



**Amazon**

## **Exam Questions DVA-C02**

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 needs 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 Systems Manager Parameter Store vs. Secrets Manager vs. Environment Variables in Lambda, when to use which

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

### NEW QUESTION 2

A developer wants to add request validation to a production environment Amazon API Gateway API. The developer needs to test the changes before the API is deployed to the production environment. For the test, the developer will send test requests to the API through a testing tool. Which solution will meet these requirements with the LEAST operational overhead?

- A. Export the existing API to an OpenAPI file.
- B. Create a new API. Modify the new API to add request validation.
- C. Import the OpenAPI file.
- D. Perform the test.
- E. Modify the existing API to add request validation.
- F. Deploy the existing API to production.
- G. Modify the existing API to add request validation.
- H. Deploy the updated API to a new API Gateway stage.
- I. Perform the test.
- J. Deploy the updated API to the API Gateway production stage.
- K. Create a new API.
- L. Add the necessary resources and methods, including new request validation.
- M. Perform the test.
- N. Modify the existing API to add request validation.
- O. Deploy the existing API to production.
- P. Clone the existing API.
- Q. Modify the new API to add request validation.
- R. Perform the test.
- S. Modify the existing API to add request validation.
- T. Deploy the existing API to production.

**Answer: B**

#### Explanation:

Amazon API Gateway allows you to create, deploy, and manage a RESTful API to expose backend HTTP endpoints, AWS Lambda functions, or other AWS services<sup>1</sup>. You can use API Gateway to perform basic validation of an API request before proceeding with the integration request<sup>1</sup>. When the validation fails, API Gateway immediately fails the request, returns a 400 error response to the caller, and publishes the validation results in CloudWatch Logs<sup>1</sup>.

To test changes before deploying to a production environment, you can modify the existing API to add request validation and deploy the updated API to a new API Gateway stage<sup>1</sup>. This allows you to perform tests without affecting the production environment. Once testing is complete and successful, you can then deploy the updated API to the API Gateway production stage<sup>1</sup>.

This approach has the least operational overhead as it avoids unnecessary creation of new APIs or exporting and importing of APIs. It leverages the existing infrastructure and only requires changes in the configuration of the existing API<sup>1</sup>.

### NEW QUESTION 3

A developer is creating a mobile app that calls a backend service by using an Amazon API Gateway REST API. For integration testing during the development phase, the developer wants to simulate different backend responses without invoking the backend service. Which solution will meet these requirements with the LEAST operational overhead?

- A. Create an AWS Lambda function.
- B. Use API Gateway proxy integration to return constant HTTP responses.
- C. Create an Amazon EC2 instance that serves the backend REST API by using an AWS CloudFormation template.
- D. Customize the API Gateway stage to select a response type based on the request.
- E. Use a request mapping template to select the mock integration response.

**Answer: D**

#### Explanation:

Amazon API Gateway supports mock integration responses, which are predefined responses that can be returned without sending requests to a backend service. Mock integration responses can be used for testing or prototyping purposes, or for simulating different backend responses based on certain conditions. A request

mapping template can be used to select a mock integration response based on an expression that evaluates some aspects of the request, such as headers, query strings, or body content. This solution does not require any additional resources or code changes and has the least operational overhead. Reference: Set up mock integrations for an API Gateway REST API

<https://docs.aws.amazon.com/apigateway/latest/developerguide/how-to-mock-integration.html>

#### NEW QUESTION 4

A company has an application that uses Amazon Cognito user pools as an identity provider. The company must secure access to user records. The company has set up multi-factor authentication (MFA). The company also wants to send a login activity notification by email every time a user logs in. What is the MOST operationally efficient solution that meets this requirement?

- A. Create an AWS Lambda function that uses Amazon Simple Email Service (Amazon SES) to send the email notification
- B. Add an Amazon API Gateway API to invoke the function
- C. Call the API from the client side when login confirmation is received.
- D. Create an AWS Lambda function that uses Amazon Simple Email Service (Amazon SES) to send the email notification
- E. Add an Amazon Cognito post authentication Lambda trigger for the function.
- F. Create an AWS Lambda function that uses Amazon Simple Email Service (Amazon SES) to send the email notification
- G. Create an Amazon CloudWatch Logs log subscription filter to invoke the function based on the login status.
- H. Configure Amazon Cognito to stream all logs to Amazon Kinesis Data Firehose
- I. Create an AWS Lambda function to process the streamed logs and to send the email notification based on the login status of each user.

**Answer:** B

#### Explanation:

Amazon Cognito user pools support Lambda triggers, which are custom functions that can be executed at various stages of the user pool workflow. A post authentication Lambda trigger can be used to perform custom actions after a user is authenticated, such as sending an email notification. Amazon SES is a cloud-based email sending service that can be used to send transactional or marketing emails. A Lambda function can use the Amazon SES API to send an email to the user's email address after the user logs in successfully. Reference: Post authentication Lambda trigger

#### NEW QUESTION 5

A developer maintains an Amazon API Gateway REST API. Customers use the API through a frontend UI and Amazon Cognito authentication. The developer has a new version of the API that contains new endpoints and backward-incompatible interface changes. The developer needs to provide beta access to other developers on the team without affecting customers. Which solution will meet these requirements with the LEAST operational overhead?

- A. Define a development stage on the API Gateway API
- B. Instruct the other developers to point the endpoints to the development stage.
- C. Define a new API Gateway API that points to the new API application code
- D. Instruct the other developers to point the endpoints to the new API.
- E. Implement a query parameter in the API application code that determines which code version to call.
- F. Specify new API Gateway endpoints for the API endpoints that the developer wants to add.

**Answer:** A

#### Explanation:

Amazon API Gateway is a service that enables developers to create, publish, maintain, monitor, and secure APIs at any scale. The developer can define a development stage on the API Gateway API and instruct the other developers to point the endpoints to the development stage. This way, the developer can provide beta access to the new version of the API without affecting customers who use the production stage. This solution will meet the requirements with the least operational overhead.

