



training and  
certification

AWS Academy Cloud Foundations (LA)  
Module 06 Student Guide  
Versión 2.0.16

100-ACCLFO-20-LA-SG

© 2024, Amazon Web Services, Inc. o sus filiales. Todos los derechos reservados.

Este contenido no puede reproducirse ni redistribuirse, total ni parcialmente, sin el permiso previo por escrito de Amazon Web Services, Inc. Queda prohibida la copia, el préstamo o la venta de carácter comercial.

Para correcciones o comentarios relacionados con el curso, contacte con nosotros en  
<https://support.aws.amazon.com/#/contacts/aws-training>.

Todas las marcas comerciales pertenecen a sus propietarios.

# Contenido

Módulo 6: Cómputo

4



# Módulo 6: Cómputo

AWS Academy Cloud Foundations

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Le damos la bienvenida al módulo 6: cómputo

## Información general sobre el módulo

### Temas

- Información general sobre los servicios de cómputo
- Amazon EC2
- Optimización de costos de Amazon EC2
- Servicios de contenedores
- Introducción a AWS Lambda
- Introducción a AWS Elastic Beanstalk

### Actividades

- Amazon EC2 versus servicios administrados
- Práctica con AWS Lambda
- Práctica con AWS Elastic Beanstalk

### Demostración

- Demostración registrada de Amazon EC2.

### Laboratorio

- Introducción a Amazon EC2



### Evaluación de conocimientos



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

2

Este módulo aborda los siguientes temas:

- Información general sobre los servicios de cómputo
- Amazon EC2
- Optimización de costos de Amazon EC2
- Servicios de contenedores
- Introducción a AWS Lambda
- Introducción a AWS Elastic Beanstalk

La Sección 2 incluye un registro de una **demonstración de Amazon EC2**. El final de esta misma sección incluye una **práctica de laboratorio**, donde practicará el lanzamiento de una instancia de EC2 mediante la Consola de administración de AWS. Además hay una actividad en esta sección que le permitirá comparar las ventajas y desventajas de ejecutar una implementación de base de datos en Amazon EC2 versus ejecutarla en Amazon Relational Database Service (RDS).

La sección 5 incluye una actividad práctica de AWS Lambda y la sección 6 incluye una actividad práctica de Elastic Beanstalk.

Finalmente, se le pedirá que complete una **evaluación de conocimientos** que pondrá a prueba su comprensión de los conceptos clave que se abordaron en este módulo.

## Objetivos del módulo

Después de completar este módulo, podrá hacer lo siguiente:

- Proporcionar información general de los diferentes servicios de cómputo de AWS en la nube.
- Mostrar cómo se usa Amazon Elastic Compute Cloud (Amazon EC2)
- Identificar la funcionalidad en la consola de Elastic Compute Cloud
- Realizar funciones básicas en Amazon EC2 para construir un entorno informático virtual
- Identificar los elementos de optimización de costos de Amazon EC2
- Mostrar cuándo usar AWS Elastic Beanstalk
- Mostrar cuándo usar AWS Lambda
- Identificar cómo ejecutar aplicaciones en contenedores en un clúster de servidores administrados



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

3

Después de completar este módulo, podrá hacer lo siguiente:

- Proporcionar información general de los diferentes servicios de cómputo de AWS en la nube.
- Mostrar cómo se usa Amazon Elastic Compute Cloud (Amazon EC2)
- Identificar la funcionalidad en la consola de Elastic Compute Cloud
- Realizar funciones básicas en Elastic Compute Cloud para construir un entorno informático virtual
- Identificar los elementos de optimización de costos de Elastic Compute Cloud
- Mostrar cuándo usar AWS Elastic Beanstalk
- Mostrar cuándo usar AWS Lambda
- Identificar cómo ejecutar aplicaciones en contenedores en un clúster de servidores administrados

# Sección 1: Información general sobre los servicios de cómputo

Módulo 6: Cómputo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Sección introductoria 1: Información general sobre los servicios de cómputo.

## Servicios de cómputo de AWS

Amazon Web Services (AWS) ofrece muchos servicios de cómputo. Este módulo analizará los servicios destacados.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

5

Amazon Web Services (AWS) ofrece muchos servicios de cómputo. Aquí se muestra un breve resumen de lo que ofrece cada servicio de cómputo:

- **Amazon Elastic Compute Cloud (Amazon EC2)**, proporciona máquinas virtuales redimensionables.
- **Amazon EC2 Auto Scaling** admite la disponibilidad de las aplicaciones al permitirle definir condiciones que iniciarán o terminarán las instancias de EC2 de manera automática.
- **Amazon Elastic Container Registry (Amazon ECR)** se utiliza para almacenar y recuperar imágenes de Docker.
- **Amazon Elastic Container Service (Amazon ECS)** es un servicio de coordinación compatible con Docker.
- **VMware Cloud on AWS** le permite aprovisionar una nube híbrida sin hardware personalizado.
- **AWS Elastic Beanstalk** proporciona una forma sencilla de ejecutar y administrar aplicaciones web.
- **AWS Lambda** es una opción de cómputo sin servidor. Solo paga por el tiempo de cómputo que utilice.
- **Amazon Elastic Kubernetes Service (Amazon EKS)** le permite ejecutar Kubernetes administrado en AWS.
- **Amazon Lightsail** proporciona un servicio fácil de usar para crear una aplicación o un sitio web.
- **AWS Batch** proporciona una herramienta para ejecutar trabajos por lotes a cualquier escala.
- **AWS Fargate** proporciona una forma de ejecutar contenedores que reducen la necesidad de

administrar servidores o clústeres.

- **AWS Outposts** proporciona una manera de ejecutar AWS seleccionados en su centro de datos en las instalaciones.
- **Repositorio de aplicaciones sin servidor de AWS** proporciona una manera de descubrir, implementar y publicar aplicaciones sin servidor.

Este módulo analizará los detalles de los servicios que se destacan en la diapositiva.

## Categorización de servicios de cómputo

Servicios	Conceptos clave	Características	Facilidad de uso
• Amazon EC2	• Infraestructura como servicio (IaaS) • Basado en instancias • <b>Máquinas virtuales</b>	• Aprovisione máquinas virtuales que pueda administrar como lo deseé.	Un concepto familiar para muchos profesionales de TI.
• AWS Lambda	• <b>Cómputo</b> sin servidor • Basado en funciones • Bajo costo	• Escriba e implemente un código que se ejecute de forma programada o que se pueda desencadenar mediante eventos • Utilícelo cuando sea posible (diseñado para la nube).	Un concepto relativamente nuevo para muchos miembros del personal de TI, pero fácil de usar después de aprender cómo hacerlo.
• Amazon ECS • Amazon EKS • AWS Fargate • Amazon ECR	• Cómputo <b>basada en contenedores</b> • Basado en instancias	• Ponga en marcha y ejecute los trabajos con mayor rapidez	AWS Fargate reduce la sobrecarga administrativa, pero puede utilizar opciones que le ofrecen mayor control.
• AWS Elastic Beanstalk	• Plataforma como servicio (PaaS) • Para <b>aplicaciones web</b>	• Concéntrese en el código (la creación de su aplicación). • Se puede vincular fácilmente con otros servicios: bases de datos, sistema de nombres de dominio (DNS), etc.	Comenzar a trabajar con este servicio es fácil y rápido.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

6

Se puede considerar que cada servicio de cómputo de AWS pertenece a una de cuatro categorías amplias: máquinas virtuales (VM) que proporcionan infraestructura como servicio (IaaS), tecnología sin servidor, tecnología basada en contenedores y plataforma como servicio (PaaS).

**Amazon EC2** proporciona máquinas virtuales y se puede considerar una forma de infraestructura como servicio (IaaS). Los servicios de IaaS ofrecen flexibilidad y le permiten estar a cargo de muchas de las responsabilidades de la administración del servidor. Puede elegir el sistema operativo, así como el tamaño y las capacidades de los recursos de los servidores que lance. El concepto de las máquinas virtuales resulta familiar para los profesionales de TI con experiencia en el uso de la informática en las instalaciones. Amazon EC2 fue uno de los primeros AWS y aún es uno de los más populares.

**AWS Lambda** es una plataforma de cómputo que no requiere ningún tipo de administración. AWS Lambda le permite ejecutar código sin necesidad de aprovisionar ni administrar servidores. Solo paga por el tiempo de cómputo que consume. Este concepto de tecnología sin servidor es relativamente nuevo para muchos profesionales de TI. Sin embargo, es cada vez más popular porque admite arquitecturas nativas en la nube, lo cual posibilita una escalabilidad masiva a un costo menor que el de la ejecución ininterrumpida de servidores para respaldar las mismas cargas de trabajo.

Los servicios basados en contenedores, incluidos **Amazon Elastic Container Service**, **Amazon**

**Elastic Kubernetes Service, AWS Fargate y Amazon Elastic Container Registry**, le permiten ejecutar varias cargas de trabajo en un solo sistema operativo. Los contenedores se implementan con mayor rapidez que las máquinas virtuales, lo que ofrece capacidad de respuesta. Las soluciones basadas en contenedores son cada vez más populares.

Por último, **AWS Elastic Beanstalk** proporciona una plataforma como servicio (PaaS). Facilita la implementación rápida de las aplicaciones que se crean al prestar todos los servicios de aplicaciones necesarios. AWS administra el sistema operativo, el servidor de aplicaciones y los demás componentes de la infraestructura para que usted pueda centrarse en desarrollar el código de la aplicación.

## Elección de un servicio de cómputo óptimo

- El servicio o servicios de cómputo óptimos que utilice dependerán de su caso práctico
- Algunos aspectos a considerar:
  - ¿Cuál es el diseño de su aplicación?
  - ¿Cuáles son los patrones de uso?
  - ¿Qué ajustes de configuración querrás administrar?
- Seleccionar la solución de cómputo incorrecta para una arquitectura puede provocar una menor eficiencia del rendimiento
  - Un buen punto de partida, comprender las opciones de cómputo disponibles



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

7

AWS ofrece muchos servicios de cómputo porque los diferentes casos prácticos se benefician de diferentes entornos de cómputo. El servicio o servicios de cómputo óptimos que utilice dependerán de su caso práctico.

A menudo, la arquitectura de cómputo que utiliza está determinada por el código heredado. Sin embargo, eso no significa que no se pueda evolucionar la arquitectura para aprovechar los diseños probados nativos de la nube.

Las prácticas recomendadas incluyen lo siguiente:

- evaluar las opciones de cómputo disponibles
- comprender las opciones de configuración de cómputo disponibles
- recopilar métricas relacionadas con el cómputo
- utilizar la elasticidad disponible de los recursos
- re-evalúe constantemente las necesidades de cómputo según las métricas

A veces, un cliente comienza con una solución de cómputo y decide cambiar el diseño en función de su análisis de métricas. Si está interesado en ver un ejemplo de cómo un cliente modificó su elección de servicios de cómputo para un caso práctico particular, vea este video de la solución de seguimiento de inventario en [https://www.youtube.com/watch?v=zr3Kib0i-OQ&feature=youtu.be&did=ta\\_card&trk=ta\\_card](https://www.youtube.com/watch?v=zr3Kib0i-OQ&feature=youtu.be&did=ta_card&trk=ta_card).

## Sección 2: Amazon Elastic Compute Cloud

Módulo 6: Cómputo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Presentamos la Sección 2: Amazon EC2.

- 
-

## Amazon Elastic Compute Cloud (Amazon EC2)

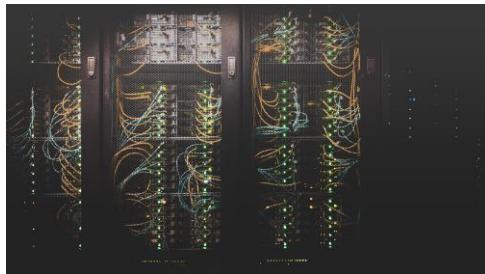
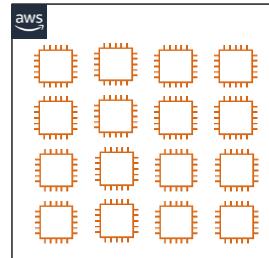


Foto de Taylor Vick en Unsplash

### Servidores en las instalaciones

#### Ejemplos de usos de instancias de EC2

- ✓ Servidor de aplicaciones
- ✓ Servidor web
- ✓ Servidor de bases de datos
- ✓ Servidor de juegos
- ✓ Servidor de correo
- ✓ Servidor de contenido multimedia
- ✓ Servidor de catálogos
- ✓ Servidor de archivos
- ✓ Servidor de cómputos
- ✓ Servidor proxy



### Instancias de Amazon EC2



Foto de panumas nikhomkhai de Pexels



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

9

**Ejecutar servidores en las instalaciones** es una tarea costosa. Se debe adquirir hardware y esta adquisición puede basarse en planes de proyecto en lugar de en la realidad de cómo se utilizan los servidores. Crear un centro de datos, así como mantenerlo o dotarlo de personal, es una tarea costosa. Las organizaciones también necesitan aprovisionar de forma permanente una cantidad suficiente de hardware para gestionar los picos de tráfico y las cargas de trabajo máximas. Una vez creadas las implementaciones en las instalaciones tradicionales, es posible que la capacidad del servidor no se utilice y esté inactiva durante una parte significativa del tiempo que los servidores están en ejecución, lo que supone un desperdicio.

Amazon Elastic Compute Cloud (Amazon EC2) proporciona máquinas virtuales en las que puede alojar los mismos tipos de aplicaciones que podría ejecutar en un servidor en las instalaciones locales tradicionales. Proporciona capacidad de cómputo segura y modificable en la nube. Las instancias EC2 pueden admitir diversas cargas de trabajo. Los usos comunes de las instancias de EC2 incluyen los siguientes, entre otros:

- Servidores de aplicaciones
- Servidores web
- Servidores de bases de datos
- Servidores de juegos
- Servidores de correo
- Servidores multimedia
- Servidores de catálogos

- Servidores de archivos
- Servidores de cómputo
- Servidores proxy

## Información general sobre Amazon EC2

### • Amazon Elastic Compute Cloud (Amazon EC2)



- Proporciona **máquinas virtuales**, denominadas **instancias EC2**, en la nube.
- Le proporciona *control absoluto* sobre el sistema operativo invitado (Windows o Linux) en cada instancia.
- Puede iniciar instancias de cualquier capacidad en una zona de disponibilidad en cualquier parte del mundo.
- Lance instancias desde **Amazon Machine Images (AMI)**.
- Inicie instancias con unos pocos clics o con una línea de código y estarán listas en cuestión de minutos.
- Puede controlar el tráfico hacia y desde las instancias.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

10

### El EC2 en Amazon EC2 significa Elastic Compute Cloud:

- Elastic se refiere al hecho de que puede aumentar o reducir fácilmente el número de servidores que ejecuta para dar soporte a una aplicación de forma automática, y también puede aumentar o disminuir el tamaño de los servidores existentes.
- El cómputo se refiere a la razón por la cual la mayoría de los usuarios ejecutan servidores en primer lugar, que es alojar aplicaciones en ejecución o procesar datos, acciones que requieren recursos de cómputo, incluida la potencia de procesamiento (CPU) y la memoria (RAM).
- Cloud se refiere al hecho de que las instancias de EC2 que ejecuta están alojadas en la nube.

Amazon EC2 proporciona máquinas virtuales en la nube y le proporciona un control administrativo total sobre el sistema operativo Microsoft Windows o Linux que se ejecuta en la instancia. Se admiten la mayoría de los sistemas operativos de servidor, incluidos Windows 2008, 2012, 2016 y 2019, Red Hat, SuSE, Ubuntu y Amazon Linux.

Un sistema operativo que se ejecuta en una máquina virtual suele denominarse sistema operativo invitado para distinguirlo del sistema operativo host. El sistema operativo host se instala de manera directa en cualquier hardware de servidor que aloja una o más máquinas virtuales.

Con Amazon EC2, puede iniciar en cuestión de minutos cualquier cantidad de instancias con cualquier capacidad en cualquier zona de disponibilidad y en cualquier parte del mundo. Las

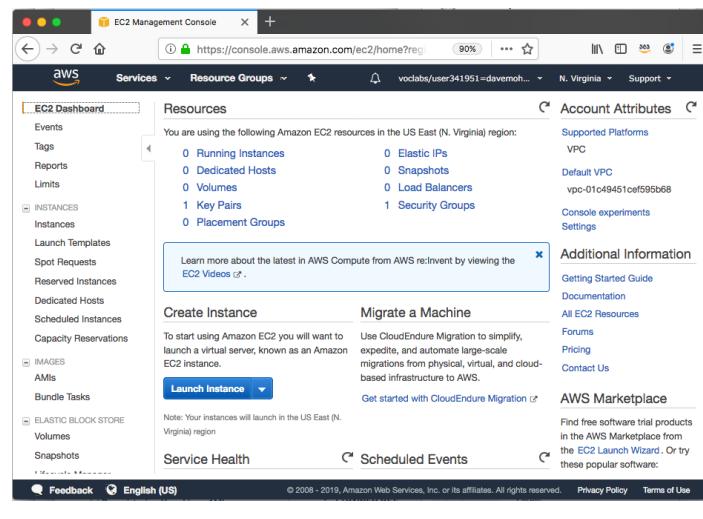
instancias se lanzan desde las Amazon Machine Images (AMI), que son en efecto plantillas de máquinas virtuales. Las AMI se analizan con más detalle más adelante en este módulo.

Puede controlar el tráfico desde las instancias y hacia ellas mediante grupos de seguridad. Además, dado que los servidores se ejecutan en la nube de AWS, puede crear soluciones que utilicen varios AWS.

## Lanzamiento de una instancia de EC2

En esta sección del módulo, se describen **nueve decisiones** clave que debe tomar al crear una instancia de EC2 mediante el **asistente de lanzamiento de** instancias de la consola de administración de AWS.

- En el camino, se explorarán los conceptos esenciales de Amazon EC2.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

11

La primera vez que lance una instancia EC2, probablemente utilice el Asistente de lanzamiento de instancias de la consola de administración de AWS. Tendrá la oportunidad de experimentar el uso del Asistente de inicio en el **laboratorio** que se encuentra en este módulo.

El **asistente de lanzamiento de instancias** facilita el lanzamiento de una instancia. Por ejemplo, si decide aceptar toda la configuración predeterminada, puede omitir la mayoría de los pasos proporcionados por el asistente e iniciar una instancia EC2 con unos pocos clics. Un ejemplo de este proceso se muestra en la **demonstración** al final de esta sección.

Sin embargo, para la mayoría de las implementaciones, querrá modificar la configuración predeterminada para que los servidores que inicie se implementen de forma que se ajusten a sus necesidades específicas.

La siguiente serie de diapositivas le presenta las decisiones esenciales que debe tomar cuando lanza una instancia. Las diapositivas cubren conceptos esenciales que es bueno saber cuándo toma estas decisiones. Estos conceptos se describen para que pueda comprender las opciones disponibles y los efectos de las decisiones que tome.

## 1. Seleccionar una AMI

Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:

1. AMI
2. Tipo de instancia
3. Configuración de red
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves



- **Imagen de máquina de Amazon (AMI)**

- Es una plantilla que se utiliza para crear una instancia EC2 (que es una **máquina virtual, o VM**, que se ejecuta en la nube de AWS)
- Contiene un sistema operativo **Windows o Linux**
- A menudo también tiene algún **software** preinstalado

- **Elecciones de AMI:**

- Inicio rápido: *AMI de Linux y Windows que proporciona AWS*
- Mis AMI: *cualquier AMI que haya creado*
- AWS Marketplace: *plantillas preconfiguradas de terceros*
- AMI de la comunidad: *AMI que comparten los demás; utilícelas bajo su propio riesgo*



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

12

Una **Amazon Machine Image (AMI)** proporciona la información necesaria para lanzar una instancia de EC2. Debe especificar una AMI de origen al lanzar una instancia. Puedes usar diferentes AMI para lanzar distintos tipos de instancias. Por ejemplo, puede elegir una AMI para lanzar una instancia que se convertirá en un servidor web y otra AMI para implementar una instancia que alojará un servidor de aplicaciones. También puede lanzar varias instancias desde una sola AMI.

Una AMI incluye los siguientes componentes:

- Una **plantilla para el volumen raíz** de la instancia. Un volumen raíz suele contener un sistema operativo (SO) y todo lo que se instaló en ese SO (aplicaciones, bibliotecas, etc.). Amazon EC2 copia la plantilla en el volumen raíz de una nueva instancia EC2 y, a continuación, la inicia.
- **Permisos de lanzamiento** que controlan qué cuentas de AWS pueden utilizar la AMI.
- Una **asignación de dispositivos de bloques** que especifica los volúmenes que deben adjuntarse a la instancia (si hay) cuando se lanza.

