

# Exam Questions DVA-C02

DVA-C02

<https://www.2passeasy.com/dumps/DVA-C02/>



**NEW QUESTION 1**

A data visualization company wants to strengthen the security of its core applications. The applications are deployed on AWS across its development, staging, pre-production, and production environments. The company needs to encrypt all of its stored sensitive credentials. The sensitive credentials need to be automatically rotated. A version of the sensitive credentials need to be stored for each environment. Which solution will meet these requirements in the MOST operationally efficient way?

- A. Configure AWS Secrets Manager versions to store different copies of the same credentials across multiple environments.
- B. Create a new parameter version in AWS Systems Manager Parameter Store for each environment. Store the environment-specific credentials in the parameter version.
- C. Configure the environment variables in the application code. Use different names for each environment type. Store the environment-specific credentials in the secret.
- D. Configure AWS Secrets Manager to create a new secret for each environment type.

**Answer:** D

**Explanation:**

AWS Secrets Manager is the best option for managing sensitive credentials across multiple environments, as it provides automatic secret rotation, auditing, and monitoring features. It also allows storing environment-specific credentials in separate secrets, which can be accessed by the applications using the SDK or CLI. AWS Systems Manager Parameter Store does not have built-in secret rotation capability, and it requires creating individual parameters or storing the entire credential set as a JSON object. Configuring the environment variables in the application code is not a secure or scalable solution, as it exposes the credentials to anyone who can access the code. References

? AWS Secrets Manager vs. Systems Manager Parameter Store

? AWS System Manager Parameter Store vs Secrets Manager vs Environment Variation in Lambda, when to use which

? AWS Secrets Manager vs. Parameter Store: Features, Cost & More

**NEW QUESTION 2**

A developer needs to perform geographic load testing of an API. The developer must deploy resources to multiple AWS Regions to support the load testing of the API.

How can the developer meet these requirements without additional application code?

- A. Create and deploy an AWS Lambda function in each desired Region.
- B. Configure the Lambda function to create a stack from an AWS CloudFormation template in that Region when the function is invoked. Create an AWS CloudFormation template that defines the load test resource.
- C. Use the AWS CLI create-stack-set command to create a stack set in the desired Regions.
- D. Use the AWS CLI create-stack-set command to create a stack set in the desired Regions.
- E. Create an AWS Systems Manager document that defines the resource.
- F. Use the document to create the resources in the desired Regions.
- G. Create an AWS CloudFormation template that defines the load test resource.
- H. Use the AWS CLI deploy command to create a stack from the template in each Region.

**Answer:** B

**Explanation:**

AWS CloudFormation is a service that allows developers to model and provision AWS resources using templates. A CloudFormation template can define the load test resources, such as EC2 instances, load balancers, and Auto Scaling groups. A CloudFormation stack set is a collection of stacks that can be created and managed from a single template in multiple Regions and accounts. The AWS CLI create-stack-set command can be used to create a stack set from a template and specify the Regions where the stacks should be created. Reference: Working with AWS CloudFormation stack sets

**NEW QUESTION 3**

A developer is building an application that uses AWS API Gateway APIs, AWS Lambda functions, and AWS DynamoDB tables. The developer uses the AWS Serverless Application Model (AWS SAM) to build and run serverless applications on AWS. Each time the developer pushes changes for only the Lambda functions, all the artifacts in the application are rebuilt.

The developer wants to implement AWS SAM Accelerate by running a command to only redeploy the Lambda functions that have changed.

Which command will meet these requirements?

- A. `sam deploy -force-upload`
- B. `sam deploy -no-execute-changeset`
- C. `sam package`
- D. `sam sync -watch`

**Answer:** D

**Explanation:**

The command that will meet the requirements is `sam sync -watch`. This command enables AWS SAM Accelerate mode, which allows the developer to only redeploy the Lambda functions that have changed. The `-watch` flag enables file watching, which automatically detects changes in the source code and triggers a redeployment. The other commands either do not enable AWS SAM Accelerate mode, or do not redeploy the Lambda functions automatically.

Reference: AWS SAM Accelerate

**NEW QUESTION 4**

A developer 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.
- B. Use unique paths in Parameter Store for each variable in each environment.
- C. Store the credentials in AWS Secrets Manager in each environment.

- D. 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.
- E. Update the application to retrieve the variables from an encrypted file that is stored with the application.
- F. Store the API URL and credentials in unique files for each environment.
- G. Update the application to retrieve the variables from each of the deployed environment.
- H. Define the authentication information and API URL in the ECS task definition as unique names during the deployment process.

**Answer:** A

**Explanation:**

AWS Systems Manager Parameter Store is a service that provides secure, hierarchical 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.

References:

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

**NEW QUESTION 5**

A developer needs to deploy an application running on AWS Fargate using Amazon ECS. The application has environment variables that must be passed to a container for the application to initialize.

How should the environment variables be passed to the container?

- A. Define an array that includes the environment variables under the environment parameter within the service definition.
- B. Define an array that includes the environment variables under the environment parameter within the task definition.
- C. Define an array that includes the environment variables under the entryPoint parameter within the task definition.
- D. Define an array that includes the environment variables under the entryPoint parameter within the service definition.

**Answer:** B

**Explanation:**

This solution allows the environment variables to be passed to the container when it is launched by AWS Fargate using Amazon ECS. The task definition is a text file that describes one or more containers that form an application. It contains various parameters for configuring the containers, such as CPU and memory requirements, network mode, and environment variables. The environment parameter is an array of key-value pairs that specify environment variables to pass to a container. Defining an array that includes the environment variables under the entryPoint parameter within the task definition

will not pass them to the container, but use them as command-line arguments for overriding the default entry point of a container.

Defining an array that includes the environment variables under the environment or entryPoint parameter within the service definition will not pass them to the container, but cause an error because these parameters are not valid for a service definition.

Reference: [Task Definition Parameters], [Environment Variables]

**NEW QUESTION 6**

A developer maintains a critical business application that uses Amazon DynamoDB as the primary data store. The DynamoDB table contains millions of documents and receives 30-60 requests each minute. The developer needs to perform processing in near-real-time on the documents when they are added or updated in the DynamoDB table.

How can the developer implement this feature with the LEAST amount of change to the existing application code?

- A. Set up a cron job on an Amazon EC2 instance. Run a script every hour to query the table for changes and process the documents.
- B. Enable a DynamoDB stream on the table. Invoke an AWS Lambda function to process the documents.
- C. Update the application to send a PutEvents request to Amazon EventBridge.
- D. Create an EventBridge rule to invoke an AWS Lambda function to process the documents.
- E. Update the application to synchronously process the documents directly after the DynamoDB write.

**Answer:** B

**Explanation:**

<https://aws.amazon.com/blogs/database/dynamodb-streams-use-cases-and-design-patterns/>

**NEW QUESTION 7**

A developer is testing a RESTful application that is deployed by using Amazon API Gateway and AWS Lambda. When the developer tests the user login by using credentials that are not valid, the developer receives an HTTP 405 METHOD\_NOT\_ALLOWED error. The developer has verified that the test is sending the correct request for the resource.

Which HTTP error should the application return in response to the request?

- A. HTTP 401
- B. HTTP 404
- C. HTTP 503
- D. HTTP 505

**Answer:** A

**Explanation:**

The HTTP 401 error indicates that the request has not been applied because it lacks valid authentication credentials for the target resource. This is the appropriate error code to return when the user login fails due to invalid credentials. The HTTP 405 error means that the method specified in the request is not allowed for the resource identified by the request URI, which is not the case here. The other error codes are not relevant to the authentication failure scenario.

References:

- ? HTTP Status Codes
- ? AWS Lambda Function Errors in API Gateway

### NEW QUESTION 8

A mobile app stores blog posts in an Amazon DynamoDB table. Millions of posts are added every day and each post represents a single item in the table. The mobile app requires only recent posts. Any post that is older than 48 hours can be removed. What is the MOST cost-effective way to delete posts that are older than 48 hours?

- A. For each item add a new attribute of type String that has a timestamp that is set to the blog post creation time
- B. Create a script to find old posts with a table scan and remove posts that are older than 48 hours by using the Batch Write Item API operation
- C. Schedule a cron job on an Amazon EC2 instance once an hour to start the script.
- D. For each item add a new attribute of type String that has a timestamp that is set to the blog post creation time
- E. Create a script to find old posts with a table scan and remove posts that are older than 48 hours by using the Batch Write item API operation
- F. Place the script in a container image
- G. Schedule an Amazon Elastic Container Service (Amazon ECS) task on AWS Fargate that invokes the container every 5 minutes.
- H. For each item, add a new attribute of type Date that has a timestamp that is set to 48 hours after the blog post creation time
- I. Create a global secondary index (GSI) that uses the new attribute as a sort key
- J. Create an AWS Lambda function that references the GSI and removes expired items by using the Batch Write item API operation. Schedule the function with an Amazon CloudWatch event every minute.
- K. For each item add a new attribute of type Number that has a timestamp that is set to 48 hours after the blog post creation time
- L. Create a global secondary index (GSI) that uses the new attribute as a sort key
- M. Create an AWS Lambda function that references the GSI and removes expired items by using the Batch Write item API operation. Schedule the function with an Amazon CloudWatch event every minute.
- N. Configure the DynamoDB table with a TTL that references the new attribute.

**Answer: D**

#### Explanation:

This solution will meet the requirements by using the Time to Live (TTL) feature of DynamoDB, which enables automatically deleting items from a table after a certain time period. The developer can add a new attribute of type Number that has a timestamp that is set to 48 hours after the blog post creation time, which represents the expiration time of the item. The developer can configure the DynamoDB table with a TTL that references the new attribute, which instructs DynamoDB to delete the item when the current time is greater than or equal to the expiration time. This solution is also cost-effective as it does not incur any additional charges for deleting expired items. Option A is not optimal because it will create a script to find and remove old posts with a table scan and a Batch Write Item API operation, which may consume more read and write capacity units and incur more costs. Option B is not optimal because it will use Amazon Elastic Container Service (Amazon ECS) and AWS Fargate to run the script, which may introduce additional costs and complexity for managing and scaling containers. Option C is not optimal because it will create a global secondary index (GSI) that uses the expiration time as a sort key, which may consume more storage space and incur more costs.

References: Time To Live, Managing DynamoDB Time To Live (TTL)

### NEW QUESTION 9

A company has an application that runs across multiple AWS Regions. The application is experiencing performance issues at irregular intervals. A developer must use AWS X-Ray to implement distributed tracing for the application to troubleshoot the root cause of the performance issues. What should the developer do to meet this requirement?

- A. Use the X-Ray console to add annotations for AWS services and user-defined services
- B. Use Region annotation that X-Ray adds automatically for AWS services. Add Region annotation for user-defined services
- C. Use the X-Ray daemon to add annotations for AWS services and user-defined services
- D. Use Region annotation that X-Ray adds automatically for user-defined services. Configure X-Ray to add Region annotation for AWS services

**Answer: B**

#### Explanation:

AWS X-Ray automatically adds Region annotation for AWS services that are integrated with X-Ray. This annotation indicates the AWS Region where the service is running. The developer can use this annotation to filter and group traces by Region and identify any performance issues related to cross-Region calls. The developer can also add Region annotation for user-defined services by using the X-Ray SDK. This option enables the developer to implement distributed tracing for the application that runs across multiple AWS Regions. References:

? AWS X-Ray Annotations

? AWS X-Ray Concepts

### NEW QUESTION 10

A developer at a company recently created a serverless application to process and show data from business reports. The application's user interface (UI) allows users to select and start processing the files. The UI displays a message when the result is available to view. The application uses AWS Step Functions with AWS Lambda functions to process the files. The developer used Amazon API Gateway and Lambda functions to create an API to support the UI. The company's UI team reports that the request to process a file is often returning timeout errors because of the size or complexity of the files. The UI team wants the API to provide an immediate response so that the UI can display a message while the files are being processed. The backend process that is invoked by the API needs to send an email message when the report processing is complete. What should the developer do to configure the API to meet these requirements?

