Exam Guides

AWS Certification

AWS Certification: Exam Guides

Table of Contents

AWS Certification Exam Guides

Overview of AWS Certification

AWS Certification(s) help individuals validate their knowledge, skills, and expertise, and help organizations identify and hire the right talent. Today, we offer certifications across three levels of proficiency: Foundational (validating fundamental knowledge of AWS Cloud), Associate and Professional (validating technical skills and expertise), as well as Specialty (showcasing expertise in specific areas of focus) certifications. Learn more.

Exam Guides - Foundational

AWS Foundational level certification exams are designed for individuals who are new to AWS and cloud computing.

Topics

AWS Certified Cloud Practitioner (CLF-C02)

The AWS Certified Cloud Practitioner (CLF-C02) exam is intended for individuals who can effectively demonstrate overall knowledge of the AWS Cloud, independent of a specific job role.

Topics

Introduction

The AWS Certified Cloud Practitioner (CLF-C02) exam is intended for individuals who can effectively demonstrate overall knowledge of the AWS Cloud, independent of a specific job role.

The exam also validates a candidate's ability to complete the following tasks:

- Explain the value of the AWS Cloud.
- Understand and explain the AWS shared responsibility model.
- Understand the AWS Well-Architected Framework.
- Understand security best practices.
- Understand AWS Cloud costs, economics, and billing practices.
- Describe and position the core AWS services, including compute, network, database, and storage services.
- Identify AWS services for common use cases.

Target Candidate Description

The target candidate has up to 6 months of exposure to AWS Cloud design, implementation, and/or operations. The candidate might be in the early stages of pursuing an AWS Cloud career or the candidate might work with people in AWS Cloud roles.

Recommended AWS knowledge

The target candidate should have AWS knowledge in the following areas:

- AWS Cloud concepts
- Security and compliance in the AWS Cloud
- Core AWS services
- Economics of the AWS Cloud

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Coding
- Designing cloud architecture
- Troubleshooting
- Implementation

- Load and performance testing

Exam Content

There are two types of questions on the exam:

- Multiple choice: Has one correct response and three incorrect responses (distractors)
- Multiple response: Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

The AWS Certified Cloud Practitioner (CLF-C02) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 700. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Unscored Content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam Results

This AWS certification exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 700. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content Outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Cloud Concepts (24% of scored content)
- Content Domain 2: Security and Compliance (30% of scored content)
- Content Domain 3: Cloud Technology and Services (34% of scored content)
- Content Domain 4: Billing, Pricing, and Support (12% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- In-Scope AWS Services

- Out-of-Scope AWS Services
- Technologies and Concepts

Content Domain 1: Cloud Concepts

Domain 1 covers Cloud Concepts and represents 24% of the scored content on the exam.

Tasks

- Task Statement 1.1: Define the benefits of the AWS Cloud.
- Task Statement 1.2: Identify design principles of the AWS Cloud.
- Task Statement 1.3: Understand the benefits of and strategies for migration to the AWS Cloud.
- Task Statement 1.4: Understand concepts of cloud economics.

Task Statement 1.1: Define the benefits of the AWS Cloud.

Knowledge of:

- Value proposition of the AWS Cloud

Skills in:

- Understanding the benefits of global infrastructure (for example, speed of deployment, global reach)
- Understanding the advantages of high availability, elasticity, and agility

Task Statement 1.2: Identify design principles of the AWS Cloud.

Knowledge of:

- AWS Well-Architected Framework

Skills in:

- Understanding the pillars of the Well-Architected Framework (for example, operational excellence, security, reliability, performance efficiency, cost optimization, sustainability)

- Identifying differences between the pillars of the Well-Architected Framework

Task Statement 1.3: Understand the benefits of and strategies for migration to the AWS Cloud.

Knowledge of:

- Cloud adoption strategies
- Resources to support the cloud migration journey

Skills in:

- Understanding the components of the AWS Cloud Adoption Framework (AWS CAF) (for example, reduced business risk; improved environmental, social, and governance [ESG] performance; increased revenue; increased operational efficiency)
- Identifying appropriate migration strategies (for example, database replication, use of AWS Snowball)

Task Statement 1.4: Understand concepts of cloud economics.

Knowledge of:

- Aspects of cloud economics
- Cost savings of moving to the cloud

Skills in:

- Understanding the role of fixed costs compared with variable costs
- Understanding costs that are associated with on-premises environments
- Understanding the differences between licensing strategies (for example, Bring Your Own License [BYOL] model compared with included licenses)
- Understanding the concept of rightsizing
- Identifying benefits of automation
- Understanding the economies of scale (for example, cost savings)

Content Domain 2: Security and Compliance

Domain 2 covers Security and Compliance and represents 30% of the scored content on the exam.

Tasks

Task Statement 2.1: Understand the AWS shared responsibility model.

Knowledge of:

- AWS shared responsibility model

Skills in:

- Recognizing the components of the AWS shared responsibility model
- Describing the customer's responsibilities on AWS
- Describing AWS responsibilities
- Describing responsibilities that the customer and AWS share
- Describing how AWS responsibilities and customer responsibilities can shift, depending on the service used (for example, Amazon RDS, AWS Lambda, Amazon EC2)

Task Statement 2.2: Understand AWS Cloud security, governance, and compliance concepts.

Knowledge of:

- AWS compliance and governance concepts
- Benefits of cloud security (for example, encryption)
- Where to capture and locate logs that are associated with cloud security

Skills in:

- Identifying where to find AWS compliance information (for example, AWS Artifact)
- Understanding compliance needs among geographic locations or industries (for example, AWS compliance)
- Describing how customers secure resources on AWS (for example, Amazon Inspector, AWS Security Hub, Amazon GuardDuty, AWS Shield)
- Identifying encryption options (for example, encryption in transit, encryption at rest)
- Recognizing services that aid in governance and compliance (for example, monitoring with Amazon CloudWatch; auditing with AWS CloudTrail, AWS Audit Manager, and AWS Config; reporting with access reports)
- Recognizing compliance requirements that vary among AWS services

Task Statement 2.3: Identify AWS access management capabilities.

Knowledge of:

- Identity and access management (for example, AWS Identity and Access Management [IAM])
- Importance of protecting the AWS root user account
- Principle of least privilege
- AWS IAM Identity Center (AWS Single Sign-On)

Skills in:

- Understanding access keys, password policies, and credential storage (for example, AWS Secrets Manager, AWS Systems Manager)
- Identifying authentication methods in AWS (for example, multi-factor authentication [MFA], IAM Identity Center, cross-account IAM roles)
- Defining groups, users, custom policies, and managed policies in compliance with the principle of least privilege
- Identifying tasks that only the account root user can perform
- Understanding which methods can achieve root user protection
- Understanding the types of identity management (for example, federated)

Task Statement 2.4: Identify components and resources for security.

Knowledge of:

- Security capabilities that AWS provides
- Security-related documentation that AWS provides

Skills in:

- Describing AWS security features and services (for example, AWS WAF, AWS Firewall Manager, AWS Shield, Amazon GuardDuty)
- Understanding that third-party security products are available from AWS Marketplace
- Identifying where AWS security information is available (for example, AWS Knowledge Center, AWS Security Center, AWS Security Blog)
- Understanding the use of AWS services for identifying security issues (for example, AWS Trusted Advisor)

Content Domain 3: Cloud Technology and Services

Domain 3 covers Cloud Technology and Services and represents 34% of the scored content on the exam.

Tasks

Task Statement 3.1: Define methods of deploying and operating in the AWS Cloud.

Knowledge of:

- Various ways of provisioning and operating in the AWS Cloud
- Various ways to access AWS services
- Types of cloud deployment models

Skills in:

- Deciding between options such as programmatic access (for example, APIs, SDKs, CLI), the AWS Management Console, and infrastructure as code (IaC)
- Evaluating requirements to determine whether to use one-time operations or repeatable processes
- Identifying deployment models (for example, cloud, hybrid, on-premises)

Task Statement 3.2: Define the AWS global infrastructure.

Knowledge of:

- AWS Regions, Availability Zones, and edge locations
- High availability
- Use of multiple Regions
- Benefits of edge locations

Skills in:

- Describing relationships among Regions, Availability Zones, and edge locations
- Describing how to achieve high availability by using multiple Availability Zones
- Recognizing that Availability Zones do not share single points of failure
- Describing when to use multiple Regions (for example, disaster recovery, business continuity, low latency for end users, data sovereignty)

Task Statement 3.3: Identify AWS compute services.

Knowledge of:

- AWS compute services

Skills in:

- Recognizing the appropriate use of various Amazon EC2 instance types (for example, compute optimized, storage optimized)
- Recognizing the appropriate use of various container options (for example, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS])
- Recognizing the appropriate use of various serverless compute options (for example, AWS Fargate, AWS Lambda)
- Recognizing that auto scaling provides elasticity
- Identifying the purposes of load balancers

Task Statement 3.4: Identify AWS database services.

Knowledge of:

- AWS database services
- Database migration

Skills in:

- Deciding when to use EC2 hosted databases or AWS managed databases
- Identifying relational databases (for example, Amazon RDS, Amazon Aurora)
- Identifying NoSQL databases (for example, Amazon DynamoDB)
- Identifying memory-based databases (for example, Amazon ElastiCache)
- Identifying database migration tools (for example AWS Database Migration Service [AWS DMS], AWS Schema Conversion Tool [AWS SCT])

Task Statement 3.5: Identify AWS network services.

Knowledge of:

- AWS network services

Skills in:

- Identifying the components of a VPC (for example, subnets, gateways)
- Understanding security in a VPC (for example, network ACLs, security groups, Amazon Inspector)
- Understanding the purpose of Amazon Route 53
- Identifying network connectivity options to AWS (for example AWS VPN, AWS Direct Connect)

Task Statement 3.6: Identify AWS storage services.

Knowledge of:

- AWS storage services

Skills in:

- Identifying the uses for object storage
- Recognizing the differences in Amazon S3 storage classes
- Identifying block storage solutions (for example, Amazon Elastic Block Store [Amazon EBS], instance store)
- Identifying file services (for example, Amazon Elastic File System [Amazon EFS], Amazon FSx)
- Identifying cached file systems (for example, AWS Storage Gateway)
- Understanding use cases for lifecycle policies
- Understanding use cases for AWS Backup

Task Statement 3.7: Identify AWS artificial intelligence and machine learning (AI/ML) services and analytics services.

Knowledge of:

- AWS AI/ML services
- AWS analytics services

Skills in:

- Understanding AI/ML services and the tasks that they accomplish (for example, Amazon SageMaker AI, Amazon Lex, Amazon Kendra)
- Identifying the services for data analytics (for example, Amazon Athena, Amazon Kinesis, AWS Glue, Amazon QuickSight)

Task Statement 3.8: Identify services from other in-scope AWS service categories.

Knowledge of:

- Application integration services of Amazon EventBridge, Amazon Simple Notification Service (Amazon SNS), and Amazon Simple Queue Service (Amazon SQS)
- Business application services of Amazon Connect and Amazon Simple Email Service (Amazon SES)
- Customer enablement services (for example, AWS Support)
- Developer tool services and capabilities (for example, AWS CodeBuild, AWS CodePipeline, and AWS X-Ray)
- End-user computing services of Amazon AppStream 2.0, Amazon WorkSpaces, and Amazon WorkSpaces Secure Browser
- Frontend web and mobile services of AWS Amplify and AWS AppSync
- IoT services (for example, AWS IoT Core)

Skills in:

- Choosing the appropriate service to deliver messages and to send alerts and notifications
- Choosing the appropriate service to meet business application needs
- Choosing the appropriate option for business support assistance
- Identifying the tools to develop, deploy, and troubleshoot applications
- Identifying the services that can present the output of virtual machines (VMs) on end-user machines
- Identifying the services that can create and deploy frontend and mobile services
- Identifying the services that manage IoT devices

Content Domain 4: Billing, Pricing, and Support

Domain 4 covers Billing, Pricing, and Support and represents 12% of the scored content on the exam.

Tasks

Task Statement 4.1: Compare AWS pricing models.

Knowledge of:

- Compute purchasing options (for example, On-Demand Instances, Reserved Instances, Spot Instances, AWS Savings Plans, Dedicated Hosts, Dedicated Instances, Capacity Reservations)
- Storage options and tiers

Skills in:

- Identifying when to use various compute purchasing options
- Describing Reserved Instance flexibility
- Describing Reserved Instance behavior in AWS Organizations
- Understanding incoming data transfer costs and outgoing data transfer costs (for example, from one AWS Region to another Region, within the same Region)
- Understanding pricing options for various storage options and tiers

Task Statement 4.2: Understand resources for billing, budget, and cost management.

Knowledge of:

- Billing support and information
- Pricing information for AWS services
- AWS Organizations

- AWS cost allocation tags

Skills in:

- Understanding the appropriate uses and capabilities of AWS Budgets, and AWS Cost Explorer
- Understanding the appropriate uses and capabilities of AWS Pricing Calculator
- Understanding AWS Organizations consolidated billing and allocation of costs
- Understanding various types of cost allocation tags and their relation to billing reports (for example, AWS Cost and Usage Report)

Task Statement 4.3: Identify AWS technical resources and AWS Support options.

Knowledge of:

- Resources and documentation available on official AWS websites
- AWS Support plans
- Role of the AWS Partner Network, including independent software vendors and system integrators
- AWS Support Center

Skills in:

- Locating AWS whitepapers, blogs, and documentation on official AWS websites
- Identifying and locating AWS technical resources (for example AWS Prescriptive Guidance, AWS Knowledge Center, AWS re:Post)
- Identifying AWS Support options for AWS customers (for example, customer service and communities, AWS Developer Support, AWS Business Support, AWS Enterprise On-Ramp Support, AWS Enterprise Support)
- Identifying the role of AWS Trusted Advisor, AWS Health Dashboard, and the AWS Health API to help manage and monitor environments for cost optimization
- Identifying the role of the AWS Trust and Safety team to report abuse of AWS resources
- Understanding the role of AWS Partners (for example AWS Marketplace, independent software vendors, system integrators)

- Identifying the benefits of being an AWS Partner (for example, partner training and certification, partner events, partner volume discounts)
- Identifying the key services that AWS Marketplace offers (for example, cost management, governance and entitlement)
- Identifying technical assistance options available at AWS (for example, AWS Professional Services, AWS Solutions Architects)

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- APIs
- Benefits of migrating to the AWS Cloud
- AWS Cloud Adoption Framework (AWS CAF)
- AWS Compliance
- Compute
- Cost management
- Databases
- Amazon EC2 instance types (for example, Reserved Instances, On-Demand Instances, Spot Instances)
- AWS global infrastructure (for example, AWS Regions, Availability Zones)
- Infrastructure as code (IaC)
- AWS Knowledge Center
- Machine learning
- Management and governance
- Migration and data transfer
- Network services
- AWS Partner Network (APN)
- AWS Prescriptive Guidance

- AWS Pricing Calculator
- AWS Professional Services
- AWS re:Post
- AWS SDKs
- Security
- AWS Security Blog
- AWS shared responsibility model
- AWS solutions architects
- Storage
- AWS Support Center
- AWS Support plans
- AWS Well-Architected Framework

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Cloud Practitioner (CLF-C02) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

- Frontend Web and Mobile
- Internet of Things (IoT)
- Machine Learning
- Management and Governance
- Migration and Transfer
- Networking and Content Delivery
- Security, Identity, and Compliance
- Serverless
- Storage

Analytics

- Amazon Athena
- Amazon EMR
- AWS Glue
- Amazon Kinesis
- Amazon OpenSearch Service
- Amazon QuickSight
- Amazon Redshift

Application Integration

- Amazon EventBridge
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Business Applications

- Amazon Connect
- Amazon Simple Email Service (Amazon SES)

Cloud Financial Management

- AWS Budgets
- AWS Cost and Usage Reports
- AWS Cost Explorer
- AWS Marketplace

Compute

- AWS Batch
- Amazon EC2
- AWS Elastic Beanstalk
- Amazon Lightsail
- AWS Outposts

Containers

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

Customer Enablement

- AWS Support

Database

- Amazon Aurora
- Amazon DocumentDB
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Neptune
- Amazon RDS

Developer Tools

- AWS CLI
- AWS CodeBuild
- AWS CodePipeline
- AWS X-Ray

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces
- Amazon WorkSpaces Secure Browser

Frontend Web and Mobile

- AWS Amplify
- AWS AppSync

Internet of Things (IoT)

- AWS IoT Core

Machine Learning

- Amazon Comprehend
- Amazon Kendra
- Amazon Lex
- Amazon Polly
- Amazon Q
- Amazon Rekognition
- Amazon SageMaker AI
- Amazon Textract
- Amazon Transcribe

- Amazon Translate

Management and Governance

- AWS Auto Scaling
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Compute Optimizer
- AWS Config
- AWS Control Tower
- AWS Health Dashboard
- AWS License Manager
- AWS Management Console
- AWS Organizations
- AWS Service Catalog
- Service Quotas
- AWS Systems Manager
- AWS Trusted Advisor
- AWS Well-Architected Tool

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Database Migration Service (AWS DMS)
- Migration Evaluator
- AWS Migration Hub
- AWS Schema Conversion Tool (AWS SCT)
- AWS Snow Family

Networking and Content Delivery

- Amazon API Gateway
- Amazon CloudFront
- AWS Direct Connect
- AWS Global Accelerator
- AWS PrivateLink
- Amazon Route 53
- AWS Transit Gateway
- Amazon VPC
- AWS VPN
- AWS Site-to-Site VPN
- AWS Client VPN

Security, Identity, and Compliance

- 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)
- AWS IAM Identity Center
- Amazon Inspector
- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Resource Access Manager (AWS RAM)

- AWS Secrets Manager
- AWS Security Hub
- AWS Shield
- AWS WAF

Serverless

- AWS Fargate
- AWS Lambda

Storage

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- AWS Elastic Disaster Recovery
- Amazon FSx
- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the AWS Certified Cloud Practitioner (CLF-C02) exam. This list is non-exhaustive and is subject to change.

Topics

- Cloud Financial Management
- Database
- Developer Tools
- Game Tech
- Internet of Things (IoT)
- Machine Learning
- Management and Governance
- Media Services
- Migration and Transfer
- Networking and Content Delivery
- Security, Identity, and Compliance
- Robotics
- Storage

Analytics

- Amazon AppFlow
- AWS Clean Rooms
- AWS Data Exchange
- Amazon DataZone
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Amazon Timestream for LiveAnalytics

Application Integration

- AWS AppFabric
- Amazon Simple Workflow Service

Business Applications

- Amazon WorkDocs
- Amazon WorkMail

Compute

- AWS App Runner
- AWS Copilot
- AWS Wavelength

Cost Management

- AWS Application Cost Profiler
- Amazon DevPay

Customer Enablement

- AWS Activate
- AWS IQ
- AWS Managed Services (AMS)

Cloud Financial Management

- AWS Billing Conductor

Database

- Amazon Keyspaces (for Apache Cassandra)
- Amazon MemoryDB for Redis OSS
- AWS AppConfig

Developer Tools

- AWS Application Composer
- AWS CodeArtifact
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CloudShell

- AWS Device Farm

Game Tech

- Amazon GameLift
- Amazon Lumberyard

Internet of Things (IoT)

- AWS IoT Device Defender
- AWS IoT Greengrass
- Amazon Monitron

Machine Learning

- Amazon Fraud Detector
- Amazon Lookout for Metrics
- Amazon Mechanical Turk
- AWS Panorama
- Amazon Personalize

Management and Governance

- AWS Chatbot
- Amazon Data Lifecycle Manager
- Amazon Elastic Transcoder
- AWS Launch Wizard

Media Services

- 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)

Migration and Transfer

- AWS Migration Hub Refactor Spaces
- AWS Transfer Family

Networking and Content Delivery

- AWS Cloud Map
- AWS Network Access Analyzer
- AWS Ground Station
- Amazon VPC Lattice

Security, Identity, and Compliance

- Amazon Cloud Directory
- AWS Network Firewall

Robotics

- AWS RoboMaker

Storage

- Amazon FSx for Lustre

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified AI Practitioner (AIF-C01)

The AWS Certified AI Practitioner (AIF-C01) exam is designed for individuals who want to demonstrate a foundational understanding of AI concepts and AWS AI tools. This certification focuses on practical business applications of AI.

Topics

Introduction

The AWS Certified AI Practitioner (AIF-C01) exam is designed for individuals who want to demonstrate a foundational understanding of AI concepts and AWS AI tools. This certification focuses on practical business applications of AI.

The exam also validates a candidate's ability to complete the following tasks:

- Describe AI, ML, and generative AI (GenAI) concepts, methods, and strategies in general and on AWS.
- Identify the appropriate use of AI/ML and GenAI technologies to solve business problems.
- Determine the correct types of AI/ML technologies to apply to specific use cases.

- Use AI, ML, and GenAI technologies responsibly.

Target candidate description

The target candidate should have up to 6 months of exposure to AI/ML technologies on AWS. The target candidate uses but does not necessarily build AI/ML solutions on AWS.

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- Familiarity with the core AWS services (for example, Amazon EC2, Amazon S3, AWS Lambda, Amazon Bedrock, and Amazon SageMaker AI) and AWS core services use cases
- Familiarity with the AWS shared responsibility model for security and compliance in the AWS Cloud
- Familiarity with AWS Identity and Access Management (IAM) for securing and controlling access to AWS resources
- Familiarity with AWS service pricing models

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Developing or coding AI/ML models or algorithms
- Implementing data engineering or feature engineering techniques
- Performing hyperparameter tuning or model optimization
- Building and deploying AI/ML pipelines or infrastructure
- Conducting mathematical or statistical analysis of AI/ML models
- Implementing security or compliance protocols for AI/ML systems
- Developing and implementing governance frameworks and policies for AI/ML solutions

Exam content

Question types

The exam contains one or more of the following question types:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors).
- **Multiple response:** Has two or more correct responses out of five or more response options. You must select all the correct responses to receive credit for the question.
- **Ordering:** Has a list of 3–5 responses to complete a specified task. You must select the correct responses and place the responses in the correct order to receive credit for the question.
- **Matching:** Has a list of responses to match with a list of 3–7 prompts. You must match all the pairs correctly to receive credit for the question.

Unanswered questions are scored as incorrect. There is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate them for future use as scored questions. The unscored questions are not identified on the exam.

Exam results

The AWS Certified AI Practitioner (AIF-C01) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 700. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, tasks, and skills for the exam. This guide does not provide a comprehensive list of the content on the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Fundamentals of AI and ML (20% of scored content)
- Content Domain 2: Fundamentals of GenAI (24% of scored content)
- Content Domain 3: Applications of Foundation Models (28% of scored content)
- Content Domain 4: Guidelines for Responsible AI (14% of scored content)
- Content Domain 5: Security, Compliance, and Governance for AI Solutions (14% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- Mentions of AWS Services on the Exam
- In-Scope AWS Services
- Out-of-Scope AWS Services

Content Domain 1: Fundamentals of AI and ML

Domain 1 covers the fundamentals of AI and ML and represents 20% of the scored content on the exam.

Tasks

- Task Statement 1.1: Explain basic AI concepts and terminologies.
- Task Statement 1.2: Identify practical use cases for AI.
- Task Statement 1.3: Describe the ML development lifecycle.

Task Statement 1.1: Explain basic AI concepts and terminologies.

Objectives:

- Define basic AI terms (for example, AI, ML, deep learning, neural networks, computer vision, natural language processing [NLP], model, algorithm, training and inferencing, bias, fairness, fit, large language models(LLMs)).
- Describe the similarities and differences between AI, ML, GenAI, and deep learning.
- Describe various types of inferencing (for example, batch, real-time).
- Describe the different types of data in AI models (for example, labeled and unlabeled, tabular, time-series, image, text, structured and unstructured).
- Describe supervised learning, unsupervised learning, and reinforcement learning.

Task Statement 1.2: Identify practical use cases for AI.

Objectives:

- Recognize applications where AI/ML can provide value (for example, assist human decision making, solution scalability, automation).
- Determine when AI/ML solutions are not appropriate (for example, cost-benefit analyses, situations when a specific outcome is needed instead of a prediction).
- Select the appropriate ML techniques for specific use cases (for example, regression, classification, clustering).
- Identify examples of real-world AI applications (for example, computer vision, NLP, speech recognition, recommendation systems, fraud detection, forecasting).
- Explain the capabilities of AWS managed AI/ML services (for example, Amazon SageMaker AI, Amazon Transcribe, Amazon Translate, Amazon Comprehend, Amazon Lex, Amazon Polly).

Task Statement 1.3: Describe the ML development lifecycle.

Objectives:

- Describe components of an ML pipeline (for example, data collection, exploratory data analysis [EDA], data pre-processing, feature engineering, model training, hyperparameter tuning, evaluation, deployment, monitoring).

- Describe sources of ML models (for example, open source pre-trained models, training custom models).
- Describe methods to use a model in production (for example, managed API service, self-hosted API).
- Identify relevant AWS services and features for each stage of an ML pipeline (for example, SageMaker AI, SageMaker Data Wrangler, SageMaker Feature Store, SageMaker Model Monitor).
- Describe fundamental concepts of ML operations (MLOps) (for example, experimentation, repeatable processes, scalable systems, managing technical debt, achieving production readiness, model monitoring, model re-training).
- Describe model performance metrics (for example, accuracy, Area Under the Curve [AUC], F1 score) and business metrics (for example, cost per user, development costs, customer feedback, return on investment [ROI]) to evaluate ML models.

Content Domain 2: Fundamentals of GenAI

Domain 2 covers the fundamentals of GenAI and represents 24% of the scored content on the exam.

Tasks

- Task Statement 2.1: Explain the basic concepts of GenAI.
- Task Statement 2.2: Understand the capabilities and limitations of GenAI for solving business problems.
- Task Statement 2.3: Describe AWS infrastructure and technologies for building GenAI applications.

Task Statement 2.1: Explain the basic concepts of GenAI.

Objectives:

- Define foundational GenAI concepts (for example, tokens, chunking, embeddings, vectors, prompt engineering, transformer-based LLMs, foundation models [FMs], multimodal models, diffusion models).
- Identify potential use cases for GenAI models (for example, image, video, and audio generation; summarization; AI assistants; translation; code generation; customer service agents; search; recommendation engines).

- Describe the foundation model lifecycle (for example, data selection, model selection, pre-training, fine-tuning, evaluation, deployment, feedback).

Task Statement 2.2: Understand the capabilities and limitations of GenAI for solving business problems.

Objectives:

- Describe the advantages of GenAI (for example, adaptability, responsiveness, simplicity).
- Identify disadvantages of GenAI solutions (for example, hallucinations, interpretability, inaccuracy, nondeterminism).
- Identify factors to consider when selecting GenAI models (for example, model types, performance requirements, capabilities, constraints, compliance).
- Determine business value and metrics for GenAI applications (for example, cross-domain performance, efficiency, conversion rate, average revenue per user, accuracy, customer lifetime value).

Task Statement 2.3: Describe AWS infrastructure and technologies for building GenAI applications.

Objectives:

- Identify AWS services and features to develop GenAI applications (for example, Amazon SageMaker JumpStart, Amazon Bedrock PartyRock, Amazon Q, Amazon Bedrock Data Automation).
- Describe the advantages of using AWS GenAI services to build applications (for example, accessibility, lower barrier to entry, efficiency, cost-effectiveness, speed to market, ability to meet business objectives).
- Describe the benefits of AWS infrastructure for GenAI applications (for example, security, compliance, responsibility, safety).
- Describe cost tradeoffs of AWS GenAI services (for example, responsiveness, availability, redundancy, performance, regional coverage, token-based pricing, provision throughput, custom models).

Content Domain 3: Applications of Foundation Models

Domain 3 covers applications of foundation models and represents 28% of the scored content on the exam.

Tasks

Task Statement 3.1: Describe design considerations for applications that use foundation models (FMs).

Objectives:

- Identify selection criteria to choose pre-trained models (for example, cost, modality, latency, multi-lingual, model size, model complexity, customization, input/output length, prompt caching).
- Describe the effect of inference parameters on model responses (for example, temperature, input/output length).
- Define Retrieval Augmented Generation (RAG) and describe its business applications (for example, Amazon Bedrock Knowledge Bases).
- Identify AWS services that help store embeddings within vector databases (for example, Amazon OpenSearch Service, Amazon Aurora, Amazon Neptune, Amazon RDS for PostgreSQL).
- Explain the cost tradeoffs of various approaches to FM customization (for example, pre-training, fine-tuning, in-context learning, RAG).
- Describe the role of agents in multi-step tasks (for example, Amazon Bedrock Agents, agentic AI, model context protocol).

Task Statement 3.2: Choose effective prompt engineering techniques.

Objectives:

- Define the concepts and constructs of prompt engineering (for example, context, instruction, negative prompts, model latent space, prompt routing).
- Define techniques for prompt engineering (for example, chain-of-thought, zero-shot, single-shot, few-shot, prompt templates).
- Identify and describe the benefits and best practices for prompt engineering (for example, response quality improvement, experimentation, guardrails, discovery, specificity and concision, using multiple comments).
- Define potential risks and limitations of prompt engineering (for example, exposure, poisoning, hijacking, jailbreaking).

Task Statement 3.3: Describe the training and fine-tuning process for FMs.

Objectives:

- Describe the key elements of training an FM (for example, pre-training, fine-tuning, continuous pre-training, distillation).
- Define methods for fine-tuning an FM (for example, instruction tuning, adapting models for specific domains, transfer learning, continuous pre-training).
- Describe how to prepare data to fine-tune an FM (for example, data curation, governance, size, labeling, representativeness, reinforcement learning from human feedback [RLHF]).

Task Statement 3.4: Describe methods to evaluate FM performance.

Objectives:

- Determine approaches to evaluate FM performance (for example, human evaluation, benchmark datasets, Amazon Bedrock Model Evaluation).
- Identify relevant metrics to assess FM performance (for example, Recall-Oriented Understudy for Gisting Evaluation [ROUGE], Bilingual Evaluation Understudy [BLEU], BERTScore).
- Determine whether a FM effectively meets business objectives (for example, productivity, user engagement, task engineering).
- Identify approaches to evaluate the performance of applications built with FMs (for example, RAG, agents, workflows).

Content Domain 4: Guidelines for Responsible AI

Domain 4 covers guidelines for responsible AI and represents 14% of the scored content on the exam.

Tasks

Task Statement 4.1: Explain the development of AI systems that are responsible.

Objectives:

- Identify features of responsible AI (for example, bias, fairness, inclusivity, robustness, safety, veracity).
- Explain how to use tools to identify features of responsible AI (for example, Amazon Bedrock Guardrails).
- Define responsible practices to select a model (for example, environmental considerations, sustainability).
- Identify legal risks of working with GenAI (for example, intellectual property infringement claims, biased model outputs, loss of customer trust, end user risk, hallucinations).
- Identify characteristics of datasets (for example, inclusivity, diversity, curated data sources, balanced datasets).
- Describe effects of bias and variance (for example, effects on demographic groups, inaccuracy, overfitting, underfitting).
- Describe tools to detect and monitor bias, trustworthiness, and truthfulness (for example, analyzing label quality, human audits, subgroup analysis, Amazon SageMaker Clarify, SageMaker Model Monitor, Amazon Augmented AI [Amazon A2I]).

Task Statement 4.2: Recognize the importance of transparent and explainable models.

Objectives:

- Describe the differences between models that are transparent and explainable and models that are not transparent and explainable.

- Describe tools to identify transparent and explainable models (for example, SageMaker Model Cards, open source models, data, licensing).
- Identify tradeoffs between model safety and transparency (for example, measure interpretability and performance).
- Describe principles of human-centered design for explainable AI.

Content Domain 5: Security, Compliance, and Governance for AI Solutions

Domain 5 covers security, compliance, and governance for AI solutions and represents 14% of the scored content on the exam.

Tasks

Task Statement 5.1: Explain methods to secure AI systems.

Objectives:

- Identify AWS services and features to secure AI systems (for example, IAM roles, policies, and permissions; encryption; Amazon Macie; AWS PrivateLink; AWS shared responsibility model).
- Describe the concept of source citation and documenting data origins (for example, data lineage, data cataloging, Amazon SageMaker Model Cards).
- Describe best practices for secure data engineering (for example, assessing data quality, implementing privacy-enhancing technologies, data access control, data integrity).
- Describe security and privacy considerations for AI systems (for example, application security, threat detection, vulnerability management, infrastructure protection, prompt injection, encryption at rest and in transit).

Task Statement 5.2: Recognize governance and compliance regulations for AI systems.

