www.atmicnetworks.com

ATMIC NETWORKS
YOUR GUIDE TO GLOBAL CERTIFIATION

"Please note that these files may not be up to date. However, the questions will help you understand the exam format and typical question patterns."

www.atmicnetworks .com

Warning: Keep connected with our support team for latest updates

www.atmicnetworks.com

Question: 1

A company has a mobile application that makes HTTP API calls to an Application Load Balancer (ALB). The ALB routes requests to an AWS Lambda function. Many different versions of the application are in use at any given time, including versions that are in testing by a subset of users. The version of the application is defined in the user-agent header that is sent with all requests to the API.
After a series of recent changes to the API, the company has observed issues with the application. The company needs to gather a metric for each API operation by response code for each version of the application that is in use. A DevOps engineer has modified the Lambda function to extract the API operation name, version information from the user-agent header and response code.
Which additional set of actions should the DevOps engineer take to gather the required metrics?

A. Modify the Lambda function to write the API operation name, response code, and version number as a log line to an Amazon CloudWatch Logs log group. Configure a CloudWatch Logs metric filter that increments a metric for each API operation name. Specify response code and application version as dimensions for the metric.

B. Modify the Lambda function to write the API operation name, response code, and version number as a log line to an Amazon CloudWatch Logs log group. Configure a CloudWatch Logs Insights query to populate CloudWatch metrics from the log lines. Specify response code and application version as dimensions for the metric.

C. Configure the ALB access logs to write to an Amazon CloudWatch Logs log group. Modify the Lambda function to respond to the ALB with the API operation name, response code, and version number as response metadata. Configure a CloudWatch Logs metric filter that increments a metric for each API operation name. Specify response code and application version as dimensions for the metric.

D. Configure AWS X-Ray integration on the Lambda function. Modify the Lambda function to create an X-Ray subsegment with the API operation name, response code, and version number. Configure X- Ray insights to extract an aggregated metric for each API operation name and to publish the metric to Amazon CloudWatch. Specify response code and application version as dimensions for the metric.

Answer: A

Explanation:

"Note that the metric filter is different from a log insights query, where the experience is interactive and provides immediate search results for the user to investigate. No automatic action can be invoked from an insights query. Metric filters, on the other hand, will generate metric data in the form of a time series. This lets you create alarms that integrate into your ITSM processes, execute AWS Lambda functions, or even create anomaly detection models."

https://aws.amazon.com/blogs/mt/quantify-custom-application-metrics-with-amazon-cloudwatch- logs-and-metric-filters/

Question: 2

A company provides an application to customers. The application has an Amazon API Gateway REST API that invokes an AWS Lambda function. On initialization, the Lambda function loads a large amount of data from an Amazon DynamoDB table. The data load process results in long cold-start times of 8-10 seconds. The DynamoDB table has DynamoDB Accelerator (DAX) configured. Customers report that the application intermittently takes a long time to respond to requests. The application receives thousands of requests throughout the day. In the middle of the day, the application experiences 10 times more requests than at any other time of the day. Near the end of the day, the application's request volume decreases to 10% of its normal total.
A DevOps engineer needs to reduce the latency of the Lambda function at all times of the day. Which solution will meet these requirements?

A. Configure provisioned concurrency on the Lambda function with a concurrency value of 1. Delete the DAX cluster for the DynamoDB table.

B. Configure reserved concurrency on the Lambda function with a concurrency value of 0.

C. Configure provisioned concurrency on the Lambda function. Configure AWS Application Auto Scaling on the Lambda function with provisioned concurrency values set to a minimum of 1 and a maximum of 100.

D. Configure reserved concurrency on the Lambda function. Configure AWS Application Auto Scaling on the API Gateway API with a reserved concurrency maximum value of 100.

Answer: C

Explanation:

The following are the steps that the DevOps engineer should take to reduce the latency of the Lambda function at all times of the day:

Configure provisioned concurrency on the Lambda function.

Configure AWS Application Auto Scaling on the Lambda function with provisioned concurrency values set to a minimum of 1 and a maximum of 100.

The provisioned concurrency setting ensures that there is always a minimum number of Lambda function instances available to handle requests. The Application Auto Scaling setting will automatically scale the number of Lambda function instances up or down based on the demand for the application.

This solution will ensure that the Lambda function is able to handle the increased load during the middle of the day, while also keeping the cold-start latency low.

The following are the reasons why the other options are not correct:

Option A is incorrect because it will not reduce the cold-start latency of the Lambda function.

Option B is incorrect because it will not scale the number of Lambda function instances up or down based on demand.

Option D is incorrect because it will only configure reserved concurrency on the API Gateway API, which will not affect the Lambda function.

Question: 3

A company is adopting AWS CodeDeploy to automate its application deployments for a Java-Apache Tomcat application with an Apache Webserver. The development team started with a proof of concept, created a deployment group for a developer environment, and performed functional tests within the application. After completion, the team will create additional deployment groups for staging and production.

The current log level is configured within the Apache settings, but the team wants to change this configuration dynamically when the deployment occurs, so that they can set different log level configurations depending on the deployment group without having a different application revision for each group.

How can these requirements be met with the LEAST management overhead and without requiring different script versions for each deployment group?

A. Tag the Amazon EC2 instances depending on the deployment group. Then place a script into the application revision that calls the metadata service and the EC2 API to identify which deployment group the instance is part of. Use this information to configure the log level settings. Reference the script as part of the AfterInstall lifecycle hook in the appspec.yml file.

B. Create a script that uses the CodeDeploy environment variable DEPLOYMENT_GROUP_ NAME to identify which deployment group the instance is part of. Use this information to configure the log level settings. Reference this script as part of the BeforeInstall lifecycle hook in the appspec.yml file.

C. Create a CodeDeploy custom environment variable for each environment. Then place a script into the application revision that checks this environment variable to identify which deployment group the instance is part of. Use this information to configure the log level settings. Reference this script as part of the ValidateService lifecycle hook in the appspec.yml file.

D. Create a script that uses the CodeDeploy environment variable DEPLOYMENT_GROUP_ID to identify which deployment group the

instance is part of to configure the log level settings. Reference this script as part of the Install lifecycle hook in the appspec.yml file.

_____ Answer: B

Explanation:

The following are the steps that the company can take to change the log level dynamically when the deployment occurs:
Create a script that uses the CodeDeploy environment variable DEPLOYMENT_GROUP_NAME to identify which deployment group the instance is part of.
Use this information to configure the log level settings.
Reference this script as part of the BeforeInstall lifecycle hook in the appspec.yml file.
The DEPLOYMENT_GROUP_NAME environment variable is automatically set by CodeDeploy when the deployment is triggered. This means that the script does not need to call the metadata service or the EC2 API to identify the deployment group.
This solution is the least complex and requires the least management overhead. It also does not require different script versions for each deployment group.
The following are the reasons why the other options are not correct:
Option A is incorrect because it would require tagging the Amazon EC2 instances, which would be a manual and time-consuming process.
Option C is incorrect because it would require creating a custom environment variable for each environment. This would be a complex and error-prone process.
Option D is incorrect because it would use the DEPLOYMENT_GROUP_ID environment variable. However, this variable is not automatically set by CodeDeploy, so the script would need to call the metadata service or the EC2 API to get the deployment group ID. This would add complexity and overhead to the solution.

Question: 4

A company requires its developers to tag all Amazon Elastic Block Store (Amazon EBS) volumes in an account to indicate a desired backup frequency. This requirement Includes EBS volumes that do not require backups. The company uses custom tags named Backup_Frequency that have values of none, dally, or weekly that correspond to the desired backup frequency. An audit finds that developers are occasionally not tagging the EBS volumes.
A DevOps engineer needs to ensure that all EBS volumes always have the Backup_Frequency tag so that the company can perform backups at least weekly unless a different value is specified. Which solution will meet these requirements?

A. Set up AWS Config in the account. Create a custom rule that returns a compliance failure for all Amazon EC2 resources that do not have a Backup Frequency tag applied. Configure a remediation action that uses a custom AWS Systems Manager Automation runbook to apply the Backup_Frequency tag with a value of weekly.
B. Set up AWS Config in the account. Use a managed rule that returns a compliance failure for EC2::Volume resources that do not have a Backup Frequency tag applied. Configure a remediation action that uses a custom AWS Systems Manager Automation runbook to apply the Backup_Frequency tag with a value of weekly.
C. Turn on AWS CloudTrail in the account. Create an Amazon EventBridge rule that reacts to EBS CreateVolume events. Configure a custom AWS Systems Manager Automation runbook to apply the Backup_Frequency tag with a value of weekly. Specify the runbook as the target of the rule.
D. Turn on AWS CloudTrail in the account. Create an Amazon EventBridge rule that reacts to EBS CreateVolume events or EBS ModifyVolume events. Configure a custom AWS Systems Manager Automation runbook to apply the Backup_Frequency tag with a value of weekly. Specify the runbook as the target of the rule.

_____ Answer: B

Explanation:

The following are the steps that the DevOps engineer should take to ensure that all EBS volumes always have the Backup_Frequency tag so that the company can perform backups at least weekly unless a different value is specified:
Set up AWS Config in the account.
Use a managed rule that returns a compliance failure for EC2::Volume resources that do not have a Backup Frequency tag applied.
Configure a remediation action that uses a custom AWS Systems Manager Automation runbook to apply the Backup_Frequency tag with a value of weekly.
The managed rule AWS::Config::EBSVolumesWithoutBackupTag will return a compliance failure for any EBS volume that does not have the Backup_Frequency tag applied. The remediation action will then use the Systems Manager Automation runbook to apply the Backup_Frequency tag with a value of weekly to the EBS volume.

Question: 5

A company is using an Amazon Aurora cluster as the data store for its application. The Aurora cluster is configured with a single DB instance. The application performs read and write operations on the database by using the cluster's instance endpoint.
The company has scheduled an update to be applied to the cluster during an upcoming maintenance window. The cluster must remain available with the least possible interruption during the maintenance window.
What should a DevOps engineer do to meet these requirements?

A. Add a reader instance to the Aurora cluster. Update the application to use the Aurora cluster endpoint for write operations. Update the Aurora cluster's reader endpoint for reads.
B. Add a reader instance to the Aurora cluster. Create a custom ANY endpoint for the cluster. Update the application to use the Aurora cluster's custom ANY endpoint for read and write operations.
C. Turn on the Multi-AZ option on the Aurora cluster. Update the application to use the Aurora cluster endpoint for write operations. Update the Aurora cluster's reader endpoint for reads.
D. Turn on the Multi-AZ option on the Aurora cluster. Create a custom ANY endpoint for the cluster. Update the application to use the Aurora cluster's custom ANY endpoint for read and write operations.

_____ Answer: C

Explanation:

To meet the requirements, the DevOps engineer should do the following:
Turn on the Multi-AZ option on the Aurora cluster.
Update the application to use the Aurora cluster endpoint for write operations.
Update the Aurora cluster's reader endpoint for reads.
Turning on the Multi-AZ option will create a replica of the database in a different Availability Zone. This will ensure that the database remains available even if one of the Availability Zones is unavailable.
Updating the application to use the Aurora cluster endpoint for write operations will ensure that all writes are sent to both the primary and replica databases. This will ensure that the data is always consistent.
Updating the Aurora cluster's reader endpoint for reads will allow the application to read data from the replica database. This will improve the performance of the application during the maintenance window.

Question: 6

A company must encrypt all AMIs that the company shares across accounts. A DevOps engineer has access to a source account where an unencrypted custom AMI has been built. The DevOps engineer also has access to a target account where an Amazon EC2 Auto Scaling group will launch EC2 instances from the AMI. The DevOps engineer must share the AMI with the target account.
The company has created an AWS Key Management Service (AWS KMS) key in the source account. Which additional steps should the DevOps engineer perform to meet the requirements? (Choose three.)

A. In the source account, copy the unencrypted AMI to an encrypted AMI. Specify the KMS key in the copy action.

B. In the source account, copy the unencrypted AMI to an encrypted AMI. Specify the default Amazon Elastic Block Store (Amazon EBS) encryption key in the copy action.

C. In the source account, create a KMS grant that delegates permissions to the Auto Scaling group service-linked role in the target account.

D. In the source account, modify the key policy to give the target account permissions to create a grant. In the target account, create a KMS grant that delegates permissions to the Auto Scaling group service-linked role.

E. In the source account, share the unencrypted AMI with the target account.

F. In the source account, share the encrypted AMI with the target account.

Answer: A, D, F

Explanation:

The Auto Scaling group service-linked role must have a specific grant in the source account in order to decrypt the encrypted AMI. This is because the service-linked role does not have permissions to assume the default IAM role in the source account.

The following steps are required to meet the requirements:

In the source account, copy the unencrypted AMI to an encrypted AMI. Specify the KMS key in the copy action.

In the source account, create a KMS grant that delegates permissions to the Auto Scaling group service-linked role in the target account.

In the source account, share the encrypted AMI with the target account.

In the target account, attach the KMS grant to the Auto Scaling group service-linked role.

The first three steps are the same as the steps that I described earlier. The fourth step is required to grant the Auto Scaling group service-linked role permissions to decrypt the AMI in the target account.

Question: 7

A company uses AWS CodePipeline pipelines to automate releases of its application A typical pipeline consists of three stages build, test, and deployment. The company has been using a separate AWS CodeBuild project to run scripts for each stage. However, the company now wants to use AWS CodeDeploy to handle the deployment stage of the pipelines.

The company has packaged the application as an RPM package and must deploy the application to a fleet of Amazon EC2 instances. The EC2 instances are in an EC2 Auto Scaling group and are launched from a common AMI.

Which combination of steps should a DevOps engineer perform to meet these requirements? (Choose two.)

A. Create a new version of the common AMI with the CodeDeploy agent installed. Update the IAM role of the EC2 instances to allow access to CodeDeploy.

B. Create a new version of the common AMI with the CodeDeploy agent installed. Create an AppSpec file that contains application deployment scripts and grants access to CodeDeploy.

C. Create an application in CodeDeploy. Configure an in-place deployment type. Specify the Auto Scaling group as the deployment target. Add a step to the CodePipeline pipeline to use EC2 Image Builder to create a new AMI. Configure CodeDeploy to deploy the newly created AMI.

D. Create an application in CodeDeploy. Configure an in-place deployment type. Specify the Auto Scaling group as the deployment target. Update the CodePipeline pipeline to use the CodeDeploy action to deploy the application.

E. Create an application in CodeDeploy. Configure an in-place deployment type. Specify the EC2 instances that are launched from the common AMI as the deployment target. Update the CodePipeline pipeline to use the CodeDeploy action to deploy the application.

Answer: A, D

Explanation:

https://docs.aws.amazon.com/codedeploy/latest/userguide/integrations-aws-auto-scaling.html

Question: 8

A company's security team requires that all external Application Load Balancers (ALBs) and Amazon API Gateway APIs are associated with AWS WAF web ACLs. The company has hundreds of AWS accounts, all of which are included in a single organization in AWS Organizations. The company has configured AWS Config for the organization. During an audit, the company finds some externally facing ALBs that are not associated with AWS WAF web ACLs.

Which combination of steps should a DevOps engineer take to prevent future violations? (Choose two.)

A. Delegate AWS Firewall Manager to a security account.
B. Delegate Amazon GuardDuty to a security account.
C. Create an AWS Firewall Manager policy to attach AWS WAF web ACLs to any newly created ALBs and API Gateway APIs.
D. Create an Amazon GuardDuty policy to attach AWS WAF web ACLs to any newly created ALBs and API Gateway APIs.
E. Configure an AWS Config managed rule to attach AWS WAF web ACLs to any newly created ALBs and API Gateway APIs.

Answer: A, C

Explanation:

If instead you want to automatically apply the policy to existing in-scope resources, choose Auto remediate any noncompliant resources. This option creates a web ACL in each applicable account within the AWS organization and associates the web ACL with the resources in the accounts. When you choose Auto remediate any noncompliant resources, you can also choose to remove existing web ACL associations from in-scope resources, for the web ACLs that aren't managed by another active Firewall Manager policy. If you choose this option, Firewall Manager first associates the policy's web ACL with the resources, and then removes the prior associations. If a resource has an association with another web ACL that's managed by a different active Firewall Manager policy, this choice doesn't affect that association.

Question: 9

A company uses AWS Key Management Service (AWS KMS) keys and manual key rotation to meet regulatory compliance requirements. The security team wants to be notified when any keys have not been rotated after 90 days.

Which solution will accomplish this?

A. Configure AWS KMS to publish to an Amazon Simple Notification Service (Amazon SNS) topic when keys are more than 90 days old.
B. Configure an Amazon EventBridge event to launch an AWS Lambda function to call the AWS Trusted Advisor API and publish to an Amazon Simple Notification Service (Amazon SNS) topic.
C. Develop an AWS Config custom rule that publishes to an Amazon Simple Notification Service (Amazon SNS) topic when keys are more than 90 days old.
D. Configure AWS Security Hub to publish to an Amazon Simple Notification Service (Amazon SNS) topic when keys are more than 90 days old.

Answer: C

Explanation:

https://aws.amazon.com/blogs/security/how-to-use-aws-config-to-determine-compliance-of-aws- kms-key-policies-to-your-specifications/

www.atmicnetworks.com

Question: 10

A security review has identified that an AWS CodeBuild project is downloading a database population script from an Amazon S3 bucket using an unauthenticated request. The security team does not allow unauthenticated requests to S3 buckets for this project.
How can this issue be corrected in the MOST secure manner?

A. Add the bucket name to the AllowedBuckets section of the CodeBuild project settings. Update the build spec to use the AWS CLI to download the database population script.
B. Modify the S3 bucket settings to enable HTTPS basic authentication and specify a token. Update the build spec to use cURL to pass the token and download the database population script.
C. Remove unauthenticated access from the S3 bucket with a bucket policy. Modify the service role for the CodeBuild project to include Amazon S3 access. Use the AWS CLI to download the database population script.
D. Remove unauthenticated access from the S3 bucket with a bucket policy. Use the AWS CLI to download the database population script using an IAM access key and a secret access key.

_____ Answer: C

Explanation:

A bucket policy is a resource-based policy that defines who can access a specific S3 bucket and what actions they can perform on it. By removing unauthenticated access from the bucket policy, you can prevent anyone without valid credentials from accessing the bucket. A service role is an IAM role that allows an AWS service, such as CodeBuild, to perform actions on your behalf. By modifying the service role for the CodeBuild project to include Amazon S3 access, you can grant the project permission to read and write objects in the S3 bucket. The AWS CLI is a command-line tool that allows you to interact with AWS services, such as S3, using commands in your terminal. By using the AWS CLI to download the database population script, you can leverage the service role credentials and encryption to secure the data transfer.
For more information, you can refer to these web pages:
[Using bucket policies and user policies - Amazon Simple Storage Service]
[Create a service role for CodeBuild - AWS CodeBuild]
[AWS Command Line Interface]

Question: 11

An ecommerce company has chosen AWS to host its new platform. The company's DevOps team has started building an AWS Control Tower landing zone. The DevOps team has set the identity store within AWS IAM Identity Center (AWS Single Sign-On) to external identity provider (IdP) and has configured SAML 2.0.
The DevOps team wants a robust permission model that applies the principle of least privilege. The model must allow the team to build and manage only the team's own resources.
Which combination of steps will meet these requirements? (Choose three.)

A. Create IAM policies that include the required permissions. Include the aws:PrincipalTag condition key.
B. Create permission sets. Attach an inline policy that includes the required permissions and uses the aws:PrincipalTag condition key to scope the permissions.
C. Create a group in the IdP. Place users in the group. Assign the group to accounts and the permission sets in IAM Identity Center.
D. Create a group in the IdP. Place users in the group. Assign the group to OUs and IAM policies.
E. Enable attributes for access control in IAM Identity Center. Apply tags to users. Map the tags as key-value pairs.

F. Enable attributes for access control in IAM Identity Center. Map attributes from the IdP as keyvalue pairs.

Answer: BCF

Explanation:

Using the principalTag in the Permission Set inline policy a logged in user belonging to a specific AD group in the IDP can be permitted access to perform operations on certain resources if their group matches the group used in the PrincipleTag. Basically you are narrowing the scope of privileges assigned via Permission policies conditionally based on whether the logged in user belongs to a specific AD Group in IDP. The mapping of the AD group to the request attributes can be done using SSO attributes where we can pass other attributes like the SAML token as well.

https://docs.aws.amazon.com/singlesignon/latest/userguide/abac.html

Question: 12

An ecommerce company is receiving reports that its order history page is experiencing delays in reflecting the processing status of orders. The order processing system consists of an AWS Lambda function that uses reserved concurrency. The Lambda function processes order messages from an Amazon Simple Queue Service (Amazon SQS) queue and inserts processed orders into an Amazon DynamoDB table. The DynamoDB table has auto scaling enabled for read and write capacity.

Which actions should a DevOps engineer take to resolve this delay? (Choose two.)

A. Check the ApproximateAgeOfOldestMessage metric for the SQS queue. Increase the Lambda function concurrency limit.
B. Check the ApproximateAgeOfOldestMessage metnc for the SQS queue Configure a redrive policy on the SQS queue.
C. Check the NumberOfMessagesSent metric for the SQS queue. Increase the SQS queue visibility timeout.
D. Check the WriteThrottleEvents metric for the DynamoDB table. Increase the maximum write capacity units (WCUs) for the table's scaling policy.
E. Check the Throttles metric for the Lambda function. Increase the Lambda function timeout.

Answer: A, D

Explanation:

A: If the ApproximateAgeOfOldestMessages indicate that orders are remaining in the SQS queue for longer than expected, the reserved concurrency limit may be set too small to keep up with the number of orders entering the queue and is being throttled. D: The DynamoDB table is using Auto Scaling. With Auto Scaling, you create a scaling policy that specifies whether you want to scale read capacity or write capacity (or both), and the minimum and maximum provisioned capacity unit settings for the table. The ThottledWriteRequests metric will indicate if there is a throttling issue on the DynamoDB table, which can be resolved by increasing the maximum write capacity units for the table's Auto Scaling policy.

https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/AutoScaling.html

Question: 13

A company has a single AWS account that runs hundreds of Amazon EC2 instances in a single AWS Region. New EC2 instances are launched and terminated each hour in the account. The account also includes existing EC2 instances that have been running for longer than a week.

The company's security policy requires all running EC2 instances to use an EC2 instance profile. If an EC2 instance does not have an instance profile attached, the EC2 instance must use a default instance profile that has no IAM permissions assigned.

A DevOps engineer reviews the account and discovers EC2 instances that are running without an instance profile. During the review, the DevOps engineer also observes that new EC2 instances are being launched without an instance profile.

Which solution will ensure that an instance profile is attached to all existing and future EC2 instances in the Region?

A. Configure an Amazon EventBridge rule that reacts to EC2 RunInstances API calls. Configure the rule to invoke an AWS Lambda function to attach the default instance profile to the EC2 instances.
B. Configure the ec2-instance-profile-attached AWS Config managed rule with a trigger type of configuration changes. Configure an automatic remediation action that invokes an AWS Systems Manager Automation runbook to attach the default instance profile to the EC2 instances.
C. Configure an Amazon EventBridge rule that reacts to EC2 StartInstances API calls. Configure the rule to invoke an AWS Systems Manager Automation runbook to attach the default instance profile to the EC2 instances.
D. Configure the iam-role-managed-policy-check AWS Config managed rule with a trigger type of configuration changes. Configure an automatic remediation action that invokes an AWS Lambda function to attach the default instance profile to the EC2 instances.

Answer: B

Explanation:

https://docs.aws.amazon.com/config/latest/developerguide/ec2-instance-profile-attached.html

Question: 14

A DevOps engineer is building a continuous deployment pipeline for a serverless application that uses AWS Lambda functions. The company wants to reduce the customer impact of an unsuccessful deployment. The company also wants to monitor for issues.
Which deploy stage configuration will meet these requirements?

A. Use an AWS Serverless Application Model (AWS SAM) template to define the serverless application. Use AWS CodeDeploy to deploy the Lambda functions with the Canary10Percent15Minutes Deployment Preference Type. Use Amazon CloudWatch alarms to monitor the health of the functions.
B. Use AWS CloudFormation to publish a new stack update, and include Amazon CloudWatch alarms on all resources. Set up an AWS CodePipeline approval action for a developer to verify and approve the AWS CloudFormation change set.
C. Use AWS CloudFormation to publish a new version on every stack update, and include Amazon CloudWatch alarms on all resources. Use the RoutingConfig property of the AWS::Lambda::Alias resource to update the traffic routing during the stack update.
D. Use AWS CodeBuild to add sample event payloads for testing to the Lambda functions. Publish a new version of the functions, and include Amazon CloudWatch alarms. Update the production alias to point to the new version. Configure rollbacks to occur when an alarm is in the ALARM state.

Answer: D

Explanation:

Use routing configuration on an alias to send a portion of traffic to a second function version. For example, you can reduce the risk of deploying a new version by configuring the alias to send most of the traffic to the existing version, and only a small percentage of traffic to the new version.

https://docs.aws.amazon.com/lambda/latest/dg/configuration-aliases.html

The following are the steps involved in the deploy stage configuration that will meet the requirements:
Use AWS CodeBuild to add sample event payloads for testing to the Lambda functions.
Publish a new version of the functions, and include Amazon CloudWatch alarms.
Update the production alias to point to the new version.
Configure rollbacks to occur when an alarm is in the ALARM state.
This configuration will help to reduce the customer impact of an unsuccessful deployment by deploying the new version of the functions

to a staging environment first. This will allow the DevOps engineer to test the new version of the functions before deploying it to production.
The configuration will also help to monitor for issues by including Amazon CloudWatch alarms. These alarms will alert the DevOps engineer if there are any problems with the new version of the functions.

Question: 15

To run an application, a DevOps engineer launches an Amazon EC2 instance with public IP addresses in a public subnet. A user data script obtains the application artifacts and installs them on the instances upon launch. A change to the security classification of the application now requires the instances to run with no access to the internet. While the instances launch successfully and show as healthy, the application does not seem to be installed.
Which of the following should successfully install the application while complying with the new rule?

A. Launch the instances in a public subnet with Elastic IP addresses attached. Once the application is installed and running, run a script to disassociate the Elastic IP addresses afterwards.
B. Set up a NAT gateway. Deploy the EC2 instances to a private subnet. Update the private subnet's route table to use the NAT gateway as the default route.
C. Publish the application artifacts to an Amazon S3 bucket and create a VPC endpoint for S3. Assign an IAM instance profile to the EC2 instances so they can read the application artifacts from the S3 bucket.
D. Create a security group for the application instances and allow only outbound traffic to the artifact repository. Remove the security group rule once the install is complete.

_____ Answer: C

Explanation:

EC2 instances running in private subnets of a VPC can now have controlled access to S3 buckets, objects, and API functions that are in the same region as the VPC. You can use an S3 bucket policy to indicate which VPCs and which VPC Endpoints have access to your S3 buckets
1-
https://aws.amazon.com/pt/blogs/aws/new-vpc-endpoint-for-amazon-s3/

Question: 16

A development team is using AWS CodeCommit to version control application code and AWS CodePipeline to orchestrate software deployments. The team has decided to use a remote main branch as the trigger for the pipeline to integrate code changes. A developer has pushed code changes to the CodeCommit repository, but noticed that the pipeline had no reaction, even after 10 minutes.
Which of the following actions should be taken to troubleshoot this issue?

A. Check that an Amazon EventBridge rule has been created for the main branch to trigger the pipeline.
B. Check that the CodePipeline service role has permission to access the CodeCommit repository.
C. Check that the developer's IAM role has permission to push to the CodeCommit repository.
D. Check to see if the pipeline failed to start because of CodeCommit errors in Amazon CloudWatch Logs.

_____ Answer: A

Explanation:

When you create a pipeline from CodePipeline during the step-by-step it creates a CloudWatch Event rule for a given branch and repo like this:

```
{
"source": [
"aws.codecommit"
],
"detail-type": [
"CodeCommit Repository State Change"
],
"resources": [
"arn:aws:codecommit:us-east-1:xxxxx:repo-name"
],
"detail": {
"event": [
"referenceCreated", "referenceUpdated"
],
"referenceType": [
"branch"
],
"referenceName": [
"master"
]
} }
```

https://docs.aws.amazon.com/codepipeline/latest/userguide/pipelines-trigger-source-repo- changes-console.html

Question: 17

A company's developers use Amazon EC2 instances as remote workstations. The company is concerned that users can create or modify EC2 security groups to allow unrestricted inbound access. A DevOps engineer needs to develop a solution to detect when users create unrestricted security group rules. The solution must detect changes to security group rules in near real time, remove unrestricted rules, and send email notifications to the security team. The DevOps engineer has created an AWS Lambda function that checks for security group ID from input, removes rules that grant unrestricted access, and sends notifications through Amazon Simple Notification Service (Amazon SNS).

What should the DevOps engineer do next to meet the requirements?

A. Configure the Lambda function to be invoked by the SNS topic. Create an AWS CloudTrail subscription for the SNS topic. Configure a subscription filter for security group modification events.

B. Create an Amazon EventBridge scheduled rule to invoke the Lambda function. Define a schedule pattern that runs the Lambda function every hour.

C. Create an Amazon EventBridge event rule that has the default event bus as the source. Define the rule's event pattern to match EC2 security group creation and modification events. Configure the rule to invoke the Lambda function.

D. Create an Amazon EventBridge custom event bus that subscribes to events from all AWS services. Configure the Lambda function to be invoked by the custom event bus.

_____ Answer: C

Explanation:

To meet the requirements, the DevOps engineer should create an Amazon EventBridge event rule that has the default event bus as the source. The rule's event pattern should match EC2 security group creation and modification events, and it should be configured to invoke

the Lambda function. This solution will allow for near real-time detection of security group rule changes and will trigger the Lambda function to remove any unrestricted rules and send email notifications to the security team.
https://repost.aws/knowledge-center/monitor-security-group-changes-ec2

Question: 18

A DevOps engineer is creating an AWS CloudFormation template to deploy a web service. The web service will run on Amazon EC2 instances in a private subnet behind an Application Load Balancer (ALB). The DevOps engineer must ensure that the service can accept requests from clients that have IPv6 addresses.

What should the DevOps engineer do with the CloudFormation template so that IPv6 clients can access the web service?

A. Add an IPv6 CIDR block to the VPC and the private subnet for the EC2 instances. Create route table entries for the IPv6 network, use EC2 instance types that support IPv6, and assign IPv6 addresses to each EC2 instance.
B. Assign each EC2 instance an IPv6 Elastic IP address. Create a target group, and add the EC2 instances as targets. Create a listener on port 443 of the ALB, and associate the target group with the ALB.
C. Replace the ALB with a Network Load Balancer (NLB). Add an IPv6 CIDR block to the VPC and subnets for the NLB, and assign the NLB an IPv6 Elastic IP address.
D. Add an IPv6 CIDR block to the VPC and subnets for the ALB. Create a listener on port 443. and specify the dualstack IP address type on the ALB. Create a target group, and add the EC2 instances as targets. Associate the target group with the ALB.

_____ Answer: D

Explanation:

it involves adding an IPv6 CIDR block to the VPC and subnets for the ALB and specifying the dualstack IP address type on the ALB listener. This allows the ALB to listen on both IPv4 and IPv6 addresses, and forward requests to the EC2 instances that are added as targets to the target group associated with the ALB.

Question: 19 A company uses AWS Organizations and AWS Control Tower to manage all the company's AWS accounts. The company uses the Enterprise Support plan.

A DevOps engineer is using Account Factory for Terraform (AFT) to provision new accounts. When new accounts are provisioned, the DevOps engineer notices that the support plan for the new accounts is set to the Basic Support plan. The DevOps engineer needs to implement a solution to provision the new accounts with the Enterprise Support plan.
Which solution will meet these requirements?

A. Use an AWS Config conformance pack to deploy the account-part-of-organizations AWS Config rule and to automatically remediate any noncompliant accounts.
B. Create an AWS Lambda function to create a ticket for AWS Support to add the account to the Enterprise Support plan. Grant the Lambda function the support:ResolveCase permission.
C. Add an additional value to the control_tower_parameters input to set the AWSEnterpriseSupport parameter as the organization's management account number.
D. Set the aft_feature_enterprise_support feature flag to True in the AFT deployment input configuration. Redeploy AFT and apply the changes.

_____ Answer: D

Explanation:

AWS Organizations is a service that helps to manage multiple AWS accounts. AWS Control Tower is a service that makes it easy to set up and govern secure, compliant multi-account AWS environments. Account Factory for Terraform (AFT) is an AWS Control Tower feature that provisions new accounts using Terraform templates. To provision new accounts with the Enterprise Support plan, the DevOps engineer can set the aft_feature_enterprise_support feature flag to True in the AFT deployment input configuration. This flag enables the Enterprise Support plan for newly provisioned accounts. https://docs.aws.amazon.com/controltower/latest/userguide/aft-feature-options.html

Question: 20

A company's DevOps engineer uses AWS Systems Manager to perform maintenance tasks during maintenance windows. The company has a few Amazon EC2 instances that require a restart after notifications from AWS Health. The DevOps engineer needs to implement an automated solution to remediate these notifications. The DevOps engineer creates an Amazon EventBridge rule.
How should the DevOps engineer configure the EventBridge rule to meet these requirements?

A. Configure an event source of AWS Health, a service of EC2. and an event type that indicates instance maintenance. Target a Systems Manager document to restart the EC2 instance.
B. Configure an event source of Systems Manager and an event type that indicates a maintenance window. Target a Systems Manager document to restart the EC2 instance.
C. Configure an event source of AWS Health, a service of EC2, and an event type that indicates instance maintenance. Target a newly created AWS Lambda function that registers an automation task to restart the EC2 instance during a maintenance window.
D. Configure an event source of EC2 and an event type that indicates instance maintenance. Target a newly created AWS Lambda function that registers an automation task to restart the EC2 instance during a maintenance window.

Answer: C

Explanation:

AWS Health provides real-time events and information related to your AWS infrastructure. It can be integrated with Amazon EventBridge to act upon the health events automatically. If the maintenance notification from AWS Health indicates that an EC2 instance requires a restart, you can set up an EventBridge rule to respond to such events. In this case, the target of this rule would be a Lambda function that would trigger a Systems Manager automation to restart the EC2 instance during a maintenance window. Remember, AWS Health is the source of the events (not EC2 or Systems Manager), and AWS Lambda can be used to execute complex remediation tasks, such as scheduling maintenance tasks via Systems Manager.
The following are the steps involved in configuring the EventBridge rule to meet these requirements: Configure an event source of AWS Health, a service of EC2, and an event type that indicates instance maintenance.
Target a newly created AWS Lambda function that registers an automation task to restart the EC2 instance during a maintenance window.
The AWS Lambda function will be triggered by the event from AWS Health. The function will then register an automation task to restart the EC2 instance during the next maintenance window.

Question: 21

A company has containerized all of its in-house quality control applications. The company is running Jenkins on Amazon EC2 instances, which require patching and upgrading. The compliance officer has requested a DevOps engineer begin encrypting build artifacts since they contain company intellectual property.
What should the DevOps engineer do to accomplish this in the MOST maintainable manner?

A. Automate patching and upgrading using AWS Systems Manager on EC2 instances and encrypt Amazon EBS volumes by default.
B. Deploy Jenkins to an Amazon ECS cluster and copy build artifacts to an Amazon S3 bucket with default encryption enabled.

C. Leverage AWS CodePipeline with a build action and encrypt the artifacts using AWS Secrets Manager.

D. Use AWS CodeBuild with artifact encryption to replace the Jenkins instance running on EC2 instances.

Answer: D

Explanation:

The following are the steps involved in accomplishing this in the most maintainable manner:
Use AWS CodeBuild with artifact encryption to replace the Jenkins instance running on EC2 instances.
Configure CodeBuild to encrypt the build artifacts using AWS Secrets Manager.
Deploy the containerized quality control applications to CodeBuild.
This approach is the most maintainable because it eliminates the need to manage Jenkins on EC2 instances. CodeBuild is a managed service, so the DevOps engineer does not need to worry about patching or upgrading the service.
https://docs.aws.amazon.com/codebuild/latest/userguide/security-encryption.html Build artifact encryption - CodeBuild requires access to an AWS KMS CMK in order to encrypt its build output artifacts. By default, CodeBuild uses an AWS Key Management Service CMK for Amazon S3 in your AWS account. If you do not want to use this CMK, you must create and configure a customermanaged CMK. For more information Creating keys.

Question: 22

An IT team has built an AWS CloudFormation template so others in the company can quickly and reliably deploy and terminate an application. The template creates an Amazon EC2 instance with a user data script to install the application and an Amazon S3 bucket that the application uses to serve static webpages while it is running.
All resources should be removed when the CloudFormation stack is deleted. However, the team observes that CloudFormation reports an error during stack deletion, and the S3 bucket created by the stack is not deleted.
How can the team resolve the error in the MOST efficient manner to ensure that all resources are deleted without errors?

A. Add a DelelionPolicy attribute to the S3 bucket resource, with the value Delete forcing the bucket to be removed when the stack is deleted.

B. Add a custom resource with an AWS Lambda function with the DependsOn attribute specifying the S3 bucket, and an IAM role. Write the Lambda function to delete all objects from the bucket when RequestType is Delete.

C. Identify the resource that was not deleted. Manually empty the S3 bucket and then delete it.

D. Replace the EC2 and S3 bucket resources with a single AWS OpsWorks Stacks resource. Define a custom recipe for the stack to create and delete the EC2 instance and the S3 bucket.

Answer: B

Explanation:

https://aws.amazon.com/premiumsupport/knowledge-center/cloudformation-s3-custom-resources/

Question: 23

A company has an AWS CodePipeline pipeline that is configured with an Amazon S3 bucket in the eu- west-1 Region. The pipeline deploys an AWS Lambda application to the same Region. The pipeline consists of an AWS CodeBuild project build action and an AWS CloudFormation deploy action. The CodeBuild project uses the aws cloudformation package AWS CLI command to build an artifact that contains the Lambda function code's .zip file and the CloudFormation template. The CloudFormation deploy action references the CloudFormation template from the output artifact of the CodeBuild project's build action.

www.atmicnetworks.com

The company wants to also deploy the Lambda application to the us-east-1 Region by using the pipeline in eu-west-1. A DevOps engineer has already updated the CodeBuild project to use the aws cloudformation package command to produce an additional output artifact for us-east-1. Which combination of additional steps should the DevOps engineer take to meet these requirements? (Choose two.)

A. Modify the CloudFormation template to include a parameter for the Lambda function code's zip file location. Create a new CloudFormation deploy action for us-east-1 in the pipeline. Configure the new deploy action to pass in the us-east-1 artifact location as a parameter override.

B. Create a new CloudFormation deploy action for us-east-1 in the pipeline. Configure the new deploy action to use the CloudFormation template from the us-east-1 output artifact.

C. Create an S3 bucket in us-east-1. Configure the S3 bucket policy to allow CodePipeline to have read and write access.

D. Create an S3 bucket in us-east-1. Configure S3 Cross-Region Replication (CRR) from the S3 bucket in eu-west-1 to the S3 bucket in us-east-1.

E. Modify the pipeline to include the S3 bucket for us-east-1 as an artifact store. Create a new CloudFormation deploy action for us-east-1 in the pipeline. Configure the new deploy action to use the CloudFormation template from the us-east-1 output artifact.

Answer: A, B

Explanation:

A) The CloudFormation template should be modified to include a parameter that indicates the location of the .zip file containing the Lambda function's code. This allows the CloudFormation deploy action to use the correct artifact depending on the region. This is critical because Lambda functions need to reference their code artifacts from the same region they are being deployed in. B. You would also need to create a new CloudFormation deploy action for the us-east-1 Region within the pipeline. This action should be configured to use the CloudFormation template from the artifact that was specifically created for us-east-1.

Question: 24

A company runs an application on one Amazon EC2 instance. Application metadata is stored in Amazon S3 and must be retrieved if the instance is restarted. The instance must restart or relaunch automatically if the instance becomes unresponsive.
Which solution will meet these requirements?

A. Create an Amazon CloudWatch alarm for the StatusCheckFailed metric. Use the recover action to stop and start the instance. Use an S3 event notification to push the metadata to the instance when the instance is back up and running.

B. Configure AWS OpsWorks, and use the auto healing feature to stop and start the instance. Use a lifecycle event in OpsWorks to pull the metadata from Amazon S3 and update it on the instance.

C. Use EC2 Auto Recovery to automatically stop and start the instance in case of a failure. Use an S3 event notification to push the metadata to the instance when the instance is back up and running.

D. Use AWS CloudFormation to create an EC2 instance that includes the UserData property for the EC2 resource. Add a command in UserData to retrieve the application metadata from Amazon S3.

_____ Answer: B

Explanation:

https://aws.amazon.com/blogs/mt/how-to-set-up-aws-opsworks-stacks-auto-healing-notifications-in-amazon-cloudwatch-events/

www.atmicnetworks.com

Question: 25

A company has multiple AWS accounts. The company uses AWS IAM Identity Center (AWS Single Sign-On) that is integrated with AWS Toolkit for Microsoft Azure DevOps. The attributes for access control feature is enabled in IAM Identity Center.
The attribute mapping list contains two entries. The department key is mapped to ${path:enterprise.department}. The costCenter key is mapped to ${path:enterprise.costCenter}.
All existing Amazon EC2 instances have a department tag that corresponds to three company departments (d1, d2, d3). A DevOps engineer must create policies based on the matching attributes. The policies must minimize administrative effort and must grant each Azure AD user access to only the EC2 instances that are tagged with the user's respective department name.
Which condition key should the DevOps engineer include in the custom permissions policies to meet these requirements?

