DOP-C02$^{Q\&As}$

AWS Certified DevOps Engineer - Professional

Pass Amazon DOP-C02 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.leads4pass.com/dop-c02.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Amazon Official Exam Center

- **Instant Download** After Purchase
- **100% Money Back** Guarantee
- **365 Days** Free Update
- **800,000+** Satisfied Customers

QUESTION 1

A company runs a workload on Amazon EC2 instances. The company needs a control that requires the use of Instance Metadata Service Version 2 (IMDSv2) on all EC2 instances in the AWS account. If an EC2 instance does not prevent the use of Instance Metadata Service Version 1 (IMDSv1), the EC2 instance must be terminated.

Which solution will meet these requirements?

A. Set up AWS Config in the account. Use a managed rule to check EC2 instances. Configure the rule to remediate the findings by using AWS Systems Manager Automation to terminate the instance.

B. Create a permissions boundary that prevents the ec2:RunInstance action if the ec2:MetadataHttpTokens condition key is not set to a value of required. Attach the permissions boundary to the IAM role that was used to launch the instance.

C. Set up Amazon Inspector in the account. Configure Amazon Inspector to activate deep inspection for EC2 instances. Create an Amazon EventBridge rule for an Inspector2 finding. Set an AWS Lambda function as the target to terminate the instance.

D. Create an Amazon EventBridge rule for the EC2 instance launch successful event. Send the event to an AWS Lambda function to inspect the EC2 metadata and to terminate the instance.

Correct Answer: B

To implement a control that requires the use of IMDSv2 on all EC2 instances in the account, the DevOps engineer can use a permissions boundary. A permissions boundary is a policy that defines the maximum permissions that an IAM entity can have. The DevOps engineer can create a permissions boundary that prevents the ec2:RunInstance action if the ec2:MetadataHttpTokens condition key is not set to a value of required. This condition key enforces the use of IMDSv2 on EC2 instances. The DevOps engineer can attach the permissions boundary to the IAM role that was used to launch the instance. This way, any attempt to launch an EC2 instance without using IMDSv2 will be denied by the permissions boundary.

QUESTION 2

A company has a new AWS account that teams will use to deploy various applications. The teams will create many Amazon S3 buckets for application- specific purposes and to store AWS CloudTrail logs. The company has enabled Amazon Macie for the account.

A DevOps engineer needs to optimize the Macie costs for the account without compromising the account\\'s functionality.

Which solutions will meet these requirements? (Select TWO.)

A. Exclude S3 buckets that contain CloudTrail logs from automated discovery.

B. Exclude S3 buckets that have public read access from automated discovery.

C. Configure scheduled daily discovery jobs for all S3 buckets in the account.

D. Configure discovery jobs to include S3 objects based on the last modified criterion.

E. Configure discovery jobs to include S3 objects that are tagged as production only.

Correct Answer: AD

To optimize the Macie costs for the account without compromising the account\\'s functionality, the DevOps engineer needs to exclude S3 buckets that do not contain sensitive data from automated discovery. S3 buckets that contain CloudTrail logs are unlikely to have sensitive data, and Macie charges for scanning and monitoring data in S3 buckets. Therefore, excluding S3 buckets that contain CloudTrail logs from automated discovery can reduce Macie costs. Similarly, configuring discovery jobs to include S3 objects based on the last modified criterion can also reduce Macie costs, as it will only scan and monitor new or updated objects, rather than all objects in the bucket.

QUESTION 3

A company\\'s application uses a fleet of Amazon EC2 On-Demand Instances to analyze and process data. The EC2 instances are in an Auto Scaling group. The Auto Scaling group is a target group for an Application Load Balancer (ALB).

The application analyzes critical data that cannot tolerate interruption. The application also analyzes noncritical data that can withstand interruption.

The critical data analysis requires quick scalability in response to real-time application demand. The noncritical data analysis involves memory consumption. A DevOps engineer must implement a solution that reduces scale-out latency for the

critical data. The solution also must process the noncritical data.

Which combination of steps will meet these requirements? (Select TWO.)

A. For the critical data, modify the existing Auto Scaling group. Create a warm pool instance in the stopped state. Define the warm pool size. Create a new version of the launch template that has detailed monitoring enabled. use Spot Instances.

B. For the critical data, modify the existing Auto Scaling group. Create a warm pool instance in the stopped state. Define the warm pool size. Create a new version of the launch template that has detailed monitoring enabled. Use On-Demand Instances.

C. For the critical data. modify the existing Auto Scaling group. Create a lifecycle hook to ensure that bootstrap scripts are completed successfully. Ensure that the application on the instances is ready to accept traffic before the instances are registered. Create a new version of the launch template that has detailed monitoring enabled.

