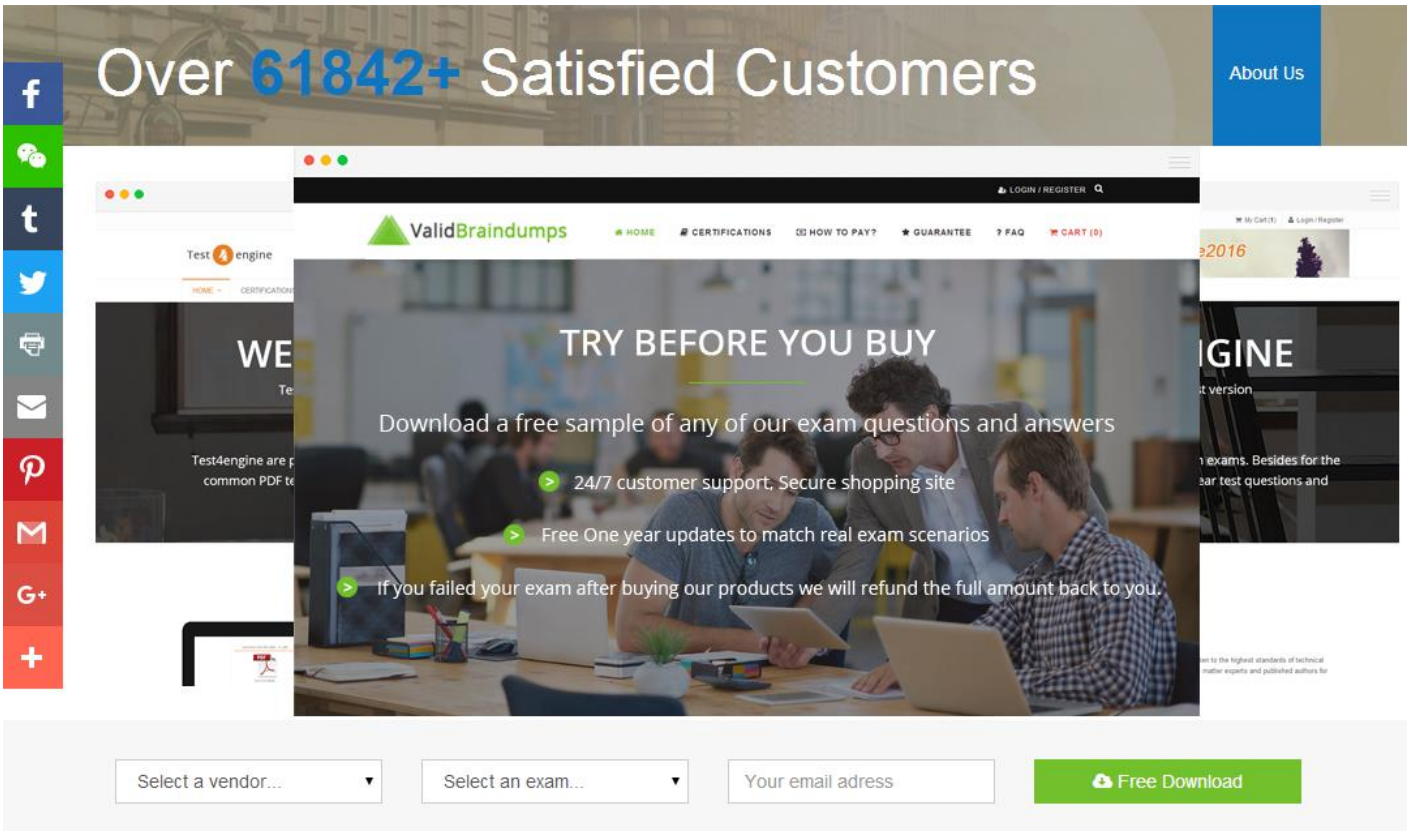
ValidBraindumps



Try Before You Buy

Download a free sample of any of our exam questions and answers

- 24/7 customer support, Secure shopping site
- Free One year updates to match real exam scenarios
- If you failed your exam after buying our products we will refund the full amount back to you.

http://www.validbraindumps.com

Free valid test braindumps for IT certification valid exam

Exam : **AWS-Developer**

Title : AWS Certified Developer Associate Exam (DVA-C02)

Vendor : Amazon

Version : DEMO

NO.1 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
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]

NO.2 A company has an online web application that includes a product catalog. The catalog is stored in an Amazon S3 bucket that is named DOC-EXAMPLE-BUCKET. The application must be able to list the objects in the S3 bucket and must be able to download objects through an 1AM policy.
Which policy allows MINIMUM access to meet these requirements?

B.

```
{
  "Effect": "Allow",
  "Action": [
    "s3:*"
  ],
  "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
}
]
}
```

C.

