

AWS Certified Solutions Architect - Associate (SAA-C03) Guía de examen

Este documento incluye las versiones en [español \(Latinoamérica\)](#) y [español \(España\)](#).

Introducción

El examen AWS Certified Solutions Architect - Associate (SAA-C03) está diseñado para personas que desempeñan un rol en la arquitectura de soluciones. En el examen, se verifica la capacidad del candidato de utilizar las tecnologías de AWS para diseñar soluciones basadas en el AWS Well-Architected Framework.

En este examen, también se certifica la capacidad de un candidato para completar las siguientes tareas:

- Diseñar soluciones que incorporan servicios de AWS para satisfacer los requisitos actuales de la empresa y las futuras necesidades pronosticadas.
- Diseñar arquitecturas seguras, resistentes, de alto rendimiento y rentables.
- Revisar las soluciones actuales y elaborar mejoras.

Descripción del candidato objetivo

El candidato objetivo debe tener, al menos, 1 año de experiencia comprobable en el diseño de soluciones de nube que utilizan servicios de AWS.

Para ver una lista detallada de las herramientas y tecnologías específicas que podrían abordarse en el examen, así como listas de los servicios de AWS que se encuentran dentro y fuera del alcance, consulte el Apéndice.

Contenido del examen

Tipos de respuesta

En el examen, hay dos tipos de preguntas:

- **Opciones múltiples:** hay una respuesta correcta y tres incorrectas (distractoras).
- **Respuesta múltiple:** hay dos o más respuestas correctas entre cinco o más opciones.

Seleccione una o más respuestas que completen la afirmación o respondan a la pregunta de la mejor manera. Las distractoras o respuestas incorrectas son opciones que podría elegir un candidato que no tenga un buen nivel de conocimientos o habilidades. Por lo general, las distractoras son respuestas verosímiles que coinciden con el área de contenido.

Las preguntas sin respuesta se califican como incorrectas. No hay penalización por adivinar. El examen incluye 50 preguntas que afectarán el puntaje.

Contenido sin puntaje

El examen incluye 15 preguntas sin puntaje que no afectan el puntaje total. AWS recopila información sobre el rendimiento de los candidatos en estas preguntas sin puntaje a fin de evaluarlas para su uso como preguntas con puntaje en el futuro. Estas preguntas sin puntaje no están identificadas en el examen.

Resultados del examen

El examen AWS Certified Solutions Architect – Associate es un examen que se aprueba o se desaprueba. El puntaje se obtiene según un estándar mínimo que establecen los profesionales de AWS en función de las prácticas recomendadas y las pautas del sector de la certificación.

El informe de los resultados del examen es un puntaje en la escala del 100 al 1000. El puntaje mínimo para aprobar es 720. El puntaje muestra cómo le fue en el examen en general y si lo aprobó o no. Los modelos de puntajes en escala ayudan a equiparar puntuaciones de varios formatos de examen que pueden tener niveles de dificultad un poco diferentes.

El informe del puntaje puede contener una tabla de clasificación de su rendimiento en cada sección. Esta información proporciona comentarios generales sobre el rendimiento en el examen. En el examen, se usa un modelo de puntaje compensatorio, lo que significa que no es necesario aprobar cada sección. Solo necesita aprobar el examen general.

Cada sección del examen tiene una ponderación específica, por lo que algunas contienen más preguntas que otras. En la tabla, se presenta información general que resalta sus fortalezas y debilidades. Interprete los comentarios de cada sección con prudencia. Los candidatos que aprueben el examen no recibirán esta información adicional.

Descripción del contenido

Esta guía de examen incluye ponderaciones, dominios de prueba y enunciados de tareas para el examen. No es una descripción completa del contenido del examen. Sin embargo, hay contexto adicional disponible para cada uno de los enunciados de las tareas que ayudan a guiar la preparación del examen. En la siguiente tabla, se enumeran los principales dominios de contenido y sus ponderaciones. La tabla precede a la descripción del contenido del examen, que incluye el contexto adicional. El porcentaje de cada dominio solo representa el contenido que recibe un puntaje.

Dominio	% del examen
Dominio 1: diseñar arquitecturas seguras	30 %
Dominio 2: diseñar arquitecturas resistentes	26 %
Dominio 3: diseñar arquitecturas de alto rendimiento	24 %
Dominio 4: diseñar arquitecturas rentables	20 %
TOTAL	100 %

Dominio 1: diseñar arquitecturas seguras

Enunciado de la tarea 1: diseñar un acceso seguro a los recursos de AWS.

Conocimientos:

- controles de acceso y administración en varias cuentas
- servicios de identidad y acceso federados de AWS (por ejemplo, AWS Identity and Access Management [IAM], AWS Single Sign-On [AWS SSO])
- infraestructura global de AWS (por ejemplo, regiones de AWS, zonas de disponibilidad)
- prácticas recomendadas de seguridad de AWS (por ejemplo, el principio de mínimo privilegio)
- el modelo de responsabilidad compartida de AWS

Habilidades:

- Aplicar las prácticas recomendadas de seguridad de AWS a los usuarios de IAM y a los usuarios raíz (por ejemplo, autenticación multifactor [MFA]).
- Diseñar un modelo de autorización flexible que incluya usuarios, grupos, roles y políticas de IAM.
- Diseñar una estrategia de control de acceso basada en roles (por ejemplo, AWS Security Token Service [AWS STS], cambio de roles, acceso entre cuentas).
- Diseñar una estrategia de seguridad para varias cuentas de AWS (por ejemplo, AWS Control Tower, políticas de control de servicios [SCP]).
- Determinar el uso adecuado de las políticas de recursos para los servicios de AWS.
- Determinar cuándo federar un servicio de directorio con roles de IAM.

Enunciado de la tarea 2: diseñar cargas de trabajo y aplicaciones seguras.

Conocimientos:

- configuración de aplicaciones y seguridad de credenciales
- puntos de enlace de servicios de AWS
- puertos de control, protocolos y tráfico en la red en AWS
- acceso seguro a las aplicaciones
- servicios de seguridad con casos de uso adecuados (por ejemplo, Amazon Cognito, Amazon GuardDuty y Amazon Macie)
- vectores de amenazas externas a AWS (por ejemplo, denegación de servicio distribuido [DDoS], inyección SQL)

Habilidades:

- Diseñar arquitecturas de VPC con componentes de seguridad (por ejemplo, grupos de seguridad, tablas de enrutamiento, ACL de red, gateways NAT).
- Determinar las estrategias de segmentación de red (por ejemplo, el uso de subredes públicas y subredes privadas).
- Integrar a los servicios de AWS para proteger las aplicaciones (por ejemplo, AWS Shield, AWS WAF, AWS SSO, AWS Secrets Manager).
- Proteger las conexiones de red externas desde la nube de AWS y hacia ella (por ejemplo, VPN, AWS Direct Connect).

Enunciado de la tarea 3: determinar los controles de seguridad de datos adecuados.

Conocimientos:

- acceso a los datos y gobernanza
- recuperación de datos
- retención y clasificación de datos
- cifrado y administración adecuada de claves

Habilidades:

- Alinear las tecnologías de AWS para cumplir los requisitos de conformidad.
- Cifrar datos en reposo (por ejemplo, AWS Key Management Service [AWS KMS]).
- Cifrar datos en tránsito (por ejemplo, AWS Certificate Manager [ACM] mediante TLS).
- Implementar políticas de acceso para claves de cifrado.
- Implementar copias de seguridad y replicaciones de datos.
- Implementar políticas de acceso, ciclo de vida y protección de datos.
- Rotar claves de cifrado y renovar certificados.

Dominio 2: diseñar arquitecturas resistentes

Enunciado de la tarea 1: diseñar arquitecturas escalables y con acoplamiento débil.

