Free Questions for DVA-C02 by ebraindumps

Shared by Grimes on 12-12-2023

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

A developer wants to add request validation to a production environment Amazon API Gateway API. The developer needs to test the changes

before the API is deployed to the production environment. For the test, the developer will send test requests to the API through a testing tool.

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

Options:

A- Export the existing API to an OpenAPI file. Create a new API. Import the OpenAPI file. Modify the new API to add request validation. Perform the tests. Modify the existing API to add request validation. Deploy the existing API to production.

B- Modify the existing API to add request validation. Deploy the updated API to a new API Gateway stage. Perform the tests. Deploy the updated API to the API Gateway production stage.

C- Create a new API. Add the necessary resources and methods, including new request validation. Perform the tests. Modify the existing API to add request validation. Deploy the existing API to production.

D- Clone the existing API. Modify the new API to add request validation. Perform the tests. Modify the existing API to add request validation. Deploy the existing API to production.

Answer:

B

Explanation:

Amazon API Gateway allows you to create, deploy, and manage a RESTful API to expose backend HTTP endpoints, AWS Lambda functions, or other AWS services1.You can use API Gateway to perform basic validation of an API request before proceeding with the integration request1.When the validation fails, API Gateway immediately fails the request, returns a 400 error response to the caller, and publishes the validation results in CloudWatch Logs1.

To test changes before deploying to a production environment, you can modify the existing API to add request validation and deploy the updated API to a new API Gateway stage1. This allows you to perform tests without affecting the production environment.Once testing is complete and successful, you can then deploy the updated API to the API Gateway production stage1.

This approach has the least operational overhead as it avoids unnecessary creation of new APIs or exporting and importing of APIs.It leverages the existing infrastructure and only requires changes in the configuration of the existing API1.

Question 2

Question Type: MultipleChoice

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?

Options:

A- Create sample events based on the Lambda documentation. Create automated test scripts that use the cdk local invoke command to invoke the Lambda functions. Check the response. Document the test scripts for the other developers on the team. Update the CI/CD pipeline to run the test scripts.

B- Install a unit testing framework that reproduces the Lambda execution environment. Create sample events based on the Lambda documentation. Invoke the handler function by using a unit testing framework. Check the response. Document how to run the unit testing framework for the other developers on the team. Update the CI/CD pipeline to run the unit testing framework.

C- Install the AWS Serverless Application Model (AWS SAM) CLI tool. Use the sam local generate-event command to generate sample events for the automated tests. Create automated test scripts that use the sam local invoke command to invoke the Lambda functions. 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- Create sample events based on the Lambda documentation. Create a Docker container from the Node.js base image to invoke the Lambda functions. Check the response. Document how to run the Docker container for the other developers on the team. Update the

CllCD pipeline to run the Docker container.

Answer:

C

Explanation:

The AWS Serverless Application Model Command Line Interface (AWS SAM CLI) is a command-line tool for local development and testing of Serverless applications3.Thesam local generate-eventcommand of AWS SAM CLI generates sample events for automated tests3.Thesam local invokecommand is used to invoke Lambda functions3. Therefore, option C is correct.

Question 3

Question Type: MultipleChoice

A developer is creating an AWS Lambda function that consumes messages from an Amazon Simple Queue Service (Amazon SQS) standard queue. The developer notices that the Lambda function processes some messages multiple times.

How should developer resolve this issue MOST cost-effectively?

Options:

A- Change the Amazon SQS standard queue to an Amazon SQS FIFO queue by using the Amazon SQS message deduplication ID.

B- Set up a dead-letter queue.

C- Set the maximum concurrency limit of the AWS Lambda function to 1

D- Change the message processing to use Amazon Kinesis Data Streams instead of Amazon SQS.

Answer:

A

Explanation:

Amazon Simple Queue Service (Amazon SQS) is a fully managed queue service that allows you to de-couple and scale for applications1.Amazon SQS offers two types of queues: Standard and FIFO (First In First Out) queues1.The FIFO queue uses themessageDeduplicationIdproperty to treat messages with the same value as duplicate2. Therefore, changing the Amazon SQS standard queue to an Amazon SQS FIFO queue using the Amazon SQS message deduplication ID can help resolve the issue of the Lambda function processing some messages multiple times. Therefore, option A is correct.

