Free Questions for DOP-C02 by braindumpscollection

Shared by Downs on 29-01-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

An ecommerce company uses a large number of Amazon Elastic Block Store (Amazon EBS) backed Amazon EC2 instances. To decrease manual work across all the instances, a DevOps engineer is tasked with automating restart actions when EC2 instance retirement events are scheduled.

How can this be accomplished?

Options:

A- Create a scheduled Amazon EventBridge rule to run an AWS Systems Manager Automation runbook that checks if any EC2 instances are scheduled for retirement once a week If the instance is scheduled for retirement the runbook will hibernate the instance

B- Enable EC2Auto Recovery on all of the instances. Create an AWS Config rule to limit the recovery to occur during a maintenance window only

C- Reboot all EC2 instances during an approved maintenance window that is outside of standard business hours Set up Amazon CloudWatch alarms to send a notification in case any instance is failing EC2 instance status checks

D- Set up an AWS Health Amazon EventBridge rule to run AWS Systems Manager Automation runbooks that stop and start the EC2 instance when a retirement scheduled event occurs.

Answer:

D

Explanation:

https://aws.amazon.com/blogs/mt/automate-remediation-actions-for-amazon-ec2-notifications-and-beyond-using-ec2-systems-manager-automation-and-aws-health/

Question 2

Question Type: MultipleChoice

A company manages a multi-tenant environment in its VPC and has configured Amazon GuardDuty for the corresponding AWS account. The company sends all GuardDuty findings to AWS Security Hub.

Traffic from suspicious sources is generating a large number of findings. A DevOps engineer needs to implement a solution to automatically deny traffic across the entire VPC when GuardDuty discovers a new suspicious source.

Which solution will meet these requirements?

Options:

A- Create a GuardDuty threat list. Configure GuardDuty to reference the list. Create an AWS Lambda function that will update the threat list Configure the Lambda function to run in response to new Security Hub findings that come from GuardDuty.

B- Configure an AWS WAF web ACL that includes a custom rule group. Create an AWS Lambda function that will create a block rule in the custom rule group Configure the Lambda function to run in response to new Security Hub findings that come from GuardDuty

C- Configure a firewall in AWS Network Firewall. Create an AWS Lambda function that will create a Drop action rule in the firewall policy Configure the Lambda function to run in response to new Security Hub findings that come from GuardDuty

D- Create an AWS Lambda function that will create a GuardDuty suppression rule. Configure the Lambda function to run in response to new Security Hub findings that come from GuardDuty.

Answer:

C

Explanation:

https://aws.amazon.com/blogs/security/automatically-block-suspicious-traffic-with-aws-network-firewall-and-amazon-guardduty/

Question 3

Question Type: MultipleChoice

A company has a single developer writing code for an automated deployment pipeline. The developer is storing source code in an Amazon S3 bucket for each project. The company wants to add more developers to the team but is concerned about code conflicts and lost work The company also wants to build a test environment to deploy newer versions of code for testing and allow developers to automatically deploy to both environments when code is changed in the repository.

What is the MOST efficient way to meet these requirements?

Options:

A- Create an AWS CodeCommit repository tor each project, use the mam branch for production code: and create a testing branch for code deployed to testing Use feature branches to develop new features and pull requests to merge code to testing and main branches.

B- Create another S3 bucket for each project for testing code, and use an AWS Lambda function to promote code changes between testing and production buckets Enable versioning on all buckets to prevent code conflicts.

C- Create an AWS CodeCommit repository for each project, and use the main branch for production and test code with different deployment pipelines for each environment Use feature branches to develop new features.

D- Enable versioning and branching on each S3 bucket, use the main branch for production code, and create a testing branch for code deployed to testing. Have developers use each branch for developing in each environment.

