

Marco de AWS Well-Architected

Pilar de sostenibilidad



Pilar de sostenibilidad: Marco de AWS Well-Architected

Copyright © 2024 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	i
Introducción	1
Sostenibilidad en la nube	3
El Modelo de responsabilidad compartida	4
Sostenibilidad de la nube	5
Sostenibilidad en la nube	5
Sostenibilidad a través de la nube	5
Principios de diseño para la sostenibilidad en la nube	6
Proceso de mejora	8
Escenario de ejemplo	9
Identificación de los objetivos de mejora	9
Recursos	10
Evaluación de las mejoras específicas	10
Métricas proxy	10
Métricas empresariales	11
Indicadores clave de rendimiento	11
Estimación de la mejora	12
Evaluación de las mejoras	12
Priorización y planificación de las mejoras	14
Prueba y validación de las mejoras	15
Implementación de cambios en producción	16
Medición de los resultados y replicación de los éxitos	16
Sostenibilidad como requisito no funcional	19
Prácticas recomendadas para la sostenibilidad en la nube	21
Selección de región	21
SUS01-BP01 Selección de la región en función de los requisitos empresariales y los objetivos de sostenibilidad	21
Alineación con la demanda	23
SUS02-BP01 Escalado de la infraestructura de la carga de trabajo dinámicamente	24
SUS02-BP02 Alineación de los SLA con los objetivos de sostenibilidad	27
SUS02-BP03 Detención de la creación y el mantenimiento de los recursos no utilizados	30
SUS02-BP04 Optimización de la ubicación geográfica de las cargas de trabajo en función de sus requisitos de red	31

SUS02-BP05 Optimización de los recursos de los miembros del equipo para las actividades efectuadas	35
SUS02-BP06 Implementación del almacenamiento en búfer o la limitación para aplanar la curva de demanda	36
Software y arquitectura	39
SUS03-BP01 Optimización del software y la arquitectura para los trabajos asíncronos y programados	40
SUS03-BP02 Eliminación o refactorización de los componentes de cargas de trabajo con uso reducido o nulo	43
SUS03-BP03 Optimización de las áreas de código que consumen la mayor parte del tiempo o de los recursos	45
SUS03-BP04 Optimización del impacto en los dispositivos y equipos	47
SUS03-BP05 Uso de los patrones de software y las arquitecturas que mejor respaldan los patrones de almacenamiento y el acceso a los datos	50
Administración de datos	52
SUS04-BP01 Implementación de una política de clasificación de datos	53
SUS04-BP02 Uso de tecnologías que admiten patrones de almacenamiento y acceso a los datos	55
SUS04-BP03 Uso de políticas para administrar el ciclo de vida de los conjuntos de datos	60
SUS04-BP04 Uso de la elasticidad y la automatización para ampliar el almacenamiento de bloques o el sistema de archivos	62
SUS04-BP05 Eliminación de datos innecesarios o redundantes	64
SUS04-BP06 Uso de sistemas de archivos o almacenamiento compartidos para acceder a datos comunes	67
SUS04-BP07 Minimización del movimiento de datos entre redes	69
SUS04-BP08 Copias de seguridad de los datos solo cuando sea difícil volver a crearlos	71
Hardware y servicios	73
SUS05-BP01 Uso de la mínima cantidad de hardware para satisfacer sus necesidades	74
SUS05-BP02 Uso de los tipos de instancia con el menor impacto	76
SUS05-BP03 Uso de servicios administrados	79
SUS05-BP04 Optimización del uso de aceleradores de computación basados en hardware	82
Proceso y cultura	84
SUS06-BP01 Adopción de métodos que permitan introducir mejoras en la sostenibilidad rápidamente	84
SUS06-BP02 Mantenimiento de una carga de trabajo actualizada	86

SUS06-BP03 Incremento del uso de los entornos de compilación	88
SUS06-BP04 Uso de granjas de dispositivos administrados para pruebas	90
Conclusión	93
Colaboradores	94
Documentación adicional	95
Revisiones del documento	96
Avisos	98
Glosario de AWS	99

Pilar de sostenibilidad: Marco de AWS Well-Architected

Fecha de publicación: 27 de junio de 2024 ([Revisiones del documento](#))

Este documento técnico se centra en el pilar de sostenibilidad del Marco de Amazon Web Services (AWS) Well-Architected. En él se incluyen principios de diseño, orientación operativa, prácticas recomendadas, posibles ventajas y desventajas, y planes de mejora que puede usar para lograr los objetivos de sostenibilidad de las cargas de trabajo de AWS.

Introducción

El Marco de AWS Well-Architected le ayuda a comprender las ventajas y desventajas de las decisiones que toma al crear cargas de trabajo 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. El marco ofrece una forma de medir sus arquitecturas de forma constante en función de las prácticas recomendadas de arquitectura y de identificar áreas de mejora. Si cuenta con cargas de trabajo con una buena arquitectura, aumentará la capacidad de respaldar los resultados empresariales.

El marco se basa en seis pilares:

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

El presente documento se centra en el pilar de sostenibilidad y, dentro del ámbito de sostenibilidad, en la sostenibilidad medioambiental. Está destinado a aquellos que ocupan puestos en tecnología, como los directores de tecnología (CTO), arquitectos, desarrolladores y miembros del equipo de operaciones.

Después de leer este documento, comprenderá mejor las recomendaciones y estrategias actuales de AWS que puede utilizar cuando diseñe arquitecturas en la nube teniendo en cuenta la

sostenibilidad. Al adoptar las prácticas que se indican en este documento, podrá crear arquitecturas que aumentarán al máximo la eficiencia y reducirán el desperdicio.