D. For the noncritical data, create a second Auto Scaling group that uses a launch template. Configure the launch template to install the unified Amazon CloudWatch agent and to configure the CloudWatch agent with a custom memory utilization metric. Use Spot Instances. Add the new Auto Scaling group as the target group for the ALB. Modify the application to use two target groups for critical data and noncritical data.

E. For the noncritical data, create a second Auto Scaling group. Choose the predefined memory utilization metric type for the target tracking scaling policy. Use Spot Instances. Add the new Auto Scaling group as the target group for the ALB. Modify the application to use two target groups for critical data and noncritical data.

Correct Answer: BD

For the critical data, using a warm pool1 can reduce the scale-out latency by having pre-initialized EC2 instances ready to serve the application traffic. Using On-Demand Instances can ensure that the instances are always available and not

interrupted by Spot interruptions2.

For the noncritical data, using a second Auto Scaling group with Spot Instances can reduce the cost and leverage the

unused capacity of EC23. Using a launch template with the CloudWatch agent4 can enable the collection of memory

utilization metrics, which can be used to scale the group based on the memory demand. Adding the second group as a target group for the ALB and modifying the application to use two target groups can enable routing the traffic based on the

data type.

References:

1: Warm pools for Amazon EC2 Auto Scaling

2: Amazon EC2 On-Demand Capacity Reservations

3: Amazon EC2 Spot Instances

4: Metrics collected by the CloudWatch agent

QUESTION 4

A company builds a container image in an AWS CodeBuild project by running Docker commands. After the container image is built, the CodeBuild project uploads the container image to an Amazon S3 bucket. The CodeBuild project has an 1AM service role that has permissions to access the S3 bucket.

A DevOps engineer needs to replace the S3 bucket with an Amazon Elastic Container Registry (Amazon ECR) repository to store the container images. The DevOps engineer creates an ECR private image repository in the same AWS Region of the CodeBuild project. The DevOps engineer adjusts the 1AM service role with the permissions that are necessary to work with the new ECR repository. The DevOps engineer also places new repository information into the docker build command and the docker push command that are used in the buildspec.yml file.

When the CodeBuild project runs a build job, the job fails when the job tries to access the ECR repository.

Which solution will resolve the issue of failed access to the ECR repository?

A. Update the buildspec.yml file to log in to the ECR repository by using the aws ecr get-login-password AWS CLI command to obtain an authentication token. Update the docker login command to use the authentication token to access the ECR repository.

B. Add an environment variable of type SECRETS_MANAGER to the CodeBuild project. In the environment variable, include the ARN of the CodeBuild project\\'s IAM service role. Update the buildspec.yml file to use the new environment variable to log in with the docker login command to access the ECR repository.

C. Update the ECR repository to be a public image repository. Add an ECR repository policy that allows the 1AM service role to have access.

D. Update the buildspec.yml file to use the AWS CLI to assume the 1AM service role for ECR operations. Add an ECR repository policy that allows the 1AM service role to have access.

Correct Answer: A

(A) When Docker communicates with an Amazon Elastic Container Registry (ECR) repository, it requires authentication. You can authenticate your Docker client to the Amazon ECR registry with the help of the AWS CLI (Command Line Interface). Specifically, you can use the "aws ecr get-login-password" command to get an authorization token and then use Docker\\'s "docker login" command with that token to authenticate to the registry. You would need to perform these steps in your buildspec.yml file before attempting to push or pull images from/to the ECR repository.

QUESTION 5

A company detects unusual login attempts in many of its AWS accounts. A DevOps engineer must implement a solution that sends a notification to the company\\'s security team when multiple failed login attempts occur. The DevOps engineer has already created an Amazon Simple Notification Service (Amazon SNS) topic and has subscribed the security team to the SNS topic.

Which solution will provide the notification with the LEAST operational effort?

A. Configure AWS CloudTrail to send log management events to an Amazon CloudWatch Logs log group. Create a CloudWatch Logs metric filter to match failed ConsoleLogin events. Create a CloudWatch alarm that is based on the metric filter. Configure an alarm action to send messages to the SNS topic.

B. Configure AWS CloudTrail to send log management events to an Amazon S3 bucket. Create an Amazon Athena query that returns a failure if the query finds failed logins in the logs in the S3 bucket. Create an Amazon EventBridge rule to periodically run the query. Create a second EventBridge rule to detect when the query fails and to send a message to the SNS topic.

C. Configure AWS CloudTrail to send log data events to an Amazon CloudWatch Logs log group. Create a CloudWatch logs metric filter to match failed Consolel_ogin events. Create a CloudWatch alarm that is based on the metric filter. Configure an alarm action to send messages to the SNS topic.

D. Configure AWS CloudTrail to send log data events to an Amazon S3 bucket. Configure an Amazon S3 event notification for the s3:ObjectCreated event type. Filter the event type by ConsoleLogin failed events. Configure the event notification to forward to the SNS topic.

Correct Answer: C