AWS SAA-C03

AWS-SAA CERTIFICATION QUESTIONS & ANSWERS

Exam Summary – Syllabus – Questions

SAA-C03

AWS Certified Solutions Architect - Associate
65 Questions Exam – 720 / 1000 Cut Score – Duration of 130 minutes

www.VMExam.com

Table of Contents

Know Your SAA-C03 Certification Well:

The SAA-C03 is best suitable for candidates who want to gain knowledge in the AWS Architect. Before you start your SAA-C03 preparation you may struggle to get all the crucial AWS-SAA materials like SAA-C03 syllabus, sample questions, study guide.

But don't worry the SAA-C03 PDF is here to help you prepare in a stress free manner.

The PDF is a combination of all your queries like-

- What is in the SAA-C03 syllabus?
- How many questions are there in the SAA-C03 exam?
- Which Practice test would help me to pass the SAA-C03 exam at the first attempt?

Passing the SAA-C03 exam makes you AWS Certified Solutions Architect - Associate. Having the AWS-SAA certification opens multiple opportunities for you. You can grab a new job, get a higher salary or simply get recognition within your current organization.

AWS SAA-C03 Solutions Architect Associate Certification Details:

Exam Name	AWS Solutions Architect Associate (AWS-SAA)
Exam Code	SAA-C03
Exam Price	$150 USD
Duration	130 minutes
Number of Questions	65
Passing Score	720 / 1000
Recommended Training / Books	**AWS Technical Essentials** **AWS Power Hour: Architecting on-demand** **Architecting on AWS**
Schedule Exam	**AWS Certification**
Sample Questions	**AWS SAA-C03 Sample Questions**
Recommended Practice	**AWS Certified Solutions Architect - Associate Practice Test**

SAA-C03 Syllabus:

Section	Objectives
Design Secure Architectures - 30%	
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 Single Sign-On [AWS SSO]) • 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
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, Amazon Cognito, Amazon GuardDuty, Amazon Macie) • Threat vectors external to AWS (for example, DDoS, SQL injection) Skills in:

Section	Objectives
	• 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, AWS SSO, AWS Secrets Manager) • Securing external network connections to and from the AWS Cloud (for example, VPN, AWS Direct Connect)
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
Design Resilient Architectures - 26%	
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], Secrets Manager) • Caching strategies • Design principles for microservices (for example, stateless workloads compared with stateful workloads) • Event-driven architectures • Horizontal scaling and vertical scaling

Section	Objectives
	• 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) • 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
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 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

Section	Objectives
	• Failover strategies • Immutable infrastructure • Load balancing concepts (for example, Application Load Balancer) • 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
Design High-Performing Architectures - 24%	
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:

Section	Objectives
	• Determining storage services and configurations that meet performance demands • Determining storage services that can scale to accommodate future needs
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
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)

Section	Objectives
	• 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
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
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

Section	Objectives
	• 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)
Design Cost-Optimized Architectures - 20%	
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:

Section	Objectives
	• 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
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, 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)

Section	Objectives
	• 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
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, 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

Section	Objectives
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, 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)

AWS SAA-C03 Sample Questions:

Question: 1

An analytics company is planning to offer a web analytics service to its users. The service will require that the users' webpages include a JavaScript script that makes authenticated GET requests to the company's Amazon S3 bucket.

What must a solutions architect do to ensure that the script will successfully execute?

a) Enable cross-origin resource sharing (CORS) on the S3 bucket.
b) Enable S3 Versioning on the S3 bucket.
c) Provide the users with a signed URL for the script.
d) Configure an S3 bucket policy to allow public execute privileges.

Answer: a

Question: 2

A company uses Amazon EC2 Reserved Instances to run its data processing workload. The nightly job typically takes 7 hours to run and must finish within a 10-hour time window.

The company anticipates temporary increases in demand at the end of each month that will cause the job to run over the time limit with the capacity of the current resources.

Once started, the processing job cannot be interrupted before completion. The company wants to implement a solution that would provide increased resource capacity as cost-effectively as possible.

