

Amazon

Exam Questions AWS-Certified-Developer-Associate

Amazon AWS Certified Developer - Associate



NEW QUESTION 1

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 of the 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 all 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 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 2

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.
- D. Update the application code to read and write configuration files from the shared folder.
- E. Deploy a multi-EC2 instance with an instance store volume.
- F. Use the host operating system to share a folder.
- 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 3

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.
- D. 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.
- 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 4

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 5

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 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 6

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 once¹. 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 it². 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 it².

? 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 7

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 for the other developers on the team
- F. Check the response Document how to run the unit testing framework.
- G. Update the CI/CD 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 the 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 documentation
- L. Create a Docker container from the Node.js base image to invoke the Lambda function
- M. Check the response Document how to run the Docker container for the other 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 8

A developer is configuring an applications deployment environment in AWS CodePipeline. 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 CodeCommit project
- B. Add the repository package's build and test commands to the project's buildspec
- C. Create an AWS CodeBuild project
- D. Add the repository package's build and test commands to the project's buildspec
- E. Create an AWS CodeDeploy project
- F. Add the repository package's build and test commands to the project's buildspec
- G. Add an action to the source stage
- H. Specify the newly created project as the action provider
- I. Specify the build artifact as the action's input artifact.
- J. Add a new stage to the pipeline after the source stage
- K. Add an action to the new stage
- L. Specify the newly created project as the action provider
- 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 9

A company runs a payment application on Amazon EC2 instances behind an Application Load Balance. The EC2 instances run in an Auto Scaling group across multiple Availability Zones. The application needs to retrieve application secrets during the application startup and export the secrets as environment variables. These secrets must be encrypted at rest and need to be rotated every month. Which solution will meet these requirements with the LEAST development effort?

- A. Save the secrets in a text file and store the text file in Amazon S3. Provision a customer managed key. Use the key for secret encryption in Amazon S3. Read the contents of the text file and read the export as environment variables. Configure S3 Object Lambda to rotate the text file every month.
- B. Save the secrets as strings in AWS Systems Manager Parameter Store and use the default AWS Key Management Service (AWS KMS) key. Configure an Amazon EC2 user data script to retrieve the secrets during the startup and export as environment variables. Configure an AWS Lambda function to rotate the secrets in Parameter Store every month.
- C. Save the secrets as base64 encoded environment variables in the application properties.
- D. Retrieve the secrets during the application startup.
- E. Reference the secrets in the application code.
- F. Write a script to rotate the secrets saved as environment variables.
- G. Store the secrets in AWS Secrets Manager. Provision a new customer master key. Use the key to encrypt the secrets. Enable automatic rotation. Configure an Amazon EC2 user data script to programmatically retrieve the secrets during the startup and export as environment variables.

Answer: D

Explanation:

AWS Secrets Manager is a service that enables the secure management and rotation of secrets, such as database credentials, API keys, or passwords. By using Secrets Manager, the company can avoid hardcoding secrets in the application code or properties files, and instead retrieve them programmatically during the application startup. Secrets Manager also supports automatic rotation of secrets by using AWS Lambda functions or built-in rotation templates. The company can provision a customer master key (CMK) to encrypt the secrets and use the AWS SDK or CLI to export the secrets as environment variables. References:

- ? What Is AWS Secrets Manager? - AWS Secrets Manager
- ? Rotating Your AWS Secrets Manager Secrets - AWS Secrets Manager
- ? Retrieving a Secret - AWS Secrets Manager

NEW QUESTION 10

An application that is hosted on an Amazon EC2 instance needs access to files that are stored in an Amazon S3 bucket. The application lists the objects that are stored in the S3 bucket and displays a table to the user. During testing, a developer discovers that the application does not show any objects in the list. What is the MOST secure way to resolve this issue?

- A. Update the IAM instance profile that is attached to the EC2 instance to include the S3:* permission for the S3 bucket.
- B. Update the IAM instance profile that is attached to the EC2 instance to include the S3:ListBucket permission for the S3 bucket.
- C. Update the developer's user permissions to include the S3:ListBucket permission for the S3 bucket.
- D. Update the S3 bucket policy by including the S3:ListBucket permission and by setting the Principal element to specify the account number of the EC2 instance.

Answer: B

Explanation:

IAM instance profiles are containers for IAM roles that can be associated with EC2 instances. An IAM role is a set of permissions that grant access to AWS resources. An IAM role can be used to allow an EC2 instance to access an S3 bucket by including the appropriate permissions in the role's policy. The S3:ListBucket permission allows listing the objects in an S3 bucket. By updating the IAM instance profile with this permission, the application on the EC2 instance can retrieve the objects from the S3 bucket and display them to the user. Reference: Using an IAM role to grant permissions to applications running on Amazon EC2 instances

NEW QUESTION 10

A developer is working on a Python application that runs on Amazon EC2 instances. The developer wants to enable tracing of application requests to debug performance issues in the code. Which combination of actions should the developer take to achieve this goal? (Select TWO)

- A. Install the Amazon CloudWatch agent on the EC2 instances.
- B. Install the AWS X-Ray daemon on the EC2 instances.
- C. Configure the application to write JSON-formatted logs to /var/log/cloudwatch.
- D. Configure the application to write trace data to /var/log/xray.
- E. Install and configure the AWS X-Ray SDK for Python in the application.