Conocimientos:

- creación y administración de API (por ejemplo, Amazon API Gateway, API de REST)
- servicios administrados de AWS con casos de uso adecuados (por ejemplo, AWS Transfer Family, Amazon Simple Queue Service [Amazon SQS], Secrets Manager)
- estrategias de almacenamiento en caché
- principios de diseño para microservicios (por ejemplo, cargas de trabajo sin estado en comparación con cargas de trabajo con estado)
- arquitecturas impulsadas por eventos
- escalado horizontal y escalado vertical
- cómo utilizar correctamente los aceleradores de borde (por ejemplo, red de entrega de contenido [CDN])
- cómo migrar aplicaciones a contenedores
- conceptos de equilibrio de carga (por ejemplo, Application Load Balancer)
- arquitecturas de varios niveles
- conceptos de colas y mensajería (por ejemplo, publicación/suscripción)
- tecnologías y patrones sin servidor (por ejemplo, AWS Fargate, AWS Lambda)
- tipos de almacenamiento con características asociadas (por ejemplo, objeto, archivo, bloque)
- el aprovisionamiento de contenedores (por ejemplo, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS])
- cuándo usar réplicas de lectura
- aprovisionamiento del flujo de trabajo (por ejemplo, AWS Step Functions)

Habilidades:

- Diseñar arquitecturas impulsadas por eventos, de microservicios o de varios niveles en función de los requisitos.
- Determinar estrategias de escalado para componentes utilizados en un diseño de arquitectura.
- Determinar los servicios de AWS necesarios para lograr un acoplamiento débil en función de los requisitos.
- Determinar cuándo usar contenedores.
- Determinar cuándo usar tecnologías y patrones sin servidor.
- Recomendar tecnologías de cómputo, almacenamiento, redes y bases de datos adecuadas en función de los requisitos.
- Usar servicios de AWS diseñados específicamente para cargas de trabajo.

Enunciado de la tarea 2: diseñar arquitecturas de alta disponibilidad o tolerantes a errores.

Conocimientos:

- infraestructura global de AWS (por ejemplo, regiones de AWS, zonas de disponibilidad, Amazon Route 53)
- servicios administrados de AWS con casos de uso adecuados (por ejemplo, Amazon Comprehend, Amazon Polly)
- conceptos básicos de redes (por ejemplo, tablas de enrutamiento)
- estrategias de recuperación de desastres (DR) (por ejemplo, copia de seguridad y restauración, luz piloto, espera activa, conmutación por error Active-Active, objetivo de punto de recuperación [RPO], objetivo de tiempo de recuperación [RTO])
- patrones de diseño distribuidos
- estrategias de conmutación por error
- infraestructura inmutable
- conceptos de equilibrio de carga (por ejemplo, Application Load Balancer)
- conceptos de proxy (por ejemplo, Amazon RDS Proxy)
- Service Quotas y reducción controlada (por ejemplo, cómo configurar Service Quotas para una carga de trabajo en un entorno en espera)
- opciones y características de almacenamiento (por ejemplo, durabilidad, replicación)
- visibilidad de las cargas de trabajo (por ejemplo, AWS X-Ray)

Habilidades:

- Determinar estrategias de automatización para garantizar la integridad de la infraestructura.
- Determinar los servicios de AWS necesarios para brindar una arquitectura de alta disponibilidad o tolerante a errores en todas las zonas de disponibilidad o regiones de AWS.
- Identificar métricas en función de los requisitos de la empresa para ofrecer una solución de alta disponibilidad.
- Implementar diseños para mitigar puntos únicos de error.
- Implementar estrategias para garantizar la durabilidad y disponibilidad de los datos (por ejemplo, copias de seguridad).
- Seleccionar una estrategia de DR adecuada para cumplir con los requisitos de la empresa.
- Usar servicios de AWS que mejoren la fiabilidad de las aplicaciones heredadas y las que no son creadas para la nube (por ejemplo, cuando no es posible realizar cambios en las aplicaciones).
- Usar servicios de AWS diseñados específicamente para cargas de trabajo.

Dominio 3: diseñar arquitecturas de alto rendimiento

Enunciado de la tarea 1: determinar soluciones de almacenamiento escalables o de alto rendimiento.

Conocimientos:

- soluciones de almacenamiento híbrido para cumplir con los requisitos de la empresa
- servicios de almacenamiento con casos de uso adecuados (por ejemplo, Amazon S3, Amazon Elastic File System [Amazon EFS], Amazon Elastic Block Store [Amazon EBS])
- tipos de almacenamiento con características asociadas (por ejemplo, objeto, archivo, bloque)

Habilidades:

- Determinar los servicios y configuraciones de almacenamiento que cumplen con las demandas de rendimiento.
- Determinar los servicios de almacenamiento que pueden escalar para adaptarse a necesidades futuras.

Enunciado de la tarea 2: diseñar soluciones de cómputo elásticas y de alto rendimiento.

Conocimientos:

- servicios de cómputo de AWS con casos de uso adecuados (por ejemplo, AWS Batch, Amazon EMR, Fargate)
- conceptos informáticos distribuidos compatibles con la infraestructura global y los servicios de borde de AWS
- conceptos de colas y mensajería (por ejemplo, publicación/suscripción)
- capacidades de escalabilidad con casos de uso adecuados (por ejemplo, Amazon EC2 Auto Scaling, AWS Auto Scaling)
- tecnologías y patrones sin servidor (por ejemplo, Lambda, Fargate)
- aprovisionamiento de contenedores (por ejemplo, Amazon ECS, Amazon EKS)

Habilidades:

- Desacoplar cargas de trabajo para que los componentes puedan escalar de forma independiente.
- Identificar métricas y condiciones para realizar acciones de escalado.
- Seleccionar opciones y características de cómputo adecuadas (por ejemplo, tipos de instancias EC2) para cumplir con los requisitos de la empresa.
- Seleccionar el tipo y el tamaño de recursos adecuados (por ejemplo, la cantidad de memoria Lambda) para cumplir con los requisitos de la empresa.

Enunciado de la tarea 3: determinar soluciones de bases de datos de alto rendimiento.

Conocimientos:

- infraestructura global de AWS (por ejemplo, regiones de AWS, zonas de disponibilidad)
- estrategias y servicios de almacenamiento en caché (por ejemplo, Amazon ElastiCache)
- patrones de acceso a datos (por ejemplo, de lectura intensiva en comparación con escritura intensiva)
- planificación de la capacidad de la base de datos (por ejemplo, unidades de capacidad, tipos de instancias, IOPS provisionadas)
- conexiones y proxies de bases de datos

- motores de bases de datos con casos de uso apropiados (por ejemplo, migraciones heterogéneas, migraciones homogéneas)
- replicación de bases de datos (por ejemplo, réplicas de lectura)
- tipos y servicios de bases de datos (por ejemplo, sin servidor, relacionales en comparación con no relacionales, en la memoria)

Habilidades:

- Configurar réplicas de lectura para cumplir con los requisitos de la empresa.
- Diseñar arquitecturas de bases de datos.
- Determinar un motor de base de datos adecuado (por ejemplo, MySQL en comparación con PostgreSQL).
- Determinar un tipo de base de datos adecuado (por ejemplo, Amazon Aurora, Amazon DynamoDB).
- Integrar el almacenamiento en caché para cumplir con los requisitos de la empresa.

Enunciado de la tarea 4: determinar arquitecturas de red escalables o de alto rendimiento.

