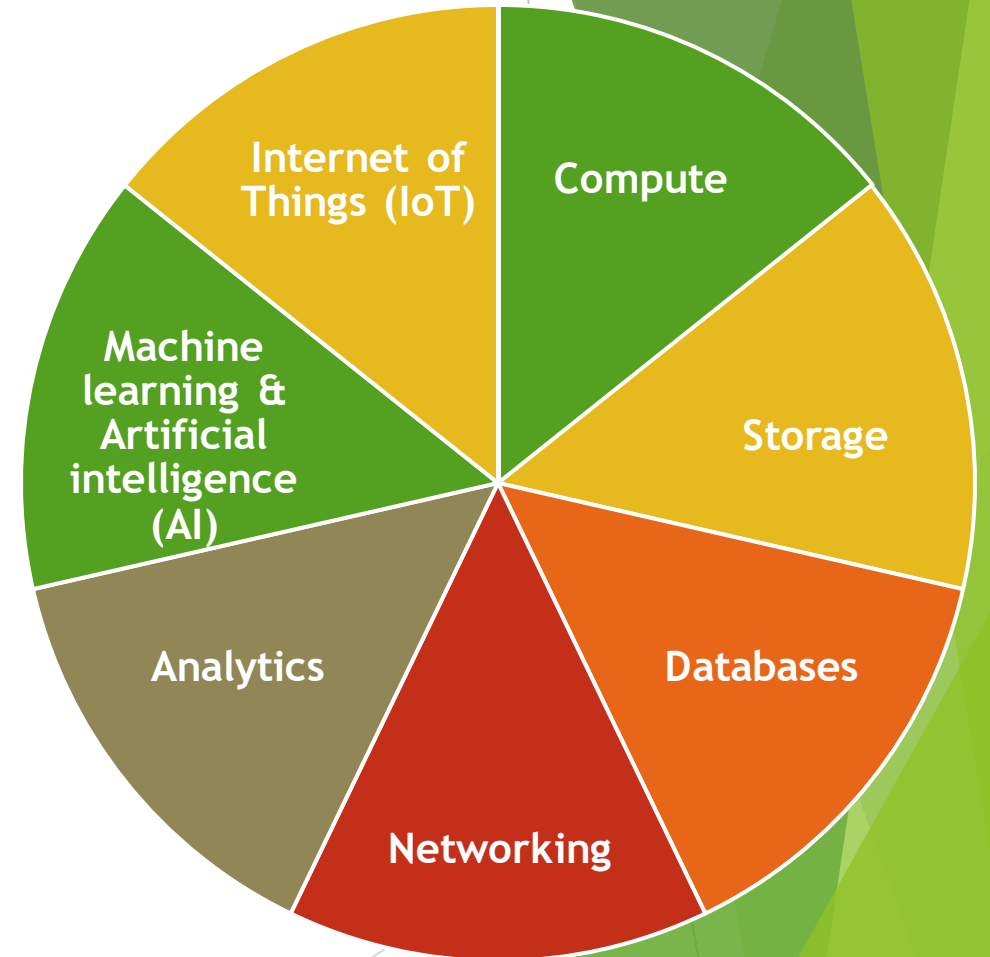


# AWS CONCEPTOS

# QUE ES AWS?

- ▶ Amazon Web Services (AWS) es la principal plataforma en la nube del mundo.
- ▶ AWS ofrece más de 165 servicios completos (¡40 de ellos no se ofrecen en ningún otro lugar!).
- ▶ AWS es adoptado por millones de clientes en todo el mundo, incluidas empresas de pequeña y gran escala.
- ▶ AWS permite que las empresas sean más ágiles, flexibles y seguras a una fracción del costo.
- ▶ AWS proporciona servicios para una amplia gama de aplicaciones, como:





# QUE ES CLOUD COMPUTING?

- ▶ La computación en la nube es la entrega bajo demanda de servicios como cómputo y almacenamiento a través de Internet con precios de pago por uso.
- ▶ En pocas palabras, en lugar de comprar un servidor físico o una computadora, ¡puede alquilarlo!
- ▶ Puede aprovechar el poder de cómputo, el almacenamiento y las bases de datos, según sea necesario, de un proveedor de nube como Amazon Web Services (AWS).

# QUE ES CLOUD COMPUTING?

- ▶ Muchas empresas de varios tamaños confían en la computación en la nube para muchas aplicaciones, tales como:
  - ▶ Escritorios virtuales
  - ▶ Almacenamiento y respaldo de datos
  - ▶ Análisis de grandes datos
  - ▶ Recuperación de desastres
  - ▶ Correo electrónico
  - ▶ Machine Learning para pronósticos y predicciones

Ejemplo: Los bancos y las instituciones financieras confían en la computación en la nube para evitar transacciones fraudulentas en tiempo real.

# ¿CUÁLES SON LOS BENEFICIOS DE LA CLOUD COMPUTING?

## AGILIDAD

- La computación en la nube permite a las empresas ser ágiles.
- En lugar de comprar hardware y software físicos y configurarlos, lo que puede llevar meses, la computación en la nube brinda a las empresas acceso a numerosas tecnologías que permiten una implementación e innovación más rápidas.
- En minutos, las empresas pueden aprovisionar e implementar recursos como cómputo, almacenamiento y bases de datos.
- Esto permite a las empresas ser más ágiles y les da libertad para experimentar con nuevas ideas más rápido.

## ELASTICIDAD

- La computación en la nube permite escalar y reducir los recursos en función de la demanda.
- Por lo tanto, no hay necesidad de sobreaprovisionar recursos por adelantado.

## AHORRO EN COSTO

- La computación en la nube intercambia gastos de capital, como la compra de servidores físicos, por gastos variables.
- Grandes ahorros con economías de escala.

