Free Questions for SAP-C02

Shared by Alexander on 16-04-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

A company is storing data on premises on a Windows file server. The company produces 5 GB of new data daily.

The company migrated part of its Windows-based workload to AWS and needs the data to be available on a file system in the cloud. The company already has established an AWS Direct Connect connection between the on-premises network and AWS.

Which data migration strategy should the company use?

Options:

A- Use the file gateway option in AWS Storage Gateway to replace the existing Windows file server, and point the existing file share to the new file gateway.
B- Use AWS DataSync to schedule a daily task to replicate data between the on-premises Windows file server and Amazon FSx.
C- Use AWS Data Pipeline to schedule a daily task to replicate data between the on-premises Windows file server and Amazon Elastic File System (Amazon EFS).
D- Use AWS DataSync to schedule a daily task lo replicate data between the on-premises Windows file server and Amazon Elastic File System (Amazon EFS),

Answer:

B

Explanation:

https://aws.amazon.com/storagegateway/file/

https://docs.aws.amazon.com/fsx/latest/WindowsGuide/migrate-files-to-fsx-datasync.html

https://docs.aws.amazon.com/systems-manager/latest/userguide/prereqs-operating-systems.html#prereqs-os-windows-server

Question 2

Question Type: MultipleChoice

A company is rearchitecting its applications to run on AWS. The company's infrastructure includes

multiple Amazon EC2 instances. The company's development team needs different levels of access. The company wants to implement a policy that requires all Windows EC2 instances to be joined to an Active Directory domain on AWS. The company also wants to Implement enhanced security processes such as multi-factor authentication (MFA). The company wants to use managed AWS services wherever possible.

Which solution will meet these requirements?

Options:

A- Create an AWS Directory Service for Microsoft Active Directory implementation. Launch an Amazon Workspace. Connect to and use the Workspace for domain security configuration tasks.
B- Create an AWS Directory Service for Microsoft Active Directory implementation. Launch an EC2 instance. Connect to and use the EC2 instance for domain security configuration tasks.
C- Create an AWS Directory Service Simple AD implementation. Launch an EC2 instance. Connect to and use the EC2 instance for domain security configuration tasks.
D- Create an AWS Directory Service Simple AD implementation. Launch an Amazon Workspace. Connect to and use the Workspace for domain security configuration tasks.

Answer:

A

Explanation:

A is the correct answer because it uses AWS Directory Service for Microsoft Active Directory to join the Windows EC2 instances to an Active Directory domain on AWS and enable MFA. AWS Directory Service for Microsoft Active Directory, also known as AWS Managed Microsoft AD, is a fully managed service that is powered by Windows Server 2019. It allows you to run directory-aware workloads in the AWS Cloud, including Microsoft SharePoint and custom .NET and SQL Server-based applications. You can also configure a trust relationship between AWS Managed Microsoft AD in the AWS Cloud and your existing on-premises Microsoft Active Directory. AWS Managed Microsoft AD supports MFA by integrating with your existing RADIUS-based MFA infrastructure. To join the Windows EC2 instances to an Active Directory domain on AWS, you can use an Amazon Workspace, which is a fully managed, secure desktop computing service that runs on AWS. You can connect to and use the Workspace for domain security configuration tasks.
Reference:

https://docs.aws.amazon.com/directoryservice/latest/admin-guide/directory_microsoft_ad.html

https://docs.aws.amazon.com/directoryservice/latest/admin-guide/ms_ad_join_instance.html

https://docs.aws.amazon.com/workspaces/latest/adminguide/amazon-workspaces.html

Question 3

Question Type: MultipleChoice

An application is using an Amazon RDS for MySQL Multi-AZ DB instance in the us-east-1 Region. After a failover test, the application lost the connections to the database and could not re-establish the connections. After a restart of the application, the application re-established the connections.

A solutions architect must implement a solution so that the application can re-establish connections to the database without requiring a restart.

Which solution will meet these requirements?

