

DVA-C02 Dumps

DVA-C02

<https://www.certleader.com/DVA-C02-dumps.html>



NEW QUESTION 1

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

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

Answer: D

Explanation:

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

? AWS Secrets Manager vs. Systems Manager Parameter Store

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

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

NEW QUESTION 2

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

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

Answer: B

Explanation:

Amazon API Gateway allows you to create, deploy, and manage a RESTful API to expose backend HTTP endpoints, AWS Lambda functions, or other AWS 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.

NEW QUESTION 3

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

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

Answer: D

Explanation:

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

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

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

NEW QUESTION 4

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

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

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

Answer: B

Explanation:

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

NEW QUESTION 5

A developer is creating an application that includes an Amazon API Gateway REST API in the us-east-2 Region. The developer wants to use Amazon CloudFront and a custom domain name for the API. The developer has acquired an SSL/TLS certificate for the domain from a third-party provider. How should the developer configure the custom domain for the application?

- A. Import the SSL/TLS certificate into AWS Certificate Manager (ACM) in the same Region as the AP
- B. Create a DNS A record for the custom domain.
- C. Import the SSL/TLS certificate into CloudFront
- D. Create a DNS CNAME record for the custom domain.
- E. Import the SSL/TLS certificate into AWS Certificate Manager (ACM) in the same Region as the AP
- F. Create a DNS CNAME record for the custom domain.
- G. Import the SSL/TLS certificate into AWS Certificate Manager (ACM) in the us-east-1 Region
- H. Create a DNS CNAME record for the custom domain.

Answer: D

Explanation:

Amazon API Gateway is a service that enables developers to create, publish, maintain, monitor, and secure APIs at any scale. Amazon CloudFront is a content delivery network (CDN) service that can improve the performance and security of web applications. The developer can use CloudFront and a custom domain name for the API Gateway REST API. To do so, the developer needs to import the SSL/TLS certificate into AWS Certificate Manager (ACM) in the us-east-1 Region. This is because CloudFront requires certificates from ACM to be in this Region. The developer also needs to create a DNS CNAME record for the custom domain that points to the CloudFront distribution.

References:

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

? [What Is Amazon CloudFront? - Amazon CloudFront]

? [Custom Domain Names for APIs - Amazon API Gateway]

NEW QUESTION 6

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 7

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 8

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

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

Answer: B

Explanation:

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

? AWS X-Ray Annotations

? AWS X-Ray Concepts

NEW QUESTION 9

An Amazon Simple Queue Service (Amazon SQS) queue serves as an event source for an AWS Lambda function. In the SQS queue, each item corresponds to a video file that the Lambda function must convert to a smaller resolution. The Lambda function is timing out on longer video files, but the Lambda function's timeout is already configured to its maximum value.

What should a developer do to avoid the timeouts without additional code changes?

- A. Increase the memory configuration of the Lambda function.
- B. Increase the visibility timeout on the SQS queue.
- C. Increase the instance size of the host that runs the Lambda function.
- D. Use multi-threading for the conversion.

Answer: A

Explanation:

Increasing the memory configuration of the Lambda function will also increase the CPU and network throughput available to the function. This can improve the performance of the video conversion process and reduce the execution time of the function. This solution does not require any code changes or additional resources. It is also recommended to follow the best practices for preventing Lambda function timeouts¹. References

? Troubleshoot Lambda function invocation timeout errors | AWS re:Post

NEW QUESTION 10

An online food company provides an Amazon API Gateway HTTP API to receive orders for partners. The API is integrated with an AWS Lambda function. The Lambda function stores the orders in an Amazon DynamoDB table.

The company expects to onboard additional partners. Some partners require additional Lambda function to receive orders. The company has created an Amazon S3 bucket. The company needs to store all orders and updates in the S3 bucket for future analysis.

How can the developer ensure that orders and updates are stored to Amazon S3 with the LEAST development effort?

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

L. Configure the new Lambda function to write to the S3 bucket as updates come through the topic.

Answer: C

Explanation:

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

NEW QUESTION 10

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

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

Which solution will meet these requirements MOST securely?

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

NEW QUESTION 11

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
- 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 12

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

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

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

Answer: D

NEW QUESTION 16

A company has a web application that is hosted on Amazon EC2 instances. The EC2 instances are configured to stream logs to Amazon CloudWatch Logs. The company needs to receive an Amazon Simple Notification Service (Amazon SNS) notification when the number of application error messages exceeds a defined threshold within a 5-minute period.

Which solution will meet these requirements?