References:

? [What Is Amazon API Gateway? - Amazon API Gateway]

? [Set up a Stage in API Gateway - Amazon API Gateway]

#### NEW QUESTION 6

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 7

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 8**

A developer is testing a new file storage application that uses an Amazon CloudFront distribution to serve content from an Amazon S3 bucket. The distribution accesses the S3 bucket by using an origin access identity (OAI). The S3 bucket's permissions explicitly deny access to all other users. The application prompts users to authenticate on a login page and then uses signed cookies to allow users to access their personal storage directories. The developer has configured the distribution to use its default cache behavior with restricted viewer access and has set the origin to point to the S3 bucket. However, when the developer tries to navigate to the login page, the developer receives a 403 Forbidden error. The developer needs to implement a solution to allow unauthenticated access to the login page. The solution also must keep all private content secure. Which solution will meet these requirements?

- A. Add a second cache behavior to the distribution with the same origin as the default cache behavior
- B. Set the path pattern for the second cache behavior to the path of the login page, and make viewer access unrestricted
- C. Keep the default cache behavior's settings unchanged.
- D. Add a second cache behavior to the distribution with the same origin as the default cache behavior
- E. Set the path pattern for the second cache behavior to \*, and make viewer access restricted
- F. Change the default cache behavior's path pattern to the path of the login page, and make viewer access unrestricted.
- G. Add a second origin as a failover origin to the default cache behavior
- H. Point the failover origin to the S3 bucket
- I. Set the path pattern for the primary origin to \*, and make viewer access restricted
- J. Set the path pattern for the failover origin to the path of the login page, and make viewer access unrestricted.
- K. Add a bucket policy to the S3 bucket to allow read access
- L. Set the resource on the policy to the Amazon Resource Name (ARN) of the login page object in the S3 bucket
- M. Add a CloudFront function to the default cache behavior to redirect unauthorized requests to the login page's S3 URL.

**Answer:** A

**Explanation:**

The solution that will meet the requirements is to add a second cache behavior to the distribution with the same origin as the default cache behavior. Set the path pattern for the second cache behavior to the path of the login page, and make viewer access unrestricted. Keep the default cache behavior's settings unchanged. This way, the login page can be accessed without authentication, while all other content remains secure and requires signed cookies. The other options either do not allow unauthenticated access to the login page, or expose private content to unauthorized users.  
Reference: Restricting Access to Amazon S3 Content by Using an Origin Access Identity

**NEW QUESTION 9**

An online food company provides an Amazon API Gateway HTTP API to receive orders for partners. The API is integrated with an AWS Lambda function. The Lambda function stores the orders in an Amazon DynamoDB table. The company expects to onboard additional partners. Some partners require additional Lambda function to receive orders. The company has created an Amazon S3 bucket. The company needs to store all orders and updates in the S3 bucket for future analysis. How can the developer ensure that orders and updates are stored to Amazon S3 with the LEAST development effort?

- A. Create a new Lambda function and a new API Gateway API endpoint
- B. Configure the new Lambda function to write to the S3 bucket
- C. Modify the original Lambda function to post updates to the new API endpoint.
- D. Use Amazon Kinesis Data Streams to create a new data stream
- E. Modify the Lambda function to publish orders to the data stream. Configure the data stream to write to the S3 bucket.
- F. Enable DynamoDB Streams on the DynamoDB table
- G. Create a new Lambda function
- H. Associate the stream's Amazon Resource Name (ARN) with the Lambda Function. Configure the Lambda function to write to the S3 bucket as records appear in the table's stream.
- I. Modify the Lambda function to publish to a new Amazon SNS topic
- J. Simple Lambda function receives order
- K. Subscribe a new Lambda function to the topic
- L. Configure the new Lambda function to write to the S3 bucket as updates come through the topic.

**Answer:** C

**Explanation:**

This solution will ensure that all orders and updates are stored to Amazon S3 with the least development effort because it uses DynamoDB Streams to capture changes in the DynamoDB table and trigger a Lambda function to write those changes to the S3 bucket. This way, the original Lambda function and API Gateway API endpoint do not need to be modified, and no additional services are required. Option A is not optimal because it will require more development effort to create a new Lambda function and a new API Gateway API endpoint, and to modify the original Lambda function to post updates to the new API endpoint. Option B is not optimal because it will introduce additional costs and complexity to use Amazon Kinesis Data Streams to create a new data stream, and to modify the Lambda function to publish orders to the data stream. Option D is not optimal because it will require more development effort to modify the Lambda function to publish to a new Amazon SNS topic, and to create and subscribe a new Lambda function to the topic. References: Using DynamoDB Streams, Using AWS Lambda with Amazon S3

**NEW QUESTION 10**

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<sup>1</sup>. 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<sup>2</sup>. 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<sup>2</sup>.

? 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 10**

A company uses Amazon API Gateway to expose a set of APIs to customers. The APIs have caching enabled in API Gateway. Customers need a way to invalidate the cache for each API when they test the API.

What should a developer do to give customers the ability to invalidate the API cache?

- A. Ask the customers to use AWS credentials to call the InvalidateCache API operation.
- B. Attach an InvalidateCache policy to the IAM execution role that the customers use to invoke the API
- C. Ask the customers to send a request that contains the HTTP header when they make an API call.
- D. Ask the customers to use the AWS SDK API Gateway class to invoke the InvalidateCache API operation.
- E. Attach an InvalidateCache policy to the IAM execution role that the customers use to invoke the API
- F. Ask the customers to add the INVALIDATE\_CACHE query string parameter when they make an API call.

**Answer:** D

**NEW QUESTION 12**

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 minutes
- 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 machine
- E. Configure the state machine to invoke the Lambda function execution role at a specified interval by using a Wait state
- F. Set the interval to 15 minutes.
- G. Provision a small Amazon EC2 instance
- 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 EventBridge<sup>1</sup>.

**NEW QUESTION 13**

A developer is creating an application that will store personal health information (PHI). The PHI needs to be encrypted at all times. An encrypted Amazon RDS for MySQL DB instance is storing the data. The developer wants to increase the performance of the application by caching frequently accessed data while adding the ability to sort or rank the cached datasets.

Which solution will meet these requirements?