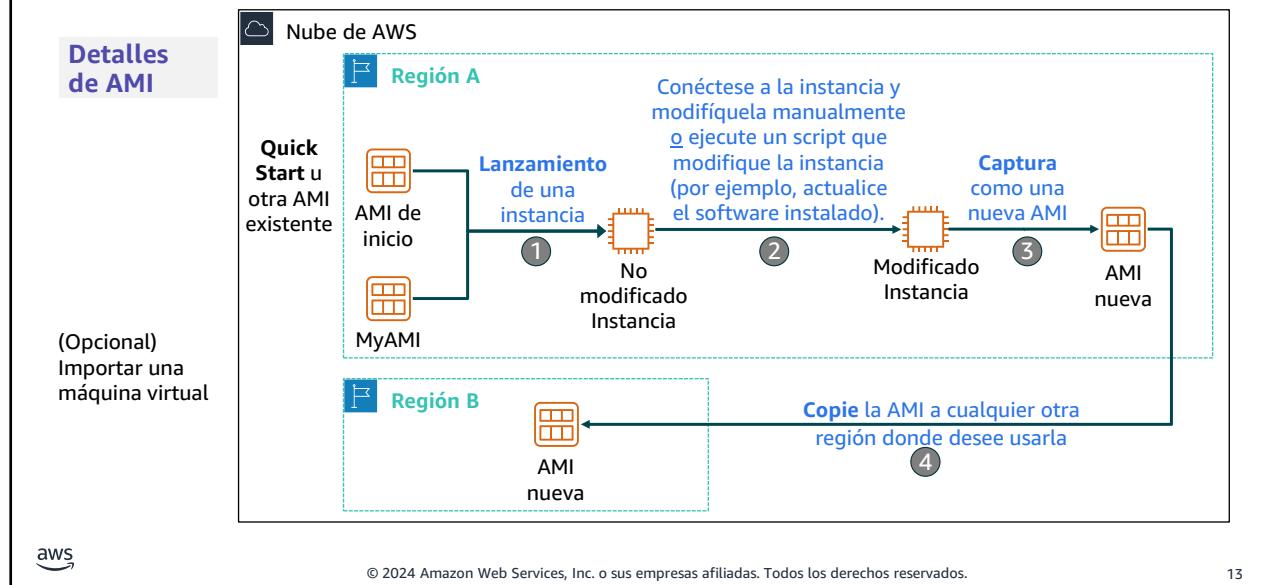
Puede elegir muchas AMI:

- **Quick Start:** AWS ofrece varias AMI prediseñadas para iniciar las instancias. Estas AMI incluyen muchas opciones de Linux y Windows.
- **Mis AMI:** estas AMI son las AMI que ha creado.
- **AWS Marketplace:** AWS Marketplace ofrece un catálogo digital que enumera miles de soluciones de software. Estas AMI pueden ofrecer casos prácticos específicos para ayudarlo a

comenzar rápidamente.

- **AMI comunitarias:** estas AMI son creadas por personas de todo el mundo. AWS no comprueba estas AMI, así que úselas bajo su propio riesgo. Las AMI comunitarias pueden ofrecer muchas soluciones diferentes a diversos problemas, pero úselas con cuidado. Evite usarlas en cualquier entorno corporativo o de producción.

## Creación de una nueva AMI: ejemplo



13

Una AMI se crea a partir de una instancia de EC2. Puede **importar** una máquina virtual para que se convierta en una instancia de EC2 y luego guardar la instancia de EC2 como una AMI. Luego puede iniciar una instancia de EC2 desde esa AMI. Alternativamente, puede comenzar con una **AMI existente**, como las AMI de Quick Start que proporciona AWS, y crear una instancia de EC2 a partir de esta.

Independientemente de las opciones que elija (paso 1), tendrá lo que el diagrama denomina una instancia sin modificar. A partir de esa instancia, se puede crear una instancia dorada, es decir, una máquina virtual que configuró con el sistema operativo específico y la configuración de la aplicación que desea (paso 2), y luego capturarla como una nueva AMI (paso 3). Cuando crea una AMI, Amazon EC2 detiene la instancia, toma una instantánea de su volumen raíz y finalmente registra la instantánea como una AMI.

Luego del registro de la AMI, esta se puede utilizar para lanzar nuevas instancias en la misma región de AWS. Ahora se puede considerar la nueva AMI como una nueva AMI inicial. Es posible que también desee copiar la AMI a otras regiones (paso 4), para que las instancias de EC2 puedan iniciarse en esas ubicaciones también.

## 2. Seleccionar un tipo de instancia

Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:

1. AMI
2. **Tipo de instancia**
3. Configuración de red
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves

- Considere su caso de uso

- ¿Cómo se utilizará la instancia EC2 que cree?

- El **tipo de instancia** que elija determina lo siguiente:

- Memoria (RAM)
  - Potencia de procesamiento (CPU)
  - Espacio en disco y tipo de disco (almacenamiento)
  - Rendimiento de red



- Categorías de tipos de instancia:

- Instancias de propósito general
  - optimizadas para cómputo
  - Optimizadas para memoria
  - Optimizadas para almacenamiento
  - Con cómputo acelerado

- Los tipos de instancia ofrecen *familia, generación y tamaño*



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

14

Después de elegir la AMI para lanzar la instancia, debe elegir un tipo de instancia.

Amazon EC2 proporciona una selección de **tipos de instancias** optimizados para adaptarse a diferentes casos prácticos. Los tipos de instancias tienen diversas combinaciones de CPU, memoria, almacenamiento y capacidad de redes. Los diferentes tipos de instancias le dan la flexibilidad de elegir la combinación adecuada de recursos para las aplicaciones. Cada tipo de instancia se ofrece en uno o varios tamaños, lo que le permite escalar los recursos según los requisitos de la carga de trabajo deseada.

Las categorías de tipos de instancia incluyen instancias de uso general, optimizadas para cómputo, optimizadas para memoria, optimizadas para almacenamiento y de cómputo acelerado. Cada categoría de tipo de instancia ofrece muchos tipos de instancias entre los que elegir.

## Nombres y tamaños de los tipos de instancia EC2

### Nombre del tipo de instancia

- Ejemplo: **t3.grande**

- T es el nombre de la familia
- 3 es el número de la generación
- Grande es el tamaño

### Tamaños de instancia de ejemplo

Nombre de instancia	vCPU	Memoria (GB)	Almacenamiento
t3.nano	2	0,5	Solo EBS
t3.micro	2	1	Solo EBS
t3.small	2	2	Solo EBS
t3.medium	2	4	Solo EBS
t3.large	2	8	Solo EBS
t3.xlarge	4	16	Solo EBS
t3.2xlarge	8	32	Solo EBS



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

15

Cuando se analiza un tipo de instancia de EC2, verá que el nombre tiene varias partes. Por ejemplo, considere el tipo T.

T es el **nombre de la familia**, al cual le sigue un número. Aquí, ese número es 3.

El número es el **número de la generación** de ese tipo. Por lo tanto, una instancia t3 es la tercera generación de la familia T. En general, los tipos de instancias que son de una generación más alta son más potentes y ofrecen un mejor relación calidad-precio.

La parte siguiente del nombre corresponde a la capacidad de la instancia. Cuando se comparan las capacidades, es importante tener en cuenta la parte del coeficiente de la categoría de capacidad.

Por ejemplo, una instancia **t3.2xlarge** tiene el doble de vCPU y memoria que una instancia t3.xlarge. La instancia t3.xlarge tiene, a su vez, el doble de vCPU y memoria que una instancia t3.large.

Además es importante tener en cuenta que el **ancho de banda de la red** también está vinculado al tamaño de la instancia de EC2. Si ejecuta trabajos que hacen un uso intensivo de la red, es posible que deba aumentar las especificaciones de las instancias para satisfacer sus necesidades.

## Seleccione el tipo de instancia: según el caso práctico

	 Instancias de propósito general	 Optimizadas para cómputo	 Optimizadas para memoria	 Con cómputo acelerado	 Optimizadas para Almacenamiento
Tipos de instancias	a1, m4, m5, t2, t3	c4, c5	r4, r5, x1, z1	f1, g3, g4, p2, p3	d2, h1, i3
Caso de uso	Amplio	Alto rendimiento	Bases de datos en memoria	Machine learning	Sistemas de archivos distribuidos



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

16

Los tipos de instancias varían de varias maneras e incluyen: tipo de CPU, CPU o número de núcleos, tipo de almacenamiento, cantidad de almacenamiento, cantidad de memoria y rendimiento de la red. El gráfico proporciona una vista de alto nivel de las distintas categorías de instancias y qué familias de tipos de instancias y números de generación se ajustan a cada tipo de categoría. Evaluemos algunos de los tipos de instancia con más detalle:

- **Las instancias T3** proporcionan instancias de propósito general con rendimiento ampliable ofrecen un nivel base de rendimiento de la CPU con la posibilidad de extenderse por encima de dicho nivel. Entre los casos prácticos de este tipo de instancia, se incluyen sitios y aplicaciones web, entornos de desarrollo, servidores de compilación, repositorios de código, microservicios, entornos de prueba y ensayo, y aplicaciones de línea de negocio.
- **Las instancias C5** están optimizadas para las cargas de trabajo con uso intensivo de cómputo y ofrecen un rendimiento rentable y alto con un precio bajo por relación de cómputo. Algunos de los casos prácticos incluyen el modelado científico, el procesamiento por lotes, la presentación de anuncios, los videojuegos multijugador altamente escalables y la codificación de video.
- **Las instancias R5** están optimizadas para las aplicaciones con uso intensivo de la memoria. Entre los casos prácticos, se incluyen las bases de datos de alto rendimiento, la extracción y el análisis de datos, las bases de datos en memoria, el almacenamiento en caché en memoria distribuida a escala web, las aplicaciones que realizan el procesamiento en tiempo real de big data no estructurados, los clústeres Apache Hadoop o Apache Spark, y otras aplicaciones empresariales.

Para obtener más información acerca de cada tipo de instancia, consulte la documentación Tipos de instancia de Amazon EC2 en <https://aws.amazon.com/ec2/instance-types/>.

## Tipos de instancias: características de redes

- La banda ancha de red (Gb/s) varía según el tipo de instancia.
  - Consulte [Tipos de instancia de Amazon EC2](#) para comparar.
- Para maximizar el rendimiento de las redes y la banda ancha del tipo de instancia:
  - Si tiene instancias interdependientes, láncelas en un **grupo con ubicación en clúster**.
  - Habilitar redes mejoradas.
- La mayoría de los tipos de instancia admiten tipos de redes mejoradas.
  - Consulte la documentación sobre las [Características de redes y almacenamiento](#) para obtener más detalles.
- Tipos de redes mejoradas:
  - **Adaptador de red elástico (ENA)**: admite velocidades de red de hasta 100 Gbps.
  - **Interfaz de función virtual Intel 82599**: admite velocidades de red de hasta 10 Gbps.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

17

Además de tener en cuenta las necesidades de CPU, RAM y almacenamiento de las cargas de trabajo, también es importante tener en cuenta los requisitos del ancho de banda de la red.

Cada tipo de instancia proporciona un nivel de rendimiento de red documentado. Por ejemplo, una instancia a1.mediana brinda hasta 10 Gb/s, pero una instancia p3dn.24xgrande proporciona hasta 100 Gb/s. Elija un tipo de instancia que satisfaga sus requisitos.

Al lanzar varias instancias EC2 nuevas, Amazon EC2 intenta distribuir de forma predeterminada las instancias en el hardware subyacente. Esto lo hace para minimizar los errores correlacionados. Sin embargo, si desea especificar criterios de colocación, puede utilizar los **grupos de ubicación** para influir en la ubicación de un grupo de instancias interdependientes para satisfacer las necesidades de su carga de trabajo. Por ejemplo, puede especificar que se deben implementar tres instancias en la misma zona de disponibilidad para garantizar una menor latencia de red y un mayor rendimiento de la red entre instancias. Consulte la documentación del grupo de ubicación en <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html> para obtener más detalles.

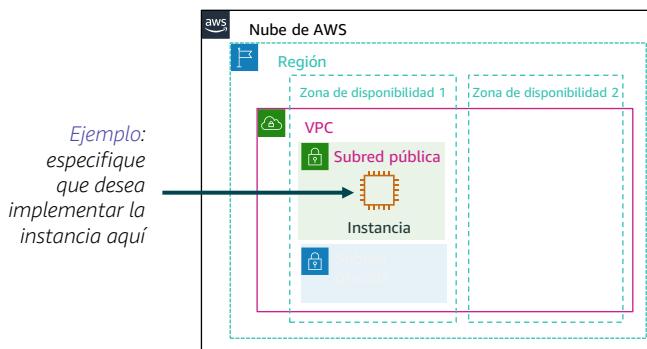
– Muchos tipos de instancias también le permiten configurar redes mejoradas para obtener un rendimiento de paquetes por segundo (PPS) mayor de forma significativa, menor variación del retraso en la llegada de paquetes a través de la red (variación de la red) y latencias más bajas. Consulte la documentación del Adaptador de red elástica (ENA) en <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/enhanced-networking-ena.html> para obtener más detalles.

### 3. Especificar la configuración de red

**Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:**

1. AMI
2. Tipo de instancia
3. **Configuración de red**
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves

- ¿Dónde debería implementarse la instancia?
- Identificar la **nube virtual privada** (VPC) y opcionalmente, la **subred**
- ¿Debería asignarse automáticamente una **dirección IP pública**?
- Para hacerla accesible a Internet



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

18

Después de elegir una AMI y un tipo de instancia, tiene que especificar la ubicación de red en la que se implementará la instancia de EC2. Se debe realizar la elección de región antes de iniciar el Asistente de lanzamiento de instancias. Compruebe que se encuentra en la página de región correcta de la consola de Amazon EC2 antes de elegir **Launch Instance (Lanzar Instancia)**.

Cuando lanza una instancia en una (**VPC**) **predeterminada**, AWS le asignará una **dirección IP pública** de forma predeterminada. Cuando lanza una instancia en una VPC no predeterminada, la subred tiene un atributo que determina si las instancias que se lanzan en esa subred reciben una dirección IP pública del grupo de direcciones IPv4 públicas. De forma predeterminada, AWS no asignará una dirección IP pública a las instancias que se lancen en una subred no predeterminada. Puede controlar si su instancia recibe una dirección IP pública mediante la modificación del atributo de dirección IP pública de su subred o la habilitación o deshabilitación de la función de dirección IP pública durante el inicio (que anula el atributo de dirección IP pública de la subred).

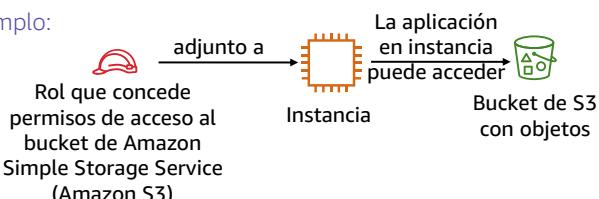
## 4. Adjuntar rol de IAM (opcional)

**Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:**

1. AMI
2. Tipo de instancia
3. Configuración de red
4. **Rol de IAM**
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves

- ¿El software de la instancia de EC2 necesita interactuar con otros AWS?
- En caso afirmativo, adjunte un **rol de IAM** adecuado.
- Un rol de AWS Identity and Access Management (IAM) que se adjunta a una instancia de EC2 se mantiene en un **perfil de instancias**.
- No se limita a adjuntar un rol solo en el lanzamiento de la instancia.
- También puede adjuntar un rol a una instancia que ya existe.

Ejemplo:



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

19

Es habitual utilizar instancias EC2 para ejecutar una aplicación que debe realizar llamadas API seguras a otros productos de AWS. Para dar soporte a estos casos prácticos, AWS le permite **adjuntar un rol de AWS Identity and Access Management (IAM) a una instancia de EC2**. Sin esta función, podría tener la tentación de colocar las credenciales de AWS en una instancia de EC2 para una aplicación que se ejecuta en esa instancia. Sin embargo, nunca debe almacenar las credenciales de AWS en una instancia EC2. Es muy poco seguro. En cambio, vincule un rol de IAM a la instancia EC2. El rol de IAM concede permisos para realizar solicitudes de Application Program Interface (API) a las aplicaciones que se ejecutan en la instancia de EC2.

Un **perfil de instancias** es un contenedor de un rol de IAM. Si utiliza la consola de administración de AWS a fin de crear un rol para Amazon EC2, la consola crea automáticamente un perfil de instancias y le asigna el mismo nombre que el rol. Cuando utiliza la consola de Amazon EC2 para lanzar una instancia con un rol de IAM, puede seleccionar un rol para asociarlo a la instancia. En la consola, la lista que se muestra es en realidad una lista de nombres de perfiles de instancias.

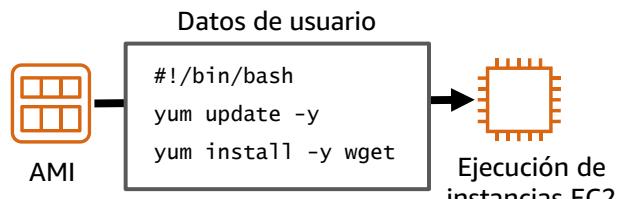
**En el ejemplo**, se utiliza un rol de IAM para otorgar permisos a una aplicación que se ejecuta en una instancia de EC2. La aplicación debe acceder a un bucket de Amazon S3.

Puede adjuntar un rol de IAM al lanzar la instancia, pero también puede adjuntar un rol a una instancia de EC2 que ya esté en ejecución. Al definir un rol que puede utilizar una instancia EC2, define qué cuentas o AWS pueden asumir el rol. También define qué acciones y recursos de API puede utilizar la aplicación después de asumir el rol. Si cambia un rol, el cambio se propaga a todas las instancias que tienen el rol asociado.

## 5. Script de datos de usuario (opcional)

Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:

1. AMI
2. Tipo de instancia
3. Configuración de red
4. Rol de IAM
5. **Datos de usuario**
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves



- Si lo desea, especifique un script de datos de usuario al lanzar la instancia
- Utilice scripts de **datos de usuario** para personalizar el entorno en tiempo de ejecución de su instancia
  - El script se ejecuta la primera vez que se inicia la instancia
- Se puede utilizar estratégicamente
  - Por ejemplo, reduzca el número de AMI personalizadas que crea y mantiene



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

20

Al crear las instancias de EC2, tiene la opción de pasar **datos de usuario** a la instancia. Los datos de usuario pueden automatizar la finalización de las instalaciones y configuraciones en el lanzamiento de la instancia. Por ejemplo, un script de datos de usuario podría aplicar parches y actualizar el sistema operativo de la instancia, buscar e instalar claves de licencia de software, o instalar software adicional.

En el script de datos de usuario de ejemplo, verá un script de shell **Linux Bash** de tres líneas sencillo. La primera línea indica que el script debe ser ejecutado por el shell de Bash. La segunda línea invoca la utilidad Yellowdog Updater, Modified (YUM), que se utiliza comúnmente en muchas distribuciones de Linux, como Amazon Linux, CentOS y Red Hat Linux, para recuperar software de un repositorio en línea e instalarlo. En la segunda línea del ejemplo, el comando indica a YUM que actualice todos los paquetes instalados a las últimas versiones conocidas por el repositorio de software al que está configurado para acceder. La tercera línea del script indica que se debe instalar la utilidad Wget. Wget es una utilidad común para descargar archivos de la Web.

En una instancia de **Windows**, el script de datos de usuario debe escribirse en un formato compatible con una ventana del símbolo del sistema (comandos por lotes) o con Windows PowerShell. Consulte la documentación de scripts de datos de usuario de Windows para obtener más detalles en <https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2-windows-user-data.html>.

Cuando se crea la instancia de EC2, **el script de datos de usuario se ejecuta con privilegios de usuario raíz** durante las fases finales del proceso de arranque. En las instancias de Linux, lo ejecuta el servicio cloud-init. En instancias de Windows, se ejecuta mediante la utilidad EC2Config o EC2Launch. **De forma predeterminada, los datos de usuario solo se ejecutan la primera vez que se inicia la instancia.** Sin embargo, si desea que el script de datos de usuario se ejecute cada vez que se inicie la instancia, puede crear un script de datos de usuario de archivos multiparte Multipurpose Internet Mail Extensions (MIME) (este proceso no es común). Consulte <https://aws.amazon.com/premiumsupport/knowledge-center/execute-user-data-ec2/> para más información.

## 6. Especificar almacenamiento

### Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:

1. AMI
2. Tipo de instancia
3. Configuración de red
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves

- Configurar el **volumen raíz**
  - Dónde está instalado el sistema operativo invitado
- Adjuntar **volúmenes de almacenamiento adicionales** (opcional)
  - Es posible que la AMI ya incluya más de un volumen
- Para cada volumen, especifique lo siguiente:
  - **Tamaño** del disco (en GB)
  - El **tipo de volumen**
    - Hay disponibles diferentes tipos de unidades de estado sólido (SSD) y unidades de disco duro (HDD)
  - Si el volumen se eliminará al finalizar la instancia
  - Si se debe utilizar el **cifrado**



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

21

Cuando lanza una instancia EC2, puede configurar las opciones de almacenamiento. Por ejemplo, puede configurar el tamaño del volumen raíz en el que está instalado el sistema operativo invitado. También puede adjuntar volúmenes de almacenamiento adicionales cuando lanza la instancia. Algunas AMI también están configuradas para lanzar más de un volumen de almacenamiento de forma predeterminada con el objetivo de proporcionar almacenamiento independiente del volumen raíz.

Para cada volumen que tendrá la instancia, puede especificar el tamaño de los discos, los tipos de volumen y si el almacenamiento se conservará si se termina la instancia. También puede especificar si se debe utilizar el cifrado.