Objectives:

- Identify AWS services and features to assist with governance and regulation compliance (for example, AWS Config, Amazon Inspector, AWS Audit Manager, AWS Artifact, AWS CloudTrail, AWS Trusted Advisor).
- Describe data governance strategies (for example, data lifecycles, logging, residency, monitoring, observation, retention).
- Describe processes to follow governance protocols (for example, policies, review cadence, review strategies, governance frameworks such as the Generative AI Security Scoping Matrix, transparency standards, team training requirements).

In-scope AWS services and features

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the AWS Certified AI Practitioner (AIF-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Analytics

- AWS Data Exchange
- Amazon EMR
- AWS Glue
- AWS Glue DataBrew
- AWS Lake Formation
- Amazon OpenSearch Service
- Amazon QuickSight
- Amazon Redshift

Cloud Financial Management

- AWS Budgets
- AWS Cost Explorer

Compute

- Amazon EC2

Containers

- Amazon Elastic Container Service (Amazon ECS)
- Amazon Elastic Kubernetes Service (Amazon EKS)

Database

- Amazon DocumentDB (with MongoDB compatibility)
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon MemoryDB
- Amazon Neptune
- Amazon RDS

Machine Learning

- Amazon Augmented AI (Amazon A2I)
- Amazon Bedrock
- Amazon Comprehend
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Nova
- Amazon Personalize
- Amazon Polly
- Amazon Q Developer
- Amazon Q Business
- Amazon Rekognition

- Amazon SageMaker AI
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Management and Governance

- AWS CloudTrail
- Amazon CloudWatch
- AWS Config
- AWS Trusted Advisor
- AWS Well-Architected Tool

Networking and Content Delivery

- Amazon CloudFront
- Amazon VPC

Security, Identity, and Compliance

- AWS Artifact
- AWS Audit Manager
- AWS Identity and Access Management (IAM)
- Amazon Inspector
- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Secrets Manager

Storage

- Amazon S3
- Amazon S3 Glacier

Out-of-scope AWS services and features

Topics

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Analytics

- AWS Clean Rooms
- Amazon CloudSearch

- Amazon FinSpace
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Application Integration

- Amazon AppFlow
- Amazon MQ
- Amazon Simple Workflow Service (Amazon SWF)

Business Applications

- Amazon Chime
- Amazon Honeycode
- Amazon Pinpoint
- Amazon Simple Email Service (Amazon SES)
- AWS Supply Chain
- AWS Wickr
- Amazon WorkDocs
- Amazon WorkMail

Cloud Financial Management

- AWS Application Cost Profiler
- AWS Billing Conductor
- AWS Marketplace

Compute

- AWS App Runner
- AWS Elastic Beanstalk
- EC2 Image Builder
- Amazon Lightsail

Containers

- Red Hat OpenShift Service on AWS (ROSA)

Customer Enablement

- AWS IQ
- AWS Managed Services (AMS)
- AWS re:Post Private
- AWS Support

Database

- Amazon Keyspaces (for Apache Cassandra)
- Amazon Quantum Ledger Database (Amazon QLDB)
- Amazon Timestream

Developer Tools

- AWS AppConfig
- AWS Application Composer
- AWS CloudShell
- Amazon CodeCatalyst
- AWS CodeStar
- AWS Fault Injection Service
- AWS X-Ray

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces
- Amazon WorkSpaces Thin Client
- Amazon WorkSpaces Web

Frontend Web and Mobile

- AWS Amplify
- AWS AppSync
- AWS Device Farm
- Amazon Location Service

Internet of Things (IoT)

- AWS IoT Analytics
- AWS IoT Core
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT Events
- AWS IoT FleetWise
- FreeRTOS
- AWS IoT Greengrass
- AWS IoT 1-Click
- AWS IoT RoboRunner
- AWS IoT SiteWise
- AWS IoT TwinMaker

Machine Learning

- AWS DeepComposer
- AWS HealthImaging
- AWS HealthOmics
- Amazon Monitron
- AWS Panorama

Management and Governance

- AWS Control Tower

- AWS Health Dashboard
- AWS Launch Wizard
- AWS License Manager
- Amazon Managed Grafana
- Amazon Managed Service for Prometheus
- AWS OpsWorks
- AWS Organizations
- AWS Proton
- AWS Resilience Hub
- AWS Resource Explorer
- AWS Resource Groups
- AWS Systems Manager Incident Manager
- AWS Service Catalog
- Service Quotas
- AWS Telco Network Builder
- AWS User Notifications

Media

- Amazon Elastic Transcoder
- AWS Elemental MediaConnect
- AWS Elemental MediaConvert
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Elemental MediaTailor
- Amazon Interactive Video Service (Amazon IVS)
- Amazon Nimble Studio

Migration and Transfer

- AWS Application Discovery Service

- AWS Application Migration Service
- AWS Database Migration Service (AWS DMS)
- AWS DataSync
- AWS Mainframe Modernization
- AWS Migration Hub
- AWS Snow Family
- AWS Transfer Family

Networking and Content Delivery

- AWS App Mesh
- AWS Cloud Map
- AWS Direct Connect
- AWS Global Accelerator
- AWS Private 5G
- Amazon Route 53
- Amazon Route 53 Application Recovery Controller
- Amazon VPC IP Address Manager (IPAM)

Security, Identity, and Compliance

- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager
- Amazon GuardDuty
- AWS IAM Identity Center
- AWS Payment Cryptography
- AWS Private Certificate Authority
- AWS Resource Access Manager (AWS RAM)

- AWS Security Hub
- Amazon Security Lake
- AWS Shield
- AWS Signer
- Amazon Verified Permissions
- AWS WAF

Storage

- AWS Backup
- AWS Elastic Disaster Recovery

Mentions of AWS Services on the Exam

AWS Certification is reducing the reading load on this exam by using official short names of well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.

- The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.
- You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.
- Note: Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

Survey

How useful was this exam guide? Let us know by taking our survey.

Exam Guides - Associate

AWS Associate level certification exams validate technical skills and experience in implementing AWS solutions.

Topics

AWS Certified Developer - Associate (DVA-C02)

The AWS Certified Developer - Associate (DVA-C02) exam is intended for individuals who perform a developer role. The exam validates a candidate's ability to demonstrate proficiency in developing, testing, deploying, and debugging AWS Cloud-based applications.

Note: AWS exam guides are periodically reviewed and revised to ensure that each certification exam tests skills and AWS services and features that are current and relevant for the job role(s) that the certification is designed to target. Exam guide revisions will be published at least one month before changes are reflected on your exam. Check the Revisions section for a summary of changes.

Topics

Introduction

The AWS Certified Developer - Associate (DVA-C02) exam is intended for individuals who perform a developer role. The exam validates a candidate's ability to demonstrate proficiency in developing, testing, deploying, and debugging AWS Cloud-based applications.

The exam also validates a candidate's ability to complete the following tasks:

- Develop and optimize applications on AWS.
- Package and deploy by using continuous integration and continuous delivery (CI/CD) workflows.
- Secure application code and data.
- Identify and resolve application issues.

Target Candidate Description

The target candidate should have 1 or more years of hands-on experience in developing and maintaining applications by using AWS services.

Recommended general IT knowledge

The target candidate should have the following general IT knowledge:

- Proficiency in at least one high-level programming language
- Understanding of application lifecycle management
- Basic understanding of cloud-focused applications to write code
- Ability to develop functional applications

- Experience in using development tools

Recommended AWS knowledge

The target candidate should be able to complete the following tasks:

- Develop and secure applications by using AWS service APIs, the AWS Command Line Interface (AWS CLI), and SDKs.
- Use a CI/CD pipeline to deploy applications on AWS.

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Design architectures (for example, distributed systems, microservices, database schemas and modeling).
- Design and create CI/CD pipelines.
- Administer IAM users and groups.
- Administer servers and operating systems.
- Design AWS networking infrastructure (for example, Amazon Virtual Private Cloud [Amazon VPC], AWS Direct Connect).

Exam Content

Response types

There are two types of questions on the exam:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)
- **Multiple response:** Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified Developer - Associate (DVA-C02) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and tasks for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Development with AWS Services (32% of scored content)
- Content Domain 2: Security (26% of scored content)
- Content Domain 3: Deployment (24% of scored content)
- Content Domain 4: Troubleshooting and Optimization (18% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- Mentions of AWS Services on the Exam
- In-Scope AWS Services
- Out-of-Scope AWS Services
- Technologies and Concepts

Survey

How useful was this exam guide? Let us know by taking our survey.

Content Domain 1: Development with AWS Services

Tasks

- Task 1: Develop code for applications hosted on AWS
- Task 2: Develop code for AWS Lambda
- Task 3: Use data stores in application development

Task 1: Develop code for applications hosted on AWS

- Skill 1.1.1: Describe architectural patterns (for example, event-driven, microservices, monolithic, choreography, orchestration, fanout)
- Skill 1.1.2: Describe differences between stateful and stateless concepts
- Skill 1.1.3: Describe differences between tightly coupled and loosely coupled components
- Skill 1.1.4: Describe differences between synchronous and asynchronous patterns
- Skill 1.1.5: Create fault-tolerant and resilient applications in a programming language (for example, Java, C#, Python, JavaScript, TypeScript, Go)
- Skill 1.1.6: Create, extend, and maintain APIs (for example, response/request transformations, enforcing validation rules, overriding status codes)
- Skill 1.1.7: Write and run unit tests in development environments (for example, using AWS SAM)

- Skill 1.1.8: Write code to use messaging services
- Skill 1.1.9: Write code that interacts with AWS services by using APIs and AWS SDKs
- Skill 1.1.10: Handle streaming data using AWS services
- Skill 1.1.11: Use Amazon Q Developer to assist with development
- Skill 1.1.12: Use Amazon EventBridge to implement event-driven patterns
- Skill 1.1.13: Implement resilient application code for third-party service integrations (for example, retry logic, circuit breakers, error handling patterns)

Task 2: Develop code for AWS Lambda

- Skill 1.2.1: Describe the access of private resources in VPCs from Lambda code
- Skill 1.2.2: Configure Lambda functions by defining environment variables and parameters (for example, memory, concurrency, timeout, runtime, handler, layers, extensions, triggers, destinations)
- Skill 1.2.3: Handle the event lifecycle and errors by using code (for example, Lambda Destinations, dead-letter queues)
- Skill 1.2.4: Write and run test code by using AWS services and tools
- Skill 1.2.5: Integrate Lambda functions with AWS services
- Skill 1.2.6: Tune Lambda functions for optimal performance
- Skill 1.2.7: Use Lambda functions to process and transform data in near real time

Task 3: Use data stores in application development

- Skill 1.3.1: Describe high-cardinality partition keys for balanced partition access
- Skill 1.3.2: Describe database consistency models (for example, strongly consistent, eventually consistent)
- Skill 1.3.3: Describe differences between query and scan operations
- Skill 1.3.4: Define Amazon DynamoDB keys and indexing
- Skill 1.3.5: Serialize and deserialize data to provide persistence to a data store
- Skill 1.3.6: Use, manage, and maintain data stores
- Skill 1.3.7: Manage data lifecycles
- Skill 1.3.8: Use data caching services

- Skill 1.3.9: Use specialized data stores based on access patterns (for example, Amazon OpenSearch Service)

Content Domain 2: Security

Tasks

Task 1: Implement authentication and/or authorization for applications and AWS services

- Skill 2.1.1: Use an identity provider to implement federated access (for example, Amazon Cognito, IAM)
- Skill 2.1.2: Secure applications by using bearer tokens
- Skill 2.1.3: Configure programmatic access to AWS
- Skill 2.1.4: Make authenticated calls to AWS services
- Skill 2.1.5: Assume an IAM role
- Skill 2.1.6: Define permissions for IAM principals
- Skill 2.1.7: Implement application-level authorization for fine-grained access control
- Skill 2.1.8: Handle cross-service authentication in microservice architectures

Task 2: Implement encryption by using AWS services

- Skill 2.2.1: Define encryption at rest and in transit
- Skill 2.2.2: Describe certificate management (for example, AWS Private CA)
- Skill 2.2.3: Describe differences between client-side encryption and server-side encryption
- Skill 2.2.4: Use encryption keys to encrypt or decrypt data
- Skill 2.2.5: Generate certificates and SSH keys for development purposes
- Skill 2.2.6: Use encryption across account boundaries
- Skill 2.2.7: Enable and disable key rotation

Task 3: Manage sensitive data in application code

- Skill 2.3.1: Describe data classification (for example, personally identifiable information [PII], protected health information [PHI])
- Skill 2.3.2: Encrypt environment variables that contain sensitive data
- Skill 2.3.3: Use secret management services to secure sensitive data
- Skill 2.3.4: Sanitize sensitive data
- Skill 2.3.5: Implement application-level data masking and sanitization
- Skill 2.3.6: Implement data access patterns for multi-tenant applications

Content Domain 3: Deployment

Tasks

Task 1: Prepare application artifacts to be deployed to AWS

- Skill 3.1.1: Manage the dependencies of the code module (for example, environment variables, configuration files, container images) within the package
- Skill 3.1.2: Organize files and a directory structure for application deployment
- Skill 3.1.3: Use code repositories in deployment environments
- Skill 3.1.4: Apply application requirements for resources (for example, memory, cores)
- Skill 3.1.5: Prepare application configurations for specific environments (for example, by using AWS AppConfig)

Task 2: Test applications in development environments

- Skill 3.2.1: Test deployed code by using AWS services and tools
- Skill 3.2.2: Write integration tests and mock APIs for external dependencies

- Skill 3.2.3: Test applications by using development endpoints (for example, configuring stages in Amazon API Gateway)
- Skill 3.2.4: Deploy application stack updates to existing environments (for example, deploying an AWS SAM template to a different staging environment)
- Skill 3.2.5: Test event-driven applications

Task 3: Automate deployment testing

- Skill 3.3.1: Create application test events (for example, JSON payloads for testing AWS Lambda, API Gateway, AWS SAM resources)
- Skill 3.3.2: Deploy API resources to various environments
- Skill 3.3.3: Create application environments that use approved versions for integration testing (for example, Lambda aliases, container image tags, AWS Amplify branches, AWS Copilot environments)
- Skill 3.3.4: Implement and deploy infrastructure as code (IaC) templates (for example, AWS SAM templates, AWS CloudFormation templates)
- Skill 3.3.5: Manage environments in individual AWS services (for example, differentiating between development, test, and production in API Gateway)
- Skill 3.3.6: Use Amazon Q Developer to generate automated tests

Task 4: Deploy code by using AWS Continuous Integration and Continuous Delivery (CI/CD) services

- Skill 3.4.1: Describe Lambda deployment packaging options
- Skill 3.4.2: Describe API Gateway stages and custom domains
- Skill 3.4.3: Update existing IaC templates (for example, AWS SAM templates, CloudFormation templates)
- Skill 3.4.4: Manage application environments by using AWS services
- Skill 3.4.5: Deploy an application version by using deployment strategies
- Skill 3.4.6: Commit code to a repository to invoke build, test, and deployment actions
- Skill 3.4.7: Use orchestrated workflows to deploy code to different environments
- Skill 3.4.8: Perform application rollbacks by using existing deployment strategies
- Skill 3.4.9: Use labels and branches for version and release management

- Skill 3.4.10: Use existing runtime configurations to create dynamic deployments (for example, using staging variables from API Gateway in Lambda functions)
- Skill 3.4.11: Configure deployment strategies (for example, blue/green, canary, rolling) for application releases

Content Domain 4: Troubleshooting and Optimization

Tasks

Task 1: Assist in a root cause analysis

- Skill 4.1.1: Debug code to identify defects
- Skill 4.1.2: Interpret application metrics, logs, and traces
- Skill 4.1.3: Query logs to find relevant data
- Skill 4.1.4: Implement custom metrics (for example, Amazon CloudWatch embedded metric format [EMF])
- Skill 4.1.5: Review application health by using dashboards and insights
- Skill 4.1.6: Troubleshoot deployment failures by using service output logs
- Skill 4.1.7: Debug service integration issues in applications

Task 2: Instrument code for observability

- Skill 4.2.1: Describe differences between logging, monitoring, and observability
- Skill 4.2.2: Implement an effective logging strategy to record application behavior and state
- Skill 4.2.3: Implement code that emits custom metrics
- Skill 4.2.4: Add annotations for tracing services
- Skill 4.2.5: Implement notification alerts for specific actions (for example, notifications about quota limits or deployment completions)
- Skill 4.2.6: Implement tracing by using AWS services and tools
- Skill 4.2.7: Implement structured logging for application events and user actions

- Skill 4.2.8: Configure application health checks and readiness probes

Task 3: Optimize applications by using AWS services and features

- Skill 4.3.1: Define concurrency
- Skill 4.3.2: Profile application performance
- Skill 4.3.3: Determine minimum memory and compute power for an application
- Skill 4.3.4: Use subscription filter policies to optimize messaging
- Skill 4.3.5: Cache content based on request headers
- Skill 4.3.6: Implement application-level caching to improve performance
- Skill 4.3.7: Optimize application resource usage
- Skill 4.3.8: Analyze application performance issues
- Skill 4.3.9: Use application logs to identify performance bottlenecks

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Analytics
- Application integration
- Compute
- Containers
- Cost and capacity management
- Database
- Developer tools
- Management and governance
- Networking and content delivery
- Security, identity, and compliance
- Storage
- Mentions of AWS services on the exam

- AWS Certification is reducing the reading load on this exam by using official short names for well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.
- The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.
- You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.
- Note: Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

Mentions of AWS Services on the Exam

AWS Certification is reducing the reading load on this exam by using official short names for well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.

- The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.
- You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.
- Note: Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Developer - Associate (DVA-C02) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

Analytics

- Amazon Athena
- Amazon Kinesis
- Amazon OpenSearch Service

Application Integration

- AWS AppSync
- Amazon EventBridge
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Compute

- Amazon EC2
- AWS Elastic Beanstalk
- AWS Lambda

Containers

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

Database

- Amazon Aurora
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon RDS

Developer Tools

- AWS Amplify
- AWS CloudShell
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline
- AWS X-Ray
- Amazon Q Developer

Management and Governance

- AWS AppConfig
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Command Line Interface (AWS CLI)

- AWS Systems Manager

Networking and Content Delivery

- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing
- Amazon Route 53
- Amazon VPC

Security, Identity, and Compliance

- Amazon Cognito
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)
- AWS Secrets Manager
- AWS Security Token Service (AWS STS)
- AWS WAF

Storage

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon S3

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the AWS Certified Developer - Associate (DVA-C02) exam. This list is non-exhaustive and is subject to change.

Topics

- Analytics
- Business Applications

- Compute
- Database
- End User Computing
- Internet of Things (IoT)
- Machine Learning
- Management and Governance
- Media Services
- Migration and Transfer
- Networking and Content Delivery
- Robotics
- Satellite
- Storage

Analytics

- Amazon EMR
- AWS Glue
- Amazon Redshift

Business Applications

- Amazon Connect
- Amazon SES

Compute

- AWS Batch
- Amazon Lightsail
- AWS Outposts

Database

- Amazon DocumentDB

- Amazon Neptune
- Amazon Quantum Ledger Database (Amazon QLDB)

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces

Internet of Things (IoT)

- AWS IoT Core
- AWS IoT Greengrass

Machine Learning

- Amazon Comprehend
- Amazon Forecast
- Amazon Lex
- Amazon Polly
- Amazon Rekognition
- Amazon SageMaker
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Management and Governance

- AWS Config
- AWS Control Tower
- AWS License Manager
- AWS Organizations
- AWS Service Catalog
- AWS Trusted Advisor

Media Services

- Amazon Elastic Transcoder
- Amazon Kinesis Video Streams

Migration and Transfer

- AWS Database Migration Service (AWS DMS)
- AWS DataSync
- AWS Migration Hub
- AWS Snow Family
- AWS Transfer Family

Networking and Content Delivery

- AWS App Mesh
- AWS Cloud Map
- AWS Direct Connect
- AWS Global Accelerator
- AWS PrivateLink
- AWS Transit Gateway

Robotics

- AWS RoboMaker

Satellite

- AWS Ground Station

Storage

- AWS Backup

- Amazon FSx
- AWS Storage Gateway

Revisions

AWS exam guides are periodically reviewed and revised to ensure that each certification exam tests skills and AWS services and features that are current and relevant for the job role(s) that the certification is designed to target. Exam guide revisions will be published at least one month before changes are reflected on your exam.

Version	Publication Date
2.1	December 12, 2024
2.0	

Changes with Version 2.1

The separate knowledge and skills in Version 2.0 of the exam guide were consolidated into one list of skills under each task. Knowledge items in Version 2.0 that overlapped with existing skills were removed in Version 2.1.

Changes to knowledge and skills

Version 2.1	Version 2.0
Skill 3.2.2: Write integration tests and mock APIs for external dependencies	Skills in: Performing mock integration for APIs and resolving integration dependencies
Skill 3.4.11: Configure deployment strategies (blue/green, canary, rolling) for application releases	Knowledge of: Deployment strategies (for example, canary, blue/green, rolling)
Skill 4.2.7: Implement structured logging for application events and user actions	Knowledge of: Structured logging

New skills added

- Skill 1.1.11: Use Amazon Q Developer for development assistance
- Skill 1.1.12: Implement event-driven patterns using Amazon EventBridge
- Skill 1.1.13: Implement resilient application code for third-party service integrations (including retry logic, circuit breakers, and error handling patterns)
- Skill 1.2.7: Implement Lambda functions for real-time data processing and transformation
- Skill 1.3.9: Use specialized data stores based on access patterns (for example, Amazon OpenSearch Service)
- Skill 2.1.7: Implement application-level authorization for fine-grained access control
- Skill 2.1.8: Handle cross-service authentication in microservices architectures
- Skill 2.3.5: Implement application-level data masking and sanitization
- Skill 2.3.6: Implement data access patterns for multi-tenant applications
- Skill 3.1.5: Prepare application configurations for different environments (for example, AWS AppConfig)
- Skill 3.2.5: Test event-driven applications
- Skill 3.3.6: Use Amazon Q Developer to generate automated tests
- Skill 4.1.7: Debug service integration issues in applications
- Skill 4.2.8: Create application health checks and readiness probes
- Skill 4.3.6: Implement application-level caching for improved performance
- Skill 4.3.7: Optimize application resource usage
- Skill 4.3.8: Analyze application performance issues
- Skill 4.3.9: Use application logs to identify performance bottlenecks

Skills removed

There are no knowledge and skills removed on Version 2 other than knowledge items that were already covered by existing skills.

Changes to in- and out-of-scope services

Services added to the in-scope list

- Amazon Q Developer

Services removed from the in-scope list

- AWS Copilot
- Amazon CodeGuru

Services added to the out-of-scope list

- No services were added to the out-of-scope list

Services removed from the out-of-scope list

- AWS Device Farm
- Amazon Lex
- AWS Service Catalog
- AWS Database Migration Service (AWS DMS)

AWS Certified CloudOps Engineer - Associate (SOA-C03)

The AWS Certified CloudOps Engineer - Associate (SOA-C03) exam is intended for CloudOps engineers. The exam validates a candidate's ability to deploy, manage, and operate workloads on AWS.

Topics

Introduction

The AWS Certified CloudOps Engineer - Associate (SOA-C03) exam is intended for CloudOps engineers. The exam validates a candidate's ability to deploy, manage, and operate workloads on AWS.

The exam also validates a candidate's ability to complete the following tasks:

- Support and maintain AWS workloads according to the AWS Well-Architected Framework.
- Perform operations by using the AWS Management Console and the AWS CLI.
- Implement security controls to meet compliance requirements.
- Monitor, log, and troubleshoot systems.
- Apply networking concepts (for example, DNS, TCP, IP, firewalls).
- Implement architectural requirements (for example, high availability, performance, capacity).
- Perform business continuity and disaster recovery procedures.
- Identify, classify, and remediate incidents.

Target candidate description

The target candidate should have 1 year of experience with deployment, management, troubleshooting, networking, and security on AWS. The target candidate also should have at least 1 year of experience in a related operations role such as system administrator.

Recommended general IT knowledge and experience

The target candidate should have the following general IT knowledge and experience:

- Techniques for monitoring, logging, and troubleshooting
- Networking concepts (for example, DNS, TCP, IP, firewalls)

- Implementation of architectural requirements (for example, high availability, performance, capacity)
- Familiarity with at least one scripting language
- Familiarity with at least one major operating system
- Understanding of cloud computing
- Containerization and orchestration basics
- Understanding of continuous integration and continuous delivery (CI/CD) and Git

Recommended AWS knowledge and experience

The target candidate should have the following AWS knowledge:

- The AWS Well-Architected Framework
- AWS storage and container solutions
- AWS monitoring tools
- How to use the AWS Management Console, the AWS CLI, infrastructure as code (IaC) solutions, and AWS CloudFormation
- AWS networking and security services
- How to implement AWS security controls and compliance requirements
- Cloud financial management
- Operations within hybrid and multi-VPC environments
- AWS database services (for example, Amazon RDS, Amazon DynamoDB, Amazon ElastiCache)
- AWS compute services (for example, Amazon EC2, AWS Lambda, Amazon Elastic Container Service [Amazon ECS])

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Design distributed architectures.
- Design CI/CD pipelines.
- Design hybrid and multi-VPC networking.

- Develop software.
- Define security, compliance, and governance requirements.
- Develop ransomware defense strategies.
- Assess and plan resource capacity.
- Analyze costs and total cost of ownership.
- Manage billing and invoicing for AWS services.

Exam content

Response types

The exam includes two types of questions:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)
- **Multiple response:** Has two or more correct responses out of five or more response options

Multiple choice and multiple response: Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

On the exam, unanswered questions are scored as incorrect. There is no penalty for guessing. The exam includes 50 questions that affect your score. These questions include multiple-choice questions and multiple-response questions. Each scored multiple-choice question and each scored multiple-response question counts as a single scored opportunity.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified CloudOps Engineer - Associate (SOA-C03) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Monitoring, Logging, Analysis, Remediation, and Performance Optimization (22% of scored content)
- Content Domain 2: Reliability and Business Continuity (22% of scored content)
- Content Domain 3: Deployment, Provisioning, and Automation (22% of scored content)
- Content Domain 4: Security and Compliance (16% of scored content)
- Content Domain 5: Networking and Content Delivery (18% of scored content)

Content Domain 1: Monitoring, Logging, Analysis, Remediation, and Performance Optimization

Tasks

- Task 1.1: Implement metrics, alarms, and filters by using AWS monitoring and logging services
- Task 1.2: Identify and remediate issues by using monitoring and availability metrics
- Task 1.3: Implement performance optimization strategies for compute, storage, and database resources

Task 1.1: Implement metrics, alarms, and filters by using AWS monitoring and logging services

- Skill 1.1.1: Configure AWS monitoring and logging by using AWS services (for example, Amazon CloudWatch, AWS CloudTrail, Amazon Managed Service for Prometheus)
- Skill 1.1.2: Configure and manage the CloudWatch agent to collect metrics and logs from Amazon EC2 instances, Amazon Elastic Container Service (Amazon ECS) clusters, or Amazon Elastic Kubernetes Service (Amazon EKS) clusters
- Skill 1.1.3: Configure, identify, and troubleshoot CloudWatch alarms that can invoke AWS services directly or through Amazon EventBridge (for example, by creating composite alarms and identifying their invokable actions)
- Skill 1.1.4: Create, implement, and manage customizable and shareable CloudWatch dashboards that display metrics and alarms for AWS resources across multiple accounts and AWS Regions
- Skill 1.1.5: Configure AWS services to send notifications to Amazon Simple Notification Service (Amazon SNS) and to invoke alarms that send notifications to Amazon SNS

Task 1.2: Identify and remediate issues by using monitoring and availability metrics

- Skill 1.2.1: Analyze performance metrics and automate remediation strategies by using AWS services and functionality (for example, CloudWatch, AWS User Notifications, AWS Lambda, AWS Systems Manager, CloudTrail, auto scaling)
- Skill 1.2.2: Use EventBridge to route, enrich, and deliver events, and troubleshoot any issues with event bus rules
- Skill 1.2.3: Create or run custom and predefined Systems Manager Automation runbooks (for example, by using AWS SDKs or custom scripts) to automate tasks and streamline processes on AWS

Task 1.3: Implement performance optimization strategies for compute, storage, and database resources

- Skill 1.3.1: Optimize compute resources and remediate performance problems by using performance metrics, resource tags, and AWS tools

- Skill 1.3.2: Analyze Amazon Elastic Block Store (Amazon EBS) performance metrics, troubleshoot issues, and optimize volume types to improve performance and reduce cost
- Skill 1.3.3: Implement and optimize Amazon S3 performance strategies (for example, AWS DataSync, S3 Transfer Acceleration, multipart uploads, S3 Lifecycle policies) to enhance data transfer, storage efficiency, and access patterns
- Skill 1.3.4: Evaluate and select shared storage solutions (for example, Amazon Elastic File System [Amazon EFS], Amazon FSx), and optimize the solutions (for example, EFS lifecycle policies) for specific use cases and requirements
- Skill 1.3.5: Monitor Amazon RDS metrics (for example, Amazon RDS Performance Insights, CloudWatch alarms), and modify configurations to increase performance efficiency (for example, Performance Insights proactive recommendations, RDS Proxy)
- Skill 1.3.6: Implement, monitor, and optimize EC2 instances and their associated storage and networking capabilities (for example, EC2 placement groups)

Content Domain 2: Reliability and Business Continuity

Tasks

Task 2.1: Implement scalability and elasticity

- Skill 2.1.1: Configure and manage scaling mechanisms in compute environments
- Skill 2.1.2: Implement caching by using AWS services to enhance dynamic scalability (for example, Amazon CloudFront, Amazon ElastiCache)
- Skill 2.1.3: Configure and manage scaling in AWS managed databases (for example, Amazon RDS, Amazon DynamoDB)

Task 2.2: Implement highly available and resilient environments

- Skill 2.2.1: Configure and troubleshoot Elastic Load Balancing (ELB) and Amazon Route 53 health checks
- Skill 2.2.2: Configure fault-tolerant systems (for example, Multi-AZ deployments)

Task 2.3: Implement backup and restore strategies

- Skill 2.3.1: Automate snapshots and backups for AWS resources (for example, Amazon EC2 instances, RDS DB instances, Amazon Elastic Block Store [Amazon EBS] volumes, Amazon S3 buckets, DynamoDB tables) by using AWS services (for example, AWS Backup)
- Skill 2.3.2: Use various methods to restore databases (for example, point-in-time restore) to meet recovery time objective (RTO), recovery point objective (RPO), and cost requirements
- Skill 2.3.3: Implement versioning for storage services (for example, Amazon S3, Amazon FSx)
- Skill 2.3.4: Follow disaster recovery procedures

Content Domain 3: Deployment, Provisioning, and Automation

Tasks

Task 3.1: Provision and maintain cloud resources

- Skill 3.1.1: Create and manage AMIs and container images (for example, Amazon EC2 Image Builder)
- Skill 3.1.2: Create and manage stacks of resources by using AWS CloudFormation and the AWS Cloud Development Kit (AWS CDK)
- Skill 3.1.3: Identify and remediate deployment issues (for example, subnet sizing issues, CloudFormation errors, permissions issues)
- Skill 3.1.4: Provision and share resources across multiple AWS Regions and accounts (for example, AWS Resource Access Manager [AWS RAM], CloudFormation StackSets)
- Skill 3.1.5: Implement deployment strategies and services
- Skill 3.1.6: Use and manage third-party tools to automate resource deployment (for example, Terraform, Git)

Task 3.2: Automate the management of existing resources

- Skill 3.2.1: Use AWS services to automate operational processes (for example, AWS Systems Manager)

- Skill 3.2.2: Implement event-driven automation by using AWS services and features (for example, AWS Lambda, Amazon S3 Event Notifications)

Content Domain 4: Security and Compliance

Tasks

Task 4.1: Implement and manage security and compliance tools and policies

- Skill 4.1.1: Implement AWS Identity and Access Management (IAM) features (for example, password policies, multi-factor authentication [MFA], roles, federated identity, resource policies, policy conditions)
- Skill 4.1.2: Troubleshoot and audit access issues by using AWS tools (for example, AWS CloudTrail, IAM Access Analyzer, IAM policy simulator)
- Skill 4.1.3: Implement multi-account strategies securely
- Skill 4.1.4: Implement remediation based on the results of AWS Trusted Advisor security checks
- Skill 4.1.5: Enforce compliance requirements (for example, AWS Region and service selections)

Task 4.2: Implement strategies to protect data and infrastructure

- Skill 4.2.1: Implement and enforce a data classification scheme
- Skill 4.2.2: Implement, configure, and troubleshoot encryption at rest (for example, AWS Key Management Service [AWS KMS])
- Skill 4.2.3: Implement, configure, and troubleshoot encryption in transit (for example, AWS Certificate Manager [ACM])
- Skill 4.2.4: Securely store secrets by using AWS services
- Skill 4.2.5: Configure reports and remediate findings from AWS services (for example, AWS Security Hub, Amazon GuardDuty, AWS Config, Amazon Inspector)