- A. Rewrite the application code to stream application logs to Amazon SNS. Configure an SNS topic to send a notification when the number of errors exceeds the defined threshold within a 5-minute period.
- B. Configure a subscription filter on the CloudWatch Logs log group.
- C. Configure the filter to send an SNS notification when the number of errors exceeds the defined threshold within a 5-minute period.
- D. Install and configure the Amazon Inspector agent on the EC2 instances to monitor for errors. Configure Amazon Inspector to send an SNS notification when the number of errors exceeds the defined threshold within a 5-minute period.
- E. Create a CloudWatch metric filter to match the application error pattern in the log data. Set up a CloudWatch alarm based on the new custom metric.
- F. Configure the alarm to send an SNS notification when the number of errors exceeds the defined threshold within a 5-minute period.

Answer: D

Explanation:

The best solution is to create a CloudWatch metric filter to match the application error pattern in the log data. This will allow you to create a custom metric that tracks the number of errors in your application. You can then set up a CloudWatch alarm based on this metric and configure it to send an SNS notification when the number of errors exceeds a defined threshold within a 5-minute period. This solution does not require any changes to your application code or installing any additional agents on your EC2 instances. It also leverages the existing integration between CloudWatch and SNS for sending notifications. References

? Create Metric Filters - Amazon CloudWatch Logs

? Creating Amazon CloudWatch Alarms - Amazon CloudWatch

? How to send alert based on log message on CloudWatch - Stack Overflow

NEW QUESTION 18

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 19

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

Which solution will meet these requirements?

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

Answer: A

Explanation:

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

References:

? [What Is Amazon ElastiCache? - Amazon ElastiCache]

? [Encryption in Transit - Amazon ElastiCache for Redis]

? [Encryption at Rest - Amazon ElastiCache for Redis]

NEW QUESTION 20

A developer wants to expand an application to run in multiple AWS Regions. The developer wants to copy Amazon Machine Images (AMIs) with the latest changes and create a new application stack in the destination Region. According to company requirements, all AMIs must be encrypted in all Regions. However, not all the AMIs that the company uses are encrypted.

How can the developer expand the application to run in the destination Region while meeting the encryption requirement?

A. Mastered

B. Not Mastered

Answer: A

Explanation:

Amazon Machine Images (AMIs) are encrypted snapshots of EC2 instances that can be used to launch new instances. The developer can create new AMIs from the existing instances and specify encryption parameters. The developer can copy the encrypted AMIs to the destination Region and use them to create a new application stack. The developer can delete the unencrypted AMIs after the encryption process is complete. This solution will meet the encryption requirement and allow the developer to expand the application to run in the destination Region.

References:

? [Amazon Machine Images (AMI) - Amazon Elastic Compute Cloud]

? [Encrypting an Amazon EBS Snapshot - Amazon Elastic Compute Cloud]

? [Copying an AMI - Amazon Elastic Compute Cloud]

NEW QUESTION 24

A company is using an AWS Lambda function to process records from an Amazon Kinesis data stream. The company recently observed slow processing of the records. A developer notices that the iterator age metric for the function is increasing and that the Lambda run duration is constantly above normal.

Which actions should the developer take to increase the processing speed? (Choose two.)

A. Increase the number of shards of the Kinesis data stream.

B. Decrease the timeout of the Lambda function.

C. Increase the memory that is allocated to the Lambda function.

D. Decrease the number of shards of the Kinesis data stream.

E. Increase the timeout of the Lambda function.

Answer: AC

Explanation:

Increasing the number of shards of the Kinesis data stream will increase the throughput and parallelism of the data processing. Increasing the memory that is allocated to the Lambda function will also increase the CPU and network performance of the function, which will reduce the run duration and improve the processing speed. Option B is not correct because decreasing the timeout of the Lambda function will not affect the processing speed, but may cause some records to fail if they exceed the timeout limit. Option D is not correct because decreasing the number of shards of the Kinesis data stream will decrease the throughput and parallelism of the data processing, which will slow down the processing speed. Option E is not correct because increasing the timeout of the Lambda function will not affect the processing speed, but may increase the cost of running the function.

References: [Amazon Kinesis Data Streams Scaling], [AWS Lambda Performance Tuning]

NEW QUESTION 27

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 32

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 a 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 tas
- B. Add a catcher to the GetResource tas
- C. Configure the catcher with an error type of TooManyRequestsExceptio
- 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 TooManyRequestsExceptio
- 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 Configure the catcher with an error type of TooManyRequestsExceptio
- I. 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 35

A developer creates a static website for their department The developer deploys the static assets for the website to an Amazon S3 bucket and serves the assets with Amazon CloudFront The developer uses origin access control (OAC) on the CloudFront distribution to access the S3 bucket The developer notices users can access the root URL and specific pages but cannot access directories without specifying a file name. For example, /products/index.html works, but /products returns an error The developer needs to enable accessing directories without specifying a file name without exposing the S3 bucket publicly. Which solution will meet these requirements'?