A.

```
''Condition": {
      "ForAllValues:StringEquals": {
            "aws:TagKeys": ["department"]
      }
}
```

B.

```
"Condition": {
    "StringEquals": {
        "aws:PrincipalTag/department"        "$(aws:ResourceTag/department)"
```

C.

```
"Condition": { "StringEquals": (
        "ec2:ResourceTag/department": )
}                                        $(aws:PrincipalTag/department)"
```

D.

"Condition": { "ForAllValues:StringEquals" "ec2:ResourceTag/department": ["dl", "d2", "d3"]

Answer: C

Explanation:
https://docs.aws.amazon.com/singlesignon/latest/userguide/configure-abac.html

Question: 26

A company hosts a security auditing application in an AWS account. The auditing application uses an IAM role to access other AWS accounts. All the accounts are in the same organization in AWS Organizations.
A recent security audit revealed that users in the audited AWS accounts could modify or delete the auditing application's IAM role. The company needs to prevent any modification to the auditing application's IAM role by any entity other than a trusted administrator IAM

role.

Which solution will meet these requirements?

A. Create an SCP that includes a Deny statement for changes to the auditing application's IAM role. Include a condition that allows the trusted administrator IAM role to make changes. Attach the SCP to the root of the organization.

B. Create an SCP that includes an Allow statement for changes to the auditing application's IAM role by the trusted administrator IAM role. Include a Deny statement for changes by all other IAM principals. Attach the SCP to the IAM service in each AWS account where the auditing application has an IAM role.

C. Create an IAM permissions boundary that includes a Deny statement for changes to the auditing application's IAM role. Include a condition that allows the trusted administrator IAM role to make changes. Attach the permissions boundary to the audited AWS accounts.

D. Create an IAM permissions boundary that includes a Deny statement for changes to the auditing application's IAM role. Include a condition that allows the trusted administrator IAM role to make changes. Attach the permissions boundary to the auditing application's IAM role in the AWS accounts.

_____ Answer: A

Explanation:

https://docs.aws.amazon.com/organizations/latest/userguide/orgs manage policies scps.html?icm pid=docs orgs console

SCPs (Service Control Policies) are the best way to restrict permissions at the organizational level, which in this case would be used to restrict modifications to the IAM role used by the auditing application, while still allowing trusted administrators to make changes to it. Options C and D are not as effective because IAM permission boundaries are applied to IAM entities (users, groups, and roles), not the account itself, and must be applied to all IAM entities in the account.

Question: 27

A company has an on-premises application that is written in Go. A DevOps engineer must move the application to AWS. The company's development team wants to enable blue/green deployments and perform A/B testing.

Which solution will meet these requirements?

A. Deploy the application on an Amazon EC2 instance, and create an AMI of the instance. Use the

AMI to create an automatic scaling launch configuration that is used in an Auto Scaling group. Use Elastic Load Balancing to distribute traffic. When changes are made to the application, a new AMI will be created, which will initiate an EC2 instance refresh.

B. Use Amazon Lightsail to deploy the application. Store the application in a zipped format in an Amazon S3 bucket. Use this zipped version to deploy new versions of the application to Lightsail. Use Lightsail deployment options to manage the deployment.

C. Use AWS CodeArtifact to store the application code. Use AWS CodeDeploy to deploy the application to a fleet of Amazon EC2 instances. Use Elastic Load Balancing to distribute the traffic to the EC2 instances. When making changes to the application, upload a new version to CodeArtifact and create a new CodeDeploy deployment.

D. Use AWS Elastic Beanstalk to host the application. Store a zipped version of the application in Amazon S3. Use that location to deploy new versions of the application. Use Elastic Beanstalk to manage the deployment options.

_____ Answer: D

Explanation:

https://aws.amazon.com/quickstart/architecture/blue-green-deployment/

www.atmicnetworks.com

Question: 28

A developer is maintaining a fleet of 50 Amazon EC2 Linux servers. The servers are part of an Amazon EC2 Auto Scaling group, and also use Elastic Load Balancing for load balancing.
Occasionally, some application servers are being terminated after failing ELB HTTP health checks. The developer would like to perform a root cause analysis on the issue, but before being able to access application logs, the server is terminated.
How can log collection be automated?

A. Use Auto Scaling lifecycle hooks to put instances in a Pending:Wait state. Create an Amazon CloudWatch alarm for EC2 Instance Terminate Successful and trigger an AWS Lambda function that invokes an SSM Run Command script to collect logs, push them to Amazon S3, and complete the lifecycle action once logs are collected.
B. Use Auto Scaling lifecycle hooks to put instances in a Terminating:Wait state. Create an AWS Config rule for EC2 Instance-terminate Lifecycle Action and trigger a step function that invokes a script to collect logs, push them to Amazon S3, and complete the lifecycle action once logs are collected.
C. Use Auto Scaling lifecycle hooks to put instances in a Terminating:Wait state. Create an Amazon CloudWatch subscription filter for EC2 Instance Terminate Successful and trigger a CloudWatch agent that invokes a script to collect logs, push them to Amazon S3, and complete the lifecycle action once logs are collected.
D. Use Auto Scaling lifecycle hooks to put instances in a Terminating:Wait state. Create an Amazon EventBridge rule for EC2 Instance-terminate Lifecycle Action and trigger an AWS Lambda function that invokes an SSM Run Command script to collect logs, push them to Amazon S3, and complete the lifecycle action once logs are collected.

_____ Answer: D

Explanation:

https://blog.fourninecloud.com/auto-scaling-lifecycle-hooks-to-export-server-logs-when-instance- terminating-58e06d7c0d6a

Question: 29

A company has an organization in AWS Organizations. The organization includes workload accounts that contain enterprise applications. The company centrally manages users from an operations account. No users can be created in the workload accounts. The company recently added an operations team and must provide the operations team members with administrator access to each workload account.
Which combination of actions will provide this access? (Choose three.)

A. Create a SysAdmin role in the operations account. Attach the AdministratorAccess policy to the role. Modify the trust relationship to allow the sts:AssumeRole action from the workload accounts.
B. Create a SysAdmin role in each workload account. Attach the AdministratorAccess policy to the role. Modify the trust relationship to allow the sts:AssumeRole action from the operations account.
C. Create an Amazon Cognito identity pool in the operations account. Attach the SysAdmin role as an authenticated role.
D. In the operations account, create an IAM user for each operations team member.
E. In the operations account, create an IAM user group that is named SysAdmins. Add an IAM policy that allows the sts:AssumeRole action for the SysAdmin role in each workload account. Add all operations team members to the group.
F. Create an Amazon Cognito user pool in the operations account. Create an Amazon Cognito user for each operations team member.

Answer: BDE

Explanation:

https://docs.aws.amazon.com/IAM/latest/UserGuide/tutorial cross-account-with-roles.html

Question: 30

A company has multiple accounts in an organization in AWS Organizations. The company's SecOps team needs to receive an Amazon Simple Notification Service (Amazon SNS) notification if any account in the organization turns off the Block Public Access feature on an Amazon S3 bucket. A DevOps engineer must implement this change without affecting the operation of any AWS accounts. The implementation must ensure that individual member accounts in the organization cannot turn off the notification.
Which solution will meet these requirements?

A. Designate an account to be the delegated Amazon GuardDuty administrator account. Turn on GuardDuty for all accounts across the organization. In the GuardDuty administrator account, create an SNS topic. Subscribe the SecOps team's email address to the SNS topic. In the same account, create an Amazon EventBridge rule that uses an event pattern for GuardDuty findings and a target of the SNS topic.
B. Create an AWS CloudFormation template that creates an SNS topic and subscribes the SecOps team's email address to the SNS topic. In the template, include an Amazon EventBridge rule that uses an event pattern of CloudTrail activity for s3:PutBucketPublicAccessBlock and a target of the SNS topic. Deploy the stack to every account in the organization by using CloudFormation StackSets.
C. Turn on AWS Config across the organization. In the delegated administrator account, create an SNS topic. Subscribe the SecOps team's email address to the SNS topic. Deploy a conformance pack that uses the s3-bucket-level-public-access-prohibited AWS Config managed rule in each account and uses an AWS Systems Manager document to publish an event to the SNS topic to notify the SecOps team.
D. Turn on Amazon Inspector across the organization. In the Amazon Inspector delegated administrator account, create an SNS topic. Subscribe the SecOps team's email address to the SNS topic. In the same account, create an Amazon EventBridge rule that uses an event pattern for public network exposure of the S3 bucket and publishes an event to the SNS topic to notify the SecOps team.

_____ Answer: C

Explanation:

Amazon GuardDuty is primarily on threat detection and response, not configuration monitoring 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
Organizations. https://docs.aws.amazon.com/config/latest/developerguide/conformance- packs.html
https://docs.aws.amazon.com/config/latest/developerguide/s3-account-level-public-access- blocks.html

Question: 31

A company has migrated its container-based applications to Amazon EKS and want to establish automated email notifications. The notifications sent to each email address are for specific activities related to EKS components. The solution will include Amazon SNS topics and an AWS Lambda function to evaluate incoming log events and publish messages to the correct SNS topic.
Which logging solution will support these requirements?

A. Enable Amazon CloudWatch Logs to log the EKS components. Create a CloudWatch subscription filter for each component with Lambda as the subscription feed destination.
B. Enable Amazon CloudWatch Logs to log the EKS components. Create CloudWatch Logs Insights queries linked to Amazon EventBridge events that invoke Lambda.
C. Enable Amazon S3 logging for the EKS components. Configure an Amazon CloudWatch subscription filter for each component with Lambda as the subscription feed destination.

D. Enable Amazon S3 logging for the EKS components. Configure S3 PUT Object event notifications with AWS Lambda as the destination.

Answer: A

Explanation:

https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/SubscriptionFilters.html#LambdaFunctionExample
https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/SubscriptionFilters.html

Question: 32

A company is implementing an Amazon Elastic Container Service (Amazon ECS) cluster to run its workload. The company architecture will run multiple ECS services on the cluster. The architecture includes an Application Load Balancer on the front end and uses multiple target groups to route traffic.
A DevOps engineer must collect application and access logs. The DevOps engineer then needs to send the logs to an Amazon S3 bucket for near-real-time analysis.
Which combination of steps must the DevOps engineer take to meet these requirements? (Choose three.)

A. Download the Amazon CloudWatch Logs container instance from AWS. Configure this instance as a task. Update the application service definitions to include the logging task.
B. Install the Amazon CloudWatch Logs agent on the ECS instances. Change the logging driver in the ECS task definition to awslogs.
C. Use Amazon EventBridge to schedule an AWS Lambda function that will run every 60 seconds and will run the Amazon CloudWatch Logs create-export-task command. Then point the output to the logging S3 bucket.
D. Activate access logging on the ALB. Then point the ALB directly to the logging S3 bucket.
E. Activate access logging on the target groups that the ECS services use. Then send the logs directly to the logging S3 bucket.
F. Create an Amazon Kinesis Data Firehose delivery stream that has a destination of the logging S3 bucket. Then create an Amazon CloudWatch Logs subscription filter for Kinesis Data Firehose.

Answer: BDF

Explanation:

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-logging-monitoring.html

Question: 33

A company that uses electronic health records is running a fleet of Amazon EC2 instances with an Amazon Linux operating system. As part of patient privacy requirements, the company must ensure continuous compliance for patches for operating system and applications running on the EC2 instances.
How can the deployments of the operating system and application patches be automated using a default and custom repository?

A. Use AWS Systems Manager to create a new patch baseline including the custom repository. Run the AWS-RunPatchBaseline document using the run command to verify and install patches.
B. Use AWS Direct Connect to integrate the corporate repository and deploy the patches using Amazon CloudWatch scheduled events, then use the CloudWatch dashboard to create reports.
C. Use yum-config-manager to add the custom repository under /etc/yum.repos.d and run yum- config-manager-enable to

activate the repository.

D. Use AWS Systems Manager to create a new patch baseline including the corporate repository. Run the AWS-AmazonLinuxDefaultPatchBaseline document using the run command to verify and install patches.

Answer: A

Explanation:

https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-manager-how-it-works-alt-source-repository.html

Question: 34

A company is using AWS CodePipeline to automate its release pipeline. AWS CodeDeploy is being used in the pipeline to deploy an application to Amazon Elastic Container Service (Amazon ECS) using the blue/green deployment model. The company wants to implement scripts to test the green version of the application before shifting traffic. These scripts will complete in 5 minutes or less. If errors are discovered during these tests, the application must be rolled back.
Which strategy will meet these requirements?

A. Add a stage to the CodePipeline pipeline between the source and deploy stages. Use AWS CodeBuild to create a runtime environment and build commands in the buildspec file to invoke test scripts. If errors are found, use the aws deploy stop-deployment command to stop the deployment.

B. Add a stage to the CodePipeline pipeline between the source and deploy stages. Use this stage to invoke an AWS Lambda function that will run the test scripts. If errors are found, use the aws deploy stop-deployment command to stop the deployment.

C. Add a hooks section to the CodeDeploy AppSpec file. Use the AfterAllowTestTraffic lifecycle event to invoke an AWS Lambda function to run the test scripts. If errors are found, exit the Lambda function with an error to initiate rollback.

D. Add a hooks section to the CodeDeploy AppSpec file. Use the AfterAllowTraffic lifecycle event to invoke the test scripts. If errors are found, use the aws deploy stop-deployment CLI command to stop the deployment.

Answer: C

Explanation:

https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure-hooks.html

Question: 35

A company uses AWS Storage Gateway in file gateway mode in front of an Amazon S3 bucket that is used by multiple resources. In the morning when business begins, users do not see the objects processed by a third party the previous evening. When a DevOps engineer looks directly at the S3 bucket, the data is there, but it is missing in Storage Gateway.
Which solution ensures that all the updated third-party files are available in the morning?

A. Configure a nightly Amazon EventBridge event to invoke an AWS Lambda function to run the RefreshCache command for Storage Gateway.

B. Instruct the third party to put data into the S3 bucket using AWS Transfer for SFTP.

C. Modify Storage Gateway to run in volume gateway mode.

D. Use S3 Same-Region Replication to replicate any changes made directly in the S3 bucket to Storage Gateway.

www.atmicnetworks.com

_____ Answer: A

Explanation:

https://docs.aws.amazon.com/storagegateway/latest/APIReference/API_RefreshCache.html " It only updates the cached inventory to reflect changes in the inventory of the objects in the S3 bucket. This operation is only supported in the S3 File Gateway types."

Question: 36

A DevOps engineer needs to back up sensitive Amazon S3 objects that are stored within an S3 bucket with a private bucket policy using S3 cross-Region replication functionality. The objects need to be copied to a target bucket in a different AWS Region and account.
Which combination of actions should be performed to enable this replication? (Choose three.)

A. Create a replication IAM role in the source account
B. Create a replication I AM role in the target account.
C. Add statements to the source bucket policy allowing the replication IAM role to replicate objects.
D. Add statements to the target bucket policy allowing the replication IAM role to replicate objects.
E. Create a replication rule in the source bucket to enable the replication.
F. Create a replication rule in the target bucket to enable the replication.

Answer: A, D, E

Explanation:

S3 cross-Region replication (CRR) automatically replicates data between buckets across different AWS Regions. To enable CRR, you need to add a replication configuration to your source bucket that specifies the destination bucket, the IAM role, and the encryption type (optional). You also need to grant permissions to the IAM role to perform replication actions on both the source and destination buckets. Additionally, you can choose the destination storage class and enable additional replication options such as S3 Replication Time Control (S3 RTC) or S3 Batch Replication. https://medium.com/cloud-techies/s3-same-region-replication-srr-and-cross-region-replication-crr-34d446806bab https://aws.amazon.com/getting-started/hands-on/replicate-data-using-amazon-s3- replication/
https://docs.aws.amazon.com/AmazonS3/latest/userguide/replication.html

Question: 37

A company has multiple member accounts that are part of an organization in AWS Organizations. The security team needs to review every Amazon EC2 security group and their inbound and outbound rules. The security team wants to programmatically retrieve this information from the member accounts using an AWS Lambda function in the management account of the organization.
Which combination of access changes will meet these requirements? (Choose three.)

A. Create a trust relationship that allows users in the member accounts to assume the management account IAM role.
B. Create a trust relationship that allows users in the management account to assume the IAM roles of the member accounts.
C. Create an IAM role in each member account that has access to the AmazonEC2ReadOnlyAccess managed policy.
D. Create an I AM role in each member account to allow the sts:AssumeRole action against the management account IAM role's ARN.
E. Create an I AM role in the management account that allows the sts:AssumeRole action against the member account IAM role's ARN.
F. Create an IAM role in the management account that has access to the AmazonEC2ReadOnlyAccess managed policy.

Answer: B, C, E

Explanation:

https://aws.amazon.com/premiumsupport/knowledge-center/lambda-function-assume-iam-role/
https://kreuzwerker.de/post/aws-multi-account-setups-reloaded

Question: 38

A space exploration company receives telemetry data from multiple satellites. Small packets of data are received through Amazon API Gateway and are placed directly into an Amazon Simple Queue Service (Amazon SQS) standard queue. A custom application is subscribed to the queue and transforms the data into a standard format.

Because of inconsistencies in the data that the satellites produce, the application is occasionally unable to transform the dat a. In these cases, the messages remain in the SQS queue. A DevOps engineer must develop a solution that retains the failed messages and makes them available to scientists for review and future processing.

Which solution will meet these requirements?

A. Configure AWS Lambda to poll the SQS queue and invoke a Lambda function to check whether the queue messages are valid. If validation fails, send a copy of the data that is not valid to an Amazon S3 bucket so that the scientists can review and correct the data. When the data is corrected, amend the message in the SQS queue by using a replay Lambda function with the corrected data.

B. Convert the SQS standard queue to an SQS FIFO queue. Configure AWS Lambda to poll the SQS queue every 10 minutes by using an Amazon EventBridge schedule. Invoke the Lambda function to identify any messages with a SentTimestamp value that is older than 5 minutes, push the data to the same location as the application's output location, and remove the messages from the queue.

C. Create an SQS dead-letter queue. Modify the existing queue by including a redrive policy that sets the Maximum Receives setting to 1 and sets the dead-letter queue ARN to the ARN of the newly created queue. Instruct the scientists to use the dead-letter queue to review the data that is not valid. Reprocess this data at a later time.

D. Configure API Gateway to send messages to different SQS virtual queues that are named for each of the satellites. Update the application to use a new virtual queue for any data that it cannot transform, and send the message to the new virtual queue. Instruct the scientists to use the virtual queue to review the data that is not valid. Reprocess this data at a later time.

_____ Answer: C

Explanation:

Create an SQS dead-letter queue. Modify the existing queue by including a redrive policy that sets the Maximum Receives setting to 1 and sets the dead-letter queue ARN to the ARN of the newly created queue. Instruct the scientists to use the dead-letter queue to review the data that is not valid. Reprocess this data at a later time.

Question: 39

A company wants to use AWS CloudFormation for infrastructure deployment. The company has strict tagging and resource requirements and wants to limit the deployment to two Regions. Developers will need to deploy multiple versions of the same application.

Which solution ensures resources are deployed in accordance with company policy?

A. Create AWS Trusted Advisor checks to find and remediate unapproved CloudFormation StackSets.

B. Create a Cloud Formation drift detection operation to find and remediate unapproved CloudFormation StackSets.

C. Create CloudFormation StackSets with approved CloudFormation templates.

D. Create AWS Service Catalog products with approved CloudFormation templates.

_____ Answer: D

Explanation:

service catalog uses stacksets and can enforce tag and restrict resources AWS Customer case with tag enforcement https://aws.amazon.com/ko/blogs/apn/enforce-centralized-tag-compliance-using-aws- service-catalog-amazon-dynamodb-aws-lambda-and-amazon-cloudwatch-events/ And Youtube video showing how to restrict resources per user with portfolio https://www.youtube.com/watch?v=LzvhTcqqyog

Question: 40

A company requires that its internally facing web application be highly available. The architecture is made up of one Amazon EC2 web server instance and one NAT instance that provides outbound internet access for updates and accessing public data. Which combination of architecture adjustments should the company implement to achieve high availability? (Choose two.)

A. Add the NAT instance to an EC2 Auto Scaling group that spans multiple Availability Zones. Update the route tables.

B. Create additional EC2 instances spanning multiple Availability Zones. Add an Application Load Balancer to split the load between them.

C. Configure an Application Load Balancer in front of the EC2 instance. Configure Amazon CloudWatch alarms to recover the EC2 instance upon host failure.

D. Replace the NAT instance with a NAT gateway in each Availability Zone. Update the route tables.

E. Replace the NAT instance with a NAT gateway that spans multiple Availability Zones. Update the route tables.

Answer: B, D

Explanation:

https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway.html

Question: 41

A DevOps engineer is building a multistage pipeline with AWS CodePipeline to build, verify, stage, test, and deploy an application. A manual approval stage is required between the test stage and the deploy stage. The development team uses a custom chat tool with webhook support that requires near-real-time notifications.

How should the DevOps engineer configure status updates for pipeline activity and approval requests to post to the chat tool?

A. Create an Amazon CloudWatch Logs subscription that filters on CodePipeline Pipeline Execution State Change. Publish subscription events to an Amazon Simple Notification Service (Amazon SNS) topic. Subscribe the chat webhook URL to the SNS topic, and complete the subscription validation.

B. Create an AWS Lambda function that is invoked by AWS CloudTrail events. When a CodePipeline Pipeline Execution State Change event is detected, send the event details to the chat webhook URL.

C. Create an Amazon EventBridge rule that filters on CodePipeline Pipeline Execution State Change. Publish the events to an Amazon Simple Notification Service (Amazon SNS) topic. Create an AWS Lambda function that sends event details to the chat webhook URL. Subscribe the function to the SNS topic.

D. Modify the pipeline code to send the event details to the chat webhook URL at the end of each stage. Parameterize the URL so that each pipeline can send to a different URL based on the pipeline environment.

Answer: C

Explanation:

https://aws.amazon.com/premiumsupport/knowledge-center/sns-lambda-webhooks-chime-slack- teams/

Question: 42

A company's application development team uses Linux-based Amazon EC2 instances as bastion hosts. Inbound SSH access to the bastion hosts is restricted to specific IP addresses, as defined in the associated security groups. The company's security team wants to receive a notification if the security group rules are modified to allow SSH access from any IP address.
What should a DevOps engineer do to meet this requirement?

A. Create an Amazon EventBridge rule with a source of aws.cloudtrail and the event name AuthorizeSecurityGroupIngress. Define an Amazon Simple Notification Service (Amazon SNS) topic as the target.
B. Enable Amazon GuardDuty and check the findings for security groups in AWS Security Hub. Configure an Amazon EventBridge rule with a custom pattern that matches GuardDuty events with an output of NON_COMPLIANT. Define an Amazon Simple Notification Service (Amazon SNS) topic as the target.
C. Create an AWS Config rule by using the restricted-ssh managed rule to check whether security groups disallow unrestricted incoming SSH traffic. Configure automatic remediation to publish a message to an Amazon Simple Notification Service (Amazon SNS) topic.
D. Enable Amazon Inspector. Include the Common Vulnerabilities and Exposures-1.1 rules package to check the security groups that are associated with the bastion hosts. Configure Amazon Inspector to publish a message to an Amazon Simple Notification Service (Amazon SNS) topic.

Answer: A

Explanation:

https://aws.amazon.com/premiumsupport/knowledge-center/monitor-security-group-changes-ec2/

Question: 43

A DevOps team manages an API running on-premises that serves as a backend for an Amazon API Gateway endpoint. Customers have been complaining about high response latencies, which the development team has verified using the API Gateway latency metrics in Amazon CloudWatch. To identify the cause, the team needs to collect relevant data without introducing additional latency. Which actions should be taken to accomplish this? (Choose two.)

A. Install the CloudWatch agent server side and configure the agent to upload relevant logs to CloudWatch.
B. Enable AWS X-Ray tracing in API Gateway, modify the application to capture request segments, and upload those segments to X-Ray during each request.
C. Enable AWS X-Ray tracing in API Gateway, modify the application to capture request segments, and use the X-Ray daemon to upload segments to X-Ray.
D. Modify the on-premises application to send log information back to API Gateway with each request.
E. Modify the on-premises application to calculate and upload statistical data relevant to the API service requests to CloudWatch metrics.

Answer: A, C

Explanation:

https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/install-CloudWatch-Agent-on-premise.html https://docs.aws.amazon.com/xray/latest/devguide/xray-api-sendingdata.html

www.atmicnetworks.com

Question: 44

A company has an application that is using a MySQL-compatible Amazon Aurora Multi-AZ DB cluster as the database. A cross-Region read replica has been created for disaster recovery purposes. A DevOps engineer wants to automate the promotion of the replica so it becomes the primary database instance in the event of a failure.
Which solution will accomplish this?

A. Configure a latency-based Amazon Route 53 CNAME with health checks so it points to both the primary and replica endpoints. Subscribe an Amazon SNS topic to Amazon RDS failure notifications from AWS CloudTrail and use that topic to invoke an AWS Lambda function that will promote the replica instance as the primary.
B. Create an Aurora custom endpoint to point to the primary database instance. Configure the application to use this endpoint. Configure AWS CloudTrail to run an AWS Lambda function to promote the replica instance and modify the custom endpoint to point to the newly promoted instance.
C. Create an AWS Lambda function to modify the application's AWS CloudFormation template to promote the replica, apply the template to update the stack, and point the application to the newly promoted instance. Create an Amazon CloudWatch alarm to invoke this Lambda function after the failure event occurs.
D. Store the Aurora endpoint in AWS Systems Manager Parameter Store. Create an Amazon EventBridge event that detects the database failure and runs an AWS Lambda function to promote the replica instance and update the endpoint URL stored in AWS Systems Manager Parameter Store. Code the application to reload the endpoint from Parameter Store if a database connection fails.

_____ Answer: D

Explanation:

EventBridge is needed to detect the database failure. Lambda is needed to promote the replica as it's in another Region (manual promotion, otherwise). Storing and updating the endpoint in Parameter store is important in updating the application. Look at High Availability section of Aurora FAQ: https://aws.amazon.com/rds/aurora/faqs/

Question: 45

A company hosts its staging website using an Amazon EC2 instance backed with Amazon EBS storage. The company wants to recover quickly with minimal data losses in the event of network connectivity issues or power failures on the EC2 instance.
Which solution will meet these requirements?

A. Add the instance to an EC2 Auto Scaling group with the minimum, maximum, and desired capacity set to 1.
B. Add the instance to an EC2 Auto Scaling group with a lifecycle hook to detach the EBS volume when the EC2 instance shuts down or terminates.
C. Create an Amazon CloudWatch alarm for the StatusCheckFailed System metric and select the EC2 action to recover the instance.
D. Create an Amazon CloudWatch alarm for the StatusCheckFailed Instance metric and select the EC2 action to reboot the instance.

_____ Answer: C

Explanation:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-recover.html

www.atmicnetworks.com

Question: 46 A company wants to use AWS development tools to replace its current bash deployment scripts. The company currently deploys a LAMP application to a group of Amazon EC2 instances behind an Application Load Balancer (ALB). During the deployments, the company unit tests the committed application, stops and starts services, unregisters and re-registers instances with the load balancer, and updates file permissions. The company wants to maintain the same deployment functionality through the shift to using AWS services.

Which solution will meet these requirements?

A. Use AWS CodeBuild to test the application. Use bash scripts invoked by AWS CodeDeploy's appspec.yml file to restart services, and deregister and register instances with the ALB. Use the appspec.yml file to update file permissions without a custom script.

B. Use AWS CodePipeline to move the application from the AWS CodeCommit repository to AWS CodeDeploy. Use CodeDeploy's deployment group to test the application, unregister and re-register instances with the ALB. and restart services. Use the appspec.yml file to update file permissions without a custom script.

C. Use AWS CodePipeline to move the application source code from the AWS CodeCommit repository to AWS CodeDeploy. Use CodeDeploy to test the application. Use CodeDeploy's appspec.yml file to restart services and update permissions without a custom script. Use AWS CodeBuild to unregister and re-register instances with the ALB.

D. Use AWS CodePipeline to trigger AWS CodeBuild to test the application. Use bash scripts invoked by AWS CodeDeploy's appspec.yml file to restart services. Unregister and re-register the instances in the AWS CodeDeploy deployment group with the ALB. Update the appspec.yml file to update file permissions without a custom script.

Answer: D

Explanation:

https://aws.amazon.com/blogs/devops/how-to-test-and-debug-aws-codedeploy-locally-before-you-ship-your-code/#:~:text=You%20can%20test%20application%20code,local%20server%20or%20EC2%20instance.

Question: 47

A company runs an application with an Amazon EC2 and on-premises configuration. A DevOps engineer needs to standardize patching across both environments. Company policy dictates that patching only happens during non-business hours.

Which combination of actions will meet these requirements? (Choose three.)

A. Add the physical machines into AWS Systems Manager using Systems Manager Hybrid Activations.

B. Attach an IAM role to the EC2 instances, allowing them to be managed by AWS Systems Manager.

C. Create IAM access keys for the on-premises machines to interact with AWS Systems Manager.

D. Run an AWS Systems Manager Automation document to patch the systems every hour.

E. Use Amazon EventBridge scheduled events to schedule a patch window.

F. Use AWS Systems Manager Maintenance Windows to schedule a patch window.

Answer: A, B, F

Explanation:

https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-managed-instance-activation.html

Question: 48

A company has chosen AWS to host a new application. The company needs to implement a multiaccount strategy. A DevOps engineer

www.atmicnetworks.com

creates a new AWS account and an organization in AWS Organizations. The DevOps engineer also creates the OU structure for the organization and sets up a landing zone by using AWS Control Tower.

The DevOps engineer must implement a solution that automatically deploys resources for new accounts that users create through AWS Control Tower Account Factory. When a user creates a new account, the solution must apply AWS CloudFormation templates and SCPs that are customized for the OU or the account to automatically deploy all the resources that are attached to the account. All the OUs are enrolled in AWS Control Tower.

Which solution will meet these requirements in the MOST automated way?

A. Use AWS Service Catalog with AWS Control Tower. Create portfolios and products in AWS Service Catalog. Grant granular permissions to provision these resources. Deploy SCPs by using the AWS CLI and JSON documents.

B. Deploy CloudFormation stack sets by using the required templates. Enable automatic deployment. Deploy stack instances to the required accounts. Deploy a CloudFormation stack set to the organization's management account to deploy SCPs.

C. Create an Amazon EventBridge rule to detect the CreateManagedAccount event. Configure AWS Service Catalog as the target to deploy resources to any new accounts. Deploy SCPs by using the AWS CLI and JSON documents.

D. Deploy the Customizations for AWS Control Tower (CfCT) solution. Use an AWS CodeCommit repository as the source. In the repository, create a custom package that includes the CloudFormation templates and the SCP JSON documents.

_____ Answer: D

Explanation:

The CfCT solution is designed for the exact purpose stated in the question. It extends the capabilities of AWS Control Tower by providing you with a way to automate resource provisioning and apply custom configurations across all AWS accounts created in the Control Tower environment. This enables the company to implement additional account customizations when new accounts are provisioned via the Control Tower Account Factory. The CloudFormation templates and SCPs can be added to a CodeCommit repository and will be automatically deployed to new accounts when they are created. This provides a highly automated solution that does not require manual intervention to deploy resources and SCPs to new accounts.

Question: 49

An online retail company based in the United States plans to expand its operations to Europe and Asia in the next six months. Its product currently runs on Amazon EC2 instances behind an Application Load Balancer. The instances run in an Amazon EC2 Auto Scaling group across multiple Availability Zones. All data is stored in an Amazon Aurora database instance.

When the product is deployed in multiple regions, the company wants a single product catalog across all regions, but for compliance purposes, its customer information and purchases must be kept in each region.

How should the company meet these requirements with the LEAST amount of application changes?

A. Use Amazon Redshift for the product catalog and Amazon DynamoDB tables for the customer information and purchases.

B. Use Amazon DynamoDB global tables for the product catalog and regional tables for the customer information and purchases.

C. Use Aurora with read replicas for the product catalog and additional local Aurora instances in each region for the customer information and purchases.

D. Use Aurora for the product catalog and Amazon DynamoDB global tables for the customer information and purchases.

_____ Answer: C

Explanation:

www.atmicnetworks.com

Question: 50

A company is implementing a well-architected design for its globally accessible API stack. The design needs to ensure both high reliability and fast response times for users located in North America and Europe.
The API stack contains the following three tiers:
Amazon API Gateway
AWS Lambda
Amazon DynamoDB
Which solution will meet the requirements?

A. Configure Amazon Route 53 to point to API Gateway APIs in North America and Europe using health checks. Configure the APIs to forward requests to a Lambda function in that Region. Configure the Lambda functions to retrieve and update the data in a DynamoDB table in the same Region as the Lambda function.

B. Configure Amazon Route 53 to point to API Gateway APIs in North America and Europe using latency-based routing and health checks. Configure the APIs to forward requests to a Lambda function in that Region. Configure the Lambda functions to retrieve and update the data in a DynamoDB global table.

C. Configure Amazon Route 53 to point to API Gateway in North America, create a disaster recovery API in Europe, and configure both APIs to forward requests to the Lambda functions in that Region. Retrieve the data from a DynamoDB global table. Deploy a Lambda function to check the North America API health every 5 minutes. In the event of a failure, update Route 53 to point to the disaster recovery API.

D. Configure Amazon Route 53 to point to API Gateway API in North America using latency-based routing. Configure the API to forward requests to the Lambda function in the Region nearest to the user. Configure the Lambda function to retrieve and update the data in a DynamoDB table.

Answer: B

Explanation:

Question: 51

A rapidly growing company wants to scale for developer demand for AWS development environments. Development environments are created manually in the AWS Management Console. The networking team uses AWS CloudFormation to manage the networking infrastructure, exporting stack output values for the Amazon VPC and all subnets. The development environments have common standards, such as Application Load Balancers, Amazon EC2 Auto Scaling groups, security groups, and Amazon DynamoDB tables.
To keep up with demand, the DevOps engineer wants to automate the creation of development environments. Because the infrastructure required to support the application is expected to grow, there must be a way to easily update the deployed infrastructure.
CloudFormation will be used to create a template for the development environments.
Which approach will meet these requirements and quickly provide consistent AWS environments for developers?

A. Use Fn::ImportValue intrinsic functions in the Resources section of the template to retrieve Virtual Private Cloud (VPC) and subnet values. Use CloudFormation StackSets for the development environments, using the Count input parameter to indicate the number of environments needed. Use the UpdateStackSet command to update existing development environments.

B. Use nested stacks to define common infrastructure components. To access the exported values, use TemplateURL to reference the networking team's template. To retrieve Virtual Private Cloud (VPC) and subnet values, use Fn::ImportValue intrinsic functions in the Parameters section of the root template. Use the CreateChangeSet and ExecuteChangeSet commands to update existing development environments.

C. Use nested stacks to define common infrastructure components. Use Fn::ImportValue intrinsic functions with the resources of the nested stack to retrieve Virtual Private Cloud (VPC) and subnet values. Use the CreateChangeSet and ExecuteChangeSet commands to update existing development environments.

D. Use Fn::ImportValue intrinsic functions in the Parameters section of the root template to retrieve Virtual Private Cloud (VPC) and subnet values. Define the development resources in the order they need to be created in the CloudFormation nested stacks. Use the CreateChangeSet. and ExecuteChangeSet commands to update existing development environments.

Answer: C

Explanation:

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/intrinsic-function-reference- importvalue.html
https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/intrinsic-function-reference- importvalue.html CF of network exports the VPC, subnet or needed information CF of application imports the above information to its stack and UpdateChangeSet/ ExecuteChangeSet

Question: 52

A company uses AWS Organizations to manage multiple accounts. Information security policies require that all unencrypted Amazon EBS volumes be marked as non-compliant. A DevOps engineer needs to automatically deploy the solution and ensure that this compliance check is always present. Which solution will accomplish this?

A. Create an AWS CloudFormation template that defines an AWS Inspector rule to check whether EBS encryption is enabled. Save the template to an Amazon S3 bucket that has been shared with all accounts within the company. Update the account creation script pointing to the CloudFormation template in Amazon S3.

B. Create an AWS Config organizational rule to check whether EBS encryption is enabled and deploy the rule using the AWS CLI. Create and apply an SCP to prohibit stopping and deleting AWS Config across the organization.

C. Create an SCP in Organizations. Set the policy to prevent the launch of Amazon EC2 instances without encryption on the EBS volumes using a conditional expression. Apply the SCP to all AWS accounts. Use Amazon Athena to analyze the AWS CloudTrail output, looking for events that deny an ec2:RunInstances action.

D. Deploy an IAM role to all accounts from a single trusted account. Build a pipeline with AWS CodePipeline with a stage in AWS Lambda to assume the IAM role, and list all EBS volumes in the account. Publish a report to Amazon S3.

Answer: B

Explanation:

https://docs.aws.amazon.com/config/latest/developerguide/ec2-ebs-encryption-by-default.html

Question: 53

A company is performing vulnerability scanning for all Amazon EC2 instances across many accounts. The accounts are in an organization in AWS Organizations. Each account's VPCs are attached to a shared transit gateway. The VPCs send traffic to the internet through a central egress VPC. The company has enabled Amazon Inspector in a delegated administrator account and has enabled scanning for all member accounts.

A DevOps engineer discovers that some EC2 instances are listed in the "not scanning" tab in Amazon Inspector.

Which combination of actions should the DevOps engineer take to resolve this issue? (Choose three.)

A. Verify that AWS Systems Manager Agent is installed and is running on the EC2 instances that Amazon Inspector is not scanning.

B. Associate the target EC2 instances with security groups that allow outbound communication on port 443 to the AWS Systems Manager service endpoint.

C. Grant inspector:StartAssessmentRun permissions to the IAM role that the DevOps engineer is using.

D. Configure EC2 Instance Connect for the EC2 instances that Amazon Inspector is not scanning.

E. Associate the target EC2 instances with instance profiles that grant permissions to communicate with AWS Systems Manager.

F. Create a managed-instance activation. Use the Activation Code and the Activation ID to register the EC2 instances.

Answer: A, B, E

Explanation:

https://docs.aws.amazon.com/inspector/latest/user/scanning-ec2.html

Question: 54

A development team uses AWS CodeCommit for version control for applications. The development team uses AWS CodePipeline, AWS CodeBuild. and AWS CodeDeploy for CI/CD infrastructure. In CodeCommit, the development team recently merged pull requests that did not pass long-running tests in the code base. The development team needed to perform rollbacks to branches in the codebase, resulting in lost time and wasted effort.

A DevOps engineer must automate testing of pull requests in CodeCommit to ensure that reviewers more easily see the results of automated tests as part of the pull request review.

What should the DevOps engineer do to meet this requirement?

A. Create an Amazon EventBridge rule that reacts to the pullRequestStatusChanged event. Create an AWS Lambda function that invokes a CodePipeline pipeline with a CodeBuild action that runs the tests for the application. Program the Lambda function to post the CodeBuild badge as a comment on the pull request so that developers will see the badge in their code review.

B. Create an Amazon EventBridge rule that reacts to the pullRequestCreated event. Create an AWS Lambda function that invokes a CodePipeline pipeline with a CodeBuild action that runs the tests for the application. Program the Lambda function to post the CodeBuild test results as a comment on the pull request when the test results are complete.

C. Create an Amazon EventBridge rule that reacts to pullRequestCreated and pullRequestSourceBranchUpdated events. Create an AWS Lambda function that invokes a CodePipeline pipeline with a CodeBuild action that runs the tests for the application. Program the Lambda function to post the CodeBuild badge as a comment on the pull request so that developers will see the badge in their code review.

D. Create an Amazon EventBridge rule that reacts to the pullRequestStatusChanged event. Create an AWS Lambda function that invokes a CodePipeline pipeline with a CodeBuild action that runs the tests for the application. Program the Lambda function to post the CodeBuild test results as a comment on the pull request when the test results are complete.

_____ Answer: C

Explanation:

https://aws.amazon.com/es/blogs/devops/complete-ci-cd-with-aws-codecommit-aws-codebuild- aws-codedeploy-and-aws-codepipeline/

Question: 55

A company has deployed an application in a production VPC in a single AWS account. The application is popular and is experiencing heavy usage. The company's security team wants to add additional security, such as AWS WAF, to the application deployment. However, the application's product manager is concerned about cost and does not want to approve the change unless the security team can prove that additional security is necessary.

The security team believes that some of the application's demand might come from users that have IP addresses that are on a deny list. The security team provides the deny list to a DevOps engineer. If any of the IP addresses on the deny list access the application, the security team wants to receive automated notification in near real time so that the security team can document that the application needs additional security. The DevOps engineer creates a VPC flow log for the production VPC. Which set of additional steps should the DevOps engineer take to meet these requirements MOST cost-effectively?