Content Domain 5: Networking and Content Delivery

Tasks

Task 5.1: Implement and optimize networking features and connectivity

- Skill 5.1.1: Configure a VPC (for example, subnets, route tables, network ACLs, security groups, NAT gateways, internet gateway, egress-only internet gateway)
- Skill 5.1.2: Configure private networking connectivity
- Skill 5.1.3: Audit AWS network protection services (for example, Amazon Route 53 Resolver DNS Firewall, AWS WAF, AWS Shield, AWS Network Firewall) in a single account
- Skill 5.1.4: Optimize the cost of network architectures

Task 5.2: Configure domains, DNS services, and content delivery

- Skill 5.2.1: Configure DNS (for example, Route 53 Resolver)
- Skill 5.2.2: Implement Route 53 routing policies, configurations, and query logging
- Skill 5.2.3: Configure content and service distribution (for example, Amazon CloudFront, AWS Global Accelerator)

Task 5.3: Troubleshoot network connectivity issues

- Skill 5.3.1: Troubleshoot VPC configurations (for example, subnets, route tables, network ACLs, security groups, transit gateways, NAT gateways)
- Skill 5.3.2: Collect and interpret networking logs to troubleshoot issues (for example, VPC flow logs, Elastic Load Balancing [ELB] access logs, AWS WAF web ACL logs, CloudFront logs, container logs)
- Skill 5.3.3: Identify and remediate CloudFront caching issues
- Skill 5.3.4: Identify and troubleshoot hybrid connectivity issues and private connectivity issues
- Skill 5.3.5: Configure and analyze Amazon CloudWatch network monitoring services

In-scope AWS services and features

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions:

Topics

- Analytics
- Application Integration
- Business Applications
- Cloud Financial Management
- Compute
- Containers
- Database
- Developer Tools
- Management and Governance
- Migration and Transfer
- Network and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- Amazon Athena
- Amazon Data Firehose

Application Integration

- Amazon EventBridge
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)

- AWS Step Functions

Business Applications

- Amazon Simple Email Service (Amazon SES)

Cloud Financial Management

- AWS Cost and Usage Reports
- AWS Cost Explorer
- Savings Plans

Compute

- Amazon EC2
- Amazon EC2 Image Builder
- AWS Lambda

Containers

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

Database

- Amazon Aurora
- Amazon Aurora Serverless v2
- Amazon DynamoDB
- Amazon DynamoDB Accelerator (DAX)
- Amazon ElastiCache
- Amazon RDS
- Amazon RDS Proxy

Developer Tools

- AWS X-Ray

Management and Governance

- AWS Auto Scaling
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Compute Optimizer
- AWS Config
- AWS Control Tower
- Amazon Managed Grafana
- AWS Managed Service for Prometheus
- AWS Organizations
- AWS Resource Access Manager (AWS RAM)
- AWS Service Catalog
- Service control policies (SCPs)
- AWS Systems Manager
- AWS Trusted Advisor
- Amazon VPC IP Address Manager (IPAM)

Migration and Transfer

- AWS DataSync

Network and Content Delivery

- Amazon Application Recovery Controller
- AWS Client VPN

- Amazon CloudFront
- Elastic IP addresses
- AWS Global Accelerator
- AWS PrivateLink
- Amazon Route 53
- Amazon Route 53 Resolver DNS Firewall
- AWS Site-to-Site VPN
- AWS Transit Gateway
- Amazon VPC
- VPC Endpoints
- VPC Flow Logs
- VPC peering
- VPC Reachability Analyzer

Security, Identity, and Compliance

- AWS Certificate Manager (ACM)
- Amazon EC2 security groups
- Egress-only internet gateways
- Elastic Load Balancing (ELB)
- Amazon GuardDuty
- AWS IAM Access Analyzer
- AWS IAM Identity Center
- AWS Identity and Access Management (IAM)
- Amazon Inspector
- Internet gateways
- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Network Firewall

- NAT gateways
- Network ACLs
- AWS Secrets Manager
- AWS Security Hub
- AWS Shield
- AWS WAF

Storage

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx
- Amazon S3
- AWS Storage Gateway

Out-of-scope AWS services and features

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Topics

- Analytics
- Application Integration
- Blockchain
- Business Applications
- Compute
- Database
- Developer Tools
- End User Computing

- Frontend Web and Mobile
- Machine Learning
- Migration and Transfer
- Network and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- AWS Clean Rooms
- AWS Data Exchange
- Amazon EMR
- Amazon FinSpace
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Application Integration

- Amazon AppFlow
- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)
- Amazon Simple Workflow Service (Amazon SWF)

Blockchain

- Amazon Managed Blockchain (AMB)

Business Applications

- AWS AppFabric
- Amazon Chime
- Amazon Connect
- AWS End User Messaging SMS
- Amazon One Enterprise

- Amazon Pinpoint
- AWS Supply Chain
- Amazon WorkDocs
- Amazon WorkMail

Compute

- Amazon Lightsail
- AWS Nitro Enclaves
- AWS Parallel Computing Service
- AWS SimSpace Weaver

Database

- Amazon Neptune
- Amazon Timestream

Developer Tools

- AWS AppConfig
- AWS App Studio
- Amazon Q Developer

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces

Frontend Web and Mobile

- AWS AppSync
- AWS Device Farm
- Amazon Location Service

Machine Learning

- Amazon Augmented AI (Amazon A2I)
- Amazon CodeGuru
- Amazon Comprehend
- AWS Deep Learning AMIs (DLAMI)
- AWS HealthLake
- AWS HealthOmics
- Amazon Kendra
- Amazon Lex
- Amazon Polly
- Amazon Rekognition
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Migration Hub
- AWS Transfer Family

Network and Content Delivery

- AWS App Mesh
- AWS Cloud WAN

Security, Identity, and Compliance

- AWS CloudHSM
- AWS Payment Cryptography

- Amazon Security Lake
- AWS Signer

Storage

- Amazon Cloud Directory
- Amazon FSx for OpenZFS

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified Solutions Architect - Associate (SAA-C03)

The AWS Certified Solutions Architect - Associate (SAA-C03) exam is intended for individuals who perform a solutions architect role. The exam validates a candidate's ability to design solutions based on the AWS Well-Architected Framework.

Topics

Introduction

The AWS Certified Solutions Architect - Associate (SAA-C03) exam is intended for individuals who perform a solutions architect role. The exam validates a candidate's ability to design solutions based on the AWS Well-Architected Framework.

The exam also validates a candidate's ability to complete the following tasks:

- Design solutions that incorporate AWS services to meet current business requirements and future projected needs
- Design architectures that are secure, resilient, high-performing, and cost-optimized
- Review existing solutions and determine improvements

Target candidate description

The target candidate should have at least 1 year of hands-on experience designing cloud solutions that use AWS services.

Exam content

Response types

There are two types of questions on the exam:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)
- **Multiple response:** Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified Solutions Architect - Associate (SAA-C03) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Design Secure Architectures (30% of scored content)
- Content Domain 2: Design Resilient Architectures (26% of scored content)
- Content Domain 3: Design High-Performing Architectures (24% of scored content)
- Content Domain 4: Design Cost-Optimized Architectures (20% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- Technologies and Concepts

- Mentions of AWS Services on the Exam
- In-Scope AWS Services
- Out-of-Scope AWS Services

Content Domain 1: Design Secure Architectures

Tasks

- Task 1.1: Design secure access to AWS resources
- Task 1.2: Design secure workloads and applications
- Task 1.3: Determine appropriate data security controls

Task 1.1: Design secure access to AWS resources

Knowledge of:

- Access controls and management across multiple accounts
- AWS federated access and identity services (for example, AWS Identity and Access Management [IAM], AWS IAM Identity Center [AWS Single Sign-On])
- AWS global infrastructure (for example, Availability Zones, AWS Regions)
- AWS security best practices (for example, the principle of least privilege)
- The AWS shared responsibility model

Skills in:

- Applying AWS security best practices to IAM users and root users (for example, multi-factor authentication [MFA])
- Designing a flexible authorization model that includes IAM users, groups, roles, and policies
- Designing a role-based access control strategy (for example, AWS Security Token Service [AWS STS], role switching, cross-account access)
- Designing a security strategy for multiple AWS accounts (for example, AWS Control Tower, service control policies [SCPs])
- Determining the appropriate use of resource policies for AWS services
- Determining when to federate a directory service with IAM roles

Task 1.2: Design secure workloads and applications

Knowledge of:

- Application configuration and credentials security
- AWS service endpoints
- Control ports, protocols, and network traffic on AWS
- Secure application access
- Security services with appropriate use cases (for example, AWS Cognito, AWS GuardDuty, AWS Macie)
- Threat vectors external to AWS (for example, DDoS, SQL injection)

Skills in:

- Designing VPC architectures with security components (for example, security groups, route tables, network ACLs, NAT gateways)
- Determining network segmentation strategies (for example, using public subnets and private subnets)
- Integrating AWS services to secure applications (for example, AWS Shield, AWS WAF, IAM Identity Center, AWS Secrets Manager)
- Securing external network connections to and from the AWS Cloud (for example, VPN, AWS Direct Connect)

Task 1.3: Determine appropriate data security controls

Knowledge of:

- Data access and governance
- Data recovery
- Data retention and classification
- Encryption and appropriate key management

Skills in:

- Aligning AWS technologies to meet compliance requirements

- Encrypting data at rest (for example, AWS Key Management Service [AWS KMS])
- Encrypting data in transit (for example, AWS Certificate Manager [ACM] using TLS)
- Implementing access policies for encryption keys
- Implementing data backups and replications
- Implementing policies for data access, lifecycle, and protection
- Rotating encryption keys and renewing certificates

Content Domain 2: Design Resilient Architectures

Tasks

Task 2.1: Design scalable and loosely coupled architectures

Knowledge of:

- API creation and management (for example, Amazon API Gateway, REST API)
- AWS managed services with appropriate use cases (for example, AWS Transfer Family, Amazon Simple Queue Service [Amazon SQS], AWS Secrets Manager)
- Caching strategies
- Design principles for microservices (for example, stateless workloads compared with stateful workloads)
- Event-driven architectures
- Horizontal scaling and vertical scaling
- How to appropriately use edge accelerators (for example, content delivery network [CDN])
- How to migrate applications into containers
- Load balancing concepts (for example, Application Load Balancer [ALB])
- Multi-tier architectures
- Queuing and messaging concepts (for example, publish/subscribe)
- Serverless technologies and patterns (for example, AWS Fargate, AWS Lambda)

- Storage types with associated characteristics (for example, object, file, block)
- The orchestration of containers (for example, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS])
- When to use read replicas
- Workflow orchestration (for example, AWS Step Functions)

Skills in:

- Designing event-driven, microservice, and/or multi-tier architectures based on requirements
- Determining scaling strategies for components used in an architecture design
- Determining the AWS services required to achieve loose coupling based on requirements
- Determining when to use containers
- Determining when to use serverless technologies and patterns
- Recommending appropriate compute, storage, networking, and database technologies based on requirements
- Using purpose-built AWS services for workloads

Task 2.2: Design highly available and/or fault-tolerant architectures

Knowledge of:

- AWS global infrastructure (for example, Availability Zones, AWS Regions, Amazon Route 53)
- AWS Managed Services (AMS) with appropriate use cases (for example, Amazon Comprehend, Amazon Polly)
- Basic networking concepts (for example, route tables)
- Disaster recovery (DR) strategies (for example, backup and restore, pilot light, warm standby, active-active failover, recovery point objective [RPO], recovery time objective [RTO])
- Distributed design patterns
- Failover strategies
- Immutable infrastructure
- Load balancing concepts (for example, ALB)
- Proxy concepts (for example, Amazon RDS Proxy)

- Service quotas and throttling (for example, how to configure the service quotas for a workload in a standby environment)
- Storage options and characteristics (for example, durability, replication)
- Workload visibility (for example, AWS X-Ray)

Skills in:

- Determining automation strategies to ensure infrastructure integrity
- Determining the AWS services required to provide a highly available and/or fault-tolerant architecture across AWS Regions or Availability Zones
- Identifying metrics based on business requirements to deliver a highly available solution
- Implementing designs to mitigate single points of failure
- Implementing strategies to ensure the durability and availability of data (for example, backups)
- Selecting an appropriate DR strategy to meet business requirements
- Using AWS services that improve the reliability of legacy applications and applications not built for the cloud (for example, when application changes are not possible)
- Using purpose-built AWS services for workloads

Content Domain 3: Design High-Performing Architectures

Tasks

Task 3.1: Determine high-performing and/or scalable storage solutions

Knowledge of:

- Hybrid storage solutions to meet business requirements

- Storage services with appropriate use cases (for example, Amazon S3, Amazon Elastic File System [Amazon EFS], Amazon Elastic Block Store [Amazon EBS])
- Storage types with associated characteristics (for example, object, file, block)

Skills in:

- Determining storage services and configurations that meet performance demands
- Determining storage services that can scale to accommodate future needs

Task 3.2: Design high-performing and elastic compute solutions

Knowledge of:

- AWS compute services with appropriate use cases (for example, AWS Batch, Amazon EMR, Fargate)
- Distributed computing concepts supported by AWS global infrastructure and edge services
- Queuing and messaging concepts (for example, publish/subscribe)
- Scalability capabilities with appropriate use cases (for example, Amazon EC2 Auto Scaling, AWS Auto Scaling)
- Serverless technologies and patterns (for example, Lambda, Fargate)
- The orchestration of containers (for example, Amazon ECS, Amazon EKS)

Skills in:

- Decoupling workloads so that components can scale independently
- Identifying metrics and conditions to perform scaling actions
- Selecting the appropriate compute options and features (for example, EC2 instance types) to meet business requirements
- Selecting the appropriate resource type and size (for example, the amount of Lambda memory) to meet business requirements

Task 3.3: Determine high-performing database solutions

Knowledge of:

- AWS global infrastructure (for example, Availability Zones, AWS Regions)
- Caching strategies and services (for example, Amazon ElastiCache)
- Data access patterns (for example, read-intensive compared with write-intensive)
- Database capacity planning (for example, capacity units, instance types, Provisioned IOPS)
- Database connections and proxies
- Database engines with appropriate use cases (for example, heterogeneous migrations, homogeneous migrations)
- Database replication (for example, read replicas)
- Database types and services (for example, serverless, relational compared with non-relational, in-memory)

Skills in:

- Configuring read replicas to meet business requirements
- Designing database architectures
- Determining an appropriate database engine (for example, MySQL compared with PostgreSQL)
- Determining an appropriate database type (for example, Amazon Aurora, Amazon DynamoDB)
- Integrating caching to meet business requirements

Task 3.4: Determine high-performing and/or scalable network architectures

Knowledge of:

- Edge networking services with appropriate use cases (for example, Amazon CloudFront, AWS Global Accelerator)
- How to design network architecture (for example, subnet tiers, routing, IP addressing)
- Load balancing concepts (for example, Application Load Balancer)
- Network connection options (for example, AWS VPN, Direct Connect, AWS PrivateLink)

Skills in:

- Creating a network topology for various architectures (for example, global, hybrid, multi-tier)
- Determining network configurations that can scale to accommodate future needs

- Determining the appropriate placement of resources to meet business requirements
- Selecting the appropriate load balancing strategy

Task 3.5: Determine high-performing data ingestion and transformation solutions

Knowledge of:

- Data analytics and visualization services with appropriate use cases (for example, Amazon Athena, AWS Lake Formation, Amazon QuickSight)
- Data ingestion patterns (for example, frequency)
- Data transfer services with appropriate use cases (for example, AWS DataSync, AWS Storage Gateway)
- Data transformation services with appropriate use cases (for example, AWS Glue)
- Secure access to ingestion access points
- Sizes and speeds needed to meet business requirements
- Streaming data services with appropriate use cases (for example, Amazon Kinesis)

Skills in:

- Building and securing data lakes
- Designing data streaming architectures
- Designing data transfer solutions
- Implementing visualization strategies
- Selecting appropriate compute options for data processing (for example, Amazon EMR)
- Selecting appropriate configurations for ingestion
- Transforming data between formats (for example, .csv to .parquet)

Content Domain 4: Design Cost-Optimized Architectures

Tasks

- Task 4.3: Design cost-optimized database solutions
- Task 4.4: Design cost-optimized network architectures

Task 4.1: Design cost-optimized storage solutions

Knowledge of:

- Access options (for example, an S3 bucket with Requester Pays object storage)
- AWS cost management service features (for example, cost allocation tags, multi-account billing)
- AWS cost management tools with appropriate use cases (for example, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- AWS storage services with appropriate use cases (for example, Amazon FSx, Amazon EFS, Amazon S3, Amazon EBS)
- Backup strategies
- Block storage options (for example, hard disk drive [HDD] volume types, solid state drive [SSD] volume types)
- Data lifecycles
- Hybrid storage options (for example, DataSync, Transfer Family, Storage Gateway)
- Storage access patterns
- Storage tiering (for example, cold tiering for object storage)
- Storage types with associated characteristics (for example, object, file, block)

Skills in:

- Designing appropriate storage strategies (for example, batch uploads to Amazon S3 compared with individual uploads)
- Determining the correct storage size for a workload
- Determining the lowest cost method of transferring data for a workload to AWS storage
- Determining when storage auto scaling is required
- Managing S3 object lifecycles
- Selecting the appropriate backup and/or archival solution
- Selecting the appropriate service for data migration to storage services

- Selecting the appropriate storage tier
- Selecting the correct data lifecycle for storage
- Selecting the most cost-effective storage service for a workload

Task 4.2: Design cost-optimized compute solutions

Knowledge of:

- AWS cost management service features (for example, cost allocation tags, multi-account billing)
- AWS cost management tools with appropriate use cases (for example, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- AWS global infrastructure (for example, Availability Zones, AWS Regions)
- AWS purchasing options (for example, Spot Instances, Reserved Instances, Savings Plans)
- Distributed compute strategies (for example, edge processing)
- Hybrid compute options (for example, AWS Outposts, AWS Snowball Edge)
- Instance types, families, and sizes (for example, memory optimized, compute optimized, virtualization)
- Optimization of compute utilization (for example, containers, serverless computing, microservices)
- Scaling strategies (for example, auto scaling, hibernation)

Skills in:

- Determining an appropriate load balancing strategy (for example, Application Load Balancer [Layer 7] compared with Network Load Balancer [Layer 4] compared with Gateway Load Balancer)
- Determining appropriate scaling methods and strategies for elastic workloads (for example, horizontal compared with vertical, EC2 hibernation)
- Determining cost-effective AWS compute services with appropriate use cases (for example, Lambda, Amazon EC2, Fargate)
- Determining the required availability for different classes of workloads (for example, production workloads, non-production workloads)
- Selecting the appropriate instance family for a workload

- Selecting the appropriate instance size for a workload

Task 4.3: Design cost-optimized database solutions

Knowledge of:

- AWS cost management service features (for example, cost allocation tags, multi-account billing)
- AWS cost management tools with appropriate use cases (for example, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- Caching strategies
- Data retention policies
- Database capacity planning (for example, capacity units)
- Database connections and proxies
- Database engines with appropriate use cases (for example, heterogeneous migrations, homogeneous migrations)
- Database replication (for example, read replicas)
- Database types and services (for example, relational compared with non-relational, Aurora, DynamoDB)

Skills in:

- Designing appropriate backup and retention policies (for example, snapshot frequency)
- Determining an appropriate database engine (for example, MySQL compared with PostgreSQL)
- Determining cost-effective AWS database services with appropriate use cases (for example, DynamoDB compared with Amazon RDS, serverless)
- Determining cost-effective AWS database types (for example, time series format, columnar format)
- Migrating database schemas and data to different locations and/or different database engines

Task 4.4: Design cost-optimized network architectures

Knowledge of:

- AWS cost management service features (for example, cost allocation tags, multi-account billing)

- AWS cost management tools with appropriate use cases (for example, AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report)
- Load balancing concepts (for example, Application Load Balancer)
- NAT gateways (for example, NAT instance costs compared with NAT gateway costs)
- Network connectivity (for example, private lines, dedicated lines, VPNs)
- Network routing, topology, and peering (for example, AWS Transit Gateway, VPC peering)
- Network services with appropriate use cases (for example, DNS)

Skills in:

- Configuring appropriate NAT gateway types for a network (for example, a single shared NAT gateway compared with NAT gateways for each Availability Zone)
- Configuring appropriate network connections (for example, Direct Connect compared with VPN compared with internet)
- Configuring appropriate network routes to minimize network transfer costs (for example, Region to Region, Availability Zone to Availability Zone, private to public, Global Accelerator, VPC endpoints)
- Determining strategic needs for content delivery networks (CDNs) and edge caching
- Reviewing existing workloads for network optimizations
- Selecting an appropriate throttling strategy
- Selecting the appropriate bandwidth allocation for a network device (for example, a single VPN compared with multiple VPNs, Direct Connect speed)

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Compute
- Cost management
- Database
- Disaster recovery

- High performance
- Management and governance
- Microservices and component delivery
- Migration and data transfer
- Networking, connectivity, and content delivery
- Resiliency
- Security
- Serverless and event-driven design principles
- Storage

Mentions of AWS Services on the Exam

AWS Certification is reducing the reading load on this exam by using official short names for well-known AWS service names that contain abbreviations or parenthetical information. For example, *Amazon Simple Notification Service (Amazon SNS)* appears on the exam as *Amazon SNS*.

- The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.
- You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.
- **Note:** Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions:

Analytics:

- 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 QuickSight
- Amazon Redshift

Application Integration:

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

AWS Cost Management:

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

Compute:

- AWS Batch
- Amazon EC2

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

Containers:

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

Database:

- 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

Developer Tools:

- AWS X-Ray

Front-End Web and Mobile:

- 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

Management and Governance:

- AWS Auto Scaling
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Command Line Interface (AWS CLI)
- AWS Compute Optimizer
- AWS Config
- AWS Control Tower

- AWS Health Dashboard
- AWS License Manager
- Amazon Managed Grafana
- Amazon Managed Service for Prometheus
- AWS Management Console
- AWS Organizations
- AWS Proton
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor
- AWS Well-Architected Tool

Media Services:

- Amazon Elastic Transcoder
- Amazon Kinesis Video Streams

Migration and Transfer:

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

Networking and Content Delivery:

- AWS Client VPN
- Amazon CloudFront

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

Security, Identity, and Compliance:

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager
- Amazon GuardDuty
- AWS IAM Identity Center (AWS Single Sign-On)
- 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 WAF

Serverless:

- AWS AppSync
- AWS Fargate
- AWS Lambda

Storage:

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx (for all types)
- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

Out-of-scope AWS services and features

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Topics

- Analytics
- Application Integration
- AR and VR
- Blockchain
- Compute
- Database
- Developer Tools
- Front-End Web and Mobile

- Game Tech
- Internet of Things
- Machine Learning
- Management and Governance
- Media Services
- Migration and Transfer
- Networking and Content Delivery
- Quantum Technologies
- Robotics
- Satellite

Analytics

- Amazon CloudSearch

Application Integration

- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)

AR and VR

- Amazon Sumerian

Blockchain

- Amazon Managed Blockchain

Compute

- Amazon Lightsail

Database

- Amazon RDS on VMware

Developer Tools

- 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 and SDKs

Front-End Web and Mobile

- Amazon Location Service

Game Tech

- Amazon GameLift
- Amazon Lumberyard

Internet of Things

- All services

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

Management and Governance

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

Media Services

- 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)

Migration and Transfer

- Migration Evaluator

Networking and Content Delivery

- AWS App Mesh
- AWS Cloud Map

Quantum Technologies

- Amazon Braket

Robotics

- AWS RoboMaker

Satellite

- AWS Ground Station

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified Machine Learning Engineer - Associate (MLA-C01)

The AWS Certified Machine Learning Engineer - Associate (MLA-C01) exam validates a candidate's ability to build, operationalize, deploy, and maintain machine learning (ML) solutions and pipelines by using the AWS Cloud.

Topics

Introduction

The AWS Certified Machine Learning Engineer - Associate (MLA-C01) exam validates a candidate's ability to build, operationalize, deploy, and maintain machine learning (ML) solutions and pipelines by using the AWS Cloud.

The exam also validates a candidate's ability to complete the following tasks:

- Ingest, transform, validate, and prepare data for ML modeling.
- Select general modeling approaches, train models, tune hyperparameters, analyze model performance, and manage model versions.
- Choose deployment infrastructure and endpoints, provision compute resources, and configure auto scaling based on requirements.

- Set up continuous integration and continuous delivery (CI/CD) pipelines to automate orchestration of ML workflows.
- Monitor models, data, and infrastructure to detect issues.
- Secure ML systems and resources through access controls, compliance features, and best practices.

Target Candidate Description

The target candidate should have at least 1 year of experience using Amazon SageMaker and other AWS services for ML engineering. The target candidate also should have at least 1 year of experience in a related role such as a backend software developer, DevOps developer, data engineer, or data scientist.

Recommended general IT knowledge

The target candidate should have the following general IT knowledge:

- Basic understanding of common ML algorithms and their use cases
- Data engineering fundamentals, including knowledge of common data formats, ingestion, and transformation to work with ML data pipelines
- Knowledge of querying and transforming data
- Knowledge of software engineering best practices for modular, reusable code development, deployment, and debugging
- Familiarity with provisioning and monitoring cloud and on-premises ML resources
- Experience with CI/CD pipelines and infrastructure as code (IaC)
- Experience with code repositories for version control and CI/CD pipelines

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- Knowledge of SageMaker capabilities and algorithms for model building and deployment
- Knowledge of AWS data storage and processing services for preparing data for modeling
- Familiarity with deploying applications and infrastructure on AWS
- Knowledge of monitoring tools for logging and troubleshooting ML systems

- Knowledge of AWS services for the automation and orchestration of CI/CD pipelines
- Understanding of AWS security best practices for identity and access management, encryption, and data protection

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Designing and architecting full end-to-end ML solutions
- Setting up best practices and guiding ML strategies
- Handling integration with a wide array of services or new tools and technologies
- Working deeply in two or more ML domains (for example, natural language processing [NLP], computer vision)
- Quantizing models and analyzing the impact on accuracy

Exam content

Question types

The exam contains one or more of the following question types:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors).
- **Multiple response:** Has two or more correct responses out of five or more response options. You must select all the correct responses to receive credit for the question.
- **Ordering:** Has a list of 3–5 responses to complete a specified task. You must select the correct responses and place the responses in the correct order to receive credit for the question.
- **Matching:** Has a list of responses to match with a list of 3–7 prompts. You must match all the pairs correctly to receive credit for the question.

Unanswered questions on the exam are scored as incorrect. There is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified Machine Learning Engineer - Associate (MLA-C01) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Data Preparation for Machine Learning (ML) (28% of scored content)
- Content Domain 2: ML Model Development (26% of scored content)
- Content Domain 3: Deployment and Orchestration of ML Workflows (22% of scored content)
- Content Domain 4: ML Solution Monitoring, Maintenance, and Security (24% of scored content)

AWS Services for the Exam

The AWS Certified Machine Learning Engineer - Associate exam covers specific AWS services that are relevant to machine learning engineers. Understanding which services are in scope can help you focus your preparation efforts.

For detailed information about the AWS services covered in the exam, see the following section:

- In-Scope AWS Services

Content Domain 1: Data Preparation for Machine Learning (ML)

Tasks

- Task 1.1: Ingest and store data
- Task 1.2: Transform data and perform feature engineering
- Task 1.3: Ensure data integrity and prepare data for modeling

Task 1.1: Ingest and store data

Knowledge of:

- Data formats and ingestion mechanisms (for example, validated and non-validated formats, Apache Parquet, JSON, CSV, Apache ORC, Apache Avro, RecordIO)
- How to use the core AWS data sources (for example, Amazon S3, Amazon Elastic File System [Amazon EFS], Amazon FSx for NetApp ONTAP)
- How to use AWS streaming data sources to ingest data (for example, Amazon Kinesis, Apache Flink, Apache Kafka)
- AWS storage options, including use cases and tradeoffs

Skills in:

- Extracting data from storage (for example, Amazon S3, Amazon Elastic Block Store [Amazon EBS], Amazon EFS, Amazon RDS, Amazon DynamoDB) by using relevant AWS service options (for example, Amazon S3 Transfer Acceleration, Amazon EBS Provisioned IOPS)
- Choosing appropriate data formats (for example, Parquet, JSON, CSV, ORC) based on data access patterns

- Ingesting data into Amazon SageMaker Data Wrangler and SageMaker Feature Store
- Merging data from multiple sources (for example, by using programming techniques, AWS Glue, Apache Spark)
- Troubleshooting and debugging data ingestion and storage issues that involve capacity and scalability
- Making initial storage decisions based on cost, performance, and data structure

Task 1.2: Transform data and perform feature engineering

Knowledge of:

- Data cleaning and transformation techniques (for example, detecting and treating outliers, imputing missing data, combining, deduplication)
- Feature engineering techniques (for example, data scaling and standardization, feature splitting, binning, log transformation, normalization)
- Encoding techniques (for example, one-hot encoding, binary encoding, label encoding, tokenization)
- Tools to explore, visualize, or transform data and features (for example, SageMaker Data Wrangler, AWS Glue, AWS Glue DataBrew)
- Services that transform streaming data (for example, AWS Lambda, Spark)
- Data annotation and labeling services that create high-quality labeled datasets

Skills in:

- Transforming data by using AWS tools (for example, AWS Glue, DataBrew, Spark running on Amazon EMR, SageMaker Data Wrangler)
- Creating and managing features by using AWS tools (for example, SageMaker Feature Store)
- Validating and labeling data by using AWS services (for example, SageMaker Ground Truth, Amazon Mechanical Turk)

Task 1.3: Ensure data integrity and prepare data for modeling

Knowledge of:

- Pre-training bias metrics for numeric, text, and image data (for example, class imbalance [CI], difference in proportions of labels [DPL])
- Strategies to address CI in numeric, text, and image datasets (for example, synthetic data generation, resampling)
- Techniques to encrypt data
- Data classification, anonymization, and masking
- Implications of compliance requirements (for example, personally identifiable information [PII], protected health information [PHI], data residency)

Skills in:

- Validating data quality (for example, by using DataBrew and AWS Glue Data Quality)
- Identifying and mitigating sources of bias in data (for example, selection bias, measurement bias) by using AWS tools (for example, SageMaker Clarify)
- Preparing data to reduce prediction bias (for example, by using dataset splitting, shuffling, and augmentation)
- Configuring data to load into the model training resource (for example, Amazon EFS, Amazon FSx)

Content Domain 2: ML Model Development

Tasks

Task 2.1: Choose a modeling approach

Knowledge of:

- Capabilities and appropriate uses of ML algorithms to solve business problems
- How to use AWS artificial intelligence (AI) services (for example, Amazon Translate, Amazon Transcribe, Amazon Rekognition, Amazon Bedrock) to solve specific business problems
- How to consider interpretability during model selection or algorithm selection

- Amazon SageMaker AI built-in algorithms and when to apply them

Skills in:

- Assessing available data and problem complexity to determine the feasibility of an ML solution
- Comparing and selecting appropriate ML models or algorithms to solve specific problems
- Choosing built-in algorithms, foundation models, and solution templates (for example, in SageMaker JumpStart and Amazon Bedrock)
- Selecting models or algorithms based on costs
- Selecting AI services to solve common business needs

Task 2.2: Train and refine models

Knowledge of:

- Elements in the training process (for example, epoch, steps, batch size)
- Methods to reduce model training time (for example, early stopping, distributed training)
- Factors that influence model size
- Methods to improve model performance
- Benefits of regularization techniques (for example, dropout, weight decay, L1 and L2)
- Hyperparameter tuning techniques (for example, random search, Bayesian optimization)
- Model hyperparameters and their effects on model performance (for example, number of trees in a tree-based model, number of layers in a neural network)
- Methods to integrate models that were built outside SageMaker AI into SageMaker AI

Skills in:

- Using SageMaker AI built-in algorithms and common ML libraries to develop ML models
- Using SageMaker AI script mode with SageMaker AI supported frameworks to train models (for example, TensorFlow, PyTorch)
- Using custom datasets to fine-tune pre-trained models (for example, Amazon Bedrock, SageMaker JumpStart)
- Performing hyperparameter tuning (for example, by using SageMaker AI automatic model tuning [AMT])