- A. Update the CloudFront distribution's settings to index.html as the default root object is set Update the Amazon S3 bucket settings and enable static website hostin
- B. Specify index html as the Index document Update the S3 bucket policy to enable acces
- D. Update the CloudFront distribution's origin to use the S3 website endpoint
- E. Create a CloudFront function that examines the request URL and appends index.html when directories are being accessed Add the function as a viewer request CloudFront function to the CloudFront distribution's behavior.
- F. Create a custom error response on the CloudFront distribution with the HTTP error code set to the HTTP 404 Not Found response code and the response page path to /index.html Set the HTTP response code to the HTTP 200 OK response code

Answer: A

Explanation:

The simplest and most efficient way to enable accessing directories without specifying a file name is to update the CloudFront distribution's settings to index.html as the default root object. This will instruct CloudFront to return the index.html object when a user requests the root URL or a directory URL for the distribution. This solution does not require enabling static website hosting on the S3 bucket, creating a CloudFront function, or creating a custom error response. References

- ? Specifying a default root object
- ? cloudfront-default-root-object-configured
- ? How to setup CloudFront default root object?
- ? Ensure a default root object is configured for AWS Cloudfront ...

NEW QUESTION 38

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 instanc
- B. Store the unique identifier for each request in a database tabl
- C. Modify the Lambda function to check the table for the identifier before processing the request.
- D. Create an Amazon DynamoDB tabl
- E. Store the unique identifier for each request in the tabl
- F. Modify the Lambda function to check the table for the identifier before processing the request.
- G. Create an Amazon DynamoDB tabl
- H. Store the unique identifier for each request in the tabl
- 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 instanc
- K. Store the unique identifier for each request in the cach
- 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 43

A company is building a web application on AWS. When a customer sends a request, the application will generate reports and then make the reports available to the customer within one hour. Reports should be accessible to the customer for 8 hours. Some reports are larger than 1 MB. Each report is unique to the customer. The application should delete all reports that are older than 2 days. Which solution will meet these requirements with the LEAST operational overhead?

- A. Generate the reports and then store the reports as Amazon DynamoDB items that have a specified TT
- B. Generate a URL that retrieves the reports from DynamoD
- C. Provide the URL to customers through the web application.
- D. Generate the reports and then store the reports in an Amazon S3 bucket that uses server-side encryptio
- E. Attach the reports to an Amazon Simple Notification Service (Amazon SNS) messag
- F. Subscribe the customer to email notifications from Amazon SNS.
- G. Generate the reports and then store the reports in an Amazon S3 bucket that uses server-side encryptio
- H. Generate a presigned URL that contains an expiration date Provide the URL to customers through the web applicatio
- I. Add S3 Lifecycle configuration rules to the S3 bucket to delete old reports.
- J. Generate the reports and then store the reports in an Amazon RDS database with a date stam
- K. Generate an URL that retrieves the reports from the RDS databas
- L. Provide the URL to customers through the web applicatio
- M. Schedule an hourly AWS Lambda function to delete database records that have expired date stamps.

Answer: C

Explanation:

This solution will meet the requirements with the least operational overhead because it uses Amazon S3 as a scalable, secure, and durable storage service for the reports. The presigned URL will allow customers to access their reports for a limited time (8 hours) without requiring additional authentication. The S3 Lifecycle configuration rules will automatically delete the reports that are older than 2 days, reducing storage costs and complying with the data retention policy. Option A is not optimal because it will incur additional costs and complexity to store the reports as DynamoDB items, which have a size limit of 400 KB. Option B is not optimal because it will not provide customers with access to their reports within one hour, as Amazon SNS email delivery is not guaranteed. Option D is not optimal because it will require more operational overhead to manage an RDS database and a Lambda function for storing and deleting the reports. References: Amazon S3 Presigned URLs, Amazon S3 Lifecycle

NEW QUESTION 44

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 49

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

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

Answer: B

Explanation:

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

NEW QUESTION 50

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 DynamoD
- C. Set up a DynamoDB Accelerator (DAX) cluster.
- D. Configure the Amazon RDS instances to use Multi-AZ deployment with one standby instanc
- 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 52

A developer is working on a serverless application that needs to process any changes to an Amazon DynamoDB table with an AWS Lambda function.

How should the developer configure the Lambda function to detect changes to the DynamoDB table?