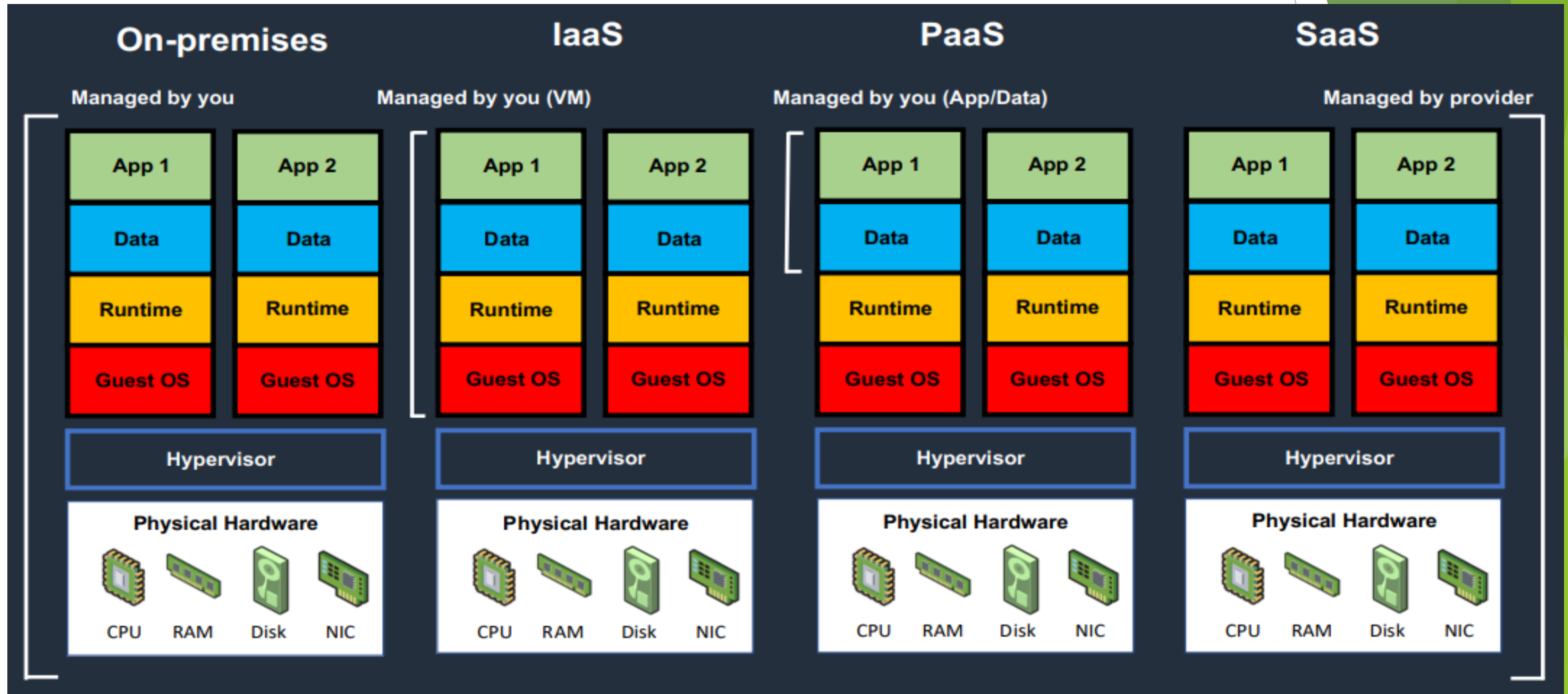
## IMPLEMENTACION GLOBAL EN MINUTOS

- La infraestructura de AWS está disponible en todo el mundo.
- Las empresas pueden implementar servicios en cualquier rincón del planeta en minutos aprovechando la nube.
- Colocar las aplicaciones cerca del cliente es fundamental para mejorar la latencia y el rendimiento.

# ¿CUÁLES SON LOS MODELOS DE SERVICIOS EN LA NUBE ?

- ▶ Los principales términos que encontrará con respecto a la computación en la nube son:
  - IaaS, Infrastructure-as-a-Service (Infraestructura como servicio)
  - PaaS, Platform-as-a-Service (Plataforma como servicio)
  - SaaS, Software-as-a-Service (Software como servicio)

# ¿CUÁLES SON LOS MODELOS DE SERVICIOS EN LA NUBE ?







## SERVICIOS BASE



### ALMACENAMIENTO Y ENTREGA DE CONTENIDO

- Storage Gateway
- Import/Export
- Glacier
- Elastic File System
- EBS
- CloudFront
- S3



### INFORMÁTICA

- Elastic Load Balancing
- Auto Scaling
- Lambda
- Elastic Beanstalk
- EC2



### BASES DE DATOS

- Redshift
- ElasticCache
- DynamoDB
- Migration Service
- RDS



## DESPLIEGUE Y ADMINISTRACIÓN



### SERVICIOS DE APLICACIONES

- API Gateway
- App Stream
- Cloud Search
- Elastic Transcoder
- SES
- SNS
- SQS
- SWF



### HERRAMIENTAS DE GESTIÓN

- Trusted Advisor
- Service Catalog
- OpsWorks
- Config
- CloudTrail
- CloudFormation
- CloudWatch



### SEGURIDAD E IDENTIDAD

- WAF
- Key Management Service
- CloudHSM
- Inspector
- Directory Service
- IAM
- Certificate Manager



### HERRAMIENTAS PARA DESARROLLADORES

- CodePipeline
- CodeDeploy
- CodeCommit



## PLATAFORMAS ENRIQUECIDAS



### ANÁLISIS

- Redshift
- Quicksight
- Machine Learning
- Elasticsearch Services
- Data Pipeline
- EMR