- Integrating automated hyperparameter optimization capabilities
- Preventing model overfitting, underfitting, and catastrophic forgetting (for example, by using regularization techniques, feature selection)
- Combining multiple training models to improve performance (for example, ensembling, stacking, boosting)
- Reducing model size (for example, by altering data types, pruning, updating feature selection, compression)
- Managing model versions for repeatability and audits (for example, by using the SageMaker Model Registry)

Task 2.3: Analyze model performance

Knowledge of:

- Model evaluation techniques and metrics (for example, confusion matrix, heat maps, F1 score, accuracy, precision, recall, Root Mean Square Error [RMSE], receiver operating characteristic [ROC], Area Under the ROC Curve [AUC])
- Methods to create performance baselines
- Methods to identify model overfitting and underfitting
- Metrics available in SageMaker Clarify to gain insights into ML training data and models
- Convergence issues

Skills in:

- Selecting and interpreting evaluation metrics and detecting model bias
- Assessing tradeoffs between model performance, training time, and cost
- Performing reproducible experiments by using AWS services
- Comparing the performance of a shadow variant to the performance of a production variant
- Using SageMaker Clarify to interpret model outputs
- Using SageMaker Model Debugger to debug model convergence

Content Domain 3: Deployment and Orchestration of ML Workflows

Tasks

Task 3.1: Select deployment infrastructure based on existing architecture and requirements

Knowledge of:

- Deployment best practices (for example, versioning, rollback strategies)
- AWS deployment services (for example, Amazon SageMaker AI)
- Methods to serve ML models in real time and in batches
- How to provision compute resources in production environments and test environments (for example, CPU, GPU)
- Model and endpoint requirements for deployment endpoints (for example, serverless endpoints, real-time endpoints, asynchronous endpoints, batch inference)
- How to choose appropriate containers (for example, provided or customized)
- Methods to optimize models on edge devices (for example, SageMaker Neo)

Skills in:

- Evaluating performance, cost, and latency tradeoffs
- Choosing the appropriate compute environment for training and inference based on requirements (for example, GPU or CPU specifications, processor family, networking bandwidth)
- Selecting the correct deployment orchestrator (for example, Apache Airflow, SageMaker Pipelines)
- Selecting multi-model or multi-container deployments
- Selecting the correct deployment target (for example, SageMaker AI endpoints, Kubernetes, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS], AWS Lambda)
- Choosing model deployment strategies (for example, real time, batch)

Task 3.2: Create and script infrastructure based on existing architecture and requirements

Knowledge of:

- Difference between on-demand and provisioned resources
- How to compare scaling policies
- Tradeoffs and use cases of infrastructure as code (IaC) options (for example, AWS CloudFormation, AWS Cloud Development Kit [AWS CDK])
- Containerization concepts and AWS container services
- How to use SageMaker AI endpoint auto scaling policies to meet scalability requirements (for example, based on demand, time)

Skills in:

- Applying best practices to enable maintainable, scalable, and cost-effective ML solutions (for example, automatic scaling on SageMaker AI endpoints, dynamically adding Spot Instances, by using Amazon EC2 instances, by using Lambda behind the endpoints)
- Automating the provisioning of compute resources, including communication between stacks (for example, by using CloudFormation, AWS CDK)
- Building and maintaining containers (for example, Amazon Elastic Container Registry [Amazon ECR], Amazon EKS, Amazon ECS, by using bring your own container [BYOC] with SageMaker AI)
- Configuring SageMaker AI endpoints within the VPC network
- Deploying and hosting models by using the SageMaker AI SDK
- Choosing specific metrics for auto scaling (for example, model latency, CPU utilization, invocations per instance)

Task 3.3: Use automated orchestration tools to set up continuous integration and continuous delivery (CI/CD) pipelines

Knowledge of:

- Capabilities and quotas for AWS CodePipeline, AWS CodeBuild, and AWS CodeDeploy
- Automation and integration of data ingestion with orchestration services
- Version control systems and basic usage (for example, Git)

- CI/CD principles and how they fit into ML workflows
- Deployment strategies and rollback actions (for example, blue/green, canary, linear)
- How code repositories and pipelines work together

Skills in:

- Configuring and troubleshooting CodeBuild, CodeDeploy, and CodePipeline, including stages
- Applying continuous deployment flow structures to invoke pipelines (for example, Gitflow, GitHub Flow)
- Using AWS services to automate orchestration (for example, to deploy ML models, automate model building)
- Configuring training and inference jobs (for example, by using Amazon EventBridge rules, SageMaker Pipelines, CodePipeline)
- Creating automated tests in CI/CD pipelines (for example, integration tests, unit tests, end-to-end tests)
- Building and integrating mechanisms to retrain models

Content Domain 4: ML Solution Monitoring, Maintenance, and Security

Tasks

Task 4.1: Monitor model inference

Knowledge of:

- Drift in ML models
- Techniques to monitor data quality and model performance
- Design principles for ML lenses relevant to monitoring

Skills in:

- Monitoring models in production (for example, by using Amazon SageMaker Model Monitor)
- Monitoring workflows to detect anomalies or errors in data processing or model inference
- Detecting changes in the distribution of data that can affect model performance (for example, by using SageMaker Clarify)
- Monitoring model performance in production by using A/B testing

Task 4.2: Monitor and optimize infrastructure and costs

Knowledge of:

- Key performance metrics for ML infrastructure (for example, utilization, throughput, availability, scalability, fault tolerance)
- Monitoring and observability tools to troubleshoot latency and performance issues (for example, AWS X-Ray, Amazon CloudWatch Lambda Insights, Amazon CloudWatch Logs Insights)
- How to use AWS CloudTrail to log, monitor, and invoke re-training activities
- Differences between instance types and how they affect performance (for example, memory optimized, compute optimized, general purpose, inference optimized)
- Capabilities of cost analysis tools (for example, AWS Cost Explorer, AWS Billing and Cost Management, AWS Trusted Advisor)
- Cost tracking and allocation techniques (for example, resource tagging)

Skills in:

- Configuring and using tools to troubleshoot and analyze resources (for example, CloudWatch Logs, CloudWatch alarms)
- Creating CloudTrail trails
- Setting up dashboards to monitor performance metrics (for example, by using Amazon QuickSight, CloudWatch dashboards)
- Monitoring infrastructure (for example, by using Amazon EventBridge events)
- Rightsizing instance families and sizes (for example, by using SageMaker AI Inference Recommender and AWS Compute Optimizer)
- Monitoring and resolving latency and scaling issues
- Preparing infrastructure for cost monitoring (for example, by applying a tagging strategy)

- Troubleshooting capacity concerns that involve cost and performance (for example, provisioned concurrency, service quotas, auto scaling)
- Optimizing costs and setting cost quotas by using appropriate cost management tools (for example, AWS Cost Explorer, AWS Trusted Advisor, AWS Budgets)
- Optimizing infrastructure costs by selecting purchasing options (for example, Spot Instances, On-Demand Instances, Reserved Instances, SageMaker AI Savings Plans)

Task 4.3: Secure AWS resources

Knowledge of:

- IAM roles, policies, and groups that control access to AWS services (for example, AWS Identity and Access Management [IAM], bucket policies, SageMaker Role Manager)
- SageMaker AI security and compliance features
- Controls for network access to ML resources
- Security best practices for CI/CD pipelines

Skills in:

- Configuring least privilege access to ML artifacts
- Configuring IAM policies and roles for users and applications that interact with ML systems
- Monitoring, auditing, and logging ML systems to ensure continued security and compliance
- Troubleshooting and debugging security issues
- Building VPCs, subnets, and security groups to securely isolate ML systems

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Machine Learning Engineer - Associate (MLA-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

- Analytics
- Application Integration
- Cloud Financial Management

- Compute
- Containers
- Database
- Developer Tools
- Machine Learning
- Management and Governance
- Media
- Migration and Transfer
- Networking and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- Amazon Athena
- Amazon Data Firehose
- Amazon EMR
- AWS Glue
- AWS Glue DataBrew
- AWS Glue Data Quality
- Amazon Kinesis
- AWS Lake Formation
- Amazon Managed Service for Apache Flink
- Amazon OpenSearch Service
- Amazon QuickSight
- Amazon Redshift

Application Integration

- Amazon EventBridge
- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)
- Amazon Simple Notification Service (Amazon SNS)

- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Cloud Financial Management

- AWS Billing and Cost Management
- AWS Budgets
- AWS Cost Explorer

Compute

- AWS Batch
- Amazon EC2
- AWS Lambda
- AWS Serverless Application Repository

Containers

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

Database

- Amazon DocumentDB (with MongoDB compatibility)
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Neptune
- Amazon RDS

Developer Tools

- AWS Cloud Development Kit (AWS CDK)

- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline
- AWS X-Ray

Machine Learning

- Amazon Augmented AI (Amazon A2I)
- Amazon Bedrock
- Amazon CodeGuru
- Amazon Comprehend
- Amazon Comprehend Medical
- Amazon DevOps Guru
- Amazon Fraud Detector
- AWS HealthLake
- Amazon Kendra
- Amazon Lex
- Amazon Lookout for Equipment
- Amazon Lookout for Metrics
- Amazon Lookout for Vision
- Amazon Mechanical Turk
- Amazon Personalize
- Amazon Polly
- Amazon Q
- Amazon Rekognition
- Amazon SageMaker
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Management and Governance

- AWS Auto Scaling
- AWS Chatbot
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS Compute Optimizer
- AWS Config
- AWS Organizations
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor

Media

- Amazon Kinesis Video Streams

Migration and Transfer

- AWS DataSync

Networking and Content Delivery

- Amazon API Gateway
- Amazon CloudFront
- AWS Direct Connect
- Amazon VPC

Security, Identity, and Compliance

- AWS Identity and Access Management (IAM)

- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Secrets Manager

Storage

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx
- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the AWS Certified Machine Learning Engineer - Associate (MLA-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list.

Topics

- Analytics
- Application Integration
- Business Applications
- Cloud Financial Management
- Compute
- Containers
- Customer Enablement
- Developer Tools
- End User Computing
- Frontend Web and Mobile
- Internet of Things (IoT)

- Machine Learning
- Management and Governance
- Media
- Migration and Transfer
- Networking and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- AWS Clean Rooms
- Amazon DataZone
- Amazon FinSpace

Application Integration

- Amazon AppFlow
- Amazon MQ
- Amazon Simple Workflow Service (Amazon SWF)

Business Applications

- Amazon Chime
- Amazon Connect
- Amazon Honeycode
- Amazon Pinpoint
- Amazon Simple Email Service (Amazon SES)
- AWS Supply Chain
- AWS Wickr
- Amazon WorkDocs
- Amazon WorkMail

Cloud Financial Management

- AWS Application Cost Profiler

Compute

- AWS App Runner
- AWS Elastic Beanstalk
- Amazon Lightsail
- AWS Outposts

Containers

- Red Hat OpenShift Service on AWS (ROSA)

Customer Enablement

- AWS Activate for startups
- AWS IQ
- AWS re:Post Private

Developer Tools

- AWS Application Composer
- AWS CloudShell
- Amazon CodeCatalyst
- AWS Fault Injection Service

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces
- Amazon WorkSpaces Secure Browser
- Amazon WorkSpaces Thin Client

Frontend Web and Mobile

- AWS Amplify
- AWS AppSync
- AWS Device Farm
- Amazon Location Service

Internet of Things (IoT)

- FreeRTOS
- AWS IoT 1-Click
- AWS IoT Core
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT Events
- AWS IoT FleetWise
- AWS IoT Greengrass
- AWS IoT RoboRunner
- AWS IoT SiteWise
- AWS IoT TwinMaker

Machine Learning

- AWS DeepRacer
- AWS HealthImaging
- AWS HealthOmics
- Amazon Monitron
- AWS Panorama

Management and Governance

- AWS AppConfig
- AWS Control Tower

- AWS Launch Wizard
- AWS License Manager
- Amazon Managed Grafana
- AWS Proton
- AWS Resilience Hub
- AWS Resource Explorer
- AWS Telco Network Builder
- AWS User Notifications

Media

- Amazon Elastic Transcoder
- 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)
- Amazon Nimble Studio

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Mainframe Modernization
- AWS Migration Hub

Networking and Content Delivery

- AWS App Mesh

- AWS Cloud Map
- AWS Global Accelerator
- AWS Private 5G
- Amazon Route 53
- Amazon Route 53 Application Recovery Controller
- Amazon VPC IP Address Manager

Security, Identity, and Compliance

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager
- Amazon GuardDuty
- Amazon Inspector
- AWS Payment Cryptography
- AWS Private Certificate Authority
- AWS Resource Access Manager (AWS RAM)
- AWS Security Hub
- AWS Shield
- AWS Signer
- Amazon Verified Permissions
- AWS WAF

Storage

- AWS Elastic Disaster Recovery

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified Data Engineer - Associate (DEA-C01)

The AWS Certified Data Engineer - Associate (DEA-C01) exam validates a candidate's ability to implement data pipelines and to monitor, troubleshoot, and optimize cost and performance issues in accordance with best practices.

Note: AWS exam guides are periodically reviewed and revised to ensure that each certification exam tests skills and AWS services and features that are current and relevant for the job role(s) that the certification is designed to target. Exam guide revisions will be published at least one month before changes are reflected on your exam. Check the Revisions section for a summary of changes.

Topics

Introduction

The AWS Certified Data Engineer - Associate (DEA-C01) exam validates a candidate's ability to implement data pipelines and to monitor, troubleshoot, and optimize cost and performance issues in accordance with best practices.

The exam also validates a candidate's ability to complete the following tasks:

- Ingest and transform data, and orchestrate data pipelines while applying programming concepts.
- Choose an optimal data store, design data models, catalog data schemas, and manage data lifecycles.
- Operationalize, maintain, and monitor data pipelines. Analyze data and ensure data quality.
- Implement appropriate authentication, authorization, data encryption, privacy, and governance. Enable logging.

Target Candidate Description

The target candidate should have the equivalent of 2–3 years of experience in data engineering. The target candidate should understand the effects of volume, variety, and velocity on data ingestion, transformation, modeling, security, governance, privacy, schema design, and optimal data store design. Additionally, the target candidate should have at least 1–2 years of hands-on experience with AWS services.

Recommended general IT knowledge

The target candidate should have the following general IT knowledge:

- Setup and maintenance of extract, transform, and load (ETL) pipelines from ingestion to destination
- Application of high-level but language-agnostic programming concepts as required by the pipeline
- How to use Git commands for source control
- How to use data lakes to store data
- General concepts for networking, storage, and compute
- General concepts of vectors

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- How to use AWS services to accomplish the tasks listed in the Introduction section of this exam guide

- An understanding of the AWS services for encryption, governance, protection, and logging of all data that is part of data pipelines
- The ability to compare AWS services to understand the cost, performance, and functional differences between services
- How to structure SQL queries and how to run SQL queries on AWS services
- An understanding of how to analyze data, verify data quality, and ensure data consistency by using AWS services

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Perform ML training and inferences.
- Demonstrate knowledge of programming language-specific syntax.
- Draw business conclusions based on data.

Exam content

Response types

There are two types of questions on the exam:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)
- **Multiple response:** Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

The AWS Certified Data Engineer - Associate (DEA-C01) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified Data Engineer - Associate (DEA-C01) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Data Ingestion and Transformation (34% of scored content)
- Content Domain 2: Data Store Management (26% of scored content)
- Content Domain 3: Data Operations and Support (22% of scored content)
- Content Domain 4: Data Security and Governance (18% of scored content)

AWS Services for the Exam

The AWS Certified Data Engineer - Associate exam covers specific AWS services that are relevant to data engineers. Understanding which services are in scope can help you focus your preparation efforts.

For detailed information about the AWS services covered in the exam, see the following section:

- In-Scope AWS Services

Content Domain 1: Data Ingestion and Transformation

Tasks

- Task 1.1: Perform data ingestion
- Task 1.2: Transform and process data
- Task 1.3: Orchestrate data pipelines
- Task 1.4: Apply programming concepts

Task 1.1: Perform data ingestion

- Skill 1.1.1: Read data from streaming sources (for example, Amazon Kinesis, Amazon Managed Streaming for Apache Kafka [Amazon MSK], Amazon DynamoDB Streams, AWS Database Migration Service [AWS DMS], AWS Glue, Amazon Redshift).

- Skill 1.1.2: Read data from batch sources (for example, Amazon S3, AWS Glue, Amazon EMR, AWS DMS, Amazon Redshift, AWS Lambda, Amazon AppFlow).
- Skill 1.1.3: Implement appropriate configuration options for batch ingestion.
- Skill 1.1.4: Consume data APIs.
- Skill 1.1.5: Set up schedulers by using Amazon EventBridge, Apache Airflow, or time-based schedules for jobs and crawlers.
- Skill 1.1.6: Set up event triggers (for example, Amazon S3 Event Notifications, EventBridge).
- Skill 1.1.7: Call a Lambda function from Kinesis.
- Skill 1.1.8: Create allowlists for IP addresses to allow connections to data sources.
- Skill 1.1.9: Implement throttling and overcoming rate limits (for example, DynamoDB, Amazon RDS, Kinesis).
- Skill 1.1.10: Manage fan-in and fan-out for streaming data distribution.
- Skill 1.1.11: Describe replayability of data ingestion pipelines.
- Skill 1.1.12: Define stateful and stateless data transactions.

Task 1.2: Transform and process data

- Skill 1.2.1: Optimize container usage for performance needs (for example, Amazon Elastic Kubernetes Service [Amazon EKS], Amazon Elastic Container Service [Amazon ECS]).
- Skill 1.2.2: Connect to different data sources (for example, Java Database Connectivity [JDBC], Open Database Connectivity [ODBC]).
- Skill 1.2.3: Integrate data from multiple sources.
- Skill 1.2.4: Optimize costs while processing data.
- Skill 1.2.5: Implement data transformation services based on requirements (for example, Amazon EMR, AWS Glue, Lambda, Amazon Redshift).
- Skill 1.2.6: Transform data between formats (for example, from .csv to Apache Parquet).
- Skill 1.2.7: Troubleshoot and debug common transformation failures and performance issues.
- Skill 1.2.8: Create data APIs to make data available to other systems by using AWS services.
- Skill 1.2.9: Define volume, velocity, and variety of data (for example, structured data, unstructured data).
- Skill 1.2.10: Integrate large language models (LLMs) for data processing.

Task 1.3: Orchestrate data pipelines

- Skill 1.3.1: Use orchestration services to build workflows for data ETL pipelines (for example, Lambda, EventBridge, Amazon Managed Workflows for Apache Airflow [Amazon MWAA], AWS Step Functions, AWS Glue workflows).
- Skill 1.3.2: Build data pipelines for performance, availability, scalability, resiliency, and fault tolerance.
- Skill 1.3.3: Implement and maintain serverless workflows.
- Skill 1.3.4: Use notification services to send alerts (for example, Amazon Simple Notification Service [Amazon SNS], Amazon Simple Queue Service [Amazon SQS]).

Task 1.4: Apply programming concepts

- Skill 1.4.1: Optimize code to reduce runtime for data ingestion and transformation.
- Skill 1.4.2: Configure Lambda functions to meet concurrency and performance needs.
- Skill 1.4.3: Use programming languages and frameworks for data engineering (for example, Python, SQL, Scala, R, Java, Bash, PowerShell).
- Skill 1.4.4: Use software engineering best practices for data engineering (for example, version control, testing, logging, monitoring).
- Skill 1.4.5: Use Infrastructure as Code (IaC) to deploy data engineering solutions.
- Skill 1.4.6: Use the AWS Serverless Application Model (AWS SAM) to package and deploy serverless data pipelines (for example, Lambda functions, Step Functions, DynamoDB tables).
- Skill 1.4.7: Use and mount storage volumes from within Lambda functions.
- Skill 1.4.8: Use infrastructure as code (IaC) for repeatable resource deployment (for example, AWS CloudFormation and AWS Cloud Development Kit [AWS CDK]).
- Skill 1.4.9: Describe continuous integration and continuous delivery (CI/CD) (implementation, testing, and deployment of data pipelines).
- Skill 1.4.10: Define distributed computing.
- Skill 1.4.11: Describe data structures and algorithms (for example, graph data structures and tree data structures).

Content Domain 2: Data Store Management

Tasks

Task 2.1: Choose a data store

- Skill 2.1.1: Implement the appropriate storage services for specific cost and performance requirements (for example, Amazon Redshift, Amazon EMR, AWS Lake Formation, Amazon RDS, Amazon DynamoDB, Amazon Kinesis Data Streams, Amazon Managed Streaming for Apache Kafka [Amazon MSK]).
- Skill 2.1.2: Configure the appropriate storage services for specific access patterns and requirements (for example, Amazon Redshift, Amazon EMR, Lake Formation, Amazon RDS, DynamoDB).
- Skill 2.1.3: Apply storage services to appropriate use cases (for example, using indexing algorithms like Hierarchical Navigable Small Worlds [HNSW] with Amazon Aurora PostgreSQL and using Amazon MemoryDB for fast key/value pair access).
- Skill 2.1.4: Integrate migration tools into data processing systems (for example, AWS Transfer Family).
- Skill 2.1.5: Implement data migration or remote access methods (for example, Amazon Redshift federated queries, Amazon Redshift materialized views, Amazon Redshift Spectrum).
- Skill 2.1.6: Manage locks to prevent access to data (for example, Amazon Redshift, Amazon RDS).
- Skill 2.1.7: Manage open table formats (for example Apache Iceberg).
- Skill 2.1.8: Describe vector index types (for example, HNSW, IVF).

Task 2.2: Understand data cataloging systems

- Skill 2.2.1: Use data catalogs to consume data from the data's source.
- Skill 2.2.2: Build and reference a technical data catalog (for example, AWS Glue Data Catalog, Apache Hive metastore).
- Skill 2.2.3: Discover schemas and use AWS Glue crawlers to populate data catalogs.
- Skill 2.2.4: Synchronize partitions with a data catalog.
- Skill 2.2.5: Create new source or target connections for cataloging (for example, AWS Glue).

- Skill 2.2.6: Create and manage business data catalogs (for example, Amazon SageMaker Catalog).

Task 2.3: Manage the lifecycle of data

- Skill 2.3.1: Perform load and unload operations to move data between Amazon S3 and Amazon Redshift.
- Skill 2.3.2: Manage S3 Lifecycle policies to change the storage tier of S3 data.
- Skill 2.3.3: Expire data when it reaches a specific age by using S3 Lifecycle policies.
- Skill 2.3.4: Manage S3 versioning and DynamoDB TTL.
- Skill 2.3.5: Delete data to meet business and legal requirements.
- Skill 2.3.6: Protect data with appropriate resiliency and availability.

Task 2.4: Design data models and schema evolution

- Skill 2.4.1: Design schemas for Amazon Redshift, DynamoDB, and Lake Formation.
- Skill 2.4.2: Address changes to the characteristics of data.
- Skill 2.4.3: Perform schema conversion (for example, by using the AWS Schema Conversion Tool [AWS SCT] and AWS Database Migration Service [AWS DMS] Schema Conversion).
- Skill 2.4.4: Establish data lineage by using AWS tools (for example, Amazon SageMaker ML Lineage Tracking and Amazon SageMaker Catalog).
- Skill 2.4.5: Describe best practices for indexing, partitioning strategies, compression, and other data optimization techniques.
- Skill 2.4.6: Describe vectorization concepts (for example, Amazon Bedrock knowledge base).

Content Domain 3: Data Operations and Support

Tasks

Task 3.1: Automate data processing by using AWS services

- Skill 3.1.1: Orchestrate data pipelines (for example, Amazon Managed Workflows for Apache Airflow [Amazon MWAA], AWS Step Functions).
- Skill 3.1.2: Troubleshoot Amazon managed workflows.
- Skill 3.1.3: Call SDKs to access Amazon features from code.
- Skill 3.1.4: Use the features of AWS services to process data (for example, Amazon EMR, Amazon Redshift, AWS Glue).
- Skill 3.1.5: Consume and maintain data APIs.
- Skill 3.1.6: Prepare data for transformation (for example, AWS Glue DataBrew and Amazon SageMaker Unified Studio).
- Skill 3.1.7: Query data (for example, Amazon Athena).
- Skill 3.1.8: Use AWS Lambda to automate data processing.
- Skill 3.1.9: Manage events and schedulers (for example, Amazon EventBridge).

Task 3.2: Analyze data by using AWS services

- Skill 3.2.1: Visualize data by using AWS services and tools (for example, DataBrew, Amazon QuickSight).
- Skill 3.2.2: Verify and clean data (for example, Lambda, Athena, QuickSight, Jupyter Notebooks, Amazon SageMaker Data Wrangler).
- Skill 3.2.3: Use SQL in Amazon Redshift and Athena to query data or to create views.
- Skill 3.2.4: Use Athena notebooks that use Apache Spark to explore data.
- Skill 3.2.5: Describe tradeoffs between provisioned services and serverless services.
- Skill 3.2.6: Define data aggregation, rolling average, grouping, and pivoting.

Task 3.3: Maintain and monitor data pipelines

- Skill 3.3.1: Extract logs for audits.
- Skill 3.3.2: Deploy logging and monitoring solutions to facilitate auditing and traceability.
- Skill 3.3.3: Use notifications during monitoring to send alerts.
- Skill 3.3.4: Troubleshoot performance issues.

- Skill 3.3.5: Use AWS CloudTrail to track API calls.
- Skill 3.3.6: Troubleshoot and maintain pipelines (for example, AWS Glue, Amazon EMR).
- Skill 3.3.7: Use Amazon CloudWatch Logs to log application data (with a focus on configuration and automation).
- Skill 3.3.8: Analyze logs with AWS services (for example, Athena, Amazon EMR, Amazon OpenSearch Service, CloudWatch Logs Insights, big data application logs).

Task 3.4: Ensure data quality

- Skill 3.4.1: Run data quality checks while processing the data (for example, checking for empty fields).
- Skill 3.4.2: Define data quality rules (for example, DataBrew).
- Skill 3.4.3: Investigate data consistency (for example, DataBrew).
- Skill 3.4.4: Describe data sampling techniques.
- Skill 3.4.5: Implement data skew mechanisms.

Content Domain 4: Data Security and Governance

Tasks

Task 4.1: Apply authentication mechanisms

- Skill 4.1.1: Update VPC security groups.
- Skill 4.1.2: Create and update AWS Identity and Access Management (IAM) groups, roles, endpoints, and services.
- Skill 4.1.3: Create and rotate credentials for password management (for example, AWS Secrets Manager).

- Skill 4.1.4: Set up IAM roles for access (for example, AWS Lambda, Amazon API Gateway, AWS CLI, AWS CloudFormation).
- Skill 4.1.5: Apply IAM policies to roles, endpoints, and services (for example, S3 Access Points, AWS PrivateLink).
- Skill 4.1.6: Describe the differences between managed services and unmanaged services.
- Skill 4.1.7: Use domain, domain units, and projects for SageMaker Unified Studio.

Task 4.2: Apply authorization mechanisms

- Skill 4.2.1: Create custom IAM policies when a managed policy does not meet the needs.
- Skill 4.2.2: Store application and database credentials (for example, Secrets Manager, AWS Systems Manager Parameter Store).
- Skill 4.2.3: Provide database users, groups, and roles access and authority in a database (for example, for Amazon Redshift).
- Skill 4.2.4: Manage permissions through AWS Lake Formation (for Amazon Redshift, Amazon EMR, Amazon Athena, and Amazon S3).
- Skill 4.2.5: Apply authorization methods that address business needs (role-based, tag-based, and attribute-based).
- Skill 4.2.6: Construct custom policies that meet the principle of least privilege.

Task 4.3: Ensure data encryption and masking

- Skill 4.3.1: Apply data masking and anonymization according to compliance laws or company policies.
- Skill 4.3.2: Use encryption keys to encrypt or decrypt data (for example, AWS Key Management Service [AWS KMS]).
- Skill 4.3.3: Configure encryption across AWS account boundaries.
- Skill 4.3.4: Enable encryption in transit or before transit for data.

Task 4.4: Prepare logs for audit

- Skill 4.4.1: Use AWS CloudTrail to track API calls.
- Skill 4.4.2: Use Amazon CloudWatch Logs to store application logs.

- Skill 4.4.3: Use AWS CloudTrail Lake for centralized logging queries.
- Skill 4.4.4: Analyze logs by using AWS services (for example, Athena, CloudWatch Logs Insights, Amazon OpenSearch Service).
- Skill 4.4.5: Integrate various AWS services to perform logging (for example, Amazon EMR in cases of large volumes of log data).

Task 4.5: Understand data privacy and governance

- Skill 4.5.1: Grant permissions for data sharing (for example, data sharing for Amazon Redshift).
- Skill 4.5.2: Implement PII identification (for example, Amazon Macie with Lake Formation).
- Skill 4.5.3: Implement data privacy strategies to prevent backups or replications of data to disallowed AWS Regions.
- Skill 4.5.4: Viewing configuration changes that have occurred in an account (for example, AWS Config).
- Skill 4.5.5: Maintain data sovereignty.
- Skill 4.5.6: Manage data access through Amazon SageMaker Catalog projects.
- Skill 4.5.7: Describe governance data framework and data sharing patterns.

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Data Engineer - Associate (DEA-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

- Analytics
- Application Integration
- Cloud Financial Management
- Compute
- Containers
- Database
- Developer Tools

- Web and Mobile
- Machine Learning
- Management and Governance
- Migration and Transfer
- Networking and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- Amazon Athena
- Amazon EMR
- AWS Glue
- AWS Glue DataBrew
- AWS Lake Formation
- Amazon Kinesis Data Firehose
- Amazon Kinesis Data Streams
- Amazon Managed Service for Apache Flink
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Amazon OpenSearch Service
- Amazon QuickSuite
- Amazon SageMaker AI

Application Integration

- Amazon AppFlow
- Amazon EventBridge
- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Cloud Financial Management

- AWS Budgets
- AWS Cost Explorer

Compute

- AWS Batch
- Amazon EC2
- AWS Lambda
- AWS Serverless Application Model (AWS SAM)

Containers

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

Database

- Amazon DocumentDB (with MongoDB compatibility)
- Amazon DynamoDB
- Amazon Keyspaces (for Apache Cassandra)
- Amazon MemoryDB for Redis
- Amazon Neptune
- Amazon RDS
- Amazon Aurora
- Amazon Redshift

Developer Tools

- AWS CLI
- AWS CloudFormation

- AWS Cloud Development Kit (AWS CDK)
- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline
- Amazon Q

Web and Mobile

- Amazon API Gateway

Machine Learning

- Amazon SageMaker AI
- Amazon Bedrock
- Amazon Kendra

Management and Governance

- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS Config
- Amazon Managed Grafana
- AWS Systems Manager
- AWS Well-Architected Tool
- AWS Data Exchange

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Database Migration Service (AWS DMS)
- AWS DataSync

- AWS Snow Family
- AWS Transfer Family

Networking and Content Delivery

- Amazon CloudFront
- AWS PrivateLink
- Amazon Route 53
- Amazon VPC

Security, Identity, and Compliance

- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Secrets Manager
- AWS Shield
- AWS WAF

Storage

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon S3
- Amazon S3 Tables
- Amazon S3 Glacier
- Amazon S3 Tables

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list.

Topics

Analytics

- Amazon FinSpace

Business Applications

- Alexa for Business
- Amazon Chime
- Amazon Connect
- AWS IQ
- Amazon WorkMail

Compute

- AWS App Runner
- AWS Elastic Beanstalk
- Amazon Lightsail
- AWS Outposts
- AWS Serverless Application Repository

Containers

- Red Hat OpenShift Service on AWS (ROSA)

Developer Tools

- AWS Fault Injection Simulator (AWS FIS)
- AWS X-Ray

Frontend Web and Mobile

- AWS Amplify
- AWS AppSync
- AWS Device Farm
- Amazon Location Service
- Amazon Pinpoint
- Amazon Simple Email Service (Amazon SES)

Internet of Things (IoT)

- FreeRTOS
- AWS IoT 1-Click
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT Events

- AWS IoT FleetWise
- AWS IoT RoboRunner
- AWS IoT SiteWise
- AWS IoT TwinMaker

Machine Learning

- Amazon DevOps Guru

Management and Governance

- AWS Activate
- AWS Managed Services (AMS)

Media Services

- Amazon Elastic Transcoder
- 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)
- Amazon Nimble Studio

Migration and Transfer

- AWS Mainframe Modernization
- AWS Migration Hub

Storage

- EC2 Image Builder

Revisions

AWS exam guides are periodically reviewed and updated to ensure that our certification exams test skills and AWS services and features that are relevant for the job role(s) that a certification is designed to target. Exam guide updates will be published approximately one month before updates will be reflected on your exam.

Topics

Change History

Version	Publication date
1.0	This is the first release.
1.1	December 12, 2025

The separate knowledge and skills in Version 1.0 of the exam guide were consolidated into one list of skills under each task. Knowledge items in Version 1.0 that overlapped with existing skills were removed in Version 1.1.