- A. Create an Amazon Kinesis data stream, and attach it to the DynamoDB tabl
- B. Create a trigger to connect the data stream to the Lambda function.
- C. Create an Amazon EventBridge rule to invoke the Lambda function on a regular schedul
- D. Connect to the DynamoDB table from the Lambda function to detect changes.
- E. Enable DynamoDB Streams on the tabl
- F. Create a trigger to connect the DynamoDB stream to the Lambda function.
- G. Create an Amazon Kinesis Data Firehose delivery stream, and attach it to the DynamoDB tabl
- H. Configure the delivery stream destination as the Lambda function.

Answer: C

Explanation:

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and consistent performance with seamless scalability. DynamoDB Streams is a feature that captures data modification events in DynamoDB tables. The developer can enable DynamoDB Streams on the table and create a trigger to connect the DynamoDB stream to the Lambda function. This solution will enable the Lambda function to detect changes to the DynamoDB table in near real time.

References:

? [Amazon DynamoDB]

? [DynamoDB Streams - Amazon DynamoDB]

? [Using AWS Lambda with Amazon DynamoDB - AWS Lambda]

NEW QUESTION 56

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?

- A. Write the encrypted key from the GenerateDataKey API to disk for later us
- B. Use the plaintext key from the GenerateDataKey API and a symmetric encryption algorithm to encrypt the file.
- C. Write the plain text key from the GenerateDataKey API to disk for later us
- D. Use the encrypted key from the GenerateDataKey API and a symmetric encryption algorithm to encrypt the file.
- E. Write the encrypted key from the GenerateDataKey API to disk for later us
- F. Use the plaintext key from the GenerateDataKey API to encrypt the file by using the KMS Encrypt API
- G. Write the plain text key from the GenerateDataKey API to disk for later us
- H. 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:

NEW QUESTION 61

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: BD

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 65

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

Which solution will meet these requirements?

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

? Using AWS Lambda destinations

? Asynchronous invocation - AWS Lambda

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

? AWS Lambda Destinations: A Complete Guide | Dashbird

NEW QUESTION 67

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

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

- A. Define a maintenance window for the Lambda functions to ensure that the functions get updated copies of the dependencies.
- B. Upgrade the Lambda functions to the most recent runtime version.
- C. Define a Lambda layer that contains all of the shared dependencies.
- D. Use an AWS CodeCommit repository to host the dependencies in a centralized location.

Answer: C

Explanation:

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

Reference: [AWS Lambda layers]

NEW QUESTION 71

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 74

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

Every record includes the following

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

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

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

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

Answer: A

Explanation:

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

Reference: [Global Secondary Indexes], [Querying]

NEW QUESTION 78

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

Which solution will meet these requirements MOST cost-effectively?

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

Answer: A

Explanation:

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

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

? AWS::StepFunctions::StateMachine - AWS CloudFormation

? Call API Gateway with Step Functions - AWS Step Functions

? amazon-web-services aws-api-gateway terraform aws-step-functions

NEW QUESTION 81

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

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

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

Answer: B

Explanation:

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

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

References:

? [Amazon DynamoDB]

? [Amazon Simple Storage Service (S3)]

NEW QUESTION 84

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 86

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 91

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 function
- F. Update the Lambda function code to download the log file, append the log entries, and upload the modified log file to Amazon EBS
- 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 95

A company has built an AWS Lambda function to convert large image files into output files that can be used in a third-party viewer application. The company recently added a new module to the function to improve the output of the generated files. However, the new module has increased the bundle size and has increased the time that is needed to deploy changes to the function code. How can a developer increase the speed of the Lambda function deployment?

- A. Use AWS CodeDeploy to deploy the function code.
- B. Use Lambda layers to package and load dependencies.
- C. Increase the memory size of the function.
- D. Use Amazon S3 to host the function dependencies.

Answer: B

Explanation:

Using Lambda layers is a way to reduce the size of the deployment package and speed up the deployment process. Lambda layers are reusable components that can contain libraries, custom runtimes, or other dependencies. By using layers, the developer can separate the core function logic from the dependencies, and avoid uploading them every time the function code changes. Layers can also be shared across multiple functions or accounts, which can improve consistency and maintainability. References

? Working with AWS Lambda layers

? AWS Lambda Layers Best Practices

? Best practices for working with AWS Lambda functions

NEW QUESTION 99

A company is developing an ecommerce application that uses Amazon API Gateway APIs. The application uses AWS Lambda as a backend. The company needs to test the code in a dedicated, monitored test environment before the company releases the code to the production environment. When 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 environment
- 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 104

A developer is working on an ecommerce website. The developer wants to review server logs without logging in to each of the application servers individually. The website runs on multiple Amazon EC2 instances, is written in Python, and needs to be highly available. How can the developer update the application to meet these requirements with MINIMUM changes?