Question 4

Question Type: MultipleChoice

A company is building a compute-intensive application that will run on a fleet of Amazon EC2 instances. The application uses attached Amazon

Elastic Block Store (Amazon EBS) volumes for storing dat

a. The Amazon EBS volumes will be created at time of initial deployment. The

application will process sensitive information. All of the data must be encrypted. The solution should not impact the application's performance.

Which solution will meet these requirements?

Options:

A- Configure the fleet of EC2 instances to use encrypted EBS volumes to store data.

B- Configure the application to write all data to an encrypted Amazon S3 bucket.

C- Configure a custom encryption algorithm for the application that will encrypt and decrypt all data.

D- Configure an Amazon Machine Image (AMI) that has an encrypted root volume and store the data to ephemeral disks.

Answer:

A

Explanation:

Amazon Elastic Block Store (Amazon EBS) provides block level storage volumes for use with Amazon EC2 instances1.Amazon EBS encryption offers a straight-forward encryption solution for your EBS resources associated with your EC2 instances1.When you create an encrypted EBS volume and attach it to a supported instance type, the following types of data are encrypted: Data at rest inside the volume, all data moving between the volume and the instance, all snapshots created from the volume, and all volumes created from those snapshots1. Therefore, option A is correct.

Question 5

Question Type: MultipleChoice

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

The developer needs to secure the API credentials and enforce automatic credentials rotation on a quarterly basis.

Which solution will meet these requirements MOST securely?

Options:

A- Use AWS Key Management Service (AWS KMS) to encrypt the configuration file. Decrypt the configuration file when users make API calls to the SaaS vendor. Enable rotation.

B- Retrieve temporary credentials from AWS Security Token Service (AWS STS) every 15 minutes. Use the temporary credentials when users make API calls to the SaaS vendor.

C- Store the credentials in AWS Secrets Manager and enable rotation. Configure the API to have Secrets Manager access.

D- Store the credentials in AWS Systems Manager Parameter Store and enable rotation. Retrieve the credentials when users make API calls to the SaaS vendor.

Answer:

C

Explanation:

Store the credentials in AWS Secrets Manager and enable rotation. Configure the API to have Secrets Manager access. This is correct. This solution will meet the requirements most securely, because it uses a service that is designed to store and manage secrets such as API credentials.AWS Secrets Manager helps you protect access to your applications, services, and IT resources by enabling you to rotate, manage, and retrieve secrets throughout their lifecycle1.You can store secrets such as passwords, database strings, API keys, and license codes as encrypted values2.You can also configure automatic rotation of your secrets on a schedule that you specify3.You can use the AWS SDK or CLI to retrieve secrets from Secrets Manager when you need them4. This way, you can avoid storing credentials in plaintext files or hardcoding them in your code.

Question 6

Question Type: MultipleChoice

An developer is building a serverless application by using the AWS Serverless Application Model (AWS SAM). The developer is currently testing the application in a development environment. When the application is nearly finsihed, the developer will need to set up additional testing and staging environments for a quality assurance team.

The developer wants to use a feature of the AWS SAM to set up deployments to multiple environments.

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

Options:

A- Add a configuration file in TOML format to group configuration entries to every environment. Add a table for each testing and staging environment. Deploy updates to the environments by using the sam deploy command and the --config-env flag that corresponds to the each environment.

B- Create additional AWS SAM templates for each testing and staging environment. Write a custom shell script that uses the sam deploy command and the --template-file flag to deploy updates to the environments.

C- Create one AWS SAM configuration file that has default parameters. Perform updates to the testing and staging environments by using the ---parameter-overrides flag in the AWS SAM CLI and the parameters that the updates will override.

D- Use the existing AWS SAM template. Add additional parameters to configure specific attributes for the serverless function and database table resources that are in each environment. Deploy updates to the testing and staging environments by using the sam

deploy command.

Answer:

A

Explanation:

The correct answer is A. Add a configuration file in TOML format to group configuration entries to every environment. Add a table for each testing and staging environment. Deploy updates to the environments by using the sam deploy command and the --config-env flag that corresponds to the each environment.

