Amazon

DVA-C02 Exam

Amazon Web Services

Questions & Answers
(DemoVersion - LimitedContent)

Version:11.2

Question: 1

Acompany is implementing an application on Amazon EC2 instances. The application needs to process incoming transactions. When the application detects a transaction that is not valid, the application must send a chat message to the company's support team. To send the message, the application needs to retrieve the access token to authenticate by using the chat API.

A developer needs to implement a solution to store the access token. The access token must be encrypted at rest and in transit. The access token must also be accessible from other AWS accounts. Which solution will meet these requirements with the LEAST management overhead?

A. Use an AWS Systems Manager Parameter Store SecureString parameter that uses an AWS Key Management Service (AWS KMS) AWS managed key to store the access token. Add a resource-based policy to the parameter to allow access from other accounts. Update the IAM role of the EC2 instances with permissions to access Parameter Store. Retrieve the token from Parameter Store with the decrypt flag enabled. Use the decrypted access token to send the message to the chat.

B. Encrypt the access token by using an AWS Key Management Service (AWS KMS) customer managed key. Store the access token in an Amazon DynamoDB table. Update the IAM role of the EC2 instances with permissions to access DynamoDB and AWS KMS. Retrieve the token from DynamoDB. Decrypt the token by using AWS KMS on the EC2 instances. Use the decrypted access token to send the message to the chat.

C. Use AWS Secrets Manager with an AWS Key Management Service (AWS KMS) customer managed key to store the access token. Add a resource-based policy to the secret to allow access from other accounts. Update the IAM role of the EC2 instances with permissions to access Secrets Manager. Retrieve the token from Secrets Manager. Use the decrypted access token to send the message to the chat.

D. Encrypt the access token by using an AWS Key Management Service (AWS KMS) AWS managed key. Store the access token in an Amazon S3 bucket. Add a bucket policy to the S3 bucket to allow access from other accounts. Update the IAM role of the EC2 instances with permissions to access Amazon S3 and AWS KMS. Retrieve the token from the S3 bucket. Decrypt the token by using AWS KMS on the EC2 instances. Use the decrypted access token to send the massage to the chat.

Answer: C

Explanation:

https://aws.amazon.com/premiumsupport/knowledge-center/secrets-manager-share-between-accounts/ https://docs.aws.amazon.com/secretsmanager/latest/userguide/auth-and-access_examples_cross.html

Question: 2

Acompany is running Amazon EC2 instances in multiple AWS accounts. A developer needs to implement an application that collects all the lifecycle events of the EC2 instances. The application needs to store the lifecycle events in a single Amazon Simple Queue Service (Amazon SQS) queue in the company's main AWS account for further processing.

Which solution will meet these requirements?

A. Configure Amazon EC2 to deliver the EC2 instance lifecycle events from all accounts to the Amazon EventBridge event bus of the main account. Add an EventBridge rule to the event bus of the main account that matches all EC2 instance lifecycle events. Add the SQS queue as a target of the rule.

B. Use the resource policies of the SQS queue in the main account to give each account permissions to write to that SQS queue. Add to the Amazon EventBridge event bus of each account an EventBridge rule that matches all EC2 instance lifecycle events. Add the SQS queue in the main account as a target of the rule.

C. Write an AWS Lambda function that scans through all EC2 instances in the company accounts to detect EC2 instance lifecycle changes. Configure the Lambda function to write a notification message to the SQS queue in the main account if the function detects an EC2 instance lifecycle change. Add an Amazon EventBridge scheduled rule that invokes the Lambda function every minute.

D. Configure the permissions on the main account event bus to receive events from all accounts. Create an Amazon EventBridge rule in each account to send all the EC2 instance lifecycle events to the main account event bus. Add an EventBridge rule to the main account event bus that matches all EC2 instance lifecycle events. Set the SQS queue as a target for the rule.

Answer: D

Explanation:

Amazon EC2 instances can send the state-change notification events to Amazon EventBridge. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/monitoring-instance-state-changes.html Amazon EventBridge can send and receive events between event buses in AWS accounts. https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-cross-account.html

Question: 3

An application is using Amazon Cognito user pools and identity pools for secure access. A developer wants to integrate the user-specific file upload and download features in the application with Amazon S3. The developer must ensure that the files are saved and retrieved in a secure manner and that users can access only their own files. The file sizes range from 3 KB to 300 MB.

Which option will meet these requirements with the HIGHEST level of security?