Sostenibilidad en la nube

La sostenibilidad como disciplina aborda el impacto medioambiental, económico y social a largo plazo de sus actividades empresariales. La [Comisión Mundial sobre el Medio Ambiente y el Desarrollo de las Naciones Unidas](#) define el desarrollo sostenible como “aquel que permite satisfacer las necesidades del presente sin comprometer la habilidad de las futuras generaciones de satisfacer sus necesidades propias”. Su organización o negocio puede tener repercusiones negativas en el medioambiente, como emisiones de carbono directas o indirectas, residuos no reciclables y daños a recursos compartidos, como el agua no contaminada.

Cuando se crean cargas de trabajo en la nube, la sostenibilidad consiste en 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. Este documento 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.

Al centrarse en los impactos ambientales, debe comprender cómo se suelen contabilizar estos impactos y los impactos posteriores en la contabilidad de emisiones de su organización. El [Protocolo de gases de efecto invernadero](#) organiza las emisiones de carbono en los siguientes ámbitos, junto con ejemplos de emisiones pertinentes dentro de cada ámbito para un proveedor de servicios en la nube, como AWS:

- **Ámbito 1:** todas las emisiones directas de las actividades de una organización o bajo su control. Por ejemplo, la quema de combustible por parte de los generadores de copias de seguridad de los centros de datos.
- **Ámbito 2:** emisiones indirectas de la electricidad comprada y utilizada para alimentar centros de datos y otras instalaciones. Por ejemplo, las emisiones de la generación de energía comercial.
- **Ámbito 3:** todas las demás emisiones indirectas de las actividades de una organización procedentes de fuentes que no controla. Algunos ejemplos de AWS incluyen las emisiones relacionadas con la construcción de centros de datos y la fabricación y el transporte del hardware de TI instalado en los centros de datos.

Desde la perspectiva del cliente de AWS, las emisiones de las cargas de trabajo que se ejecutan en AWS se contabilizan como emisiones indirectas y forman parte de las emisiones de ámbito 3. Cada carga de trabajo implementada genera una fracción de las emisiones totales de AWS de

cada uno de los ámbitos anteriores. La cantidad real varía según la carga de trabajo y depende de varios factores, como los servicios de AWS utilizados, la energía consumida por esos servicios, la intensidad de carbono de las redes eléctricas que dan servicio a los centros de datos de AWS en los que se ejecutan y la adquisición de energía renovable de AWS.

En primer lugar, en este documento se describe un modelo de responsabilidad compartida para la sostenibilidad ambiental y, a continuación, se proporcionan las prácticas recomendadas de arquitectura para minimizar el impacto de sus cargas de trabajo al reducir los recursos totales necesarios para que funcionen en los centros de datos de AWS.

El Modelo de responsabilidad compartida

La sostenibilidad ambiental es una responsabilidad compartida entre clientes y AWS.

- AWS es responsable de optimizar la sostenibilidad de la nube: proporcionar infraestructuras eficientes y compartidas, administrar el agua y suministrar energía renovable.
- Los clientes son responsables de la sostenibilidad en la nube: optimizar las cargas de trabajo y la utilización de los recursos y minimizar el total de recursos necesarios para implementar sus cargas de trabajo.



Modelo de responsabilidad compartida

Sostenibilidad de la nube

Los proveedores de servicios en la nube tienen una huella de carbono más baja y son más eficientes desde el punto de vista energético que las alternativas en las instalaciones habituales, ya que invierten en tecnologías eficientes de alimentación y refrigeración, gestionan poblaciones de servidores eficientes desde el punto de vista energético y logran altas tasas de utilización de los servidores. Las cargas de trabajo en la nube reducen el impacto al aprovechar los recursos compartidos, como las redes, la alimentación, la refrigeración y las instalaciones físicas. Puede migrar sus cargas de trabajo en la nube a tecnologías más eficientes a medida que estén disponibles y utilizar los servicios basados en la nube para transformar sus cargas de trabajo y lograr una mayor sostenibilidad.

Recursos

- [The Carbon Reduction Opportunity of Moving to Amazon Web Services](#)
- [AWS hace posible las soluciones para la sostenibilidad](#)

Sostenibilidad en la nube

La sostenibilidad en la nube es un esfuerzo continuo que se centra principalmente en la reducción de la energía y la eficiencia en todos los componentes de una carga de trabajo, obteniendo el máximo beneficio de los recursos provisionados 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.

Sostenibilidad a través de la nube

Además de minimizar el impacto de las cargas de trabajo que ha implementado, puede utilizar la Nube de AWS para ejecutar cargas de trabajo diseñadas para respaldar sus desafíos de sostenibilidad más amplios. Algunos ejemplos de estos desafíos incluyen reducir las emisiones de carbono, reducir el consumo de energía, reciclar el agua o reducir los residuos en otras áreas de su empresa u organización.

La sostenibilidad a través de la nube es cuando se utiliza la tecnología de AWS para resolver un desafío de sostenibilidad más amplio. Por ejemplo, puede utilizar un servicio de machine learning

como [Amazon Monitron](#) para detectar comportamientos anormales en la maquinaria industrial. Con estos datos de detección, puede hacer un mantenimiento preventivo para reducir el riesgo de incidentes ambientales causados por fallas inesperadas en los equipos y garantizar que la maquinaria siga funcionando con la máxima eficiencia.

Principios de diseño para la sostenibilidad en la nube

Aplique estos principios de diseño al diseñar la arquitectura de sus cargas de trabajo en la nube para maximizar la sostenibilidad y minimizar el impacto.

- **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 sus 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, elimine 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 o elimine 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.

Proceso de mejora

El proceso de mejora de la arquitectura incluye entender qué tiene y qué puede hacer para mejorar, para lo que debe seleccionar objetivos de mejora, probar las mejoras, adoptar las mejoras de éxito, cuantificar el éxito y compartir lo aprendido para poder replicarlo en otras partes y así, después, repetir el ciclo.