Conocimientos:

- servicios de redes de borde con casos de uso apropiados (por ejemplo, Amazon CloudFront, AWS Global Accelerator)
- cómo diseñar una arquitectura de red (por ejemplo, niveles de subred, enrutamiento, direccionamiento IP)
- conceptos de equilibrio de carga (por ejemplo, Application Load Balancer)
- opciones de conexión de red (por ejemplo, AWS VPN, Direct Connect, AWS PrivateLink)

Habilidades:

- Crear una topología de red para varias arquitecturas (por ejemplo, global, híbrida o de varios niveles).
- Determinar las configuraciones de red que pueden escalar para adaptarse a necesidades futuras.
- Determinar la ubicación adecuada de los recursos para cumplir con los requisitos de la empresa.
- Seleccionar la estrategia de equilibrio de carga adecuada.

Enunciado de la tarea 5: determinar soluciones de transformación e ingesta de datos de alto rendimiento.

Conocimientos:

- servicios de análisis y visualización de datos con casos de uso adecuados (por ejemplo, Amazon Athena, AWS Lake Formation, Amazon QuickSight)
- patrones de ingesta de datos (por ejemplo, frecuencia)
- servicios de transferencia de datos con casos de uso adecuados (por ejemplo, AWS DataSync, AWS Storage Gateway)
- servicios de transformación de datos con casos de uso adecuados (por ejemplo, AWS Glue)
- acceso seguro a los puntos de acceso de ingesta
- tamaños y velocidades necesarias para cumplir con los requisitos de la empresa
- servicios de transmisión de datos con casos de uso adecuados (por ejemplo, Amazon Kinesis)

Habilidades:

- Crear y proteger lagos de datos.
- Diseñar arquitecturas de flujos de datos.
- Diseñar soluciones de transferencia de datos.
- Implementar estrategias de visualización.
- Seleccionar las opciones de cómputo adecuadas para el procesamiento de datos (por ejemplo, Amazon EMR).
- Seleccionar configuraciones adecuadas para la ingesta.
- Transformar datos entre formatos (por ejemplo, de .csv a .parquet).

Dominio 4: diseñar arquitecturas rentables

Enunciado de la tarea 1: diseñar soluciones de almacenamiento rentables.

Conocimientos:

- opciones de acceso (por ejemplo, un S3 bucket con almacenamiento de objetos con pagos por solicitante)
- características del servicio de administración de costos de AWS (por ejemplo, etiquetas de asignación de costos, facturación en varias cuentas)
- herramientas de administración de costos de AWS con casos de uso apropiados (por ejemplo, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- servicios de almacenamiento de AWS con casos de uso apropiados (por ejemplo, Amazon FSx, Amazon EFS, Amazon S3, Amazon EBS)
- estrategias de copias de seguridad
- opciones de almacenamiento en bloque (por ejemplo, tipos de volumen de unidad de disco duro [HDD], tipos de volumen de unidad de estado sólido [SSD])
- ciclos de vida de los datos
- opciones de almacenamiento híbrido (por ejemplo, DataSync, Transfer Family, Storage Gateway)
- patrones de acceso al almacenamiento
- clasificación de almacenamiento (por ejemplo, clasificación en frío para el almacenamiento de objetos)
- tipos de almacenamiento con características asociadas (por ejemplo, objeto, archivo, bloque)

Habilidades:

- Diseñar estrategias de almacenamiento adecuadas (por ejemplo, cargas por lotes a Amazon S3 en comparación con cargas individuales).
- Determinar el tamaño de almacenamiento correcto para una carga de trabajo.
- Determinar el método de menor costo de transferencia de datos para una carga de trabajo al almacenamiento de AWS.
- Determinar cuándo se requiere el Auto Scaling del almacenamiento.
- Administrar los ciclos de vida de objetos S3.
- Seleccionar la solución de copia de seguridad o archivado adecuada.
- Seleccionar el servicio adecuado para la migración de datos a los servicios de almacenamiento.
- Seleccionar el nivel de almacenamiento adecuado.
- Seleccionar el ciclo de vida de datos correcto para el almacenamiento.
- Seleccionar el servicio de almacenamiento más redituable para una carga de trabajo.

Enunciado de la tarea 2: diseñar soluciones de cómputo rentables.

Conocimientos:

- características del servicio de administración de costos de AWS (por ejemplo, etiquetas de asignación de costos, facturación en varias cuentas)
- herramientas de administración de costos de AWS con casos de uso apropiados (por ejemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- infraestructura global de AWS (por ejemplo, regiones de AWS, zonas de disponibilidad)
- opciones de compra de AWS (por ejemplo, instancias tipo spot, instancias reservadas, Savings Plans)
- estrategias de cómputo distribuidas (por ejemplo, procesamiento de borde)
- opciones de cómputo híbridas (por ejemplo, AWS Outposts, AWS Snowball Edge)
- tipos, familias y tamaños de instancias (por ejemplo, memoria optimizada, cómputo optimizado, virtualización)
- optimización de la utilización de cómputo (por ejemplo, contenedores, informática sin servidor, microservicios)
- estrategias de escalado (por ejemplo, Auto Scaling, hibernación)

Habilidades:

- Determinar una estrategia de equilibrio de carga adecuada (por ejemplo, Application Load Balancer [capa 7] en comparación con Network Load Balancer [capa 4], en comparación con Gateway Load Balancer).
- Determinar métodos y estrategias de escalado adecuados para cargas de trabajo elásticas (por ejemplo, horizontal en comparación con vertical, hibernación EC2).
- Determinar servicios de cómputo de AWS redituables con casos de uso adecuados (por ejemplo, Lambda, Amazon EC2, Fargate).
- Determinar la disponibilidad necesaria para diferentes clases de cargas de trabajo (por ejemplo, cargas de trabajo de producción, cargas de trabajo que no son de producción).
- Seleccionar la familia de instancias adecuada para una carga de trabajo.
- Seleccionar el tamaño de instancia adecuado para una carga de trabajo.

Enunciado de la tarea 3: diseñar soluciones de bases de datos rentables.

Conocimientos:

- características del servicio de administración de costos de AWS (por ejemplo, etiquetas de asignación de costos, facturación en varias cuentas)
- herramientas de administración de costos de AWS con casos de uso apropiados (por ejemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- estrategias de almacenamiento en caché
- políticas de retención de datos
- planificación de capacidad de las bases de datos (por ejemplo, unidades de capacidad)
- conexiones y proxies de bases de datos
- motores de bases de datos con casos de uso apropiados (por ejemplo, migraciones heterogéneas, migraciones homogéneas)
- replicación de bases de datos (por ejemplo, réplicas de lectura)
- tipos y servicios de bases de datos (por ejemplo, relacionales en comparación con no relacionales, Aurora o DynamoDB)

Habilidades:

- Diseñar políticas de copias de seguridad y retención adecuadas (por ejemplo, frecuencia de instantáneas).
- Determinar un motor de base de datos adecuado (por ejemplo, MySQL en comparación con PostgreSQL).
- Determinar servicios de bases de datos de AWS redituables con casos de uso adecuados (por ejemplo, DynamoDB en comparación con Amazon RDS, sin servidor).
- Determinar tipos de bases de datos de AWS redituables (por ejemplo, formato de serie temporal, formato de columnas).
- Migrar esquemas y datos de bases de datos a diferentes ubicaciones y diferentes motores de bases de datos.

Enunciado de la tarea 4: diseñar arquitecturas de redes rentables.