A. Use S3 Event Notifications to validate the file upload and download requests and update the user interface (UI).

B. Save the details of the uploaded files in a separate Amazon DynamoDB table. Filter the list of files in the user interface (UI) by comparing the current user ID with the user ID associated with the file in the table.

C. Use Amazon API Gateway and an AWS Lambda function to upload and download files. Validate each request in the Lambda function before performing the requested operation.

D. Use an IAM policy within the Amazon Cognito identity prefix to restrict users to use their own folders in Amazon S3.

Answer: D

Explanation:

https://docs.aws.amazon.com/cognito/latest/developerguide/amazon-cognito-integrating-user-pools-with-identity-pools.html

Question: 4

Acompany is building a scalable data management solution by using AWS services to improve the speed and agility of development. The solution will ingest large volumes of data from various sources and will process this data through multiple business rules and transformations.

The solution requires business rules to run in sequence and to handle reprocessing of data if errors occur when the business rules run. The company needs the solution to be scalable and to require the least possible maintenance.

Which AWS service should the company use to manage and automate the orchestration of the data flows to meet these requirements?

A. AWS Batch

B. AWS Step Functions

C. AWS Glue

D. AWS Lambda

Answer: B

Explanation:

https://docs.aws.amazon.com/step-functions/latest/dg/welcome.html

Question: 5

Adeveloper has created an AWS Lambda function that is written in Python. The Lambda function reads data from objects in Amazon S3 and writes data to an Amazon DynamoDB table. The function is successfully invoked from an S3 event notification when an object is created. However, the function fails when it attempts to write to the DynamoDB table.

What is the MOST likely cause of this issue?

A. The Lambda function's concurrency limit has been exceeded.

B. DynamoDB table requires a global secondary index (GSI) to support writes.

C. The Lambda function does not have IAM permissions to write to DynamoDB.

D. The DynamoDB table is not running in the same Availability Zone as the Lambda function.

Answer: C

Explanation:

https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_examples_lambda-access-dynamodb.html

Question: 6

Adeveloper is creating an AWS CloudFormation template to deploy Amazon EC2 instances across multiple AWS accounts. The developer must choose the EC2 instances from a list of approved instance types.

How can the developer incorporate the list of approved instance types in the CloudFormation template?

A. Create a separate CloudFormation template for each EC2 instance type in the list.

B. In the Resources section of the CloudFormation template, create resources for each EC2 instance type in the list.

C. In the CloudFormation template, create a separate parameter for each EC2 instance type in the list.

D. In the CloudFormation template, create a parameter with the list of EC2 instance types as AllowedValues.

Answer: D

Explanation:

In the CloudFormation template, the developer should create a parameter with the list of approved EC2 instance types as AllowedValues. This way, users can select the instance type they want to use when launching the CloudFormation stack, but only from the approved list.

Question: 7

Adeveloper has an application that makes batch requests directly to Amazon DynamoDB by using the BatchGetItem low-level API operation. The responses frequently return values in the UnprocessedKeys element.

Which actions should the developer take to increase the resiliency of the application when the batch response includes values in UnprocessedKeys? (Choose two.)

A. Retry the batch operation immediately.

B. Retry the batch operation with exponential backoff and randomized delay.

C. Update the application to use an AWS software development kit (AWS SDK) to make the requests.

D. Increase the provisioned read capacity of the DynamoDB tables that the operation accesses.

E. Increase the provisioned write capacity of the DynamoDB tables that the operation accesses.

Answer: B, C

Explanation:

The UnprocessedKeys element indicates that the BatchGetItem operation did not process all of the requested items in the current response. This can happen if the response size limit is exceeded or if the table's provisioned throughput is exceeded. To handle this situation, the developer should retry the batch operation with exponential backoff and randomized delay to avoid throttling errors and reduce the load on the table. The developer should also use an AWS SDK to make the requests, as the SDKs automatically retry requests that return UnprocessedKeys.

Reference:

[BatchGetItem - Amazon DynamoDB]

[Working with Queries and Scans - Amazon DynamoDB]

[Best Practices for Handling DynamoDB Throttling Errors]

Question: 8

Acompany is running a custom application on a set of on-premises Linux servers that are accessed using Amazon API Gateway. AWS X-Ray tracing has been enabled on the API test stage.

How can a developer enable X-Ray tracing on the on-premises servers with the LEAST amount of configuration?

A. Install and run the X-Ray SDK on the on-premises servers to capture and relay the data to the X-Ray service.
B. Install and run the X-Ray daemon on the on-premises servers to capture and relay the data to the X-Ray service.
C. Capture incoming requests on-premises and configure an AWS Lambda function to pull, process, and relay relevant data to X-Ray using the PutTraceSegments API call.
D. Capture incoming requests on-premises and configure an AWS Lambda function to pull, process, and relay relevant data to X-Ray using the PutTelemetryRecords API call.