- A. Create an Amazon ElastiCache for Redis instance
- B. Enable encryption of data in transit and at rest
- C. Store frequently accessed data in the cache.
- D. Create an Amazon ElastiCache for Memcached instance
- E. Enable encryption of data in transit and at rest
- F. Store frequently accessed data in the cache.
- G. Create an Amazon RDS for MySQL read replica
- H. Connect to the read replica by using SSL
- I. Configure the read replica to store frequently accessed data.
- J. Create an Amazon DynamoDB table and a DynamoDB Accelerator (DAX) cluster for the table
- K. Store frequently accessed data in the DynamoDB table.

**Answer:** A

**Explanation:**

Amazon ElastiCache is a service that offers fully managed in-memory data stores that are compatible with Redis or Memcached. The developer can create an ElastiCache for Redis instance and enable encryption of data in transit and at rest. This will ensure that the PHI is encrypted at all times. The developer can store frequently accessed data in the cache and use Redis features such as sorting and ranking to enhance the performance of the application.

References:

? [What Is Amazon ElastiCache? - Amazon ElastiCache]

? [Encryption in Transit - Amazon ElastiCache for Redis]

? [Encryption at Rest - Amazon ElastiCache for Redis]

**NEW QUESTION 16**

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 sam local start-api subcommand allows you to run your serverless application locally for quick development and testing<sup>1</sup>. It creates a local HTTP server that acts as a proxy for API Gateway and invokes your Lambda functions based on the AWS SAM template<sup>1</sup>. You can use the sam local start-api subcommand to test your REST API locally by sending HTTP requests to the local endpoint<sup>1</sup>.

**NEW QUESTION 21**

A developer designed an application on an Amazon EC2 instance. The application makes API requests to objects in an Amazon S3 bucket. Which combination of steps will ensure that the application makes the API requests in the MOST secure manner? (Select TWO.)

- A. Create an IAM user that has permissions to the S3 bucket
- B. Add the user to an IAM group
- C. Create an IAM role that has permissions to the S3 bucket
- D. Add the IAM role to an instance profile
- E. Attach the instance profile to the EC2 instance.
- F. Create an IAM role that has permissions to the S3 bucket. Assign the role to an IAM group
- G. Store the credentials of the IAM user in the environment variables on the EC2 instance

**Answer:** BC

**Explanation:**

- Create an IAM role that has permissions to the S3 bucket. - Add the IAM role to an instance profile. Attach the instance profile to the EC2 instance. We first need to create an IAM Role with permissions to read and eventually write a specific S3 bucket. Then, we need to attach the role to the EC2 instance through an instance profile. In this

way, the ec2 instance has the permissions to read and eventually write the specified S3 bucket

**NEW QUESTION 25**

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 27**

A developer is creating an application that will give users the ability to store photos from their cellphones in the cloud. The application needs to support tens of thousands of users. The application uses an Amazon API Gateway REST API that is integrated with AWS Lambda functions to process the photos. The application stores details about the photos in Amazon DynamoDB.

Users need to create an account to access the application. In the application, users must be able to upload photos and retrieve previously uploaded photos. The photos will range in size from 300 KB to 5 MB.

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

- A. Use Amazon Cognito user pools to manage user account
- B. Create an Amazon Cognito user pool authorizer in API Gateway to control access to the AP
- C. Use the Lambda function to store the photos and details in the DynamoDB tabl
- D. Retrieve previously uploaded photos directly from the DynamoDB table.
- E. Use Amazon Cognito user pools to manage user account
- F. Create an Amazon Cognito user pool authorizer in API Gateway to control access to the AP
- G. Use the Lambda function to store the photos in Amazon S3. Store the object's S3 key as part of the photo details in the DynamoDB tabl
- H. Retrieve previously uploaded photos by querying DynamoDB for the S3 key.
- I. Create an IAM user for each user of the application during the sign-up proces
- J. Use IAM authentication to access the API Gateway AP
- K. Use the Lambda function to store the photos in Amazon S3. Store the object's S3 key as part of the photo details in the DynamoDB tabl
- L. Retrieve previously uploaded photos by querying DynamoDB for the S3 key.
- M. Create a users table in DynamoD
- N. Use the table to manage user account
- O. Create a Lambda authorizer that validates user credentials against the users tabl
- P. Integrate the Lambda authorizer with API Gateway to control access to the AP
- Q. Use the Lambda function to store the photos in Amazon S3. Store the object's S3 key as par of the photo details in the DynamoDB tabl
- R. Retrieve previously uploaded photos by querying DynamoDB for the S3 key.

DynamoDB

**Answer: B**

**Explanation:**

Amazon Cognito user pools is a service that provides a secure user directory that scales to hundreds of millions of users. The developer can use Amazon Cognito user pools to manage user accounts and create an Amazon Cognito user pool authorizer in API Gateway to control access to the API. The developer can use the Lambda function to store the photos in Amazon S3, which is a highly scalable, durable, and secure object storage service. The developer can store the object's S3 key as part of the photo details in the DynamoDB table, which is a fast and flexible NoSQL database service. The developer can retrieve previously uploaded photos by querying DynamoDB for the S3 key and fetching the photos from S3. This solution will meet the requirements with the least operational overhead.

References:

- ? [Amazon Cognito User Pools]
- ? [Use Amazon Cognito User Pools - Amazon API Gateway]
- ? [Amazon Simple Storage Service (S3)]
- ? [Amazon DynamoDB]

**NEW QUESTION 30**

A company has an application that runs as a series of AWS Lambda functions. Each Lambda function receives data from an Amazon Simple Notification Service (Amazon SNS) topic and writes the data to an Amazon Aurora DB instance.

To comply with an information security policy, the company must ensure that the Lambda functions all use a single securely encrypted database connection string to access Aurora.

Which solution will meet these requirements'?

- A. Use IAM database authentication for Aurora to enable secure database connections for ail the Lambda functions.
- B. Store the credentials and read the credentials from an encrypted Amazon RDS DB instance.
- C. Store the credentials in AWS Systems Manager Parameter Store as a secure string parameter.
- D. Use Lambda environment variables with a shared AWS Key Management Service (AWS KMS) key for encryption.