## Opciones de almacenamiento de Amazon EC2

### • **Amazon Elastic Block Store (Amazon EBS):**

- Volúmenes de almacenamiento duraderos a nivel de bloque.
- Puede detener la instancia e iniciarla de nuevo y los datos seguirán ahí.

### • **Amazon EC2 Instance Store:**

- El almacenamiento efímero se proporciona en los discos que se encuentran conectados a la computadora anfitrión en el que se ejecuta la instancia de EC2.
- Si la instancia se detiene, los datos almacenados allí se eliminan.
- Otras opciones de almacenamiento (no para el volumen raíz):
  - Montaje de un sistema de **archivos de Amazon Elastic File System (Amazon EFS)**.
  - Conéctese a **Amazon Simple Storage Service (Amazon S3)**.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

22

**Amazon Elastic Block Store (Amazon EBS)** es un servicio de almacenamiento en bloque duradero fácil de usar y de alto rendimiento que se ha diseñado con el objetivo de utilizarse con Amazon EC2 para cargas de trabajo de alto rendimiento y de transacciones intensivas. Con Amazon EBS, puede elegir entre cuatro tipos de volúmenes diferentes para equilibrar el precio y el rendimiento óptimos. Puede cambiar los tipos de volúmenes o aumentar el tamaño del volumen sin interrumpir las aplicaciones críticas, de modo que pueda disponer de un almacenamiento rentable cuando lo necesite.

**Amazon EC2 Instance Store** proporciona almacenamiento efímero o temporal a nivel de bloques para su instancia. Este almacenamiento está ubicado en los discos que se adjuntan físicamente a la computadora host. Instance Store funciona bien cuando debe almacenar temporalmente información que cambia con frecuencia, como búferes, cachés, datos temporales y otro contenido temporal. También puede utilizar Instance Store para los datos que se replican en una flota de instancias, tales como un grupo de servidores web con balanceo de carga. Si se detienen las instancias, ya sea por un error del usuario o por un mal funcionamiento, se eliminarán los datos del almacén de instancias.

**Amazon Elastic File System (Amazon EFS)** proporciona un sistema de archivos de Network File System (NFS) elástico sencillo, escalable y completamente administrado para su uso con servicios en la nube de AWS y recursos en las instalaciones. Está preparado para escalar a petabytes bajo demanda sin interrumpir las aplicaciones. Crece y se reduce automáticamente a medida que

agrega y elimina archivos, lo que minimiza la necesidad de aprovisionar y administrar la capacidad para adaptarse al crecimiento.

**Amazon Simple Storage Service (Amazon S3)** es un servicio de almacenamiento de objetos que ofrece escalabilidad, disponibilidad de datos, seguridad y rendimiento. Puede almacenar y proteger cualquier cantidad de datos para diversos casos de uso, como sitios web, aplicaciones móviles, copia de seguridad y restauración, archivado, aplicaciones para empresas, dispositivos de Internet de las cosas (IoT) y análisis de big data.

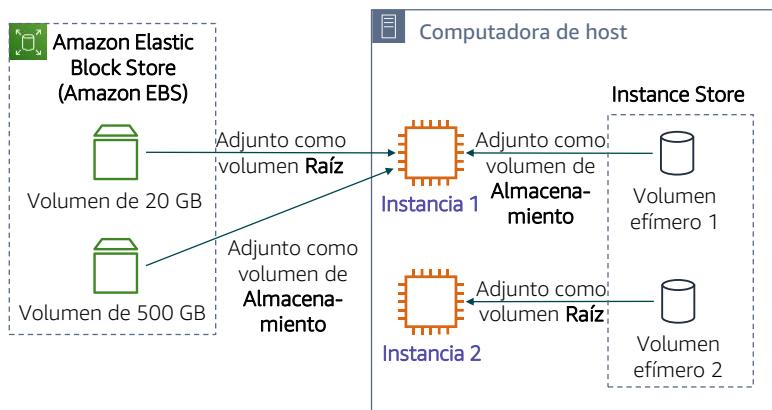
## Opciones de almacenamiento de ejemplo

- Características de la **instancia 1**:

- Tiene un tipo de *volumen raíz de Amazon EBS* para el sistema operativo.
- ¿Qué ocurre si la instancia se detiene y se vuelve a iniciar?

- Características de la **instancia 2**:

- Tiene un tipo de *volumen raíz de Instance Store* para el sistema operativo.
- ¿Qué ocurre si la instancia se detiene (debido a un error del usuario o a un mal funcionamiento del sistema)?



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

23

Aquí verá dos ejemplos de cómo se pueden configurar las opciones de almacenamiento para instancias de EC2.

El ejemplo de la **Instancia 1** muestra que el volumen raíz, que contiene el SO y posiblemente otros datos, se almacena en Amazon EBS. Esta instancia también tiene dos volúmenes adjuntos. Un volumen es un volumen de almacenamiento de EBS de 500 GB y el otro volumen es un volumen de Instance Store. **Si esta instancia se detuviera y luego se reiniciara**, el sistema operativo sobreviviría y cualquier dato almacenado en el volumen de Amazon EBS de 20 GB o en el volumen de Amazon EBS de 500 GB permanecería intacto. Sin embargo, cualquier dato almacenado en el volumen efímero 1 se perderá de forma permanente. Recuerde que Instance Store funciona bien para almacenar de forma temporal información que cambia con frecuencia tales como búferes, cachés, datos de Scratch y otro contenido temporal.

El **ejemplo de la Instancia 2** muestra que el volumen raíz se encuentra en un almacén de instancias (volumen efímero 2). **Una llamada a la API de Amazon EC2 no puede detener una instancia con un volumen raíz de Instance Store. Solo se puede terminar.** Sin embargo, podría detenerse desde el sistema operativo de la instancia (por ejemplo, emitiendo un comando de apagado), o podría detenerse debido a un error del sistema operativo o del disco, lo que provocaría la finalización de la instancia. Si se termina la instancia, se perderán todos los datos almacenados en el volumen efímero 2, incluido el SO. No podrá volver a iniciar la instancia. Por lo tanto, no confíe en Instance Store para guardar datos valiosos a largo plazo. En su lugar, utilice un

almacenamiento de datos más duradero, como Amazon EBS, Amazon EFS o Amazon S3.

Si una instancia se reinicia (de manera intencionada o involuntaria), los datos del volumen raíz del almacén de instancias persisten.

## 7. Agregar etiquetas

**Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:**

1. AMI
2. Tipo de instancia
3. Configuración de red
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. **Etiquetas**
8. Grupo de seguridad
9. Par de claves

- Una **etiqueta** es un rótulo que se puede asignar a un recurso de AWS.
  - Consta de una *clave* y un *valor opcional*.
- El etiquetado es la forma en que se pueden adjuntar **metadatos** a una instancia de EC2.
- Beneficios potenciales de las etiquetas: filtro, automatización, asignación de costos y control de acceso.

Ejemplo:

Key	(128 characters maximum)	Value	(256 characters maximum)
Name		WebServer1	
<b>Add another tag</b> (Up to 50 tags maximum)			



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

24

Una etiqueta es un rótulo que se asigna a un recurso de AWS. Cada etiqueta está formada por una clave y un valor opcional que usted mismo define. Las etiquetas le permiten clasificar los recursos de AWS tales como instancias de EC2 de distintas formas. Por ejemplo, puede etiquetar las instancias por finalidad, propietario o entorno.

El etiquetado es la forma en que se pueden adjuntar metadatos a una instancia de EC2.

Las claves y los valores de las etiquetas distinguen entre mayúsculas y minúsculas. Por ejemplo, una etiqueta de uso común para las instancias de EC2 es una clave de etiqueta denominada Name (Nombre) y un valor de etiqueta que describe la instancia, como My Web Server (Mi servidor web). La etiqueta Name (Nombre) se muestra de forma predeterminada en la página Instances (Instancias) de la consola de Amazon EC2. Sin embargo, si crea una clave llamada nombre (con n minúscula), no aparecerá en la columna **Name** (Nombre) de la lista de instancias (aunque seguirá apareciendo en el panel de detalles de la instancia en la pestaña **Tags** [Etiquetas]).

Es recomendable desarrollar estrategias de etiquetado. Utilizar un conjunto coherente de claves de etiqueta facilita la administración de los recursos. También puede buscar y filtrar los recursos según las etiquetas que agregue. Para más información, consulte:  
<https://d1.awsstatic.com/whitepapers/aws-tagging-best-practices.pdf>.

## 8. Configuración del grupo de seguridad

Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:

1. AMI
2. Tipo de instancia
3. Configuración de red
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves

- Un **grupo de seguridad** es un **conjunto de reglas de firewall** que controlan el tráfico a la instancia.
  - Existe *frente* del SO invitado de la instancia.
- Cree **reglas** que especifiquen la **fuente** y **los puertos** que pueden utilizar las comunicaciones de red.
  - Especifique el número de **puerto** y el **protocolo**, como Protocolo de control de transmisión (TCP), Protocolo de datagramas de usuario (UDP) o Protocolo de mensajes de control de Internet (ICMP).
  - Especifique la **fuente** (por ejemplo, una dirección IP u otro grupo de seguridad) a la que se le permita utilizar la regla.

Regla de ejemplo:

Type	Protocol	Port Range	Source
SSH	TCP	22	My IP 72.21.198.67/32

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

25

Un **grupo de seguridad** funciona como un firewall virtual que controla el tráfico de una o varias instancias. Al lanzar una instancia, puede especificar uno o varios grupos de seguridad. De lo contrario, se utiliza el grupo de seguridad predeterminado.

Puede agregar **reglas** a cada grupo de seguridad. Las reglas permiten el tráfico hacia sus instancias asociadas o desde estas. Las reglas de un grupo de seguridad se pueden modificar en cualquier momento; las nuevas reglas se aplican automáticamente a todas las instancias que estén asociadas al grupo de seguridad. Cuando AWS decide si permite que el tráfico llegue a una instancia, se evalúan todas las reglas de todos los grupos de seguridad asociados a la instancia. Cuando lanza una instancia en una virtual private cloud (VPC), debe crear un nuevo grupo de seguridad o utilizar uno que ya exista en esa VPC. Después de lanzar una instancia, puede cambiar sus grupos de seguridad.

Al **definir una regla**, puede especificar la fuente permitida de la comunicación de red (reglas de entrada) o el destino (reglas de salida). El **origen** puede ser una dirección IP, un intervalo de direcciones IP, otro grupo de seguridad, un punto de enlace de la VPC de gateway o cualquier lugar (lo que significa que se permitirán todas las fuentes). De forma predeterminada, un grupo de seguridad **incluye una** regla de salida que permite todo el **tráfico** saliente. Puede quitar la **regla** y agregar **reglas salientes** que solo permitan tráfico saliente específico. Si un **grupo de seguridad no tiene** reglas de salida, no se permite el tráfico saliente que se origina en la instancia.

**En la regla de ejemplo,** la regla permite el tráfico de Secure Shell (SSH) a través del puerto 22 del Protocolo de control de transmisión (TCP) si la fuente de la solicitud es My IP (Mi IP). La dirección IP My IP se calcula al definir la regla mediante la determinación de la dirección IP que utiliza para conectarse a la nube de AWS.

Las listas de control de acceso a la red (ACL de red) también se pueden utilizar como firewalls para proteger las subredes de una VPC.

**Para accesibilidad:** captura de pantalla de la consola Elastic Compute Cloud donde puede definir una regla de grupo de seguridad. Muestra una regla con tipo SSH, protocolo TCP, rango de puerto 22, origen My IP (Mi IP) y un bloque CIDR que muestra un ejemplo de Mi dirección IP. **Fin de la descripción de accesibilidad.**

## 9. Identificar o crear el par de claves

### Elecciones que se realizaron mediante el asistente de lanzamiento de instancias:

1. AMI
2. Tipo de instancia
3. Configuración de red
4. Rol de IAM
5. Datos de usuario
6. Opciones de almacenamiento
7. Etiquetas
8. Grupo de seguridad
9. Par de claves

- En el lanzamiento de la instancia, se especifica un par de claves existente o se crea un nuevo par de claves.



- Un **par de claves** consiste en:

- una **clave pública** que AWS almacena.
  - un archivo de **clave privada** que usted almacena.

- Permite conexiones seguras a la instancia.



- Para **AMI de Windows**:

- utilice la clave privada con el fin de obtener la contraseña de administrador que necesita para iniciar sesión en su instancia.

- Para **AMI de Linux**:

- utilice la clave privada a fin de emplear SSH para conectarse de forma segura a su instancia.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

26

Después de especificar todas las configuraciones necesarias para lanzar una instancia de EC2 y después de personalizar cualquier configuración opcional del Launch Wizard de Elastic Compute Cloud, se abre la **ventana Review Instance Launch** (Revisar lanzamiento de instancias). Si selecciona **Launch** (Lanzar), **un cuadro de diálogo le pedirá que elija un par de claves existente, que proceda sin un par de claves o que cree un nuevo par de claves antes de elegir Launch Instances** (Lanzar instancias) y crear la instancia de EC2.

Amazon EC2 utiliza la criptografía de clave pública para cifrar y descifrar la información de inicio de sesión. La tecnología utiliza una **clave pública** para cifrar un dato y luego el destinatario usa la clave privada para descifrar los datos. El conjunto de clave pública y clave privada se denomina **par de claves**. La criptografía de clave pública le permite acceder de forma segura a sus instancias mediante una clave privada en lugar de una contraseña.

Cuando lanza una instancia, especifica un par de claves. Puede especificar un par de claves existente o uno nuevo que cree en el lanzamiento. Si crea un nuevo par de claves, descárguelo y guárdelo en un lugar seguro. Esta oportunidad es la única posibilidad de guardar el archivo de clave privada.

Para conectarse a una instancia de Windows, utilice la clave privada a fin de obtener la contraseña de administrador y, a continuación, inicie sesión en el escritorio de Windows de la instancia de EC2 mediante el Protocolo de escritorio remoto (RDP). Para establecer una conexión

SSH desde una máquina Windows a una instancia de EC2, puede utilizar una herramienta como PuTTY, que requerirá la misma clave privada.

Con las instancias de **Linux**, en el momento de arranque, se coloca el contenido de la **clave pública** en la instancia. Se crea una entrada en `~/.ssh/authorized_keys`. Para iniciar sesión en su instancia de Linux (por ejemplo, mediante SSH), debe proporcionar la **clave privada** cuando establezca la conexión.

## Vista de la consola de Amazon EC2 de una instancia EC2 en ejecución

The screenshot shows the AWS EC2 Management console with the following details:

**Instance Details:**

- Name: i-092b6f3efba959a53
- Instance Type: t2.micro
- Instance State: running
- Status Checks: Initializing
- Public DNS (IPv4): ec2-54-159-171-63.compute-1.amazonaws.com
- IPv4 Public IP: 54.159.171.63

**Description Tab (Selected):**

Attribute	Value
Instance ID	i-092b6f3efba959a53
Instance state	running
Instance type	t2.micro
Elastic IPs	None
Availability zone	us-east-1c
Security groups	launch-wizard-1, view inbound rules, view outbound rules
Scheduled events	No scheduled events
AMI ID	amzn2-ami-hvm-2.0.20190623.1-x86_64-gp2 (ami-0b69ea66ff7391e80)
Platform	None
Public DNS (IPv4)	ec2-54-159-171-63.compute-1.amazonaws.com
IPv4 Public IP	54.159.171.63
IPv6 IPs	-
Private DNS	ip-172-31-82-44.ec2.internal
Private IPs	172.31.82.44
Secondary private IPs	-
VPC ID	vpc-e4e9859e
Subnet ID	subnet-c22277fc
Network interfaces	eth0

**Footer:**

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

27

Después de elegir **Launch Instances** (Iniciar instancias) y luego elegir **View Instances** (Ver instancias), se le presentará una pantalla similar al ejemplo.

Muchas de las configuraciones especificadas durante el inicio se encuentran en el panel **Description** (Descripción).

La información sobre la instancia disponible incluye información sobre la dirección IP y la dirección DNS, el tipo de instancia, el ID de instancia único que se asignó a la instancia, el ID de AMI de la AMI que utilizó para lanzar la instancia, el ID de VPC, el ID de subred y mucho más.

Muchos de estos detalles proporcionan hipervínculos que puede elegir para obtener más información sobre los recursos que son relevantes para la instancia de EC2 que lanzó.

## Otra opción: lanzar una instancia de EC2 con AWS Command Line Interface

- Las instancias EC2 también se pueden crear mediante programación.



Interfaz de la línea de comandos de AWS (AWS CLI)

- En este ejemplo, se muestra lo sencillo que puede ser el comando.
  - Este comando supone que el par de claves y el grupo de seguridad ya existen.
  - Se podrían especificar más opciones. Consulte la [Referencia de comandos de AWS CLI](#) para obtener más de detalles.

### Comando de ejemplo:

```
aws ec2 run-instances \
--image-id ami-1a2b3c4d \
--count 1 \
--instance-type c3.large \
--key-name MyKeyPair \
--security-groups MySecurityGroup \
--region us-east-1
```



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

28

También puede lanzar instancias EC2 mediante programación, a través de AWS Command Line Interface (AWS CLI) o uno de los kits de desarrollo de software (SDK) de AWS.

En el comando de ejemplo de AWS CLI, verá un único comando que especifica la información mínima necesaria para lanzar una instancia. El comando incluye la siguiente información:

- aws: especifica una invocación de la utilidad de línea de comandos aws.
- ec2: especifica una invocación del comando de servicio ec2.
- run-instances: es el subcomando que se invoca.

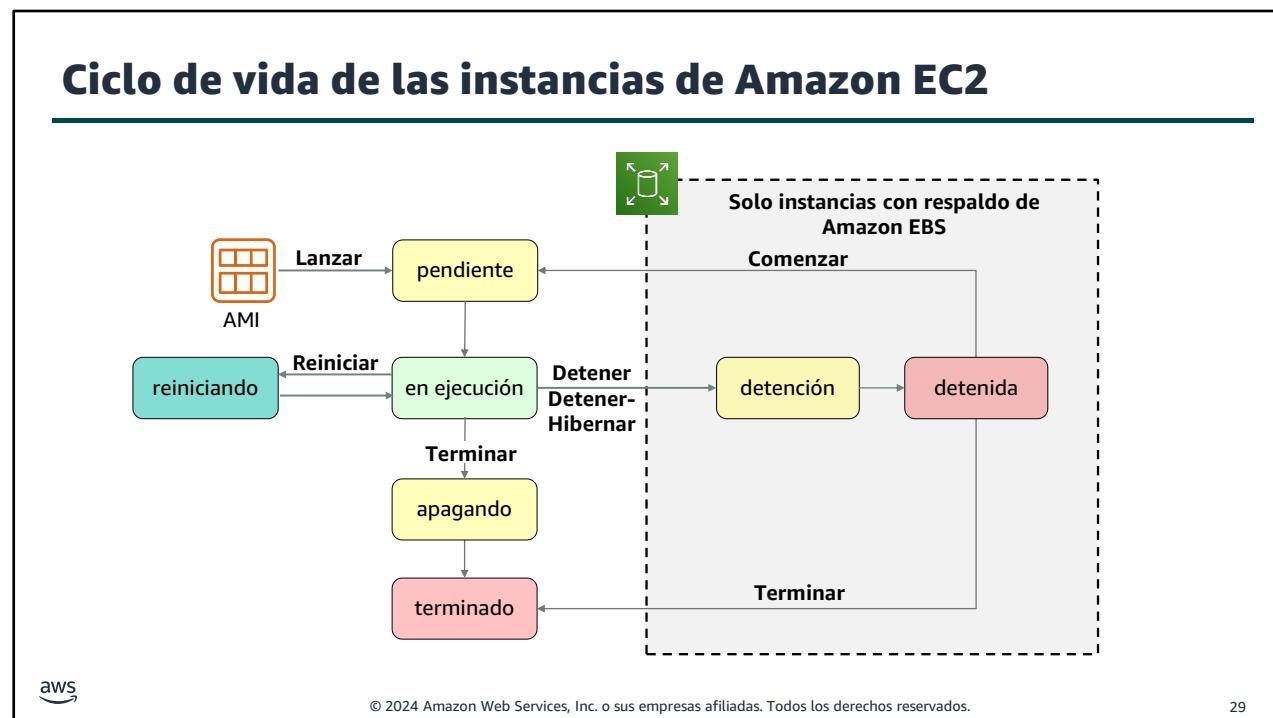
El resto del comando especifica varios parámetros, entre ellos:

- image-id: este parámetro va seguido de un ID de AMI. Todas las AMI tienen un ID de AMI único.
- count: puede especificar más de uno.
- instance-type: puede especificar el tipo de instancia que desea crear (por ejemplo, una instancia c3.grande).
- key-name: en el ejemplo, supongamos que MyKeyPair ya existe.
- security-groups: en este ejemplo, supongamos que MySecurityGroup ya existe.
- region: las AMI existen en una región de AWS, por lo que debe especificar la región en la que AWS CLI encontrará la AMI y lanzará la instancia de EC2.

El comando debería crear de manera correcta una instancia EC2 en los siguientes casos:

- Si el comando se ha formado correctamente
- Si ya existen los recursos que necesita el comando
- Si tiene permisos suficientes para ejecutar el comando
- Si tiene capacidad suficiente en la cuenta de AWS

Si el comando se ejecuta correctamente, la API responde al comando con el ID de instancia y otros datos relevantes para que la aplicación los utilice en solicitudes de API posteriores.



