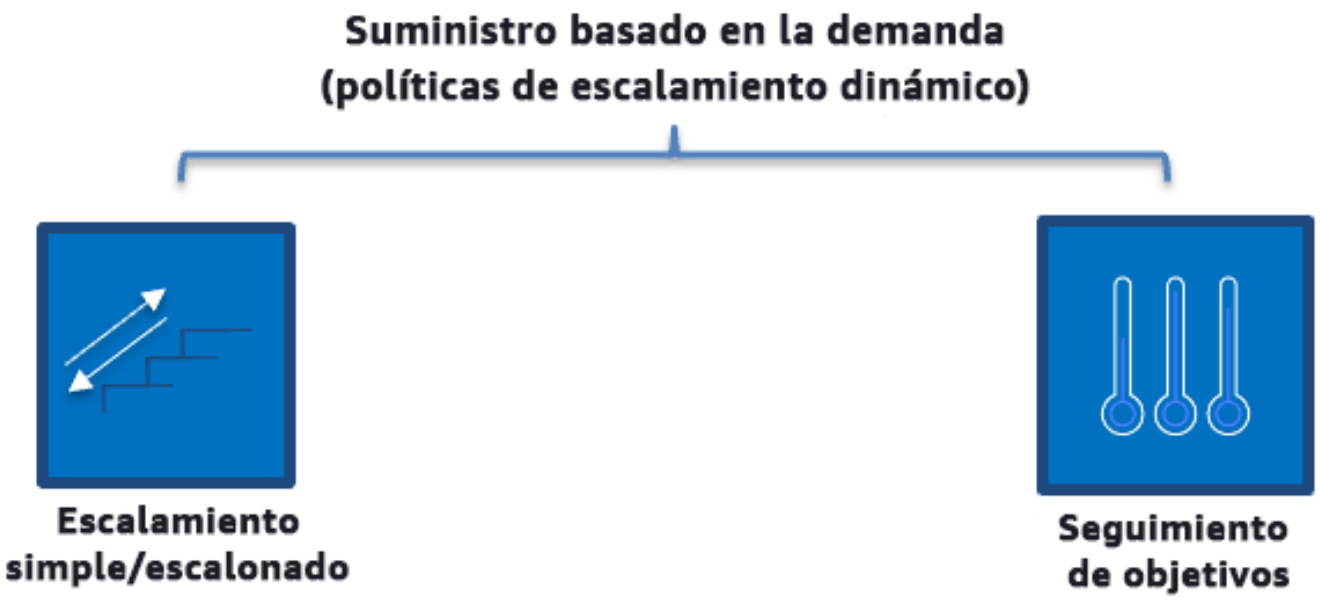
crear un subconjunto del conjunto completo de parámetros. A continuación, puede reutilizarlo para crear otras versiones de la misma plantilla de lanzamiento.

Puede usar AWS Auto Scaling o incorporar el escalado en su código con las API o SDK de AWS. Esto reduce los costos generales de la carga de trabajo al eliminar el costo operativo de aplicar los cambios manualmente en su entorno. Además, los cambios se pueden aplicar mucho más rápido. De este modo, también se adapta la dotación de recursos de la carga de trabajo a su demanda en cualquier momento. Para seguir esta práctica recomendada y suministrar recursos de forma dinámica a su organización, debe comprender el escalado horizontal y vertical en la Nube de AWS, así como la naturaleza de las aplicaciones que se ejecutan en las instancias de Amazon EC2. Es mejor que su equipo de administración financiera en la nube colabore con los equipos técnicos para seguir esta práctica recomendada.

Elastic Load Balancing (Elastic Load Balancing) lo ayuda a escalar mediante la distribución de la demanda entre varios recursos. Mediante ASG y Elastic Load Balancing, puede administrar las solicitudes entrantes mediante el enrutamiento óptimo del tráfico para que ninguna instancia se sobrecargue en un grupo de escalado automático. Las solicitudes se distribuirían entre todos los destinatarios de un grupo objetivo por turnos, sin tener en cuenta la capacidad ni la utilización.

Las métricas habituales pueden ser métricas de Amazon EC2 estándar, como el uso de la CPU, el rendimiento de la red y la latencia de solicitud y respuesta observada de Elastic Load Balancing. Si es posible, debe usar una métrica indicativa de la experiencia del cliente. Suele ser una métrica personalizada que se puede originar en el código de la aplicación en la carga de trabajo. Para explicar cómo satisfacer la demanda de forma dinámica, vamos a agrupar el escalado automático en dos categorías (modelos de suministro basados en la demanda y modelos de suministro basados en el tiempo) y analizaremos en profundidad cada una de ellas.

Suministro basado en la demanda: aproveche la elasticidad de la nube para suministrar recursos que satisfagan los cambios en la demanda utilizando el estado de la demanda casi en tiempo real. Para el suministro basado en la demanda, utilice las API o las características del servicio para cambiar mediante programación la cantidad de recursos en la nube de su arquitectura. De esta forma, puede escalar componentes en su arquitectura y aumentar la cantidad de recursos durante los picos de demanda para mantener el rendimiento, así como disminuir la capacidad cuando la demanda disminuya para reducir los costos.

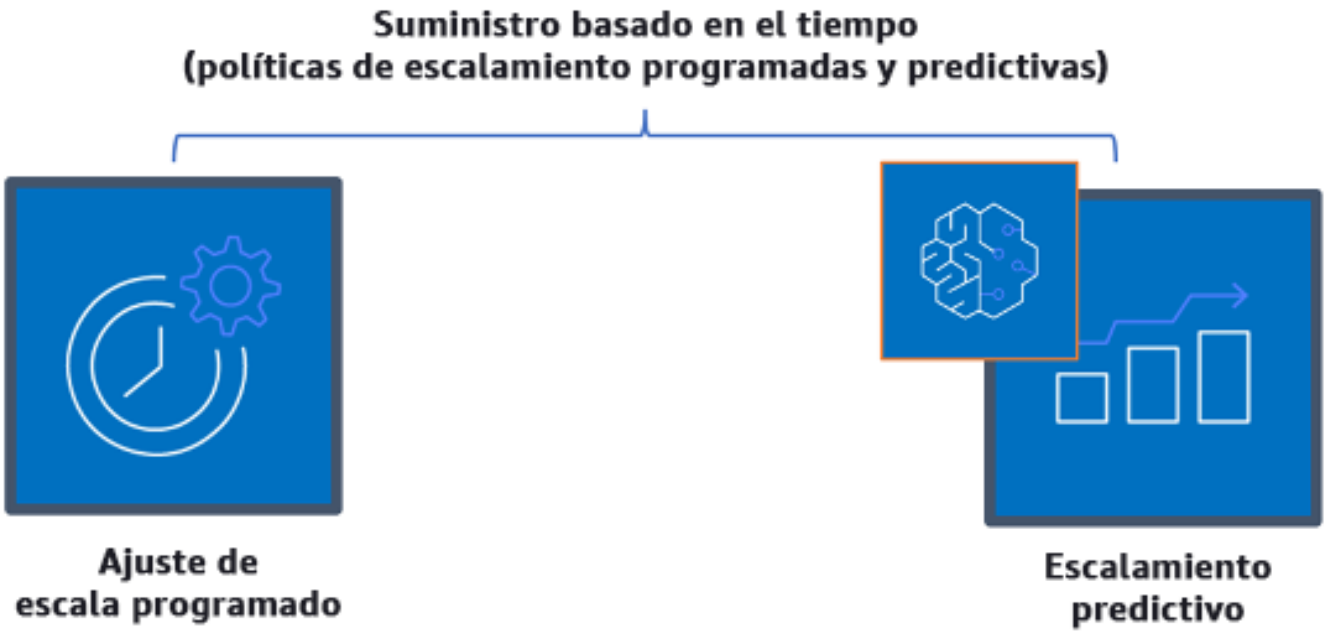



Políticas de escalado dinámico basadas en la demanda

- Escalado sencillo o por pasos: supervisa las métricas y agrega o elimina instancias de acuerdo con los pasos definidos manualmente por los clientes.
- Seguimiento de objetivos: mecanismo de control similar a un termostato que agrega o elimina instancias automáticamente para mantener las métricas en un objetivo definido por el cliente.

Al diseñar con un enfoque basado en la demanda, tenga en cuenta dos consideraciones clave. La primera: debe conocer la rapidez con la que necesita aprovisionar recursos nuevos. La segunda: tenga en cuenta que el tamaño del margen entre la oferta y la demanda cambiará. Debe estar preparado para poder hacer frente a la velocidad del cambio en la demanda y también a los errores de recursos.

Suministro basado en el tiempo: el enfoque basado en el tiempo adapta la capacidad de los recursos a una demanda que es predecible o que está bien definida por el tiempo. Normalmente, este enfoque no depende de los niveles de utilización de los recursos. El enfoque basado en el tiempo garantiza que los recursos estén disponibles en el momento específico en que se necesiten y que se puedan proporcionar sin retrasos debidos a los procedimientos de inicio y comprobaciones del sistema o de coherencia. Con el enfoque basado en el tiempo, puede brindar recursos adicionales o aumentar la capacidad durante los periodos de mayor actividad.




Políticas de escalado basadas en el tiempo

Puede utilizar el escalado automático programado o predictivo para implementar un enfoque basado en el tiempo. Las cargas de trabajo se pueden programar para escalarse o reducirse horizontalmente en momentos definidos (como el inicio del horario laboral). De este modo, los recursos están disponibles cuando lleguen los usuarios o aumente la demanda. El escalado predictivo utiliza patrones para escalar horizontalmente, mientras que el escalado programado utiliza tiempos predefinidos para escalar horizontalmente. También puede utilizar una estrategia de selección de tipo de instancia basada en atributos (ABS) en los grupos de escalado automático (ASG), lo que le permite expresar sus requisitos de instancia como un conjunto de atributos, por ejemplo, vCPU, memoria y almacenamiento. De este modo, también puede utilizar automáticamente los tipos de instancia de nueva generación cuando se lancen y acceder a una gama más amplia de capacidad con las instancias de spot de Amazon EC2. Flota de Amazon EC2 y Amazon EC2 Auto Scaling seleccionan y lanzan instancias que se ajusten a los atributos especificados, por lo que no es necesario elegir manualmente los tipos de instancia.

También puede utilizar las API y los SDK de AWS y AWS CloudFormation para aprovisionar y retirar entornos completos de manera automática según los necesite. Este enfoque es ideal para los entornos de desarrollo o pruebas que se ejecutan únicamente en horarios laborales o periodos definidos. Puede usar las API para escalar el tamaño de los recursos dentro de un entorno (escalado

vertical). Por ejemplo, puede escalar verticalmente una carga de trabajo de producción mediante el cambio del tamaño o la clase de instancia. Para ello, hay que detener o iniciar la instancia y seleccionar el tamaño o la clase de instancia diferente. Esta técnica también se puede aplicar a otros recursos tales como los volúmenes elásticos de Amazon EBS, los cuales se pueden modificar para aumentar el tamaño, ajustar el rendimiento (IOPS) o cambiar el tipo de volumen mientras están en uso.

Al diseñar con un enfoque basado en el tiempo, tenga en cuenta dos consideraciones clave. La primera: ¿qué grado de consistencia presenta el patrón? La segunda: ¿en qué afectaría el patrón si cambiara? Puede aumentar la precisión de las predicciones mediante la supervisión de sus cargas de trabajo y el uso de la inteligencia empresarial. Si observa cambios considerables en el patrón de uso, puede ajustar los tiempos para asegurarse de que se proporcione cobertura.

Pasos para la implementación

- Configuración del escalado programado: en caso de cambios predecibles en la demanda, el escalado basado en el tiempo puede proporcionar el número correcto de recursos de manera oportuna. También es útil si la creación y configuración de recursos no son suficientemente rápidas a la hora de responder a los cambios en la demanda. Use el análisis de las cargas de trabajo para configurar el escalado programado con AWS Auto Scaling. Para configurar la programación en función del tiempo, puede utilizar el escalado predictivo del escalado programado para aumentar por adelantado el número de instancias de Amazon EC2 de sus grupos de escalado automático en función de los cambios de carga previstos o predecibles.
- Configuración del escalado predictivo: el escalado predictivo le permite aumentar el número de instancias de Amazon EC2 en su grupo de escalado automático antes de los patrones diarios y semanales de flujos de tráfico. Si tiene picos de tráfico regulares y aplicaciones que tardan mucho en iniciarse, debería plantearse el uso del escalado predictivo. El escalado predictivo puede ayudarlo a escalar más rápidamente mediante la inicialización de la capacidad antes de la carga prevista si se compara con el escalado dinámico únicamente, que es de naturaleza reactiva. Por ejemplo, si los usuarios empiezan a utilizar su carga de trabajo con el inicio del horario laboral y no la utilizan fuera de dicho horario, el escalado predictivo puede agregar capacidad antes del horario laboral, lo que elimina el retraso del escalado dinámico para reaccionar ante los cambios en el tráfico.
- Configuración del escalado automático dinámico: use el escalado automático para configurar el escalado según las métricas de carga de trabajo activas. Use los análisis y configure el escalado automático para que se lance en los niveles de recursos correctos y verifique que la carga de trabajo se escala en el tiempo requerido. Puede lanzar y escalar automáticamente una flota de

instancias en diferido e instancias de spot en un solo grupo de Auto Scaling. Además de los descuentos relacionados con las instancias de spot, puede utilizar instancias reservadas o un Savings Plan para conseguir mejores precios de los habituales en las instancias en diferido. La combinación de todos estos factores le permite optimizar el ahorro de costos en las instancias de Amazon EC2, a la vez que se asegura de obtener la escala y el rendimiento deseados para su aplicación.

Recursos

Documentos relacionados:

- AWS Auto Scaling
- AWS Instance Scheduler
- Escalar el tamaño de un grupo de escalado automático
- Getting Started with Amazon EC2 Auto Scaling
- Getting started with Amazon SQS
- Scheduled Scaling for Amazon EC2 Auto Scaling
- Predictive scaling for Amazon EC2 Auto Scaling

Videos relacionados:

- Target Tracking Scaling Policies for Auto Scaling
- AWS Instance Scheduler

Ejemplos relacionados:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive Scaling with Amazon EC2 Auto Scaling
- ¿Cómo utilizo Instance Scheduler con CloudFormation para programar instancias de Amazon EC2?

## Optimización a lo largo del tiempo

Preguntas

- COST 10. ¿Cómo evalúa los servicios nuevos?
- COST 11. ¿Cómo evalúa el costo del esfuerzo?


## COST 10. ¿Cómo evalúa los servicios nuevos?

A medida que AWS presenta nuevos servicios y características, se recomienda que revise sus decisiones de diseño actuales para comprobar que sigan siendo las más rentables.

Prácticas recomendadas


- COST10-BP01 Desarrollo de un proceso de revisión de la carga de trabajo
- COST10-BP02 Revisión y análisis regulares de esta carga de trabajo


### COST10-BP01 Desarrollo de un proceso de revisión de la carga de trabajo

Desarrolle un proceso que defina los criterios y el proceso para la revisión de las cargas de trabajo. El esfuerzo de revisión debe reflejar la ventaja potencial. Por ejemplo, las cargas de trabajo principales o las cargas de trabajo con un valor por encima del diez por ciento de la factura se revisan trimestral o semestralmente, mientras que las cargas de trabajo por debajo del diez por ciento se revisan anualmente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: alto

Guía para la implementación

Para tener siempre la carga de trabajo más rentable, debe revisar periódicamente la carga de trabajo para saber si hay oportunidades de implementar nuevos servicios, características y componentes. Para conseguir reducir los costos totales, el proceso debe ser proporcional al volumen potencial de ahorro. Por ejemplo, las cargas de trabajo que suponen el 50 % de sus gastos totales se deben revisar con mayor regularidad y más a fondo que las cargas de trabajo que constituyen el cinco por ciento de sus gastos totales. Tenga en cuenta los factores externos o la volatilidad. Si la carga de trabajo da servicio a un segmento geográfico o de mercado específico y se prevén cambios en ese ámbito, unas revisiones más frecuentes podrían suponer un ahorro de costos. Otro factor de revisión es el esfuerzo para implementar los cambios. Si las pruebas y la validación de los cambios suponen un costo importante, las revisiones deberían ser menos frecuentes.

Hay que tener en cuenta el costo a largo plazo del mantenimiento de componentes y recursos obsoletos y heredados, y la imposibilidad de implementar nuevas características en ellos. El costo actual de las pruebas y la validación puede superar el beneficio propuesto. Sin embargo, con el

tiempo, el costo de aplicar el cambio puede aumentar significativamente a medida que se incrementa la brecha entre la carga de trabajo y las tecnologías actuales, lo que se traduce en costo aún mayores. Por ejemplo, el costo de pasar a un nuevo lenguaje de programación puede no ser rentable en la actualidad. No obstante, dentro de cinco años, el costo de las personas con competencias en ese lenguaje puede aumentar y, debido al crecimiento de la carga de trabajo, estaría trasladando un sistema aún mayor al nuevo lenguaje, lo que requeriría un esfuerzo aún mayor que el anterior.

Divida la carga de trabajo en componentes, asigne el costo del componente (basta con una estimación) y, a continuación, enumere los factores (por ejemplo, el esfuerzo y los mercados externos) junto a cada componente. Utilice estos indicadores para determinar la frecuencia de revisión de cada carga de trabajo. Por ejemplo, es posible que los servidores web tengan un costo elevado, un esfuerzo de cambio bajo y unos factores externos elevados, lo que da lugar a una frecuencia de revisión alta. Una base de datos central puede tener un costo medio, un esfuerzo de cambio alto y unos factores externos bajos, lo que da lugar a una frecuencia de revisión media.

Defina un proceso para evaluar nuevos servicios, patrones de diseño, tipos de recursos y configuraciones para optimizar el costo de la carga de trabajo a medida que estén disponibles. Al igual que en los procesos de revisión del pilar de rendimiento y revisión del pilar de fiabilidad, identifique, valide y priorice las actividades de optimización y mejora y la solución de problemas e incorpórelas a su cartera de trabajo.

Pasos para la implementación

- Defina la frecuencia de revisión: defina con qué frecuencia se deben revisar la carga de trabajo y sus componentes. Asigne tiempo y recursos a la mejora continua y revise la frecuencia para mejorar la eficacia y la optimización de su carga de trabajo. Se trata de una combinación de factores y puede diferir de una carga de trabajo a otra en su organización y entre los componentes de la carga de trabajo. Entre los factores más comunes se encuentran la importancia para la organización medida en cuanto a los ingresos o la marca, el costo total de la ejecución de la carga de trabajo (incluidos los costos de funcionamiento y de recursos), la complejidad de la carga de trabajo, la facilidad para implementar un cambio, cualquier acuerdo de licencia de software y si un cambio supusiera un aumento significativo de los costos de licencia debido a las licencias punitivas. Los componentes pueden definirse funcional o técnicamente, como servidores web y bases de datos, o recursos de computación y almacenamiento. Equilibre los factores de la forma correspondiente y desarrolle un periodo para la carga de trabajo y sus componentes. Puede decidir revisar toda la carga de trabajo cada 18 meses, revisar los servidores web cada seis meses, la base de datos cada 12 meses, la computación y el almacenamiento a corto plazo cada seis meses y el almacenamiento a largo plazo cada 12 meses.

- Definición de la exhaustividad de la revisión: defina cuánto esfuerzo se dedica a la revisión de la carga de trabajo o de los componentes de la carga de trabajo. Al igual que sucede con la frecuencia de revisión, se trata de equilibrar múltiples factores. Evalúe y priorice las oportunidades de mejora para centrar los esfuerzos donde aporten los mayores beneficios, al tiempo que estima el esfuerzo necesario para estas actividades. Si los resultados previstos no alcanzan los objetivos y el esfuerzo necesario cuesta más, repita el proceso con acciones alternativas. Sus procesos de revisión deben incluir tiempo y recursos de sus procesos para hacer posibles las mejoras incrementales continuas. Por ejemplo, puede decidir dedicar una semana de análisis al componente de base de datos, una semana de análisis a los recursos de computación y cuatro horas a las revisiones de almacenamiento.