Los objetivos de sus mejoras pueden ser los siguientes:

- Eliminar los recursos desaprovechados, inactivos, no utilizados o con una baja utilización
- Maximizar el valor de los recursos que consume

Note

Utilice todos los recursos que aprovisione y complete el mismo trabajo con el mínimo de recursos posible.

En las etapas tempranas de la optimización, elimine primero las áreas desaprovechadas o con una baja utilización y, después, pase a optimizaciones más específicas que se ajusten a su carga de trabajo concreta.

Supervise los cambios en el consumo de los recursos a lo largo del tiempo. Identifique si la acumulación de cambios da como resultado un aumento ineficaz o significativo en el consumo de los recursos. Determine la necesidad de mejoras para abordar los cambios en el consumo e implementar las mejoras que establezca como prioritarias.

Los siguientes pasos están diseñados para ser un proceso iterativo que evalúa, prioriza, prueba e implementa mejoras centradas en la sostenibilidad para las cargas de trabajo en la nube.

1. Identificación de los objetivos de mejora: revise las cargas de trabajo en relación con las prácticas recomendadas de sostenibilidad que se identifican en este documento e identifique los objetivos de mejora.
2. Evaluación de las mejoras específicas: evalúe los cambios específicos para determinar las posibles mejoras, los costos proyectados y el riesgo empresarial.

3. Priorización y planificación de las mejoras: priorice los cambios que ofrezcan las mejoras más importantes con el menor costo y riesgo y establezca un plan para las pruebas y la implementación.
4. Prueba y validación de las mejoras: implemente cambios en los entornos de prueba para validar su potencial de mejora.
5. Implementación de cambios en producción: implemente cambios en todos los entornos de producción.
6. Medición de los resultados y repetición de los éxitos: busque oportunidades para replicar los éxitos en todas las cargas de trabajo y revertir los cambios con resultados inaceptables.

Escenario de ejemplo

Más adelante en este documento se hace referencia al siguiente escenario de ejemplo para ilustrar cada paso del proceso de mejora.

Su empresa tiene una carga de trabajo que manipula imágenes complejas en las instancias de Amazon EC2 y almacena los archivos modificados y originales para que los usuarios puedan acceder a ellos. Las actividades de procesamiento requieren un uso intensivo de la CPU y los archivos de salida son extremadamente grandes.

Identificación de los objetivos de mejora

Comprenda las prácticas recomendadas que le pueden ser útiles para lograr sus objetivos de sostenibilidad. Encontrará descripciones detalladas de estas [prácticas recomendadas](#) y recomendaciones de mejora más adelante en este documento.

Revise sus cargas de trabajo y los recursos utilizados. Identifique los puntos de acceso, como las grandes implementaciones y los recursos de uso frecuente. Evalúe estos puntos de acceso para ver si hay oportunidades de mejorar la utilización eficaz de sus recursos y reducir el total de recursos necesarios para lograr sus resultados empresariales.

Revise su carga de trabajo en relación con las prácticas recomendadas e identifique los elementos susceptibles de mejora.

Al aplicar este paso al [Escenario de ejemplo](#), identificará las siguientes prácticas recomendadas como posibles objetivos de mejora:

- Uso de la mínima cantidad de hardware para cumplir sus necesidades

- Uso de tecnologías que mejor admitan patrones de almacenamiento y acceso a los datos

Recursos

- [Optimizing your AWS Infrastructure for Sustainability, Part I: Compute](#)
- [Optimizing your AWS Infrastructure for Sustainability, Part II: Storage](#)
- [Optimizing your AWS Infrastructure for Sustainability, Part III: Networking](#)

Evaluación de las mejoras específicas

Conozca los recursos que aprovisiona su carga de trabajo para completar una unidad de trabajo. Evalúe las posibles mejoras y calcule su impacto potencial, el costo de implementación y los riesgos asociados.

Para medir las mejoras a lo largo del tiempo, primero comprenda lo que ha aprovisionado en AWS y cómo se consumen esos recursos.

Comience con una visión general completa de su uso de AWS y utilice los informes de costos y uso de AWS para ayudar a identificar los puntos de acceso. Utilice este [código de ejemplo de AWS](#) para revisar y analizar su informe con la ayuda de Amazon Athena.

Métricas proxy

Al evaluar cambios específicos, también debe evaluar qué métricas cuantifican mejor el efecto de ese cambio en el recurso asociado. Estas métricas se denominan métricas proxy. Seleccione las métricas proxy que mejor reflejen el tipo de mejora que está evaluando y los recursos a los que apunta la mejora. Estas métricas pueden evolucionar con el tiempo.

Los recursos aprovisionados para respaldar su carga de trabajo incluyen computación, almacenamiento y redes. Evalúe los recursos aprovisionados mediante sus métricas proxy para ver cómo se consumen esos recursos.

Use las métricas proxy para medir los recursos aprovisionados para lograr resultados empresariales.

Resource	Ejemplo de métricas proxy	Objetivos de mejora
Cálculo	Minutos de vCPU	Maximizar la utilización de los recursos aprovisionados

Resource	Ejemplo de métricas proxy	Objetivos de mejora
Almacenamiento	GB aprovisionados	Reducir el total aprovisionado
Network	GB transferidos o paquetes transferidos	Reducir la distancia transferida y la distancia total transferida

Métricas empresariales

Seleccione métricas empresariales para cuantificar el logro de los resultados empresariales. Las métricas empresariales deben reflejar el valor que aporta su carga de trabajo, por ejemplo, el número de usuarios activos simultáneos, las llamadas a la API atendidas o el número de transacciones completadas. Estas métricas pueden evolucionar con el tiempo. Tenga cuidado al evaluar las métricas empresariales de base financiera, ya que la incoherencia en el valor de las transacciones invalida las comparaciones.