A. Create a log group in Amazon CloudWatch Logs. Configure the VPC flow log to capture accepted traffic and to send the data to the log group. Create an Amazon CloudWatch metric filter for IP addresses on the deny list. Create a CloudWatch alarm with the metric filter as input. Set the period to 5 minutes and the datapoints to alarm to 1. Use an Amazon Simple Notification Service (Amazon SNS) topic to send alarm notices to the security team.

B. Create an Amazon S3 bucket for log files. Configure the VPC flow log to capture all traffic and to send the data to the S3 bucket. Configure Amazon Athena to return all log files in the S3 bucket for IP addresses on the deny list. Configure Amazon QuickSight to accept data from Athena and to publish the data as a dashboard that the security team can access. Create a threshold alert of 1 for successful access. Configure the alert to automatically notify the security team as frequently as possible when the alert threshold is met.

C. Create an Amazon S3 bucket for log files. Configure the VPC flow log to capture accepted traffic and to send the data to the S3 bucket. Configure an Amazon OpenSearch Service cluster and domain for the log files. Create an AWS Lambda function to retrieve the logs from the S3 bucket, format the logs, and load the logs into the OpenSearch Service cluster. Schedule the Lambda function to run every 5 minutes. Configure an alert and condition in OpenSearch Service to send alerts to the security team through an Amazon Simple Notification Service (Amazon SNS) topic when access from the IP addresses on the deny list is detected.

D. Create a log group in Amazon CloudWatch Logs. Create an Amazon S3 bucket to hold query results. Configure the VPC flow log to capture all traffic and to send the data to the log group. Deploy an Amazon Athena CloudWatch connector in AWS Lambda. Connect the connector to the log group. Configure Athena to periodically query for all accepted traffic from the IP addresses on the deny list and to store the results in the S3 bucket. Configure an S3 event notification to automatically notify the security team through an Amazon Simple Notification Service (Amazon SNS) topic when new objects are added to the S3 bucket.

_____ Answer: A

Explanation:

Question: 56

A DevOps engineer has automated a web service deployment by using AWS CodePipeline with the following steps:

1) An AWS CodeBuild project compiles the deployment artifact and runs unit tests.
2) An AWS CodeDeploy deployment group deploys the web service to Amazon EC2 instances in the staging environment.
3) A CodeDeploy deployment group deploys the web service to EC2 instances in the production environment.

The quality assurance (QA) team requests permission to inspect the build artifact before the deployment to the production environment occurs. The QA team wants to run an internal penetration testing tool to conduct manual tests. The tool will be invoked by a REST API call. Which combination of actions should the DevOps engineer take to fulfill this request? (Choose two.)

A. Insert a manual approval action between the test actions and deployment actions of the pipeline.
B. Modify the buildspec.yml file for the compilation stage to require manual approval before completion.
C. Update the CodeDeploy deployment groups so that they require manual approval to proceed.
D. Update the pipeline to directly call the REST API for the penetration testing tool.
E. Update the pipeline to invoke an AWS Lambda function that calls the REST API for the penetration testing tool.

Answer: A, E

Explanation:

Question: 57

A company is hosting a web application in an AWS Region. For disaster recovery purposes, a second region is being used as a standby. Disaster recovery requirements state that session data must be replicated between regions in near-real time and 1% of requests should route to the secondary region to continuously verify system functionality. Additionally, if there is a disruption in service in the main region, traffic should be automatically routed to the secondary region, and the secondary region must be able to scale up to handle all traffic.
How should a DevOps engineer meet these requirements?

A. In both regions, deploy the application on AWS Elastic Beanstalk and use Amazon DynamoDB global tables for session data. Use an Amazon Route 53 weighted routing policy with health checks to distribute the traffic across the regions.
B. In both regions, launch the application in Auto Scaling groups and use DynamoDB for session data. Use a Route 53 failover routing policy with health checks to distribute the traffic across the regions.
C. In both regions, deploy the application in AWS Lambda, exposed by Amazon API Gateway, and use Amazon RDS for PostgreSQL with cross-region replication for session data. Deploy the web application with client-side logic to call the API Gateway directly.
D. In both regions, launch the application in Auto Scaling groups and use DynamoDB global tables for session data. Enable an Amazon CloudFront weighted distribution across regions. Point the Amazon Route 53 DNS record at the CloudFront distribution.

Answer: D

Explanation:

Question: 58

A company runs an application on Amazon EC2 instances. The company uses a series of AWS CloudFormation stacks to define the application resources. A developer performs updates by building and testing the application on a laptop and then uploading the build output and CloudFormation stack templates to Amazon S3. The developer's peers review the changes before the developer performs the CloudFormation stack update and installs a new version of the application onto the EC2 instances.
The deployment process is prone to errors and is time-consuming when the developer updates each EC2 instance with the new application. The company wants to automate as much of the application deployment process as possible while retaining a final manual approval step before the modification of the application or resources.
The company already has moved the source code for the application and the CloudFormation templates to AWS CodeCommit. The company also has created an AWS CodeBuild project to build and test the application.
Which combination of steps will meet the company's requirements? (Choose two.)

A. Create an application group and a deployment group in AWS CodeDeploy. Install the CodeDeploy agent on the EC2 instances.
B. Create an application revision and a deployment group in AWS CodeDeploy. Create an environment in CodeDeploy. Register the EC2 instances to the CodeDeploy environment.
C. Use AWS CodePipeline to invoke the CodeBuild job, run the CloudFormation update, and pause for a manual approval step. After approval, start the AWS CodeDeploy deployment.
D. Use AWS CodePipeline to invoke the CodeBuild job, create CloudFormation change sets for each of the application stacks, and pause for a manual approval step. After approval, run the CloudFormation change sets and start the AWS CodeDeploy deployment.
E. Use AWS CodePipeline to invoke the CodeBuild job, create CloudFormation change sets for each of the application stacks, and pause for a manual approval step. After approval, start the AWS CodeDeploy deployment.

Answer: A, D

Explanation:

A- https://docs.aws.amazon.com/codedeploy/latest/userguide/codedeploy-agent.html D - This option correctly utilizes AWS CodePipeline to invoke the CodeBuild job and create CloudFormation change sets. It adds a manual approval step before executing the change sets and starting the AWS CodeDeploy deployment. This ensures that the deployment process is automated while retaining the final manual approval step.

Question: 59

A DevOps engineer manages a web application that runs on Amazon EC2 instances behind an Application Load Balancer (ALB). The instances run in an EC2 Auto Scaling group across multiple Availability Zones. The engineer needs to implement a deployment strategy that: Launches a second fleet of instances with the same capacity as the original fleet.
Maintains the original fleet unchanged while the second fleet is launched. Transitions traffic to the second fleet when the second fleet is fully deployed.
Terminates the original fleet automatically 1 hour after transition.
Which solution will satisfy these requirements?

A. Use an AWS CloudFormation template with a retention policy for the ALB set to 1 hour. Update the Amazon Route 53 record to reflect the new ALB.
B. Use two AWS Elastic Beanstalk environments to perform a blue/green deployment from the original environment to the new one. Create an application version lifecycle policy to terminate the original environment in 1 hour.
C. Use AWS CodeDeploy with a deployment group configured with a blue/green deployment configuration Select the option Terminate the original instances in the deployment group with a waiting period of 1 hour.
D. Use AWS Elastic Beanstalk with the configuration set to Immutable. Create an .ebextension using the Resources key that sets the deletion policy of the ALB to 1 hour, and deploy the application.

Answer: C

Explanation:

https://docs.aws.amazon.com/codedeploy/latest/APIReference/API BlueInstanceTerminationOption .html
The original revision termination settings are configured to wait 1 hour after traffic has been rerouted before terminating the blue task set.
https://docs.aws.amazon.com/AmazonECS/latest/developerguide/deployment-type-bluegreen.html

Question: 60

A video-sharing company stores its videos in Amazon S3. The company has observed a sudden increase in video access requests, but the company does not know which videos are most popular. The company needs to identify the general access pattern for the video files. This pattern includes the number of users who access a certain file on a given day, as well as the numb A DevOps engineer manages a large commercial website that runs on Amazon EC2 The website uses Amazon Kinesis Data Streams to collect and process web togs The DevOps engineer manages the Kinesis consumer application, which also runs on Amazon EC2
Sudden increases of data cause the Kinesis consumer application to (all behind and the Kinesis data streams drop records before the records can be processed The DevOps engineer must implement a solution to improve stream handling
Which solution meets these requirements with the MOST operational efficiency''
er of pull requests for certain files.
How can the company meet these requirements with the LEAST amount of effort?

A. Activate S3 server access logging. Import the access logs into an Amazon Aurora database. Use an Aurora SQL query to analyze

the access patterns.

B. Activate S3 server access logging. Use Amazon Athena to create an external table with the log files. Use Athena to create a SQL query to analyze the access patterns.

C. Invoke an AWS Lambda function for every S3 object access event. Configure the Lambda function to write the file access information, such as user. S3 bucket, and file key, to an Amazon Aurora database. Use an Aurora SQL query to analyze the access patterns.

D. Record an Amazon CloudWatch Logs log message for every S3 object access event. Configure a CloudWatch Logs log stream to write the file access information, such as user, S3 bucket, and file key, to an Amazon Kinesis Data Analytics for SQL application. Perform a sliding window analysis.

Answer: B

Explanation:

Activating S3 server access logging and using Amazon Athena to create an external table with the log files is the easiest and most cost-effective way to analyze access patterns. This option requires minimal setup and allows for quick analysis of the access patterns with SQL queries. Additionally,
Amazon Athena scales automatically to match the query load, so there is no need for additional infrastructure provisioning or management.

Question: 61

A development team wants to use AWS CloudFormation stacks to deploy an application. However, the developer IAM role does not have the required permissions to provision the resources that are specified in the AWS CloudFormation template. A DevOps engineer needs to implement a solution that allows the developers to deploy the stacks. The solution must follow the principle of least privilege.
Which solution will meet these requirements?

A. Create an IAM policy that allows the developers to provision the required resources. Attach the policy to the developer IAM role.

B. Create an IAM policy that allows full access to AWS CloudFormation. Attach the policy to the developer IAM role.

C. Create an AWS CloudFormation service role that has the required permissions. Grant the developer IAM role a cloudformation:* action. Use the new service role during stack deployments.

D. Create an AWS CloudFormation service role that has the required permissions. Grant the developer IAM role the iam:PassRole permission. Use the new service role during stack deployments.

Answer: D

Explanation:

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-iam-servicerole.html

Question: 62

A production account has a requirement that any Amazon EC2 instance that has been logged in to manually must be terminated within 24 hours. All applications in the production account are using Auto Scaling groups with the Amazon CloudWatch Logs agent configured.
How can this process be automated?

A. Create a CloudWatch Logs subscription to an AWS Step Functions application. Configure an AWS Lambda function to add a tag to the EC2 instance that produced the login event and mark the instance to be decommissioned. Create an Amazon EventBridge rule to invoke a second Lambda function once a day that will terminate all instances with this tag.

B. Create an Amazon CloudWatch alarm that will be invoked by the login event. Send the notification to an Amazon Simple Notification Service (Amazon SNS) topic that the operations team is subscribed to, and have them terminate the EC2 instance within 24 hours.
C. Create an Amazon CloudWatch alarm that will be invoked by the login event. Configure the alarm to send to an Amazon Simple Queue Service (Amazon SQS) queue. Use a group of worker instances to process messages from the queue, which then schedules an Amazon EventBridge rule to be invoked.
D. Create a CloudWatch Logs subscription to an AWS Lambda function. Configure the function to add a tag to the EC2 instance that produced the login event and mark the instance to be decommissioned. Create an Amazon EventBridge rule to invoke a daily Lambda function that terminates all instances with this tag.

_____ Answer: D

Explanation:

"You can use subscriptions to get access to a real-time feed of log events from CloudWatch Logs and have it delivered to other services such as an Amazon Kinesis stream, an Amazon Kinesis Data Firehose stream, or AWS Lambda for custom processing, analysis, or loading to other systems. When log events are sent to the receiving service, they are Base64 encoded and compressed with the gzip format." See https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/Subscriptions.html

Question: 63

A company has enabled all features for its organization in AWS Organizations. The organization contains 10 AWS accounts. The company has turned on AWS CloudTrail in all the accounts. The company expects the number of AWS accounts in the organization to increase to 500 during the next year. The company plans to use multiple OUs for these accounts.
The company has enabled AWS Config in each existing AWS account in the organization. A DevOps engineer must implement a solution that enables AWS Config automatically for all future AWS accounts that are created in the organization.
Which solution will meet this requirement?

A. In the organization's management account, create an Amazon EventBridge rule that reacts to a CreateAccount API call. Configure the rule to invoke an AWS Lambda function that enables trusted access to AWS Config for the organization.
B. In the organization's management account, create an AWS CloudFormation stack set to enable AWS Config. Configure the stack set to deploy automatically when an account is created through Organizations.
C. In the organization's management account, create an SCP that allows the appropriate AWS Config API calls to enable AWS Config. Apply the SCP to the root-level OU.
D. In the organization's management account, create an Amazon EventBridge rule that reacts to a CreateAccount API call. Configure the rule to invoke an AWS Systems Manager Automation runbook to enable AWS Config for the account.

_____ Answer: B

Explanation:

https://aws.amazon.com/about-aws/whats-new/2020/02/aws-cloudformation-stacksets-introduces-automatic-deployments-across-accounts-and-regions-through-aws-organizations/

Question: 64

A company has many applications. Different teams in the company developed the applications by using multiple languages and frameworks. The applications run on premises and on different servers with different operating systems. Each team has its own release protocol and process. The company wants to reduce the complexity of the release and maintenance of these applications.

The company is migrating its technology stacks, including these applications, to AWS. The company wants centralized control of source code, a consistent and automatic delivery pipeline, and as few maintenance tasks as possible on the underlying infrastructure.
What should a DevOps engineer do to meet these requirements?

A. Create one AWS CodeCommit repository for all applications. Put each application's code in a different branch. Merge the branches, and use AWS CodeBuild to build the applications. Use AWS CodeDeploy to deploy the applications to one centralized application server.
B. Create one AWS CodeCommit repository for each of the applications. Use AWS CodeBuild to build the applications one at a time. Use AWS CodeDeploy to deploy the applications to one centralized application server.
C. Create one AWS CodeCommit repository for each of the applications. Use AWS CodeBuild to build the applications one at a time and to create one AMI for each server. Use AWS CloudFormation StackSets to automatically provision and decommission Amazon EC2 fleets by using these AMIs.
D. Create one AWS CodeCommit repository for each of the applications. Use AWS CodeBuild to build one Docker image for each application in Amazon Elastic Container Registry (Amazon ECR). Use AWS CodeDeploy to deploy the applications to Amazon Elastic Container Service (Amazon ECS) on infrastructure that AWS Fargate manages.

Answer: D

Explanation:

because of "as few maintenance tasks as possible on the underlying infrastructure". Fargate does that better than "one centralized application server"

Question: 65

A company's application is currently deployed to a single AWS Region. Recently, the company opened a new office on a different continent. The users in the new office are experiencing high latency. The company's application runs on Amazon EC2 instances behind an Application Load Balancer (ALB) and uses Amazon DynamoDB as the database layer. The instances run in an EC2 Auto Scaling group across multiple Availability Zones. A DevOps engineer is tasked with minimizing application response times and improving availability for users in both Regions.
Which combination of actions should be taken to address the latency issues? (Choose three.)

A. Create a new DynamoDB table in the new Region with cross-Region replication enabled.
B. Create new ALB and Auto Scaling group global resources and configure the new ALB to direct traffic to the new Auto Scaling group.
C. Create new ALB and Auto Scaling group resources in the new Region and configure the new ALB to direct traffic to the new Auto Scaling group.
D. Create Amazon Route 53 records, health checks, and latency-based routing policies to route to the ALB.
E. Create Amazon Route 53 aliases, health checks, and failover routing policies to route to the ALB.
F. Convert the DynamoDB table to a global table.

Answer: C, D, F

Explanation:
C) Create new ALB and Auto Scaling group resources in the new Region and configure the new ALB to direct traffic to the new Auto Scaling group. This will allow users in the new Region to access the application with lower latency by reducing the network hops between the user and the application servers.

D) Create Amazon Route 53 records, health checks, and latency-based routing policies to route to the ALB. This will enable Route 53 to route user traffic to the nearest healthy ALB, based on the latency between the user and the ALBs.

F) Convert the DynamoDB table to a global table. This will enable reads and writes to the table in both Regions with low latency, improving the overall response time of the application

Question: 66

A DevOps engineer needs to apply a core set of security controls to an existing set of AWS accounts. The accounts are in an organization in AWS Organizations. Individual teams will administer individual accounts by using the AdministratorAccess AWS managed policy. For all accounts. AWS CloudTrail and AWS Config must be turned on in all available AWS Regions. Individual account administrators must not be able to edit or delete any of the baseline resources. However, individual account administrators must be able to edit or delete their own CloudTrail trails and AWS Config rules. Which solution will meet these requirements in the MOST operationally efficient way?

A. Create an AWS CloudFormation template that defines the standard account resources. Deploy the template to all accounts from the organization's management account by using CloudFormation StackSets. Set the stack policy to deny Update:Delete actions.
B. Enable AWS Control Tower. Enroll the existing accounts in AWS Control Tower. Grant the individual account administrators access to CloudTrail and AWS Config.
C. Designate an AWS Config management account. Create AWS Config recorders in all accounts by using AWS CloudFormation StackSets. Deploy AWS Config rules to the organization by using the AWS Config management account. Create a CloudTrail organization trail in the organization's management account. Deny modification or deletion of the AWS Config recorders by using an SCP.
D. Create an AWS CloudFormation template that defines the standard account resources. Deploy the template to all accounts from the organization's management account by using Cloud Formation StackSets Create an SCP that prevents updates or deletions to CloudTrail resources or AWS Config resources unless the principal is an administrator of the organization's management account.

Answer: D

Explanation:

Question: 67

A company has its AWS accounts in an organization in AWS Organizations. AWS Config is manually configured in each AWS account. The company needs to implement a solution to centrally configure AWS Config for all accounts in the organization The solution also must record resource changes to a central account.
Which combination of actions should a DevOps engineer perform to meet these requirements?

(Choose two.)

A. Configure a delegated administrator account for AWS Config. Enable trusted access for AWS Config in the organization.
B. Configure a delegated administrator account for AWS Config. Create a service-linked role for AWS Config in the organization's management account.
C. Create an AWS CloudFormation template to create an AWS Config aggregator. Configure a CloudFormation stack set to deploy the template to all accounts in the organization.
D. Create an AWS Config organization aggregator in the organization's management account. Configure data collection from all AWS accounts in the organization and from all AWS Regions.
E. Create an AWS Config organization aggregator in the delegated administrator account. Configure data collection from all AWS accounts in the organization and from all AWS Regions.

Answer: AE

Explanation:

https://aws.amazon.com/blogs/mt/org-aggregator-delegated-admin/

https://docs.aws.amazon.com/organizations/latest/userguide/services-that-can-integrate- config.html

Question: 68

A company wants to migrate its content sharing web application hosted on Amazon EC2 to a serverless architecture. The company currently deploys changes to its application by creating a new Auto Scaling group of EC2 instances and a new Elastic Load Balancer, and then shifting the traffic away using an Amazon Route 53 weighted routing policy.

For its new serverless application, the company is planning to use Amazon API Gateway and AWS Lambda. The company will need to update its deployment processes to work with the new application. It will also need to retain the ability to test new features on a small number of users before rolling the features out to the entire user base.

Which deployment strategy will meet these requirements?

A. Use AWS CDK to deploy API Gateway and Lambda functions. When code needs to be changed, update the AWS CloudFormation stack and deploy the new version of the APIs and Lambda functions. Use a Route 53 failover routing policy for the canary release strategy.

B. Use AWS CloudFormation to deploy API Gateway and Lambda functions using Lambda function versions. When code needs to be changed, update the CloudFormation stack with the new Lambda code and update the API versions using a canary release strategy. Promote the new version when testing is complete.

C. Use AWS Elastic Beanstalk to deploy API Gateway and Lambda functions. When code needs to be changed, deploy a new version of the API and Lambda functions. Shift traffic gradually using an Elastic Beanstalk blue/green deployment.

D. Use AWS OpsWorks to deploy API Gateway in the service layer and Lambda functions in a custom layer. When code needs to be changed, use OpsWorks to perform a blue/green deployment and shift traffic gradually.

Answer: B

Explanation:

https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/automating- updates-to-serverless-apps.html

Question: 69

A development team uses AWS CodeCommit, AWS CodePipeline, and AWS CodeBuild to develop and deploy an application. Changes to the code are submitted by pull requests. The development team reviews and merges the pull requests, and then the pipeline builds and tests the application. Over time, the number of pull requests has increased. The pipeline is frequently blocked because of failing tests. To prevent this blockage, the development team wants to run the unit and integration tests on each pull request before it is merged.

Which solution will meet these requirements?

A. Create a CodeBuild project to run the unit and integration tests. Create a CodeCommit approval rule template. Configure the template to require the successful invocation of the CodeBuild project. Attach the approval rule to the project's CodeCommit repository.

B. Create an Amazon EventBridge rule to match pullRequestCreated events from CodeCommit Create a CodeBuild project to run the unit and integration tests. Configure the CodeBuild project as a target of the EventBridge rule that includes a custom event payload with the CodeCommit repository and branch information from the event.

C. Create an Amazon EventBridge rule to match pullRequestCreated events from CodeCommit. Modify the existing CodePipeline pipeline to not run the deploy steps if the build is started from a pull request. Configure the EventBridge rule to run the pipeline with a custom payload that contains the CodeCommit repository and branch information from the event.

D. Create a CodeBuild project to run the unit and integration tests. Create a CodeCommit notification rule that matches when a pull request is created or updated. Configure the notification rule to invoke the CodeBuild project.

www.atmicnetworks.com

_____ Answer: B

Explanation:

CodeCommit generates events in CloudWatch, CloudWatch triggers the CodeBuild https://aws.amazon.com/es/blogs/devops/complete-ci-cd-with-aws-codecommit-aws-codebuild- aws-codedeploy-and-aws-codepipeline/

Question: 70

A company has an application that runs on a fleet of Amazon EC2 instances. The application requires frequent restarts. The application logs contain error messages when a restart is required. The application logs are published to a log group in Amazon CloudWatch Logs.
An Amazon CloudWatch alarm notifies an application engineer through an Amazon Simple Notification Service (Amazon SNS) topic when the logs contain a large number of restart-related error messages. The application engineer manually restarts the application on the instances after the application engineer receives a notification from the SNS topic.
A DevOps engineer needs to implement a solution to automate the application restart on the instances without restarting the instances.
Which solution will meet these requirements in the MOST operationally efficient manner?

A. Configure an AWS Systems Manager Automation runbook that runs a script to restart the application on the instances. Configure the SNS topic to invoke the runbook.
B. Create an AWS Lambda function that restarts the application on the instances. Configure the Lambda function as an event destination of the SNS topic.
C. Configure an AWS Systems Manager Automation runbook that runs a script to restart the application on the instances. Create an AWS Lambda function to invoke the runbook. Configure the Lambda function as an event destination of the SNS topic.
D. Configure an AWS Systems Manager Automation runbook that runs a script to restart the application on the instances. Configure an Amazon EventBridge rule that reacts when the CloudWatch alarm enters ALARM state. Specify the runbook as a target of the rule.

_____ Answer: D

Explanation:

This solution meets the requirements in the most operationally efficient manner by automating the application restart process on the instances without restarting them. When the CloudWatch alarm enters the ALARM state, the EventBridge rule is triggered, which in turn invokes the Systems Manager Automation runbook that contains the script to restart the application on the instances.

Question: 71

A DevOps engineer at a company is supporting an AWS environment in which all users use AWS IAM Identity Center (AWS Single Sign-On). The company wants to immediately disable credentials of any new IAM user and wants the security team to receive a notification.
Which combination of steps should the DevOps engineer take to meet these requirements? (Choose three.)

A. Create an Amazon EventBridge rule that reacts to an IAM CreateUser API call in AWS CloudTrail.
B. Create an Amazon EventBridge rule that reacts to an IAM GetLoginProfile API call in AWS CloudTrail.
C. Create an AWS Lambda function that is a target of the EventBridge rule. Configure the Lambda function to disable any access keys and delete the login profiles that are associated with the IAM user.
D. Create an AWS Lambda function that is a target of the EventBridge rule. Configure the Lambda function to delete the login profiles that are associated with the IAM user.
E. Create an Amazon Simple Notification Service (Amazon SNS) topic that is a target of the EventBridge rule. Subscribe the security team's group email address to the topic.
F. Create an Amazon Simple Queue Service (Amazon SQS) queue that is a target of the Lambda function. Subscribe the security team's

group email address to the queue.

Answer: A, C, E

Explanation:

Question: 72

A company wants to set up a continuous delivery pipeline. The company stores application code in a private GitHub repository. The company needs to deploy the application components to Amazon Elastic Container Service (Amazon ECS). Amazon EC2, and AWS Lambd a. The pipeline must support manual approval actions.

Which solution will meet these requirements?

A. Use AWS CodePipeline with Amazon ECS. Amazon EC2, and Lambda as deploy providers.

B. Use AWS CodePipeline with AWS CodeDeploy as the deploy provider.

C. Use AWS CodePipeline with AWS Elastic Beanstalk as the deploy provider.

D. Use AWS CodeDeploy with GitHub integration to deploy the application.

_____ Answer: B

Explanation:

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

Question: 73

A company has an application that runs on Amazon EC2 instances that are in an Auto Scaling group. When the application starts up. the application needs to process data from an Amazon S3 bucket before the application can start to serve requests.

The size of the data that is stored in the S3 bucket is growing. When the Auto Scaling group adds new instances, the application now takes several minutes to download and process the data before the application can serve requests. The company must reduce the time that elapses before new EC2 instances are ready to serve requests.

Which solution is the MOST cost-effective way to reduce the application startup time?

A. Configure a warm pool for the Auto Scaling group with warmed EC2 instances in the Stopped state. Configure an autoscaling:EC2_INSTANCE_LAUNCHING lifecycle hook on the Auto Scaling group. Modify the application to complete the lifecycle hook when the application is ready to serve requests.

B. Increase the maximum instance count of the Auto Scaling group. Configure an autoscaling:EC2_INSTANCE_LAUNCHING lifecycle hook on the Auto Scaling group. Modify the application to complete the lifecycle hook when the application is ready to serve requests.

C. Configure a warm pool for the Auto Scaling group with warmed EC2 instances in the Running state. Configure an autoscaling:EC2_INSTANCE_LAUNCHING lifecycle hook on the Auto Scaling group. Modify the application to complete the lifecycle hook when the application is ready to serve requests.

D. Increase the maximum instance count of the Auto Scaling group. Configure an autoscaling:EC2_INSTANCE_LAUNCHING lifecycle hook on the Auto Scaling group. Modify the application to complete the lifecycle hook and to place the new instance in the Standby state when the application is ready to serve requests.

_____ Answer: A

Explanation:

Option A is the most cost-effective solution. By configuring a warm pool of EC2 instances in the Stopped state, the company can reduce the time it takes for new instances to be ready to serve requests. When the Auto Scaling group launches a new instance, it can attach the stopped EC2 instance from the warm pool. The instance can then be started up immediately, rather than having to wait for the data to be downloaded and processed. This reduces the overall startup time for the application.

Question: 74

A company is using an AWS CodeBuild project to build and package an application. The packages are copied to a shared Amazon S3 bucket before being deployed across multiple AWS accounts.
The buildspec.yml file contains the following:

```
version: 0.2
phases:
  build:
    commands:
      - go build -o myapp
  post_build:
    commands:
      - aws s3 cp —acl authenticated-read myapp s3://artifacts/
```

The DevOps engineer has noticed that anybody with an AWS account is able to download the artifacts.
What steps should the DevOps engineer take to stop this?

A. Modify the post_build command to use --acl public-read and configure a bucket policy that grants read access to the relevant AWS accounts only.
B. Configure a default ACL for the S3 bucket that defines the set of authenticated users as the relevant AWS accounts only and grants read-only access.
C. Create an S3 bucket policy that grants read access to the relevant AWS accounts and denies read access to the principal "*".
D. Modify the post_build command to remove --acl authenticated-read and configure a bucket policy that allows read access to the relevant AWS accounts only.

_____ Answer: D

Explanation:

When setting the flag authenticated-read in the command line, the owner gets FULL_CONTROL. The AuthenticatedUsers group (Anyone with an AWS account) gets READ access. Reference: https://docs.aws.amazon.com/AmazonS3/latest/userguide/acl-overview.html

Question: 75

A company has developed a serverless web application that is hosted on AWS. The application consists of Amazon S3. Amazon API Gateway, several AWS Lambda functions, and an Amazon RDS for MySQL database. The company is using AWS CodeCommit to store the source code. The source code is a combination of AWS Serverless Application Model (AWS SAM) templates and Python code.
A security audit and penetration test reveal that user names and passwords for authentication to the database are hardcoded within CodeCommit repositories. A DevOps engineer must implement a solution to automatically detect and prevent hardcoded secrets.
What is the MOST secure solution that meets these requirements?

A. Enable Amazon CodeGuru Profiler. Decorate the handler function with @with_lambda_profiler(). Manually review the recommendation report. Write the secret to AWS Systems Manager Parameter Store as a secure string. Update the SAM templates and the Python code to pull the secret from Parameter Store.
B. Associate the CodeCommit repository with Amazon CodeGuru Reviewer. Manually check the code review for any recommendations. Choose the option to protect the secret. Update the SAM templates and the Python code to pull the secret from AWS Secrets Manager.
C. Enable Amazon CodeGuru Profiler. Decorate the handler function with @with_lambda_profiler(). Manually review the

recommendation report. Choose the option to protect the secret. Update the SAM templates and the Python code to pull the secret from AWS Secrets Manager.

D. Associate the CodeCommit repository with Amazon CodeGuru Reviewer. Manually check the code review for any recommendations. Write the secret to AWS Systems Manager Parameter Store as a string. Update the SAM templates and the Python code to pull the secret from Parameter Store.

_____ Answer: B

Explanation:

https://docs.aws.amazon.com/codecommit/latest/userguide/how-to-amazon-codeguru- reviewer.html

Question: 76

A company uses AWS CloudFormation stacks to deploy updates to its application. The stacks consist of different resources. The resources include AWS Auto Scaling groups, Amazon EC2 instances, Application Load Balancers (ALBs), and other resources that are necessary to launch and maintain independent stacks. Changes to application resources outside of CloudFormation stack updates are not allowed.

The company recently attempted to update the application stack by using the AWS CLI. The stack failed to update and produced the following error message: "ERROR: both the deployment and the CloudFormation stack rollback failed. The deployment failed because the following resource(s) failed to update: [AutoScalingGroup]."

The stack remains in a status of UPDATE_ROLLBACK_FAILED.

*

Which solution will resolve this issue?

A. Update the subnet mappings that are configured for the ALBs. Run the aws cloudformation update-stack-set AWS CLI command.

B. Update the 1AM role by providing the necessary permissions to update the stack. Run the aws cloudformation continue-update-rollback AWS CLI command.

C. Submit a request for a quota increase for the number of EC2 instances for the account. Run the aws cloudformation cancel-update-stack AWS CLI command.

D. Delete the Auto Scaling group resource. Run the aws cloudformation rollback-stack AWS CLI command.

Answer: B

Explanation:

https://repost.aws/knowledge-center/cloudformation-update-rollback-failed If your stack is stuck in the UPDATE_ROLLBACK_FAILED state after a failed update, then the only actions that you can perform on the stack are the ContinueUpdateRollback or DeleteStack operations.

Question: 77

A company uses AWS Secrets Manager to store a set of sensitive API keys that an AWS Lambda function uses. When the Lambda function is invoked, the Lambda function retrieves the API keys and makes an API call to an external service. The Secrets Manager secret is encrypted with the default AWS Key Management Service (AWS KMS) key.

A DevOps engineer needs to update the infrastructure to ensure that only the Lambda function's execution role can access the values in Secrets Manager. The solution must apply the principle of least privilege.

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

A. Update the default KMS key for Secrets Manager to allow only the Lambda function's execution role to decrypt.
B. Create a KMS customer managed key that trusts Secrets Manager and allows the Lambda function's execution role to decrypt. Update Secrets Manager to use the new customer managed key.
C. Create a KMS customer managed key that trusts Secrets Manager and allows the account's :root principal to decrypt. Update Secrets Manager to use the new customer managed key.
D. Ensure that the Lambda function's execution role has the KMS permissions scoped on the resource level. Configure the permissions so that the KMS key can encrypt the Secrets Manager secret.
E. Remove all KMS permissions from the Lambda function's execution role.

Answer: B, D

Explanation:

The requirement is to update the infrastructure to ensure that only the Lambda function's execution role can access the values in Secrets Manager. The solution must apply the principle of least privilege, which means granting the minimum permissions necessary to perform a task.
To do this, the DevOps engineer needs to use the following steps:
Create a KMS customer managed key that trusts Secrets Manager and allows the Lambda function's execution role to decrypt. A customer managed key is a symmetric encryption key that is fully managed by the customer. The customer can define the key policy, which specifies who can use and manage the key. By creating a customer managed key, the DevOps engineer can restrict the decryption permission to only the Lambda function's execution role, and prevent other principals from accessing the secret values. The customer managed key also needs to trust Secrets Manager, which means allowing Secrets Manager to use the key to encrypt and decrypt secrets on behalf of the customer.
Update Secrets Manager to use the new customer managed key. Secrets Manager allows customers to choose which KMS key to use for encrypting each secret. By default, Secrets Manager uses the default KMS key for Secrets Manager, which is a service-managed key that is shared by all customers in the same AWS Region. By updating Secrets Manager to use the new customer managed key, the DevOps engineer can ensure that only the Lambda function's execution role can decrypt the secret values using that key.
Ensure that the Lambda function's execution role has the KMS permissions scoped on the resource level. The Lambda function's execution role is an IAM role that grants permissions to the Lambda function to access AWS services and resources. The role needs to have KMS permissions to use the customer managed key for decryption. However, to apply the principle of least privilege, the role should have the permissions scoped on the resource level, which means specifying the ARN of the customer managed key as a condition in the IAM policy statement. This way, the role can only use that specific key and not any other KMS keys in the account.

Question: 78

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 lAM 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.

_____ Answer: A

Explanation:

(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: 79

A company is hosting a static website from an Amazon S3 bucket. The website is available to customers at example.com. The company uses an Amazon Route 53 weighted routing policy with a TTL of 1 day. The company has decided to replace the existing static website with a dynamic web application. The dynamic web application uses an Application Load Balancer (ALB) in front of a fleet of Amazon EC2 instances.

On the day of production launch to customers, the company creates an additional Route 53 weighted DNS record entry that points to the ALB with a weight of 255 and a TTL of 1 hour. Two days later, a DevOps engineer notices that the previous static website is displayed sometimes when customers navigate to example.com.

How can the DevOps engineer ensure that the company serves only dynamic content for example.com?

A. Delete all objects, including previous versions, from the S3 bucket that contains the static website content.

B. Update the weighted DNS record entry that points to the S3 bucket. Apply a weight of 0. Specify the domain reset option to propagate changes immediately.

C. Configure webpage redirect requests on the S3 bucket with a hostname that redirects to the ALB.

D. Remove the weighted DNS record entry that points to the S3 bucket from the example.com hosted zone. Wait for DNS propagation to become complete.

_____ Answer: D

Explanation:

Question: 80

A company manages multiple AWS accounts in AWS Organizations. The company's security policy states that AWS account root user credentials for member accounts must not be used. The company monitors access to the root user credentials.

A recent alert shows that the root user in a member account launched an Amazon EC2 instance. A DevOps engineer must create an SCP at the organization's root level that will prevent the root user in member accounts from making any AWS service API calls.

Which SCP will meet these requirements?