Answer: BE

Explanation:

This solution will meet the requirements by using AWS X-Ray to enable tracing of application requests to debug performance issues in the code. 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 install the AWS X-Ray daemon on the EC2 instances, which is a software that listens for traffic on UDP port 2000, gathers raw segment data, and relays it to the X-Ray API. The developer can also install and configure the AWS X-Ray SDK for Python in the application, which is a library that enables instrumenting Python code to generate and send trace data to the X-Ray daemon. Option A is not optimal because it will install the Amazon CloudWatch agent on the EC2 instances, which is a software that collects metrics and logs from EC2 instances and on-premises servers, not application performance data. Option C is not optimal because it will configure the application to write JSON-formatted logs to /var/log/cloudwatch, which is not a valid path or destination for CloudWatch logs. Option D is not optimal because it will configure the application to write trace data to /var/log/xray, which is also not a valid path or destination for X-Ray trace data.

References: [AWS X-Ray], [Running the X-Ray Daemon on Amazon EC2]

NEW QUESTION 15

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 17

A developer is using AWS Step Functions to automate a workflow. The workflow defines each step as an AWS Lambda function task. The developer notices that runs of the Step Functions state machine fail in the GetResource task with either an UlegalArgumentException error or a TooManyRequestsException error. The developer wants the state machine to stop running when the state machine encounters an UlegalArgumentException error. The state machine needs to retry the GetResource task one additional time after 10 seconds if the state machine encounters a TooManyRequestsException error. If the second attempt fails, the developer wants the state machine to stop running.

How can the developer implement the Lambda retry functionality without adding unnecessary complexity to the state machine'?

- A. Add a Delay task after the GetResource task.
- B. Add a catcher to the GetResource task.
- C. Configure the catcher with an error type of TooManyRequestsException.
- D. Configure the next step to be the Delay task. Configure the Delay task to wait for an interval of 10 seconds. Configure the next step to be the GetResource task.
- E. Add a catcher to the GetResource task. Configure the catcher with an error type of TooManyRequestsException.
- F. an interval of 10 seconds, and a maximum attempts value of 1. Configure the next step to be the GetResource task.
- G. Add a retrier to the GetResource task. Configure the retrier with an error type of TooManyRequestsException, an interval of 10 seconds, and a maximum attempts value of 1.
- H. Duplicate the GetResource task. Rename the new GetResource task to TryAgain. Add a catcher to the original GetResource task.
- I. Configure the catcher with an error type of TooManyRequestsException.
- J. Configure the next step to be TryAgain.

Answer: C

Explanation:

The best way to implement the Lambda retry functionality is to use the Retry field in the state definition of the GetResource task. The Retry field allows the developer to specify an array of retriers, each with an error type, an interval, and a maximum number of attempts. By setting the error type to TooManyRequestsException, the interval to 10 seconds, and the maximum attempts to 1, the developer can achieve the desired behavior of retrying the GetResource task once after 10 seconds if it encounters

a TooManyRequestsException error. If the retry fails, the state machine will stop running. If the GetResource task encounters an UlegalArgumentException error, the state machine will also stop running without retrying, as this error type is not specified in the Retry field. References

- ? Error handling in Step Functions
- ? Handling Errors, Retries, and adding Alerting to Step Function State Machine Executions
- ? The Jitter Strategy for Step Functions Error Retries on the New Workflow Studio

NEW QUESTION 19

A company receives food orders from multiple partners. The company has a microservices application that uses Amazon API Gateway APIs with AWS Lambda integration. Each partner sends orders by calling a customized API that is exposed through API Gateway. The API call invokes a shared Lambda function to process the orders.

Partners need to be notified after the Lambda function processes the orders. Each partner must receive updates for only the partner's own orders. The company wants to add new partners in the future with the fewest code changes possible.

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

- A. Create a different Amazon Simple Notification Service (Amazon SNS) topic for each partner.
- B. Configure the Lambda function to publish messages for each partner to the partner's SNS topic.
- C. Create a different Lambda function for each partner.
- D. Configure the Lambda function to notify each partner's service endpoint directly.
- E. Create an Amazon Simple Notification Service (Amazon SNS) topic.
- F. Configure the Lambda function to publish messages with specific attributes to the SNS topic.
- G. Subscribe each partner to the SNS topic.
- H. Apply the appropriate filter policy to the topic subscriptions.
- I. Create one Amazon Simple Notification Service (Amazon SNS) topic.
- J. Subscribe all partners to the SNS topic.

Answer: C

Explanation:

Amazon Simple Notification Service (Amazon SNS) is a fully managed messaging service that enables pub/sub communication between distributed systems. The developer can create an SNS topic and configure the Lambda function to publish messages with specific attributes to the topic. The developer can subscribe each partner to the SNS topic and apply the appropriate filter policy to the topic subscriptions. This way, each partner will receive updates for only their own orders based on the message attributes. This solution will meet the requirements in the most scalable way and allow adding new partners in the future with minimal code changes.

References:

- ? [Amazon Simple Notification Service (SNS)]

? [Filtering Messages with Attributes - Amazon Simple Notification Service]

NEW QUESTION 20

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 22

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.
- J. Store credentials in a config file
- K. 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 monitoring¹. 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 purposes². 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 deployed³.

References:

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

NEW QUESTION 27

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 31

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

NEW QUESTION 32

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 33

An application uses Lambda functions to extract metadata from files uploaded to an S3 bucket; the metadata is stored in Amazon DynamoDB. The application starts behaving unexpectedly, and the developer wants to examine the logs of the Lambda function code for errors.

Based on this system configuration, where would the developer find the logs?

- A. Amazon S3
- B. AWS CloudTrail
- C. Amazon CloudWatch
- D. Amazon DynamoDB

Answer: C

Explanation:

Amazon CloudWatch is the service that collects and stores logs from AWS Lambda functions. The developer can use CloudWatch Logs Insights to query and analyze the logs for errors and metrics. Option A is not correct because Amazon S3 is a storage service that does not store Lambda function logs. Option B is not correct because AWS CloudTrail is a service that records API calls and events for AWS services, not Lambda function logs. Option D is not correct because Amazon DynamoDB is a database service that does not store Lambda function logs.