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

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action":"s3:ListBucket",
      "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
    },
```

A. Option A
B. Option B
C. Option C
D. Option D
Answer: A

NO.3 A developer is writing an application to analyze the traffic to a fleet of Amazon EC2 instances. The EC2 instances run behind a public Application Load Balancer (ALB). An HTTP server runs on each of the EC2 instances, logging all requests to a log file.
The developer wants to capture the client public IP addresses. The developer analyzes the log files and notices only the IP address of the ALB.
What must the developer do to capture the client public IP addresses in the log file?
A. Add a Host header to the HTTP server log configuration file.
B. Install the Amazon CloudWatch Logs agent on each EC2 instance. Configure the agent to write to the log file.
C. Install the AWS X-Ray daemon on each EC2 instance. Configure the daemon to write to the log file.
D. Add an X-Forwarded-For header to the HTTP server log configuration file.
Answer: D

NO.4 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. Create a new API. Import the OpenAPI file. Modify the new API to add request validation. Perform the tests. Modify the existing API to add request validation.
Deploy the existing API to production.
B. Modify the existing API to add request validation. Deploy the updated API to a new API Gateway stage. Perform the tests. Deploy the updated API to the API Gateway production stage.
C. Create a new API. Add the necessary resources and methods, including new request validation. Perform the tests. Modify the existing API to add request validation. Deploy the existing API to production.
D. Clone the existing API. Modify the new API to add request validation. Perform the tests. Modify the existing API to add request validation. Deploy the existing API to production.
Answer: B
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.

NO.5 A company runs an application on AWS The application stores data in an Amazon DynamoDB table Some queries are taking a long time to run These slow queries involve an attribute that is not

the table's partition key or sort key The amount of data that the application stores in the DynamoDB table is expected to increase significantly. A developer must increase the performance of the queries. Which solution will meet these requirements'?

A. Increase the page size for each request by setting the Limit parameter to be higher than the default value Configure the application to retry any request that exceeds the provisioned throughput.

B. Create a global secondary index (GSI). Set query attribute to be the partition key of the index

C. Perform a parallel scan operation by issuing individual scan requests in the parameters specify the segment for the scan requests and the total number of segments for the parallel scan.

D. Turn on read capacity auto scaling for the DynamoDB table. Increase the maximum read capacity units (RCUs).

Answer: B

* Global Secondary Index (GSI): GSIs enable alternative query patterns on a DynamoDB table by using different partition and sort keys.
* Addressing Query Bottleneck: By making the slow-query attribute the GSI's partition key, you optimize queries on that attribute.
* Scalability: GSIs automatically scale to handle increasing data volumes.

References:
Amazon DynamoDB Global Secondary Indexes:
https://docs.aws.amazon.com/amazondynamodb/latest
/developerguide/GSI.html

NO.6 A team is developing an application that is deployed on Amazon EC2 instances. During testing, the team receives an error. The EC2 instances are unable to access an Amazon S3 bucket.
Which steps should the team take to troubleshoot this issue? (Select TWO.)

A. Check whether the policy that is assigned to the JAM role that is attached to the EC2 instances grants access to Amazon S3.

B. Check the S3 bucket policy to validate the access permissions for the S3 bucket.

C. Check whether the policy that is assigned to the 1AM user that is attached to the EC2 instances grants access to Amazon S3.

D. Check the S3 Lifecycle policy to validate the permissions that are assigned to the S3 bucket.

E. Check the security groups that are assigned to the EC2 instances. Make sure that a rule is not blocking the access to Amazon S3.

Answer: A,B

NO.7 A company is creating an AWS Step Functions state machine to run a set of tests for an application. The tests need to run when a specific AWS Cloud Formation stack is deployed.
Which combination of steps will meet these requirements? (Select TWO.)

A. Create an AWS Lambda function to invoke the state machine.

B. Create an Amazon EventBridge rule on the default bus that matches on a detail type of CloudFormation stack status change, a status of UPDATE_IN_PROGRESS, and the stack ID of the CloudFormation stack.

C. Create a pipe in Amazon EventBridge Pipes that has a source of the default event bus. Set the Lambda function as a target. Filter on a detail type of CloudFormation stack status change, a status of UPDATE_IN_PROGRESS, and the stack ID of the CloudFormation stack.

D. Create a pipe in Amazon EventBridge Pipes that has a source of the EventBridge rule. Set the state machine as a target.

E. Add the state machine as a target of the EventBridge rule.

Answer: A,E

Requirement Summary:

* Trigger an AWS Step Functions state machine (test execution)
* Only when a specific AWS CloudFormation stack is deployed

Option A: Create a Lambda function to invoke the state machine

* # Valid approach: Lambda can be used as an intermediary trigger for Step Functions using the SDK (e.g., StartExecution API).
* Offers flexibility (custom filtering, additional logic).

Option B: Create EventBridge rule filtering on UPDATE_IN_PROGRESS

* # Incorrect: UPDATE_IN_PROGRESS triggers before the stack is fully deployed.
* You need to trigger after deployment, such as UPDATE_COMPLETE or CREATE_COMPLETE.

Option C: EventBridge Pipes with Lambda target filtering on UPDATE_IN_PROGRESS

* # Incorrect for same reason as B (wrong timing).
* Also, EventBridge Pipes are not necessary here if you're using rules directly.

Option D: Pipe with EventBridge Rule as source and Step Functions as target

* # Invalid setup: EventBridge Pipes use event sources, not rules, as input.