Aquí, puede ver el ciclo de vida de una instancia. Las flechas muestran las **acciones** que puede realizar y los cuadros muestran el **estado** en el que entrará la instancia luego de esa acción. Una instancia puede tener uno de los siguientes estados:

- **Pendiente**: cuando una instancia se inicia por primera vez desde una AMI o cuando se inicia una instancia detenida, entra en estado pendiente una vez que la instancia se inicia y se implementa en una computadora host. El tipo de instancia especificado en el inicio determina el hardware anfitrión de la instancia.
- **En ejecución**: cuando la instancia se ha iniciado completamente y se encuentra lista, sale del estado pendiente y entra en estado de en ejecución. Puede conectarse a su instancia en ejecución a través de Internet.
- **Reinicio**: AWS recomienda reiniciar una instancia mediante la consola de Amazon EC2, la CLI de AWS o los SDK de AWS en lugar de invocar un reinicio desde el sistema operativo (SO) invitado. Una instancia reiniciada permanece en el mismo host físico, mantiene el mismo nombre DNS público y dirección IP pública y, si tiene **volúmenes de almacenamiento** de instancias, conserva los datos de esos volúmenes.
- **Apagando**: este estado es un estado intermedio entre en ejecución y terminada.
- **Terminada**: una instancia terminada permanece en la consola de Amazon EC2 durante algún tiempo antes de que se elimine la máquina virtual. Sin embargo, no puede conectarse a una instancia terminada ni recuperarla.
- **Deteniendo**: las instancias respaldadas por Amazon EBS se pueden detener. Entran en el estado de en proceso de detención antes de alcanzar el estado de detenida por completo.

- **Detenida:** Una instancia detenida no incurrirá en los mismos costos que una instancia en ejecución. Al iniciar una instancia detenida, se vuelve a poner en estado pendiente, lo que la mueve a un nuevo equipo host.

## Consideré utilizar una dirección IP elástica

- **Reiniciar** una instancia *no* cambiará ninguna dirección IP ni nombre de host DNS.
- ¿Qué ocurre si la instancia se **detiene** y se vuelve a **iniciar**?
  - La dirección IPv4 *pública* y el nombre de host DNS *externo* se modificarán.
  - La dirección IPv4 *privada* y el nombre de host DNS interno *no* se modifican.



Dirección  
IP elástica

- Si necesita una dirección IP pública persistente:
  - puede asociar una **dirección IP elástica** con la instancia.
- Características de la dirección IP elástica:
  - se puede asociar con instancias en la Región según sea necesario.
  - permanece asignado a su cuenta hasta que decida liberarlo.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

31

Una **dirección IP pública** es una dirección IPv4 a la que se puede acceder desde Internet. Cada instancia que recibe una dirección IP pública también recibe un nombre de host DNS externo. Por ejemplo, si la dirección IP pública asignada a la instancia es 203.0.113.25, entonces el nombre de host DNS externo podría ser ec2-203-0-113-25.compute-1.amazonaws.com.

Si especifica que se debe asignar una dirección IP pública a su instancia, se asigna desde el grupo de direcciones IPv4 públicas de AWS. La dirección IP pública no está asociada con su cuenta de AWS. Cuando una dirección IP pública se desasocia de su instancia, se devuelve al grupo de direcciones IPv4 públicas y no podrá especificar que desea reutilizarla. AWS publica la dirección IP pública de su instancia cuando la instancia se detiene o termina. La instancia detenida recibe una nueva dirección IP pública cuando se inicia de nuevo.

Si necesita una dirección IP pública persistente, es posible que desee asociar una **dirección IP elástica** con la instancia. Para asociar una dirección IP elástica, primero debe asignar una nueva dirección IP elástica en la región donde existe la instancia. Luego de la asignación de la dirección IP elástica, puede asociar la dirección IP elástica con una instancia de EC2.

De forma predeterminada, todas las cuentas de AWS están limitadas a cinco (5) direcciones IP elásticas por región, ya que las direcciones de Internet públicas (IPv4) son un recurso público escaso. Sin embargo, este es un límite suave y usted puede solicitar un aumento del límite (que podría aprobarse).

## Metadatos de la instancia de EC2

- **Los metadatos de la instancia** son datos sobre su instancia.
- Mientras esté conectado a la instancia, podrá visualizarla:
  - En un navegador: <http://169.254.169.254/latest/meta-data/>
  - En una ventana de terminal: `curl http://169.254.169.254/latest/meta-data/`
- **Ejemplos de valores recuperables:**
  - Dirección IP pública, dirección IP privada, nombre de host público, ID de instancia, grupos de seguridad, región, zona de disponibilidad.
  - También se puede acceder a cualquier dato de usuario especificado en el lanzamiento de la instancia en: <http://169.254.169.254/latest/user-data/>
- **Se puede utilizar para configurar o administrar una instancia en ejecución.**
  - Por ejemplo, cree un script de configuración que lea los metadatos y los utilice para configurar aplicaciones o ajustes del sistema operativo.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

32

Los metadatos de la instancia son datos sobre su instancia. Mientras esté conectado a la instancia, puede visualizarla. Para acceder a él en un navegador, vaya a la siguiente URL: <http://169.254.169.254/latest/meta-data/>. Los datos también se pueden leer mediante programación, como desde una ventana de terminal que tenga la utilidad curl.

En la ventana de terminal, ejecute `curl http://169.254.169.254/latest/meta-data/` para recuperarlo. La dirección IP `169.254.169.254` es una dirección de enlace local y solo es válida desde la instancia.

Los metadatos de la instancia proporcionan gran parte de la misma información sobre la instancia en ejecución que puede encontrar en la Consola de administración de AWS. Por ejemplo, puede descubrir la dirección IP pública, dirección IP privada, nombre de host público, ID de instancia, grupos de seguridad, región, zona de disponibilidad y más.

También se puede acceder a cualquier dato de usuario que se especifique en el inicio de la instancia en la siguiente URL: <http://169.254.169.254/latest/user-data>.

Los metadatos de la instancia de EC2 se puede utilizar para configurar o administrar una instancia en ejecución. Por ejemplo, puede crear un script de configuración que lea los metadatos y los utilice para configurar aplicaciones o ajustes del sistema operativo.

## Se utiliza CloudWatch para la supervisión

- Utilice **Amazon CloudWatch** para monitorear instancias de EC2
  - Proporciona métricas en tiempo real
  - Proporciona gráficos en la pestaña **Monitoreo** de la consola Amazon EC2 que puede ver
  - Mantiene 15 meses de datos históricos

### Supervisión básica

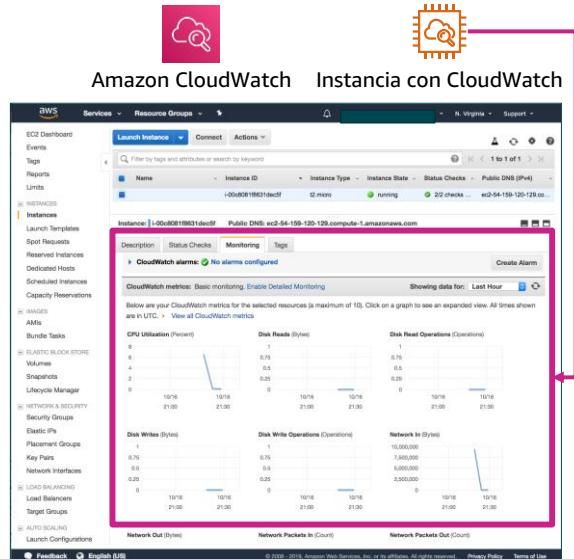
- Predeterminado, sin costo adicional
- Datos de métricas enviados a CloudWatch cada 5 minutos

### Monitoreo detallado

- Tarifa mensual fija para siete métricas preseleccionadas
- Datos de métricas entregados cada 1 minuto



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.



33

Puede monitorear las instancias mediante Amazon CloudWatch, que recopila y procesa datos sin procesar de Amazon EC2 en métricas legibles casi en tiempo real. Estas estadísticas se registran durante un periodo de 15 meses, de forma que pueda acceder a información histórica y obtener una mejor perspectiva sobre el rendimiento de su aplicación web o servicio.

De forma predeterminada, Amazon EC2 proporciona **supervisión básica**, que envía datos de métrica a CloudWatch en periodos de 5 minutos. Para enviar datos de métricas para su instancia a CloudWatch en períodos de 1 minuto, puede habilitar la **supervisión detallada** de la instancia. Para más información, consulte Habilite o deshabilite la supervisión detallada para sus instancias en <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-cloudwatch-new.html>.

La consola de Amazon EC2 muestra una serie de gráficos basados en los datos sin procesar de Amazon CloudWatch. De acuerdo a sus necesidades, es posible que prefiera obtener datos para sus instancias desde Amazon CloudWatch en lugar de a través de los gráficos de la consola. De forma predeterminada, Amazon CloudWatch no proporciona métricas de RAM para instancias de EC2, aunque es una opción que puede configurar si desea que CloudWatch recopile esos datos.

## Conclusiones importantes de la Sección 2



- **Amazon EC2** le permite ejecutar **máquinas virtuales** Microsoft Windows y Linux en la nube.
- Lanza **instancias de EC2** desde una plantilla **AMI** en una VPC en su cuenta.
- Puede elegir entre muchos **tipos de instancias**. Cada tipo de instancia ofrece diferentes combinaciones de CPU, RAM, almacenamiento y capacidades de red.
- Puede configurar **grupos de seguridad** para controlar el acceso a las instancias (especifique los puertos y las fuentes permitidos).
- **Los datos del usuario** le permiten especificar un script para ejecutar la primera vez que se inicia una instancia.
- Solo **las instancias que tienen el respaldo de Amazon EBS pueden detenerse**.
- Puede utilizar **Amazon CloudWatch** para capturar y revisar métricas en instancias de EC2.

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

34

Entre los aprendizajes clave de esta sección del módulo, se incluyen los siguientes:

- Amazon EC2 le permite ejecutar máquinas virtuales Microsoft Windows y Linux en la nube.
- Lanza instancias de EC2 desde una plantilla AMI en una VPC en su cuenta.
- Puede elegir entre muchos tipos de instancias. Cada tipo de instancia ofrece diferentes combinaciones de CPU, RAM, almacenamiento y capacidades de red.
- Puede configurar grupos de seguridad para controlar el acceso a las instancias (especifique los puertos y las fuentes permitidos).
- Los datos del usuario le permiten especificar un script para ejecutar la primera vez que se inicia una instancia.
- Las instancias que tienen el respaldo de Amazon EBS pueden detenerse.
- Puede utilizar Amazon CloudWatch para capturar y revisar métricas en instancias de EC2.

## Demostración registrada de Amazon EC2



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

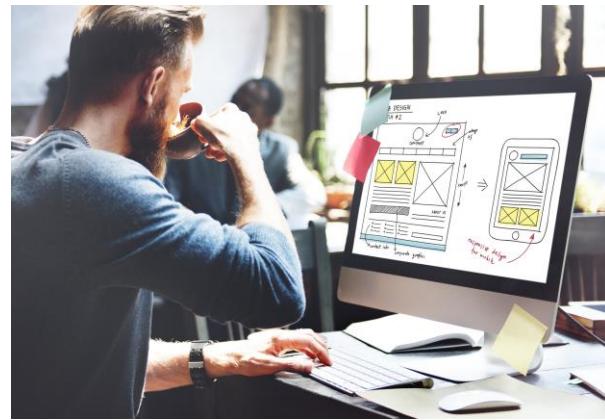
35

Ahora tómese el tiempo de mirar la demostración de Elastic Compute Cloud en [https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/ILT-TF-100-ACFNDS-20-EN/Module\\_6\\_EC2+v2.0.mp4](https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/ILT-TF-100-ACFNDS-20-EN/Module_6_EC2+v2.0.mp4). La grabación dura un poco más de 3 minutos y refuerza muchos de los conceptos que mencionamos en esta sección del módulo.

La demostración muestra:

- Cómo utilizar la consola de administración de AWS para lanzar una instancia de Amazon EC2, con toda la configuración de instancias predeterminada aceptada.
- Cómo conectarse a la instancia de Windows mediante un cliente de Escritorio remoto y el par de claves que se identificó durante el inicio de la instancia para descifrar la contraseña de Windows para iniciar sesión.
- Cómo terminar la instancia cuando ya no es necesaria.

## Laboratorio 3: Introducción a Amazon EC2



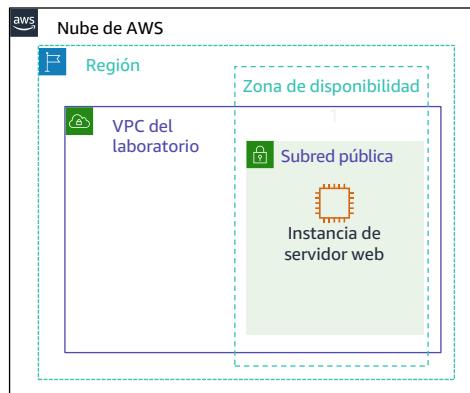
© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

36

Laboratorio introductorio 3: Introducción a Amazon EC2. En este laboratorio, se proporciona información general básica sobre iniciar, modificar el tamaño, administrar y supervisar una instancia de Amazon EC2.

## Tarea 3: Situación

En este laboratorio, iniciará y configurará su primera máquina virtual que se ejecuta en Amazon EC2.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

37

Laboratorio introductorio 3: Introducción a Amazon EC2.

En este laboratorio, iniciará y configurará una máquina virtual que se ejecuta en Amazon EC2.

## Laboratorio 3: Tareas

- Tarea 1: lanzar una instancia de Amazon EC2
- Tarea 2: supervisar la instancia
- Tarea 3: actualizar el grupo de seguridad y acceder al servidor web
- Tarea 4: modificar el tamaño de la instancia, tipo de instancia y volumen de EBS
- Tarea 5: explorar los límites de Elastic Compute Cloud
- Tarea 6: probar la protección contra terminación



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

38

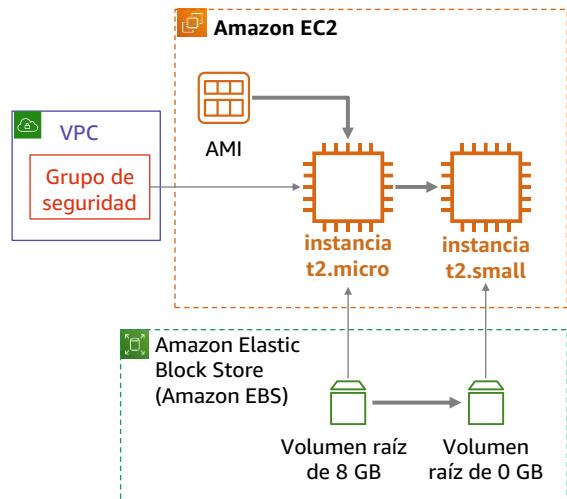
En este laboratorio práctico podrá:

- lanzar una instancia de Amazon EC2
- supervisar la instancia
- actualizar el grupo de seguridad y acceder al servidor web
- modificar el tamaño de la instancia, tipo de instancia y volumen de EBS
- explorar los límites de EC2
- probar la protección contra terminación

## Laboratorio 3: Producto final

Al final del laboratorio, habrás:

1. Lanzado una instancia que está configurada como servidor web
2. Visto el registro del sistema de instancia
3. Re-configurado un grupo de seguridad
4. Modificado el tipo de instancia y el tamaño del volumen raíz



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

39

Al final del laboratorio, habrás:

1. Lanzado una instancia que está configurada como servidor web
2. Visto el registro del sistema de instancia
3. Re-configurado un grupo de seguridad
4. Modificado el tipo de instancia y el tamaño del volumen raíz



A blue cup of coffee with a wooden spoon containing coffee beans.

**Comenzar laboratorio 3:  
Introducción  
a Amazon EC2**

~ 35 minutos

aws

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

40

Es momento de comenzar con el laboratorio.

## Análisis posterior del laboratorio: Aprendizajes clave



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

41

El instructor puede optar por dirigir una conversación sobre los aprendizajes clave del laboratorio después de que lo haya completado.

## Actividad: Amazon EC2



Foto de Pixabay de Pexels.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

42

En esta actividad dirigida por un educador, analizará las ventajas y desventajas de utilizar Amazon EC2 frente a un servicio administrado como Amazon Relational Database Service (Amazon RDS).

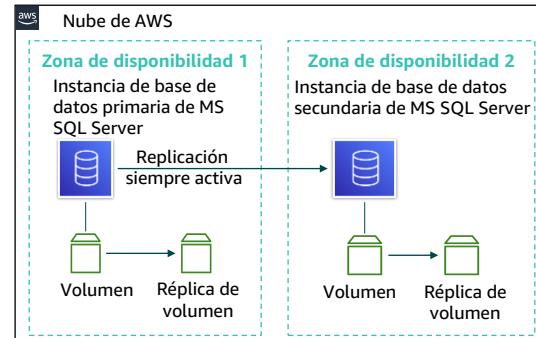
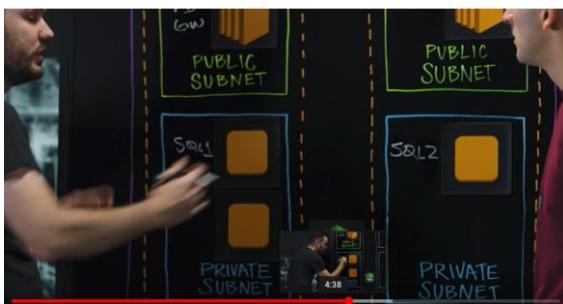
## Actividad: Recopilar información



Amazon EC2



Amazon RDS



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

43

El objetivo de esta actividad es demostrar que comprende las diferencias entre crear una implementación que utiliza Amazon EC2 y utilizar un servicio completamente administrado, como Amazon RDS, para implementar su solución. Al final de esta actividad, debe estar preparado para analizar las ventajas y desventajas de implementar Microsoft SQL Server en Amazon EC2 versus implementarlo en Amazon RDS.

El educador le pedirá lo siguiente:

1. Mirar un video de 8 minutos en [https://www.youtube.com/watch?v=UYy-UeQ29jo&did=ta\\_card&trk=ta\\_card](https://www.youtube.com/watch?v=UYy-UeQ29jo&did=ta_card&trk=ta_card) que explica los beneficios de implementar Microsoft SQL Server en Amazon EC2 mediante el uso de AWS Quick Start. Se le anima a tomar notas.
2. Leer una publicación de blog en <https://aws.amazon.com/blogs/publicsector/the-scoop-on-moving-your-microsoft-sql-server-to-aws/> sobre los beneficios de ejecutar Microsoft SQL Server en Amazon RDS. Se le anima a tomar notas nuevamente.
3. Participar en la conversación en clase sobre las preguntas planteadas en la siguiente diapositiva.

## Actividad: Verifique sus conocimientos

1. Entre Amazon EC2 o Amazon RDS, ¿cuál proporciona un servicio administrado? ¿Qué significa *servicio administrado*?
  - **RESPUESTA:** Amazon RDS proporciona un servicio administrado. Amazon RDS se encarga del aprovisionamiento, la instalación y la aplicación de parches, los respaldos automatizados, la restauración de instantáneas desde momentos puntuales, la alta disponibilidad y la supervisión.
2. Mencione al menos una ventaja de implementar Microsoft SQL Server en Amazon EC2 en lugar de Amazon RDS.
  - **RESPUESTA:** Amazon EC2 ofrece control total sobre cada configuración, el sistema operativo y la pila de software.
3. ¿Qué ventaja ofrece el Quick Start sobre una instalación manual en Amazon EC2?
  - **RESPUESTA:** Quick Start es una arquitectura de referencia con prácticas recomendadas comprobadas integradas en el diseño.
4. ¿Qué opción de implementación ofrece el mejor enfoque para todos los casos prácticos?
  - **RESPUESTA:** Ninguna. La opción de implementación correcta depende de sus necesidades específicas.
5. ¿Qué enfoque cuesta más: usar Amazon EC2 o usar Amazon RDS?
  - **RESPUESTA:** Depende. Administrar la implementación de la base de datos en Amazon EC2 requiere más tiempo y supervisión del cliente. Si el tiempo es su prioridad, entonces Amazon RDS podría resultar menos costoso. Si tiene conocimiento experto interno, Amazon EC2 podría resultar más rentable.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

44

El educador dirigirá la clase en una conversación a medida que se revele cada pregunta. Luego, el educador mostrará las respuestas sugeridas por escrito y usted podrá analizar estos puntos más en detalle.