A)

www.atmicnetworks.com

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "StringNotLike": { "aws:PrincipalArn": "arn:aws:iam::*:root" }
            }
        }
    ]
}
```

B)

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "*",
            "Resource": "*",
            "Principal": { "AWS": "arn:aws:iam::*:root" }
        }
    ]
}
```

C)

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action":
            "Resource":
            "Condition": {
                "StringLike": ( "aws:PrincipalArn": "arn:aws:iam::*:root" } )
        }
    ]
```

D)

"Version": "201 "Statement": (

(

Effect": "Allow

Action": n *11 9

www.atmicnetworks.com

A. Option A
B. Option B
C. Option C
D. Option D

_____Answer: D

Explanation:

Question: 81

A global company manages multiple AWS accounts by using AWS Control Tower. The company hosts internal applications and public applications.

Each application team in the company has its own AWS account for application hosting. The accounts are consolidated in an organization in AWS Organizations. One of the AWS Control Tower member accounts serves as a centralized DevOps account with CI/CD pipelines that application teams use to deploy applications to their respective target AWS accounts. An 1AM role for deployment exists in the centralized DevOps account.

An application team is attempting to deploy its application to an Amazon Elastic Kubernetes Service (Amazon EKS) cluster in an application AWS account. An 1AM role for deployment exists in the application AWS account. The deployment is through an AWS CodeBuild project that is set up in the centralized DevOps account. The CodeBuild project uses an 1AM service role for CodeBuild. The deployment is failing with an Unauthorized error during attempts to connect to the cross-account EKS cluster from CodeBuild.

Which solution will resolve this error?

A. Configure the application account's deployment 1AM role to have a trust relationship with the centralized DevOps account. Configure the trust relationship to allow the sts:AssumeRole action. Configure the application account's deployment 1AM role to have the required access to the EKS cluster. Configure the EKS cluster aws-auth ConfigMap to map the role to the appropriate system permissions.

B. Configure the centralized DevOps account's deployment I AM role to have a trust relationship with the application account. Configure the trust relationship to allow the sts:AssumeRole action. Configure the centralized DevOps account's deployment 1AM role to allow the required access to CodeBuild.

C. Configure the centralized DevOps account's deployment 1AM role to have a trust relationship with the application account. Configure the trust relationship to allow the sts:AssumeRoleWithSAML action. Configure the centralized DevOps account's deployment 1AM role to allow the required access to CodeBuild.

D. Configure the application account's deployment 1AM role to have a trust relationship with the AWS Control Tower management account. Configure the trust relationship to allow the sts:AssumeRole action. Configure the application account's deployment 1AM role to have the required access to the EKS cluster. Configure the EKS cluster aws-auth ConfigMap to map the role to the appropriate system permissions.

_____Answer: A

Explanation:

In the source AWS account, the IAM role used by the CI/CD pipeline should have permissions to access the source code repository, build artifacts, and any other resources required for the build process. In the destination AWS accounts, the IAM role used for deployment should have permissions to access the AWS resources required for deploying the application, such as EC2 instances, RDS databases, S3 buckets, etc. The exact permissions required will depend on the specific resources being used by the application. the IAM role used for deployment in the destination accounts should also have permissions to assume the IAM role for deployment in the centralized DevOps account. This is typically done using an IAM role trust policy that allows the destination account to assume the DevOps account role.

Question: 82

A company is implementing AWS CodePipeline to automate its testing process The company wants to be notified when the execution state fails and used the following custom event pattern in Amazon EventBridge:

```
{
  "source": [
    "aws.codepipeline"
  ],
  "detail-type": [
    "CodePipeline Action Execution State Change"
  ],
  "detail": {
    "state": [
      "FAILED"
    ],
    "type": {
      "category": ["Approval"]
    }
  }
}
```

Which type of events will match this event pattern?

A. Failed deploy and build actions across all the pipelines
B. All rejected or failed approval actions across all the pipelines
C. All the events across all pipelines
D. Approval actions across all the pipelines

Answer: B

Explanation:

Action-level states in events
Action state Description
STARTED The action is currently running.
SUCCEEDED The action was completed successfully.
FAILED For Approval actions, the FAILED state means the action was either rejected by the reviewer or failed due to an incorrect action configuration.
CANCELED The action was canceled because the pipeline structure was updated.

www.atmicnetworks.com

Question: 83

A DevOps engineer has implemented a CI/CO pipeline to deploy an AWS Cloud Format ion template that provisions a web application. The web application consists of an Application Load Balancer (ALB) a target group, a launch template that uses an Amazon Linux 2 AMI an Auto Scaling group of Amazon EC2 instances, a security group and an Amazon RDS for MySQL database The launch template includes user data that specifies a script to install and start the application.

The initial deployment of the application was successful. The DevOps engineer made changes to update the version of the application with the user dat

a. The CI/CD pipeline has deployed a new version of the template However, the health checks on the ALB are now failing The health checks have marked all targets as unhealthy.

During investigation the DevOps engineer notices that the Cloud Formation stack has a status of UPDATE_COMPLETE. However, when the DevOps engineer connects to one of the EC2 instances and checks /varar/log messages, the DevOps engineer notices that the Apache web server failed to start successfully because of a configuration error

How can the DevOps engineer ensure that the CloudFormation deployment will fail if the user data fails to successfully finish running?

A. Use the cfn-signal helper script to signal success or failure to CloudFormation Use the WaitOnResourceSignals update policy within the CloudFormation template Set an appropriate timeout for the update policy.

B. Create an Amazon CloudWatch alarm for the UnhealthyHostCount metric. Include an appropriate alarm threshold for the target group Create an Amazon Simple Notification Service (Amazon SNS) topic as the target to signal success or failure to CloudFormation

C. Create a lifecycle hook on the Auto Scaling group by using the AWS AutoScaling LifecycleHook resource Create an Amazon Simple Notification Service (Amazon SNS) topic as the target to signal success or failure to CloudFormation Set an appropriate timeout on the lifecycle hook.

D. Use the Amazon CloudWatch agent to stream the cloud-init logs Create a subscription filter that includes an AWS Lambda function with an appropriate invocation timeout Configure the Lambda function to use the SignalResource API operation to signal success or failure to CloudFormation.

Answer: A

Explanation:

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-attribute- updatepolicy.html

Question: 84

A company has a legacy application A DevOps engineer needs to automate the process of building the deployable artifact for the legacy application. The solution must store the deployable artifact in an existing Amazon S3 bucket for future deployments to reference Which solution will meet these requirements in the MOST operationally efficient way?

A. Create a custom Docker image that contains all the dependencies tor the legacy application Store the custom Docker image in a new Amazon Elastic Container Registry (Amazon ECR) repository Configure a new AWS CodeBuild project to use the custom Docker image to build the deployable artifact and to save the artifact to the S3 bucket.

B. Launch a new Amazon EC2 instance Install all the dependencies (or the legacy application on the EC2 instance Use the EC2 instance to build the deployable artifact and to save the artifact to the S3 bucket.

C. Create a custom EC2 Image Builder image Install all the dependencies for the legacy application on the image Launch a new Amazon EC2 instance from the image Use the new EC2 instance to build the deployable artifact and to save the artifact to the S3 bucket.

D. Create an Amazon Elastic Kubernetes Service (Amazon EKS) cluster with an AWS Fargate profile that runs in multiple Availability Zones Create a custom Docker image that contains all the dependencies for the legacy application Store the custom Docker image in a new Amazon Elastic Container Registry (Amazon ECR) repository Use the custom Docker image inside the EKS cluster to build the deployable artifact and to save the artifact to the S3 bucket.

www.atmicnetworks.com

_____ Answer: A

Explanation:

This approach is the most operationally efficient because it leverages the benefits of containerization, such as isolation and reproducibility, as well as AWS managed services. AWS CodeBuild is a fully managed build service that can compile your source code, run tests, and produce deployable software packages. By using a custom Docker image that includes all dependencies, you can ensure that the environment in which your code is built is consistent. Using Amazon ECR to store Docker images lets you easily deploy the images to any environment. Also, you can directly upload the build artifacts to Amazon S3 from AWS CodeBuild, which is beneficial for version control and archival purposes.

Question: 85

A company has a data ingestion application that runs across multiple AWS accounts. The accounts are in an organization in AWS Organizations. The company needs to monitor the application and consolidate access to the application. Currently the company is running the application on Amazon EC2 instances from several Auto Scaling groups. The EC2 instances have no access to the internet because the data is sensitive Engineers have deployed the necessary VPC endpoints. The EC2 instances run a custom AMI that is built specifically tor the application.

To maintain and troubleshoot the application, system administrators need the ability to log in to the EC2 instances. This access must be automated and controlled centrally. The company's security team must receive a notification whenever the instances are accessed.

Which solution will meet these requirements?

A. Create an Amazon EventBridge rule to send notifications to the security team whenever a user logs in to an EC2 instance Use EC2 Instance Connect to log in to the instances. Deploy Auto Scaling groups by using AWS Cloud Formation Use the cfn-init helper script to deploy appropriate VPC routes for external access Rebuild the custom AMI so that the custom AMI includes AWS Systems Manager Agent.

B. Deploy a NAT gateway and a bastion host that has internet access Create a security group that allows incoming traffic on all the EC2 instances from the bastion host Install AWS Systems Manager Agent on all the EC2 instances Use Auto Scaling group lifecycle hooks for monitoring and auditing access Use Systems Manager Session Manager to log in to the instances Send logs to a log group m Amazon CloudWatch Logs. Export data to Amazon S3 for auditing Send notifications to the security team by using S3 event notifications.

C. Use EC2 Image Builder to rebuild the custom AMI Include the most recent version of AWS Systems Manager Agent in the Image Configure the Auto Scaling group to attach the AmazonSSMManagedinstanceCore role to all the EC2 instances Use Systems Manager Session Manager to log in to the instances Enable logging of session details to Amazon S3 Create an S3 event notification for new file uploads to send a message to the security team through an Amazon Simple Notification Service (Amazon SNS) topic.

D. Use AWS Systems Manager Automation to build Systems Manager Agent into the custom AMI Configure AWS Configure to attach an SCP to the root organization account to allow the EC2 instances to connect to Systems Manager Use Systems Manager Session Manager to log in to the instances Enable logging of session details to Amazon S3 Create an S3 event notification for new file uploads to send a message to the security team through an Amazon Simple Notification Service (Amazon SNS) topic.

Answer: C

Explanation:

Even if AmazonSSMManagedInstanceCore is a managed policy and not an IAM role I will go with C because this policy is to be attached to an IAM role for EC2 to access System Manager.

Question: 86

A company recently migrated its legacy application from on-premises to AWS. The application is hosted on Amazon EC2 instances behind

www.atmicnetworks.com

an Application Load Balancer which is behind Amazon API Gateway. The company wants to ensure users experience minimal disruptions during any deployment of a new version of the application. The company also wants to ensure it can quickly roll back updates if there is an issue.

Which solution will meet these requirements with MINIMAL changes to the application?

A. Introduce changes as a separate environment parallel to the existing one Configure API Gateway to use a canary release deployment to send a small subset of user traffic to the new environment.

B. Introduce changes as a separate environment parallel to the existing one Update the application's DNS alias records to point to the new environment.

C. Introduce changes as a separate target group behind the existing Application Load Balancer Configure API Gateway to route user traffic to the new target group in steps.

D. Introduce changes as a separate target group behind the existing Application Load Balancer Configure API Gateway to route all traffic to the Application Load Balancer which then sends the traffic to the new target group.

_____ Answer: A

Explanation:

API Gateway supports canary deployment on a deployment stage before you direct all traffic to that stage. A parallel environment means we will create a new ALB and a target group that will target a new set of EC2 instances on which the newer version of the app will be deployed. So the canary setting associated to the new version of the API will connect with the new ALB instance which in turn will direct the traffic to the new EC2 instances on which the newer version of the application is deployed.

Question: 87

A development team manually builds an artifact locally and then places it in an Amazon S3 bucket. The application has a local cache that must be cleared when a deployment occurs. The team runs a command to do this downloads the artifact from Amazon S3 and unzips the artifact to complete the deployment.

A DevOps team wants to migrate to a CI/CD process and build in checks to stop and roll back the deployment when a failure occurs. This requires the team to track the progression of the deployment.

Which combination of actions will accomplish this? (Select THREE)

A. Allow developers to check the code into a code repository Using Amazon EventBridge on every pull into the mam branch invoke an AWS Lambda function to build the artifact and store it in Amazon S3.

B. Create a custom script to clear the cache Specify the script in the BeforeInstall lifecycle hook in the AppSpec file.

C. Create user data for each Amazon EC2 instance that contains the clear cache script Once deployed test the application If it is not successful deploy it again.

D. Set up AWS CodePipeline to deploy the application Allow developers to check the code into a code repository as a source tor the pipeline.

E. Use AWS CodeBuild to build the artifact and place it in Amazon S3 Use AWS CodeDeploy to deploy the artifact to Amazon EC2 instances.

F. Use AWS Systems Manager to fetch the artifact from Amazon S3 and deploy it to all the instances.

Answer: BDE

Explanation:

Question: 88 A DevOps engineer is working on a data archival project that requires the migration of on-premises data to an Amazon S3 bucket. The DevOps engineer develops a script that incrementally archives onpremises data that is older than 1 month to Amazon S3. Data that is transferred to Amazon S3 is deleted from the on-premises location The script uses the S3 PutObject operation.

During a code review the DevOps engineer notices that the script does not verity whether the data was successfully copied to Amazon S3. The DevOps engineer must update the script to ensure that data is not corrupted during transmission. The script must use MD5 checksums to verify data integrity before the on-premises data is deleted.

Which solutions for the script will meet these requirements'? (Select TWO.)

A. Check the returned response for the Versioned Compare the returned Versioned against the MD5 checksum.

B. Include the MD5 checksum within the Content-MD5 parameter. Check the operation call's return status to find out if an error was returned.

C. Include the checksum digest within the tagging parameter as a URL query parameter.

D. Check the returned response for the ETag. Compare the returned ETag against the MD5 checksum.

E. Include the checksum digest within the Metadata parameter as a name-value pair After upload use the S3 HeadObject operation to retrieve metadata from the object.

Answer: BD

Explanation:

https://docs.aws.amazon.com/AmazonS3/latest/userguide/checking-object-integrity.html

Question: 89

A DevOps engineer used an AWS Cloud Formation custom resource to set up AD Connector. The AWS Lambda function ran and created AD Connector, but Cloud Formation is not transitioning from CREATE_IN_PROGRESS to CREATE_COMPLETE.

Which action should the engineer take to resolve this issue?

A. Ensure the Lambda function code has exited successfully.

B. Ensure the Lambda function code returns a response to the pre-signed URL.

C. Ensure the Lambda function IAM role has cloudformation UpdateStack permissions for the stack ARN.

D. Ensure the Lambda function IAM role has ds ConnectDirectory permissions for the AWS account.

Answer: B

Explanation:

Reference: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/crpg-ref- responses.html

Question: 90

A DevOps engineer is working on a project that is hosted on Amazon Linux and has failed a security review. The DevOps manager has been asked to review the company buildspec. yaml die for an AWS

CodeBuild project and provide recommendations. The buildspec. yaml file is configured as follows:

```
env:
  variables:
    AWS_ACCESS_KEY_ID: AF.:AJF?BRFW JBA4GHXNA
    AWS_SE€RET_ACCESS_KEY: ORj 'ns3At2aIh4O4Ata0*zHxZqz?cNAvMLYRehCi AMS_D£FAVLT_REG1ON:
    us~east~l
    DB_PASSWORD:          >RptFa3va s3«//db* deploy b'ic ke t/ay• m£• teaplate . tap/ioy.mr
  phases:                 s/DB_FW/$[DB_fASSWORD]I i trap-'av.cnf
    build:                •3://db-deploy-bucket/instance.key /tap/instance.key /tap/instance.key
      commands:                 - scp -i /trap/instance.key /tap/ay.cnf root»10.25.15.23:/etc/my.cnf
        - aws s3 cp             - ssh -i /trap/instance.key rcctdlO.25.15.23 /etc/init.d/aysqld restart
        - sed -i ''
        - aws s3 cp
        - chmod 600
```

What changes should be recommended to comply with AWS security best practices? (Select THREE.)

A. Add a post-build command to remove the temporary files from the container before termination to ensure they cannot be seen by other CodeBuild users.

B. Update the CodeBuild project role with the necessary permissions and then remove the AWS credentials from the environment variable.

C. Store the db_password as a SecureString value in AWS Systems Manager Parameter Store and then remove the db_password from the environment variables.

D. Move the environment variables to the 'db.-deploy-bucket 'Amazon S3 bucket, add a prebuild stage to download then export the variables.

E. Use AWS Systems Manager run command versus sec and ssh commands directly to the instance.

Answer: BCE

Explanation:

B) Update the CodeBuild project role with the necessary permissions and then remove the AWS credentials from the environment variable. C. Store the DB_PASSWORD as a SecureString value in AWS Systems Manager Parameter Store and then remove the DB_PASSWORD from the environment variables. E. Use AWS Systems Manager run command versus scp and ssh commands directly to the instance.

Question: 91

A DevOps engineer is researching the least expensive way to implement an image batch processing cluster on AWS. The application cannot run in Docker containers and must run on Amazon EC2. The batch job stores checkpoint data on an NFS volume and can tolerate interruptions. Configuring the cluster software from a generic EC2 Linux image takes 30 minutes.
What is the MOST cost-effective solution?

A. Use Amazon EFS (or checkpoint data. To complete the job, use an EC2 Auto Scaling group and an On-Demand pricing model to provision EC2 instances temporally.

B. Use GlusterFS on EC2 instances for checkpoint data. To run the batch job configure EC2 instances manually When the job completes shut down the instances manually.

C. Use Amazon EFS for checkpoint data Use EC2 Fleet to launch EC2 Spot Instances and utilize user data to configure the EC2 Linux instance on startup.

D. Use Amazon EFS for checkpoint data Use EC2 Fleet to launch EC2 Spot Instances Create a custom AMI for the cluster and use the latest AMI when creating instances.

_____ Answer: D

Explanation:

Question: 92

A company has a guideline that every Amazon EC2 instance must be launched from an AMI that the company's security team produces Every month the security team sends an email message with the latest approved AMIs to all the development teams.
The development teams use AWS CloudFormation to deploy their applications. When developers launch a new service they have to search their email for the latest AMIs that the security department sent. A DevOps engineer wants to automate the process that the security team uses to provide the AMI IDs to the development teams.
What is the MOST scalable solution that meets these requirements?

A. Direct the security team to use CloudFormation to create new versions of the AMIs and to list! the AMI ARNs in an encrypted Amazon S3 object as part of the stack's Outputs Section Instruct the developers to use a cross-stack reference to load the encrypted S3 object and obtain the most recent AMI ARNs.
B. Direct the security team to use a CloudFormation stack to create an AWS CodePipeline pipeline that builds new AMIs and places the latest AMI ARNs in an encrypted Amazon S3 object as part of the pipeline output Instruct the developers to use a cross-stack reference within their own CloudFormation template to obtain the S3 object location and the most recent AMI ARNs.
C. Direct the security team to use Amazon EC2 Image Builder to create new AMIs and to place the AMI ARNs as parameters in AWS Systems Manager Parameter Store Instruct the developers to specify a parameter of type SSM in their CloudFormation stack to obtain the most recent AMI ARNs from Parameter Store.
D. Direct the security team to use Amazon EC2 Image Builder to create new AMIs and to create an Amazon Simple Notification Service (Amazon SNS) topic so that every development team can receive notifications. When the development teams receive a notification instruct them to write an AWS Lambda function that will update their CloudFormation stack with the most recent AMI ARNs.

Answer: C

Explanation:

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/dynamic-references.html

Question: 93

A DevOps team is merging code revisions for an application that uses an Amazon RDS Multi-AZ DB cluster for its production database. The DevOps team uses continuous integration to periodically verify that the application works. The DevOps team needs to test the changes before the changes are

deployed to the production database.
Which solution will meet these requirements'?

A. Use a buildspec file in AWS CodeBuild to restore the DB cluster from a snapshot of the production database run integration tests, and drop the restored database after verification.
B. Deploy the application to production. Configure an audit log of data control language (DCL) operations to capture database activities to perform if verification fails.
C. Create a snapshot of the DB duster before deploying the application Use the Update requires Replacement property on the DB instance in AWS CloudFormation to deploy the application and apply the changes.
D. Ensure that the DB cluster is a Multi-AZ deployment. Deploy the application with the updates. Fail over to the standby instance if verification fails.

Answer: A

Explanation:

This solution will meet the requirements because it will create a temporary copy of the production database using a snapshot, run the integration tests on the copy, and delete the copy after the tests are done. This way, the production database will not be affected by the code revisions, and the DevOps team can test the changes before deploying them to production. A buildspec file is a YAML file that contains the commands and settings that CodeBuild uses to run a build1. The buildspec file can specify the steps to restore the DB cluster from a snapshot, run the integration tests, and drop the restored database2

Question: 94

A company is divided into teams Each team has an AWS account and all the accounts are in an organization in AWS Organizations. Each team must retain full administrative rights to its AWS account. Each team also must be allowed to access only AWS services that the company approves for use AWS services must gam approval through a request and approval process.
How should a DevOps engineer configure the accounts to meet these requirements?

A. Use AWS CloudFormation StackSets to provision IAM policies in each account to deny access to restricted AWS services. In each account configure AWS Config rules that ensure that the policies are attached to IAM principals in the account.
B. Use AWS Control Tower to provision the accounts into OUs within the organization Configure AWS Control Tower to enable AWS IAM identity Center (AWS Single Sign-On). Configure 1AM Identity Center to provide administrative access Include deny policies on user roles for restricted AWS services.
C. Place all the accounts under a new top-level OU within the organization Create an SCP that denies access to restricted AWS services Attach the SCP to the OU.
D. Create an SCP that allows access to only approved AWS services. Attach the SCP to the root OU of the organization. Remove the FullAWSAccess SCP from the root OU of the organization.

Answer: C

Explanation:

https://docs.aws.amazon.com/vpc/latest/userguide/managed-prefix-lists.html A managed prefix list is a set of one or more CIDR blocks. You can use prefix lists to make it easier to configure and maintain your security groups and route tables.
https://docs.aws.amazon.com/vpc/latest/userguide/sharing-managed-prefix-lists.html With AWS Resource Access Manager (AWS RAM), the owner of a prefix list can share a prefix list with the following: Specific AWS accounts inside or outside of its organization in AWS Organizations An organizational unit inside its organization in AWS Organizations An entire organization in AWS Organizations

Question: 95

An Amazon EC2 instance is running in a VPC and needs to download an object from a restricted Amazon S3 bucket. When the DevOps engineer tries to download the object, an AccessDenied error is received,
What are the possible causes tor this error? (Select TWO,)

A. The 53 bucket default encryption is enabled.
B. There is an error in the S3 bucket policy.
C. The object has been moved to S3 Glacier.
D. There is an error in the IAM role configuration.
E. S3 Versioning is enabled.

Answer: BD

Explanation:

www.atmicnetworks.com

These are the possible causes for the AccessDenied error because they affect the permissions to access the S3 object from the EC2 instance. An S3 bucket policy is a resource-based policy that defines who can access the bucket and its objects, and what actions they can perform. An IAM role is an identity that can be assumed by an EC2 instance to grant it permissions to access AWS services and resources. If there is an error in the S3 bucket policy or the IAM role configuration, such as a missing or incorrect statement, condition, or principal, then the EC2 instance may not have the necessary permissions to download the object from the S3 bucket .
https://docs.aws.amazon.com/AmazonS3/latest/userguide/example-bucket-policies.htmlhttps://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html

Question: 96

A company uses a single AWS account lo test applications on Amazon EC2 instances. The company has turned on AWS Config in the AWS account and has activated the restricted-ssh AWS Config managed rule.
The company needs an automated monitoring solution that will provide a customized notification in real time if any security group in the account is not compliant with the restricted-ssh rule. The customized notification must contain the name and ID of the noncompliant security group.
A DevOps engineer creates an Amazon Simple Notification Service (Amazon SNS) topic in the account and subscribes the appropriate personnel to the topic.
What should me DevOps engineer do next to meet these requirements?

A. Create an Amazon EventBridge rule that matches an AWS Config evaluation result of NON_COMPLIANT tor the restricted-ssh rule. Configure an input transformer for the EventBridge rule Configure the EventBridge rule to publish a notification to the SNS topic.

B. Configure AWS Config to send all evaluation results for the restricted-ssh rule to the SNS topic. Configure a filter policy on the SNS topic to send only notifications that contain the text of NON_COMPLIANT in the notification to subscribers.

C. Create an Amazon EventBridge rule that matches an AWS Config evaluation result of NON_COMPLIANT for the restricted-ssh rule Configure the EventBridge rule to invoke AWS Systems Manager Run Command on the SNS topic to customize a notification and to publish the notification to the SNS topic

D. Create an Amazon EventBridge rule that matches all AWS Config evaluation results of NON_COMPLIANT Configure an input transformer for the restricted-ssh rule Configure the EventBridge rule to publish a notification to the SNS topic.

Answer: A

Explanation:

Create an Amazon EventBridge (Amazon CloudWatch Events) rule that matches an AWS Config evaluation result of NON_COMPLIANT for the restricted-ssh rule. Configure an input transformer for the EventBridge (CloudWatch Events) rule. Configure the EventBridge (CloudWatch Events) rule to publish a notification to the SNS topic. This approach uses Amazon EventBridge (previously known as Amazon CloudWatch Events) to filter AWS Config evaluation results based on the restricted-ssh rule and its compliance status (NON_COMPLIANT). An input transformer can be used to customize the information contained in the notification, such as the name and ID of the noncompliant security group. The EventBridge (CloudWatch Events) rule can then be configured to publish a notification to the SNS topic, which will notify the appropriate personnel in real-time.

Question: 97

A DevOps engineer is architecting a continuous development strategy for a company's software as a service (SaaS) web application running on AWS. For application and security reasons users subscribing to this application are distributed across multiple. Application Load Balancers (ALBs) each of which has a dedicated Auto Scaling group and fleet of Amazon EC2 instances The application does not require a build stage and when it is committed to AWS CodeCommit, the application must trigger a simultaneous deployment to all ALBs Auto Scaling groups and EC2 fleets.
Which architecture will meet these requirements with the LEAST amount of configuration?

A. Create a single AWS CodePipeline pipeline that deploys the application in parallel using unique AWS CodeDeploy applications and deployment groups created for each ALB-Auto Scaling group pair.

B. Create a single AWS CodePipeline pipeline that deploys the application using a single AWS CodeDeploy application and single deployment group.

C. Create a single AWS CodePipeline pipeline that deploys the application in parallel using a single AWS CodeDeploy application and unique deployment group for each ALB-Auto Scaling group pair.

D. Create an AWS CodePipeline pipeline for each ALB-Auto Scaling group pair that deploys the application using an AWS CodeDeploy application and deployment group created for the same ALB- Auto Scaling group pair.

_____ Answer: C

Explanation:

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

Question: 98

A company recently created a new AWS Control Tower landing zone in a new organization in AWS Organizations. The landing zone must be able to demonstrate compliance with the Center tor Internet Security (CIS) Benchmarks tor AWS Foundations.
The company's security team wants to use AWS Security Hub to view compliance across all accounts Only the security team can be allowed to view aggregated Security Hub Findings. In addition specific users must be able to view findings from their own accounts within the organization All accounts must be enrolled m Security Hub after the accounts are created.
Which combination of steps will meet these requirements in the MOST automated way? (Select THREE.)

A. Turn on trusted access for Security Hub in the organization's management account. Create a new security account by using AWS Control Tower Configure the new security account as the delegated administrator account for Security Hub. In the new security account provide. Security Hub with the CIS Benchmarks for AWS Foundations standards.

B. Turn on trusted access for Security Hub in the organ ration's management account. From the management account, provide Security Hub with the CIS Benchmarks for AWS Foundations standards.

C. Create an AWS IAM identity Center (AWS Single Sign-On) permission set that includes the required permissions Use the CreateAccountAssignment API operation to associate the security team users with the permission set and with the delegated security account.

D. Create an SCP that explicitly denies any user who is not on the security team from accessing Security Hub.

E. In Security Hub, turn on automatic enablement.

F. In the organization's management account create an Amazon EventBridge rule that reacts to the CreateManagedAccount event Create an AWS Lambda function that uses the Security Hub CreateMembers API operation to add new accounts to Security Hub. Configure the EventBridge rule to invoke the Lambda function.

Answer: ACE

Explanation:

https://docs.aws.amazon.com/securityhub/latest/userguide/accounts-orgs-auto-enable.html

Question: 99

A highly regulated company has a policy that DevOps engineers should not log in to their Amazon EC2 instances except in emergencies. It a DevOps engineer does log in the security team must be notified within 15 minutes of the occurrence.
Which solution will meet these requirements'?

A. Install the Amazon Inspector agent on each EC2 instance Subscribe to Amazon EventBridge notifications Invoke an AWS Lambda function to check if a message is about user logins If it is send a notification to the security team using Amazon SNS.
B. Install the Amazon CloudWatch agent on each EC2 instance Configure the agent to push all logs to Amazon CloudWatch Logs and set up a CloudWatch metric filter that searches for user logins. If a login is found send a notification to the security team using Amazon SNS.
C. Set up AWS CloudTrail with Amazon CloudWatch Logs. Subscribe CloudWatch Logs to Amazon Kinesis Attach AWS Lambda to Kinesis to parse and determine if a log contains a user login If it does, send a notification to the security team using Amazon SNS.
D. Set up a script on each Amazon EC2 instance to push all logs to Amazon S3 Set up an S3 event to invoke an AWS Lambda function which invokes an Amazon Athena query to run. The Athena query checks tor logins and sends the output to the security team using Amazon SNS.

_____Answer: B

Explanation:

https://aws.amazon.com/blogs/security/how-to-monitor-and-visualize-failed-ssh-access-attempts- to-amazon-ec2-linux-instances/

Question: 100

A company wants to ensure that their EC2 instances are secure. They want to be notified if any new vulnerabilities are discovered on their instances and they also want an audit trail of all login activities on the instances.
Which solution will meet these requirements'?

A. Use AWS Systems Manager to detect vulnerabilities on the EC2 instances Install the Amazon Kinesis Agent to capture system logs and deliver them to Amazon S3.
B. Use AWS Systems Manager to detect vulnerabilities on the EC2 instances Install the Systems Manager Agent to capture system logs and view login activity in the CloudTrail console.
C. Configure Amazon CloudWatch to detect vulnerabilities on the EC2 instances Install the AWS Config daemon to capture system logs and view them in the AWS Config console.
D. Configure Amazon Inspector to detect vulnerabilities on the EC2 instances Install the Amazon CloudWatch Agent to capture system logs and record them via Amazon CloudWatch Logs.

_____Answer: D

Explanation:

This solution will meet the requirements because it will use Amazon Inspector to scan the EC2 instances for any new vulnerabilities and generate findings that can be viewed in the Inspector console or sent as notifications via Amazon Simple Notification Service (SNS). It will also use the Amazon CloudWatch Agent to collect and send system logs from the EC2 instances to Amazon CloudWatch Logs, where they can be stored, searched, and analyzed. The system logs can provide an audit trail of all login activities on the instances, as well as other useful information such as performance metrics, errors, and events.

https://docs.aws.amazon.com/inspector/latest/user/what-is-inspector.html

Question: 101

A company uses a series of individual Amazon Cloud Formation templates to deploy its multi-Region Applications. These templates must be deployed in a specific order. The company is making more changes to the templates than previously expected and wants to deploy new templates more efficiently. Additionally, the data engineering team must be notified of all changes to the templates. What should the company do to accomplish these goals?

A. Create an AWS Lambda function to deploy the Cloud Formation templates m the required order Use stack policies to alert the data engineering team.

B. Host the Cloud Formation templates in Amazon S3 Use Amazon S3 events to directly trigger CloudFormation updates and Amazon SNS notifications.

C. Implement CloudFormation StackSets and use drift detection to trigger update alerts to the data engineering team.

D. Leverage CloudFormation nested stacks and stack sets (or deployments Use Amazon SNS to notify the data engineering team.

Answer: D

Explanation:

This solution will meet the requirements because it will use CloudFormation nested stacks and stack sets to deploy the templates more efficiently and consistently across multiple regions. Nested stacks allow the company to separate out common components and reuse templates, while stack sets allow the company to create stacks in multiple accounts and regions with a single template. The company can also use Amazon SNS to send notifications to the data engineering team whenever a change is made to the templates or the stacks. Amazon SNS is a service that allows you to publish messages to subscribers, such as email addresses, phone numbers, or other AWS services. By using Amazon SNS, the company can ensure that the data engineering team is aware of all changes to the templates and can take appropriate actions if needed. What is Amazon SNS? - Amazon Simple Notification Service

Question: 102

A DevOps engineer is deploying a new version of a company's application in an AWS CodeDeploy deployment group associated with its Amazon EC2 instances. After some time, the deployment fails. The engineer realizes that all the events associated with the specific deployment ID are in a Skipped status and code was not deployed in the instances associated with the deployment group.
What are valid reasons for this failure? (Select TWO.).

A. The networking configuration does not allow the EC2 instances to reach the internet via a NAT gateway or internet gateway and the CodeDeploy endpoint cannot be reached.

B. The IAM user who triggered the application deployment does not have permission to interact with the CodeDeploy endpoint.

C. The target EC2 instances were not properly registered with the CodeDeploy endpoint.

D. An instance profile with proper permissions was not attached to the target EC2 instances.

E. The appspec. yml file was not included in the application revision.

Answer: AD

Explanation:

https://docs.aws.amazon.com/codedeploy/latest/userguide/troubleshooting-deployments.html#troubleshooting-skipped-lifecycle-events

Question: 103

A company manages an application that stores logs in Amazon CloudWatch Logs. The company wants to archive the logs to an Amazon S3 bucket Logs are rarely accessed after 90 days and must be retained tor 10 years.
Which combination of steps should a DevOps engineer take to meet these requirements? (Select TWO.)

A. Configure a CloudWatch Logs subscription filter to use AWS Glue to transfer all logs to an S3 bucket.

B. Configure a CloudWatch Logs subscription filter to use Amazon Kinesis Data Firehose to stream all logs to an S3 bucket.

C. Configure a CloudWatch Logs subscription fitter to stream all logs to an S3 bucket.

D. Configure the S3 bucket lifecycle policy to transition logs to S3 Glacier after 90 days and to expire logs after 3.650 days.

E. Configure the S3 bucket lifecycle policy to transition logs to Reduced Redundancy after 90 days and to expire logs after 3.650 days.

Answer: BD

Explanation:

https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/SubscriptionFilters.html

Question: 104

A DevOps engineer needs to configure a blue green deployment for an existing three-tier application. The application runs on Amazon EC2 instances and uses an Amazon RDS database The EC2 instances run behind an Application Load Balancer (ALB) and are in an Auto Scaling group.
The DevOps engineer has created a launch template and an Auto Scaling group for the blue environment. The DevOps engineer also has created a launch template and an Auto Scaling group for the green environment. Each Auto Scaling group deploys to a matching blue or green target group. The target group also specifies which software blue or green gets loaded on the EC2 instances. The ALB can be configured to send traffic to the blue environments target group or the green environments target group. An Amazon Route 53 record for www example com points to the ALB. The deployment must move traffic all at once between the software on the blue environment's EC2 instances to the newly deployed software on the green environments EC2 instances What should the DevOps engineer do to meet these requirements?

A. Start a rolling restart to the Auto Scaling group tor the green environment to deploy the new software on the green environment's EC2 instances When the rolling restart is complete, use an AWS CLI command to update the ALB to send traffic to the green environment's target group.

B. Use an AWS CLI command to update the ALB to send traffic to the green environment's target group. Then start a rolling restart of the Auto Scaling group for the green environment to deploy the new software on the green environment's EC2 instances.

C. Update the launch template to deploy the green environment's software on the blue environment's EC2 instances Keep the target groups and Auto Scaling groups unchanged in both environments Perform a rolling restart of the blue environment's EC2 instances.

D. Start a rolling restart of the Auto Scaling group for the green environment to deploy the new software on the green environment's EC2 instances When the rolling restart is complete, update the Route 53 DNS to point to the green environments endpoint on the ALB.

Answer: A

Explanation:

This solution will meet the requirements because it will use a rolling restart to gradually replace the EC2 instances in the green environment with new instances that have the new software version installed. A rolling restart is a process that terminates and launches instances in batches, ensuring that there is always a minimum number of healthy instances in service. This way, the green environment can be updated without affecting the availability or performance of the application. When the rolling restart is complete, the DevOps engineer can use an AWS CLI command to modify the listener rules of the ALB and change the default action to forward traffic to the green environment's target group. This will switch the traffic from the blue environment to the green environment all at once, as required by the question.

Question: 105

A large enterprise is deploying a web application on AWS. The application runs on Amazon EC2 instances behind an Application Load

Balancer. The instances run in an Auto Scaling group across multiple Availability Zones. The application stores data in an Amazon RDS for Oracle DB instance and Amazon DynamoDB. There are separate environments tor development testing and production. What is the MOST secure and flexible way to obtain password credentials during deployment?

A. Retrieve an access key from an AWS Systems Manager securestring parameter to access AWS services. Retrieve the database credentials from a Systems Manager SecureString parameter.

B. Launch the EC2 instances with an EC2 1AM role to access AWS services Retrieve the database credentials from AWS Secrets Manager.

C. Retrieve an access key from an AWS Systems Manager plaintext parameter to access AWS services. Retrieve the database credentials from a Systems Manager SecureString parameter.

D. Launch the EC2 instances with an EC2 1AM role to access AWS services Store the database passwords in an encrypted config file with the application artifacts.

_____ Answer: B

Explanation:

AWS Secrets Manager is a secrets management service that helps you protect access to your applications, services, and IT resources. This service enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle. Using Secrets Manager, you can secure and manage secrets used to access resources in the AWS Cloud, on third- party services, and on-premises. SSM parameter store and AWS Secret manager are both a secure option. However, Secrets manager is more flexible and has more options like password generation. Reference: https://www.1strategy.com/blog/2019/02/28/aws-parameter-store-vs-aws-secrets-manager/

Question: 106

A DevOps engineer is designing an application that integrates with a legacy REST API. The application has an AWS Lambda function that reads records from an Amazon Kinesis data stream. The Lambda function sends the records to the legacy REST API.
Approximately 10% of the records that the Lambda function sends from the Kinesis data stream have data errors and must be processed manually. The Lambda function event source configuration has an Amazon Simple Queue Service (Amazon SQS) dead-letter queue as an on-failure destination. The DevOps engineer has configured the Lambda function to process records in batches and has implemented retries in case of failure.
During testing the DevOps engineer notices that the dead-letter queue contains many records that have no data errors and that already have been processed by the legacy REST API. The DevOps engineer needs to configure the Lambda function's event source options to reduce the number of errorless records that are sent to the dead-letter queue.
Which solution will meet these requirements?

A. Increase the retry attempts
B. Configure the setting to split the batch when an error occurs
C. Increase the concurrent batches per shard
D. Decrease the maximum age of record

_____ Answer: B

Explanation:

This solution will meet the requirements because it will reduce the number of errorless records that are sent to the dead-letter queue. When you configure the setting to split the batch when an error occurs, Lambda will retry only the records that caused the error, instead of retrying the entire batch. This way, the records that have no data errors and have already been processed by the legacy REST API will

not be retried and sent to the dead-letter queue unnecessarily. https://docs.aws.amazon.com/lambda/latest/dg/with-kinesis.html

Question: 107

A company is storing 100 GB of log data in csv format in an Amazon S3 bucket SQL developers want to query this data and generate graphs to visualize it. The SQL developers also need an efficient automated way to store metadata from the csv file. Which combination of steps will meet these requirements with the LEAST amount of effort? (Select THREE.)

A. Fitter the data through AWS X-Ray to visualize the data.
B. Filter the data through Amazon QuickSight to visualize the data.
C. Query the data with Amazon Athena.
D. Query the data with Amazon Redshift.
E. Use the AWS Glue Data Catalog as the persistent metadata store.
F. Use Amazon DynamoDB as the persistent metadata store.

Answer: BCE

Explanation:

https://docs.aws.amazon.com/glue/latest/dg/components-overview.html

Question: 108

A company manages AWS accounts for application teams in AWS Control Tower. Individual application teams are responsible for securing their respective AWS accounts.
A DevOps engineer needs to enable Amazon GuardDuty for all AWS accounts in which the application teams have not already enabled GuardDuty. The DevOps engineer is using AWS CloudFormation StackSets from the AWS Control Tower management account.
How should the DevOps engineer configure the CloudFormation template to prevent failure during the StackSets deployment?

A. Create a CloudFormation custom resource that invokes an AWS Lambda function. Configure the Lambda function to conditionally enable GuardDuty if GuardDuty is not already enabled in the accounts.
B. Use the Conditions section of the CloudFormation template to enable GuardDuty in accounts where GuardDuty is not already enabled.
C. Use the CloudFormation Fn. GetAtt intrinsic function to check whether GuardDuty is already enabled If GuardDuty is not already enabled use the Resources section of the CloudFormation template to enable GuardDuty.
D. Manually discover the list of AWS account IDs where GuardDuty is not enabled Use the CloudFormation Fn: ImportValue intrinsic function to import the list of account IDs into the CloudFormation template to skip deployment for the listed AWS accounts.

Answer: A

Explanation:

This solution will meet the requirements because it will use a CloudFormation custom resource to execute custom logic during the stack set operation. A custom resource is a resource that you define in your template and that is associated with an AWS Lambda function. The Lambda function runs whenever the custom resource is created, updated, or deleted, and can perform any actions that are supported by the AWS SDK. In this case, the Lambda function can use the GuardDuty API to check whether GuardDuty is already enabled in each target account, and if not, enable it. This way, the DevOps engineer can avoid deploying the stack set to accounts that already have GuardDuty

enabled, and prevent failure during the deployment.

Question: 109

A company has 20 service learns Each service team is responsible for its own microservice. Each service team uses a separate AWS account for its microservice and a VPC with the 192 168 0 0/22 CIDR block. The company manages the AWS accounts with AWS Organizations.

Each service team hosts its microservice on multiple Amazon EC2 instances behind an Application Load Balancer. The microservices communicate with each other across the public internet. The company's security team has issued a new guideline that all communication between microservices must use HTTPS over private network connections and cannot traverse the public internet.

A DevOps engineer must implement a solution that fulfills these obligations and minimizes the number of changes for each service team

Which solution will meet these requirements?

A. Create a new AWS account in AWS Organizations Create a VPC in this account and use AWS Resource Access Manager to share the private subnets of this VPC with the organization Instruct the service teams to launch a new. Network Load Balancer (NLB) and EC2 instances that use the shared private subnets Use the NLB DNS names for communication between microservices.

B. Create a Network Load Balancer (NLB) in each of the microservice VPCs Use AWS PrivateLink to create VPC endpoints in each AWS account for the NLBs Create subscriptions to each VPC endpoint in each of the other AWS accounts Use the VPC endpoint DNS names for communication between microservices.

C. Create a Network Load Balancer (NLB) in each of the microservice VPCs Create VPC peering connections between each of the microservice VPCs Update the route tables for each VPC to use the peering links Use the NLB DNS names for communication between microservices.

D. Create a new AWS account in AWS Organizations Create a transit gateway in this account and use AWS Resource Access Manager to share the transit gateway with the organization. In each of the microservice VPCs. create a transit gateway attachment to the shared transit gateway Update the route tables of each VPC to use the transit gateway Create a Network Load Balancer (NLB) in each of the microservice VPCs Use the NLB DNS names for communication between microservices.

____ Answer: B

Explanation:

https://aws.amazon.com/blogs/networking-and-content-delivery/connecting-networks-with- overlapping-ip-ranges/ Private link is the best option because Transit Gateway doesn't support overlapping CIDR ranges.

Question: 110

A company is building a new pipeline by using AWS CodePipeline and AWS CodeBuild in a build account. The pipeline consists of two stages. The first stage is a CodeBuild job to build and package an AWS Lambda function. The second stage consists of deployment actions that operate on two different AWS accounts a development environment account and a production environment account. The deployment stages use the AWS Cloud Format ion action that CodePipeline invokes to deploy the infrastructure that the Lambda function requires.

A DevOps engineer creates the CodePipeline pipeline and configures the pipeline to encrypt build artifacts by using the AWS Key Management Service (AWS KMS) AWS managed key for Amazon S3 (the aws/s3 key). The artifacts are stored in an S3 bucket When the pipeline runs, the Cloud Formation actions fail with an access denied error.

Which combination of actions must the DevOps engineer perform to resolve this error? (Select TWO.)

A. Create an S3 bucket in each AWS account for the artifacts Allow the pipeline to write to the S3 buckets. Create a CodePipeline S3

action to copy the artifacts to the S3 bucket in each AWS account Update the CloudFormation actions to reference the artifacts S3 bucket in the production account.

B. Create a customer managed KMS key Configure the KMS key policy to allow the IAM roles used by the CloudFormation action to perform decrypt operations Modify the pipeline to use the customer managed KMS key to encrypt artifacts.

C. Create an AWS managed KMS key Configure the KMS key policy to allow the development account and the production account to perform decrypt operations. Modify the pipeline to use the KMS key to encrypt artifacts.

D. In the development account and in the production account create an IAM role for CodePipeline. Configure the roles with permissions to perform CloudFormation operations and with permissions to retrieve and decrypt objects from the artifacts S3 bucket. In the CodePipeline account configure the CodePipeline CloudFormation action to use the roles.

E. In the development account and in the production account create an IAM role for CodePipeline Configure the roles with permissions to perform CloudFormation
operations and with permissions to retrieve and decrypt objects from the artifacts S3 bucket. In the CodePipelme account modify the artifacts S3 bucket policy to allow the roles access Configure the CodePipeline CloudFormation action to use the roles.

Answer: BE

Explanation:

Question: 111

A company has many AWS accounts. During AWS account creation the company uses automation to create an Amazon CloudWatch Logs log group in every AWS Region that the company operates in. The automaton configures new resources in the accounts to publish logs to the provisioned log groups in their Region.

The company has created a logging account to centralize the logging from all the other accounts. A DevOps engineer needs to aggregate the log groups from all the accounts to an existing Amazon S3 bucket in the logging account.

Which solution will meet these requirements in the MOST operationally efficient manner?

A. In the logging account create a CloudWatch Logs destination with a destination policy. For each new account subscribe the CloudWatch Logs log groups to the. Destination Configure a single Amazon Kinesis data stream and a single Amazon Kinesis Data Firehose delivery stream to deliver the logs from the CloudWatch Logs destination to the S3 bucket.

B. In the logging account create a CloudWatch Logs destination with a destination policy for each Region. For each new account subscribe the CloudWatch Logs log groups to the destination. Configure a single Amazon Kinesis data stream and a single Amazon Kinesis Data Firehose delivery stream to deliver the logs from all the CloudWatch Logs destinations to the S3 bucket.

C. In the logging account create a CloudWatch Logs destination with a destination policy for each Region. For each new account subscribe the CloudWatch Logs log groups to the destination Configure an Amazon Kinesis data stream and an Amazon Kinesis Data Firehose delivery stream for each Region to deliver the logs from the CloudWatch Logs destinations to the S3 bucket.

D. In the logging account create a CloudWatch Logs destination with a destination policy. For each new account subscribe the CloudWatch Logs log groups to the destination. Configure a single Amazon Kinesis data stream to deliver the logs from the CloudWatch Logs destination to the S3
bucket.

Answer: C

Explanation:

This solution will meet the requirements in the most operationally efficient manner because it will use CloudWatch Logs destination to aggregate the log groups from all the accounts to a single S3 bucket in the logging account. However, unlike option A, this solution will create a CloudWatch Logs destination for each region, instead of a single destination for all regions. This will improve the performance and reliability of the log delivery, as it will avoid cross-region data transfer and latency issues. Moreover, this solution will use an Amazon Kinesis data stream and an Amazon Kinesis Data Firehose delivery stream for each region, instead of a single stream for all regions. This

www.atmicnetworks.com

will also improve the scalability and throughput of the log delivery, as it will avoid bottlenecks and throttling issues that may occur with a single stream.

Question: 112

A company uses AWS CodeArtifact to centrally store Python packages. The CodeArtifact repository is configured with the following repository policy.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "codeartifact:DescribePackageVersion",
                "codeartifact:DescribeRepository",
                "codeartifact:GetPackageVersionReadme",
                "codeartifact:GetRepositoryEndpoint",
                "codeartifact:ListPackageVersionAssets",
                "codeartifact:ListPackageVersionDependencies",
                "codeartifact:ListPackageVersions",
                "codeartifact:ListPackages",
                "codeartifact:ReadFromRepository"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Principal": "*",
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalOrgID": [
                        "o-xxxxxxxxxx"
                    ]
                }
            }
        }
    ]
}
```

A development team is building a new project in an account that is in an organization in AWS Organizations. The development team wants to use a Python library that has already been stored in the CodeArtifact repository in the organization. The development team uses AWS CodePipeline and AWS CodeBuild to build the new application. The CodeBuild job that the development team uses to build the application is configured to run in a VPC Because of compliance requirements the VPC has no internet connectivity.
The development team creates the VPC endpoints for CodeArtifact and updates the CodeBuild buildspec yaml file. However, the development team cannot download the Python library from the repository.
Which combination of steps should a DevOps engineer take so that the development team can use Code Artifact? (Select TWO.)