Recursos

Documentos relacionados:

- Blog de noticias de AWS
- Tipos de computación en la nube
- Novedades de AWS

Ejemplos relacionados:

- AWS Support Proactive Services
- Revisiones regulares de las cargas de trabajo de SAP

COST10-BP02 Revisión y análisis regulares de esta carga de trabajo

Las cargas de trabajo existentes se revisan periódicamente en función de cada proceso definido para averiguar si se pueden adoptar nuevos servicios, reemplazar los existentes o rediseñar las cargas de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

AWS agrega constantemente nuevas características para que pueda experimentar e innovar más rápidamente con la tecnología más reciente. En Novedades de AWS se detalla cómo le está yendo a AWS y se proporciona una descripción general rápida de los servicios, las características

y los anuncios de expansión regional de AWS a medida que se lanzan. Puede profundizar en los lanzamientos que se han anunciado y utilizarlos para revisar y analizar sus cargas de trabajo existentes. Para obtener las ventajas de los nuevos servicios y características de AWS, debe revisar sus cargas de trabajo e implementar los nuevos servicios y características según sea necesario. Esto significa que es posible que tenga que reemplazar los servicios existentes que utiliza para la carga de trabajo o modernizarla para adoptar estos nuevos servicios de AWS. Por ejemplo, podría revisar sus cargas de trabajo y reemplazar el componente de mensajería por Amazon Simple Email Service. Esto elimina el costo de utilizar y mantener una flota de instancias, a la vez que proporciona toda la funcionalidad a un costo reducido.

Para analizar su carga de trabajo y destacar las posibles oportunidades, debe tener en cuenta no solo nuevos servicios, sino también nuevas formas de crear soluciones. Consulte los videos This is My Architecture en AWS para obtener información sobre los diseños arquitectónicos de otros clientes, sus desafíos y sus soluciones. Consulte la serie All-In para conocer las aplicaciones de los servicios de AWS en el mundo real y las historias de los clientes. También puede ver la serie de videos Back to Basics, en la que se explican, examinan y desglosan las prácticas recomendadas básicas sobre los patrones de arquitectura de nube. Otra fuente son los videos How to Build This, que están diseñados para ayudar a las personas con grandes ideas sobre cómo hacer realidad su producto mínimo viable (MVP) mediante los servicios de AWS. Es una forma de que los creadores de todo el mundo que tengan una idea sólida reciban orientación sobre arquitectura de arquitectos de soluciones de AWS experimentados. Por último, puede revisar los materiales de recursos de Introducción, que incluyen tutoriales paso a paso.

Antes de empezar el proceso de revisión, cumpla los requisitos de su empresa en cuanto a carga de trabajo, seguridad y privacidad de los datos para poder utilizar un servicio específico o los requisitos de la región y rendimiento mientras sigue el proceso de revisión acordado.

Pasos para la implementación

- Revisión periódica de la carga de trabajo: lleve a cabo las revisiones con la frecuencia especificada aplicando el proceso que haya definido. Compruebe que dedica el esfuerzo adecuado a cada componente. Este proceso sería similar al del diseño inicial, en el que seleccionó los servicios para la optimización de costos. Analice los servicios y las ventajas que aportarían. Esta vez, tenga en cuenta el costo de aplicar el cambio, no solo las ventajas a largo plazo.
- Implementación de nuevos servicios: si la conclusión del análisis es implementar cambios, establezca primero una base de referencia de la carga de trabajo para conocer el costo actual de cada resultado. Implemente los cambios y, a continuación, haga un análisis para confirmar el nuevo costo de cada resultado.

Recursos

Documentos relacionados:

- Blog de noticias de AWS
- Novedades de AWS
- Documentación de AWS
- Introducción a AWS
- Recursos generales de AWS

Videos relacionados:

- AWS - This is My Architecture
- AWS - Back to Basics
- AWS - Serie All-In
- How to Build This

## COST 11. ¿Cómo evalúa el costo del esfuerzo?

Prácticas recomendadas

- COST11-BP01 Automatización de las operaciones

### COST11-BP01 Automatización de las operaciones

Evalúe los costos operativos en la nube, centrándose en cuantificar el ahorro de tiempo y esfuerzo en tareas administrativas, implementaciones, mitigación del riesgo de errores humanos, cumplimiento y otras operaciones mediante la automatización. Evalúe el tiempo y los costos asociados que son necesarios para los esfuerzos operativos e implemente la automatización de las tareas administrativas para minimizar el esfuerzo manual siempre que sea factible.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

La automatización de las operaciones reduce la frecuencia de las tareas manuales, mejora la eficacia y beneficia a los clientes, ya que ofrece una experiencia coherente y fiable a la hora de implementar,

administrar u operar cargas de trabajo. Puede liberar recursos de la infraestructura de las tareas operativas manuales y utilizarlos para tareas de mayor valor e innovaciones, lo que mejora el valor empresarial. Las empresas necesitan una forma probada y contrastada de administrar sus cargas de trabajo en la nube. Esta solución debe ser segura, rápida y rentable, con un riesgo mínimo y máxima fiabilidad.

Comience por priorizar sus actividades operativas en función del esfuerzo necesario examinando el costo global de las operaciones. Por ejemplo, ¿cuánto tiempo se tarda en implementar nuevos recursos en la nube, aplicar cambios de optimización en los existentes o implementar las configuraciones necesarias? Analice el costo total de las acciones humanas teniendo en cuenta el costo de las operaciones y de la administración. Dé prioridad a las automatizaciones de las tareas administrativas para reducir el esfuerzo manual.

El esfuerzo de revisión debe reflejar la ventaja potencial. Por ejemplo, fíjese en el tiempo que se dedica a llevar a cabo tareas manuales comparado con el que se tarda en efectuarlas de forma automática. Priorice la automatización de actividades repetitivas, de alto valor, lentas y complejas. Las actividades de alto valor o que entrañan un mayor riesgo de errores humanos suelen ser las mejores con las que empezar la automatización, ya que el riesgo suele plantear un costo operativo adicional no deseado (por ejemplo, que el equipo de operaciones trabaje horas extra).

Utilice herramientas de automatización como AWS Systems Manager o AWS Config para optimizar las operaciones, el cumplimiento, la supervisión, el ciclo de vida y los procesos de terminación. Con servicios de AWS, herramientas y productos de terceros, puede personalizar las automatizaciones que implemente para satisfacer sus requisitos específicos. En la tabla siguiente, se muestran algunas de las funciones y capacidades de funcionamiento básicas que puede conseguir con los servicios de AWS para automatizar la administración y el funcionamiento:

- AWS Audit Manager: audite continuamente su uso de AWS para simplificar la evaluación de riesgos y cumplimiento.
- AWS Backup: administre y automatice la protección de datos de forma centralizada.
- AWS Config: configure los recursos de computación, evalúe y audite las configuraciones y el inventario de recursos.
- AWS CloudFormation: lance recursos de alta disponibilidad con infraestructura como código.
- AWS CloudTrail: administración, cumplimiento y control de los cambios de TI.
- Amazon EventBridge: programe eventos y active AWS Lambda para que actúen.
- AWS Lambda: automatice los procesos repetitivos activándolos con eventos o ejecutándolos según una programación fija con AWS EventBridge.

- AWS Systems Manager: inicie y detenga las cargas de trabajo, aplique revisiones a los sistemas operativos y automatice la configuración y la administración continua.
- AWS Step Functions: programe trabajos y automatice los flujos de trabajo.
- AWS Service Catalog: consumo de plantillas, infraestructura como código con cumplimiento y control.

Si desea adoptar las automatizaciones de forma inmediata con el uso de productos y servicios de AWS y no tiene destrezas en la organización, contacte con AWS Managed Services (AMS), AWS Professional Services o los socios de AWS para aumentar la adopción de la automatización y mejorar su excelencia operativa en la nube.

AWS Managed Services (AMS) es un servicio que utiliza la infraestructura de AWS en nombre de los socios y clientes de la empresa. Proporciona un entorno seguro y conforme a las normativas en el que puede implementar sus cargas de trabajo. AMS utiliza modelos operativos de nube empresarial con automatización para permitirle satisfacer los requisitos de su organización, trasladarse a la nube más rápidamente y reducir los costos de administración continua.

AWS Professional Services también puede ayudarlo a conseguir los resultados empresariales deseados y a automatizar las operaciones con AWS. Ayudan a los clientes a implementar operaciones de TI automatizadas, sólidas y ágiles, así como capacidades de gobernanza optimizadas para la nube. Para ver ejemplos detallados de supervisión y las prácticas recomendadas, consulte el documento técnico sobre el pilar de excelencia operativa.

Pasos para la implementación

- Creación única e implementación múltiple: utilice infraestructura como código, como CloudFormation, el SDK de AWS o la AWS CLI, para implementar una vez y utilice la creación varias veces en entornos similares o en escenarios de recuperación de desastres. Etiquete mientras implementa para hacer un seguimiento de su consumo, tal y como se define en otras prácticas recomendadas. Utilice AWS Launch Wizard para reducir el tiempo de implementación de muchas cargas de trabajo empresariales populares. AWS Launch Wizard le guía en el proceso de dimensionamiento, configuración e implementación de cargas de trabajo empresariales siguiendo las prácticas recomendadas de AWS. También puede usar Service Catalog, que lo ayuda a crear y administrar plantillas aprobadas de infraestructura como código para su uso en AWS, de modo que cualquiera pueda descubrir recursos de nube aprobados y de autoservicio.
- Automatice el cumplimiento continuo: considere la posibilidad de automatizar la evaluación y la corrección de las configuraciones registradas en función de los estándares predefinidos. Si

combina AWS Organizations con las capacidades de AWS Config y AWS CloudFormation, puede administrar y automatizar de manera eficiente el cumplimiento de la configuración a escala para cientos de cuentas de miembro. Puede revisar los cambios en las configuraciones y las relaciones entre los recursos de AWS y profundizar en el historial de una configuración de recursos.

- Automatice las tareas de supervisión: AWS proporciona varias herramientas que puede utilizar para supervisar servicios. Puede configurar estas herramientas para automatizar las tareas de supervisión. Cree e implemente un plan de supervisión con el que se recopilen datos de supervisión de todas las partes de su carga de trabajo para que pueda depurar más fácilmente un error multipunto en el caso de que se produzca. Por ejemplo, puede usar las herramientas de supervisión automatizadas para observar a Amazon EC2 y que le informe cuando algo va mal en las comprobaciones del estado del sistema, las comprobaciones del estado de las instancias y las alarmas de Amazon CloudWatch.
- Automatice el mantenimiento y las operaciones: ejecute las operaciones de rutina automáticamente sin intervención humana. Usando los servicios y las herramientas de AWS, puede elegir qué automatizaciones de AWS implementar y personalizar según sus requisitos específicos. Por ejemplo, utilice el Generador de imágenes de EC2 para crear, probar e implementar imágenes de máquinas virtuales y contenedores para su uso en AWS, en las instalaciones o para aplicar revisiones a las instancias de EC2 con AWS SSM. Si la acción deseada no se puede ejecutar con los servicios de AWS o si necesita acciones más complejas con los recursos de filtrado, automatice sus operaciones mediante AWS Command Line Interface (AWS CLI) o las herramientas de AWS SDK. AWS CLI ofrece la posibilidad de automatizar todo el proceso de control y administración de los servicios de AWS mediante scripts sin utilizar la Consola de administración de AWS. Seleccione sus SDK de AWS preferidos para interactuar con los servicios de AWS. Para ver otros ejemplos de código, consulte el repositorio de ejemplos de código del SDK de AWS.
- Creación de un ciclo de vida continuo con las automatizaciones: es importante que establezca y conserve políticas de ciclo de vida consolidadas, no solo en lo que respecta a las normativas o la redundancia, sino también a fin de optimizar los costos. Puede usar AWS Backup para administrar y automatizar de manera centralizada la protección de datos de los almacenes de datos, como buckets, volúmenes, bases de datos y sistemas de archivos. Puede utilizar Amazon Data Lifecycle Manager para automatizar la creación, retención y eliminación de instantáneas de EBS y las AMI respaldadas por EBS.
- Eliminación de recursos innecesarios: es bastante común acumular recursos no utilizados de Cuentas de AWS en un entorno de pruebas o de desarrollo. Los desarrolladores crean y experimentan con diversos servicios y recursos como parte del ciclo de desarrollo normal y, después, no eliminan esos recursos cuando ya no los necesitan. Los recursos no utilizados

pueden dar lugar a costos innecesarios para la organización que a veces son elevados. La eliminación de estos recursos puede reducir los costos de operación de estos entornos. Asegúrese de que sus datos no son necesarios o haga una copia de seguridad si no está claro. Puede utilizar AWS CloudFormation para limpiar las pilas implementadas, con lo que se elimina automáticamente la mayoría de los recursos definidos en la plantilla. Como alternativa, puede crear una automatización para la eliminación de recursos de AWS con herramientas como aws-nuke.

Recursos

Documentos relacionados:

- Modernizing operations in the Nube de AWS
- AWS Services for Automation
- Infraestructura y automatización
- AWS Systems Manager Automation
- Supervisión automatizada y manual
- AWS automations for SAP administration and operations
- AWS Managed Services
- AWS Servicios profesionales de

Videos relacionados:

- Automate Continuous Compliance at Scale in AWS
- AWS Backup Demo: Cross-Account & Cross-Region Backup
- Patching for your Amazon EC2 Instances

Ejemplos relacionados:

- Reinventing automated operations (Part I)
- Reinventing automated operations (Part II)
- Automate deletion of AWS resources by using aws-nuke
- Delete unused Amazon EBS volumes by using AWS Config and AWS SSM
- Automate continuous compliance at scale in AWS

- IT Automations with AWS Lambda

# Sostenibilidad

El pilar de sostenibilidad incluye comprender las repercusiones de los servicios que se usan, cuantificar el impacto durante todo el ciclo de vida de la carga de trabajo y aplicar tanto principios de diseño como prácticas recomendadas para reducir estas repercusiones al diseñar cargas de trabajo en la nube. Encontrará una guía prescriptiva acerca de la implementación en el documento técnico sobre el pilar de sostenibilidad.

Áreas de prácticas recomendadas

- Selección de región
- Alineación con la demanda
- Software y arquitectura
- Datos
- Hardware y servicios
- Proceso y cultura

## Selección de región

Pregunta

- SUS 1: ¿Cómo selecciona las regiones para la carga de trabajo?

### SUS 1: ¿Cómo selecciona las regiones para la carga de trabajo?

La elección de la región para su carga de trabajo afecta significativamente a sus KPI, incluidos el rendimiento, el costo y la huella de carbono. Para mejorar eficazmente estos KPI, debe elegir las regiones para sus cargas de trabajo basándose tanto en los requisitos empresariales como en los objetivos de sostenibilidad.

Prácticas recomendadas

- SUS01-BP01 Selección de la región en función de los requisitos empresariales y los objetivos de sostenibilidad

SUS01-BP01 Selección de la región en función de los requisitos empresariales y los objetivos de sostenibilidad

Elija una región para su carga de trabajo en función tanto de los requisitos empresariales como de los objetivos de sostenibilidad para optimizar sus KPI, incluidos el rendimiento, el costo y la huella de carbono.

Patrones comunes de uso no recomendados:

- Seleccionar la región de la carga de trabajo en función de la propia ubicación.
- Consolida todos los recursos de la carga de trabajo en una ubicación geográfica.

Beneficios de establecer esta práctica recomendada: la colocación de una carga de trabajo cerca de proyectos de energías renovables de Amazon o de regiones con una baja intensidad de carbono publicada puede ayudar a reducir la huella de carbono de una carga de trabajo en la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La Nube de AWS es una red en constante expansión de regiones y puntos de presencia (POP), con una infraestructura de red global que los une. La elección de la región para su carga de trabajo afecta significativamente a sus KPI, incluidos el rendimiento, el costo y la huella de carbono. Para mejorar eficazmente estos KPI, debe elegir las regiones para su carga de trabajo en función tanto de los requisitos empresariales como de los objetivos de sostenibilidad.

Pasos para la implementación

- Preseleccione las regiones potenciales: siga estos pasos para evaluar y preseleccionar las posibles regiones para la carga de trabajo en función de los requisitos empresariales, incluido el cumplimiento, las características disponibles, el coste y la latencia:
  - Confirme que estas regiones cumplen con la normativa, en función de la legislación local aplicable (por ejemplo, sobre soberanía de datos).
  - Utilice las listas de servicios regionales de AWS para comprobar si las regiones cuentan con los servicios y las características que necesita para gestionar su carga de trabajo.
  - Calcule el costo de la carga de trabajo en cada región mediante Calculadora de precios de AWS.
  - Pruebe la latencia de la red entre las ubicaciones de sus usuarios finales y cada Región de AWS.

- Elija regiones: elija regiones cerca de proyectos de energías renovables de Amazon y regiones en las que la intensidad de carbono recogida en la cuadrícula sea más baja que en otras ubicaciones (o regiones).
  - Identifique sus directrices de sostenibilidad relevantes para rastrear y comparar las emisiones de carbono de un año a otro según el protocolo de gases de efecto invernadero (métodos basados en el mercado y basados la ubicación).
  - Elija la región en función del método que utilice para hacer un seguimiento de las emisiones de carbono. Para obtener más información sobre cómo elegir una región en función de tus directrices de sostenibilidad, consulte How to select a Region for your workload based on sustainability goals.

Recursos

Documentos relacionados:

- Understanding your carbon emission estimations
- Amazon Around the Globe
- Renewable Energy Methodology
- What to Consider when Selecting a Region for your Workloads

Videos relacionados:

- AWS re:Invent 2023 - Sustainability innovation in AWS Global Infrastructure
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure

## Alineación con la demanda

Pregunta

- SUS 2 ¿Cómo alinea los recursos en la nube a su demanda?

## SUS 2 ¿Cómo alinea los recursos en la nube a su demanda?

La forma en que los usuarios y las aplicaciones consumen las cargas de trabajo y otros recursos puede ayudarle a identificar las mejoras necesarias para alcanzar sus objetivos de sostenibilidad. Escale la infraestructura para adaptarla continuamente a la demanda y compruebe que solo utiliza los recursos mínimos necesarios para prestar asistencia a sus usuarios. Alinee los niveles de servicio con las necesidades de los clientes. Posicione los recursos de forma que se limite el uso de red necesario para que los usuarios puedan consumirlos. Elimine los activos que no se usan. Proporcione a los miembros de su equipo dispositivos que satisfagan sus necesidades con un impacto mínimo en la sostenibilidad.

Prácticas recomendadas

- SUS02-BP01 Escalado de la infraestructura de la carga de trabajo dinámicamente
- SUS02-BP02 Alineación de los SLA con los objetivos de sostenibilidad
- SUS02-BP03 Detención de la creación y el mantenimiento de los recursos no utilizados
- SUS02-BP04 Optimización de la ubicación geográfica de las cargas de trabajo en función de sus requisitos de red
- SUS02-BP05 Optimización de los recursos de los miembros del equipo para las actividades efectuadas
- SUS02-BP06 Implementación del almacenamiento en búfer o la limitación para aplanar la curva de demanda

### SUS02-BP01 Escalado de la infraestructura de la carga de trabajo dinámicamente

Utilice la elasticidad de la nube y escale su infraestructura de forma dinámica para adaptar la oferta de recursos en la nube a la demanda y evitar un exceso de capacidad en su carga de trabajo.

Patrones comunes de uso no recomendados:

- No escalar la infraestructura con la carga de usuarios.
- Escalar la infraestructura manualmente todo el tiempo.
- Dejar la capacidad aumentada después de un evento de ajuste de escala en lugar de volver a desescalar verticalmente.

Beneficios de establecer esta práctica recomendada: configurar y probar la elasticidad de la carga de trabajo ayuda a adaptar de manera eficiente el suministro de recursos de la nube a la demanda

y a evitar el exceso de aprovisionamiento de la capacidad. Puede aprovechar la elasticidad de la nube para escalar automáticamente la capacidad durante y después de los picos de demanda para asegurarse de que solo utiliza el número correcto de recursos necesarios para satisfacer los requisitos empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La nube ofrece la flexibilidad de ampliar o reducir sus recursos de forma dinámica a través de diversos mecanismos para satisfacer los cambios en la demanda. La correspondencia óptima entre la oferta y la demanda ofrece el menor impacto medioambiental para una carga de trabajo.

