

Amazon

Exam Questions DVA-C02

DVA-C02



NEW QUESTION 1

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

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

Answer: B

Explanation:

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

NEW QUESTION 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
- 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 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 4

A company is offering APIs as a service over the internet to provide unauthenticated read access to statistical information that is updated daily. The company uses Amazon API Gateway and AWS Lambda to develop the APIs. The service has become popular, and the company wants to enhance the responsiveness of the APIs.

Which action can help the company achieve this goal?

- A. Enable API caching in API Gateway.
- B. Configure API Gateway to use an interface VPC endpoint.
- C. Enable cross-origin resource sharing (CORS) for the APIs.
- D. Configure usage plans and API keys in API Gateway.

Answer: A

Explanation:

Amazon API Gateway is a service that enables developers to create, publish, maintain, monitor, and secure APIs at any scale. The developer can enable API caching in API Gateway to cache responses from the backend integration point for a specified time-to-live (TTL) period. This can improve the responsiveness of the APIs by reducing the number

of calls made to the backend service. References:

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

? [Enable API Caching to Enhance Responsiveness - Amazon API Gateway]

NEW QUESTION 5

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

The developer needs to implement a solution to allow unauthenticated access to the login page. The solution also must keep all private content secure.

Which solution will meet these requirements?

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

Answer: A

Explanation:

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

Reference: Restricting Access to Amazon S3 Content by Using an Origin Access Identity

NEW QUESTION 6

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 7

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

- A. Implement Kinesis Data Firehose data transformation as an AWS Lambda function
- B. Configure the function to remove the customer identifier
- C. Set an Amazon S3 bucket as the destination of the delivery stream.
- D. Launch an Amazon EC2 instance
- E. Set the EC2 instance as the destination of the delivery stream
- F. Run an application on the EC2 instance to remove the customer identifier
- G. Store the transformed data in an Amazon S3 bucket.
- H. Create an Amazon OpenSearch Service instance
- I. Set the OpenSearch Service instance as the destination of the delivery stream
- J. Use search and replace to remove the customer identifier
- K. Export the data to an Amazon S3 bucket.
- L. Create an AWS Step Functions workflow to remove the customer identifier
- M. As the last step in the workflow, store the transformed data in an Amazon S3 bucket
- N. Set the workflow as the destination of the delivery stream.

Answer: A

Explanation:

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

NEW QUESTION 8

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

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

Answer: D

Explanation:

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

NEW QUESTION 9

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

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

Answer: B

Explanation:

This solution allows the Lambda function to be invoked by the DynamoDB stream whenever updates are made to items in the DynamoDB table. Event source mapping is a feature of Lambda that enables a function to be triggered by an event source, such as a DynamoDB stream, an Amazon Kinesis stream, or an Amazon Simple Queue Service (SQS) queue. The developer can configure event source mapping for the Lambda function using the AWS Management Console, the AWS CLI, or the AWS SDKs. Changing the StreamViewType parameter value to NEW_AND_OLD_IMAGES for the DynamoDB table will not affect the

invocation of the Lambda function, but only change the information that is written to the stream record. Mapping an Amazon Simple Notification Service (Amazon SNS) topic to the DynamoDB stream will not invoke the Lambda function directly, but require an additional subscription from the Lambda function to the SNS topic. Increasing the maximum runtime (timeout) setting of the Lambda function will not affect the invocation of the Lambda function, but only change how long the function can run before it is terminated.

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

NEW QUESTION 10

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

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

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

Answer: C

Explanation:

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

NEW QUESTION 10

A developer is creating a simple proof-of-concept demo by using AWS CloudFormation and AWS Lambda functions. The demo will use a CloudFormation template to deploy an existing Lambda function. The Lambda function uses deployment packages and dependencies stored in Amazon S3. The developer defined an AWS Lambda Function resource in a CloudFormation template. The developer needs to add the S3 bucket to the CloudFormation template. What should the developer do to meet these requirements with the LEAST development effort?