- A. Rewrite the application to be cloud native and to run on AWS Lambda, where the logs can be reviewed in Amazon CloudWatch
- B. Set up centralized logging by using Amazon OpenSearch Service, Logstash, and OpenSearch Dashboards
- C. Scale down the application to one larger EC2 instance where only one instance is recording logs
- D. Install the unified Amazon CloudWatch agent on the EC2 instances. Configure the agent to push the application logs to CloudWatch

Answer: D

Explanation:

The unified Amazon CloudWatch agent can collect both system metrics and log files from Amazon EC2 instances and on-premises servers. By installing and configuring the agent on the EC2 instances, the developer can easily access and analyze the application logs in CloudWatch without logging in to each server individually. This option requires minimum changes to the existing application and does not affect its availability or scalability. References

? Using the CloudWatch Agent

? Collecting Metrics and Logs from Amazon EC2 Instances and On-Premises Servers with the CloudWatch Agent

NEW QUESTION 108

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 109

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

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

Answer: D

Explanation:

FunctionName: MyLambdaFunction Code:

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

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

Variables:

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

NEW QUESTION 112

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

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

- A. Inspect the frontend logs for API failure
- B. Call the POST API manually by using the requests from the log file.
- C. Create and inspect the Lambda dead-letter queue
- D. Troubleshoot the failed function
- E. Reprocess the events.
- F. Inspect the Lambda logs in Amazon CloudWatch for possible error
- G. Fix the errors.
- H. Make sure that caching is disabled for the POST API in API Gateway.

Answer: B

Explanation:

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

Reference: Asynchronous invocation

NEW QUESTION 117

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

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

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

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

Answer: BE

Explanation:

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

? Set up AWS CodeDeploy to deploy the most recent version of the application at

runtime. This will ensure that the application code is always up to date and does not depend on the AMI.

? Remove any commands that perform operating system patching from the

UserData script. This will reduce the time that the UserData script takes to run and speed up the instance launch process.

References:

? [What Is AWS CloudFormation? - AWS CloudFormation]

? [What Is AWS CodeDeploy? - AWS CodeDeploy]

? [Running Commands on Your Linux Instance at Launch - Amazon Elastic Compute Cloud]

NEW QUESTION 119

A developer wants to store information about movies. Each movie has a title, release year, and genre. The movie information also can include additional properties about the cast and production crew. This additional information is inconsistent across movies. For example, one movie might have an assistant director, and another movie might have an animal trainer.

The developer needs to implement a solution to support the following use cases:

For a given title and release year, get all details about the movie that has that title and release year.

For a given title, get all details about all movies that have that title. For a given genre, get all details about all movies in that genre. Which data store configuration will meet these requirements?

- A. Create an Amazon DynamoDB tabl
- B. Configure the table with a primary key that consists of the title as the partition key and the release year as the sort ke
- C. Create a global secondary index that uses the genre as the partition key and the title as the sort key.
- D. Create an Amazon DynamoDB tabl
- E. Configure the table with a primary key that consists of the genre as the partition key and the release year as the sort ke
- F. Create a global secondary index that uses the title as the partition key.
- G. On an Amazon RDS DB instance, create a table that contains columns for title, release year, and genr
- H. Configure the title as the primary key.
- I. On an Amazon RDS DB instance, create a table where the primary key is the title and all other data is encoded into JSON format as one additional column.

Answer: A

Explanation:

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and consistent performance with seamless scalability. The developer can create a DynamoDB table and configure the table with a primary key that consists of the title as the partition key and the release year as the sort key. This will enable querying for a given title and release year efficiently. The developer can also create a global secondary index that uses the genre as the partition key and the title as the sort key. This will enable querying for a given genre efficiently. The developer can store additional properties about the cast and production crew as attributes in the DynamoDB table. These attributes can have different data types and structures, and they do not need to be consistent across items.

References:

- ? [Amazon DynamoDB]
- ? [Working with Queries - Amazon DynamoDB]
- ? [Working with Global Secondary Indexes - Amazon DynamoDB]

NEW QUESTION 122

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

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

GetSecretValue Secrets Manager API operation requests.

Answer: B

Explanation:

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

NEW QUESTION 125

An application that is deployed to Amazon EC2 is using Amazon DynamoDB. The application calls the DynamoDB REST API. Periodically the application receives a `ProvisionedThroughputExceededException` error when the application writes to a DynamoDB table.