**Answer: A**

**Explanation:**

This solution will meet the requirements by using IAM database authentication for Aurora, which enables using IAM roles or users to authenticate with Aurora databases instead of using passwords or other secrets. The developer can use IAM database authentication for Aurora to enable secure database connections for all the Lambda functions that access Aurora DB instance. The developer can create an IAM role with permission to connect to Aurora DB instance and attach it to each Lambda function. The developer can also configure Aurora DB instance to use IAM database authentication and enable encryption in transit using SSL certificates. This way, the Lambda functions can use a single securely encrypted database connection string to access Aurora without needing any secrets or passwords. Option B is not optimal because it will store the credentials and read them from an encrypted Amazon RDS DB instance, which may introduce additional costs and complexity for managing and accessing another RDS DB instance. Option C is not optimal because it will store the credentials in AWS Systems Manager Parameter Store as a secure string parameter, which may require additional steps or permissions to retrieve and decrypt the credentials from Parameter Store. Option D is not optimal because it will use Lambda environment variables with a shared AWS Key Management Service (AWS KMS) key for encryption, which may not be secure or scalable as environment variables are stored as plain text unless encrypted with AWS KMS. References: [IAM Database Authentication for MySQL and PostgreSQL], [Using SSL/TLS to Encrypt a Connection to a DB Instance]

**NEW QUESTION 34**

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 mad
- 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 mad
- 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 consol
- F. Select the removePii functio
- G. Use S3 Access Points to access the object without PII.
- H. Create an S3 access point from the S3 consol



- 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 39**

A developer is building a serverless application by using AWS Serverless Application Model (AWS SAM) on multiple AWS Lambda functions. When the application is deployed, the developer wants to shift 10% of the traffic to the new deployment of the application for the first 10 minutes after deployment. If there are no issues, all traffic must switch over to the new version.

Which change to the AWS SAM template will meet these requirements?

- A. Set the Deployment Preference Type to Canary and Percent10Minute.
- B. Set the AutoPublishAlias property to the Lambda alias.
- C. Set the Deployment Preference Type to Linear and PercentEvery10Minute.
- D. Set AutoPublishAlias property to the Lambda alias.
- E. Set the Deployment Preference Type to Canary and Percent10Minute.
- F. Set the PreTraffic and PostTraffic properties to the Lambda alias.
- G. Set the Deployment Preference Type to Linear and PercentEvery10Minute.
- H. Set PreTraffic and PostTraffic properties to the Lambda alias.

**Answer:** A

**Explanation:**

? The Deployment Preference Type property specifies how traffic should be shifted between versions of a Lambda function. The Canary10Percent10Minutes option means that 10% of the traffic is immediately shifted to the new version, and after 10 minutes, the remaining 90% of the traffic is shifted. This matches the requirement of shifting 10% of the traffic for the first 10 minutes, and then switching all traffic to the new version.

? The AutoPublishAlias property enables AWS SAM to automatically create and update a Lambda alias that points to the latest version of the function. This is required to use the Deployment Preference Type property. The alias name can be specified by the developer, and it can be used to invoke the function with the latest code.

**NEW QUESTION 42**

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 44**

A company has an analytics application that uses an AWS Lambda function to process transaction data asynchronously. A developer notices that asynchronous invocations of the Lambda function sometimes fail. When failed Lambda function invocations occur, the developer wants to invoke a second Lambda function to handle errors and log details.

Which solution will meet these requirements?

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

Configuring a Lambda function destination with a failure condition is the best solution for invoking a second Lambda function to handle errors and log details. A Lambda function destination is a resource that Lambda sends events to after a function is invoked. The developer can specify the destination type as Lambda function and the ARN of the error-handling Lambda function as the resource. The developer can also specify the failure condition, which means that the destination is invoked only when the initial Lambda function fails. The destination event will include the response from the initial function, the request ID, and the timestamp. The other solutions are either not feasible or not efficient. Enabling AWS X-Ray active tracing on the initial Lambda function will help to monitor and troubleshoot the function performance, but it will not automatically invoke the error-handling Lambda function. Configuring a Lambda function trigger with a failure condition is not a valid option, as triggers are used to invoke Lambda functions, not to send events from Lambda functions. Creating a status check alarm on the initial Lambda function will incur additional costs and complexity, and it will not capture the details of the failed invocations. References

? Using AWS Lambda destinations

? Asynchronous invocation - AWS Lambda

? AWS Lambda Destinations: What They Are and Why to Use Them



? AWS Lambda Destinations: A Complete Guide | Dashbird

#### NEW QUESTION 47

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 49

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 52

A developer is using AWS Amplify Hosting to build and deploy an application. The developer is receiving an increased number of bug reports from users. The developer wants to add end-to-end testing to the application to eliminate as many bugs as possible before the bugs reach production.

Which solution should the developer implement to meet these requirements?

- A. Run the amplify add test command in the Amplify CLI.
- B. Create unit tests in the applicatio
- C. Deploy the unit tests by using the amplify push command in the Amplify CLI.
- D. Add a test phase to the amplify.yml build settings for the application.
- E. Add a test phase to the aws-exports.js file for the application.

**Answer:** C

#### Explanation:

The solution that will meet the requirements is to add a test phase to the amplify.yml build settings for the application. This way, the developer can run end-to-end tests on every code commit and catch any bugs before deploying to production. The other options either do not support end-to-end testing, or do not run tests automatically.

Reference: End-to-end testing

#### NEW QUESTION 57

A team of developed is using an AWS CodePipeline pipeline as a continuous integration and continuous delivery (CI/CD) mechanism for a web application. A developer has written unit tests to programmatically test the functionality of the application code. The unit tests produce a test report that shows the results of each individual check. The developer now

wants to run these tests automatically during the CI/CD process.

- A. Write a Git pre-commit hook that runs the test before every commi
- B. Ensure that each developer who is working on the project has the pre-commit hook instated locall
- C. Review the test report and resolve any issues before pushing changes to AWS CodeCommit.
- D. Add a new stage to the pipelin
- E. Use AWS CodeBuild as the provide
- F. Add the new stage after the stage that deploys code revisions to the test environmen
- G. Write a buildspec that fails the CodeBuild stage if any test does not pas
- H. Use the test reports feature of Codebuild to integrate the report with the CodoBuild consol
- I. View the test results in CodeBuild Resolve any issues.
- J. Add a new stage to the pipelin
- K. Use AWS CodeBuild at the provide
- L. Add the new stage before the stage that deploys code revisions to the test environmen
- M. Write a buildspec that fails the CodeBuild stage it any test does not pas
- N. Use the test reports feature of CodeBuild to integrate the report with the CodeBuild consol
- O. View the test results in codeBuild Resolve any issues.
- P. Add a new stage to the pipelin
- Q. Use Jenkins as the provide
- R. Configure CodePipeline to use Jenkins to run the unit test
- S. Write a Jenkinsfile that fails the stage if any test does not pas
- T. Use the test report plugin for Jenkins to integrate the repot with the Jenkins dashboar
- . View the test results in Jenkin
- . Resolve any issues.