- A. Add the function code in the CloudFormation template inline as the code property
- B. Add the function code in the CloudFormation template as the ZipFile property.
- C. Find the S3 key for the Lambda function. Add the S3 key as the ZipFile property in the CloudFormation template.
- D. Add the relevant key and bucket to the S3Bucket and S3Key properties in the CloudFormation template

Answer: D

Explanation:

The easiest way to add the S3 bucket to the CloudFormation template is to use the S3Bucket and S3Key properties of the AWS::Lambda::Function resource. These properties specify the name of the S3 bucket and the location of the .zip file that contains the function code and dependencies. This way, the developer does not need to modify the function code or upload it to a different location. The other options are either not feasible or not efficient. The code property can only be used for inline code, not for code stored in S3. The ZipFile property can only be used for code that is less than 4096 bytes, not for code that has dependencies. Finding the S3 key for the Lambda function and adding it as the ZipFile property would not work, as the ZipFile property expects a base64-encoded .zip file, not an S3 location. References
? AWS::Lambda::Function - AWS CloudFormation
? Deploying Lambda functions as .zip file archives
? AWS Lambda Function Code - AWS CloudFormation

NEW QUESTION 12

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 14

An online sales company is developing a serverless application that runs on AWS. The application uses an AWS Lambda function that calculates order success rates and stores the data in an Amazon DynamoDB table. A developer wants an efficient way to invoke the Lambda function every 15 minutes. Which solution will meet this requirement with the LEAST development effort?

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

Answer: A

Explanation:

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

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 25

A developer is creating an AWS CloudFormation template to deploy Amazon EC2 instances across multiple AWS accounts. The developer must choose the EC2 instances from a list of approved instance types.

How can the developer incorporate the list of approved instance types in the CloudFormation template?

- A. Create a separate CloudFormation template for each EC2 instance type in the list.
- B. In the Resources section of the CloudFormation template, create resources for each EC2 instance type in the list.
- C. In the CloudFormation template, create a separate parameter for each EC2 instance type in the list.
- D. In the CloudFormation template, create a parameter with the list of EC2 instance types as AllowedValues.

Answer: D

Explanation:

In the CloudFormation template, the developer should create a parameter with the list of approved EC2 instance types as AllowedValues. This way, users can select the instance type they want to use when launching the CloudFormation stack, but only from the approved list.

NEW QUESTION 28

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 hosting.
- B. Specify `index.html` as the Index document. Update the S3 bucket policy to enable access.
- C. Update the CloudFront distribution's origin to use the S3 website endpoint.
- 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 33

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

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

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

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

DynamoDB

Answer: B

Explanation:

Amazon Cognito user pools is a service that provides a secure user directory that scales to hundreds of millions of users. The developer can use Amazon Cognito

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

References:

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

NEW QUESTION 38

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 42

A developer is testing an application that invokes an AWS Lambda function asynchronously. During the testing phase the Lambda function fails to process after two retries.

How can the developer troubleshoot the failure?

- A. Configure AWS CloudTrail logging to investigate the invocation failures.
- B. Configure Dead Letter Queues by sending events to Amazon SQS for investigation.
- C. Configure Amazon Simple Workflow Service to process any direct unprocessed events.
- D. Configure AWS Config to process any direct unprocessed events.

Answer: B

Explanation:

This solution allows the developer to troubleshoot the failure by capturing unprocessed events in a queue for further analysis. Dead Letter Queues (DLQs) are queues that store messages that could not be processed by a service, such as Lambda, for various reasons, such as configuration errors, throttling limits, or permissions issues. The developer can configure DLQs for Lambda functions by sending events to either an Amazon Simple Queue Service (SQS) queue or an Amazon Simple Notification Service (SNS) topic. The developer can then inspect the messages in the queue or topic to identify and fix the root cause of the failure. Configuring AWS CloudTrail logging will not capture invocation failures for asynchronous Lambda invocations, but only record API calls made by or on behalf of Lambda. Configuring Amazon Simple Workflow Service (SWF) or AWS Config will not process any direct unprocessed events, but require additional integration and configuration.

Reference: [Using AWS Lambda with DLQs], [Asynchronous invocation]

NEW QUESTION 45

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 46

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

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

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

D. Use AWS CodePipeline for deployments and rollbacks.

Answer: B

Explanation:

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

NEW QUESTION 51

A company needs to set up secure database credentials for all its AWS Cloud resources. The company's resources include Amazon RDS DB instances Amazon DocumentDB clusters and Amazon Aurora DB instances. The company's security policy mandates that database credentials be encrypted at rest and rotated at a regular interval.