A) Add a configuration file in TOML format to group configuration entries to every environment. Add a table for each testing and staging environment. Deploy updates to the environments by using the sam deploy command and the --config-env flag that corresponds to the each environment. This is correct. This solution will meet the requirements with the least development effort, because it uses a feature of the AWS SAM CLI that supports a project-level configuration file that can be used to configure AWS SAM CLI command parameter values1. The configuration file can have multiple environments, each with its own set of parameter values, such as stack name, region, capabilities, and more2. The developer can use the --config-env option to specify which environment to use when deploying the application3. This way, the developer can avoid creating multiple templates or scripts, or manually overriding parameters for each environment.

B) Create additional AWS SAM templates for each testing and staging environment. Write a custom shell script that uses the sam deploy command and the --template-file flag to deploy updates to the environments. This is incorrect. This solution will not meet the requirements with the least development effort, because it requires creating and maintaining multiple templates and scripts for each environment. This can introduce duplication, inconsistency, and complexity in the deployment process.

C) Create one AWS SAM configuration file that has default parameters. Perform updates to the testing and staging environments by using the ---parameter-overrides flag in the AWS SAM CLI and the parameters that the updates will override. This is incorrect. This solution will not meet the requirements with the least development effort, because it requires manually specifying and overriding parameters for each environment every time the developer deploys the application. This can be error-prone, tedious, and inefficient.

D) Use the existing AWS SAM template. Add additional parameters to configure specific attributes for the serverless function and database table resources that are in each environment. Deploy updates to the testing and staging environments by using the sam deploy command. This is incorrect. This solution will not meet the requirements with the least development effort, because it requires modifying the existing template and adding complexity to the resource definitions for each environment. This can also make it difficult to manage and track changes across different environments.

1: AWS SAM CLI configuration file - AWS Serverless Application Model

2: Configuration file basics - AWS Serverless Application Model

3: Specify a configuration file - AWS Serverless Application Model

Question 7

Question Type: MultipleChoice

A developer at a company needs to create a small application that makes the same API call once each day at a designated time. The company does not have infrastructure in the AWS Cloud yet, but the company wants to implement this functionality on AWS.

Which solution meets these requirements in the MOST operationally efficient manner?

Options:

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

B- Use an Amazon Linux crontab scheduled job that runs on Amazon EC2.

C- Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event.

D- Use an AWS Batch job that is submitted to an AWS Batch job queue.

Answer:

C

Explanation:

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

C) Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event. This is correct. AWS Lambda is a serverless compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging1. Amazon EventBridge is a serverless event bus service that enables you to connect your applications with data from a variety of sources2. EventBridge can create rules that run on a schedule, either at regular intervals or at specific times and dates, and invoke targets such as Lambda functions3. This solution meets the

requirements of creating a small application that makes the same API call once each day at a designated time, without requiring any infrastructure in the AWS Cloud or any operational overhead.

A) Use a Kubernetes cron job that runs on Amazon Elastic Kubernetes Service (Amazon EKS). This is incorrect. Amazon EKS is a fully managed Kubernetes service that allows you to run containerized applications on AWS4. Kubernetes cron jobs are tasks that run periodically on a given schedule5. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to provision and manage an EKS cluster, which would incur additional costs and complexity.

B) Use an Amazon Linux crontab scheduled job that runs on Amazon EC2. This is incorrect. Amazon EC2 is a web service that provides secure, resizable compute capacity in the cloud6. Crontab is a Linux utility that allows you to schedule commands or scripts to run automatically at a specified time or date7. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to provision and manage an EC2 instance, which would incur additional costs and complexity.

D) Use an AWS Batch job that is submitted to an AWS Batch job queue. This is incorrect. AWS Batch enables you to run batch computing workloads on the AWS Cloud8. Batch jobs are units of work that can be submitted to job queues, where they are executed in parallel or sequentially on compute environments9. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to configure and manage an AWS Batch environment, which would incur additional costs and complexity.