Options:

A- Create an Amazon Aurora MySQL Serverless v1 DB instance. Migrate the RDS DB instance to the Aurora Serverless v1 DB instance. Update the connection settings in the application to point to the Aurora reader endpoint.
B- Create an RDS proxy. Configure the existing RDS endpoint as a target. Update the connection settings in the application to point to the RDS proxy endpoint.
C- Create a two-node Amazon Aurora MySQL DB cluster. Migrate the RDS DB instance to the Aurora DB cluster. Create an RDS proxy. Configure the existing RDS endpoint as a target. Update the connection settings in the application to point to the RDS proxy endpoint.
D- Create an Amazon S3 bucket. Export the database to Amazon S3 by using AWS Database Migration Service (AWS DMS). Configure Amazon Athena to use the S3 bucket as a data store. Install the latest Open Database Connectivity (ODBC) driver for the application. Update the connection settings in the application to point to the Athena endpoint

Answer:

B

Explanation:

Amazon RDS Proxy is a fully managed database proxy service for Amazon Relational Database Service (RDS) that makes applications more scalable, resilient, and secure. It allows applications to pool and share connections to an RDS database, which can help reduce database connection overhead, improve scalability, and provide automatic failover and high availability.

Question 4

Question Type: MultipleChoice

An environmental company is deploying sensors in major cities throughout a country to measure air quality The sensors connect to AWS loT Core to ingest timesheets data readings. The company stores the data in Amazon DynamoDB

For business continuity the company must have the ability to ingest and store data in two AWS Regions

Which solution will meet these requirements?

Options:

A- Create an Amazon Route 53 alias failover routing policy with values for AWS loT Core data endpoints in both Regions Migrate data to Amazon Aurora global tables
B- Create a domain configuration for AWS loT Core in each Region Create an Amazon Route 53 latency-based routing policy Use AWS loT Core data endpoints in both Regions as values Migrate the data to Amazon MemoryDB for Radis and configure Cross-Region replication
C- Create a domain configuration for AWS loT Core in each. Region Create an AmazonRoute 53 health check that evaluates domain configuration health Create a failover routing policy with values for the domain name from the AWS loT Core domain configurations Update the DynamoDB table to a global table
D- Create an Amazon Route 53 latency-based routing policy. Use AWS loT Core data endpoints in both Regions as values. Configure DynamoDB streams and Cross-Region data replication

Answer:

C

Explanation:

https://aws.amazon.com/solutions/implementations/disaster-recovery-for-aws-iot/

Question 5

Question Type: MultipleChoice

A company has several AWS Lambda functions written in Python. The functions are deployed with the .zip package deployment type. The functions use a Lambda layer that contains common

libraries and packages in a .zip file. The Lambda .zip packages and the Lambda layer .zip file are stored in an Amazon S3 bucket. The company must implement automatic scanning of the Lambda functions and the Lambda layer to identify CVEs. A subset of the Lambda functions must receive automated code scans to detect potential data leaks and other vulnerabilities. The code scans must occur only for selected Lambda functions, not all the Lambda functions. Which combination of actions will meet these requirements? (Select THREE.)

Options:

A- Activate Amazon Inspector. Start automated CVE scans.
B- Activate Lambda standard scanning and Lambda code scanning in Amazon Inspector.
C- Enable Amazon GuardDuty. Enable the Lambda Protection feature in GuardDuty.
D- Enable scanning in the Monitor settings of the Lambda functions that need code scans.
E- Tag Lambda functions that do not need code scans. In the tag, include a key of InspectorCodeExclusion and a value of LambdaCodeScanning.F. Use Amazon Inspector to scan the S3 bucket that contains the Lambda .zip packages and the Lambda layer .zip file for code scans.

Answer:

B, D, E

Explanation:

Comprehensive and Detailed Explanation From Exact Extract: The company requires two different types of security analysis: vulnerability scanning for CVEs and code-level scanning for sensitive data exposure. Amazon Inspector provides both Lambda Standard Scanning and Lambda Code Scanning. These capabilities allow Inspector to examine Lambda deployment packages and Lambda layers for software vulnerabilities, and also perform static code analysis to detect insecure coding patterns. To meet the requirement that only certain Lambda functions receive deeper code scanning, Inspector allows enabling or disabling code scanning at the individual Lambda function level through the function's Monitor settings. Inspector additionally supports resource exclusion based on tagging, allowing administrators to prevent specific Lambda functions from participating in code scans. Tag-based exclusion satisfies the requirement to scan only a subset of Lambda functions.

Option B is required because enabling both Lambda standard scanning and Lambda code scanning activates the functionality necessary for CVE detection and code security analysis. Option D is required because enabling scanning at the function level ensures that code scanning runs only for selected Lambda functions. Option E is required because resource exclusion using tags allows the administrator to exclude non-target Lambda functions from code scanning automatically.

