Free Questions for DOP-C01 by braindumpscollection

Shared by Anthony on 07-06-2022

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

A mobile application running on eight Amazon EC2 instances is relying on a third-party API endpoint. The third-party service has a high failure rate because of limited capacity which is expected to be resolved in a few weeks.

In the meantime the mobile application developers have added a retry mechanism and are logging failed API requests. A DevOps Engineer must automate the monitoring of application togs and count the specific error messages if there are more than 10 errors within a 1-minute window the system must issue an alert

How can the requirements be met with MINIMAL management overhead?

Options:

A) Install AfterAllowTraffic hook to the AppSpec file that forces traffic
not having fully propagated before the
push the application logs to CloudWatch Logs Use metric fitters to count the error messages every minute and trigger a CloudWatch alarm if the count exceeds errors.

B) Install the Amazon CloudWatch Logs agent on all instances to push the access logs to CloudWatch Logs Create a CloudWatch Events rule to count the error messages every minute and trigger a CloudWatch alarm if the count exceeds 10 errors

C) Install the Amazon CloudWatch Logs agent on all instances to push the application logs to CloudWatch Logs Use a metric filter to generate a custom CloudWatch metric that records the number of failures and triggers a CloudWatch alarm if the custom metric reaches 10 errors in a 1-mmute period

D) Deploy a custom script on all instances to check application logs regularly in a job Count the number of error messages every minute and push a data point to a custom CloudWatch metric Trigger a CloudWatch alarm if the custom metric reaches 10 errors in a 1-minute period

Answer:

C

Question 2

Question Type: **MultipleChoice**

A global company with distributed Development teams built a web application using a microservices architecture running on Amazon ECS. Each application service is independent and runs as a service in the ECS cluster. The container build files and source code reside in a private GitHub source code repository. Separate ECS clusters exist for development, testing, and production environments. Developers are required to push features to branches in the GitHub repository and then merge the changes into an environment-specific branch (development, test, or production). This merge needs to trigger an automated pipeline to run a build and a deployment to the appropriate ECS cluster. What should the DevOps Engineer recommend as an automated solution to these requirements?

Options:

A) Create an AWS CloudFormation stack for the ECS cluster and AWS CodePipeline services. Store the container build files in an Amazon S3 bucket. Use a post-commit hook to trigger a CloudFormation stack update that deploys the ECS cluster. Add a task in the ECS cluster to build and push images to Amazon ECR, based on the container build files in S3.

B) Create a separate pipeline in AWS CodePipeline for each environment. Trigger each pipeline based on commits to the corresponding environment branch in GitHub. Add a build stage to launch AWS CodeBuild to create the container image from the build file and push it to Amazon ECR. Then add another stage to update the Amazon ECS task and service definitions in the appropriate cluster for that environment.

C) Create a pipeline in AWS CodePipeline. Configure it to be triggered by commits to the master branch in GitHub. Add a stage to use the Git commit message to determine which environment the commit should be applied to, then call the create-image Amazon ECR command to build the image, passing it to the container build file. Then add a stage to update the ECS task and service definitions in the appropriate cluster for that environment.

D) Create a new repository in AWS CodeCommit. Configure a scheduled project in AWS CodeBuild to synchronize the GitHub repository to the new CodeCommit repository. Create a separate pipeline for each environment triggered by changes to the CodeCommit repository. Add a stage using AWS Lambda to build the container image and push to Amazon ECR. Then add another stage to update the ECS task and service definitions in the appropriate cluster for that environment.

Answer:

B

Explanation:

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-cd-pipeline.html

Question 3

Question Type: MultipleChoice

A company is using AWS CodeBuild, AWS CodeDeploy, and AWS CodePipeline to deploy applications automatically to an Amazon EC2 instance. A DevOps Engineer needs to perform a security assessment scan of the operating system on every application deployment to the environment. How should this be automated?

Options:

A) Use Amazon CloudWatch Events to monitor for Auto Scaling event notifications of new instances and configure CloudWatch Events to trigger an Amazon Inspector scan.

