

AWS Certified Solutions Architect - Associate

Guia do exame (SAA-C03)

Introdução

O exame AWS Certified Solutions Architect - Associate (SAA-C03) destina-se a pessoas que desempenham a função de arquiteto de soluções. O exame valida a capacidade do candidato de usar as tecnologias da AWS para projetar soluções com base no AWS Well-Architected Framework.

O exame também valida a capacidade de um candidato de concluir as seguintes tarefas:

- Projetar soluções que incorporam os serviços da AWS para atender aos requisitos empresariais atuais e às necessidades futuras projetadas.
- Projetar arquiteturas seguras, resilientes, de alto desempenho e econômicas.
- Analisar as soluções existentes e determinar como realizar melhorias.

Descrição do candidato

O candidato deve ter pelo menos um ano de experiência prática em design de soluções de nuvem que usam serviços da AWS.

Para obter uma lista detalhada de ferramentas e tecnologias específicas que podem ser abordadas no exame, bem como listas de serviços da AWS que estão dentro e fora do escopo, consulte o Apêndice.

Conteúdo do exame

Tipos de resposta

Existem dois tipos de perguntas no exame:

- **Múltipla escolha:** tem uma resposta correta e três respostas incorretas (distratores)
- **Resposta múltipla:** tem duas ou mais respostas corretas dentre cinco ou mais opções de resposta

Selecione uma ou mais respostas que completem melhor a afirmação ou respondam à pergunta. Distratores, ou respostas incorretas, são opções de resposta que um candidato com habilidades ou conhecimentos incompletos pode escolher. Geralmente, os distratores são respostas plausíveis que correspondem à área de conteúdo.

As perguntas não respondidas são pontuadas como incorretas; não há penalidade por tentar adivinhar. O exame inclui 50 perguntas que afetarão sua pontuação.

Conteúdo não pontuado

O exame inclui 15 perguntas não pontuadas que não afetam sua pontuação. A AWS coleta informações sobre a performance do candidato nas perguntas não pontuadas a fim de avaliá-las para uso futuro como perguntas pontuadas. As perguntas não pontuadas não são identificadas no exame.

Resultados do exame

O exame AWS Certified Solutions Architect – Associate é um exame de aprovação ou reprovação. O exame é pontuado de acordo com um padrão mínimo estabelecido por profissionais da AWS que seguem as práticas recomendadas e as diretrizes do setor de certificação.

Os resultados do exame são fornecidos como uma pontuação em escala de 100 a 1.000. A pontuação mínima de aprovação é de 720. A pontuação mostra como foi sua performance no exame como um todo e se você obteve aprovação ou não. Os modelos de pontuação em escala ajudam a correlacionar as pontuações em várias formas de exame que podem ter níveis de dificuldade ligeiramente diferentes.

O relatório de pontuação pode conter uma tabela de classificação de sua performance em cada nível de seção. Essas informações fornecem feedback geral sobre sua performance no exame. O exame usa um modelo de pontuação compensatória, o que significa que não é necessário obter uma pontuação de aprovação em cada seção. Você precisa passar apenas no exame geral.

Cada seção do exame tem um peso específico, portanto algumas seções têm mais perguntas do que outras. A tabela contém informações gerais que destacam seus pontos fortes e fracos. Tenha cuidado ao interpretar o feedback no nível da seção. Os candidatos que passarem no exame não receberão essas informações adicionais.

Descrição do conteúdo

Este guia do exame inclui os pesos, os domínios do teste e as declarações de tarefas do exame. Não é uma lista abrangente do conteúdo do exame. No entanto, disponibilizamos um contexto adicional para cada uma das declarações de tarefas visando ajudar a orientar sua preparação para o exame. A tabela a seguir lista os principais domínios de conteúdo e seus pesos. A tabela precede a descrição completa do conteúdo do exame, que inclui o contexto adicional. A porcentagem de cada domínio representa apenas o conteúdo pontuado.