Question 6

Question Type: MultipleChoice

A company runs an application in an on-premises data center. The application gives users the ability to upload media files. The files persist in a file server. The web application has many users. The application server is overutilized, which causes data uploads to fail occasionally. The company frequently adds new storage to the file server. The company wants to resolve these challenges by migrating the application to AWS.

Users from across the United States and Canada access the application. Only authenticated usersshould have the ability to access the application to upload files. The company will consider a solution that refactors the application, and the company needs to accelerate application development.

Which solution will meet these requirements with the LEAST operational overhead?

Options:

A- Use AWS Application Migration Service to migrate the application server to Amazon EC2 instances. Create an Auto Scaling group for the EC2 instances. Use an Application Load Balancer to distribute the requests. Modify the application to use Amazon S3 to persist the files. Use Amazon Cognito to authenticate users.
B- Use AWS Application Migration Service to migrate the application server to Amazon EC2 instances. Create an Auto Scaling group for the EC2 instances. Use an Application Load Balancer to distribute the requests. Set up AWS IAM Identity Center (AWS Single Sign-On) to give users the ability to sign in to the application. Modify the application to use Amazon S3 to persist the files.
C- Create a static website for uploads of media files. Store the static assets in Amazon S3. Use AWS AppSync to create an API. Use AWS Lambda resolvers to upload the media files to Amazon S3. Use Amazon Cognito to authenticate users.
D- Use AWS Amplify to create a static website for uploads of media files. Use Amplify Hosting to serve the website through Amazon CloudFront. Use Amazon S3 to store the uploaded media files. Use Amazon Cognito to authenticate users.

Answer:

D

Explanation:

The company should use AWS Amplify to create a static website for uploads of media files. The company should use Amplify Hosting to serve the website through Amazon CloudFront. The

company should use Amazon S3 to store the uploaded media files. The company should use Amazon Cognito to authenticate users. This solution will meet the requirements with the least operational overhead because AWS Amplify is a complete solution that lets frontend web and mobile developers easily build, ship, and host full-stack applications on AWS, with the flexibility to leverage the breadth of AWS services as use cases evolve.No cloud expertise needed1. By using AWS Amplify, the company can refactor the application to a serverless architecture that reduces operational complexity and costs. AWS Amplify offers the following features and benefits:

Amplify Studio: A visual interface that enables you to build and deploy a full-stack app quickly, including frontend UI and backend.

Amplify CLI: A local toolchain that enables you to configure and manage an app backend with just a few commands.

Amplify Libraries: Open-source client libraries that enable you to build cloud-powered mobile and web apps.

Amplify UI Components: Open-source design system with cloud-connected components for building feature-rich apps fast.

Amplify Hosting: Fully managed CI/CD and hosting for fast, secure, and reliable static and server-side rendered apps.

By using AWS Amplify to create a static website for uploads of media files, the company can leverage Amplify Studio to visually build a pixel-perfect UI and connect it to a cloudbackend in clicks. By using Amplify Hosting to serve the website through Amazon CloudFront, the company can easily deploy its web app or website to the fast, secure, and reliable AWS content delivery network (CDN), with hundreds of points of presence globally.By using Amazon S3 to store the uploaded media files, the company can benefit from a highly scalable, durable, and cost-effective object storage service that can handle any amount of data2.By using Amazon Cognito to authenticate users, the company can add user sign-up, sign-in, and access control to its web app with a fully managed service that scales to support millions of users3.

The other options are not correct because:

Using AWS Application Migration Service to migrate the application server to Amazon EC2 instances would not refactor the application or accelerate development. AWS Application Migration Service (AWS MGN) is a service that enables you to migrate physical servers, virtual machines (VMs), or cloud servers from any source infrastructure to AWS without requiring agents or specialized tools. However, this would not address the challenges of overutilization and data uploads failures. It would also not reduce operational overhead or costs compared to a serverless architecture.

Creating a static website for uploads of media files and using AWS AppSync to create an API would not be as simple or fast as using AWS Amplify. AWS AppSync is a service that enables you to create flexible APIs for securely accessing, manipulating, and combining data from one or more data sources. However, this would require more configuration and management than using Amplify Studio and Amplify Hosting. It would also not provide authentication features like Amazon

Cognito.

