

Amazon

Exam Questions DVA-C02

DVA-C02



NEW QUESTION 1

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

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

Answer: D

Explanation:

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

? AWS Secrets Manager vs. Systems Manager Parameter Store

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

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

NEW QUESTION 2

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

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

Answer: B

Explanation:

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

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

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

NEW QUESTION 3

A developer is incorporating AWS X-Ray into an application that handles personally identifiable information (PII). The application is hosted on Amazon EC2 instances. The application trace messages include encrypted PII and go to Amazon CloudWatch. The developer needs to ensure that no PII goes outside of the EC2 instances. Which solution will meet these requirements?

- A. Manually instrument the X-Ray SDK in the application code.
- B. Use the X-Ray auto-instrumentation agent.
- C. Use Amazon Macie to detect and hide PII.
- D. Call the X-Ray API from AWS Lambda.
- E. Use AWS Distro for Open Telemetry.

Answer: A

Explanation:

This solution will meet the requirements by allowing the developer to control what data is sent to X-Ray and CloudWatch from the application code. The developer

can filter out any PII from the trace messages before sending them to X-Ray and CloudWatch, ensuring that no PII goes outside of the EC2 instances. Option B is not optimal because it will automatically instrument all incoming and outgoing requests from the application, which may include PII in the trace messages. Option C is not optimal because it will require additional services and costs to use Amazon Macie and AWS Lambda, which may not be able to detect and hide all PII from the trace messages. Option D is not optimal because it will use Open Telemetry instead of X-Ray, which may not be compatible with CloudWatch and other AWS services.

References: [AWS X-Ray SDKs]

NEW QUESTION 4

A developer is creating a mobile app that calls a backend service by using an Amazon API Gateway REST API. For integration testing during the development phase, the developer wants to simulate different backend responses without invoking the backend service.

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

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

Answer: D

Explanation:

Amazon API Gateway supports mock integration responses, which are predefined responses that can be returned without sending requests to a backend service. Mock integration responses can be used for testing or prototyping purposes, or for simulating different backend responses based on certain conditions. A request mapping template can be used to select a mock integration response based on an expression that evaluates some aspects of the request, such as headers, query strings, or body content. This solution does not require any additional resources or code changes and has the least operational overhead. Reference: Set up mock integrations for an API Gateway REST API

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

NEW QUESTION 5

A company has an application that uses Amazon Cognito user pools as an identity provider. The company must secure access to user records. The company has set up multi-factor authentication (MFA). The company also wants to send a login activity notification by email every time a user logs in.

What is the MOST operationally efficient solution that meets this requirement?

- A. Create an AWS Lambda function that uses Amazon Simple Email Service (Amazon SES) to send the email notification
- B. Add an Amazon API Gateway API to invoke the function
- C. Call the API from the client side when login confirmation is received.
- D. Create an AWS Lambda function that uses Amazon Simple Email Service (Amazon SES) to send the email notification
- E. Add an Amazon Cognito post authentication Lambda trigger for the function.
- F. Create an AWS Lambda function that uses Amazon Simple Email Service (Amazon SES) to send the email notification
- G. Create an Amazon CloudWatch Logs log subscription filter to invoke the function based on the login status.
- H. Configure Amazon Cognito to stream all logs to Amazon Kinesis Data Firehose
- I. Create an AWS Lambda function to process the streamed logs and to send the email notification based on the login status of each user.

Answer: B

Explanation:

Amazon Cognito user pools support Lambda triggers, which are custom functions that can be executed at various stages of the user pool workflow. A post authentication Lambda trigger can be used to perform custom actions after a user is authenticated, such as sending an email notification. Amazon SES is a cloud-based email sending service that can be used to send transactional or marketing emails. A Lambda function can use the Amazon SES API to send an email to the user's email address after the user logs in successfully. Reference: Post authentication Lambda trigger

NEW QUESTION 6

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

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

Answer: D

Explanation:

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

References:

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

? [What Is Amazon CloudFront? - Amazon CloudFront]

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

NEW QUESTION 7

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 8

A company is building a serverless application on AWS. The application uses an AWS Lambda function to process customer orders 24 hours a day, 7 days a week. The Lambda function calls an external vendor's HTTP API to process payments.