Domínio	% do exame
Domínio 1: Design de arquiteturas seguras	30%
Domínio 2: Design de arquiteturas resilientes	26%
Domínio 3: Design de arquiteturas de alta performance	24%
Domínio 4: Design de arquiteturas econômicas	20%
TOTAL	100%

Domínio 1: Design de arquiteturas seguras

Declaração de tarefa 1: Projetar acesso seguro aos recursos da AWS.

Conhecimento sobre:

- Controles de acesso e gerenciamento em várias contas.
- Serviços de identidade e acesso federado da AWS (por exemplo, AWS Identity and Access Management [IAM], AWS Single Sign-On [AWS SSO]).
- Infraestrutura global da AWS (por exemplo, Zonas de Disponibilidade, Regiões AWS).
- Práticas recomendadas de segurança da AWS (por exemplo, o princípio de menor privilégio).
- O modelo de responsabilidade compartilhada da AWS.

Habilidades em:

- Aplicar as práticas recomendadas de segurança da AWS a usuários do IAM e usuários-raiz (por exemplo, autenticação com multifator [MFA]).
- Projetar um modelo de autorização flexível que inclua usuários, grupos, funções e políticas do IAM.
- Projetar uma estratégia de controle de acesso baseada em função (por exemplo, AWS Security Token Service [AWS STS], mudança de função, acesso entre contas).
- Projetar uma estratégia de segurança para várias contas da AWS (por exemplo, AWS Control Tower, políticas de controle de serviço [SCPs]).
- Determinar o uso apropriado de políticas de recursos para os serviços da AWS.
- Determinar quando federar um serviço de diretório com funções do IAM.

Declaração de tarefa 2: Projetar cargas de trabalho e aplicativos seguros.

Conhecimento sobre:

- Configuração de aplicativos e segurança de credenciais.
- Endpoints de serviço da AWS.
- Controle de portas, protocolos e tráfego de rede na AWS.
- Acesso seguro a aplicativos.
- Serviços de segurança com casos de uso apropriados (por exemplo, Amazon Cognito, Amazon GuardDuty, Amazon Macie).
- Vetores de ameaças externos à AWS (por exemplo, DDoS, SQL injection).

Habilidades em:

- Projetar arquiteturas de VPC com componentes de segurança (por exemplo, security groups, tabelas de rotas, ACLs de rede, gateways NAT).
- Determinar estratégias de segmentação de rede (por exemplo, usando sub-redes públicas e privadas).
- Integrar serviços da AWS para proteger aplicativos (por exemplo, AWS Shield, AWS WAF, AWS SSO, AWS Secrets Manager).
- Proteger conexões de rede externas de e para a nuvem AWS (por exemplo, VPN, AWS Direct Connect).

Declaração de tarefa 3: Determinar os controles de segurança de dados apropriados.

Conhecimento sobre:

- Acesso e governança de dados.
- Recuperação de dados.
- Classificação e retenção de dados.
- Criptografia e gerenciamento de chaves apropriado.

Habilidades em:

- Alinhar as tecnologias da AWS para atender aos requisitos de conformidade.
- Criptografia de dados em repouso (por exemplo, AWS Key Management Service [AWS KMS]).
- Criptografia de dados em trânsito (por exemplo, AWS Certificate Manager [ACM] usando TLS).
- Implementar políticas de acesso para chaves de criptografia.
- Implementar backups e replicações de dados.
- Implementar políticas para acesso, ciclo de vida e proteção de dados.
- Alternar chaves de criptografia e renovar certificados.

Domínio 2: Design de arquiteturas resilientes

Declaração de tarefa 1: Projetar arquiteturas dimensionáveis e com acoplamento fraco.