A. Create an Amazon S3 gateway endpoint Update the route tables for the subnets that are running the CodeBuild job.
B. Update the repository policy's Principal statement to include the ARN of the role that the CodeBuild project uses.
C. Share the CodeArtifact repository with the organization by using AWS Resource Access Manager (AWS RAM).
D. Update the role that the CodeBuild project uses so that the role has sufficient permissions to use the CodeArtifact repository.

E. Specify the account that hosts the repository as the delegated administrator for CodeArtifact in the organization.

Answer: AD

Explanation:

"AWS CodeArtifact operates in multiple Availability Zones and stores artifact data and metadata in Amazon S3 and Amazon DynamoDB. Your encrypted data is redundantly stored across multiple facilities and multiple devices in each facility, making it highly available and highly durable." https://aws.amazon.com/codeartifact/features/ With no internet connectivity, a gateway endpoint becomes necessary to access S3.

Question: 113

A company is deploying a new application that uses Amazon EC2 instances. The company needs a solution to query application logs and AWS account API activity Which solution will meet these requirements?

A. Use the Amazon CloudWatch agent to send logs from the EC2 instances to Amazon CloudWatch Logs Configure AWS CloudTrail to deliver the API logs to Amazon S3 Use CloudWatch to query both sets of logs.
B. Use the Amazon CloudWatch agent to send logs from the EC2 instances to Amazon CloudWatch Logs Configure AWS CloudTrail to deliver the API logs to CloudWatch Logs Use CloudWatch Logs Insights to query both sets of logs.
C. Use the Amazon CloudWatch agent to send logs from the EC2 instances to Amazon Kinesis Configure AWS CloudTrail to deliver the API logs to Kinesis Use Kinesis to load the data into Amazon Redshift Use Amazon Redshift to query both sets of logs.
D. Use the Amazon CloudWatch agent to send logs from the EC2 instances to Amazon S3 Use AWS CloudTrail to deliver the API togs to Amazon S3 Use Amazon Athena to query both sets of logs in Amazon S3.

_____ Answer: D

Explanation:

This solution will meet the requirements because it will use Amazon S3 as a common data lake for both the application logs and the API logs. Amazon S3 is a service that provides scalable, durable, and secure object storage for any type of data. You can use the Amazon CloudWatch agent to send logs from your EC2 instances to S3 buckets, and use AWS CloudTrail to deliver the API logs to S3 buckets as well. You can also use Amazon Athena to query both sets of logs in S3 using standard SQL, without loading or transforming them. Athena is a serverless interactive query service that allows you to analyze data in S3 using a variety of data formats, such as JSON, CSV, Parquet, and ORC.

Question: 114

A company wants to use a grid system for a proprietary enterprise m-memory data store on top of AWS. This system can run in multiple server nodes in any Linux-based distribution. The system must be able to reconfigure the entire cluster every time a node is added or removed. When adding or removing nodes an /etc./cluster/nodes config file must be updated listing the IP addresses of the current node members of that cluster.
The company wants to automate the task of adding new nodes to a cluster.
What can a DevOps engineer do to meet these requirements?

A. Use AWS OpsWorks Stacks to layer the server nodes of that cluster. Create a Chef recipe that populates the content of the 'etc./cluster/nodes config file and restarts the service by using the current members of the layer. Assign that recipe to the Configure lifecycle event.

B. Put the file nodes config in version control. Create an AWS CodeDeploy deployment configuration and deployment group based on an Amazon EC2 tag value for thecluster nodes. When adding a new node to the cluster update the file with all tagged instances and make a commit in version control. Deploy the new file and restart the services.

C. Create an Amazon S3 bucket and upload a version of the /etc./cluster/nodes config file Create a crontab script that will poll for that S3 file and download it frequently. Use a process manager such as Monit or system, to restart the cluster services when it detects that the new file was modified. When adding a node to the cluster edit the file's most recent members Upload the new file to the S3 bucket.

D. Create a user data script that lists all members of the current security group of the cluster and automatically updates the /etc/cluster/. nodes config. Tile whenever a new instance is added to the cluster.

_____ Answer: A

Explanation:

You can run custom recipes manually, but the best approach is usually to have AWS OpsWorks Stacks run them automatically. Every layer has a set of built-in recipes assigned each of five lifecycle events— Setup, Configure, Deploy, Undeploy, and Shutdown. Each time an event occurs for an instance, AWS OpsWorks Stacks runs the associated recipes for each of the instance's layers, which handle the corresponding tasks. For example, when an instance finishes booting, AWS OpsWorks Stacks triggers a Setup event. This event runs the associated layer's Setup recipes, which typically handle tasks such as installing and configuring packages

Question: 115

A company is testing a web application that runs on Amazon EC2 instances behind an Application Load Balancer. The instances run in an Auto Scaling group across multiple Availability Zones. The company uses a blue green deployment process with immutable instances when deploying new software.

During testing users are being automatically logged out of the application at random times. Testers also report that when a new version of the application is deployed all users are logged out. The development team needs a solution to ensure users remain logged m across scaling events and application deployments.

What is the MOST operationally efficient way to ensure users remain logged in?

A. Enable smart sessions on the load balancer and modify the application to check tor an existing session.

B. Enable session sharing on the toad balancer and modify the application to read from the session store.

C. Store user session information in an Amazon S3 bucket and modify the application to read session information from the bucket.

D. Modify the application to store user session information in an Amazon ElastiCache cluster.

_____ Answer: D

Explanation:

https://aws.amazon.com/caching/session-management/

Question: 116

A company's DevOps engineer is creating an AWS Lambda function to process notifications from an Amazon Simple Notification Service (Amazon SNS) topic. The Lambda function will process the notification messages and will write the contents of the notification messages to an Amazon RDS Multi-AZ DB instance.

During testing a database administrator accidentally shut down the DB instance. While the database was down the company lost several of the SNS notification messages that were delivered during that time.

www.atmicnetworks.com

The DevOps engineer needs to prevent the loss of notification messages in the future
Which solutions will meet this requirement? (Select TWO.)

A. Replace the RDS Multi-AZ DB instance with an Amazon DynamoDB table.

B. Configure an Amazon Simple Queue Service (Amazon SQS) queue as a destination of the Lambda function.

C. Configure an Amazon Simple Queue Service (Amazon SQS> dead-letter queue for the SNS topic.

D. Subscribe an Amazon Simple Queue Service (Amazon SQS) queue to the SNS topic Configure the Lambda function to process messages from the SQS queue.

E. Replace the SNS topic with an Amazon EventBridge event bus Configure an EventBridge rule on the new event bus to invoke the Lambda function for each event.

Answer: CD

Explanation:

These solutions will meet the requirement because they will prevent the loss of notification messages in the future. An Amazon SQS queue is a service that provides a reliable, scalable, and secure message queue for asynchronous communication between distributed components. You can use an SQS queue to buffer messages from an SNS topic and ensure that they are delivered and processed by a Lambda function, even if the function or the database is temporarily unavailable. Option C will configure an SQS dead-letter queue for the SNS topic. A dead-letter queue is a queue that receives messages that could not be delivered to any subscriber after a specified number of retries. You can use a dead-letter queue to store and analyze failed messages, or to reprocess them later. This way, you can avoid losing messages that could not be delivered to the Lambda function due to network errors, throttling, or other issues. Option D will subscribe an SQS queue to the SNS topic and configure the Lambda function to process messages from the SQS queue. This will decouple the SNS topic from the Lambda function and provide more flexibility and control over the message delivery and processing. You can use an SQS queue to store messages from the SNS topic until they are ready to be processed by the Lambda function, and also to retry processing in case of failures. This way, you can avoid losing messages that could not be processed by the Lambda function due to database errors, timeouts, or other issues.

Question: 117

A media company has several thousand Amazon EC2 instances in an AWS account. The company is using Slack and a shared email inbox for team communications and important updates. A DevOps engineer needs to send all AWS-scheduled EC2 maintenance notifications to the Slack channel and the shared inbox. The solution must include the instances' Name and Owner tags.
Which solution will meet these requirements?

A. Integrate AWS Trusted Advisor with AWS Config Configure a custom AWS Config rule to invoke an AWS Lambda function to publish notifications to an Amazon Simple Notification Service (Amazon SNS) topic Subscribe a Slack channel endpoint and the shared inbox to the topic.

B. Use Amazon EventBridge to monitor for AWS Health Events Configure the maintenance events to target an Amazon Simple Notification Service (Amazon SNS) topic Subscribe an AWS Lambda function to the SNS topic to send notifications to the Slack channel and the shared inbox.

C. Create an AWS Lambda function that sends EC2 maintenance notifications to the Slack channel and the shared inbox Monitor EC2 health events by using Amazon CloudWatch metrics Configure a CloudWatch alarm that invokes the Lambda function when a maintenance notification is received.

D. Configure AWS Support integration with AWS CloudTrail Create a CloudTrail lookup event to invoke an AWS Lambda function to pass EC2 maintenance notifications to Amazon Simple Notification Service (Amazon SNS) Configure Amazon SNS to target the Slack channel and the shared inbox.

Answer: B

Explanation:

https://docs.aws.amazon.com/health/latest/ug/cloudwatch-events-health.html

Question: 118

A company's DevOps engineer is working in a multi-account environment. The company uses AWS Transit Gateway to route all outbound traffic through a network operations account. In the network operations account all account traffic passes through a firewall appliance for inspection before the traffic goes to an internet gateway.

The firewall appliance sends logs to Amazon CloudWatch Logs and includes event seventies of CRITICAL, HIGH, MEDIUM, LOW, and INFO. The security team wants to receive an alert if any CRITICAL events occur.

What should the DevOps engineer do to meet these requirements?

A. Create an Amazon CloudWatch Synthetics canary to monitor the firewall state. If the firewall reaches a CRITICAL state or logs a CRITICAL event use a CloudWatch alarm to publish a notification to an Amazon Simple Notification Service (Amazon SNS) topic Subscribe the security team's email address to the topic.

B. Create an Amazon CloudWatch metric filter by using a search for CRITICAL events Publish a custom metric for the finding. Use a CloudWatch alarm based on the custom metric to publish a notification to an Amazon Simple Notification Service (Amazon SNS) topic. Subscribe the security team's email address to the topic.

C. Enable Amazon GuardDuty in the network operations account. Configure GuardDuty to monitor flow logs Create an Amazon EventBridge event rule that is invoked by GuardDuty events that are CRITICAL Define an Amazon Simple Notification Service (Amazon SNS) topic as a target Subscribe the security team's email address to the topic.

D. Use AWS Firewall Manager to apply consistent policies across all accounts. Create an Amazon. EventBridge event rule that is invoked by Firewall Manager events that are CRITICAL Define an Amazon Simple Notification Service (Amazon SNS) topic as a target Subscribe the security team's email address to the topic.

Answer: B

Explanation:

"The firewall appliance sends logs to Amazon CloudWatch Logs and includes event severities of CRITICAL, HIGH, MEDIUM, LOW, and INFO"

Question: 119

An application running on a set of Amazon EC2 instances in an Auto Scaling group requires a configuration file to operate. The instances are created and maintained with AWS CloudFormation. A DevOps engineer wants the instances to have the latest configuration file when launched and wants changes to the configuration file to be reflected on all the instances with a minimal delay when the CloudFormation template is updated. Company policy requires that application configuration files be maintained along with AWS infrastructure configuration files m source control.

Which solution will accomplish this?

A. In the CloudFormaiion template add an AWS Config rule. Place the configuration file content in the rule's InputParameters property and set the Scope property to the EC2 Auto Scaling group. Add an AWS Systems Manager Resource Data Sync resource to the template to poll for updates to the configuration.

B. In the CloudFormation template add an EC2 launch template resource. Place the configuration file content in the launch template. Configure the cfn-mit script to run when the instance is launched and configure the cfn-hup script to poll for updates to the configuration.

C. In the CloudFormation template add an EC2 launch template resource. Place the configuration file content in the launch template. Add an AWS Systems Manager Resource Data Sync resource to the template to poll for updates to the configuration.

D. In the CloudFormation template add CloudFormation imt metadata. Place the configuration file content m the metadata. Configure the cfn-init script to run when the instance is launched and configure the cfn-hup script to poll for updates to the configuration.

_____ Answer: D

Explanation:

Use the AWS::CloudFormation::Init type to include metadata on an Amazon EC2 instance for the cfn- init helper script. If your template calls the cfn-init script, the script looks for resource metadata rooted in the AWS::CloudFormation::Init metadata key. Reference: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-resource-init.html

Question: 120

A company updated the AWS Cloud Formation template for a critical business application. The stack update process failed due to an error in the updated template and AWS CloudFormation automatically began the stack rollback process Later a DevOps engineer discovered that the application was still unavailable and that the stack was in the UPDATE_ROLLBACK_FAILED state. Which combination of actions should the DevOps engineer perform so that the stack rollback can complete successfully? (Select TWO.)

A. Attach the AWSC loud Formation FullAccess IAM policy to the AWS CtoudFormation role.
B. Automatically recover the stack resources by using AWS CloudFormation drift detection.
C. Issue a ContinueUpdateRollback command from the AWS CloudFormation console or the AWS CLI.
D. Manually adjust the resources to match the expectations of the stack.
E. Update the existing AWS CloudFormation stack by using the original template.

Answer: CD

Explanation:

https://docs.aws.amazon.com/cli/latest/reference/cloudformation/continue-update-rollback.html For a specified stack that is in the UPDATE_ROLLBACK_FAILED state, continues rolling it back to the UPDATE_ROLLBACK_COMPLETE state. Depending on the cause of the failure, you can manually fix the error and continue the rollback. By continuing the rollback, you can return your stack to a working state (the UPDATE_ROLLBACK_COMPLETE state), and then try to update the stack again.

Question: 121 A company's production environment uses an AWS CodeDeploy blue/green deployment to deploy an application. The deployment incudes Amazon EC2 Auto Scaling groups that launch instances that run Amazon Linux 2.

A working appspec. ymi file exists in the code repository and contains the following text.

f l s z

369 c xnjk* x on z /V9Z/www/ntntl/ 6pcxxc6*xon

A DevOps engineer needs to ensure that a script downloads and installs a license file onto the instances before the replacement instances start to handle request traffic. The DevOps engineer adds a hooks section to the appspec. yml file.

Which hook should the DevOps engineer use to run the script that downloads and installs the license file?

A. AfterBlockTraffic

B. BeforeBlockTraffic

C. BeforeInstall

D. Down load Bundle

Answer: C

Explanation:

This hook runs before the new application version is installed on the replacement instances. This is the best place to run the script because it ensures that the license file is downloaded and installed before the replacement instances start to handle request traffic. If you use any other hook, you may encounter errors or inconsistencies in your application.

Question: 122

A company runs applications in AWS accounts that are in an organization in AWS Organizations The applications use Amazon EC2 instances and Amazon S3.

The company wants to detect potentially compromised EC2 instances suspicious network activity and unusual API activity in its existing AWS accounts and in any AWS accounts that the company creates in the future When the company detects one to these events the company wants to use an existing Amazon Simple Notification Service (Amazon SNS) topic to send a notification to its operational support team for investigation and remediation.

Which solution will meet these requirements in accordance with AWS best practices?

A. In the organization's management account configure an AWS account as the Amazon GuardDuty administrator account. In the GuardDuty administrator account add the company's existing AWS accounts to GuardDuty as members In the GuardDuty administrator account create an Amazon EventBridge rule with an event pattern to match GuardDuty events and to forward matching events to the SNS topic.

B. In the organization's management account configure Amazon GuardDuty to add newly created

AWS accounts by invitation and to send invitations to the existing AWS accounts Create an AWS Cloud Formation stack set that accepts the GuardDuty invitation and creates an Amazon EventBridge rule Configure the rule with an event pattern to match. GuardDuty events and to forward matching events to the SNS topic. Configure the Cloud Formation stack set to deploy into all AWS accounts in the organization.

C. In the organization's management account. create an AWS CloudTrail organization trail Activate the organization trail in all AWS accounts in the organization. Create an SCP that enables VPC Flow Logs in each account in the organization. Configure AWS Security Hub for the organization Create an Amazon EventBridge rule with an even pattern to match Security Hub events and to forward matching events to the SNS topic.

D. In the organization's management account configure an AWS account as the AWS CloudTrail administrator account in the CloudTrail administrator account create a CloudTrail organization trail. Add the company's existing AWS accounts to the organization trail Create an SCP that enables VPC Flow Logs in each account in the organization. Configure AWS Security Hub for the organization. Create an Amazon EventBridge rule with an event pattern to match Security Hub events and to forward matching events to the SNS topic.

Answer: B

Explanation:

It allows the company to detect potentially compromised EC2 instances, suspicious network activity, and unusual API activity in its existing AWS accounts and in any AWS accounts that the company creates in the future using Amazon GuardDuty. It also provides a solution for automatically adding future AWS accounts to GuardDuty by configuring GuardDuty to add newly created AWS accounts by invitation and to send invitations to the existing AWS accounts.

www.atmicnetworks.com

Question: 123

An application runs on Amazon EC2 instances behind an Application Load Balancer (ALB). A DevOps engineer is using AWS CodeDeploy to release a new version. The deployment fails during the AllowTraffic lifecycle event, but a cause for the failure is not indicated in the deployment logs. What would cause this?

A. The appspec. yml file contains an invalid script that runs in the AllowTraffic lifecycle hook.
B. The user who initiated the deployment does not have the necessary permissions to interact with the ALB.
C. The health checks specified for the ALB target group are misconfigured.
D. The CodeDeploy agent was not installed in the EC2 instances that are pad of the ALB target group.

Answer: C

Explanation:

This failure is typically due to incorrectly configured health checks in Elastic Load Balancing for the Classic Load Balancer, Application Load Balancer, or Network Load Balancer used to manage traffic for the deployment group. To resolve the issue, review and correct any errors in the health check configuration for the load balancer.
https://docs.aws.amazon.com/codedeploy/latest/userguide/troubleshooting-deployments.html#troubleshooting-deployments-allowtraffic-no-logs

Question: 124

An AWS CodePipeline pipeline has implemented a code release process. The pipeline is integrated with AWS CodeDeploy to deploy versions of an application to multiple Amazon EC2 instances for each CodePipeline stage.
During a recent deployment the pipeline failed due to a CodeDeploy issue. The DevOps team wants to improve monitoring and notifications during deployment to decrease resolution times.
What should the DevOps engineer do to create notifications. When issues are discovered?

A. Implement Amazon CloudWatch Logs for CodePipeline and CodeDeploy create an AWS Config rule to evaluate code deployment issues, and create an Amazon Simple Notification Service (Amazon SNS) topic to notify stakeholders of deployment issues.
B. Implement Amazon EventBridge for CodePipeline and CodeDeploy create an AWS Lambda function to evaluate code deployment issues, and create an Amazon Simple Notification Service (Amazon SNS) topic to notify stakeholders of deployment issues.
C. Implement AWS CloudTrail to record CodePipeline and CodeDeploy API call information create an AWS Lambda function to evaluate code deployment issues and create an Amazon Simple Notification Service (Amazon SNS) topic to notify stakeholders of deployment issues.
D. Implement Amazon EventBridge for CodePipeline and CodeDeploy create an Amazon. Inspector assessment target to evaluate code deployment issues and create an Amazon Simple. Notification Service (Amazon SNS) topic to notify stakeholders of deployment issues.

Answer: B

Explanation:

AWS CloudWatch Events can be used to monitor events across different AWS resources, and a CloudWatch Event Rule can be created to trigger an AWS Lambda function when a deployment issue is detected in the pipeline. The Lambda function can then evaluate the issue and send a notification to the appropriate stakeholders through an Amazon SNS topic. This approach allows for real-time notifications and faster resolution times.

www.atmicnetworks.com

Question: 125

A Company uses AWS CodeCommit for source code control. Developers apply their changes to various feature branches and create pull requests to move those changes to the main branch when the changes are ready for production.
The developers should not be able to push changes directly to the main branch. The company applied the AWSCodeCommitPowerUser managed policy to the developers' IAM role, and now these developers can push changes to the main branch directly on every repository in the AWS account. What should the company do to restrict the developers' ability to push changes to the main branch directly?

A. Create an additional policy to include a Deny rule for the GitPush and PutFile actions. Include a restriction for the specific restriction for the specific repositories in the policy repositories in the policy statement with a condition that references the main branch.

A Create an additional policy to include a Deny rule for the GitPush and PutFile actions Include a restriction for the specific repositories in the policy statement with a condition that references the main branch
B. Remove the IAM policy, and add an AWSCodeCommitReadOnly managed policy. Add an Allow rule for the GitPush and PutFile actions for the specific repositories in the policy statement with a condition that references the mam branch.
C. Modify the IAM policy Include a Deny rule for the GitPush and PutFile actions for the specific repositories in the policy statement with a condition that references the main branch.
D. Create an additional policy to include an Allow rule for the GitPush and PutFile actions. Include a restriction for the specific repositories in the policy statement with a condition that references the feature branches.

_____ Answer: A

Explanation:

By default, the AWSCodeCommitPowerUser managed policy allows users to push changes to any branch in any repository in the AWS account. To restrict the developers' ability to push changes to the main branch directly, an additional policy is needed that explicitly denies these actions for the main branch.

The Deny rule should be included in a policy statement that targets the specific repositories and includes a condition that references the main branch. The policy statement should look something like this:

```
{
"Effect": "Deny",
"Action": [
"codecommit:GitPush", "codecommit:PutFile"
],
"Resource": "arn:aws:codecommit:<region>:<account-id>:<repository-name>", "Condition": {
"StringEqualsIfExists": { "codecommit:References": [ "refs/heads/main"
] } }
```

Question: 126

A company deploys updates to its Amazon API Gateway API several times a week by using an AWS CodePipeline pipeline. As part of the update process the company exports the JavaScript SDK for the API from the API. Gateway console and uploads the SDK to an Amazon S3 bucket
The company has configured an Amazon CloudFront distribution that uses the S3 bucket as an origin Web client then download the SDK by using the CloudFront distribution's endpoint. A DevOps engineer needs to implement a solution to make the new SDK available automatically during new API deployments.

www.atmicnetworks.com

Which solution will meet these requirements?

A. Create a CodePipeline action immediately after the deployment stage of the API. Configure the action to invoke an AWS Lambda function. Configure the Lambda function to download the SDK from API Gateway, upload the SDK to the S3 bucket and create a CloudFront invalidation for the SDK path.
B. Create a CodePipeline action immediately after the deployment stage of the API Configure the action to use the CodePipelme integration with API. Gateway to export the SDK to Amazon S3 Create another action that uses the CodePipeline integration with Amazon S3 to invalidate the cache for the SDK path.
C. Create an Amazon EventBridge rule that reacts to UpdateStage events from aws apigateway Configure the rule to invoke an AWS Lambda function to download the SDK from API Gateway upload the SDK to the S3 bucket and call the CloudFront API to create an invalidation for the SDK path.
D. Create an Amazon EventBridge rule that reacts to Create. Deployment events from aws apigateway. Configure the rule to invoke an AWS Lambda function to download the SDK from API. Gateway upload the SDK to the S3 bucket and call the S3 API to invalidate the cache for the SDK path.

Answer: A

Explanation:

This solution would allow the company to automate the process of updating the SDK and making it available to web clients. By adding a CodePipeline action immediately after the deployment stage of the API, the Lambda function will be invoked automatically each time the API is updated. The Lambda function should be able to download the new SDK from API Gateway, upload it to the S3 bucket and also create a CloudFront invalidation for the SDK path so that the latest version of the SDK is available for the web clients. This is the most straight forward solution and it will meet the requirements.

Question: 127

A DevOps engineer manages a large commercial website that runs on Amazon EC2. The website uses Amazon Kinesis Data Streams to collect and process web togs. The DevOps engineer manages the Kinesis consumer application, which also runs on Amazon EC2.
Sudden increases of data cause the Kinesis consumer application to (all behind and the Kinesis data streams drop records before the records can be processed. The DevOps engineer must implement a solution to improve stream handling.
Which solution meets these requirements with the MOST operational efficiency?

A. Modify the Kinesis consumer application to store the logs durably in Amazon S3 Use Amazon EMR to process the data directly on Amazon S3 to derive customer insights Store the results in Amazon S3.
B. Horizontally scale the Kinesis consumer application by adding more EC2 instances based on the Amazon CloudWatch GetRecords IteratorAgeMilliseconds metric Increase the retention period of the Kinesis data streams.
C. Convert the Kinesis consumer application to run as an AWS Lambda function. Configure the Kinesis data streams as the event source for the Lambda function to process the data streams
D. Increase the number of shards in the Kinesis data streams to increase the overall throughput so that the consumer application processes the data faster.

Answer: B

Explanation:

https://docs.aws.amazon.com/streams/latest/dev/monitoring-with-cloudwatch.html
GetRecords.IteratorAgeMilliseconds - The age of the last record in all GetRecords calls made against a Kinesis stream, measured over the specified time period. Age is the difference between the current time and when the last record of the GetRecords call was written to the stream. The Minimum and Maximum statistics can be used to track the progress of Kinesis consumer applications. A value of zero

indicates that the records being read are completely caught up.

Question: 128

A business has an application that consists of five independent AWS Lambda functions.
The DevOps engineer has built a CI/CD pipeline using AWS CodePipeline and AWS CodeBuild that builds tests packages and deploys each Lambda function in sequence. The pipeline uses an Amazon EventBridge rule to ensure the pipeline starts as quickly as possible after a change is made to the application source code.
After working with the pipeline for a few months the DevOps engineer has noticed the pipeline takes too long to complete.
What should the DevOps engineer implement to BEST improve the speed of the pipeline?

A. Modify the CodeBuild projects within the pipeline to use a compute type with more available network throughput.
B. Create a custom CodeBuild execution environment that includes a symmetric multiprocessing configuration to run the builds in parallel.
C. Modify the CodePipeline configuration to run actions for each Lambda function in parallel by specifying the same runorder.
D. Modify each CodeBuild protect to run within a VPC and use dedicated instances to increase throughput.

_____ Answer: C

Explanation:

https://docs.aws.amazon.com/codepipeline/latest/userguide/reference-pipeline-structure.html AWS doc: "To specify parallel actions, use the same integer for each action you want to run in parallel. For example, if you want three actions to run in sequence in a stage, you would give the first action the runOrder value of 1, the second action the runOrder value of 2, and the third the runOrder value of 3. However, if you want the second and third actions to run in parallel, you would give the first action the runOrder value of 1 and both the second and third actions the runOrder value of 2."

Question: 129

A company has developed an AWS Lambda function that handles orders received through an API. The company is using AWS CodeDeploy to deploy the Lambda function as the final stage of a CI/CD pipeline.
A DevOps engineer has noticed there are intermittent failures of the ordering API for a few seconds after deployment. After some investigation the DevOps engineer believes the failures are due to database changes not having fully propagated before the Lambda function is invoked How should the DevOps engineer overcome this?

A. Add a BeforeAllowTraffic hook to the AppSpec file that tests and waits for any necessary database changes before traffic can flow to the new version of the Lambda function.
B. Add an AfterAllowTraffic hook to the AppSpec file that forces traffic to wait for any pending database changes before allowing the new version of the Lambda function to respond.
C. Add a BeforeAllowTraffic hook to the AppSpec file that tests and waits for any necessary database changes before deploying the new version of the Lambda function.
D. Add a validateService hook to the AppSpec file that inspects incoming traffic and rejects the payload if dependent services such as the database are not yet ready.

_____ Answer: A

Explanation:

https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure- hooks.html#appspec-hooks-lambda

Question: 130

A company uses Amazon S3 to store proprietary information. The development team creates buckets for new projects on a daily basis. The security team wants to ensure that all existing and future buckets have encryption logging and versioning enabled. Additionally, no buckets should ever be publicly read or write accessible.
What should a DevOps engineer do to meet these requirements?

A. Enable AWS CloudTrail and configure automatic remediation using AWS Lambda.
B. Enable AWS Conflg rules and configure automatic remediation using AWS Systems Manager documents.
C. Enable AWS Trusted Advisor and configure automatic remediation using Amazon EventBridge.
D. Enable AWS Systems Manager and configure automatic remediation using Systems Manager documents.

Answer: B

Explanation:

https://aws.amazon.com/blogs/mt/aws-config-auto-remediation-s3-compliance/
https://aws.amazon.com/blogs/aws/aws-config-rules-dynamic-compliance-checking-for-cloud- resources/

Question: 131

A company manages a web application that runs on Amazon EC2 instances behind an Application Load Balancer (ALB). The EC2 instances run in an Auto Scaling group across multiple Availability Zones. The application uses an Amazon RDS for MySQL DB instance to store the dat
a. The company has configured Amazon Route 53 with an alias record that points to the ALB.
A new company guideline requires a geographically isolated disaster recovery (DR> site with an RTO of 4 hours and an RPO of 15 minutes.
Which DR strategy will meet these requirements with the LEAST change to the application stack?

A. Launch a replica environment of everything except Amazon RDS in a different Availability Zone Create an RDS read replica in the new Availability Zone: and configure the new stack to point to the local RDS DB instance. Add the new stack to the Route 53 record set by using a hearth check to configure a failover routing policy.
B. Launch a replica environment of everything except Amazon RDS in a different AWS. Region Create an RDS read replica in the new Region and configure the new stack to point to the local RDS DB instance. Add the new stack to the Route 53 record set by using a health check to configure a latency routing policy.
C. Launch a replica environment of everything except Amazon RDS ma different AWS Region. In the event of an outage copy and restore the latest RDS snapshot from the primary. Region to the DR Region Adjust the Route 53 record set to point to the ALB in the DR Region.
D. Launch a replica environment of everything except Amazon RDS in a different AWS Region. Create an RDS read replica in the new Region and configure the new environment to point to the local RDS DB instance. Add the new stack to the Route 53 record set by using a health check to configure a failover routing policy. In the event of an outage promote the read replica to primary.

Answer: D

Explanation:

www.atmicnetworks.com

Question: 132

A company is running an application on Amazon EC2 instances in an Auto Scaling group. Recently an issue occurred that prevented EC2 instances from launching successfully and it took several hours for the support team to discover the issue. The support team wants to be notified by email whenever an EC2 instance does not start successfully.
Which action will accomplish this?

A. Add a health check to the Auto Scaling group to invoke an AWS Lambda function whenever an instance status is impaired.
B. Configure the Auto Scaling group to send a notification to an Amazon SNS topic whenever a failed instance launch occurs.
C. Create an Amazon CloudWatch alarm that invokes an AWS Lambda function when a failed Attachinstances Auto Scaling API call is made.
D. Create a status check alarm on Amazon EC2 to send a notification to an Amazon SNS topic whenever a status check fail occurs.

_____ Answer: B

Explanation:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/ASGettingNotifications.html#auto-scaling- sns-notifications

Question: 133

A company requires an RPO of 2 hours and an RTO of 10 minutes for its data and application at all times. An application uses a MySQL database and Amazon EC2 web servers. The development team needs a strategy for failover and disaster recovery.
Which combination of deployment strategies will meet these requirements? (Select TWO.)

A. Create an Amazon Aurora cluster in one Availability Zone across multiple Regions as the data store Use Aurora's automatic recovery capabilities in the event of a disaster
B. Create an Amazon Aurora global database in two Regions as the data store. In the event of a failure promote the secondary Region as the primary for the application.
C. Create an Amazon Aurora multi-master cluster across multiple Regions as the data store. Use a Network Load Balancer to balance the database traffic in different Regions.
D. Set up the application in two Regions and use Amazon Route 53 failover-based routing that points to the Application Load Balancers in both Regions. Use hearth checks to determine the availability in a given Region. Use Auto Scaling groups in each Region to adjust capacity based on demand.
E. Set up the application m two Regions and use a multi-Region Auto Scaling group behind Application Load Balancers to manage the capacity based on demand. In the event of a disaster adjust the Auto Scaling group's desired instance count to increase baseline capacity in the failover Region.

Answer: BD

Explanation:

Question: 134

The security team depends on AWS CloudTrail to detect sensitive security issues in the company's AWS account. The DevOps engineer needs a solution to auto-remediate CloudTrail being turned off in an AWS account.
What solution ensures the LEAST amount of downtime for the CloudTrail log deliveries?

A. Create an Amazon EventBridge rule for the CloudTrail StopLogging event. Create an AWS Lambda (unction that uses the AWS SDK to call StartLogging on the ARN of the resource in which StopLogging was called. Add the Lambda function ARN as a target to the EventBridge rule.

B. Deploy the AWS-managed CloudTrail-enabled AWS Config rule set with a periodic interval to 1 hour. Create an Amazon EventBridge rule tor AWS Config rules compliance change. Create an AWS Lambda function that uses the AWS SDK to call StartLogging on the ARN of the resource in which StopLoggmg was called. Add the Lambda function ARN as a target to the EventBridge rule.

C. Create an Amazon EventBridge rule for a scheduled event every 5 minutes. Create an AWS Lambda function that uses the AWS SDK to call StartLogging on a CloudTrail trail in the AWS account. Add the Lambda function ARN as a target to the EventBridge rule.

D. Launch a t2 nano instance with a script running every 5 minutes that uses the AWS SDK to query CloudTrail in the current account. If the CloudTrail trail is disabled have the script re-enable the trail.

Answer: A

Explanation:

https://aws.amazon.com/blogs/mt/monitor-changes-and-auto-enable-logging-in-aws-cloudtrail/

Question: 135 A company is using an organization in AWS Organizations to manage multiple AWS accounts. The company's development team wants to use AWS Lambda functions to meet resiliency requirements and is rewriting all applications to work with Lambda functions that are deployed in a VPC. The development team is using Amazon Elastic Pile System (Amazon EFS) as shared storage in Account A in the organization.

The company wants to continue to use Amazon EPS with Lambda Company policy requires all serverless projects to be deployed in Account B.

A DevOps engineer needs to reconfigure an existing EFS file system to allow Lambda functions to access the data through an existing EPS access point.

Which combination of steps should the DevOps engineer take to meet these requirements? (Select THREE.)

A. Update the EFS file system policy to provide Account B with access to mount and write to the EFS file system in Account A.

B. Create SCPs to set permission guardrails with fine-grained control for Amazon EFS.

C. Create a new EFS file system in Account B Use AWS Database Migration Service (AWS DMS) to keep data from Account A and Account B synchronized.

D. Update the Lambda execution roles with permission to access the VPC and the EFS file system.

E. Create a VPC peering connection to connect Account A to Account B.

F. Configure the Lambda functions in Account B to assume an existing IAM role in Account A.

Answer: AEF

Explanation:

A Lambda function in one account can mount a file system in a different account. For this scenario, you configure VPC peering between the function VPC and the file system VPC.

https://docs.aws.amazon.com/lambda/latest/dg/services-efs.htmlhttps://aws.amazon.com/ru/blogs/storage/mount-amazon-efs-file-systems-cross-account-from- amazon-eks/

1. Need to update the file system policy on EFS to allow mounting the file system into Account B.

File System Policy

```
$ cat file-system-policy.json
{
"Statement": [
{
"Effect": "Allow",
"Action": [
```

www.atmicnetworks.com

"elasticfilesystem:ClientMount",
"elasticfilesystem:ClientWrite"
],
"Principal": {
"AWS": "arn:aws:iam::<aws-account-id-A>:root" # Replace with AWS account ID of EKS cluster
}
}
]
}

2. Need VPC peering between Account A and Account B as the pre-requisite
3. Need to assume cross-account IAM role to describe the mounts so that a specific mount can be chosen.

Question: 136

A company deploys its corporate infrastructure on AWS across multiple AWS Regions and Availability Zones. The infrastructure is deployed on Amazon EC2 instances and connects with AWS loT Greengrass devices. The company deploys additional resources on on-premises servers that are located in the corporate headquarters.

The company wants to reduce the overhead involved in maintaining and updating its resources. The company's DevOps team plans to use AWS Systems Manager to implement automated management and application of patches. The DevOps team confirms that Systems Manager is available in the Regions that the resources are deployed m Systems Manager also is available in a Region near the corporate headquarters.

Which combination of steps must the DevOps team take to implement automated patch and configuration management across the company's EC2 instances loT devices and on-premises infrastructure? (Select THREE.)

A. Apply tags lo all the EC2 instances. AWS loT Greengrass devices, and on-premises servers. Use Systems Manager Session Manager to push patches to all the tagged devices.

B. Use Systems Manager Run Command to schedule patching for the EC2 instances AWS loT Greengrass devices and on-premises servers.

C. Use Systems Manager Patch Manager to schedule patching loT the EC2 instances AWS loT Greengrass devices and on-premises servers as a Systems Manager maintenance window task.

D. Configure Amazon EventBridge to monitor Systems Manager Patch Manager for updates to patch baselines. Associate Systems Manager Run Command with the event lo initiate a patch action for all EC2 instances AWS loT Greengrass devices and on-premises servers.

E. Create an IAM instance profile for Systems Manager Attach the instance profile to all the EC2 instances in the AWS account. For the AWS loT Greengrass devices and on-premises servers create an IAM service role for Systems Manager.

F. Generate a managed-instance activation Use the Activation Code and Activation ID to install Systems Manager Agent (SSM Agent) on each server in the on-premises environment Update the AWS loT Greengrass IAM token exchange role Use the role to deploy SSM Agent on all the loT devices.

Answer: CEF

Explanation:

https://aws.amazon.com/blogs/mt/how-to-centrally-manage-aws-iot-greengrass-devices-using-aws-systems-manager/?force isolation=true

www.atmicnetworks.com

Question: 137

A company is developing a new application. The application uses AWS Lambda functions for its compute tier. The company must use a canary deployment for any changes to the Lambda functions. Automated rollback must occur if any failures are reported.

The company's DevOps team needs to create the infrastructure as code (IaC) and the CI/CD pipeline for this solution.

Which combination of steps will meet these requirements? (Choose three.)

A. Create an AWS CloudFormation template for the application. Define each Lambda function in the template by using the AWS::Lambda::Function resource type. In the template, include a version for the Lambda function by using the AWS::Lambda::Version resource type. Declare the CodeSha256 property. Configure an AWS::Lambda::Alias resource that references the latest version of the Lambda function.

B. Create an AWS Serverless Application Model (AWS SAM) template for the application. Define each Lambda function in the template by using the AWS::Serverless::Function resource type. For each function, include configurations for the AutoPublishAlias property and the DeploymentPreference property. Configure the deployment configuration type to LambdaCanary10Percent10Minutes.