En relación con la **pregunta 5**, utilice estas páginas para ver los precios de las instancias de RDS y de EC2 con las características enumeradas:

**Amazon RDS para SQL Server:** <https://aws.amazon.com/rds/sqlserver/pricing/>

- Instancia: instancia estándar (Single-AZ)
- Tamaño de instancia: db.m5.large
- Región: Este de EE. UU. (Ohio)
- Precios: instancia bajo demanda

**Amazon EC2:** <https://aws.amazon.com/ec2/pricing/>

- Instancia: instancia de Windows
- Tamaño de instancia: m5.large
- Región: Este de EE. UU. (Ohio)
- Precios: instancia bajo demanda

Tenga en cuenta que una instancia de Amazon RDS tiene un costo por hora superior al de una instancia de EC2 de tamaño similar. Al considerar el costo, no olvide agregar el costo de la mano de obra. Por ejemplo, tenga en cuenta que con una implementación estándar Single-AZ de

Amazon RDS (que es la base de la referencia de precios de ejemplo) se proporcionan respaldos automatizados. Con Amazon RDS, si un componente de la instancia de base de datos falla y se requiere una operación de restauración iniciada por el usuario, tendrá un respaldo restaurable que podría usar. Si ejecuta la base de datos en Amazon EC2, puede configurar un procedimiento de respaldo igualmente sólido para Microsoft SQL Server. No obstante, se necesitaría tiempo, conocimiento y habilidad técnica para crear la solución. Además deberá configurar de antemano la solución antes de encontrar la situación en la que la necesite. Por estos motivos, cuando tiene en cuenta las necesidades de sus implementaciones de manera integral, es posible que descubra que usar Amazon RDS resulta menos costoso que usar Amazon EC2. Sin embargo, si su personal está compuesto por administradores de bases de datos capacitados (y también tiene requisitos de implementación muy específicos que hacen que sea preferible tener control total sobre todos los aspectos de la implementación), podría utilizar Amazon EC2. En este caso, es posible que Amazon EC2 sea la solución más rentable.

## Sección 3: Optimización de costos de Amazon EC2

Módulo 6: Cómputo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Sección introductoria 3: Optimización de costos de Amazon EC2.

## Modelos de precios de Amazon EC2

### Instancias bajo demanda

- Pago por hora
- No asume compromisos a largo plazo.
- Apto para el [nivel gratuito de AWS](#).

### Servidores dedicados

- Un servidor físico con capacidad de instancias de EC2 totalmente dedicado a su uso.

### Instancias dedicadas

- Instancias que se ejecutan en una VPC en hardware dedicado a un solo cliente.

### Instancias reservadas

- Pago total, parcial o sin pago inicial para la instancia que reserva.
- Descuento en el cargo horario por uso de dicha instancia.
- Plazo de 1 o 3 años.

### Instancias reservadas programadas

- Adquiera una reserva de capacidad que esté disponible en un horario recurrente que especifique.
- Plazo de 1 año.

### Instancias de spot

- Las instancias se ejecutan siempre que estén disponibles y su puja sea superior al precio de las instancias de spot.
- AWS puede interrumpirlas con una notificación de 2 minutos.
- Las opciones de interrupción incluyen terminación, parada o hibernación.
- Los precios pueden ser de forma significativa menos costosos en comparación con las instancias bajo demanda
- Una buena elección cuando tiene flexibilidad en cuanto a cuándo se pueden ejecutar sus aplicaciones.

La **facturación por segundo** solo está disponible para las instancias bajo demanda, las instancias reservadas y las instancias de spot que ejecutan Amazon Linux o Ubuntu.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

46

Amazon ofrece diferentes modelos de precios entre los cuales elegir cuando desee ejecutar instancias EC2.

- **La facturación por segundo** solo está disponible para las instancias bajo demanda, las instancias reservadas y las instancias de spot que ejecutan Amazon Linux o Ubuntu.
- **Las instancias bajo demanda** son aptas para la capa gratuita de AWS (<https://aws.amazon.com/free/>). Tienen el costo inicial más bajo y la mayor flexibilidad. No es necesario efectuar contratos por adelantado ni compromisos a largo plazo. Son una buena opción para aplicaciones con cargas de trabajo a corto plazo, con picos de demanda o impredecibles.
- **Los alojamientos dedicados** son servidores físicos con capacidad de instancia dedicada para su uso. Le permiten utilizar sus licencias de software existentes por socket, por núcleo o por máquina virtual, como Microsoft Windows o Microsoft SQL Server.
- **Las instancias dedicadas** son instancias de EC2 que se ejecutan en una nube virtual privada (VPC) en un hardware dedicado a un único cliente. Se aíslan físicamente de las instancias que pertenecen a otras cuentas de AWS a nivel del hardware que es host.
- **Las instancias reservadas** le permiten reservar capacidad de cómputo durante un año o tres años, con costos de funcionamiento menores por hora. El precio de uso con descuento se fija mientras sea propietario de la instancia reservada. Si espera un uso uniforme e intensivo, pueden proporcionar ahorros sustanciales en comparación con las instancias bajo demanda.
- **Las instancias reservadas programadas** le permiten comprar reservas de capacidad que se

repiten diaria, semanal o mensualmente, con una duración especificada, por un periodo de 1 año. Paga por el tiempo en que se programan las instancias, incluso si no las utiliza.

- **Las** instancias de spot le permiten pujar por instancias EC2 no utilizadas, lo que puede reducir los costos. El precio por hora de una instancia de spot fluctúa en función de la oferta y la demanda. Su instancia de spot se ejecuta siempre que su puja supere el precio actual del mercado.

## Modelos de precios de Amazon EC2: beneficios



Instancias bajo demanda	Instancias de spot	Instancias reservadas	Servidores dedicados
<ul style="list-style-type: none"><li>Ofrece bajo costo y flexibilidad</li></ul>	<ul style="list-style-type: none"><li>Carga de trabajo dinámica y a gran escala</li></ul>	<ul style="list-style-type: none"><li>Ofrece previsibilidad, lo que garantiza que la capacidad de cómputo esté disponible cuando sea necesario</li></ul>	<ul style="list-style-type: none"><li>Permite ahorrar en costos de licencia</li><li>Ayuda a cumplir los requisitos normativos y de conformidad</li></ul>



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

47

Cada modelo de precios de Amazon EC2 proporciona un conjunto diferente de beneficios.

**Las instancias bajo demanda ofrecen la mayor flexibilidad, sin contratos a largo plazo y con tarifas bajas.**

**Las instancias de spot ofrecen gran escala a un precio con un descuento significativo.**

**Las instancias reservadas son una buena opción si tiene necesidades de cómputo predecibles o de estado estable (por ejemplo, una instancia que sabe que desea seguir ejecutando casi todo el tiempo o durante meses o años).**

**Los servidores dedicados son una buena opción cuando existen restricciones de licencia para el software que desea ejecutar en Amazon EC2 o cuando tiene requisitos normativos o de cumplimiento específicos que le impiden utilizar las demás opciones de implementación.**

## Modelos de precios de Amazon EC2: casos de uso



Cargas de trabajo con picos de demanda



Cargas de trabajo independientes del tiempo



Cargas de trabajo de estado estable



Cargas de trabajo muy dependientes

Instancias bajo demanda	Instancias de spot	Instancias reservadas	Servidores dedicados
<ul style="list-style-type: none"> <li>Cargas de trabajo a corto plazo, con picos de demanda o impredecibles</li> <li>Desarrollo o prueba de aplicaciones</li> </ul>	<ul style="list-style-type: none"> <li>Aplicaciones con tiempos de inicio y finalización flexibles.</li> <li>Aplicaciones que solo son viables con precios de cómputo muy bajos</li> <li>Usuarios con necesidades de cómputo urgentes de grandes cantidades de capacidad adicional</li> </ul>	<ul style="list-style-type: none"> <li>Cargas de trabajo de uso predecible o estado estable</li> <li>Aplicaciones que requieren capacidad reservada, incluida la recuperación ante desastres (DR)</li> <li>Usuarios capaces de afrontar pagos iniciales para reducir aún más los costos de cómputo</li> </ul>	<ul style="list-style-type: none"> <li>Licencia "Bring your own license" (BYOL)</li> <li>Conformidad y restricciones normativas</li> <li>Seguimiento del uso y las licencias</li> <li>Control en una ubicación de instancia</li> </ul>



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

48

Esta es una revisión de algunos casos de uso de las distintas opciones de precios.

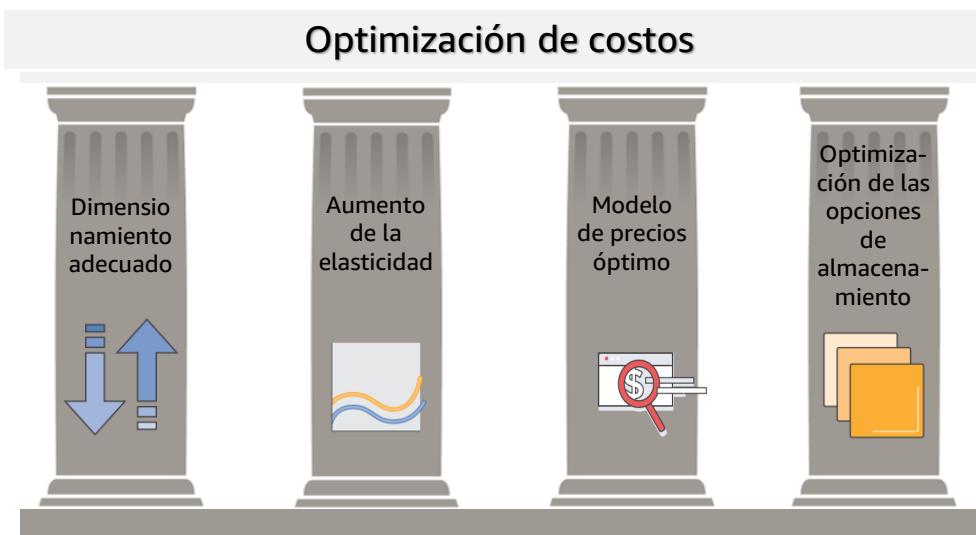
**Los precios de las** instancias bajo demanda funcionan bien para cargas de trabajo con picos de demanda o si solo necesita probar o ejecutar una aplicación durante un breve periodo (por ejemplo, durante el desarrollo o las pruebas de aplicaciones). A veces, las cargas de trabajo son impredecibles y las instancias bajo demanda son una buena opción para estos casos.

**Las** instancias de spot son una buena opción si sus aplicaciones pueden tolerar interrupciones con una notificación de advertencia de 2 minutos. De forma predeterminada, las instancias se terminan, pero puede configurarlas para que se detengan o hibernen en su lugar. Los casos prácticos habituales incluyen aplicaciones tolerantes a errores, como servidores web, backends de API y procesamiento de big data. Las cargas de trabajo que guardan datos de forma constante en almacenamiento persistente (como Amazon S3) también son buenas candidatas.

**Las** instancias reservadas son una buena opción cuando tiene cargas de trabajo a largo plazo con patrones de uso predecibles, como servidores que sabe que deseará ejecutar de manera uniforme durante muchos meses.

**Los** alojamientos dedicados son una buena opción cuando tiene licencias de software por socket, por núcleo o por máquina virtual, o cuando debe cumplir requisitos normativos y de conformidad corporativa específicos.

## Los cuatro pilares de la optimización de costos



aws

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

49

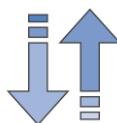
Para optimizar los costos, debe considerar cuatro factores potentes y consistentes:

- **Tamaño adecuado:** elija el equilibrio adecuado de tipos de instancias. Observe cuándo los servidores pueden reducirse o apagarse y aun así cumplir con los requisitos de rendimiento.
- **Aumento de la elasticidad:** diseña implementaciones para reducir la cantidad de capacidad del servidor que está inactiva mediante implementaciones que sean elásticas, tales como implementaciones que utilizan el escalado automático para manejar cargas máximas.
- **Modelo de precios óptimo:** reconoce las opciones de precios disponibles. Analice sus patrones de uso de manera que pueda ejecutar las instancias EC2 con la combinación adecuada de opciones de precios.
- **Optimización de las opciones de almacenamiento:** analiza los requisitos de almacenamiento de las implementaciones. Reduzca la sobrecarga de almacenamiento sin utilizar cuando sea posible y elegir opciones de almacenamiento más económicas si siguen cumpliendo con los requisitos de rendimiento de almacenamiento.

## Pilar 1: Dimensionamiento adecuado

### Pilares:

1. Dimensionamiento adecuado
2. Aumento de la elasticidad
3. Modelo de precios óptimo
4. Optimización de las opciones de almacenamiento



- ✓ Aprovisionar instancias para satisfacer la necesidad
  - CPU, memoria, almacenamiento y rendimiento de red
  - Seleccione los tipos de instancia apropiados para su uso
- ✓ Uso de métricas de Amazon CloudWatch
  - ¿Qué tan inactivas están las instancias? ¿Cuándo?
  - Reducir el tamaño de las instancias
- ✓ Práctica recomendada: tamaño correcto, luego reserve



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

50

Primero, considere el tamaño correcto. AWS ofrece aproximadamente 60 tipos y tamaños de instancias. La amplia variedad de opciones permite a los clientes seleccionar la instancia que mejor se adapte a su carga de trabajo. Puede resultar difícil saber por dónde empezar y qué instancia será la mejor, tanto desde el punto de vista técnico como de costos. El ajuste del tamaño es el proceso de revisar los recursos implementados y buscar oportunidades para reducir el tamaño de las implementaciones cuando sea posible.

#### Dimensión correcta:

- **Seleccione** la instancia más barata disponible que aún cumpla con sus requisitos de rendimiento.
- **Verifica** el uso de CPU, RAM, almacenamiento y red para identificar las instancias cuyo tamaño podría reducirse. Es posible que desee aprovisionar una variedad de tipos y tamaños de instancias en un entorno de prueba y luego probar su aplicación en esas diferentes implementaciones de prueba para identificar qué instancias ofrecen la mejor relación rendimiento-costo. Para lograr el tamaño correcto, utilice técnicas como las pruebas de carga a su favor.
- **Utilice** métricas de Amazon CloudWatch y configure métricas personalizadas. Una métrica representa un conjunto de valores ordenados en el tiempo que se publican en CloudWatch (por ejemplo, el uso de CPU de una instancia de EC2 particular). Los propios puntos de datos pueden proceder de cualquier aplicación o actividad empresarial desde la que se recopilen datos.

## Pilar 2: Aumento de la elasticidad

### Pilares:

1. Dimensión correcta
2. **Aumentar elasticidad**
3. Modelo de precios óptimo
4. Optimización de las opciones de almacenamiento



✓ **Detener o hibernar** instancias respaldadas por Amazon EBS que estén en uso inactivo

- Ejemplo: desarrollo no productivo o instancias de prueba

✓ Utilice el **escalado** automático para satisfacer las necesidades de acuerdo con el uso

- Elasticidad automatizada y basada en el tiempo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

51

Una forma de **elasticidad** es crear, iniciar o usar instancias de EC2 cuando sean necesarias, pero luego desactivarlas cuando no estén en uso. La elasticidad es uno de los principios centrales de la nube, pero los clientes a menudo pasan por un proceso de aprendizaje para poner en práctica la elasticidad y generar ahorros de costos.

La forma más fácil para que los grandes clientes adopten la elasticidad es buscar recursos que parezcan buenos candidatos para detenerse o hibernar, como entornos que no son de producción, cargas de trabajo de desarrollo o cargas de trabajo de prueba. Por ejemplo, si ejecuta cargas de trabajo de desarrollo o prueba en una única zona horaria, puede desactivar fácilmente esas instancias fuera del horario comercial y así reducir los costos de tiempo de ejecución quizás en un 65 por ciento. El concepto es similar a por qué hay un interruptor de luz al lado de la puerta y por qué la mayoría de las oficinas alientan a los empleados *a apagar las luces al salir de la oficina cada noche*.

Para las cargas de trabajo de producción, configurar políticas de escalado automático más precisas y granulares puede ayudarle a aprovechar el escalado horizontal para satisfacer las necesidades de capacidad máxima y no pagar por la capacidad máxima todo el tiempo.

Como regla general, debe apuntar a que entre el 20 y el 30 por ciento de sus instancias Amazon EC2 se ejecuten como instancias bajo demanda o instancias de spot, y también debe buscar activamente formas de maximizar la elasticidad.

## Pilar 3: Modelo de precios óptimo

### Pilares:

1. Dimensión correcta
2. Aumentar elasticidad
- 3. Modelo de precios óptimo**
4. Optimización de las opciones de almacenamiento



✓ Aproveche el modelo de precios adecuado para su caso práctico

- ¿Cuáles son los patrones de uso?

✓ Optimizar y *combinar tipos de compra*

✓ Ejemplos:

- Utilice **instancias bajo demanda e instancias de spot** para cargas de trabajo variables
- Utilice **instancias reservadas** para cargas de trabajo predecibles

✓ Considere soluciones sin servidor (AWS Lambda)



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

52

AWS ofrece varios modelos de precios para Amazon EC2 para ayudar a los clientes a ahorrar dinero. Los modelos disponibles se analizaron en detalle antes en este módulo. Los clientes pueden combinar múltiples tipos de compras para optimizar los precios en función de sus necesidades de capacidad actuales y previstas.

Además se anima a los clientes a considerar la arquitectura de su aplicación. Por ejemplo, ¿es necesario que la funcionalidad proporcionada por su aplicación se ejecute en una máquina virtual de Elastic Compute Cloud? Quizás al utilizar el servicio AWS Lambda, podría reducir significativamente sus costos.

Analizaremos AWS Lambda más adelante en este módulo.

## Pilar 4: Optimización de las opciones de almacenamiento

### Pilares:

1. Dimensión correcta
2. Aumentar elasticidad
3. Modelo de precios óptimo
4. Optimización de las opciones de almacenamiento



- ✓ Reduzca los costos manteniendo el rendimiento y la disponibilidad del almacenamiento
- ✓ Modificar el tamaño de los volúmenes de Amazon EBS
- ✓ Modificar los tipos de volúmenes de Amazon EBS
  - ✓ ¿Se pueden cumplir los requisitos de rendimiento con un almacenamiento menos costoso?
  - ✓ Ejemplo: el almacenamiento **HDD (st1) con rendimiento optimizado de Amazon EBS** por lo general cuesta la mitad que la opción de almacenamiento **SSD de uso general (gp2)** predeterminada.
- ✓ Elimine las instantáneas de EBS que ya no sean necesarias
- ✓ Identificar el destino más apropiado para tipos específicos de datos.
  - ✓ ¿La aplicación necesita que la instancia resida en Amazon EBS?
  - ✓ Las opciones de almacenamiento de Amazon S3 con políticas de ciclo de vida pueden reducir los costos



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

53

Los clientes también pueden reducir los costos de almacenamiento. Cuando inicia instancias de EC2, diferentes tipos de instancias ofrecen diferentes opciones de almacenamiento. Es una práctica recomendada el intentar reducir los costos y al mismo tiempo mantener el rendimiento y la disponibilidad del almacenamiento.

Una forma de lograrlo es mediante el redimensionamiento de los volúmenes de EBS. Por ejemplo, si en el inicio aprovisionó un volumen de 500 GB para una instancia de EC2 que solo necesitará un máximo de 20 GB de espacio de almacenamiento, puede reducir el tamaño del volumen y ahorrar costos.

También hay una variedad de **tipos de volúmenes de EBS**. Elija el tipo menos costoso que aún cumpla con sus requisitos de rendimiento. Por ejemplo, el almacenamiento HDD (st1) con rendimiento optimizado de Amazon EBS por lo general cuesta la mitad que la opción de almacenamiento SSD de uso general (gp2) predeterminada.

Si una unidad st1 satisface las necesidades de su carga de trabajo, aproveche el ahorro de costos.

Los clientes suelen utilizar **instantáneas de EBS** para crear respaldos de datos. Sin embargo, algunos clientes se olvidan de eliminar las instantáneas que son innecesarias. Elimine estas instantáneas innecesarias para ahorrar costos.

Finalmente, trate de identificar el **destino más apropiado para tipos específicos de datos**. ¿Su

aplicación necesita que los datos que utiliza residan en Amazon EBS? ¿La aplicación se ejecutaría igual de bien si utilizara Amazon S3 para el almacenamiento? La configuración de políticas de ciclo de vida de datos también puede reducir los costos. Por ejemplo, puede automatizar la migración de datos antiguos a los que se accede a ubicaciones de almacenamiento más económicas con poca frecuencia, como Amazon Simple Storage Service Glacier.

## Medición, supervisión y mejora

- La optimización de costos es un proceso continuo.



- Recomendaciones:

- Definir y aplicar el **etiquetado de asignación de costos**.
- Defina métricas, establezca objetivos y revíselos periódicamente.
- Aliente a los equipos a **diseñar por costos**.
- Asigne la responsabilidad de la optimización a un individuo o a un equipo.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

54

Si se hace correctamente, la optimización de costos no es un proceso único que completa un cliente. En cambio, al medir y analizar sus sistemas de forma rutinaria, puede mejorar y ajustar sus costos de manera continua.

El etiquetado *ayuda a proporcionar información sobre qué recursos utiliza* quién y con qué propósito. Puede activar etiquetas de asignación de costos en la consola de facturación y administración de costos, y AWS puede generar un informe de asignación de costos con el uso y los costos agrupados por sus etiquetas activas. Pueden aplicarse etiquetas que representen categorías empresariales (por ejemplo, centros de costos, nombres de aplicación o propietarios) para organizar los costos entre varios servicios.

**Aliente a los equipos a diseñar por costos.** Explorador de costos de AWS es una herramienta que puede utilizar para ver y analizar sus costos y uso. Puede utilizar el explorador de costos para identificar las áreas que deben estudiarse más a fondo y analizar tendencias que puede utilizar para comprender los costos.

Utilice AWS como **AWS Trusted Advisor**, que proporciona orientación en tiempo real para ayudarlo a aprovisionar recursos que siguen las prácticas recomendadas de AWS.