During load tests, a developer discovers that the external vendor payment processing API occasionally times out and returns errors. The company expects that some payment processing API calls will return errors.

The company wants the support team to receive notifications in near real time only when

the payment processing external API error rate exceeds 5% of the total number of transactions in an hour. Developers need to use an existing Amazon Simple Notification Service (Amazon SNS) topic that is configured to notify the support team.

Which solution will meet these requirements?

- A. Write the results of payment processing API calls to Amazon CloudWatch.
- B. Use Amazon CloudWatch Logs Insights to query the CloudWatch log.
- C. Schedule the Lambda function to check the CloudWatch logs and notify the existing SNS topic.
- D. Publish custom metrics to CloudWatch that record the failures of the external payment processing API call.
- E. Configure a CloudWatch alarm to notify the existing SNS topic when error rate exceeds the specified rate.
- F. Publish the results of the external payment processing API calls to a new Amazon SNS topic.
- G. Subscribe the support team members to the new SNS topic.
- H. Write the results of the external payment processing API calls to Amazon S3. Schedule an Amazon Athena query to run at regular interval.
- I. Configure Athena to send notifications to the existing SNS topic when the error rate exceeds the specified rate.

Answer: B

Explanation:

Amazon CloudWatch is a service that monitors AWS resources and applications. The developer can publish custom metrics to CloudWatch that record the failures of the external payment processing API calls. The developer can configure a CloudWatch alarm to notify the existing SNS topic when the error rate exceeds 5% of the total number of transactions in an hour. This solution will meet the requirements in a near real-time and scalable way.

References:

? [What Is Amazon CloudWatch? - Amazon CloudWatch]

? [Publishing Custom Metrics - Amazon CloudWatch]

? [Creating Amazon CloudWatch Alarms - Amazon CloudWatch]

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 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 10

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 15

A company has a multi-node Windows legacy application that runs on premises. The application uses a network shared folder as a centralized configuration repository to store configuration files in .xml format. The company is migrating the application to Amazon EC2 instances. As part of the migration to AWS, a developer must identify a solution that provides high availability for the repository.

Which solution will meet this requirement MOST cost-effectively?

- A. Mount an Amazon Elastic Block Store (Amazon EBS) volume onto one of the EC2 instances
- B. Deploy a file system on the EBS volume
- C. Use the host operating system to share a folder
- D. Update the application code to read and write configuration files from the shared folder.
- E. Deploy a micro EC2 instance with an instance store volume
- F. Use the host operating system to share a folder
- G. Update the application code to read and write configuration files from the shared folder.
- H. Create an Amazon S3 bucket to host the repository
- I. Migrate the existing .xml files to the S3 bucket
- J. Update the application code to use the AWS SDK to read and write configuration files from Amazon S3.
- K. Create an Amazon S3 bucket to host the repository
- L. Migrate the existing .xml files to the S3 bucket
- M. Mount the S3 bucket to the EC2 instances as a local volume
- N. Update the application code to read and write configuration files from the disk.

Answer: C

Explanation:

Amazon S3 is a service that provides highly scalable, durable, and secure object storage. The developer can create an S3 bucket to host the repository and migrate the existing .xml files to the S3 bucket. The developer can update the application code to use the AWS SDK to read and write configuration files from S3. This solution will meet the requirement of high availability for the repository in a cost-effective way.