Answer: B

Explanation:

The X-Ray daemon is a software that collects trace data from the X-Ray SDK and relays it to the X-Ray service. The X-Ray daemon can run on any platform that supports Go, including Linux, Windows, and macOS. The developer can install and run the X-Ray daemon on the on-premises servers to capture and relay the data to the X-Ray service with minimal configuration. The X-Ray SDK is used to instrument the application code, not to capture and relay data. The Lambda function solutions are more complex and require additional configuration.
Reference:

[AWS X-Ray concepts - AWS X-Ray]

[Setting up AWS X-Ray - AWS X-Ray]

Question: 9

Acompany wants to share information with a third party. The third party has an HTTP API endpoint that the company can use to share the information. The company has the required API key to access the HTTP API.

The company needs a way to manage the API key by using code. The integration of the API key with

the application code cannot affect application performance.

Which solution will meet these requirements MOST securely?

A. Store the API credentials in AWS Secrets Manager. Retrieve the API credentials at

runtime by using
the AWS SDK. Use the credentials to make the API call.
B. Store the API credentials in a local code variable. Push the code to a secure Git

repository. Use the
local code variable at runtime to make the API call.
C. Store the API credentials as an object in a private Amazon S3 bucket. Restrict access to

the S3
object by using IAM policies. Retrieve the API credentials at runtime by using the AWS SDK.
Use the
credentials to make the API call. D. Store the API credentials in an Amazon DynamoDB table.

Restrict access to the table by using

Answer: A

resource-based policies. Retrieve the API credentials at runtime by using the AWS SDK.
Use the
credentials to make the API call.

Explanation:

AWS Secrets Manager is a service that helps securely store, rotate, and manage secrets such as API
keys, passwords, and tokens. The developer can store the API credentials in AWS Secrets Manager
and retrieve them at runtime by using the AWS SDK. This solution will meet the requirements of
security, code management, and performance. Storing the API credentials in a local code variable or
an S3 object is not secure, as it exposes the credentials to unauthorized access or leakage. Storing the
API credentials in a DynamoDB table is also not secure, as it requires additional encryption and
access control measures. Moreover, retrieving the credentials from S3 or DynamoDB may affect
application performance due to network latency.
Reference:

[What Is AWS Secrets Manager? - AWS Secrets Manager]

[Retrieving a Secret - AWS Secrets Manager]

Question: 10

Adeveloper is deploying a new application to Amazon Elastic Container Service (Amazon
ECS). The developer needs to securely store and retrieve different types of variables. These
variables include authentication information for a remote API, the URL for the API, and
credentials. The authentication information and API URL must be available to all current
and future deployed versions of the application across development, testing, and
production environments.

How should the developer retrieve the variables with the FEWEST application changes?

A. Update the application to retrieve the variables from AWS Systems Manager Parameter

Store. Use

unique paths in Parameter Store for each variable in each environment. Store the
credentials in AWS

Secrets Manager in each environment. B. Update the application to retrieve the variables

from AWS Key Management Service (AWS KMS).
Store the API URL and credentials as unique keys for each environment.

C. Update the application to retrieve the variables from an encrypted file that is stored with

the

application. Store the API URL and credentials in unique files for each environment.

D. Update the application to retrieve the variables from each of the deployed environments.

Define

Answer: A

Explanation:
the authentication information and API URL in the ECS task definition as unique names
during the
AWS Systems Manager Parameter Store is a service that provides secure, hierarchical
deployment process.
storage for

configuration data management and secrets management. The developer can update the
application

to retrieve the variables from Parameter Store by using the AWS SDK or the AWS CLI. The

developer

can use unique paths in Parameter Store for each variable in each environment, such as
/dev/api-url,

/test/api-url, and /prod/api-url. The developer can also store the credentials in AWS

Secrets

Manager, which is integrated with Parameter Store and provides additional features such
as

automatic rotation and encryption. Reference: [What Is AWS Systems Manager? - AWS

Systems Manager] [Parameter Store - AWS Systems Manager] [What Is AWS Secrets

Manager? - AWS Secrets Manager]

Thank You for trying DVA-C02 PDF Demo

Start Your DVA-C02 Preparation

[Limited Time Offer] Use Coupon "Save25" for extra 25% discount the purchase of PDF file. Test your DVA-C02 preparation with actual exam questions