- A. Change the API Gateway route to add an X-Amz-Invocation-Type header with a static value of 'Event' in the integration request. Deploy the API Gateway stage to apply the changes.
- B. Change the configuration of the Lambda function that implements the request to process a file
- C. Configure the maximum age of the event so that the Lambda function will run asynchronously.
- D. Change the API Gateway timeout value to match the Lambda function timeout value
- E. Deploy the API Gateway stage to apply the changes.
- F. Change the API Gateway route to add an X-Amz-Target header with a static value of 'A sync' in the integration request. Deploy the API Gateway stage to apply the changes.

**Answer: A**

#### Explanation:

This solution allows the API to invoke the Lambda function asynchronously, which means that the API will return an immediate response without waiting for the function to complete. The X-Amz-Invocation-Type header specifies the invocation type of the Lambda function, and setting it to 'Event' means that the function will be invoked asynchronously. The function can then use Amazon Simple Email Service (SES) to send an email message when the report processing is complete.

Reference: [Asynchronous invocation], [Set up Lambda proxy integrations in API Gateway]

#### NEW QUESTION 10

A developer needs to build an AWS CloudFormation template that self-populates the AWS Region variable that deploys the CloudFormation template. What is the MOST operationally efficient way to determine the Region in which the template is being deployed?

- A. Use the AWS::Region pseudo parameter
- B. Require the Region as a CloudFormation parameter
- C. Find the Region from the AWS::StackId pseudo parameter by using the Fn::Split intrinsic function
- D. Dynamically import the Region by referencing the relevant parameter in AWS Systems Manager Parameter Store

**Answer:** A

#### Explanation:

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/mappings-section-structure.html>  
<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/pseudo-parameter-reference.html>  
<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/pseudo-parameter-reference.html>

#### NEW QUESTION 11

A company is building an application for stock trading. The application needs sub-millisecond latency for processing trade requests. The company uses Amazon DynamoDB to store all the trading data that is used to process each trading request. A development team performs load testing on the application and finds that the data retrieval time is higher

than expected. The development team needs a solution that reduces the data retrieval time with the least possible effort.

Which solution meets these requirements?

- A. Add local secondary indexes (LSIs) for the trading data.
- B. Store the trading data in Amazon S3 and use S3 Transfer Acceleration.
- C. Add retries with exponential back off for DynamoDB queries.
- D. Use DynamoDB Accelerator (DAX) to cache the trading data.

**Answer:** D

#### Explanation:

This solution will meet the requirements by using DynamoDB Accelerator (DAX), which is a fully managed, highly available, in-memory cache for DynamoDB that delivers up to a 10 times performance improvement - from milliseconds to microseconds - even at millions of requests per second. The developer can use DAX to cache the trading data that is used to process each trading request, which will reduce the data retrieval time with the least possible effort. Option A is not optimal because it will add local secondary indexes (LSIs) for the trading data, which may not improve the performance or reduce the latency of data retrieval, as LSIs are stored on the same partition as the base table and share the same provisioned throughput. Option B is not optimal because it will store the trading data in Amazon S3 and use S3 Transfer Acceleration, which is a feature that enables fast, easy, and secure transfers of files over long distances between S3 buckets and clients, not between DynamoDB and clients. Option C is not optimal because it will add retries with exponential backoff for DynamoDB queries, which is a strategy to handle transient errors by retrying failed requests with increasing delays, not by reducing data retrieval time.

References: [DynamoDB Accelerator (DAX)], [Local Secondary Indexes]

#### NEW QUESTION 12

A developer needs to migrate an online retail application to AWS to handle an anticipated increase in traffic. The application currently runs on two servers: one server for the web application and another server for the database. The web server renders webpages and manages session state in memory. The database server hosts a MySQL database that contains order details. When traffic to the application is heavy, the memory usage for the web server approaches 100% and the application slows down considerably.

The developer has found that most of the memory increase and performance decrease is related to the load of managing additional user sessions. For the web server migration, the developer will use Amazon EC2 instances with an Auto Scaling group behind an Application Load Balancer.

Which additional set of changes should the developer make to the application to improve the application's performance?

- A. Use an EC2 instance to host the MySQL database
- B. Store the session data and the application data in the MySQL database.
- C. Use Amazon ElastiCache for Memcached to store and manage the session data
- D. Use an Amazon RDS for MySQL DB instance to store the application data.
- E. Use Amazon ElastiCache for Memcached to store and manage the session data and the application data.
- F. Use the EC2 instance store to manage the session data
- G. Use an Amazon RDS for MySQL DB instance to store the application data.

**Answer:** B

#### Explanation:

Using Amazon ElastiCache for Memcached to store and manage the session data will reduce the memory load and improve the performance of the web server. Using Amazon RDS for MySQL DB instance to store the application data will provide a scalable, reliable, and managed database service. Option A is not optimal because it does not address the memory issue of the web server. Option C is not optimal because it does not provide a persistent storage for the application data. Option D is not optimal because it does not provide a high availability and durability for the session data.

References: Amazon ElastiCache, Amazon RDS

#### NEW QUESTION 17

An Amazon Kinesis Data Firehose delivery stream is receiving customer data that contains personally identifiable information. A developer needs to remove pattern-based customer identifiers from the data and store the modified data in an Amazon S3 bucket.

What should the developer do to meet these requirements?

- A. Implement Kinesis Data Firehose data transformation as an AWS Lambda function
- B. Configure the function to remove the customer identifier
- C. Set an Amazon S3 bucket as the destination of the delivery stream.
- D. Launch an Amazon EC2 instance
- E. Set the EC2 instance as the destination of the delivery stream
- F. Run an application on the EC2 instance to remove the customer identifier
- G. Store the transformed data in an Amazon S3 bucket.

- H. Create an Amazon OpenSearch Service instance
- I. Set the OpenSearch Service instance as the destination of the delivery stream
- J. Use search and replace to remove the customer identifier
- K. Export the data to an Amazon S3 bucket.
- L. Create an AWS Step Functions workflow to remove the customer identifier
- M. As the last step in the workflow, store the transformed data in an Amazon S3 bucket
- N. Set the workflow as the destination of the delivery stream.

**Answer:** A

**Explanation:**

Amazon Kinesis Data Firehose is a service that delivers real-time streaming data to destinations such as Amazon S3, Amazon Redshift, Amazon OpenSearch Service, and Amazon Kinesis Data Analytics. The developer can implement Kinesis Data Firehose data transformation as an AWS Lambda function. The function can remove pattern-based customer identifiers from the data and return the modified data to Kinesis Data Firehose. The developer can set an Amazon S3 bucket as the destination of the delivery stream. References:

- ? [What Is Amazon Kinesis Data Firehose? - Amazon Kinesis Data Firehose]
- ? [Data Transformation - Amazon Kinesis Data Firehose]

**NEW QUESTION 19**

A company notices that credentials that the company uses to connect to an external software as a service (SaaS) vendor are stored in a configuration file as plaintext.

The developer needs to secure the API credentials and enforce automatic credentials rotation on a quarterly basis. Which solution will meet these requirements MOST securely?

- A. Use AWS Key Management Service (AWS KMS) to encrypt the configuration file
- B. Decrypt the configuration file when users make API calls to the SaaS vendor
- C. Enable rotation.
- D. Retrieve temporary credentials from AWS Security Token Service (AWS STS) every 15 minutes
- E. Use the temporary credentials when users make API calls to the SaaS vendor.
- F. Store the credentials in AWS Secrets Manager and enable rotation
- G. Configure the API to have Secrets Manager access.  
 Store the credentials in AWS Systems Manager Parameter Store and enable rotation
- H. Retrieve the credentials when users make API calls to the SaaS vendor.

**Answer:** C

**Explanation:**

Store the credentials in AWS Secrets Manager and enable rotation. Configure the API to have Secrets Manager access. This is correct. This solution will meet the requirements most securely, because it uses a service that is designed to store and manage secrets such as API credentials. AWS Secrets Manager helps you protect access to your applications, services, and IT resources by enabling you to rotate, manage, and retrieve secrets throughout their lifecycle<sup>1</sup>. You can store secrets such as passwords, database strings, API keys, and license codes as encrypted values<sup>2</sup>. You can also configure automatic rotation of your secrets on a schedule that you specify<sup>3</sup>. You can use the AWS SDK or CLI to retrieve secrets from Secrets Manager when you need them<sup>4</sup>. This way, you can avoid storing credentials in plaintext files or hardcoding them in your code.

**NEW QUESTION 22**

A company has a multi-node Windows legacy application that runs on premises. The application uses a network shared folder as a centralized configuration repository to store configuration files in .xml format. The company is migrating the application to Amazon EC2 instances. As part of the migration to AWS, a developer must identify a solution that provides high availability for the repository.

Which solution will meet this requirement MOST cost-effectively?

- A. Mount an Amazon Elastic Block Store (Amazon EBS) volume onto one of the EC2 instances
- B. Deploy a file system on the EBS volume
- C. Use the host operating system to share a folder on the EC2 instances
- D. Update the application code to read and write configuration files from the shared folder.
- E. Deploy a micro EC2 instance with an instance store volume
- F. Use the host operating system to share a folder on the EC2 instances
- G. Update the application code to read and write configuration files from the shared folder.
- H. Create an Amazon S3 bucket to host the repository
- I. Migrate the existing .xml files to the S3 bucket
- J. Update the application code to use the AWS SDK to read and write configuration files from Amazon S3.
- K. Create an Amazon S3 bucket to host the repository
- L. Migrate the existing .xml files to the S3 bucket
- M. Mount the S3 bucket to the EC2 instances as a local volume
- N. Update the application code to read and write configuration files from the disk.

**Answer:** C

**Explanation:**

Amazon S3 is a service that provides highly scalable, durable, and secure object storage. The developer can create an S3 bucket to host the repository and migrate the existing .xml files to the S3 bucket. The developer can update the application code to use the AWS SDK to read and write configuration files from S3. This solution will meet the requirement of high availability for the repository in a cost-effective way.

References:

- ? [Amazon Simple Storage Service (S3)]
- ? [Using AWS SDKs with Amazon S3]

**NEW QUESTION 27**

A developer is troubleshooting an Amazon API Gateway API. Clients are receiving HTTP 400 response errors when the clients try to access an endpoint of the API. How can the developer determine the cause of these errors?

- A. Create an Amazon Kinesis Data Firehose delivery stream to receive API call logs from API Gateway

- B. Configure Amazon CloudWatch Logs as the delivery stream's destination.
- C. Turn on AWS CloudTrail Insights and create a trail Specify the Amazon Resource Name (ARN) of the trail for the stage of the API.  
 Turn on AWS X-Ray for the API stage Create an Amazon CloudWatch Logs log group Specify the Amazon Resource Name (ARN) of the log group for the API stage.
- D. Turn on execution logging and access logging in Amazon CloudWatch Logs for the API stage.
- E. Turn on execution logging and access logging in Amazon CloudWatch Logs for the API stage.
- F. Create a CloudWatch Logs log group
- G. Specify the Amazon Resource Name (ARN) of the log group for the API stage.