* This configuration is unsupported.

Option E: Add the state machine as a target of the EventBridge rule

* # Direct and low-overhead approach.
* EventBridge natively supports Step Functions as a target.
* You can trigger the state machine without a Lambda if the filter matches (e.g., ResourceStatus = CREATE_COMPLETE, with the correct StackId).
* Step Functions as EventBridge target: https://docs.aws.amazon.com/eventbridge/latest/userguide/eventbridge-target-step-functions.html
* EventBridge CloudFormation events: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-cfn-listing-event-history.html
* StartExecution API: https://docs.aws.amazon.com/step-functions/latest/apireference/API_StartExecution.html

NO.8 A developer has a financial application. The application uses AWS Secrets Manager to manage an Amazon RDS for PostgreSQL database's username and password. The developer needs to rotate the password while maintaining the application's high availability. Which solution will meet these requirements with LEAST development effort?

A. Rotate the secret by using the alternating-users rotation strategy. Update the application with an appropriate retry strategy to handle authentication failures.

B. Use the PostgreSQL client to create a new database username and password. Include the new secret values by performing an immediate rotation. Use the AWS CLI to update the RDS database password.

Perform an immediate rotation of the Secrets Manager secrets.

C. Rotate the secret by using multivalue answer rotation. Update the application with an appropriate retry strategy to handle authentication failures.

D. Rotate the secret by using the single-user rotation strategy. Update the application with an

appropriate retry strategy to handle authentication failures.

Answer: D

Requirement Summary:
* Secrets managed in AWS Secrets Manager
* DB: Amazon RDS for PostgreSQL
* Need automated password rotation
* Must maintain high availability
* Least development effort

Rotation Strategies:
Single-user rotation strategy
* # Simplest to implement
* The secret contains one set of credentials used by app and rotation logic
* # Supports automated rotation
* AWS provides built-in Lambda rotation templates for RDS

A). Alternating-users strategy
* ## More complex
* Requires application to switch users during rotation window

B). Manual secret + CLI rotation
* # Too much manual work
* Not scalable or reliable

C). Multivalue answer rotation
* # Not a valid strategy in this context
* Doesn't apply to Secrets Manager
* Secrets Manager rotation strategies:
https://docs.aws.amazon.com/secretsmanager/latest/userguide
/rotating-secrets.html
* RDS PostgreSQL secret rotation: https://docs.aws.amazon.com/secretsmanager/latest/userguide
/rotating-secrets_strategies.html#rotating-secrets-single-user

NO.9 A company caches session information for a web application in an Amazon DynamoDB table. The company wants an automated way to delete old items from the table.
What is the simplest way to do this?

A. Write a script that deletes old records; schedule the script as a cron job on an Amazon EC2 instance.

B. Add an attribute with the expiration time; enable the Time To Live feature based on that attribute.

C. Each day, create a new table to hold session data; delete the previous day's table.

D. Add an attribute with the expiration time; name the attribute ItemExpiration.

Answer: B

NO.10 A developer migrated a legacy application to an AWS Lambda function. The function uses a third-party service to pull data with a series of API calls at the end of each month. The function than processes the data to generate the monthly reports. The function has Been working with no issues so far.
The third-party service recently issued a restriction to allow a feed number to API calls each minute and each day. If the API calls exceed the limit tor each minute or each day, then the service will produce errors. The API also provides the minute limit and daily limit in the response header. This

restriction might extend the overall process to multiple days because the process is consuming more API calls than the available limit.
What is the MOST operationally efficient way to refactor the server less application to accommodate this change?

A. Use an AWS Step Functions State machine to monitor API failures. Use the Wait state to delay calling the Lambda function.

B. Use an Amazon Simple Queue Service (Amazon SQS) queue to hold the API calls. Configure the Lambda function to poll the queue within the API threshold limits.

C. Use an Amazon CloudWatch Logs metric to count the number of API calls. Configure an Amazon CloudWatch alarm flat slops the currently running instance of the Lambda function when the metric exceeds the API threshold limits.

D. Use Amazon Kinesis Data Firehose to batch me API calls and deliver them to an Amazon S3 bucket win an event notification to invoke the Lambda function.

Answer: A

The solution that will meet the requirements is to use an AWS Step Functions state machine to monitor API failures. Use the Wait state to delay calling the Lambda function. This way, the developer can refactor the serverless application to accommodate the change in a way that is automated and scalable. The developer can use Step Functions to orchestrate the Lambda function and handle any errors or retries. The developer can also use the Wait state to pause the execution for a specified duration or until a specified timestamp, which can help avoid exceeding the API limits. The other options either involve using additional services that are not necessary or appropriate for this scenario, or do not address the issue of API failures.
Reference: AWS Step Functions Wait state

NO.11 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 Aversion of the sensitive credentials need 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

D. Configure AWS Secrets Manager to create a new secret for each environment type. Store the environment-specific credentials in the secret

Answer: D

* Secrets Management: AWS Secrets Manager is designed specifically for storing and managing sensitive credentials.
* Environment Isolation: Creating separate secrets for each environment (development, staging, etc.) ensures clear separation and prevents accidental leaks.
* Automatic Rotation: Secrets Manager provides built-in rotation capabilities, enhancing security posture.