References: AWS Lambda Monitoring, [CloudWatch Logs Insights]

NEW QUESTION 35

A company is planning to securely manage one-time fixed license keys in AWS. The company's development team needs to access the license keys in automation scripts that run in Amazon EC2 instances and in AWS CloudFormation stacks.

Which solution will meet these requirements MOST cost-effectively?

- A. Amazon S3 with encrypted files prefixed with "config"
- B. AWS Secrets Manager secrets with a tag that is named SecretString
- C. AWS Systems Manager Parameter Store SecureString parameters
- D. CloudFormation NoEcho parameters

Answer: C

Explanation:

AWS Systems Manager Parameter Store is a service that provides secure, hierarchical storage for configuration data and secrets. Parameter Store supports SecureString parameters, which are encrypted using AWS Key Management Service (AWS KMS) keys. SecureString parameters can be used to store license keys in AWS and retrieve them securely from automation scripts that run in EC2 instances or CloudFormation stacks. Parameter Store is a cost-effective solution because it does not charge for storing parameters or API calls. Reference: Working with Systems Manager parameters

NEW QUESTION 38

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 39

A company is migrating its PostgreSQL database into the AWS Cloud. The company wants to use a database that will secure and regularly rotate database credentials. The company wants a solution that does not require additional programming overhead.

Which solution will meet these requirements?

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

This solution meets the requirements because it uses a PostgreSQL-compatible database that can secure and regularly rotate database credentials without requiring additional programming overhead. Amazon Aurora PostgreSQL is a relational database service that is compatible with PostgreSQL and offers high performance, availability, and scalability. AWS Secrets Manager is a service that helps you protect secrets needed to access your applications, services, and IT resources. You can store database credentials in AWS Secrets Manager and use them to access your Aurora PostgreSQL database. You can also enable automatic rotation of your secrets according to a schedule or an event. AWS Secrets Manager handles the complexity of rotating secrets for you, such as generating new passwords and updating your database with the new credentials. Using Amazon DynamoDB for the database will not meet the requirements because it is a NoSQL database that is not compatible with PostgreSQL. Using AWS Systems Manager Parameter Store for storing and rotating database credentials will require additional programming overhead to integrate with your database.

Reference: [What Is Amazon Aurora?], [What Is AWS Secrets Manager?]

NEW QUESTION 44

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

A developer needs to implement a solution to store the access token. The access token must be encrypted at rest and in transit. The access token must also be accessible from other AWS accounts.

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

- A. Use an AWS Systems Manager Parameter Store SecureString parameter that uses an AWS Key Management Service (AWS KMS) AWS managed key to store the access token
- B. Add a resource-based policy to the parameter to allow access from other account
- C. Update the IAM role of the EC2 instances with permissions to access Parameter Store
- D. Retrieve the token from Parameter Store with the decrypt flag enable
- E. Use the decrypted access token to send the message to the chat.
- F. Encrypt the access token by using an AWS Key Management Service (AWS KMS) customer managed key
- G. Store the access token in an Amazon DynamoDB table
- H. Update the IAM role of the EC2 instances with permissions to access DynamoDB and AWS KMS
- I. Retrieve the token from DynamoDB
- J. Decrypt the token by using AWS KMS on the EC2 instance
- K. Use the decrypted access token to send the message to the chat.
- L. Use AWS Secrets Manager with an AWS Key Management Service (AWS KMS) customer managed key to store the access token
- M. Add a resource-based policy to the secret to allow access from other account
- N. Update the IAM role of the EC2 instances with permissions to access Secrets Manager
- O. Retrieve the token from Secrets Manager
- P. Use the decrypted access token to send the message to the chat.
- Q. Encrypt the access token by using an AWS Key Management Service (AWS KMS) AWS managed key
- R. Store the access token in an Amazon S3 bucket
- S. Add a bucket policy to the S3 bucket to allow access from other account

- T. Update the IAM role of the EC2 instances with permissions to access Amazon S3 and AWS KM
- . Retrieve the token from the S3 bucket
 - . Decrypt the token by using AWS KMS on the EC2 instance
 - . Use the decrypted access token to send the message to the chat.

Answer: C

Explanation:

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

NEW QUESTION 47

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 50

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 55

A company has installed smart meters in all its customer locations. The smart meter's measure power usage at 1-minute intervals and send the usage readings to a remote endpoint for collection. The company needs to create an endpoint that will receive the smart meter readings and store the readings in a database. The company wants to store the location ID and timestamp information.

The company wants to give its customers low-latency access to their current usage and historical usage on demand. The company expects demand to increase significantly. The solution must not impact performance or include downtime while seeing.

When solution will meet these requirements MOST cost-effectively?

- A. Store the smart meter readings in an Amazon RDS database
- B. Create an index on the location ID and timestamp columns. Use the columns to filter on the customers' data.
- C. Store the smart meter readings in an Amazon DynamoDB table. Create a composite key by using the location ID and timestamp column.
- D. Use the columns to filter on the customers' data.
- E. Store the smart meter readings in Amazon ElastiCache for Redis. Create a Sorted set key by using the location ID and timestamp column.
- F. Use the columns to filter on the customers' data.
- G. Store the smart meter readings in Amazon S3. Partition the data by using the location ID and timestamp column.
- H. Use Amazon Athena to filter on the customers' data.

Answer: B

Explanation:

The solution that will meet the requirements most cost-effectively is to store the smart meter readings in an Amazon DynamoDB table. Create a composite key by using the location ID and timestamp columns. Use the columns to filter on the customers' data. This way, the company can leverage the scalability, performance, and low latency of DynamoDB to store and retrieve the smart meter readings. The company can also use the composite key to query the data by location ID and timestamp efficiently. The other options either involve more expensive or less scalable services, or do not provide low-latency access to the current usage.