La demanda puede ser fija o variable, lo que requiere métricas y automatización para garantizar que la administración no resulte difícil. Las aplicaciones pueden escalarse o desescalarse verticalmente mediante la modificación del tamaño de la instancia, escalarse o desescalarse horizontalmente mediante la modificación del número de instancias, o una combinación de ambas.

Puede usar distintos enfoques para hacer que el suministro de recursos coincida con la demanda.

- Enfoque de seguimiento de objetivos: supervise la métrica de escalado y aumente o reduzca de forma automática la capacidad en función de sus necesidades.
- Escalado predictivo: reduzca horizontalmente de antemano según las tendencias diarias y semanales previstas.
- Enfoque basado en la programación: establezca su propia programación de escalado según los cambios de carga predecibles.
- Escalado de servicios: elija servicios (como los servicios sin servidor) que se escalen de forma nativa por diseño o que incluyan el escalado automático como característica.

Identifique los periodos de uso reducido o inexistente y escale los recursos en consonancia para eliminar el exceso de capacidad y mejorar la eficiencia.

Pasos para la implementación

- La elasticidad hace coincidir la oferta de los recursos que tiene con la demanda de esos recursos. Las instancias, los contenedores y las funciones proporcionan mecanismos de elasticidad, ya sea en combinación con el escalado automático o como características del servicio. AWS proporciona una serie de mecanismos de escalado automático para garantizar que las cargas de trabajo

puedan reducirse verticalmente de forma rápida y sencilla durante los periodos con poca carga de usuarios. A continuación, se presentan algunos ejemplos de mecanismos de escalado automático:

| Mecanismo de escalado automático | Dónde se usa |
| --- | --- |
| Amazon EC2 Auto Scaling | Se usa para verificar que tiene el número correcto de instancias de Amazon EC2 disponibles para gestionar la carga de usuarios de su aplicación. |
| Aplicación de escalado automático | Se usa para escalar automáticamente los recursos de servicios de AWS más allá de Amazon EC2, como las funciones de Lambda o los servicios de Amazon Elastic Container Service (Amazon ECS). |
| Escalador automático de clústeres de Kubernetes | Se usa para escalar automáticamente clústeres de Kubernetes en AWS. |

- Normalmente, se habla del escalado en relación con los servicios de computación, como las instancias de Amazon EC2 o las funciones de AWS Lambda. Considere la posibilidad de configurar servicios no computacionales, como las unidades de capacidad de lectura y escritura de Amazon DynamoDB o las particiones de Amazon Kinesis Data Streams, para satisfacer la demanda.
- Verifique que las métricas para escalar o reducir verticalmente se validan con respecto al tipo de carga de trabajo que se está implementando. Si está implementando una aplicación de transcodificación de vídeo, se espera un uso del 100 % de la CPU y no debería ser su métrica principal. Si es necesario, puede utilizar una métrica personalizada (como el uso de la memoria) para su política de escalado. Para elegir las métricas adecuadas, tenga en cuenta las siguientes directrices para Amazon EC2:
  - La métrica debe ser una métrica de utilización válida y describir el grado de ocupación de una instancia.
  - El valor de la métrica debe aumentar o disminuir proporcionalmente al número de instancias del grupo de escalado automático.
- Utilice el escalado dinámico en lugar del escalado manual para su grupo de escalado automático. También le recomendamos que utilice políticas de escalado de seguimiento objetivo en su escalado dinámico.

- Verifique que las implementaciones de la carga de trabajo puedan manejar los eventos de escalado y desescalado horizontales. Cree escenarios de prueba para los eventos de escalado con el fin de verificar que la carga de trabajo se comporta del modo previsto y no afecta a la experiencia del usuario (como la pérdida de sesiones persistentes). Puede utilizar el historial de actividad para verificar una actividad de escalado para un grupo de escalado automático.
- Evalúe los patrones predecibles de su carga de trabajo y escale de forma proactiva para anticiparse a los cambios previstos y planeados en la demanda. Con el escalado predictivo, puede eliminar la necesidad de aprovisionar capacidad en exceso. Para más información, consulte Predictive Scaling with Amazon EC2 Auto Scaling.

Recursos

Documentos relacionados:

- Getting Started with Amazon EC2 Auto Scaling
- Predictive Scaling for EC2, Powered by Machine Learning
- Analyze user behavior using Amazon OpenSearch Service, Amazon Data Firehose and Kibana
- ¿Qué es Amazon CloudWatch?
- Monitoreo de la carga de base de datos con Performance Insights en Amazon RDS
- Introducing Native Support for Predictive Scaling with Amazon EC2 Auto Scaling
- Introducing Karpenter - An Open-Source, High-Performance Kubernetes Cluster Autoscaler
- Deep Dive on Amazon ECS Cluster Auto Scaling

Videos relacionados:

- AWS re:Invent 2023 - Scaling on AWS for the first 10 million users
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment
- AWS re:Invent 2022 - Scaling containers from one user to millions
- AWS re:Invent 2.023 - Scaling FM inference to hundreds of models with Amazon SageMaker AI
- AWS re:Invent 2023 - Harness the power of Karpenter to scale, optimize & upgrade Kubernetes

Ejemplos relacionados:

- Autoscaling

SUS02-BP02 Alineación de los SLA con los objetivos de sostenibilidad

Revise y optimice los acuerdos de nivel de servicio (SLA) de la carga de trabajo en función de sus objetivos de sostenibilidad a fin de minimizar los recursos necesarios para admitir la carga de trabajo sin dejar de satisfacer las necesidades empresariales.

Patrones comunes de uso no recomendados:

- Los SLA de carga de trabajo se desconocen o son ambiguos.
- Define su SLA solo para la disponibilidad y el rendimiento.
- Utiliza el mismo patrón de diseño (como la arquitectura Multi-AZ) para todas sus cargas de trabajo.

Beneficios de establecer esta práctica recomendada: la alineación de los SLA con los objetivos de sostenibilidad conlleva un uso óptimo de los recursos, al tiempo que se satisfacen las necesidades empresariales.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Los SLA definen el nivel de servicio que se espera de una carga de trabajo en la nube, como el tiempo de respuesta, la disponibilidad y la retención de datos. Influyen en la arquitectura, el uso de recursos y el impacto medioambiental de una carga de trabajo en la nube. Con una cadencia regular, revise los SLA y haga concesiones para reducir significativamente el uso de recursos a cambio de disminuciones aceptables en los niveles de servicio.

Pasos para la implementación

- Comprensión de los objetivos de sostenibilidad: identifique los objetivos de sostenibilidad de su organización, como la reducción de emisiones de carbono o la mejora del uso de los recursos.
- Revisión de los SLA: evalúe sus SLA para determinar si cumplen con los requisitos de su empresa. Si está superando los SLA, lleve a cabo una revisión adicional.
- Comprensión de las compensaciones: comprenda cuáles son las compensaciones de la complejidad de su carga de trabajo (como el alto volumen de usuarios simultáneos), el rendimiento (como la latencia) y el impacto en la sostenibilidad (como los recursos necesarios). Por lo general, priorizar dos de los factores se produce a expensas del tercero.

- Ajuste de los SLA: ajuste los SLA para hacer que las compensaciones disminuyan de forma considerable las repercusiones en la sostenibilidad a cambio de reducciones aceptables en los niveles de servicio.
  - Sostenibilidad y fiabilidad: las cargas de trabajo de alta disponibilidad tienden a consumir más recursos.
  - Sostenibilidad y rendimiento: el uso de más recursos para aumentar el rendimiento podría tener un mayor impacto medioambiental.
  - Sostenibilidad y seguridad: las cargas de trabajo excesivamente seguras podrían tener un mayor impacto medioambiental.
- Definición de los SLA de sostenibilidad si es posible: incluya los SLA de sostenibilidad en su carga de trabajo. Por ejemplo, defina un nivel de uso mínimo como un SLA de sostenibilidad para sus instancias de computación.
- Uso de patrones de diseño eficaces: use patrones de diseño, como microservicios en AWS, que den prioridad a las funciones esenciales para el negocio y permitan unos niveles de servicio más bajos (como objetivos de tiempo de respuesta o de tiempo de recuperación) para las funciones no críticas.
- Comunicación y establecimiento de responsabilidades: comparta los SLA con todas las partes interesadas pertinentes, incluidos su equipo de desarrollo y los clientes. Utilice los informes para hacer un seguimiento de los SLA y supervisarlos. Asigne responsabilidades para cumplir con los objetivos de sostenibilidad de los SLA.
- Uso de incentivos y recompensas: utilice incentivos y recompensas para lograr o superar los SLA que están en consonancia con los objetivos de sostenibilidad.
- Revisión e iteración: revise y ajuste periódicamente los SLA para asegurarse de que estén en consonancia con los objetivos de sostenibilidad y rendimiento en constante cambio.

Recursos

Documentos relacionados:

- Understand resiliency patterns and trade-offs to architect efficiently in the cloud
- Importance of Service Level Agreement for SaaS Providers

Videos relacionados:

- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment

SUS02-BP03 Detención de la creación y el mantenimiento de los recursos no utilizados

Retire los activos no utilizados de su carga de trabajo para reducir el número de recursos en la nube necesarios para atender su demanda y minimizar los residuos.

Patrones comunes de uso no recomendados:

- No analiza su aplicación en busca de activos redundantes o que ya no son necesarios.
- No elimina los activos que son redundantes o que ya no son necesarios.

Beneficios de establecer esta práctica recomendada: la eliminación de los activos no utilizados libera recursos y mejora la eficiencia general de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Los activos no utilizados consumen recursos de la nube, como espacio de almacenamiento y potencia de computación. Con la identificación y eliminación de estos activos, podrá liberar estos recursos, lo que dará lugar a una arquitectura en la nube más eficiente. Lleve a cabo análisis periódicos en los activos de aplicaciones (como los informes precompilados, los conjuntos de datos y las imágenes estáticas) y los patrones de acceso a los activos para identificar cualquier tipo de redundancia, infrautilización y los posibles objetivos de retirada. Elimine esos activos redundantes para reducir el consumo de recursos en su carga de trabajo.

Pasos para la implementación

- Inventario: lleve a cabo un inventario exhaustivo para identificar todos los activos de su carga de trabajo.
- Análisis del uso: utilice herramientas de supervisión continua para identificar los activos estáticos que ya no sean necesarios.
- Eliminación de los activos que no se usan: elabore un plan para eliminar los activos que ya no sean necesarios.

  - Antes de eliminar un activo, evalúe el impacto de su eliminación en la arquitectura.
  - Consolide los recursos generados superpuestos para eliminar el procesamiento redundante.
  - Actualice las aplicaciones para que dejen de producir y almacenar activos que no sean necesarios.
- Comunicación con terceros: indique a terceros que dejen de producir y almacenar activos administrados en su nombre que ya no sean necesarios. Solicite la consolidación de los activos redundantes.
- Uso de políticas de ciclo de vida: utilice políticas de ciclo de vida para eliminar automáticamente los activos no utilizados.
  - Puede usar Amazon S3 Lifecycle para administrar los objetos a lo largo de su ciclo de vida.
  - Puede utilizar Amazon Data Lifecycle Manager para automatizar la creación, conservación y eliminación de instantáneas de Amazon EBS y las AMI basadas en Amazon EBS.
- Revisión y optimización: revise periódicamente la carga de trabajo para identificar y eliminar los activos no utilizados.

Recursos

Documentos relacionados:

- Optimizing your AWS Infrastructure for Sustainability, Part II: Storage
- How do I terminate active resources that I no longer need on my Cuenta de AWS?

Videos relacionados:

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Preserving and maximizing the value of digital media assets using Amazon S3
- AWS re:Invent 2023 - Optimize costs in your multi-account environments

SUS02-BP04 Optimización de la ubicación geográfica de las cargas de trabajo en función de sus requisitos de red

Seleccione para su carga de trabajo una ubicación y unos servicios en la nube que acorten la distancia que debe recorrer el tráfico de red y reduzcan el total de recursos de red necesarios para admitir su carga de trabajo.

Patrones comunes de uso no recomendados:

- Selecciona la región de la carga de trabajo en función de la propia ubicación.
- Consolida todos los recursos de la carga de trabajo en una ubicación geográfica.
- Todo el tráfico fluye a través de sus centros de datos existentes.

Beneficios de establecer esta práctica recomendada: colocar una carga de trabajo cerca de sus usuarios permite obtener la menor latencia, al tiempo que disminuye el movimiento de datos a través de la red y reduce el impacto medioambiental.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La infraestructura de Nube de AWS se crea en torno a opciones de ubicación como regiones, zonas de disponibilidad, grupos de ubicaciones y ubicaciones periféricas como AWS Outposts y zonas locales de AWS. Estas opciones de ubicación son las responsables de mantener la conectividad entre los componentes de las aplicaciones, los servicios en la nube, las redes periféricas y los centros de datos en las instalaciones.

Analice los patrones de acceso a la red en su carga de trabajo para identificar cómo utilizar estas opciones de ubicación en la nube y reducir la distancia que debe recorrer el tráfico de red.

Pasos para la implementación

- Analice los patrones de acceso a la red en su carga de trabajo para identificar cómo utilizan los usuarios su aplicación.
  - Utilice herramientas de supervisión, como Amazon CloudWatch y AWS CloudTrail, para recopilar datos sobre las actividades de la red.
  - Analice los datos para identificar el patrón de acceso a la red.
- Seleccione las regiones para la implementación de la carga de trabajo en función de los siguientes elementos clave:
  - Su objetivo de sostenibilidad: tal como se explica en Selección de regiones.
  - Ubicación de los datos: en el caso de las aplicaciones con gran cantidad de datos (como macrodatos y machine learning), el código de la aplicación debe ejecutarse lo más cerca posible de los datos.
  - Ubicación de los usuarios: para las aplicaciones orientadas al usuario, elija una región (o regiones) cercana a los usuarios de su carga de trabajo.

  - Otras restricciones: tenga en cuenta las limitaciones, como el costo y el cumplimiento, tal y como se explica en What to Consider when Selecting a Region for your Workloads.
- Utilice almacenamiento en caché local o soluciones de almacenamiento en caché de AWS para los activos de uso frecuente con el fin de mejorar el rendimiento, reducir el movimiento de datos y disminuir el impacto medioambiental.

| Servicio | Cuándo se debe usar |
| --- | --- |
| Amazon CloudFront | Se usa para almacenar en caché el contenido estático como imágenes, scripts y videos, así como el contenido dinámico como respuestas de API y aplicaciones web. |
| Amazon ElastiCache | Se usa para almacenar en caché el contenido de las aplicaciones web. |
| DynamoDB Accelerator | Se usa para agregar aceleración en memoria a sus tablas de DynamoDB. |

- Utilice servicios que puedan ayudarle a ejecutar el código más cerca de los usuarios de su carga de trabajo:

| Servicio | Cuándo se debe usar |
| --- | --- |
| Lambda@Edge | Se usa para las operaciones que utilizan muchos recursos de computación que se inician cuando los objetos no están en la memoria caché. |
| Amazon CloudFront Functions | Se usan para casos de uso sencillos como las manipulaciones de solicitudes o respuesta s HTTP(s) que pueden iniciarse mediante funciones de corta duración. |
| AWS IoT Greengrass | Se usa para ejecutar la computación local, la mensajería y el almacenamiento en caché de datos para los dispositivos conectados. |

- Use la agrupación de conexiones para permitir reutilizar las conexiones y reducir la cantidad de recursos necesarios.
- Use los almacenes de datos distribuidos que no se basen en conexiones persistentes y en actualizaciones sincrónicas por coherencia para atender a las poblaciones regionales.
- Reemplace la capacidad de red estática preaprovisionada por capacidad dinámica compartida y comparta el impacto en la sostenibilidad de la capacidad de red con otros suscriptores.

Recursos

Documentos relacionados:

- Optimizing your AWS Infrastructure for Sustainability, Part III: Networking
- Documentación de Amazon ElastiCache
- ¿Qué es Amazon CloudFront?
- Características clave de Amazon CloudFront
- Infraestructura global de AWS
- AWS Local Zones and AWS Outposts, choosing the right technology for your edge workload
- Grupos de ubicación
- Zonas locales de AWS
- AWS Outposts

Videos relacionados:

- Demystifying data transfer on AWS
- Scaling network performance on next-gen Amazon EC2 instances
- AWS Local Zones Explainer Video
- AWS Outposts: Overview and How it Works
- AWS re:Invent 2023 - A migration strategy for edge and on-premises workloads
- AWS re:Invent 2021 - AWS Outposts: Bringing the AWS experience on premises
- AWS re:Invent 2020 - AWS Wavelength: Run apps with ultra-low latency at 5G edge
- AWS re:Invent 2022 - AWS Local Zones: Building applications for a distributed edge
- AWS re:Invent 2021 - Building low-latency websites with Amazon CloudFront

- AWS re:Invent 2022 - Improve performance and availability with AWS Global Accelerator
- AWS re:Invent 2022 - Build your global wide area network using AWS
- AWS re:Invent 2020: Global traffic management with Amazon Route 53

Ejemplos relacionados:

- Talleres de redes de AWS
- Architecting for sustainability - Minimize data movement across networks

SUS02-BP05 Optimización de los recursos de los miembros del equipo para las actividades efectuadas

Optimice los recursos proporcionados a los miembros del equipo para minimizar el impacto en la sostenibilidad medioambiental a la vez que se cubren sus necesidades.

Patrones comunes de uso no recomendados:

- Ignora el impacto de los dispositivos utilizados por los miembros de su equipo en la eficacia global de su aplicación en la nube.
- Administra y actualiza manualmente los recursos que utilizan los miembros del equipo.

Beneficios de establecer esta práctica recomendada: la optimización de los recursos de los miembros del equipo mejora la eficiencia general de las aplicaciones basadas en la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Analice los dispositivos que usan los miembros de su equipo para consumir sus servicios, el ciclo de vida que se espera que tengan y el impacto económico y en la sostenibilidad. Implemente estrategias para optimizar estos recursos. Por ejemplo, lleve a cabo operaciones complejas (como la representación y la compilación) en escritorios en una infraestructura escalable con un uso intensivo, en lugar de hacerlo en sistemas de usuarios únicos de gran potencia infrautilizados.

Pasos para la implementación

- Uso de estaciones de trabajo de bajo consumo energético: proporcione a los miembros del equipo estaciones de trabajo y periféricos que ahorren energía. Utilice características de administración de

energía eficiente (como el modo de bajo consumo) en estos dispositivos para reducir el consumo de energía.

- Uso de la virtualización: use escritorios virtuales y streaming de aplicaciones para limitar los requisitos de dispositivos y actualizaciones.
- Fomento de la colaboración remota: anime a los miembros del equipo a utilizar herramientas de colaboración remota, como, por ejemplo, Amazon Chime o AWS Wickr para reducir la necesidad de viajar y las emisiones de carbono asociadas.
- Uso de software de bajo consumo energético: proporcione a los miembros del equipo un software de bajo consumo energético mediante la eliminación y la desactivación de características y procesos innecesarios.
- Administración de los ciclos de vida: evalúe el impacto de los procesos y los sistemas en el ciclo de vida de los dispositivos y seleccione aquellas soluciones que minimizan los requisitos para el reemplazo de dispositivos a la vez que satisfacen los requisitos empresariales. Mantenga y actualice periódicamente las estaciones de trabajo o el software para mantener y mejorar la eficiencia.
- Administración remota de dispositivos: implemente la administración remota de los dispositivos para reducir la necesidad de hacer viajes de negocios.
  - Administrador de flotas de AWS Systems Manager es una experiencia de interfaz de usuario (IU) unificada que le ayuda a administrar de forma remota los nodos que se ejecutan en AWS o en un entorno en las instalaciones.

Recursos

Documentos relacionados:

- What is Amazon WorkSpaces?
- Cost Optimizer for Amazon WorkSpaces
- Documentación de Amazon AppStream 2.0
- NICE DCV

Videos relacionados:

- Managing cost for Amazon WorkSpaces on AWS

SUS02-BP06 Implementación del almacenamiento en búfer o la limitación para aplanar la curva de demanda

El almacenamiento en búfer y la limitación aplanan la curva de demanda y reducen la capacidad aprovisionada necesaria para su carga de trabajo.