What should a solutions architect do to accomplish this?

a) Deploy On-Demand Instances during periods of high demand.
b) Create a second EC2 reservation for additional instances
c) Deploy Spot Instances during periods of high demand.
d) Increase the EC2 instance size in the EC2 reservation to support the increased workload.

Answer: a

Question: 3

A company's security team requires that all data stored in the cloud be encrypted at rest at all times using encryption keys stored on premises.

Which encryption options meet these requirements?

(Select TWO.)

a) Use server-side encryption with Amazon S3 managed encryption keys (SSE-S3)
b) Use server-side encryption with AWS KMS managed encryption keys (SSE-KMS).
c) Use server-side encryption with customer-provided encryption keys (SSE-C).
d) Use client-side encryption to provide at-rest encryption.
e) Use an AWS Lambda function invoked by Amazon S3 events to encrypt the data using the customer's keys.

Answer: c, d

Question: 4

A company has a two-tier application architecture that runs in public and private subnets. Amazon EC2 instances running the web application are in the public subnet and an EC2 instance for the database runs on the private subnet. The web application instances and the database are running in a single Availability Zone (AZ).

Which combination of steps should a solutions architect take to provide high availability for this architecture?

(Select TWO.)

a) Create new public and private subnets in the same AZ.
b) Create an Amazon EC2 Auto Scaling group and Application Load Balancer spanning multiple AZs for the web application instances.
c) Add the existing web application instances to an Auto Scaling group behind an Application Load Balancer.
d) Create new public and private subnets in a new AZ. Create a database using an EC2 instance in the public subnet in the new AZ. Migrate the old database contents to the new database.
e) Create new public and private subnets in the same VPC, each in a new AZ. Create an Amazon RDS Multi-AZ DB instance in the private subnets. Migrate the old database contents to the new DB instance.

Answer: b, e

Question: 5

A company runs an online voting system for a weekly live television program. During broadcasts, users submit hundreds of thousands of votes within minutes to a front-end fleet of Amazon EC2 instances that run in an Auto Scaling group. The EC2 instances write the votes to an Amazon RDS database.

However, the database is unable to keep up with the requests that come from the EC2 instances. A solutions architect must design a solution that processes the votes in the most efficient manner and without downtime.

Which solution meets these requirements?

a) Migrate the front-end application to AWS Lambda. Use Amazon API Gateway to route user requests to the Lambda functions.
b) Scale the database horizontally by converting it to a Multi-AZ deployment. Configure the front-end application to write to both the primary and secondary DB instances
c) Configure the front-end application to send votes to an Amazon Simple Queue Service (Amazon SQS) queue. Provision worker instances to read the SQS queue and write the vote information to the database.
d) Use Amazon EventBridge (Amazon CloudWatch Events) to create a scheduled event to re-provision the database with larger, memory optimized instances during voting periods. When voting ends, re-provision the database to use smaller instances.

Answer: c

Question: 6

A company plans to run a monitoring application on an Amazon EC2 instance in a VPC. Connections are made to the EC2 instance using the instance's private IPv4 address.

A solutions architect needs to design a solution that will allow traffic to be quickly directed to a standby EC2 instance if the application fails and becomes unreachable.

Which approach will meet these requirements?

a) Deploy an Application Load Balancer configured with a listener for the private IP address and register the primary EC2 instance with the load balancer. Upon failure, de-register the instance and register the standby EC2 instance.
b) Configure a custom DHCP option set. Configure DHCP to assign the same private IP address to the standby EC2 instance when the primary EC2 instance fails.
c) Attach a secondary elastic network interface to the EC2 instance configured with the private IP address. Move the network interface to the standby EC2 instance if the primary EC2 instance becomes unreachable.
d) Associate an Elastic IP address with the network interface of the primary EC2 instance. Disassociate the Elastic IP from the primary instance upon failure and associate it with a standby EC2 instance.

Answer: c

Question: 7

A company runs a public-facing three-tier web application in a VPC across multiple Availability Zones.