* Versioning: Tracking changes to secrets is essential for auditing and compliance.
References:
AWS Secrets Manager: https://aws.amazon.com/secrets-manager/
Secrets Manager Rotation: https://docs.aws.amazon.com/secretsmanager/latest/userguide/rotating-secrets.html

NO.12 A company is developing a serverless application by using AWS Lambda functions. One of the Lambda functions needs to access an Amazon RDS DB instance. The DB instance is in a private subnet inside a VPC.
The company creates a role that includes the necessary permissions to access the DB instance. The company then assigns the role to the Lambda function. A developer must take additional action to give the Lambda function access to the DB instance.
What should the developer do to meet these requirements?
A. Assign a public IP address to the DB instance. Modify the security group of the DB instance to allow inbound traffic from the IP address of the Lambda function.
B. Set up an AWS Direct Connect connection between the Lambda function and the DB instance.
C. Configure an Amazon CloudFront distribution to create a secure connection between the Lambda function and the DB instance.
D. Configure the Lambda function to connect to the private subnets in the VPC. Add security group rules to allow traffic to the DB instance from the Lambda function.
Answer: D

NO.13 A developer is deploying a new application to Amazon Elastic Container Service (Amazon ECS). The developer needs to securely store and retrieve different types of variables. These variables include authentication information for a remote API, the URL for the API, and credentials. The authentication information and API URL must be available to all current and future deployed versions of the application across development, testing, and production environments.
How should the developer retrieve the variables with the FEWEST application changes?
A. Update the application to retrieve the variables from AWS Systems Manager Parameter Store. Use unique paths in Parameter Store for each variable in each environment. Store the credentials in AWS Secrets Manager in each environment.
B. Update the application to retrieve the variables from AWS Key Management Service (AWS KMS). Store the API URL and credentials as unique keys for each environment.
C. Update the application to retrieve the variables from an encrypted file that is stored with the application.
Store the API URL and credentials in unique files for each environment.
D. Update the application to retrieve the variables from each of the deployed environments. Define the authentication information and API URL in the ECS task definition as unique names during the deployment process.
Answer: A
AWS Systems Manager Parameter Store is a service that provides secure, hierarchical storage for configuration data management and secrets management. The developer can update the application to retrieve the variables from Parameter Store by using the AWS SDK or the AWS CLI. The developer can use unique paths in Parameter Store for each variable in each environment, such as /dev/api-url, /test/api-url, and /prod

/api-url. The developer can also store the credentials in AWS Secrets Manager, which is integrated with Parameter Store and provides additional features such as automatic rotation and encryption.
References:
[What Is AWS Systems Manager? - AWS Systems Manager]
[Parameter Store - AWS Systems Manager]
[What Is AWS Secrets Manager? - AWS Secrets Manager]

NO.14 A developer is building a web and mobile application for two types of users: regular users and guest users. Regular users are required to log in, but guest users do not log in. Users should see only their data, regardless of whether they authenticate. Users need AWS credentials before they can access AWS resources.

A. Use an Amazon Cognito identity pool to generate temporary AWS credentials that are linked to an unauthenticated role that has access to the required resources.

B. Set up an IAM user that has permissions to the required resources. Hardcode the IAM credentials in the web and mobile application.

C. Generate temporary keys that are stored in AWS KMS. Use the temporary keys to access the required resources.

D. Generate temporary credentials. Store the temporary credentials in AWS Secrets Manager. Use the temporary credentials to access the required resources.

Answer: A

Comprehensive and Detailed Step-by-Step Explanation:
* Option A: Amazon Cognito Identity Pool with Unauthenticated Role
* Cognito identity pools can generate temporary AWS credentials for both authenticated and unauthenticated users.
* For guest users, Cognito assigns an unauthenticated role with limited permissions, ensuring secure access to only their resources.
* This is the most secure and efficient solution for managing AWS credentials dynamically without hardcoding or storing them.
* Why Other Options Are Incorrect:
* Option B: Hardcoding IAM credentials in the application is insecure and violates best practices.
* Option C and D: Temporary keys stored in KMS or Secrets Manager require additional implementation overhead and do not inherently manage user-specific access.
References:
Amazon Cognito Identity Pools

NO.15 A developer is creating an AWS Lambda function that needs network access to private resources in a VPC.

A. Attach the Lambda function to the VPC through private subnets. Create a security group that allows network access to the private resources. Associate the security group with the Lambda function.

B. Configure the Lambda function to route traffic through a VPN connection. Create a security group that allows network access to the private resources. Associate the security group with the Lambda function.

C. Configure a VPC endpoint connection for the Lambda function. Set up the VPC endpoint to route traffic through a NAT gateway.

D. Configure an AWS PrivateLink endpoint for the private resources. Configure the Lambda function to reference the PrivateLink endpoint.

Answer: A

Comprehensive Detailed Step by Step Explanation with All AWS Developer References:

When you need to provide an AWS Lambda function access to private resources in a VPC, the most common and straightforward approach is to attach the Lambda function to a VPC via private subnets. Once the Lambda function is associated with the VPC, you need to configure appropriate security groups to control the access to the private resources.