Answer:

A

Explanation:

Creating an AWS CodeCommit repository for each project, using the main branch for production code, and creating a testing branch for code deployed to testing will meet the requirements.AWS CodeCommit is a managed revision control service that hosts Git repositories and works with all Git-based tools1. By using feature branches to develop new features and pull requests to merge code to testing and main branches, the developers can avoid code conflicts and lost work, and also implement code reviews and approvals. Option B is incorrect because creating another S3 bucket for each project for testing code and using an AWS Lambda function to promote code changes between testing and production buckets will not provide the benefits of revision control, such as tracking changes, branching, merging, and collaborating. Option C is incorrect because using the main branch for production and test code with different deployment pipelines for each environment will not allow the developers to test their code changes before deploying them to production. Option D is incorrect because enabling versioning and branching on each S3 bucket will not work with Git-based tools and will not provide the same level of revision control as AWS CodeCommit.Reference:

AWS CodeCommit

Certified DevOps Engineer - Professional (DOP-C02) Study Guide(page 182)

Question 4

Question Type: MultipleChoice

A DevOps engineer notices that all Amazon EC2 instances running behind an Application Load Balancer in an Auto Scaling group are failing to respond to user requests. The EC2 instances are also failing target group HTTP health checks

Upon inspection, the engineer notices the application process was not running in any EC2 instances. There are a significant number of out of memory messages in the system logs. The engineer needs to improve the resilience of the application to cope with a potential application memory leak. Monitoring and notifications should be enabled to alert when there is an issue

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

Options:

A- Change the Auto Scaling configuration to replace the instances when they fail the load balancer's health checks.

B- Change the target group health check HealthCheckIntervalSeconds parameter to reduce the interval between health checks.

C- Change the target group health checks from HTTP to TCP to check if the port where the application is listening is reachable.

D- Enable the available memory consumption metric within the Amazon CloudWatch dashboard for the entire Auto Scaling group Create an alarm when the memory utilization is high Associate an Amazon SNS topic to the alarm to receive notifications when the alarm goes off

E- Use the Amazon CloudWatch agent to collect the memory utilization of the EC2 instances in the Auto Scaling group Create an alarm when the memory utilization is high and associate an Amazon SNS topic to receive a notification.

Answer:

A, E

Explanation:

https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/metrics-collected-by-CloudWatch-agent.html

Question 5

Question Type: MultipleChoice

A company has microservices running in AWS Lambda that read data from Amazon DynamoDB. The Lambda code is manually deployed by developers after successful testing The company now needs the tests and deployments be automated and run in the cloud Additionally, traffic to the new versions of each microservice should be incrementally shifted over time after deployment.

What solution meets all the requirements, ensuring the MOST developer velocity?

Options:

A- Create an AWS CodePipelme configuration and set up a post-commit hook to trigger the pipeline after tests have passed Use AWS CodeDeploy and create a Canary deployment configuration that specifies the percentage of traffic and interval

B- Create an AWS CodeBuild configuration that triggers when the test code is pushed Use AWS CloudFormation to trigger an AWS CodePipelme configuration that deploys the new Lambda versions and specifies the traffic shift percentage and interval

C- Create an AWS CodePipelme configuration and set up the source code step to trigger when code is pushed. Set up the build step to use AWS CodeBuild to run the tests Set up an AWS CodeDeploy configuration to deploy, then select the

CodeDeployDefault.LambdaLinearlDPercentEvery3Minut.es Option.

D- Use the AWS CLI to set up a post-commit hook that uploads the code to an Amazon S3 bucket after tests have passed. Set up an S3 event trigger that runs a Lambda function that deploys the new version. Use an interval in the Lambda function to deploy the code over time at the required percentage

Answer:

C

Explanation:

https://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-configurations.html

Question 6

Question Type: MultipleChoice

A company has a mission-critical application on AWS that uses automatic scaling The company wants the deployment lilecycle to meet the following parameters.

* The application must be deployed one instance at a time to ensure the remaining fleet continues to serve traffic

* The application is CPU intensive and must be closely monitored

* The deployment must automatically roll back if the CPU utilization of the deployment instance exceeds 85%.

Which solution will meet these requirements?