Indicadores clave de rendimiento

Con la siguiente fórmula, divida los recursos aprovisionados entre los resultados empresariales obtenidos para determinar los recursos aprovisionados por unidad de trabajo.

$$\text{Recursos aprovisionados por unidad de trabajo} = \frac{\text{Métrica de proxy para recurso aprovisionado}}{\text{Métrica empresarial para resultados}}$$

Fórmula de KPI

Utilice sus recursos por unidad de trabajo como KPI. Establezca bases de referencia basadas en los recursos aprovisionados como base para las comparaciones.

Resource	Ejemplos de KPI	Objetivos de mejora
Cálculo	Minutos de vCPU por transacción	Maximizar la utilización de los recursos aprovisionados
Almacenamiento	GB por transacción	Reducir el total aprovisionado

Resource	Ejemplos de KPI	Objetivos de mejora
Network	GB transferidos por transacción o paquetes transferidos por transacción	Reducir la distancia transferida y la distancia total transferida

Estimación de la mejora

Estime la mejora como la reducción cuantitativa de los recursos aprovisionados (según lo indican las métricas proxy) y el cambio porcentual con respecto a los recursos aprovisionados de base de referencia por unidad de trabajo.

Resource	Ejemplos de KPI	Objetivos de mejora
Cálculo	% de reducción de minutos de vCPU por transacción	Maximizar el uso
Almacenamiento	% de reducción en GB por transacción	Reducir el total aprovisionado
Network	% de reducción de GB transferidos por transacción o de paquetes transferidos por transacción	Reducir la distancia transferida y la distancia total transferida

Evaluación de las mejoras

Evalúe las posibles mejoras según el beneficio neto previsto. Evalúe el tiempo, el costo y el nivel de esfuerzo necesarios para implementarlas y mantenerlas, así como los riesgos empresariales, como los impactos imprevistos.

Las mejoras específicas suelen representar compensaciones entre los tipos de recursos consumidos. Por ejemplo, para reducir el consumo de computación, puede almacenar un resultado o, para limitar la transferencia de datos, puede procesar los datos antes de enviar el resultado a un cliente. Estas [compensaciones](#) se analizarán con más detalle más adelante.

Incluya los requisitos no funcionales al evaluar los riesgos para su carga de trabajo, como la seguridad, la fiabilidad, la eficiencia del rendimiento, la optimización de los costos y el impacto de las mejoras en su capacidad para gestionar su carga de trabajo.

Al aplicar este paso al [Escenario de ejemplo](#), se evalúan las mejoras previstas y se obtienen los siguientes resultados:

Práctica recomendada	Mejora específica	Potencial	Costo	Riesgo
Uso de la mínima cantidad de hardware para cumplir sus necesidades	Implementación del escalado predictivo para reducir los periodos de baja utilización	Medio	Baja	Baja
Uso de tecnologías que mejor admitan patrones de almacenamiento y acceso a los datos	Implementación de mecanismos de compresión más efectivos para reducir el almacenamiento total y el tiempo necesario para lograrlo	Alta	Baja	Baja

La implementación del escalado predictivo reduce las horas de vCPU consumidas por las instancias infrutilizadas o no utilizadas, lo que proporciona beneficios moderados en comparación con los mecanismos de escalado existentes, con una reducción estimada del 11 % en los recursos consumidos. Los costos involucrados son bajos e incluyen la configuración de los recursos de la nube y la operación de escalado predictivo para Amazon EC2 Auto Scaling. El riesgo se traduce en un rendimiento limitado cuando el escalado horizontal se efectúa de forma reactiva en respuesta a una demanda que supera las previsiones.

La implementación de una compresión más eficaz puede tener un impacto significativo, ya que se reduce considerablemente el tamaño de los archivos en todas las imágenes originales y

manipuladas, y se calcula que las necesidades de almacenamiento en producción se reducen en un 25 %. La implementación del nuevo algoritmo es una sustitución que requiere poco esfuerzo y conlleva poco riesgo.

Priorización y planificación de las mejoras

Priorice las mejoras identificadas en función del mayor impacto previsto con los costos más bajos y el riesgo aceptable.

Decida en qué mejoras centrarse inicialmente e inclúyalas en su hoja de ruta de planificación y desarrollo de recursos.

Al aplicar este paso al [Escenario de ejemplo](#), prioriza las mejoras específicas de la siguiente manera:

Priority (Prioridad)	Mejora	Potencial	Costo	Riesgo
1	Implementación de mecanismos de compresión más efectivos	Alta	Baja	Baja
2	Implementación del escalado predictivo	Medio	Baja	Baja

El alto potencial, el bajo costo y el riesgo de actualizar la compresión de archivos hacen que sea un objetivo de gran valor para su empresa y una prioridad por encima de la implementación del escalado predictivo. Usted determina que implementar el escalado predictivo, con un impacto potencial medio, un bajo costo y un bajo riesgo, debe ser la mejora prioritaria una vez finalizada la compresión de archivos.

Asigna a un miembro del equipo la tarea de implementar una compresión de archivos mejorada y agregar el escalado predictivo a sus tareas pendientes.

Prueba y validación de las mejoras

Haga pequeñas pruebas con una inversión mínima para reducir el riesgo de un esfuerzo a gran escala.

Implemente una copia representativa de su carga de trabajo en su entorno de pruebas para limitar el costo y el riesgo de hacer las pruebas y la validación. Haga un conjunto predefinido de transacciones de prueba, mida los recursos aprovisionados y determine los recursos utilizados por unidad de trabajo para establecer una base de referencia de pruebas.

Implemente su objetivo de mejora en el entorno de pruebas y repita la prueba con la misma metodología en las mismas condiciones. A continuación, mida los recursos aprovisionados y los recursos utilizados por unidad de trabajo con la mejora implementada.