* Lambda with VPC Access: Lambda functions can be attached to private subnets in a VPC, allowing them to access resources like RDS, EC2, or internal services within that VPC.
* Security Groups: A security group acts as a virtual firewall for the Lambda function, ensuring that it can access only the necessary resources and ports in the VPC.
* Alternatives:
* Option B involves routing traffic through a VPN, which adds unnecessary complexity and operational overhead compared to simply attaching the Lambda to the VPC.
* Option C requires configuring a VPC endpoint and a NAT gateway, which can be complex and costly.
* Option D refers to AWS PrivateLink, which is used to access services over private connections, but it's unnecessary in this scenario unless you need a cross-VPC connection.

:

Lambda functions in a VPC

NO.16 A developer at a company needs to create a small application that makes the same API call once each day at a designated time. The company does not have infrastructure in the AWS Cloud yet, but the company wants to implement this functionality on AWS.

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

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

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

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

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

Answer: C

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

NO.17 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. Mount the S3 bucket folder in the Lambda function Import the library by using the proper folder in the mount point.

C. Load the external library to the Lambda function's /tmp directory during deployment of the Lambda package. Import the library from the /tmp directory.

D. Create an Amazon Elastic File System (Amazon EFS) volume. Upload the external library to the EFS volume Mount the EFS volume in the Lambda function. Import the library by using the proper folder in the mount point.

Answer: A

* Lambda Layers: These are designed to package dependencies that you can share across functions.
* How to Use:

* Create a layer, upload your 100MB library as a zip.
* Attach the layer to your function.
* In your function code, import the library from the standard layer path.
References:
Lambda Layers: https://docs.aws.amazon.com/lambda/latest/dg/configuration-layers.html

NO.18 A developer manages a website that distributes its content by using Amazon CloudFront. The website's static artifacts are stored in an Amazon S3 bucket.
The developer deploys some changes and can see the new artifacts in the S3 bucket. However, the changes do not appear on the webpage that the CloudFront distribution delivers.
How should the developer resolve this issue?
A. Configure S3 Object Lock to update to the latest version of the files every time an S3 object is updated.
B. Configure the S3 bucket to clear all old objects from the bucket before new artifacts are uploaded.
C. Set CloudFront to invalidate the cache after the artifacts have been deployed to Amazon S3.
D. Set CloudFront to modify the distribution origin after the artifacts have been deployed to Amazon S3.
Answer: C

NO.19 An application that is hosted on an Amazon EC2 instance needs access to files that are stored in an Amazon S3 bucket. The application lists the objects that are stored in the S3 bucket and displays a table to the user.
During testing, a developer discovers that the application does not show any objects in the list.
What is the MOST secure way to resolve this issue?
A. Update the IAM instance profile that is attached to the EC2 instance to include the S3:* permission for the S3 bucket.
B. Update the IAM instance profile that is attached to the EC2 instance to include the S3:ListBucket permission for the S3 bucket.
C. Update the developer's user permissions to include the S3:ListBucket permission for the S3 bucket .
D. Update the S3 bucket policy by including the S3:ListBucket permission and by setting the Principal element to specify the account number of the EC2 instance.
Answer: B
IAM instance profiles are containers for IAM roles that can be associated with EC2 instances. An IAM role is a set of permissions that grant access to AWS resources. An IAM role can be used to allow an EC2 instance to access an S3 bucket by including the appropriate permissions in the role's policy. The S3:ListBucket permission allows listing the objects in an S3 bucket. By updating the IAM instance profile with this permission, the application on the EC2 instance can retrieve the objects from the S3 bucket and display them to the user. Reference: Using an IAM role to grant permissions to applications running on Amazon EC2 instances

NO.20 A large company has its application components distributed across multiple AWS accounts. The company needs to collect and visualize trace data across these accounts.
What should be used to meet these requirements?
A. AWS X-Ray

B. Amazon CloudWatch
C. Amazon VPC flow logs
D. Amazon OpenSearch Service
Answer: A

NO.21 A developer wants the ability to roll back to a previous version of an AWS Lambda function in the event of errors caused by a new deployment. How can the developer achieve this with MINIMAL impact on users?
A. Change the application to use an alias that points to the current version. Deploy the new version of the code Update the alias to use the newly deployed version. If too many errors are encountered, point the alias back to the previous version.
B. Change the application to use an alias that points to the current version. Deploy the new version of the code. Update the alias to direct 10% of users to the newly deployed version. If too many errors are encountered, send 100% of traffic to the previous version
C. Do not make any changes to the application. Deploy the new version of the code. If too many errors are encountered, point the application back to the previous version using the version number in the Amazon Resource Name (ARN).
D. Create three aliases: new, existing, and router. Point the existing alias to the current version. Have the router alias direct 100% of users to the existing alias. Update the application to use the router alias.
Deploy the new version of the code. Point the new alias to this version. Update the router alias to direct