Conhecimento sobre:

- Criação e gerenciamento de APIs (por exemplo, Amazon API Gateway, API REST).
- AWS Managed Services com casos de uso apropriados (por exemplo, AWS Transfer Family, Amazon Simple Queue Service [Amazon SQS], Secrets Manager).
- Estratégias de armazenamento em cache.
- Princípios de design para microsserviços (por exemplo, cargas de trabalho stateless em comparação com cargas de trabalho stateful).
- Arquiteturas orientadas por eventos.
- Scaling horizontal e vertical.
- Como usar adequadamente os aceleradores de borda (por exemplo, rede de entrega de conteúdo [CDN]).
- Como migrar aplicativos para contêineres.
- Conceitos de balanceamento de carga (por exemplo, Application Load Balancer).
- Arquiteturas multicamadas.
- Conceitos de enfileiramento e sistema de mensagens (por exemplo, publicar/assinar).
- Tecnologias e padrões sem servidor (por exemplo, AWS Fargate, AWS Lambda).
- Tipos de armazenamento com características associadas (por exemplo, objeto, arquivo, bloco).
- A orquestração de contêineres (por exemplo, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS]).
- Quando usar réplicas de leitura.
- Orquestração de fluxo de trabalho (por exemplo, AWS Step Functions).

Habilidades em:

- Projetar arquiteturas orientadas por eventos, microsserviços e/ou multicamadas com base em requisitos.
- Determinar estratégias de scaling para componentes usados em um projeto de arquitetura.
- Determinar os serviços da AWS necessários para obter um atendimento fraco com base em requisitos.
- Determinar quando usar contêineres.
- Determinar quando usar tecnologias e padrões sem servidor.
- Recomendar tecnologias apropriadas de computação, armazenamento, redes e banco de dados com base em requisitos.
- Usar serviços da AWS com propósito específico para cargas de trabalho.

Declaração de tarefa 2: Projetar arquiteturas altamente disponíveis e/ou tolerantes a falhas.

Conhecimento sobre:

- Infraestrutura global da AWS (por exemplo, Zonas de Disponibilidade, Regiões AWS, Amazon Route 53).
- AWS Managed Services com casos de uso apropriados (por exemplo, Amazon Comprehend, Amazon Polly).
- Conceitos básicos de redes (por exemplo, tabelas de rotas).
- Estratégias de recuperação de desastres (DR) (por exemplo, backup e restauração, luz piloto, warm standby, failover ativo-ativo, objetivo de ponto de recuperação [RPO], objetivo de tempo de recuperação [RTO]).
- Padrões de design distribuídos.
- Estratégias de failover.
- Infraestrutura imutável.
- Conceitos de balanceamento de carga (por exemplo, Application Load Balancer).
- Conceitos de proxy (por exemplo, Amazon RDS Proxy).
- Cotas de serviço e limitação de largura de banda (por exemplo, como configurar as cotas de serviço para uma carga de trabalho em um ambiente de standby).
- Opções e características de armazenamento (por exemplo, durabilidade, replicação).
- Visibilidade da carga de trabalho (por exemplo, AWS X-Ray).

Habilidades em:

- Determinar estratégias de automação para garantir a integridade da infraestrutura.
- Determinar os serviços da AWS necessários para fornecer uma arquitetura altamente disponível e/ou tolerante a falhas nas Zonas de Disponibilidade ou Regiões AWS.
- Identificar métricas com base nos requisitos empresariais para oferecer uma solução altamente disponível.
- Implementar designs para mitigar pontos únicos de falha.
- Implementar estratégias para garantir a durabilidade e a disponibilidade dos dados (por exemplo, backups).
- Selecionar uma estratégia de DR apropriada para atender aos requisitos empresariais.
- Usar serviços da AWS que melhoram a confiabilidade de aplicativos legados e aplicativos que não foram criados para a nuvem (por exemplo, quando não é possível fazer alterações nos aplicativos).
- Usar serviços da AWS com propósito específico para cargas de trabalho.