Patrones comunes de uso no recomendados:

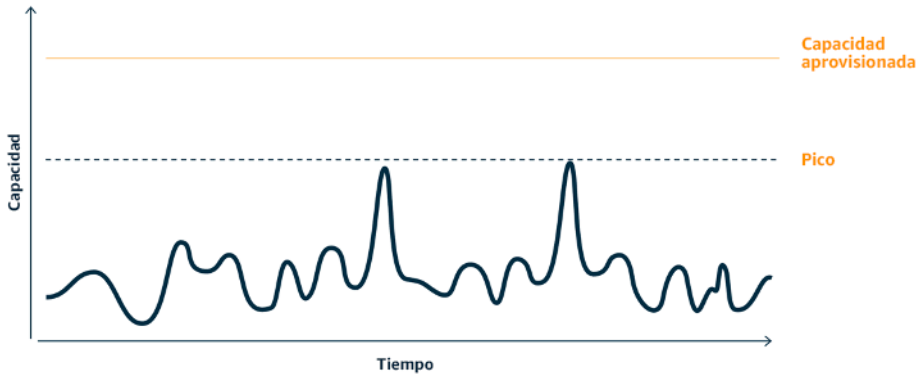
- Procesa las solicitudes de los clientes inmediatamente cuando no es necesario.
- No analiza los requisitos de las solicitudes de los clientes.

Beneficios de establecer esta práctica recomendada: al aplanar la curva de demanda, se reduce la capacidad aprovisionada requerida para la carga de trabajo. La reducción de la capacidad aprovisionada implica un menor consumo de energía y un menor impacto medioambiental.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

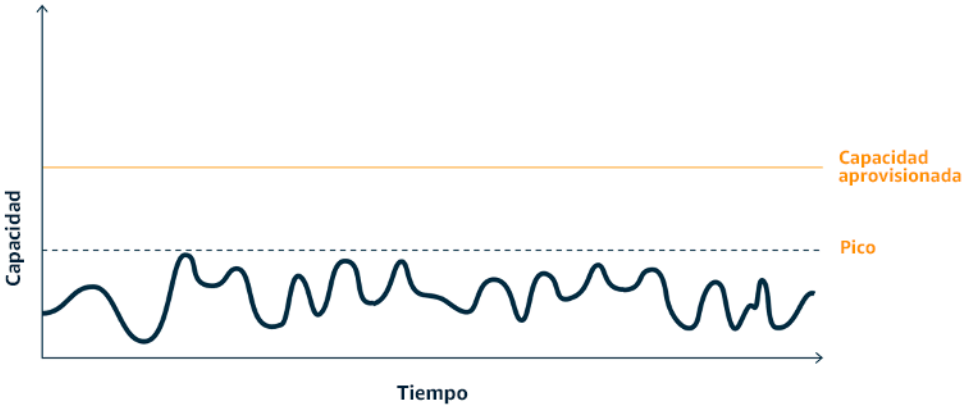
Guía para la implementación

El aplanamiento de la curva de demanda de la carga de trabajo puede ayudarle a reducir la capacidad aprovisionada para una carga de trabajo y a reducir su impacto medioambiental. Supongamos una carga de trabajo con la curva de demanda que se muestra en la siguiente figura. Esta carga de trabajo tiene dos picos y, para gestionarlos, se aprovisiona la capacidad de recursos que muestra la línea naranja. Los recursos y la energía utilizados para esta carga de trabajo no están indicados por el área situada debajo de la curva de demanda, sino por el área situada debajo de la línea de capacidad aprovisionada, ya que esta capacidad se necesita para gestionar esos dos picos.




Curva de demanda con dos picos distintos que requieren una elevada capacidad aprovisionada.

Puede utilizar el almacenamiento en búfer o la limitación para modificar la curva de demanda y suavizar los picos, lo que significa menos capacidad aprovisionada y menos energía consumida. Implemente limitaciones cuando sus clientes puedan llevar a cabo reintentos. Implemente el almacenamiento en búfer para almacenar la solicitud y aplazar el procesamiento para más adelante.




Efecto de la limitación en la curva de demanda y en la capacidad aprovisionada.

Pasos para la implementación

- Analice las solicitudes de los clientes para determinar cómo responder a ellas. Entre las preguntas que hay que tener en cuenta se incluyen las siguientes:
  - ¿Puede procesarse esta solicitud de forma asíncrona?
  - ¿Tiene el cliente capacidad de reintentos?
- Si el cliente tiene capacidad de reintentos, puede implementar la limitación, que le indica al origen que si no puede atender la solicitud en el momento actual debe intentarlo más tarde.
  - Puede utilizar Amazon API Gateway para implementar la limitación.
- En el caso de los clientes que no pueden hacer reintentos, es necesario implementar un búfer para aplanar la curva de demanda. Un búfer aplaza el procesamiento de las solicitudes, por lo que permite a las aplicaciones que se ejecutan a diferentes ritmos comunicarse de forma efectiva. El enfoque basado en búfer utiliza una cola o una secuencia para aceptar mensajes de los productores. De este modo, los consumidores pueden leer y procesar los mensajes, lo que permite que dichos mensajes se ejecuten a la velocidad que cumpla con los requisitos empresariales de los consumidores.
  - Amazon Simple Queue Service (Amazon SQS) es un servicio gestionado que proporciona colas que permiten a un solo consumidor leer mensajes individuales.

  - Amazon Kinesis ofrece una secuencia que permite que muchos consumidores lean los mismos mensajes.
- Analice la demanda general, la tasa de cambio y el tiempo de respuesta requerido para dimensionar correctamente la limitación o el búfer requeridos.

Recursos

Documentos relacionados:

- Getting started with Amazon SQS
- Application integration Using Queues and Messages
- Managing and monitoring API throttling in your workloads
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Application integration Using Queues and Messages

Videos relacionados:

- AWS re:Invent 2022 - Application integration patterns for microservices
- AWS re:Invent 2023 - Smart savings: Amazon EC2 cost-optimization strategies
- AWS re:Invent 2023 - Advanced integration patterns & trade-offs for loosely coupled systems

## Software y arquitectura

Pregunta

- SUS 3 ¿Cómo puede sacar partido de los patrones de software y de arquitectura para respaldar sus objetivos de sostenibilidad?

### SUS 3 ¿Cómo puede sacar partido de los patrones de software y de arquitectura para respaldar sus objetivos de sostenibilidad?

Implemente patrones que permitan suavizar la carga y mantener un uso elevado consistente de los recursos implementados para minimizar los recursos consumidos. Puede haber componentes que queden inactivos debido a la falta de uso relacionada con los cambios en el comportamiento de los usuarios a lo largo del tiempo. Revise los patrones y la arquitectura para consolidar los componentes infrautilizados a fin de incrementar el uso general. Retire los componentes que ya no son necesarios.

Analice el rendimiento de los componentes de su carga de trabajo y optimice aquellos que consumen la mayor cantidad de recursos. Tenga en cuenta los dispositivos que usan los clientes para acceder a sus servicios e implemente patrones para minimizar la necesidad de llevar a cabo actualizaciones de los dispositivos.

Prácticas recomendadas

- SUS03-BP01 Optimización del software y la arquitectura para los trabajos asíncronos y programados
- SUS03-BP02 Eliminación o refactorización de los componentes de cargas de trabajo con uso reducido o nulo
- SUS03-BP03 Optimización de las áreas de código que consumen la mayor parte del tiempo o de los recursos
- SUS03-BP04 Optimización del impacto en los dispositivos y equipos
- SUS03-BP05 Uso de los patrones de software y las arquitecturas que mejor respaldan los patrones de almacenamiento y el acceso a los datos

SUS03-BP01 Optimización del software y la arquitectura para los trabajos asíncronos y programados

Utilice patrones de software y arquitectura eficientes, como los basados en colas, para mantener una utilización elevada y coherente de los recursos implementados.

Patrones comunes de uso no recomendados:

- Aprovisiona en exceso los recursos de su carga de trabajo en la nube para hacer frente a picos imprevistos de la demanda.
- Usa una arquitectura que no desacopla los emisores y los receptores de mensajes asíncronos mediante un componente de mensajería.

Beneficios de establecer esta práctica recomendada:

- Los patrones de software y arquitectura eficientes minimizan los recursos no utilizados en la carga de trabajo y mejoran la eficiencia global.
- Posibilidad de escalar el procesamiento independientemente de la recepción de mensajes asíncronos.
- Mediante un componente de mensajería, tendrá unos requisitos de disponibilidad más relajados que podrá cumplir con menos recursos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Utilice patrones de arquitectura eficientes, como una arquitectura basada en eventos, que permita un uso uniforme de los componentes y minimice el aprovisionamiento excesivo de la carga de trabajo. El uso de patrones de arquitectura eficientes minimiza los recursos inactivos por falta de uso debido a cambios en la demanda a lo largo del tiempo.

Comprenda los requisitos de los componentes de la carga de trabajo y adopte patrones de arquitectura que aumenten la utilización global de los recursos. Retire los componentes que ya no son necesarios.

Pasos para la implementación

- Analice la demanda de su carga de trabajo para determinar cómo responder a ella.
- En el caso de solicitudes o trabajos que no requieran respuestas síncronas, utilice arquitecturas basadas en colas y empleados de escalado automático para maximizar la utilización. A continuación, encontrará algunos ejemplos de cuándo podría plantearse una arquitectura basada en colas:

| Mecanismo de colas | Descripción |
| --- | --- |
| Colas de trabajo de AWS Batch | Los trabajos de AWS Batch se envían a una cola de trabajos en la que permanecen hasta que pueden programarse para ejecutarse en un entorno de computación. |
| Amazon Simple Queue Service e instancias puntuales de Amazon EC2 | Emparejamiento de instancias de Spot y Amazon SQS para crear una arquitectura eficiente y tolerante a errores. |

- En el caso de solicitudes o trabajos que puedan procesarse en cualquier momento, utilice mecanismos de programación para procesar los trabajos por lotes y obtener una mayor eficacia. A continuación, se presentan algunos ejemplos de mecanismos de programación en AWS:

| Mecanismo de programación | Descripción |
| --- | --- |
| Programador de Amazon EventBridge | Funcionalidad de Amazon EventBridge que permite crear, ejecutar y administrar tareas programadas a escala. |
| Programación basada en el tiempo de AWS Glue | Defina una programación basada en el tiempo para sus rastreadores y trabajos en AWS Glue. |
| Tareas programadas de Amazon Elastic Container Service (Amazon ECS) | Amazon ECS admite la creación de tareas programadas. Las tareas programadas utilizan las reglas de Amazon EventBridge para ejecutar tareas según una programación o en respuesta a un evento de EventBridge. |
| Programador de instancias | Configure los horarios de inicio y parada para sus instancias de Amazon EC2 y Amazon Relational Database Service. |

- Si utiliza mecanismos de sondeo y webhooks en su arquitectura, reemplácelos por eventos. Utilice arquitecturas basadas en eventos para crear cargas de trabajo altamente eficientes.
- Aproveche la tecnología sin servidor en AWS para eliminar la infraestructura con exceso de aprovisionamiento.
- Dimensione correctamente los componentes individuales de su arquitectura para evitar recursos inactivos mientras se espera la entrada.
  - Puede utilizar las recomendaciones de redimensionamiento de AWS Cost Explorer o AWS Compute Optimizer para identificar oportunidades de redimensionamiento.
  - Para obtener más información, consulte Ajuste del tamaño: aprovisionamiento de instancias para adaptarse a las cargas de trabajo.

Recursos

Documentos relacionados:

- What is Amazon Simple Queue Service?

- ¿Qué es Amazon MQ?
- Escalado basado en Amazon SQS
- Qué es AWS Step Functions?
- Qué es AWS Lambda?
- Uso de AWS Lambda con Amazon SQS
- ¿Qué es Amazon EventBridge?
- Managing Asynchronous Workflows with a REST API

Videos relacionados:

- AWS re:Invent 2023 - Navigating the journey to serverless event-driven architecture
- AWS re:Invent 2023 - Using serverless for event-driven architecture & domain-driven design
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- Asynchronous Message Patterns | AWS Events

Ejemplos relacionados:

- Event-driven architecture with AWS Graviton Processors and Amazon EC2 Spot Instances

SUS03-BP02 Eliminación o refactorización de los componentes de cargas de trabajo con uso reducido o nulo

Elimine los componentes que ya no se usan ni se necesitan y refactorice aquellos con un uso reducido para minimizar el desperdicio en su carga de trabajo.

Patrones comunes de uso no recomendados:

- No comprueba periódicamente el nivel de uso de los componentes individuales de la carga de trabajo.
- No comprueba ni analiza recomendaciones de herramientas de dimensionamiento de AWS como AWS Compute Optimizer.

Beneficios de establecer esta práctica recomendada: la eliminación de los componentes no utilizados minimiza el desperdicio y mejora la eficiencia general de la carga de trabajo en la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los componentes no utilizados o infrautilizados de una carga de trabajo en la nube consumen recursos de computación, de almacenamiento o de red innecesarios. Elimine o refactorice estos componentes para reducir directamente el uso innecesario y mejorar la eficiencia general de una carga de trabajo en la nube. Este es un proceso de mejora iterativo que puede iniciarse por cambios en la demanda o por el lanzamiento de un nuevo servicio en la nube. Por ejemplo, una disminución significativa en el tiempo de ejecución de una función de AWS Lambda puede ser un indicador de que es necesario reducir el tamaño de la memoria. Además, a medida que AWS lanza nuevos servicios y características, los servicios y la arquitectura óptimos para su carga de trabajo también pueden cambiar.

Supervise continuamente la actividad de la carga de trabajo y busque oportunidades para mejorar el nivel de uso de los componentes individuales. Con la eliminación de los componentes ociosos y con las actividades de redimensionamiento, cumplirá los requisitos de su empresa con el menor número de recursos en la nube.

Pasos para la implementación

- Haga un inventario de sus recursos de AWS: cree un inventario de sus recursos de AWS. En AWS, puede activar Explorador de recursos de AWS para explorar y organizar los recursos de AWS. Para obtener más información, consulte AWS re:Invent 2022 - How to manage resources and applications at scale on AWS.
- Monitoree el uso: monitoree y capture las métricas de uso de los componentes críticos de su carga de trabajo (como el uso de la CPU, el uso de la memoria o el rendimiento de la red en las métricas de Amazon CloudWatch).
- Identifique los componentes no utilizados: identifique los componentes no utilizados o infrautilizados de su arquitectura.
  - Para las cargas de trabajo estables, compruebe las herramientas de dimensionamiento de AWS, como, por ejemplo, AWS Compute Optimizer, a intervalos regulares para identificar los componentes inactivos, no utilizados o infrautilizados.
  - En el caso de las cargas de trabajo efímeras, evalúe las métricas de uso para identificar los componentes inactivos, no utilizados o infrautilizados.
- Retire los componentes no utilizados: retire los componentes y los recursos asociados (como las imágenes de Amazon ECR) que ya no sean necesarios.

  - Automated Cleanup of Unused Images in Amazon ECR
  - Eliminar volúmenes de Amazon Elastic Block Store (Amazon EBS) no utilizados con AWS Config y AWS Systems Manager
- Refactorice los componentes infrautilizados: refactorice o consolide los componentes infrautilizados con otros recursos para mejorar la eficiencia de uso. Por ejemplo, puede aprovisionar varias bases de datos pequeñas en una sola instancia de base de datos de Amazon RDS en lugar de ejecutar las bases de datos en instancias individuales infrautilizadas.
- Evalúe las mejoras: conozca los recursos que aprovisiona su carga de trabajo para completar una unidad de trabajo. Utilice esta información para evaluar las mejoras logradas al eliminar o refactorizar los componentes.
  - Mida la eficiencia de la nube y haga un seguimiento de esta con métricas indirectas de sostenibilidad. Parte I: ¿Qué son las métricas intermedias?
  - Mida la eficiencia de la nube y haga un seguimiento de esta con métricas indirectas de sostenibilidad. Parte II: Establezca una cartera de métricas

Recursos

Documentos relacionados:

- AWS Trusted Advisor
- Qué es Amazon CloudWatch?
- Ajuste del tamaño: aprovisionamiento de instancias para adaptarse a las cargas de trabajo
- Optimizing your cost with Rightsizing Recommendations

Videos relacionados:

- AWS re:Invent 2023 - Capacity, availability, cost efficiency: Pick three

SUS03-BP03 Optimización de las áreas de código que consumen la mayor parte del tiempo o de los recursos

Optimice el código que se ejecuta en los distintos componentes de su arquitectura para minimizar el uso de los recursos y, a la vez, maximizar el rendimiento.

Patrones comunes de uso no recomendados:

- Ignora la optimización del código para el uso de recursos.
- Normalmente responde a los problemas de rendimiento con un aumento de los recursos.
- Su proceso de revisión y desarrollo del código no hace un seguimiento de los cambios de rendimiento.

Beneficios de establecer esta práctica recomendada: el uso de código eficiente minimiza el consumo de recursos y mejora el rendimiento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Es fundamental examinar cada área funcional, incluido el código de una aplicación con arquitectura de nube, para optimizar el consumo de recursos y el rendimiento. Supervise continuamente el rendimiento de la carga de trabajo en los entornos de creación y producción e identifique oportunidades para mejorar los fragmentos de código que tienen un uso de recursos especialmente elevado. Adopte un proceso de revisión periódico para identificar errores o antipatrones en su código que utilicen los recursos de forma ineficiente. Use algoritmos sencillos y eficaces que produzcan los mismos resultados para su caso de uso.

Pasos para la implementación

- Uso de un lenguaje de programación eficiente: utilice un sistema operativo y un lenguaje de programación eficientes para la carga de trabajo. Para obtener más información sobre los lenguajes de programación energéticamente eficientes (incluido Rust), consulte Sustainability with Rust.
- Uso de un complemento de codificación de IA: considere la posibilidad de utilizar un complemento de codificación de IA, como Amazon Q Developer, para escribir código de manera eficiente.
- Automatización de las revisiones de código: durante el desarrollo de sus cargas de trabajo, adopte un proceso automatizado de revisión del código para mejorar la calidad e identificar errores y antipatrones.
  - Automate code reviews with Amazon CodeGuru Reviewer
  - Detecting concurrency bugs with Amazon CodeGuru
  - Raising code quality for Python applications using Amazon CodeGuru
- Uso de un generador de perfiles de código: use un generador de perfiles de código para identificar las áreas de código que emplean más tiempo o recursos como destino de la optimización.
  - Reducing your organization's carbon footprint with Amazon CodeGuru Profiler

  - Understanding memory usage in your Java application with Amazon CodeGuru Profiler
  - Improving customer experience and reducing cost with Amazon CodeGuru Profiler
- Supervisión y optimización: use recursos de supervisión continua para identificar los componentes con altos requisitos de recursos o con una configuración subóptima.
  - Reemplace los algoritmos que hacen un uso intensivo de la computación por versiones más sencillas y eficientes que produzcan el mismo resultado.
  - Elimine el código innecesario, como la ordenación y el formato.
- Uso de la refactorización o la transformación del código: explore la posibilidad de transformar el código de Amazon Q para el mantenimiento y las actualizaciones de las aplicaciones.
  - Upgrade language versions with Amazon Q Code Transformation
  - AWS re:Invent 2023 - Automate app upgrades & maintenance using Amazon Q Code Transformation

Recursos

Documentos relacionados:

- What is Amazon CodeGuru Profiler?
- FPGA instances
- SDK de AWS en Herramientas para crear en AWS

Videos relacionados:

- Improve Code Efficiency Using Amazon CodeGuru Profiler
- Automate Code Reviews and Application Performance Recommendations with Amazon CodeGuru

SUS03-BP04 Optimización del impacto en los dispositivos y equipos

Comprenda los dispositivos y los equipos empleados en la arquitectura y utilice estrategias para reducir su uso. Esto puede minimizar el impacto medioambiental global de su carga de trabajo en la nube.

Patrones comunes de uso no recomendados:

- Ignora el impacto medioambiental de los dispositivos que utilizan sus clientes.

- Administra y actualiza manualmente los recursos que utilizan los clientes.