Reference: Working with Queries in DynamoDB

NEW QUESTION 59

A company is creating an application that processes CSV files from Amazon S3. A developer has created an S3 bucket. The developer has also created an AWS Lambda function to process the CSV files from the S3 bucket.

Which combination of steps will invoke the Lambda function when a CSV file is uploaded to Amazon S3? (Select TWO.)

- A. Create an Amazon EventBridge rule. Configure the rule with a pattern to match the S3 object created event.
- B. Schedule an Amazon EventBridge rule to run a new Lambda function to scan the S3 bucket.
- C. Add a trigger to the existing Lambda function.
- D. Set the trigger type to EventBridge. Select the Amazon EventBridge rule.
- E. Create a new Lambda function to scan the S3 bucket for recently added S3 objects.
- F. Add S3 Lifecycle rules to invoke the existing Lambda function.

Answer: AC

Explanation:

To invoke a Lambda function when a CSV file is uploaded to Amazon S3, you can use Amazon EventBridge to create a rule that matches the S3 object created event. Then, you can add a trigger to the existing Lambda function and set the trigger type to EventBridge. This way, the Lambda function will be invoked whenever a new CSV file is added to the S3 bucket. References

? Tutorial: Using an Amazon S3 trigger to invoke a Lambda function

? How to trigger my Lambda Function once the file is uploaded to S3 bucket

? Lambda Function to be invoked or triggered by S3 (CSV file upload ...)

NEW QUESTION 63

A developer has code that is stored in an Amazon S3 bucket. The code must be deployed as an AWS Lambda function across multiple accounts in the same AWS Region as the S3 bucket. An AWS CloudFormation template that runs for each account will deploy the Lambda function.

What is the MOST secure way to allow CloudFormation to access the Lambda code in the S3 bucket?

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

This solution allows the CloudFormation service role to access the S3 bucket from any account, as long as it has the S3 GetObject permission. The bucket policy grants access to any principal with the GetObject permission, which is the least privilege needed to deploy the Lambda code. This is more secure than granting ListBucket permission, which is not required for deploying Lambda code, or using a service-based link, which is not supported for Lambda functions.

Reference: AWS CloudFormation Service Role, Using AWS Lambda with Amazon S3

NEW QUESTION 65

A company is building a microservices application that consists of many AWS Lambda functions. The development team wants to use AWS Serverless Application Model (AWS SAM) templates to automatically test the Lambda functions. The development team plans to test a small percentage of traffic that is directed to new updates before the team commits to a full deployment of the application.

Which combination of steps will meet these requirements in the MOST operationally efficient way? (Select TWO.)

- A. Use AWS SAM CLI commands in AWS CodeDeploy to invoke the Lambda functions to test the deployment.
- B. Declare the EventInvokeConfig on the Lambda functions in the AWS SAM templates with OnSuccess and OnFailure configurations. Enable gradual deployments through AWS SAM templates.
- C. Set the deployment preference type to Canary10Percent130Minutes. Use hooks to test the deployment.
- D. Set the deployment preference type to Linear10PercentEvery10Minutes. Use hooks to test the deployment.

Answer: CD

Explanation:

This solution will meet the requirements by using AWS Serverless Application Model (AWS SAM) templates and gradual deployments to automatically test the Lambda functions. AWS SAM templates are configuration files that define serverless applications and resources such as Lambda functions. Gradual deployments are a feature of AWS SAM that enable deploying new versions of Lambda functions incrementally, shifting traffic gradually, and performing validation tests during deployment. The developer can enable gradual deployments through AWS SAM templates by adding a DeploymentPreference property to each Lambda function resource in the template. The developer can set the deployment preference type to Canary10Percent30Minutes, which means that 10 percent of traffic will be shifted to the new version of the Lambda function for 30 minutes before shifting 100 percent of traffic. The developer can also use hooks to test the deployment,

which are custom Lambda functions that run before or after traffic shifting and perform validation tests or rollback actions.
References: [AWS Serverless Application Model (AWS SAM)], [Gradual Code Deployment]