Conocimientos:

- características del servicio de administración de costos de AWS (por ejemplo, etiquetas de asignación de costos, facturación en varias cuentas)
- herramientas de administración de costos de AWS con casos de uso apropiados (por ejemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- conceptos de equilibrio de carga (por ejemplo, Application Load Balancer)
- gateways NAT (por ejemplo, costos de instancias NAT en comparación con los costos de gateway NAT)
- conectividad de red (por ejemplo, líneas privadas, líneas dedicadas, VPN)
- enrutamiento, topología e interconexión de redes (por ejemplo, AWS Transit Gateway, interconexión de VPC)
- servicios de redes con casos de uso adecuados (por ejemplo, DNS)

Habilidades:

- Configurar los tipos de gateway NAT adecuados para una red (por ejemplo, una única gateway NAT compartida en comparación con las gateway NAT para cada zona de disponibilidad).
- Configurar las conexiones de red adecuadas (por ejemplo, Direct Connect en comparación con una VPN, en comparación con Internet).
- Configurar las rutas de red adecuadas para minimizar los costos de transferencia de red (por ejemplo, de región a región, de zona de disponibilidad a zona de disponibilidad, de privada a pública, Global Accelerator, puntos de enlace de la VPC).
- Determinar las necesidades estratégicas para las redes de entrega de contenido (CDN) y el almacenamiento en caché en borde.
- Revisar cargas de trabajo existentes para optimizaciones de red.
- Seleccionar una estrategia de reducción controlada adecuada.
- Seleccionar la asignación de banda ancha adecuada para un dispositivo de red (por ejemplo, una sola VPN en comparación con varias VPN, velocidad de Direct Connect).

Apéndice

¿Qué herramientas, tecnologías y conceptos clave podrían incluirse en el examen?

La siguiente es una lista parcial de las herramientas y tecnologías que podrían aparecer en el examen. Esta lista está sujeta a cambios y se proporciona para ayudarlo a comprender el alcance general de los servicios, las características o las tecnologías que se presentan en el examen. Las herramientas y las tecnologías generales de esta lista no aparecen en un orden particular. Los servicios de AWS se agrupan según sus funciones principales. Aunque es probable que el examen abarque algunas de estas tecnologías en mayor medida, el orden y su ubicación en esta lista no son indicios de su relevancia o importancia:

- Cómputo
- Administración de costos
- Base de datos
- Recuperación de desastres
- Alto rendimiento
- Administración y gobernanza
- Microservicios y desacoplamiento de componentes
- Migración y transferencia de datos
- Redes, conectividad y entrega de contenido
- Resistencia
- Seguridad
- Principios de diseño sin servidor e impulsados por eventos
- Almacenamiento

Servicios y características de AWS

Análisis:

- Amazon Athena
- AWS Data Exchange
- AWS Data Pipeline
- Amazon EMR
- AWS Glue
- Amazon Kinesis
- AWS Lake Formation
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Amazon OpenSearch Service (Amazon Elasticsearch Service)
- Amazon QuickSight
- Amazon Redshift

Integración de aplicaciones:

- Amazon AppFlow
- AWS AppSync
- Amazon EventBridge (Amazon CloudWatch Events)
- Amazon MQ
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Administración de costos de AWS:

- AWS Budgets
- AWS Cost and Usage Report
- AWS Cost Explorer
- Savings Plans

Cómputo:

- AWS Batch
- Amazon EC2
- Amazon EC2 Auto Scaling
- AWS Elastic Beanstalk
- AWS Outposts
- AWS Serverless Application Repository
- VMware Cloud on AWS
- AWS Wavelength

Contenedores:

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon ECS Anywhere
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Amazon EKS Anywhere
- Amazon EKS Distro

Base de datos:

- Amazon Aurora
- Amazon Aurora Serverless
- Amazon DocumentDB (with MongoDB compatibility)
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Keyspaces (for Apache Cassandra)
- Amazon Neptune
- Amazon Quantum Ledger Database (Amazon QLDB)
- Amazon RDS
- Amazon Redshift
- Amazon Timestream

Herramientas para desarrolladores:

- AWS X-Ray

Frontend web y móvil:

- AWS Amplify
- Amazon API Gateway
- AWS Device Farm
- Amazon Pinpoint

Machine learning:

- Amazon Comprehend
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Polly
- Amazon Rekognition
- Amazon SageMaker
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Administración y gobernanza:

- AWS Auto Scaling
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Command Line Interface (AWS CLI)
- AWS Compute Optimizer
- AWS Config
- AWS Control Tower
- AWS License Manager
- Amazon Managed Grafana
- Amazon Managed Service for Prometheus
- AWS Management Console
- AWS Organizations
- AWS Personal Health Dashboard
- AWS Proton
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor
- AWS Well-Architected Tool

Servicios multimedia:

- Amazon Elastic Transcoder
- Amazon Kinesis Video Streams

Migración y transferencia:

- AWS Application Discovery Service
- AWS Application Migration Service (CloudEndure Migration)
- AWS Database Migration Service (AWS DMS)
- AWS DataSync
- AWS Migration Hub
- AWS Server Migration Service (AWS SMS)
- AWS Snow Family
- AWS Transfer Family

Redes y entrega de contenido:

- Amazon CloudFront
- AWS Direct Connect
- Elastic Load Balancing (ELB)
- AWS Global Accelerator
- AWS PrivateLink
- Amazon Route 53
- AWS Transit Gateway
- Amazon VPC
- AWS VPN

Seguridad, identidad y conformidad:

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager
- Amazon GuardDuty
- AWS Identity and Access Management (IAM)
- Amazon Inspector
- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Network Firewall
- AWS Resource Access Manager (AWS RAM)
- AWS Secrets Manager
- AWS Security Hub
- AWS Shield
- AWS Single Sign-On
- AWS WAF

Sin servidor:

- AWS AppSync
- AWS Fargate
- AWS Lambda

Almacenamiento:

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx (para todos los tipos)
- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

Servicios y características de AWS fuera de alcance

A continuación, se muestra una lista parcial de los servicios y las características de AWS que no se incluyen en el examen. Estos servicios y características no representan todas las ofertas de AWS que no se incluyen en el contenido del examen.

Análisis:

- Amazon CloudSearch

Integración de aplicaciones:

- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)

RA y RV:

- Amazon Sumerian

Cadena de bloques:

- Amazon Managed Blockchain

Cómputo:

- Amazon Lightsail

Base de datos:

- Amazon RDS on VMware

Herramientas para desarrolladores:

- AWS Cloud9
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudShell
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeCommit
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CodeStar
- Amazon Corretto
- AWS Fault Injection Simulator (AWS FIS)
- AWS Tools y SDK

Frontend web y móvil:

- Amazon Location Service

Tecnología de juegos:

- Amazon GameLift
- Amazon Lumberyard

Internet de las cosas:

- Todos los servicios

Machine learning:

- Apache MXNet on AWS
- Amazon Augmented AI (Amazon A2I)
- AWS DeepComposer
- AWS Deep Learning AMIs (DLAMI)
- AWS Deep Learning Containers
- AWS DeepLens
- AWS DeepRacer
- Amazon DevOps Guru
- Amazon Elastic Inference
- Amazon HealthLake
- AWS Inferentia
- Amazon Lookout for Equipment
- Amazon Lookout for Metrics
- Amazon Lookout for Vision
- Amazon Monitron
- AWS Panorama
- Amazon Personalize
- PyTorch on AWS
- Amazon SageMaker Data Wrangler
- Amazon SageMaker Ground Truth
- TensorFlow on AWS

Administración y gobernanza:

- AWS Chatbot
- AWS Console Mobile Application
- AWS Distro for OpenTelemetry
- AWS OpsWorks

Servicios multimedia:

- AWS Elemental Appliances and Software
- AWS Elemental MediaConnect
- AWS Elemental MediaConvert
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Elemental MediaTailor
- Amazon Interactive Video Service (Amazon IVS)

Migración y transferencia:

- Migration Evaluator (anteriormente TSO Logic)

Redes y entrega de contenido:

- AWS App Mesh
- AWS Cloud Map

Tecnologías cuánticas:

- Amazon Braket

Robótica:

- AWS RoboMaker

Servicios satelitales:

- AWS Ground Station

AWS Certified Solutions Architect – Associate (SAA-C03) Guía de examen

Este documento incluye las versiones en [español \(Latinoamérica\)](#) y [español \(España\)](#).