Which solution will meet these requirements MOST securely?

- A. Set up IAM database authentication for token-based access
- B. Generate user tokens to provide centralized access to RDS DB instance
- C. Amazon DocumentDB clusters and Aurora DB instances.
- D. Create parameters for the database credentials in AWS Systems Manager Parameter Store Set the Type parameter to Secure String
- E. Set up automatic rotation on the parameters.
- F. Store the database access credentials as an encrypted Amazon S3 object in an S3 bucket Block all public access on the S3 bucket automatic rotation on the encryption key.
- G. Use S3 server-side encryption to set up
- H. Create an AWS Lambda function by using the SecretsManagerRotationTemplate template in the AWS Secrets Manager console
- I. Create secrets for the database credentials in Secrets Manager Set up secrets rotation on a schedule.

Answer: D

Explanation:

This solution will meet the requirements by using AWS Secrets Manager, which is a service that helps protect secrets such as database credentials by encrypting them with AWS Key Management Service (AWS KMS) and enabling automatic rotation of secrets. The developer can create an AWS Lambda function by using the SecretsManagerRotationTemplate template in the AWS Secrets Manager console, which provides a sample code for rotating secrets for RDS DB instances, Amazon DocumentDB clusters, and Amazon Aurora DB instances. The developer can also create secrets for the database credentials in Secrets Manager, which encrypts them at rest and provides secure access to them. The developer can set up secrets rotation on a schedule, which changes the database credentials periodically according to a specified interval or event. Option A is not optimal because it will set up IAM database authentication for token-based access, which may not be compatible with all database engines and may require additional configuration and management of IAM roles or users. Option B is not optimal because it will create parameters for the database credentials in AWS Systems Manager Parameter Store, which does not support automatic rotation of secrets. Option C is not optimal because it will store the database access credentials as an encrypted Amazon S3 object in an S3 bucket, which may introduce additional costs and complexity for accessing and securing the data.

References: [AWS Secrets Manager], [Rotating Your AWS Secrets Manager Secrets]

NEW QUESTION 56

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 61

A company has deployed infrastructure on AWS. A development team wants to create an AWS Lambda function that will retrieve data from an Amazon Aurora database. The Amazon Aurora database is in a private subnet in company's VPC. The VPC is named VPC1. The data is relational in nature. The Lambda function needs to access the data

securely.

Which solution will meet these requirements?

- A. Create the Lambda function
- B. Configure VPC1 access for the function
- C. Attach a security group named SG1 to both the Lambda function and the database
- D. Configure the security group inbound and outbound rules to allow TCP traffic on Port 3306.
- E. Create and launch a Lambda function in a new public subnet that is in a new VPC named VPC2. Create a peering connection between VPC1 and VPC2.
- F. Create the Lambda function
- G. Configure VPC1 access for the function
- H. Assign a security group named SG1 to the Lambda function
- I. Assign a second security group named SG2 to the database

J. Add an inbound rule to SG1 to allow TCP traffic from Port 3306.

K. Export the data from the Aurora database to Amazon S3. Create and launch a Lambda function in VPC1. Configure the Lambda function query the data from Amazon S3.

Answer: A

Explanation:

AWS Lambda is a service that lets you run code without provisioning or managing servers. Lambda functions can be configured to access resources in a VPC, such as an Aurora database, by specifying one or more subnets and security groups in the VPC settings of the function. A security group acts as a virtual firewall that controls inbound and outbound traffic for the resources in a VPC. To allow a Lambda function to communicate with an Aurora database, both resources need to be associated with the same security group, and the security group rules need to allow TCP traffic on Port 3306, which is the default port for MySQL databases. Reference: [Configuring a Lambda function to access resources in a VPC]

NEW QUESTION 62

A developer has a legacy application that is hosted on-premises. Other applications hosted on AWS depend on the on-premises application for proper functioning. In case of any application errors, the developer wants to be able to use Amazon CloudWatch to monitor and troubleshoot all applications from one place. How can the developer accomplish this?

- A. Install an AWS SDK on the on-premises server to automatically send logs to CloudWatch.
- B. Download the CloudWatch agent to the on-premises server
- C. Configure the agent to use IAM user credentials with permissions for CloudWatch.
- D. Upload log files from the on-premises server to Amazon S3 and have CloudWatch read the files.
- E. Upload log files from the on-premises server to an Amazon EC2 instance and have the instance forward the logs to CloudWatch.