1: What is AWS Lambda? - AWS Lambda

2: What is Amazon EventBridge? - Amazon EventBridge

3: Creating an Amazon EventBridge rule that runs on a schedule - Amazon EventBridge

4: What is Amazon EKS? - Amazon EKS

5: CronJob - Kubernetes

6: What is Amazon EC2? - Amazon EC2

7: Crontab in Linux with 20 Useful Examples to Schedule Jobs - Tecmint

8: What is AWS Batch? - AWS Batch

9: Jobs - AWS Batch

Question 8

Question Type: **MultipleChoice**

A developer is migrating an application to Amazon Elastic Kubernetes Service (Amazon EKS). The developer migrates the application to Amazon Elastic Container Registry (Amazon ECR) with an EKS cluster.

As part of the application migration to a new backend, the developer creates a new AWS account. The developer makes configuration changes to the application to point the application to the new AWS account and to use new backend resources. The developer successfully tests the changes within the application by deploying the pipeline.

The Docker image build and the pipeline deployment are successful, but the application is still connecting to the old backend. The developer finds that the application's configuration is still referencing the original EKS cluster and not referencing the new backend

resources.

Which reason can explain why the application is not connecting to the new resources?

Options:

A- The developer did not successfully create the new AWS account.

B- The developer added a new tag to the Docker image.

C- The developer did not update the Docker image tag to a new version.

D- The developer pushed the changes to a new Docker image tag.

Answer:

C

Explanation:

The correct answer is C) The developer did not update the Docker image tag to a new version.

C) The developer did not update the Docker image tag to a new version. This is correct. When deploying an application to Amazon EKS, the developer needs to specify the Docker image tag that contains the application code and configuration. If the developer does not update the Docker image tag to a new version after making changes to the application, the EKS cluster will continue to use the old Docker image tag that references the original backend resources. To fix this issue, the developer should update the Docker image tag to

a new version and redeploy the application to the EKS cluster.

A) The developer did not successfully create the new AWS account. This is incorrect. The creation of a new AWS account is not related to the application's connection to the backend resources. The developer can use any AWS account to host the EKS cluster and the backend resources, as long as they have the proper permissions and configurations.

B) The developer added a new tag to the Docker image. This is incorrect. Adding a new tag to the Docker image is not enough to deploy the changes to the application. The developer also needs to update the Docker image tag in the EKS cluster configuration, so that the EKS cluster can pull and run the new Docker image.

D) The developer pushed the changes to a new Docker image tag. This is incorrect. Pushing the changes to a new Docker image tag is not enough to deploy the changes to the application. The developer also needs to update the Docker image tag in the EKS cluster configuration, so that the EKS cluster can pull and run the new Docker image.

1: Amazon EKS User Guide, "Deploying applications to your Amazon EKS cluster", https://docs.aws.amazon.com/eks/latest/userguide/deploying-applications.html

2: Amazon ECR User Guide, "Pushing an image", https://docs.aws.amazon.com/AmazonECR/latest/userguide/docker-push-ecr-image.html

3: Amazon EKS User Guide, "Updating an Amazon EKS cluster", https://docs.aws.amazon.com/eks/latest/userguide/update-cluster.html

Question 9

Question Type: MultipleChoice

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

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

Options:

A- Create an Amazon EventBridge rule that has a rate expression that will run the rule every 15 minutes. Add the Lambda function as the target of the
EventBridge rule.

B- Create an AWS Systems Manager document that has a script that will invoke the Lambda function on Amazon EC2. Use a Systems Manager Run Command
task to run the shell script every 15 minutes.

C- Create an AWS Step Functions state machine. Configure the state machine to invoke the Lambda function execution role at a specified interval by using a Wait state. Set the interval to 15 minutes.

D- Provision a small Amazon EC2 instance. Set up a cron job that invokes the Lambda function every 15 minutes.

Answer:

A

Explanation:

The best solution for this requirement is option

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

Question 10

Question Type: MultipleChoice

A developer is troubleshooting an application in an integration environment. In the application, an Amazon Simple Queue Service (Amazon SQS) queue consumes messages and then an AWS Lambda function processes the messages. The Lambda function transforms the messages and makes an API call to a third-party service.