Introducción

El examen AWS Certified Solutions Architect — Associate (SAA-C03) está dirigido a personas que desempeñan un rol de arquitecto de soluciones. El examen valida la capacidad de un candidato para utilizar las tecnologías de AWS para diseñar soluciones basadas en AWS Well-Architected Framework.

El examen también valida la capacidad del candidato para completar las siguientes tareas:

- Diseñar soluciones que incorporen servicios de AWS para cumplir con los requisitos empresariales actuales y las necesidades proyectadas
- Diseñar arquitecturas que sean seguras, resilientes, de alto rendimiento y con precios optimizados
- Revisar las soluciones existentes y determinar las mejoras

Descripción del candidato objetivo

El candidato objetivo debe tener al menos 1 año de experiencia práctica en el diseño de soluciones en la nube que utilicen los servicios de AWS.

Para obtener una lista detallada de herramientas y tecnologías específicas que podrían incluirse en el examen, así como listas de servicios de AWS dentro y fuera del alcance, consulta el Apéndice.

Contenido del examen

Tipos de respuesta

Hay dos tipos de preguntas en el examen:

- **Tipo test con opciones múltiples:** una respuesta correcta y tres respuestas incorrectas (distractores)
- **Tipo test de respuesta múltiple:** dos o más respuestas correctas de cinco o más opciones de respuesta

Selecciona una o más respuestas que mejor se adapten al enunciado o respondan a la pregunta. Los distractores, o las respuestas incorrectas, son opciones de respuesta que puede elegir un candidato con conocimientos o habilidades incompletos. Los distractores son respuestas generalmente verosímiles que coinciden con el área de contenido.

Las preguntas sin respuesta se califican como incorrectas; no hay penalización por intentar adivinar. El examen incluye 50 preguntas que afectarán a tu puntuación.

Contenido sin puntuar

El examen incluye 15 preguntas sin calificación que no afectan a tu puntuación. AWS recopila información sobre el rendimiento de los candidatos en estas preguntas sin calificación para evaluar estas preguntas para usarlas en el futuro como preguntas puntuadas. Estas preguntas sin calificación no se identifican en el examen.

Resultados del examen

El examen AWS Certified Solutions Architect — Associate es un examen con calificación de aprobado o suspendido. El examen se califica según un estándar mínimo establecido por profesionales de AWS que siguen las prácticas recomendadas y las directrices del sector de la certificación.

Los resultados del examen se notifican como una puntuación que va de 100 a 1000. La puntuación mínima para aprobar es 720. Tu puntuación indica tu comportamiento en el examen en conjunto y si has aprobado o no. Los modelos de puntuación escalados ayudan a igualar las puntuaciones en varios formularios de examen que puedan tener niveles de dificultad ligeramente diferentes.

El informe de puntuación puede contener una tabla de clasificaciones de tu rendimiento en cada nivel de sección. Esta información proporciona comentarios generales sobre tu rendimiento en el examen. El examen utiliza un modelo de calificación compensatoria, lo que significa que no necesitas obtener una calificación de aprobado en cada sección. Solo debes aprobar el examen general.

Cada sección del examen tiene una ponderación específica, por lo que algunas secciones tienen más preguntas que otras. La tabla contiene información general que destaca tus puntos fuertes y débiles. Interpreta con precaución los comentarios de las secciones. Los candidatos que aprueben el examen no recibirán esta información adicional.

Descripción del contenido

Esta guía del examen incluye ponderaciones, dominios de prueba y enunciados de tareas para el examen. No es una lista exhaustiva del contenido del examen. Sin embargo, hay un contexto adicional para cada uno de los enunciados de tareas que te ayudará como guía en la preparación del examen. En la siguiente tabla se enumeran los principales dominios de contenido y sus ponderaciones. La tabla precede a la descripción completa del contenido del examen, que incluye el contexto adicional. El porcentaje en cada dominio representa solo el contenido puntuado.

Dominio	% del examen
Dominio 1: Diseñar arquitecturas seguras	30 %
Dominio 2: Diseñar arquitecturas resilientes	26 %
Dominio 3: Diseñar arquitecturas de alto rendimiento	24 %
Dominio 4: Diseñar arquitecturas con precio optimizado	20 %
TOTAL	100 %

Dominio 1: Diseñar arquitecturas seguras

Enunciado de tarea 1: Diseñar un acceso seguro a los recursos de AWS.

Conocimientos de:

- Controles de acceso y administración en varias cuentas
- Servicios de identidad y acceso federados de AWS (por ejemplo, AWS Identity and Access Management [IAM], AWS Single Sign-On [AWS SSO])
- Infraestructura global de AWS (por ejemplo, zonas de disponibilidad, regiones de AWS)
- Prácticas recomendadas de seguridad de AWS (por ejemplo, el principio de mínimo privilegio)
- El modelo de responsabilidad compartida de AWS

Competencias en:

- Aplicación de prácticas recomendadas de seguridad de AWS a los usuarios de IAM y a los usuarios raíz (por ejemplo, autenticación multifactor [MFA])
- Diseño de un modelo de autorización flexible que incluya usuarios, grupos, roles y políticas de IAM
- Diseño de una estrategia de control de acceso basada en roles (por ejemplo, AWS Security Token Service [AWS STS], cambio de roles, acceso entre cuentas)
- Diseño una estrategia de seguridad para varias cuentas de AWS (por ejemplo, AWS Control Tower, políticas de control de servicios [SCP])
- Determinación del uso adecuado de las políticas de recursos para los servicios de AWS
- Determinación de cuándo federar un servicio de directorios con roles de IAM

Enunciado de tarea 2: Diseñar cargas de trabajo y aplicaciones seguras.

Conocimientos de:

- Configuración de aplicaciones y seguridad de credenciales
- Puntos de enlace de servicios de AWS
- Controlar puertos, protocolos y tráfico de red en AWS
- Acceso seguro a las aplicaciones
- Servicios de seguridad con casos prácticos apropiados (por ejemplo, Amazon Cognito, Amazon GuardDuty, Amazon Macie)
- Vectores de amenazas externos a AWS (por ejemplo, DDoS, inyección SQL)

Competencias en:

- Diseño de arquitecturas de VPC con componentes de seguridad (por ejemplo, grupos de seguridad, tablas de enrutamiento, ACL de red y gateways NAT)
- Determinación de estrategias de segmentación de red (por ejemplo, el uso de subredes públicas y subredes privadas)
- Integración de los servicios de AWS para proteger las aplicaciones (por ejemplo, AWS Shield, AWS WAF, AWS SSO, AWS Secrets Manager)
- Protección de las conexiones de red externas desde y hacia la nube de AWS (por ejemplo, VPN y AWS Direct Connect)

Enunciado de tarea 3: Determinar los controles de seguridad de datos apropiados

Conocimientos de:

- Acceso a los datos y dirección
- Recuperación de datos
- Retención y clasificación de datos
- Cifrado y administración de claves adecuada

Competencias en:

- Alineación de las tecnologías de AWS para cumplir los requisitos de conformidad
- Cifrado de datos en reposo (por ejemplo, AWS Key Management Service [AWS KMS])
- Cifrado de datos en tránsito (por ejemplo, AWS Certificate Manager [ACM] mediante TLS)
- Implementación de políticas de acceso para claves de cifrado
- Implementación de backups y replicaciones de datos
- Implementación de políticas de acceso, ciclo de vida y protección de datos
- Rotación de claves de cifrado y renovación de certificados

Dominio 2: Diseñar arquitecturas resilientes

Enunciado de tarea 1: Diseñar arquitecturas escalables y poco acopladas.