Answer: B

Explanation:

Amazon CloudWatch is a service that monitors AWS resources and applications. The developer can use CloudWatch to monitor and troubleshoot all applications from one place. To do so, the developer needs to download the CloudWatch agent to the on-premises server and configure the agent to use IAM user credentials with permissions for CloudWatch. The agent will collect logs and metrics from the on-premises server and send them to CloudWatch.

References:

? [What Is Amazon CloudWatch? - Amazon CloudWatch]

? [Installing and Configuring the CloudWatch Agent - Amazon CloudWatch]

NEW QUESTION 67

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

- A. Query the instance metadata from `http://169.254.169.254/latest/meta-data/`.
- B. Query the instance user data from `http://169.254.169.254/latest/user-data/`
- C. Query the Amazon Machine Image (AMI) information from `http://169.254.169.254/latest/meta-data/ami/`.
- D. Check the hosts file of the operating system

Answer: A

Explanation:

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

? Instance metadata and user data

? Get Public IP Address on current EC2 Instance

? Get the public IP address of your EC2 instance quickly

NEW QUESTION 70

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

Answer: D

Explanation:

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

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

NEW QUESTION 72

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

A developer needs to implement a solution that optimizes the photos that are served to each device to reduce load time and increase photo quality.

Which solution will meet these requirements MOST cost-effective?

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

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

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

Answer: D

Explanation:

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

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

NEW QUESTION 74

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

A.

Configure the CloudFront cache

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

Answer: B

Explanation:

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

NEW QUESTION 77

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

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

How can the developer resolve this issue?

- A. Increase the SQS event source's batch size setting.
- B. Configure provisioned concurrency for the Lambda function based on the third-party API's documented rate limits.
- C. Increase the retry attempts and maximum event age in the Lambda function's asynchronous configuration.
- D. Configure maximum concurrency on the SQS event source based on the third-party service's documented rate limits.

Answer: D

Explanation:

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

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

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

NEW QUESTION 80

A developer is creating a template that uses AWS CloudFormation to deploy an application. The application is serverless and uses Amazon API Gateway, Amazon DynamoDB, and AWS Lambda.

Which AWS service or tool should the developer use to define serverless resources in YAML?

- A. CloudFormation serverless intrinsic functions
- B. AWS Elastic Beanstalk
- C. AWS Serverless Application Model (AWS SAM)
- D. AWS Cloud Development Kit (AWS CDK)

Answer: C

Explanation:

AWS Serverless Application Model (AWS SAM) is an open-source framework that enables developers to build and deploy serverless applications on AWS. AWS SAM uses a template specification that extends AWS CloudFormation to simplify the

definition of serverless resources such as API Gateway, DynamoDB, and Lambda. The developer can use AWS SAM to define serverless resources in YAML and deploy them using the AWS SAM CLI.

References:

? [What Is the AWS Serverless Application Model (AWS SAM)? - AWS Serverless Application Model]

? [AWS SAM Template Specification - AWS Serverless Application Model]

NEW QUESTION 85

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

Which solution will meet these requirements'?

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

Answer: A

Explanation:

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

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

NEW QUESTION 88

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

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

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

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

Answer: BE

Explanation:

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

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

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

? Remove any commands that perform operating system patching from the

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

References:

? [What Is AWS CloudFormation? - AWS CloudFormation]

? [What Is AWS CodeDeploy? - AWS CodeDeploy]

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

NEW QUESTION 89

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

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

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

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

Answer: D

Explanation:

AWS Secrets Manager is a service that helps you store, distribute, and rotate secrets securely. You can use Secrets Manager to migrate your credentials from your application code to a secure and encrypted storage. You can also enable automatic rotation of your secrets by using AWS Lambda functions or custom logic.

