



Amazon

Exam Questions DVA-C02

DVA-C02

About ExamBible

Your Partner of IT Exam

Found in 1998

ExamBible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, ExamBible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

All examinations will be up to date.

* 24/7 Quality Support

We will provide service round the clock.

* 100% Pass Rate

Our guarantee that you will pass the exam.

* Unique Gurantee

If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

NEW QUESTION 1

A company is using Amazon OpenSearch Service to implement an audit monitoring system. A developer needs to create an AWS CloudFormation custom resource that is

associated with an AWS Lambda function to configure the OpenSearch Service domain. The Lambda function must access the OpenSearch Service domain by using Open Search Service internal master user credentials. What is the MOST secure way to pass these credentials to the Lambda function?

- A. Use a CloudFormation parameter to pass the master user credentials at deployment to the OpenSearch Service domain's MasterUserOptions and the Lambda function's environment variable
- B. Set the No Echo attribute to true.
- C. Use a CloudFormation parameter to pass the master user credentials at deployment to the OpenSearch Service domain's MasterUserOptions and to create a parameter
- D. In AWS Systems Manager Parameter Store
- E. Set the No Echo attribute to true
- F. Create an IAM role that has the ssm GetParameter permission
- G. Assign the role to the Lambda function
- H. Store the parameter name as the Lambda function's environment variable
- I. Resolve the parameter's value at runtime.
- J. Use a CloudFormation parameter to pass the master user credentials at deployment to the OpenSearch Service domain's MasterUserOptions and the Lambda function's environment variable. We encrypt the parameter's value by using the AWS Key Management Service (AWS KMS) encrypt command.
- K. Use CloudFormation to create an AWS Secrets Manager secret
- L. Use a CloudFormation dynamic reference to retrieve the secret's value for the OpenSearch Service domain's MasterUserOptions
- M. Create an IAM role that has the secretsmanager:GetSecretValue permission
- N. Assign the role to the Lambda function. Store the secret's name as the Lambda function's environment variable
- O. Resolve the secret's value at runtime.

Answer: D

Explanation:

The solution that will meet the requirements is to use CloudFormation to create an AWS Secrets Manager secret. Use a CloudFormation dynamic reference to retrieve the secret's value for the OpenSearch Service domain's MasterUserOptions. Create an IAM role that has the secretsmanager:GetSecretValue permission. Assign the role to the Lambda function. Store the secret's name as the Lambda function's environment variable. Resolve the secret's value at runtime. This way, the developer can pass the credentials to the Lambda function in a secure way, as AWS Secrets Manager encrypts and manages the secrets. The developer can also use a dynamic reference to avoid exposing the secret's value in plain text in the CloudFormation template. The other options either involve passing the credentials as plain text parameters, which is not secure, or encrypting them with AWS KMS, which is less convenient than using AWS Secrets Manager.

Reference: Using dynamic references to specify template values

NEW QUESTION 2

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 environments
- 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 3

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

Which solution will meet these requirements?

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

Answer: C

Explanation:

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

? Using AWS Lambda with Amazon SQS

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

? Lambda event source mappings - AWS Lambda

? aws_lambda_event_source_mapping - Terraform Registry

NEW QUESTION 4

A developer maintains a critical business application that uses Amazon DynamoDB as the primary data store. The DynamoDB table contains millions of documents and receives 30-60 requests each minute. The developer needs to perform processing in near-real time on the documents when they are added or updated in the DynamoDB table.

How can the developer implement this feature with the LEAST amount of change to the existing application code?

- A. Set up a cron job on an Amazon EC2 instance. Run a script every hour to query the table for changes and process the documents.
- B. Enable a DynamoDB stream on the table. Invoke an AWS Lambda function to process the documents.
- C. Update the application to send a PutEvents request to Amazon EventBridge.
- D. Create an EventBridge rule to invoke an AWS Lambda function to process the documents.
- E. Update the application to synchronously process the documents directly after the DynamoDB write.

Answer: B

Explanation:

<https://aws.amazon.com/blogs/database/dynamodb-streams-use-cases-and-design-patterns/>

NEW QUESTION 5

A company is building a new application that runs on AWS and uses Amazon API Gateway to expose APIs. Teams of developers are working on separate components of the application in parallel. The company wants to publish an API without an integrated backend, so that teams that depend on the application backend can continue the development work before the API backend development is complete. Which solution will meet these requirements?