**Answer:** C

**Explanation:**

The solution that will meet the requirements is to add a new stage to the pipeline. Use AWS CodeBuild as the provider. Add the new stage before the stage that deploys code revisions to the test environment. Write a buildspec that fails the CodeBuild stage if any test does not pass. Use the test reports feature of CodeBuild to integrate the report with the CodeBuild console. View the test results in CodeBuild. Resolve any issues. This way, the developer can run the unit tests automatically during the CI/CD process and catch any bugs before deploying to the test environment. The developer can also use the test reports feature of CodeBuild to view and analyze the test results in a graphical interface. The other options either involve running the tests manually, running them after deployment, or using a different provider that requires additional configuration and integration.

Reference: Test reports for CodeBuild

**NEW QUESTION 59**

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 ate experiencing high latency flue lo sialic content on theEC2 instance. even during non-peak hours.  
When companion of steps mill resolves 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 sialic 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 64**

A developer warns to add request validation to a production environment Amazon API Gateway API. The developer needs to test the changes before the API is deployed to the production environment. For the lest the developer will send test requests to the API through a testing tool.  
Which solution will meet these requirements with the LEAST operational overhead?

- A. Export the existing API to an OpenAPI fil
- B. Create a new API Import the OpenAPI file Modify the new API to add request validatio
- C. Perform the tests Modify the existing API to add request validatio
- D. Deploy the existing API to production.
- E. Modify the existing API to add request validatio
- F. Deploy the updated API to a new API Gateway stage Perform the tests Deploy the updated API to the API Gateway production stage.
- G. Create a new API Add the necessary resources and methods including new request validatio
- H. Perform the tests Modify the existing API to add request validatio
- I. Deploy the existing API to production.
- J. Clone the exiting API Modify the new API lo add request validatio  
Modify the existing API to add request validation Deploy the existing API to production.
- K. Perform the tests

**Answer:** D

**Explanation:**

This solution allows the developer to test the changes without affecting the production environment. Cloning an API creates a copy of the API definition that can be modified independently. The developer can then add request validation to the new API and test it using a testing tool. After verifying that the changes work as expected, the developer can apply the same changes to the existing API and deploy it to production.

Reference: Clone an API, [Enable Request Validation for an API in API Gateway]

#### NEW QUESTION 68

A company is expanding the compatibility of its photo-sharing mobile app to hundreds of additional devices with unique screen dimensions and resolutions. Photos are stored in Amazon S3 in their original format and resolution. The company uses an Amazon CloudFront distribution to serve the photos. The app includes the dimension and resolution of the display as GET parameters with every request.

A developer needs to implement a solution that optimizes the photos that are served to each device to reduce load time and increase photo quality. Which solution will meet these requirements MOST cost-effectively?

- A. Use S3 Batch Operations to invoke an AWS Lambda function to create new variants of the photos with the required dimensions and resolution.
- B. Create a dynamic CloudFront origin that automatically maps the request of each device to the corresponding photo variant.
- C. Use S3 Batch Operations to invoke an AWS Lambda function to create new variants of the photos with the required dimensions and resolution.
- D. Create a Lambda@Edge function to route requests to the corresponding photo variant by using request headers.
- E. Create a Lambda@Edge function that optimizes the photos upon request and returns the photos as a response.
- F. Change the CloudFront TTL cache policy to the maximum value possible.

Create a Lambda@Edge function that optimizes the photos upon request and returns the photos as a response.

G. In the same function store a copy of the processed photos on Amazon S3 for subsequent requests.

**Answer: D**

#### Explanation:

This solution meets the requirements most cost-effectively because it optimizes the photos on demand and caches them for future requests. Lambda@Edge allows the developer to run Lambda functions at AWS locations closer to viewers, which can reduce latency and improve photo quality. The developer can create a Lambda@Edge function that uses the GET parameters from each request to optimize the photos with the required dimensions and resolutions and returns them as a response. The function can also store a copy of the processed photos on Amazon S3 for subsequent requests, which can reduce processing time and costs. Using S3 Batch Operations to create new variants of the photos will incur additional storage costs and may not cover all possible dimensions and resolutions. Creating a dynamic CloudFront origin or a Lambda@Edge function to route requests to corresponding photo variants will require maintaining a mapping of device types and photo variants, which can be complex and error-prone.

Reference: [Lambda@Edge Overview], [Resizing Images with Amazon CloudFront & Lambda@Edge]

#### NEW QUESTION 73

A developer is planning to migrate on-premises company data to Amazon S3. The data must be encrypted, and the encryption keys must support automated annual rotation. The company must use AWS Key Management Service (AWS KMS) to encrypt the data. When type of keys should the developer use to meet these requirements?

- A. Amazon S3 managed keys
- B. Symmetric customer managed keys with key material that is generated by AWS
- C. Asymmetric customer managed keys with key material that is generated by AWS
- D. Symmetric customer managed keys with imported key material

**Answer: B**

#### Explanation:

The type of keys that the developer should use to meet the requirements is symmetric customer managed keys with key material that is generated by AWS. This way, the developer can use AWS Key Management Service (AWS KMS) to encrypt the data with a symmetric key that is managed by the developer. The developer can also enable automatic annual rotation for the key, which creates new key material for the key every year. The other options either involve using Amazon S3 managed keys, which do not support automatic annual rotation, or using asymmetric keys or imported key material, which are not supported by S3 encryption.