You can use IAM policies and roles to grant your Amazon ECS Fargate tasks permissions to access your secrets from Secrets Manager. This solution minimizes the operational overhead of managing your credentials and enhances the security of your application. References

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

News Blog

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

? Top 5 AWS root account best practices - TheServerSide

NEW QUESTION 91

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

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

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

? How to Deploy Docker Containers | AWS

? Deploy a Web App Using AWS App Runner

? How to Deploy Containerized Apps on AWS Using ECR and Docker

NEW QUESTION 94

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 97

A company has an Amazon S3 bucket containing premier content that it intends to make available to only paid subscribers of its website. The S3 bucket currently has default permissions of all objects being private to prevent inadvertent exposure of the premier content to non-paying website visitors.

How can the company Limit the ability to download a premier content file in the S3 Bucket to paid subscribers only?

- A. Apply a bucket policy that allows anonymous users to download the content from the S3 bucket.
- B. Generate a pre-signed object URL for the premier content file when a pad subscriber requests a download.
- C. Add a Docket policy that requires multi-factor authentication for request to access the S3 bucket objects.
- D. Enable server-side encryption on the S3 bucket for data protection against the non- paying website visitors.

Answer: B

Explanation:

This solution will limit the ability to download a premier content file in the S3 bucket to paid subscribers only because it uses a pre-signed object URL that grants temporary access to an S3 object for a specified duration. The pre-signed object URL can be generated by the company's website when a paid subscriber requests a download, and can be verified by Amazon S3 using the signature in the URL. Option A is not optimal because it will allow anyone to download the content from the S3 bucket without verifying their subscription status. Option C is not optimal because it will require additional steps and costs to configure multi-factor authentication for accessing the S3 bucket objects, which may not be feasible or user-friendly for paid subscribers. Option D is not optimal because it will not prevent non-paying website visitors from accessing the S3 bucket objects, but only encrypt them at rest.

References: Share an Object with Others, [Using Amazon S3 Pre-Signed URLs]

NEW QUESTION 100

When using the AWS Encryption SDK how does the developer keep track of the data encryption keys used to encrypt data?

- A. The developer must manually keep track of the data encryption keys used for each data object.
- B. The SDK encrypts the data encryption key and stores it (encrypted) as part of the returned ciphertext.
- C. The SDK stores the data encryption keys automatically in Amazon S3.
- D. The data encryption key is stored in the user data for the EC2 instance.

Answer: B

Explanation:

This solution will meet the requirements by using AWS Encryption SDK, which is a client-side encryption library that enables developers to encrypt and decrypt data using data encryption keys that are protected by AWS Key Management Service (AWS KMS). The SDK encrypts the data encryption key with a customer master key (CMK) that is managed by AWS KMS, and stores it (encrypted) as part of the returned ciphertext. The developer does not need to keep track of the data encryption keys used to encrypt data, as they are stored with the encrypted data and can be retrieved and decrypted by using AWS KMS when needed. Option A is not optimal because it will require manual tracking of the data encryption keys used for each data object, which is error-prone and inefficient. Option C is not optimal because it will store the data encryption keys automatically in Amazon S3, which is unnecessary and insecure as Amazon S3 is not designed for storing encryption keys. Option D is not optimal because it will store the data encryption key in the user data for the EC2 instance, which is also unnecessary and insecure as user data is not encrypted by default.

References: [AWS Encryption SDK], [AWS Key Management Service]

NEW QUESTION 102

A developer is designing an AWS Lambda function that creates temporary files that are less than 10 MB during invocation. The temporary files will be accessed and modified multiple times during invocation. The developer has no need to save or retrieve these files in the future.

Where should the temporary files be stored?

- A. the /tmp directory
- B. Amazon Elastic File System (Amazon EFS)
- C. Amazon Elastic Block Store (Amazon EBS)
- D. Amazon S3

Answer: A

Explanation:

AWS Lambda is a service that lets developers run code without provisioning or managing servers. Lambda provides a local file system that can be used to store temporary files during invocation. The local file system is mounted under the /tmp directory and has a limit of 512 MB. The temporary files are accessible only by the Lambda function that created them and are deleted after the function execution ends. The developer can store temporary files that are less than 10 MB in the /tmp directory and access and modify them multiple times during invocation.