- A. Create API Gateway resources and set the integration type value to MOCK. Configure the method integration request and integration response to associate a response with an HTTP status code. Create an API Gateway stage and deploy the API.
- B. Create an AWS Lambda function that returns mocked responses and various HTTP status codes.
- C. Create API Gateway resources and set the integration type value to AWS_PROXY. Deploy the API.
- D. Create an EC2 application that returns mocked HTTP responses. Create API Gateway resources and set the integration type value to AWS. Create an API Gateway stage and deploy the API.
- E. Create API Gateway resources and set the integration type value set to HTTP_PROXY.
- F. Add mapping templates and deploy the API.
- G. Create an AWS Lambda layer that returns various HTTP status codes. Associate the Lambda layer with the API deployment.

Answer: A

Explanation:

The best solution for publishing an API without an integrated backend is to use the MOCK integration type in API Gateway. This allows the developer to return a static response to the client without sending the request to a backend service. The developer can configure the method integration request and integration response to associate a response with an HTTP status code, such as 200 OK or 404 Not Found. The developer can also create an API Gateway stage and deploy the API to make it available to the teams that depend on the application backend. The other solutions are either not feasible or not efficient. Creating an AWS Lambda function, an EC2 application, or an AWS Lambda layer would require additional resources and code to generate the mocked responses and HTTP status codes. These solutions would also incur additional costs and complexity, and would not leverage the built-in functionality of API Gateway. References

? Set up mock integrations for API Gateway REST APIs

? Mock Integration for API Gateway - AWS CloudFormation

? Mocking API Responses with API Gateway

? How to mock API Gateway responses with AWS SAM

NEW QUESTION 6

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

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

Answer: B

Explanation:

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

for the application that runs across multiple AWS Regions. References

? AWS X-Ray Annotations

? AWS X-Ray Concepts

NEW QUESTION 7

A developer is optimizing an AWS Lambda function and wants to test the changes in production on a small percentage of all traffic. The Lambda function serves requests to a REST API in Amazon API Gateway. The developer needs to deploy their changes and perform a test in production without changing the API Gateway URL. Which solution will meet these requirements?

- A. Define a function version for the currently deployed production Lambda function
- B. Update the API Gateway endpoint to reference the new Lambda function version
- C. Upload and publish the optimized Lambda function code
- D. On the production API Gateway stage, define a canary release and set the percentage of traffic to direct to the canary release
- E. Update the API Gateway endpoint to use the \$LATEST version of the Lambda function
- F. Publish the API to the canary stage.
- G. Define a function version for the currently deployed production Lambda function
- H. Update the API Gateway endpoint to reference the new Lambda function version
- I. Upload and publish the optimized Lambda function code
- J. Update the API Gateway endpoint to use the \$LATEST version of the Lambda function
- K. Deploy a new API Gateway stage.
- L. Define an alias on the \$LATEST version of the Lambda function
- M. Update the API Gateway endpoint to reference the new Lambda function alias
- N. Upload and publish the optimized Lambda function code
- O. On the production API Gateway stage, define a canary release and set the percentage of traffic to direct to the canary release
- P. Update the API Gateway endpoint to use the \$LATEST version of the Lambda function
- Q. Publish to the canary stage.
- R. Define a function version for the currently deployed production Lambda function
- S. Update the API Gateway endpoint to reference the new Lambda function version
- T. Upload and publish the optimized Lambda function code
- . Update the API Gateway endpoint to use the \$LATEST version of the Lambda function
- . Deploy the API to the production API Gateway stage.

Answer: C

Explanation:

? A Lambda alias is a pointer to a specific Lambda function version or another alias¹. A Lambda alias allows you to invoke different versions of a function using the same name¹. You can also split traffic between two aliases by assigning weights to them¹.

? In this scenario, the developer needs to test their changes in production on a small percentage of all traffic without changing the API Gateway URL. To achieve this, the developer can follow these steps:

? By using this solution, the developer can test their changes in production on a small percentage of all traffic without changing the API Gateway URL. The developer can also monitor and compare metrics between the canary and production releases, and promote or disable the canary as needed².

NEW QUESTION 8

A developer is designing a serverless application for a game in which users register and log in through a web browser. The application makes requests on behalf of users to a set of AWS Lambda functions that run behind an Amazon API Gateway HTTP API.

The developer needs to implement a solution to register and log in users on the application's sign-in page. The solution must minimize operational overhead and must minimize ongoing management of user identities.