Domínio 3: Design de arquiteturas de alta performance

Declaração de tarefa 1: Determinar soluções de armazenamento dimensionáveis e/ou de alto desempenho.

Conhecimento sobre:

- Soluções de armazenamento híbrido para atender aos requisitos empresariais
- Serviços de armazenamento com casos de uso apropriados (por exemplo, Amazon S3, Amazon Elastic File System [Amazon EFS], Amazon Elastic Block Store [Amazon EBS])
- Tipos de armazenamento com características associadas (por exemplo, objeto, arquivo, bloco).

Habilidades em:

- Determinar quais serviços e configurações de armazenamento atendem às demandas de desempenho.
- Determinar quais serviços de armazenamento que podem ser dimensionados para atender às necessidades futuras.

Declaração de tarefa 2: Projetar soluções de computação elásticas e de alto desempenho.

Conhecimento sobre:

- Serviços de computação da AWS com casos de uso apropriados (por exemplo, AWS Batch, Amazon EMR, Fargate).
- Conceitos de computação distribuída com base na infraestrutura global e nos serviços de borda da AWS.
- Conceitos de enfileiramento e sistema de mensagens (por exemplo, publicar/assinar).
- Recursos de escalabilidade com casos de uso apropriados (por exemplo, Amazon EC2 Auto Scaling, AWS Auto Scaling).
- Tecnologias e padrões sem servidor (por exemplo, Lambda, Fargate).
- A orquestração de contêineres (por exemplo, Amazon ECS, Amazon EKS).

Habilidades em:

- Desacoplar cargas de trabalho para que os componentes possam ser dimensionados de forma independente.
- Identificar métricas e condições para realizar ações de scaling.
- Selecionar as opções e os recursos de computação apropriados (por exemplo, tipos de instância do EC2) para atender aos requisitos empresariais.
- Selecionar o tipo e o tamanho de recurso apropriados (por exemplo, a quantidade de memória do Lambda) para atender aos requisitos empresariais.

Declaração de tarefa 3: Determinar soluções de banco de dados de alto desempenho.

Conhecimento sobre:

- Infraestrutura global da AWS (por exemplo, Zonas de Disponibilidade, Regiões AWS).
- Estratégias e serviços de armazenamento em cache (por exemplo, Amazon ElastiCache).
- Padrões de acesso a dados (por exemplo, leitura intensiva em comparação com gravação intensiva).
- Planejamento da capacidade de bancos de dados (por exemplo, unidades de capacidade, tipos de instância, IOPS provisionado).
- Proxies e conexões de banco de dados.

- Mecanismos de banco de dados com casos de uso apropriados (por exemplo, migrações homogêneas e heterogêneas).
- Replicação de banco de dados (por exemplo, réplicas de leitura).
- Tipos e serviços de banco de dados (por exemplo, sem servidor, relacional em comparação com não relacional, na memória).

Habilidades em:

- Configurar réplicas de leitura para atender aos requisitos empresariais.
- Projetar arquiteturas de banco de dados.
- Determinar um mecanismo de banco de dados apropriado (por exemplo, MySQL em comparação com o PostgreSQL).
- Determinar um tipo de banco de dados apropriado (por exemplo, Amazon Aurora, Amazon DynamoDB).
- Integrar o armazenamento em cache para atender aos requisitos empresariais.

Declaração de tarefa 4: Determinar arquiteturas de rede dimensionáveis e/ou de alto desempenho.

Conhecimento sobre:

- Serviços de redes de borda com casos de uso apropriados (por exemplo, Amazon CloudFront, AWS Global Accelerator).
- Como projetar arquitetura de rede (por exemplo, camadas de sub-rede, roteamento, endereçamento IP).
- Conceitos de balanceamento de carga (por exemplo, Application Load Balancer).
- Opções de conexão de rede (por exemplo, AWS VPN, Direct Connect, AWS PrivateLink).