Which solutions will mitigate this error MOST cost-effectively? (Select TWO)

- A. Modify the application code to perform exponential back off when the error is received.
- B. Modify the application to use the AWS SDKs for DynamoDB.
- C. Increase the read and write throughput of the DynamoDB table.
- D. Create a DynamoDB Accelerator (DAX) cluster for the DynamoDB table.
- E. Create a second DynamoDB table. Distribute the reads and writes between the two tables.

Answer: AB

Explanation:

These solutions will mitigate the error most cost-effectively because they do not require increasing the provisioned throughput of the DynamoDB table or creating additional resources. Exponential backoff is a retry strategy that increases the waiting time between retries to reduce the number of requests sent to DynamoDB. The AWS SDKs for DynamoDB implement exponential backoff by default and also provide other features such as automatic pagination and encryption. Increasing the read and write throughput of the DynamoDB table, creating a DynamoDB Accelerator (DAX) cluster, or creating a second DynamoDB table will incur additional costs and complexity.

Reference: [Error Retries and Exponential Backoff in AWS], [Using the AWS SDKs with DynamoDB]

NEW QUESTION 129

Users are reporting errors in an application. The application consists of several micro services that are deployed on Amazon Elastic Container Service (Amazon ECS) with AWS Fargate.

When combination of steps should a developer take to fix the errors? (Select TWO)

- A. Deploy AWS X-Ray as a sidecar container to the micro service
- B. Update the task role policy to allow access to the X-Ray API.
- C. Deploy AWS X-Ray as a daemon set to the Fargate cluster
- D. Update the service role

policy to allow access to the X-Ray API.

- E. Instrument the application by using the AWS X-Ray SDK

- F. Update the application to use the PutXrayTrace API call to communicate with the X-Ray API.
- G. Instrument the application by using the AWS X-Ray SD
- H. Update the application to communicate with the X-Ray daemon.
- I. Instrument the ECS task to send the stout and spider- output to Amazon CloudWatch Log
- J. Update the task role policy to allow the cloudwatch Putlogs action.

Answer: AE

Explanation:

The combination of steps that the developer should take to fix the errors is to deploy AWS X-Ray as a sidecar container to the microservices and instrument the ECS task to send the stdout and stderr output to Amazon CloudWatch Logs. This way, the developer can use AWS X-Ray to analyze and debug the performance of the microservices and identify any issues or bottlenecks. The developer can also use CloudWatch Logs to monitor and troubleshoot the logs from the ECS task and detect any errors or exceptions. The other options either involve using AWS X-Ray as a daemon set, which is not supported by Fargate, or using the PutTraceSegments API call, which is not necessary when using a sidecar container.

Reference: Using AWS X-Ray with Amazon ECS

NEW QUESTION 132

When a developer tries to run an AWS Code Build project, it raises an error because the length of all environment variables exceeds the limit for the combined maximum of characters. What is the recommended solution?

- A. Add the export LC- _ALL" on _ US, tuft" command to the pre _ build section to ensure POSIX Localization.
- B. Use Amazon Cognate to store key-value pairs for large numbers of environment variables
- C. Update the settings for the build project to use an Amazon S3 bucket for large numbers of environment variables
- D. Use AWS Systems Manager Parameter Store to store large numbers of environment variables

Answer: D

Explanation:

This solution allows the developer to overcome the limit for the combined maximum of characters for environment variables in AWS CodeBuild. AWS Systems Manager Parameter Store provides secure, hierarchical storage for configuration data management and secrets management. The developer can store large numbers of environment variables as parameters in Parameter Store and reference them in the buildspec file using parameter references. Adding export LC_ALL="en_US.utf8" command to the pre_build section will not affect the environment variables limit. Using Amazon Cognito or an Amazon S3 bucket to store key-value pairs for environment variables will require additional configuration and integration.

Reference: [Build Specification Reference for AWS CodeBuild], [What Is AWS Systems Manager Parameter Store?]

NEW QUESTION 134

A developer is migrating some features from a legacy monolithic application to use AWS Lambda functions instead. The application currently stores data in an Amazon Aurora DB cluster that runs in private subnets in a VPC. The AWS account has one VPC deployed. The Lambda functions and the DB cluster are deployed in the same AWS Region in the same AWS account.

The developer needs to ensure that the Lambda functions can securely access the DB cluster without crossing the public internet.

Which solution will meet these requirements?

- A. Configure the DB cluster's public access setting to Yes.
- B. Configure an Amazon RDS database proxy for the Lambda functions.
- C. Configure a NAT gateway and a security group for the Lambda functions.
- D. Configure the VPC, subnets, and a security group for the Lambda functions.