Reference: Using AWS KMS keys to encrypt S3 objects

#### NEW QUESTION 75

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 libraries.
- 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 78

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 79

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 83

A company has an Amazon S3 bucket that contains sensitive data. The data must be encrypted in transit and at rest. The company

encrypts the data in the S3 bucket by using an AWS Key Management Service (AWS KMS) key. A developer needs to grant several other AWS accounts the permission to use the S3 GetObject operation to retrieve the data from the S3 bucket. How can the developer enforce that all requests to retrieve the data provide encryption in transit?

- A. Define a resource-based policy on the S3 bucket to deny access when a request meets the condition “aws:SecureTransport”: “false”.
- B. Define a resource-based policy on the S3 bucket to allow access when a request meets the condition “aws:SecureTransport”: “false”.
- C. Define a role-based policy on the other accounts' roles to deny access when a request meets the condition of “aws:SecureTransport”: “false”.
- D. Define a resource-based policy on the KMS key to deny access when a request meets the condition of “aws:SecureTransport”: “false”.

**Answer:** A

**Explanation:**

Amazon S3 supports resource-based policies, which are JSON documents that specify the permissions for accessing S3 resources. A resource-based policy can be used to enforce encryption in transit by denying access to requests that do not use HTTPS. The condition key aws:SecureTransport can be used to check if the request was sent using SSL. If the value of this key is false, the request is denied; otherwise, the request is allowed. Reference: How do I use an S3 bucket policy to require requests to use Secure Socket Layer (SSL)?

**NEW QUESTION 84**

A company is using Amazon API Gateway to invoke a new AWS Lambda function. The company has Lambda function versions in its PROD and DEV environments. In each environment, there is a Lambda function alias pointing to the corresponding Lambda function version. API Gateway has one stage that is configured to point at the PROD alias.

The company wants to configure API Gateway to enable the PROD and DEV Lambda function versions to be simultaneously and distinctly available. Which solution will meet these requirements?

- A. Enable a Lambda authorizer for the Lambda function alias in API Gateway. Republish PROD and create a new stage for DEV. Create API Gateway stage variables for the PROD and DEV stage.
- B. Point each stage variable to the PROD Lambda authorizer to the DEV Lambda authorizer.
- C. Set up a gateway response in API Gateway for the Lambda function alias.
- D. Republish PROD and create a new stage for DEV.
- E. Create gateway responses in API Gateway for PROD and DEV Lambda aliases.
- F. Use an environment variable for the Lambda function alias in API Gateway.
- G. Republish PROD and create a new stage for development.
- H. Create API gateway environment variables for PROD and DEV stage.
- I. Point each stage variable to the PROD Lambda function alias to the DEV Lambda function alias.
- J. Use an API Gateway stage variable to configure the Lambda function alias. Republish PROD and create a new stage for development. Create API Gateway stage variables for PROD and DEV stages. Point each stage variable to the PROD Lambda function alias and to the DEV Lambda function alias.

**Answer:** D

**Explanation:**

The best solution is to use an API Gateway stage variable to configure the Lambda function alias. This allows you to specify the Lambda function name and its alias or version using the syntax `function_name:$ {stageVariables.variable_name}` in the Integration Request. You can then create different stages in API Gateway, such as PROD and DEV, and assign different values to the stage variable for each stage. This way, you can invoke different Lambda function versions or aliases based on the stage that you are using, without changing the function name in the Integration Request. References

- ? Using API Gateway stage variables to manage Lambda functions
- ? How to point AWS API gateway stage to specific lambda function alias?
- ? Setting stage variables using the Amazon API Gateway console
- ? Amazon API Gateway stage variables reference

**NEW QUESTION 89**

A developer is writing a serverless application that requires an AWS Lambda function to be invoked every 10 minutes. What is an automated and serverless way to invoke the function?

- A. Deploy an Amazon EC2 instance based on Linux, and edit its `/etc/crontab` file by adding a command to periodically invoke the lambda function.
- B. Configure an environment variable named `PERIOD` for the Lambda function.
- C. Set the value to 600.
- D. Create an Amazon EventBridge rule that runs on a regular schedule to invoke the Lambda function.
- E. Create an Amazon Simple Notification Service (Amazon SNS) topic that has a subscription to the Lambda function with a 600-second timer.

**Answer:** C

**Explanation:**

The solution that will meet the requirements is to create an Amazon EventBridge rule that runs on a regular schedule to invoke the Lambda function. This way, the developer can use an automated and serverless way to invoke the function every 10 minutes. The developer can also use a cron expression or a rate expression to specify the schedule for the rule. The other options either involve using an Amazon EC2 instance, which is not serverless, or using environment variables or query parameters, which do not trigger the function.

Reference: Schedule AWS Lambda functions using EventBridge

#### NEW QUESTION 92

A company needs to distribute firmware updates to its customers around the world. Which service will allow easy and secure control of the access to the downloads at the lowest cost?

- A. Use Amazon CloudFront with signed URLs for Amazon S3.
- B. Create a dedicated Amazon CloudFront Distribution for each customer.
- C. Use Amazon CloudFront with AWS Lambda@Edge.
- D. Use Amazon API Gateway and AWS Lambda to control access to an S3 bucket.

**Answer:** A

#### Explanation:

This solution allows easy and secure control of access to the downloads at the lowest cost because it uses a content delivery network (CDN) that can cache and distribute firmware updates to customers around the world, and uses a mechanism that can restrict access to specific files or versions. Amazon CloudFront is a CDN that can improve performance, availability, and security of web applications by delivering content from edge locations closer to customers. Amazon S3 is a storage service that can store firmware updates in buckets and objects. Signed URLs are URLs that include additional information, such as an expiration date and time, that give users temporary access to specific objects in S3 buckets. The developer can use CloudFront to serve firmware updates from S3 buckets and use signed URLs to control who can download them and for how long. Creating a dedicated CloudFront distribution for each customer will incur unnecessary costs and complexity. Using Amazon CloudFront with AWS Lambda@Edge will require additional programming overhead to implement custom logic at the edge locations. Using Amazon API Gateway and AWS Lambda to control access to an S3 bucket will also require additional programming overhead and may not provide optimal performance or availability.

Reference: [Serving Private Content through CloudFront], [Using CloudFront with Amazon S3]

#### NEW QUESTION 97

A company wants to automate part of its deployment process. A developer needs to automate the process of checking for and deleting unused resources that supported previously deployed stacks but that are no longer used. The company has a central application that uses the AWS Cloud Development Kit (AWS CDK) to manage all deployment stacks. The stacks are spread out across multiple accounts. The developer's solution must integrate as seamlessly as possible within the current deployment process. Which solution will meet these requirements with the LEAST amount of configuration?