10% of users to the new alias. If too many errors are encountered, send 100% of traffic to the existing alias.
Answer: A

NO.22 A company is building a scalable data management solution by using AWS services to improve the speed and agility of development. The solution will ingest large volumes of data from various sources and will process this data through multiple business rules and transformations.
The solution requires business rules to run in sequence and to handle reprocessing of data if errors occur when the business rules run. The company needs the solution to be scalable and to require the least possible maintenance.
Which AWS service should the company use to manage and automate the orchestration of the data flows to meet these requirements?
A. AWS Batch
B. AWS Step Functions
C. AWS Glue
D. AWS Lambda
Answer: B
https://docs.aws.amazon.com/step-functions/latest/dg/welcome.html

NO.23 A company needs to distribute firmware updates to its customers around the world.
Which service will allow easy and secure control of the access to the downloads at the lowest cost?
A. Use Amazon CloudFront with signed URLs for Amazon S3.

B. Create a dedicated Amazon CloudFront Distribution for each customer.
C. Use Amazon CloudFront with AWS Lambda@Edge.
D. Use Amazon API Gateway and AWS Lambda to control access to an S3 bucket.
Answer: A
This solution allows easy and secure control of access to the downloads at the lowest cost because it uses a content delivery network (CDN) that can cache and distribute firmware updates to customers around the world, and uses a mechanism that can restrict access to specific files or versions. Amazon CloudFront is a CDN that can improve performance, availability, and security of web applications by delivering content from edge locations closer to customers. Amazon S3 is a storage service that can store firmware updates in buckets and objects. Signed URLs are URLs that include additional information, such as an expiration date and time, that give users temporary access to specific objects in S3 buckets. The developer can use CloudFront to serve firmware updates from S3 buckets and use signed URLs to control who can download them and for how long.
Creating a dedicated CloudFront distribution for each customer will incur unnecessary costs and complexity.
Using Amazon CloudFront with AWS Lambda@Edge will require additional programming overhead to implement custom logic at the edge locations. Using Amazon API Gateway and AWS Lambda to control access to an S3 bucket will also require additional programming overhead and may not provide optimal performance or availability.
Reference: [Serving Private Content through CloudFront], [Using CloudFront with Amazon S3]

NO.24 A developer is building a highly secure healthcare application using serverless components. This application requires writing temporary data to /Imp storage on an AWS Lambda function.
How should the developer encrypt this data?
A. Enable Amazon EBS volume encryption with an AWS KMS key in the Lambda function configuration so that all storage attached to the Lambda function is encrypted.
B. Set up the Lambda function with a role and key policy to access an AWS KMS key. Use the key to generate a data key used to encrypt all data prior to writing to Amp storage.
C. Use OpenSSL to generate a symmetric encryption key on Lambda startup. Use this key to encrypt the data prior to writing to /tmp.
D. Use an on-premises hardware security module (HSM) to generate keys, where the Lambda function requests a data key from the HSM and uses that to encrypt data on all requests to the function.
Answer: B

NO.25 A developer is building an application that uses an AWS Lambda function to process data. The application requires minimum latency. The Lambda function must have predictable function start times. All setup activities for the execution environment must happen before invocation of the Lambda function.
Which solution will meet these requirements?
A. Increase the memory of the Lambda function to the maximum amount. Configure an Amazon EventBridge rule to schedule invocations of the Lambda function every minute to keep the execution environment active.
B. Optimize the static initialization code that runs when a new execution environment is prepared for the first time. Decrease and compress the size of the Lambda function package and the imported

libraries and dependencies.

C. Increase the reserved concurrency of the Lambda function to the maximum value for unreserved account concurrency. Run any setup activities manually before the initial invocation of the Lambda function.

D. Publish a new version of the Lambda function. Configure provisioned concurrency for the Lambda function with the required minimum number of execution environments.

Answer: D

NO.26 A company has many microservices that are comprised of AWS Lambda functions. Multiple teams within the company split ownership of the microservices.

An application reads configuration values from environment variables that are contained in the Lambda functions. During a security audit, the company discovers that some of the environment variables contain sensitive information.

The company's security policy requires each team to have full control over the rotation of AWS KMS keys that the team uses for its respective microservices.

A. Create AWS managed keys for all Lambda functions. Use the new AWS managed keys to encrypt the environment variables. Add kms:Decrypt permissions to the Lambda function execution roles.

B. Create customer managed keys for all Lambda functions. Use the new customer managed keys to encrypt the environment variables. Add kms:Decrypt permission to the Lambda function execution roles.

C. Create customer managed keys for all Lambda functions. Use the new customer managed keys to encrypt the environment variables. Add kms:CreateGrant permission and kms:Encrypt permission to the Lambda function execution roles.

D. Create AWS managed keys for all Lambda functions. Use the new AWS managed keys to encrypt the environment variables. Add kms:CreateGrant permission and kms:Encrypt permission to the Lambda function execution roles.

Answer: B

Comprehensive and Detailed Step-by-Step Explanation:

* Customer Managed Keys (CMK) for Granular Control (Option B):
* Customer-managed KMS keys are required to meet the security policy requirement of team- specific control over KMS key rotation. Each team can manage the lifecycle of its own key.