Calcule el cambio porcentual de los recursos aprovisionados por unidad de trabajo con respecto a su base de referencia y determine la reducción cuantitativa esperada de los recursos aprovisionados en su entorno de producción. Compare estos valores con los valores previstos. Determine si el resultado es un nivel de mejora aceptable. Evalúe si alguna compensación en los recursos adicionales consumidos hace que el beneficio neto de la mejora sea inaceptable.

Determine si la mejora es un éxito y si se deben invertir recursos en implementar el cambio en producción. Si se considera que el cambio no ha tenido éxito en este momento, redirija sus recursos para probar y validar su próximo objetivo y continuar con el ciclo de mejora.

% de reducción de los recursos aprovisionados por unidad de trabajo	Reducción cuantitativa de los recursos aprovisionados	Action
Cumplió las expectativas	Cumplió las expectativas	Continuar con la mejora
No cumplió las expectativas	Cumplió las expectativas	Continuar con la mejora
Cumplió las expectativas	No cumplió las expectativas	Buscar una mejora alternativa
No cumplió las expectativas	No cumplió las expectativas	Buscar una mejora alternativa

Al aplicar este paso al [Escenario de ejemplo](#), se hacen pruebas para validar el éxito.

Tras hacer las pruebas con el algoritmo de compresión mejorado, el porcentaje de reducción de los recursos provisionados por unidad de trabajo (el almacenamiento necesario tanto para la imagen original como para la imagen modificada) cumplió las expectativas, con una reducción media del 30 % en el almacenamiento provisionado y un aumento insignificante de la carga de computación.

Determina que los recursos de computación adicionales necesarios para aplicar el algoritmo de compresión mejorado a los archivos existentes en producción son insignificantes en comparación con la reducción de almacenamiento lograda. Ha confirmado el éxito de la reducción cuantitativa de los recursos necesarios (TB de almacenamiento) y ha aprobado la mejora para su implementación en producción.

Implementación de cambios en producción

Implemente mejoras probadas, validadas y aprobadas en producción. Lleve a cabo implementaciones limitadas, confirme la funcionalidad de su carga de trabajo, pruebe la reducción real de los recursos provisionados y de los recursos consumidos por unidad de trabajo durante la implementación limitada y compruebe las consecuencias imprevistas del cambio. Continúe con las implementaciones completas después de hacer las pruebas correctas.

Revierta los cambios si las pruebas no son satisfactorias o si el cambio tiene consecuencias imprevistas e inaceptables.

Al aplicar este paso al [Escenario de ejemplo](#), se llevan a cabo las siguientes acciones.

Los cambios en producción se aplican mediante una implementación limitada a través de una metodología de implementación azul/verde. Las pruebas de funcionalidad de las instancias recién implementadas se llevan a cabo correctamente. Se observa una reducción media del 26 % en el almacenamiento provisionado para los archivos de imagen originales y manipulados. No se observa ningún indicio de un aumento en la carga de computación al comprimir archivos nuevos.

Observa una disminución imprevista en el tiempo necesario para comprimir los archivos de imagen y lo atribuye al código altamente optimizado del nuevo algoritmo de compresión.

Continúe con la implementación completa de la nueva versión.

Medición de los resultados y replicación de los éxitos

Mida los resultados y replique los éxitos de las siguientes maneras:

- Mida la mejora inicial de los recursos aprovisionados por unidad de trabajo y la disminución cuantitativa de los recursos aprovisionados.
- Compare las estimaciones iniciales y los resultados de las pruebas con sus mediciones de producción. Identifique los factores que podrían haber contribuido a las diferencias y actualice sus metodologías de estimación y pruebas cuando proceda.
- Determine el éxito y el grado de éxito y comparta los resultados con las partes interesadas.
- Si tuvo que revertir los cambios debido a que las pruebas no fueron satisfactorias o a las consecuencias negativas no deseadas del cambio, identifique los factores que contribuyeron a ello. Efectúe iteraciones cuando sea viable o evalúe nuevos enfoques para lograr los objetivos del cambio.
- Aproveche lo que ha aprendido, establezca estándares y aplique las mejoras exitosas a otros sistemas que puedan beneficiarse de manera similar. Recopile y comparta su metodología, los artefactos relacionados y los beneficios netos entre equipos y organizaciones para que otros puedan adoptar su estándar y replicar su éxito.
- Supervise los recursos aprovisionados por unidad de trabajo y haga un seguimiento de los cambios y del impacto total a lo largo del tiempo. Los cambios en la carga de trabajo, o la forma en que los clientes consumen la carga de trabajo, pueden repercutir en la eficacia de la mejora. Vuelva a evaluar las oportunidades de mejora si observa una disminución significativa a corto plazo en la eficacia de la mejora o una reducción acumulada de la eficacia a lo largo del tiempo.
- Cuantifique el beneficio neto de la mejora a lo largo del tiempo (incluidos los beneficios recibidos por otros equipos que aplicaron la mejora, si están disponibles) para mostrar el retorno de la inversión derivado de las actividades de mejora.

Al aplicar este paso al [Escenario de ejemplo](#), se miden los siguientes resultados.

Su carga de trabajo muestra una mejora inicial de una reducción del 23 % en los requisitos de almacenamiento tras implementar y aplicar el nuevo algoritmo de compresión a los archivos de imagen existentes.

El valor medido coincide en gran medida con las estimaciones iniciales (25 %) y se determina que la diferencia significativa en comparación con las pruebas (30 %) se debe a que los archivos de imagen utilizados en las pruebas no son representativos de los archivos de imagen presentes en producción. El conjunto de imágenes de prueba se modifica para que refleje de forma más adecuada las imágenes en producción.

La mejora se considera un éxito total. La reducción total del almacenamiento aprovisionado es un 2 % inferior al 25 % estimado, pero un 23 % sigue siendo una enorme mejora en cuanto al impacto en la sostenibilidad y va acompañada de un ahorro de costos equivalente.