C. Create an AWS CodeCommit repository. Create an AWS CodePipeline pipeline. Use the CodeCommit repository in a new source stage that starts the pipeline. Create an AWS CodeBuild project to deploy the AWS Serverless Application Model (AWS SAM) template. Upload the template and source code to the CodeCommit repository. In the CodeCommit repository, create a buildspec.yml file that includes the commands to build and deploy the SAM application.

D. Create an AWS CodeCommit repository. Create an AWS CodePipeline pipeline. Use the CodeCommit repository in a new source stage that starts the pipeline. Create an AWS CodeDeploy deployment group that is configured for canary deployments with a DeploymentPreference type of Canary10Percent10Minutes. Upload the AWS CloudFormation template and source code to the CodeCommit repository. In the CodeCommit repository, create an appspec.yml file that includes the commands to deploy the CloudFormation template.

E. Create an Amazon CloudWatch composite alarm for all the Lambda functions. Configure an evaluation period and dimensions for Lambda. Configure the alarm to enter the ALARM state if any errors are detected or if there is insufficient data.

F. Create an Amazon CloudWatch alarm for each Lambda function. Configure the alarms to enter the ALARM state if any errors are detected. Configure an evaluation period, dimensions for each Lambda function and version, and the namespace as AWS/Lambda on the Errors metric.

Answer: BCF

Explanation:

The requirement is to create the infrastructure as code (IaC) and the CI/CD pipeline for the Lambda application that uses canary deployment and automated rollback. To do this, the DevOps team needs to use the following steps:

Create an AWS Serverless Application Model (AWS SAM) template for the application. AWS SAM is a framework that simplifies the development and deployment of serverless applications on AWS. AWS SAM allows customers to define Lambda functions and other resources in a template by using a simplified syntax. For each Lambda function, the DevOps team can include configurations for the AutoPublishAlias property and the DeploymentPreference property. The AutoPublishAlias property specifies the name of the alias that points to the latest version of the function. The DeploymentPreference property specifies how CodeDeploy deploys new versions of the function. By configuring the deployment configuration type to LambdaCanary10Percent10Minutes, the DevOps team can enable canary deployment with 10% of traffic shifted to the new version every 10 minutes.

Create an AWS CodeCommit repository. Create an AWS CodePipeline pipeline. Use the CodeCommit repository in a new source stage that starts the pipeline. Create an AWS CodeBuild project to deploy the AWS SAM template. CodeCommit is a fully managed source control service that hosts Git repositories. CodePipeline is a fully managed continuous delivery service that automates the release process of software applications. CodeBuild is a fully managed continuous integration service that compiles source code and runs tests. By using

www.atmicnetworks.com

these services, the DevOps team can create a CI/CD pipeline for the Lambda application. The pipeline should use the CodeCommit repository as the source stage, where the DevOps team can upload the SAM template and source code. The pipeline should also use a CodeBuild project as the build stage, where the SAM template can be built and deployed.
Create an Amazon CloudWatch alarm for each Lambda function. Configure the alarms to enter the ALARM state if any errors are detected. Configure an evaluation period, dimensions for each Lambda function and version, and the namespace as AWS/Lambda on the Errors metric. CloudWatch is a service that monitors and collects metrics from AWS resources and applications. CloudWatch alarms are actions that are triggered when a metric crosses a specified threshold. By creating CloudWatch alarms for each Lambda function, the DevOps team can monitor the health and performance of each function version during deployment. By configuring the alarms to enter the ALARM state if any errors are detected, the DevOps team can enable automated rollback if any failures are reported.

Question: 138

A company that runs many workloads on AWS has an Amazon EBS spend that has increased over time. The DevOps team notices there are many unattached
EBS volumes. Although there are workloads where volumes are detached, volumes over 14 days old are stale and no longer needed. A DevOps engineer has been tasked with creating automation that deletes unattached EBS volumes that have been unattached for 14 days. Which solution will accomplish this?

A. Configure the AWS Config ec2-volume-inuse-check managed rule with a configuration changes trigger type and an Amazon EC2 volume resource target. Create a new Amazon CloudWatch Events rule scheduled to execute an AWS Lambda function in 14 days to delete the specified EBS volume.
B. Use Amazon EC2 and Amazon Data Lifecycle Manager to configure a volume lifecycle policy. Set the interval period for unattached EBS volumes to 14 days and set the retention rule to delete. Set the policy target volumes as *.
C. Create an Amazon CloudWatch Events rule to execute an AWS Lambda function daily. The Lambda function should find unattached EBS volumes and tag them with the current date, and delete unattached volumes that have tags with dates that are more than 14 days old.
D. Use AWS Trusted Advisor to detect EBS volumes that have been detached for more than 14 days. Execute an AWS Lambda function that creates a snapshot and then deletes the EBS volume.

_____ Answer: C

Explanation:

The requirement is to create automation that deletes unattached EBS volumes that have been unattached for 14 days. To do this, the DevOps engineer needs to use the following steps: Create an Amazon CloudWatch Events rule to execute an AWS Lambda function daily. CloudWatch Events is a service that enables event-driven architectures by delivering events from various sources to targets. Lambda is a service that lets you run code without provisioning or managing servers. By

creating a CloudWatch Events rule that executes a Lambda function daily, the DevOps engineer can schedule a recurring task to check and delete unattached EBS volumes.
The Lambda function should find unattached EBS volumes and tag them with the current date, and delete unattached volumes that have tags with dates that are more than 14 days old. The Lambda function can use the EC2 API to list and filter unattached EBS volumes based on their state and tags. The function can then tag each unattached volume with the current date using the create-tags command. The function can also compare the tag value with the current date and delete any unattached volume that has been tagged more than 14 days ago using the delete-volume command.

Question: 139

AnyCompany is using AWS Organizations to create and manage multiple AWS accounts AnyCompany recently acquired a smaller

www.atmicnetworks.com

company, Example Corp. During the acquisition process, Example Corp's single AWS account joined AnyCompany's management account through an Organizations invitation. AnyCompany moved the new member account under an OU that is dedicated to Example Corp.

AnyCompany's DevOps engineer has an IAM user that assumes a role that is named OrganizationAccountAccessRole to access member accounts. This role is configured with a full access policy When the DevOps engineer tries to use the AWS Management Console to assume the role in Example Corp's new member account, the DevOps engineer receives the following error message "Invalid information in one or more fields. Check your information or contact your administrator."

Which solution will give the DevOps engineer access to the new member account?

A. In the management account, grant the DevOps engineer's IAM user permission to assume the OrganzatlonAccountAccessR01e IAM role in the new member account.

B. In the management account, create a new SCR In the SCP, grant the DevOps engineer's IAM user full access to all resources in the new member account. Attach the SCP to the OU that contains the new member account,

C. In the new member account, create a new IAM role that is named OrganizationAccountAccessRole. Attach the AdmInistratorAccess AVVS managed policy to the role. In the role's trust policy, grant the management account permission to assume the role.

D. In the new member account edit the trust policy for the Organ zationAccountAccessRole IAM role. Grant the management account permission to assume the role.

Answer: C

Explanation:

The problem is that the DevOps engineer cannot assume the OrganizationAccountAccessRole IAM role in the new member account that joined AnyCompany's management account through an Organizations invitation. The solution is to create a new IAM role with the same name and trust policy in the new member account.

Option A is incorrect, as it does not address the root cause of the error. The DevOps engineer's IAM user already has permission to assume the OrganizationAccountAccessRole IAM role in any member account, as this is the default role name that AWS Organizations creates when a new account joins an organization. The error occurs because the new member account does not have this role, as it was not created by AWS Organizations.

Option B is incorrect, as it does not address the root cause of the error. An SCP is a policy that defines the maximum permissions for account members of an organization or organizational unit (OU). An SCP does not grant permissions to IAM users or roles, but rather limits the permissions that identitybased policies or resource-based policies grant to them. An SCP also does not affect how IAM roles are assumed by other principals.

Option C is correct, as it addresses the root cause of the error. By creating a new IAM role with the same name and trust policy as the OrganizationAccountAccessRole IAM role in the new member account, the DevOps engineer can assume this role and access the account. The new role should have the AdministratorAccess AWS managed policy attached, which grants full access to all AWS resources in the account. The trust policy should allow the management account to assume the role, which can be done by specifying the management account ID as a principal in the policy statement. Option D is incorrect, as it assumes that the new member account already has the OrganizationAccountAccessRole IAM role, which is not true. The new member account does not have this role, as it was not created by AWS Organizations. Editing the trust policy of a non-existent role will not solve the problem.

Question: 140

A DevOps engineer has developed an AWS Lambda function The Lambda function starts an AWS CloudFormation drift detection operation on all supported resources for a specific CloudFormation stack The Lambda function then exits Its invocation The DevOps engineer has created an Amazon EventBrdge scheduled rule that Invokes the Lambda function every hour. An Amazon Simple Notification Service (Amazon SNS) topic already exists In the AWS account. The DevOps engineer has subscribed to the SNS topic to receive notifications

The DevOps engineer needs to receive a notification as soon as possible when drift is detected in this specific stack configuration.

Which solution Will meet these requirements?

A. Configure the existing EventBridge rule to also target the SNS topic Configure an SNS subscription filter policy to match the Cloud Formation stack. Attach the subscription filter policy to the SNS tomc

B. Create a second Lambda function to query the CloudFormation API for the drift detection results for the stack Configure the second Lambda function to publish a message to the SNS topic If drift ts detected Adjust the existing EventBridge rule to also target the second Lambda function

C. Configure Amazon GuardDuty in the account with drift detection for all CloudFormation stacks. Create a second EventBndge rule that reacts to the GuardDuty drift detection event finding for the specific CloudFormation stack. Configure the SNS topic as a target of the second EventBridge rule.

D. Configure AWS Config in the account. Use the cloudformation-stack-drift-detection-check managed rule. Create a second EventBndge rule that reacts to a compliance change event for the

CloudFormaUon stack. Configure the SNS topc as a target of the second EventBridge rule.

Answer: D

Explanation:

A comprehensive and detailed explanation is:
Option A is incorrect because EventBridge rules cannot filter events based on the message body or attributes of the target service. Therefore, configuring an SNS subscription filter policy to match the CloudFormation stack will not work. The SNS topic will receive all events from the EventBridge rule, regardless of the stack name or drift status.
Option B is incorrect because it introduces unnecessary complexity and cost. Creating a second Lambda function to query the CloudFormation API for the drift detection results is redundant, since CloudFormation already publishes drift detection events to EventBridge. Moreover, invoking two Lambda functions every hour will incur more charges than invoking one.
Option C is incorrect because GuardDuty does not provide drift detection for CloudFormation stacks. GuardDuty is a threat detection service that monitors for malicious activity and unauthorized behavior in AWS accounts and workloads. It does not monitor or report on configuration changes or drifts in CloudFormation stacks.
Option D is correct because it leverages AWS Config and its managed rule for drift detection. AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources. It can detect configuration changes and drifts in CloudFormation stacks using the cloudformation-stack-drift-detection-check managed rule. This rule triggers an AWS Config event when a stack drifts from its expected template configuration. By creating a second EventBridge rule that reacts to this event for the specific stack, the DevOps engineer can configure the SNS topic as a target and receive a notification as soon as possible when drift is detected.

References:
AWS Config
Amazon SNS subscription filter policies

Amazon EventBridge rules

Question: 141

A DevOps team uses AWS CodePipeline, AWS CodeBuild, and AWS CodeDeploy to deploy an application. The application is a REST API that uses AWS Lambda functions and Amazon API Gateway Recent deployments have introduced errors that have affected many customers.

The DevOps team needs a solution that reverts to the most recent stable version of the application when an error is detected. The solution must affect the fewest customers possible.

Which solution Will meet these requirements With the MOST operational efficiency?

A. Set the deployment configuration in CodeDeploy to LambdaAllAtOnce Configure automatic rollbacks on the deployment group Create an Amazon CloudWatch alarm that detects HTTP Bad Gateway errors on API Gateway Configure the deployment group to roll back when the number of alarms meets the alarm threshold

B. Set the deployment configuration in CodeDeploy to LambdaCanary10Percent10Minutes. Configure automatic rollbacks on the deployment group Create an Amazon CloudWatch alarm that detects HTTP Bad Gateway errors on API Gateway Configure the deployment group to roll back when the number of alarms meets the alarm threshold

C. Set the deployment configuration in CodeDeploy to LambdaAllAtOnce Configure manual rollbacks on the deployment group. Create an Amazon Simple Notification Service (Amazon SNS) topc to send notifications every time a deployrnent fads. Configure the SNS topc to Invoke a new Lambda function that stops the current deployment and starts the most recent successful deployment

D. Set the deployment configuration in CodeDeploy to LambdaCanaryIOPercentIOMinutes Configure manual rollbacks on the deployment group Create a metric filter on an Amazon CloudWatch log group for API Gateway to monitor HTTP Bad Gateway errors. Configure the metric filter to Invoke a new Lambda function that stops the current eployment and starts the most recent successful deployment

_____ Answer: B

Explanation:

Option A is incorrect because setting the deployment configuration to LambdaAllAtOnce means that the new version of the application will be deployed to all Lambda functions at once, affecting all customers. This does not meet the requirement of affecting the fewest customers possible. Moreover, configuring automatic rollbacks on the deployment group is not operationally efficient, as it requires manual intervention to fix the errors and redeploy the application.

Option B is correct because setting the deployment configuration to LambdaCanary10Percent10Minutes means that the new version of the application will be deployed to 10 percent of the Lambda functions first, and then to the remaining 90 percent after 10 minutes. This minimizes the impact of errors on customers, as only 10 percent of them will be affected by a faulty deployment. Configuring automatic rollbacks on the deployment group also meets the requirement of reverting to the most recent stable version of the application when an error is detected. Creating a CloudWatch alarm that detects HTTP Bad Gateway errors on API Gateway is a valid way to monitor the health of the application and trigger a rollback if needed.

Option C is incorrect because setting the deployment configuration to LambdaAllAtOnce means that the new version of the application will be deployed to all Lambda functions at once, affecting all customers. This does not meet the requirement of affecting the fewest customers possible. Moreover, configuring manual rollbacks on the deployment group is not operationally efficient, as it requires human intervention to stop the current deployment and start a new one. Creating an SNS topic to send notifications every time a deployment fails is not sufficient to detect errors in the application, as it does not monitor the API Gateway responses.

Option D is incorrect because configuring manual rollbacks on the deployment group is not operationally efficient, as it requires human intervention to stop the current deployment and start a new one. Creating a metric filter on a CloudWatch log group for API Gateway to monitor HTTP Bad Gateway errors is a valid way to monitor the health of the application, but invoking a new Lambda function to perform a rollback is unnecessary and complex, as CodeDeploy already provides automatic rollback functionality.

References:

AWS CodeDeploy Deployment Configurations

[AWS CodeDeploy Rollbacks]

Amazon CloudWatch Alarms

Question: 142

A company has multiple development teams in different business units that work in a shared single AWS account All Amazon EC2 resources that are created in the account must include tags that specify who created the resources. The tagging must occur within the first hour of resource creation.

A DevOps engineer needs to add tags to the created resources that Include the user ID that created the resource and the cost center ID The DevOps engineer configures an AWS Lambda function With the cost center mappings to tag the resources. The DevOps engineer also sets up AWS CloudTrail in the AWS account. An Amazon S3 bucket stores the CloudTrail event logs

Which solution will meet the tagging requirements?

A. Create an S3 event notification on the S3 bucket to invoke the Lambda function for s3.ObJectTagging:Put events. Enable bucket versioning on the S3 bucket.

B. Enable server access logging on the S3 bucket. Create an S3 event notification on the S3 bucket for s3. ObjectTagglng.^ events

C. Create a recurring hourly Amazon EventBridge scheduled rule that invokes the Larnbda function. Modify the Lambda function to read the logs from the S3 bucket

D. Create an Amazon EventBridge rule that uses Amazon EC2 as the event source. Configure the rule to match events delivered by CloudTrail. Configure the rule to target the Lambda function

Answer: D

Explanation:

Option A is incorrect because S3 event notifications do not support s3.ObjectTagging:Put events. S3 event notifications only support events related to object creation, deletion, replication, and restore. Moreover, enabling bucket versioning on the S3 bucket is not relevant to the tagging requirements, as it only keeps multiple versions of objects in the bucket.

Option B is incorrect because enabling server access logging on the S3 bucket does not help with tagging the resources. Server access logging only records requests for access to the bucket or its objects. It does not capture the user ID or the cost center ID of the resources. Furthermore, creating an S3 event notification on the S3 bucket for s3.ObjectTagging:Put events is not possible, as explained in option A.

Option C is incorrect because creating a recurring hourly Amazon EventBridge scheduled rule that invokes the Lambda function is not efficient or timely. The Lambda function would have to read the logs from the S3 bucket every hour and tag the resources accordingly, which could incur unnecessary costs and delays. A better solution would be to trigger the Lambda function as soon as a resource is created, rather than waiting for an hourly schedule.

Option D is correct because creating an Amazon EventBridge rule that uses Amazon EC2 as the event source and matches events delivered by CloudTrail is a valid way to tag the resources. CloudTrail records all API calls made to AWS services, including EC2, and delivers them as events to EventBridge. The EventBridge rule can filter the events based on the user ID and the resource type, and then

target the Lambda function to tag the resources with the cost center ID. This solution meets the tagging requirements in a timely and efficient manner.

References:

S3 event notifications

Server access logging

Amazon EventBridge rules

AWS CloudTrail

Question: 143

A company sells products through an ecommerce web application The company wants a dashboard that shows a pie chart of product transaction details. The company wants to integrate the dashboard With the company's existing Amazon CloudWatch dashboards

Which solution Will meet these requirements With the MOST operational efficiency?

A. Update the ecommerce application to emit a JSON object to a CloudWatch log group for each processed transaction. Use CloudWatch Logs Insights to query the log group and to visualize the results in a pie chart format Attach the results to the desired CloudWatch dashboard.

B. Update the ecommerce application to emit a JSON object to an Amazon S3 bucket for each processed transaction. Use Amazon Athena to query the S3 bucket and to visualize the results In a Pie chart format. Export the results from Athena Attach the results to the desired CloudWatch dashboard

C. Update the ecommerce application to use AWS X-Ray for instrumentation. Create a new X-Ray subsegment Add an annotation for each processed transaction. Use X-Ray traces to query the data and to visualize the results in a pie chart format Attach the results to the desired CloudWatch dashboard

D. Update the ecommerce application to emit a JSON object to a CloudWatch log group for each processed transaction_ Create an AWS Lambda function to aggregate and write the results to Amazon DynamoDB. Create a Lambda subscription filter for the log file. Attach the results to the desired CloudWatch dashboard.

_____ Answer: A

Explanation:

The correct answer is A.

A comprehensive and detailed explanation is:

Option A is correct because it meets the requirements with the most operational efficiency. Updating the ecommerce application to emit a JSON object to a CloudWatch log group for each processed transaction is a simple and cost-effective way to collect the data needed for the dashboard. Using CloudWatch Logs Insights to query the log group and to visualize the results in a pie chart format is also a convenient and integrated solution that leverages the existing CloudWatch dashboards. Attaching the results to the desired CloudWatch dashboard is straightforward and does not require any additional steps or services.

Option B is incorrect because it introduces unnecessary complexity and cost. Updating the ecommerce application to emit a JSON object to an Amazon S3 bucket for each processed transaction is a valid way to store the data, but it requires creating and managing an S3 bucket and its permissions. Using Amazon Athena to query the S3 bucket and to visualize the results in a pie chart format is also a valid way to analyze the data, but it incurs charges based on the amount of data scanned by each query. Exporting the results from Athena and attaching them to the desired

CloudWatch dashboard is also an extra step that adds more overhead and latency.

Option C is incorrect because it uses AWS X-Ray for an inappropriate purpose. Updating the ecommerce application to use AWS X-Ray for instrumentation is a good practice for monitoring and tracing distributed applications, but it is not designed for aggregating product transaction details. Creating a new X-Ray subsegment and adding an annotation for each processed transaction is possible, but it would clutter the X-Ray service map and make it harder to debug performance issues. Using X-Ray traces to query the data and to visualize the results in a pie chart format is also possible, but it would require custom code and logic that are not supported by X-Ray natively. Attaching the results to the desired CloudWatch dashboard is also not supported by X-Ray directly, and would require additional steps or services.

Option D is incorrect because it introduces unnecessary complexity and cost. Updating the ecommerce application to emit a JSON object to a CloudWatch log group for each processed transaction is a simple and cost-effective way to collect the data needed for the dashboard, as in option A) However, creating an AWS Lambda function to aggregate and write the results to Amazon DynamoDB is redundant, as CloudWatch Logs Insights can already perform aggregation queries on log data. Creating a Lambda subscription filter for the log file is also redundant, as CloudWatch Logs Insights can already access log data directly. Attaching the results to the desired CloudWatch dashboard would also require additional steps or services, as DynamoDB does not support native integration with CloudWatch dashboards.

References:

CloudWatch Logs Insights

Amazon Athena

AWS X-Ray

AWS Lambda

Amazon DynamoDB

Question: 144

A company has configured an Amazon S3 event source on an AWS Lambda function The company needs the Lambda function to run when a new object is created or an existing object IS modified In a particular S3 bucket The Lambda function will use the S3 bucket name and the S3 object key of the incoming event to read the contents of the created or modified S3 object The Lambda function will parse the contents and save the parsed contents to an Amazon DynamoDB table.

The Lambda function's execution role has permissions to read from the S3 bucket and to write to the DynamoDB table, During testing, a DevOps engineer discovers that the Lambda

function does not run when objects are added to the S3 bucket or when existing objects are modified.

Which solution will resolve this problem?

A. Increase the memory of the Lambda function to give the function the ability to process large files from the S3 bucket.

B. Create a resource policy on the Lambda function to grant Amazon S3 the permission to invoke the Lambda function for the S3 bucket

C. Configure an Amazon Simple Queue Service (Amazon SQS) queue as an OnFailure destination for the Lambda function

D. Provision space in the /tmp folder of the Lambda function to give the function the ability to process large files from the S3 bucket

www.atmicnetworks.com

Answer: B

Explanation:

Option A is incorrect because increasing the memory of the Lambda function does not address the root cause of the problem, which is that the Lambda function is not triggered by the S3 event source. Increasing the memory of the Lambda function might improve its performance or reduce its execution time, but it does not affect its invocation. Moreover, increasing the memory of the Lambda function might incur higher costs, as Lambda charges based on the amount of memory allocated to the function.

Option B is correct because creating a resource policy on the Lambda function to grant Amazon S3 the permission to invoke the Lambda function for the S3 bucket is a necessary step to configure an S3 event source. A resource policy is a JSON document that defines who can access a Lambda resource and under what conditions. By granting Amazon S3 permission to invoke the Lambda function, the company ensures that the Lambda function runs when a new object is created or an existing object is modified in the S3 bucket1.

Option C is incorrect because configuring an Amazon Simple Queue Service (Amazon SQS) queue as an On-Failure destination for the Lambda function does not help with triggering the Lambda function. An On-Failure destination is a feature that allows Lambda to send events to another service, such as SQS or Amazon Simple Notification Service (Amazon SNS), when a function invocation fails. However, this feature only applies to asynchronous invocations, and S3 event sources use synchronous invocations. Therefore, configuring an SQS queue as an On-Failure destination would have no effect on the problem.

Option D is incorrect because provisioning space in the /tmp folder of the Lambda function does not address the root cause of the problem, which is that the Lambda function is not triggered by the S3 event source. Provisioning space in the /tmp folder of the Lambda function might help with processing large files from the S3 bucket, as it provides temporary storage for up to 512 MB of data. However, it does not affect the invocation of the Lambda function.

References:

Using AWS Lambda with Amazon S3
Lambda resource access permissions
AWS Lambda destinations
[AWS Lambda file system]

Question: 145

A company manages multiple AWS accounts by using AWS Organizations with OUS for the different business divisions, The company is updating their corporate network to use new IP address ranges. The company has 10 Amazon S3 buckets in different AWS accounts. The S3 buckets store reports for the different divisions. The S3 bucket configurations allow only private corporate network IP addresses to access the S3 buckets.

A DevOps engineer needs to change the range of IP addresses that have permission to access the contents of the S3 buckets The DevOps engineer also needs to revoke the permissions of two OUS in the company

Which solution will meet these requirements?

A. Create a new SCP that has two statements, one that allows access to the new range of IP addresses for all the S3 buckets and one that demes access to the old range of IP addresses for all the S3 buckets. Set a permissions boundary for the OrganzauonAccountAccessRole role In the two OUS to deny access to the S3 buckets.

B. Create a new SCP that has a statement that allows only the new range of IP addresses to access the S3 buckets. Create another SCP that denies access to the S3 buckets. Attach the second SCP to the two OUS

C. On all the S3 buckets, configure resource-based policies that allow only the new range of IP addresses to access the S3 buckets. Create a new SCP that denies access to the S3 buckets. Attach the SCP to the two OUs.

D. On all the S3 buckets, configure resource-based policies that allow only the new range of IP addresses to access the S3 buckets. Set a permissions boundary for the OrganizationAccountAccessRole role in the two OUS to deny access to the S3 buckets.

_____ Answer: C

Explanation:

The correct answer is C.
A comprehensive and detailed explanation is:
Option A is incorrect because creating a new SCP that has two statements, one that allows access to the new range of IP addresses for all the S3 buckets and one that denies access to the old range of IP addresses for all the S3 buckets, is not a valid solution. SCPs are not resource-based policies, and they cannot specify the S3 buckets or the IP addresses as resources or conditions. SCPs can only control the actions that can be performed by the principals in the organization, not the access to specific resources. Moreover, setting a permissions boundary for the OrganizationAccountAccessRole role in the two OUs to deny access to the S3 buckets is not sufficient to revoke the permissions of the two OUs, as there might be other roles or users in those OUs that can still access the S3 buckets.
Option B is incorrect because creating a new SCP that has a statement that allows only the new range of IP addresses to access the S3 buckets is not a valid solution, for the same reason as option A) SCPs are not resource-based policies, and they cannot specify the S3 buckets or the IP addresses as resources or conditions. Creating another SCP that denies access to the S3 buckets and attaching it to the two OUs is also not a valid solution, as SCPs cannot specify the S3 buckets as resources either. Option C is correct because it meets both requirements of changing the range of IP addresses that have permission to access the contents of the S3 buckets and revoking the permissions of two OUs in the company. On all the S3 buckets, configuring resource-based policies that allow only the new range of IP addresses to access the S3 buckets is a valid way to update the IP address ranges, as resource-based policies can specify both resources and conditions. Creating a new SCP that denies access to the S3 buckets and attaching it to the two OUs is also a valid way to revoke the permissions of those OUs, as SCPs can deny actions such as s3:PutObject or s3:GetObject on any resource. Option D is incorrect because setting a permissions boundary for the OrganizationAccountAccessRole role in the two OUs to deny access to the S3 buckets is not sufficient to revoke the permissions of the two OUs, as there might be other roles or users in those OUs that can still access the S3 buckets. A permissions boundary is a policy that defines the maximum permissions that an IAM entity can have. However, it does not revoke any existing permissions that are granted by other policies. References:
AWS Organizations
S3 Bucket Policies
Service Control Policies
Permissions Boundaries

Question: 146

A company hosts applications in its AWS account Each application logs to an individual Amazon CloudWatch log group. The company's CloudWatch costs for ingestion are increasing

A DevOps engineer needs to Identify which applications are the source of the increased logging costs.

Which solution Will meet these requirements?

A. Use CloudWatch metrics to create a custom expression that Identifies the CloudWatch log groups that have the most data being written to them.

B. Use CloudWatch Logs Insights to create a set of queries for the application log groups to Identify the number of logs written for

a period of time

C. Use AWS Cost Explorer to generate a cost report that details the cost for CloudWatch usage

D. Use AWS CloudTrail to filter for CreateLogStream events for each application

_____ Answer: C

Explanation:

The correct answer is C.
A comprehensive and detailed explanation is:
Option A is incorrect because using CloudWatch metrics to create a custom expression that identifies the CloudWatch log groups that have the most data being written to them is not a valid solution. CloudWatch metrics do not provide information about the size or volume of data being ingested by CloudWatch logs. CloudWatch metrics only provide information about the number of events, bytes, and errors that occur within a log group or stream. Moreover, creating a custom expression with CloudWatch metrics would require using the search_web tool, which is not necessary for this use case.
Option B is incorrect because using CloudWatch Logs Insights to create a set of queries for the application log groups to identify the number of logs written for a period of time is not a valid solution. CloudWatch Logs Insights can help analyze and filter log events based on patterns and expressions, but it does not provide information about the cost or billing of CloudWatch logs. CloudWatch Logs Insights also charges based on the amount of data scanned by each query, which could increase the logging costs further.
Option C is correct because using AWS Cost Explorer to generate a cost report that details the cost for CloudWatch usage is a valid solution. AWS Cost Explorer is a tool that helps visualize, understand, and manage AWS costs and usage over time. AWS Cost Explorer can generate custom reports that show the breakdown of costs by service, region, account, tag, or any other dimension. AWS Cost Explorer can also filter and group costs by usage type, which can help identify the specific CloudWatch log groups that are the source of the increased logging costs.
Option D is incorrect because using AWS CloudTrail to filter for CreateLogStream events for each application is not a valid solution. AWS CloudTrail is a service that records API calls and account activity for AWS services, including CloudWatch logs. However, AWS CloudTrail does not provide information about the cost or billing of CloudWatch logs. Filtering for CreateLogStream events would only show when a new log stream was created within a log group, but not how much data was ingested or stored by that log stream.
References:
CloudWatch Metrics
CloudWatch Logs Insights
AWS Cost Explorer
AWS CloudTrail

Question: 147

A company is developing an application that will generate log events. The log events consist of five distinct metrics every one tenth of a second and produce a large amount of data The company needs to configure the application to write the logs to Amazon Time stream The company will configure a daily query against the Timestream table.

Which combination of steps will meet these requirements with the FASTEST query performance? (Select THREE.)

A. Use batch writes to write multiple log events in a Single write operation

B. Write each log event as a single write operation

C. Treat each log as a single-measure record

D. Treat each log as a multi-measure record

E. Configure the memory store retention period to be longer than the magnetic store retention period

F. Configure the memory store retention period to be shorter than the magnetic store retention period

Answer: ADF

Explanation:

A comprehensive and detailed explanation is:
Option A is correct because using batch writes to write multiple log events in a single write operation is a recommended practice for optimizing the performance and cost of data ingestion in Timestream. Batch writes can reduce the number of network round trips and API calls, and can also take advantage of parallel processing by Timestream. Batch writes can also improve the compression ratio of data in the memory store and the magnetic store, which can reduce the storage costs and improve the query performance!
Option B is incorrect because writing each log event as a single write operation is not a recommended practice for optimizing the performance and cost of data ingestion in Timestream. Writing each log event as a single write operation would increase the number of network round trips and API calls, and would also reduce the compression ratio of data in the memory store and the magnetic store. This would increase the storage costs and degrade the query performance!
Option C is incorrect because treating each log as a single-measure record is not a recommended practice for optimizing the query performance in Timestream. Treating each log as a single-measure record would result in creating multiple records for each timestamp, which would increase the storage size and the query latency. Moreover, treating each log as a single-measure record would require using joins to query multiple measures for the same timestamp, which would add complexity and overhead to the query processing2.
Option D is correct because treating each log as a multi-measure record is a recommended practice for optimizing the query performance in Timestream. Treating each log as a multi-measure record would result in creating a single record for each timestamp, which would reduce the storage size and the query latency. Moreover, treating each log as a multi-measure record would allow querying multiple measures for the same timestamp without using joins, which would simplify and speed up the query processing2.
Option E is incorrect because configuring the memory store retention period to be longer than the magnetic store retention period is not a valid option in Timestream. The memory store retention period must always be shorter than or equal to the magnetic store retention period. This ensures that data is moved from the memory store to the magnetic store before it expires out of the memory store3.
Option F is correct because configuring the memory store retention period to be shorter than the magnetic store retention period is a valid option in Timestream. The memory store retention period determines how long data is kept in the memory store, which is optimized for fast point-in-time queries. The magnetic store retention period determines how long data is kept in the magnetic store, which is optimized for fast analytical queries. By configuring these retention periods appropriately, you can balance your storage costs and query performance according to your application needs3. References: 1: Batch writes
2: Multi-measure records vs. single-measure records
3: Storage

Question: 148

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 IAM 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 IAM 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 IAM service role to have access.

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

_____ Answer: A

Explanation:

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 authentica-tion token. Update the docker login command to use the authentication token to access the ECR repository.
This is the correct solution. The aws ecr get-login-password AWS CLI command retrieves and displays an authentication token that can be used to log in to an ECR repository. The docker login command can use this token as a password to authenticate with the ECR repository. This way, the CodeBuild project can push and pull images from the ECR repository without any errors. For more information, see Using Amazon ECR with the AWS CLI and get-login-password.

Question: 149

A company recently launched multiple applications that use Application Load Balancers. Application response time often slows down when the applications experience problems A DevOps engineer needs to Implement a monitoring solution that alerts the company when the applications begin to perform slowly The DevOps engineer creates an Amazon Simple Notification Semce (Amazon SNS) topic and subscribe the company's email address to the topic

What should the DevOps engineer do next to meet the requirements?

A. Create an Amazon EventBridge rule that invokes an AWS Lambda function to query the applications on a 5-minute interval Configure the Lambda function to publish a notification to the SNS topic when the applications return errors.

B. Create an Amazon CloudWatch Synthetics canary that runs a custom script to query the applications on a 5-minute interval.

Configure the canary to use the SNS topic when the applications return errors.

C. Create an Amazon CloudWatch alarm that uses the AWS/AppljcabonELB namespace RequestCountPerTarget metric Configure the CloudWatch alarm to send a notification when the number of connections becomes greater than the configured number of threads that the application supports Configure the CloudWatch alarm to use the SNS topic.

D. Create an Amazon CloudWatch alarm that uses the AWS/ApplicationELB namespace RequestCountPerTarget metric Configure the CloudWatch alarm to send a notification when the average response time becomes greater than the longest response time that the application supports Configure the CloudWatch alarm to use the SNS topic

Answer: B

Explanation:

Option A is incorrect because creating an Amazon EventBridge rule that invokes an AWS Lambda function to query the applications on a 5-minute interval is not a valid solution. EventBridge rules can only trigger Lambda functions based on events, not on time intervals. Moreover, querying the applications on a 5-minute interval might incur unnecessary costs and network overhead, and might not detect performance issues in real time.

Option B is correct because creating an Amazon CloudWatch Synthetics canary that runs a custom script to query the applications on a 5-minute interval is a valid solution. CloudWatch Synthetics canaries are configurable scripts that monitor endpoints and APIs by simulating customer behavior. Canaries can run as often as once per minute, and can measure the latency and availability of the applications. Canaries can also send notifications to an Amazon SNS topic when they detect errors or performance issues1.

Option C is incorrect because creating an Amazon CloudWatch alarm that uses the AWS/ApplicationELB namespace RequestCountPerTarget metric is not a valid solution. The RequestCountPerTarget metric measures the number of requests completed or connections made per target in a target group2. This metric does not reflect the application response time, which is the requirement. Moreover, configuring the CloudWatch alarm to send a notification when the number of connections becomes greater than the configured number of threads that the application supports is not a valid way to measure the application performance, as it depends on the application design and implementation.

Option D is incorrect because creating an Amazon CloudWatch alarm that uses the AWS/ApplicationELB namespace RequestCountPerTarget metric is not a valid solution, for the same reason as option C. The RequestCountPerTarget metric does not reflect the application response time, which is the requirement. Moreover, configuring the CloudWatch alarm to send a notification when the average response time becomes greater than the longest response time that the application supports is not a valid way to measure the application performance, as it does not account for variability or outliers in the response time distribution.

References:

1: Using synthetic monitoring

2: Application Load Balancer metrics

Question: 150

A company's development team uses AVMS Cloud Formation to deploy its application resources The team must use for an changes to the environment The team cannot use AWS Management Console or the AWS CLI to make manual changes directly.

The team uses a developer IAM role to access the environment The role is configured with the Admnistratoraccess managed policy. The company has created a new Cloudformationdeployment IAM role that has the following policy.

www.atmicnetworks.com

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

The company wants ensure that only CloudFormation can use the new role. The development team cannot make any manual changes to the deployed resources.

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

A. Remove the AdministratorAccess policy. Assign the ReadOnlyAccess managed IAM policy to the developer role. Instruct the developers to use the CloudFormationDeployment role as a CloudFormation service role when the developers deploy new stacks.

B. Update the trust of CloudFormationDeployment role to allow the developer IAM role to assume the CloudFormationDepoyment role.

C. Configure the IAM to be to get and pass the CloudFormationDeployment role if cloudformation actions for resources,

D. Update the trust Of the CloudFormationDepoyment role to anow the cloudformation.amazonaws.com AWS principal to perform the iam:AssumeR01e action

E. Remove me Administratoraccess policy. Assign the ReadOnly/Access managed IAM policy to the developer role Instruct the developers to assume the CloudFormatondeployment role when the developers new stacks

F. Add an IAM policy to CloudFormationDeplyment to allow cloudformation * on an Add a policy that allows the iam.PassR01e action for ARN of if iam PassedT0Service equal cloudformation.amazonaws.com

---

Answer: ADF

Explanation:

A comprehensive and detailed explanation is:

Option A is correct because removing the AdministratorAccess policy and assigning the ReadOnlyAccess managed IAM policy to the developer role is a valid way to prevent the developers from making any manual changes to the deployed resources. The AdministratorAccess policy grants full access to all AWS resources and actions, which is not necessary for the developers. The ReadOnlyAccess policy grants read-only access to most AWS resources and actions, which is sufficient for the developers to view the status of their stacks. Instructing the developers to use the CloudFormationDeployment role as a CloudFormation service role when they deploy new stacks is also a valid way to ensure that only CloudFormation can use the new role. A CloudFormation service role is an IAM

role that allows CloudFormation to make calls to resources in a stack on behalf of the user1. The user can specify a service role when they create or update a stack, and CloudFormation will use that role's credentials for all operations that are performed on that stack1.

Option B is incorrect because updating the trust of CloudFormationDeployment role to allow the developer IAM role to assume the CloudFormationDeployment role is not a valid solution. This would allow the developers to manually assume the CloudFormationDeployment role and perform actions on the deployed resources, which is not what the company wants. The trust of CloudFormationDeployment role should only allow the cloudformation.amazonaws.com AWS principal to assume the role, as in option D.

Option C is incorrect because configuring the IAM user to be able to get and pass the CloudFormationDeployment role if cloudformation actions for resources is not a valid solution. This would allow the developers to manually pass the CloudFormationDeployment role to other services or resources, which is not what the company wants. The IAM user should only be able to pass the CloudFormationDeployment role as a service role when they create or update a stack with CloudFormation, as in option A.

Option D is correct because updating the trust of CloudFormationDeployment role to allow the cloudformation.amazonaws.com AWS principal to perform the iam:AssumeRole action is a valid solution. This allows CloudFormation to assume the CloudFormationDeployment role and access resources in other services on behalf of the user2. The trust policy of an IAM role defines which entities can assume the role2. By specifying cloudformation.amazonaws.com as the principal, you grant permission only to CloudFormation to assume this role.

Option E is incorrect because instructing the developers to assume the CloudFormationDeployment role when they deploy new stacks is not a valid solution. This would allow the developers to manually assume the CloudFormationDeployment role and perform actions on the deployed resources, which is not what the company wants. The developers should only use the CloudFormationDeployment role as a service role when they deploy new stacks with CloudFormation, as in option A.

Option F is correct because adding an IAM policy to CloudFormationDeployment that allows cloudformation:* on all resources and adding a policy that allows the iam:PassRole action for ARN of CloudFormationDeployment if iam:PassedToService equals cloudformation.amazonaws.com are valid solutions. The first policy grants permission for CloudFormationDeployment to perform any action with any resource using cloudformation.amazonaws.com as a service principals. The second policy grants permission for passing this role only if it is passed by cloudformation.amazonaws.com as a service principal. This ensures that only CloudFormation can use this role.

References:

1: AWS CloudFormation service roles

2: How to use trust policies with IAM roles

3: AWS::IAM::Policy

4: IAM: Pass an IAM role to a specific AWS service

Question: 151

A company wants to deploy a workload on several hundred Amazon EC2 instances. The company will provision the EC2 instances in an Auto Scaling group by using a launch template.

The workload will pull files from an Amazon S3 bucket, process the data, and put the results into a different S3 bucket. The EC2 instances must have least-privilege permissions and must use temporary security credentials.

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

A. Create an IAM role that has the appropriate permissions for S3 buckets. Add the IAM role to an instance profile.

B. Update the launch template to include the IAM instance profile.

C. Create an IAM user that has the appropriate permissions for Amazon S3. Generate a secret key and token.

D. Create a trust anchor and profile. Attach the IAM role to the profile.

E. Update the launch template. Modify the user data to use the new secret key and token.

Answer: AB

Explanation:

To meet the requirements of deploying a workload on several hundred EC2 instances with leastprivilege permissions and temporary security credentials, the company should use an IAM role and an instance profile. An IAM role is a way to grant permissions to an entity that you trust, such as an EC2 instance. An instance profile is a container for an IAM role that you can use to pass role information to an EC2 instance when the instance starts. By using an IAM role and an instance profile, the EC2 instances can automatically receive temporary security credentials from the AWS Security Token Service (STS) and use them to access the S3 buckets. This way, the company does not need to manage or rotate any long-term credentials, such as IAM users or access keys.