Conocimientos de:

- Creación y administración de API (por ejemplo, Amazon API Gateway, API REST)
- Servicios administrados de AWS con casos prácticos apropiados (por ejemplo, AWS Transfer Family, Amazon Simple Queue Service [Amazon SQS], Secrets Manager)
- Estrategias de almacenamiento en caché
- Diseñar principios para microservicios (por ejemplo, cargas de trabajo sin estado en comparación con cargas de trabajo con estado)
- Arquitecturas impulsadas por eventos
- Escalado horizontal y escalado vertical
- Cómo utilizar correctamente los aceleradores perimetrales (por ejemplo, red de entrega de contenido [CDN])
- Cómo migrar aplicaciones a contenedores
- Conceptos de balanceo de carga (por ejemplo, Application Load Balancer)
- Arquitecturas de varios niveles
- Conceptos de listas de espera y mensajería (por ejemplo, publicación/suscripción)
- Tecnologías y patrones sin servidor (por ejemplo, AWS Fargate, AWS Lambda)
- Tipos de almacenamiento con características asociadas (por ejemplo, objeto, archivo, bloque)
- El aprovisionamiento de contenedores (por ejemplo, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS])
- Cuándo usar réplicas de lectura
- Aprovisionamiento de flujo de trabajo (por ejemplo, AWS Step Functions)

Competencias en:

- Diseño de arquitecturas impulsadas por eventos, de microservicios o de varios niveles en función de los requisitos
- Determinación de estrategias de escalado para componentes utilizados en un diseño de arquitectura
- Determinación de servicios de AWS necesarios para lograr un acoplamiento débil en función de los requisitos
- Determinación de cuándo usar contenedores
- Determinación de cuándo usar tecnologías y patrones sin servidor
- Recomendación de tecnologías de cómputo, almacenamiento, redes y bases de datos adecuadas en función de los requisitos
- Uso de servicios de AWS diseñados específicamente para cargas de trabajo

Enunciado de tarea 2: Diseñar arquitecturas de alta disponibilidad o tolerantes a fallos.

Conocimientos de:

- Infraestructura global de AWS (por ejemplo, zonas de disponibilidad, regiones de AWS, Amazon Route 53)
- Servicios administrados de AWS con casos prácticos apropiados (por ejemplo, Amazon Comprehend y Amazon Polly)
- Conceptos básicos de redes (por ejemplo, tablas de enrutamiento)
- Estrategias de recuperación de desastres (RD) (por ejemplo, backup y restauración, luz piloto, espera activa, conmutación por error activa-activa, objetivo de punto de recuperación [RPO], objetivo de tiempo de recuperación [RTO])
- Patrones de diseño distribuidos
- Estrategias de conmutación por error
- Infraestructura inmutable
- Conceptos de balanceo de carga (por ejemplo, Application Load Balancer)
- Conceptos de proxy (por ejemplo, proxy de Amazon RDS)
- Cuotas de servicio y limitación (por ejemplo, cómo configurar las cuotas de servicio para una carga de trabajo en un entorno en espera)
- Opciones y características de almacenamiento (por ejemplo, durabilidad, replicación)
- Visibilidad de cargas de trabajo (por ejemplo, AWS X-Ray)

Competencias en:

- Determinación de las estrategias de automatización para garantizar la integridad
- Determinación de los servicios de AWS necesarios para proporcionar una arquitectura de alta disponibilidad o tolerante a fallos en todas las regiones o zonas de disponibilidad de AWS
- Identificación de métricas en función de los requisitos empresariales para ofrecer una solución de alta disponibilidad
- Implementación de diseños para mitigar puntos únicos de falla
- Implementación de estrategias para garantizar la durabilidad y disponibilidad de los datos (por ejemplo, copias de seguridad)
- Selección de una estrategia de DR adecuada para cumplir con los requisitos empresariales
- Uso de servicios de AWS que mejoren la fiabilidad de las aplicaciones heredadas y las aplicaciones no creadas para la nube (por ejemplo, cuando no es posible realizar cambios en las aplicaciones)
- Uso de servicios de AWS diseñados específicamente para cargas de trabajo

Dominio 3: Diseñar arquitecturas de alto rendimiento

Enunciado de tarea 1: Determinar soluciones de almacenamiento escalables o de alto rendimiento.

Conocimientos de:

- Soluciones de almacenamiento híbrido para cumplir con los requisitos empresariales
- Servicios de almacenamiento con casos prácticos adecuados (por ejemplo, Amazon S3, Amazon Elastic File System [Amazon EFS], Amazon Elastic Block Store [Amazon EBS])
- Tipos de almacenamiento con características asociadas (por ejemplo, objeto, archivo, bloque)

Competencias en:

- Determinación de los servicios y configuraciones de almacenamiento que cumplen con las demandas
- Determinación de los servicios de almacenamiento de información que pueden ampliarse para adaptarse

Enunciado de tarea 2: Diseñar soluciones de computación elásticas y de alto rendimiento.

Conocimientos de:

- Servicios de computación de AWS con casos prácticos apropiados (por ejemplo, AWS Batch, Amazon EMR, Fargate)
- Conceptos de computación distribuidos compatibles con la infraestructura global y los servicios perimetrales de AWS
- Conceptos de listas de espera y mensajería (por ejemplo, publicación/suscripción)
- Capacidades de escalabilidad con casos prácticos apropiados (por ejemplo, Amazon EC2 Auto Scaling, AWS Auto Scaling)
- Tecnologías y patrones sin servidor (por ejemplo, Lambda, Fargate)
- El aprovisionamiento de contenedores (por ejemplo, Amazon ECS, Amazon EKS)

Competencias en:

- Desacoplamiento de cargas de trabajo para que los componentes puedan escalarse de forma independiente
- Identificación de métricas y condiciones para realizar acciones de escalado
- Selección de las opciones y características de computación adecuadas (por ejemplo, tipos de instancias EC2) para cumplir con los requisitos empresariales
- Selección del tipo y el tamaño de recursos adecuados (por ejemplo, la cantidad de memoria Lambda) para cumplir con los requisitos empresariales

Enunciado de tarea 3: Determinar soluciones de bases de datos de alto

Conocimientos de:

- Infraestructura global de AWS (por ejemplo, zonas de disponibilidad, regiones de AWS)
- Estrategias y servicios de almacenamiento en caché (por ejemplo, Amazon ElastiCache)
- Patrones de acceso a datos (por ejemplo, lectura intensiva en comparación con escritura intensiva)
- Planificación de la capacidad de la base de datos (por ejemplo, unidades de capacidad, tipos de instancias, IOPS provisionadas)
- Conexiones y proxies de bases de datos

- Motores de bases de datos con casos prácticos apropiados (por ejemplo, migraciones heterogéneas, migraciones homogéneas)
- Replicación de bases de datos (por ejemplo, réplica de lectura)
- Tipos y servicios de bases de datos (por ejemplo, sin servidor, relacionales en comparación con no relacionales, en memoria)

Competencias en:

- Configuración de réplicas de lectura para cumplir con los requisitos empresariales
- Diseño de arquitecturas de bases de datos
- Determinación de un motor de base de datos apropiado (por ejemplo, MySQL en comparación con PostgreSQL)
- Determinación de un tipo de base de datos adecuado (por ejemplo, Amazon Aurora, Amazon DynamoDB)
- Integración del almacenamiento en caché para cumplir con los requisitos empresariales

Enunciado de tarea 4: Determinar arquitecturas de red escalables o de alto rendimiento.