**Answer: D**

**Explanation:**

This solution will meet the requirements by using Amazon CloudWatch Logs to capture and analyze the logs from API Gateway. Amazon CloudWatch Logs is a service that monitors, stores, and accesses log files from AWS resources. The developer can turn on execution logging and access logging in Amazon CloudWatch Logs for the API stage, which enables logging information about API execution and client access to the API. The developer can create a CloudWatch Logs log group, which is a collection of log streams that share the same retention, monitoring, and access control settings. The developer can specify the Amazon Resource Name (ARN) of the log group for the API stage, which instructs API Gateway to send the logs to the specified log group. The developer can then examine the logs to determine the cause of the HTTP 400 response errors. Option A is not optimal because it will create an Amazon Kinesis Data Firehose delivery stream to receive API call logs from API Gateway, which may introduce additional costs and complexity for delivering and processing streaming data. Option B is not optimal because it will turn on AWS CloudTrail Insights and create a trail, which is a feature that helps identify and troubleshoot unusual API activity or operational issues, not HTTP response errors. Option C is not optimal because it will turn on AWS X-Ray for the API stage, which is a service that helps analyze and debug distributed applications, not HTTP response errors. References: [Setting Up CloudWatch Logging for a REST API], [CloudWatch Logs Concepts]

**NEW QUESTION 31**

A company needs to deploy all its cloud resources by using AWS CloudFormation templates A developer must create an Amazon Simple Notification Service (Amazon SNS) automatic notification to help enforce this rule. The developer creates an SNS topic and subscribes the email address of the company's security team to the SNS topic.

The security team must receive a notification immediately if an IAM role is created without the use of CloudFormation.

Which solution will meet this requirement?

- A. Create an AWS Lambda function to filter events from CloudTrail if a role was created without CloudFormation Configure the Lambda function to publish to the SNS topic
- B. Create an Amazon EventBridge schedule to invoke the Lambda function every 15 minutes
- C. Create an AWS Fargate task in Amazon Elastic Container Service (Amazon ECS) to filter events from CloudTrail if a role was created without CloudFormation Configure the Fargate task to publish to the SNS topic Create an Amazon EventBridge schedule to run the Fargate task every 15 minutes
- D. Launch an Amazon EC2 instance that includes a script to filter events from CloudTrail if a role was created without CloudFormation Configure the script to publish to the SNS topic
- E. Configure the script to publish to the SNS topic
- F. Create a cron job to run the script on the EC2 instance every 15 minutes.
- G. Create an Amazon EventBridge rule to filter events from CloudTrail if a role was created without CloudFormation Specify the SNS topic as the target of the EventBridge rule.

**Answer: D**

**Explanation:**

Creating an Amazon EventBridge rule is the most efficient and scalable way to monitor and react to events from CloudTrail, such as the creation of an IAM role without CloudFormation. EventBridge allows you to specify a filter pattern to match the events you are interested in, and then specify an SNS topic as the target to send notifications. This solution does not require any additional resources or code, and it can trigger notifications in near real-time. The other solutions involve creating and managing additional resources, such as Lambda functions, Fargate tasks, or EC2 instances, and they rely on polling CloudTrail events every 15 minutes, which can introduce delays and increase costs. References

- ? Using Amazon EventBridge rules to process AWS CloudTrail events
- ? Using AWS CloudFormation to create and manage AWS Batch resources
- ? How to use AWS CloudFormation to configure auto scaling for Amazon Cognito and AWS AppSync
- ? Using AWS CloudFormation to automate the creation of AWS WAF web ACLs, rules, and conditions

**NEW QUESTION 32**

A developer has been asked to create an AWS Lambda function that is invoked any time updates are made to items in an Amazon DynamoDB table. The function has been created and appropriate permissions have been added to the Lambda execution role Amazon DynamoDB streams have been enabled for the table, but the function is still not being invoked.

Which option would enable the DynamoDB table updates to invoke the Lambda function?

- A. Change the StreamViewType parameter value to NEW\_AND\_OLD\_IMAGES for the DynamoDB table.
- B. Configure event source mapping for the Lambda function.
- C. Map an Amazon Simple Notification Service (Amazon SNS) topic to the DynamoDB streams.
- D. Increase the maximum runtime (timeout) setting of the Lambda function.

**Answer: B**

**Explanation:**

This solution allows the Lambda function to be invoked by the DynamoDB stream whenever updates are made to items in the DynamoDB table. Event source mapping is a feature of Lambda that enables a function to be triggered by an event source, such as a DynamoDB stream, an Amazon Kinesis stream, or an Amazon Simple Queue Service (SQS) queue. The developer can configure event source mapping for the Lambda function using the AWS Management Console, the AWS CLI, or the AWS SDKs. Changing the StreamViewType parameter value to NEW\_AND\_OLD\_IMAGES for the DynamoDB table will not affect the invocation of the Lambda function, but only change the information that is written to the stream record. Mapping an Amazon Simple Notification Service (Amazon SNS) topic to the DynamoDB stream will not invoke the Lambda function directly, but require an additional subscription from the Lambda function to the SNS topic. Increasing the maximum runtime (timeout) setting of the Lambda function will not affect the invocation of the Lambda function, but only change how long the function can run before it is terminated.

Reference: [Using AWS Lambda with Amazon DynamoDB], [Using AWS Lambda with Amazon SNS]

**NEW QUESTION 36**

An application that runs on AWS receives messages from an Amazon Simple Queue Service (Amazon SQS) queue and processes the messages in batches. The

application sends the data to another SQS queue to be consumed by another legacy application. The legacy system can take up to 5 minutes to process some transaction data.

A developer wants to ensure that there are no out-of-order updates in the legacy system. The developer cannot alter the behavior of the legacy system. Which solution will meet these requirements?

- A. Use an SQS FIFO queue
- B. Configure the visibility timeout value.
- C. Use an SQS standard queue with a SendMessageBatchRequestEntry data type
- D. Configure the DelaySeconds values.
- E. Use an SQS standard queue with a SendMessageBatchRequestEntry data type
- F. Configure the visibility timeout value.
- G. Use an SQS FIFO queue
- H. Configure the DelaySeconds value.

**Answer:** A

**Explanation:**

? An SQS FIFO queue is a type of queue that preserves the order of messages and ensures that each message is delivered and processed only once. This is suitable for the scenario where the developer wants to ensure that there are no out-of-order updates in the legacy system.

? The visibility timeout value is the amount of time that a message is invisible in the queue after a consumer receives it. This prevents other consumers from processing the same message simultaneously. If the consumer does not delete the message before the visibility timeout expires, the message becomes visible again and another consumer can receive it.

? In this scenario, the developer needs to configure the visibility timeout value to be longer than the maximum processing time of the legacy system, which is 5 minutes. This will ensure that the message remains invisible in the queue until the legacy system finishes processing it and deletes it. This will prevent duplicate or out-of-order processing of messages by the legacy system.

**NEW QUESTION 41**

A developer is creating an AWS Lambda function that needs credentials to connect to an Amazon RDS for MySQL database. An Amazon S3 bucket currently stores the credentials. The developer needs to improve the existing solution by implementing credential rotation and secure storage. The developer also needs to provide integration with the Lambda function.

Which solution should the developer use to store and retrieve the credentials with the LEAST management overhead?

- A. Store the credentials in AWS Systems Manager Parameter Store
- B. Select the database that the parameter will access
- C. Use the default AWS Key Management Service (AWS KMS) key to encrypt the parameter
- D. Enable automatic rotation for the parameter
- E. Use the parameter from Parameter Store on the Lambda function to connect to the database.
- F. Encrypt the credentials with the default AWS Key Management Service (AWS KMS) key
- G. Store the credentials as environment variables for the Lambda function
- H. Create a second Lambda function to generate new credentials and to rotate the credentials by updating the environment variables of the first Lambda function
- I. Invoke the second Lambda function by using an Amazon EventBridge rule that runs on a schedule
- J. Update the database to use the new credential
- K. On the first Lambda function, retrieve the credentials from the environment variable
- L. Decrypt the credentials by using AWS KMS, connect to the database.
- M. Store the credentials in AWS Secrets Manager
- N. Set the secret type to Credentials for Amazon RDS databases
- O. Select the database that the secret will access
- P. Use the default AWS Key Management Service (AWS KMS) key to encrypt the secret
- Q. Enable automatic rotation for the secret
- R. Use the secret from Secrets Manager on the Lambda function to connect to the database.
- S. Encrypt the credentials by using AWS Key Management Service (AWS KMS). Store the credentials in an Amazon DynamoDB table
- T. Create a second Lambda function to rotate the credential
- . Invoke the second Lambda function by using an Amazon EventBridge rule that runs on a schedule
- . Update the DynamoDB table
- . Update the database to use the generated credential
- . Retrieve the credentials from DynamoDB with the first Lambda function
- . Connect to the database.

**Answer:** C

**Explanation:**

AWS Secrets Manager is a service that helps you protect secrets needed to access your applications, services, and IT resources. Secrets Manager enables you to store, retrieve, and rotate secrets such as database credentials, API keys, and passwords. Secrets Manager supports a secret type for RDS databases, which allows you to select an existing RDS database instance and generate credentials for it. Secrets Manager encrypts the secret using AWS Key Management Service (AWS KMS) keys and enables automatic rotation of the secret at a specified interval. A Lambda function can use the AWS SDK or CLI to retrieve the secret from Secrets Manager and use it to connect to the database. Reference: Rotating your AWS Secrets Manager secrets

**NEW QUESTION 45**

A developer has observed an increase in bugs in the AWS Lambda functions that a development team has deployed in its Node.js application. To minimize these bugs, the developer wants to implement automated testing of Lambda functions in an environment that closely simulates the Lambda environment.

The developer needs to give other developers the ability to run the tests locally. The developer also needs to integrate the tests into the team's continuous integration and continuous delivery (CI/CD) pipeline before the AWS Cloud Development Kit (AWS CDK) deployment.

Which solution will meet these requirements?

- A. Create sample events based on the Lambda documentation
- B. Create automated test scripts that use the `cdk local invoke` command to invoke the Lambda function
- C. Check the response Document the test scripts for the other developers on the team Update the CI/CD pipeline to run the test scripts.
- D. Install a unit testing framework that reproduces the Lambda execution environment
- E. Create sample events based on the Lambda Documentation Invoke the handler function by using a unit testing framework framework for the other developers on the team
- F. Check the response Document how to run the unit testing

- G. Update the OCD pipeline to run the unit testing framework.
- H. Install the AWS Serverless Application Model (AWS SAM) CLI tool Use the Sam local generate-event command to generate sample events for me automated test
- I. Create automated test scripts that use the Sam local invoke command to invoke the Lambda function
- J. Check the response Document the test scripts for the other developers on the team Update the CI/CD pipeline to run the test scripts.
- K. Create sample events based on the Lambda documentatio
- L. Create a Docker container from the Node is base image to invoke the Lambda function
- M. Check the response Document how to run the Docker container for the more developers on the team update the CI/CD pipeline to run the Docker container.

**Answer: C**

**Explanation:**

This solution will meet the requirements by using AWS SAM CLI tool, which is a command line tool that lets developers locally build, test, debug, and deploy serverless applications defined by AWS SAM templates. The developer can use sam local generate- event command to generate sample events for different event sources such as API Gateway or S3. The developer can create automated test scripts that use sam local invoke command to invoke Lambda functions locally in an environment that closely simulates Lambda environment. The developer can check the response from Lambda functions and document how to run the test scripts for other developers on the team. The developer can also update CI/CD pipeline to run these test scripts before deploying with AWS CDK. Option A is not optimal because it will use cdk local invoke command, which does not exist in AWS CDK CLI tool. Option B is not optimal because it will use a unit testing framework that reproduces Lambda execution environment, which may not be accurate or consistent with Lambda environment. Option D is not optimal because it will create a Docker container from Node.js base image to invoke Lambda functions, which may introduce additional overhead and complexity for creating and running Docker containers.