Setting up AWS IAM Identity Center (AWS Single Sign-On) to give users the ability to sign in to the application would not be as suitable as using Amazon Cognito. AWS Single Sign-On (AWS SSO) is a service that enables you to centrally manage SSO access and user permissions across multiple AWS accounts and business applications. However, this service is designed for enterprise customers who need to manage access for employees or partners across multiple resources. It is not intended for authenticating end users of web or mobile apps.

https://aws.amazon.com/amplify/

https://aws.amazon.com/s3/

https://aws.amazon.com/cognito/

https://aws.amazon.com/mgn/

https://aws.amazon.com/appsync/

https://aws.amazon.com/single-sign-on/

Question 7

Question Type: MultipleChoice

A solutions architect is designing the data storage and retrieval architecture for a new application that a company will be launching soon. The application is designed to ingest millions of small records per minute from devices all around the world. Each record is less than 4 KB in size and needs to be stored in a durable location where it can be retrieved with low latency. The data is ephemeral and the company is required to store the data for 120 days only, after which the data can be deleted.

The solutions architect calculates that, during the course of a year, the storage requirements would be about 10-15 TB.

Which storage strategy is the MOST cost-effective and meets the design requirements?

Options:

A- Design the application to store each incoming record as a single .csv file in an Amazon S3 bucket to allow for indexed retrieval. Configure a lifecycle policy to delete data older than 120 days.

B- Design the application to store each incoming record in an Amazon DynamoDB table properly configured for the scale. Configure the DynamoOB Time to Live (TTL) feature to delete records

older than 120 days.
C- Design the application to store each incoming record in a single table in an Amazon RDS MySQL database. Run a nightly cron job that executes a query to delete any records older than 120 days.
D- Design the application to batch incoming records before writing them to an Amazon S3 bucket. Update the metadata for the object to contain the list of records in the batch and use the Amazon S3 metadata search feature to retrieve the data. Configure a lifecycle policy to delete the data after 120 days.

Answer:

B

Explanation:

DynamoDB with TTL, cheaper for sustained throughput of small items + suited for fast retrievals. S3 cheaper for storage only, much higher costs with writes. RDS not designed for this use case.

Question 8

Question Type: MultipleChoice

A company wants to run a custom network analysis software package to inspect traffic as traffic leaves and enters a VPC. The company has deployed the solution by using AWS Cloud Formation on three Amazon EC2 instances in an Auto Scaling group. All network routing has been established to direct traffic to the EC2 instances.

Whenever the analysis software stops working, the Auto Scaling group replaces an instance. The network routes are not updated when the instance replacement occurs.

Which combination of steps will resolve this issue? {Select THREE.)

Options:

A- Create alarms based on EC2 status check metrics that will cause the Auto Scaling group to replace the failed instance.
B- Update the Cloud Formation template to install the Amazon CloudWatch agent on the EC2 instances. Configure the CloudWatch agent to send process metrics for the application.
C- Update the Cloud Formation template to install AWS Systems Manager Agent on the EC2 instances. Configure Systems Manager Agent to send process metrics for the application.
D- Create an alarm for the custom metric in Amazon CloudWatch for the failure scenarios.

Configure the alarm to publish a message to an Amazon Simple Notification Service {Amazon SNS) topic.
E- Create an AWS Lambda function that responds to the Amazon Simple Notification Service (Amazon SNS) message to take the instance out of service. Update the network routes to point to the replacement instance.
F- In the Cloud Formation template, write a condition that updates the network routes when a replacement instance is launched.

Answer:

B, D, E

Question 9

Question Type: MultipleChoice

An adventure company has launched a new feature on its mobile app. Users can use the feature to upload their hiking and ratting photos and videos anytime. The photos and videos are stored in Amazon S3 Standard storage in an S3 bucket and are served through Amazon CloudFront.

The company needs to optimize the cost of the storage. A solutions architect discovers that most of the uploaded photos and videos are accessed infrequently after 30 days. However, some of the uploaded photos and videos are accessed frequently after 30 days. The solutions architect needs to implement a solution that maintains millisecond retrieval availability of the photos and videos at the lowest possible cost.

Which solution will meet these requirements?

Options:

A- Configure S3 Intelligent-Tiering on the S3 bucket.
B- Configure an S3 Lifecycle policy to transition image objects and video objects from S3 Standard to S3 Glacier Deep Archive after 30 days.
C- Replace Amazon S3 with an Amazon Elastic File System (Amazon EFS) file system that is mounted on Amazon EC2 instances.
D- Add a Cache-Control: max-age header to the S3 image objects and S3 video objects. Set the header to 30 days.