B) Use Amazon CloudWatch Events to monitor for AWS CodeDeploy notifications of a successful code deployment and configure CloudWatch Events to trigger an Amazon Inspector scan.

C) Use Amazon CloudWatch Events to monitor for CodePipeline notifications of a successful code deployment and configure CloudWatch Events to trigger an AWS X-Ray scan.

D) Use Amazon Inspector as a CodePipeline task after the successful use of CodeDeploy to deploy the code to the systems.

Answer:

B

Question 4

Question Type: MultipleChoice

A Security team is concerned that a Developer can unintentionally attach an Elastic IP address to an Amazon EC2 instance in production. No Developer should be allowed to attach an Elastic IP address to an instance. The Security team must be notified if any production server has an Elastic IP address at any time. How can this task be automated?

Options:

A) Use Amazon Athena to query AWS CloudTrail logs to check for any associate-address attempts. Create an AWS Lambda function to dissociate the Elastic IP address from the instance, and alert the Security team.

B) Attach an IAM policy to the Developer's IAM group to deny associate-address permissions. Create a custom AWS Config rule to check whether an Elastic IP address is associated with any instance tagged as production, and alert the Security team.

C) Ensure that all IAM groups are associated with Developers do not have associate-address permissions. Create a scheduled AWS Lambda function to check whether an Elastic IP address is associated with any instance tagged as production, and alert the Security team if an instance has an Elastic IP address associated with it.

D) Create an AWS Config rule to check that all production instances have the EC2 IAM roles that include deny associate-address permissions. Verify whether there is an Elastic IP address associated with any instance, and alert the Security team if an instance has an Elastic IP address associated with it.

Answer:

B

Explanation:

https://docs.aws.amazon.com/vpc/latest/userguide/vpc-migrate-ipv6.html#vpc-migrate-ipv6-sg-rules

Question 5

Question Type: MultipleChoice

A government agency has multiple AWS accounts, many of which store sensitive citizen information. A Security team wants to detect anomalous account and network activities (such as SSH brute force attacks) in any account and centralize that information in a dedicated security account. Event information should be stored in an Amazon S3 bucket in the security account, which is monitored by the department's Security Information and Even Manager (SIEM) system. How can this be accomplished?

Options:

A) Enable Amazon Macie in every account. Configure the security account as the Macie Administrator for every member account using invitation/acceptance. Create an Amazon CloudWatch Events rule in the security account to send all findings to Amazon Kinesis Data Firehouse, which should push the findings to the S3 bucket.

B) Enable Amazon Macie in the security account only. Configure the security account as the Macie Administrator for every member account using invitation/ acceptance. Create an Amazon CloudWatch Events rule in the security account to send all findings to Amazon Kinesis Data Streams. Write and application using KCL to read data from the Kinesis Data Streams and write to the S3 bucket.

C) Enable Amazon GuardDuty in every account. Configure the security account as the GuardDuty Administrator for every member account using invitation/ acceptance. Create an Amazon CloudWatch rule in the security account to send all findings to Amazon Kinesis Data Firehouse, which will push the findings to the S3 bucket.

D) Enable Amazon GuardDuty in the security account only. Configure the security account as the GuardDuty Administrator for every member account using invitation/acceptance. Create an Amazon CloudWatch rule in the security account to send all findings to Amazon Kinesis Data Streams. Write and application using KCL to read data from Kinesis Data Streams and write to the S3 bucket.

Answer:

C

Explanation:

https://aws.amazon.com/blogs/security/how-to-manage-amazon-guardduty-security-findings-across-multiple-accounts/

Question 6

Question Type: MultipleChoice

A healthcare company has a critical application running in AWS. Recently, the company experienced some down time. if it happens again, the company needs to be able to recover its application in another AWS Region. The application uses Elastic Load Balancing and Amazon EC2 instances. The company also maintains a custom AMI that contains its application. This AMI is changed frequently. The workload is required to run in the primary region, unless there is a regional service disruption, in which case traffic should fail over to the new region. Additionally, the cost for the second region needs to be low. The RTO is 2 hours. Which solution allows the company to fail over to another region in the event of a failure, and also meet the above requirements?