Which solution will meet these requirements?

- A. Create Amazon Cognito user pools for external social identity providers. Configure IAM roles for the identity pools.
- B. Program the sign-in page to create users' IAM groups with the IAM roles attached to the groups.
- C. Create an Amazon RDS for SQL Server DB instance to store the users and manage the permissions to the backend resources in AWS.
- D. Configure the sign-in page to register and store the users and their passwords in an Amazon DynamoDB table with an attached IAM policy.

Answer: A

Explanation:

<https://docs.aws.amazon.com/cognito/latest/developerguide/signing-up-users-in-your-app.html>

NEW QUESTION 9

A developer needs to migrate an online retail application to AWS to handle an anticipated increase in traffic. The application currently runs on two servers: one server for the web application and another server for the database. The web server renders webpages and manages session state in memory. The database server hosts a MySQL database that contains order details. When traffic to the application is heavy, the memory usage for the web server approaches 100% and the application slows down considerably.

The developer has found that most of the memory increase and performance decrease is related to the load of managing additional user sessions. For the web server migration, the developer will use Amazon EC2 instances with an Auto Scaling group behind an Application Load Balancer.

Which additional set of changes should the developer make to the application to improve the application's performance?

- A. Use an EC2 instance to host the MySQL database.
- B. Store the session data and the application data in the MySQL database.
- C. Use Amazon ElastiCache for Memcached to store and manage the session data.
- D. Use an Amazon RDS for MySQL DB instance to store the application data.
- E. Use Amazon ElastiCache for Memcached to store and manage the session data and the application data.
- F. Use the EC2 instance store to manage the session data.
- G. Use an Amazon RDS for MySQL DB instance to store the application data.

Answer: B

Explanation:

Using Amazon ElastiCache for Memcached to store and manage the session data will reduce the memory load and improve the performance of the web server. Using Amazon RDS for MySQL DB instance to store the application data will provide a scalable, reliable, and managed database service. Option A is not optimal because it does not address the memory issue of the web server. Option C is not optimal because it does not provide a persistent storage for the application data. Option D is not optimal because it does not provide a high availability and durability for the session data.

References: Amazon ElastiCache, Amazon RDS

NEW QUESTION 10

A developer has observed an increase in bugs in the AWS Lambda functions that a development team has deployed in its Node.js application. To minimize these bugs, the developer wants to implement automated testing of Lambda functions in an environment that closely simulates the Lambda environment.

The developer needs to give other developers the ability to run the tests locally. The developer also needs to integrate the tests into the team's continuous integration and continuous delivery (CI/CD) pipeline before the AWS Cloud Development Kit (AWS CDK) deployment.

Which solution will meet these requirements?

- A. Create sample events based on the Lambda documentation
- B. Create automated test scripts that use the `cdk local invoke` command to invoke the Lambda function
- C. Check the response. Document the test scripts for the other developers on the team. Update the CI/CD pipeline to run the test scripts.
- D. Install a unit testing framework that reproduces the Lambda execution environment
- E. Create sample events based on the Lambda Documentation. Invoke the handler function by using a unit testing framework for the other developers on the team
- F. Check the response. Document how to run the unit testing framework.
- G. Update the CI/CD pipeline to run the unit testing framework.
- H. Install the AWS Serverless Application Model (AWS SAM) CLI tool. Use the `Sam local generate-event` command to generate sample events for the automated test
- I. Create automated test scripts that use the `Sam local invoke` command to invoke the Lambda function
- J. Check the response. Document the test scripts for the other developers on the team. Update the CI/CD pipeline to run the test scripts.
- K. Create sample events based on the Lambda documentation
- L. Create a Docker container from the Node.js base image to invoke the Lambda function
- M. Check the response. Document how to run the Docker container for the other developers on the team. Update the CI/CD pipeline to run the Docker container.

Answer: C

Explanation:

This solution will meet the requirements by using the AWS SAM CLI tool, which is a command-line tool that lets developers locally build, test, debug, and deploy serverless applications defined by AWS SAM templates. The developer can use `sam local generate-event` to generate sample events for different event sources such as API Gateway or S3. The developer can create automated test scripts that use `sam local invoke` to invoke Lambda functions locally in an environment that closely simulates the Lambda environment. The developer can check the response from Lambda functions and document how to run the test scripts for other developers on the team. The developer can also update the CI/CD pipeline to run these test scripts before deploying with AWS CDK. Option A is not optimal because it will use `cdk local invoke`, which does not exist in the AWS CDK CLI tool. Option B is not optimal because it will use a unit testing framework that reproduces the Lambda execution environment, which may not be accurate or consistent with the Lambda environment. Option D is not optimal because it will create a Docker container from the Node.js base image to invoke Lambda functions, which may introduce additional overhead and complexity for creating and running Docker containers.