NEW QUESTION 67

A developer is troubleshooting an application that uses Amazon DynamoDB in the us-west-2 Region. The application is deployed to an Amazon EC2 instance. The application requires read-only permissions to a table that is named Cars. The EC2 instance has an attached IAM role that contains the following IAM policy.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ReadOnlyAPIActions",
      "Effect": "Allow",
      "Action": [
        "dynamodb:GetItem",
        "dynamodb:BatchGetItem",
        "dynamodb:Scan",
        "dynamodb:Query",
        "dynamodb:ConditionCheckItem"
      ],
      "Resource": "arn:aws:dynamodb:us-west-2:account-id:table/Cars"
    }
  ]
}
```

When the application tries to read from the Cars table, an Access Denied error occurs. How can the developer resolve this error?

- A. Modify the IAM policy resource to be "arn:aws:dynamodb:us-west-2:account-id:table/*"
- B. Modify the IAM policy to include the dynamodb:* action
- C. Create a trust policy that specifies the EC2 service principal
- D. Associate the role with the policy.
- E. Create a trust relationship between the role and dynamodb.amazonaws.com.

Answer: C

Explanation:

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/access-control-overview.html#access-control-resource-ownership>

NEW QUESTION 69

A developer is preparing to begin development of a new version of an application. The previous version of the application is deployed in a production environment. The developer needs to deploy fixes and updates to the current version during the development of the new version of the application. The code for the new version of the application is stored in AWS CodeCommit.

Which solution will meet these requirements?

- A. From the main branch, create a feature branch for production bug fixes
- B. Create a second feature branch from the main branch for development of the new version.
- C. Create a Git tag of the code that is currently deployed in production
- D. Create a Git tag for the development of the new version
- E. Push the two tags to the CodeCommit repository.
- F. From the main branch, create a branch of the code that is currently deployed in production
- G. Apply an IAM policy that ensures no other users can push or merge to the branch.
- H. Create a new CodeCommit repository for development of the new version of the application
- I. Create a Git tag for the development of the new version.

Answer: A

Explanation:

? A feature branch is a branch that is created from the main branch to work on a specific feature or task. Feature branches allow developers to isolate their work from the main branch and avoid conflicts with other changes. Feature branches can be merged back to the main branch when the feature or task is completed and tested.

? In this scenario, the developer needs to maintain two parallel streams of work: one for fixing and updating the current version of the application that is deployed in production, and another for developing the new version of the application. The developer can use feature branches to achieve this goal.

? The developer can create a feature branch from the main branch for production bug fixes. This branch will contain the code that is currently deployed in production, and any fixes or updates that need to be applied to it. The developer can push this branch to the CodeCommit repository and use it to deploy changes to the production environment.

? The developer can also create a second feature branch from the main branch for development of the new version of the application. This branch will contain the code that is under development for the new version, and any changes or enhancements that are part of it. The developer can push this branch to the CodeCommit repository and use it to test and deploy the new version of the application in a separate environment.

? By using feature branches, the developer can keep the main branch stable and clean, and avoid mixing code from different versions of the application. The developer can also easily switch between branches and merge them when needed.

NEW QUESTION 73

A development team maintains a web application by using a single AWS CloudFormation template. The template defines web servers and an Amazon RDS database. The team uses the CloudFormation template to deploy the CloudFormation stack to different environments.

During a recent application deployment, a developer caused the primary development database to be dropped and recreated. The result of this incident was a loss of data. The team needs to avoid accidental database deletion in the future.

Which solutions will meet these requirements? (Choose two.)

- A. Add a CloudFormation DeletionPolicy attribute with the Retain value to the database resource.
- B. Update the CloudFormation stack policy to prevent updates to the database.
Modify the database to use a Multi-AZ deployment.
- C. Create a CloudFormation stack set for the web application and database deployments.
- E. Add a CloudFormation DeletionPolicy attribute with the Retain value to the stack.

Answer: AB

Explanation:

AWS CloudFormation is a service that enables developers to model and provision AWS resources using templates. The developer can add a CloudFormation Deletion Policy attribute with the Retain value to the database resource. This will prevent the database from being deleted when the stack is deleted or updated. The developer can also update the CloudFormation stack policy to prevent updates to the database. This will prevent accidental changes to the database configuration or properties.

References:

? [What Is AWS CloudFormation? - AWS CloudFormation]

? [DeletionPolicy Attribute - AWS CloudFormation]

? [Protecting Resources During Stack Updates - AWS CloudFormation]

NEW QUESTION 76

A company has an application that stores data in Amazon RDS instances. The application periodically experiences surges of high traffic that cause performance problems.

During periods of peak traffic, a developer notices a reduction in query speed in all database queries.

The team's technical lead determines that a multi-threaded and scalable caching solution should be used to offload the heavy read traffic. The solution needs to improve performance.

Which solution will meet these requirements with the LEAST complexity?

- A. Use Amazon ElastiCache for Memcached to offload read requests from the main database.
- B. Replicate the data to Amazon DynamoDB
- C. Set up a DynamoDB Accelerator (DAX) cluster.
- D. Configure the Amazon RDS instances to use Multi-AZ deployment with one standby instance.
- E. Offload read requests from the main database to the standby instance.
- F. Use Amazon ElastiCache for Redis to offload read requests from the main database.

Answer: A

Explanation:

? Amazon ElastiCache for Memcached is a fully managed, multithreaded, and scalable in-memory key-value store that can be used to cache frequently accessed data and improve application performance¹. By using Amazon ElastiCache for Memcached, the developer can reduce the load on the main database and handle high traffic surges more efficiently.

? To use Amazon ElastiCache for Memcached, the developer needs to create a cache cluster with one or more nodes, and configure the application to store and retrieve data from the cache cluster². The developer can use any of the supported Memcached clients to interact with the cache cluster³. The developer can also use Auto Discovery to dynamically discover and connect to all cache nodes in a cluster⁴.

? Amazon ElastiCache for Memcached is compatible with the Memcached protocol, which means that the developer can use existing tools and libraries that work with