* The kms:Decrypt permission allows the Lambda function execution roles to decrypt the environment variables during runtime.
* This solution adheres to the principle of least privilege and satisfies the need for team-specific key control.
* Why Other Options Are Incorrect:
* Option A: AWS-managed keys cannot provide team-specific control or support the custom rotation policy required by the teams.
* Option C: Adding kms:CreateGrant and kms:Encrypt permissions to Lambda roles is unnecessary for this scenario. The key usage is limited to decryption at runtime.
* Option D: AWS-managed keys still lack team-specific control, and adding kms:CreateGrant and kms:Encrypt is redundant.

References:

AWS Lambda Environment Variables

AWS Key Management Service Documentation

NO.27 For a deployment using AWS Code Deploy, what is the run order of the hooks for in-place deployments?

A. BeforeInstall -> ApplicationStop -> ApplicationStart -> AfterInstall

B. ApplicationStop -> BeforeInstall -> AfterInstall -> ApplicationStart

C. BeforeInstall -> ApplicationStop -> ValidateService -> ApplicationStart

D. ApplicationStop -> BeforeInstall -> ValidateService -> ApplicationStart

Answer: B

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

* ApplicationStop: This hook runs first on all instances and stops the current application that is running on the instances.
* BeforeInstall: This hook runs after ApplicationStop on all instances and performs any tasks required before installing the new application revision.
* AfterInstall: This hook runs after BeforeInstall on all instances and performs any tasks required after installing the new application revision.
* ApplicationStart: This hook runs after AfterInstall on all instances and starts the new application that has been installed on the instances.
* ValidateService: This hook runs last on all instances and verifies that the new application is running properly on the instances.

Reference: [AWS CodeDeploy lifecycle event hooks reference]

NO.28 A company has an application that is deployed on AWS Elastic Beanstalk. The application generates user- specific PDFs and stores the PDFs in an Amazon S3 bucket. The application then uses Amazon Simple Email Service (Amazon SES) to send the PDFs by email to subscribers.

Users no longer access the PDFs 90 days after the PDFs are generated. The S3 bucket is not versioned and contains many obsolete PDFs.

A developer must reduce the number of files in the S3 bucket by removing PDFs that are older than 90 days.

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

A. Update the application code. In the code, add a rule to scan all the objects in the S3 bucket every day and to delete objects after 90 days.

B. Create an AWS Lambda function. Program the Lambda function to scan all the objects in the S3 bucket every day and to delete objects after 90 days.

C. Create an S3 Lifecycle rule for the S3 bucket to expire objects after 90 days.

D. Partition the S3 objects with a <year>/<month>/<day> key prefix. Create an AWS Lambda function to remove objects that have prefixes that have reached the expiration date.

Answer: C

NO.29 A developer is creating a microservices application that runs across multiple compute environments.

The application must securely access secrets that are stored in AWS Secrets Manager with minimal network latency. The developer wants a solution that reduces the number of direct calls to Secrets

Manager and simplifies secrets management across environments. Which solution will meet these requirements with the LEAST operational overhead?

A. Create a custom script that retrieves secrets directly from Secrets Manager and caches the secrets in a local database for each compute environment.

B. Install the Secrets Manager Agent in each compute environment. Configure the agent to cache secrets locally. Securely retrieve the secrets from Secrets Manager as needed.

C. Implement lazy loading logic in the application to fetch secrets directly from Secrets Manager and to cache the secrets in Redis.

D. Store the secrets in an Amazon S3 bucket. Retrieve and load the secrets as environment variables during application startup for each compute environment.

Answer: B

The Secrets Manager Agent provides an out-of-the-box solution for securely caching secrets locally, reducing latency and operational overhead.

* Why Option B is Correct:
* Caching: The agent securely caches secrets locally, minimizing Secrets Manager API calls.
* Security: Secrets remain secure during retrieval and storage.
* Low Operational Overhead: Managed solution eliminates the need for custom logic.
* Why Not Other Options:
* Option A: Custom scripts introduce complexity and require ongoing maintenance.
* Option C: Using Redis requires managing an additional service, increasing overhead.
* Option D: Storing secrets in S3 lacks the fine-grained security controls of Secrets Manager.

:

Caching Secrets in AWS Secrets Manager

NO.30 An application ingests data from an Amazon Kinesis data stream. The shards in the data stream are set for normal traffic.

During tests for peak traffic, the application ingests data slowly. A developer needs to adjust the data stream to handle the peak traffic.

What should the developer do to meet this requirement MOST cost-effectively?

A. Install the Kinesis Producer Library {KPL) to ingest data into the data stream.

B. Switch to on-demand capacity mode for the data stream. Specify a partition key when writing data to the data stream.

C. Decrease the amount of time that data is kept in the data stream by using the DecreaseStreamRetention Period API operation.

D. Increase the shard count in the data stream by using the UpdateShardCount API operation.

Answer: D

NO.31 An ecommerce application is running behind an Application Load Balancer. A developer observes some unexpected load on the application during non-peak hours. The developer wants to analyze patterns for the client IP addresses that use the application. Which HTTP header should the developer use for this analysis?

A. The X-Forwarded-Proto header

B. The X-F Forwarded-Host header

C. The X-Forwarded-For header

D. The X-Forwarded-Port header

Answer: C