Los esfuerzos de optimización de costos suelen tener más éxito cuando la responsabilidad de la optimización de costos se asigna a un individuo o a un equipo.

## Conclusiones importantes de la Sección 3



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

55

- Los **modelos de precios de Amazon EC2** incluyen instancias bajo demanda, instancias reservadas, instancias de spot, instancias dedicadas y servidores dedicados.
- Las **instancias de spot** pueden ser interrumpidas con una notificación de 2 minutos. Sin embargo, pueden ofrecer importantes ahorros de costos en comparación con las instancias bajo demanda.
- Los **cuatro pilares de la optimización de costos** son:
  - Dimensionamiento adecuado
  - Aumento de la elasticidad
  - Modelo de precios óptimo
  - Optimización de las opciones de almacenamiento

Algunos puntos clave de esta sección del módulo son los siguientes:

- Los **modelos de precios de Amazon EC2** incluyen instancias bajo demanda, instancias reservadas, instancias de spot, instancias dedicadas y servidores dedicados. La facturación por segundo solo está disponible para las instancias bajo demanda, las instancias reservadas y las instancias de spot que ejecutan Amazon Linux y Ubuntu.
- Las **instancias de spot** pueden ser interrumpidas con una notificación de 2 minutos. Sin embargo, pueden ofrecer importantes ahorros de costos en comparación con las instancias bajo demanda.
- Los **cuatro pilares de la optimización de costos** son:
  - Dimensionamiento adecuado
  - Aumento de la elasticidad
  - Modelo de precios óptimo
  - Optimización de las opciones de almacenamiento

## Sección 4: Servicios de contenedores

Módulo 6: Cómputo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Sección introductoria 4: Servicios de contenedores.

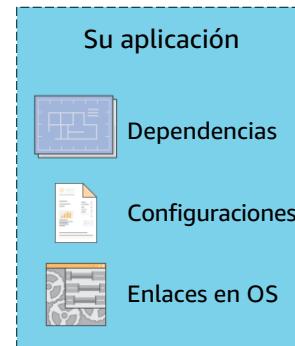
## Conceptos básicos de contenedores

- Los **contenedores** son un método de virtualización del sistema operativo.

Su contenedor

- **Beneficios:**

- Repetible.
- Entornos autónomos.
- El software se ejecuta igual en diferentes entornos.
  - Prueba, producción, portátil del desarrollador.
- Más rápido para iniciar, detener o terminar que las máquinas virtuales



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

57

Los contenedores **son un método de** virtualización del sistema operativo que le permite ejecutar una aplicación y sus dependencias en procesos aislados de recursos. Mediante el uso de contenedores, puede empaquetar fácilmente las dependencias, las configuraciones y el código de una aplicación en bloques de creación de uso fácil que ofrecen uniformidad del entorno, eficiencia operativa, productividad de los desarrolladores y control de versiones.

Los contenedores son más pequeños que las máquinas virtuales y no tienen un sistema operativo completo. En cambio, los contenedores *comparten un sistema operativo virtualizado* y se ejecutan como procesos aislados de los recursos, lo que garantiza implementaciones rápidas, confiables y uniformes. Los contenedores incluyen todo lo que el software necesita para ejecutarse, como bibliotecas, herramientas del sistema, código y tiempo de ejecución.

Los contenedores proporcionan **uniformidad de entorno** porque el código, las configuraciones y las dependencias de la aplicación se empaquetan en un único objeto.

En términos de espacio, las imágenes de contenedores suelen ser un orden de magnitud más pequeñas que las máquinas virtuales. Hacer girar un contenedor ocurre en cientos de milisegundos. Por lo tanto, al utilizar contenedores, puede utilizar entornos rápidos, portátiles y agnósticos de la infraestructura.

Los contenedores pueden ayudar a garantizar que las aplicaciones se implementen de forma

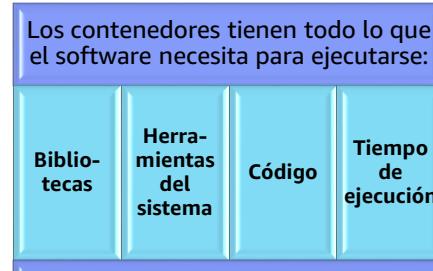
rápida, fiable y coherente, cualquiera que sea el entorno de implementación. Los contenedores también le ofrecen más control granular sobre los recursos, lo que le brinda una eficiencia mejorada de la infraestructura.

## ¿Qué es Docker?

- Docker es una plataforma de software que le permite crear, probar e implementar aplicaciones rápidamente.
- Puedes ejecutar contenedores en Docker.
  - Los contenedores se crean a partir de una plantilla llamada *imagen*.
  - Un **contenedor** tiene todo lo que una aplicación de software necesita para ejecutarse.



Contenedor



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

58

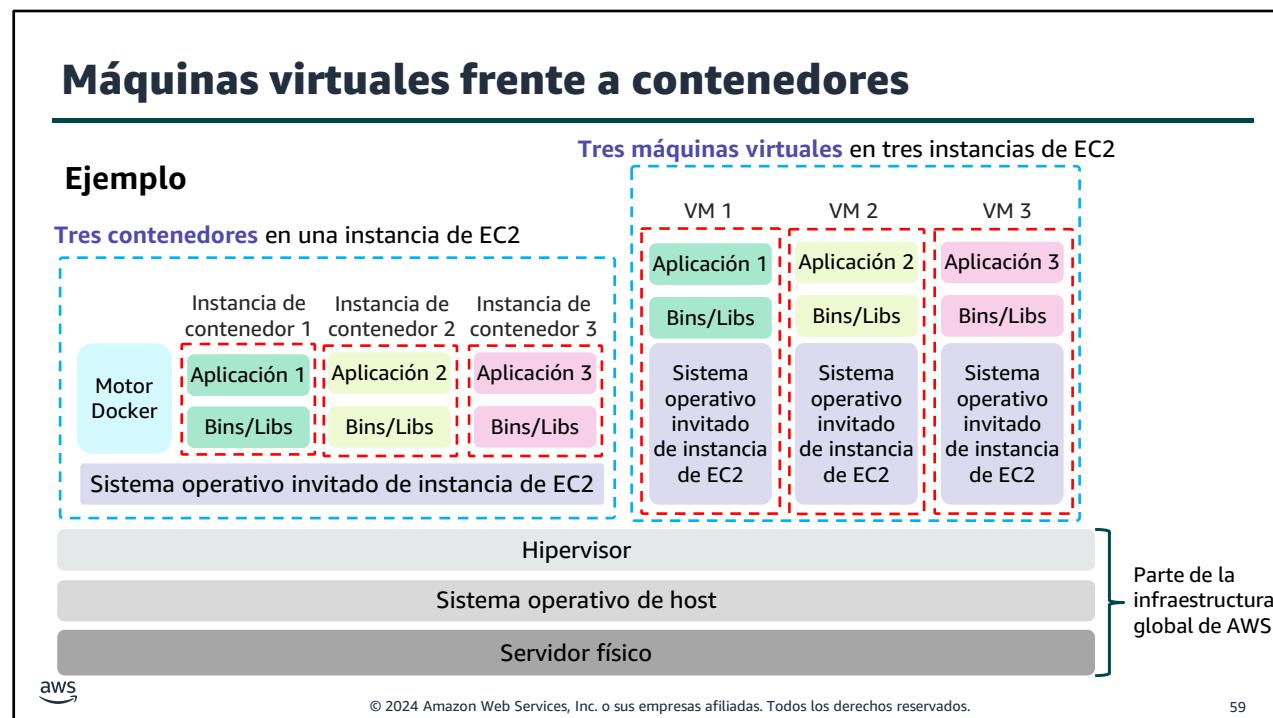
Docker es una plataforma de software que empaqueta software (tales como aplicaciones) en contenedores.

Docker se instala en cada servidor que alojará a los contenedores y proporciona comandos sencillos que puede utilizar para crear, comenzar o detener contenedores.

Con Docker, puede implementar y escalar las aplicaciones rápidamente en cualquier entorno.

Docker se utiliza mejor como una solución cuando desea:

- Estandarizar entornos
- Reducir los conflictos entre las pilas de idiomas y versiones
- Utilizar contenedores como un servicio
- Ejecutar microservicios mediante implementaciones de código estandarizadas
- Requerir portabilidad para el procesamiento de datos



Muchas personas que conocen por primera vez el concepto de contenedor piensan que los contenedores son exactamente como máquinas virtuales. Sin embargo, las diferencias están en los detalles. Una diferencia significativa es que las máquinas virtuales se ejecutan directamente en un hipervisor, pero los contenedores pueden ejecutarse en cualquier sistema operativo Linux si tienen el soporte de funciones del kernel adecuado y el daemon Docker está presente. Esto hace que los contenedores sean altamente portátiles. Su laptop, la máquina virtual, la instancia de EC2 y el servidor bare metal son posibles hosts donde puede ejecutar un contenedor.

**La derecha del diagrama tiene una implementación basada en máquina virtual (VM).** Cada una de las tres instancias de EC2 se ejecuta directamente en el hipervisor proporcionado por la infraestructura global de AWS. Cada instancia de EC2 ejecuta una máquina virtual. En esta implementación basada en VM, cada una de las tres aplicaciones se ejecuta en su propia VM, lo que proporciona aislamiento de procesos.

**La izquierda del diagrama tiene una implementación que se basa en contenedores.** Sólo hay una instancia de EC2 que ejecuta una máquina virtual. El motor Docker está instalado en el sistema operativo invitado Linux de la instancia de EC2 y hay tres contenedores. En esta implementación basada en contenedores, cada aplicación se ejecuta en su propio contenedor (lo que proporciona aislamiento de procesos), pero todos los contenedores se ejecutan en una única instancia de EC2. Los procesos que se ejecutan en los contenedores se comunican directamente con el kernel en el sistema operativo invitado Linux y en gran medida desconocen su silo de

contenedor. El motor Docker está presente para administrar cómo se ejecutan los contenedores en el sistema operativo huésped Linux y también proporciona funciones de administración esenciales durante todo el ciclo de vida del contenedor.

En una implementación real basada en contenedores, una instancia de EC2 grande podría ejecutar cientos de contenedores.

## Amazon Elastic Container Service (Amazon ECS)

- Amazon Elastic Container Service (**Amazon ECS**):

- Un servicio de administración de contenedores altamente escalable y rápido



Amazon Elastic  
Container Service

- Beneficios clave:

- Coordina la ejecución de contenedores de Docker
  - Mantiene y escala la flota de nodos que ejecutan sus contenedores
  - Elimina la complejidad de crear la infraestructura

- Integrado con características que son conocidas para los usuarios del servicio Amazon EC2:

- Balanceo de carga elástica
  - Grupos de seguridad de Amazon EC2
  - Volúmenes de Amazon EBS
  - Roles de IAM



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

60

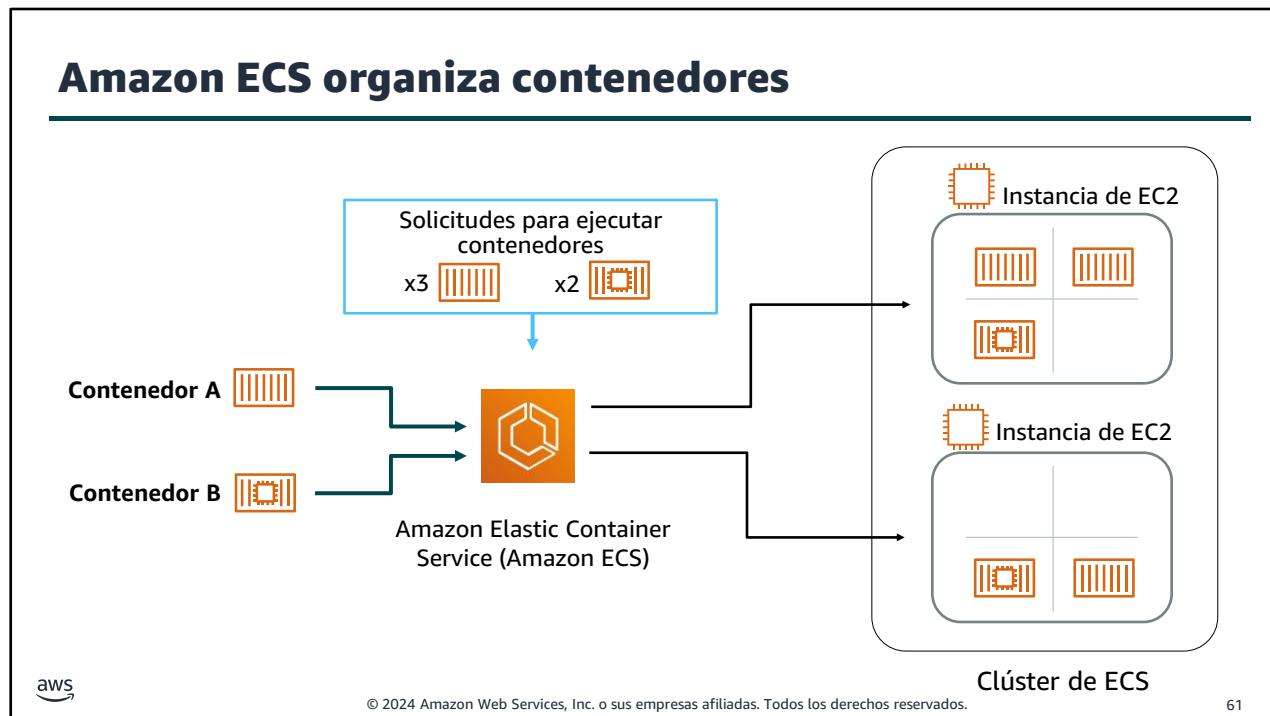
Dado lo que ahora sabe sobre los contenedores, podría pensar que podría iniciar una o más instancias de Amazon EC2, instalar Docker en cada instancia y administrar y ejecutar los contenedores de Docker en esas instancias de Amazon EC2 por su cuenta. Si bien esa es una opción, AWS proporciona un servicio llamado Amazon Elastic Container Service (Amazon ECS) que simplifica la administración de contenedores.

**Amazon Elastic Container Service (Amazon ECS)** es un servicio de administración de contenedores altamente escalables y de gran rendimiento que es compatible con los contenedores de Docker. Utilice Amazon ECS para ejecutar aplicaciones fácilmente en un clúster administrado de instancias de Amazon EC2.

Las funciones esenciales de Amazon ECS incluyen la capacidad de:

- **Iniciar** decenas de miles de contenedores de Docker en segundos
- **Supervisa** la implementación de contenedores
- **Administra** el estado del clúster que ejecuta los contenedores
- **Programar** contenedores con un programador integrado o de terceros (por ejemplo: Apache Mesos o Blox)

Los clústeres de Amazon ECS también pueden utilizar instancias de spot e instancias reservadas.



61

Para preparar su aplicación para que se ejecute en Amazon ECS, cree una **definición de tarea**, que es un archivo de texto que **describe uno o más contenedores**, hasta un máximo de diez, que forman su aplicación. Puede considerarlo como un esquema para su aplicación. Las definiciones de tareas especifican parámetros para su aplicación, por ejemplo, qué contenedores usar, qué puertos deben abrirse para su aplicación y qué volúmenes de datos se deben utilizar con los contenedores en la tarea.

Una **tarea** es la instancia de una definición de tareas dentro de un clúster. Puede especificar el número de tareas que se ejecutarán en el clúster. El **programador de tareas de Amazon ECS** es responsable de colocar las tareas dentro del clúster. Una tarea se ejecutará entre uno y diez contenedores, según la definición de tarea que haya definido.

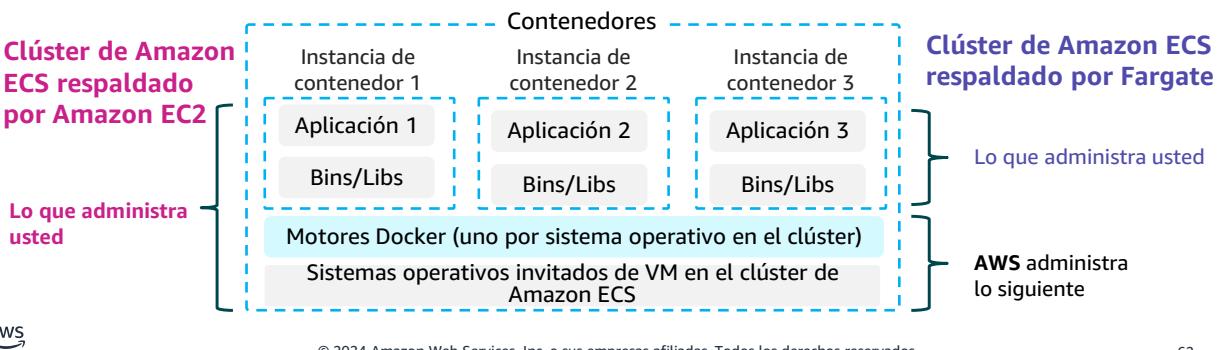
-

Cuando Amazon ECS ejecuta los contenedores que componen su tarea, los coloca en un **clúster** de ECS. El clúster (cuando elige el tipo de lanzamiento EC2) consta de un grupo de instancias de EC2, cada una de las cuales ejecuta un **agente de contenedor de Amazon ECS**.

Amazon ECS proporciona múltiples estrategias de programación que ubicarán contenedores en sus clústeres de acuerdo a sus necesidades de recursos (por ejemplo, CPU o RAM) y requisitos de disponibilidad.

## Opciones de clúster de Amazon ECS

- **Pregunta clave:** ¿Quiere administrar el clúster de Amazon ECS que ejecuta los contenedores?
  - En caso **afirmativo**, cree un **clúster de Amazon ECS respaldado por Amazon EC2** (proporciona un control más granular sobre la infraestructura)
  - En caso **negativo**, cree un clúster de **Amazon ECS respaldado por AWS Fargate** (más fácil de mantener, céntrese en sus aplicaciones)



Cuando se crea un clúster de Amazon ECS, existen tres opciones:

- Un **clúster** solo de red (con tecnología de AWS Fargate)
- Un **clúster** EC2 Linux + Networking
- Un **clúster** EC2 Windows + Networking

Si elige una de las dos opciones de **tipo de lanzamiento EC2**, se le pedirá que elija si las instancias de EC2 del clúster se ejecutarán como instancias bajo demanda o instancias de spot. Además, deberá especificar muchos detalles sobre las instancias de EC2 que conformarán su clúster, los mismos detalles que debe especificar cuando inicia una instancia de EC2 independiente. De esta manera, el tipo de inicio Elastic Compute Cloud proporciona un control más granular sobre la infraestructura que ejecuta sus aplicaciones de contenedor porque administra las instancias de EC2 que conforman el clúster.

Amazon ECS realiza un seguimiento de toda la CPU, la memoria y otros recursos del clúster. Amazon ECS también encuentra el mejor servidor para su contenedor en función de los requisitos de recursos especificados.

Si elige el **tipo de lanzamiento de Fargate** solo en red, AWS administrará el clúster que ejecutará sus contenedores. Con esta opción, solo necesita empaquetar la aplicación en contenedores, especificar los requisitos de CPU y memoria, definir las políticas de redes e IAM e iniciar la aplicación.

No es necesario aprovisionar, configurar ni escalar el clúster. Elimina la necesidad de elegir tipos

de servidores, decidir cuándo escalar los clústeres u optimizar conjuntos de clústeres. La opción Fargate le permite concentrarse en diseñar y crear sus aplicaciones.

## ¿Qué es Kubernetes?

- Kubernetes es un software de código abierto para la coordinación de contenedores.
  - Le permite desplegar y **administrar aplicaciones en contenedores** a escala.
  - El mismo conjunto de herramientas se puede utilizar en las instalaciones y en la nube.
- Complementa a Docker.
  - Docker le permite ejecutar varios contenedores en un único host de sistema operativo.
  - Kubernetes **coordina** múltiples hosts (nodos) de Docker.
- Automatiza:
  - Aprovisionamiento de contenedores.
  - Redes.
  - Distribución de carga.
  - Escalado.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

63

**Kubernetes** es un software de código abierto para la coordinación de contenedores. Kubernetes puede funcionar con muchas tecnologías de contenedorización, incluido Docker. Debido a que es un proyecto popular de código abierto, una gran comunidad de desarrolladores y empresas crean extensiones, integraciones y complementos que mantienen el software relevante, y con frecuencia se agregan funciones nuevas y demandadas.

Kubernetes le permite desplegar y administrar **aplicaciones en contenedores** a escala. Con Kubernetes, puede ejecutar cualquier tipo de aplicación en contenedores mediante el mismo conjunto de herramientas en las instalaciones y en la nube. Kubernetes funciona mediante la administración de un **clúster** de instancias de cómputo (llamados **nodos**). Ejecuta contenedores en el clúster, que se basan en dónde están disponibles los recursos de cómputo y los requisitos de recursos de cada contenedor. Los contenedores se ejecutan en agrupaciones lógicas llamadas **pods**. Puede ejecutar y escalar uno o varios contenedores juntos como un pod. A cada pod se le asigna una dirección IP y un único nombre de Sistema de nombres de dominio (DNS), que Kubernetes utiliza para conectar sus servicios entre sí y con el tráfico externo.