References:

? [Amazon Simple Storage Service (S3)]

? [Using AWS SDKs with Amazon S3]

NEW QUESTION 19

A company has an application that is hosted on Amazon EC2 instances. The application stores objects in an Amazon S3 bucket and allows users to download objects from the S3 bucket. A developer turns on S3 Block Public Access for the S3 bucket. After this change, users report errors when they attempt to download objects. The developer needs to implement a solution so that only users who are signed in to the application can access objects in the S3 bucket.

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

- A. Create an EC2 instance profile and role with an appropriate policy. Associate the role with the EC2 instances.
- B. Create an IAM user with an appropriate policy.
- C. Store the access key ID and secret access key on the EC2 instances.
- D. Modify the application to use the S3 GeneratePresignedUrl API call.
- E. Modify the application to use the S3 GetObject API call and to return the object handle to the user.
- F. Modify the application to delegate requests to the S3 bucket.

Answer: AC

Explanation:

The most secure way to allow the EC2 instances to access the S3 bucket is to use an EC2 instance profile and role with an appropriate policy that grants the necessary permissions. This way, the EC2 instances can use temporary security credentials that are automatically rotated and do not need to store any access keys on the instances. To allow the users who are signed in to the application to download objects from the S3 bucket, the application can use the S3 GeneratePresignedUrl API call to create a pre-signed URL that grants temporary access to a specific object. The pre-signed URL can be returned to the user, who can then use it to download the object within a specified time period. References:

? Use Amazon S3 with Amazon EC2

? How to Access AWS S3 Bucket from EC2 Instance in a Secured Way

? Sharing an Object with Others

NEW QUESTION 24

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

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

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

Answer: D

Explanation:

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

? What Is AWS Secrets Manager? - AWS Secrets Manager

? Rotating Your AWS Secrets Manager Secrets - AWS Secrets Manager

? Retrieving a Secret - AWS Secrets Manager

NEW QUESTION 29

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

Which solution will meet these requirements?

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

Answer: D

Explanation:

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

? Create Metric Filters - Amazon CloudWatch Logs

? Creating Amazon CloudWatch Alarms - Amazon CloudWatch

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

NEW QUESTION 31

A developer is creating an AWS Lambda function that searches for Items from an Amazon DynamoDB table that contains customer contact information. The DynamoDB table items have the customers as the partition and additional properties such as customer_type, name, and job_title.

The Lambda function runs whenever a user types a new character into the customer_type text Input. The developer wants to search to return partial matches of all the email_address property of a particular customer type. The developer does not want to recreate the DynamoDB table.

What should the developer do to meet these requirements?

A. Add a global secondary index (GSI) to the DynamoDB table with customer_type input, as the partition key and email_address as the sort key

B. Perform a query operation on the GSI by using the begins_with key condition expression with the email_address property.

C. Add a global secondary index (GSI) to the DynamoDB table with email_address as the partition key and customer_type as the sort key

D. Perform a query operation on the GSI by using the begins_with key condition expression with the email_address property.

E. Address property.

F. Add a local secondary index (LSI) to the DynamoDB table with customer_type as the partition Key and email_address as the sort Key

G. Perform a query operation on the LSI by using the begins_with Key condition expression with the email_address property.

H. Add a local secondary index (LSI) to the DynamoDB table with job_title as the partition key and email_address as the sort key

I. Perform a query operation on the LSI by using the begins_with key condition expression with the email_address property.

Answer: A

Explanation:

The solution that will meet the requirements is to add a global secondary index (GSI) to the DynamoDB table with customer_type as the partition key and email_address as the sort key. Perform a query operation on the GSI by using the begins_with key condition expression with the email_address property. This way, the developer can search for partial matches of the email_address property of a particular customer type without recreating the DynamoDB table. The other options either involve using a local secondary index (LSI), which requires recreating the table, or using a different partition key, which does not allow filtering by customer_type.