References: [AWS Serverless Application Model (AWS SAM)], [AWS Cloud Development Kit (AWS CDK)]

NEW QUESTION 10

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

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

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

Answer: D

NEW QUESTION 14

A developer is creating a new REST API by using Amazon API Gateway and AWS Lambda. The development team tests the API and validates responses for the known use cases before deploying the API to the production environment.

locally.

The developer wants to make the REST API available for testing by using API Gateway. Which AWS Serverless Application Model Command Line Interface (AWS SAM CLI) subcommand will meet these requirements?

- A. `Sam local invoke`
- B. `Sam local generate-event`
- C. `Sam local start-lambda`
- D. `Sam local start-api`

Answer: D

Explanation:

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

NEW QUESTION 16

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 18

A developer is creating a mobile application that will not require users to log in. What is the MOST efficient method to grant users access to AWS resources'?

- A. Use an identity provider to securely authenticate with the application.
- B. Create an AWS Lambda function to create an IAM user when a user accesses the application.
- C. Create credentials using AWS KMS and apply these credentials to users when using the application.
- D. Use Amazon Cognito to associate unauthenticated users with an IAM role that has limited access to resources.

Answer: D

Explanation:

This solution is the most efficient method to grant users access to AWS resources without requiring them to log in. Amazon Cognito is a service that provides user sign-up, sign-in, and access control for web and mobile applications. Amazon Cognito identity pools support both authenticated and unauthenticated users. Unauthenticated users receive access to your AWS resources even if they aren't logged in with any of your identity providers (IdPs). You can use Amazon Cognito to associate unauthenticated users with an IAM role that has limited access to resources, such as Amazon S3 buckets or DynamoDB tables. This degree of access is useful to display content to users before they log in or to allow them to perform certain actions without signing up. Using an identity provider to securely authenticate with the application will require users to log in, which does not meet the requirement. Creating an AWS Lambda function to create an IAM user when a user accesses the application will incur unnecessary costs and complexity, and may pose security risks if not implemented properly. Creating credentials using AWS KMS and applying them to users when using the application will also incur unnecessary costs and complexity, and may not provide fine-grained access control for resources.

Reference: Switching unauthenticated users to authenticated users (identity pools), Allow user access to your API without authentication (Anonymous user access)

NEW QUESTION 22

A company is using an AWS Lambda function to process records from an Amazon Kinesis data stream. The company recently observed slow processing of the records. A developer notices that the iterator age metric for the function is increasing and that the Lambda run duration is constantly above normal. Which actions should the developer take to increase the processing speed? (Choose two.)

- A. Increase the number of shards of the Kinesis data stream.
- B. Decrease the timeout of the Lambda function.
- C. Increase the memory that is allocated to the Lambda function.
- D. Decrease the number of shards of the Kinesis data stream.
- E. Increase the timeout of the Lambda function.

Answer: AC

Explanation:

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

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

NEW QUESTION 26

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 30

A company has an existing application that has hardcoded database credentials A developer needs to modify the existing application The application is deployed in two AWS Regions with an active-passive failover configuration to meet company's disaster recovery strategy The developer needs a solution to store the credentials outside the code. The solution must comply With the company's disaster recovery strategy Which solution Will meet these requirements in the MOST secure way?

- A. Store the credentials in AWS Secrets Manager in the primary Regio
- B. Enable secret replication to the secondary Region Update the application to use the Amazon Resource Name (ARN) based on the Region.
- C. Store credentials in AWS Systems Manager Parameter Store in the primary Regio
- D. Enable parameter replication to the secondary Regio
- E. Update the application to use the Amazon Resource Name (ARN) based on the Region.
- F. Store credentials in a config fil
- G. Upload the config file to an S3 bucket in me primary Regio
- H. Enable Cross-Region Replication (CRR) to an S3 bucket in the secondary regio
- I. Update the application to access the config file from the S3 bucket based on the Region.
Store credentials in a config fil
- K. Upload the config file to an Amazon Elastic File System (Amazon EFS) file syste
- L. Update the application to use the Amazon EFS file system Regional endpoints to access the config file in the primary and secondary Regions.