Beneficios de establecer esta práctica recomendada: implementación de patrones y características de software optimizados para el dispositivo del cliente puede reducir el impacto medioambiental general de la carga de trabajo en la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La implementación de patrones y características de software optimizados para el dispositivo del cliente puede reducir el impacto medioambiental general de la carga de trabajo en la nube de varias maneras:

- La implementación de nuevas características compatibles con versiones anteriores puede reducir el número de reemplazos de hardware.
- La optimización de una aplicación para que funcione de forma eficiente en los dispositivos puede contribuir a reducir su consumo de energía y a prolongar la duración de su batería (si funcionan con ella).
- La optimización de una aplicación para dispositivos también puede reducir la transferencia de datos a través de la red.

Comprenda los dispositivos y equipos utilizados en su arquitectura, su ciclo de vida previsto y el impacto de reemplazar esos componentes. Implemente patrones y características de software que puedan minimizar el consumo de energía del dispositivo, así como la necesidad de los clientes de reemplazar el dispositivo y actualizarlo manualmente.

Pasos para la implementación

- Inventario: haga un inventario de los dispositivos utilizados en su arquitectura. Los dispositivos pueden ser móviles, tabletas, dispositivos IoT, luces inteligentes o incluso dispositivos inteligentes en una fábrica.
- Uso de dispositivos que ahorren energía: considere la posibilidad de utilizar dispositivos que ahorren energía en su arquitectura. Use las configuraciones de administración de energía en los dispositivos para entrar en el modo de bajo consumo cuando no estén en uso.
- Ejecución de aplicaciones eficientes: optimice la aplicación que se ejecuta en los dispositivos:

    - Utilice estrategias como la ejecución de tareas en segundo plano para reducir su consumo de energía.
    - Tenga en cuenta la latencia y el ancho de banda de la red al crear cargas e implemente capacidades que ayuden al funcionamiento óptimo de las aplicaciones en enlaces de alta latencia y ancho de banda bajo.
    - Convierta las cargas útiles y los archivos a los formatos optimizados que requieren los dispositivos. Por ejemplo, puede usar Amazon Elastic Transcoder o AWS Elemental MediaConvert para convertir archivos multimedia digitales de gran calidad y tamaño en formatos que los usuarios puedan reproducir en dispositivos móviles, tablets, navegadores web y televisiones conectadas.
    - Lleve a cabo las actividades con un uso intensivo de los recursos de computación (como la representación de imágenes) del servidor o use el streaming de aplicaciones para mejorar la experiencia del usuario en los dispositivos más antiguos.
    - Segmente y pagine los resultados, sobre todo en las sesiones interactivas, para administrar las cargas y limitar los requisitos de almacenamiento local.
- Implicación de los proveedores: trabaje con proveedores de dispositivos que usen materiales sostenibles y que ofrezcan transparencia en sus cadenas de suministro y certificaciones medioambientales.
- Uso de actualizaciones vía inalámbrica (OTA): use el mecanismo automatizado vía inalámbrica (OTA) para implementar actualizaciones en uno o varios dispositivos.
    - Puede utilizar una canalización de CI/CD para actualizar las aplicaciones móviles.
    - Puede utilizar AWS IoT Device Management para gestionar de forma remota los dispositivos conectados a gran escala.
- Uso de granjas de dispositivos administrados: para probar nuevas características y actualizaciones, utilice granjas de dispositivos administrados con conjuntos representativos de hardware e itere el desarrollo para maximizar los dispositivos admitidos. Para obtener más información, consulte SUS06-BP05 Uso de granjas de dispositivos administrados para pruebas.
- Continuación de la supervisión y mejora: haga un seguimiento del consumo de energía de los dispositivos para identificar las áreas de mejora. Utilice nuevas tecnologías o prácticas recomendadas para mejorar los impactos medioambientales de estos dispositivos.

Recursos

Documentos relacionados:

- Qué es AWS Device Farm?
- Documentación de aplicaciones de WorkSpaces
- NICE DCV
- OTA tutorial for updating firmware on devices running FreeRTOS
- Optimizing Your IoT Devices for Environmental Sustainability

Videos relacionados:

- AWS re:Invent 2023 - Improve your mobile and web app quality using AWS Device Farm

SUS03-BP05 Uso de los patrones de software y las arquitecturas que mejor respaldan los patrones de almacenamiento y el acceso a los datos

Analice cómo se usan los datos en la carga de trabajo, cómo los consumen los usuarios, cómo se transfieren y cómo se almacenan. Utilice patrones y arquitecturas de software que admitan mejor el acceso a los datos y el almacenamiento para minimizar los recursos de computación, redes y almacenamiento necesarios para admitir la carga de trabajo.

Patrones comunes de uso no recomendados:

- Supone que todas las cargas de trabajo tienen patrones similares de almacenamiento y acceso a los datos.
- Solo utiliza un nivel de almacenamiento, asumiendo que todas las cargas de trabajo encajan en ese nivel.
- Supone que los patrones de acceso a los datos se mantendrán coherentes a lo largo del tiempo.
- Su arquitectura admite una posible ampliación de acceso a los datos, lo que provoca que los recursos permanezcan inactivos la mayor parte del tiempo.

Beneficios de establecer esta práctica recomendada: la selección y optimización de su arquitectura en función de los patrones de acceso y almacenamiento de datos le ayudará a reducir la complejidad del desarrollo y a aumentar la utilización general. Saber cuándo utilizar las tablas globales, las particiones de datos y el almacenamiento en caché le ayudará a disminuir la sobrecarga operativa y a escalar en función de sus necesidades de carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Para mejorar la sostenibilidad de las cargas de trabajo a largo plazo, utilice patrones de arquitectura que admitan las características de acceso y almacenamiento de datos para su carga de trabajo. Estos patrones lo ayudan a recuperar y procesar los datos de manera eficiente. Por ejemplo, utilice una arquitectura de datos moderna en AWS con servicios diseñados específicamente y optimizados para sus casos de uso de análisis exclusivos. Estos patrones de arquitectura permiten un procesamiento de datos eficaz y reducen el uso de recursos.

Pasos para la implementación

- Conozca las características de los datos: analice las características de los datos y los patrones de acceso para identificar la configuración correcta de sus recursos en la nube. Entre las características clave que se deben tener en cuenta se incluyen las siguientes:
  - Tipo de datos: estructurados, semiestructurados y no estructurados
  - Crecimiento de datos: limitado, ilimitado
  - Durabilidad de los datos: persistentes, efímeros o transitorios
  - Patrones de acceso: lecturas o escrituras, frecuencia de actualización, con picos o constantes
- Utilice patrones de arquitectura óptimos: utilice los patrones de arquitectura que mejor admitan los patrones de acceso y almacenamiento de datos.
  - Patrones para habilitar la persistencia de datos
  - Let's Architect! Modern data architectures
  - Databases on AWS: The Right Tool for the Right Job
- Utilice servicios diseñados específicamente: utilice tecnologías que se adapten a su propósito.
  - Utilice tecnologías que funcionen de forma nativa con datos comprimidos.
    - Compatibilidad con la compresión de Athena
    - Opciones de formato para las entradas y salidas de ETL en AWS Glue
    - Carga de archivos de datos comprimidos desde Amazon S3 con Amazon Redshift
  - Utilice servicios de análisis diseñados específicamente para el procesamiento de datos en su arquitectura. Para obtener más información sobre los servicios de análisis diseñados específicamente de AWS, consulte AWS re:Invent 2022 - Building modern data architectures on AWS.
  - Use el motor de base de datos que mejor admita su patrón de consulta dominante. Administre los índices de las bases de datos para que se hagan consultas de forma eficaz. Para obtener

más información, consulte Bases de datos de AWS y AWS re:Invent 2022 - Modernize apps with purpose-built databases.

- Minimice la transferencia de datos: seleccione protocolos de red que reduzcan la cantidad de capacidad de red consumida en su arquitectura.

Recursos

Documentos relacionados:

- Uso de COPY con formatos de datos de columnas con Amazon Redshift
- Conversión del formato de registros de entrada en Kinesis Data Firehose
- Mejora del rendimiento de las consultas en Amazon Athena con la conversión a formato de columnas
- Monitoreo de la carga de base de datos con Performance Insights en Amazon Aurora
- Monitoreo de la carga de base de datos con Performance Insights en Amazon RDS
- Clase de almacenamiento Amazon S3 Intelligent-Tiering
- Build a CQRS event store with Amazon DynamoDB

Videos relacionados:

- AWS re:Invent 2022 - Building data mesh architectures on AWS
- AWS re:Invent 2023 - Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023 - Improve Amazon EBS efficiency and be more cost-efficient
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2023 - Advanced event-driven patterns with Amazon EventBridge

Ejemplos relacionados:

- AWS Purpose Built Databases Workshop
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS

# Datos

Pregunta

- SUS 4 ¿Cómo puede aprovechar los patrones y las políticas de administración de datos para admitir sus objetivos de sostenibilidad?

## SUS 4 ¿Cómo puede aprovechar los patrones y las políticas de administración de datos para admitir sus objetivos de sostenibilidad?

Implemente prácticas de administración de datos para reducir el almacenamiento aprovisionado que se necesita para admitir la carga de trabajo y los recursos necesarios para su uso. Comprenda sus datos y use las configuraciones y tecnologías de almacenamiento que respalden con más eficacia al valor empresarial de los datos y la forma en que se usan. Haga que el ciclo de vida de los datos incluya un almacenamiento más eficaz con un menor rendimiento cuando disminuyan los requisitos y elimine los datos que ya no se requieran.

Prácticas recomendadas

- SUS04-BP01 Implementación de una política de clasificación de datos
- SUS04-BP02 Uso de tecnologías que admiten patrones de almacenamiento y acceso a los datos
- SUS04-BP03 Uso de políticas para administrar el ciclo de vida de los conjuntos de datos
- SUS04-BP04 Uso de la elasticidad y la automatización para ampliar el almacenamiento de bloques o el sistema de archivos
- SUS04-BP05 Eliminación de datos innecesarios o redundantes
- SUS04-BP06 Uso de sistemas de archivos o almacenamiento compartidos para acceder a datos comunes
- SUS04-BP07 Minimización del movimiento de datos entre redes
- SUS04-BP08 Copias de seguridad de los datos solo cuando sea difícil volver a crearlos

### SUS04-BP01 Implementación de una política de clasificación de datos

Clasifique los datos para comprender su criticidad para los resultados empresariales y elija el nivel de almacenamiento de bajo consumo adecuado para almacenar los datos.

Patrones comunes de uso no recomendados:

- No identificar activos de datos con características similares (como sensibilidad, criticidad empresarial o requisitos normativos) que se estén procesando o almacenando.
- No ha implementado un catálogo de datos para inventariar sus activos de datos.

Beneficios de establecer esta práctica recomendada: la implementación de una política de clasificación de datos le permite determinar el nivel de almacenamiento de mayor eficiencia energética para los datos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

La clasificación de datos implica la identificación de los tipos de datos que se están procesando y almacenando en un sistema de información propiedad de una organización o controlado por ella. También implica tomar una determinación sobre la criticidad de los datos y la posible repercusión de su divulgación, pérdida o uso indebido.

Implemente la política de clasificación de datos mediante un trabajo en sentido inverso a partir del uso contextual de los datos y la creación de un esquema de categorización que tenga en cuenta el nivel de criticidad de un conjunto de datos determinado para las operaciones de una organización.

Pasos para la implementación

- Inventario de los datos: haga un inventario de los distintos tipos de datos que existen para su carga de trabajo.
- Agrupación de los datos: determine la gravedad, la confidencialidad, la integridad y la disponibilidad de los datos en función del riesgo para la organización. Utilice estos requisitos para agrupar los datos en uno de los niveles de clasificación de datos que adopte. Como ejemplo, consulte Cuatro sencillos pasos para clasificar los datos y proteger una startup.
- Definición de los niveles y las políticas de clasificación de datos: en cada grupo de datos, defina el nivel de clasificación de datos (por ejemplo, públicos o confidenciales) y las políticas de gestión. Etiquete los datos en consecuencia. Para obtener más información sobre las categorías de clasificación de datos, consulte el documento técnico Data Classification.
- Revisiones periódicas: revise y audite periódicamente su entorno para detectar datos sin etiquetar ni clasificar. Utilice la automatización para identificar estos datos y clasifique y etiquete los datos de forma adecuada. Como ejemplo, consulte Data Catalog and crawlers in AWS Glue.
- Establecimiento de un catálogo de datos: establezca un catálogo de datos que proporcione capacidades de auditoría y gobierno.

- Documentación: documente las políticas de clasificación de datos y los procedimientos de gestión para cada clase de datos.

Recursos

Documentos relacionados:

- Leveraging Nube de AWS to Support Data Classification
- Tag policies from AWS Organizations

Videos relacionados:

- AWS re:Invent 2022 - Enabling agility with data governance on AWS
- AWS re:Invent 2023 - Data protection and resilience with AWS storage

SUS04-BP02 Uso de tecnologías que admiten patrones de almacenamiento y acceso a los datos

Use las tecnologías de almacenamiento que mejor respalden la forma en que accede y guarda sus datos a fin de minimizar los recursos aprovisionados para admitir la carga de trabajo.

Patrones comunes de uso no recomendados:

- Supone que todas las cargas de trabajo tienen patrones similares de almacenamiento y acceso a los datos.
- Solo utiliza un nivel de almacenamiento, asumiendo que todas las cargas de trabajo encajan en ese nivel.
- Supone que los patrones de acceso a los datos se mantendrán coherentes a lo largo del tiempo.

Beneficios de establecer esta práctica recomendada: seleccionar y optimizar sus tecnologías de almacenamiento en función de los patrones de acceso y almacenamiento de datos le ayudará a reducir los recursos necesarios en la nube para satisfacer sus necesidades empresariales y a mejorar la eficacia general de la carga de trabajo en la nube.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Seleccione la solución de almacenamiento que mejor se adapte a sus patrones de acceso, o bien considere cambiar sus patrones de acceso de modo que se adapten a la solución de almacenamiento, a fin de maximizar la eficiencia del rendimiento.

Pasos para la implementación

- Evaluación de las características de los datos y el acceso: evalúe las características de sus datos y su patrón de acceso para recopilar las características clave de sus necesidades de almacenamiento. Entre las características clave que se deben tener en cuenta se incluyen las siguientes:
  - Tipo de datos: estructurados, semiestructurados y no estructurados
  - Crecimiento de datos: limitado, ilimitado
  - Durabilidad de los datos: persistentes, efímeros o transitorios
  - Patrones de acceso: lecturas o escrituras, frecuencia de actualización, con picos o constantes
- Elección de la tecnología de almacenamiento adecuada: migre los datos a la tecnología de almacenamiento adecuada que sea compatible con las características de sus datos y su patrón de acceso. A continuación, le presentamos algunos ejemplos de tecnologías de almacenamiento de AWS y sus principales características:

| Tipo | Tecnología | Características clave |
| --- | --- | --- |
| Almacenamiento de objetos | Amazon S3 | Servicio de almacenamiento de objetos con escalabilidad ilimitada, alta disponibilidad y varias opciones de accesibilidad. La transferencia y el acceso a objetos dentro y fuera de Amazon S3 puede utilizar un servicio, como Transfer Acceleration o Puntos de acceso, para respaldar su ubicación, sus necesidades de seguridad y sus patrones de acceso. |

| Tipo | Tecnología | Características clave |
| --- | --- | --- |
| Almacenamiento de archivos | de Amazon Glacier | Clase de almacenamiento de Amazon S3 desarrollada para el archivado de datos. |
| Sistema de archivos compartidos | Amazon Elastic File System (Amazon EFS) | Sistema de archivos montable al que pueden acceder varios tipos de soluciones de computaci ón. Amazon EFS aumenta y reduce automáticamente el almacenamiento y optimiza el rendimiento para ofrecer latencias bajas y consisten tes. |
| Sistema de archivos compartidos | Amazon FSx | Se basa en las últimas soluciones de computación de AWS para admitir cuatro sistemas de archivos de uso común: NetApp ONTAP, OpenZFS, Windows File Server y Lustre. La latencia, el rendimiento y las E/S por segundo de Amazon FSx varían según el sistema de archivos y deben tenerse en cuenta a la hora de seleccion ar el sistema de archivos adecuado para sus necesidad es de carga de trabajo. |

| Tipo | Tecnología | Características clave |
| --- | --- | --- |
| Almacenamiento en bloque | Amazon Elastic Block Store (Amazon EBS) | Servicio de almacenamiento en bloque de alto rendimiento, escalable y fácil de usar diseñado para Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS incluye almacenamiento respaldado por SSD para cargas de trabajo transaccionales y de IOPS intensivas, así como almacenamiento respaldado por HDD para cargas de trabajo de rendimiento intensivo. |
| Base de datos relacional | Amazon Aurora, Amazon RDS, Amazon Redshift | Se han diseñado para respaldar las transacciones ACID (atomicidad, coherencia, aislamiento, durabilidad) y mantener la integridad referencial y una fuerte coherencia de datos. Muchas aplicaciones tradicionales, la planificación de recursos empresariales (ERP), la administración de las relaciones con los clientes (CRM) y los sistemas de comercio electrónico utilizan bases de datos relacionales para almacenar sus datos. |

| Tipo | Tecnología | Características clave |
| --- | --- | --- |
| Base de datos de clave-valor | Amazon DynamoDB | Optimizada para patrones de acceso comunes, normalmente para almacenar y recuperar grandes volúmenes de datos. Las aplicaciones web con mucho tráfico, los sistemas de comercio electrónico y las aplicaciones de juegos son casos de uso típicos para las bases de datos de clave-valor. |

- Automatización de la asignación de almacenamiento: para los sistemas de almacenamiento que tienen un tamaño fijo, como Amazon EBS o Amazon FSx, supervise el espacio de almacenamiento disponible y automatice la asignación de almacenamiento al alcanzar un umbral. Puede usar Amazon CloudWatch para recopilar y analizar diferentes métricas para Amazon EBS y Amazon FSx.
- Elección de la clase de almacenamiento adecuada: elija la clase de almacenamiento adecuada para sus datos.
  - Las clases de almacenamiento de Amazon S3 se pueden configurar en el nivel de objeto. Un único bucket puede incluir objetos almacenados en todas las clases de almacenamiento.
  - Puede utilizar las políticas de ciclo de vida de Amazon S3 para hacer transiciones automáticas de objetos entre clases de almacenamiento o eliminar datos sin necesidad de hacer cambios en la aplicación. En general, tiene que equilibrar la eficiencia de los recursos, la latencia de acceso y la fiabilidad cuando considere estos mecanismos de almacenamiento.

Recursos

Documentos relacionados:

- Amazon EBS volume types
- Almacén de instancias de Amazon EC2
- Amazon S3 Intelligent-Tiering
- Amazon EBS I/O Characteristics

- Uso de las clases de almacenamiento de Amazon S3
- ¿Qué es Amazon Glacier?

Videos relacionados:

- AWS re:Invent 2023 - Improve Amazon EBS efficiency and be more cost-efficient
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- AWS re:Invent 2023 - Building and optimizing a data lake on Amazon S3
- AWS re:Invent 2022 - Building modern data architectures on AWS
- AWS re:Invent 2022 - Modernize apps with purpose-built databases
- AWS re:Invent 2022 - Building data mesh architectures on AWS
- AWS re:Invent 2023 - Deep dive into Amazon Aurora and its innovations
- AWS re:Invent 2023 - Advanced data modeling with Amazon DynamoDB

Ejemplos relacionados:

- Ejemplos de Amazon S3
- AWS Purpose Built Databases Workshop
- Databases for Developers
- AWS Modern Data Architecture Immersion Day
- Build a Data Mesh on AWS

SUS04-BP03 Uso de políticas para administrar el ciclo de vida de los conjuntos de datos

Administre el ciclo de vida de todos sus datos y aplique automáticamente la eliminación para minimizar el almacenamiento total necesario para su carga de trabajo.

Patrones comunes de uso no recomendados:

- Elimina los datos manualmente.
- No elimina ningún dato de su carga de trabajo.
- No traslada los datos a niveles de almacenamiento de mayor eficiencia energética en función de sus requisitos de retención y acceso.