Answer: D

Explanation:

This solution will meet the requirements by allowing the Lambda functions to access the DB cluster securely within the same VPC without crossing the public internet. The developer can configure a VPC endpoint for RDS in a private subnet and assign it to the Lambda functions. The developer can also configure a security group for the Lambda functions that allows inbound traffic from the DB cluster on port 3306 (MySQL). Option A is not optimal because it will expose the DB cluster to public access, which may compromise its security and data integrity. Option B is not optimal because it will introduce additional latency and complexity to use an RDS database proxy for accessing the DB cluster from Lambda functions within the same VPC. Option C is not optimal because it will require

additional costs and configuration to use a NAT gateway for accessing resources in private subnets from Lambda functions.
References: [Configuring a Lambda Function to Access Resources in a VPC]

NEW QUESTION 138

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 142

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

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

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

Answer: B

Explanation:

The most secure way to grant the Lambda function access to the S3 bucket and the DynamoDB table is to create an IAM role for the Lambda function and attach the existing IAM policy to the role. This way, you can use the principle of least privilege and avoid exposing any credentials in your function code or environment variables. You can also leverage the temporary security credentials that AWS provides to the Lambda function when it assumes the role. This solution follows the best practices for working with AWS Lambda functions¹ and designing and architecting with DynamoDB².
References:
? Best practices for working with AWS Lambda functions
? Best practices for designing and architecting with DynamoDB

NEW QUESTION 146

An application runs on multiple EC2 instances behind an ELB.

Where is the session data best written so that it can be served reliably across multiple requests?

- A. Write data to Amazon ElastiCache
- B. Write data to Amazon Elastic Block Store
- C. Write data to Amazon EC2 instance Store
- D. Write data to the root filesystem

Answer: A

Explanation:

The solution that will meet the requirements is to write data to Amazon ElastiCache. This way, the application can write session data to a fast, scalable, and reliable in-memory data store that can be served reliably across multiple requests. The other options either involve writing data to persistent storage, which is slower and more expensive than in-memory storage, or writing data to the root filesystem, which is not shared among multiple EC2 instances.
Reference: Using ElastiCache for session management

NEW QUESTION 149

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

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

Answer: C

Explanation:

This solution will meet the requirements by using AWS X-Ray to analyze and debug the performance issues with the distributed Lambda applications. AWS X-Ray is a service that collects data about requests that the applications serve, and provides tools to view, filter, and gain insights into that data. The developer can use AWS X-Ray to identify the root cause of the performance issues by examining the segments and errors that show the details of each request and the components

that make up the applications. Option A is not optimal because it will use logging statements and Amazon CloudWatch, which may not provide enough information or visibility into the distributed applications. Option B is not

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

NEW QUESTION 152

A company has multiple Amazon VPC endpoints in the same VPC. A developer needs configure an Amazon S3 bucket policy so users can access an S3 bucket only by using these VPC endpoints.

Which solution will meet these requirements?

- A. Create multiple S3 bucket policies by using each VPC endpoint ID that have the aws SourceVpce value in the StringNotEquals condition.
- B. Create a single S3 bucket policy that has the aws SourceVpc value and in the StingNotEquals condition to use VPC ID.
- C. Create a single S3 bucket policy that the multiple aws SourceVpce value and in the SringNotEquals condton to use vpce.
- D. Create a single S3 bucket policy that has multiple aws sourceVpce value in the StingNotEquale conditio
- E. Repeat for all the VPC endpoint IDs.

Answer: D

Explanation:

This solution will meet the requirements by creating a single S3 bucket policy that denies access to the S3 bucket unless the request comes from one of the specified VPC endpoints. The aws:SourceVpce condition key is used to match the ID of the VPC endpoint that is used to access the S3 bucket. The

allowed.

StringNotEquals condition operator is used to negate the condition so that only requests from the listed VPC endpoints are allowed. Option A is not optimal because it will create multiple S3 bucket policies, which is not possible as only one bucket policy can be attached to an S3 bucket. Option B is not optimal because it will use the aws:SourceVpc condition key, which matches the ID of the VPC that is used to access the S3 bucket, not the VPC endpoint. Option C is not optimal because it will use the StringNotEquals condition operator with a single value, which will deny access to the S3 bucket from all VPC endpoints except one.

References: Using Amazon S3 Bucket Policies and User Policies, AWS Global Condition Context Keys

NEW QUESTION 156

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 157

A company runs an application on AWS The application stores data in an Amazon DynamoDB table Some queries are taking a long time to run These slow queries involve an attribute that is not the table's partition key or sort key

The amount of data that the application stores in the DynamoDB table is expected to increase significantly. A developer must increase the performance of the queries.