Memcached¹. Amazon ElastiCache for Memcached also supports data partitioning, which allows the developer to distribute data among multiple nodes and scale out the cache cluster as needed.

? Using Amazon ElastiCache for Memcached is a simple and effective solution that meets the requirements with the least complexity. The developer does not need to change the database schema, migrate data to a different service, or use a different caching model. The developer can leverage the existing Memcached ecosystem and easily integrate it with the application.

NEW QUESTION 80

A developer has an application that stores data in an Amazon S3 bucket. The application uses an HTTP API to store and retrieve objects. When the PutObject API operation adds objects to the S3 bucket the developer must encrypt these objects at rest by using server-side encryption with Amazon S3 managed keys (SSE-S3).

Which solution will meet this requirement?

- A. Create an AWS Key Management Service (AWS KMS) key
- B. Assign the KMS key to the S3 bucket.
- C. Set the x-amz-server-side-encryption header when invoking the PutObject API operation.
- D. Provide the encryption key in the HTTP header of every request.
- E. Apply TLS to encrypt the traffic to the S3 bucket.

Answer: B

Explanation:

Amazon S3 supports server-side encryption, which encrypts data at rest on the server that stores the data. One of the encryption options is SSE-S3, which uses keys managed by S3. To use SSE-S3, the x-amz-server-side-encryption header must be set to AES256 when invoking the PutObject API operation. This instructs

S3 to encrypt the object data with SSE-S3 before saving it on disks in its data centers, and decrypt it when it is downloaded. Reference:

Protecting data using server-side encryption with Amazon S3-managed encryption keys (SSE-S3)

NEW QUESTION 83

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 Canary10Percent10Minute
- B. Set the Deployment Preference Type to Linear10PercentEvery10Minute
- C. Set the Deployment Preference Type to Canary10Percent10Minute
- D. Set the AutoPublishAlias property to the Lambda alias.
- E. Set the Deployment Preference Type to Canary10Percent10Minute
- F. Set the PreTraffic and PostTraffic properties to the Lambda alias.
- G. Set the Deployment Preference Type to Linear10PercentEvery10Minute
- H. Set the PreTraffic and PostTraffic properties to the Lambda alias.

Answer: A

Explanation:

The AWS Serverless Application Model (AWS SAM) comes built-in with CodeDeploy to provide gradual AWS Lambda deployments¹. The DeploymentPreference property in AWS SAM allows you to specify the type of deployment that you want. The Canary10Percent10Minutes option means that 10 percent of your customer traffic is immediately shifted to your new version. After 10 minutes, all traffic is shifted to the new version¹. The AutoPublishAlias property in AWS SAM allows AWS SAM to automatically create an alias that points to the updated version of the Lambda function¹. Therefore, option A is correct.

NEW QUESTION 84

An application that runs on AWS Lambda requires access to specific highly confidential objects in an Amazon S3 bucket. In accordance with the principle of least privilege a company grants access to the S3 bucket by using only temporary credentials. How can a developer configure access to the S3 bucket in the MOST secure way?

- A. Hardcode the credentials that are required to access the S3 objects in the application code
- B. Use the credentials to access the required S3 objects.
 - Create a secret access key and access key ID with permission to access the S3 bucket
- C. Store the key and key ID in AWS Secrets Manager**
- D. Configure the application to retrieve the Secrets Manager secret and use the credentials to access the S3 objects.
- E. Create a Lambda function execution role. Attach a policy to the role that grants access to specific objects in the S3 bucket.
- F. Create a secret access key and access key ID with permission to access the S3 bucket. Store the key and key ID as environment variables in Lambda
- G. Use the environment variables to access the required S3 objects.

Answer: C

Explanation:

This solution will meet the requirements by creating a Lambda function execution role, which is an IAM role that grants permissions to a Lambda function to access AWS resources such as Amazon S3 objects. The developer can attach a policy to the role that grants access to specific objects in the S3 bucket that are required by the application, following the principle of least privilege. Option A is not optimal because it will hardcode the credentials that are required to access S3 objects in the application code, which is insecure and difficult to maintain. Option B is not optimal because it will create a secret access key and access key ID with permission to access the S3 bucket, which will introduce additional security risks and complexity for storing and managing credentials. Option D is not optimal because it will store the secret access key and access key ID as environment variables in Lambda, which is also insecure and difficult to maintain. References: [AWS Lambda Execution Role], [Using AWS Lambda with Amazon S3]

NEW QUESTION 87

An organization is using Amazon CloudFront to ensure that its users experience low-latency access to its web application. The organization has identified a need to encrypt all traffic between users and CloudFront, and all traffic between CloudFront and the web application. How can these requirements be met? (Select TWO)

- A. Use AWS KMS to encrypt traffic between CloudFront and the web application.
- B. Set the Origin Protocol Policy to "HTTPS Only".**
- C. Set the Origin's HTTP Port to 443.**
- D. Set the Viewer Protocol Policy to "HTTPS Only" or Redirect HTTP to HTTPS"
- E. Enable the CloudFront option Restrict Viewer Access.

Answer: BC

Explanation:

This solution will meet the requirements by ensuring that all traffic between users and CloudFront, and all traffic between CloudFront and the web application, are encrypted using HTTPS protocol. The Origin Protocol Policy determines how CloudFront communicates with the origin server (the web application), and setting it to "HTTPS Only" will force CloudFront to use HTTPS for every request to the origin server. The Viewer Protocol Policy determines how CloudFront responds to HTTP or HTTPS requests from users, and setting it to "HTTPS Only" or "Redirect HTTP to HTTPS" will force CloudFront to use HTTPS for every response to users. Option A is not optimal because it will use AWS KMS to encrypt traffic between CloudFront and the web application, which is not necessary or supported by CloudFront. Option C is not optimal because it will set the origin's HTTP port to 443, which is incorrect as port 443 is used for HTTPS protocol, not HTTP protocol. Option E is not optimal because it will enable the CloudFront option Restrict Viewer Access, which is used for controlling access to private content using signed URLs or signed cookies, not for encrypting traffic. References: [Using HTTPS with CloudFront], [Restricting Access to Amazon S3 Content by Using an Origin Access Identity]

NEW QUESTION 89

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 91

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 93

A developer created an AWS Lambda function that performs a series of operations that involve multiple AWS services. The function's duration time is higher than normal. To determine the cause of the issue, the developer must investigate traffic between the services without changing the function code

Which solution will meet these requirements?

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