Changes to knowledge and skills

Version 1.1	Version 1.0
Skill 1.4.8: Use infrastructure as code (IaC) for repeatable resource deployment (for	Knowledge of: Infrastructure as code (IaC) for repeatable deployments (for example,

Version 1.1	Version 1.0
example AWS CloudFormation and AWS Cloud Development Kit (AWS CDK)).	AWS Cloud Development Kit [AWS CDK], AWS CloudFormation)
Skill 2.1.3: Apply storage services to appropria te use cases (for example, using indexing algorithms like HNSW with Amazon Aurora PostgreSQL and using Amazon MemoryDB for fast key/value pair access).	Skills in: Applying storage services to appropriate use cases (for example, Amazon S3)
Skill 2.2.2: Build and reference a technical data catalog (for example, AWS Glue Data Catalog, Apache Hive metastore).	Skills in: Building and referencing a data catalog (for example, AWS Glue Data Catalog, Apache Hive metastore)
Skill 2.4.4: Establish data lineage by using AWS tools (for example, Amazon SageMaker ML Lineage Tracking and Amazon SageMaker Catalog).	Skills in: Establishing data lineage by using AWS tools (for example, Amazon SageMaker ML Lineage Tracking)
Skill 3.1.6: Prepare data for transformation (for example, AWS Glue DataBrew and Amazon SageMaker Unified Studio).	Skills in: Preparing data transformation (for example, AWS Glue DataBrew)
Skill 3.2.3: Use SQL in Redshift and Athena to query data or to create views.	Skills in: Using Athena to query data or to create views
Skill 4.2.5: Apply authorization methods that address business needs (role-based, tag-based , and attribute-based).	Knowledge of: Authorization methods (role-based, policy-based, tag-based, and attribute based)
Skill 4.2.6: Construct custom policies that meet the principle of least privilege.	Knowledge of: Principle of least privilege as it applies to AWS security
Skill 4.3.4: Enable encryption in transit or before transit for data.	Skills in: Enabling encryption in transit for data

Version 1.1	Version 1.0
Skill 4.5.4: Viewing configuration changes that have occurred in an account (for example, AWS Config).	Skills in: Managing configuration changes that have occurred in an account (for example, AWS Config)
Skill 4.5.5: Maintain data sovereignty.	Knowledge of: Data sovereignty

New skills added

- Skill 1.2.10: Integrate Large Language Models (LLM) for data processing.
- Skill 2.1.7: Manage open table formats (for example Apache Iceberg).
- Skill 2.1.8: Describe vector index types (for example, HNSW, IVF).
- Skill 2.2.6: Create and manage business data catalogs (for example Amazon SageMaker Catalog).
- Skill 2.4.6: Describe vectorization concepts (for example, Amazon Bedrock knowledge base).
- Skill 4.1.7: Use domain, domain units, and projects for SageMaker Unified Studio.
- Skill 4.5.6: Manage data access through Amazon SageMaker Catalog projects.
- Skill 4.5.7: Describe governance data framework and data sharing patterns.

Skills removed

There are no knowledge and skills removed on Version 1.1 other than knowledge items that were already covered by existing skills.

Changes to in- and out-of-scope services

Services added to the in-scope list

- Amazon Aurora
- Amazon Q
- Amazon Bedrock
- Amazon Kendra
- AWS Data Exchange
- Amazon S3 Tables

Services removed from the in-scope list

- AWS Cloud9
- AWS CodeCommit
- AWS Schema Conversion Tool (AWS SCT)

Services added to the out-of-scope list

No services were added to the out-of-scope list

Services removed from the out-of-scope list (all due to service changes or deprecations)

- Amazon Honeycode
- Amazon WorkDocs
- Amazon Timestream
- Amazon CodeWhisperer

Survey

How useful was this exam guide? Let us know by taking our survey.

Exam Guides - Professional

AWS Professional level certification exams validate advanced technical skills and experience in designing and implementing AWS solutions.

Topics

AWS Certified Generative AI Developer - Professional (AIP-C01)

The AWS Certified Generative AI Developer - Professional (AIP-C01) exam is intended for individuals who perform a GenAI developer role. The exam validates a candidate's ability to effectively integrate foundation models (FMs) into applications and business workflows. This certification demonstrates practical knowledge of how to implement GenAI solutions into production environments by using AWS technologies.

Topics

- Survey

Introduction

The AWS Certified Generative AI Developer - Professional (AIP-C01) exam is intended for individuals who perform a GenAI developer role. The exam validates a candidate's ability to effectively integrate foundation models (FMs) into applications and business workflows. This certification demonstrates practical knowledge of how to implement GenAI solutions into production environments by using AWS technologies.

The exam also validates a candidate's ability to complete the following tasks:

- Design and implement solutions by using vector stores, Retrieval Augmented Generation (RAG), knowledge bases, and other GenAI architectures.
- Integrate FMs into applications and business workflows.
- Apply prompt engineering and management techniques.
- Implement agentic AI solutions.
- Optimize GenAI applications for cost, performance, and business value.
- Implement security, governance, and Responsible AI practices.
- Troubleshoot, monitor, and optimize GenAI applications.
- Evaluate FMs for quality and responsibility.

Target candidate description

The target candidate should have 2 or more years of experience building production-grade applications on AWS or with open-source technologies, general AI/ML or data engineering experience, and 1 year of hands-on experience implementing GenAI solutions.

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- Experience with AWS compute, storage, and networking services
- Understanding of AWS security best practices and identity management
- Experience with AWS deployment and infrastructure as code (IaC) tools
- Familiarity with AWS monitoring and observability services

- Understanding of AWS cost optimization principles

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Model development and training
- Advanced ML techniques
- Data engineering and feature engineering

Refer to the Appendix for a list of technologies and concepts that might appear on the exam, a list of in-scope AWS services and features, and a list of out-of-scope AWS services and features.

Exam content

Question types

The exam contains one or more of the following question types:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors).
- **Multiple response:** Has two or more correct responses out of five or more response options. You must select all the correct responses to receive credit for the question.
- **Ordering:** Has a list of 3–5 responses to complete a specified task. You must select the correct responses and place the responses in the correct order to receive credit for the question.
- **Matching:** Has a list of responses to match with a list of 3–7 prompts. You must match all the pairs correctly to receive credit for the question.

Unanswered questions are scored as incorrect. There is no penalty for guessing. The exam includes 65 questions that affect your score.

Unscored content

The exam includes 10 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate them for future use as scored questions. The unscored questions are not identified on the exam.

Exam results

The AWS Certified Generative AI Developer - Professional (AIP-C01) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 750. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, tasks, and skills for the exam. This guide does not provide a comprehensive list of the content on the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Foundation Model Integration, Data Management, and Compliance (31% of scored content)
- Content Domain 2: Implementation and Integration (26% of scored content)
- Content Domain 3: AI Safety, Security, and Governance (20% of scored content)
- Content Domain 4: Operational Efficiency and Optimization for GenAI Applications (12% of scored content)
- Content Domain 5: Testing, Validation, and Troubleshooting (11% of scored content)

Content Domain 1: Foundation Model Integration, Data Management, and Compliance

Task 1.1: Analyze requirements and design GenAI solutions.

- Skill 1.1.1: Create comprehensive architectural designs that align with specific business needs and technical constraints (for example, by using appropriate FMs, integration patterns, deployment strategies).
- Skill 1.1.2: Develop technical proof-of-concept implementations to validate feasibility, performance characteristics, and business value before proceeding to full-scale deployment (for example, by using Amazon Bedrock).
- Skill 1.1.3: Create standardized technical components to ensure consistent implementation across multiple deployment scenarios (for example, by using the AWS Well-Architected Framework, AWS WA Tool Generative AI Lens).

Task 1.2: Select and configure FMs.

- Skill 1.2.1: Assess and choose FMs to ensure optimal alignment with specific business use cases and technical requirements (for example, by using performance benchmarks, capability analysis, limitation evaluation).
- Skill 1.2.2: Create flexible architecture patterns to enable dynamic model selection and provider switching without requiring code modifications (for example, by using AWS Lambda, Amazon API Gateway, AWS AppConfig).
- Skill 1.2.3: Design resilient AI systems to ensure continuous operation during service disruptions (for example, by using AWS Step Functions circuit breaker patterns, Amazon Bedrock Cross-Region Inference for models that have limited regional availability, cross-Region model deployment, graceful degradation strategies).
- Skill 1.2.4: Implement FM customization deployment and lifecycle management (for example, by using Amazon SageMaker AI to deploy domain-specific fine-tuned models, parameter-efficient adaptation techniques such as low-rank adaptation [LoRA] and adapters for model deployment, SageMaker Model Registry for versioning and to deploy customized models, automated deployment pipelines to update models, rollback strategies for failed deployments, lifecycle management to retire and replace models).

Task 1.3: Implement data validation and processing pipelines for FM consumption.

- Skill 1.3.1: Create comprehensive data validation workflows to ensure data meets quality standards for FM consumption (for example, by using AWS Glue Data Quality, SageMaker Data Wrangler, custom Lambda functions, Amazon CloudWatch metrics).
- Skill 1.3.2: Create data processing workflows to handle complex data types including text, image, audio, and tabular data with specialized processing requirements for FM consumption (for example, by using Amazon Bedrock multimodal models, SageMaker Processing, AWS Transcribe, advanced multimodal pipeline architectures).
- Skill 1.3.3: Format input data for FM inference according to model-specific requirements (for example, by using JSON formatting for Amazon Bedrock API requests, structured data preparation for SageMaker AI endpoints, conversation formatting for dialog-based applications).
- Skill 1.3.4: Enhance input data quality to improve FM response quality and consistency (for example, by using Amazon Bedrock to reformat text, Amazon Comprehend to extract entities, Lambda functions to normalize data).

Task 1.4: Design and implement vector store solutions.

- Skill 1.4.1: Create advanced vector database architectures specifically for FM augmentation to enable efficient semantic retrieval beyond traditional search capabilities (for example, by using Amazon Bedrock Knowledge Bases for hierarchical organization, Amazon OpenSearch Service with the Neural plugin for Amazon Bedrock integration for topic-based segmentation, Amazon RDS with Amazon S3 document repositories, Amazon DynamoDB with vector databases for metadata and embeddings).
- Skill 1.4.2: Develop comprehensive metadata frameworks to improve search precision and context awareness for FM interactions (for example, by using S3 object metadata for document timestamps, custom attributes for authorship information, tagging systems for domain classification).
- Skill 1.4.3: Implement high-performance vector database architectures to optimize semantic search performance at scale for FM retrieval (for example, by using OpenSearch sharding strategies, multi-index approaches for specialized domains, hierarchical indexing techniques).
- Skill 1.4.4: Use AWS services to create integration components to connect with resources (for example, document management systems, knowledge bases, internal wikis for comprehensive data integration in GenAI applications).

- Skill 1.4.5: Design and deploy data maintenance systems to ensure that vector stores contain current and accurate information for FM augmentation (for example, by using incremental update mechanisms, real-time change detection systems, automated synchronization workflows, scheduled refresh pipelines).

Task 1.5: Design retrieval mechanisms for FM augmentation.

- Skill 1.5.1: Develop effective document segmentation approaches to optimize retrieval performance for FM context augmentation (for example, by using Amazon Bedrock chunking capabilities, Lambda functions to implement fixed-size chunking, custom processing for hierarchical chunking based on content structure).
- Skill 1.5.2: Select and configure optimal embedding solutions to create efficient vector representations for semantic search (for example, by using Amazon Titan embeddings based on dimensionality and domain fit, by evaluating performance characteristics of Amazon Bedrock embedding models, by using Lambda functions to batch generate embeddings).
- Skill 1.5.3: Deploy and configure vector search solutions to enable semantic search capabilities for FM augmentation (for example, by using OpenSearch Service with vector search capabilities, Amazon Aurora with the pgvector extension, Amazon Bedrock Knowledge Bases with managed vector store functionality).
- Skill 1.5.4: Create advanced search architectures to improve the relevance and accuracy of retrieved information for FM context (for example, by using OpenSearch for semantic search, hybrid search that combines keywords and vectors, Amazon Bedrock reranker models).
- Skill 1.5.5: Develop sophisticated query handling systems to improve the retrieval effectiveness and result quality for FM augmentation (for example, by using Amazon Bedrock for query expansion, Lambda functions for query decomposition, Step Functions for query transformation).
- Skill 1.5.6: Create consistent access mechanisms to enable seamless integration with FMs (for example, by using function calling interfaces for vector search, Model Context Protocol [MCP] clients for vector queries, standardized API patterns for retrieval augmentation).

Task 1.6: Implement prompt engineering strategies and governance for FM interactions.

- Skill 1.6.1: Create effective model instruction frameworks to control FM behavior and outputs (for example, by using Amazon Bedrock Prompt Management to enforce role definitions,

Amazon Bedrock Guardrails to enforce responsible AI guidelines, template configurations to format responses).

- Skill 1.6.2: Build interactive AI systems to maintain context and improve user interactions with FMs (for example, by using Step Functions for clarification workflows, Amazon Comprehend for intent recognition, DynamoDB for conversation history storage).
- Skill 1.6.3: Implement comprehensive prompt management and governance systems to ensure consistency and oversight of FM operations (for example, by using Amazon Bedrock Prompt Management to create parameterized templates and approval workflows, Amazon S3 to store template repositories, AWS CloudTrail to track usage, Amazon CloudWatch Logs to log access).
- Skill 1.6.4: Develop quality assurance systems to ensure prompt effectiveness and reliability for FMs (for example, by using Lambda functions to verify expected output, Step Functions to test edge cases, CloudWatch to test prompt regression).
- Skill 1.6.5: Enhance FM performance to refine prompts iteratively and improve response quality beyond basic prompting techniques (for example, by using structured input components, output format specifications, chain-of-thought instruction patterns, feedback loops).
- Skill 1.6.6: Design complex prompt systems to handle sophisticated tasks with FMs (for example, by using Amazon Bedrock Prompt Flows for sequential prompt chains, conditional branching based on model responses, reusable prompt components, integrated pre-processing and post-processing steps).

Content Domain 2: Implementation and Integration

Task 2.1: Implement agentic AI solutions and tool integrations.

- Skill 2.1.1: Develop intelligent autonomous systems with appropriate memory and state management capabilities (for example, by using Strands Agents and AWS Agent Squad for multi-agent systems, MCP for agent-tool interactions).
- Skill 2.1.2: Create advanced problem-solving systems to give FMs the ability to break down and solve complex problems by following structured reasoning steps (for example, by using Step Functions to implement ReAct patterns and chain-of-thought reasoning approaches).
- Skill 2.1.3: Develop safeguarded AI workflows to ensure controlled FM behavior (for example, by using Step Functions to implement stopping conditions, Lambda functions to implement timeout mechanisms, IAM policies to enforce resource boundaries, circuit breakers to mitigate failures).

- Skill 2.1.4: Create sophisticated model coordination systems to optimize performance across multiple capabilities (for example, by using specialized FMs to perform complex tasks, custom aggregation logic for model ensembles, model selection frameworks).
- Skill 2.1.5: Develop collaborative AI systems to enhance FM capabilities with human expertise (for example, by using Step Functions to orchestrate review and approval processes, API Gateway to implement feedback collection mechanisms, human augmentation patterns).
- Skill 2.1.6: Implement intelligent tool integrations to extend FM capabilities and to ensure reliable tool operations (for example, by using the Strands API to implement custom behaviors, standardized function definitions, Lambda functions to implement error handling and parameter validation).
- Skill 2.1.7: Develop model extension frameworks to enhance FM capabilities (for example, by using Lambda functions to implement stateless MCP servers that provide lightweight tool access, Amazon ECS to implement MCP servers that provide complex tools, MCP client libraries to ensure consistent access patterns).

Task 2.2: Implement model deployment strategies.

- Skill 2.2.1: Deploy FMs based on specific application needs and performance requirements (for example, by using Lambda functions for on-demand invocation, Amazon Bedrock provisioned throughput configurations, SageMaker AI endpoints to implement hybrid solutions).
- Skill 2.2.2: Deploy FM solutions by addressing unique challenges of large language models (LLMs) that differ from traditional ML deployments (for example, by implementing container-based deployment patterns that are optimized for memory requirements, GPU utilization, and token processing capacity, by following specialized model loading strategies).
- Skill 2.2.3: Develop optimized FM deployment approaches to balance performance and resource requirements for GenAI workloads (for example, by selecting appropriate models, by using smaller pre-trained models for specific tasks, by using API-based model cascading to perform routine queries).

Task 2.3: Design and implement enterprise integration architectures.

- Skill 2.3.1: Create enterprise connectivity solutions to seamlessly incorporate FM capabilities into existing enterprise environments (for example, by using API-based integrations with legacy systems, event-driven architectures to implement loose coupling, data synchronization patterns).

- Skill 2.3.2: Develop integrated AI capabilities to enhance existing applications with GenAI functionality (for example, by using API Gateway to implement microservice integrations, Lambda functions for webhook handlers, Amazon EventBridge to implement event-driven integrations).
- Skill 2.3.3: Create secure access frameworks to ensure appropriate security controls (for example, by using identity federation between FM services and enterprise systems, role-based access control for model and data access, least privilege API access to FMs).
- Skill 2.3.4: Develop cross-environment AI solutions to ensure data compliance across jurisdictions while enabling FM access (for example, by using AWS Outposts for on-premises data integration, AWS Wavelength to perform edge deployments, secure routing between cloud and on-premises resources).
- Skill 2.3.5: Implement CI/CD pipelines and GenAI gateway architectures to implement secure and compliant consumption patterns in enterprise environments (for example, by using AWS CodePipeline, AWS CodeBuild, automated testing frameworks for continuous deployment and testing of GenAI components with security scans and rollback support, centralized abstraction layers, observability and control mechanisms).

Task 2.4: Implement FM API integrations.

- Skill 2.4.1: Create flexible model interaction systems (for example, by using Amazon Bedrock APIs to manage synchronous requests from various compute environments, language-specific AWS SDKs and Amazon SQS for asynchronous processing, API Gateway to provide custom API clients with request validation).
- Skill 2.4.2: Develop real-time AI interaction systems to provide immediate feedback from FM (for example, by using Amazon Bedrock streaming APIs for incremental response delivery, WebSockets or server-sent events to generate text in real time, API Gateway to implement chunked transfer encoding).
- Skill 2.4.3: Create resilient FM systems to ensure reliable operations (for example, by using the AWS SDK for exponential backoff, API Gateway to manage rate limiting, fallback mechanisms for graceful degradation, AWS X-Ray to provide observability across service boundaries).
- Skill 2.4.4: Develop intelligent model routing systems to optimize model selection (for example, by using application code to implement static routing configurations, Step Functions for dynamic content-based routing to specialized FMs, intelligent model routing based on metrics, API Gateway with request transformations for routing logic).

Task 2.5: Implement application integration patterns and development tools.

- Skill 2.5.1: Create FM API interfaces to address the specific requirements of GenAI workloads (for example, by using API Gateway to handle streaming responses, token limit management, retry strategies to handle model timeouts).
- Skill 2.5.2: Develop accessible AI interfaces to accelerate adoption and integration of FMs (for example, by using AWS Amplify to develop declarative UI components, OpenAPI specifications for API-first development approaches, Amazon Bedrock Prompt Flows for no-code workflow builders).
- Skill 2.5.3: Create business system enhancements (for example, by using Lambda functions to implement customer relationship management [CRM] enhancements, Step Functions to orchestrate document processing systems, Amazon Q Business data sources to provide internal knowledge tools, Amazon Bedrock Data Automation to manage automated data processing workflows).
- Skill 2.5.4: Enhance developer productivity to accelerate development workflows for GenAI applications (for example, by using Amazon Q Developer to generate and refactor code, code suggestions for API assistance, AI component testing, performance optimization).
- Skill 2.5.5: Develop advanced GenAI applications to implement sophisticated AI capabilities (for example, by using Strands Agents and AWS Agent Squad for AWS native orchestration, Step Functions to orchestrate agent design patterns, Amazon Bedrock to manage prompt chaining patterns).
- Skill 2.5.6: Improve troubleshooting efficiency for FM applications (for example, by using CloudWatch Logs Insights to analyze prompts and responses, X-Ray to trace FM API calls, Amazon Q Developer to implement GenAI-specific error pattern recognition).

Content Domain 3: AI Safety, Security, and Governance

Task 3.1: Implement input and output safety controls.

- Skill 3.1.1: Develop comprehensive content safety systems to protect against harmful user inputs to FMs (for example, by using Amazon Bedrock guardrails to filter content, Step Functions and Lambda functions to implement custom moderation workflows, real-time validation mechanisms).

- Skill 3.1.2: Create content safety frameworks to prevent harmful outputs (for example, by using Amazon Bedrock guardrails to filter responses, specialized FM evaluations for content moderation and toxicity detection, text-to-SQL transformations to ensure deterministic results).
- Skill 3.1.3: Develop accuracy verification systems to reduce hallucinations in FM responses (for example, by using Amazon Bedrock Knowledge Base to ground responses and perform fact-checking, confidence scoring and semantic similarity search for verification, JSON Schema to enforce structured outputs).
- Skill 3.1.4: Create defense-in-depth safety systems to provide comprehensive protection against FM misuse (for example, by using Amazon Comprehend to develop pre-processing filters, Amazon Bedrock to implement model-based guardrails, Lambda functions to perform post-processing validation, API Gateway to implement API response filtering).
- Skill 3.1.5: Implement advanced threat detection to protect against adversarial inputs and security vulnerabilities (for example, by using prompt injection and jailbreak detection mechanisms, input sanitization and content filters, safety classifiers, automated adversarial testing workflows).

Task 3.2: Implement data security and privacy controls.

- Skill 3.2.1: Develop protected AI environments to ensure comprehensive security for FM deployments (for example, by using VPC endpoints to isolate networks, IAM policies to enforce secure data access patterns, AWS Lake Formation to provide granular data access, CloudWatch to monitor data access).
- Skill 3.2.2: Develop privacy-preserving systems to protect sensitive information during FM interactions (for example, by using Amazon Comprehend and Amazon Macie to detect personally identifiable information [PII], Amazon Bedrock native data privacy features, Amazon Bedrock guardrails to filter outputs, Amazon S3 Lifecycle configurations to implement data retention policies).
- Skill 3.2.3: Create privacy-focused AI systems to protect user privacy while maintaining FM utility and effectiveness (for example, by using data masking techniques, Amazon Comprehend PII detection, anonymization strategies for sensitive information, Amazon Bedrock guardrails).

Task 3.3: Implement AI governance and compliance mechanisms.

- Skill 3.3.1: Develop compliance frameworks to ensure regulatory compliance for FM deployments (for example, by using SageMaker AI to develop programmatic model cards, AWS Glue to

automatically track data lineage, metadata tagging for systematic data source attribution, CloudWatch Logs to collect comprehensive decision logs).

- Skill 3.3.2: Implement data source tracking to maintain traceability in GenAI applications (for example, by using AWS Glue Data Catalog to register data sources, metadata tagging for source attribution in FM-generated content, CloudTrail for audit logging).
- Skill 3.3.3: Create organizational governance systems to ensure consistent oversight of FM implementations (for example, by using comprehensive frameworks that align with organizational policies, regulatory requirements, and responsible AI principles).
- Skill 3.3.4: Implement continuous monitoring and advanced governance controls to support safety audits and regulatory readiness (for example, by using automated detection for misuse, drift, and policy violations, bias drift monitoring, automated alerting and remediation workflows, token-level redaction, response logging, AI output policy filters).

Task 3.4: Implement responsible AI principles.

- Skill 3.4.1: Develop transparent AI systems in FM outputs (for example, by using reasoning displays to provide user-facing explanations, CloudWatch to collect confidence metrics and quantify uncertainty, evidence presentation for source attribution, Amazon Bedrock agent tracing to provide reasoning traces).
- Skill 3.4.2: Apply fairness evaluations to ensure unbiased FM outputs (for example, by using pre-defined fairness metrics in CloudWatch, Amazon Bedrock Prompt Management and Amazon Bedrock Prompt Flows to perform systematic A/B testing, Amazon Bedrock with LLM-as-a-judge solutions to perform automated model evaluations).
- Skill 3.4.3: Develop policy-compliant AI systems to ensure adherence to responsible AI practices (for example, by using Amazon Bedrock guardrails based on policy requirements, model cards to document FM limitations, Lambda functions to perform automated compliance checks).

Content Domain 4: Operational Efficiency and Optimization for GenAI Applications

Task 4.1: Implement cost optimization and resource efficiency strategies.

- Skill 4.1.1: Develop token efficiency systems to reduce FM costs while maintaining effectiveness (for example, by using token estimation and tracking, context window optimization, response size controls, prompt compression, context pruning, response limiting).

- Skill 4.1.2: Create cost-effective model selection frameworks (for example, by using cost-capability tradeoff evaluation, tiered FM usage based on query complexity, inference cost balancing against response quality, price-to-performance ratio measurement, efficient inference patterns).
- Skill 4.1.3: Develop high-performance FM systems to maximize resource utilization and throughput for GenAI workloads (for example, by using batching strategies, capacity planning, utilization monitoring, auto-scaling configurations, provisioned throughput optimization).
- Skill 4.1.4: Create intelligent caching systems to reduce costs and improve response times by avoiding unnecessary FM invocations (for example, by using semantic caching, result fingerprinting, edge caching, deterministic request hashing, prompt caching).

Task 4.2: Optimize application performance.

- Skill 4.2.1: Create responsive AI systems to address latency-cost tradeoffs and improve the user experience with FMs (for example, by using pre-computation to perform predictable queries, latency-optimized Amazon Bedrock models for time-sensitive applications, parallel requests for complex workflows, response streaming, performance benchmarking).
- Skill 4.2.2: Enhance retrieval performance to improve the relevance and speed of retrieved information for FM context augmentation (for example, by using index optimization, query preprocessing, hybrid search implementation with custom scoring).
- Skill 4.2.3: Implement FM throughput optimization to address the specific throughput challenges of GenAI workloads (for example, by using token processing optimization, batch inference strategies, concurrent model invocation management).
- Skill 4.2.4: Enhance FM performance to achieve optimal results for specific GenAI use cases (for example, by using model-specific parameter configurations, A/B testing to evaluate improvements, appropriate temperature and top-k/top-p selection based on requirements).
- Skill 4.2.5: Create efficient resource allocation systems specifically for FM workloads (for example, by using capacity planning for token processing requirements, utilization monitoring for prompt and completion patterns, auto-scaling configurations that are optimized for GenAI traffic patterns).
- Skill 4.2.6: Optimize FM system performance for GenAI workflows (for example, by using API call profiling for prompt-completion patterns, vector database query optimization for retrieval augmentation, latency reduction techniques specific to LLM inference, efficient service communication patterns).

Task 4.3: Implement monitoring systems for GenAI applications.

- Skill 4.3.1: Create holistic observability systems to provide complete visibility into FM application performance (for example, by using operational metrics, performance tracing, FM interaction tracing, business impact metrics with custom dashboards).
- Skill 4.3.2: Implement comprehensive GenAI monitoring systems to proactively identify issues and evaluate key performance indicators specific to FM implementations (for example, by using CloudWatch to track token usage; prompt effectiveness; hallucination rates; and response quality, anomaly detection for token burst patterns and response drift, Amazon Bedrock Model Invocation Logs to perform detailed request and response analysis, performance benchmarks, cost anomaly detection).
- Skill 4.3.3: Develop integrated observability solutions to provide actionable insights for FM applications (for example, by using operational metric dashboards, business impact visualizations, compliance monitoring, forensic traceability and audit logging, user interaction tracking, model behavior pattern tracking).
- Skill 4.3.4: Create tool performance frameworks to ensure optimal tool operation and utilization for FMs (for example, by using call pattern tracking, performance metric collection, tool calling observability and multi-agent coordination tracking, usage baselines for anomaly detection).
- Skill 4.3.5: Create vector store operational management systems to ensure optimal vector store operation and reliability for FM augmentation (for example, by using performance monitoring for vector databases, automated index optimization routines, data quality validation processes).
- Skill 4.3.6: Develop FM-specific troubleshooting frameworks to identify unique GenAI failure modes that are not present in traditional ML systems (for example, by using golden datasets to detect hallucinations, output diffing techniques to conduct response consistency analysis, reasoning path tracing to identify logical errors, specialized observability pipelines).

Content Domain 5: Testing, Validation, and Troubleshooting

Task 5.1: Implement evaluation systems for GenAI.

- Skill 5.1.1: Develop comprehensive assessment frameworks to evaluate the quality and effectiveness of FM outputs beyond traditional ML evaluation approaches (for example, by using metrics for relevance, factual accuracy, consistency, and fluency).
- Skill 5.1.2: Create systematic model evaluation systems to identify optimal configurations (for example, by using Amazon Bedrock Model Evaluations, A/B testing and canary testing of FMs,

multi-model evaluation, cost-performance analysis to measure token efficiency, latency-to-quality ratios, and business outcomes).

- Skill 5.1.3: Develop user-centered evaluation mechanisms to continuously improve FM performance based on user experience (for example, by using feedback interfaces, rating systems for model outputs, annotation workflows to assess response quality).
- Skill 5.1.4: Create systematic quality assurance processes to maintain consistent performance standards for FMs (for example, by using continuous evaluation workflows, regression testing for model outputs, automated quality gates for deployments).
- Skill 5.1.5: Develop comprehensive assessment systems to ensure thorough evaluation from multiple perspectives for FM outputs (for example, by using RAG evaluation, automated quality assessment with LLM-as-a-Judge techniques, human feedback collection interfaces).
- Skill 5.1.6: Implement retrieval quality testing to evaluate and optimize information retrieval components for FM augmentation (for example, by using relevance scoring, context matching verification, retrieval latency measurements).
- Skill 5.1.7: Develop agent performance frameworks to ensure that agents perform tasks correctly and efficiently (for example, by using task completion rate measurements, tool usage effectiveness evaluations, Amazon Bedrock Agent evaluations, reasoning quality assessment in multi-step workflows).
- Skill 5.1.8: Create comprehensive reporting systems to communicate performance metrics and insights effectively to stakeholders for FM implementations (for example, by using visualization tools, automated reporting mechanisms, model comparison visualizations).
- Skill 5.1.9: Create deployment validation systems to maintain reliability during FM updates (for example, by using synthetic user workflows, AI-specific output validation for hallucination rates and semantic drift, automated quality checks to ensure response consistency).

Task 5.2: Troubleshoot GenAI applications.

- Skill 5.2.1: Resolve content handling issues to ensure that necessary information is processed completely in FM interactions (for example, by using context window overflow diagnostics, dynamic chunking strategies, prompt design optimization, truncation-related error analysis).
- Skill 5.2.2: Diagnose and resolve FM integration issues to identify and fix API integration problems specific to GenAI services (for example, by using error logging, request validation, response analysis).

- Skill 5.2.3: Troubleshoot prompt engineering problems to improve FM response quality and consistency beyond basic prompt adjustments (for example, by using prompt testing frameworks, version comparison, systematic refinement).
- Skill 5.2.4: Troubleshoot retrieval system issues to identify and resolve problems that affect information retrieval effectiveness for FM augmentation (for example, by using model response relevance analysis, embedding quality diagnostics, drift monitoring, vectorization issue resolution, chunking and preprocessing remediation, vector search performance optimization).
- Skill 5.2.5: Troubleshoot prompt maintenance issues to continuously improve the performance of FM interactions (for example, by using template testing and CloudWatch Logs to diagnose prompt confusion, X-Ray to implement prompt observability pipelines, schema validation to detect format inconsistencies, systematic prompt refinement workflows).

Technologies and concepts that might appear on the exam

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Retrieval Augmented Generation (RAG)
- Vector databases and embeddings
- Prompt engineering and management
- Foundation model (FM) integration
- Agentic AI systems
- Responsible AI practices
- Content safety and moderation
- Model evaluation and validation
- Cost optimization for AI workloads
- Performance tuning for AI applications
- Monitoring and observability for AI systems
- Security and governance for AI applications
- API design and integration patterns
- Event-driven architectures
- Serverless computing

- Container orchestration
- Infrastructure as code (IaC)
- CI/CD for AI applications
- Hybrid cloud architectures
- Enterprise system integration

Mentions of AWS services on the exam

AWS Certification is reducing the reading load on this exam by using official short names of well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.

The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.

You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.