Answer: A

Explanation:

AWS Secrets Manager is a service that allows you to store and manage secrets, such as database credentials, API keys, and passwords, in a secure and centralized way. It also provides features such as automatic secret rotation, auditing, and monitoring¹. By using AWS Secrets Manager, you can avoid hardcoding credentials in your code, which is a bad security practice and makes it difficult to update them. You can also replicate your secrets to another Region, which is useful for disaster recovery purposes². To access your secrets from your application, you can use the ARN of the secret, which is a unique identifier that includes the Region name. This way, your application can use the appropriate secret based on the Region where it is deployed³.

References:

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

NEW QUESTION 31

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

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

Answer: C

Explanation:

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

NEW QUESTION 32

A financial company must store original customer records for 10 years for legal reasons. A complete record contains personally identifiable information (PII).

According to local regulations, PII is available to only certain people in the company and must not be shared with third parties. The company needs to make the records available to third-party organizations for statistical analysis without sharing the PII.

A developer wants to store the original immutable record in Amazon S3. Depending on who accesses the S3 document, the document should be returned as is or with all the PII removed. The developer has written an AWS Lambda function to remove the PII from the document. The function is named `removePii`.

What should the developer do so that the company can meet the PII requirements while maintaining only one copy of the document?

- A. Set up an S3 event notification that invokes the `removePii` function when an S3 GET request is made
- B. Call Amazon S3 by using a GET request to access the object without PII.
- C. Set up an S3 event notification that invokes the `removePii` function when an S3 PUT request is made
- D. Call Amazon S3 by using a PUT request to access the object without PII.
- E. Create an S3 Object Lambda access point from the S3 console
- F. Select the `removePii` function
- G. Use S3 Access Points to access the object without PII.
- H. Create an S3 access point from the S3 console
- I. Use the access point name to call the `GetObjectLegalHold` S3 API function
- J. Pass in the `removePii` function name to access the object without PII.

Answer: C

Explanation:

S3 Object Lambda allows you to add your own code to process data retrieved from S3 before returning it to an application. You can use an AWS Lambda function to modify the data, such as removing PII, redacting confidential information, or resizing images. You can create an S3 Object Lambda access point and associate it with your Lambda function. Then, you can use the access point to request objects from S3 and get the modified data back. This way, you can maintain only one copy of the original

document in S3 and apply different transformations depending on who accesses it. Reference: Using AWS Lambda with Amazon S3

NEW QUESTION 33

For a deployment using AWS Code Deploy, what is the run order of the hooks for in-place deployments?

- A. `BeforeInstall` -> `ApplicationStop` -> `ApplicationStart` -> `AfterInstall`
- B. `ApplicationStop` -> `BeforeInstall` -> `AfterInstall` -> `ApplicationStart`
- C. `BeforeInstall` -> `ApplicationStop` -> `ValidateService` -> `ApplicationStart`
- D. `ApplicationStop` -> `BeforeInstall` -> `ValidateService` -> `ApplicationStart`

Answer: B

Explanation:

For in-place deployments, AWS CodeDeploy uses a set of predefined hooks that run in a specific order during each deployment lifecycle event. The hooks are `ApplicationStop`, `BeforeInstall`, `AfterInstall`, `ApplicationStart`, and `ValidateService`. The run order of the hooks for in-place deployments is as follows:

? `ApplicationStop`: This hook runs first on all instances and stops the current application that is running on the instances.

? `BeforeInstall`: This hook runs after `ApplicationStop` on all instances and performs any tasks required before installing the new application revision.

? `AfterInstall`: This hook runs after `BeforeInstall` on all instances and performs any tasks required after installing the new application revision.

? `ApplicationStart`: This hook runs after `AfterInstall` on all instances and starts the new application that has been installed on the instances.

? `ValidateService`: This hook runs last on all instances and verifies that the new application is running properly on the instances.

Reference: [AWS CodeDeploy lifecycle event hooks reference]

NEW QUESTION 35

A company is migrating an on-premises database to Amazon RDS for MySQL. The company has read-heavy workloads. The company wants to refactor the code to achieve optimum read performance for queries.

Which solution will meet this requirement with LEAST current and future effort?