Conocimientos de:

- Servicios de redes perimetrales con casos prácticos apropiados (por ejemplo, Amazon CloudFront, AWS Global Accelerator)
- Cómo diseñar una arquitectura de red (por ejemplo, niveles de subred, enrutamiento, direccionamiento IP)
- Conceptos de balanceo de carga (por ejemplo, Application Load Balancer)
- Opciones de conexión de red (por ejemplo, AWS VPN, Direct Connect y AWS PrivateLink)

Competencias en:

- Creación de una topología de red para varias arquitecturas (por ejemplo, global, híbrida o de varios niveles)
- Determinación de las configuraciones de red que se puedan escalar para adaptarse a las necesidades
- Determinación de la ubicación adecuada de los recursos para cumplir con los requisitos comerciales
- Selección de la estrategia de balanceo de carga adecuada

Enunciado de tarea 5: Determinar soluciones de transformación e ingesta de datos de alto rendimiento.

Conocimientos de:

- Servicios de análisis y visualización de datos con casos prácticos apropiados (por ejemplo, Amazon Athena, AWS Lake Formation y Amazon QuickSight)
- Patrones de ingesta de datos (por ejemplo, frecuencia)
- Servicios de transferencia de datos con casos prácticos apropiados (por ejemplo, AWS DataSync y AWS Storage Gateway)
- Servicios de transformación de datos con casos prácticos apropiados (por ejemplo, AWS Glue)
- Acceso seguro a los puntos de acceso de ingesta
- Tamaños y velocidades necesarios para cumplir con los requisitos empresariales
- Servicios de streaming con casos prácticos apropiados (por ejemplo, Amazon Kinesis)

Competencias en:

- Creación y protección de data lakes
- Diseño de arquitecturas de streaming
- Diseño de soluciones de transferencia de datos
- Implementación de estrategias de visualización
- Selección de las opciones de computación adecuadas para el procesamiento de datos (por ejemplo, Amazon EMR)
- Selección de configuraciones apropiadas para la ingesta
- Transformación de datos entre formatos (por ejemplo, de .csv a .parquet)

Dominio 4: Diseñar arquitecturas con precio optimizado

Enunciado de tarea 1: Diseñar soluciones de almacenamiento optimizadas para precios

Conocimientos de:

- Opciones de acceso (por ejemplo, un S3 Bucket con almacenamiento de objetos de Pago por solicitante)
- Características del servicio de administración de costes de AWS (por ejemplo, etiquetas de asignación de precios y facturación multicuenta)
- Herramientas de administración de costes de AWS con casos prácticos apropiados (por ejemplo, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- Servicios de almacenamiento de AWS con casos prácticos apropiados (por ejemplo, Amazon FSx, Amazon EFS, Amazon S3 y Amazon EBS)
- Estrategias de backup
- Opciones de almacenamiento en bloque (por ejemplo, tipos de volumen de unidad de disco duro [HDD], tipos de volumen de unidad de estado sólido [SSD])
- Ciclos de vida de los datos
- Opciones de almacenamiento híbrido (por ejemplo, DataSync, Transfer Family y Storage Gateway)
- Modelos de acceso al almacenamiento
- Organización en niveles de almacenamiento (por ejemplo, estratificación en frío para el almacenamiento de objetos)
- Tipos de almacenamiento con características asociadas (por ejemplo, objeto, archivo, bloque)

Competencias en:

- Diseño de estrategias de almacenamiento adecuadas (por ejemplo, cargas por lotes a Amazon S3 en comparación con cargas individuales)
- Determinación del tamaño de almacenamiento correcto para una carga de trabajo
- Determinación del método de menor precio para transferir datos para una carga de trabajo al almacenamiento de AWS
- Determinación de cuándo se requiere el escalado automático del almacenamiento
- Administración de los ciclos de vida de objetos S3
- Selección de la solución de backup o archivado adecuada
- Selección del servicio adecuado para la migración de datos a los servicios de almacenamiento
- Selección del nivel de almacenamiento adecuado
- Selección del ciclo de vida de datos correcto para el almacenamiento
- Selección del servicio de almacenamiento más rentable para una carga de trabajo

Enunciado de tarea 2: Diseñar soluciones de computación con precios optimizados

Conocimientos de:

- Características del servicio de administración de costes de AWS (por ejemplo, etiquetas de asignación de precios y facturación multicuenta)
- Herramientas de administración de costes de AWS con casos prácticos apropiados (por ejemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- Infraestructura global de AWS (por ejemplo, zonas de disponibilidad, regiones de AWS)
- Opciones de compra de AWS (por ejemplo, instancias tipo spot, instancias reservadas, planes de ahorro)
- Estrategias de computación distribuidas (por ejemplo, procesamiento perimetral)
- Opciones de computación híbridas (por ejemplo, AWS Outposts y AWS Snowball Edge)
- Tipos, familias y tamaños de instancias (por ejemplo, optimizados para memoria, optimización de computación y virtualización)
- Optimización de la utilización de computación (por ejemplo, contenedores, computación sin servidor y microservicios)
- Estrategias de escalado (por ejemplo, escalado automático e hibernación)

Competencias en:

- Determinación de una estrategia de balanceo de carga adecuada (por ejemplo, Application Load Balancer [capa 7] en comparación con Network Load Balancer [capa 4] en comparación con Gateway Load Balancer)
- Determinación de métodos y estrategias de escalado adecuados para cargas de trabajo elásticas (por ejemplo, horizontal en comparación con vertical e hibernación de EC2)
- Determinación de servicios computación de AWS rentables con casos prácticos adecuados (por ejemplo, Lambda, Amazon EC2, Fargate)
- Determinación de la disponibilidad requerida para diferentes clases de cargas de trabajo (por ejemplo, cargas de trabajo de producción, cargas de trabajo que no son de producción)
- Selección de la familia de instancias adecuada para una carga de trabajo
- Selección del tamaño de instancia adecuado para una carga de trabajo

Enunciado de tarea 3: Diseñar soluciones de bases de datos optimizadas

Conocimientos de:

- Características del servicio de administración de costes de AWS (por ejemplo, etiquetas de asignación de precios y facturación multicuenta)
- Herramientas de administración de costes de AWS con casos prácticos apropiados (por ejemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- Estrategias de almacenamiento en caché
- Políticas de retención de datos
- Planificación de capacidad de base de datos (por ejemplo, unidades de capacidad)
- Conexiones y proxies de bases de datos
- Motores de bases de datos con casos prácticos apropiados (por ejemplo, migraciones heterogéneas, migraciones homogéneas)
- Replicación de bases de datos (por ejemplo, réplica de lectura)
- Tipos y servicios de bases de datos (por ejemplo, relacionales en comparación con no relacionales, Aurora o DynamoDB)

Competencias en:

- Diseño de políticas de backup y retención adecuadas (por ejemplo, frecuencia de instantáneas)
- Determinación de un motor de base de datos apropiado (por ejemplo, MySQL en comparación con PostgreSQL)
- Determinación de servicios de base de datos de AWS rentables con casos prácticos adecuados (por ejemplo, DynamoDB en comparación con Amazon RDS, sin servidor)
- Determinación de tipos de bases de datos de AWS rentables (por ejemplo, formato de serie temporal, formato de columnas)
- Migración de esquemas y datos de bases de datos a diferentes ubicaciones o diferentes motores de bases de datos

Enunciado de tarea 4: Diseñar arquitecturas de red con precios optimizados

Conocimientos de:

- Características del servicio de administración de costes de AWS (por ejemplo, etiquetas de asignación de precios y facturación multicuenta)
- Herramientas de administración de costes de AWS con casos prácticos apropiados (por ejemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- Conceptos de balanceo de carga (por ejemplo, Application Load Balancer)
- Gateways NAT (por ejemplo, precios de instancias NAT en comparación con los precios de gateway NAT)
- Conectividad de red (por ejemplo, líneas privadas, líneas dedicadas, VPN)
- Enrutamiento, topología e interconexión de redes (por ejemplo, AWS Transit Gateway e interconexión de VPC)
- Servicios de red con casos prácticos apropiados (por ejemplo, DNS)

Competencias en:

- Configurar los tipos de gateway NAT adecuados para una red (por ejemplo, una gateway NAT compartida única en comparación con las gateways NAT para cada zona de disponibilidad)
- Configuración de las conexiones de red apropiadas (por ejemplo, Direct Connect en comparación con una VPN en comparación con Internet)
- Configuración de las rutas de red adecuadas para minimizar los precios de transferencia de red (por ejemplo, de región a región, de zona de disponibilidad a zona de disponibilidad, de privada a pública, Global Accelerator y puntos de enlace de la VPC)
- Determinación de las necesidades estratégicas para las redes de entrega de contenido (CDN) y el almacenamiento en caché perimetral
- Revisión de cargas de trabajo existentes para optimizaciones de red
- Selección de una estrategia de limitación adecuada
- Selección de la asignación de ancho de banda adecuada para un dispositivo de red (por ejemplo, una sola VPN en comparación con varias VPN, velocidad Direct Connect)

Apéndice

¿Qué herramientas, tecnologías y conceptos clave podrían incluirse en el examen?