To use an IAM role and an instance profile, the company should create an IAM role that has the appropriate permissions for S3 buckets. The permissions should allow the EC2 instances to read from the source S3 bucket and write to the destination S3 bucket. The company should also create a trust policy for the IAM role that specifies that EC2 is allowed to assume the role. Then, the company should add the IAM role to an instance profile. An instance profile can have only one IAM role, so the company does not need to create multiple roles or profiles for this scenario.

Next, the company should update the launch template to include the IAM instance profile. A launch template is a way to save launch parameters for EC2 instances, such as the instance type, security group, user data, and IAM instance profile. By using a launch template, the company can ensure that all EC2 instances in the Auto Scaling group have consistent configuration and permissions. The company should specify the name or ARN of the IAM instance profile in the launch template. This way, when the Auto Scaling group launches new EC2 instances based on the launch template, they will automatically receive the IAM role and its permissions through the instance profile.

The other options are not correct because they do not meet the requirements or follow best practices. Creating an IAM user and generating a secret key and token is not a good option because it involves managing long-term credentials that need to be rotated regularly. Moreover, embedding credentials in user data is not secure because user data is visible to anyone who can describe the EC2 instance. Creating a trust anchor and profile is not a valid option because trust anchors are used for certificate-based authentication, not for IAM roles or instance profiles. Modifying user data to use a new secret key and token is also not a good option because it requires updating user data every time the credentials change, which is not scalable or efficient.

References:

1: AWS Certified DevOps Engineer - Professional Certification | AWS Certification | AWS

2: DevOps Resources - Amazon Web Services (AWS)

3: Exam Readiness: AWS Certified DevOps Engineer - Professional

: IAM Roles for Amazon EC2 - AWS Identity and Access Management : Working with Instance Profiles - AWS Identity and Access Management : Launching an Instance Using a Launch Template - Amazon Elastic Compute Cloud : Temporary Security Credentials - AWS Identity and Access Management

Question: 152

A company needs to ensure that flow logs remain configured for all existing and new VPCs in its AWS account. The company uses an AWS CloudFormation stack to manage its VPCs. The company needs a solution that will work for any VPCs that any IAM user creates.

Which solution will meet these requirements?

A. Add the resource to the CloudFormation stack that creates the VPCs.

B. Create an organization in AWS Organizations. Add the company's AWS account to the organization. Create an SCP to prevent users from modifying VPC flow logs.

C. Turn on AWS Config. Create an AWS Config rule to check whether VPC flow logs are turned on. Configure automatic remediation to turn on VPC flow logs.

D. Create an IAM policy to deny the use of API calls for VPC flow logs. Attach the IAM policy to all IAM users.

_____ Answer: C

Explanation:

To meet the requirements of ensuring that flow logs remain configured for all existing and new VPCs in the AWS account, the company should use AWS Config and automatic remediation. AWS Config is a service that enables customers to assess, audit, and evaluate the configurations of their AWS resources. AWS Config continuously monitors and records the configuration changes of the AWS resources and evaluates them against desired configurations. Customers can use AWS Config rules to define the desired configuration state of their AWS resources and trigger actions when a resource configuration violates a rule.

One of the AWS Config rules that customers can use is vpc-flow-logs-enabled, which checks whether VPC flow logs are enabled for all VPCs in an AWS account. Customers can also configure automatic remediation for this rule, which means that AWS Config will automatically enable VPC flow logs for any VPCs that do not have them enabled. Customers can specify the destination (CloudWatch Logs or S3) and the traffic type (all, accept, or reject) for the flow logs as remediation parameters. By using AWS Config and automatic remediation, the company can ensure that flow logs remain configured for all existing and new VPCs in its AWS account, regardless of who creates them or how they are created.

The other options are not correct because they do not meet the requirements or follow best practices. Adding the resource to the CloudFormation stack that creates the VPCs is not a sufficient solution because it will only work for VPCs that are created by using the CloudFormation stack. It will not work for VPCs that are created by using other methods, such as the console or the API. Creating an organization in AWS Organizations and creating an SCP to prevent users from modifying VPC flow logs is not a good solution because it will not ensure that flow logs are enabled for all VPCs in the first place. It will only prevent users from disabling or changing flow logs after they are enabled. Creating an IAM policy to deny the use of API calls for VPC flow logs and attaching it to all IAM users is not a valid solution because it will prevent users from enabling or disabling flow logs at all. It will also not work for VPCs that are created by using other methods, such as the console or CloudFormation. References:

1: AWS::EC2::FlowLog - AWS CloudFormation

2: Amazon VPC Flow Logs extends CloudFormation Support to custom format subscriptions, 1-minute aggregation intervals and tagging

3: Logging IP traffic using VPC Flow Logs - Amazon Virtual Private Cloud

: About AWS Config - AWS Config

: vpc-flow-logs-enabled - AWS Config

: Remediate Noncompliant Resources with AWS Config Rules - AWS Config

Question: 153

A company is migrating its on-premises Windows applications and Linux applications to AWS. The company will use automation to launch Amazon EC2 instances to mirror the on-premises configurations. The migrated applications require access to shared storage that uses SMB for Windows and NFS for Linux.

The company is also creating a pilot light disaster recovery (DR) environment in another AWS Region. The company will use automation to launch and configure the EC2 instances in the DR Region. The company needs to replicate the storage to the DR Region.

Which storage solution will meet these requirements?

A. Use Amazon S3 for the application storage. Create an S3 bucket in the primary Region and an S3 bucket in the DR Region. Configure S3 Cross-Region Replication (CRR) from the primary Region to the DR Region.

B. Use Amazon Elastic Block Store (Amazon EBS) for the application storage. Create a backup plan in AWS Backup that creates snapshots of the EBS volumes that are in the primary Region and replicates the snapshots to the DR Region.

C. Use a Volume Gateway in AWS Storage Gateway for the application storage. Configure CrossRegion Replication (CRR) of the Volume Gateway from the primary Region to the DR Region.

D. Use Amazon FSx for NetApp ONTAP for the application storage. Create an FSx for ONTAP instance in the DR Region. Configure NetApp SnapMirror replication from the primary Region to the DR Region.

Answer: D

Explanation:

To meet the requirements of migrating its on-premises Windows and Linux applications to AWS and creating a pilot light DR environment in another AWS Region, the company should use Amazon FSx for NetApp ONTAP for the application storage. Amazon FSx for NetApp ONTAP is a fully managed service that provides highly reliable, scalable, high-performing, and feature-rich file storage built on NetApp's popular ONTAP file system. FSx for ONTAP supports multiple protocols, including SMB for Windows and NFS for Linux, so the company can access the shared storage from both types of applications. FSx for ONTAP also supports NetApp SnapMirror replication, which enables the company to replicate the storage to the DR Region. NetApp SnapMirror replication is efficient, secure, and incremental, and it preserves the data deduplication and compression benefits of FSx for ONTAP. The company can use automation to launch and configure the EC2 instances in the DR Region and then use NetApp SnapMirror to restore the data from the primary Region.
The other options are not correct because they do not meet the requirements or follow best practices. Using Amazon S3 for the application storage is not a good option because S3 is an object storage service that does not support SMB or NFS protocols natively. The company would need to use additional services or software to mount S3 buckets as file systems, which would add complexity and cost. Using Amazon EBS for the application storage is also not a good option because EBS is a block storage service that does not support SMB or NFS protocols natively. The company would need to set up and manage file servers on EC2 instances to provide shared access to the EBS volumes, which would add overhead and maintenance. Using a Volume Gateway in AWS Storage Gateway for the application storage is not a valid option because Volume Gateway does not support SMB protocol. Volume Gateway only supports iSCSI protocol, which means that only Linux applications can access the shared storage.

References:

1: What is Amazon FSx for NetApp ONTAP? - FSx for ONTAP
2: Amazon FSx for NetApp ONTAP
3: Amazon FSx for NetApp ONTAP | NetApp
4: AWS Announces General Availability of Amazon FSx for NetApp ONTAP
: Replicating Data with NetApp SnapMirror - FSx for ONTAP
: What Is Amazon S3? - Amazon Simple Storage Service
: What Is Amazon Elastic Block Store (Amazon EBS)? - Amazon Elastic Compute Cloud
: What Is AWS Storage Gateway? - AWS Storage Gateway

Question: 154

A company deploys a web application on Amazon EC2 instances that are behind an Application Load Balancer (ALB). The company stores

the application code in an AWS CodeCommit repository. When code is merged to the main branch, an AWS Lambda function invokes an AWS CodeBuild project. The CodeBuild project packages the code, stores the packaged code in AWS CodeArtifact, and invokes AWS Systems Manager Run Command to deploy the packaged code to the EC2 instances.

Previous deployments have resulted in defects, EC2 instances that are not running the latest version of the packaged code, and inconsistencies between instances.

Which combination of actions should a DevOps engineer take to implement a more reliable deployment solution? (Select TWO.)

A. Create a pipeline in AWS CodePipeline that uses the CodeCommit repository as a source provider. Configure pipeline stages that run the CodeBuild project in parallel to build and test the application. In the pipeline, pass the CodeBuild project output artifact to an AWS CodeDeploy action.

B. Create a pipeline in AWS CodePipeline that uses the CodeCommit repository as a source provider. Create separate pipeline stages that run a CodeBuild project to build and then test the application. In the pipeline, pass the CodeBuild project output artifact to an AWS CodeDeploy action.

C. Create an AWS CodeDeploy application and a deployment group to deploy the packaged code to the EC2 instances. Configure the ALB for the deployment group.

D. Create individual Lambda functions that use AWS CodeDeploy instead of Systems Manager to run build, test, and deploy actions.

E. Create an Amazon S3 bucket. Modify the CodeBuild project to store the packages in the S3 bucket instead of in CodeArtifact. Use deploy actions in CodeDeploy to deploy the artifact to the EC2 instances.

Answer: AC

Explanation:

To implement a more reliable deployment solution, a DevOps engineer should take the following actions:
Create a pipeline in AWS CodePipeline that uses the CodeCommit repository as a source provider. Configure pipeline stages that run the CodeBuild project in parallel to build and test the application. In the pipeline, pass the CodeBuild project output artifact to an AWS CodeDeploy action. This action will improve the deployment reliability by automating the entire process from code commit to deployment, reducing human errors and inconsistencies. By running the build and test stages in parallel, the pipeline can also speed up the delivery time and provide faster feedback. By using CodeDeploy as the deployment action, the pipeline can leverage the features of CodeDeploy, such as traffic shifting, health checks, rollback, and deployment configuration123
Create an AWS CodeDeploy application and a deployment group to deploy the packaged code to the EC2 instances. Configure the ALB for the deployment group. This action will improve the deployment reliability by using CodeDeploy to orchestrate the deployment across multiple EC2 instances behind an ALB. CodeDeploy can perform blue/green deployments or in-place deployments with traffic shifting, which can minimize downtime and reduce risks. CodeDeploy can also monitor the health of the instances during and after the deployment, and automatically roll back if any issues are detected. By configuring the ALB for the deployment group, CodeDeploy can register and deregister instances from the load balancer as needed, ensuring that only healthy instances receive traffic45 The other options are not correct because they do not improve the deployment reliability or follow best practices. Creating separate pipeline stages that run a CodeBuild project to build and then test the application is not a good option because it will increase the pipeline execution time and delay the feedback loop. Creating individual Lambda functions that use CodeDeploy instead of Systems Manager to run build, test, and deploy actions is not a valid option because it will add unnecessary complexity and cost to the solution. Lambda functions are not designed for long-running tasks such as building or deploying applications. Creating an Amazon S3 bucket and modifying the

CodeBuild project to store the packages in the S3 bucket instead of in CodeArtifact is not a necessary option because it will not affect the deployment reliability. CodeArtifact is a secure, scalable, and costeffective package management service that can store and share software packages for application development67

References:

1: What is AWS CodePipeline? - AWS CodePipeline
2: Create a pipeline in AWS CodePipeline - AWS CodePipeline
3: Deploy an application with AWS CodeDeploy - AWS CodePipeline
4: What is AWS CodeDeploy? - AWS CodeDeploy
5: Configure an Application Load Balancer for your blue/green deployments - AWS CodeDeploy 6: What is AWS Lambda? - AWS Lambda
7: What is AWS CodeArtifact? - AWS CodeArtifact

Question: 155

A company is launching an application. The application must use only approved AWS services. The account that runs the application was created less than 1 year ago and is assigned to an AWS Organizations OU.

The company needs to create a new Organizations account structure. The account structure must have an appropriate SCP that supports the use of only services that are currently active in the AWS account.

The company will use AWS Identity and Access Management (IAM) Access Analyzer in the solution.

Which solution will meet these requirements?

A. Create an SCP that allows the services that IAM Access Analyzer identifies. Create an OU for the account. Move the account into the new OU. Attach the new SCP to the new OU. Detach the default FullAWSAccess SCP from the new OU.

B. Create an SCP that denies the services that IAM Access Analyzer identifies. Create an OU for the account. Move the account into the new OIJ. Attach the new SCP to the new OU.

C. Create an SCP that allows the services that IAM Access Analyzer identifies. Attach the new SCP to the organization's root.

D. Create an SCP that allows the services that IAM Access Analyzer identifies. Create an OU for the account. Move the account into the new OU. Attach the new SCP to the management account. Detach the default FullAWSAccess SCP from the new OU.

Answer: A

Explanation:

To meet the requirements of creating a new Organizations account structure with an appropriate SCP that supports the use of only services that are currently active in the AWS account, the company should use the following solution:
Create an SCP that allows the services that IAM Access Analyzer identifies. IAM Access Analyzer is a service that helps identify potential resource-access risks by analyzing resource-based policies in the AWS environment. IAM Access Analyzer can also generate IAM policies based on access activity in the AWS CloudTrail logs. By using IAM Access Analyzer, the company can create an SCP that grants only the permissions that are required for the application to run, and denies all other services. This way, the company can enforce the use of only approved AWS services and reduce the risk of unauthorized access12
Create an OU for the account. Move the account into the new OU. An OU is a container for accounts within an organization that enables you to group accounts that have similar business or security requirements. By creating an OU for the account, the company can apply policies and manage settings for the account as a group. The company should move the account into the new OU to make it subject to

the policies attached to the OU3

Attach the new SCP to the new OU. Detach the default FullAWSAccess SCP from the new OU. An SCP is a type of policy that specifies the maximum permissions for an organization or organizational unit (OU). By attaching the new SCP to the new OU, the company can restrict the services that are available to all accounts in that OU, including the account that runs the application. The company should also detach the default FullAWSAccess SCP from the new OU, because this policy allows all actions on all AWS services and might override or conflict with the new SCP45

The other options are not correct because they do not meet the requirements or follow best practices. Creating an SCP that denies the services that IAM Access Analyzer identifies is not a good option because it might not cover all possible services that are not approved or required for the application. A deny policy is also more difficult to maintain and update than an allow policy. Creating an SCP that allows the services that IAM Access Analyzer identifies and attaching it to the organization's root is not a good option because it might affect other accounts and OUs in the organization that have different service requirements or approvals. Creating an SCP that allows the services that IAM Access Analyzer identifies and attaching it to the management account is not a valid option because SCPs cannot be attached directly to accounts, only to OUs or roots. References:

1: Using AWS Identity and Access Management Access Analyzer - AWS Identity and Access Management
2: Generate a policy based on access activity - AWS Identity and Access Management
3: Organizing your accounts into OUs - AWS Organizations
4: Service control policies - AWS Organizations
5: How SCPs work - AWS Organizations

Question: 156

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.

_____ Answer: C

Explanation:

Question: 157

A company needs to implement failover for its application. The application includes an Amazon CloudFront distribution and a public Application Load Balancer (ALB) in an AWS Region. The company has configured the ALB as the default origin for the distribution.

After some recent application outages, the company wants a zero-second RTO. The company deploys the application to a secondary Region in a warm standby configuration. A DevOps engineer needs to automate the failover of the application to the secondary Region so that HTTP GET requests meet the desired R TO.

Which solution will meet these requirements?

A. Create a second CloudFront distribution that has the secondary ALB as the default origin. Create Amazon Route 53 alias records that have a failover policy and Evaluate Target Health set to Yes for both CloudFront distributions. Update the application to use the new record set.

B. Create a new origin on the distribution for the secondary ALB. Create a new origin group. Set the original ALB as the primary origin. Configure the origin group to fail over for HTTP 5xx status codes. Update the default behavior to use the origin group.

C. Create Amazon Route 53 alias records that have a failover policy and Evaluate Target Health set to Yes for both ALBs. Set the TTL of both records to O. Update the distribution's origin to use the new record set.

D. Create a CloudFront function that detects HTTP 5xx status codes. Configure the function to return a 307 Temporary Redirect error response to the secondary ALB if the function detects 5xx status codes. Update the distribution's default behavior to send origin responses to the function.

Answer: B

Explanation:

To implement failover for the application to the secondary Region so that HTTP GET requests meet the desired RTO, the DevOps engineer should use the following solution:

Create a new origin on the distribution for the secondary ALB. A CloudFront origin is the source of the content that CloudFront delivers to viewers. By creating a new origin for the secondary ALB, the DevOps engineer can configure CloudFront to route traffic to the secondary Region when the primary Region is unavailable1

Create a new origin group. Set the original ALB as the primary origin. Configure the origin group to fail over for HTTP 5xx status codes. An origin group is a logical grouping of two origins: a primary origin and a secondary origin. By creating an origin group, the DevOps engineer can specify which origin CloudFront should use as a fallback when the primary origin fails. The DevOps engineer can also define which HTTP status codes should trigger a failover from the primary origin to the secondary origin. By setting the original ALB as the primary origin and configuring the origin group to fail over for HTTP 5xx status codes, the DevOps engineer can ensure that CloudFront will switch to the secondary ALB when the primary ALB returns server errors2

Update the default behavior to use the origin group. A behavior is a set of rules that CloudFront applies when it receives requests for specific URLs or file types. The default behavior applies to all requests that do not match any other behaviors. By updating the default behavior to use the origin group, the DevOps engineer can enable failover routing for all requests that are sent to the distribution3

This solution will meet the requirements because it will automate the failover of the application to the secondary Region with zero-second RTO. When CloudFront receives an HTTP GET request, it will first try to route it to the primary ALB in the primary Region. If the primary ALB is healthy and returns a successful response, CloudFront will deliver it to the viewer. If the primary ALB is unhealthy or returns an HTTP 5xx status code, CloudFront will automatically route the request to the secondary ALB in the secondary Region and deliver its response to the viewer.

The other options are not correct because they either do not provide zero-second RTO or do not work as expected. Creating a second CloudFront distribution that has the secondary ALB as the default origin and creating Amazon Route 53 alias records that have a failover policy is not a good option because it will introduce additional latency and complexity to the solution. Route 53 health checks and DNS propagation can take several minutes or longer, which means that viewers might experience delays or errors when accessing the application during a failover event. Creating Amazon Route 53 alias records that have a failover policy and Evaluate Target Health set to Yes for both ALBs and setting the TTL of both records to O is not a valid option because it will not work with CloudFront distributions. Route 53 does not support health checks for alias records that point to CloudFront distributions, so it cannot detect if an ALB behind a distribution is healthy or not. Creating a CloudFront function that detects HTTP 5xx status codes and returns a 307 Temporary Redirect error response to the secondary ALB is not a valid option because it will not provide zero-second RTO. A 307 Temporary Redirect error response tells viewers to retry their requests with a different URL, which means that viewers will have to make an additional request and wait for another response from CloudFront before reaching the secondary ALB.

References:

1: Adding, Editing, and Deleting Origins - Amazon CloudFront

2: Configuring Origin Failover - Amazon CloudFront

3: Creating or Updating a Cache Behavior - Amazon CloudFront

Question: 158

A company runs its container workloads in AWS App Runner. A DevOps engineer manages the company's container repository in Amazon Elastic Container Registry (Amazon ECR).

The DevOps engineer must implement a solution that continuously monitors the container repository. The solution must create a new container image when the solution detects an operating system vulnerability or language package vulnerability.

Which solution will meet these requirements?

A. Use EC2 Image Builder to create a container image pipeline. Use Amazon ECR as the target repository. Turn on enhanced scanning on the ECR repository. Create an Amazon EventBridge rule to capture an Inspector2 finding event. Use the event to invoke the image pipeline. Re-upload the container to the repository.

B. Use EC2 Image Builder to create a container image pipeline. Use Amazon ECR as the target repository. Enable Amazon GuardDuty Malware Protection on the container workload. Create an Amazon EventBridge rule to capture a GuardDuty finding event. Use the event to invoke the image pipeline.

C. Create an AWS CodeBuild project to create a container image. Use Amazon ECR as the target repository. Turn on basic scanning on the repository. Create an Amazon EventBridge rule to capture an ECR image action event. Use the event to invoke the CodeBuild project. Re-upload the container to the repository.

D. Create an AWS CodeBuild project to create a container image. Use Amazon ECR as the target repository. Configure AWS Systems Manager Compliance to scan all managed nodes. Create an Amazon EventBridge rule to capture a configuration compliance state change event. Use the event to invoke the CodeBuild project.

Answer: A

Explanation:

The solution that meets the requirements is to use EC2 Image Builder to create a container image pipeline, use Amazon ECR as the target repository, turn on enhanced scanning on the ECR repository, create an Amazon EventBridge rule to capture an Inspector2 finding event, and use the event to invoke the image pipeline. Re-upload the container to the repository.

www.atmicnetworks.com

This solution will continuously monitor the container repository for vulnerabilities using enhanced scanning, which is a feature of Amazon ECR that provides detailed information and guidance on how to fix security issues found in your container images. Enhanced scanning uses Inspector2, a security assessment service that integrates with Amazon ECR and generates findings for any vulnerabilities detected in your images. You can use Amazon EventBridge to create a rule that triggers an action when an Inspector2 finding event occurs. The action can be to invoke an EC2 Image Builder pipeline, which is a service that automates the creation of container images. The pipeline can use the latest patches and updates to build a new container image and upload it to the same ECR repository, replacing the vulnerable image.

The other options are not correct because they do not meet all the requirements or use services that are not relevant for the scenario.

Option B is not correct because it uses Amazon GuardDuty Malware Protection, which is a feature of GuardDuty that detects malicious activity and unauthorized behavior on your AWS accounts and resources. GuardDuty does not scan container images for vulnerabilities, nor does it integrate with Amazon ECR or EC2 Image Builder.

Option C is not correct because it uses basic scanning on the ECR repository, which only provides a summary of the vulnerabilities found in your container images. Basic scanning does not use Inspector2 or generate findings that can be captured by Amazon EventBridge. Moreover, basic scanning does not provide guidance on how to fix the vulnerabilities.

Option D is not correct because it uses AWS Systems Manager Compliance, which is a feature of Systems Manager that helps you monitor and manage the compliance status of your AWS resources based on AWS Config rules and AWS Security Hub standards. Systems Manager Compliance does not scan container images for vulnerabilities, nor does it integrate with Amazon ECR or EC2 Image Builder.

Question: 159

A DevOps engineer is building an application that uses an AWS Lambda function to query an Amazon Aurora MySQL DB cluster. The Lambda function performs only read queries. Amazon EventBridge events invoke the Lambda function.

As more events invoke the Lambda function each second, the database's latency increases and the database's throughput decreases. The DevOps engineer needs to improve the performance of the application.

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

A. Use Amazon RDS Proxy to create a proxy. Connect the proxy to the Aurora cluster reader endpoint. Set a maximum connections percentage on the proxy.

B. Implement database connection pooling inside the Lambda code. Set a maximum number of connections on the database connection pool.

C. Implement the database connection opening outside the Lambda event handler code.

D. Implement the database connection opening and closing inside the Lambda event handler code.

E. Connect to the proxy endpoint from the Lambda function.

F. Connect to the Aurora cluster endpoint from the Lambda function.

www.atmicnetworks.com

Answer: ACE

Explanation:

Verified Answer: A, C, and E.

Explanation:

Short To improve the performance of the application, the DevOps engineer should use Amazon RDS Proxy, implement the database connection opening outside the Lambda event handler code, and connect to the proxy endpoint from the Lambda function.

Reference:

Amazon RDS Proxy is a fully managed, highly available database proxy for Amazon Relational Database Service (RDS) that makes applications more scalable, more resilient to database failures, and more secure1. By using Amazon RDS Proxy, the DevOps engineer can reduce the overhead of opening and closing connections to the database, which can improve latency and throughput2.

The DevOps engineer should connect the proxy to the Aurora cluster reader endpoint, which allows read-only connections to one of the Aurora Replicas in the DB cluster3. This can help balance the load across multiple read replicas and improve performance for read-intensive workloads4.

The DevOps engineer should implement the database connection opening outside the Lambda event handler code, which means using a global variable to store the database connection object5. This can enable connection reuse across multiple invocations of the Lambda function, which can reduce latency and improve performance.

The DevOps engineer should connect to the proxy endpoint from the Lambda function, which is a unique URL that represents the proxy. This can allow the Lambda function to access the database through the proxy, which can provide benefits such as connection pooling, load balancing, failover handling, and enhanced security.

The other options are incorrect because:

Implementing database connection pooling inside the Lambda code is unnecessary and redundant when using Amazon RDS Proxy, which already provides connection pooling as a service.

Implementing the database connection opening and closing inside the Lambda event handler code is inefficient and costly, as it can increase latency and consume more resources for each invocation of the Lambda function.

Connecting to the Aurora cluster endpoint from the Lambda function is not optimal for read-only queries, as it can direct traffic to either the primary instance or one of the Aurora Replicas in the DB cluster. This can result in inconsistent performance and potential conflicts with write operations on the primary instance.

Question: 160

A company's application teams use AWS CodeCommit repositories for their applications. The application teams have repositories in multiple AWS accounts. All accounts are in an organization in AWS Organizations.

Each application team uses AWS IAM Identity Center (AWS Single Sign-On) configured with an external IdP to assume a developer IAM role. The developer role allows the application teams to use Git to work with the code in the repositories.

A security audit reveals that the application teams can modify the main branch in any repository. A DevOps engineer must implement a solution that allows the application teams to modify the main branch of only the repositories that they manage.

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

A. Update the SAML assertion to pass the user's team name. Update the IAM role's trust policy to add an access-team session tag that has the team name.

B. Create an approval rule template for each team in the Organizations management account. Associate the template with all the

repositories. Add the developer role ARN as an approver.

C. Create an approval rule template for each account. Associate the template with all repositories. Add the "aws:ResourceTag/access-team":"$;{aws:PrincipalTag/access-team}" condition to the approval rule template.

D. For each CodeCommit repository, add an access-team tag that has the value set to the name of the associated team.

E. Attach an SCP to the accounts. Include the following statement:

```
{
    "Effect": "Deny",
    "Action": [
        "codecommit:GitPush",
        "codecommit:PutFile",
        "codecommit:Merge*"
    1,
    "Resource":
    "Condition": {
        "StringEqualsIfExists":   {
            "codecommit:References": ["refs/heads/main"]
        },
        "StringNotEquals": {
            "aws:ResourceTag/access-team": "$ ;(aws:PrincipalTag/access-team}" },
        "Null": {
            "codecommit:References": "false"
        }
```

F. Create an IAM permissions boundary in each account. Include the following statement:

```
"Effect": "Allow",
"Action": [
    "codecommit:GitPush",
    "codecommit:PutFile",
    "codecommit:Merge*"
],
"Resource": "*",
"Condition": {
    "StringEqualsIfExists": {
        "codecommit:References":    ["refs/heads/main"]
    ),
    "StringNotEquals": {
        "aws:ResourceTag/access-team": "$ ;(aws:PrincipalTag/access-team}"
    ),
    "Null": {
        "codecommit:References": "false"
    )
```

Answer: ADF

Explanation:

Short To meet the requirements, the DevOps engineer should update the SAML assertion to pass the user's team name, update the IAM role's trust policy to add an access-team session tag that has the team name, create an IAM permissions boundary in each account, and for each CodeCommit repository, add an access-team tag that has the value set to the name of the associated team. Reference:
Updating the SAML assertion to pass the user's team name allows the DevOps engineer to use IAM tags to identify which team a user belongs to. This can help enforce fine-grained access control based on the user's team membership!
Updating the IAM role's trust policy to add an access-team session tag that has the team name allows the DevOps engineer to use IAM condition keys to restrict access based on the session tag value2. For example, the DevOps engineer can use the aws:PrincipalTag condition key to match the access-team tag of the user with the access-team tag of the repository3.
Creating an IAM permissions boundary in each account allows the DevOps engineer to set the maximum permissions that an identity-based policy can grant to an IAM entity. An entity's permissions boundary allows it to perform only the actions that are allowed by both its identitybased policies and its permissions boundaries4. For example, the DevOps engineer can use a permissions boundary policy to limit the actions that a user can perform on CodeCommit repositories based on their access-team tag5.
For each CodeCommit repository, adding an access-team tag that has the value set to the name of the associated team allows the DevOps engineer to use resource tags to identify which team manages a repository. This can help enforce fine-grained access control based on the resource tag value6.
The other options are incorrect because:
Creating an approval rule template for each team in the Organizations management account is not a valid option, as approval rule templates are not supported by AWS Organizations. Approval rule templates are specific to CodeCommit and can only be associated with one or more repositories in the same AWS Region where they are created7.
Creating an approval rule template for each account is not a valid option, as approval rule templates are not designed to restrict access to modify branches. Approval rule templates are designed to require approvals from specified users or groups before merging pull requests8.
Attaching an SCP to the accounts is not a valid option, as SCPs are not designed to restrict access based on tags. SCPs are designed to restrict access based on service actions and resources across all users and roles in an organization's account9.

Question: 161

A company needs a strategy for failover and disaster recovery of its data and application. The application uses a MySQL database and Amazon EC2 instances. The company requires a maximum RPO of 2 hours and a maximum RTO of 10 minutes for its data and application at all times.

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

A. Create an Amazon Aurora Single-AZ cluster in multiple AWS Regions as the data store. Use Aurora's automatic recovery capabilities in the event of a disaster.

B. Create an Amazon Aurora global database in two AWS Regions as the data store. In the event of a failure, promote the secondary Region to the primary for the application. Update the application to use the Aurora cluster endpoint in the secondary Region.

C. Create an Amazon Aurora cluster in multiple AWS Regions as the data store. Use a Network Load Balancer to balance the database traffic in different Regions.

D. Set up the application in two AWS Regions. Use Amazon Route 53 failover routing that points to Application Load Balancers in both Regions. Use health checks and Auto Scaling groups in each Region.

E. Set up the application in two AWS Regions. Configure AWS Global Accelerator to point to Application Load Balancers (ALBs) in both

Regions. Add both ALBs to a single endpoint group. Use health checks and Auto Scaling groups in each Region.

Answer: BE

Explanation:

Verified Answer: B and E.

Explanation:

Short To meet the requirements of failover and disaster recovery, the company should use the following deployment strategies:

Create an Amazon Aurora global database in two AWS Regions as the data store. In the event of a failure, promote the secondary Region to the primary for the application. Update the application to use the Aurora cluster endpoint in the secondary Region. This strategy can provide a low RPO and RTO for the data, as Aurora global database replicates data with minimal latency across Regions and allows fast and easy failover12. The company can use the Amazon Aurora cluster endpoint to connect to the current primary DB cluster without needing to change any application code1.

Set up the application in two AWS Regions. Configure AWS Global Accelerator to point to Application Load Balancers (ALBs) in both Regions. Add both ALBs to a single endpoint group. Use health checks and Auto Scaling groups in each Region. This strategy can provide high availability and performance for the application, as AWS Global Accelerator uses the AWS global network to route traffic to the closest healthy endpoint3. The company can also use static IP addresses that are assigned by Global Accelerator as a fixed entry point for their application!. By using health checks and Auto Scaling groups, the company can ensure that their application can scale up or down based on demand and handle any instance failures4.

The other options are incorrect because:

Creating an Amazon Aurora Single-AZ cluster in multiple AWS Regions as the data store would not provide a fast failover or disaster recovery solution, as the company would need to manually restore data from backups or snapshots in another Region in case of a failure.

Creating an Amazon Aurora cluster in multiple AWS Regions as the data store and using a Network Load Balancer to balance the database traffic in different Regions would not work, as Network Load Balancers do not support cross-Region routing. Moreover, this strategy would not provide a consistent view of the data across Regions, as Aurora clusters do not replicate data automatically between Regions unless they are part of a global database.

Setting up the application in two AWS Regions and using Amazon Route 53 failover routing that points to Application Load Balancers in both Regions would not provide a low RTO, as Route 53 failover routing relies on DNS resolution, which can take time to propagate changes across different DNS servers and clients. Moreover, this strategy would not provide deterministic routing, as Route 53

failover routing depends on DNS caching behavior, which can vary depending on different factors.

Question: 162

A company uses AWS Directory Service for Microsoft Active Directory as its identity provider (IdP). The company requires all infrastructure to be

defined and deployed by AWS CloudFormation.

A DevOps engineer needs to create a fleet of Windows-based Amazon EC2 instances to host an application. The DevOps engineer has created a

CloudFormation template that contains an EC2 launch template, IAM role, EC2 security group, and EC2 Auto Scaling group. The DevOps engineer must implement a solution that joins all EC2 instances to the domain of the AWS Managed Microsoft AD directory.

Which solution will meet these requirements with the MOST operational efficiency?

A. In the CloudFormation template, create an AWS::SSM::Document resource that joins the EC2 instance to the AWS Managed Microsoft AD domain by using the parameters for the existing directory. Update the launch template to include the SSMAssociation property to use the new SSM document. Attach the AmazonSSMManagedInstanceCore and AmazonSSMDirectoryServiceAccess AWS managed policies to the IAM role that the EC2 instances use.

B. In the CloudFormation template, update the launch template to include specific tags that propagate on launch. Create an AWS::SSM::Association resource to associate the AWS- JoinDirectoryServiceDomain Automation runbook with the EC2 instances that have the specified tags. Define the required parameters to join the AWS Managed Microsoft AD directory. Attach the AmazonSSMManagedInstanceCore and AmazonSSMDirectoryServiceAccess AWS managed policies to the IAM role that the EC2 instances use.

C. Store the existing AWS Managed Microsoft AD domain connection details in AWS Secrets Manager. In the CloudFormation template, create an AWS::SSM::Association resource to associate the AWS- CreateManagedWindowsInstanceWithApproval Automation runbook with the EC2 Auto Scaling group. Pass the ARNs for the parameters from Secrets Manager to join the domain. Attach the AmazonSSMDirectoryServiceAccess and SecretsManagerReadWrite AWS managed policies to the IAM role that the EC2 instances use.

D. Store the existing AWS Managed Microsoft AD domain administrator credentials in AWS Secrets Manager. In the CloudFormation template, update the EC2 launch template to include user data. Configure the user data to pull the administrator credentials from Secrets Manager and to join the AWS Managed Microsoft AD domain. Attach the AmazonSSMManagedInstanceCore and SecretsManagerReadWrite AWS managed policies to the IAM role that the EC2 instances use.

Answer: B

Explanation:

To meet the requirements, the DevOps engineer needs to create a solution that joins all EC2 instances to the domain of the AWS Managed Microsoft AD directory with the most operational efficiency. The DevOps engineer can use AWS Systems Manager Automation to automate the domain join process using an existing runbook called AWS-JoinDirectoryServiceDomain. This runbook can join Windows instances to an AWS Managed Microsoft AD or Simple AD directory by using PowerShell commands. The DevOps engineer can create an AWS::SSM::Association resource in the CloudFormation template to associate the runbook with the EC2 instances that have specific tags. The tags can be defined in the launch template and propagated on launch to the EC2 instances. The DevOps engineer can also define the required parameters for the runbook, such as the directory ID, directory name, and organizational unit. The DevOps engineer can attach the AmazonSSMManagedInstanceCore and AmazonSSMDirectoryServiceAccess AWS managed policies to the IAM role that the EC2 instances use. These policies grant the necessary permissions for Systems Manager and Directory Service operations.

Question: 163

A company has multiple development groups working in a single shared AWS account. The Senior Manager of the groups wants to be alerted via a third-party API call when the creation of resources approaches the service limits for the account.
Which solution will accomplish this with the LEAST amount of development effort?

A. Create an Amazon CloudWatch Event rule that runs periodically and targets an AWS Lambda function. Within the Lambda function, evaluate the current state of the AWS environment and compare deployed resource values to resource limits on the account. Notify the Senior Manager if the account is approaching a service limit.

B. Deploy an AWS Lambda function that refreshes AWS Trusted Advisor checks, and configure an Amazon CloudWatch Events rule to run the Lambda function periodically. Create another CloudWatch Events rule with an event pattern matching Trusted Advisor events and a target Lambda function. In the target Lambda function, notify the Senior Manager.

C. Deploy an AWS Lambda function that refreshes AWS Personal Health Dashboard checks, and configure an Amazon CloudWatch

Events rule to run the Lambda function periodically. Create another CloudWatch Events rule with an event pattern matching Personal Health Dashboard events and a target Lambda function. In the target Lambda function, notify the Senior Manager.

D. Add an AWS Config custom rule that runs periodically, checks the AWS service limit status, and streams notifications to an Amazon SNS topic. Deploy an AWS Lambda function that notifies the Senior Manager, and subscribe the Lambda function to the SNS topic.

Answer: B

Explanation:

To meet the requirements, the company needs to create a solution that alerts the Senior Manager when the creation of resources approaches the service limits for the account with the least amount of development effort. The company can use AWS Trusted Advisor, which is a service that provides best practice recommendations for cost optimization, performance, security, and service limits. The company can deploy an AWS Lambda function that refreshes Trusted Advisor checks, and configure an Amazon CloudWatch Events rule to run the Lambda function periodically. This will ensure that Trusted Advisor checks are up to date and reflect the current state of the account. The company can then create another CloudWatch Events rule with an event pattern matching Trusted Advisor events and a target Lambda function. The event pattern can filter for events related to service limit checks and their status. The target Lambda function can notify the Senior Manager via a third-party API call if the event indicates that the account is approaching or exceeding a service limit.

Question: 164

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.

Answer: AD

Explanation:

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: 165

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.

Answer: B

Explanation:

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: 166

A company builds an application that uses an Application Load Balancer in front of Amazon EC2 instances that are in an Auto Scaling group. The

application is stateless. The Auto Scaling group uses a custom AMI that is fully prebuilt. The EC2 instances do not have a custom bootstrapping process.

The AMI that the Auto Scaling group uses was recently deleted. The Auto Scaling group's scaling activities show failures because the AMI ID does not exist.

Which combination of steps should a DevOps engineer take to meet these requirements? (Select THREE.)

A. Create a new launch template that uses the new AMI.

B. Update the Auto Scaling group to use the new launch template.

C. Reduce the Auto Scaling group's desired capacity to O.

D. Increase the Auto Scaling group's desired capacity by I.

E. Create a new AMI from a running EC2 instance in the Auto Scaling group.

F. Create a new AMI by copying the most recent public AMI of the operating system that the EC2 instances use.

Answer: ABF

Explanation:

To restore the functionality of the Auto Scaling group after the AMI was deleted, the DevOps engineer needs to create a new AMI and update the Auto Scaling group to use it. The DevOps engineer can create a new AMI by copying the most recent public AMI of the operating system that the EC2 instances use. This will ensure that the new AMI has the same operating system as the custom AMI that was deleted. The DevOps engineer can then create a new launch template that uses the new AMI and update the Auto Scaling group to use the new launch template. This will allow the Auto Scaling group to launch new instances with the new AMI.

Question: 167

A company plans to use Amazon CloudWatch to monitor its Amazon EC2 instances. The company needs to stop EC2 instances when the average of the NetworkPacketsIn metric is less than 5 for at least 3 hours in a 12-hour time window. The company must evaluate the metric every hour. The EC2 instances must continue to run if there is missing data for the NetworkPacketsIn metric during the evaluation period.

A DevOps engineer creates a CloudWatch alarm for the NetworkPacketsIn metric. The DevOps engineer configures a threshold value of 5 and an evaluation period of 1 hour.

Which set of additional actions should the DevOps engineer take to meet these requirements?

A. Configure the Datapoints to Alarm value to be 3 out of 12. Configure the alarm to treat missing data as breaching the threshold. Add an AWS Systems Manager action to stop the instance when the alarm enters the ALARM state.

B. Configure the Datapoints to Alarm value to be 3 out of 12. Configure the alarm to treat missing data as not breaching the threshold. Add an EC2 action to stop the instance when the alarm enters the ALARM state.

C. Configure the Datapoints to Alarm value to be 9 out of 12. Configure the alarm to treat missing data as breaching the threshold. Add an EC2 action to stop the instance when the alarm enters the ALARM state.

D. Configure the Datapoints to Alarm value to be 9 out of 12. Configure the alarm to treat missing data as not breaching the threshold. Add an AWS Systems Manager action to stop the instance when the alarm enters the ALARM state.