- A. Use a multi-AZ Amazon RDS deployment
- B**: Increase the number of connections that the code makes to the database or increase the connection pool size if a connection pool is in use.
- C. Use a multi-AZ Amazon RDS deployment
- D. Modify the code so that queries access the secondary RDS instance.
- E. Deploy Amazon RDS with one or more read replicas
- F. Modify the application code so that queries use the URL for the read replicas.
- G. Use open source replication software to create a copy of the MySQL database on an Amazon EC2 instance
- H. Modify the application code so that queries use the IP address of the EC2 instance.

Answer: C

Explanation:

Amazon RDS for MySQL supports read replicas, which are copies of the primary database instance that can handle read-only queries. Read replicas can improve the read performance of the database by offloading the read workload from the primary instance and distributing it across multiple replicas. To use read replicas, the application code needs to be modified to direct read queries to the URL of the read replicas, while write queries still go to the URL of the primary instance. This solution requires less current and future effort than using a multi-AZ deployment, which does not provide read scaling benefits, or using open source replication software, which requires additional configuration and maintenance. Reference: Working with read replicas

NEW QUESTION 40

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

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

- A. Set the `DeploymentPreferenceType` to `Canary` and `OPercent10Minute`
- B. Set the `AutoPublishAlias` property to the Lambda alias.

- C. Set the Deployment Preference Type to Linearl OPercentEverylOMinute
- D. Set AutoPublishAlias property to the Lambda alias.
- E. Set the Deployment Preference Type to Canaryl OPercentlOMinute
- F. Set the PreTraffic and PostTraffic properties to the Lambda alias.
- G. Set the Deployment Preference Type to Linearl OPercentEvery10Minute
- H. Set PreTraffic and PostTraffic properties to the Lambda alias.

Answer: A

Explanation:

? The Deployment Preference Type property specifies how traffic should be shifted between versions of a Lambda function¹. The Canary10Percent10Minutes option means that 10% of the traffic is immediately shifted to the new version, and after 10 minutes, the remaining 90% of the traffic is shifted¹. This matches the requirement of shifting 10% of the traffic for the first 10 minutes, and then switching all traffic to the new version.

? The AutoPublishAlias property enables AWS SAM to automatically create and update a Lambda alias that points to the latest version of the function¹. This is required to use the Deployment Preference Type property¹. The alias name can be specified by the developer, and it can be used to invoke the function with the latest code.

NEW QUESTION 41

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

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

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

- A: Use an Amazon Linux crontab scheduled job that runs on Amazon EC2.
- C. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event.
- D. Use an AWS Batch job that is submitted to an AWS Batch job queue.

Answer: C

Explanation:

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

* C. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event. This is correct. AWS Lambda is a serverless compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging¹. Amazon EventBridge is a serverless event bus service that enables you to connect your applications with data from a variety of sources². EventBridge can create rules that run on a schedule, either at regular intervals or at specific times and dates, and invoke targets such as Lambda functions³. This solution meets the requirements of creating a small application that makes the same API call once each day at a designated time, without requiring any infrastructure in the AWS Cloud or any operational overhead.

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

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

* D. Use an AWS Batch job that is submitted to an AWS Batch job queue. This is incorrect. AWS Batch enables you to run batch computing workloads on the AWS or sequentially on compute environments⁸. Batch jobs are units of work that can be submitted to job queues, where they are executed in parallel. This solution could meet the functional requirements of creating a small application that makes the same API call once each day at a designated time, but it would not be the most operationally efficient manner. The company would need to configure and manage an AWS Batch environment, which would incur additional costs and complexity.

References:

- ? 1: What is AWS Lambda? - AWS Lambda
- ? 2: What is Amazon EventBridge? - Amazon EventBridge
- ? 3: Creating an Amazon EventBridge rule that runs on a schedule - Amazon EventBridge
- ? 4: What is Amazon EKS? - Amazon EKS
- ? 5: CronJob - Kubernetes
- ? 6: What is Amazon EC2? - Amazon EC2
- ? 7: Crontab in Linux with 20 Useful Examples to Schedule Jobs - Tecmint
- ? 8: What is AWS Batch? - AWS Batch
- ? 9: Jobs - AWS Batch

NEW QUESTION 45

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

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

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

Which solution will meet these requirements with the LEAST complexity?

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

Answer: A

Explanation:

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

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

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

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

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

NEW QUESTION 47

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

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

Answer: BD

Explanation:

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

References: [Using HTTPS with CloudFront], [Restricting Access to Amazon S3 Content by Using an Origin Access Identity]

NEW QUESTION 48

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

Which solution will meet these requirements?