Las únicas consecuencias imprevistas del cambio son la reducción beneficiosa del tiempo transcurrido para efectuar la compresión y una reducción equivalente del consumo de vCPU. Estas mejoras se atribuyen al código altamente optimizado.

Establece un proyecto interno de código abierto en el que comparte su código, los artefactos asociados, la orientación sobre cómo implementar el cambio y los resultados de la implementación. El proyecto interno de código abierto facilita a sus equipos la adopción del código para todos sus casos de uso del almacenamiento persistente de archivos. Sus equipos adoptan la mejora como norma. Los beneficios secundarios del proyecto interno de código abierto son que todos los que adoptan la solución se benefician de las mejoras de la solución y cualquier persona puede contribuir con mejoras al proyecto.

Publica su éxito y comparte el proyecto de código abierto en toda su organización. Cada equipo que adopta la solución replica el beneficio con una inversión mínima y aumenta el beneficio neto obtenido de su inversión. Publica estos datos como una historia de éxito continuo.

Seguirá supervisando el impacto de la mejora a lo largo del tiempo y aplicará los cambios necesarios en el proyecto interno de código abierto.

Sostenibilidad como requisito no funcional

Incorporar la sostenibilidad a su lista de requisitos empresariales puede resultar en soluciones más rentables. Centrarse en obtener más valor de los recursos que utiliza y utilizar menos recursos se traduce directamente en un ahorro de costos en AWS, ya que solo paga por lo que utiliza.

Es posible que el cumplimiento de los objetivos de sostenibilidad no requiera compensaciones equivalentes en una o más métricas tradicionales, como el tiempo de actividad, la disponibilidad o el tiempo de respuesta. Puede lograr ganancias significativas en materia de sostenibilidad sin un impacto mensurable en los niveles de servicio. Cuando se requieren compensaciones menores, las mejoras de sostenibilidad que se obtengan mediante estas compensaciones pueden compensar el cambio en la calidad del servicio.

Anime a los miembros de su equipo a experimentar continuamente con las mejoras de sostenibilidad a medida que desarrollan los requisitos funcionales. Los equipos también deberían incorporar métricas proxy al establecer los objetivos para garantizar que evalúan la intensidad de los recursos al desarrollar sus cargas de trabajo.

A continuación, se muestran algunos ejemplos de compensaciones que pueden reducir los recursos de la nube que consume:

Ajuste de la calidad del resultado: puede cambiar la calidad de los resultados (QoR) por una reducción de la intensidad de la carga de trabajo con un cálculo aproximado. La práctica de la computación aproximada busca oportunidades para aprovechar la brecha entre lo que los clientes necesitan y lo que realmente se produce. Por ejemplo, si coloca los datos en una estructura de datos establecida, puede eliminar el operador ORDER BY en SQL para eliminar el procesamiento innecesario, lo que ahorra recursos y, al mismo tiempo, proporciona una respuesta aceptable.

Ajuste del tiempo de respuesta: una respuesta con un tiempo de respuesta más lento puede reducir las emisiones de carbono al minimizar los gastos compartidos. El procesamiento de tareas efímeras y puntuales puede generar una sobrecarga de inicio. Agrupe y procese las tareas en lotes en lugar de pagar los gastos generales cada vez que llegue una tarea. El procesamiento por lotes cambia el tiempo de respuesta por una reducción de la sobrecarga compartida que supone activar una instancia, descargar el código fuente y ejecutar el proceso.

Ajuste de la disponibilidad: con AWS, puede agregar redundancia y cumplir los objetivos de alta disponibilidad con tan solo unos clics. Puede aumentar la redundancia mediante técnicas como la estabilidad estática mediante el aprovisionamiento de recursos inactivos, lo que siempre se

traduce en una disminución de la utilización. Evalúe las necesidades de la empresa al establecer los objetivos. Las compensaciones relativamente menores en cuanto a la disponibilidad pueden traducirse en mejoras mucho mayores en la utilización. Por ejemplo, el patrón de arquitectura de estabilidad estática implica el aprovisionamiento de una capacidad de conmutación por error inactiva para que asuma la carga inmediatamente después de que se produzca un fallo en un componente. Reducir el requisito de disponibilidad puede eliminar la necesidad de contar con capacidad en línea inactiva, ya que permite que la automatización implemente los recursos de reemplazo. La incorporación de la capacidad de conmutación por error bajo demanda impulsa una mayor utilización general sin afectar a la empresa durante las operaciones normales y tiene la ventaja secundaria de reducir los costos.

Prácticas recomendadas para la sostenibilidad en la nube

Optimice la ubicación de la carga de trabajo y la arquitectura en función de la demanda, el software, los datos, el hardware y los procesos para aumentar la eficiencia energética. En cada una de estas áreas, pueden aplicarse prácticas recomendadas para reducir el impacto en la sostenibilidad de su carga de trabajo en la nube; para ello, se maximiza su utilización y se minimiza el desaprovechamiento y el total de recursos que se implementan y se conectan para admitir la carga de trabajo.

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 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

- 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 costo y la latencia:
 - Confirme que estas regiones cumplen con la normativa local vigente.
 - 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 [AWS Pricing Calculator](#).
 - Pruebe la latencia de la red entre las ubicaciones de sus usuarios finales y cada Región de AWS.
- 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

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 desescalarsse verticalmente mediante la modificación del tamaño de la instancia, escalarse o desescalarsse 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

Mecanismo de escalado automático	Dónde se usa
	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 2023 - Scaling FM inference to hundreds of models with Amazon SageMaker](#)
- [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.

- Defina 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 respuestas 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 infrutilizados.