Beneficios de establecer esta práctica recomendada: el uso de políticas de ciclo de vida de los datos garantiza el acceso y la conservación eficientes de los datos en una carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los conjuntos de datos suelen tener distintos requisitos de conservación y acceso durante su ciclo de vida. Por ejemplo, su aplicación puede necesitar acceso frecuente a algunos conjuntos de datos durante un periodo de tiempo limitado. Después, se accede a esos conjuntos de datos con poca frecuencia. Para mejorar la eficiencia del almacenamiento y la computación de datos a lo largo del tiempo, implemente políticas de ciclo de vida, que son reglas que definen cómo se manejan los datos a lo largo del tiempo.

Con las reglas de configuración del ciclo de vida, puede indicar al servicio de almacenamiento específico que haga la transición de un conjunto de datos a niveles de almacenamiento de mayor eficiencia energética, que lo archive o que lo elimine. Esta práctica minimiza el almacenamiento y la recuperación activos de datos, lo que se traduce en un menor consumo de energía. Además, prácticas como archivar o eliminar datos obsoletos respaldan el cumplimiento normativo y la gobernanza de datos.

Pasos para la implementación

- Utilice la clasificación de datos: clasifique los conjuntos de datos en su carga de trabajo.
- Defina reglas de administración: defina procedimientos de administración para cada clase de datos.
- Habilite la automatización: establezca políticas de ciclo de vida automatizadas para aplicar reglas de ciclo de vida. A continuación, se ofrecen algunos ejemplos de configuración de políticas automatizadas de ciclo de vida para distintos servicios de almacenamiento de AWS:

| Servicio de almacenamiento | Manera de establecer políticas de ciclo de vida automatizadas |
| --- | --- |
| Amazon S3 | Puede usar Amazon S3 Lifecycle para administrar los objetos a lo largo de su ciclo de vida. Si sus patrones de acceso son desconocidos, cambiantes o impredecibles, puede utilizar Amazon S3 Intelligent-Tiering, que supervisa los patrones de acceso y |

| Servicio de almacenamiento | Manera de establecer políticas de ciclo de vida automatizadas |
| --- | --- |
| | mueve automáticamente los objetos a los que no se ha accedido a niveles de acceso de menor costo. Puede aprovechar las métricas de Lente de almacenamiento de Amazon S3 para identificar las oportunidades de optimización y las brechas en la administración del ciclo de vida. |
| Amazon Elastic Block Store (EBS) | Puede utilizar Amazon Data Lifecycle Manager para automatizar la creación, conservación y eliminación de instantáneas de Amazon EBS y las AMI basadas en Amazon EBS. |
| Amazon Elastic File System | La administración del ciclo de vida de Amazon EFS administra automáticamente el almacenamiento de los archivos para sus sistemas de archivos. |
| Amazon Elastic Container Registry | Las políticas de ciclo de vida de Amazon ECR automatizan la limpieza de las imágenes de contenedor al hacer caducar las imágenes por antigüedad o cantidad. |
| AWS Elemental MediaStore | Puede utilizar una política de ciclo de vida de objetos que rija el tiempo que los objetos deben almacenarse en el contenedor de MediaStore. |

- Elimine los recursos que no utilice: elimine los volúmenes, las instantáneas y los datos no utilizados que estén fuera de su periodo de retención. Saque partido de las características nativas del servicio, como el tiempo de vida (TTL) de Amazon DynamoDB o la conservación de registros de Amazon CloudWatch para la eliminación.
- Agregue y comprima: agregue y comprima datos cuando proceda en función de las reglas de ciclo de vida.

Recursos

Documentos relacionados:

- Optimice las reglas de ciclo de vida de Amazon S3 con el análisis de clases de almacenamiento de Amazon S3
- Evaluating Resources with Reglas de AWS Config

Videos relacionados:

- AWS re:Invent 2021 - Amazon S3 Lifecycle best practices to optimize your storage spend
- AWS re:Invent 2023 - Optimizing storage price and performance with Amazon S3
- Simplify Your Data Lifecycle and Optimize Storage Costs With Amazon S3 Lifecycle
- Reduce Your Storage Costs Using Amazon S3 Storage Lens

SUS04-BP04 Uso de la elasticidad y la automatización para ampliar el almacenamiento de bloques o el sistema de archivos

Utilice la elasticidad y la automatización para ampliar el almacenamiento de bloques o el sistema de archivos a medida que crecen los datos para minimizar el almacenamiento total aprovisionado.

Patrones comunes de uso no recomendados:

- Adquiere un almacenamiento de bloques grande o un sistema de archivos de gran tamaño para necesidades futuras.
- Aprovisiona en exceso las operaciones de entrada y salida por segundo (IOPS) de su sistema de archivos.
- No supervisa el uso de sus volúmenes de datos.

Beneficios de establecer esta práctica recomendada: minimizar el exceso de aprovisionamiento del sistema de almacenamiento reduce los recursos inactivos y mejora la eficiencia general de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Cree almacenamiento de bloques y sistemas de archivos con una asignación de tamaño, rendimiento y latencia adecuados para su carga de trabajo. Utilice la elasticidad y la automatización para ampliar el almacenamiento de bloques o el sistema de archivos a medida que crecen los datos sin tener que aprovisionar en exceso estos servicios de almacenamiento.

Pasos para la implementación

- Para el almacenamiento de tamaño fijo, como Amazon EBS, asegúrese de supervisar la cantidad de almacenamiento utilizada en relación con el tamaño total del almacenamiento y cree una automatización, si es posible, para aumentar el tamaño de almacenamiento cuando se alcance un umbral.
- Use volúmenes elásticos y servicios administrados de datos en bloque para automatizar la asignación de almacenamiento adicional a medida que aumentan sus datos persistentes. Como ejemplo, puede usar volúmenes elásticos de Amazon EBS para cambiar el tamaño del volumen, el tipo de volumen o ajustar el rendimiento de sus volúmenes de Amazon EBS.
- Elija la clase de almacenamiento, el modo de rendimiento y el modo de caudal adecuados para que su sistema de archivos responda a su necesidad empresarial, sin excederse.
  - Amazon EFS performance
  - Amazon EBS volume performance on Linux instances
- Establezca niveles como objetivo de uso para los volúmenes de datos y ajuste el tamaño de los volúmenes que estén fuera de los intervalos esperados.
- Establezca el tamaño correcto de los volúmenes de solo lectura según los datos.
- Migre los datos a almacenes de objetos para evitar el aprovisionamiento del exceso de capacidad de los tamaños de volúmenes fijos en el almacenamiento en bloque.
- Revise periódicamente los volúmenes elásticos y los sistemas de archivos para terminar los volúmenes inactivos y reducir los recursos aprovisionados en exceso para ajustarlos al tamaño actual de los datos.

Recursos

Documentos relacionados:

- Extend the file system after resizing an EBS volume
- Modify a volume using Amazon EBS Elastic Volumes

- Documentación de Amazon FSx
- ¿Qué es Amazon Elastic File System?

Videos relacionados:

- Deep Dive on Amazon EBS Elastic Volumes
- Amazon EBS and Snapshot Optimization Strategies for Better Performance and Cost Savings
- Optimizing Amazon EFS for cost and performance, using best practices

SUS04-BP05 Eliminación de datos innecesarios o redundantes

Elimine datos innecesarios o redundantes para minimizar los recursos de almacenamiento necesarios para guardar sus conjuntos de datos.

Patrones comunes de uso no recomendados:

- Duplica datos que se pueden obtener o recrear fácilmente.
- Hace una copia de seguridad de todos los datos sin tener en cuenta su criticidad.
- Elimina solo datos de forma irregular, en eventos operativos o no los elimina en absoluto.
- Almacena datos de forma redundante independientemente de la durabilidad del servicio de almacenamiento.
- Activa el control de versiones de Amazon S3 sin ninguna justificación empresarial.

Beneficios de establecer esta práctica recomendada: la eliminación de los datos innecesarios reduce el tamaño de almacenamiento necesario para la carga de trabajo y el impacto medioambiental de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Al eliminar conjuntos de datos innecesarios y redundantes, puede reducir el coste de almacenamiento y la huella ambiental. Esta práctica también puede hacer que la computación sea más eficiente, ya que los recursos de computación solo procesan datos importantes en lugar de datos innecesarios. Automatice la eliminación de datos innecesarios. Use tecnologías que desdupliquen los datos en el nivel de archivo y de bloque. Utilice las características de los servicios para la replicación y la redundancia de datos nativos.

Pasos para la implementación

- Evalúe conjunto de datos públicos: evalúe si puede evitar el almacenamiento de datos mediante los conjuntos de datos existentes y disponibles públicamente en AWS Data Exchange y Datos abiertos en AWS.
- Desduplique datos: use mecanismos que puedan desduplicar los datos en los bloques y objetos. A continuación, se ofrecen algunos ejemplos de cómo desduplicar datos en AWS:

| Servicio de almacenamiento | Mecanismo de desduplicación |
| --- | --- |
| Amazon S3 | Use AWS Lake Formation FindMatches para buscar registros coincidentes en un conjunto de datos (incluidos los que no tienen identific adores) con la nueva transformación de ML FindMatches. |
| Amazon FSx | Use la desduplicación de datos en Amazon FSx para Windows. |
| Amazon Elastic Block Store snapshots | Las instantáneas son copias de seguridad incrementales, lo que significa que solo se guardan los bloques que han cambiado en el dispositivo después de la instantánea más reciente. |

- Uso de políticas de ciclo de vida: utilice políticas de ciclo de vida para eliminar automáticamente los activos no utilizados. Utilice las características de los servicios nativos, como Tiempo de vida de Amazon DynamoDB, Amazon S3 Lifecycle o la retención de registros de Amazon CloudWatch para la eliminación.
- Utilice la virtualización de datos: utilice las capacidades de virtualización de datos en AWS para mantener los datos en su origen y evitar la duplicación de datos.
  - Cloud Native Data Virtualization on AWS
  - Optimize Data Pattern Using Amazon Redshift Data Sharing
- Utilice copias de seguridad incrementales: use una tecnología de copia de seguridad que pueda crear copias incrementales.

- Utilice la durabilidad nativa: aproveche la durabilidad de Amazon S3 y la replicación de Amazon EBS para cumplir sus objetivos de durabilidad en lugar de utilizar tecnologías autogestionadas (como una matriz redundante de discos independientes [RAID]).
- Use registros eficientes: centralice los datos de registro y de seguimiento, desduplique las entradas de registro que sean idénticas y establezca mecanismos para ajustar los detalles cuando sea necesario.
- Utilice un almacenamiento en caché eficiente: Rellene las memorias caché previamente solo en casos justificados.
- Establezca la supervisión y la automatización de la memoria caché para ajustar el tamaño de esta en consonancia.
- Quite los recursos de versiones antiguas: quite las implementaciones y los recursos desfasados de los almacenes de objetos y las memorias caché periféricas al introducir nuevas versiones de su carga de trabajo.

Recursos

Documentos relacionados:

- Change log data retention in CloudWatch Logs
- Data deduplication on Amazon FSx for Windows File Server
- Features of Amazon FSx for ONTAP including data deduplication
- Invalidación de archivos en Amazon CloudFront
- Using AWS Backup to back up and restore Amazon EFS file systems
- What is Amazon CloudWatch Logs?
- Introducción a las copias de seguridad en Amazon RDS
- Integrate and deduplicate datasets using AWS Lake Formation

Videos relacionados:

- Amazon Redshift Data Sharing Use Cases

Ejemplos relacionados:

- ¿Cómo puedo utilizar Amazon Athena para analizar mis registros de acceso al servidor de Amazon S3?

### SUS04-BP06 Uso de sistemas de archivos o almacenamiento compartidos para acceder a datos comunes

Adopte sistemas de archivos o almacenamiento compartidos para evitar la duplicación de datos y posibilitar una infraestructura más eficiente para la carga de trabajo.

Patrones comunes de uso no recomendados:

- Aprovisiona almacenamiento para cada cliente.
- No desconecta el volumen de datos de los clientes inactivos.
- No proporciona acceso al almacenamiento a través de plataformas y sistemas.

Beneficios de establecer esta práctica recomendada: usar sistemas de archivos o almacenamiento compartidos permite compartir datos con uno o varios consumidores sin tener que copiarlos. De este modo, se reducen los recursos de almacenamiento necesarios para la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si tiene varios usuarios o aplicaciones que acceden a los mismos conjuntos de datos, el uso de la tecnología de almacenamiento compartido es esencial para utilizar una infraestructura eficiente para la carga de trabajo. La tecnología de almacenamiento compartido proporciona una ubicación central para almacenar y administrar conjuntos de datos y evitar la duplicación de datos. También refuerza la coherencia de los datos entre los distintos sistemas. Además, la tecnología de almacenamiento compartido permite un uso más eficaz de la potencia de computación, ya que varios recursos de computación pueden acceder a los datos y procesarlos simultáneamente en paralelo.

Obtenga datos de estos servicios de almacenamiento compartido solo cuando los necesite y desconecte los volúmenes que no utilice para liberar recursos.

Pasos para la implementación

- Use el almacenamiento compartido: migre los datos al almacenamiento compartido cuando tengan varios consumidores. A continuación le mostramos algunos ejemplos de tecnología de almacenamiento compartido en AWS:

| Opción de almacenamiento | Cuándo se debe usar |
| --- | --- |
| Amazon EBS Multi-Attach | Amazon EBS Multi-Attach le permite asociar un único volumen SSD de IOPS aprovisionadas (io1 o io2) a varias instancias que se encuentren en la misma zona de disponibilidad. |
| Amazon EFS | Consulte When to Choose Amazon EFS. |
| Amazon FSx | Consulte Elegir un sistema de archivos de Amazon FSx. |
| Amazon S3 | Las aplicaciones que no requieren una estructura de sistema de archivos y están diseñadas para colaborar con el almacenamiento de objetos pueden utilizar Amazon S3 como una solución de almacenamiento de objetos escalable de forma masiva, duradera y de bajo costo. |

- Obtenga datos según las necesidades: copie datos en sistemas de archivos compartidos, o recupérelos de ellos solo cuando sea necesario. Por ejemplo, puede crear un sistema de archivos de Amazon FSx para Lustre respaldado por Amazon S3 y cargar solo el subconjunto de datos necesario para procesar los trabajos en Amazon FSx.
- Elimine los datos que no necesite: elimine los datos según corresponda a sus patrones de uso, tal y como se describe en SUS04-BP03 Uso de políticas para administrar el ciclo de vida de los conjuntos de datos.
- Desconecte los volúmenes inactivos: desconecte los volúmenes de los clientes que no los estén usando de forma activa.

Recursos

Documentos relacionados:

- Linking your file system to an Amazon S3 bucket
- Using Amazon EFS for AWS Lambda in your serverless applications

- Amazon EFS Intelligent-Tiering Optimizes Costs for Workloads with Changing Access Patterns
- Using Amazon FSx with your on-premises data repository

Videos relacionados:

- Storage cost optimization with Amazon EFS
- AWS re:Invent 2023 - What's new with AWS file storage
- AWS re:Invent 2023 - File storage for builders and data scientists on Amazon Elastic File System

SUS04-BP07 Minimización del movimiento de datos entre redes

Utilice almacenamiento de objetos o sistemas de archivos compartidos para acceder a los datos comunes y minimizar el total de recursos de redes necesarios para admitir el movimiento de datos para su carga de trabajo.

Patrones comunes de uso no recomendados:

- Almacena todos los datos en la misma Región de AWS independientemente de dónde se encuentren los usuarios de los datos.
- No optimiza el tamaño ni el formato de los datos antes de moverlos por la red.

Beneficios de establecer esta práctica recomendada: la optimización del movimiento de datos por la red reduce los recursos de redes totales necesarios para la carga de trabajo y disminuye su impacto medioambiental.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

El movimiento de datos por la organización requiere recursos de computación, red y almacenamiento. Utilice técnicas para minimizar el movimiento de datos y mejorar la eficacia general de la carga de trabajo.

Pasos para la implementación

- Utilice la proximidad: tenga en cuenta la proximidad a los datos o a los usuarios como factor de decisión al seleccionar una región para la carga de trabajo.

- Particione los servicios: particione los servicios que se consumen regionalmente para que los datos específicos de una región se almacenen en la región en la que se consumen.
- Utilice formatos de archivo eficientes: utilice formatos de archivo eficientes (como Parquet u ORC) y comprima los datos antes de moverlos por la red.
- Minimice el movimiento de datos: no mueva los datos no utilizados. Algunos ejemplos que pueden ayudarle a evitar mover datos no utilizados:
  - Reduzca las respuestas de la API solo a los datos relevantes.
  - Agregue los datos cuando estén detallados (no se requiere información en el nivel de registro).
  - Consulte Well-Architected Lab - Optimize Data Pattern Using Amazon Redshift Data Sharing.
  - Tenga en cuenta el uso compartido de datos entre cuentas en AWS Lake Formation.
- Utilice servicios de la periferia: utilice servicios que puedan ayudarlo a ejecutar el código más cerca de los usuarios de la carga de trabajo.

| Servicio | Cuándo se debe usar |
| --- | --- |
| Lambda@Edge | Se usa para las operaciones que utilizan muchos recursos de computación que se ejecutan cuando los objetos no están en la memoria caché. |
| CloudFront Functions | Se usan en casos de uso sencillos como las manipulaciones de solicitudes o respuesta s HTTP(s) que pueden iniciarse mediante funciones de corta duración. |
| AWS IoT Greengrass | Ejecuta la computación local, la mensajería y el almacenamiento en caché de datos para los dispositivos conectados. |

Recursos

Documentos relacionados:

- Optimizing your AWS Infrastructure for Sustainability, Part III: Networking
- Infraestructura global de AWS

- Características clave de Amazon CloudFront, incluida la red periférica global de CloudFront
- Compresión de solicitudes HTTP en Amazon OpenSearch Service
- Compresión de datos intermedia con Amazon EMR
- Carga de archivos de datos comprimidos desde Amazon S3 en Amazon Redshift
- Distribución de archivos comprimidos con Amazon CloudFront

Videos relacionados:

- Demystifying data transfer on AWS

SUS04-BP08 Copias de seguridad de los datos solo cuando sea difícil volver a crearlos

Evite hacer copias de seguridad de datos que no tengan valor empresarial para minimizar los requisitos de recursos de almacenamiento para su carga de trabajo.

Patrones comunes de uso no recomendados:

- No dispone de una estrategia de copia de seguridad para los datos.
- Hace copias de seguridad de datos que pueden volver a crearse fácilmente.

Beneficios de establecer esta práctica recomendada: evitar hacer copias de seguridad de datos que no son críticos reduce los recursos de almacenamiento necesarios para la carga de trabajo y reduce su impacto medioambiental.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Evitar la copia de seguridad de datos innecesarios puede contribuir a reducir los costos y los recursos de almacenamiento utilizados por la carga de trabajo. Haga copias de seguridad únicamente de aquellos datos que tengan valor empresarial o que sean necesarios para satisfacer los requisitos de cumplimiento. Examine las políticas de copia de seguridad y excluya el almacenamiento efímero que no proporcione valor alguno en un escenario de recuperación.

Pasos para la implementación

- Clasifique los datos: implemente la política de clasificación de datos tal como se describe en SUS04-BP01 Implementación de una política de clasificación de datos.

- Diseñe una estrategia de copia de seguridad: utilice la importancia crítica de la clasificación de datos y diseñe una estrategia de copia de seguridad en función del objetivo de tiempo de recuperación (RTO) y objetivo de punto de recuperación (RPO). Evite hacer copias de seguridad de datos no esenciales.
  - Excluya los datos que puedan volver a crearse fácilmente.
  - Excluya los datos efímeros de sus copias de seguridad.
  - Excluya las copias locales de los datos, a menos que el tiempo necesario para restaurar esos datos desde una ubicación común supere lo establecido en los acuerdos de nivel de servicio (SLA).
- Utilice copias de seguridad automatizadas: utilice una solución automatizada o un servicio administrado para hacer copias de seguridad de los datos fundamentales para la empresa.
  - AWS Backup es un servicio totalmente administrado que facilita la centralización y automatización de la protección de datos en todos los servicios de AWS, en la nube y en las instalaciones. Para obtener orientación práctica sobre cómo crear copias de seguridad automatizadas mediante AWS Backup, consulte Well-Architected Labs - Testing Backup and Restore of Data.
  - Automatice las copias de seguridad y optimice los costos de las copias de seguridad de Amazon EFS con AWS Backup.