Habilidades em:

- Criar uma topologia de rede para várias arquiteturas (por exemplo, global, híbrida, multicamadas).
- Determinar quais configurações de rede podem ser dimensionadas para acomodar necessidades futuras.
- Determinar o posicionamento adequado dos recursos para atender aos requisitos empresariais.
- Selecionar a estratégia de balanceamento de carga apropriada.

Declaração de tarefa 5: Determinar soluções de transformação e ingestão de dados de alto desempenho.

Conhecimento sobre:

- Serviços de análise e visualização de dados com casos de uso apropriados (por exemplo, Amazon Athena, AWS Lake Formation, Amazon QuickSight).
- Padrões de ingestão de dados (por exemplo, frequência).
- Serviços de transferência de dados com casos de uso apropriados (por exemplo, AWS DataSync, AWS Storage Gateway).
- Serviços de transformação de dados com casos de uso apropriados (por exemplo, AWS Glue).
- Acesso seguro a pontos de acesso de ingestão.
- Tamanhos e velocidades necessários para atender aos requisitos empresariais.
- Serviços de streaming de dados com casos de uso apropriados (por exemplo, Amazon Kinesis).

Habilidades em:

- Criar e proteger data lakes.
- Projetar arquiteturas de streaming de dados.
- Projetar soluções de transferência de dados.
- Implementar estratégias de visualização.
- Selecionar opções de computação apropriadas para processamento de dados (por exemplo, Amazon EMR).
- Selecionar configurações apropriadas para ingestão.
- Transformar dados entre formatos (por exemplo, .csv para .parquet).

Domínio 4: Design de arquiteturas econômicas

Declaração de tarefa 1: Projetar soluções de armazenamento econômicas.

Conhecimento sobre:

- Opções de acesso (por exemplo, um bucket do S3 com armazenamento de objetos de Requester Pays).
- Recursos do serviço de gerenciamento de custos da AWS (por exemplo, tags de alocação de custos, cobrança de várias contas).
- Ferramentas de gerenciamento de custos da AWS com casos de uso apropriados (por exemplo, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report).
- Serviços de armazenamento da AWS com casos de uso apropriados (por exemplo, Amazon FSx, Amazon EFS, Amazon S3, Amazon EBS).
- Estratégias de backup.
- Opções de armazenamento em bloco (por exemplo, tipos de volume da unidade de disco rígido [HDD], tipos de volume da unidade de estado sólido [SSD]).
- Ciclos de vida dos dados.
- Opções de armazenamento híbrido (por exemplo, DataSync, Transfer Family, Storage Gateway).
- Padrões de acesso ao armazenamento.
- Armazenamento em camadas (por exemplo, camadas frias para armazenamento de objetos).
- Tipos de armazenamento com características associadas (por exemplo, objeto, arquivo, bloco).

Habilidades em:

- Projetar estratégias de armazenamento apropriadas (por exemplo, fazer upload em lote para o Amazon S3 em comparação com upload individual).
- Determinar o tamanho de armazenamento correto para uma carga de trabalho.
- Determinar o método de menor custo de transferência de dados de uma carga de trabalho para o armazenamento da AWS.
- Determinar quando o auto scaling de armazenamento é necessário.
- Gerenciar ciclos de vida de objetos do S3.
- Selecionar a solução apropriada de backup e/ou arquivamento.
- Selecionar o serviço apropriado para a migração de dados aos serviços de armazenamento.
- Selecionar o nível de armazenamento apropriado.
- Selecionar o ciclo de vida de dados correto para armazenamento.
- Selecionar o serviço de armazenamento mais econômico para uma carga de trabalho.

Declaração de tarefa 2: Projetar soluções de computação econômicas.