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

? Using AWS Lambda destinations

? Asynchronous invocation - AWS Lambda

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

? AWS Lambda Destinations: A Complete Guide | Dashbird

NEW QUESTION 53

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

Which solution will meet these requirements?

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

? Monitoring and troubleshooting Lambda functions - AWS Lambda

? Using AWS Lambda with AWS X-Ray

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

NEW QUESTION 55

A developer is working on a web application that uses Amazon DynamoDB as its data store. The application has two DynamoDB tables: one table that is named `artists` and one table that is named `songs`. The `artists` table has `artistName` as the partition key. The `songs` table has `songName` as the partition key and `artistName` as the sort key.

The table usage patterns include the retrieval of multiple songs and artists in a single database operation from the webpage. The developer needs a way to retrieve this information with minimal network traffic and optimal application performance.

Which solution will meet these requirements?

- A. Perform a `BatchGetItem` operation that returns items from the two tables.
- B. Use the list of `songName` `artistName` keys for the `songs` table and the list of `artistName` key for the `artists` table.
- C. Create a local secondary index (LSI) on the `songs` table that uses `artistName` as the partition key. Perform a query operation for each `artistName` on the `songs` table that filters by the list of `songName`. Perform a query operation for each `artistName` on the `artists` table.
- D. Perform a `BatchGetItem` operation on the `songs` table that uses the `songName/artistName` key.
- E. Perform a `BatchGetItem` operation on the `artists` table that uses `artistName` as the key.
- F. Perform a `Scan` operation on each table that filters by the list of `songName/artistName` for the `songs` table and the list of `artistName` in the `artists` table.

Answer: A

Explanation:

`BatchGetItem` can return one or multiple items from one or more tables. For reference, check the link below:

https://docs.aws.amazon.com/amazondynamodb/latest/APIReference/API_BatchGetItem.html

NEW QUESTION 60

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

Which solution will meet these requirements MOST cost-effectively?

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

Answer: A

Explanation:

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

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

References:

- ? `AWS::StepFunctions::StateMachine` - AWS CloudFormation
- ? Call API Gateway with Step Functions - AWS Step Functions
- ? `amazon-web-services aws-api-gateway terraform aws-step-functions`

NEW QUESTION 61

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. Use a software-as-a-service (SaaS) method.
- C. Use Amazon Cognito user pools to implement the employee directory in a fully managed, serverless way.
- 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 66

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://M69.254.169.254/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 69

A company's website runs on an Amazon EC2 instance and uses Auto Scaling to scale the environment during peak times. Website users across the world are experiencing high latency due to static content on the EC2 instance, even during non-peak hours. When combination of steps will resolve the latency issue? (Select TWO)

- A. Double the Auto Scaling group's maximum number of servers
- B. Host the application code on AWS Lambda
- C. Scale vertically by resizing the EC2 instances
- D. Create an Amazon CloudFront distribution to cache the static content
- E. Store the application's static content in Amazon S3

Answer: DE

Explanation:

The combination of steps that will resolve the latency issue is to create an Amazon CloudFront distribution to cache the static content and store the application's static content in Amazon S3. This way, the company can use CloudFront to deliver the static content from edge locations that are closer to the website users, reducing latency and improving performance. The company can also use S3 to store the static content reliably and cost-effectively, and integrate it with CloudFront easily. The other options either do not address the latency issue, or are not necessary or feasible for the given scenario.

Reference: Using Amazon S3 Origins and Custom Origins for Web Distributions

NEW QUESTION 73

A developer is creating an AWS Lambda function. The Lambda function needs an external library to connect to a third-party solution. The external library is a collection of files with a total size of 100 MB. The developer needs to make the external library available to the Lambda execution environment and reduce the Lambda package space.

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

A.

Create a Lambda layer to store the external library. Configure the Lambda function to use the layer.

- B. Create an Amazon S3 bucket. Upload the external library into the S3 bucket.
- C. Mount the S3 bucket folder in the Lambda function. Import the library by using the proper folder in the mount point.
- D. Load the external library to the Lambda function's `/tmp` directory during deployment of the Lambda package.
- E. Import the library from the `/tmp` directory.
- F. Create an Amazon Elastic File System (Amazon EFS) volume.
- G. Upload the external library to the EFS volume. Mount the EFS volume in the Lambda function.
- H. Import the library by using the proper folder in the mount point.