There has been an increase in application usage. The third-party API frequently returns an HTTP 429 Too Many Requests error message. The error message prevents a significant number of messages from being processed successfully.

How can the developer resolve this issue?

Options:

A- Increase the SQS event source's batch size setting.

B- Configure provisioned concurrency for the Lambda function based on the third-party API's documented rate limits.

C- Increase the retry attempts and maximum event age in the Lambda function's asynchronous configuration.

D- Configure maximum concurrency on the SQS event source based on the third-party service's documented rate limits.

Answer:

D

Explanation:

Maximum concurrency for SQS as an event source allows customers to control the maximum concurrent invokes by the SQS event source1.When multiple SQS event sources are configured to a function, customers can control the maximum concurrent invokes of individual SQS event source1.

In this scenario, the developer needs to resolve the issue of the third-party API frequently returning an HTTP 429 Too Many Requests error message, which prevents a significant number of messages from being processed successfully. To achieve this, the developer can follow these steps:

Find out the documented rate limits of the third-party API, which specify how many requests can be made in a given time period.

Configure maximum concurrency on the SQS event source based on the rate limits of the third-party API. This will limit the number of concurrent invokes by the SQS event source and prevent exceeding the rate limits of the third-party API.

Test and monitor the application performance and adjust the maximum concurrency value as needed.

By using this solution, the developer can reduce the frequency of HTTP 429 errors and improve the message processing success rate. The developer can also avoid throttling or blocking by the third-party API.

Question 11

Question Type: **MultipleChoice**

A developer is writing an application that will retrieve sensitive data from a third-party system. The application will format the data into a PDF file. The PDF file could be more than 1 MB. The application will encrypt the data to disk by using AWS Key Management Service (AWS KMS). The application will decrypt the file when a user requests to download it. The retrieval and formatting portions of the application are complete.

The developer needs to use the GenerateDataKey API to encrypt the PDF file so that the PDF file can be decrypted later. The developer needs to use an AWS KMS symmetric customer managed key for encryption.

Which solutions will meet these requirements?

Options:

A- Write the encrypted key from the GenerateDataKey API to disk for later use. Use the plaintext key from the GenerateDataKey API and a symmetric encryption
algorithm to encrypt the file.

B- Write the plain text key from the GenerateDataKey API to disk for later use. Use the encrypted key from the GenerateDataKey API and a symmetric encryption
algorithm to encrypt the file.

C- Write the encrypted key from the GenerateDataKey API to disk for later use. Use the plaintext key from the GenerateDataKey API to encrypt the file by using the KMS Encrypt API

D- Write the plain text key from the GenerateDataKey API to disk for later use. Use the encrypted key from the GenerateDataKey API to encrypt the file by using the KMS Encrypt API

Answer:

A

Explanation:

The GenerateDataKey API returns a data key that is encrypted under a symmetric encryption KMS key that you specify, and a plaintext copy of the same data key1.The data key is a random byte string that can be used with any standard encryption algorithm, such as AES or SM42.The plaintext data key can be used to encrypt or decrypt data outside of AWS KMS, while the encrypted data key can be stored with the encrypted data and later decrypted by AWS KMS1.

In this scenario, the developer needs to use the GenerateDataKey API to encrypt the PDF file so that it can be decrypted later. The developer also needs to use an AWS KMS symmetric customer managed key for encryption. To achieve this, the developer can follow these steps:

Call the GenerateDataKey API with the symmetric customer managed key ID and the desired length or specification of the data key. The API will return an encrypted data key and a plaintext data key.

Write the encrypted data key to disk for later use. This will allow the developer to decrypt the data key and the PDF file later by using AWS KMS.

Use the plaintext data key and a symmetric encryption algorithm to encrypt the PDF file. The developer can use any standard encryption library or tool to perform this operation, such as OpenSSL or AWS Encryption SDK.

Discard the plaintext data key from memory as soon as possible after using it. This will prevent unauthorized access or leakage of the data key.

To Get Premium Files for DVA-C02 Visit

https://www.p2pexams.com/products/dva-c02

For More Free Questions Visit

https://www.p2pexams.com/amazon/pdf/dva-c02