Conhecimento sobre:

- Recursos do serviço de gerenciamento de custos da AWS (por exemplo, tags de alocação de custos, cobrança de várias contas)
- Ferramentas de gerenciamento de custos da AWS com casos de uso apropriados (por exemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report).
- Infraestrutura global da AWS (por exemplo, Zonas de Disponibilidade, Regiões AWS).
- Opções de compra da AWS (por exemplo, instâncias spot, instâncias reservadas, Savings Plans).
- Estratégias de computação distribuída (por exemplo, processamento de borda).
- Opções de computação híbrida (por exemplo, AWS Outposts, AWS Snowball Edge).
- Famílias, tamanhos e tipos de instâncias (por exemplo, otimizadas para memória, otimizadas para computação, virtualização).
- Otimização da utilização da computação (por exemplo, contêineres, computação sem servidor, microsserviços).
- Estratégias de scaling (por exemplo, auto scaling, hibernação).

Habilidades em:

- Determinar uma estratégia de balanceamento de carga apropriada (por exemplo, Application Load Balancer [camada 7] em comparação com o Network Load Balancer [camada 4] em comparação com o Gateway Load Balancer).
- Determinar métodos e estratégias de scaling apropriados para cargas de trabalho elásticas (por exemplo, horizontal em comparação com vertical, hibernação do EC2).
- Determinar serviços de computação da AWS econômicos com casos de uso apropriados (por exemplo, Lambda, Amazon EC2, Fargate)
- Determinar a disponibilidade necessária para diferentes classes de cargas de trabalho (por exemplo, cargas de trabalho de produção e de não produção)
- Escolher a família de instâncias apropriada para uma carga de trabalho.
- Escolher o tamanho de instância apropriado para uma carga de trabalho.

Declaração de tarefa 3: Projetar soluções de banco de dados econômicas.

Conhecimento sobre:

- Recursos do serviço de gerenciamento de custos da AWS (por exemplo, tags de alocação de custos, cobrança de várias contas)
- Ferramentas de gerenciamento de custos da AWS com casos de uso apropriados (por exemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report).
- Estratégias de armazenamento em cache.
- Políticas de retenção de dados.
- Planejamento da capacidade do banco de dados (por exemplo, unidades de capacidade).
- Proxies e conexões de banco de dados.
- Mecanismos de banco de dados com casos de uso apropriados (por exemplo, migrações homogêneas e heterogêneas).
- Replicação de banco de dados (por exemplo, réplicas de leitura).
- Tipos e serviços de banco de dados (por exemplo, relacional em comparação com não relacional, Aurora, DynamoDB).

Habilidades em:

- Projetar políticas de backup e retenção apropriadas (por exemplo, frequência de snapshots).
- Determinar um mecanismo de banco de dados apropriado (por exemplo, MySQL em comparação com o PostgreSQL).
- Determinar serviços de banco de dados da AWS econômicos com casos de uso apropriados (por exemplo, DynamoDB em comparação com o Amazon RDS, sem servidor).
- Determinar tipos de banco de dados da AWS econômicos (por exemplo, formato de série temporal, formato colunar).
- Migrar esquemas e dados de banco de dados para diferentes locais e/ou diferentes mecanismos de banco de dados.

Declaração de tarefa 4: Projetar arquiteturas de rede econômicas.

Conhecimento sobre:

- Recursos do serviço de gerenciamento de custos da AWS (por exemplo, tags de alocação de custos, cobrança de várias contas)
- Ferramentas de gerenciamento de custos da AWS com casos de uso apropriados (por exemplo, Cost Explorer, AWS Budgets, AWS Cost and Usage Report).
- Conceitos de balanceamento de carga (por exemplo, Application Load Balancer).
- Gateways NAT (por exemplo, custos de instância NAT em comparação com os custos do gateway NAT).
- Conectividade de rede (por exemplo, linhas privadas, linhas dedicadas, VPNs).
- Roteamento de rede, topologia e peering (por exemplo, transit gateway da AWS, peering de VPC).
- Serviços de rede com casos de uso apropriados (por exemplo, DNS).