### SERVICIOS MÓVILES

- SNS
- Mobile Analytics
- Device Farm
- Cognito
- API Gateway
- Mobile Hub



### REDES

- Route 53
- Elastic Load Balancing
- Direct Connect
- VPC



### APLICACIONES EMPRESARIALES

- WorkDocs
- WorkMail
- WorkSpaces



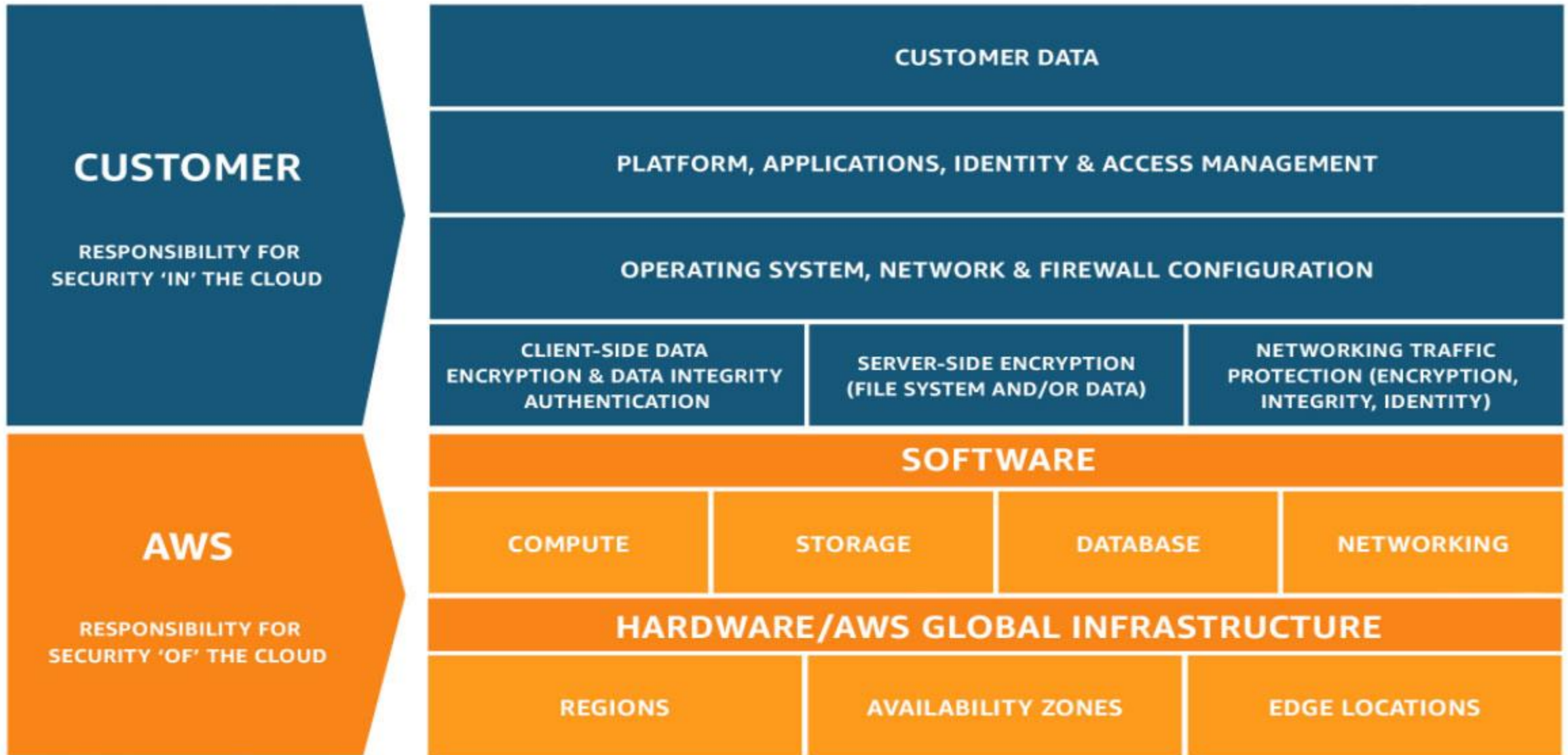
### INTERNET DE LAS COSAS

- IoT

# ¿CUÁLES SON LOS MODELOS DE SERVICIOS EN LA NUBE ?



# ¿Modelo de responsabilidad compartida?



# AWS Well-Architected Framework

<https://wa.aws.amazon.com/wat.pillars.wa-pillars.es.html>

Nombre completo	Description (Descripción)
<b>Excelencia operativa</b>	Capacidad de apoyar el desarrollo y ejecutar cargas de trabajo eficazmente, conocer sus operaciones y mejorar continuamente los procesos y procedimientos de soporte para ofrecer valor empresarial.
<b>Seguridad</b>	Capacidad de proteger datos, sistemas y activos para sacar partido de las tecnologías de la nube con el fin de mejorar el nivel de seguridad.
<b>Fiabilidad</b>	La capacidad de una carga de trabajo de llevar a cabo la función prevista de forma correcta y coherente cuando se espera que lo haga. Esto incluye la capacidad de utilizar y probar la carga de trabajo a lo largo de todo su ciclo de vida. En este documento se incluye orientación sobre las prácticas recomendadas para la implementación de cargas de trabajo fiables en AWS.
<b>Eficiencia del rendimiento</b>	Es la capacidad de utilizar de forma eficaz los recursos informáticos para satisfacer los requisitos del sistema, así como de mantener la eficiencia a medida que la demanda cambia y las tecnologías evolucionan.
<b>Optimización de costos</b>	Capacidad de ejecutar sistemas para ofrecer valor empresarial al menor precio posible.
<b>Sostenibilidad</b>	La sostenibilidad como disciplina aborda el impacto medioambiental, económico y social a largo plazo de sus actividades empresariales. La Comisión Mundial sobre el Medio Ambiente y el Desarrollo de las Naciones Unidas define el desarrollo sostenible como «aquel que permite satisfacer las necesidades del presente sin comprometer la habilidad de las futuras generaciones de satisfacer sus necesidades propias». Su organización o negocio puede tener repercusiones negativas en el medioambiente, como emisiones de carbono directas o indirectas, residuos no reciclables y daños a recursos compartidos, como el agua no contaminada.

# ¿QUÉ ES UNA REGIÓN?

- ▶ Una región de AWS es una ubicación geográfica que contiene varias zonas de disponibilidad (centros de datos).
- ▶ Cada región está físicamente separada de todas las demás regiones.
- ▶ Cada región tiene su propio suministro independiente de energía y agua.
- ▶ Las regiones son importantes para garantizar:
  - ▶ (1) Cumplimiento de datos
  - ▶ (2) Latencia (los centros de datos se colocan cerca de los usuarios para reducir la latencia)