Reference: Using Global Secondary Indexes in DynamoDB

NEW QUESTION 35

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 minutes

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 39

A developer has written the following IAM policy to provide access to an Amazon S3 bucket:


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/secrets*"
    }
  ]
}
```

Which access does the policy allow regarding the s3:GetObject and s3:PutObject actions?

- A. Access on all buckets except the "DOC-EXAMPLE-BUCKET" bucket
- B. Access on all buckets that start with "DOC-EXAMPLE-BUCKET" except the "DOC-EXAMPLE-BUCKET/secrets" bucket
- C. Access on all objects in the "DOC-EXAMPLE-BUCKET" bucket along with access to all S3 actions for objects in the "DOC-EXAMPLE-BUCKET" bucket that start with "secrets"
- D. Access on all objects in the "DOC-EXAMPLE-BUCKET" bucket except on objects that start with "secrets"

Answer: D

Explanation:

The IAM policy shown in the image is a resource-based policy that grants or denies access to an S3 bucket based on certain conditions. The first statement allows access to any S3 action on any object in the "DOC-EXAMPLE-BUCKET" bucket when the request is made over HTTPS (the value of aws:SecureTransport is true). The second statement denies access to the s3:GetObject and s3:PutObject actions on any object in the "DOC-EXAMPLE-BUCKET/secrets" prefix when the request is made over HTTP (the value of aws:SecureTransport is false). Therefore, the policy allows access on all objects in the "DOC-EXAMPLE-BUCKET" bucket except on objects that start with "secrets".

Reference: Using IAM policies for Amazon S3

NEW QUESTION 40

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.
- J. 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 41

A developer is working on an ecommerce platform that communicates with several third- party payment processing APIs The third-party payment services do not provide a test environment. The developer needs to validate the ecommerce platform's integration with the third-party payment processing APIs. The developer must test the API integration code without invoking the third-party payment processing APIs. Which solution will meet these requirements'?

- A. Set up an Amazon API Gateway REST API with a gateway response configured for status code 200 Add response templates that contain sample responses captured from the real third-party API.
- B. Set up an AWS AppSync GraphQL API with a data source configured for each third- party API Specify an integration type of Mock Configure integration responses by using sample responses captured from the real third-party API.
- C. Create an AWS Lambda function for each third-party AP
- D. Embed responses captured from the real third-party AP
- E. Configure Amazon Route 53 Resolver with an inbound endpoint for each Lambda function's Amazon Resource Name (ARN).
- F. Set up an Amazon API Gateway REST API for each third-party API Specify an integration request type of Mock Configure integration responses by using sample responses captured from the real third-party API

Answer: D

Explanation:

Amazon API Gateway can mock responses for testing purposes without requiring any integration backend. This allows the developer to test the API integration code without invoking the third-party payment processing APIs. The developer can configure integration responses by using sample responses captured from the real third- party API. References:

- ? Mocking Integration Responses in API Gateway
- ? Set up Mock Integrations for an API in API Gateway

NEW QUESTION 43

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

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

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

Answer: C

Explanation:

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

References: Amazon S3 Presigned URLs, Amazon S3 Lifecycle

NEW QUESTION 44

A 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 mad
- 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 mad
- 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 consol
- F. Select the removePii functio
- G. Use S3 Access Points to access the object without PII.
- H. Create an S3 access point from the S3 consol
- I. Use the access point name to call the GetObjectLegalHold S3 API functio
- 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 45

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 46

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.

The developer wants to make the REST API available for testing by using API Gateway locally.

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 AWS Serverless Application Model Command Line Interface (AWS SAM CLI) is a command-line tool for local development and testing of Serverless applications². The sam local start-api subcommand of AWS SAM CLI is used to simulate a REST API by starting a new local endpoint³. Therefore, option D is correct.

NEW QUESTION 48

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

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 50

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 51

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

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

Answer: A

Explanation:

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

NEW QUESTION 55

A developer is writing an application that will retrieve sensitive data from a third-party system. The application will format the data into a PDF file. The PDF file could be more than 1 MB. The application will encrypt the data to disk by using AWS Key Management Service (AWS KMS). The application will decrypt the file when a user requests to download it. The retrieval and formatting portions of the application are complete. The developer needs to use the GenerateDataKey API to encrypt the PDF file so that the PDF file can be decrypted later. The developer needs to use an AWS KMS symmetric customer managed key for encryption. Which solutions will meet these requirements?

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

Answer: A

Explanation:

? The GenerateDataKey API returns a data key that is encrypted under a symmetric encryption KMS key that you specify, and a plaintext copy of the same data key1. The data key is a random byte string that can be used with any standard encryption algorithm, such as AES or SM42. The plaintext data key can be used to encrypt or decrypt data outside of AWS KMS, while the encrypted data key can be stored with the encrypted data and later decrypted by AWS KMS1. ? In this scenario, the developer needs to use the GenerateDataKey API to encrypt the PDF file so that it can be decrypted later. The developer also needs to use an AWS KMS symmetric customer managed key for encryption. To achieve this, the developer can follow these steps:

NEW QUESTION 58

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

How can a developer configure access to the S3 bucket in the MOST secure way?

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

Answer: C

Explanation:

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

NEW QUESTION 59

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 61

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

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

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

Answer: C

Explanation:

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

NEW QUESTION 66

A company wants to deploy and maintain static websites on AWS. Each website's source code is hosted in one of several version control systems, including AWS CodeCommit, Bitbucket, and GitHub.

The company wants to implement phased releases by using development, staging, user acceptance testing, and production environments in the AWS Cloud. Deployments to each environment must be started by code merges on the relevant Git branch. The company wants to use HTTPS for all data exchange. The company needs a solution that does not require servers to run continuously.

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

- A. Host each website by using AWS Amplify with a serverless backend
- B. Connect the repository branches that correspond to each of the desired environments
- C. Start deployments by merging code changes to a desired branch.
- D. Host each website in AWS Elastic Beanstalk with multiple environments

- E. Use the EB CLI to link each repository branch
- F. Integrate AWS CodePipeline to automate deployments from version control code merges.
- G. Host each website in different Amazon S3 buckets for each environment
- H. Configure AWS CodePipeline to pull source code from version control
- I. Add an AWS CodeBuild stage to copy source code to Amazon S3.
- J. Host each website on its own Amazon EC2 instance
- K. Write a custom deployment script to bundle each website's static assets
- L. Copy the assets to Amazon EC2. Set up a workflow to run the script when code is merged.

Answer: A

Explanation:

AWS Amplify is a set of tools and services that enables developers to build and deploy full-stack web and mobile applications that are powered by AWS. AWS Amplify supports hosting static websites on Amazon S3 and Amazon CloudFront, with HTTPS enabled by default. AWS Amplify also integrates with various version control systems, such as AWS CodeCommit, Bitbucket, and GitHub, and allows developers to connect different branches to different environments. AWS Amplify automatically builds and deploys the website whenever code changes are merged to a connected branch, enabling phased releases with minimal operational overhead. Reference: AWS Amplify Console

NEW QUESTION 70

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 SecureString
- 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. Set up 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 73

A developer is using AWS Amplify Hosting to build and deploy an application. The developer is receiving an increased number of bug reports from users. The developer wants to add end-to-end testing to the application to eliminate as many bugs as possible before the bugs reach production.

Which solution should the developer implement to meet these requirements?

- A. Run the amplify add test command in the Amplify CLI.
- B. Create unit tests in the application
- C. Deploy the unit tests by using the amplify push command in the Amplify CLI.
- D. Add a test phase to the amplify.yml build settings for the application.
- E. Add a test phase to the aws-exports.js file for the application.