References: [AWS Serverless Application Model (AWS SAM)], [AWS Cloud Development Kit (AWS CDK)]

**NEW QUESTION 48**

A developer is configuring an applications deployment environment in AWS CodePipeine. The application code is stored in a GitHub repository. The developer wants to ensure that the repository package's unit tests run in the new deployment environment. The deployment has already set the pipeline's source provider to GitHub and has specified the repository and branch to use in the deployment.

When combination of steps should the developer take next to meet these requirements with the least the LEAST overhead' (Select TWO).

- A. Create an AWS CodeCommt projec
- B. Add the repository package's build and test commands to the protects buildspec
- C. Create an AWS CodeBuid projec
- D. Add the repository package's build and test commands to the projects buildspec
- E. Create an AWS CodeDeploy protec
- F. Add the repository package's build and test commands to the project's buildspec
- G. Add an action to the source stag
- H. Specify the newly created project as the action provide
- I. Specify the build attract as the actions input artifact.
- J. Add a new stage to the pipeline alter the source stag
- K. Add an action to the new stag
- L. Speedy the newly created protect as the action provide
- M. Specify the source artifact as the action's input artifact.

**Answer: BE**

**Explanation:**

This solution will ensure that the repository package's unit tests run in the new deployment environment with the least overhead because it uses AWS CodeBuild to build and test the code in a fully managed service, and AWS CodePipeline to orchestrate the deployment stages and actions. Option A is not optimal because it will use AWS CodeCommit instead of AWS CodeBuild, which is a source control service, not a build and test service. Option C is not optimal because it will use AWS CodeDeploy instead of AWS CodeBuild, which is a deployment service, not a build and test service. Option D is not optimal because it will add an action to the source stage instead of creating a new stage, which will not follow the best practice of separating different deployment phases. References: AWS CodeBuild, AWS CodePipeline

**NEW QUESTION 53**

An online sales company is developing a serverless application that runs on AWS. The application uses an AWS Lambda function that calculates order success rates and stores the data in an Amazon DynamoDB table. A developer wants an efficient way to invoke the Lambda function every 15 minutes.

Which solution will meet this requirement with the LEAST development effort?

- A. Create an Amazon EventBridge rule that has a rate expression that will run the rule every 15 minute
- B. Add the Lambda function as the target of the EventBridge rule.
- C. Create an AWS Systems Manager document that has a script that will invoke the Lambda function on Amazon EC2. Use a Systems Manager Run Command task to run the shell script every 15 minutes.
- D. Create an AWS Step Functions state machin
- E. Configure the state machine to invoke the Lambda function execution role at a specified interval by using a Wait stat
- F. Set the interval to 15 minutes.
- G. Provision a small Amazon EC2 instanc
- H. Set up a cron job that invokes the Lambda function every 15 minutes.

**Answer: A**

**Explanation:**

The best solution for this requirement is option A. Creating an Amazon EventBridge rule that has a rate expression that will run the rule every 15 minutes and adding the Lambda function as the target of the EventBridge rule is the most efficient way to invoke the Lambda function periodically. This solution does not require any additional resources or development effort, and it leverages the built-in scheduling capabilities of EventBridge1.

**NEW QUESTION 57**

A developer 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.

**NEW QUESTION 59**

A company uses a custom root certificate authority certificate chain (Root CA Cert) that is 10 KB in size generate SSL certificates for its on-premises HTTPS endpoints. One of the company's cloud based applications has hundreds of AWS Lambda functions that pull data from these endpoints. A developer updated the trust store of the Lambda execution environment to use the Root CA Cert when the Lambda execution environment is initialized. The developer bundled the Root CA Cert as a text file in the Lambdas deployment bundle.

After 3 months of development the root CA Cert is no longer valid and must be updated. The developer needs a more efficient solution to update the Root CA Cert for all deployed Lambda functions. The solution must not include rebuilding or updating all Lambda functions that use the Root CA Cert. The solution must also work for all development, testing and production environment. Each environment is managed in a separate AWS account.

When combination of steps Would the developer take to meet these environments MOST cost-effectively? (Select TWO)

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

This solution will meet the requirements by storing the Root CA Cert as a Secure String parameter in AWS Systems Manager Parameter Store, which is a secure and scalable service for storing and managing configuration data and secrets. The resource-based policy will allow IAM users in different AWS accounts and environments to access the parameter without requiring cross-account roles or permissions. The Lambda code will be refactored to load the Root CA Cert from the parameter store and modify the runtime trust store outside the Lambda function handler, which will improve performance and reduce latency by avoiding repeated calls to Parameter Store and trust store modifications for each invocation of the Lambda function. Option A is not optimal because it will use AWS Secrets Manager instead of AWS Systems Manager Parameter Store, which will incur additional costs and complexity for storing and managing a non-secret configuration data such as Root CA Cert. Option C is not optimal because it will deactivate the application secrets and monitor the application error logs temporarily, which will cause application downtime and potential data loss. Option D is not optimal because it will modify the runtime trust store inside the Lambda function handler, which will degrade performance and increase latency by repeating unnecessary operations for each invocation of the Lambda function.

References: AWS Systems Manager Parameter Store, [Using SSL/TLS to Encrypt a Connection to a DB Instance]

**NEW QUESTION 60**

A developer 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:** BC

**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.

References:

? [BatchGetItem - Amazon DynamoDB]

? [Working with Queries and Scans - Amazon DynamoDB]

? [Best Practices for Handling DynamoDB Throttling Errors]