The HTTP header that the developer should use for this analysis is the X-Forwarded-For header. This header contains the IP address of the client that made the request to the Application Load Balancer. The developer can use this header to analyze patterns for the client IP addresses that use the application. The other headers either contain information about the protocol, host, or port of the request, which are not relevant for the analysis.

Reference: How Application Load Balancer works with your applications

NO.32 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 API. Embed responses captured from the real third-party API. Configure Amazon Route 53 Resolver with an inbound endpoint for each Lambda function's Amazon Resource Name (ARN).

D. 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

* Mocking API Responses: API Gateway's Mock integration type enables simulating API behavior without invoking backend services.
* Testing with Sample Data: Using captured responses from the real third-party API ensures realistic testing of the integration code.
* Focus on Integration Logic: This solution allows the developer to isolate and test the application's interaction with the payment APIs, even without a test environment from the third-party providers.

References:

Amazon API Gateway Mock Integrations:
https://docs.aws.amazon.com/apigateway/latest/developerguide
/how-to-mock-integration.html

NO.33 A financial company must store original customer records for 10 years for legal reasons. A complete record contains personally identifiable information (PII). According to local regulations, PII is available to only certain people in the company and must not be shared with third parties. The company needs to make the records available to third-party organizations for statistical analysis without sharing the PII.

A developer wants to store the original immutable record in Amazon S3. Depending on who accesses the S3 document, the document should be returned as is or with all the PII removed. The developer

has written an AWS Lambda function to remove the PII from the document. The function is named removePii.
What should the developer do so that the company can meet the PII requirements while maintaining only one copy of the document?
A. Set up an S3 event notification that invokes the removePii function when an S3 GET request is made.
Call Amazon S3 by using a GET request to access the object without PII.
B. Set up an S3 event notification that invokes the removePii function when an S3 PUT request is made.
Call Amazon S3 by using a PUT request to access the object without PII.
C. Create an S3 Object Lambda access point from the S3 console. Select the removePii function. Use S3 Access Points to access the object without PII.
D. Create an S3 access point from the S3 console. Use the access point name to call the GetObjectLegalHold S3 API function. Pass in the removePii function name to access the object without PII.
Answer: C
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

NO.34 A developer is preparing to deploy an AWS CloudFormation stack for an application from a template that includes an IAM user.
The developer needs to configure the application's resources to retain the IAM user after successful creation.
However, the developer also needs to configure the application to delete the IAM user if the stack rolls back.
A. Update CloudFormation template with the following deletion policy:
AWSTemplateFormatVersion: '2010-05-09'
Resources:
appUser:
Type: AWS::IAM::User
DeletionPolicy: Retain
B. Update CloudFormation template with the following deletion policy:
AWSTemplateFormatVersion: '2010-09-09'
Resources:
appUser:
Type: AWS::IAM::User
DeletionPolicy: RetainExceptOnCreate
C. Update the CloudFormation service role to include the following policy:
{
"Version": "2012-10-17",

"Statement": [{
"Effect": "Allow",
"Action": ["cloudformation:UpdateTerminationProtection"],
"Resource": "*"
}]
}

D. Update the stack policy to include the following statements:

{
"Statement": [{
"Effect": "Deny",
"Action": "Update:*",
"Principal": "*",
"Resource": "*",
"Condition": {
"StringEquals": {
"ResourceType": "AWS::IAM::User"
}
}
}]
}

Answer: B

* Why Option B is Correct:The RetainExceptOnCreate deletion policy ensures that the IAM user is retained after successful stack creation but is deleted if the stack creation fails or rolls back. This meets both requirements.
* Why Other Options are Incorrect:
* Option A: The Retain policy retains the resource regardless of stack status and does not delete the IAM user upon rollback.
* Option C: Updating the service role for termination protection does not address the specific deletion behavior for the IAM user.
* Option D: Stack policy controls updates, not resource deletion behavior during rollbacks.
* AWS Documentation References:
* CloudFormation DeletionPolicy Attribute

NO.35 An online food company provides an Amazon API Gateway HTTP API 1o receive orders for partners. The API is integrated with an AWS Lambda function. The Lambda function stores the orders in an Amazon DynamoDB table.

The company expects to onboard additional partners Some to me panthers require additional Lambda function to receive orders. The company has created an Amazon S3 bucket. The company needs 10 store all orders and updates m the S3 bucket for future analysis How can the developer ensure that an orders and updates are stored to Amazon S3 with the LEAST development effort?

A. Create a new Lambda function and a new API Gateway API endpoint. Configure the new Lambda function to write to the S3 bucket. Modify the original Lambda function to post updates to the new API endpoint.

B. Use Amazon Kinesis Data Streams to create a new data stream. Modify the Lambda function to publish orders to the oats stream Configure in data stream to write to the S3 bucket.

C. Enable DynamoDB Streams on me DynamoOB table. Create a new lambda function. Associate the

stream's Amazon Resource Name (ARN) with the Lambda Function Configure the Lambda function to write to the S3 bucket as records appear in the table's stream.

D. Modify the Lambda function to punish to a new Amazon. Simple Lambda function receives orders.Subscribe a new Lambda function to the topic. Configure the new Lambda function to write to the S3 bucket as updates come through the topic.

Answer: C

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

References: Using DynamoDB Streams, Using AWS Lambda with Amazon S3