Una ventaja clave de Kubernetes es que puede usarlo para ejecutar aplicaciones en contenedores en cualquier entorno sin la necesidad de tener que cambiar sus herramientas operativas. Por ejemplo, las aplicaciones se pueden trasladar desde máquinas de desarrollo en las instalaciones a implementaciones de producción en la nube mediante las mismas herramientas operativas.

## Amazon Elastic Kubernetes Service (Amazon EKS)

- Amazon Elastic Kubernetes Service (**Amazon EKS**)
  - Habilita la ejecución de Kubernetes en AWS
  - Certificado conforme a Kubernetes (admite una fácil migración)
  - Admite contenedores de Linux y Windows
  - Compatible con las herramientas de la comunidad de Kubernetes y admite complementos populares de Kubernetes
- Utilice Amazon EKS para:
  - administrar clústeres de instancias de cómputo EC2
  - ejecute contenedores coordinados por Kubernetes en esas instancias



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

64

Podría pensar que podría iniciar una o más instancias de Amazon EC2, instalar Docker en cada instancia, instalar Kubernetes en el clúster y administrar y ejecutar Kubernetes por su cuenta. Si bien esa es una opción, AWS proporciona un servicio llamado Amazon Elastic Container Service (Amazon ECS) que simplifica la administración de los clústeres de Kubernetes.

**Amazon Elastic Kubernetes Service (Amazon EKS)** es un servicio administrado que le facilita la ejecución de Kubernetes en AWS sin necesidad de instalar, operar y mantener su propio plano de control de Kubernetes. Tiene certificación de conformidad con Kubernetes, por lo que las aplicaciones existentes que se ejecutan en Kubernetes ascendentes son compatibles con Amazon EKS.

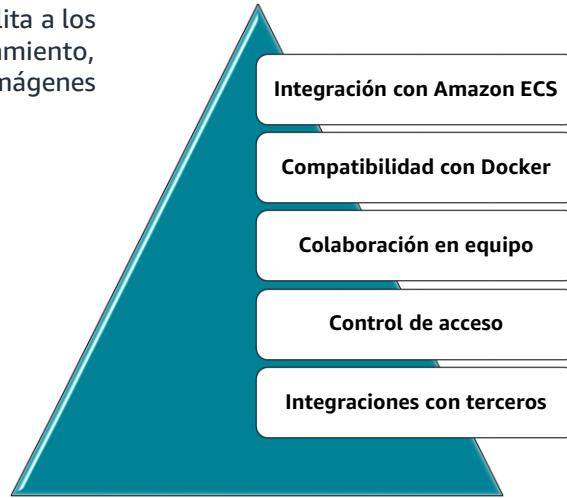
Amazon EKS administra automáticamente la disponibilidad y escalabilidad de los nodos del clúster que son responsables de iniciar y detener contenedores, programar contenedores en máquinas virtuales, almacenar datos del clúster y otras tareas. Detecta y reemplaza automáticamente los nodos del plano de control en mal estado para cada clúster. Puede aprovechar el rendimiento, la escala, la confiabilidad y la disponibilidad de la nube de AWS, que incluye redes de AWS y servicios de seguridad como equilibrador de carga de aplicación para distribución de carga, IAM para control de acceso basado en roles y VPC para redes de pods.

Quizás se pregunte por qué Amazon ofrece Amazon ECS y Amazon EKS, ya que ambos son capaces de coordinar contenedores de Docker. La razón por la que existen ambos servicios es

para brindar a los clientes opciones flexibles. Podrá decidir qué opción se adapta mejor a tus necesidades.

## Amazon Elastic Container Registry (Amazon ECR)

Amazon ECR es un **registro de contenedores Docker** completamente administrado que facilita a los desarrolladores las tareas de almacenamiento, administración e implementación de imágenes de contenedores de Docker.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

65

Amazon Elastic Container Registry (Amazon ECR) es un registro de contenedores de Docker completamente administrado que facilita a los desarrolladores las tareas de almacenamiento, administración e implementación de imágenes de contenedores de Docker. Está **integrado con Amazon ECS** para que pueda almacenar, ejecutar y administrar con facilidad imágenes de contenedores para aplicaciones que se ejecutan en Amazon ECS. Especifique el repositorio de Amazon ECR en la definición de tareas y Amazon ECS recuperará las imágenes correspondientes para sus aplicaciones.

Amazon ECR es compatible con API HTTP de Docker Registry, versión 2, que le permite interactuar con Amazon ECR mediante comandos de la CLI de Docker o las herramientas preferidas de Docker. Por lo tanto, puede mantener su flujo de trabajo de desarrollo existente y acceder a Amazon ECR desde cualquier entorno Docker, ya sea en la nube, en las instalaciones locales o en su equipo local.

Puede transferir las imágenes del contenedor a y desde Amazon ECS mediante HTTPS. Las imágenes también se cifran automáticamente en reposo mediante el cifrado del lado servidor de Amazon S3.

También es posible utilizar imágenes de Amazon ECR con **Amazon EKS**. Consulte la documentación Uso de imágenes de Amazon ECR con Amazon EKS en [https://docs.aws.amazon.com/AmazonECR/latest/userguide/ECR\\_on\\_EKS.html](https://docs.aws.amazon.com/AmazonECR/latest/userguide/ECR_on_EKS.html) for details.

## Conclusiones importantes de la Sección 4



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

66

- Los **contenedores** pueden contener todo lo que una aplicación necesita para ejecutarse.
- **Docker** es una plataforma de software que empaqueta software en contenedores.
  - Una sola aplicación puede abarcar varios contenedores.
- Amazon Elastic Container Service (**Amazon ECS**) coordina la ejecución de contenedores de Docker.
- **Kubernetes** es un software de código abierto para la coordinación de contenedores.
- Amazon Elastic Kubernetes Service (**Amazon EKS**) le permite ejecutar Kubernetes administrado en AWS
- Amazon Elastic Container Registry (**Amazon ECR**) permite almacenar, administrar e implementar los contenedores de Docker.

Algunos aprendizajes clave de esta sección incluyen:

- Los contenedores pueden contener todo lo que una aplicación necesita para ejecutarse.
- Docker es una plataforma de software que empaqueta software en contenedores.
- Una sola aplicación puede abarcar varios contenedores.
- Amazon Elastic Container Service (Amazon ECS) coordina la ejecución de contenedores de Docker.
- Kubernetes es un software de código abierto para la coordinación de contenedores.
- Amazon Elastic Kubernetes Service (Amazon EKS) le permite ejecutar Kubernetes en AWS
- Amazon Elastic Container Registry (Amazon ECR) le permite almacenar, administrar e implementar los contenedores de Docker.

## Sección 5: Introducción a AWS Lambda

Módulo 6: Cómputo

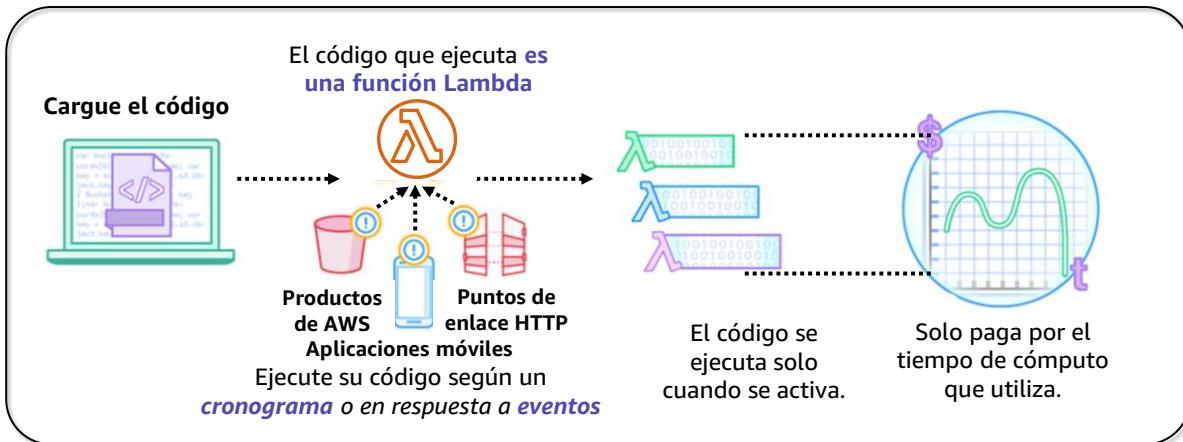


© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Sección introductoria 5: Introducción a AWS Lambda.

## AWS Lambda: ejecutar código sin servidores

AWS Lambda es una opción de cómputo **sin servidor**.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

68

Como vio en las secciones anteriores de este módulo, AWS ofrece muchas opciones de cómputo. Por ejemplo, **Amazon EC2** proporciona máquinas virtuales. Como otro ejemplo, **Amazon ECS** y **Amazon EKS** son servicios de cómputo basados en contenedores.

Sin embargo, existe otro enfoque del cómputo, que no requiere aprovisionar ni administrar servidores. Este tercer enfoque se conoce a menudo como cómputo **sin servidor**.

**AWS Lambda** es un servicio de cómputo sin servidor que se basa en eventos. Lambda le permite ejecutar código sin necesidad de aprovisionar ni administrar servidores.

Crea una **función Lambda**, que es el recurso de AWS que contiene el código que carga. Luego, configura la función Lambda para que se active, ya sea de forma programada o en respuesta a un evento. El código se ejecuta solo cuando se activa.

**Solo pagará por el tiempo de cómputo que consuma:** no se incurre en cargos cuando el código no se está en ejecución.

## Beneficios de Lambda



AWS  
Lambda



Es compatible con varios lenguajes de programación



Administración automatizada por completo



Tolerancia a fallas integrada



Admite la coordinación de múltiples funciones



Precios de pago por uso



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

69

Con AWS Lambda, no es necesario aprender idiomas, herramientas ni marcos de trabajo nuevos. Lambda **admite múltiples lenguajes de programación**, entre ellos Java, Go, PowerShell, Node.js, C#, Python y Ruby. Su código puede utilizar cualquier biblioteca, ya sea nativa o de terceros.

Lambda **automatiza completamente la administración**. Administra toda la infraestructura para ejecutar su código en una infraestructura altamente disponible y tolerante a fallas, lo que le permite concentrarse en crear servicios backend diferenciados. Lambda implementa su código sin interrupciones; realiza toda la administración, mantenimiento y parches de seguridad; y proporciona registro y supervisión integrados a través de Amazon CloudWatch.

Lambda proporciona **tolerancia a errores integrada**. Mantiene la capacidad de cómputo en múltiples zonas de disponibilidad en cada región para ayudar a proteger su código contra fallas de máquinas individuales o fallas del centro de datos. No hay ventanas de mantenimiento ni tiempos de inactividad programados.

Puede **coordinar varias funciones de Lambda** para tareas complejas o de larga duración mediante la creación de flujos de trabajo con AWS Step Functions. Utilice funciones de pasos para definir flujos de trabajo. Estos flujos de trabajo desencadenan una colección de funciones Lambda mediante el uso de pasos secuenciales, paralelos, de bifurcación y de manejo de errores. Con Step Functions y Lambda, puede crear procesos con estado y de larga duración para aplicaciones y backends.

Con Lambda, **se paga solo por las solicitudes atendidas y el tiempo de cómputo necesario para ejecutar su código**. La facturación se mide en incrementos de 100 milisegundos, lo que la hace rentable y fácil de escalar automáticamente desde unas pocas solicitudes diarias hasta miles de solicitudes por segundo.

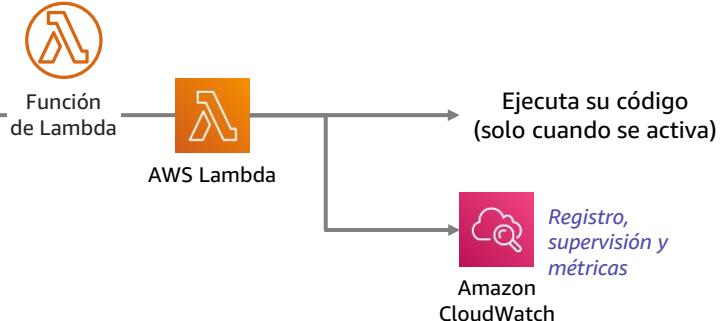
## Fuente de eventos de AWS Lambda

### Fuentes de eventos



Configure otros AWS como **orígenes de eventos** para invocar su función como se muestra aquí.

Como alternativa, invoque una función de Lambda desde la consola de Lambda, el SDK de AWS o la CLI de AWS.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

70

Una **fuente de eventos** es un servicio de AWS o una aplicación creada por el desarrollador que desencadena una función de AWS Lambda para que se ejecute.

Algunos servicios publican eventos en Lambda invocando la función Lambda directamente. Estos servicios que invocan funciones Lambda de forma **asincrónica** incluyen, entre otros, Amazon S3, Amazon Simple Notification Service (Amazon SNS) y Amazon CloudWatch Events.

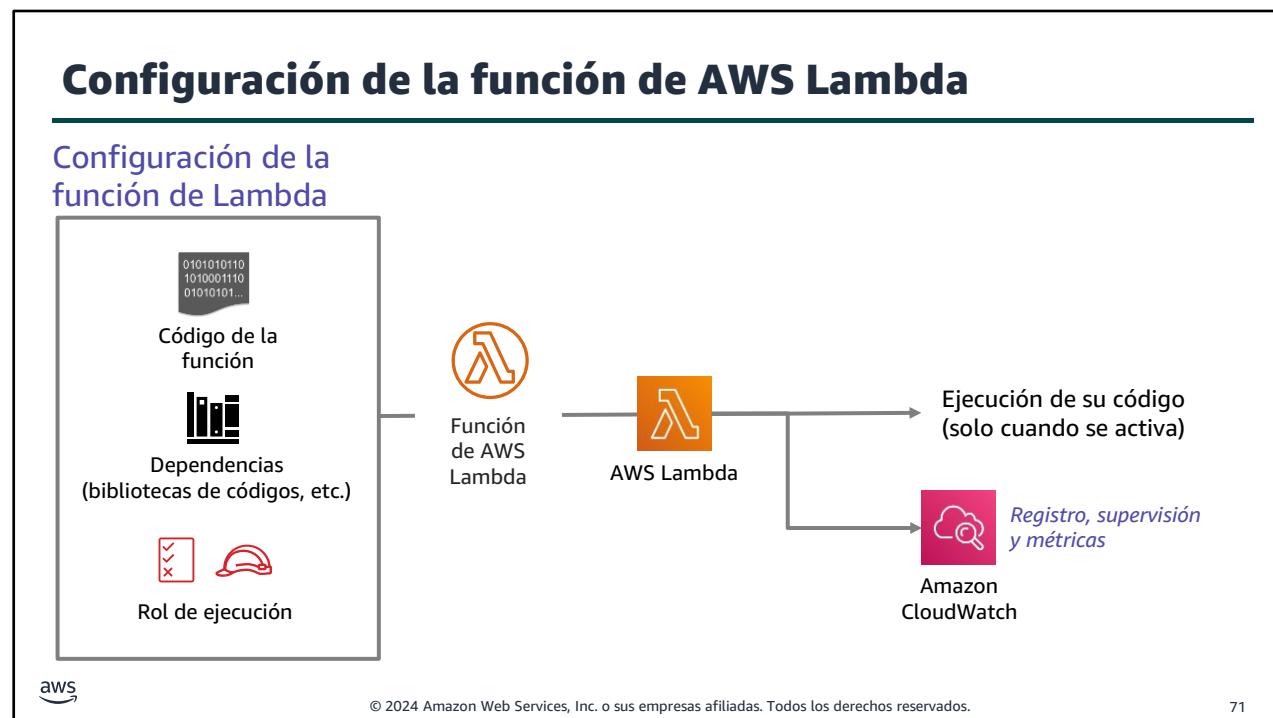
Lambda también puede sondear recursos en otros servicios que no publican eventos en Lambda. Por ejemplo, Lambda puede extraer registros de una cola de **Amazon Simple Queue Service (Amazon SQS)** y ejecutar una función Lambda para cada mensaje obtenido. Lambda puede leer eventos de **Amazon DynamoDB**.

Algunos servicios, como Elastic Load Balancing (Application Load Balancer) y Amazon API Gateway, pueden **invocar su función Lambda directamente**.

Puede invocar funciones de Lambda directamente con la consola de Lambda, la API de Lambda, el kit de desarrollo de software (SDK) de AWS, la CLI de AWS y los conjuntos de herramientas de AWS. El enfoque de invocación directa puede resultar útil, por ejemplo, cuando está desarrollando una aplicación móvil y desea que la aplicación llame a funciones Lambda. Para obtener más detalles sobre todos los servicios compatibles, consulte la documentación sobre el uso de Lambda con otros servicios en <https://docs.aws.amazon.com/lambda/latest/dg/lambda-integrations.html>

services.html.

**AWS Lambda supervisa automáticamente las funciones de Lambda a través de Amazon CloudWatch.** Para ayudarlo a solucionar los errores de una función, Lambda registra todas las solicitudes gestionadas por la función. Lambda también **almacena automáticamente los registros generados por el código** a través de Amazon CloudWatch Logs.



Recuerde que una función Lambda es el código personalizado que escribe para procesar eventos y que Lambda ejecuta la función Lambda en su nombre.

Al utilizar la Consola de administración de AWS para crear una **función Lambda**, primero otórguele un nombre a la función. Luego, lo que usted especifica:

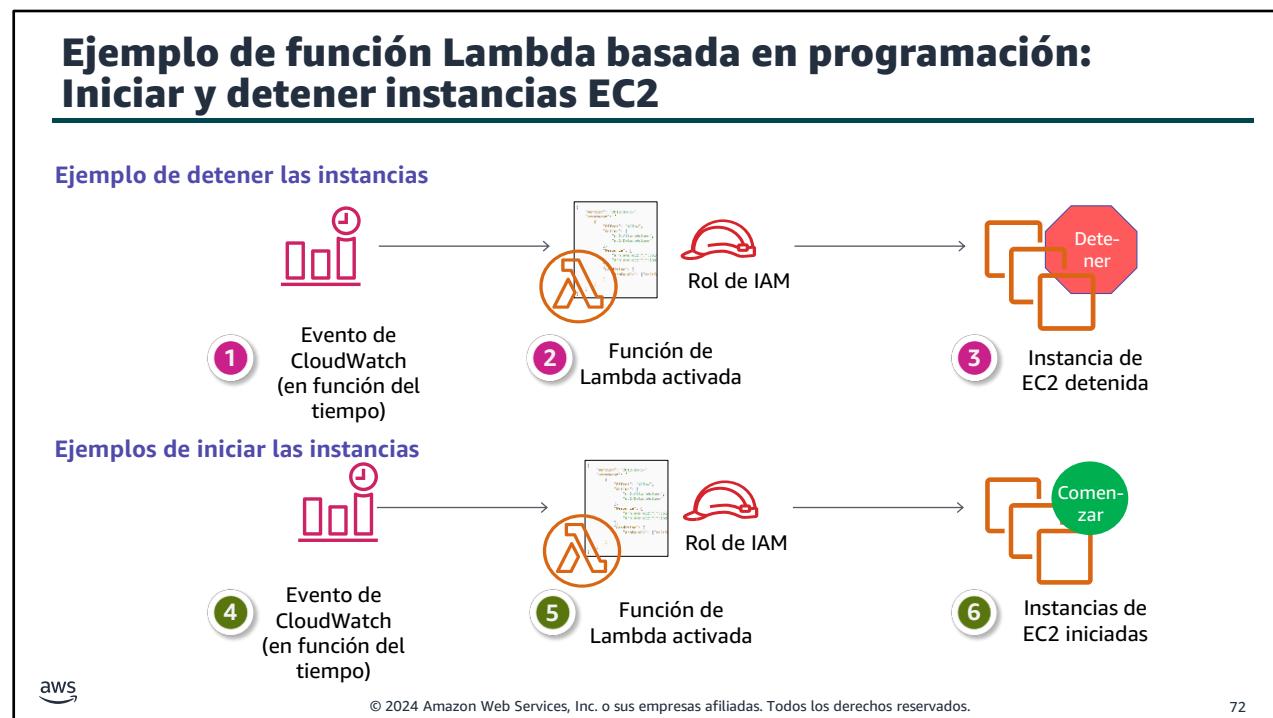
- El **entorno en tiempo de ejecución** que utilizará la función (por ejemplo, una versión de Python o Node.js)
- Un **rol de ejecución** (para otorgar permiso de IAM a la función para que pueda interactuar con otros AWS según sea necesario)

Lo siguiente, después de hacer clic en **Crear función**, configure la función. Las configuraciones incluyen:

- Agregue un **desencadenador** (**especifique uno de las fuentes de eventos disponibles en la diapositiva anterior**)
- Agregue su **código de función** (utilice el editor de código proporcionado o cargue un archivo que contenga su código)
- Especifique la **memoria** en MB para asignar a su función (128 MB a 10 240 MB)
- De manera opcional, especifique las variables de entorno, la descripción, el tiempo de espera, la nube privada virtual (VPC) específica para ejecutar la función, las etiquetas que le gustaría usar y otras configuraciones. Para obtener más información, consulte Configuración de funciones en la consola de AWS Lambda

<https://docs.aws.amazon.com/lambda/latest/dg/configuration-console.html> en la documentación de AWS.