**NEW QUESTION 64**

A developer has written the following IAM policy to provide access to an Amazon S3 bucket:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/secrets*"
    }
  ]
}
```

Which access does the policy allow regarding the s3:GetObject and s3:PutObject actions?

- A. Access on all buckets except the "DOC-EXAMPLE-BUCKET" bucket
- B. Access on all buckets that start with "DOC-EXAMPLE-BUCKET" except the "DOC-EXAMPLE-BUCKET/secrets" bucket
- C. Access on all objects in the "DOC-EXAMPLE-BUCKET" bucket along with access to all S3 actions for objects in the "DOC-EXAMPLE-BUCKET" bucket that start with "secrets"
- D. Access on all objects in the "DOC-EXAMPLE-BUCKET" bucket except on objects that start with "secrets"

**Answer:** D

**Explanation:**

The IAM policy shown in the image is a resource-based policy that grants or denies access to an S3 bucket based on certain conditions. The first statement allows access to any S3 action on any object in the "DOC-EXAMPLE-BUCKET" bucket when the request is made over HTTPS (the value of aws:SecureTransport is true). The second statement denies access to the s3:GetObject and s3:PutObject actions on any object in the "DOC-EXAMPLE-BUCKET/secrets" prefix when the request is made over HTTP (the value of aws:SecureTransport is false). Therefore, the policy allows access on all objects in the "DOC-EXAMPLE-BUCKET" bucket except on objects that start with "secrets".

Reference: Using IAM policies for Amazon S3

**NEW QUESTION 67**

A developer is creating an application that will be deployed on IoT devices. The application will send data to a RESTful API that is deployed as an AWS Lambda function. The application will assign each API request a unique identifier. The volume of API requests from the application can randomly increase at any given time of day.

During periods of request throttling, the application might need to retry requests. The API must be able to handle duplicate requests without inconsistencies or data loss.

Which solution will meet these requirements?

- A. Create an Amazon RDS for MySQL DB instance
- B. Store the unique identifier for each request in a database table
- C. Modify the Lambda function to check the table for the identifier before processing the request.
- D. Create an Amazon DynamoDB table
- E. Store the unique identifier for each request in the table
- F. Modify the Lambda function to check the table for the identifier before processing the request.
- G. Create an Amazon DynamoDB table
- H. Store the unique identifier for each request in the table
- I. Modify the Lambda function to return a client error response when the function receives a duplicate request.
- J. Create an Amazon ElastiCache for Memcached instance
- K. Store the unique identifier for each request in the cache
- L. Modify the Lambda function to check the cache for the identifier before processing the request.

**Answer:** B

**Explanation:**

Amazon DynamoDB is a fully managed NoSQL database service that can store and retrieve any amount of data with high availability and performance. DynamoDB can handle concurrent requests from multiple IoT devices without throttling or data loss. To prevent duplicate requests from causing inconsistencies or data loss, the Lambda function can use DynamoDB conditional writes to check if the unique identifier for each request already exists in the table before processing the request. If the identifier exists, the function can skip or abort the request; otherwise, it can process the request and store the identifier in the table. Reference: Using conditional writes

**NEW QUESTION 72**

A company has an existing application that has hardcoded database credentials. A developer needs to modify the existing application. The application is deployed in two AWS Regions with an active-passive failover configuration to meet company's disaster recovery strategy.

The developer needs a solution to store the credentials outside the code. The solution must comply with the company's disaster recovery strategy. Which solution will meet these requirements in the MOST secure way?

- A. Store the credentials in AWS Secrets Manager in the primary Region
- B. Enable secret replication to the secondary Region. Update the application to use the Amazon Resource Name (ARN) based on the Region.

- C. Store credentials in AWS Systems Manager Parameter Store in the primary Region
- D. Enable parameter replication to the secondary Region
- E. Update the application to use the Amazon Resource Name (ARN) based on the Region.
- F. Store credentials in a config file
- G. Upload the config file to an S3 bucket in the primary Region
- H. Enable Cross-Region Replication (CRR) to an S3 bucket in the secondary region
- I. Update the application to access the config file from the S3 bucket based on the Region.  
  - Store credentials in a config file
- J. Upload the config file to an Amazon Elastic File System (Amazon EFS) file system
- L. Update the application to use the Amazon EFS file system Regional endpoints to access the config file in the primary and secondary Regions.

**Answer:** A

**Explanation:**

AWS Secrets Manager is a service that allows you to store and manage secrets, such as database credentials, API keys, and passwords, in a secure and centralized way. It also provides features such as automatic secret rotation, auditing, and monitoring<sup>1</sup>. By using AWS Secrets Manager, you can avoid hardcoding credentials in your code, which is a bad security practice and makes it difficult to update them. You can also replicate your secrets to another Region, which is useful for disaster recovery purposes<sup>2</sup>. To access your secrets from your application, you can use the ARN of the secret, which is a unique identifier that includes the Region name. This way, your application can use the appropriate secret based on the Region where it is deployed<sup>3</sup>.

References:

- ? AWS Secrets Manager
- ? Replicating and sharing secrets
- ? Using your own encryption keys

**NEW QUESTION 75**

A developer is working on an ecommerce platform that communicates with several third-party payment processing APIs. The third-party payment services do not provide a test environment.

The developer needs to validate the ecommerce platform's integration with the third-party payment processing APIs. The developer must test the API integration code without invoking the third-party payment processing APIs.

Which solution will meet these requirements?

- A. Set up an Amazon API Gateway REST API with a gateway response configured for status code 200. Add response templates that contain sample responses captured from the real third-party API.
- B. Set up an AWS AppSync GraphQL API with a data source configured for each third-party API. Specify an integration type of Mock. Configure integration responses by using sample responses captured from the real third-party API.
- C. Create an AWS Lambda function for each third-party API.
- D. Embed responses captured from the real third-party API.
- E. Configure Amazon Route 53 Resolver with an inbound endpoint for each Lambda function's Amazon Resource Name (ARN).
- F. Set up an Amazon API Gateway REST API for each third-party API. Specify an integration request type of Mock. Configure integration responses by using sample responses captured from the real third-party API.

**Answer:** D

**Explanation:**

Amazon API Gateway can mock responses for testing purposes without requiring any integration backend. This allows the developer to test the API integration code without invoking the third-party payment processing APIs. The developer can configure integration responses by using sample responses captured from the real third-party API. References:

- ? Mocking Integration Responses in API Gateway
- ? Set up Mock Integrations for an API in API Gateway

**NEW QUESTION 78**

A financial company must store original customer records for 10 years for legal reasons. A complete record contains personally identifiable information (PII). According to local regulations, PII is available to only certain people in the company and must not be shared with third parties. The company needs to make the records available to third-party organizations for statistical analysis without sharing the PII.

A developer wants to store the original immutable record in Amazon S3. Depending on who accesses the S3 document, the document should be returned as is or with all the PII removed. The developer has written an AWS Lambda function to remove the PII from the document. The function is named removePii.

What should the developer do so that the company can meet the PII requirements while maintaining only one copy of the document?

- A. Set up an S3 event notification that invokes the removePii function when an S3 GET request is made
- B. Call Amazon S3 by using a GET request to access the object without PII.
- C. Set up an S3 event notification that invokes the removePii function when an S3 PUT request is made
- D. Call Amazon S3 by using a PUT request to access the object without PII.
- E. Create an S3 Object Lambda access point from the S3 console
- F. Select the removePii function
- G. Use S3 Access Points to access the object without PII.
- H. Create an S3 access point from the S3 console
- I. Use the access point name to call the GetObjectLegalHold S3 API function
- J. Pass in the removePii function name to access the object without PII.

**Answer:** C

**Explanation:**

S3 Object Lambda allows you to add your own code to process data retrieved from S3 before returning it to an application. You can use an AWS Lambda function to modify the data, such as removing PII, redacting confidential information, or resizing images. You can create an S3 Object Lambda access point and associate it with your Lambda function. Then, you can use the access point to request objects from S3 and get the modified data back. This way, you can maintain only one copy of the original

document in S3 and apply different transformations depending on who accesses it. Reference: Using AWS Lambda with Amazon S3

**NEW QUESTION 81**

A company 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. Mastered
- B. Not Mastered

**Answer:** A

**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.

References:

- ? [What Is AWS Secrets Manager? - AWS Secrets Manager]
- ? [Retrieving a Secret - AWS Secrets Manager]

**NEW QUESTION 82**

A developer has observed an increase in bugs in the AWS Lambda functions that a development team has deployed in its Node.js application. To minimize these bugs, the developer wants to implement automated testing of Lambda functions in an environment that closely simulates the Lambda environment.

The developer needs to give other developers the ability to run the tests locally. The developer also needs to integrate the tests into the team's continuous integration and continuous delivery (CI/CD) pipeline before the AWS Cloud Development Kit (AWS CDK) deployment.

Which solution will meet these requirements?

- A. Create sample events based on the Lambda documentatio
- B. Create automated test scripts that use the cdk local invoke command to invoke the Lambda function
- C. Check the respons
- D. Document the test scripts for the other developers on the tea
- E. Update the CI/CD pipeline to run the test scripts.

Create sample events based on the Lambda

- F. Install a unit testing framework that reproduces the Lambda execution environment.
- G. Invoke the handler function by using a unit testing framewor
- H. Check the respons
- I. Document how to run the unit testing framework for the other developers on the tea
- J. Update the CI/CD pipeline to run the unit testing framework.
- K. Install the AWS Serverless Application Model (AWS SAM) CLI too
- L. Use the sam local generate-event command to generate sample events for the automated test
- M. Create automated test scripts that use the sam local invoke command to invoke the Lambda function
- N. Check the respons
- O. Document the test scripts for the other developers on the tea
- P. Update the CI/CD pipeline to run the test scripts.
- Q. Create sample events based on the Lambda documentatio
- R. Create a Docker container from the Node.js base image to invoke the Lambda function
- S. Check the respons
- T. Document how to run the Docker container for the other developers on the tea
- . Update the CI/CD pipeline to run the Docker container.

**Answer:** C

**Explanation:**

The AWS Serverless Application Model Command Line Interface (AWS SAM CLI) is a command-line tool for local development and testing of Serverless applications<sup>3</sup>. The sam local generate-event command of AWS SAM CLI generates sample events for automated tests<sup>3</sup>. The sam local invoke command is used to invoke Lambda functions<sup>3</sup>. Therefore, option C is correct.

**NEW QUESTION 85**

A developer is creating a new REST API by using Amazon API Gateway and AWS Lambda. The development team tests the API and validates responses for the known use cases before deploying the API to the production environment.

The developer wants to make the REST API available for testing by using API Gateway locally.

Which AWS Serverless Application Model Command Line Interface (AWS SAM CLI) subcommand will meet these requirements?

- A. Sam local invoke
- B. Sam local generate-event
- C. Sam local start-lambda
- D. Sam local start-api

**Answer:** D

**Explanation:**

The AWS Serverless Application Model Command Line Interface (AWS SAM CLI) is a command-line tool for local development and testing of Serverless applications<sup>2</sup>. The sam local start-api subcommand of AWS SAM CLI is used to simulate a REST API by starting a new local endpoint<sup>3</sup>. Therefore, option D is correct.

**NEW QUESTION 87**

A developer at a company needs to create a small application that makes the same API call once each day at a designated time. The company does not have

infrastructure in the AWS Cloud yet, but the company wants to implement this functionality on AWS. Which solution meets these requirements in the MOST operationally efficient manner?

Use a Kubernetes cron job that runs on Amazon Elastic Kubernetes Service (Amazon EKS).

- A. Use an Amazon Linux crontab scheduled job that runs on Amazon EC2.
- B. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event.
- C. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event.
- D. Use an AWS Batch job that is submitted to an AWS Batch job queue.

**Answer: C**

**Explanation:**

The correct answer is C. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event.

\* C. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event. This is correct. AWS Lambda is a serverless compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging<sup>1</sup>. Amazon EventBridge is a serverless event bus service that enables you to connect your applications with data from a variety of sources<sup>2</sup>. EventBridge can create rules that run on a schedule, either at regular intervals or at specific times and dates, and invoke targets such as Lambda functions<sup>3</sup>. This solution meets the requirements of creating a small application that makes the same API call once each day at a designated time, without requiring any infrastructure in the AWS Cloud or any operational overhead.

\* A. Use a Kubernetes cron job that runs on Amazon Elastic Kubernetes Service (Amazon EKS). This is incorrect. Amazon EKS is a fully managed Kubernetes service that allows you to run containerized applications on AWS<sup>4</sup>. Kubernetes cron jobs are tasks that run periodically on a given schedule<sup>5</sup>. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to provision and manage an EKS cluster, which would incur additional costs and complexity.

\* B. Use an Amazon Linux crontab scheduled job that runs on Amazon EC2. This is incorrect. Amazon EC2 is a web service that provides secure, resizable compute capacity in the cloud<sup>6</sup>. Crontab is a Linux utility that allows you to schedule commands or scripts to run automatically at a specified time or date<sup>7</sup>. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to provision and manage an EC2 instance, which would incur additional costs and complexity.