**us-east-1** es la región más grande de AWS y contiene cinco zonas.



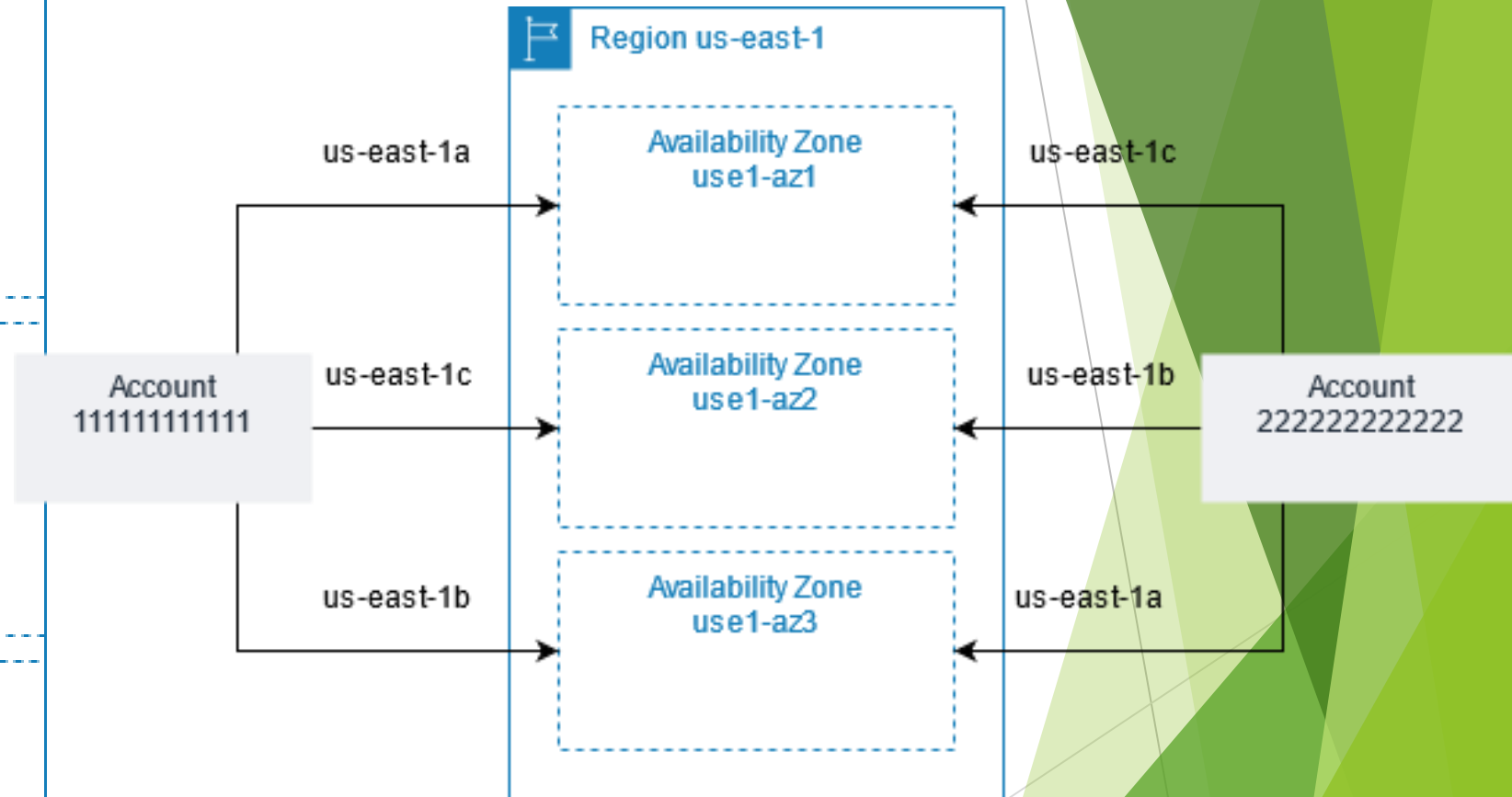
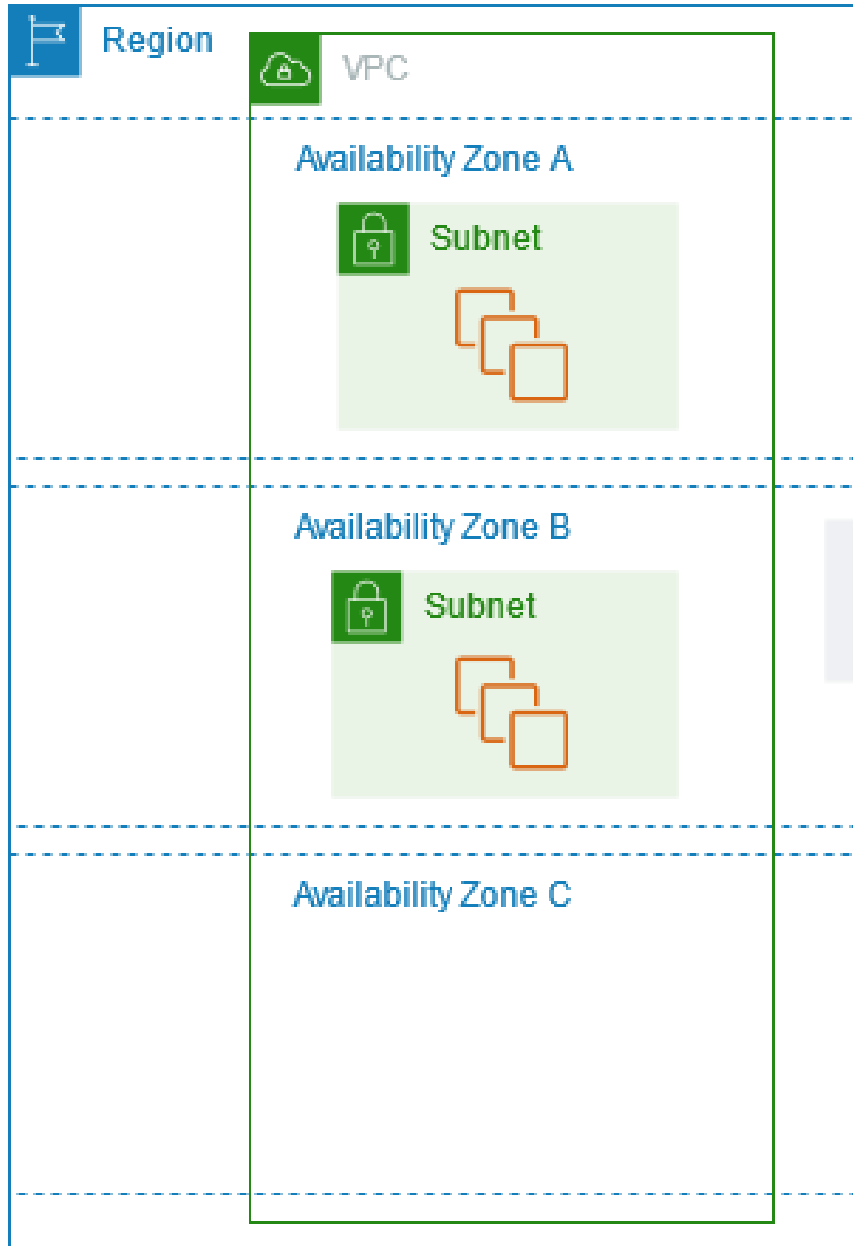


# ¿QUÉ ES UNA ZONA DE DISPONIBILIDAD?

- ▶ Una zona de disponibilidad de AWS es un centro de datos lógico que se encuentra en una determinada región.
- ▶ Hay dos o más zonas de disponibilidad en cada región de AWS.
- ▶ Con el fin de garantizar la disponibilidad y reducir la probabilidad de que dos zonas se caigan al mismo tiempo, cada zona tiene una red y alimentación redundante e independiente.

Nota: un centro de datos consta de un montón de servidores

Nota: un concepto erróneo común es que una sola zona equivale a un solo centro de datos. De hecho, cada zona está respaldada por uno o más centros de datos físicos.