> **Note**
>
> Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

In-scope AWS services and features

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions:

Analytics

- Amazon Athena

- Amazon EMR
- AWS Glue
- Amazon Kinesis
- Amazon OpenSearch Service
- Amazon QuickSight
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Application Integration

- Amazon AppFlow
- AWS AppConfig
- Amazon EventBridge
- Amazon SNS
- Amazon SQS
- AWS Step Functions

Compute

- AWS App Runner
- Amazon EC2
- AWS Lambda
- AWS Lambda@Edge
- AWS Outposts
- AWS Wavelength

Containers

- Amazon ECR
- Amazon ECS
- Amazon EKS
- AWS Fargate

Customer Engagement

- Amazon Connect

Database

- Amazon Aurora
- Amazon DocumentDB
- Amazon DynamoDB
- Amazon DynamoDB Streams
- Amazon ElastiCache
- Amazon Neptune
- Amazon RDS

Developer Tools

- AWS Amplify
- AWS CDK
- AWS CLI
- AWS CloudFormation
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline
- AWS Tools and SDKs
- AWS X-Ray

Machine Learning

- Amazon Augmented AI
- Amazon Bedrock
- Amazon Bedrock AgentCore
- Amazon Bedrock Knowledge Bases

- Amazon Bedrock Prompt Management
- Amazon Bedrock Prompt Flows
- Amazon Comprehend
- Amazon Kendra
- Amazon Lex
- Amazon Q Business
- Amazon Q Business Apps
- Amazon Q Developer
- Amazon Rekognition
- Amazon SageMaker AI
- Amazon SageMaker Clarify
- Amazon SageMaker Data Wrangler
- Amazon SageMaker Ground Truth
- Amazon SageMaker JumpStart
- Amazon SageMaker Model Monitor
- Amazon SageMaker Model Registry
- Amazon SageMaker Neo
- Amazon SageMaker Processing
- Amazon SageMaker Unified Studio
- Amazon Textract
- Amazon Titan
- Amazon Transcribe

Management and Governance

- AWS Auto Scaling
- AWS Chatbot
- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Logs

- Amazon CloudWatch Synthetics
- AWS Cost Anomaly Detection
- AWS Cost Explorer
- Amazon Managed Grafana
- AWS Service Catalog
- AWS Systems Manager
- AWS Well-Architected Tool

Migration and Transfer

- AWS DataSync
- AWS Transfer Family

Networking and Content Delivery

- Amazon API Gateway
- AWS AppSync
- Amazon CloudFront
- Elastic Load Balancing (ELB)
- AWS Global Accelerator
- AWS PrivateLink
- Amazon Route 53
- Amazon VPC

Security, Identity, and Compliance

- Amazon Cognito
- AWS Encryption SDK
- IAM
- IAM Access Analyzer
- IAM Identity Center
- AWS KMS

- Amazon Macie
- AWS Secrets Manager
- AWS WAF

Storage

- Amazon EBS
- Amazon EFS
- Amazon S3
- Amazon S3 Intelligent-Tiering
- Amazon S3 Lifecycle policies
- Amazon S3 Cross-Region Replication

Out-of-scope AWS services and features

Topics

- Application Integration
- Analytics
- Blockchain
- Business Applications
- Cloud Financial Management
- Compute
- Containers
- Customer Engagement
- Database
- Developer Tools
- End User Computing
- Frontend Web and Mobile
- Game Development
- Internet of Things (IoT)
- Management and Governance

- Machine Learning
- Media Services
- Migration and Transfer
- Networking and Content Delivery
- Quantum Technologies
- Robotics
- Satellite

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Application Integration

- Amazon MQ

Analytics

- AWS Clean Rooms
- AWS Data Exchange
- Amazon DataZone
- Amazon FinSpace

Blockchain

- Amazon Managed Blockchain (AMB)

Business Applications

- Alexa for Business
- Amazon Chime
- AWS Wickr
- Amazon WorkDocs

- Amazon WorkMail

Cloud Financial Management

- AWS Budgets
- AWS Cost and Usage Report
- Reserved Instance reports
- AWS Savings Plans

Compute

- AWS Batch
- Amazon EC2 Image Builder
- Amazon ECS Anywhere
- Amazon EKS Anywhere
- AWS Elastic Beanstalk
- Amazon Lightsail
- AWS Local Zones
- AWS Serverless Application Repository

Containers

- AWS App2Container
- AWS Copilot
- Red Hat OpenShift Service on AWS (ROSA)

Customer Engagement

- Amazon SES

Database

- Amazon Keyspaces

- Amazon Quantum Ledger Database (Amazon QLDB)
- Amazon Redshift
- Amazon Timestream

Developer Tools

- AWS Cloud9
- AWS CloudShell
- Amazon CodeGuru
- AWS CodeStar
- Amazon Corretto

End User Computing

- Amazon AppStream 2.0
- Amazon WorkLink
- Amazon WorkSpaces
- Amazon WorkSpaces Web

Frontend Web and Mobile

- AWS Device Farm
- Amazon Location Service
- Amazon Pinpoint

Game Development

- Amazon GameLift
- Amazon Lumberyard

Internet of Things (IoT)

- AWS IoT 1-Click

- AWS IoT Analytics
- AWS IoT Button
- AWS IoT Core
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT Events
- AWS IoT FleetWise
- AWS IoT Greengrass
- AWS IoT SiteWise
- AWS IoT TwinMaker

Management and Governance

- AWS Console Mobile Application
- AWS Health Dashboard
- AWS License Manager
- AWS Proton
- AWS Trusted Advisor

Machine Learning

- AWS DeepComposer
- AWS DeepRacer
- Amazon DevOps Guru
- Amazon Forecast
- Amazon Fraud Detector
- Amazon HealthLake
- Amazon Lookout for Equipment
- Amazon Lookout for Metrics
- Amazon Lookout for Vision

- Amazon Monitron
- AWS Panorama

Media Services

- Amazon Elastic Transcoder
- AWS Elemental MediaConnect
- AWS Elemental MediaConvert
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Elemental MediaTailor
- Amazon Interactive Video Service
- Amazon Kinesis Video Streams
- Amazon Nimble Studio

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- CloudEndure Migration
- AWS Migration Hub
- AWS Snow Family

Networking and Content Delivery

- AWS App Mesh
- AWS Cloud Map
- AWS Direct Connect
- AWS Private 5G
- AWS Transit Gateway

- AWS VPN

Quantum Technologies

- Amazon Braket

Robotics

- AWS RoboMaker

Satellite

- AWS Ground Station

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified Solutions Architect - Professional (SAP-C02)

The AWS Certified Solutions Architect - Professional (SAP-C02) exam is intended for individuals who perform a solutions architect role. The exam validates a candidate's advanced technical skills and experience in designing optimized AWS solutions that are based on the AWS Well-Architected Framework.

Topics

Introduction

The exam also validates a candidate's ability to complete the following tasks within the scope of the AWS Well-Architected Framework:

- Design for organizational complexity.
- Design for new solutions.
- Continuously improve existing solutions.
- Accelerate workload migration and modernization.

Target candidate description

The target candidate has 2 or more years of experience in using AWS services to design and implement cloud solutions. This candidate has the ability to evaluate cloud application requirements and make architectural recommendations for deployment of applications on AWS. This candidate also can provide expert guidance about architectural design that extends across multiple applications and projects within a complex organization.

Refer to the Appendix for a list of technologies and concepts that might appear on the exam, a list of in-scope AWS services and features, and a list of out-of-scope AWS services and features.

Job tasks and knowledge that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform and knowledge that the target candidate is not expected to have. This list is non-exhaustive. These tasks and knowledge are out of scope for the exam:

- Frontend development for mobile apps
- 12-factor app methodology

- In-depth knowledge of operating systems

Exam content

Response types

The exam contains one or more of the following question types:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors).
- **Multiple response:** Has two or more correct responses out of five or more response options. You must select all the correct responses to receive credit for the question.

Unanswered questions are scored as incorrect. There is no penalty for guessing. The exam includes 65 questions that affect your score.

Unscored content

The exam includes 10 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate them for future use as scored questions. The unscored questions are not identified on the exam.

Exam results

The AWS Certified Solutions Architect - Professional (SAP-C02) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 750. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and tasks for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Design Solutions for Organizational Complexity (26% of scored content)
- Content Domain 2: Design for New Solutions (29% of scored content)
- Content Domain 3: Continuous Improvement for Existing Solutions (25% of scored content)
- Content Domain 4: Accelerate Workload Migration and Modernization (20% of scored content)

Content Domain 1: Design Solutions for Organizational Complexity

Tasks

- Task 1.1: Architect network connectivity strategies.
- Task 1.2: Prescribe security controls.
- Task 1.3: Design reliable and resilient architectures.
- Task 1.4: Design a multi-account AWS environment.
- Task 1.5: Determine cost optimization and visibility strategies.

Task 1.1: Architect network connectivity strategies.

Knowledge of:

- AWS Global Infrastructure
- AWS networking concepts (for example, Amazon Virtual Private Cloud [Amazon VPC], AWS Direct Connect, AWS VPN, transitive routing, AWS container services)
- Hybrid DNS concepts (for example, Amazon Route 53 Resolver, on-premises DNS integration)
- Network segmentation (for example, subnetting, IP addressing, connectivity among VPCs)
- Network traffic monitoring

Skills in:

- Evaluating connectivity options for multiple VPCs
- Evaluating connectivity options for on-premises, co-location, and cloud integration
- Selecting AWS Regions and Availability Zones based on network and latency requirements
- Troubleshooting traffic flows by using AWS tools
- Using service endpoints for service integrations

Task 1.2: Prescribe security controls.

Knowledge of:

- AWS Identity and Access Management (IAM) and AWS IAM Identity Center
- Route tables, security groups, and network ACLs
- Encryption keys and certificate management (for example, AWS Key Management Service [AWS KMS], AWS Certificate Manager [ACM])
- AWS security, identity, and compliance tools (for example, AWS CloudTrail, AWS Identity and Access Management Access Analyzer, AWS Security Hub, Amazon Inspector)

Skills in:

- Evaluating cross-account access management
- Integrating with third-party identity providers
- Deploying encryption strategies for data at rest and data in transit
- Developing a strategy for centralized security event notifications and auditing

Task 1.3: Design reliable and resilient architectures.

Knowledge of:

- Recovery time objectives (RTOs) and recovery point objectives (RPOs)
- Disaster recovery strategies (for example, using AWS Elastic Disaster Recovery, pilot light, warm standby, and multi-site)
- Data backup and restoration

Skills in:

- Designing disaster recovery solutions based on RTO and RPO requirements
- Implementing architectures to automatically recover from failure
- Developing the optimal architecture by considering scale-up and scale-out options
- Designing an effective backup and restoration strategy

Task 1.4: Design a multi-account AWS environment.

Knowledge of:

- AWS Organizations and AWS Control Tower
- Multi-account event notifications
- AWS resource sharing across environments

Skills in:

- Evaluating the most appropriate account structure for organizational requirements
- Recommending a strategy for central logging and event notifications
- Developing a multi-account governance model

Task 1.5: Determine cost optimization and visibility strategies.

Knowledge of:

- AWS cost and usage monitoring tools (for example, AWS Trusted Advisor, AWS Pricing Calculator, AWS Cost Explorer, AWS Budgets)
- AWS purchasing options (for example, Reserved Instances, Savings Plans, Spot Instances)
- AWS rightsizing visibility tools (for example, AWS Compute Optimizer, Amazon Simple Storage Service [Amazon S3] Storage Lens)

Skills in:

- Monitoring cost and usage with AWS tools
- Developing an effective tagging strategy that maps costs to business units
- Understanding how purchasing options affect cost and performance

Content Domain 2: Design for New Solutions

Tasks

Task 2.1: Design a deployment strategy to meet business requirements

Knowledge of:

- Infrastructure as code (IaC) (for example, AWS CloudFormation)
- Continuous integration and continuous delivery (CI/CD)
- Change management processes
- Configuration management tools (for example, AWS Systems Manager)

Skills in:

- Determining an application or upgrade path for new services and features
- Selecting services to develop deployment strategies and implement appropriate rollback mechanisms
- Adopting managed services as needed to reduce infrastructure provisioning and patching overhead
- Making advanced technologies accessible by delegating complex development and deployment tasks to AWS

Task 2.2: Design a solution to ensure business continuity

Knowledge of:

- AWS Global Infrastructure

- AWS networking concepts (for example, Amazon Route 53, routing methods)
- RTOs and RPOs
- Disaster recovery scenarios (for example, backup and restore, pilot light, warm standby, multi-site)
- Disaster recovery solutions on AWS

Skills in:

- Configuring disaster recovery solutions
- Configuring data and database replication
- Performing disaster recovery testing
- Architecting a backup solution that is automated, is cost-effective, and supports business continuity across multiple Availability Zones or AWS Regions
- Designing an architecture that provides application and infrastructure availability in the event of a disruption
- Using processes and components for centralized monitoring to proactively recover from system failures

Task 2.3: Determine security controls based on requirements

Knowledge of:

- IAM
- Route tables, security groups, and network ACLs
- Encryption options for data at rest and data in transit
- AWS service endpoints
- Credential management services
- AWS managed security services (for example, AWS Shield, AWS WAF, Amazon GuardDuty, AWS Security Hub)

Skills in:

- Specifying IAM users and IAM roles that adhere to the principle of least privilege access

- Specifying inbound and outbound network flows by using security group rules and network ACL rules
- Developing attack mitigation strategies for large-scale web applications
- Developing encryption strategies for data at rest and data in transit
- Specifying service endpoints for service integrations
- Developing strategies for patch management to remain compliant with organizational standards

Task 2.4: Design a strategy to meet reliability requirements

Knowledge of:

- AWS Global Infrastructure
- AWS storage services and replication strategies (for example Amazon S3, Amazon RDS, Amazon ElastiCache)
- Multi-AZ and multi-Region architectures
- Auto scaling policies and events
- Application integration (for example, Amazon SNS, Amazon SQS, AWS Step Functions)
- Service quotas and limits

Skills in:

- Designing highly available application environments based on business requirements
- Using advanced techniques to design for failure and ensure seamless system recoverability
- Implementing loosely coupled dependencies
- Operating and maintaining high-availability architectures (for example, application failovers, database failovers)
- Using AWS managed services for high availability
- Implementing DNS routing policies (for example, Route 53 latency-based routing, geolocation routing, simple routing)

Task 2.5: Design a solution to meet performance objectives

Knowledge of:

- Performance monitoring technologies
- Storage options on AWS
- Instance families and use cases
- Purpose-built databases

Skills in:

- Designing large-scale application architectures for a variety of access patterns
- Designing an elastic architecture based on business objectives
- Applying design patterns to meet performance objectives with caching, buffering, and replicas
- Developing a process methodology for selecting purpose-built services for required tasks
- Designing a rightsizing strategy

Task 2.6: Determine a cost optimization strategy to meet solution goals and objectives

Knowledge of:

- AWS cost and usage monitoring tools (for example, AWS Cost Explorer, AWS Trusted Advisor, AWS Pricing Calculator)
- Pricing models (for example, Reserved Instances, AWS Savings Plans)
- Storage tiering
- Data transfer costs
- AWS managed service offerings

Skills in:

- Identifying opportunities to select and rightsize infrastructure for cost-effective resources
- Identifying appropriate pricing models
- Performing data transfer modeling and selecting services to reduce data transfer costs
- Developing a strategy and implementing controls for expenditure and usage awareness

Content Domain 3: Continuous Improvement for Existing Solutions

Tasks

Task 3.1: Determine a strategy to improve overall operational excellence.

Knowledge of:

- Alerting and automatic remediation strategies
- Disaster recovery planning
- Monitoring and logging solutions (for example, Amazon CloudWatch)
- CI/CD pipelines and deployment strategies (for example, blue/green, all-at-once, rolling)
- Configuration management tools (for example, AWS Systems Manager)

Skills in:

- Determining the most appropriate logging and monitoring strategy
- Evaluating current deployment processes for improvement opportunities
- Prioritizing opportunities for automation within a solution stack
- Recommending the appropriate AWS solution to enable configuration management automation
- Engineering failure scenario activities to support and exercise an understanding of recovery actions

Task 3.2: Determine a strategy to improve security.

Knowledge of:

- Data retention, data sensitivity, and data regulatory requirements
- Automated monitoring and remediation strategies (for example, AWS Config rules)

- Secrets management (for example, Systems Manager, AWS Secrets Manager)
- Principle of least privilege access
- Security-specific AWS solutions
- Patching practices
- Backup practices and methods

Skills in:

- Evaluating a strategy for the secure management of secrets and credentials
- Auditing an environment for least privilege access
- Reviewing implemented solutions to ensure security at every layer
- Reviewing comprehensive traceability of users and services
- Prioritizing automated responses to the detection of vulnerabilities
- Designing and implementing a patch and update process
- Designing and implementing a backup process
- Employing remediation techniques

Task 3.3: Determine a strategy to improve performance.

Knowledge of:

- High-performing systems architectures (for example, auto scaling, instance fleets, placement groups)
- Global service offerings (for example, AWS Global Accelerator, Amazon CloudFront, edge computing services)
- Monitoring tool sets and services (for example, CloudWatch)
- Service level agreements (SLAs) and key performance indicators (KPIs)

Skills in:

- Translating business requirements to measurable metrics
- Testing potential remediation solutions and making recommendations
- Proposing opportunities for the adoption of new technologies and managed services

- Assessing solutions and applying rightsizing based on requirements
- Identifying and examining performance bottlenecks

Task 3.4: Determine a strategy to improve reliability.

Knowledge of:

- AWS Global Infrastructure
- Data replication methods
- Scaling methodologies (for example, load balancing, auto scaling)
- High availability and resiliency
- Disaster recovery methods and tools
- Service quotas and limits

Skills in:

- Understanding application growth and usage trends
- Evaluating existing architecture to determine areas that are not sufficiently reliable
- Remediating single points of failure
- Enabling data replication, self-healing, and elastic features and services

Task 3.5: Identify opportunities for cost optimizations.

Knowledge of:

- Cost-conscious architecture choices (for example, using Spot Instances, scaling policies, and rightsizing resources)
- Price model adoptions (for example, Reserved Instances, AWS Savings Plans)
- Networking and data transfer costs
- Cost management, alerting, and reporting

Skills in:

- Analyzing usage reports to identify underutilized and overutilized resources

- Using AWS solutions to identify unused resources
- Designing billing alarms based on expected usage patterns
- Investigating AWS Cost and Usage Reports at a granular level
- Using tagging for cost allocation and reporting

Content Domain 4: Accelerate Workload Migration and Modernization

Tasks

Task 4.1: Select existing workloads and processes for potential migration.

Knowledge of:

- Migration assessment and tracking tools (for example, AWS Migration Hub)
- Portfolio assessment
- Asset planning
- Prioritization and migration of workloads (for example, wave planning)

Skills in:

- Completing an application migration assessment
- Evaluating applications according to the seven common migration strategies (7Rs)
- Evaluating total cost of ownership (TCO)

Task 4.2: Determine the optimal migration approach for existing workloads.

Knowledge of:

- Data migration options and tools (for example, AWS DataSync, AWS Transfer Family, AWS Snow Family, Amazon S3 Transfer Acceleration)

- Application migration tools (for example, AWS Application Discovery Service, AWS Application Migration Service)
- AWS networking services and DNS (for example, AWS Direct Connect, AWS Site-to-Site VPN, Amazon Route 53)
- Identity services (for example, AWS IAM Identity Center, AWS Directory Service)
- Database migration tools (for example, AWS DMS, AWS SCT)
- Governance tools (for example, AWS Control Tower, AWS Organizations)

Skills in:

- Selecting the appropriate database transfer mechanism
- Selecting the appropriate application transfer mechanism
- Selecting the appropriate data transfer service and migration strategy
- Applying the appropriate security methods to migration tools
- Selecting the appropriate governance model

Task 4.3: Determine a new architecture for existing workloads.

Knowledge of:

- Compute services (for example, Amazon EC2, AWS Elastic Beanstalk)
- Containers (for example, Amazon ECS, Amazon EKS, AWS Fargate, Amazon ECR)
- AWS storage services (for example, Amazon EBS, Amazon EFS, Amazon FSx, Amazon S3, AWS Storage Gateway Volume Gateway)
- Databases (for example, Amazon DynamoDB, Amazon OpenSearch Service, Amazon RDS, self-managed databases on Amazon EC2)

Skills in:

- Selecting the appropriate compute platform
- Selecting the appropriate container hosting platform
- Selecting the appropriate storage service
- Selecting the appropriate database platform

Task 4.4: Determine opportunities for modernization and enhancements.

Knowledge of:

- Serverless compute offerings (for example, AWS Lambda)
- Containers (for example, Amazon ECS, Amazon EKS, Fargate)
- AWS storage services (for example, Amazon S3, Amazon EFS)
- Purpose-built databases (for example, DynamoDB, Amazon Aurora Serverless, Amazon ElastiCache)
- Integration services (for example, Amazon SQS, Amazon SNS, Amazon EventBridge, AWS Step Functions)

Skills in:

- Identifying opportunities to decouple application components
- Identifying opportunities for serverless solutions
- Selecting the appropriate service for containers
- Identifying opportunities for purpose-built databases
- Selecting the appropriate application integration service

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- Technologies and Concepts
- In-Scope AWS Services
- Out-of-Scope AWS Services
- Mentions of AWS Services on the Exam

Survey

How useful was this exam guide? Let us know by taking our survey.

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Compute
- Cost management
- Database
- Disaster recovery
- High availability
- Management and governance
- Microservices and component decoupling
- Migration and data transfer
- Networking, connectivity, and content delivery
- Security
- Serverless design principles
- Storage

In-scope AWS services and features

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the AWS Certified Solutions Architect - Professional (SAP-C02) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

- Analytics
- Application Integration
- Blockchain
- Business Applications
- Cloud Financial Management
- Compute

- Containers
- Database
- Developer Tools
- End User Computing
- Frontend Web and Mobile
- Internet of Things (IoT)
- Machine Learning
- Management and Governance
- Migration and Transfer
- Networking and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- Amazon Athena
- AWS Data Exchange
- Amazon Data Firehose
- Amazon EMR
- AWS Glue
- Amazon Kinesis Data Streams
- AWS Lake Formation
- Amazon Managed Service for Apache Flink
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Amazon OpenSearch Service
- Amazon QuickSight

Application Integration

- Amazon AppFlow
- AWS AppSync
- Amazon EventBridge

- Amazon MQ
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Blockchain

- Amazon Managed Blockchain

Business Applications

- Amazon Simple Email Service (Amazon SES)

Cloud Financial Management

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

Compute

- AWS App Runner
- AWS Auto Scaling
- AWS Batch
- AWS Elastic Beanstalk
- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon EC2 Auto Scaling
- AWS Fargate
- AWS Lambda
- Amazon Lightsail
- AWS Outposts
- AWS Wavelength

Containers

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon Elastic Kubernetes Service (Amazon EKS)
- AWS Fargate

Database

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

Developer Tools

- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CodePipeline
- AWS X-Ray

End User Computing

- Amazon AppStream 2.0

- Amazon WorkSpaces

Frontend Web and Mobile

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

Internet of Things (IoT)

- AWS IoT Core
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT Events
- AWS IoT Greengrass
- AWS IoT SiteWise
- AWS IoT Things Graph
- AWS IoT 1-Click

Machine Learning

- Amazon Comprehend
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Personalize
- Amazon Polly
- Amazon Rekognition
- Amazon SageMaker AI (previously known as Amazon SageMaker)
- Amazon Textract
- Amazon Transcribe

- Amazon Translate

Management and Governance

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

Migration and Transfer

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Database Migration Service (AWS DMS)
- AWS DataSync

- AWS Migration Hub
- AWS Schema Conversion Tool (AWS SCT)
- AWS Snow Family
- AWS Transfer Family

Networking and Content Delivery

- Amazon API Gateway
- AWS App Mesh
- AWS Client VPN
- Amazon CloudFront
- AWS Cloud Map
- AWS Direct Connect
- Elastic Load Balancing
- AWS Global Accelerator
- AWS PrivateLink
- Amazon Route 53
- AWS Site-to-Site VPN
- AWS Transit Gateway
- Amazon VPC
- AWS VPN

Security, Identity, and Compliance

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager

- Amazon GuardDuty
- AWS IAM Identity Center
- 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 Security Token Service (AWS STS)
- AWS Shield
- AWS WAF

Storage

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- AWS Elastic Disaster Recovery
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx (for all types)
- Amazon Simple Storage Service (Amazon S3)
- Amazon S3 Glacier
- AWS Storage Gateway

Out-of-scope AWS services and features

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Topics

- Game Tech

Game Tech

- Amazon GameLift

Mentions of AWS Services on the Exam

AWS Certification is reducing the reading load on this exam by using official short names for well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.

The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.

You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.

Note: Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

AWS Certified DevOps Engineer - Professional (DOP-C02)

The AWS Certified DevOps Engineer - Professional (DOP-C02) exam is intended for individuals who perform a DevOps engineer role. The exam validates a candidate's technical expertise in provisioning, operating, and managing distributed systems and services on AWS.

Topics

Introduction

The AWS Certified DevOps Engineer - Professional (DOP-C02) exam is intended for individuals who perform a DevOps engineer role. The exam validates a candidate's technical expertise in provisioning, operating, and managing distributed systems and services on AWS.

The exam also validates a candidate's ability to complete the following tasks:

- Implement and manage continuous delivery systems and methodologies on AWS.
- Implement and automate security controls, governance processes, and compliance validation.
- Define and deploy monitoring, metrics, and logging systems on AWS.
- Implement systems that are highly available, scalable, and self-healing on AWS.
- Design, manage, and maintain tools to automate operational processes.

Target Candidate Description

The target candidate should have 2 or more years of experience in provisioning, operating, and managing AWS environments. The target candidate also has experience with the software development lifecycle and programming and/or scripting.

Recommended General IT Knowledge and Experience

The target candidate should have the following experience:

- Experience in building highly automated infrastructure
- Experience in administering operating systems

- Experience with modern development and operations processes and methodologies

Recommended AWS Knowledge and Experience

The target candidate should have experience in securing AWS infrastructure.

Job Tasks That Are Out of Scope for the Target Candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Possess advanced networking knowledge (for example, advanced routing algorithms, failover techniques).
- Provide deep-level security recommendations to developers.
- Design, query, and optimize the performance of databases.
- Develop full-stack application code.

Exam Content

Response Types

There are two types of questions on the exam:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)
- **Multiple response:** Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 65 questions that affect your score.

Unscored Content

The exam includes 10 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam Results

The AWS Certified DevOps Engineer - Professional (DOP-C02) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 750. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content Outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: SDLC Automation (22% of scored content)
- Content Domain 2: Configuration Management and IaC (17% of scored content)
- Content Domain 3: Resilient Cloud Solutions (15% of scored content)
- Content Domain 4: Monitoring and Logging (15% of scored content)
- Content Domain 5: Incident and Event Response (14% of scored content)
- Content Domain 6: Security and Compliance (17% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- Technologies and Concepts
- In-Scope AWS Services

Content Domain 1: SDLC Automation

Task Statement 1.1: Implement CI/CD pipelines.

Knowledge of:

- Software development lifecycle (SDLC) concepts, phases, and models
- Pipeline deployment patterns for single- and multi-account environments

Skills in:

- Configuring code, image, and artifact repositories
- Using version control to integrate pipelines with application environments
- Setting up build processes (for example, AWS CodeBuild)
- Managing build and deployment secrets (for example, AWS Secrets Manager, AWS Systems Manager Parameter Store)
- Determining appropriate deployment strategies (for example, AWS CodeDeploy)

Task Statement 1.2: Integrate automated testing into CI/CD pipelines.

Knowledge of:

- Different types of tests (for example, unit tests, integration tests, acceptance tests, user interface tests, security scans)
- Reasonable use of different types of tests at different stages of the CI/CD pipeline

Skills in:

- Running builds or tests when generating pull requests or code merges (for example, CodeBuild)
- Running load/stress tests, performance benchmarking, and application testing at scale
- Measuring application health based on application exit codes

- Automating unit tests and code coverage
- Invoking AWS services in a pipeline for testing

Task Statement 1.3: Build and manage artifacts.

Knowledge of:

- Artifact use cases and secure management
- Methods to create and generate artifacts
- Artifact lifecycle considerations

Skills in:

- Creating and configuring artifact repositories (for example, AWS CodeArtifact, Amazon S3, Amazon Elastic Container Registry [Amazon ECR])
- Configuring build tools for generating artifacts (for example, CodeBuild, AWS Lambda)
- Automating Amazon EC2 instance and container image build processes (for example, EC2 Image Builder)

Task Statement 1.4: Implement deployment strategies for instance, container, and serverless environments.

Knowledge of:

- Deployment methodologies for various platforms (for example, Amazon EC2, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS], Lambda)
- Application storage patterns (for example, Amazon Elastic File System [Amazon EFS], Amazon S3, Amazon Elastic Block Store [Amazon EBS])
- Mutable deployment patterns in contrast to immutable deployment patterns
- Tools and services available for distributing code (for example, CodeDeploy, Image Builder)

Skills in:

- Configuring security permissions to allow access to artifact repositories (for example, AWS Identity and Access Management [IAM], CodeArtifact)

- Configuring deployment agents (for example, CodeDeploy agent)
- Troubleshooting deployment issues
- Using different deployment methods (for example, blue/green, canary)

Content Domain 2: Configuration Management and IaC

Task Statement 2.1: Define cloud infrastructure and reusable components to provision and manage systems throughout their lifecycle.

Knowledge of:

- Infrastructure as code (IaC) options and tools for AWS
- Change management processes for IaC-based platforms
- Configuration management services and strategies

Skills in:

- Composing and deploying IaC templates (for example, AWS Serverless Application Model [AWS SAM], AWS CloudFormation, AWS Cloud Development Kit [AWS CDK])
- Applying CloudFormation stack sets across multiple accounts and AWS Regions
- Determining optimal configuration management services (for example, AWS OpsWorks, AWS Systems Manager, AWS Config, AWS AppConfig)
- Implementing infrastructure patterns, governance controls, and security standards into reusable IaC templates (for example, AWS Service Catalog, CloudFormation modules, AWS CDK)

Task Statement 2.2: Deploy automation to create, onboard, and secure AWS accounts in a multi-account or multi-Region environment.

Knowledge of:

- AWS account structures, best practices, and related AWS services

Skills in:

- Standardizing and automating account provisioning and configuration

- Creating, consolidating, and centrally managing accounts (for example, AWS Organizations, AWS Control Tower)
- Applying IAM solutions for multi-account and complex organization structures (for example, SCPs, assuming roles)
- Implementing and developing governance and security controls at scale (AWS Config, AWS Control Tower, AWS Security Hub, Amazon Detective, Amazon GuardDuty, Service Catalog, SCPs)

Task Statement 2.3: Design and build automated solutions for complex tasks and large-scale environments.

Knowledge of:

- AWS services and solutions to automate tasks and processes
- Methods and strategies to interact with the AWS software-defined infrastructure

Skills in:

- Automating system inventory, configuration, and patch management (for example, Systems Manager, AWS Config)
- Developing AWS Lambda function automations for complex scenarios (for example, AWS SDKs, Lambda, AWS Step Functions)
- Automating the configuration of software applications to the desired state (for example, OpsWorks, Systems Manager State Manager)
- Maintaining software compliance (for example, Systems Manager)

Content Domain 3: Resilient Cloud Solutions

Task Statement 3.1: Implement highly available solutions to meet resilience and business requirements.

Knowledge of:

- Multi-AZ and multi-Region deployments (for example, compute layer, data layer)
- SLAs
- Replication and failover methods for stateful services

- Techniques to achieve high availability (for example, Multi-AZ, multi-Region)

Skills in:

- Translating business requirements into technical resiliency needs
- Identifying and remediating single points of failure in existing workloads
- Enabling cross-Region solutions where available (for example, Amazon DynamoDB, Amazon RDS, Amazon Route 53, Amazon S3, Amazon CloudFront)
- Configuring load balancing to support cross-AZ services
- Configuring applications and related services to support multiple Availability Zones and AWS Regions while minimizing downtime

Task Statement 3.2: Implement solutions that are scalable to meet business requirements.

Knowledge of:

- Appropriate metrics for scaling services
- Loosely coupled and distributed architectures
- Serverless architectures
- Container platforms

Skills in:

- Identifying and remediating scaling issues
- Identifying and implementing appropriate auto scaling, load balancing, and caching solutions
- Deploying container-based applications (for example, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS])
- Deploying workloads in multiple Regions for global scalability
- Configuring serverless applications (for example, Amazon API Gateway, AWS Lambda, AWS Fargate)

Task Statement 3.3: Implement automated recovery processes to meet RTO and RPO requirements.