Answer: C

Explanation:

The solution that will meet the requirements is to add a test phase to the amplify.yml build settings for the application. This way, the developer can run end-to-end tests on every code commit and catch any bugs before deploying to production. The other options either do not support end-to-end testing, or do not run tests automatically.

Reference: End-to-end testing

NEW QUESTION 76

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 78

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 80

A developer is planning to migrate on-premises company data to Amazon S3. The data must be encrypted, and the encryption keys must support automatic annual rotation. The company must use AWS Key Management Service (AWS KMS) to encrypt the data.

When type of keys should the developer use to meet these requirements?

- A. Amazon S3 managed keys
- B. Symmetric customer managed keys with key material that is generated by AWS
- C. Asymmetric customer managed keys with key material that generated by AWS
- D. Symmetric customer managed keys with imported key material

Answer: B

Explanation:

The type of keys that the developer should use to meet the requirements is symmetric customer managed keys with key material that is generated by AWS. This way, the developer can use AWS Key Management Service (AWS KMS) to encrypt the data with a symmetric key that is managed by the developer. The developer can also enable automatic annual rotation for the key, which creates new key material for the key every year. The other options either involve using Amazon S3 managed keys, which do not support automatic annual rotation, or using asymmetric keys or imported key material, which are not supported by S3 encryption.

Reference: Using AWS KMS keys to encrypt S3 objects

NEW QUESTION 84

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

When solution will meet these requirements?

- A. Use a single stage in API Gateway
- B. Create a Lambda function for each environment
- C. Configure API clients to send a query parameter that indicates the environment and the specific lambda function.
- D. Use multiple stages in API Gateway
- E. Create a single Lambda function for all environment
- F. Add different code blocks for different environments in the Lambda function based on Lambda environment variables.
- G. Use multiple stages in API Gateway
- H. Create a Lambda function for each environment
- I. Configure API Gateway stage variables to route traffic to the Lambda function in different environments.
- J. Use a single stage in API Gateway
- K. Configure a API client to send a query parameter that indicated the environment
- L. Add different code blocks for different environments in the Lambda function to match the value of the query parameter.

Answer: C

Explanation:

The solution that will meet the requirements is to use multiple stages in API Gateway. Create a Lambda function for each environment. Configure API Gateway stage variables to route traffic to the Lambda function in different environments. This way, the company can test the code in a dedicated, monitored test environment before releasing it to the production environment. The company can also use stage variables to specify the Lambda function version or alias for each stage, and avoid hard-coding the Lambda function name in the API Gateway integration. The other options either involve using a single stage in API Gateway, which does not allow testing in different environments, or adding different code blocks for different environments in the Lambda function, which increases complexity and maintenance.

Reference: Set up stage variables for a REST API in API Gateway

NEW QUESTION 89

A developer is working on an ecommerce website. The developer wants to review server logs without logging in to each of the application servers individually. The website runs on multiple Amazon EC2 instances, is written in Python, and needs to be highly available.

How can the developer update the application to meet these requirements with MINIMUM changes?

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

Answer: D

Explanation:

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

? Using the CloudWatch Agent

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

NEW QUESTION 94

A developer is building a web application that uses Amazon API Gateway to expose an AWS Lambda function to process requests from clients. During testing, the developer notices that the API Gateway times out even though the Lambda function finishes under the set time limit.

Which of the following API Gateway metrics in Amazon CloudWatch can help the developer troubleshoot the issue? (Choose two.)

- A. CacheHitCount
- B. IntegrationLatency
- C. CacheMissCount
- D. Latency
- E. Count

Answer: BD

Explanation:

Amazon API Gateway is a service that enables developers to create, publish, maintain, monitor, and secure APIs at any scale. Amazon CloudWatch is a service that monitors AWS resources and applications. API Gateway provides several CloudWatch metrics to help developers troubleshoot issues with their APIs. Two of the metrics that can help the developer troubleshoot the issue of API Gateway timing out are:

? IntegrationLatency: This metric measures the time between when API Gateway