## Section 3: Global vs Regional Services

Global Service  
Regional Service



### Identity and Access Management

AWS IAM

### AWS Compute

Amazon EC2

Amazon ECS

AWS Lambda

Amazon LightSail

### AWS Storage

Amazon S3

Amazon EBS

Amazon EFS

AWS Storage Gateway

### AWS Networking

Amazon VPC

AWS Direct Connect

### Databases

Amazon RDS

Amazon DynamoDB

Amazon RedShift

Amazon ElastiCache

### Elastic Load Balancing and Auto Scaling

Elastic Load Balancing

Auto Scaling

### Content Delivery and DNS Services

Amazon Route 53

Amazon CloudFront

### Monitoring and Logging Services

Amazon CloudWatch

AWS CloudTrail

### Notification Services

Amazon SNS

### Migration and Transfer

Database Migration Service

Server Migration Service

AWS Snowball

### Cloud Governance and Security

AWS GuardDuty

AWS KMS

AWS WAF & Shield

AWS CloudHSM

AWS Artifact

AWS Inspector

AWS Trusted Advisor

AWS Config

AWS Service Catalog

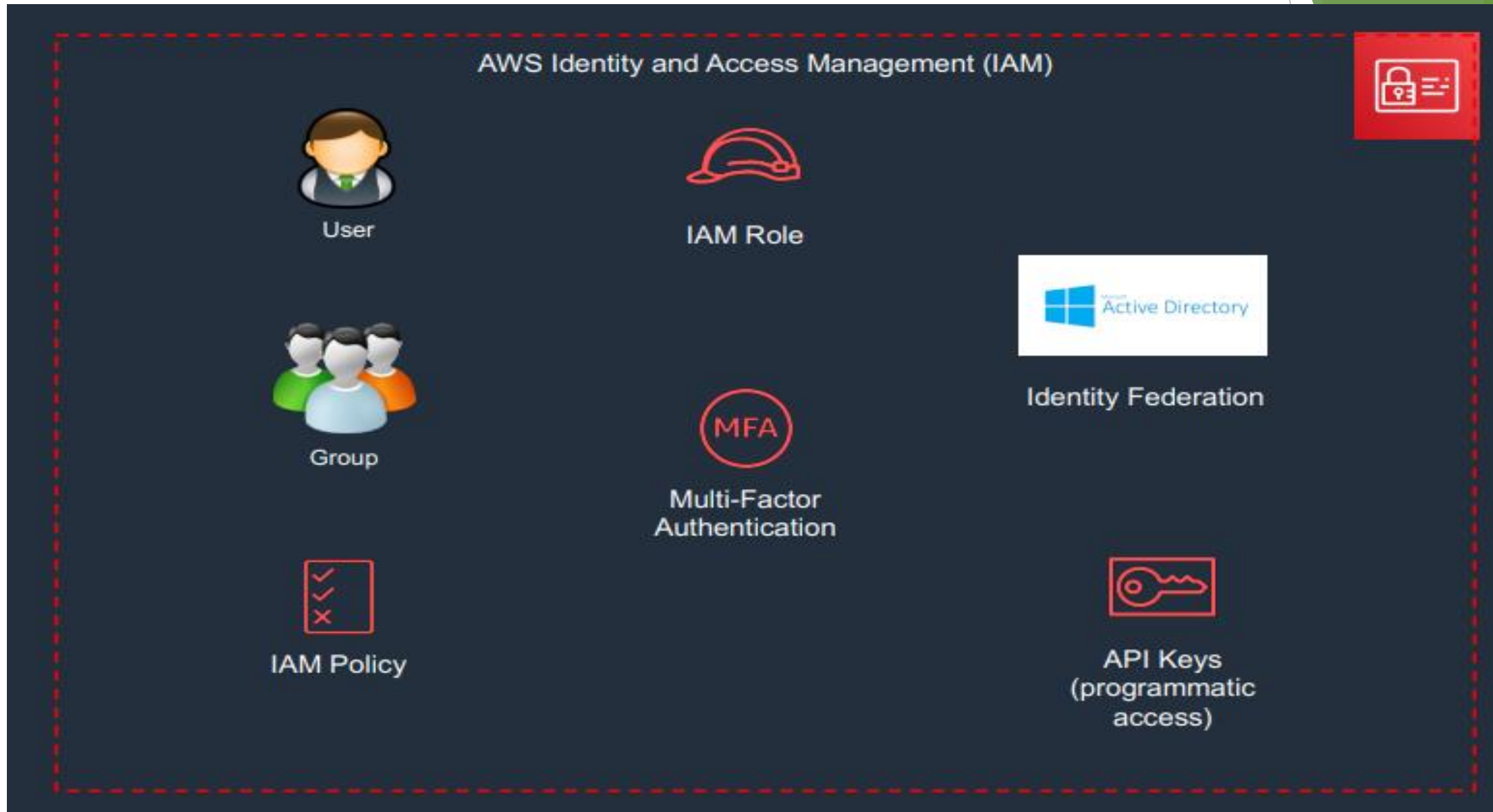
AWS Personal Health Dashboard



# MODELO DE PRECIOS AWS

- ▶ Bajo demanda (On-demand)
  - ▶ Utilizado para capacidad de cómputo y base de datos
  - ▶ Sin compromisos a largo plazo ni pagos por adelantado
- ▶ Instancias dedicadas (Dedicated Instances)
  - ▶ Disponible para Amazon EC2
  - ▶ El hardware está dedicado a un solo cliente
- ▶ Instancias puntuales (Spot Instances)
  - ▶ Adquirir capacidad sobrante sin compromisos
  - ▶ Grandes descuentos de tarifas por hora
- ▶ Reservas (Reservations)
  - ▶ Hasta un 75% de descuento a cambio de un compromiso a plazo

# Identity and Access Management (IAM)



# Identity and Access Management (IAM)

## ► IAM Users

Un usuario de IAM es una entidad que representa a una persona o servicio

### ► Se puede asignar:

- Un ID de clave de acceso y una clave de acceso secreta para el acceso programático a la API de AWS, CLI, SDK y otras herramientas de desarrollo
- Una contraseña para acceder a la consola de gestión
- Por defecto, los usuarios no pueden acceder a nada en su cuenta
- Las credenciales del usuario raíz de la cuenta son la dirección de correo electrónico utilizada para crear la cuenta y una contraseña
- La cuenta raíz tiene permisos administrativos completos y estos no se pueden restringir
- Mejores prácticas para cuentas raíz:
  - No use las credenciales de usuario raíz
  - No comparta las credenciales de usuario raíz
  - Cree un usuario de IAM y asigne permisos administrativos según sea necesario
  - Habilitar autenticación multifactor (MFA)

# Identity and Access Management (IAM)

## ► IAM Groups

Los grupos son colecciones de usuarios y tienen políticas adjuntas.

- Un grupo no es una identidad y no se puede identificar como principal en una política de IAM
- Use grupos para asignar permisos a los usuarios
- Usar el principio de privilegio mínimo al asignar permisos
- No puede anidar grupos (grupos dentro de grupos)

# Identity and Access Management (IAM)

## ► IAM Roles

### Funciones de gestión de identidades y accesos

- Los roles son creados y luego "asumidos" por entidades confiables y definen un conjunto de permisos para realizar solicitudes de servicio de AWS
- Con los roles de IAM puede delegar permisos a recursos para usuarios y servicios sin usar credenciales permanentes (por ejemplo, nombre de usuario y contraseña)
- Los usuarios de IAM o los servicios de AWS pueden asumir un rol para obtener credenciales de seguridad temporales que se puede utilizar para realizar llamadas a la API de AWS
- Puedes delegar usando roles
- No hay credenciales asociadas a un rol (contraseña o claves de acceso)