Options:

A- Use AWS CloudFormalion to create an AWS Step Functions state machine and Auto Scaling hfecycle hooks to move to one instance at a time into a wait state Use AWS Systems Manager automation to deploy the update to each instance and move it back into the Auto Scaling group using the heartbeat timeout

B- Use AWS CodeDeploy with Amazon EC2 Auto Scaling. Configure an alarm tied to the CPU utilization metric. Use the CodeDeployDefault OneAtAtime configuration as a deployment strategy Configure automatic rollbacks within the deployment group to roll back the deployment if the alarm thresholds are breached

C- Use AWS Elastic Beanstalk for load balancing and AWS Auto Scaling Configure an alarm tied to the CPU utilization metric Configure rolling deployments with a fixed batch size of one instance Enable enhanced health to monitor the status of the deployment and roll back based on the alarm previously created.

D- Use AWS Systems Manager to perform a blue/green deployment with Amazon EC2 Auto Scaling Configure an alarm tied to the CPU utilization metric Deploy updates one at a time Configure automatic rollbacks within the Auto Scaling group to roll back the deployment if the alarm thresholds are breached

Answer:

B

Explanation:

https://aws.amazon.com/about-aws/whats-new/2016/09/aws-codedeploy-introduces-deployment-monitoring-with-amazon-cloudwatch-alarms-and-automatic-deployment-rollback/

Question 7

Question Type: MultipleChoice

A company is reviewing its 1AM policies. One policy written by the DevOps engineer has been (lagged as too permissive. The policy is used by an AWS Lambda function that issues a stop command to Amazon EC2 instances tagged with Environment: NonProduccion over the weekend. The current policy is:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:*",
      "Resource": "*"
    }
  ]
}
```

What changes should the engineer make to achieve a policy ot least permission? (Select THREE.)

A.

A. Add the following conditional expression:

```
"Condition": {
  "StringEquals": {
    "aws:principaltype": "lambda.amazonaws.com"
  }
}
```

B.

Change `"Resource": "*"` to `"Resource": "arn:aws:ec2:*:*:instance/*"`

C.

Add the following conditional expression:

```
"Condition": {
  "StringNotEquals": {
    "ec2:ResourceTag/Environment": "Production"
  }
}
```

D.

Add the following conditional expression:

```
"Condition": {
  "StringEquals": {
    "ec2:ResourceTag/Environment": "NonProduction"
  }
}
```

E.

Change `"Action": "ec2:*"` to `"Action": "ec2:StopInstances"`

F.

Add the following conditional expression:

```
  "Condition" :  {
    "DateGreaterThan" : {
      "aws:CurrentTime" : "$ ;{aws:DateTime:Friday}"
    },
    "DateLessThan": {
      "aws:CurrentTime" : "$ ;{aws:DateTime:Monday}"
    }
}
```

Options:

A- Option A

B- Option B

C- Option C

D- Option D

Answer:

A, B, D

Explanation:

The engineer should make the following changes to achieve a policy of least permission:

A:Add a condition to ensure that the principal making the request is an AWS Lambda function. This ensures that only Lambda functions can execute this policy.

B:Narrow down the resources by specifying the ARN of EC2 instances instead of allowing all resources. This ensures that the policy only affects EC2 instances.

D:Add a condition to ensure that this policy only applies to EC2 instances tagged with "Environment: NonProduction". This ensures that production environments are not affected by this policy.

AWS Identity and Access Management (IAM) - AWS Documentation

Certified DevOps Engineer - Professional (DOP-C02) Study Guide(page 179)

Question 8

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 disassociate the Elastic IP address from the instance, and alert the security team.

B- Attach an 1AM policy to the developers' 1AM 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 1AM groups 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 secunty 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 EC2 1AM 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:

To prevent developers from unintentionally attaching an Elastic IP address to an Amazon EC2 instance in production, the best approach is to use IAM policies and AWS Config rules. By attaching an IAM policy that denies theassociate-addresspermission to the developers' IAM group, you ensure that developers cannot perform this action. Additionally, creating a custom AWS Config rule to check for Elastic IP addresses associated with instances tagged as production provides ongoing monitoring. If the rule detects an Elastic IP address, it can trigger an alert to notify the security team.This method is proactive and enforces the necessary permissions while also providing a mechanism for detection and notification.Reference: from Amazon DevOps sources

Question 9

Question Type: MultipleChoice

A company has an AWS CodeDeploy application. The application has a deployment group that uses a single tag group to identify instances for the deployment of Application

Options:

A- The single tag group configuration identifies instances that have Environment=Production and Name=ApplicattonA tags for the deployment of ApplicationA.
The company launches an additional Amazon EC2 instance with Department=Marketing Environment^Production. and Name=ApplicationB tags. On the next CodeDeploy deployment of ApplicationA. the additional instance has ApplicationA installed on it. A DevOps engineer needs to configure the existing deployment group to prevent ApplicationA from being installed on the additional instance
Which solution will meet these requirements?

A- Change the current single tag group to include only the Environment=Production tag Add another single tag group that includes only the Name=ApplicationA tag.

B- Change the current single tag group to include the Department=Marketmg Environment=Production and Name=ApplicationAtags

C- Add another single tag group that includes only the Department=Marketing tag. Keep the Environment=Production and Name=ApplicationA tags with the current single tag group

D- Change the current single tag group to include only the Environment=Production tag Add another single tag group that includes only the Department=Marketing tag

Answer:

A, A

Explanation:

To prevent ApplicationA from being installed on the additional instance, the deployment group configuration needs to be more specific. By changing the current single tag group to include only theEnvironment=Productiontag and adding another single tag group that includes only theName=ApplicationAtag, the deployment process will target only the instances that match both tag groups. This ensures that only instances intended for ApplicationA with the correct environment and name tags will receive the deployment, thus excluding the additional instance with theDepartment=MarketingandName=ApplicationBtags.

AWS CodeDeploy Documentation:Working with instances for CodeDeploy

AWS CodeDeploy Documentation:Stop a deployment with CodeDeploy

Stack Overflow Discussion:CodeDeploy Deployment failed to stop Application

Question 10

Question Type: **MultipleChoice**

A company is using AWS Organizations to create separate AWS accounts for each of its departments The company needs to automate the following tasks

* Update the Linux AMIs with new patches periodically and generate a golden image

* Install a new version to Chef agents in the golden image, is available

* Provide the newly generated AMIs to the department's accounts

Which solution meets these requirements with the LEAST management overhead'?

Options:

A- Write a script to launch an Amazon EC2 instance from the previous golden image Apply the patch updates Install the new version of the Chef agent, generate a new golden image, and then modify the AMI permissions to share only the new image with the department's accounts.

B- Use Amazon EC2 Image Builder to create an image pipeline that consists of the base Linux AMI and components to install the Chef agent Use AWS Resource Access Manager to share EC2 Image Builder images with the department's accounts

C- Use an AWS Systems Manager Automation runbook to update the Linux AMI by using the previous image Provide the URL for the script that will update the Chef agent Use AWS Organizations to replace the previous golden image in the department's accounts.

D- Use Amazon EC2 Image Builder to create an image pipeline that consists of the base Linux AMI and components to install the Chef agent Create a parameter in AWS Systems Manager Parameter Store to store the new AMI ID that can be referenced by the department's accounts

Answer:

B

Explanation:

Amazon EC2 Image Builder is a service that automates the creation, management, and deployment of customized, secure, and up-to-date server images that are pre-installed with software and configuration settings tailored to meet specific IT standards. EC2 Image Builder simplifies the creation and maintenance of golden images, and makes it easy to generate images for multiple platforms, such as Amazon EC2 and on-premises. EC2 Image Builder also integrates with AWS Resource Access Manager, which allows you to share your images across accounts within your organization or with external AWS accounts. This solution meets the requirements of automating the tasks of updating the Linux AMIs, installing the Chef agent, and providing the images to the department's accounts with the least management overhead.Reference:

Amazon EC2 Image Builder

Sharing EC2 Image Builder images

Question 11

Question Type: MultipleChoice

A company is using AWS Organizations to centrally manage its AWS accounts. The company has turned on AWS Config in each member account by using AWS Cloud Formation StackSets The company has configured trusted access in Organizations for AWS Config and has configured a member account as a delegated administrator account for AWS Config

A DevOps engineer needs to implement a new security policy The policy must require all current and future AWS member accounts to use a common baseline of AWS Config rules that contain remediation actions that are managed from a central account Non-administrator users who can access member accounts must not be able to modify this common baseline of AWS Config rules that are

deployed into each member account

Which solution will meet these requirements?

Options:

A- Create a CloudFormation template that contains the AWS Config rules and remediation actions. Deploy the template from the Organizations management account by using CloudFormation StackSets.

B- Create an AWS Config conformance pack that contains the AWS Config rules and remediation actions Deploy the pack from the Organizations management account by using CloudFormation StackSets.

C- Create a CloudFormation template that contains the AWS Config rules and remediation actions Deploy the template from the delegated administrator account by using AWS Config.

D- Create an AWS Config conformance pack that contains the AWS Config rules and remediation actions. Deploy the pack from the delegated administrator account by using AWS Config.

Answer:

D

Explanation:

The correct answer is D. Creating an AWS Config conformance pack that contains the AWS Config rules and remediation actions and deploying it from the delegated administrator account by using AWS Config will meet the requirements.A conformance pack is a

collection of AWS Config rules and remediation actions that can be easily deployed as a single entity in an account and a region or across an organization in AWS Organizations1. By using the delegated administrator account, the DevOps engineer can centrally manage the conformance pack and prevent non-administrator users from modifying it in the member accounts. Option A is incorrect because creating a CloudFormation template that contains the AWS Config rules and remediation actions and deploying it from the Organizations management account by using CloudFormation StackSets will not prevent non-administrator users from modifying the AWS Config rules in the member accounts. Option B is incorrect because deploying the conformance pack from the Organizations management account by using CloudFormation StackSets will not use the trusted access feature of AWS Config and will require additional permissions and resources. Option C is incorrect because creating a CloudFormation template that contains the AWS Config rules and remediation actions and deploying it from the delegated administrator account by using AWS Config will not leverage the benefits of conformance packs, such as simplified deployment and management.Reference:

Conformance Packs - AWS Config

Certified DevOps Engineer - Professional (DOP-C02) Study Guide(page 176)

Question 12

Question Type: MultipleChoice

A company's application runs on Amazon EC2 instances. The application writes to a log file that records the username, date, time: and source IP address of the login. The log is published to a log group in Amazon CloudWatch Logs

The company is performing a root cause analysis for an event that occurred on the previous day The company needs to know the number of logins for a specific user from the past 7 days

Which solution will provide this information'?

Options:

A- Create a CloudWatch Logs metric filter on the log group Use a filter pattern that matches the username. Publish a CloudWatch metric that sums the number of logins over the past 7 days.

B- Create a CloudWatch Logs subscription on the log group Use a filter pattern that matches the username Publish a CloudWatch metric that sums the number of logins over the past 7 days

C- Create a CloudWatch Logs Insights query that uses an aggregation function to count the number of logins for the username over the past 7 days. Run the query against the log group

D- Create a CloudWatch dashboard. Add a number widget that has a filter pattern that counts the number of logins for the username over the past 7 days directly from the log group

Answer:

C

Explanation:

To analyze and find the number of logins for a specific user from the past 7 days, a CloudWatch Logs Insights query is the most suitable solution. CloudWatch Logs Insights enables you to interactively search and analyze your log data in Amazon CloudWatch Logs. You can use the query language to perform queries that contain multiple commands, including aggregation functions, which can count the occurrences of logins for a specific username over a specified time period. This approach is more direct and efficient than creating a metric filter or subscription, which would require additional steps to publish and sum a metric.Reference:AWS Certified DevOps Engineer - Professional,CloudWatch Logs Insights query syntax,Tutorial: Run a query with an aggregation function,Add or remove a number widget from a CloudWatch dashboard.