\* D. Use an AWS Batch job that is submitted to an AWS Batch job queue. This is incorrect. AWS Batch enables you to run batch computing workloads on the AWS or sequentially on compute environments<sup>8</sup>. Batch jobs are units of work that can be submitted to job queues, where they are executed in parallel. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to configure and manage an AWS Batch environment, which would incur additional costs and complexity.

**References:**

- ? 1: What is AWS Lambda? - AWS Lambda
- ? 2: What is Amazon EventBridge? - Amazon EventBridge
- ? 3: Creating an Amazon EventBridge rule that runs on a schedule - Amazon EventBridge
- ? 4: What is Amazon EKS? - Amazon EKS
- ? 5: CronJob - Kubernetes
- ? 6: What is Amazon EC2? - Amazon EC2
- ? 7: Crontab in Linux with 20 Useful Examples to Schedule Jobs - Tecmint
- ? 8: What is AWS Batch? - AWS Batch
- ? 9: Jobs - AWS Batch

**NEW QUESTION 92**

A company has a web application that runs on Amazon EC2 instances with a custom Amazon Machine Image (AMI) The company uses AWS CloudFormation to provision the application The application runs in the us-east-1 Region, and the company needs to deploy the application to the us-west-1 Region An attempt to create the AWS CloudFormation stack in us-west-1 fails. An error message states that the AMI ID does not exist. A developer must resolve this error with a solution that uses the least amount of operational overhead Which solution meets these requirements?

- A. Change the AWS CloudFormation templates for us-east-1 and us-west-1 to use an AWS AM
- B. Relaunch the stack for both Regions.
- C. Copy the custom AMI from us-east-1 to us-west-1. Update the AWS CloudFormation template for us-west-1 to refer to AMI ID for the copied AMI Relaunch the stack
- D. Build the custom AMI in us-west-1 Create a new AWS CloudFormation template to launch the stack in us-west-1 with the new AMI ID
- E. Manually deploy the application outside AWS CloudFormation in us-west-1.

**Answer: B**

**Explanation:**

<https://aws.amazon.com/blogs/aws/ec2-ami-copy-between-regions/>

**NEW QUESTION 93**

A company built an online event platform For each event the company organizes quizzes and generates leaderboards that are based on the quiz scores. The company stores the leaderboard data in Amazon DynamoDB and retains the data for 30 days after an event is complete The company then uses a scheduled job to delete the old leaderboard data The DynamoDB table is configured with a fixed write capacity. During the months when many events occur, the DynamoDB write API requests are throttled when the scheduled delete job runs. A developer must create a long-term solution that deletes the old leaderboard data and optimizes write throughput Which solution meets these requirements?

- A. Configure a TTL attribute for the leaderboard data
- B. Use DynamoDB Streams to schedule and delete the leaderboard data
- C. Use AWS Step Functions to schedule and delete the leaderboard data.
- D. Set a higher write capacity when the scheduled delete job runs

**Answer: A**

**Explanation:**

"deletes the item from your table without consuming any write throughput" <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/TTL.html>

**NEW QUESTION 94**

A company 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
- B. Add an EventBridge rule to the event bus of the main account that matches all EC2 instance lifecycle event
- C. Add the SQS queue as a target of the rule.
- D. Use the resource policies of the SQS queue in the main account to give each account permissions to write to that SQS queue
- E. Add to the Amazon EventBridge event bus of each account an EventBridge rule that matches all EC2 instance lifecycle event
- F. Add the SQS queue in the main account as a target of the rule.
- G. Write an AWS Lambda function that scans through all EC2 instances in the company accounts to detect EC2 instance lifecycle change
- H. 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
- I. Add an Amazon EventBridge scheduled rule that invokes the Lambda function every minute.
- J. Configure the permissions on the main account event bus to receive events from all accounts
- K. Create an Amazon EventBridge rule in each account to send all the EC2 instance lifecycle events to the main account event bus
- L. Add an EventBridge rule to the main account event bus that matches all EC2 instance lifecycle event
- M. 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>

**NEW QUESTION 97**

A developer is deploying an AWS Lambda function. The developer wants the ability to return to older versions of the function quickly and seamlessly.

How can the developer achieve this goal with the LEAST operational overhead?

- A. Use AWS OpsWorks to perform blue/green deployments.
- B. Use a function alias with different versions.
- C. Maintain deployment packages for older versions in Amazon S3.
- D. Use AWS CodePipeline for deployments and rollbacks.

**Answer:** B

**Explanation:**

A function alias is a pointer to a specific Lambda function version. You can use aliases to create different environments for your function, such as development, testing, and production. You can also use aliases to perform blue/green deployments by shifting traffic between two versions of your function gradually. This way, you can easily roll back to a previous version if something goes wrong, without having to redeploy your code or change your configuration. Reference: AWS Lambda function aliases

**NEW QUESTION 98**

A developer has created an AWS Lambda function that makes queries to an Amazon Aurora MySQL DB instance. When the developer performs a test the DB instance shows an error for too many connections.

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

- A. Create a read replica for the DB instance. Query the replica DB instance instead of the primary DB instance.
- B. Migrate the data to an Amazon DynamoDB database.
- C. Configure the Amazon Aurora MySQL DB instance for Multi-AZ deployment.
- D. Create a proxy in Amazon RDS Proxy. Query the proxy instead of the DB instance.

**Answer:** D

**Explanation:**

This solution will meet the requirements by using Amazon RDS Proxy, which is a fully managed, highly available database proxy for Amazon RDS that makes applications more scalable, more resilient to database failures, and more secure. The developer can create a proxy in Amazon RDS Proxy, which sits between the application

and the DB instance and handles connection management, pooling, and routing. The developer can query the proxy instead of the DB instance, which reduces the number of open connections to the DB instance and avoids errors for too many connections. Option A is not optimal because it will create a read replica for the DB instance, which may not solve the problem of too many connections as read replicas also have connection limits and may incur additional costs. Option B is not optimal because it will migrate the data to an Amazon DynamoDB database, which may introduce additional complexity and overhead for migrating and accessing data from a different database service. Option C is not optimal because it will configure the Amazon Aurora MySQL DB instance for Multi-AZ deployment, which may improve availability and durability of the DB instance but not reduce the number of connections.

References: [Amazon RDS Proxy], [Working with Amazon RDS Proxy]

**NEW QUESTION 101**

A developer has an application that is composed of many different AWS Lambda functions. The Lambda functions all use some of the same dependencies. To avoid security issues the developer is constantly updating the dependencies of all of the Lambda functions. The result is duplicated effort to reach function.

How can the developer keep the dependencies of the Lambda functions up to date with the LEAST additional complexity?

- A. Define a maintenance window for the Lambda functions to ensure that the functions get updated copies of the dependencies.
- B. Upgrade the Lambda functions to the most recent runtime version.

- C. Define a Lambda layer that contains all of the shared dependencies.
- D. Use an AWS CodeCommit repository to host the dependencies in a centralized location.

**Answer:** C

**Explanation:**

This solution allows the developer to keep the dependencies of the Lambda functions up to date with the least additional complexity because it eliminates the need to update each function individually. A Lambda layer is a ZIP archive that contains libraries, custom runtimes, or other dependencies. The developer can create a layer that contains all of the shared dependencies and attach it to multiple Lambda functions. When the developer updates the layer, all of the functions that use the layer will have access to the latest version of the dependencies.

Reference: [AWS Lambda layers]

**NEW QUESTION 106**

A company wants to deploy and maintain static websites on AWS. Each website's source code is hosted in one of several version control systems, including AWS CodeCommit, Bitbucket, and GitHub.

The company wants to implement phased releases by using development, staging, user acceptance testing, and production environments in the AWS Cloud. Deployments to each environment must be started by code merges on the relevant Git branch. The company wants to use HTTPS for all data exchange. The company needs a solution that does not require servers to run continuously.

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

- A. Host each website by using AWS Amplify with a serverless backen
- B. Conned the repository branches that correspond to each of the desired environment
- C. Start deployments by merging code changes to a desired branch.
- D. Host each website in AWS Elastic Beanstalk with multiple environment
- E. Use the EB CLI to link each repository branc
- F. Integrate AWS CodePipeline to automate deployments from version control code merges.
- G. Host each website in different Amazon S3 buckets for each environmen
- H. Configure AWS CodePipeline to pull source code from version contro
- I. Add an AWS CodeBuild stage to copy source code to Amazon S3.
- J. Host each website on its own Amazon EC2 instanc
- K. Write a custom deployment script to bundle each website's static asset
- L. Copy the assets to Amazon EC2. Set up a workflow to run the script when code is merged.

**Answer:** A

**Explanation:**

AWS Amplify is a set of tools and services that enables developers to build and deploy full-stack web and mobile applications that are powered by AWS. AWS Amplify supports hosting static websites on Amazon S3 and Amazon CloudFront, with HTTPS enabled by default. AWS Amplify also integrates with various version control systems, such as AWS CodeCommit, Bitbucket, and GitHub, and allows developers to connect different branches to different environments. AWS Amplify automatically builds and deploys the website whenever code changes are merged to a connected branch, enabling phased releases with minimal operational overhead. Reference: AWS Amplify Console

**NEW QUESTION 107**

A company's developer has deployed an application in AWS by using AWS CloudFormation The CloudFormation stack includes parameters in AWS Systems Manager Parameter Store that the application uses as configuration settings. The application can modify the parameter values

When the developer updated the stack to create additional resources with tags, the developer noted that the parameter values were reset and that the values ignored the latest changes made by the application. The developer needs to change the way the company deploys the CloudFormation stack. The developer also needs to avoid resetting the parameter values outside the stack.

Which solution will meet these requirements with the LEAST development effort?

- A. Modify the CloudFormation stack to set the deletion policy to Retain for the Parameter Store parameters.
- B. Create an Amazon DynamoDB table as a resource in the CloudFormation stack to hold configuration data for the application Migrate the parameters that the application is modifying from Parameter Store to the DynamoDB table
- C. Create an Amazon RDS DB instance as a resource in the CloudFormation stac
- D. Create a table in the database for parameter configuratio
- E. Migrate the parameters that the application is modifying from Parameter Store to the configuration table
- F. Modify the CloudFormation stack policy to deny updates on Parameter Store parameters

**Answer:** D

**Explanation:**

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/protect-stack-resources.html#stack-policy-samples>

**NEW QUESTION 108**

A company has an ecommerce application. To track product reviews, the company's development team uses an Amazon DynamoDB table. Every record includes the following

- A Review ID a 16-digit universally unique identifier (UUID)
- A Product ID and User ID 16 digit UUIDs that reference other tables
- A Product Rating on a scale of 1-5
- An optional comment from the user

The table partition key is the Review ID. The most performed query against the table is to find the 10 reviews with the highest rating for a given product.

Which index will provide the FASTEST response for this query"?

- A. A global secondary index (GSI) with Product ID as the partition key and Product Rating as the sort key
- B. A global secondary index (GSI) with Product ID as the partition key and Review ID as the sort key
- C. A local secondary index (LSI) with Product ID as the partition key and Product Rating as the sort key
- D. A local secondary index (LSI) with Review ID as the partition key and Product ID as the sort key

**Answer:** A

**Explanation:**

This solution allows the fastest response for the query because it enables the query to use a single partition key value (the Product ID) and a range of sort key values (the Product Rating) to find the matching items. A global secondary index (GSI) is an index that has a partition key and an optional sort key that are different from those on the base table. A GSI can be created at any time and can be queried or scanned independently of the base table. A local secondary index (LSI) is an index that has the same partition key as the base table, but a different sort key. An LSI can only be created when the base table is created and must be queried together with the base table partition key. Using a GSI with Product ID as the partition key and Review ID as the sort key will not allow the query to use a range of sort key values to find the highest ratings. Using an LSI with Product ID as the partition key and Product Rating as the sort key will not work because Product ID is not the partition key of the base table. Using an LSI with Review ID as the partition key and Product ID as the sort key will not allow the query to use a single partition key value to find the matching items.

Reference: [Global Secondary Indexes], [Querying]

**NEW QUESTION 112**

A developer is working on a web application that uses Amazon DynamoDB as its data store. The application has two DynamoDB tables: one table that is named `artists` and one table that is named `songs`. The `artists` table has `artistName` as the partition key. The `songs` table has `songName` as the partition key and `artistName` as the sort key.

The table usage patterns include the retrieval of multiple songs and artists in a single database operation from the webpage. The developer needs a way to retrieve this information with minimal network traffic and optimal application performance.

Which solution will meet these requirements?

- A. Perform a `BatchGetItem` operation that returns items from the two tables.
- B. Use the list of `songName/artistName` keys for the `songs` table and the list of `artistName` key for the `artists` table.
- C. Create a local secondary index (LSI) on the `songs` table that uses `artistName` as the partition key. Perform a query operation for each `artistName` on the `songs` table that filters by the list of `songName`. Perform a query operation for each `artistName` on the `artists` table.