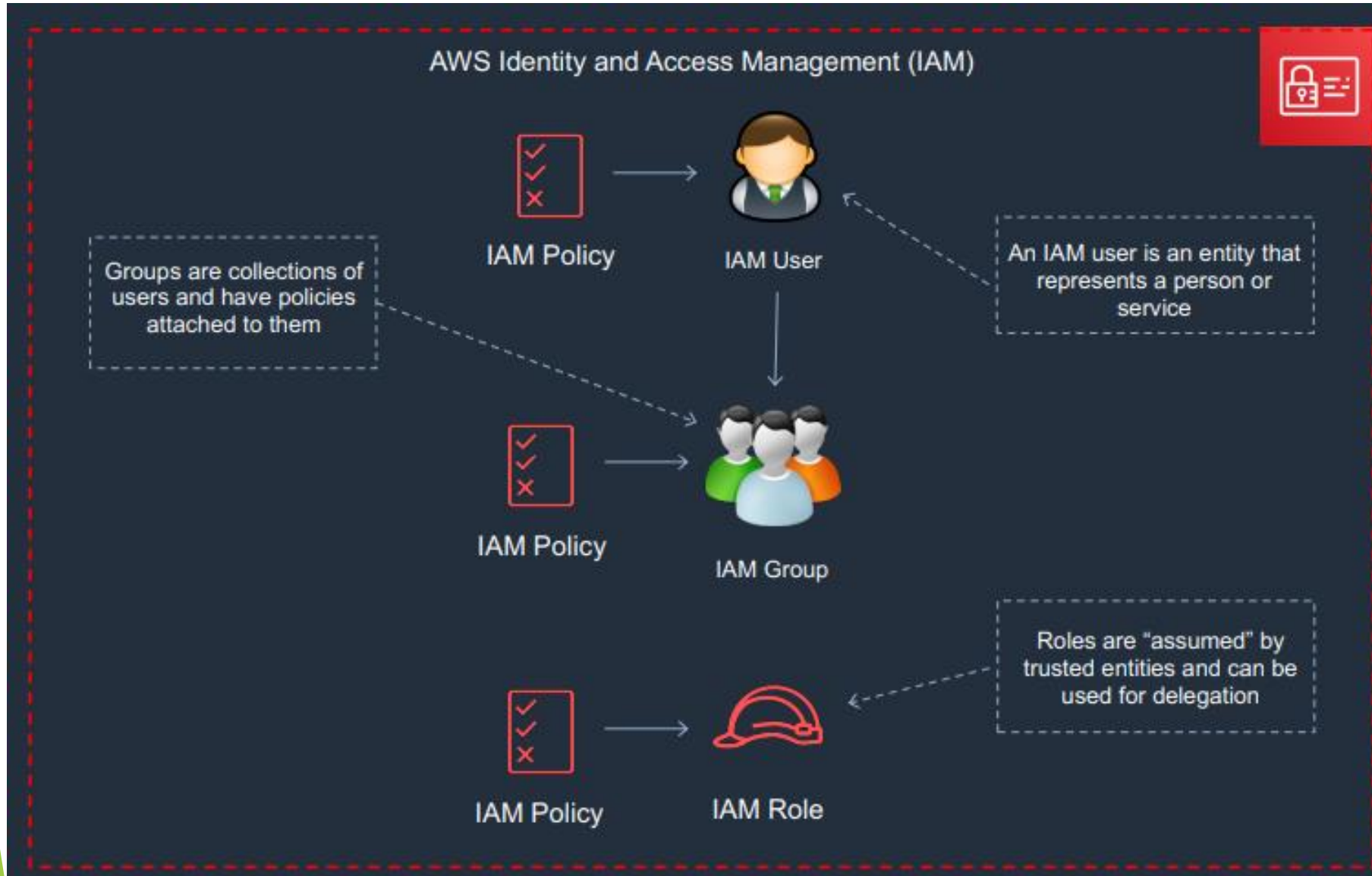
# Identity and Access Management (IAM)

## ► IAM Policies

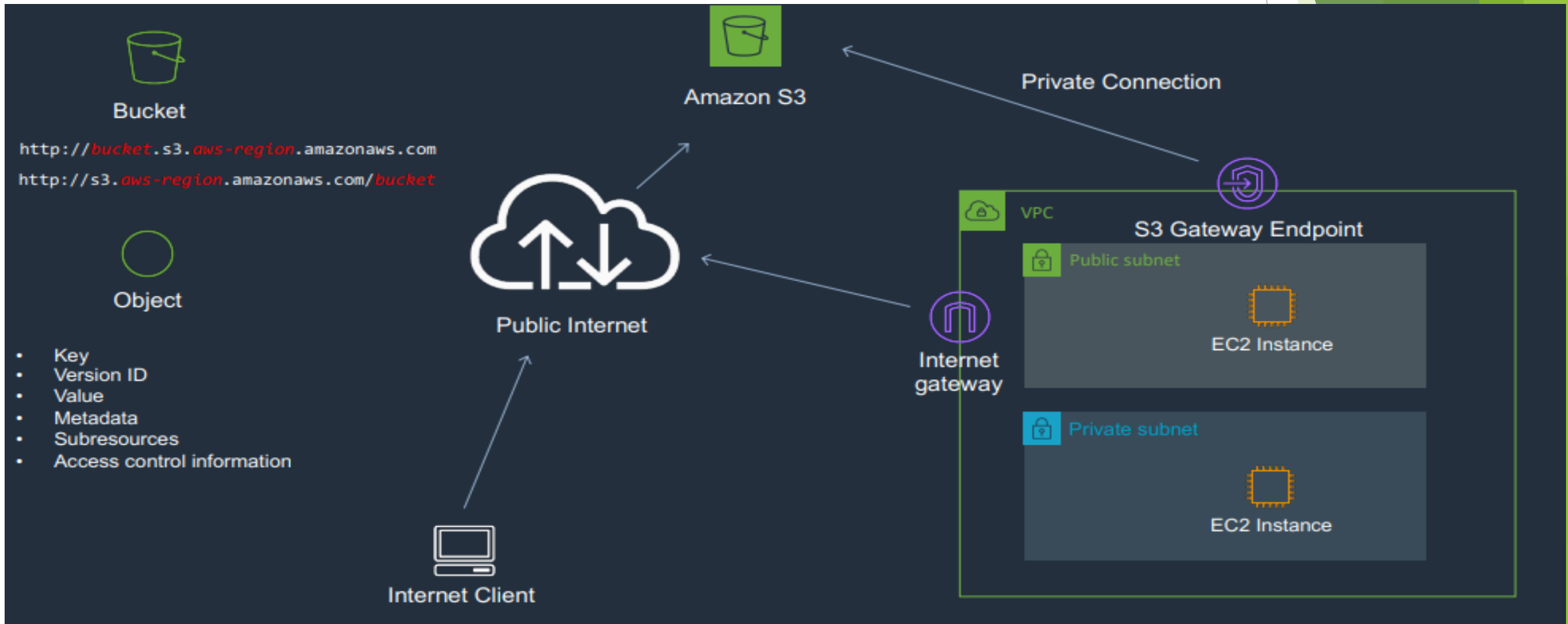
### Políticas de gestión de identidades y accesos

- Las políticas son documentos que definen permisos y se pueden aplicar a usuarios, grupos y roles
- Los documentos de política están escritos en JSON (par clave-valor que consta de un atributo y un valor)
- Todos los permisos se niegan implícitamente por defecto
- Se aplica la política más restrictiva
- El simulador de políticas de IAM es una herramienta para ayudarlo a comprender, probar y validar el efecto de las políticas de control de acceso

# Identity and Access Management (IAM)



# ¿Qué es Amazon Simple Storage Service (S3)?





# ¿Qué es Amazon Simple Storage Service (S3)?

Amazon S3 es un almacenamiento de objetos creado para almacenar y recuperar cualquier cantidad de datos desde cualquier lugar, sitios web y aplicaciones móviles, aplicaciones corporativas y datos de sensores o dispositivos IoT

Puede almacenar cualquier tipo de archivo en S3

S3 está diseñado para ofrecer una durabilidad del 99,999999999 %

Los casos de uso típicos incluyen:

- Copia de seguridad y almacenamiento: proporciona servicios de copia de seguridad y almacenamiento de datos para otros
- Alojamiento de aplicaciones: proporciona servicios que implementan, instalan y administran aplicaciones web
- Alojamiento de medios: cree una infraestructura redundante, escalable y de alta disponibilidad que aloje cargas y descargas de videos, fotos o música
- Entrega de software: aloje sus aplicaciones de software que los clientes pueden descargar
- Sitio web estático: puede configurar un sitio web estático para que se ejecute desde un depósito S3