Answer: A

Explanation:

Create a Lambda layer to store the external library. Configure the Lambda function to use the layer. This will allow the developer to make the external library available to the Lambda execution environment without having to include it in the Lambda package, which will reduce the Lambda package space. Using a Lambda layer is a simple and straightforward solution that requires minimal operational overhead. <https://docs.aws.amazon.com/lambda/latest/dg/configuration-layers.html>

NEW QUESTION 76

A developer is investigating an issue in part of a company's application. In the application messages are sent to an Amazon Simple Queue Service (Amazon SQS) queue. The AWS Lambda function polls messages from the SQS queue and sends email messages by using Amazon Simple Email Service (Amazon SES). Users have been receiving duplicate email messages during periods of high traffic.

Which reasons could explain the duplicate email messages? (Select TWO.)

- A. Standard SQS queues support at-least-once message delivery
- B. Standard SQS queues support exactly-once processing, so the duplicate email messages are because of user error.
- C. Amazon SES has the DomainKeys Identified Mail (DKIM) authentication incorrectly configured
- D. The SQS queue's visibility timeout is lower than or the same as the Lambda function's timeout.
- E. The Amazon SES bounce rate metric is too high.

Answer: AD

Explanation:

Standard SQS queues support at-least-once message delivery, which means that a message can be delivered more than once to the same or different consumers. This can happen if the message is not deleted from the queue before the visibility timeout expires, or if there is a network issue or a system failure. The SQS queue's visibility timeout is the period of time that a message is invisible to other consumers after it is received by one consumer. If the visibility timeout is lower than or the same as the Lambda function's timeout, the Lambda function might not be able to process and delete the message before it becomes visible again, leading to duplicate processing and email messages. To avoid this, the visibility timeout should be set to at least 6 times the length of the Lambda function's timeout. The other options are not related to the issue of duplicate email messages. References

- ? Using the Amazon SQS message deduplication ID
- ? Exactly-once processing - Amazon Simple Queue Service
- ? Amazon SQS duplicated messages in queue - Stack Overflow
- ? amazon web services - How long can duplicate SQS messages persist ...
- ? Standard SQS - Duplicate message | AWS re:Post - Amazon Web Services, Inc.

NEW QUESTION 79

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 82

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

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

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

policy to allow access to the X-Ray API.

- E. Instrument the application by using the AWS X-Ray SDK
- F. Update the application to use the PutXrayTrace API call to communicate with the X-Ray API.

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

Answer: AE

Explanation:

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

Reference: Using AWS X-Ray with Amazon ECS

NEW QUESTION 85

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

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

Answer: C

Explanation:

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

References: AWS Lambda Monitoring, [CloudWatch Logs Insights]

NEW QUESTION 90

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

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

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

Answer: B

Explanation:

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

? Best practices for working with AWS Lambda functions

? Best practices for designing and architecting with DynamoDB

NEW QUESTION 91

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

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

- A. Use a Kubernetes cron job that runs on Amazon Elastic Kubernetes Service (Amazon EKS)
- B. Use an Amazon Linux crontab scheduled job that runs on Amazon EC2
- C. Use an AWS Lambda function that is invoked by an Amazon EventBridge scheduled event.
- D. Use an AWS Batch job that is submitted to an AWS Batch job queue.

Answer: C

Explanation:

This solution meets the requirements in the most operationally efficient manner because it does not require any infrastructure provisioning or management. The developer can create a Lambda function that makes the API call and configure an EventBridge rule that triggers the function once a day at a designated time. This is a serverless solution that scales automatically and only charges for the execution time of the function.

Reference: [Using AWS Lambda with Amazon EventBridge], [Schedule Expressions for Rules]

NEW QUESTION 93

An application runs on multiple EC2 instances behind an ELB.

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

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

Answer: A

Explanation:

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

Reference: Using ElastiCache for session management

NEW QUESTION 95

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 100

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 102

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

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

The developer needs to create/delete branches
 Which specific IAM permissions need to be added based on the principle of least privilege?

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

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

Answer: A

Explanation:

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

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

NEW QUESTION 104

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

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

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

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

Answer: C

Explanation:

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

NEW QUESTION 106

.....

Relate Links

100% Pass Your DVA-C02 Exam with Exam Bible Prep Materials

<https://www.exambible.com/DVA-C02-exam/>

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - <https://www.exambible.com/>