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:

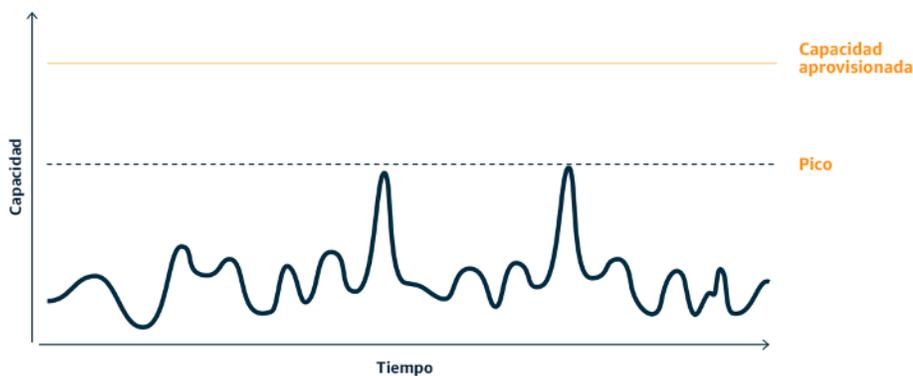
- Procesa las solicitudes de los clientes inmediatamente cuando no es necesario.
- No analiza los requisitos de las solicitudes de los clientes.

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

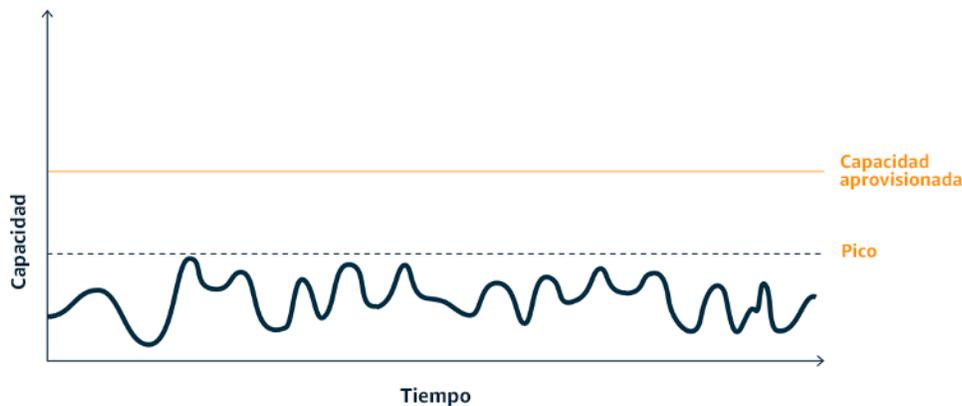
Guía para la implementación

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

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 actualizar 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:

- Aprovechona 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

Mecanismo de programación	Descripción
Programador de instancias	ejecutar tareas según una programación o en respuesta a un evento de EventBridge. 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

Revise la carga de trabajo para identificar los componentes inactivos o no utilizados. 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 los 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](#).
- Supervise 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 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 y activos 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 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.
- Conozca los [recursos que aprovisiona su carga de trabajo para completar una unidad de trabajo](#).

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](#)

Ejemplos relacionados:

- [Optimize Hardware Patterns and Observe Sustainability KPIs](#)

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 CodeWhisperer](#), 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](#)
- [AWS re:Invent 2023 - Best practices for Amazon CodeWhisperer](#)
- [Automate Code Reviews and Application Performance Recommendations with Amazon CodeGuru](#)

Ejemplos relacionados:

- [Optimizing Code 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 pague 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-BP04 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?](#)
- [AppStream 2.0 Documentation](#)
- [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

Utilice los patrones de software y arquitectura que mejor se adapten a las características de sus datos y a sus patrones de acceso. Por ejemplo, utilice una [arquitectura de datos moderna en AWS](#) que le permita utilizar 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

- 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 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 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 base de datos para garantizar 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](#).
- 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](#)

Administración de datos

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. Analice sus datos y use las configuraciones y tecnologías de almacenamiento que mejor admitan el 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.
Almacenamiento de archivos	Amazon S3 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 computación. Amazon EFS aumenta y reduce automáticamente el almacenamiento y optimiza el rendimiento para ofrecer latencias bajas y consistentes.

Tipo	Tecnología	Características clave
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 seleccionar el sistema de archivos adecuado para sus necesidades de carga de trabajo.
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.

Tipo	Tecnología	Características clave
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.
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](#)
- [What is Amazon S3 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 administrar eficazmente los conjuntos de datos a lo largo de su ciclo de vida, configure las políticas de ciclo de vida, que son reglas que definen cómo administrar los conjuntos de datos.

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.

Pasos para la implementación

- [Clasifique los conjuntos de datos de su carga de trabajo.](#)
- Defina procedimientos de gestión para cada clase de datos.
- Establezca políticas de ciclo de vida automatizadas para la aplicación de 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 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

Servicio de almacenamiento	Manera de establecer políticas de ciclo de vida automatizadas
	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 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 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

No almacene datos que no necesite. Automatice la eliminación de datos innecesarios. Use tecnologías que desdupliquen los datos en el nivel de archivo y de bloque. Aproveche las características de replicación y redundancia de datos nativos de los servicios.

Pasos para la implementación

- Evalúe si puede evitar el almacenamiento de datos mediante los conjuntos de datos existentes y disponibles públicamente en [AWS Data Exchange](#) y [Open Data en AWS](#).
- 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 identificadores) 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.

- Analice el acceso de datos para identificar los datos innecesarios. Automatice las políticas de ciclo de vida. Aproveche las características de los servicios nativos, como [Amazon DynamoDB Time To Live](#), [Amazon S3 Lifecycle](#) o la [retención de registros de Amazon CloudWatch](#) para la eliminación.

- 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](#)
- Use una tecnología de copia de seguridad que pueda crear copias incrementales.
- 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]).
- Centralice los datos de registro y de seguimiento, deduplique las entradas de registro que sean idénticas y establezca mecanismos para ajustar los detalles cuando sea necesario.
- Rellene las memorias caché previamente solo cuando se justifique.
- Establezca la supervisión y la automatización de la memoria caché para ajustar el tamaño de esta en consonancia.
- 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

- 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.