Habilidades em:

- Configurar tipos de gateway NAT apropriados para uma rede (por exemplo, um único gateway NAT compartilhado em comparação com gateways NAT para cada Zona de Disponibilidade).
- Configurar conexões de rede apropriadas (por exemplo, Direct Connect em comparação com a VPN e com a Internet).
- Configurar rotas de rede apropriadas para minimizar os custos de transferência de rede (por exemplo, região para região, Zona de Disponibilidade para Zona de Disponibilidade, privado para público, Global Accelerator, endpoints de VPC).
- Determinar necessidades estratégicas para redes de entrega de conteúdo (CDNs) e cache de borda.
- Analisar as cargas de trabalho para otimizações de rede.
- Selecionar uma estratégia de limitação de largura de banda apropriada.
- Selecionar a alocação de largura de banda apropriada para um dispositivo de rede (por exemplo, uma única VPN em comparação com várias VPNs, velocidade do Direct Connect).

Apêndice

Quais são as principais ferramentas, tecnologias e conceitos que podem ser abordados no exame?

Veja a seguir uma lista (não completa) de ferramentas e tecnologias que podem aparecer no exame. Essa lista está sujeita a alterações e é fornecida para ajudar a entender o escopo geral de serviços, recursos ou tecnologias no exame. As ferramentas e tecnologias gerais dessa lista não aparecem em nenhuma ordem específica. Os produtos da AWS são agrupados de acordo com suas funções principais. Embora algumas dessas tecnologias provavelmente sejam abordadas mais do que outras no exame, a ordem e colocação delas na lista não são indicativos de importância nem de peso relativo:

- Computação
- Gerenciamento de custos
- Banco de dados
- Recuperação de desastres
- Alto desempenho
- Gerenciamento e governança
- Microsserviços e desacoplamento de componentes
- Migração e transferência de dados
- Redes, conectividade e entrega de conteúdo
- Resiliência
- Segurança
- Princípios de design sem servidor e orientados por eventos
- Armazenamento

Recursos e produtos da AWS

Análise:

- 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

Integração de aplicativos:

- 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

Gerenciamento de custos da AWS:

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

Computação:

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

Contêineres:

- 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

Banco de dados:

- 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

Ferramentas do desenvolvedor:

- AWS X-Ray

Web e dispositivos móveis de front-end:

- 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

Gerenciamento e governança:

- 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

Serviços de mídia:

- Amazon Elastic Transcoder
- Amazon Kinesis Video Streams

Migração e transferência:

- 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 e entrega de conteúdo:

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

Segurança, identidade e compatibilidade:

- 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

Sem servidor:

- AWS AppSync
- AWS Fargate
- AWS Lambda

Armazenamento:

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

Recursos e produtos da AWS fora do escopo

Veja a seguir uma lista (não completa) de recursos e produtos da AWS que não são abordados no exame. Estes recursos e produtos não representam todos os serviços da AWS que foram excluídos do conteúdo do exame.

Análise:

- Amazon CloudSearch

Integração de aplicativos:

- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)

AR e VR:

- Amazon Sumerian

Blockchain:

- Amazon Managed Blockchain

Computação:

- Amazon Lightsail

Banco de dados:

- Amazon RDS on VMware

Ferramentas do desenvolvedor:

- 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)
- Ferramentas e SDKs da AWS

Web e dispositivos móveis de front-end:

- Amazon Location Service

Tecnologia de jogos:

- Amazon GameLift
- Amazon Lumberyard

Internet das Coisas:

- Todos os serviços

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

Gerenciamento e governança:

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

Serviços de mídia:

- Dispositivos e software do 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)

Migração e transferência:

- Migration Evaluator (o antigo TSO Logic)

Redes e entrega de conteúdo:

- AWS App Mesh
- AWS Cloud Map

Tecnologias quânticas:

- Amazon Braket

Robótica:

- AWS RoboMaker

Satélite:

- AWS Ground Station