References:

? [What Is AWS Lambda? - AWS Lambda]

? [AWS Lambda Execution Environment - AWS Lambda]

NEW QUESTION 103

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

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

Answer: B

Explanation:

This solution meets the requirements with the least amount of configuration because it uses a feature of AWS CDK that allows custom logic to be executed during stack deployment or deletion. The AWS Cloud Development Kit (AWS CDK) is a software development framework that allows you to define cloud infrastructure as code and provision it through CloudFormation. An AWS CDK custom resource is a construct that enables you to create resources that are not natively supported by CloudFormation or perform tasks that are not supported by CloudFormation during stack deployment or deletion. The developer can write a handler function in the code that uses AWS SDK calls to check for and delete unused resources, and create an AWS CDK custom resource that attaches the function code to a Lambda function and invokes it when the deployment stack runs. This way, the developer can automate the cleanup process without requiring additional configuration or integration. Creating a CloudFormation template from a JSON file will require additional configuration and integration with the central AWS CDK application. Creating an API in AWS Amplify will require additional configuration and integration with the central AWS CDK application and may not provide optimal performance or availability. Writing a handler function in the AWS Lambda console will require additional configuration and integration with the central AWS CDK application.

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

NEW QUESTION 106

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 107

A developer is using an AWS Lambda function to generate avatars for profile pictures that are uploaded to an Amazon S3 bucket. The Lambda function is automatically invoked for profile pictures that are saved under the /original/ S3 prefix. The developer notices that some pictures cause the Lambda function to time out. The developer wants to implement a fallback mechanism by using another Lambda function that resizes the profile picture.

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

- A. Set the image resize Lambda function as a destination of the avatar generator Lambda function for the events that fail processing.
- B. Create an Amazon Simple Queue Service (Amazon SQS) queue
- C. Set the SQS queue as a destination with an on failure condition for the avatar generator Lambda function
- D. Configure the image resize Lambda function to poll from the SQS queue.
- E. Create an AWS Step Functions state machine that invokes the avatar generator Lambda function and uses the image resize Lambda function as a fallback
- F. Create an Amazon EventBridge rule that matches events from the S3 bucket to invoke the state machine.
- G. Create an Amazon Simple Notification Service (Amazon SNS) topic
- H. Set the SNS topic as a destination with an on failure condition for the avatar generator Lambda function
- I. Subscribe the image resize Lambda function to the SNS topic.

Answer: A

Explanation:

The solution that will meet the requirements with the least development effort is to set the image resize Lambda function as a destination of the avatar generator Lambda function for the events that fail processing. This way, the fallback mechanism is automatically triggered by the Lambda service without requiring any

additional components or configuration. The other options involve creating and managing additional resources such as queues, topics, state machines, or rules, which would increase the complexity and cost of the solution.

Reference: Using AWS Lambda destinations

NEW QUESTION 109

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 111

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

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

NEW QUESTION 113

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 table.
- 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 DynamoDB. References: ? Working with Global Secondary Indexes - Amazon DynamoDB ? DynamoDB Performance & Latency - Everything You Need To Know

NEW QUESTION 116

A developer is migrating an application to Amazon Elastic Kubernetes Service (Amazon EKS). The developer migrates the application to Amazon Elastic Container Registry (Amazon ECR) with an EKS cluster. As part of the application migration to a new backend, the developer creates a new AWS account. The developer makes configuration changes to the application to point the application to the new AWS account and to use new backend resources. The developer successfully tests the changes within the application by deploying the pipeline.

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

- A. The developer did not successfully create the new AWS account.
- B. The developer added a new tag to the Docker image.
- C. The developer did not update the Docker image tag to a new version.
- D. The developer pushed the changes to a new Docker image tag.

Answer: C

Explanation:

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

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

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

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

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

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

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

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

NEW QUESTION 120

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 functio
- 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 121

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

Which solution will meet these requirements?

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

Answer: A

Explanation:

? A feature branch is a branch that is created from the main branch to work on a specific feature or task1. Feature branches allow developers to isolate their work from the main branch and avoid conflicts with other changes1. Feature branches can be merged back to the main branch when the feature or task is completed and tested1.

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

? The developer can create a feature branch from the main branch for production bug fixes. This branch will contain the code that is currently deployed in

production, and any fixes or updates that need to be applied to it. The developer can push this branch to the CodeCommit repository and use it to deploy changes to the production environment.

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

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

NEW QUESTION 123

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 126

.....

Thank You for Trying Our Product

We offer two products:

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

2nd - Questions and Answers in PDF Format

DVA-C02 Practice Exam Features:

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

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