Recursos

Prácticas recomendadas relacionadas:

- REL09-BP01 Identificación de todos los datos de los que se debe hacer una copia de seguridad, creación de la copia de seguridad o reproducción de los datos a partir de los orígenes
- REL09-BP03 Copias de seguridad automáticas de los datos
- REL13-BP02 Uso de estrategias de recuperación definidas para cumplir los objetivos de recuperación

Documentos relacionados:

- Using AWS Backup to back up and restore Amazon EFS file systems
- Instantáneas de Amazon EBS
- Trabajo con copias de seguridad en Amazon Relational Database Service
- Socio de APN: socios que pueden ayudar con la copia de seguridad

- AWS Marketplace: productos que pueden usarse para la copia de seguridad
- Backing Up Amazon EFS
- Backing Up Amazon FSx for Windows File Server
- Copia de seguridad y restauración de Amazon ElastiCache (Redis OSS)

Videos relacionados:

- AWS re:Invent 2023 - Backup and disaster recovery strategies for increased resilience
- AWS re:Invent 2023 - What's new with AWS Backup
- AWS re:Invent 2021 - Backup, disaster recovery, and ransomware protection with AWS

## Hardware y servicios

Pregunta

- SUS 5 ¿Cómo selecciona y usa el hardware y los servicios en la nube de su arquitectura para lograr sus objetivos de sostenibilidad?

### SUS 5 ¿Cómo selecciona y usa el hardware y los servicios en la nube de su arquitectura para lograr sus objetivos de sostenibilidad?

Haga cambios en sus prácticas de administración de hardware como forma de reducir el impacto en la sostenibilidad de las cargas de trabajo. Minimice la cantidad de hardware necesario para aprovisionar e implementar y seleccione el hardware y los servicios más eficaces para su carga de trabajo individual.

Prácticas recomendadas

- SUS05-BP01 Uso de la mínima cantidad de hardware para satisfacer sus necesidades
- SUS05-BP02 Uso de los tipos de instancia con el menor impacto
- SUS05-BP03 Uso de servicios administrados
- SUS05-BP04 Optimización del uso de aceleradores de computación basados en hardware

SUS05-BP01 Uso de la mínima cantidad de hardware para satisfacer sus necesidades

Utilice la cantidad mínima de hardware para su carga de trabajo a fin de satisfacer eficazmente sus necesidades empresariales.

Patrones comunes de uso no recomendados:

- No supervisa el uso de los recursos.
- Tiene recursos con un bajo nivel de uso en su arquitectura.
- No revisa el uso del hardware estático para determinar si debe redimensionarse.
- No establece objetivos de uso de hardware para su infraestructura de computación en función de los KPI empresariales.

Beneficios de establecer esta práctica recomendada: redimensionar correctamente los recursos en la nube ayuda a reducir el impacto medioambiental de la carga de trabajo, a ahorrar dinero y a mantener los niveles de referencia de rendimiento.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Seleccione de forma óptima el número total de hardware necesario para su carga de trabajo con el fin de mejorar la eficacia global. La Nube de AWS ofrece la flexibilidad de ampliar o reducir sus recursos de forma dinámica a través de diversos mecanismos, como AWS Auto Scaling, para satisfacer los cambios en la demanda. También proporciona API y SDK que permiten modificar los recursos con el esfuerzo mínimo. Use estas capacidades para hacer cambios frecuentes en las implementaciones de la carga de trabajo. Además, utilice las directrices de dimensionamiento de las herramientas de AWS para usar eficazmente sus recursos en la nube y satisfacer sus necesidades empresariales.

Pasos para la implementación

- Elección del tipo de instancias: elija el tipo de instancia correcto que mejor se adapte a sus necesidades. Para obtener información sobre cómo elegir instancias de Amazon Elastic Compute Cloud y usar mecanismos como la selección de instancias basada en atributos, consulte lo siguiente:
  - ¿Cómo elijo el tipo de instancia de Amazon EC2 apropiado para mi carga de trabajo?
  - Seleccione el tipo de instancia basada en atributos para la flota de Amazon EC2.

    - Cree un grupo de escalado automático mediante la selección del tipo de instancia basada en atributos.
- Escalado: use pequeños incrementos para escalar cargas de trabajo variables.
- Uso de varias opciones de compra de computación: equilibre la flexibilidad, la escalabilidad y el ahorro de costos de las instancias con múltiples opciones de compra de computación.
    - Las instancias bajo demanda de Amazon EC2 son las más adecuadas para cargas de trabajo nuevas, con estado y con picos que no pueden ser flexibles en cuanto al tipo de instancia, la ubicación o el tiempo.
    - Las instancias de spot de Amazon EC2 son una excelente forma de complementar las demás opciones para aplicaciones flexibles y tolerantes a errores.
    - Aproveche los Savings Plans para computación para obtener cargas de trabajo estables que ofrezcan flexibilidad si cambian sus necesidades (como la zona de disponibilidad, la región y las familias o los tipos de instancias).
- Uso de la diversidad de instancias y zonas de disponibilidad: maximice la disponibilidad de las aplicaciones y aproveche el exceso de capacidad diversificando sus instancias y zonas de disponibilidad.
- Dimensionamiento correcto de las instancias: use las recomendaciones de tamaño adecuado de las herramientas de AWS para adaptar su carga de trabajo. Para obtener más información, consulte Optimización del costo con recomendaciones de redimensionamiento y Ajuste del tamaño: aprovisionamiento de instancias para adaptarse a las cargas de trabajo
    - Siga las recomendaciones de redimensionamiento de AWS Cost Explorer o AWS Compute Optimizer para identificar oportunidades de redimensionamiento.
- Negocio de acuerdos de nivel de servicio (SLA): negocie acuerdos de nivel de servicio (SLA) que permitan una reducción temporal de la capacidad mientras la automatización implementa recursos de reemplazo.

Recursos

Documentos relacionados:

- Optimizing your AWS Infrastructure for Sustainability, Part I: Compute
- Selección del tipo de instancia basada en atributos para el escalado automático de la Flota de Amazon EC2
- Documentación de AWS Compute Optimizer

- Operación de Lambda: optimización del rendimiento
- Documentación sobre el escalado automático

Videos relacionados:

- AWS re:Invent 2023 - What's new with Amazon EC2
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2022 - Optimizing Amazon Elastic Kubernetes Service for performance and cost on AWS
- AWS re:Invent 2023 - Sustainable compute: reducing costs and carbon emissions with AWS

SUS05-BP02 Uso de los tipos de instancia con el menor impacto

Supervise y utilice continuamente nuevos tipos de instancias para aprovechar las mejoras de la eficiencia energética.

Patrones comunes de uso no recomendados:

- Solo utiliza una familia de instancias.
- Solo utiliza instancias x86.
- Especifica un tipo de instancia en su configuración de Amazon EC2 Auto Scaling.
- Utiliza instancias de AWS para fines para los que no fueron diseñadas (por ejemplo, utiliza instancias optimizadas para la computación para una carga de trabajo que hace un uso intensivo de la memoria).
- No evalúa de forma regular nuevos tipos de instancia.
- No comprueba recomendaciones de herramientas de dimensionamiento de AWS como AWS Compute Optimizer.

Beneficios de establecer esta práctica recomendada: al utilizar instancias energéticamente eficientes y del tamaño adecuado, podrá reducir en gran medida el impacto medioambiental y el costo de su carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Usar instancias eficientes en la carga de trabajo en la nube es fundamental para reducir el uso de los recursos y mejorar la rentabilidad. Supervise de forma continuada el lanzamiento de nuevos tipos de instancia y aproveche las mejoras de la eficiencia energética; se incluyen los tipos de instancia diseñados para admitir cargas de trabajo específicas, como el entrenamiento y la inferencia en machine learning y la transcodificación de vídeo.

Pasos para la implementación

- Conocimiento y exploración de los tipos de instancia: descubra los tipos de instancia que pueden reducir el impacto medioambiental de su carga de trabajo.
  - Suscríbase a Novedades de AWS para mantenerse al día de las últimas tecnologías e instancias de AWS.
  - Conozca los diferentes tipos de instancias de AWS.
  - Conozca las instancias basadas en AWS Graviton que ofrecen el mejor rendimiento por vatio de consumo energético en Amazon EC2 viendo re:Invent 2020: Conocer en profundidad las instancias de Amazon EC2 con procesador AWS Graviton2 y Conocer en profundidad las instancias C7g de Amazon EC2 y AWS Graviton3.
- Uso de los tipos de instancia con el menor impacto: planifique y haga la transición de su carga de trabajo a los tipos de instancia con el menor impacto.
  - Defina un proceso para evaluar nuevas funciones o instancias para su carga de trabajo. Aproveche la agilidad de la nube para probar rápidamente cómo los nuevos tipos de instancia pueden mejorar la sostenibilidad medioambiental de su carga de trabajo. Utilice las métricas proxy para medir cuántos recursos necesita para completar una unidad de trabajo.
  - Si es posible, modifique su carga de trabajo para que funcione con diversas cantidades de vCPU y de memoria para sacar el máximo partido del tipo de instancia que haya elegido.
  - Considere la posibilidad de cambiar su carga de trabajo a instancias basadas en Graviton para mejorar la eficiencia del rendimiento de la carga de trabajo. Para obtener más información sobre cómo cambiar las cargas de trabajo a AWS Graviton, consulte AWS Graviton Fast Start y Consideraciones al trasladar cargas de trabajo a instancias de Amazon Elastic Compute Cloud basadas en AWS Graviton.
  - Considere seleccionar la opción de AWS Graviton al usar los servicios administrados de AWS.
  - Migre su carga de trabajo a las regiones que ofrezcan las instancias con menor impacto en la sostenibilidad y que sigan cumpliendo sus requisitos empresariales.

    - Para las cargas de trabajo de machine learning, utilice hardware personalizado específico para su carga de trabajo, como AWS Trainium, AWS Inferentia y Amazon EC2 DL1. AWS Inferentia, como las instancias Inf2, ofrecen hasta un 50 % más de rendimiento por vatio que las instancias de Amazon EC2 comparables.
    - Utilice el Recomendador de inferencias de Amazon SageMaker AI para ajustar el tamaño correcto del punto de conexión de inferencia de ML.
    - Para cargas de trabajo con picos (cargas de trabajo con requisitos poco frecuentes de capacidad adicional), utilice instancias de rendimiento ampliable.
    - Para las cargas de trabajo sin estado y tolerantes a errores, use instancias de spot de Amazon EC2 para incrementar el uso global de la nube y reducir el impacto en la sostenibilidad de los recursos no utilizados.
- Operación y optimización: opere y optimice la instancia de la carga de trabajo.
    - En el caso de las cargas de trabajo efímeras, evalúe las métricas de Amazon CloudWatch de la instancia, como `CPUUtilization`, para identificar si la instancia está inactiva o infrautilizada.
    - En las cargas de trabajo estables, consulte regularmente las herramientas de dimensionamiento de AWS, como AWS Compute Optimizer, para identificar oportunidades de optimizar y dimensionar las instancias de forma correcta. Para ver más ejemplos y recomendaciones, vea los siguientes laboratorios:
        - Laboratorio de Well-Architected: Recomendaciones de redimensionamiento
        - Laboratorio de Well-Architected: Redimensionamiento con Compute Optimizer
        - Laboratorio de Well-Architected: Optimizar los patrones de hardware y observar los KPI de sostenibilidad

Recursos

Documentos relacionados:

- Optimizing your AWS Infrastructure for Sustainability, Part I: Compute
- AWS Graviton
- Amazon EC2 DL1
- Flotas de reservas de capacidad de Amazon EC2
- Flota de spot de Amazon EC2
- Funciones: configuración de funciones de Lambda
- Selección de tipo de instancia basada en atributos para la flota de Amazon EC2

- Creación de aplicaciones sostenibles, eficientes y con optimización de costos en AWS
- Cómo ayuda el panel de sostenibilidad de Contino a los clientes a optimizar su huella de carbono

Videos relacionados:

- AWS re:Invent 2023 - AWS Graviton: The best price performance for your AWS workloads
- AWS re:Invent 2023 - New Amazon Elastic Compute Cloud generative AI capabilities in Consola de administración de AWS
- AWS re:Invent 2023 = What's new with Amazon Elastic Compute Cloud
- AWS re:Invent 2023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies
- AWS re:Invent 2021 - Deep dive into AWS Graviton3 and Amazon EC2 C7g instances
- AWS re:Invent 2022 - Build a cost-, energy-, and resource-efficient compute environment

Ejemplos relacionados:

- Solución: guía para optimizar las cargas de trabajo de aprendizaje profundo para la sostenibilidad en AWS

SUS05-BP03 Uso de servicios administrados

Utilice los servicios administrados para operar con mayor eficacia en la nube.

Patrones comunes de uso no recomendados:

- Utiliza instancias de Amazon EC2 con poco uso para ejecutar sus aplicaciones.
- Su equipo interno solo administra la carga de trabajo, sin tiempo para centrarse en la innovación o las simplificaciones.
- Implementa y mantiene tecnologías para tareas que pueden ejecutarse con mayor eficacia en servicios administrados.

Beneficios de establecer esta práctica recomendada:

- El uso de servicios administrados traslada la responsabilidad a AWS, que dispone de información sobre millones de clientes que puede ayudar a impulsar nuevas innovaciones y eficiencias.
- El servicio administrado distribuye el impacto medioambiental del servicio entre muchos usuarios gracias a los planos de control de varios principios.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los servicios administrados traspasan a AWS la responsabilidad de mantener un uso elevado y optimizar la sostenibilidad del hardware implementado. Los servicios administrados también eliminan la carga operativa y administrativa del mantenimiento de un servicio, lo que permite al equipo tener más tiempo para centrarse en la innovación.

Revise la carga de trabajo para identificar los componentes que se pueden reemplazar por servicios administrados de AWS. Por ejemplo, Amazon RDS, Amazon Redshift y Amazon ElastiCache proporcionan un servicio administrado de base de datos. Amazon Athena, Amazon EMR y Amazon OpenSearch Service proporcionan un servicio de análisis administrado.

Pasos para la implementación

1. Inventario de la carga de trabajo: haga un inventario de la carga de trabajo para servicios y componentes.
2. Identificación de los candidatos: evalúe e identifique los componentes que se pueden reemplazar por servicios administrados. A continuación, encontrará algunos ejemplos de cuándo podría plantearse el uso de un servicio administrado:

| Tarea | Qué usar en AWS |
| --- | --- |
| Alojamiento de una base de datos | Utilice instancias administradas de Amazon Relational Database Service (Amazon RDS) en lugar de mantener sus propias instancias de Amazon RDS en Amazon Elastic Compute Cloud (Amazon EC2). |
| Alojamiento de una carga de trabajo de contenedores | Use AWS Fargate, en vez de implementar su propia infraestructura de contenedores. |
| Alojamiento de aplicaciones web | Utilice AWS Amplify Hosting como un servicio de CI/CD y alojamiento totalmente administr ado para sitios web estáticos y aplicaciones web renderizadas en el servidor. |

3. Creación de un plan de migración: identifique las dependencias y cree un plan de migración. Actualice los manuales de procedimientos y las guías de estrategias según corresponda.
    - AWS Application Discovery Service recopila y presenta de modo automático la información detallada sobre el uso y las dependencias de aplicaciones para que pueda tomar decisiones más fundamentadas cuando planifique la migración
4. Pruebas: pruebe el servicio antes de migrar al servicio administrado.
5. Sustitución de los servicios autoalojados: utilice el plan de migración para sustituir los servicios autoalojados por servicios administrados.
6. Supervisión y ajuste: supervise continuamente el servicio una vez finalizada la migración para llevar a cabo los ajustes necesarios y optimizar el servicio.

Recursos

Documentos relacionados:

- Productos en la Nube de AWS
- Calculadora del costo total de propiedad (TCO) de AWS
- Amazon DocumentDB
- Amazon Elastic Kubernetes Service (EKS)
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Videos relacionados:

- AWS re:Invent 2021 - Cloud operations at scale with AWS Managed Services
- AWS re:Invent 2023 - Best practices for operating on AWS

SUS05-BP04 Optimización del uso de aceleradores de computación basados en hardware

Optimice el uso de instancias de computación acelerada para reducir las demandas de infraestructura física de la carga de trabajo.

Patrones comunes de uso no recomendados:

- No supervisa el uso de GPU.
- Utiliza una instancia de uso general para la carga de trabajo cuando una instancia personalizada podría ofrecer mayor rendimiento, menor costo y mejor rendimiento por vatio.

- Utiliza aceleradores de computación basados en hardware para tareas en las que es más eficiente utilizar alternativas basadas en CPU.

Beneficios de establecer esta práctica recomendada: al optimizar el uso de los aceleradores basados en hardware, puede reducir las demandas de infraestructura física de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Si necesita una gran capacidad de procesamiento, puede beneficiarse del uso de instancias de computación acelerada, que proporcionan acceso a aceleradores de computación basados en hardware, como unidades de procesamiento gráfico (GPU) y matrices de puertas programables en campo (FPGA). Estos aceleradores de hardware llevan a cabo ciertas funciones, como el procesamiento gráfico o la concordancia de patrones de datos, de forma más eficiente que las alternativas basadas en CPU. Muchas cargas de trabajo aceleradas, como el renderizado, la transcodificación y el machine learning, son muy variables en cuanto al uso de recursos. Ejecute este hardware solo durante el tiempo que sea necesario y retírelo mediante automatización cuando no se requiera para minimizar los recursos consumidos.

Pasos para la implementación

- Explore los aceleradores de la computación: identifique qué instancias de computación acelerada pueden satisfacer sus requisitos.
- Use hardware diseñado específicamente: para las cargas de trabajo de machine learning, utilice hardware diseñado específicamente para su carga de trabajo, como AWS Trainium, AWS Inferentia y Amazon EC2 DL1. Las instancias de AWS Inferentia, como las instancias Inf2, ofrecen hasta un 50 % más de rendimiento por vatio que las instancias de Amazon EC2 comparables.
- Monitoree las métricas de uso: recopile las métricas de uso de las instancias de computación acelerada. Por ejemplo, puede usar el agente de CloudWatch para recopilar métricas como `utilization_gpu` y `utilization_memory` para sus GPU, como se muestra en Recopilación de métricas de GPU NVIDIA con Amazon CloudWatch.
- Ajuste el tamaño: optimice el código, el funcionamiento de la red y la configuración de los aceleradores de hardware para asegurarse de que se aprovecha al máximo el hardware subyacente.
  - Optimización de las configuraciones de GPU
  - GPU Monitoring and Optimization in the Deep Learning AMI

  - Optimizing I/O for GPU performance tuning of deep learning training in Amazon SageMaker AI
- Manténgase al día: utilice las bibliotecas de alto rendimiento y los controladores de GPU más recientes.
- Libere las instancias innecesarias: use la automatización para liberar instancias de GPU cuando no se estén usando.

Recursos

Documentos relacionados:

- Computación acelerada
- Let's Architect! Architecting with custom chips and accelerators
- Cómo elijo el tipo de instancia de Amazon EC2 apropiado para mi carga de trabajo?
- Amazon EC2 VT1 Instances
- Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker AI

Videos relacionados:

- AWS re:Invent 2021 - How to select Amazon EC2 GPU instances for deep learning
- AWS Online Tech Talks - Deploying Cost-Effective Deep Learning Inference
- AWS re:Invent 2023 - Cutting-edge AI with AWS and NVIDIA
- AWS re:Invent 2022 - [NEW LAUNCH!] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances
- AWS re:Invent 2022 - Accelerate deep learning and innovate faster with AWS Trainium
- AWS re:Invent 2022 - Deep learning on AWS with NVIDIA: From training to deployment

## Proceso y cultura

Pregunta

- SUS 6 ¿Cómo respaldan sus procesos organizativos sus objetivos de sostenibilidad?

## SUS 6 ¿Cómo respaldan sus procesos organizativos sus objetivos de sostenibilidad?

Haga cambios en sus prácticas de desarrollo, prueba e implementación como forma de reducir el impacto en la sostenibilidad.