Knowledge of:

- Disaster recovery concepts (for example, RTO, RPO)
- AWS Backup and recovery strategies (for example, pilot light, warm standby)
- Recovery procedures

Skills in:

- Testing failover of Multi-AZ and multi-Region workloads (for example, Amazon RDS, Amazon Aurora, Route 53, CloudFront)
- Identifying and implementing appropriate cross-Region AWS Backup and recovery strategies (for example, AWS Backup, Amazon S3, AWS Systems Manager)
- Configuring a load balancer to recover from backend failure

Content Domain 4: Monitoring and Logging

Task Statement 4.1: Configure the collection, aggregation, and storage of logs and metrics.

Knowledge of:

- How to monitor applications and infrastructure
- Amazon CloudWatch metrics (for example, namespaces, metrics, dimensions, and resolution)
- Real-time log ingestion
- Encryption options for at-rest and in-transit logs and metrics (for example, client-side and server-side, AWS Key Management Service [AWS KMS])
- Security configurations (for example, IAM roles and permissions to allow for log collection)

Skills in:

- Securely storing and managing logs
- Creating CloudWatch metrics from log events by using metric filters

- Creating CloudWatch metric streams (for example, Amazon S3 or Amazon Kinesis Data Firehose options)
- Collecting custom metrics (for example, using the CloudWatch agent)
- Managing log storage lifecycles (for example, Amazon S3 lifecycles, CloudWatch log group retention)
- Processing log data by using CloudWatch log subscriptions (for example, Amazon Kinesis, AWS Lambda, Amazon OpenSearch Service)
- Searching log data by using filter and pattern syntax or Amazon CloudWatch Logs Insights
- Configuring encryption of log data (for example, AWS KMS)

Task Statement 4.2: Audit, monitor, and analyze logs and metrics to detect issues.

Knowledge of:

- Anomaly detection alarms (for example, CloudWatch anomaly detection)
- Common CloudWatch metrics and logs (for example, CPU utilization with Amazon EC2, queue length with Amazon RDS, 5xx errors with an Application Load Balancer [ALB])
- Amazon Inspector and common assessment templates
- AWS Config rules
- AWS CloudTrail log events

Skills in:

- Building CloudWatch dashboards and Amazon QuickSight visualizations
- Associating CloudWatch alarms with CloudWatch metrics (standard and custom)
- Configuring AWS X-Ray for different services (for example, containers, Amazon API Gateway, Lambda)
- Analyzing real-time log streams (for example, using Amazon Kinesis Data Streams)
- Analyzing logs with AWS services (for example, Amazon Athena, CloudWatch Logs Insights)

Task Statement 4.3: Automate monitoring and event management of complex environments.

Knowledge of:

- Event-driven, asynchronous design patterns (for example, S3 Event Notifications or Amazon EventBridge events to Amazon Simple Notification Service [Amazon SNS] or Lambda)
- Capabilities of auto scaling for a variety of AWS services (for example, EC2 Auto Scaling groups, RDS storage auto scaling, Amazon DynamoDB, Amazon Elastic Container Service [Amazon ECS] capacity provider, Amazon Elastic Kubernetes Service [Amazon EKS] autoscalers)
- Alert notification and action capabilities (for example, CloudWatch alarms to Amazon SNS, Lambda, EC2 automatic recovery)
- Health check capabilities in AWS services (for example, ALB target groups, Amazon Route 53)

Skills in:

- Configuring solutions for auto scaling (for example, DynamoDB, EC2 Auto Scaling groups, RDS storage auto scaling, ECS capacity provider)
- Creating CloudWatch custom metrics and metric filters, alarms, and notifications (for example, Amazon SNS, Lambda)
- Configuring S3 events to process log files (for example, by using Lambda) and deliver log files to another destination (for example, OpenSearch Service, CloudWatch Logs)
- Configuring EventBridge to send notifications based on a particular event pattern
- Installing and configuring agents on EC2 instances (for example, AWS Systems Manager Agent [SSM Agent], CloudWatch agent)
- Configuring AWS Config rules to remediate issues
- Configuring health checks (for example, Route 53, ALB)

Content Domain 5: Incident and Event Response

Task Statement 5.1: Manage event sources to process, notify, and take action in response to events.

Knowledge of:

- AWS services that generate, capture, and process events (for example, AWS Health, Amazon EventBridge, AWS CloudTrail)
- Event-driven architectures (for example, fan out, event streaming, queuing)

Skills in:

- Integrating AWS event sources (for example, AWS Health, EventBridge, CloudTrail)
- Building event processing workflows (for example, Amazon Simple Queue Service [Amazon SQS], Amazon Kinesis, Amazon Simple Notification Service [Amazon SNS], AWS Lambda, AWS Step Functions)

Task Statement 5.2: Implement configuration changes in response to events.

Knowledge of:

- Fleet management services (for example, AWS Systems Manager, AWS Auto Scaling)
- Configuration management services (for example, AWS Config)

Skills in:

- Applying configuration changes to systems
- Modifying infrastructure configurations in response to events
- Remediating a non-desired system state

Task Statement 5.3: Troubleshoot system and application failures.

Knowledge of:

- AWS metrics and logging services (for example, Amazon CloudWatch, AWS X-Ray)

- AWS service health services (for example, AWS Health, CloudWatch, Systems Manager OpsCenter)
- Root cause analysis

Skills in:

- Analyzing failed deployments (for example, AWS CodePipeline, AWS CodeBuild, AWS CodeDeploy, AWS CloudFormation, CloudWatch synthetic monitoring)
- Analyzing incidents regarding failed processes (for example, auto scaling, Amazon Elastic Container Service [Amazon ECS], Amazon Elastic Kubernetes Service [Amazon EKS])

Content Domain 6: Security and Compliance

Task Statement 6.1: Implement techniques for identity and access management at scale.

Knowledge of:

- Appropriate usage of different IAM entities for human and machine access (for example, users, groups, roles, identity providers, identity-based policies, resource-based policies, session policies)
- Identity federation techniques (for example, using IAM identity providers and AWS IAM Identity Center)
- Permission management delegation by using IAM permissions boundaries
- Organizational SCPs

Skills in:

- Designing policies to enforce least privilege access
- Implementing role-based and attribute-based access control patterns
- Automating credential rotation for machine identities (for example, AWS Secrets Manager)
- Managing permissions to control access to human and machine identities (for example, enabling multi-factor authentication [MFA], AWS Security Token Service [AWS STS], IAM profiles)

Task Statement 6.2: Apply automation for security controls and data protection.

Knowledge of:

- Network security components (for example, security groups, network ACLs, routing, AWS Network Firewall, AWS WAF, AWS Shield)
- Certificates and public key infrastructure (PKI)
- Data management (for example, data classification, encryption, key management, access controls)

Skills in:

- Automating the application of security controls in multi-account and multi-Region environments (for example, AWS Security Hub, AWS Organizations, AWS Control Tower, AWS Systems Manager)
- Combining security controls to apply defense in depth (for example, AWS Certificate Manager [ACM], AWS WAF, AWS Config, AWS Config rules, Security Hub, Amazon GuardDuty, security groups, network ACLs, Amazon Detective, Network Firewall)
- Automating the discovery of sensitive data at scale (for example, Amazon Macie)
- Encrypting data in transit and data at rest (for example, AWS Key Management Service [AWS KMS], AWS CloudHSM, ACM)

Task Statement 6.3: Implement security monitoring and auditing solutions.

Knowledge of:

- Security auditing services and features (for example, AWS CloudTrail, AWS Config, VPC Flow Logs, AWS CloudFormation drift detection)
- AWS services for identifying security vulnerabilities and events (for example, GuardDuty, Amazon Inspector, IAM Access Analyzer, AWS Config)
- Common cloud security threats (for example, insecure web traffic, exposed AWS access keys, S3 buckets with public access enabled or encryption disabled)

Skills in:

- Implementing robust security auditing
- Configuring alerting based on unexpected or anomalous security events

- Configuring service and application logging (for example, CloudTrail, Amazon CloudWatch Logs)
- Analyzing logs, metrics, and security findings

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Application deployment
- Application integration
- Application pipelines
- Automation
- Code repository best practices
- Cost optimization
- Deployment requirements
- Hybrid deployments
- IAM policies
- Metrics, monitoring, alarms, and logging
- Network ACL and security group design and implementation
- Operational best practices
- Rollback procedures

In-scope AWS services and features

The following list contains AWS services and features that are in scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions:

Analytics:

- Amazon Athena
- Amazon EMR
- Amazon Kinesis Data Firehose

- Amazon Kinesis Data Streams
- Amazon OpenSearch Service
- Amazon QuickSight

Application Integration:

- Amazon AppFlow
- Amazon EventBridge

Compute:

- AWS App Runner
- Amazon EC2
- Amazon EC2 Auto Scaling
- EC2 Image Builder
- AWS Elastic Beanstalk
- AWS Serverless Application Repository

Containers:

- AWS App2Container
- AWS Copilot
- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Amazon EKS Distro
- AWS Fargate
- Red Hat OpenShift Service on AWS (ROSA)

Database:

- Amazon Aurora
- Amazon Aurora Serverless v2

- AWS Database Migration Service (AWS DMS)
- Amazon DocumentDB (with MongoDB compatibility)
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon MemoryDB for Redis
- Amazon RDS
- Amazon Redshift

Developer Tools:

- AWS CLI
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudShell
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CodePipeline
- AWS CodeStar
- AWS Fault Injection Simulator (AWS FIS)
- AWS SDKs and Tools
- AWS X-Ray

Management and Governance:

- AWS Auto Scaling
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS Compute Optimizer
- AWS Config

- AWS Control Tower
- AWS Health
- AWS License Manager
- Amazon Managed Grafana
- Amazon Managed Service for Prometheus
- AWS OpsWorks
- AWS Organizations
- AWS Proton
- AWS Resilience Hub
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor

Networking and Content Delivery:

- Amazon API Gateway
- AWS Client VPN
- Amazon CloudFront
- Elastic Load Balancing (ELB)
- AWS PrivateLink
- Amazon Route 53
- AWS Site-to-Site VPN
- AWS Transit Gateway
- Amazon VPC

Security, Identity, and Compliance:

- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service

- Amazon GuardDuty
- AWS IAM Identity Center
- 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 Security Token Service (AWS STS)
- AWS Shield
- AWS WAF

Serverless:

- AWS Lambda
- AWS Serverless Application Model (AWS SAM)
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Storage:

- AWS Backup
- Amazon Elastic Block Store (Amazon EBS)
- AWS Elastic Disaster Recovery
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx for Lustre
- Amazon FSx for NetApp ONTAP
- Amazon FSx for OpenZFS
- Amazon FSx for Windows File Server

- Amazon S3
- Amazon S3 Glacier
- AWS Storage Gateway

In-scope AWS services and features

Survey

How useful was this exam guide? Let us know by taking our survey.

Exam Guides - Specialty

AWS Specialty certification exams validate technical skills and experience in specific technical domains.

Topics

AWS Certified Machine Learning - Specialty (MLS-C01)

The AWS Certified Machine Learning - Specialty (MLS-C01) exam is intended for individuals who perform an artificial intelligence and machine learning (AI/ML) development or data science role. The exam validates a candidate's ability to design, build, deploy, optimize, train, tune, and maintain ML solutions for given business problems by using the AWS Cloud.

Topics

Introduction

The AWS Certified Machine Learning - Specialty (MLS-C01) exam is intended for individuals who perform an artificial intelligence and machine learning (AI/ML) development or data science role. The exam validates a candidate's ability to design, build, deploy, optimize, train, tune, and maintain ML solutions for given business problems by using the AWS Cloud.

The exam also validates a candidate's ability to complete the following tasks:

- Select and justify the appropriate ML approach for a given business problem.
- Identify appropriate AWS services to implement ML solutions.
- Design and implement scalable, cost-optimized, reliable, and secure ML solutions.

Target Candidate Description

The target candidate should have 2 or more years of experience developing, architecting, and running ML or deep learning workloads in the AWS Cloud.

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- Experience performing basic hyperparameter optimization
- Experience with ML and deep learning frameworks

Job tasks that are out of scope for the target candidate

The following list contains knowledge that the target candidate is not expected to have. This list is non-exhaustive. Knowledge in the following areas is out of scope for the exam:

- Extensive or complex algorithm development
- Extensive hyperparameter optimization
- Complex mathematical proofs and computations
- Advanced networking and network design
- Advanced database, security, and DevOps concepts
- DevOps-related tasks for Amazon EMR

Exam Content

There are two types of questions on the exam:

- Multiple choice: Has one correct response and three incorrect responses (distractors)
- Multiple response: Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

The AWS Certified Machine Learning - Specialty (MLS-C01) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 750. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Data Engineering (20% of scored content)
- Content Domain 2: Exploratory Data Analysis (24% of scored content)
- Content Domain 3: Modeling (36% of scored content)
- Content Domain 4: Machine Learning Implementation and Operations (20% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- In-Scope AWS Services
- Out-of-Scope AWS Services
- Technologies and Concepts

Content Domain 1: Data Engineering

Tasks

- Task 1.1: Create data repositories for ML
- Task 1.2: Identify and implement a data ingestion solution
- Task 1.3: Identify and implement a data transformation solution

Task 1.1: Create data repositories for ML

- Identify data sources (for example, content and location, primary sources such as user data).
- Determine storage mediums (for example, databases, Amazon S3, Amazon Elastic File System [Amazon EFS], Amazon Elastic Block Store [Amazon EBS]).

Task 1.2: Identify and implement a data ingestion solution

- Identify data job styles and job types (for example, batch load, streaming).
- Orchestrate data ingestion pipelines (batch-based ML workloads and streaming-based ML workloads).

  - Amazon Kinesis
  - Amazon Data Firehose
  - Amazon EMR
  - AWS Glue
  - Amazon Managed Service for Apache Flink
- Schedule jobs.

Task 1.3: Identify and implement a data transformation solution

- Transform data in transit (ETL, AWS Glue, Amazon EMR, AWS Batch).
- Handle ML-specific data by using MapReduce (for example, Apache Hadoop, Apache Spark, Apache Hive).

Content Domain 2: Exploratory Data Analysis

Tasks

Task 2.1: Sanitize and prepare data for modeling

- Identify and handle missing data, corrupt data, and stop words.
- Format, normalize, augment, and scale data.
- Determine whether there is sufficient labeled data.
  - Identify mitigation strategies.
  - Use data labelling tools (for example, Amazon Mechanical Turk).

Task 2.2: Perform feature engineering

- Identify and extract features from datasets, including from data sources such as text, speech, images, and public datasets.

- Analyze and evaluate feature engineering concepts (for example, binning, tokenization, outliers, synthetic features, one-hot encoding, reducing dimensionality of data).

Task 2.3: Analyze and visualize data for ML

- Create graphs (for example, scatter plots, time series, histograms, box plots).
- Interpret descriptive statistics (for example, correlation, summary statistics, p-value).
- Perform cluster analysis (for example, hierarchical, diagnosis, elbow plot, cluster size).

Content Domain 3: Modeling

Tasks

Task 3.1: Frame business problems as ML problems

- Determine when to use and when not to use ML.
- Know the difference between supervised and unsupervised learning.
- Select from among classification, regression, forecasting, clustering, recommendation, and foundation models.

Task 3.2: Select the appropriate model(s) for a given ML problem

- XGBoost, logistic regression, k-means, linear regression, decision trees, random forests, RNN, CNN, ensemble, transfer learning, and large language models (LLMs)
- Express the intuition behind models.

Task 3.3: Train ML models

- Split data between training and validation (for example, cross validation).
- Understand optimization techniques for ML training (for example, gradient descent, loss functions, convergence).
- Choose appropriate compute resources (for example GPU or CPU, distributed or non-distributed).
  - Choose appropriate compute platforms (Spark or non-Spark).
- Update and retrain models.
  - Batch or real-time/online

Task 3.4: Perform hyperparameter optimization

- Perform regularization.
  - Dropout
  - L1/L2
- Perform cross-validation.
- Initialize models.
- Understand neural network architecture (layers and nodes), learning rate, and activation functions.
- Understand tree-based models (number of trees, number of levels).
- Understand linear models (learning rate).

Task 3.5: Evaluate ML models

- Avoid overfitting or underfitting.
  - Detect and handle bias and variance.
- Evaluate metrics (for example, area under curve [AUC]-receiver operating characteristics [ROC], accuracy, precision, recall, Root Mean Square Error [RMSE], F1 score).
- Interpret confusion matrices.
- Perform offline and online model evaluation (A/B testing).
- Compare models by using metrics (for example, time to train a model, quality of model, engineering costs).
- Perform cross-validation.

Content Domain 4: Machine Learning Implementation and Operations

Tasks

Task 4.1: Build ML solutions for performance, availability, scalability, resiliency, and fault tolerance

- Log and monitor AWS environments.
  - AWS CloudTrail and Amazon CloudWatch
  - Build error monitoring solutions.
- Deploy to multiple AWS Regions and multiple Availability Zones.
- Create AMIs and golden images.
- Create Docker containers.
- Deploy Auto Scaling groups.
- Rightsize resources (for example, instances, Provisioned IOPS, volumes).
- Perform load balancing.
- Follow AWS best practices.

Task 4.2: Recommend and implement the appropriate ML services and features for a given problem

- ML on AWS (application services), for example:
  - Amazon Polly
  - Amazon Lex
  - Amazon Transcribe
  - Amazon Q

- Understand AWS service quotas.
- Determine when to build custom models and when to use Amazon SageMaker built-in algorithms.
- Understand AWS infrastructure (for example, instance types) and cost considerations.
  - Use Spot Instances to train deep learning models by using AWS Batch.

Task 4.3: Apply basic AWS security practices to ML solutions

- AWS Identity and Access Management (IAM)
- S3 bucket policies
- Security groups
- VPCs
- Encryption and anonymization

Task 4.4: Deploy and operationalize ML solutions

- Expose endpoints and interact with them.
- Understand ML models.
- Perform A/B testing.
- Retrain pipelines.
- Debug and troubleshoot ML models.
  - Detect and mitigate drops in performance.
  - Monitor performance of the model.

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Machine Learning - Specialty (MLS-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

- Containers
- Database
- Internet of Things
- Machine Learning
- Management and Governance
- Networking and Content Delivery
- Security, Identity, and Compliance
- Storage

Analytics

- Amazon Athena
- Amazon Data Firehose
- Amazon EMR
- AWS Glue
- Amazon Kinesis
- Amazon Kinesis Data Streams
- AWS Lake Formation
- Amazon Managed Service for Apache Flink
- Amazon OpenSearch Service
- Amazon QuickSight

Compute

- AWS Batch
- Amazon EC2
- AWS Lambda

Containers

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)

- Amazon Elastic Kubernetes Service (Amazon EKS)
- AWS Fargate

Database

- Amazon Redshift

Internet of Things

- AWS IoT Greengrass

Machine Learning

- Amazon Bedrock
- Amazon Comprehend
- AWS Deep Learning AMIs (DLAMI)
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Lex
- Amazon Kendra
- Amazon Mechanical Turk
- Amazon Polly
- Amazon Q
- Amazon Rekognition
- Amazon SageMaker
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Management and Governance

- AWS CloudTrail

- Amazon CloudWatch

Networking and Content Delivery

- Amazon VPC

Security, Identity, and Compliance

- AWS Identity and Access Management (IAM)

Storage

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx
- Amazon S3

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the AWS Certified Machine Learning - Specialty (MLS-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list.

Topics

Analytics

- AWS Data Pipeline

Machine Learning

- AWS DeepRacer

- Amazon Machine Learning (Amazon ML)

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Ingestion and collection
- Processing and ETL
- Data analysis and visualization
- Model training
- Model deployment and inference
- Operationalizing ML
- AWS ML application services
- Language relevant to ML (for example, Python, Java, Scala, R, SQL)
- Notebooks and integrated development environments (IDEs)

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified Security - Specialty (SCS-C03)

The AWS Certified Security - Specialty exam is intended for individuals who have a responsibility to secure cloud solutions. The exam validates a candidate's ability to effectively demonstrate knowledge about securing AWS products and services.

Topics

Introduction

The AWS Certified Security - Specialty exam is intended for individuals who have a responsibility to secure cloud solutions. The exam validates a candidate's ability to effectively demonstrate knowledge about securing AWS products and services.

The exam also validates a candidate's ability to complete the following tasks:

- Apply specialized data classifications and AWS data protection mechanisms.
- Implement data-encryption methods and AWS encryption mechanisms.
- Implement AWS mechanisms to follow secure internet protocols.
- Use AWS security services and features to ensure secure production environments.
- Make decisions that account for tradeoffs between cost, security, and deployment complexity to meet a set of application requirements.
- Understand security operations and risks.

Target candidate description

The target candidate should have the equivalent of 3–5 years of experience securing cloud solutions.

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- The AWS shared responsibility model and its application
- Managing identity at scale
- Multi-account governance
- Managing software supply chain risks
- Security incident prevention and response strategies
- Vulnerability management in the cloud
- Developing firewall rules at scale for layers 3–7
- Incident root cause analysis
- Experience responding to an audit
- Logging and monitoring strategies
- Data encryption methodologies, both at-rest and in-transit
- Disaster recovery controls, including backup strategies

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Design cryptographic algorithms
- Analyze traffic on the packet level
- Architect overall cloud deployments
- Manage end-user compute resources
- Train machine learning models

Exam content

Response types

The exam includes one or more of the following question types:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)

- **Multiple response:** Has two or more correct responses out of five or more response options
- **Ordering:** Has a list of 3–5 responses to complete a specified task. You must select the correct responses and place the responses in the correct order to receive credit for the question.
- **Matching:** Has a list of responses to match with a list of 3–7 prompts. You must match all the pairs correctly to receive credit for the question.

Unanswered questions are scored as incorrect. There is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified Security - Specialty (SCS-C03) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 750. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam.

The exam has the following content domains and weightings:

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

Content Domain 1: Detection

Tasks

Task 1.1: Design and implement monitoring and alerting solutions for an AWS account or organization

Skills in:

- Skill 1.1.1: Analyze workloads to determine monitoring requirements.
- Skill 1.1.2: Design and implement workload monitoring strategies (for example, by configuring resource health checks).
- Skill 1.1.3: Aggregate security and monitoring events.

- Skill 1.1.4: Create metrics, alerts, and dashboards to detect anomalous data and events (for example, Amazon GuardDuty, Amazon Security Lake, AWS Security Hub, Amazon Macie).
- Skill 1.1.5: Create and manage automations to perform regular assessments and investigations (for example, by deploying AWS Config conformance packs, Security Hub, AWS Systems Manager State Manager).

Task 1.2: Design and implement logging solutions

Skills in:

- Skill 1.2.1: Identify sources for log ingestion and storage based on requirements.
- Skill 1.2.2: Configure logging for AWS services and applications (for example, by configuring an AWS CloudTrail trail for an organization, by creating a dedicated Amazon CloudWatch logging account, by configuring the Amazon CloudWatch Logs agent).
- Skill 1.2.3: Implement log storage and log data lakes (for example, Security Lake) and integrate with third-party security tools.
- Skill 1.2.4: Use AWS services to analyze logs (for example, CloudWatch Logs Insights, Amazon Athena, Security Hub findings).
- Skill 1.2.5: Use AWS services to normalize, parse, and correlate logs (for example, Amazon OpenSearch Service, AWS Lambda, Amazon Managed Grafana).
- Skill 1.2.6: Determine and configure appropriate log sources based on network design, threats, and attacks (for example, VPC Flow Logs, transit gateway flow logs, Amazon Route 53 Resolver logs).

Task 1.3: Troubleshoot security monitoring, logging, and alerting solutions

Skills in:

- Skill 1.3.1: Analyze the functionality, permissions, and configuration of resources (for example, Lambda function logging, Amazon API Gateway logging, health checks, Amazon CloudFront logging).
- Skill 1.3.2: Remediate misconfiguration of resources (for example, by troubleshooting CloudWatch Agent configurations, troubleshooting missing logs).

Content Domain 2: Incident Response

Tasks

Task 2.1: Design and test an incident response plan

Skills in:

- Skill 2.1.1: Design and implement response plans and runbooks to respond to security incidents (for example, Systems Manager OpsCenter, Amazon SageMaker AI notebooks).
- Skill 2.1.2: Use AWS service features and capabilities to configure services to be prepared for incidents (for example, by provisioning access, deploying security tools, minimizing the blast radius, configuring AWS Shield Advanced protections).
- Skill 2.1.3: Recommend procedures to test and validate the effectiveness of an incident response plan (for example, AWS Fault Injection Service, AWS Resilience Hub).
- Skill 2.1.4: Use AWS services to automatically remediate incidents (for example, Systems Manager, Automated Forensics Orchestrator for Amazon EC2, AWS Step Functions, Amazon Application Recovery Controller, Lambda functions).

Task 2.2: Respond to security events

Skills in:

- Skill 2.2.1: Capture and store relevant system and application logs as forensic artifacts.
- Skill 2.2.2: Search and correlate logs for security events across applications and AWS services.
- Skill 2.2.3: Validate findings from AWS security services to assess the scope and impact of an event.
- Skill 2.2.4: Respond to affected resources by containing and eradicating threats, and recover resources (for example, by implementing network containment controls, restoring backups).
- Skill 2.2.5: Describe methods to conduct root cause analysis (for example, Amazon Detective).

Content Domain 3: Infrastructure Security

Tasks

Task 3.1: Design, implement, and troubleshoot security controls for network edge services

Skills in:

- Skill 3.1.1: Define and select edge security strategies based on anticipated threats and attacks.
- Skill 3.1.2: Implement appropriate network edge protection (for example, CloudFront headers, AWS WAF, AWS IoT policies, protecting against OWASP Top 10 threats, Amazon S3 cross-origin resource sharing [CORS], Shield Advanced).
- Skill 3.1.3: Design and implement AWS edge controls and rules based on requirements (for example, geography, geolocation, rate limiting, client fingerprinting).
- Skill 3.1.4: Configure integrations with AWS edge services and third-party services (for example, by ingesting data in Open Cybersecurity Schema Framework [OCSF] format, by using third-party WAF rules).

Task 3.2: Design, implement, and troubleshoot security controls for compute workloads

Skills in:

- Skill 3.2.1: Design and implement hardened Amazon EC2 AMIs and container images to secure compute workloads and embed security controls (for example, Systems Manager, EC2 Image Builder).
- Skill 3.2.2: Apply instance profiles, service roles, and execution roles appropriately to authorize compute workloads.
- Skill 3.2.3: Scan compute resources for known vulnerabilities (for example, scan container images and Lambda functions by using Amazon Inspector, monitor compute runtimes by using GuardDuty).

- Skill 3.2.4: Deploy patches across compute resources to maintain secure and compliant environments by automating update processes and by integrating continuous validation (for example, Systems Manager Patch Manager, Amazon Inspector).
- Skill 3.2.5: Configure secure administrative access to compute resources (for example, Systems Manager Session Manager, EC2 Instance Connect).
- Skill 3.2.6: Configure security tools to discover and remediate vulnerabilities within a pipeline (for example, Amazon Q Developer, Amazon CodeGuru Security).
- Skill 3.2.7: Implement protections and guardrails for generative AI applications (for example, by applying GenAI OWASP Top 10 for LLM Applications protections).

Task 3.3: Design and troubleshoot network security controls

Skills in:

- Skill 3.3.1: Design and troubleshoot appropriate network controls to permit or prevent network traffic as required (for example, security groups, network ACLs, AWS Network Firewall).
- Skill 3.3.2: Design secure connectivity between hybrid and multi-cloud networks (for example, AWS Site-to-Site VPN, AWS Direct Connect, MAC Security [MACsec]).
- Skill 3.3.3: Determine and configure security workload requirements for communication between hybrid environments and AWS (for example, by using AWS Verified Access).
- Skill 3.3.4: Design network segmentation based on security requirements (for example, north/south and east/west traffic protections, isolated subnets).
- Skill 3.3.5: Identify unnecessary network access (for example, AWS Verified Access, Network Access Analyzer, Amazon Inspector network reachability findings).

Content Domain 4: Identity and Access Management

Tasks

Task 4.1: Design, implement, and troubleshoot authentication strategies

Skills in:

- Skill 4.1.1: Design and establish identity solutions for human, application, and system authentication (for example, AWS IAM Identity Center, Amazon Cognito, multi-factor authentication [MFA], identity provider [IdP] integration).
- Skill 4.1.2: Configure mechanisms to issue temporary credentials (for example, AWS Security Token Service [AWS STS], Amazon S3 presigned URLs).
- Skill 4.1.3: Troubleshooting authentication issues (for example, CloudTrail, Amazon Cognito, IAM Identity Center permission sets, AWS Directory Service).

Task 4.2: Design, implement, and troubleshoot authorization strategies

Skills in:

- Skill 4.2.1: Design and evaluate authorization controls for human, application, and system access (for example, Amazon Verified Permissions, IAM paths, IAM Roles Anywhere, resource policies for cross-account access, IAM role trust policies).
- Skill 4.2.2: Design attribute-based access control (ABAC) and role-based access control (RBAC) strategies (for example, by configuring resource access based on tags or attributes).
- Skill 4.2.3: Design, interpret, and implement IAM policies by following the principle of least privilege (for example, permission boundaries, session policies).
- Skill 4.2.4: Analyze authorization failures to determine causes or effects (for example, IAM Policy Simulator, IAM Access Analyzer).
- Skill 4.2.5: Investigate and correct unintended permissions, authorizations, or privileges granted to a resource, service, or entity (for example, IAM Access Analyzer).

Content Domain 5: Data Protection

Tasks

Task 5.1: Design and implement controls for data in transit

Skills in:

- Skill 5.1.1: Design and configure mechanisms to require encryption when connecting to connect to resources (for example, by configuring Elastic Load Balancing [ELB] security policies, by enforcing TLS configurations).
- Skill 5.1.2: Design and configure mechanisms for secure and private access to resources (for example, AWS PrivateLink, VPC endpoints, AWS Client VPN, AWS Verified Access).
- Skill 5.1.3: Design and configure inter-resource encryption in transit (for example, inter-node encryption configurations for Amazon EMR, Amazon Elastic Kubernetes Service [Amazon EKS], SageMaker AI, Nitro encryption).

Task 5.2: Design and implement controls for data at rest

Skills in:

- Skill 5.2.1: Design, implement, and configure data encryption at rest based on specific requirements (for example, by selecting the appropriate encryption key service such as AWS CloudHSM or AWS Key Management Service [AWS KMS] or by selecting the appropriate encryption type such as client-side encryption or server-side encryption).
- Skill 5.2.2: Design and configure mechanisms to protect data integrity (for example, S3 Object Lock, S3 Glacier Vault Lock, versioning, digital code signing, file validation).
- Skill 5.2.3: Design automatic lifecycle management and retention solutions for data (for example, S3 Lifecycle policies, S3 Object Lock, Amazon Elastic File System [Amazon EFS] Lifecycle policies, Amazon FSx for Lustre backup policies).
- Skill 5.2.4: Design and configure secure data replication and backup solutions (for example, Amazon Data Lifecycle Manager, AWS Backup, ransomware protection, AWS DataSync).

Task 5.3: Design and implement controls to protect confidential data, credentials, secrets, and cryptographic key materials

Skills in:

- Skill 5.3.1: Design management and rotation of credentials and secrets (for example, AWS Secrets Manager).
- Skill 5.3.2: Manage and use imported key material (for example, by managing and rotating imported key material, by managing and configuring external key stores).
- Skill 5.3.3: Describe the differences between imported key material and AWS generated key material.

- Skill 5.3.4: Mask sensitive data (for example, CloudWatch Logs data protection policies, Amazon Simple Notification Service [Amazon SNS] message data protection).
- Skill 5.3.5: Create and manage encryption keys and certificates across a single AWS Region or multiple Regions (for example, AWS KMS customer managed AWS KMS keys, AWS Private Certificate Authority).

Content Domain 6: Security Foundations and Governance

Tasks

Task 6.1: Develop a strategy to centrally deploy and manage AWS accounts

Skills in:

- Skill 6.1.1: Deploy and configure organizations by using AWS Organizations.
- Skill 6.1.2: Implement and manage AWS Control Tower in new and existing environments, and deploy optional and custom controls.
- Skill 6.1.3: Implement organization policies to manage permissions (for example, SCPs, RCPs, AI service opt-out policies, declarative policies).
- Skill 6.1.4: Centrally manage security services (for example, delegated administrator accounts).
- Skill 6.1.5: Manage AWS account root user credentials (for example, by centralizing root access for member accounts, managing MFA, designing break-glass procedures).

Task 6.2: Implement a secure and consistent deployment strategy for cloud resources

Skills in:

- Skill 6.2.1: Use infrastructure as code (IaC) to deploy cloud resources consistently and securely across accounts (for example, CloudFormation stack sets, third-party IaC tools, CloudFormation Guard, cfn-lint).