Answer:

A

Explanation:

Amazon S3 Intelligent-Tiering is a storage class that automatically moves objects between two access tiers based on changing access patterns. Objects that are accessed frequently are stored in the frequent access tier and objects that are accessed infrequently are stored in the infrequent access tier. This allows for cost optimization without requiring manual intervention. This makes it an ideal solution for the scenario described, as it can automatically move objects that are infrequently accessed after 30 days to a lower-cost storage tier while still maintaining millisecond retrieval availability.

Question 10

Question Type: MultipleChoice

A company is planning to migrate its on-premises transaction-processing application to AWS. The application runs inside Docker containers that are hosted on VMS in the company's data center. The Docker containers have shared storage where the application records transaction data.

The transactions are time sensitive. The volume of transactions inside the application is unpredictable. The company must implement a low-latency storage solution that will automatically scale throughput to meet increased demand. The company cannot develop the application further and cannot continue to administer the Docker hosting environment.

How should the company migrate the application to AWS to meet these requirements?

Options:

A- Migrate the containers that run the application to Amazon Elastic Kubernetes Service (Amazon EKS). Use Amazon S3 to store the transaction data that the containers share.
B- Migrate the containers that run the application to AWS Fargate for Amazon Elastic Container Service (Amazon ECS). Create an Amazon Elastic File System (Amazon EFS) file system. Create a Fargate task definition. Add a volume to the task definition to point to the EFS file system
C- Migrate the containers that run the application to AWS Fargate for Amazon Elastic Container Service (Amazon ECS). Create an Amazon Elastic Block Store (Amazon EBS) volume. Create a Fargate task definition. Attach the EBS volume to each running task.
D- Launch Amazon EC2 instances. Install Docker on the EC2 instances. Migrate the containers to the EC2 instances. Create an Amazon Elastic File System (Amazon EFS) file system. Add a mount point to the EC2 instances for the EFS file system.

Answer:

B

Explanation:

Migrating the containers that run the application to AWS Fargate for Amazon Elastic Container Service (Amazon ECS) will meet the requirement of not administering the Docker hosting environment.AWS Fargate is a serverless compute engine that runs containerswithout requiring any infrastructure management3. Creating an Amazon Elastic File System (Amazon EFS) file system and adding a volume to the Fargate task definition to point to the EFS file system will meet the requirement of low-latency storage that will automatically scale throughput to meet increased demand.Amazon EFS is a fully managed file system service that provides shared access to data from multiple containers, supports NFSv4 protocol, and offers consistent performance and high availability4.Amazon EFS also supports automatic scaling of throughput based on the amount of data stored in the file system5.

Question 11

Question Type: MultipleChoice

A medical company is running a REST API on a set of Amazon EC2 instances The EC2 instances run in an Auto Scaling group behind an Application Load Balancer (ALB) The ALB runs in three public subnets, and the EC2 instances run in three private subnets The company has deployed an Amazon CloudFront distribution that has the ALB as the only origin

Which solution should a solutions architect recommend to enhance the origin security?

Options:

A- Store a random string in AWS Secrets Manager Create an AWS Lambda function for automatic secret rotation Configure CloudFront to inject the random string as a custom HTTP header for the origin request Create an AWS WAF web ACL rule with a string match rule for the custom header Associate the web ACL with the ALB

B- Create an AWS WAF web ACL rule with an IP match condition of the CloudFront service IP address ranges Associate the web ACL with the ALB Move the ALB into the three private subnets

C- Store a random string in AWS Systems Manager Parameter Store Configure Parameter Store automatic rotation for the string Configure CloudFront to inject the random string as a custom HTTP header for the origin request Inspect the value of the custom HTTP header, and block access in the ALB

D- Configure AWS Shield Advanced. Create a security group policy to allow connections from CloudFront service IP address ranges. Add the policy to AWS Shield Advanced, and attach the policy to the ALB

Answer:

A

Explanation:

Store Secret in AWS Secrets Manager:

Create a random string in AWS Secrets Manager to be used as a custom HTTP header value.

Set Up Automatic Rotation:

Implement a Lambda function to handle automatic rotation of the secret in AWS Secrets Manager, ensuring the header value remains secure.

Configure CloudFront Custom Header:

In the CloudFront distribution settings, configure an origin custom header with the name and value from AWS Secrets Manager. This header will be included in requests forwarded to the ALB.

Create AWS WAF Web ACL:

Create a Web ACL in AWS WAF with a string match rule to allow requests that include the custom header with the correct value.

Associate the Web ACL with the ALB to filter incoming traffic based on the custom header.

By using this method, you can ensure that only requests coming through CloudFront (which injects the custom header) can reach the ALB, enhancing the origin security