AWS X-Ray is a service that helps you analyze and debug your applications. You can use X-Ray to trace requests made to your Lambda function and other AWS services, and identify performance bottlenecks and errors. Enabling active tracing in your Lambda function allows X-Ray to collect data from the function invocation and the downstream services that it calls. You can then review the logs and service maps in X-Ray to diagnose the issue. References

? Monitoring and troubleshooting Lambda functions - AWS Lambda

? Using AWS Lambda with AWS X-Ray

? Troubleshoot Lambda function cold start issues | AWS re:Post

NEW QUESTION 96

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 table
- 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.h tml](https://docs.aws.amazon.com/amazondynamodb/latest/APIReference/API_BatchGetItem.html)

NEW QUESTION 97

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

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

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

API call.

D. Capture incoming requests on-premises and configure an AWS Lambda function to pull, process, and relay relevant data to X-Ray using the PutTelemetryRecords API call.

Answer: B

Explanation:

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

References:

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

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

NEW QUESTION 100

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 101

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 106

A developer migrated a legacy application to an AWS Lambda function. The function uses a third-party service to pull data with a series of API calls at the end of each month. The function then processes the data to generate the monthly reports. The function has been working with no issues so far. The third-party service recently issued a restriction to allow a fixed number of API calls each minute and each day. If the API calls exceed the limit for each minute or each day, then the service will produce errors. The API also provides the minute limit and daily limit in the response header. This restriction might extend the overall process to multiple days because the process is consuming more API calls than the available limit. What is the MOST operationally efficient way to refactor the serverless application to accommodate this change?

- A. Use an AWS Step Functions state machine to monitor API failure
- B. Use the Wait state to delay calling the Lambda function.
- C. Use an Amazon Simple Queue Service (Amazon SQS) queue to hold the API call
- D. Configure the Lambda function to poll the queue within the API threshold limits.
- E. Use an Amazon CloudWatch Logs metric to count the number of API call
- F. Configure an Amazon CloudWatch alarm that stops the currently running instance of the Lambda function when the metric exceeds the API threshold limits.
- G. Use Amazon Kinesis Data Firehose to batch the API calls and deliver them to an Amazon S3 bucket with an event notification to invoke the Lambda function.

Answer: A

Explanation:

The solution that will meet the requirements is to use an AWS Step Functions state machine to monitor API failures. Use the Wait state to delay calling the Lambda function. This way, the developer can refactor the serverless application to accommodate the change in a way that is automated and scalable. The developer can use Step Functions to orchestrate the Lambda function and handle any errors or retries. The developer can also use the Wait state to pause the execution for a specified duration or until a specified timestamp, which can help avoid exceeding the API limits. The other options either involve using additional services that are not necessary or appropriate for this scenario, or do not address the issue of API failures.

Reference: AWS Step Functions Wait state

NEW QUESTION 111

A developer is creating an AWS Lambda function in VPC mode. An Amazon S3 event will invoke the Lambda function when an object is uploaded into an S3 bucket. The Lambda function will process the object and produce some analytic results that will be recorded into a file. Each processed object will also generate a log entry that will be recorded into a file. Other Lambda functions, AWS services, and on-premises resources must have access to the result files and log file. Each log entry must also be appended to the same shared log file. The developer needs a solution that can share files and append results into an existing file. Which solution should the developer use to meet these requirements?

- A. Create an Amazon Elastic File System (Amazon EFS) file system
- B. Mount the EFS file system in Lambda
- C. Store the result files and log file in the mount point
- D. Append the log entries to the log file.
- E. Create an Amazon Elastic Block Store (Amazon EBS) Multi-Attach enabled volume. Attach the EBS volume to all Lambda functions. download the log file, append the log entries, and upload the modified log file to Amazon EBS
- F. Update the Lambda function code to
- G. Create a reference to the /tmp local directory
- H. Store the result files and log file by using the directory reference
- I. Append the log entry to the log file.
- J. Create a reference to the /opt storage directory. Store the result files and log file by using the directory reference. Append the log entry to the log file

Answer: A

Explanation:

<https://aws.amazon.com/blogs/compute/using-amazon-efs-for-aws-lambda-in-your-serverless-applications/>

NEW QUESTION 114

A company is developing an e-commerce 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 environment 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 119

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 120

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 cach

- B. Update the application to return cached content based upon the default request headers.
- C. Override the cache method in the selected stage of API Gatewa
- D. Select the POST method.

- E. Save the latest request response in Lambda /tmp director
- F. Update the Lambda function to check the /tmp directory.
- G. Save the latest request in AWS Systems Manager Parameter Store
- H. Modify the Lambda function to take the latest request response from Parameter Store.

Answer: B

Explanation:

Amazon API Gateway provides tools for creating and documenting web APIs that route HTTP requests to Lambda functions². You can secure access to your API with authentication and authorization controls. Your APIs can serve traffic over the internet or can be accessible only within your VPC². You can override the cache method in the selected stage of API Gateway². Therefore, option B is correct.

NEW QUESTION 121

A developer is building an application that gives users the ability to view bank account from multiple sources in a single dashboard. The developer has automated the process to retrieve API credentials for these sources. The process invokes an AWS Lambda function that is associated with an AWS CloudFormation custom resource.

The developer wants a solution that will store the API credentials with minimal operational overhead.

Which solution will meet these requirements?

- A. Add an AWS Secrets Manager GenerateSecretString resource to the CloudFormation template
- B. Set the value to reference new credentials to the CloudFormation resource.
- C. Use the AWS SDK ssm PutParameter operation in the Lambda function from the existing, custom resource to store the credentials as a parameter
- D. Set the parameter value to reference the new credential
- E. Set the parameter type to SecureString.
- F. Add an AWS Systems Manager Parameter Store resource to the CloudFormation template
- G. Set the CloudFormation resource value to reference the new credentials. Set the resource NoEcho attribute to true.