- A. In the central AWS CDK application, write a handler function in the code that uses AWS SDK calls to check for and delete unused resource
- B. Create an AWS CloudFormation template from a JSON file
- C. Use the template to attach the function code to an AWS Lambda function and to invoke the Lambda function when the deployment stack runs.
- D. In the central AWS CDK application
- E. write a handler function in the code that uses AWS SDK calls to check for and delete unused resource
- F. Create an AWS CDK custom resource Use the custom resource to attach the function code to an AWS Lambda function and to invoke the Lambda function when the deployment stack runs.
- G. In the central AWS CDK, write a handler function in the code that uses AWS SDK calls to check for and delete unused resource
- H. Create an API in AWS Amplify Use the API to attach the function code to an AWS Lambda function and to invoke the Lambda function when the deployment stack runs.
- I. In the AWS Lambda console write a handler function in the code that uses AWS SDK calls to check for and delete unused resource
- J. Create an AWS CDK custom resource
- K. Use the custom resource to import the Lambda function into the stack and to invoke the Lambda function when the deployment stack runs.

**Answer:** B

#### Explanation:

This solution meets the requirements with the least amount of configuration because it uses a feature of AWS CDK that allows custom logic to be executed during stack deployment or deletion. The AWS Cloud Development Kit (AWS CDK) is a software development framework that allows you to define cloud infrastructure as code and provision it through CloudFormation. An AWS CDK custom resource is a construct that enables you to create resources that are not natively supported by CloudFormation or perform tasks that are not supported by CloudFormation during stack deployment or deletion. The developer can write a handler function in the code that uses AWS SDK calls to check for and delete unused resources, and create an AWS CDK custom resource that attaches the function code to a

Lambda function and invokes it when the deployment stack runs. This way, the developer can automate the cleanup process without requiring additional configuration or integration. Creating a CloudFormation template from a JSON file will require additional configuration and integration with the central AWS CDK application. Creating an API in AWS Amplify will require additional configuration and integration with the central AWS CDK application and may not provide optimal performance or availability. Writing a handler function in the AWS Lambda console will require additional configuration and integration with the central AWS CDK application.

Reference: [AWS Cloud Development Kit (CDK)], [Custom Resources]

#### NEW QUESTION 99

An AWS Lambda function requires read access to an Amazon S3 bucket and requires read/write access to an Amazon DynamoDB table. The correct IAM policy already exists.

What is the MOST secure way to grant the Lambda function access to the S3 bucket and the DynamoDB table?

- A. Attach the existing IAM policy to the Lambda function.
- B. Create an IAM role for the Lambda function. Attach the existing IAM policy to the role. Attach the role to the Lambda function.
- C. Create an IAM user with programmatic access. Attach the existing IAM policy to the user.
- D. Add the user access key ID and secret access key as environment variables in the Lambda function.
- E. Add the AWS account root user access key ID and secret access key as encrypted environment variables in the Lambda function.

**Answer: B**

#### Explanation:

The most secure way to grant the Lambda function access to the S3 bucket and the DynamoDB table is to create an IAM role for the Lambda function and attach the existing IAM policy to the role. This way, you can use the principle of least privilege and avoid exposing any credentials in your function code or environment variables. You can also leverage the temporary security credentials that AWS provides to the Lambda function when it assumes the role. This solution follows the best practices for working with AWS Lambda functions<sup>1</sup> and designing and architecting with DynamoDB<sup>2</sup>. References

? Best practices for working with AWS Lambda functions

? Best practices for designing and architecting with DynamoDB

#### NEW QUESTION 102

A company hosts a client-side web application for one of its subsidiaries on Amazon S3. The web application can be accessed through Amazon CloudFront from <https://www.example.com>. After a successful rollout, the company wants to host three more client-side web applications for its remaining subsidiaries on three separate S3 buckets.

To achieve this goal, a developer moves all the common JavaScript files and web fonts to a central S3 bucket that serves the web applications. However, during testing, the developer notices that the browser blocks the JavaScript files and web fonts.

What should the developer do to prevent the browser from blocking the JavaScript files and web fonts?

- A. Create four access points that allow access to the central S3 bucket.
- B. Assign an access point to each web application bucket.
- C. Create a bucket policy that allows access to the central S3 bucket.
- D. Attach the bucket policy to the central S3 bucket.
- E. Create a cross-origin resource sharing (CORS) configuration that allows access to the central S3 bucket.
- F. Add the CORS configuration to the central S3 bucket.
- G. Create a Content-MD5 header that provides a message integrity check for the central S3 bucket.
- H. Insert the Content-MD5 header for each web application request.

**Answer: C**

#### Explanation:

This is a frequent trouble. Web applications cannot access the resources in other domains by default, except some exceptions. You must configure CORS on the resources to be accessed. <https://docs.aws.amazon.com/AmazonS3/latest/userguide/cors.html>

#### NEW QUESTION 104

A developer must analyze performance issues with production-distributed applications written as AWS Lambda functions. These distributed Lambda applications invoke other components that make up the applications. How should the developer identify and troubleshoot the root cause of the performance issues in production?

- A. Add logging statements to the Lambda function.
- B. then use Amazon CloudWatch to view the logs.
- C. Use AWS CloudTrail and then examine the logs.
- D. Use AWS X-Ray.
- E. then examine the segments and errors.
- F. Run Amazon Inspector agents and then analyze performance.

**Answer: C**

#### Explanation:

This solution will meet the requirements by using AWS X-Ray to analyze and debug the performance issues with the distributed Lambda applications. AWS X-Ray is a service that collects data about requests that the applications serve, and provides tools to view, filter, and gain insights into that data. The developer can use AWS X-Ray to identify the root cause of the performance issues by examining the segments and errors that show the details of each request and the components that make up the applications. Option A is not optimal because it will use logging statements and Amazon CloudWatch, which may not provide enough information or visibility into the distributed applications. Option B is not

optimal because it will use AWS CloudTrail, which is a service that records API calls and events for AWS services, not application performance data. Option D is not optimal because it will use Amazon Inspector, which is a service that helps improve the security and compliance of applications on Amazon EC2 instances, not Lambda functions. References: AWS X-Ray, Using AWS X-Ray with AWS Lambda

#### NEW QUESTION 109

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



What is the MOST likely cause of this issue?