- Copie datos en sistemas de archivos compartidos, o recupérellos 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 según corresponda para 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 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

- 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 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 (como Parquet u ORC) y comprima los datos antes de moverlos por la red.
- 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 que puedan ayudarle 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 respuestas 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](#)

Ejemplos relacionados:

- [Architecting for sustainability - Minimize data movement across networks](#)

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

- Implemente la política de clasificación de datos tal como se describe en [SUS04-BP01 Implementación de una política de clasificación de datos](#).
- Aproveche la importancia 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 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](#)

Ejemplos relacionados:

- [Laboratorio de Well-Architected: Copia de seguridad de los datos](#)

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.

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](#) 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 infrutilizada.
 - 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 AWS Management Console](#)
- [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](#)
- [Migración de las bases de datos de Amazon Relational Database Service a Graviton](#)

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 administrado.

Tarea	Qué usar en AWS
	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

- 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 Inferentia, como las instancias Inf2, ofrecen hasta un [50 % más de rendimiento por vatio que las instancias de Amazon EC2 comparables](#).

- Recopile la métrica 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](#)
- 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:

- [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](#)

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

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

Prácticas recomendadas

- [SUS06-BP01 Adopción de métodos que permitan introducir mejoras en la sostenibilidad rápidamente](#)
- [SUS06-BP02 Mantenimiento de una carga de trabajo actualizada](#)
- [SUS06-BP03 Incremento del uso de los entornos de compilación](#)
- [SUS06-BP04 Uso de granjas de dispositivos administrados para pruebas](#)

SUS06-BP01 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](#)
- [Prácticas de desarrollo ágiles y escalables basadas en AWS CodeCommit](#)

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](#)

Ejemplos relacionados:

- [Well-Architected Lab - Turning cost & usage reports into efficiency reports](#)

SUS06-BP02 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](#)

Ejemplos relacionados:

- [Well-Architected Labs - Inventory and Patch Management](#)
- [Laboratorio: AWS Systems Manager](#)

SUS06-BP03 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?](#)

- [What is AWS CodeBuild?](#)
- [Programador de instancias de AWS](#)

Videos relacionados:

- [AWS re:Invent 2023 - Continuous integration and delivery for AWS](#)

SUS06-BP04 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 su 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](#)

Conclusión

Un número cada vez mayor de organizaciones está estableciendo objetivos de sostenibilidad en respuesta a los cambios en la regulación gubernamental, la ventaja competitiva y la demanda de los clientes, los empleados y los inversores. Los CTO, arquitectos, desarrolladores y miembros de los equipos de operaciones buscan formas de contribuir directamente a los objetivos de sostenibilidad de su organización. Al utilizar estos principios de diseño y las prácticas recomendadas respaldados por los servicios de AWS, puede tomar decisiones informadas que equilibren la seguridad, el costo, el rendimiento, la fiabilidad y la excelencia operativa con los resultados de sostenibilidad de sus cargas de trabajo de Nube de AWS. Cada medida que tome para reducir el uso de recursos y aumentar la eficiencia de sus cargas de trabajo contribuye a reducir el impacto ambiental y contribuye a los objetivos de sostenibilidad más amplios de sus organizaciones.

Colaboradores

Los colaboradores de este documento son:

- Sam Mokhtari, Senior Efficiency Lead Solutions Architect, Amazon Web Services
- Brendan Sisson, Principal Sustainability Solutions Architect, Amazon Web Services
- Margaret O'Toole, Sustainability Tech Leader, Amazon Web Services
- Steffen Grunwald, Principal Sustainability Solutions Architect, Amazon Web Services
- Ryan Eccles, Principal Engineer, Sustainability, Amazon
- Rodney Lester, Principal Architect, Amazon Web Services
- Adrian Cockcroft, VP Sustainability Architecture, Amazon Web Services
- Ian Meyers, Director of Technology, Solutions Architecture, Amazon Web Services

Documentación adicional

Para obtener información adicional, consulte los siguientes recursos:

- [AWS Well-Architected](#)
- [Centro de arquitectura de AWS](#)
- [Sustainability in the Cloud](#)
- [AWS hace posible las soluciones para la sostenibilidad](#)
- [The Climate Pledge](#)
- [Objetivos de desarrollo sostenible de Naciones Unidas](#)
- [Greenhouse Gas Protocol](#)

Revisiones del documento

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

Cambio	Descripción	Fecha
Actualización de las directrices de prácticas recomendadas	Se hicieron pequeños cambios en todo el pilar.	27 de junio de 2024
Actualizaciones de los niveles de riesgo	Se aplicaron actualizaciones menores de los niveles de riesgo de las prácticas recomendadas.	3 de octubre de 2023
Actualización de las directrices de prácticas recomendadas	Las prácticas recomendadas se actualizaron con nuevas directrices en las siguientes áreas: adaptación a la demanda , software y arquitectura , datos y hardware y servicios .	13 de julio de 2023
Actualizaciones del nuevo marco	Se actualizaron las prácticas recomendadas con una guía prescriptiva y se agregaron nuevas prácticas recomendadas.	10 de abril de 2023
Documento técnico actualizado	Se actualizaron las prácticas recomendadas con una nueva guía de implementación.	15 de diciembre de 2022
Documento técnico actualizado	Se ampliaron las prácticas recomendadas y se agregaron planes de mejora.	20 de octubre de 2022

[Publicación inicial](#)

Se publicó Pilar de sostenibilidad: Marco de AWS Well-Architected.

2 de diciembre de 2021

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.

© 2023 Amazon Web Services, Inc. o sus filiales. Todos los derechos reservados.

Glosario de AWS

Para ver la terminología más reciente de AWS, consulte el [Glosario de AWS](#) en la Referencia de Glosario de AWS.