# ¿Qué es Amazon Simple Storage Service (S3)?

Los archivos se almacenan en Bucket:

- Los bucket son carpetas de nivel raíz
- Los archivos pueden tener desde 0 bytes hasta 5 TB
- Hay almacenamiento ilimitado disponible
- S3 es un espacio de nombres universal, por lo que los nombres de los depósitos deben ser únicos a nivel mundial
- Sin embargo, creas tus bucket dentro de una REGIÓN
- Es una buena práctica crear depósitos en las regiones que están físicamente más cerca de sus usuarios para reducir la latencia
- Los objetos consisten en:
  - Clave (nombre del objeto)
  - Valor (dato formado por una secuencia de bytes)
  - ID de versión (utilizado para el control de versiones)
  - Metadatos (datos sobre los datos que se almacenan)

# Precios S3

## Almacenamiento

- ▶ Solicitudes
- ▶ Precios de gestión de almacenamiento
- ▶ Precios de transferencia de datos
- ▶ Transferencia de aceleración

# Clases de almacenamiento de Amazon S3



Frecuente		Poco frecuente		Infrecuente	
<ul style="list-style-type: none"> <li>• Activo, datos de acceso frecuente</li> <li>• Acceso en milisegundos</li> <li>• <math>\geq 3</math> AZ</li> <li>• Desde: \$0.0210/GB</li> </ul>	<ul style="list-style-type: none"> <li>• Datos con patrón de acceso cambiante</li> <li>• Acceso en milisegundos</li> <li>• <math>\geq 3</math> AZ</li> <li>• Desde: \$ 0.0125 hasta \$ 0.0210</li> <li>• Costo x monitoreo de obj.</li> <li>• Duración min de almacenamiento</li> </ul>	<ul style="list-style-type: none"> <li>• Datos accedidos de forma infrecuente</li> <li>• Acceso en milisegundos</li> <li>• <math>\geq 3</math> AZ</li> <li>• Desde: \$0.0125/GB</li> <li>• Costo por obtención del dato por GB</li> <li>• Min storage duration</li> <li>• Min object size</li> </ul>	<ul style="list-style-type: none"> <li>• Datos reproducible</li> <li>• Acceso en milisegundos</li> <li>• 1 AZ</li> <li>• Desde: \$0.0100/GB</li> <li>• Costo por obtención del dato por GB</li> <li>• Duración min de almacenamiento y tamaño</li> </ul>	<ul style="list-style-type: none"> <li>• Datos historicos</li> <li>• Acceso en min o horas</li> <li>• <math>\geq 3</math> AZ</li> <li>• Desde: \$0.0040/GB</li> <li>• Costo por obtención del dato por GB</li> <li>• Duración min de almacenamiento y tamaño</li> </ul>	<ul style="list-style-type: none"> <li>• Datos historicos</li> <li>• Acceso en horas</li> <li>• <math>\geq 3</math> AZ</li> <li>• Desde: \$0.00099/GB</li> <li>• Costo por obtención del dato por GB</li> <li>• Duración min de almacenamiento y tamaño</li> </ul>

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes



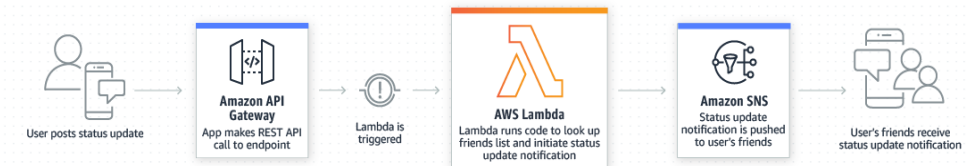
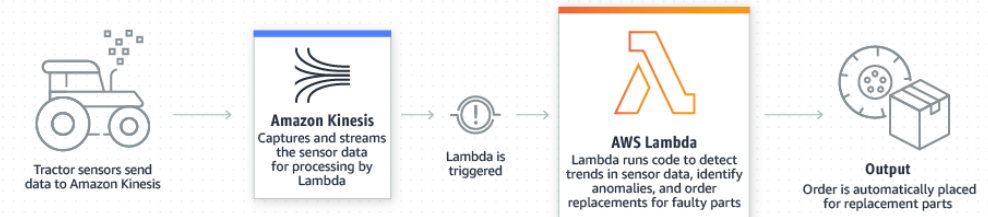
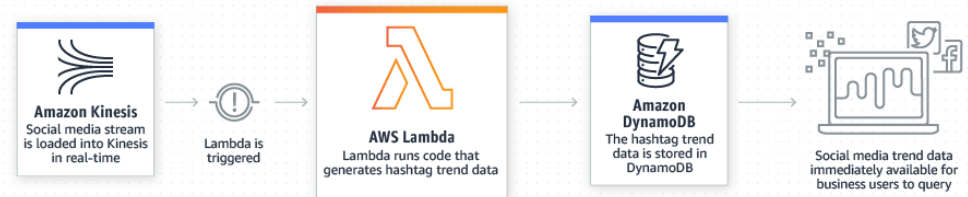
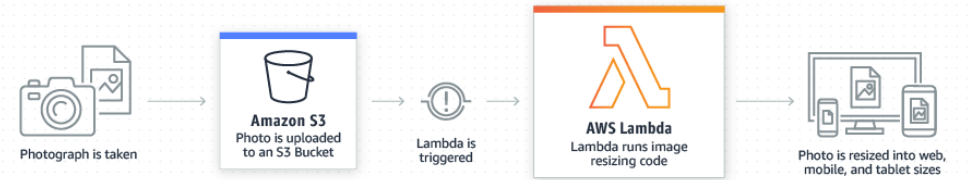
# AWS LAMBDA

Lambda es un servicio informático que le permite ejecutar código sin aprovisionar ni administrar servidores. Lambda ejecuta su código en una infraestructura informática de alta disponibilidad y realiza toda la administración de los recursos informáticos, incluido el mantenimiento del sistema operativo y del servidor, el aprovisionamiento de capacidad y el escalado automático.

**Serverless:** La ejecución de aplicaciones sin servidor significa que nosotros y nuestro equipo podemos centrarnos en el desarrollo de código y el dominio comercial en lugar de gastar esfuerzos en tareas operativas y administrativas.

Con Lambda podemos ejecutar casi cualquier tipo de aplicación o servicio de back-end siempre que el código esté en uno de los lenguajes y marcos compatibles : **Node.js, Python, Java, .Net, Go y Ruby.**

Las funciones de Lambda se pueden invocar mediante la API de Lambda o reaccionando a eventos de otros servicios de AWS, como una solicitud de API Gateway, una actualización en DynamoDB o S3, etc.



# Conceptos Clave

## ► Trigger (Disparador)

Un disparador puede ser un servicio de AWS configurado para invocar una función de Lambda. También puede invocar un mapeo de origen de eventos que es un recurso de Lambda que lee de un flujo de datos o una cola y luego invoca una función.