- H. Use the AWS SDK ssm PutParameter operation in the Lambda function from the existing custom resources to store the credentials as a parameter
- I. Set the parameter value to reference the new credential
- J. Set the parameter NoEcho attribute to true.

Answer: B

Explanation:

The solution that will meet the requirements is to use the AWS SDK ssm PutParameter operation in the Lambda function from the existing custom resource to store the credentials as a parameter. Set the parameter value to reference the new credentials. Set the parameter type to SecureString. This way, the developer can store the API credentials with minimal operational overhead, as AWS Systems Manager Parameter Store provides secure and scalable storage for configuration data. The SecureString parameter type encrypts the parameter value with AWS Key Management Service (AWS KMS). The other options either involve adding additional resources to the CloudFormation template, which increases complexity and cost, or do not encrypt the parameter value, which reduces security.

Reference: Creating Systems Manager parameters

NEW QUESTION 126

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.ECSLinear10PercentEvery1 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 130

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 134

A developer is storing sensitive data generated by an application in Amazon S3. The developer wants to encrypt the data at rest. A company policy requires an audit trail of when the AWS Key Management Service (AWS KMS) key was used and by whom. Which encryption option will meet these requirements?

- A. Server-side encryption with Amazon S3 managed keys (SSE-S3)
- B. Server-side encryption with AWS KMS managed keys (SSE-KMS)
- C. Server-side encryption with customer-provided keys (SSE-C)
- D. Server-side encryption with self-managed keys

Answer: B

Explanation:

This solution meets the requirements because it encrypts data at rest using AWS KMS keys and provides an audit trail of when and by whom they were used. Server-side encryption with AWS KMS managed keys (SSE-KMS) is a feature of Amazon S3 that encrypts data using keys that are managed by AWS KMS. When SSE-KMS is enabled for an S3 bucket or object, S3 requests AWS KMS to generate data keys and encrypts data using these keys. AWS KMS logs every use of its keys in AWS CloudTrail, which records all API calls to AWS KMS as events. These events include information such as who made the request, when it was made, and which key was used. The company policy can use CloudTrail logs to audit critical events related to their data encryption and access. Server-side encryption with Amazon S3 managed keys (SSE-S3) also encrypts data at rest using keys that are managed by S3, but does not provide an audit trail of key usage. Server-side encryption with customer-provided keys (SSE-C) and server-side encryption with self-managed keys also encrypt data at rest using keys that are provided or managed by customers, but do not provide an audit trail of key usage and require additional overhead for key management.

Reference: [Protecting Data Using Server-Side Encryption with AWS KMS–Managed Encryption Keys (SSE-KMS)], [Logging AWS KMS API calls with AWS CloudTrail]

NEW QUESTION 138

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 141

A developer is building an application that uses AWS API Gateway APIs, AWS Lambda function, 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 146

A development team wants to build a continuous integration/continuous delivery (CI/CD) pipeline. The team is using AWS CodePipeline to automate the code build and deployment. The team wants to store the program code to prepare for the CI/CD pipeline.

Which AWS service should the team use to store the program code?

- A. AWS CodeDeploy
B. AWS CodeArtifact
C. AWS CodeCommit
D. Amazon CodeGuru

Answer: C

Explanation:

AWS CodeCommit is a service that provides fully managed source control for hosting secure and scalable private Git repositories. The development team can use CodeCommit to store the program code and prepare for the CI/CD pipeline. CodeCommit integrates with other AWS services such as CodePipeline, CodeBuild, and CodeDeploy to automate the code build and deployment process.

References:

? [What Is AWS CodeCommit? - AWS CodeCommit]

? [AWS CodePipeline - AWS CodeCommit]

NEW QUESTION 150

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 155

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 156

A company runs an application on AWS. The application uses an AWS Lambda function that is configured with an Amazon Simple Queue Service (Amazon SQS) queue called high priority queue as the event source. A developer is updating the Lambda function with another SQS queue called low priority queue as the event source. The Lambda function must always read up to 10 simultaneous messages from the high priority queue before processing messages from low priority queue. The Lambda function must be limited to 100 simultaneous invocations.

Which solution will meet these requirements?

- A. Set the event source mapping batch size to 10 for the high priority queue and to 90 for the low priority queue.
- B. Set the delivery delay to 0 seconds for the high priority queue and to 10 seconds for the low priority queue.
- C. Set the event source mapping maximum concurrency to 10 for the high priority queue and to 90 for the low priority queue.
- D. Set the event source mapping batch window to 10 for the high priority queue and to 90 for the low priority queue.

Answer: C

Explanation:

Setting the event source mapping maximum concurrency is the best way to control how many messages from each queue are processed by the Lambda function. processed concurrently from the same event

at a time. The maximum concurrency setting limits the number of batches that can be processed concurrently from the same event source. By setting it to 10 for the high priority queue and to 90 for the low priority queue, the developer can ensure that the Lambda function always reads up to 10 simultaneous messages from the high priority queue before processing messages from the low priority queue, and that the total number of concurrent invocations does not exceed 100. The other solutions are either not effective or not relevant. The batch size setting controls how many messages are sent to the Lambda function in a single invocation, not how many invocations are allowed at a time. The delivery delay setting controls how long a message is invisible in the queue after it is sent, not how often it is processed by the Lambda function. The batch window setting controls how long the event source mapping can buffer messages before sending a batch, not how many batches are processed concurrently. References

? Using AWS Lambda with Amazon SQS

? AWS Lambda Event Source Mapping - Examples and best practices | Shisho Dojo

? Lambda event source mappings - AWS Lambda

? aws_lambda_event_source_mapping - Terraform Registry

NEW QUESTION 157

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 160

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 asynchronously.
- 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 'AWS::Lambda::Invoke' 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 165

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 169

.....

Thank You for Trying Our Product

We offer two products:

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

2nd - Questions and Answers in PDF Format

AWS-Certified-Developer-Associate Practice Exam Features:

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

100% Actual & Verified — Instant Download, Please Click
[Order The AWS-Certified-Developer-Associate Practice Test Here](#)