Options:

A) Maintain a copy of the AMI from the main region in the backup region. Create an Auto Scaling group with one instance using a launch configuration that contains the copied AMI. Use an Amazon Route 53 record to direct traffic to the load balancer in the backup region in the event of failure, as required. Allow the Auto Scaling group to scale out as needed during a failure.

B) Automate the copying of the AMI in the main region to the backup region. Generate an AWS Lambda function that will create an EC2 instance from the AMI and place it behind a load balancer. Using the same Lambda function, point the Amazon Route 53 record to the load balancer in the backup region. Trigger the Lambda function in the event of a failure.

C) Place the AMI in a replicated Amazon S3 bucket. Generate an AWS Lambda function that can create a launch configuration and assign it to an already created Auto Scaling group. Have one instance in this Auto Scaling group ready to accept traffic. Trigger the Lambda function in the event of a failure. Use an Amazon Route 53 record and modify it with the same Lambda function to point to the load balancer in the backup region.

D) Automate the copying of the AMI to the backup region. Create an AWS Lambda function that can create a launch configuration and assign it to an already created Auto Scaling group. Set the Auto Scaling group maximum size to 0 and only increase it with the Lambda function during a failure. Trigger the Lambda function in the event of a failure. Use an Amazon Route 53 record and modify it with the same Lambda function to point to the load balancer in the backup region.

Answer:

C

Question 7

Question Type: **MultipleChoice**

A DevOps Engineer just joined a new company that is already running workloads on Amazon EC2 instances. AWS has been adopted incrementally with no central governance. The Engineer must now assess how well the existing deployments comply with the following requirements: *EC2 instances are running only approved AMIs. *Amazon EBS volumes are encrypted. *EC2 instances have an Owner tag. *Root login over SSH is disabled on EC2 instances. Which services should the Engineer use to perform this assessment with the LEAST amount of effort? (Select TWO.)

Options:

A) AWS Config

B) Amazon GuardDuty

C) AWS System Manager

D) AWS Directory Service

E) Amazon Inspector

Answer:

A, E

Explanation:

https://docs.aws.amazon.com/ja_jp/inspector/latest/userguide/inspector_security-best-practices.html

Question 8

Question Type: **MultipleChoice**

A DevOps Engineer needs to deploy a scalable three-tier Node.js application in AWS. The application must have zero downtime during deployments and be able to roll back to previous versions. Other applications will also connect to the same MySQL backend database.

The CIO has provided the following guidance for logging: *Centrally view all current web access server logs. *Search and filter web and application logs in near-real time. *Retain log data for three months. How should these requirements be met?

Options:

A) Deploy the application using AWS Elastic Beanstalk. Configure the environment type for Elastic Load Balancing and Auto Scaling. Create an Amazon RDS MySQL instance inside the Elastic Beanstalk stack. Configure the Elastic Beanstalk log options to stream logs to Amazon CloudWatch Logs. Set retention to 90 days.

B) Deploy the application on Amazon EC2. Configure Elastic Load Balancing and Auto Scaling. Use an Amazon RDS MySQL instance for the database tier. Configure the application to store log files in Amazon S3. Use Amazon EMR to search and filter the data. Set an Amazon S3 lifecycle rule to expire objects after 90 days.

C) Deploy the application using AWS Elastic Beanstalk. Configure the environment type for Elastic Load Balancing and Auto Scaling. Create the Amazon RDS MySQL instance outside the Elastic Beanstalk stack. Configure the Elastic Beanstalk log options to stream logs to Amazon CloudWatch Logs. Set retention to 90 days.

D) Deploy the application on Amazon EC2. Configure Elastic Load Balancing and Auto Scaling. Use an Amazon RDS MySQL instance for the database tier. Configure the application to load streaming log data using Amazon Kinesis Data Firehouse into Amazon ES. Delete and create a new Amazon ES domain every 90 days.

Answer:

B

Explanation:

https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-debugging.html