

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

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