Amazon EC2 instances for the application tier running in private subnets need to download software patches from the internet. However, the EC2 instances cannot be directly accessible from the internet.

Which actions should be taken to allow the EC2 instances to download the needed patches?

(Select TWO.)

a) Configure a NAT gateway in a public subnet.
b) Define a custom route table with a route to the NAT gateway for internet traffic and associate it with the private subnets for the application tier.
c) Assign Elastic IP addresses to the EC2 instances.
d) Define a custom route table with a route to the internet gateway for internet traffic and associate it with the private subnets for the application tier.
e) Configure a NAT instance in a private subnet.

Answer: a, b

Question: 8

A website runs a custom web application that receives a burst of traffic each day at noon. The users upload new pictures and content daily, but have been complaining of timeouts.

The architecture uses Amazon EC2 Auto Scaling groups, and the application consistently takes 1 minute to initiate upon boot up before responding to user requests.

How should a solutions architect redesign the architecture to better respond to changing traffic?

a) Configure a Network Load Balancer with a slow start configuration
b) Configure Amazon ElastiCache for Redis to offload direct requests from the EC2 instances.
c) Configure Amazon CloudFront to use an Application Load Balancer as the origin.
d) Configure an Auto Scaling step scaling policy with an EC2 instance warmup condition.

Answer: d

Question: 9

An application running on AWS uses an Amazon Aurora Multi-AZ DB cluster deployment for its database.

When evaluating performance metrics, a solutions architect discovered that the database reads are causing high I/O and adding latency to the write requests against the database.

What should the solutions architect do to separate the read requests from the write requests?

a) Enable read-through caching on the Aurora database.
b) Update the application to read from the Multi-AZ standby instance.
c) Create a second Aurora database and link it to the primary database as a read replica.
d) Create an Aurora replica and modify the application to use the appropriate endpoints.

Answer: d

Question: 10

A solutions architect wants to design a solution to save costs for Amazon EC2 instances that do not need to run during a 2-week company shutdown.

The applications running on the EC2 instances store data in instance memory that must be present when the instances resume operation.

Which approach should the solutions architect recommend to shut down and resume the EC2 instances?

a) Modify the application to store the data on instance store volumes. Reattach the volumes while restarting them.
b) Run the applications on EC2 instances enabled for hibernation. Hibernate the instances before the 2-week company shutdown.
c) Snapshot the EC2 instances before stopping them. Restore the snapshot after restarting the instances.
d) Note the Availability Zone for each EC2 instance before stopping it. Restart the instances in the same Availability Zones after the 2-week company shutdown.

Answer: b

Study Guide to Crack AWS-SAA-C03 Exam:

- Getting details of the SAA-C03 syllabus, is the first step of a study plan. This pdf is going to be of ultimate help. Completion of the syllabus is must to pass the SAA-C03 exam.
- Making a schedule is vital. A structured method of preparation leads to success. A candidate must plan his schedule and follow it rigorously to attain success.
- Joining the AWS provided training for SAA-C03 exam could be of much help. If there is specific training for the exam, you can discover it from the link above.
- Read from the SAA-C03 sample questions to gain your idea about the actual exam questions. In this PDF useful sample questions are provided to make your exam preparation easy.
- Practicing on SAA-C03 practice tests is must. Continuous practice will make you an expert in all syllabus areas.

Reliable Online Practice Test for SAA-C03 Certification

Make VMExam.com your best friend during your AWS Solutions Architect Associate exam preparation. We provide authentic practice tests for the SAA-C03 exam. Experts design these online practice tests, so we can offer you an exclusive experience of taking the actual SAA-C03 exam. We guarantee you 100% success in your first exam attempt if you continue practicing regularly. Don't bother if you don't get 100% marks in initial practice exam attempts. Just utilize the result section to know your strengths and weaknesses and prepare according to that until you get 100% with our practice tests. Our evaluation makes you confident, and you can score high in the SAA-C03 exam.

Start Online practice of SAA-C03 Exam by visiting URL

https://www.vmexam.com/aws/saa-c03-aws-solutions-architect-associate