Prácticas recomendadas

- SUS06-BP01 Comunicación y ampliación en cascada de los objetivos de sostenibilidad
- SUS06-BP02 Adopción de métodos que permitan introducir mejoras en la sostenibilidad rápidamente
- SUS06-BP03 Mantenimiento de una carga de trabajo actualizada
- SUS06-BP04 Incremento del uso de los entornos de compilación
- SUS06-BP05 Uso de granjas de dispositivos administrados para pruebas

### SUS06-BP01 Comunicación y ampliación en cascada de los objetivos de sostenibilidad

La tecnología es un factor clave de la sostenibilidad. Los equipos de TI desempeñan un papel crucial a la hora de impulsar un cambio significativo hacia los objetivos de sostenibilidad de su organización. Estos equipos deben comprender con claridad los objetivos de sostenibilidad de la empresa y trabajar para comunicar esas prioridades e integrarlas en todas sus operaciones.

Patrones comunes de uso no recomendados:

- No conoce los objetivos de sostenibilidad de su organización ni cómo se aplican a su equipo.
- No tiene suficiente conocimiento ni formación sobre el impacto medioambiental de las cargas de trabajo en la nube.
- No sabe con certeza las áreas específicas a las que debe dar prioridad.
- No implica a sus empleados y clientes en las iniciativas de sostenibilidad.

Beneficios de establecer esta práctica recomendada: desde la optimización de la infraestructura y los sistemas hasta el uso de tecnologías innovadoras, los equipos de TI pueden reducir las emisiones de carbono de la organización y minimizar el consumo de recursos. La comunicación de los objetivos de sostenibilidad puede proporcionar a los equipos de TI la capacidad de mejorar y adaptarse continuamente a los desafíos de sostenibilidad que evolucionan. Además, estas optimizaciones sostenibles a menudo también se traducen en ahorros de costes, lo que refuerza el modelo de negocio.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Los principales objetivos de sostenibilidad de los equipos de TI deberían ser optimizar los sistemas y las soluciones para aumentar la eficiencia de los recursos y minimizar la huella de carbono de la organización y el impacto ambiental general. Los servicios e iniciativas compartidos, como los programas de formación y los paneles operativos, pueden ayudar a las organizaciones a optimizar las operaciones de TI y crear soluciones que pueden ayudar a reducir significativamente la huella de carbono. La nube ofrece una oportunidad no solo para transferir las responsabilidades de la infraestructura física y el aprovisionamiento de energía a una responsabilidad compartida del proveedor de la nube, sino también para optimizar continuamente la eficiencia de los recursos de los servicios basados en la nube.

Cuando los equipos utilizan el modelo de eficiencia inherente y responsabilidad compartida de la nube, pueden reducir significativamente el impacto ambiental de la organización. Esto, a su vez, puede contribuir a los objetivos generales de sostenibilidad de la organización y demostrar el valor de estos equipos como socios estratégicos en el camino hacia un futuro más sostenible.

Pasos para la implementación

- Defina metas y objetivos: establezca metas bien definidas para su programa de TI. Esto implica obtener la opinión de las partes interesadas responsables de diferentes departamentos, como TI, sostenibilidad y finanzas. Estos equipos deben definir objetivos medibles en consonancia con los objetivos de sostenibilidad de la organización, incluidas áreas como la reducción de emisiones de carbono y la optimización de los recursos.
- Conozca los límites de su empresa para cuantificar las emisiones de carbono: comprenda cómo los métodos de cuantificación del carbono, como el protocolo de gases de efecto invernadero (GEI), se relacionan con sus cargas de trabajo en la nube (para obtener más información, consulte Sostenibilidad de la nube).
- Utilice soluciones en la nube para la cuantificación del carbono: utilice soluciones en la nube, como las soluciones de cuantificación del carbono en AWS, para hacer un seguimiento de las emisiones de GEI de alcance uno, dos y tres en sus operaciones, carteras y cadenas de valor. Con estas soluciones, las organizaciones pueden agilizar la adquisición de datos sobre las emisiones de GEI, simplificar la elaboración de informes y obtener información que sirva de base para sus estrategias climáticas.
- Monitoree la huella de carbono de su cartera de TI: haga un seguimiento de las emisiones de carbono de sus sistemas de TI e informe al respecto. Utilice la herramienta de huella de carbono del cliente de AWS para rastrear, medir, revisar y pronosticar las emisiones de carbono generadas por el uso que hace de AWS.

- Comunique a sus equipos el uso de los recursos mediante métricas por proxy: realice un seguimiento del uso de los recursos e informe al respecto mediante métricas por proxy. En los modelos de precios bajo demanda de la nube, el uso de los recursos está relacionado con el coste, que es una métrica generalmente comprensible. Como mínimo, utilice el coste como una métrica por proxy para comunicar el uso de los recursos y las mejoras de cada equipo.
  - Habilite la subdivisión por horas en el explorador de costes y elabore un informe de costes y uso (CUR): el CUR proporciona información detallada de uso diario o por horas, tarifas, costes y atributos de uso de todos los servicios de AWS. Utilice los paneles de Cloud Intelligence y su panel de métricas por proxy de sostenibilidad como punto de partida para procesar y visualizar los datos basados en los costes y el uso. Para obtener más información, consulte lo siguiente:
  - Mida la eficiencia de la nube y haga un seguimiento de esta con métricas indirectas de sostenibilidad. Parte I: ¿Qué son las métricas intermedias?
  - Mida la eficiencia de la nube y haga un seguimiento de esta con métricas indirectas de sostenibilidad. Parte II: Establezca una cartera de métricas
- Optimice y evalúe continuamente: utilice un proceso de mejora para optimizar continuamente sus sistemas de TI, incluida la carga de trabajo en la nube, para lograr eficiencia y sostenibilidad. Monitoree la huella de carbono antes y después de implementar la estrategia de optimización. Utilice la reducción de la huella de carbono para evaluar la eficacia.
- Fomente una cultura de la sostenibilidad: utilice programas de formación (como AWSSkill Builder) para inculcar a sus empleados el valor de la sostenibilidad. Implíquelos en las iniciativas de sostenibilidad. Comparta y celebre sus historias de éxito. Utilice incentivos para premiarlos si alcanzan los objetivos de sostenibilidad.

Recursos

Documentos relacionados:

- Understanding your carbon emission estimations

Videos relacionados:

- AWS re:Invent 2023 - Accelerate data-driven circular economy initiatives with AWS
- AWS re:Invent 2023 - Sustainability innovation in AWS Global Infrastructure
- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures

- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure

Ejemplos relacionados:

- Well-Architected Lab - Turning cost & usage reports into efficiency reports

Formación relacionada:

- Sustainability Transformation on AWS
- SimuLearn - Sustainability Reporting
- Decarbonization with AWS

SUS06-BP02 Adopción de métodos que permitan introducir mejoras en la sostenibilidad rápidamente

Adopte métodos y procesos para validar las mejoras potenciales, minimizar los costos de las pruebas y ofrecer pequeñas mejoras.

Patrones comunes de uso no recomendados:

- La revisión de su solicitud de sostenibilidad es una tarea que se hace solo una vez al comienzo de un proyecto.
- Su carga de trabajo se ha quedado obsoleta, ya que el proceso de lanzamiento es demasiado complejo para incorporar pequeños cambios para la eficiencia de los recursos.
- No dispone de mecanismos para mejorar la carga de trabajo en materia de sostenibilidad.

Beneficios de establecer esta práctica recomendada: al establecer un proceso para introducir mejoras de sostenibilidad y hacer un seguimiento, podrá adoptar continuamente nuevas funciones y capacidades, eliminar problemas y mejorar la eficiencia de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: medio

Guía para la implementación

Pruebe y valide las mejoras de sostenibilidad potenciales antes de implementarlas en producción. Tenga en cuenta el costo de las pruebas al calcular las posibles ventajas futuras de una mejora. Desarrolle métodos de prueba de bajo costo para ofrecer pequeñas mejoras.

Pasos para la implementación

- Comprensión y comunicación de los objetivos de sostenibilidad de la organización: comprenda los objetivos de sostenibilidad de la organización, como la reducción de emisiones de carbono o la administración del agua. Convierta estos objetivos en requisitos de sostenibilidad para las cargas de trabajo en la nube. Comunique estos requisitos a las partes interesadas clave.
- Agregación de requisitos de sostenibilidad a las tareas pendientes: agregue requisitos para mejorar la sostenibilidad a las tareas pendientes de desarrollo.
- Iteración y mejora : utilice un proceso de mejora iterativo para identificar, evaluar, priorizar, probar e implementar las mejoras.
- Pruebas con un producto mínimo viable (MVP): desarrolle y pruebe posibles mejoras mediante los componentes representativos mínimos viables para reducir el costo y el impacto medioambiental de las pruebas.
- Agilización del proceso: mejore y optimice continuamente sus procesos de desarrollo. Por ejemplo, automatice su proceso de entrega de software mediante canalizaciones de integración y entrega continuas (CI/CD) para probar e implementar posibles mejoras con el fin de reducir el nivel de esfuerzo y limitar los errores provocados por los procesos manuales.
- Formación y concienciación: organice programas de formación para los miembros del equipo para formarlos en sostenibilidad y mostrarles cómo sus actividades afectan a los objetivos de sostenibilidad de la organización.
- Evaluación y ajuste: evalúe continuamente el impacto de las mejoras y haga los ajustes necesarios.

Recursos

Documentos relacionados:

- AWS hace posible las soluciones para la sostenibilidad

Videos relacionados:

- AWS re:Invent 2023 - Sustainable architecture: Past, present, and future
- AWS re:Invent 2022 - Delivering sustainable, high-performing architectures
- AWS re:Invent 2022 - Architecting sustainably and reducing your AWS carbon footprint
- AWS re:Invent 2022 - Sustainability in AWS global infrastructure

- AWS re:Invent 2023 - What's new with AWS observability and operations

SUS06-BP03 Mantenimiento de una carga de trabajo actualizada

Mantenga actualizada su carga de trabajo para adoptar características eficaces, eliminar problemas y mejorar la eficacia general de su carga de trabajo.

Patrones comunes de uso no recomendados:

- Asume que su arquitectura actual es estática y no se actualizará con el tiempo.
- No dispone de sistemas ni de una cadencia regular para evaluar si los programas y paquetes actualizados son compatibles con la carga de trabajo.

Beneficios de establecer esta práctica recomendada: al establecer un proceso para mantener la carga de trabajo actualizada, podrá adoptar nuevas características y capacidades, resolver problemas y mejorar la eficiencia de la carga de trabajo.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

La actualización de sistemas operativos, tiempos de ejecución, middlewares, bibliotecas y aplicaciones puede mejorar la eficacia de la carga de trabajo y facilitar la adopción de tecnologías más eficientes. Un software actualizado también puede incluir características que midan el impacto de la carga de trabajo en la sostenibilidad de forma más precisa, ya que los proveedores ofrecen características para cumplir sus propios objetivos de sostenibilidad. Adopte una cadencia periódica para mantener la carga de trabajo al día de las últimas características y versiones.

Pasos para la implementación

- Definición de un proceso: use un proceso y una programación para evaluar nuevas funciones o instancias de la carga de trabajo. Aproveche la agilidad de la nube para probar rápidamente cómo las nuevas funciones pueden mejorar su carga de trabajo para:
  - Reduzca el impacto en la sostenibilidad.
  - Logre la eficacia operativa.
  - Elimine las barreras para una mejora planificada.
  - Mejore su capacidad a la hora de medir y administrar las repercusiones en la sostenibilidad.

- Inventario: haga inventario del software y la arquitectura de la carga de trabajo e identifique los componentes que deben actualizarse.
  - Puede usar AWS Systems Manager Inventory para recopilar los metadatos del sistema operativo (SO), de las aplicaciones y de las instancias de Amazon EC2 y comprender rápidamente qué instancias están ejecutando el software y las configuraciones requeridas por su política de software, así como las instancias que deben actualizarse.
- Conocimiento del procedimiento de actualización: entienda cómo actualizar los componentes de la carga de trabajo.

| Componente de la carga de trabajo | Cómo actualizar |
| --- | --- |
| Imágenes de máquina | Use el Generador de imágenes de EC2 para gestionar las actualizaciones de las imágenes de máquina de Amazon (AMI) para Linux o las imágenes de Windows Server. |
| Imágenes de contenedor | Utilice Amazon Elastic Container Registry (Amazon ECR) con la canalización existente para gestionar imágenes de Amazon Elastic Container Service (Amazon ECS). |
| AWS Lambda | AWS Lambda incluye características de administración de versiones. |

- Uso de la automatización: automatice las actualizaciones para reducir el nivel de esfuerzo para implementar nuevas funciones y limitar los errores causados por los procesos manuales.
  - Puede utilizar CI/CD para actualizar automáticamente las AMI, las imágenes de contenedor y otros artefactos relacionados con la aplicación en la nube.
  - Puede utilizar herramientas como AWS Systems Manager Patch Manager para automatizar el proceso de actualizaciones del sistema y programar la actividad mediante Ventanas de mantenimiento de AWS Systems Manager.

Recursos

Documentos relacionados:

- Centro de arquitectura de AWS
- Novedades de AWS
- Herramientas para desarrolladores de AWS

Videos relacionados:

- AWS re:Invent 2022 - Optimize your AWS workloads with best-practice guidance
- All Things Patch: AWS Systems Manager

SUS06-BP04 Incremento del uso de los entornos de compilación

Aumente el uso de recursos para desarrollar, probar y compilar cargas de trabajo.

Patrones comunes de uso no recomendados:

- Aprovisiona o finaliza manualmente sus entornos de compilación.
- Mantiene sus entornos de compilación en funcionamiento independientemente de las actividades de prueba, compilación o lanzamiento (por ejemplo, ejecución de un entorno fuera del horario laboral de los miembros de su equipo de desarrollo).
- Aprovisiona en exceso los recursos para sus entornos de compilación.

Beneficios de establecer esta práctica recomendada: al aumentar el uso de los entornos de compilación, puede mejorar la eficiencia general de la carga de trabajo en la nube y, al mismo tiempo, asignar los recursos a los desarrolladores para que desarrollen, prueben y compilen de manera eficiente.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

Use la automatización y la infraestructura como código para incorporar los entornos de compilación cuando sea necesario y retirarlos cuando no se utilicen. Un patrón común consiste en programar periodos de disponibilidad que coincidan con las horas de trabajo de los miembros del equipo de desarrollo. Los entornos de prueba deben parecerse mucho a la configuración de producción. Aun así, busque oportunidades para utilizar tipos de instancia con capacidad de ampliación, instancias de spot de Amazon EC2, servicios de base de datos de escalamiento automático, contenedores y tecnologías sin servidor para coordinar el desarrollo y la capacidad de prueba con el uso. Limite el

volumen de datos para cumplir únicamente los requisitos de prueba. Si utiliza datos de producción en las pruebas, estudie las posibilidades de compartir los datos de producción y no trasladarlos.

Pasos para la implementación

- Uso de la infraestructura como código: utilice la infraestructura como código para aprovisionar los entornos de compilación.
- Uso de la automatización: use la automatización para administrar el ciclo de vida de los entornos de desarrollo y pruebas y maximizar la eficiencia de los recursos de compilación.
- Maximización del uso: utilice estrategias para maximizar el uso de los entornos de desarrollo y prueba.
  - Use el mínimo viable de entornos representativos para desarrollar y probar mejoras potenciales.
  - Utilice tecnologías sin servidor si es posible.
  - Use instancias bajo demanda para complementar los dispositivos de desarrollador.
  - Use tipos de instancia con capacidad de ampliación, instancias de spot y otras tecnologías para alinear la capacidad de creación con el uso.
  - Adopte servicios nativos en la nube para obtener un acceso seguro al intérprete de comandos de instancias en lugar de implementar flotas de hosts bastión.
  - Escale automáticamente sus recursos de compilación en función de sus tareas de compilación.

Recursos

Documentos relacionados:

- Administrador de sesiones de AWS Systems Manager
- Instancias de rendimiento ampliable de Amazon EC2
- ¿Qué es AWS CloudFormation?
- ¿Qué es AWS CodeBuild?
- Programador de instancias de AWS

Videos relacionados:

- AWS re:Invent 2023 - Continuous integration and delivery for AWS

SUS06-BP05 Uso de granjas de dispositivos administrados para pruebas

Utilice granjas de dispositivos administrados para probar eficazmente una nueva característica en un conjunto representativo de hardware.

Patrones comunes de uso no recomendados:

- Prueba e implementa manualmente su aplicación en dispositivos físicos individuales.
- No utiliza el servicio de pruebas de aplicaciones para probar e interactuar con sus aplicaciones (por ejemplo, Android, iOS y aplicaciones web) en dispositivos físicos reales.

Beneficios de establecer esta práctica recomendada: el uso de granjas de dispositivos administrados para probar aplicaciones preparadas para la nube ofrece una serie de ventajas:

- Incluyen características más eficaces para probar la aplicación en una amplia gama de dispositivos.
- Eliminan la necesidad de una infraestructura interna para las pruebas.
- Ofrecen diversos tipos de dispositivos, incluido el hardware más antiguo y menos popular, lo que elimina la necesidad de actualizaciones innecesarias de los dispositivos.

Nivel de riesgo expuesto si no se establece esta práctica recomendada: bajo

Guía para la implementación

El uso de granjas de dispositivos administrados puede ayudarlo a agilizar el proceso de prueba de nuevas características en un conjunto representativo de hardware. Las granjas de dispositivos administrados ofrecen diversos tipos de dispositivos, incluido el hardware más antiguo y menos popular, y evitan el impacto en la sostenibilidad para el cliente que tienen las actualizaciones innecesarias de dispositivos.

Pasos para la implementación

- Definición de los requisitos de prueba: defina los requisitos y el plan de pruebas (como el tipo de prueba, los sistemas operativos y el calendario de pruebas).
  - Puede usar Amazon CloudWatch RUM para recopilar y analizar datos del cliente y configurar su plan de pruebas.
- Selección de una granja de dispositivos administrada: seleccione una granja de dispositivos administrada que pueda cumplir con sus requisitos de prueba. Por ejemplo, puede usar AWS

Device Farm para probar y comprender el impacto de los cambios en un conjunto representativo de hardware.

- Uso de la automatización: utilice la integración continua/implementación continua (CI/CD) para programar y ejecutar las pruebas.
  - Integración de AWS Device Farm con la canalización de CI/CD para ejecutar pruebas de Selenium en varios navegadores
  - Creación y prueba de aplicaciones iOS y iPadOS con DevOps de AWS y servicios móviles
- Revisión y ajuste: revise continuamente los resultados de las pruebas y efectúe las mejoras necesarias.

Recursos

Documentos relacionados:

- Lista de dispositivos de AWS Device Farm
- Visualización del panel de CloudWatch RUM

Videos relacionados:

- AWS re:Invent 2023 - Improve your mobile and web app quality using AWS Device Farm
- AWS re:Invent 2021 - Optimize applications through end user insights with Amazon CloudWatch RUM

Ejemplos relacionados:

- Ejemplo de aplicación de AWS Device Farm para Android
- Ejemplo de aplicación de AWS Device Farm para iOS
- Pruebas web de Appium para AWS Device Farm

# Avisos

Es responsabilidad de los clientes realizar su propia evaluación independiente de la información que contiene este documento. El presente documento: (a) tiene sólo fines informativos, (b) representa las ofertas y prácticas actuales de los productos de AWS, que están sujetas a cambios sin previo aviso, y (c) no supone ningún compromiso ni garantía por parte de AWS y sus filiales, proveedores o licenciantes. Los productos o servicios de AWS se proporcionan “tal cual”, sin garantías, afirmaciones ni condiciones de ningún tipo, ya sean expresas o implícitas. Las responsabilidades y obligaciones de AWS con respecto a sus clientes se controlan mediante los acuerdos de AWS y este documento no forma parte ni modifica ningún acuerdo entre AWS y sus clientes.

Copyright © 2024 Amazon Web Services, Inc. o sus filiales.

# Glosario de AWS

Para ver la terminología más reciente de AWS, consulte el Glosario de AWS en la Referencia de Glosario de AWS.