- D. Perform a `BatchGetItem` operation on the `songs` table that uses the `songName/artistName` key.
- E. Perform a `BatchGetItem` operation on the `artists` table that uses `artistName` as the key.
- F. Perform a `Scan` operation on each table that filters by the list of `songName/artistName` for the `songs` table and the list of `artistName` in the `artists` table.

**Answer:** A

**Explanation:**

`BatchGetItem` can return one or multiple items from one or more tables. For reference, check the link below:

[https://docs.aws.amazon.com/amazondynamodb/latest/APIReference/API\\_BatchGetItem.html](https://docs.aws.amazon.com/amazondynamodb/latest/APIReference/API_BatchGetItem.html)

**NEW QUESTION 113**

A developer uses AWS CloudFormation to deploy an Amazon API Gateway API and an AWS Step Functions state machine. The state machine must reference the API Gateway API after the CloudFormation template is deployed. The developer needs a solution that uses the state machine to reference the API Gateway endpoint.

Which solution will meet these requirements MOST cost-effectively?

- A. Configure the CloudFormation template to reference the API endpoint in the `DefinitionSubstitutions` property for the `AWS::StepFunctions::StateMachine` resource.
- B. Configure the CloudFormation template to store the API endpoint in an environment variable for the `AWS::StepFunctions::StateMachine` resource. Configure the state machine to reference the environment variable.
- C. Configure the CloudFormation template to store the API endpoint in a standard `AWS::SecretsManager::Secret` resource. Configure the state machine to reference the resource.
- D. Configure the CloudFormation template to store the API endpoint in a standard `AWS::AppConfig::ConfigurationProfile` resource. Configure the state machine to reference the resource.

**Answer:** A

**Explanation:**

The most cost-effective solution is to use the `DefinitionSubstitutions` property of the `AWS::StepFunctions::StateMachine` resource to inject the API endpoint as a variable in the state machine definition. This way, the developer can use the intrinsic function

`Fn::GetAtt` to get the API endpoint from the `AWS::ApiGateway::RestApi` resource, and pass it to the state machine without creating any additional resources or environment variables. The other solutions involve creating and managing extra resources, such as `Secrets Manager` secrets or `AppConfig` configuration profiles, which incur additional costs and complexity. References:

- ? `AWS::StepFunctions::StateMachine` - AWS CloudFormation
- ? Call API Gateway with Step Functions - AWS Step Functions
- ? `amazon-web-services aws-api-gateway terraform aws-step-functions`

**NEW QUESTION 116**

A company is migrating legacy internal applications to AWS. Leadership wants to rewrite the internal employee directory to use native AWS services. A developer needs to create a solution for storing employee contact details and high-resolution photos for use with the new application.

Which solution will enable the search and retrieval of each employee's individual details and high-resolution photos using AWS APIs?

- A. Encode each employee's contact information and photos using Base64. Store the information in an Amazon DynamoDB table using a sort key.
- B. Store each employee's contact information in an Amazon DynamoDB table along with the object keys for the photos stored in Amazon S3.
- C. Use Amazon Cognito user pools to implement the employee directory in a fully managed software-as-a-service (SaaS) method.
- D. Store employee contact information in an Amazon RDS DB instance with the photos stored in Amazon Elastic File System (Amazon EFS).

**Answer:** B

**Explanation:**

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and consistent performance with seamless scalability. The developer can store each employee's contact information in a DynamoDB table along with the object keys for the photos stored in Amazon S3. Amazon S3 is an object storage

service that offers industry-leading scalability, data availability, security, and performance. The developer can use AWS APIs to search and retrieve the employee details and photos from DynamoDB and S3.

References:

? [Amazon DynamoDB]

? [Amazon Simple Storage Service (S3)]

#### NEW QUESTION 119

A developer deployed an application to an Amazon EC2 instance. The application needs to know the public IPv4 address of the instance. How can the application find this information?

Query the instance metadata from `http://169.254.169.254/latest/meta-data/`.

A. Query the instance user data from `http://169.254.169.254/latest/user-data/`

C. Query the Amazon Machine Image (AMI) information from `http://169.254.169.254/latest/meta-data/ami/`.

D. Check the hosts file of the operating system

**Answer:** A

#### Explanation:

The instance metadata service provides information about the EC2 instance, including the public IPv4 address, which can be obtained by querying the endpoint `http://169.254.169.254/latest/meta-data/public-ipv4`. References

? Instance metadata and user data

? Get Public IP Address on current EC2 Instance

? Get the public ip address of your EC2 instance quickly

#### NEW QUESTION 124

A company's website runs on an Amazon EC2 instance and uses Auto Scaling to scale the environment during peak times. Website users across the world are experiencing high latency due to static content on the EC2 instance, even during non-peak hours.

Which combination of steps will resolve the latency issue? (Select TWO)

A. Double the Auto Scaling group's maximum number of servers

B. Host the application code on AWS Lambda

C. Scale vertically by resizing the EC2 instances

D. Create an Amazon CloudFront distribution to cache the static content

E. Store the application's static content in Amazon S3

**Answer:** DE

#### Explanation:

The combination of steps that will resolve the latency issue is to create an Amazon CloudFront distribution to cache the static content and store the application's static content in Amazon S3. This way, the company can use CloudFront to deliver the static content from edge locations that are closer to the website users, reducing latency and improving performance. The company can also use S3 to store the static content reliably and cost-effectively, and integrate it with CloudFront easily. The other options either do not address the latency issue, or are not necessary or feasible for the given scenario.

Reference: Using Amazon S3 Origins and Custom Origins for Web Distributions

#### NEW QUESTION 125

A company runs a batch processing application by using AWS Lambda functions and Amazon API Gateway APIs with deployment stages for development, user acceptance testing and production. A development team needs to configure the APIs in the deployment stages to connect to third-party service endpoints. Which solution will meet this requirement?

A. Store the third-party service endpoints in Lambda layers that correspond to the stage

B. Store the third-party service endpoints in API Gateway stage variables that correspond to the stage

C. Encode the third-party service endpoints as query parameters in the API Gateway request URL.

D. Store the third-party service endpoint for each environment in AWS AppConfig

**Answer:** B

#### Explanation:

API Gateway stage variables are name-value pairs that can be defined as configuration attributes associated with a deployment stage of a REST API. They act like environment variables and can be used in the API setup and mapping templates. For example, the development team can define a stage variable named `endpoint` and assign it different values for each stage, such as `dev.example.com` for development, `uat.example.com` for user acceptance testing, and `prod.example.com` for production. Then, the team can use the stage variable value in the integration request URL, such as `http://${stageVariables.endpoint}/api`. This way, the team can use the same API setup with different endpoints at each stage by resetting the stage variable value. The other solutions are either not feasible or not cost-effective. Lambda layers are used to package and load dependencies for Lambda functions, not for storing endpoints. Encoding the endpoints as query parameters would expose them to the public and make the request URL unnecessarily long. Storing the endpoints in AWS AppConfig would incur additional costs and complexity, and would require additional logic to retrieve the values from the configuration store. References

? Using Amazon API Gateway stage variables

? Setting up stage variables for a REST API deployment

? Setting stage variables using the Amazon API Gateway console

#### NEW QUESTION 128

A company is developing an ecommerce application that uses Amazon API Gateway APIs. The application uses AWS Lambda as a backend. The company needs to test the code in a dedicated, monitored test environment before the company releases the code to the production environment.

Which solution will meet these requirements?

A. Use a single stage in API Gateway

B. Create a Lambda function for each environment

C. Configure API clients to send a query parameter that indicates the environment and the specific lambda function.

D. Use multiple stages in API Gateway

E. Create a single Lambda function for all environments

- F. Add different code blocks for different environments in the Lambda function based on Lambda environments variables.
- G. Use multiple stages in API Gateway
- H. Create a Lambda function for each environment
- I. Configure API Gateway stage variables to route traffic to the Lambda function in different environments.
- J. Use a single stage in API Gateway
- K. Configure a API client to send a query parameter that indicated the environment
- L. Add different code blocks for different environments in the Lambda function to match the value of the query parameter.

**Answer:** C

**Explanation:**

The solution that will meet the requirements is to use multiple stages in API Gateway. Create a Lambda function for each environment. Configure API Gateway stage variables to route traffic to the Lambda function in different environments. This way, the company can test the code in a dedicated, monitored test environment before releasing it to the production environment. The company can also use stage variables to specify the Lambda function version or alias for each stage, and avoid hard-coding the Lambda function name in the API Gateway integration. The other options either involve using a single stage in API Gateway, which does not allow testing in different environments, or adding different code blocks for different environments in the Lambda function, which increases complexity and maintenance.

Reference: Set up stage variables for a REST API in API Gateway

**NEW QUESTION 129**

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
- C. 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.
- D. Use Amazon API Gateway and an AWS Lambda function to upload and download file
- E. Validate each request in the Lambda function before performing the requested operation.
- F. 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>

**NEW QUESTION 130**

A developer is designing a serverless application with two AWS Lambda functions to process photos. One Lambda function stores objects in an Amazon S3 bucket and stores the associated metadata in an Amazon DynamoDB table. The other Lambda function fetches the objects from the S3 bucket by using the metadata from the DynamoDB table. Both Lambda functions use the same Python library to perform complex computations and are approaching the quota for the maximum size of zipped deployment packages.

What should the developer do to reduce the size of the Lambda deployment packages with the LEAST operational overhead?

- A. Package each Python library in its own .zip file archive
- B. Deploy each Lambda function with its own copy of the library.
- C. Create a Lambda layer with the required Python library
- D. Use the Lambda layer in both Lambda functions.
- E. Combine the two Lambda functions into one Lambda function
- F. Deploy the Lambda function as a single .zip file archive.
- G. Download the Python library to an S3 bucket
- H. Program the Lambda functions to reference the object URLs.

**Answer:** B

**Explanation:**

AWS Lambda is a service that lets developers run code without provisioning or managing servers. Lambda layers are a distribution mechanism for libraries, custom runtimes, and other dependencies. The developer can create a Lambda layer with the

required Python library and use the layer

in both Lambda functions. This will reduce the size of the Lambda deployment packages and avoid reaching the quota for the maximum size of zipped deployment packages. The developer can also benefit from using layers to manage dependencies separately from function code.

References:

? [What Is AWS Lambda? - AWS Lambda]

? [AWS Lambda Layers - AWS Lambda]

#### NEW QUESTION 131

A developer wants to deploy a new version of an AWS Elastic Beanstalk application. During deployment, the application must maintain full capacity and avoid service interruption. Additionally, the developer must minimize the cost of additional resources that support the deployment.

Which deployment method should the developer use to meet these requirements?

A.

All at once

B. Rolling with additional batch

C. Blue/green

D. Immutable

**Answer: D**

#### Explanation:

The immutable deployment method is the best option for this scenario, because it meets the requirements of maintaining full capacity, avoiding service interruption, and minimizing the cost of additional resources.

The immutable deployment method creates a new set of instances in a separate Auto Scaling group and deploys the new version of the application to them. Then, it swaps the new instances with the old ones and terminates the old instances. This way, the application maintains full capacity during the deployment and avoids any downtime. The cost of additional resources is also minimized, because the new instances are only created for a short time and then replaced by the old ones.

The other deployment methods do not meet all the requirements:

? The all at once method deploys the new version to all instances simultaneously, which causes a short period of downtime and reduced capacity.

? The rolling with additional batch method deploys the new version in batches, but for the first batch it creates new instances instead of using the existing ones.

This increases the cost of additional resources and reduces the capacity of the original environment.

? The blue/green method creates a new environment with a new set of instances and deploys the new version to them. Then, it swaps the URLs between the old and new environments. This method maintains full capacity and avoids service interruption, but it also increases the cost of additional resources significantly, because it duplicates the entire environment.

#### NEW QUESTION 136

A developer is creating a serverless application that uses an AWS Lambda function. The developer will use AWS CloudFormation to deploy the application. The application will write logs to Amazon CloudWatch Logs. The developer has created a log group in a CloudFormation template for the application to use. The developer needs to modify the CloudFormation template to make the name of the log group available to the application at runtime. Which solution will meet this requirement?

- A. Use the `AWS::Include` transform in CloudFormation to provide the log group's name to the application.
- B. Pass the log group's name to the application in the user data section of the CloudFormation template.
- C. Use the CloudFormation template's Mappings section to specify the log group's name for the application.
- D. Pass the log group's Amazon Resource Name (ARN) as an environment variable to the Lambda function.

**Answer:** D

**Explanation:**

FunctionName: MyLambdaFunction Code:

S3Bucket: your-lambda-code-bucket S3Key: lambda-code.zip

Runtime: nodejs14.x # Specify the desired runtime for your Lambda function Environment:

Variables:

LOG\_GROUP\_NAME: !Ref MyLogGroup <https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-logs-loggroup.html>

**NEW QUESTION 138**

A company is planning to use AWS CodeDeploy to deploy an application to Amazon Elastic Container Service (Amazon ECS). During the deployment of a new version of the application, the company initially must expose only 10% of live traffic to the new version of the deployed application. Then, after 15 minutes elapse, the company must route all the remaining live traffic to the new version of the deployed application.

Which CodeDeploy predefined configuration will meet these requirements?

- A. CodeDeployDefault\_ECSCanary10Percent15Minutes
- B. CodeDeployDefault\_LambdaCanary10Percent5Minutes
- C. CodeDeployDefault\_LambdaCanary10Percent15Minutes
- D. CodeDeployDefault\_ECSELinear10PercentEvery1 Minutes

**Answer:** A

**Explanation:**

The predefined configuration "CodeDeployDefault\_ECSCanary10Percent15Minutes" is designed for Amazon Elastic Container Service (Amazon ECS) deployments and meets the specified requirements. It will perform a canary deployment, which means it will initially route 10% of live traffic to the new version of the application, and then after 15 minutes elapse, it will automatically route all the remaining live traffic to the new version. This gradual deployment approach allows

the company to verify the health and performance of the new version with a small portion of traffic before fully deploying it to all users.

#### NEW QUESTION 140

An ecommerce application is running behind an Application Load Balancer. A developer observes some unexpected load on the application during non-peak hours. The developer wants to analyze patterns for the client IP addresses that use the application. Which HTTP header should the developer use for this analysis?

- A. The X-Forwarded-Proto header
- B. The X-F Forwarded-Host header
- C. The X-Forwarded-For header
- D. The X-Forwarded-Port header

**Answer:** C

#### Explanation:

The HTTP header that the developer should use for this analysis is the X-Forwarded-For header. This header contains the IP address of the client that made the request to the Application Load Balancer. The developer can use this header to analyze patterns for the client IP addresses that use the application. The other headers either contain information about the protocol, host, or port of the request, which are not relevant for the analysis.

Reference: How Application Load Balancer works with your applications

#### NEW QUESTION 142

An ecommerce company is using an AWS Lambda function behind Amazon API Gateway

as its application tier. To process orders during checkout, the application calls a POST API from the frontend. The POST API invokes the Lambda function asynchronously. In rare situations, the application has not processed orders. The Lambda application logs show no errors or failures. What should a developer do to solve this problem?

- A. Inspect the frontend logs for API failure
- B. Call the POST API manually by using the requests from the log file.
- C. Create and inspect the Lambda dead-letter queue
- D. Troubleshoot the failed function
- E. Reprocess the events.
- F. Inspect the Lambda logs in Amazon CloudWatch for possible error

- G. Fix the errors.
- H. Make sure that caching is disabled for the POST API in API Gateway.

**Answer:** B

**Explanation:**

The solution that will solve this problem is to create and inspect the Lambda dead-letter queue. Troubleshoot the failed functions. Reprocess the events. This way, the developer can identify and fix any issues that caused the Lambda function to fail when invoked asynchronously by API Gateway. The developer can also reprocess any orders that were not processed due to failures. The other options either do not address the root cause of the problem, or do not help recover from failures.

Reference: Asynchronous invocation

**NEW QUESTION 143**

A company developed an API application on AWS by using Amazon CloudFront, Amazon API Gateway, and AWS Lambda. The API has a minimum of four requests every second. A developer notices that many API users run the same query by using the POST method. The developer wants to cache the POST request to optimize the API resources.

Which solution will meet these requirements?

- A. Configure the CloudFront cache. Update the application to return cached content based upon the default request headers.
- B. Override the cache method in the selected stage of API Gateway. Select the POST method.
- C. Save the latest request response in Lambda /tmp directory. Update the Lambda function to check the /tmp directory.
- D. Save the latest request in AWS Systems Manager Parameter Store. Modify the Lambda function to take the latest request response from Parameter Store.

**Answer:** A

**Explanation:**

This solution will meet the requirements by using Amazon CloudFront, which is a content delivery network (CDN) service that speeds up the delivery of web content and APIs to end users. The developer can configure the CloudFront cache, which is a set of edge locations that store copies of popular or recently accessed content close to the viewers. The developer can also update the application to return cached content based upon the default request headers, which are a set of HTTP headers that CloudFront automatically forwards to the origin server and uses to determine whether an object in an edge location is still valid. By caching the POST requests, the developer can optimize the API resources and reduce the latency for repeated queries. Option B is not optimal because it will override the cache method in the selected stage of API Gateway, which is not possible or effective as API Gateway does not support caching for POST methods by default. Option C is not optimal because it will save the latest request response in Lambda /tmp directory, which is a local storage space that is available for each Lambda function invocation, not a cache that can be shared across multiple invocations or requests. Option D is not optimal because it will save the latest request in AWS Systems Manager Parameter Store, which is a service that provides secure and scalable storage for configuration data and secrets, not a cache for API responses.

References: [Amazon CloudFront], [Caching Content Based on Request Headers]

**NEW QUESTION 147**

An application uses an Amazon EC2 Auto Scaling group. A developer notices that EC2 instances are taking a long time to become available during scale-out events. The UserData script is taking a long time to run.

The developer must implement a solution to decrease the time that elapses before an EC2 instance becomes available. The solution must make the most recent version of the application available at all times and must apply all available security updates. The solution also must minimize the number of images that are created. The images must be validated.

Which combination of steps should the developer take to meet these requirements? (Choose two.)

- A. Use EC2 Image Builder to create an Amazon Machine Image (AMI). Install all the patches and agents that are needed to manage and run the application.
- B. Update the Auto Scaling group launch configuration to use the AMI.
- C. Use EC2 Image Builder to create an Amazon Machine Image (AMI). Install the latest version of the application and all the patches and agents that are needed to manage and run the application.
- D. Update the Auto Scaling group launch configuration to use the AMI.
- E. Set up AWS CodeDeploy to deploy the most recent version of the application at runtime.
- F. Set up AWS CodePipeline to deploy the most recent version of the application at runtime.
- G. Remove any commands that perform operating system patching from the UserData script.

**Answer:** BE

**Explanation:**

AWS CloudFormation is a service that enables developers to model and provision AWS resources using templates. The developer can use the following steps to avoid accidental database deletion in the future:

- ? Set up AWS CodeDeploy to deploy the most recent version of the application at runtime. This will ensure that the application code is always up to date and does not depend on the AMI.
- ? Remove any commands that perform operating system patching from the UserData script. This will reduce the time that the UserData script takes to run and speed up the instance launch process.

References:

- ? [What Is AWS CloudFormation? - AWS CloudFormation]
- ? [What Is AWS CodeDeploy? - AWS CodeDeploy]
- ? [Running Commands on Your Linux Instance at Launch - Amazon Elastic Compute Cloud]

**NEW QUESTION 151**

A company hosts its application on AWS. The application runs on an Amazon Elastic Container Service (Amazon ECS) cluster that uses AWS Fargate. The cluster runs behind an Application Load Balancer. The application stores data in an Amazon Aurora database. A developer encrypts and manages database credentials inside the application.

The company wants to use a more secure credential storage method and implement periodic credential rotation.

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

- A. Migrate the secret credentials to Amazon RDS parameter group
- B. Encrypt the parameter by using an AWS Key Management Service (AWS KMS) key. Turn on secret rotation.
- C. Use IAM policies and roles to grant AWS KMS permissions to access Amazon RDS.
- D. Migrate the credentials to AWS Systems Manager Parameter Store.
- E. Encrypt the parameter by using an AWS Key Management Service (AWS KMS) key.
- F. Turn on secret rotation.
- G. Use IAM policies and roles to grant Amazon ECS Fargate permissions to access to AWS Secrets Manager.
- H. Migrate the credentials to ECS Fargate environment variable.
- I. Encrypt the credentials by using an AWS Key Management Service (AWS KMS) key. Turn on secret rotation.
- J. Use IAM policies and roles to grant Amazon ECS Fargate permissions to access to AWS Secrets Manager.
- K. Migrate the credentials to AWS Secrets Manager.

L. Encrypt the credentials by using an AWS Key Management Service (AWS KMS) key Turn on secret rotation Use IAM policies and roles to grant Amazon ECS Fargate permissions to access to AWS Secrets Manager by using keys.

**Answer:** D

**Explanation:**

AWS Secrets Manager is a service that helps you store, distribute, and rotate secrets securely. You can use Secrets Manager to migrate your credentials from your application code to a secure and encrypted storage. You can also enable automatic rotation of your secrets by using AWS Lambda functions or custom logic. You can use IAM policies and roles to grant your Amazon ECS Fargate tasks permissions to access your secrets from Secrets Manager. This solution minimizes the operational overhead of managing your credentials and enhances the security of your application. References

? AWS Secrets Manager: Store, Distribute, and Rotate Credentials Securely | AWS

News Blog

? Why You Should Audit and Rotate Your AWS Credentials Periodically - Cloud Academy

? Top 5 AWS root account best practices - TheServerSide

**NEW QUESTION 154**

A developer maintains applications that store several secrets in AWS Secrets Manager. The applications use secrets that have changed over time. The developer needs to identify required secrets that are still in use. The developer does not want to cause any application downtime.

What should the developer do to meet these requirements?

- A. Configure an AWS CloudTrail log file delivery to an Amazon S3 bucket
- B. Create an Amazon CloudWatch alarm for the GetSecretValue
- C. Secrets Manager API operation requests
- D. Create a secrets manager-secret-unused AWS Config managed rule
- E. Create an Amazon EventBridge rule to initiate notification when the AWS Config managed rule is met.
- F. Deactivate the applications secrets and monitor the applications error logs temporarily.
- G. Configure AWS X-Ray for the application
- H. Create a sampling rule to match the

GetSecretValue Secrets Manager API operation requests.

**Answer:** B

**Explanation:**

This solution will meet the requirements by using AWS Config to monitor and evaluate whether Secrets Manager secrets are unused or have been deleted, based on specified time periods. The secrets manager-secret-unused managed rule is a predefined rule that checks whether Secrets Manager secrets have been rotated within a specified number of days or have been deleted within a specified number of days after last accessed date. The Amazon EventBridge rule will trigger a notification when the AWS Config managed rule is met, alerting the developer about unused secrets that can be removed without causing application downtime. Option A is not optimal because it will use AWS CloudTrail log file delivery to an Amazon S3 bucket, which will incur additional costs and complexity for storing and analyzing log files that may not contain relevant information about secret usage. Option C is not optimal because it will deactivate the application secrets and monitor the application error logs temporarily, which will cause application downtime and potential data loss. Option D is not optimal because it will use AWS X-Ray to trace secret usage, which will introduce additional overhead and latency for instrumenting and sampling requests that may not be related to secret usage. References: [AWS Config Managed Rules], [Amazon EventBridge]

**NEW QUESTION 155**

A company is preparing to migrate an application to the company's first AWS environment Before this migration, a developer is creating a proof-of-concept application to validate a model for building and deploying container-based applications on AWS.

Which combination of steps should the developer take to deploy the containerized proof-of-concept application with the LEAST operational effort? (Select TWO.)

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

To deploy a containerized application on AWS with the least operational effort, the developer should package the application into a container image by using the Docker CLI and upload the image to Amazon ECR, which is a fully managed container registry service. Then, the developer should deploy the application to Amazon ECS on AWS Fargate, which is a serverless compute engine for containers that eliminates the need to provision and manage servers or clusters. Amazon ECS will automatically scale, load balance, and monitor the application. References

- ? How to Deploy Docker Containers | AWS
- ? Deploy a Web App Using AWS App Runner
- ? How to Deploy Containerized Apps on AWS Using ECR and Docker

#### NEW QUESTION 156

A developer is building a serverless application that is based on AWS Lambda. The developer initializes the AWS software development kit (SDK) outside of the Lambda handler function.

What is the PRIMARY benefit of this action?

- A. Improves legibility and stylistic convention
- B. Takes advantage of runtime environment reuse
- C. Provides better error handling
- D. Creates a new SDK instance for each invocation

**Answer:** B

#### **Explanation:**

This benefit occurs when initializing the AWS SDK outside of the Lambda handler function because it allows the SDK instance to be reused across multiple invocations of the same function. This can improve performance and reduce latency by avoiding unnecessary initialization overhead. If the SDK is initialized inside the handler function, it will create a new SDK instance for each invocation, which can increase memory usage and execution time.

Reference: [AWS Lambda execution environment], [Best Practices for Working with AWS Lambda Functions]

#### NEW QUESTION 160

.....

## THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual DVA-C02 Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the DVA-C02 Product From:

<https://www.2passeasy.com/dumps/DVA-C02/>

### Money Back Guarantee

#### **DVA-C02 Practice Exam Features:**

- \* DVA-C02 Questions and Answers Updated Frequently
- \* DVA-C02 Practice Questions Verified by Expert Senior Certified Staff
- \* DVA-C02 Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- \* DVA-C02 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year