Básicamente, un disparador es todo lo que termina ejecutando una función de Lambda.

## ► Event (Evento)

Es un documento que contiene datos formateados en JSON que utiliza Lambda para procesar convirtiéndolos en un objeto que se pasa al código de la función.

```
{  
  "nombre" : "Kevin",  
  "apellido" : "Reyes"  
}
```

## ► Execution environment (Entorno de ejecución)

Es un entorno seguro y aislado para el tiempo de ejecución de la función Lambda. Dado que Lambda es un servicio completamente administrado, este entorno de ejecución maneja todos los procesos y recursos que nuestra función y todas sus dependencias requieren para ejecutarse.

# Conceptos Clave

## ► **Deployment package (Paquete de implementación)**

La función Lambda se puede implementar en dos tipos de paquetes:

- .ziparchivo: contiene todo el código y las dependencias, y Lambda ejecuta el entorno de ejecución
- imagen de contenedor: imagen que contiene el código y las dependencias, así como el sistema operativo y el tiempo de ejecución de Lambda Trigger (Disparador)

## ► **Capa (Layer)**

Para reducir los archivos de implementación y hacer que la implementación sea más rápida, las capas se pueden usar para empaquetar bibliotecas y dependencias requeridas por la función.

Las capas mejoran el código compartido y la separación de responsabilidades y se nos permite tener hasta 5 capas por función. Todo el contenido de la Capa estará disponible en el /opt directorio del Event (Evento)

## ► **Extensión (Extension)**

Como su nombre lo dice, puede extender nuestras funcionalidades de Lambdas integrándolas con otras herramientas como monitoreo, observabilidad, etc.



# Conceptos Clave

## ► **Deployment package (Paquete de implementación)**

La función Lambda se puede implementar en dos tipos de paquetes:

- .ziparchivo: contiene todo el código y las dependencias, y Lambda ejecuta el entorno de ejecución
- imagen de contenedor: imagen que contiene el código y las dependencias, así como el sistema operativo y el tiempo de ejecución de Lambda Trigger (Disparador)

## ► **Capa (Layer)**

Para reducir los archivos de implementación y hacer que la implementación sea más rápida, las capas se pueden usar para empaquetar bibliotecas y dependencias requeridas por la función.

Las capas mejoran el código compartido y la separación de responsabilidades y se nos permite tener hasta 5 capas por función. Todo el contenido de la Capa estará disponible en el /opt directorio del Event (Evento)

## ► **Extensión (Extension)**

Como su nombre lo dice, puede extender nuestras funcionalidades de Lambdas integrándolas con otras herramientas como monitoreo, observabilidad, etc.

# Conceptos Clave

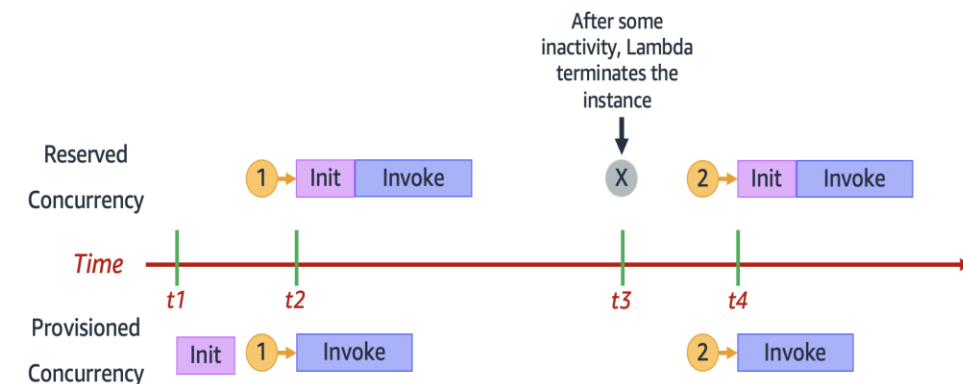
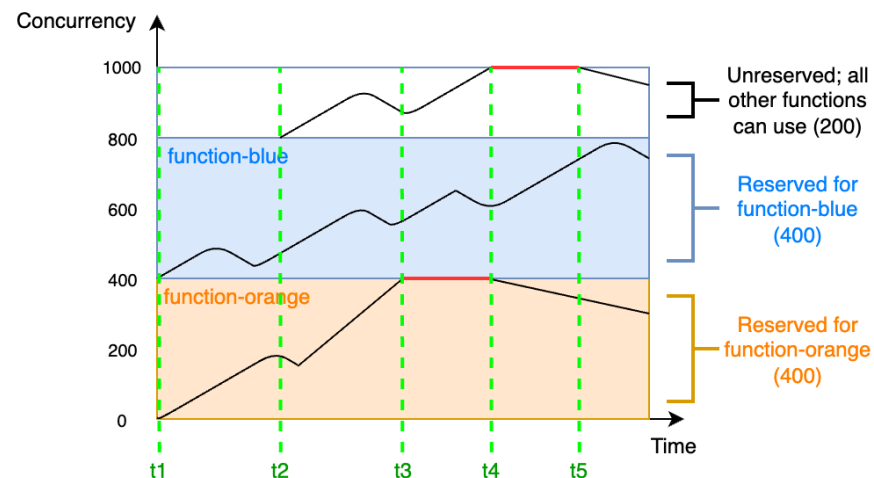
## ► Concurrencia / Escalado (Concurrency / Scaling)

Cuando se activa la función, se aprovisiona una instancia de la función que le permite procesar los eventos entrantes. Cada evento creará una nueva instancia. Si una instancia se está ejecutando cuando llega un nuevo evento, Lambda aprovisionará una nueva instancia creando un escenario de simultaneidad.

Cuando la función se invoca más rápido de lo que puede procesar el resultado, Lambda comienza a aumentar la cantidad de instancias que ejecutan nuestro código. Lo contrario sucede cuando el tráfico entrante disminuye y la cantidad de instancias en ejecución se reduce para congelarse o detenerse.

Tipos de concurrencia:

- Reservada
- Aprovisionada



# Conceptos Clave

## ► **Asíncrono / Destino (Asynchronous / Destination)**

Es posible configurar una función Lambda para que se invoque de forma síncrona (el cliente espera la respuesta de la función) o asíncrona (donde Lambda pone en cola los eventos para que se procesen en un orden específico).

En escenarios de arquitectura asíncrona, Lambda usa destinos para enviar los eventos. Estos destinos se pueden usar para eventos fallidos (que deben ponerse en cola y luego reprocesarse) o procesarse correctamente (para enviarse a otros servicios de AWS).

## ► **Asignaciones de fuentes de eventos (Event source mappings)**

Esta es la funcionalidad que permite a Lambda procesar eventos de flujos de datos o colas mediante la lectura de elementos de Amazon SQS, Amazon Kinesis, DynamoDB u otros.

Los elementos de estos orígenes se pueden enviar en lotes que recibe el recurso de asignación de orígenes de eventos y cada elemento puede tener cientos o miles de elementos.

# Links

- ▶ <https://aws-sdk-pandas.readthedocs.io/en/2.14.0/install.html#aws-lambda-layer>