- Skill 6.2.2: Use tags to organize AWS resources into groups for management (for example, by grouping by department, cost center, environment).
- Skill 6.2.3: Deploy and enforce policies and configurations from a central source (for example, AWS Firewall Manager).
- Skill 6.2.4: Securely share resources across AWS accounts (for example, AWS Service Catalog, AWS Resource Access Manager [AWS RAM]).

Task 6.3: Evaluate the compliance of AWS resources

Skills in:

- Skill 6.3.1: Create or enable rules to detect and remediate noncompliant AWS resources and to send notifications (for example, by using AWS Config to aggregate alerts and remediate non-compliant resources, Security Hub).
- Skill 6.3.2: Use AWS audit services to collect and organize evidence (for example, AWS Audit Manager, AWS Artifact).
- Skill 6.3.3: Use AWS services to evaluate architecture for compliance with AWS security best practices (for example, AWS Well-Architected Framework tool).

In-scope AWS services and features

In-scope AWS services and features

Note: Security affects all AWS services. Many services do not appear in this list because the overall service is out of scope, but the security aspects of the service are in scope. For example, a candidate for this exam would not be asked about the steps to set up replication for an S3 bucket. However, the candidate might be asked about configuring an S3 bucket policy.

The following list contains AWS services and features that are in scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions:

Topics

- Developer Tools
- Internet of Things
- Machine Learning
- Management and Governance
- Networking and Content Delivery
- Security, Identity, and Compliance
- Storage and Data Management

Analytics

- Amazon Athena
- Amazon OpenSearch Service

Application Integration

- Amazon Simple Notification Service (Amazon SNS)
- AWS Step Functions

Compute

- Amazon API Gateway
- Amazon EC2 (including EC2 Image Builder, EC2 Instance Connect)
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Amazon EMR
- AWS Lambda
- Amazon Data Lifecycle Manager

Developer Tools

- AWS Fault Injection Service

Internet of Things

- AWS IoT Core

Machine Learning

- Amazon Bedrock
- Amazon CodeGuru Security
- Amazon Q Business
- Amazon Q Developer
- Amazon SageMaker AI

Management and Governance

- AWS CloudFormation
- AWS CloudTrail
- AWS CloudTrail Lake
- Amazon CloudWatch
- AWS Config
- AWS Control Tower
- Amazon Managed Grafana
- AWS Organizations
- AWS Resilience Hub
- AWS Resource Access Manager (AWS RAM)
- AWS Service Catalog
- AWS Systems Manager
- AWS Trusted Advisor
- AWS User Notifications
- AWS Well-Architected Tool

Networking and Content Delivery

- Amazon Application Recovery Controller
- Amazon VPC
  - Network Access Analyzer
  - Network ACLs

  - Security groups
  - VPC endpoints
  - AWS Site-to-Site VPN
  - Flow Logs
  - VPC Endpoints
  - AWS Verified Access
- AWS Client VPN
- Amazon CloudFront
- Amazon Verified Permissions
- Amazon Route 53 (including Route 53 Resolver DNS Firewall)
- AWS Direct Connect
- Elastic Load Balancing (ELB)
- Network Access Analyzer
- AWS Transit Gateway

Security, Identity, and Compliance

- AWS Artifact
- AWS Audit Manager
- AWS Certificate Manager (ACM)
- AWS CloudHSM
- Amazon Cognito
- Amazon Detective
- AWS Directory Service
- AWS Firewall Manager
- Automated Forensics Orchestrator for Amazon EC2
- Amazon GuardDuty
- AWS IAM Identity Center
- AWS Identity and Access Management (IAM)
- Amazon Inspector

- AWS Key Management Service (AWS KMS)
- Amazon Macie
- AWS Network Firewall
- AWS Private Certificate Authority
- AWS Secrets Manager
- AWS Security Hub
- Amazon Security Lake
- AWS Security Token Service (AWS STS)
- AWS Shield
- AWS Shield Advanced
- AWS WAF

Storage and Data Management

- Amazon S3
- AWS Backup
- AWS DataSync
- Amazon Elastic File System (Amazon EFS) (including EFS Lifecycle policies)
- Amazon FSx for Lustre

Out-of-scope AWS services and features

The following list contains AWS services and features that are out of scope for the exam. This list is non-exhaustive and is subject to change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list:

Topics

- Application Integration
- Security, Identity, and Compliance

Application Integration

- Amazon Managed Workflows for Apache Airflow (Amazon MWAA)

Security, Identity, and Compliance

- AWS Payment Cryptography

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- AWS CLI
- AWS SDKs
- AWS Management Console
- Secure remote access
- Certificate management
- Infrastructure as code (IaC)

Appendix: Comparison of SCS-C02 and SCS-C03

Side-by-side comparison

The following table shows the domains and the percentage of scored questions in each domain for the SCS-C02 exam (in use until December 1, 2025) and the SCS-C03 exam (in use beginning December 2, 2025).

SCS-C02 Domain	SCS-C03 Domain
Domain 1: Threat Detection and Incident Response (14%)	Content Domain 1: Detection (16% of scored content)
Domain 2: Security Logging and Monitoring (18%)	Content Domain 2: Incident Response (14%)
Domain 3: Infrastructure Security (20%)	Content Domain 3: Infrastructure Security (18%)

SCS-C02 Domain	SCS-C03 Domain
Domain 4: Identity and Access Management (16%)	Content Domain 4: Identity and Access Management (20%)
Domain 5: Data Protection (18%)	Content Domain 5: Data Protection (18%)
Domain 6: Management and Security Governance (14%)	Content Domain 6: Security Foundations and Governance (14%)

Additions of content for SCS-C03

In Task 2.2.3, the following content was added:

- 2.2.3 Validate findings from AWS security services to assess the scope and impact of an event.

In Task 3.1.4, the following content was added:

- 3.1.4 Configure integrations with AWS edge services and third-party services (for example, by ingesting data in Open Cybersecurity Schema Framework [OCSF] format, by using third-party WAF rules).

In Task 3.2.7, the following content was added:

- 3.2.7 Implement protections and guardrails for generative AI applications (for example, by applying GenAI OWASP Top 10 for LLM Applications protections).

In Task 5.1.3, the following content was added:

- 5.1.3 Design and configure inter-resource encryption in-transit (for example, inter-node encryption configurations for Amazon EMR, Amazon Elastic Kubernetes Service [Amazon EKS], SageMaker AI, Nitro encryption).

In Task 5.3.3, the following content was added:

- 5.3.3 Describe the differences between imported key material and AWS generated key material.

In Task 5.3.4, the following content was added:

- 5.3.4 Mask sensitive data (for example, CloudWatch Logs data protection policies, Amazon Simple Notification Service [Amazon SNS] message data protection).

In Task 5.3.5, the following content was added:

- 5.3.5 Create and manage encryption keys and certificates across a single AWS Region or multiple Regions (for example, AWS KMS customer managed AWS KMS keys, AWS Private Certificate Authority).

Deletions of content for SCS-C03

In Task 6.4, the following content was removed:

- Identify security gaps through architectural reviews and cost analysis.

In Task 1.1, the following content was removed:

- AWS Security Finding Format (ASFF)

In Task 1.3, the following content was removed:

- AWS Security Incident Response Guide

In Task 2.5 the following content was removed:

- Log format and components (for example, CloudTrail logs)

In Task 3.3, the following content was removed:

- Host-based security (for example, firewalls, hardening)
- Activating host-based security mechanisms (for example, host-based firewalls)

In Task 3.4, the following content was removed:

- How to analyze reachability (for example, by using VPC Reachability Analyzer and Amazon Inspector)
- Fundamental TCP/IP networking concepts (for example, UDP compared with TCP, ports, Open Systems Interconnection [OSI] model, network operating system utilities)
- Identifying, interpreting, and prioritizing problems in network connectivity (for example, by using Amazon Inspector Network Reachability)

In Task 4.2, the following content was removed:

- Components and impact of a policy (for example, Principal, Action, Resource, Condition)

In Task 5.1, the following content was removed:

- TLS concepts
- Designing cross-Region networking by using private VIFs and public VIFs

In Task 5.2, the following content was removed:

- Configure S3 static website hosting.

Recategorizations of content for SCS-C03

The following major content reorganizations have occurred in the transition from SCS-C02 to SCS-C03:

SCS-C02 Domains 1 and 2 have been restructured:

- "Threat Detection and Incident Response" and "Security Logging and Monitoring" are now:
  - Domain 1: Detection
  - Domain 2: Incident Response

Domain 6 has been renamed for SCS-C03:

- From "Management and Security Governance" to "Security Foundations and Governance"

The following task statement have been recategorized:

SCS-C02 Task Statement 1.1 is mapped to the following tasks in SCS-C03:

- 1.1 Design and implement monitoring and alerting for an AWS account or organization.
- 1.2 Design and implement logging.
- 2.1 Design and test an incident response plan.
- 2.2 Respond to security events.

SCS-C02 Task Statement 1.2 is mapped to the following tasks in SCS-C03:

- 1.1 Design and implement monitoring and alerting for an AWS account or organization.
- 1.2 Design and implement logging.

SCS-C02 Task Statement 1.3 is mapped to the following tasks in SCS-C03:

- 2.1 Design and test an incident response plan.
- 2.2 Respond to security events.

SCS-C02 Task Statement 2.1 is mapped to the following tasks in SCS-C03:

- 1.1 Design and implement monitoring and alerting for an AWS account or organization.

SCS-C02 Task Statement 2.2 is mapped to the following tasks in SCS-C03:

- 1.1 Design and implement monitoring and alerting for an AWS account or organization.
- 1.2 Design and implement logging.
- 1.3 Troubleshoot security monitoring, logging and alerting.

SCS-C02 Task Statement 2.3 is mapped to the following tasks in SCS-C03:

- 1.2 Design and implement logging.

SCS-C02 Task Statement 2.4 is mapped to the following tasks in SCS-C03:

- 1.2 Design and implement logging.
- 1.3 Troubleshoot security monitoring, logging and alerting.

SCS-C02 Task Statement 2.5 is mapped to the following tasks in SCS-C03:

- 1.2 Design and implement logging.

SCS-C02 Task Statement 3.1 is mapped to the following tasks in SCS-C03:

- 1.2 Design and implement logging.
- 3.1 Design, implement, and troubleshoot security controls for network edge services.

SCS-C02 Task Statement 3.2 is mapped to the following tasks in SCS-C03:

- 1.2 Design and implement logging.
- 3.3 Design and troubleshoot network security controls.
- 5.1 Design and implement controls for data in transit.
- 6.2 Implement a secure and consistent deployment strategy for cloud resources.

SCS-C02 Task Statement 3.3 is mapped to the following tasks in SCS-C03:

- 3.2 Design, implement, and troubleshoot security controls for compute workloads.
- 5.3 Design and implement controls to protect confidential data, credentials, secrets, and cryptographic key materials.

SCS-C02 Task Statement 3.4 is mapped to the following tasks in SCS-C03:

- 1.2 Design and implement logging.
- 3.3 Design and troubleshoot network security controls.

SCS-C02 Task Statement 4.1 is mapped to the following tasks in SCS-C03:

- 4.1 Design, implement, and troubleshoot authentication strategies

SCS-C02 Task Statement 4.2 is mapped to the following tasks in SCS-C03:

- 4.2 Design, implement, and troubleshoot authorization strategies

SCS-C02 Task Statement 5.1 is mapped to the following tasks in SCS-C03:

- 3.2 Design, implement, and troubleshoot security controls for compute workloads.
- 3.3 Design and troubleshoot network security controls.
- 5.1 Design and implement controls for data in transit.

SCS-C02 Task Statement 5.2 is mapped to the following tasks in SCS-C03:

- 4.2 Design, implement, and troubleshoot authorization strategies
- 5.2 Design and implement controls for data at rest.

SCS-C02 Task Statement 5.3 is mapped to the following tasks in SCS-C03:

- 5.2 Design and implement controls for data at rest.

SCS-C02 Task Statement 5.4 is mapped to the following tasks in SCS-C03:

- 5.2 Design and implement controls for data at rest.
- 5.3 Design and implement controls to protect confidential data, credentials, secrets, and cryptographic key materials.

SCS-C02 Task Statement 6.1 is mapped to the following tasks in SCS-C03:

- 4.2 Design, implement, and troubleshoot authorization strategies
- 6.1 Develop a strategy to centrally deploy and manage AWS accounts.

SCS-C02 Task Statement 6.2 is mapped to the following tasks in SCS-C03:

- 6.2 Implement a secure and consistent deployment strategy for cloud resources.

SCS-C02 Task Statement 6.3 is mapped to the following tasks in SCS-C03:

- 1.1 Design and implement monitoring and alerting for an AWS account or organization.
- 5.2 Design and implement controls for data at rest.
- 6.3 Evaluate the compliance of AWS resources.

SCS-C02 Task Statement 6.4 is mapped to the following tasks in SCS-C03:

- 2.1 Design and test an incident response plan.
- 1.1 Design and implement monitoring and alerting for an AWS account or organization.
- 6.3 Evaluate the compliance of AWS resources.

Survey

How useful was this exam guide? Let us know by taking our survey.

AWS Certified Advanced Networking - Specialty (ANS-C01)

The AWS Certified Advanced Networking - Specialty (ANS-C01) exam is intended for individuals who perform an AWS networking specialist's role. The exam validates a candidate's ability to design, implement, manage, and secure AWS and hybrid network architectures at scale.

Topics

Introduction

The AWS Certified Advanced Networking - Specialty (ANS-C01) exam is intended for individuals who perform an AWS networking specialist's role. The exam validates a candidate's ability to design, implement, manage, and secure AWS and hybrid network architectures at scale.

The exam also validates a candidate's ability to complete the following tasks:

- Design and develop hybrid and cloud-based networking solutions by using AWS
- Implement core AWS networking services according to AWS best practices
- Operate and maintain hybrid and cloud-based network architecture for all AWS services
- Use tools to deploy and automate hybrid and cloud-based AWS networking tasks
- Implement secure AWS networks using AWS native networking constructs and services

Target Candidate Description

The target candidate should have 5 or more years of networking experience with 2 or more years of cloud and hybrid networking experience.

Recommended AWS knowledge

The target candidate should have the following AWS knowledge:

- AWS networking nuances and how they relate to the integration of AWS services
- AWS security best practices
- AWS compute and storage options and their underlying consistency models

Exam Content

Question Types

The exam contains the following question types:

- **Multiple response:** Has two or more correct responses out of five or more response options. You must select all the correct responses to receive credit for the question.
- **Matching:** Has a list of responses to match with a list of 3–7 prompts. You must match all the pairs correctly to receive credit for the question.

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored Content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam Results

This AWS certification exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 700. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content Outline

This exam guide includes weightings, content domains, and task statements for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task statement is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Network Design (30% of scored content)
- Content Domain 2: Network Implementation (26% of scored content)
- Content Domain 3: Network Management and Operation (20% of scored content)
- Content Domain 4: Network Security, Compliance, and Governance (24% of scored content)

AWS Services for the Exam

The AWS Certified Advanced Networking - Specialty exam covers specific AWS services that are relevant to networking specialists. Understanding which services are in scope and which are out of scope can help you focus your preparation efforts.

For detailed information about the AWS services covered in the exam, see the following sections:

- In-Scope AWS Services
- Out-of-Scope AWS Services

Content Domain 1: Network Design

Tasks

- Task 1.1: Design a solution that incorporates edge network services to optimize user performance and traffic management for global architectures
- Task 1.2: Design DNS solutions that meet public, private, and hybrid requirements
- Task 1.3: Design solutions that integrate load balancing to meet high availability, scalability, and security requirements
- Task 1.4: Define logging and monitoring requirements across AWS and hybrid networks
- Task 1.5: Design a routing strategy and connectivity architecture between on-premises networks and the AWS Cloud
- Task 1.6: Design a routing strategy and connectivity architecture that include multiple AWS accounts, AWS Regions, and VPCs to support different connectivity patterns

Task 1.1: Design a solution that incorporates edge network services to optimize user performance and traffic management for global architectures

Knowledge of:

- Design patterns for the usage of content distribution networks (for example, Amazon CloudFront)
- Design patterns for global traffic management (for example, AWS Global Accelerator)
- Integration patterns for content distribution networks and global traffic management with other services (for example, Elastic Load Balancing [ELB], Amazon API Gateway)

Skills in:

- Evaluating requirements of global inbound and outbound traffic from the internet to design an appropriate content distribution solution

Task 1.2: Design DNS solutions that meet public, private, and hybrid requirements

Knowledge of:

- DNS protocol (for example, DNS records, TTL, DNSSEC, DNS delegation, zones)
- DNS logging and monitoring
- Amazon Route 53 features (for example, alias records, traffic policies, resolvers, health checks)
- Integration of Route 53 with other AWS networking services (for example, Amazon VPC)
- Integration of Route 53 with hybrid, multi-account, and multi-Region options
- Domain registration

Skills in:

- Using Route 53 public hosted zones
- Using Route 53 private hosted zones
- Using Route 53 Resolver endpoints in hybrid and AWS architectures
- Using Route 53 for global traffic management
- Creating and managing domain registrations

Task 1.3: Design solutions that integrate load balancing to meet high availability, scalability, and security requirements

Knowledge of:

- How load balancing works at layer 3, layer 4, and layer 7 of the OSI model
- Different types of load balancers and how they meet requirements for network design, high availability, and security
- Connectivity patterns that apply to load balancing based on the use case (for example, internal load balancers, external load balancers)
- Scaling factors for load balancers
- Integrations of load balancers and other AWS services (for example, Global Accelerator, CloudFront, AWS WAF, Route 53, Amazon Elastic Kubernetes Service [Amazon EKS], AWS Certificate Manager [ACM])
- Configuration options for load balancers (for example, proxy protocol, cross-zone load balancing, session affinity [sticky sessions], routing algorithms)
- Configuration options for load balancer target groups (for example, TCP, GENEVE, IP compared with instance)
- AWS Load Balancer Controller for Kubernetes clusters
- Considerations for encryption and authentication with load balancers (for example, TLS termination, TLS passthrough)

Skills in:

- Selecting an appropriate load balancer based on the use case
- Integrating auto scaling with load balancing solutions
- Integrating load balancers with existing application deployments

Task 1.4: Define logging and monitoring requirements across AWS and hybrid networks

Knowledge of:

- Amazon CloudWatch metrics, agents, logs, alarms, dashboards, and insights in AWS architectures to provide visibility

- AWS Transit Gateway Network Manager in architectures to provide visibility
- VPC Reachability Analyzer in architectures to provide visibility
- Flow logs and traffic mirroring in architectures to provide visibility
- Access logging (for example, load balancers, CloudFront)

Skills in:

- Identifying the logging and monitoring requirements
- Recommending appropriate metrics to provide visibility of the network status
- Capturing baseline network performance

Task 1.5: Design a routing strategy and connectivity architecture between on-premises networks and the AWS Cloud

Knowledge of:

- Routing fundamentals (for example, dynamic compared with static, BGP)
- Layer 1 and layer 2 concepts for physical interconnects (for example, VLAN, link aggregation group [LAG], optics, jumbo frames)
- Encapsulation and encryption technologies (for example, Generic Routing Encapsulation [GRE], IPsec)
- Resource sharing across AWS accounts
- Overlay networks

Skills in:

- Identifying the requirements for hybrid connectivity
- Designing a redundant hybrid connectivity model with AWS services (for example, AWS Direct Connect, AWS Site-to-Site VPN)
- Designing BGP routing with BGP attributes to influence the traffic flows based on the desired traffic patterns (load sharing, active/passive)
- Designing for integration of a software-defined wide area network (SD-WAN) with AWS (for example, Transit Gateway Connect, overlay networks)

Task 1.6: Design a routing strategy and connectivity architecture that include multiple AWS accounts, AWS Regions, and VPCs to support different connectivity patterns

Knowledge of:

- Different connectivity patterns and use cases (for example, VPC peering, Transit Gateway, AWS PrivateLink)
- Capabilities and advantages of VPC sharing
- IP subnets and solutions accounting for IP address overlaps

Skills in:

- Connecting multiple VPCs by using the most appropriate services based on requirements (for example, using VPC peering, Transit Gateway, PrivateLink)
- Using VPC sharing in a multi-account setup
- Managing IP overlaps by using different available services and options (for example, NAT, PrivateLink, Transit Gateway routing)

Content Domain 2: Network Implementation

Tasks

Task 2.1: Implement routing and connectivity between on-premises networks and the AWS Cloud

Knowledge of:

- Routing protocols (for example, static, dynamic)
- VPNs (for example, security, accelerated VPN)

- Layer 1 and types of hardware to use (for example, Letter of Authorization [LOA] documents, colocation facilities, Direct Connect)
- Layer 2 and layer 3 (for example, VLANs, IP addressing, gateways, routing, switching)
- Traffic management and SD-WAN (for example, Transit Gateway Connect)
- DNS (for example, conditional forwarding, hosted zones, resolvers)
- Security appliances (for example, firewalls)
- Load balancing (for example, layer 4 compared with layer 7, reverse proxies, layer 3)
- Infrastructure automation
- AWS Organizations and AWS Resource Access Manager (AWS RAM) (for example, multi-account Transit Gateway, Direct Connect, Amazon VPC, Route 53)
- Test connectivity (for example, Route Analyzer, Reachability Analyzer)
- Networking services of VPCs

Skills in:

- Configuring the physical network requirements for hybrid connectivity solutions
- Configuring static or dynamic routing protocols to work with hybrid connectivity solutions
- Configuring existing on-premises networks to connect with the AWS Cloud
- Configuring existing on-premises name resolution with the AWS Cloud
- Configuring and implementing load balancing solutions
- Configuring network monitoring and logging for AWS services
- Testing and validating connectivity between environments

Task 2.2: Implement routing and connectivity across multiple AWS accounts, Regions, and VPCs to support different connectivity patterns

Knowledge of:

- Inter-VPC and multi-account connectivity (for example, VPC peering, Transit Gateway, VPN, third-party vendors, SD-WAN, multi-protocol label switching [MPLS])
- Private application connectivity (for example, PrivateLink)
- Methods of expanding AWS networking connectivity (for example, Organizations, AWS RAM)

- Host and service name resolution for applications and clients (for example, DNS)
- Infrastructure automation
- Authentication and authorization (for example, SAML, Active Directory)
- Security (for example, security groups, network ACLs, AWS Network Firewall)
- Test connectivity (for example, Route Analyzer, Reachability Analyzer, tooling)

Skills in:

- Configuring network connectivity architectures by using AWS services in a single-VPC or multi-VPC design (for example, DHCP, routing, security groups)
- Configuring hybrid connectivity with existing third-party vendor solutions
- Configuring a hub-and-spoke network architecture (for example, Transit Gateway, transit VPC)
- Configuring a DNS solution to make hybrid connectivity possible
- Implementing security between network boundaries
- Configuring network monitoring and logging by using AWS solutions

Task 2.3: Implement complex hybrid and multi-account DNS architectures

Knowledge of:

- When to use private hosted zones and public hosted zones
- Methods to alter traffic management (for example, based on latency, geography, weighting)
- DNS delegation and forwarding (for example, conditional forwarding)
- Different DNS record types (for example, A, AAAA, TXT, pointer records, alias records)
- DNSSEC
- How to share DNS services between accounts (for example, AWS RAM)
- Requirements and implementation options for outbound and inbound endpoints

Skills in:

- Configuring DNS zones and conditional forwarding
- Configuring traffic management by using DNS solutions
- Configuring DNS for hybrid networks

- Configuring appropriate DNS records
- Configuring DNSSEC on Route 53
- Configuring DNS within a centralized or distributed network architecture
- Configuring DNS monitoring and logging on Route 53

Task 2.4: Automate and configure network infrastructure

Knowledge of:

- Infrastructure as code (IaC) (for example, AWS Cloud Development Kit [AWS CDK], AWS CloudFormation, AWS CLI, AWS SDK, APIs)
- Event-driven network automation
- Common problems of using hardcoded instructions in IaC templates when provisioning cloud networking resources

Skills in:

- Creating and managing repeatable network configurations
- Integrating event-driven networking functions
- Integrating hybrid network automation options with AWS native IaC
- Eliminating risk and achieving efficiency in a cloud networking environment while maintaining the lowest possible cost
- Automating the process of optimizing cloud network resources with IaC

Content Domain 3: Network Management and Operation

Tasks

Task 3.1: Maintain routing and connectivity on AWS and hybrid networks

Knowledge of:

- Industry-standard routing protocols that are used in AWS hybrid networks (for example, BGP over Direct Connect)
- Connectivity methods for AWS and hybrid networks (for example, Direct Connect gateway, Transit Gateway, VIFs)
- How limits and quotas affect AWS networking services (for example, bandwidth limits, route limits)
- Available private and public access methods for custom services (for example, PrivateLink, VPC peering)
- Available inter-Regional and intra-Regional communication patterns

Skills in:

- Managing routing protocols for AWS and hybrid connectivity options (for example, over a Direct Connect connection, VPN)
- Maintaining private access to custom services (for example, PrivateLink, VPC peering)
- Using route tables to direct traffic appropriately (for example, automatic propagation, BGP)
- Setting up private access or public access to AWS services (for example, Direct Connect, VPN)
- Optimizing routing over dynamic and static routing protocols (for example, summarizing routes, CIDR overlap)

Task 3.2: Monitor and analyze network traffic to troubleshoot and optimize connectivity patterns

Knowledge of:

- Network performance metrics and reachability constraints (for example, routing, packet size)
- Appropriate logs and metrics to assess network performance and reachability issues (for example, packet loss)
- Tools to collect and analyze logs and metrics (for example, CloudWatch, VPC Flow Logs, VPC Traffic Mirroring)
- Tools to analyze routing patterns and issues (for example, Reachability Analyzer, Transit Gateway Network Manager)

Skills in:

- Analyzing tool output to assess network performance and troubleshoot connectivity (for example, VPC Flow Logs, Amazon CloudWatch Logs)
- Mapping or understanding network topology (for example, Transit Gateway Network Manager)
- Analyzing packets to identify issues in packet shaping (for example, VPC Traffic Mirroring)
- Troubleshooting connectivity issues that are caused by network misconfiguration (for example, Reachability Analyzer)
- Verifying that a network configuration meets network design requirements (for example, Reachability Analyzer)
- Automating the verification of connectivity intent as a network configuration changes (for example, Reachability Analyzer)
- Troubleshooting packet size mismatches in a VPC to restore network connectivity

Task 3.3: Optimize AWS networks for performance, reliability, and cost-effectiveness

Knowledge of:

- Situations in which a VPC peer or a transit gateway are appropriate
- Different methods to reduce bandwidth utilization (for example, unicast compared with multicast, CloudFront)
- Cost-effective connectivity options for data transfer between a VPC and on-premises environments
- Different types of network interfaces on AWS
- High-availability features in Route 53 (for example, DNS load balancing using health checks with latency and weighted record sets)
- Availability of options from Route 53 that provide reliability
- Load balancing and traffic distribution patterns
- VPC subnet optimization
- Frame size optimization for bandwidth across different connection types

Skills in:

- Optimizing for network throughput

- Selecting the right network interface for the best performance (for example, elastic network interface, Elastic Network Adapter [ENA], Elastic Fabric Adapter [EFA])
- Choosing between VPC peering, proxy patterns, or a transit gateway connection based on analysis of the network requirements provided
- Implementing a solution on an appropriate network connectivity service (for example, VPC peering, Transit Gateway, VPN connection) to meet network requirements
- Implementing a multicast capability within a VPC and on-premises environments
- Creating Route 53 public hosted zones and private hosted zones and records to optimize application availability (for example, private zonal DNS entry to route traffic to multiple Availability Zones)
- Updating and optimizing subnets for auto scaling configurations to support increased application load
- Updating and optimizing subnets to prevent the depletion of available IP addresses within a VPC (for example, secondary CIDR)
- Configuring jumbo frame support across connection types
- Optimizing network connectivity by using Global Accelerator to improve network performance and application availability

Content Domain 4: Network Security, Compliance, and Governance

Tasks

Task 4.1: Implement and maintain network features to meet security and compliance needs and requirements

Knowledge of:

- Different threat models based on application architecture
- Common security threats
- Mechanisms to secure different application flows

- AWS network architecture that meets security and compliance requirements

Skills in:

- Securing inbound traffic flows into AWS (for example, AWS WAF, AWS Shield, Network Firewall)
- Securing outbound traffic flows from AWS (for example, Network Firewall, proxies, Gateway Load Balancers)
- Securing inter-VPC traffic within an account or across multiple accounts (for example, security groups, network ACLs, VPC endpoint policies)
- Implementing an AWS network architecture to meet security and compliance requirements (for example, untrusted network, perimeter VPC, three-tier architecture)
- Developing a threat model and identifying appropriate mitigation strategies for a given network architecture
- Testing compliance with the initial requirements (for example, failover test, resiliency)
- Automating security incident reporting and alerting using AWS

Task 4.2: Validate and audit security by using network monitoring and logging services

Knowledge of:

- Network monitoring and logging services that are available in AWS (for example, CloudWatch, AWS CloudTrail, VPC Traffic Mirroring, VPC Flow Logs, Transit Gateway Network Manager)
- Alert mechanisms (for example, CloudWatch alarms)
- Log creation in different AWS services (for example, VPC flow logs, load balancer access logs, CloudFront access logs)
- Log delivery mechanisms (for example, Amazon Kinesis, Route 53, CloudWatch)
- Mechanisms to audit network security configurations (for example, security groups, AWS Firewall Manager, AWS Trusted Advisor)

Skills in:

- Creating and analyzing a VPC flow log (including base and extended fields of flow logs)
- Creating and analyzing network traffic mirroring (for example, using VPC Traffic Mirroring)

- Implementing automated alarms by using CloudWatch
- Implementing customized metrics by using CloudWatch
- Correlating and analyzing information across single or multiple AWS log sources
- Implementing log delivery solutions
- Implementing a network audit strategy across single or multiple AWS network services and accounts (for example, Firewall Manager, security groups, network ACLs)

Task 4.3: Implement and maintain confidentiality of data and communications of the network

Knowledge of:

- Network encryption options that are available on AWS
- VPN connectivity over Direct Connect
- Encryption methods for data in transit (for example, IPsec)
- Network encryption under the AWS shared responsibility model
- Security methods for DNS communications (for example, DNSSEC)

Skills in:

- Implementing network encryption methods to meet application compliance requirements (for example, IPsec, TLS)
- Implementing encryption solutions to secure data in transit (for example, CloudFront, Application Load Balancers and Network Load Balancers, VPN over Direct Connect, AWS managed databases, Amazon S3, custom solutions on Amazon EC2, Transit Gateway)
- Implementing a certificate management solution by using a certificate authority (for example, ACM, AWS Private Certificate Authority [ACM PCA])
- Implementing secure DNS communications

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Advanced Networking - Specialty (ANS-C01) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

Application Integration

- Amazon EventBridge
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)

Compute

- Amazon EC2
- Amazon EC2 Auto Scaling
- AWS Lambda

Containers

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon Elastic Kubernetes Service (Amazon EKS)
- AWS Fargate

Cost Management

- AWS Cost Explorer

Front-End Web and Mobile

- Amazon API Gateway

Management and Governance

- AWS Auto Scaling
- AWS CLI
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Config
- AWS Control Tower
- AWS Health Dashboard
- AWS Management Console
- AWS Organizations
- AWS Trusted Advisor
- AWS Well-Architected Tool

Networking and Content Delivery

- Amazon API Gateway
- AWS App Mesh
- AWS Client VPN
- AWS Cloud Map
- Amazon CloudFront
- AWS Direct Connect
- Elastic Load Balancing (ELB)
- AWS Global Accelerator

- AWS PrivateLink
- Amazon Route 53
- AWS Site-to-Site VPN
- AWS Transit Gateway
- Amazon VPC

Security, Identity, and Compliance

- AWS Firewall Manager
- AWS Identity and Access Management (IAM)
- AWS Network Firewall
- AWS Resource Access Manager (AWS RAM)
- AWS Shield
- AWS WAF

Serverless

- Amazon API Gateway
- Amazon EventBridge
- AWS Fargate
- AWS Lambda
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- Amazon Simple Storage Service (Amazon S3)

Storage

- Amazon S3

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the AWS Certified Advanced Networking - Specialty (ANS-C01) exam. This list is non-exhaustive and is subject to

change. AWS offerings that are entirely unrelated to the target job roles for the exam are excluded from this list.

Topics

Analytics

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

AR and VR

- Amazon Sumerian

Blockchain

- Amazon Managed Blockchain
- Amazon Quantum Ledger Database (Amazon QLDB)

Developer Tools

- AWS Device Farm
- AWS X-Ray

Robotics

- AWS RoboMaker

Satellite

- AWS Ground Station

Survey

How useful was this exam guide? Let us know by taking our survey.