relays a request to the backend and when it receives a response from the backend. A high value for this metric indicates that the backend is taking too long to respond and may cause API Gateway to time out.

? Latency: This metric measures the time between when API Gateway receives a

request from a client and when it returns a response to the client. A high value for this metric indicates that either the integration latency is high or API Gateway is taking too long to process the request or response.

References:

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

? [Amazon API Gateway Metrics and Dimensions - Amazon CloudWatch]

? [Troubleshooting API Errors - Amazon API Gateway]

NEW QUESTION 97

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 99

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 104

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

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

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

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

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

Answer: A

Explanation:

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

References:

? [Amazon DynamoDB]

? [Working with Queries - Amazon DynamoDB]

? [Working with Global Secondary Indexes - Amazon DynamoDB]

NEW QUESTION 105

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 107

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

What should the developer do to meet these requirements?

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

GetSecretValue Secrets Manager API operation requests.

Answer: B

Explanation:

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

NEW QUESTION 112

A developer wants to insert a record into an Amazon DynamoDB table as soon as a new file is added to an Amazon S3 bucket.

Which set of steps would be necessary to achieve this?

- A. Create an event with Amazon EventBridge that will monitor the S3 bucket and then insert the records into DynamoDB.
- B. Configure an S3 event to invoke an AWS Lambda function that inserts records into DynamoDB.
- C. Create an AWS Lambda function that will poll the S3 bucket and then insert the records into DynamoDB.
- D. Create a cron job that will run at a scheduled time and insert the records into DynamoDB.

Answer: B

Explanation:

Amazon S3 is a service that provides highly scalable, durable, and secure object storage. Amazon DynamoDB is a fully managed NoSQL database service that

provides fast and consistent performance with seamless scalability. AWS Lambda is a service that lets developers run code without provisioning or managing servers. The developer can configure an S3 event to invoke a Lambda function that inserts records into DynamoDB whenever a new file is added to the S3 bucket. This solution will meet the requirement of inserting a record into DynamoDB as soon as a new file is added to S3. References:

? [Amazon Simple Storage Service (S3)]

? [Amazon DynamoDB]

? [What Is AWS Lambda? - AWS Lambda]

? [Using AWS Lambda with Amazon S3 - AWS Lambda]

NEW QUESTION 115

Users are reporting errors in an application. The application consists of several micro services that are deployed on Amazon Elastic Container Services (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

- 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 SDK
- H. Update the application to communicate with the X-Ray daemon.
- I. Instrument the ECS task to send the stdout and stderr 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 120

A developer is modifying an existing AWS Lambda function. While checking the code, the developer notices hardcoded parameter values for an Amazon RDS for SQL Server user name, password, database, host, and port. There also are hardcoded parameter values for an Amazon DynamoDB table, an Amazon S3 bucket, and an Amazon Simple Notification Service (Amazon SNS) topic. The developer wants to securely store the parameter values outside the code in an encrypted format and wants to turn on rotation for the credentials. The developer also wants to be able to reuse the parameter values from other applications and to update the parameter values without modifying code. Which solution will meet these requirements with the LEAST operational overhead?

- A. Create an RDS database secret in AWS Secrets Manager.
- B. Set the user name, password, database, host, and port.
- C. Turn on secret rotation.
- D. Create encrypted Lambda environment variables for the DynamoDB table, S3 bucket, and SNS topic.
- E. Create an RDS database secret in AWS Secrets Manager.
- F. Set the user name, password, database, host, and port.
- G. Turn on secret rotation.
- H. Create Secure String parameters in AWS Systems Manager Parameter Store for the DynamoDB table, S3 bucket, and SNS topic.
- I. Create RDS database parameters in AWS Systems Manager Parameter Store.
- J. Store the user name, password, database, host, and port.
- K. Create encrypted Lambda environment variables for the DynamoDB table, S3 bucket, and SNS topic.
- L. Create a Lambda function and set the logic for the credentials rotation. Schedule the credentials rotation task in Amazon EventBridge.