Which solution will meet these requirements'?

- A. Increase the page size for each request by setting the Limit parameter to be higher than the default value Configure the application to retry any request that exceeds the provisioned throughput.
- B. Create a global secondary index (GSI). Set query attribute to be the partition key of the index
- C. Perform a parallel scan operation by issuing individual scan requests in the parameters specify the segment for the scan requests and the total number of segments for the parallel scan.
- D. Turn on read capacity auto scaling for the DynamoDB tabl
- E. Increase the maximum read capacity units (RCUs).

Answer: B

Explanation:

Creating a global secondary index (GSI) is the best solution to improve the performance of the queries that involve an attribute that is not the table's partition key or sort key. A GSI allows you to define an alternate key for your table and query the data using that key. This way, you can avoid scanning the entire table and reduce the latency and cost of your queries. You should also follow the best practices for designing and using GSIs in DynamoDB¹². References

? Working with Global Secondary Indexes - Amazon DynamoDB

? DynamoDB Performance & Latency - Everything You Need To Know

NEW QUESTION 160

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 paramete
- 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 163

A developer accesses AWS CodeCommit over SSH. The SSH keys configured to access AWS CodeCommit are tied to a user with the following permissions:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:BatchGetRepositories",
        "codecommit:Get*",
        "codecommit:List*",
        "codecommit:GitPull"
      ],
      "Resource": "*"
    }
  ]
}
```

The developer needs to create/delete branches

Which specific IAM permissions need to be added based on the principle of least privilege?

- A. "codecommit:CreateBranch"
"codecommit>DeleteBranch"
- B. "codecommit:Put*"
- C. "codecommit:Update*"
- D. "codecommit:*"

- A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

Explanation:

This solution allows the developer to create and delete branches in AWS CodeCommit by granting the codecommit:CreateBranch and codecommit>DeleteBranch permissions. These are the minimum permissions required for this task, following the principle of least privilege. Option B grants too many permissions, such as codecommit:Put*, which allows the developer to create, update, or delete any resource in CodeCommit. Option C grants too few permissions, such as codecommit:Update*, which does not allow the developer to create or delete branches. Option D grants all permissions, such as codecommit:*, which is not secure or recommended.

Reference: [AWS CodeCommit Permissions Reference], [Create a Branch (AWS CLI)]

NEW QUESTION 164

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 167

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 168

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 Deletion Policy attribute with the Retain value to the database resource.
- B. Update the CloudFormation stack policy to prevent updates to the database.
- C. Modify the database to use a Multi-AZ deployment.
- D. 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 173

A developer is creating a service that uses an Amazon S3 bucket for image uploads. The service will use an AWS Lambda function to create a thumbnail of each image. Each time an image is uploaded, the service needs to send an email notification and create the thumbnail. The developer needs to configure the image processing and email notifications setup.

Which solution will meet these requirements?

- A. Create an Amazon Simple Notification Service (Amazon SNS) topic. Configure S3 event notifications with a destination of the SNS topic. Subscribe the Lambda function to the SNS topic. Create an email notification subscription to the SNS topic.

- B. Create an Amazon Simple Notification Service (Amazon SNS) topic
C. Configure S3 event notifications with a destination of the SNS topic
D. Subscribe the Lambda function to the SNS topic
E. Create an Amazon Simple Queue Service (Amazon SQS) queue Subscribe the SQS queue to the SNS topic Create an email notification subscription to the SQS queue.
F. Create an Amazon Simple Queue Service (Amazon SQS) queue Configure S3 event notifications with a destination of the SQS queue Subscribe the Lambda function to the SQS queue Create an email notification subscription to the SQS queue.
G. Create an Amazon Simple Queue Service (Amazon SQS) queue
H. Send S3 event notifications to Amazon EventBridge
I. Create an EventBridge rule that runs the Lambda function when images are uploaded to the S3 bucket Create an EventBridge rule that sends notifications to the SQS queue Create an email notification subscription to the SQS queue

Answer: A

Explanation:

This solution will allow the developer to receive notifications for each image uploaded to the S3 bucket, and also create a thumbnail using the Lambda function. The SNS topic will serve as a trigger for both the Lambda function and the email notification subscription. When an image is uploaded, S3 will send a notification to the SNS topic, which will trigger the Lambda function to create the thumbnail and also send an email notification to the specified email address.

NEW QUESTION 175

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your DVA-C02 Exam with Our Prep Materials Via below:

<https://www.certleader.com/DVA-C02-dumps.html>