- A. The Lambda function's concurrency limit has been exceeded.
- B. DynamoDB table requires a global secondary index (GSI) to support writes.
- C. The Lambda function does not have IAM permissions to write to DynamoDB.
- D. The DynamoDB table is not running in the same Availability Zone as the Lambda function.

**Answer: C**

**Explanation:**

[https://docs.aws.amazon.com/IAM/latest/UserGuide/reference\\_policies\\_examples\\_lambda-access-dynamodb.html](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_examples_lambda-access-dynamodb.html)

#### NEW QUESTION 110

A company built a new application in the AWS Cloud. The company automated the bootstrapping of new resources with an Auto Scaling group by using AWS CloudFormation templates. The bootstrap scripts contain sensitive data.

The company needs a solution that is integrated with CloudFormation to manage the sensitive data in the bootstrap scripts.

Which solution will meet these requirements in the MOST secure way?

- A. Put the sensitive data into a CloudFormation parameter
- B. Encrypt the CloudFormation templates by using an AWS Key Management Service (AWS KMS) key.
- C. Put the sensitive data into an Amazon S3 bucket Update the CloudFormation templates to download the object from Amazon S3 during bootstrap.
- D. Put the sensitive data into AWS Systems Manager Parameter Store as a secure string parameter
- E. Update the CloudFormation templates to use dynamic references to specify template values.
- F. Put the sensitive data into Amazon Elastic File System (Amazon EFS) Enforce EFS encryption after file system creation
- G. Update the CloudFormation templates to retrieve data from Amazon EFS.

**Answer: C**

**Explanation:**

This solution meets the requirements in the most secure way because it uses a service that is integrated with CloudFormation to manage sensitive data in encrypted form. AWS Systems Manager Parameter Store provides secure, hierarchical storage for configuration data management and secrets management. You can store sensitive data as secure string parameters, which are encrypted using an AWS Key Management Service (AWS KMS) key of your choice. You can also use dynamic references in your CloudFormation templates to specify template values that are stored in Parameter Store or Secrets Manager without having to include them in your templates. Dynamic references are resolved only during stack creation or update operations, which reduces exposure risks for sensitive data. Putting sensitive data into a CloudFormation parameter will not encrypt them or protect them from unauthorized access. Putting sensitive data into an Amazon S3 bucket or Amazon Elastic File System (Amazon EFS) will require additional configuration and integration with CloudFormation and may not provide fine-grained access control or encryption for sensitive data.

Reference: [What Is AWS Systems Manager Parameter Store?], [Using Dynamic

References to Specify Template Values]

#### NEW QUESTION 113

A developer is creating an AWS Lambda function that searches for items from an Amazon DynamoDB table that contains customer contact information- The DynamoDB table items have the customer's email\_address as the partition key and additional properties such as customer\_type, name, and job\_title.

The Lambda function runs whenever a user types a new character into the customer\_type text input The developer wants the search to return partial matches of all the email\_address property of a particular customer\_type The developer does not want to recreate the DynamoDB table.

What should the developer do to meet these requirements?

- A. Add a global secondary index (GSI) to the DynamoDB table with customer\_type as the partition key and email\_address as the sort key Perform a query operation on the GSI by using the begins\_with key condition expression With the email\_address property
- B. Add a global secondary index (GSI) to the DynamoDB table With email\_address as the partition key and customer\_type as the sort key Perform a query operation on the GSI by using the begins\_with key condition expression With the email\_address property.
- C. Add a local secondary index (LSI) to the DynamoDB table With customer\_type as the partition key and email\_address as the sort key Perform a query operation on the LSI by using the begins\_with key condition expression With the email\_address property
- D. Add a local secondary Index (LSI) to the DynamoDB table With job\_title as the partition key and email\_address as the sort key Perform a query operation on the LSI by using the begins\_with key condition expression With the email\_address property

**Answer: A**

**Explanation:**

By adding a global secondary index (GSI) to the DynamoDB table with customer\_type as the partition key and email\_address as the sort key, the developer can perform a query operation on the GSI using the Begins\_with key condition expression with the email\_address property. This will return partial matches of all email\_address properties of a specific customer\_type.

#### NEW QUESTION 117

A developer is writing an AWS Lambda function. The developer wants to log key events that occur while the Lambda function runs. The developer wants to include a unique identifier to associate the events with a specific function invocation. The developer adds the following code to the Lambda function:

```
function handler(event, context) {  
  
}
```

Which solution will meet this requirement?

- A. Obtain the request identifier from the AWS request ID field in the context object
- B. Configure the application to write logs to standard output.

- C. Obtain the request identifier from the AWS request ID field in the event object
- D. Configure the application to write logs to a file.
- E. Obtain the request identifier from the AWS request ID field in the event object
- F. Configure the application to write logs to standard output.
- G. Obtain the request identifier from the AWS request ID field in the context object
- H. Configure the application to write logs to a file.

**Answer:** A

**Explanation:**

<https://docs.aws.amazon.com/lambda/latest/dg/nodejs-context.html> <https://docs.aws.amazon.com/lambda/latest/dg/nodejs-logging.html>

There is no explicit information for the runtime, the code is written in Node.js.

AWS Lambda is a service that lets developers run code without provisioning or managing servers. The developer can use the AWS request ID field in the context object to obtain a unique identifier for each function invocation. The developer can configure the application to write logs to standard output, which will be captured by Amazon CloudWatch Logs. This solution will meet the requirement of logging key events with a unique identifier.

References:

? [What Is AWS Lambda? - AWS Lambda]

? [AWS Lambda Function Handler in Node.js - AWS Lambda]

? [Using Amazon CloudWatch - AWS Lambda]

**NEW QUESTION 119**

.....

## Thank You for Trying Our Product

### We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questons and Answers in PDF Format

### 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

**100% Actual & Verified — Instant Download, Please Click**  
**[Order The DVA-C02 Practice Test Here](#)**