Todas las configuraciones anteriores terminan en un **paquete de implementación de Lambda**, que es un archivo ZIP que contiene su código de función y sus dependencias. Cuando utiliza la consola Lambda para crear su función, la consola administra el paquete por usted. Sin embargo, debe crear un paquete de implementación si utiliza la API Lambda para administrar funciones.



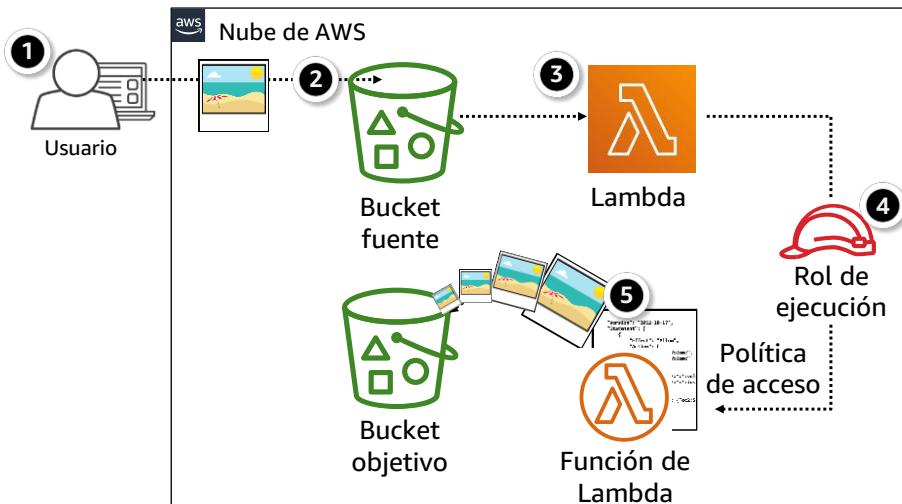
Considere un caso práctico de ejemplo para una función Lambda basada en programación. Supongamos que se encuentra en una situación en la que desea reducir el uso de Amazon EC2. Usted decide que desea detener las instancias a una hora predefinida (por ejemplo, por la noche, cuando nadie accede a ellas) y luego desea reiniciar las instancias por la mañana (antes de iniciar la jornada laboral).

En esta situación, puede configurar **AWS Lambda** y Amazon CloudWatch Events para realizar estas acciones automáticamente.

El ejemplo muestra lo que sucede en cada paso:

1. Está programado un evento de CloudWatch para ejecutar una función Lambda para detener sus instancias de EC2 (por ejemplo) a las 22:00 GMT.
2. La función de Lambda se activa y se ejecuta con el rol de IAM que le otorga permisos para detener las instancias de EC2.
3. Las instancias de Amazon EC2 entran en estado detenido.
4. Más tarde, a las (por ejemplo) 05:00 a. m. UTC, se programa un evento de CloudWatch Events para ejecutar una función de Lambda para iniciar las instancias de EC2.
5. La función de Lambda se activa y se ejecuta con el rol de IAM que le otorga permisos para iniciar las instancias de EC2.
6. Las instancias de Amazon EC2 entran en estado en ejecución.

## Ejemplo de función Lambda basada en eventos: Crear imágenes en miniatura



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

73

Ahora, considere un caso práctico de ejemplo para una función Lambda basada en eventos. Imagínese que desea crear una miniatura para cada imagen (objeto .jpg o .png) que se carga en un bucket de S3.

Para crear una solución, puede crear una función Lambda que Amazon S3 invoca cuando se cargan objetos. Luego, la función Lambda lee el objeto de imagen del bucket de origen y crea una imagen en miniatura en un bucket de destino. Funciona de esta manera:

1. El usuario carga un objeto en el bucket de origen de Amazon S3 (evento `object-created`).
2. Amazon S3 detecta el evento `object-created`.
3. Amazon S3 publica el evento `object-created` en Lambda mediante la invocación de la función de Lambda y el pase de los datos del evento.
4. Lambda ejecuta la función Lambda asumiendo el rol de ejecución que especificó cuando creó la función Lambda.
5. A partir de los datos del evento que recibe, la función de Lambda conoce el nombre del bucket de origen y el nombre de la clave del objeto. La función de Lambda lee el objeto y crea una miniatura con las bibliotecas de gráficos y la guarda en el bucket de destino.

## Cuotas de AWS Lambda

Límites suaves por región:

- Ejecuciones simultáneas = 1.000
- Almacenamiento de funciones y capas = 75 GB

Límites estrictos para funciones individuales:

- Asignación máxima de memoria de función = 10 240 MB
- Tiempo de espera de la función = 15 minutos
- Tamaño del paquete de implementación = 250 MB descomprimidos, incluidas las capas
- Tamaño del paquete del código de imagen del contenedor = 10 GB

También existen límites adicionales. Los detalles se encuentran en la documentación de cuotas de AWS Lambda en <https://docs.aws.amazon.com/lambda/latest/dg/gettingstarted-limits.html>.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

74

AWS Lambda tiene algunas cuotas que debe conocer al crear e implementar funciones Lambda.

AWS Lambda limita la cantidad de recursos de cómputo y almacenamiento que puede usar para ejecutar y almacenar funciones. Por ejemplo, en el momento de redactar este documento, el máximo de asignación de memoria para una sola función de Lambda es de 10,240 GB. También tiene límites de 1.000 ejecuciones simultáneas en una Región. Las funciones Lambda se pueden configurar para que se ejecuten hasta 15 minutos por ejecución. Puede establecer el tiempo de espera en cualquier valor entre 1 segundo y 15 minutos. Si solucionará problemas de una implementación de Lambda, tenga en cuenta estos límites.

Existen límites en el **tamaño del paquete de implementación** de una función (250 MB). Una **capa** es un archivo ZIP que contiene bibliotecas, un tiempo de ejecución personalizado u otras dependencias. Con las capas, puede utilizar las bibliotecas en la función sin necesidad de incluirlas en el **paquete de implementación**. El uso de capas puede ayudar a evitar alcanzar el límite de tamaño del paquete de implementación. Las capas también son una buena forma de compartir código y datos entre funciones Lambda.

Para cargas de trabajo más grandes que dependen de dependencias considerables, como machine learning o cargas de trabajo con uso intensivo de datos, puede implementar su función Lambda en una imagen de contenedor de hasta 10 GB de tamaño.

Los límites son suaves o duros. **Los límites suaves** de una cuenta se pueden suavizar potencialmente mediante el envío de un ticket de soporte y proporcionando una justificación de la solicitud. **Los límites duros** no se pueden aumentar.

Los detalles se encuentran en la documentación de cuotas de AWS Lambda en <https://docs.aws.amazon.com/lambda/latest/dg/gettingstarted-limits.html>.

## Conclusiones importantes de la Sección 5



- El **cómputo sin servidor** le permite crear y ejecutar aplicaciones y servicios sin aprovisionar ni administrar servidores.
- **Lambda es un servicio de cómputo sin servidor** que proporciona tolerancia a errores integrada y escalado automático.
- Un **origen de eventos** es un servicio de AWS o una aplicación creada por el desarrollador que desencadena una función de Lambda para que se ejecute.
- El máximo de asignación de memoria para una sola función de Lambda es de 10,240 GB.
- El máximo de tiempo de ejecución para una función de Lambda es de 15 minutos.

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

75

Entre los aprendizajes clave de esta sección del módulo, se incluyen los siguientes:

- El cómputo sin servidor le permite crear y ejecutar aplicaciones y servicios sin aprovisionar ni administrar servidores.
- AWS Lambda es un servicio de cómputo sin servidor que proporciona tolerancia a errores integrada y escalado automático.
- Una fuente de eventos es un servicio de AWS o una aplicación creada por el desarrollador que desencadena una función de Lambda para que se ejecute.
- El máximo de asignación de memoria para una sola función de Lambda es de 10,240 GB.
- El máximo de tiempo de ejecución para una función de Lambda es de 15 minutos.

## Actividad: Crear una función Stopinator de AWS Lambda



### Para completar esta actividad:

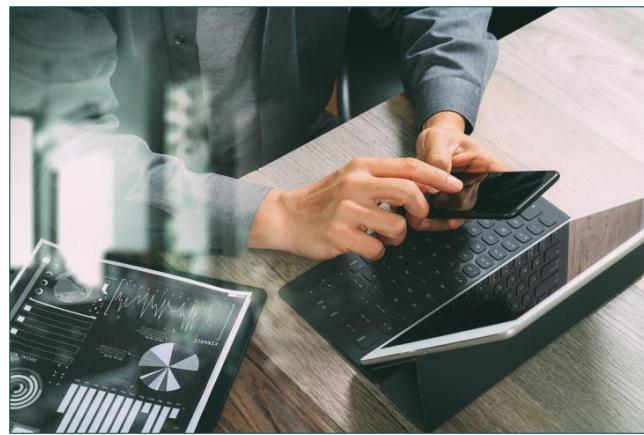
- vaya al entorno de laboratorio práctico e inicie la actividad AWS Lambda.
- siga las instrucciones que se proporcionan en el entorno de laboratorio práctico.

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

76

En esta actividad práctica, creará una función Lambda básica que detiene una instancia de EC2.

## Informe de actividad: Aprendizajes clave



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

77

El instructor dirigirá una conversación sobre los aprendizajes clave de la actividad después de que los estudiantes lo hayan completado.

## Sección 6: Introducción a AWS Elastic Beanstalk

Módulo 6: Cómputo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Sección introductoria 6: Introducción a AWS Elastic Beanstalk.

## AWS Elastic Beanstalk

- Una forma sencilla de poner **aplicaciones web** en funcionamiento
- Un **servicio administrado** que maneja automáticamente:
  - Administración de aprovisionamiento y configuración
  - Implementación
  - Balanceo de carga
  - Escalado automático
  - Monitoreo de la salud
  - Análisis y depuración
  - Registros
- No se aplican cargos adicionales por utilizar Elastic Beanstalk
  - Solo paga por los servicios subyacentes que se utilizan



**AWS Elastic  
Beanstalk**



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

79

AWS Elastic Beanstalk es otra opción de servicio de cómputo. Elastic Beanstalk es una plataforma como servicio (PaaS) que facilita la rápida implementación, el escalado y la administración de las aplicaciones web y servicios.

Mantenga el control. Una vez desarrollada toda la plataforma, lo único que tiene que hacer es cargar el código. Elija su tipo de instancia, su base de datos, configure y ajuste el escalado automático, actualice la aplicación, acceda a los archivos de registro del servidor y habilite HTTP en el equilibrador de carga.

Tiene que cargar el código y Elastic Beanstalk administrará de manera automática la implementación, desde el aprovisionamiento de la capacidad, el equilibrio de carga y el escalado automático hasta la monitorización del estado de la aplicación. Además, tiene el control absoluto de los recursos de AWS que alimentan la aplicación y puede acceder a los recursos subyacentes en cualquier momento.

→

No se aplican cargos adicionales por utilizar AWS Elastic Beanstalk. Paga por los recursos de AWS (por ejemplo, instancias de EC2 o buckets de S3) que crea para almacenar y ejecutar su aplicación. Solo paga por lo que usa, cuando lo usa. No hay tarifas mínimas ni compromisos iniciales.

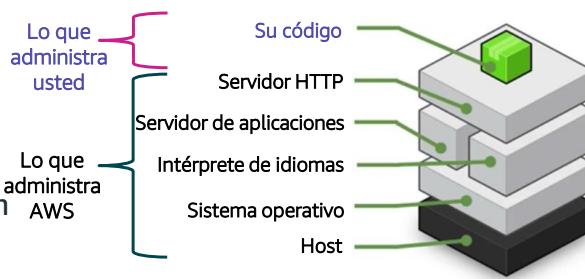
## Implementación de AWS Elastic Beanstalk

- Admite aplicaciones web escritas para plataformas comunes.

- **Java, .NET, PHP, Node.js, Python, Ruby, Go y Docker**

- Cargue su código

- Elastic Beanstalk maneja automáticamente la implementación
- Se desarrollan en servidores tales como Apache, NGINX, Passenger, Puma y Microsoft Internet Information Services (IIS).



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

80

AWS Elastic Beanstalk le permite implementar su código a través de la Consola de administración de AWS, la interfaz de línea de comandos de AWS (AWS CLI), Visual Studio y Eclipse. Elastic Beanstalk ofrece todos los servicios de aplicaciones que necesita para la aplicación. Solo tiene que crear su código. Elastic Beanstalk está diseñado para hacer que la implementación de su aplicación sea un proceso rápido y sencillo.

Elastic Beanstalk admite una gran variedad de plataformas. Las plataformas compatibles incluyen Docker, Go, Java, .NET, Node.js, PHP, Python y Ruby.

AWS Elastic Beanstalk implementa su código en **Apache Tomcat** para aplicaciones Java; **Servidor HTTP Apache** para aplicaciones PHP y Python; NGINX o Apache HTTP Server para aplicaciones Node.js; **Passenger** o **Puma** para aplicaciones Ruby; y **Microsoft Internet Information Services (IIS)** para aplicaciones .NET, Java SE, Docker y Go.

## Beneficios de Elastic Beanstalk



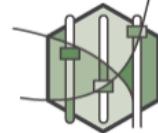
Rápido y sencillo  
de empezar a usar



Productividad  
para los  
desarrolladores



Difícil de  
optimizar



Control completo  
de recursos



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

81

Elastic Beanstalk es **rápido y sencillo de empezar a utilizar**. Puede utilizar la consola de administración de AWS, un repositorio Git o un entorno de desarrollo integrado (IDE) tales como Apache Eclipse o Microsoft Visual Studio para cargar la aplicación. Elastic Beanstalk gestiona de forma automática los detalles de implementación de aprovisionamiento de capacidad, balanceo de carga, escalado automático y monitoreo del estado de las aplicaciones.

Puede mejorar su **productividad del desarrollador** al enfocarse en escribir código en lugar de administrar y configurar servidores, bases de datos,平衡adores de carga, firewalls y redes. AWS actualiza la plataforma subyacente que ejecuta su aplicación con parches y actualizaciones.

Elastic Beanstalk es **difícil de superar**. Con Elastic Beanstalk, la aplicación puede manejar las cargas máximas de trabajo o tráfico y usted puede minimizar los costos al mismo tiempo. Escala de forma automática la aplicación de forma ascendente o descendente, en función de las necesidades de la aplicación, mediante el uso de configuraciones de escalado automático fácilmente ajustables. Puede usar las métricas de utilización de la CPU para activar las acciones de escalado automático.

–Tiene la **libertad de seleccionar los recursos de AWS** (tales como el tipo de instancia de Amazon EC2) que sean óptimos para su aplicación.

Elastic Beanstalk le permite mantener el control total de los recursos de AWS que alimentan a su aplicación. Si decide que desea hacerse cargo de algunos (o todos) los elementos de su

infraestructura, puede hacerlo sin problemas mediante las capacidades de administración que proporciona Elastic Beanstalk.

## Actividad: AWS Elastic Beanstalk



### Para completar esta actividad:

- vaya al entorno de laboratorio práctico e inicie la actividad AWS Elastic Beanstalk.
- siga las instrucciones que se proporcionan en el entorno de laboratorio práctico.

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

82

En esta actividad práctica, comprenderá por qué es posible que desee utilizar Elastic Beanstalk para implementar una aplicación web en AWS.

## Informe de actividad: Aprendizajes clave



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

83

Después de completar el laboratorio, su instructor puede optar por dirigir una conversación sobre los aprendizajes clave de la actividad.

## Conclusiones importantes de la Sección 6



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

84

- AWS Elastic Beanstalk mejora la productividad de los desarrolladores.
  - Simplifica el proceso de implementación de su aplicación.
  - Reduce la complejidad de administración.
- Elastic Beanstalk es compatible con **Java, .NET, PHP, Node.js, Python, Ruby, Go y Docker**
- No se aplican cargos adicionales por utilizar Elastic Beanstalk. Pague solo por los recursos de AWS que utilice.

Entre los aprendizajes clave de esta sección del módulo, se incluyen los siguientes:

- AWS Elastic Beanstalk mejora la productividad de los desarrolladores.
  - Simplifica el proceso de implementación de su aplicación.
  - Reduce la complejidad de administración.
- Elastic Beanstalk es compatible con Java, .NET, PHP, Node.js, Python, Ruby, Go y Docker.
- No se aplican cargos adicionales por utilizar Elastic Beanstalk. Pague solo por los recursos AWS que utilice.

# Conclusión del módulo

Módulo 6: Cómputo



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

Ahora es el momento de revisar el módulo y concluir con una evaluación de conocimientos y una discusión sobre una pregunta del examen de certificación de práctica.

## Resumen del módulo

En resumen, en este módulo aprendieron a hacer lo siguiente:

- Proporcionar información general de los diferentes servicios de cómputo de AWS en la nube.
- Mostrar cómo se usa Amazon Elastic Compute Cloud (Amazon EC2)
- Identificar la funcionalidad en la consola de Amazon EC2
- Realizar funciones básicas en Amazon EC2 para construir un entorno informático virtual
- Identificar los elementos de optimización de costos de Amazon EC2
- Mostrar cuándo usar AWS Elastic Beanstalk
- Mostrar cuándo usar AWS Lambda
- Identificar cómo ejecutar aplicaciones en contenedores en un clúster de servidores administrados



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

86

En resumen, en este módulo aprendieron a hacer lo siguiente:

- Proporcionar información general de los diferentes servicios de cómputo de AWS en la nube.
- Mostrar cómo se usa Amazon Elastic Compute Cloud (Amazon EC2)
- Identificar la funcionalidad en la consola de Amazon EC2
- Realizar funciones básicas en Amazon EC2 para construir un entorno informático virtual
- Identificar los elementos de optimización de costos de Amazon EC2
- Mostrar cuándo usar AWS Elastic Beanstalk
- Mostrar cuándo usar AWS Lambda
- Identificar cómo ejecutar aplicaciones en contenedores en un clúster de servidores administrados

# Completar la evaluación de conocimientos



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

87

Ahora es el momento de completar la evaluación de conocimientos para este módulo.

## Pregunta de examen de ejemplo



¿Qué servicio de AWS ayuda a los desarrolladores a implementar rápidamente recursos que pueden utilizar diferentes lenguajes de programación, como .NET y Java?

Opción Respuesta

- | Opción | Respuesta                                 |
|--------|---|
| A      | AWS CloudFormation                        |
| B      | AWS SQS                                   |
| C      | AWS Elastic Beanstalk                     |
| D      | Amazon Elastic Compute Cloud (Amazon EC2) |

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

88

Mire las opciones de respuesta y descártelas según las palabras clave.

## Respuesta a la pregunta de examen de ejemplo



¿Qué servicio de AWS ayuda a los desarrolladores a implementar rápidamente recursos que pueden utilizar diferentes lenguajes de programación, como .NET y Java?

La respuesta correcta es la opción C.

Las palabras clave en la pregunta son que los desarrolladores implementan rápidamente recursos y diferentes lenguajes de programación.

© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

89

Las siguientes son las palabras clave a reconocer: los desarrolladores implementan recursos y diferentes lenguajes de programación de manera rápida.

La respuesta correcta es **C. AWS Elastic Beanstalk**

Respuestas incorrectas:

Respuesta A: AWS CloudFormation

Respuesta B: AWS SQS

Respuesta D: Amazon Elastic Compute Cloud (Amazon EC2)

## Recursos adicionales

- Documentación de Amazon EC2: <https://docs.aws.amazon.com/ec2/>
- Precios de Amazon EC2: <https://aws.amazon.com/ec2/pricing>
- Taller de Amazon ECS: <https://ecsworkshop.com/>
- Ejecución de contenedores en AWS: <https://containersonaws.com/>
- Taller de Amazon EKS: <https://www.eksworkshop.com/>
- Documentación de AWS Lambda: <https://docs.aws.amazon.com/lambda/>
- Documentación de AWS Elastic Beanstalk:  
<https://docs.aws.amazon.com/elastic-beanstalk/>
- Optimización de costos de Playbook:  
[https://d1.awsstatic.com/pricing/AWS\\_CO\\_Playbook\\_Final.pdf](https://d1.awsstatic.com/pricing/AWS_CO_Playbook_Final.pdf)



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

90

Los servicios de cómputo en AWS son un tema amplio y este módulo solo proporcionó una introducción al tema. Los siguientes recursos proporcionan más detalles:

- Documentación de Amazon EC2: <https://docs.aws.amazon.com/ec2/>
- Precios de Amazon EC2: [https://aws.amazon.com/ec2/pricing/](https://aws.amazon.com/ec2/pricing)
- Taller de Amazon ECS: <https://ecsworkshop.com/>
- Ejecución de contenedores en AWS: <https://containersonaws.com/>
- Taller de Amazon EKS: <https://www.eksworkshop.com/>
- Documentación de AWS Lambda: <https://docs.aws.amazon.com/lambda/>
- Documentación de AWS Elastic Beanstalk: <https://docs.aws.amazon.com/elastic-beanstalk/>
- Optimización de costos de Playbook:  
[https://d1.awsstatic.com/pricing/AWS\\_CO\\_Playbook\\_Final.pdf](https://d1.awsstatic.com/pricing/AWS_CO_Playbook_Final.pdf)

# Gracias

Todas las marcas comerciales pertenecen a sus propietarios.



© 2024 Amazon Web Services, Inc. o sus empresas afiliadas. Todos los derechos reservados.

91

Gracias por completar este módulo.