- M. Create RDS database parameters in AWS Systems Manager Parameter Store.
- N. Store the user name, password, database, host, and port.
- O. Store the DynamoDB table, S3 bucket, and SNS topic in Amazon S3.
- P. Create a Lambda function and set the logic for the credentials rotation. Invoke the Lambda function on a schedule.

Answer: B

Explanation:

This solution will meet the requirements by using AWS Secrets Manager and AWS Systems Manager Parameter Store to securely store the parameter values outside the code in an encrypted format. AWS Secrets Manager 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 RDS database secret in AWS Secrets Manager and set the user name, password, database, host, and port for accessing the RDS database. The developer can also turn on secret rotation, which will change the database credentials periodically according to a specified schedule or event. AWS Systems Manager Parameter Store is a service that provides secure and scalable storage for configuration data and secrets. The developer can create Secure String parameters in AWS Systems Manager Parameter Store for the DynamoDB table, S3 bucket, and SNS topic, which will encrypt them with AWS KMS. The developer can also reuse the parameter values from other applications and update them without modifying code. Option A is not optimal because it will create encrypted Lambda

environment variables for the DynamoDB table, S3 bucket, and SNS topic, which may not be reusable or updatable without modifying code. Option C is not optimal because it will create RDS database parameters in AWS Systems Manager Parameter Store, which does not support automatic rotation of secrets. Option D is not optimal because it will store the DynamoDB table, S3 bucket, and SNS topic in Amazon S3, which may introduce additional costs and complexity for accessing configuration data. References: AWS Secrets Manager, [AWS Systems Manager Parameter Store]

NEW QUESTION 123

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 124

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 125

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

Which solution will meet these requirements MOST cost-effectively?

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

Answer: C

Explanation:

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

NEW QUESTION 128

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 132

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

Which solution will meet these requirements?

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

Answer: D

Explanation:

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

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

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

NEW QUESTION 133

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 137

A company built a new application in the AWS Cloud. The company automated the bootstrapping of new resources with an Auto Scaling group by using AWS CloudFormation templates. The bootstrap scripts contain sensitive data. The company needs a solution that is integrated with CloudFormation to manage the sensitive data in the bootstrap scripts. Which solution will meet these requirements in the MOST secure way?

- A. Put the sensitive data into a CloudFormation parameter
- B. Encrypt the CloudFormation templates by using an AWS Key Management Service (AWS KMS) key.
- C. Put the sensitive data into an Amazon S3 bucket. Update the CloudFormation templates to download the object from Amazon S3 during bootstrap.
- D. Put the sensitive data into AWS Systems Manager Parameter Store as a secure string parameter.
- E. Update the CloudFormation templates to use dynamic references to specify template values.
- F. Put the sensitive data into Amazon Elastic File System (Amazon EFS). Enforce EFS encryption after file system creation.
- G. Update the CloudFormation templates to retrieve data from Amazon EFS.

Answer: C

Explanation:

This solution meets the requirements in the most secure way because it uses a service that is integrated with CloudFormation to manage sensitive data in encrypted form. AWS Systems Manager Parameter Store provides secure, hierarchical storage for configuration data management and secrets management. You can store sensitive data as secure string parameters, which are encrypted using an AWS Key Management Service (AWS KMS) key of your choice. You can also use dynamic references in your CloudFormation templates to specify template values that are stored in Parameter Store or Secrets Manager without having to include them in your templates. Dynamic references are resolved only during stack creation or update operations, which reduces exposure risks for sensitive data. Putting sensitive data into a CloudFormation parameter will not encrypt them or protect them from unauthorized access. Putting sensitive data into an Amazon S3 bucket or Amazon Elastic File System (Amazon EFS) will require additional configuration and integration with CloudFormation and may not provide fine-grained access control or encryption for sensitive data.

Reference: [What Is AWS Systems Manager Parameter Store?], [Using Dynamic References to Specify Template Values]

NEW QUESTION 139

.....

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](#)