A continuación se incluye una lista no exhaustiva de las herramientas y tecnologías que podrían aparecer en el examen. Esta lista está sujeta a cambios y se proporciona para ayudarle a comprender el alcance general de los servicios, las características o las tecnologías del examen. Las herramientas y tecnologías generales de esta lista no aparecen en ningún orden en particular. Los servicios de AWS se agrupan de acuerdo con sus funciones principales. Si bien es probable que algunas de estas tecnologías se cubran más que otras en el examen, el orden y la ubicación de las mismas en esta lista no indican el peso o la importancia relativos:

- Computación
- Gestión de precios
- Bases de datos
- Recuperación de desastres
- Alto rendimiento
- Administración y dirección
- Desacoplamiento de microservicios y componentes
- Migración y transferencia de datos
- Redes, conectividad y entrega de contenido
- Resiliencia
- Seguridad
- Principios de diseño sin servidor e impulsados por eventos
- Almacenamiento

Servicios y características de AWS

Análisis:

- Amazon Athena
- AWS Data Exchange
- AWS Data Pipeline
- Amazon EMR
- AWS Glue
- Amazon Kinesis
- AWS Lake Formation
- Amazon Managed Streaming para Apache Kafka (Amazon MSK)
- Amazon OpenSearch Service (Amazon Elasticsearch Service)
- Amazon QuickSight
- Amazon Redshift

Integración de aplicaciones:

- Amazon AppFlow
- AWS AppSync
- Amazon EventBridge (Amazon CloudWatch Events)
- Amazon MQ
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Administración de costes de AWS:

- AWS Budgets
- AWS Cost and Usage Report
- AWS Cost Explorer
- Planes de ahorro

Computación:

- AWS Batch
- Amazon EC2
- Amazon EC2 Auto Scaling
- AWS Elastic Beanstalk
- AWS Outposts
- AWS Serverless Application Repository
- VMware Cloud on AWS
- AWS Wavelength

Contenedores:

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon ECS Anywhere
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Amazon EKS Anywhere
- Amazon EKS Distro

Base de datos:

- Amazon Aurora
- Amazon Aurora sin servidor
- Amazon DocumentDB (compatible con MongoDB)
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Keyspaces (para Apache Cassandra)
- Amazon Neptune
- Amazon Quantum Ledger Database (Amazon QLDB)
- Amazon RDS
- Amazon Redshift
- Amazon Timestream

Herramientas para desarrolladores:

- AWS X-Ray

Interfaz web y móvil:

- AWS Amplify
- Amazon API Gateway
- AWS Device Farm
- Amazon Pinpoint

Machine Learning:

- Amazon Comprehend
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Polly
- Amazon Rekognition
- Amazon SageMaker
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Administración y dirección:

- AWS Auto Scaling
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Command Line Interface (AWS CLI)
- AWS Compute Optimizer
- AWS Config
- AWS Control Tower
- AWS License Manager
- Managed Service for Grafana
- Amazon Managed Service for Prometheus
- AWS Management Console
- AWS Organizations
- AWS Personal Health Dashboard
- AWS Proton
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor
- AWS Well-Architected Tool

Servicios multimedia:

- Amazon Elastic Transcoder
- Amazon Kinesis Video Streams

Migración y transferencia:

- AWS Application Discovery Service
- Servicio de migración de aplicaciones de AWS (CloudEndure Migration)
- AWS Database Migration Service (AWS DMS)
- AWS DataSync
- AWS Migration Hub
- AWS Server Migration Service (AWS SMS)
- AWS Snow Family
- AWS Transfer Family

Redes y entrega de contenido:

- Amazon CloudFront
- AWS Direct Connect
- Elastic Load Balancing (ELB)
- AWS Global Accelerator
- AWS PrivateLink
- Amazon Route 53
- AWS Transit Gateway
- Amazon VPC
- AWS VPN

Seguridad, identidad y conformidad:

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager
- Amazon GuardDuty
- AWS Identity and Access Management (IAM)
- Amazon Inspector
- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Network Firewall
- AWS Resource Access Manager (AWS RAM)
- AWS Secrets Manager
- AWS Security Hub
- AWS Shield
- AWS Single Sign-On
- AWS WAF

Sin servidor:

- AWS AppSync
- AWS Fargate
- AWS Lambda

Almacenamiento:

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx (para todos los tipos)
- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

Servicios y características de AWS fuera del ámbito

A continuación se incluye una lista no exhaustiva de los servicios y características de AWS que no se tratan en el examen. Estos servicios y características no representan todas las ofertas de AWS que se excluyen del contenido del examen.

Análisis:

- Amazon CloudSearch

Integración de aplicaciones:

- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)

RA y RV:

- Amazon Sumerian

Blockchain:

- Amazon Managed Blockchain

Computación:

- Amazon Lightsail

Base de datos:

- Amazon RDS en VMware

Herramientas para desarrolladores:

- AWS Cloud9
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudShell
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeCommit
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CodeStar
- Amazon Corretto
- AWS Fault Injection Simulator (AWS FIS)
- Herramientas y SDK de AWS

Interfaz web y móvil:

- Amazon Location Service

Tecnología de juegos:

- Amazon GameLift
- Amazon Lumberyard

Internet de las cosas:

- Todos los servicios

Machine Learning:

- Apache MXNet en AWS
- Amazon Augmented AI (Amazon A2I)
- AWS DeepComposer
- AMI de AWS Deep Learning (DLAMI)
- AWS Deep Learning Containers
- AWS DeepLens
- AWS DeepRacer
- Amazon DevOps Guru
- Amazon Elastic Inference
- Amazon HealthLake
- AWS Inferentia
- Amazon Lookout for Equipment
- Amazon Lookout for Metrics
- Amazon Lookout for Vision
- Amazon Monitron
- AWS Panorama
- Amazon Personalize
- PyTorch en AWS
- Amazon SageMaker Data Wrangler
- Amazon SageMaker Ground Truth
- TensorFlow en AWS

Administración y dirección:

- AWS Chatbot
- AWS Console Mobile Application
- AWS Distro for OpenTelemetry
- AWS OpsWorks

Servicios multimedia:

- Aplicaciones y software de AWS Elemental
- AWS Elemental MediaConnect
- AWS Elemental MediaConvert
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Elemental MediaTailor
- Amazon Interactive Video Service (Amazon IVS)

Migración y transferencia:

- Migration Evaluator (anteriormente TSO Logic)

Redes y entrega de contenido:

- AWS App Mesh
- AWS Cloud Map

Tecnologías cuánticas:

- Amazon Braket

Robótica:

- AWS RoboMaker

Satélite:

- AWS Ground Station