Answer: B

Explanation:

To meet the requirements, the DevOps engineer needs to configure the CloudWatch alarm to stop the EC2 instances when the average of the NetworkPacketsIn metric is less than 5 for at least 3 hours in a 12-hour time window. This means that the alarm should trigger when 3 out of 12 datapoints are below the threshold of 5. The alarm should also treat missing data as not breaching the threshold, so that the EC2 instances continue to run if there is no data for the metric during the evaluation period. The DevOps engineer can add an EC2 action to stop the instance when the alarm enters the ALARM state, which is a built-in action type for CloudWatch alarms.

Question: 168

A company uses AWS and has a VPC that contains critical compute infrastructure with predictable traffic patterns. The company has configured VPC flow logs that are published to a log group in Amazon CloudWatch Logs.

The company's DevOps team needs to configure a monitoring solution for the VPC flow logs to identify anomalies in network traffic to the VPC over time. If the monitoring solution detects an anomaly, the company needs the ability to initiate a response to the anomaly.

www.atmicnetworks.com

How should the DevOps team configure the monitoring solution to meet these requirements?

A. Create an Amazon Kinesis data stream. Subscribe the log group to the data stream. Configure Amazon Kinesis Data Analytics to detect log anomalies in the data stream. Create an
AWS Lambda function to use as the output of the data stream. Configure the Lambda function to write to the default Amazon EventBridge event bus in the event of an anomaly finding.

B. Create an Amazon Kinesis Data Firehose delivery stream that delivers events to an Amazon S3 bucket. Subscribe the log group to the delivery stream. Configure Amazon Lookout for Metrics to monitor the data in the S3 bucket for anomalies. Create an AWS Lambda function to run in response to Lookout for Metrics anomaly findings. Configure the Lambda function to publish to the default Amazon EventBridge event bus.

C. Create an AWS Lambda function to detect anomalies. Configure the Lambda function to publish an event to the default Amazon EventBridge event bus if the Lambda function detects an anomaly. Subscribe the Lambda function to the log group.

D. Create an Amazon Kinesis data stream. Subscribe the log group to the data stream. Create an AWS Lambda function to detect log anomalies. Configure the Lambda function to write to the default Amazon EventBridge event bus if the Lambda function detects an anomaly. Set the Lambda function as the processor for the data stream.

Answer: D

Explanation:

To meet the requirements, the DevOps team needs to configure a monitoring solution for the VPC flow logs that can detect anomalies in network traffic over time and initiate a response to the anomaly. The DevOps team can use Amazon Kinesis Data Streams to ingest and process streaming data from CloudWatch Logs. The DevOps team can subscribe the log group to a Kinesis data stream, which will deliver log events from CloudWatch Logs to Kinesis Data Streams in near real-time. The DevOps team can then create an AWS Lambda function to detect log anomalies using machine learning or statistical methods. The Lambda function can be set as a processor for the data stream, which means that it will process each record from the stream before sending it to downstream applications or destinations. The Lambda function can also write to the default Amazon EventBridge event bus if it detects an anomaly, which will allow other AWS services or custom applications to respond to the anomaly event.

Question: 169 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.

Answer: BD

Explanation:

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. Reference: 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: 170

A growing company manages more than 50 accounts in an organization in AWS Organizations. The company has configured its applications to send logs to Amazon CloudWatch Logs.

A DevOps engineer needs to aggregate logs so that the company can quickly search the logs to respond to future security incidents. The DevOps engineer has created a new AWS account for centralized monitoring.

Which combination of steps should the DevOps engineer take to make the application logs searchable from the monitoring account? (Select THREE.)

A. In the monitoring account, download an AWS CloudFormation template from CloudWatch to use in Organizations. Use CloudFormation StackSets in the organization's management account to deploy the CloudFormation template to the entire organization.

B. Create an AWS CloudFormation template that defines an IAM role. Configure the role to allow logs-amazonaws.com to perform the logs:Link action if the aws:ResourceAccount property is equal to the monitoring account ID. Use CloudFormation StackSets in the organization's management account to deploy the CloudFormation template to the entire organization.

C. Create an IAM role in the monitoring account. Attach a trust policy that allows logs.amazonaws.com to perform the iam:CreateSink

action if the aws:PrincipalOrgId property is equal to the organization ID.

D. In the organization's management account, enable the logging policies for the organization.

E. use CloudWatch Observability Access Manager in the monitoring account to create a sink. Allow logs to be shared with the monitoring account. Configure the monitoring account data selection to view the Observability data from the organization ID.

F. In the monitoring account, attach the CloudWatchLogsReadOnlyAccess AWS managed policy to an IAM role that can be assumed to search the logs.

Answer: BCF

Explanation:

To aggregate logs from multiple accounts in an organization, the DevOps engineer needs to create a cross-account subscription1 that allows the monitoring account to receive log events from the sharing accounts.

To enable cross-account subscription, the DevOps engineer needs to create an IAM role in each sharing account that grants permission to CloudWatch Logs to link the log groups to the destination in the monitoring account2. This can be done using a CloudFormation template and StackSets3 to deploy the role to all accounts in the organization.
The DevOps engineer also needs to create an IAM role in the monitoring account that allows CloudWatch Logs to create a sink for receiving log events from other accounts4. The role must have a trust policy that specifies the organization ID as a condition.
Finally, the DevOps engineer needs to attach the CloudWatchLogsReadOnlyAccess policy5 to an IAM role in the monitoring account that can be used to search the logs from the cross-account subscription.
Reference: 1: Cross-account log data sharing with subscriptions 2: Create an IAM role for CloudWatch Logs in each sharing account 3: AWS CloudFormation StackSets 4: Create an IAM role for CloudWatch Logs in your monitoring account 5: CloudWatchLogsReadOnlyAccess policy

Question: 171

A company's security policies require the use of security hardened AMIS in production environments.
A DevOps engineer has used EC2 Image Builder to create a pipeline that builds the AMIs on a recurring schedule.

The DevOps engineer needs to update the launch templates of the companys Auto Scaling groups. The Auto Scaling groups must use the newest AMIS during the launch of Amazon EC2 instances.

Which solution will meet these requirements with the MOST operational efficiency?

A. Configure an Amazon EventBridge rule to receive new AMI events from Image Builder. Target an AWS Systems Manager Run Command document that updates the launch templates of the Auto Scaling groups with the newest AMI ID.

B. Configure an Amazon EventBridge rule to receive new AMI events from Image Builder. Target an AWS Lambda function that updates the launch templates of the Auto Scaling groups with the newest AMI ID.

C. Configure the launch template to use a value from AWS Systems Manager Parameter Store for the AMI ID. Configure the Image Builder pipeline to update the Parameter Store value with the newest AMI ID.

D. Configure the Image Builder distribution settings to update the launch templates with the newest AMI ID. Configure the Auto Scaling groups to use the newest version of the launch template.

Answer: C

Explanation:

The most operationally efficient solution is to use AWS Systems Manager Parameter Storel to store the AMI ID and reference it in the launch template2. This way, the launch template does not need to be updated every time a new AMI is created by Image Builder. Instead, the Image Builder pipeline can update the Parameter Store value with the newest AMI ID3, and the Auto Scaling group can launch instances using the latest value from Parameter Store.

The other solutions require updating the launch template or creating a new version of it every time a new AMI is created, which adds complexity and overhead. Additionally, using EventBridge rules and Lambda functions or Run Command documents introduces additional dependencies and potential points of failure.

Reference: 1: AWS Systems Manager Parameter Store 2: Using AWS Systems Manager parameters instead of AMI IDs in launch templates 3: Update an SSM parameter with Image Builder

Question: 172

A company is launching an application that stores raw data in an Amazon S3 bucket. Three applications need to access the data to generate reports. The data must be redacted differently for each application before the applications can access the data.

Which solution will meet these requirements?

A. Create an S3 bucket for each application. Configure S3 Same-Region Replication (SRR) from the raw data's S3 bucket to each application's S3 bucket. Configure each application to consume data from its own S3 bucket.

B. Create an Amazon Kinesis data stream. Create an AWS Lambda function that is invoked by object creation events in the raw data's S3 bucket. Program the Lambda function to redact data for each application. Publish the data on the Kinesis data stream. Configure each application to consume data from the Kinesis data stream.

C. For each application, create an S3 access point that uses the raw data's S3 bucket as the destination. Create an AWS Lambda function that is invoked by object creation events in the raw data's S3 bucket. Program the Lambda function to redact data for each application. Store the data in each application's S3 access point. Configure each application to consume data from its own S3 access point.

D. Create an S3 access point that uses the raw data's S3 bucket as the destination. For each application, create an S3 Object Lambda access point that uses the S3 access point. Configure the AWS Lambda function for each S3 Object Lambda access point to redact data when objects are retrieved. Configure each application to consume data from its own S3 Object Lambda access point.

Answer: D

Explanation:

The best solution is to use S3 Object Lambda1, which allows you to add your own code to S3 GET, LIST, and HEAD requests to modify and process data as it is returned to an application2. This way, you can redact the data differently for each application without creating and storing multiple copies of the data or running proxies.

The other solutions are less efficient or scalable because they require replicating the data to multiple buckets, streaming the data

through Kinesis, or storing the data in S3 access points.

Reference: 1: Amazon S3 Features | Object Lambda | AWS 2: Transforming objects with S3 Object Lambda - Amazon Simple Storage Service

Question: 173

A DevOps engineer manages a company's Amazon Elastic Container Service (Amazon ECS) cluster.

The cluster runs on several Amazon EC2 instances that are in an Auto Scaling group. The DevOps engineer must implement a solution that logs and reviews all stopped tasks for errors.

Which solution will meet these requirements?

A. Create an Amazon EventBridge rule to capture task state changes. Send the event to Amazon CloudWatch Logs. Use CloudWatch Logs Insights to investigate stopped tasks.

B. Configure tasks to write log data in the embedded metric format. Store the logs in Amazon CloudWatch Logs. Monitor the ContainerInstanceCount metric for changes.

C. Configure the EC2 instances to store logs in Amazon CloudWatch Logs. Create a CloudWatch Contributor Insights rule that uses the EC2 instance log data. Use the Contributor Insights rule to investigate stopped tasks.

D. Configure an EC2 Auto Scaling lifecycle hook for the EC2_INSTANCE_TERMINATING scale-in event. Write the SystemEventLog file to Amazon S3. Use Amazon Athena to query the log file for errors.

Answer: A

Explanation:

The best solution to log and review all stopped tasks for errors is to use Amazon EventBridge and Amazon CloudWatch Logs. Amazon EventBridge allows the DevOps engineer to create a rule that matches task state change events from Amazon ECS. The rule can then send the event data to Amazon CloudWatch Logs as the target. Amazon CloudWatch Logs can store and monitor the log data, and also provide CloudWatch Logs Insights, a feature that enables the DevOps engineer to interactively search and analyze the log data. Using CloudWatch Logs Insights, the DevOps engineer can filter and aggregate the log data based on various fields, such as cluster, task, container, and reason. This way, the DevOps engineer can easily identify and investigate the stopped tasks and their errors.

The other options are not as effective or efficient as the solution in option A. Option B is not suitable because the embedded metric format is designed for custom metrics, not for logging task state changes. Option C is not feasible because the EC2 instances do not store

the task state change events in their logs. Option D is not relevant because the EC2_INSTANCE_TERMINATING lifecycle hook is triggered when an EC2 instance is terminated by the Auto Scaling group, not when a task is stopped by Amazon ECS.

Reference:

: Creating a CloudWatch Events Rule That Triggers on an Event - Amazon Elastic Container Service

: Sending and Receiving Events Between AWS Accounts - Amazon EventBridge

: Working with Log Data - Amazon CloudWatch Logs

: Analyzing Log Data with CloudWatch Logs Insights - Amazon CloudWatch Logs

: Embedded Metric Format - Amazon CloudWatch

: Amazon EC2 Auto Scaling Lifecycle Hooks - Amazon EC2 Auto Scaling

Question: 174

A company uses an organization in AWS Organizations that has all features enabled. The company uses AWS Backup in a primary account and uses an AWS Key Management Service (AWS KMS) key to encrypt the backups.

The company needs to automate a cross-account backup of the resources that AWS Backup backs up in the primary account. The company configures cross-account backup in the Organizations management account. The company creates a new AWS account in the organization and configures an AWS Backup backup vault in the new account. The company creates a KMS key in the new account to encrypt the backups. Finally, the company configures a new backup plan in the primary account. The destination for the new backup plan is the backup vault in the new account.

When the AWS Backup job in the primary account is invoked, the job creates backups in the primary account. However, the backups are not copied to the new account's backup vault.

Which combination of steps must the company take so that backups can be copied to the new account's backup vault? (Select TWO.)

A. Edit the backup vault access policy in the new account to allow access to the primary account.

B. Edit the backup vault access policy in the primary account to allow access to the new account.

C. Edit the backup vault access policy in the primary account to allow access to the KMS key in the new account.

D. Edit the key policy of the KMS key in the primary account to share the key with the new account.

E. Edit the key policy of the KMS key in the new account to share the key with the primary account.

Answer: AE

Explanation:

To enable cross-account backup, the company needs to grant permissions to both the backup vault and the KMS key in the destination account. The backup vault access policy in the destination account must allow the primary account to copy backups into the vault. The key policy of the KMS key in the destination account must allow the primary account to use the key to encrypt and decrypt the backups.

These steps are described in the AWS documentation^. Therefore, the correct answer is A and E.

Reference:

1: Creating backup copies across AWS accounts - AWS Backup

2: Using AWS Backup with AWS Organizations - AWS Backup

Question: 175

A company has an application that runs on AWS Lambda and sends logs to Amazon CloudWatch Logs. An Amazon Kinesis data stream is subscribed to the log groups in CloudWatch Logs. A single consumer Lambda function processes the logs from the data stream and stores the logs in an Amazon S3 bucket.

The company's DevOps team has noticed high latency during the processing and ingestion of some logs.

Which combination of steps will reduce the latency? (Select THREE.)

A. Create a data stream consumer with enhanced fan-out. Set the Lambda function that processes the logs as the consumer.

B. Increase the ParallelizationFactor setting in the Lambda event source mapping.

C. Configure reserved concurrency for the Lambda function that processes the logs.

Increase the batch size in the Kinesis data stream.

D. Turn off the ReportBatchItemFailures setting in the Lambda event source mapping.

Increase the number of shards in the Kinesis data stream.

Answer: ABC

Explanation:

www.atmicnetworks.com

The latency in processing and ingesting logs can be caused by several factors, such as the throughput of the Kinesis data stream, the concurrency of the Lambda function, and the configuration of the event source mapping. To reduce the latency, the following steps can be taken:

Create a data stream consumer with enhanced fan-out. Set the Lambda function that processes the logs as the consumer. This will allow the Lambda function to receive records from the data stream with dedicated throughput of up to 2 MB per second per shard, independent of other consumers1. This will reduce the contention and delay in accessing the data stream.

Increase the ParallelizationFactor setting in the Lambda event source mapping. This will allow the Lambda service to invoke more instances of the function concurrently to process the records from the data stream2. This will increase the processing capacity and reduce the backlog of records in the data stream.

Configure reserved concurrency for the Lambda function that processes the logs. This will ensure that the function has enough concurrency available to handle the increased load from the data stream3. This will prevent the function from being throttled by the account-level concurrency limit.

The other options are not effective or may have negative impacts on the latency. Option D is not suitable because increasing the batch size in the Kinesis data stream will increase the amount of data that the Lambda function has to process in each invocation, which may increase the execution time and latency4. Option E is not advisable because turning off the ReportBatchItemFailures setting in the Lambda event source mapping will prevent the Lambda service from retrying the failed records, which may result in data loss. Option F is not necessary because increasing the number of shards in the Kinesis data stream will increase the throughput of the data stream, but it will not affect the processing speed of the Lambda function, which is the bottleneck in this scenario.

Reference:

1: Using AWS Lambda with Amazon Kinesis Data Streams - AWS Lambda

2: AWS Lambda event source mappings - AWS Lambda

3: Managing concurrency for a Lambda function - AWS Lambda

4: AWS Lambda function scaling - AWS Lambda

: AWS Lambda event source mappings - AWS Lambda

: Scaling Amazon Kinesis Data Streams with AWS CloudFormation - Amazon Kinesis Data Streams

Question: 176

A company needs to implement failover for its application. The application includes an Amazon CloudFront distribution and a public Application Load Balancer (ALB) in an AWS Region. The company has configured the ALB as the default origin for the distribution.

After some recent application outages, the company wants a zero-second RTO. The company deploys the application to a secondary Region in a warm standby configuration. A DevOps engineer needs to automate the failover of the application to the secondary Region so

www.atmicnetworks.com

that HTTP GET requests meet the desired RTO.

Which solution will meet these requirements?

A. Create a second CloudFront distribution that has the secondary ALB as the default origin. Create Amazon Route 53 alias records that have a failover policy and Evaluate Target Health set to Yes for both CloudFront distributions. Update the application to use the new record set.

B. Create a new origin on the distribution for the secondary ALB. Create a new origin group. Set the original ALB as the primary origin. Configure the origin group to fail over for HTTP 5xx status codes. Update the default behavior to use the origin group.

C. Create Amazon Route 53 alias records that have a failover policy and Evaluate Target Health set to Yes for both ALBs. Set the TTL of both records to 0. Update the distribution's origin to use the new record set.

D. Create a CloudFront function that detects HTTP 5xx status codes. Configure the function to return a 307 Temporary Redirect error response to the secondary ALB if the function detects 5xx status codes. Update the distribution's default behavior to send origin responses to the function.

Answer: B

Explanation:

The best solution to implement failover for the application is to use CloudFront origin groups. Origin groups allow CloudFront to automatically switch to a secondary origin when the primary origin is unavailable or returns specific HTTP status codes that indicate a failurel. This way, CloudFront can serve the requests from the secondary ALB in the secondary Region without any delay or redirection. To set up origin groups, the DevOps engineer needs to create a new origin on the distribution for the secondary ALB, create a new origin group with the original ALB as the primary origin and the secondary ALB as the secondary origin, and configure the origin group to fail over for HTTP 5xx status codes. Then, the DevOps engineer needs to update the default behavior to use the origin group instead of the single origin2.

The other options are not as effective or efficient as the solution in option B. Option A is not suitable because creating a second CloudFront distribution will increase the complexity and cost of the application. Moreover, using Route 53 alias records with a failover policy will introduce some delay in detecting and switching to the secondary CloudFront distribution, which may not meet the zero-second RTO requirement. Option C is not feasible because CloudFront does not support using Route 53 alias records as origins3. Option D is not advisable because using a CloudFront function to redirect the requests to the secondary ALB will add an extra round-trip and latency to the failover process, which may also not meet the zero-second RTO requirement.

Reference:

1: Optimizing high availability with CloudFront origin failover - Amazon CloudFront

2: Creating an origin group - Amazon CloudFront

3: Values That You Specify When You Create or Update a Web Distribution - Amazon CloudFront

Question: 177

A company uses an organization in AWS Organizations to manage its AWS accounts. The company recently acquired another company that has standalone AWS accounts. The acquiring company's DevOps team needs to consolidate the administration of the AWS accounts for both companies and retain full administrative control of the accounts. The DevOps team also needs to collect and group findings across all the accounts to implement and maintain a security posture.

Which combination of steps should the DevOps team take to meet these requirements? (Select TWO.)

A. Invite the acquired company's AWS accounts to join the organization. Create an SCP that has full administrative privileges. Attach the SCP to the management account.

B. Invite the acquired company's AWS accounts to join the organization. Create the OrganizationAccountAccessRole 1AM role in the invited accounts. Grant permission to the management account to assume the role.

C. Use AWS Security Hub to collect and group findings across all accounts. Use Security Hub to automatically detect new accounts as the accounts are added to the organization.

D. Use AWS Firewall Manager to collect and group findings across all accounts. Enable all features for the organization. Designate an account in the organization as the delegated administrator account for Firewall Manager.

E. Use Amazon Inspector to collect and group findings across all accounts. Designate an account in the organization as the delegated administrator account for Amazon Inspector.

Answer: BC

Explanation:

The correct answer is B and C. Option B is correct because inviting the acquired company's AWS accounts to join the organization and creating the OrganizationAccountAccessRole IAM role in the invited accounts allows the management account to assume the role and gain full administrative access to the member accounts. Option C is correct because using AWS Security Hub to collect and group findings across all accounts enables the DevOps team to monitor and improve the security posture of the organization. Security Hub can automatically detect new accounts as the accounts are added to the organization and enable Security Hub for them. Option A is incorrect because creating an SCP that has full administrative privileges and attaching it to the management account does not grant the management account access to the member accounts. SCPs are used to restrict the permissions of the member accounts, not to grant permissions to the management account. Option D is incorrect because using AWS Firewall Manager to collect and group findings across all accounts is not a valid use case for Firewall Manager. Firewall Manager is used to centrally configure and manage firewall rules across the organization, not to collect and group security findings. Option E is incorrect because using Amazon Inspector to collect and group findings across all accounts is not a valid use case for Amazon Inspector. Amazon Inspector is used to assess the security and compliance of applications running on Amazon EC2 instances, not to collect and group security findings across accounts. Reference:

Inviting an AWS account to join your organization

Enabling and disabling AWS Security Hub

Service control policies

AWS Firewall Manager

Amazon Inspector

Question: 178

A company has an application and a CI/CD pipeline. The CI/CD pipeline consists of an AWS CodePipeline pipeline and an AWS CodeBuild project. The CodeBuild project runs tests against the application as part of the build process and outputs a test report. The company must keep the test reports for 90 days.

Which solution will meet these requirements?

A. Add a new stage in the CodePipeline pipeline after the stage that contains the CodeBuild project. Create an Amazon S3 bucket to store the reports. Configure an S3 deploy action type in the new CodePipeline stage with the appropriate path and format for the reports.

B. Add a report group in the CodeBuild project buildspec file with the appropriate path and format for the reports. Create an Amazon S3 bucket to store the reports. Configure an Amazon EventBridge rule that invokes an AWS Lambda function to copy the reports to the S3 bucket when a build is completed. Create an S3 Lifecycle rule to expire the objects after 90 days.

C. Add a new stage in the CodePipeline pipeline. Configure a test action type with the appropriate path and format for the reports. Configure the report expiration time to be 90 days in the CodeBuild project buildspec file.

D. Add a report group in the CodeBuild project buildspec file with the appropriate path and format for the reports. Create an Amazon S3 bucket to store the reports. Configure the report group as an artifact in the CodeBuild project buildspec file. Configure the S3 bucket as the artifact destination. Set the object expiration to 90 days.

Answer: B

Explanation:

The correct solution is to add a report group in the AWS CodeBuild project buildspec file with the appropriate path and format for the reports. Then, create an Amazon S3 bucket to store the reports. You should configure an Amazon EventBridge rule that invokes an AWS Lambda function to copy the reports to the S3 bucket when a build is completed. Finally, create an S3 Lifecycle rule to expire the objects after 90 days. This approach allows for the automated transfer of reports to long-term storage and ensures they are retained for the required duration without manual intervention!

Reference:

www.atmicnetworks.com

AWS CodeBuild User Guide on test reporting1.

AWS CodeBuild User Guide on working with report groups2.

AWS Documentation on using AWS CodePipeline with AWS CodeBuild3.

Question: 179

A company uses an Amazon API Gateway regional REST API to host its application API. The REST API has a custom domain. The REST API's default endpoint is deactivated.

The company's internal teams consume the API. The company wants to use mutual TLS between the API and the internal teams as an additional layer of authentication.

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

A. Use AWS Certificate Manager (ACM) to create a private certificate authority (CA). Provision a client certificate that is signed by the private CA.

B. Provision a client certificate that is signed by a public certificate authority (CA). Import the certificate into AWS Certificate Manager (ACM).

C. Upload the provisioned client certificate to an Amazon S3 bucket. Configure the API Gateway mutual TLS to use the client certificate that is stored in the S3 bucket as the trust store.

D. Upload the provisioned client certificate private key to an Amazon S3 bucket. Configure the API Gateway mutual TLS to use the private key that is stored in the S3 bucket as the trust store.

E. Upload the root private certificate authority (CA) certificate to an Amazon S3 bucket. Configure the API Gateway mutual TLS to use the private CA certificate that is stored in the S3 bucket as the trust store.

Answer: AE

Explanation:

Mutual TLS (mTLS) authentication requires two-way authentication between the client and the server. For Amazon API Gateway, you can enable mTLS for a custom domain name, which requires clients to present X.509 certificates to verify their identity to access your API. To set up mTLS, you would typically use AWS Certificate Manager (ACM) to create a private certificate authority (CA) and provision a client certificate signed by this private CA. The root CA certificate is then uploaded to an Amazon S3 bucket and configured in API Gateway as the trust store12.

Reference:

Introducing mutual TLS authentication for Amazon API Gateway1.

Configuring mutual TLS authentication for a REST API2.

AWS Private Certificate Authority details3.

AWS Certificate Manager Private Certificate Authority updates4.

Question: 180

A company has an application that includes AWS Lambda functions. The Lambda functions run Python code that is stored in an AWS CodeCommit repository. The company has recently experienced failures in the production environment because of an error in the Python code. An engineer has written unit tests for the Lambda functions to help avoid releasing any future defects into the production environment.

The company's DevOps team needs to implement a solution to integrate the unit tests into an existing AWS CodePipeline pipeline. The solution must produce reports about the unit tests for the company to view.

Which solution will meet these requirements?

A. Associate the CodeCommit repository with Amazon CodeGuru Reviewer. Create a new AWS CodeBuild project. In the CodePipeline pipeline, configure a test stage that uses the new CodeBuild project. Create a buildspec.yml file in the CodeCommit repository. In the buildspec.yml file, define the actions to run a CodeGuru review.

B. Create a new AWS CodeBuild project. In the CodePipeline pipeline, configure a test stage that uses the new CodeBuild project. Create a CodeBuild report group. Create a buildspec.yml file in the CodeCommit repository. In the buildspec.yml file, define the actions to run the unit tests with an output of JUNITXML in the build phase section. Configure the test reports to be uploaded to the new CodeBuild report group.

C. Create a new AWS CodeArtifact repository. Create a new AWS CodeBuild project. In the CodePipeline pipeline, configure a test stage that uses the new CodeBuild project. Create an appspec.yml file in the original CodeCommit repository. In the appspec.yml file, define the actions to run the unit tests with an output of CUCUMBERJSON in the build phase section. Configure the tests reports to be sent to the new CodeArtifact repository.

D. Create a new AWS CodeBuild project. In the CodePipeline pipeline, configure a test stage that uses the new CodeBuild project. Create a new Amazon S3 bucket. Create a buildspec.yml file in the CodeCommit repository. In the buildspec.yml file, define the actions to run the unit tests with an output of HTML in the phases section. In the reports section, upload the test reports to the S3 bucket.

www.atmicnetworks.com

Answer: B

Explanation:

The correct answer is B. Creating a new AWS CodeBuild project and configuring a test stage in the AWS CodePipeline pipeline that uses the new CodeBuild project is the best way to integrate the unit tests into the existing pipeline. Creating a CodeBuild report group and uploading the test reports to the new CodeBuild report group will produce reports about the unit tests for the company to view. Using JUNITXML as the output format for the unit tests is supported by CodeBuild and will generate a valid report.

Option A is incorrect because Amazon CodeGuru Reviewer is a service that provides automated code reviews and recommendations for improving code quality and performance. It is not a tool for running unit tests or producing test reports. Therefore, option A will not meet the requirements.

Option C is incorrect because AWS CodeArtifact is a service that provides secure, scalable, and costeffective artifact management for software development. It is not a tool for running unit tests or producing test reports. Moreover, option C uses CUCUMBERJSON as the output format for the unit tests, which is not supported by CodeBuild and will not generate a valid report.

Option D is incorrect because uploading the test reports to an Amazon S3 bucket is not the best way to produce reports about the unit tests for the company to view. CodeBuild has a built-in feature to create and manage test reports, which is more convenient and efficient than using S3. Furthermore, option D uses HTML as the output format for the unit tests, which is not supported by CodeBuild and will not generate a valid report.

Question: 181

A company uses AWS Organizations to manage its AWS accounts. The company has a root OU that has a child OU. The root OU has an SCP that allows all actions on all resources. The child OU has an SCP that allows all actions for Amazon DynamoDB and AWS Lambda, and denies all other actions.

The company has an AWS account that is named vendor-data in the child OU. A DevOps engineer has an 1AM user that is attached to the AdministratorAccess 1AM policy in the vendor-data account. The DevOps engineer attempts to launch an Amazon EC2 instance in the vendor-data account but receives an access denied error.

Which change should the DevOps engineer make to launch the EC2 instance in the vendor-data account?

A. Attach the AmazonEC2FullAccess 1AM policy to the 1AM user.

B. Create a new SCP that allows all actions for Amazon EC2. Attach the SCP to the vendor-data account.

C. Update the SCP in the child OU to allow all actions for Amazon EC2.

www.atmicnetworks.com

D. Create a new SCP that allows all actions for Amazon EC2. Attach the SCP to the root OU.

Answer: C

Explanation:

The correct answer is C. Updating the SCP in the child OU to allow all actions for Amazon EC2 will enable the DevOps engineer to launch the EC2 instance in the vendor-data account. SCPs are applied to OUs and accounts in a hierarchical manner, meaning that the SCPs attached to the parent OU are inherited by the child OU and accounts. Therefore, the SCP in the child OU overrides the SCP in the root OU and denies all actions except for DynamoDB and Lambda. By adding EC2 to the allowed actions in the child OU's SCP, the DevOps engineer can access EC2 resources in the vendor-data account.

Option A is incorrect because attaching the AmazonEC2FullAccess IAM policy to the IAM user will not grant the user access to EC2 resources. IAM policies are evaluated after SCPs, so even if the IAM policy allows EC2 actions, the SCP will still deny them.

Option B is incorrect because creating a new SCP that allows all actions for EC2 and attaching it to the vendor-data account will not work. SCPs are not cumulative, meaning that only one SCP is applied to an account at a time. The SCP attached to the account will be the SCP attached to the OU that contains the account. Therefore, option B will not change the SCP that is applied to the vendor-data account.

Option D is incorrect because creating a new SCP that allows all actions for EC2 and attaching it to the root OU will not work. As explained earlier, the SCP in the child OU overrides the SCP in the root OU and denies all actions except for DynamoDB and Lambda. Therefore, option D will not affect the SCP that is applied to the vendor-data account.

Question: 182

A company has deployed a critical application in two AWS Regions. The application uses an Application Load Balancer (ALB) in both Regions. The company has Amazon Route 53 alias DNS records for both ALBs.

The company uses Amazon Route 53 Application Recovery Controller to ensure that the application can fail over between the two Regions. The Route 53 ARC configuration includes a routing control for both Regions. The company uses Route 53 ARC to perform quarterly disaster recovery (DR) tests.

During the most recent DR test, a DevOps engineer accidentally turned off both routing controls. The company needs to ensure that at least one routing control is turned on at all times.

Which solution will meet these requirements?

A. In Route 53 ARC. create a new assertion safety rule. Apply the assertion safety rule to the two routing controls. Configure the rule with the ATLEAST type with a threshold of 1.

B. In Route 53 ARC, create a new gating safety rule. Apply the assertion safety rule to the two routing controls. Configure the rule with the OR type with a threshold of 1.

C. In Route 53 ARC, create a new resource set. Configure the resource set with an AWS: Route53: HealthCheck resource type. Specify the ARNs of the two routing controls as the target resource. Create a new readiness check for the resource set.

D. In Route 53 ARC, create a new resource set. Configure the resource set with an AWS: Route53RecoveryReadiness: DNSTargetResource resource type. Add the domain names of the two Route 53 alias DNS records as the target resource. Create a new readiness check for the resource set.

Answer: A

Explanation:

The correct solution is to create a new assertion safety rule in Route 53 ARC and apply it to the two routing controls. An assertion safety rule is a type of safety rule that ensures that a minimum number of routing controls are always enabled. The ATLEAST type of assertion safety rule specifies the minimum number of routing controls that must be enabled for the rule to evaluate as healthy. By setting the threshold to 1, the rule ensures that at least one routing control is always turned on. This prevents the scenario where both routing controls are accidentally turned off and the application becomes unavailable in both Regions.

The other solutions are incorrect because they do not use safety rules to prevent both routing controls from being turned off. A gating safety rule is a type of safety rule that prevents routing control state changes that violate the rule logic. The OR type of gating safety rule specifies that one or more routing controls must be enabled for the rule to evaluate as healthy. However, this rule does not prevent a user from turning off both routing controls manually. A resource set is a collection of resources that are tested for readiness by Route 53 ARC. A readiness check is a test that verifies that all the resources in a resource set are operational. However, these concepts are not related to routing control states or safety rules. Therefore, creating a new resource set and a new readiness check will not ensure that at least one routing control is turned on at all times. Reference:

Routing control in Amazon Route 53 Application Recovery Controller

Viewing and updating routing control states in Route 53 ARC

Creating a control panel in Route 53 ARC

Creating safety rules in Route 53 ARC

Question: 183

A healthcare services company is concerned about the growing costs of software licensing for an application for monitoring patient wellness. The company wants to create an audit process to ensure that the application is running exclusively on Amazon EC2 Dedicated Hosts. A DevOps engineer must create a workflow to audit the application to ensure compliance.

www.atmicnetworks.com

What steps should the engineer take to meet this requirement with the LEAST administrative overhead?

A. Use AWS Systems Manager Configuration Compliance. Use calls to the put-compliance-items API action to scan and build a database of noncompliant EC2 instances based on their host placement configuration. Use an Amazon DynamoDB table to store these instance IDs for fast access. Generate a report through Systems Manager by calling the list-compliance-summaries API action.

B. Use custom Java code running on an EC2 instance. Set up EC2 Auto Scaling for the instance depending on the number of instances to be checked. Send the list of noncompliant EC2 instance IDs to an Amazon SQS queue. Set up another worker instance to process instance IDs from the SQS queue and write them to Amazon DynamoDB. Use an AWS Lambda function to terminate noncompliant instance IDs obtained from the queue, and send them to an Amazon SNS email topic for distribution.

C. Use AWS Config. Identify all EC2 instances to be audited by enabling Config Recording on all Amazon EC2 resources for the region. Create a custom AWS Config rule that triggers an AWS Lambda function by using the "config-rule-change-triggered" blueprint. Modify the Lambda evaluateCompliance () function to verify host placement to return a NON_COMPLIANT result if the instance is not running on an EC2 Dedicated Host. Use the AWS Config report to address noncompliant instances.

D. Use AWS CloudTrail. Identify all EC2 instances to be audited by analyzing all calls to the EC2 RunCommand API action. Invoke a AWS Lambda function that analyzes the host placement of the instance. Store the EC2 instance ID of noncompliant resources in an Amazon RDS for MySQL DB instance. Generate a report by querying the RDS instance and exporting the query results to a CSV text file.

_____ Answer: C

Explanation:

The correct answer is C. Using AWS Config to identify and audit all EC2 instances based on their host placement configuration is the most efficient and scalable solution to ensure compliance with the software licensing requirement. AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources. By creating a custom AWS Config rule that triggers a Lambda function to verify host placement, the DevOps engineer can automate the process of checking whether the instances are running on EC2 Dedicated Hosts or not. The Lambda function can return a NON_COMPLIANT result if the instance is not running on an EC2 Dedicated Host, and the AWS Config report can provide a summary of the compliance status of the instances. This solution requires the least administrative overhead compared to the other options.

Option A is incorrect because using AWS Systems Manager Configuration Compliance to scan and build a database of noncompliant EC2 instances based on their host placement configuration is a more complex and costly solution than using AWS Config. AWS Systems Manager Configuration Compliance is a feature of AWS Systems Manager that enables you to scan your managed instances for patch compliance and configuration inconsistencies. To use this feature, the DevOps engineer would need to install the Systems Manager Agent on each EC2 instance, create a State Manager association to run the put-compliance-items API action periodically, and use a DynamoDB table to store the instance IDs of noncompliant resources. This solution would also require more API calls and storage costs than using AWS Config.

Option B is incorrect because using custom Java code running on an EC2 instance to check and terminate noncompliant EC2 instances is a more cumbersome and error-prone solution than using AWS Config. This solution would require the DevOps engineer to write and maintain the Java code, set up EC2 Auto Scaling for the instance, use an SQS queue and another worker instance to process the instance

IDs, use a Lambda function and an SNS topic to terminate and notify the noncompliant instances, and handle any potential failures or exceptions in the workflow. This solution would also incur more compute, storage, and messaging costs than using AWS Config.

Option D is incorrect because using AWS CloudTrail to identify and audit EC2 instances by analyzing the EC2 RunCommand API action is a less reliable and accurate solution than using AWS Config. AWS CloudTrail is a service that enables you to monitor and log the API activity in your AWS account. The EC2 RunCommand API action is used to execute commands on one or more EC2 instances. However, this API action does not necessarily indicate the host placement of the instance, and it may not capture all the instances that are running on EC2 Dedicated Hosts or not. Therefore, option D would not provide a comprehensive and consistent audit of the EC2 instances.

Question: 184

A company is examining its disaster recovery capability and wants the ability to switch over its daily operations to a secondary AWS Region. The company uses AWS CodeCommit as a source control tool in the primary Region.

A DevOps engineer must provide the capability for the company to develop code in the secondary Region. If the company needs to use the secondary Region, developers can add an additional remote URL to their local Git configuration.

Which solution will meet these requirements?

A. Create a CodeCommit repository in the secondary Region. Create an AWS CodeBuild project to perform a Git mirror operation of the primary Region's CodeCommit repository to the secondary Region's CodeCommit repository. Create an AWS Lambda function that invokes the CodeBuild project. Create an Amazon EventBridge rule that reacts to merge events in the primary Region's CodeCommit repository. Configure the EventBridge rule to invoke the Lambda function.

B. Create an Amazon S3 bucket in the secondary Region. Create an AWS Fargate task to perform a Git mirror operation of the primary Region's CodeCommit repository and copy the result to the S3 bucket. Create an AWS Lambda function that initiates the Fargate task. Create an Amazon EventBridge rule that reacts to merge events in the CodeCommit repository. Configure the EventBridge rule to invoke the Lambda function.

C. Create an AWS CodeArtifact repository in the secondary Region. Create an AWS CodePipeline pipeline that uses the primary Region's CodeCommit repository for the source action. Create a CrossRegion stage in the pipeline that packages the CodeCommit repository contents and stores the contents in the CodeArtifact repository when a pull request is merged into the CodeCommit repository.

D. Create an AWS Cloud9 environment and a CodeCommit repository in the secondary Region. Configure the primary Region's CodeCommit repository as a remote repository in the AWS Cloud9 environment. Connect the secondary Region's CodeCommit repository to the AWS Cloud9 environment.

Answer: A

Explanation:

The best solution to meet the disaster recovery capability and allow developers to switch over to a secondary AWS Region for code development is option A. This involves creating a CodeCommit repository in the secondary Region and setting up an AWS CodeBuild project to perform a Git mirror operation of the primary Region's CodeCommit repository to the secondary Region's repository. An AWS Lambda function is then created to invoke the CodeBuild project. Additionally, an Amazon EventBridge rule is configured to react to merge events in the primary Region's CodeCommit repository and invoke the Lambda function12. This setup ensures that the secondary Region's repository is always up-to-date with the primary repository, allowing for a seamless transition in case of a disaster recovery event1.

Reference:

AWS CodeCommit User Guide on resilience and disaster recovery1.

AWS Documentation on monitoring CodeCommit events in Amazon EventBridge and Amazon CloudWatch Events2.

Question: 185

A DevOps engineer is planning to deploy a Ruby-based application to production. The application needs to interact with an Amazon RDS for MySQL database and should have automatic scaling and high availability. The stored data in the database is critical and should persist regardless of the state of the application stack.

The DevOps engineer needs to set up an automated deployment strategy for the application with automatic rollbacks. The solution also must alert the application team when a deployment fails.

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

A. Deploy the application on AWS Elastic Beanstalk. Deploy an Amazon RDS for MySQL DB instance as part of the Elastic Beanstalk configuration.

B. Deploy the application on AWS Elastic Beanstalk. Deploy a separate Amazon RDS for MySQL DB instance outside of Elastic Beanstalk.

C. Configure a notification email address that alerts the application team in the AWS Elastic Beanstalk configuration.

D. Configure an Amazon EventBridge rule to monitor AWS Health events. Use an Amazon Simple Notification Service (Amazon SNS) topic as a target to alert the application team.

E. Use the immutable deployment method to deploy new application versions.

F. Use the rolling deployment method to deploy new application versions.

Answer: BED

Explanation:

www.atmicnetworks.com

For deploying a Ruby-based application with requirements for interaction with an Amazon RDS for MySQL database, automatic scaling, high availability, and data persistence, the following steps will meet the requirements:

B. Deploy the application on AWS Elastic Beanstalk. Deploy a separate Amazon RDS for MySQL DB instance outside of Elastic Beanstalk. This approach ensures that the database persists independently of the Elastic Beanstalk environment, which can be torn down and recreated without affecting the database123.

E) Use the immutable deployment method to deploy new application versions. Immutable deployments provide a zero-downtime deployment method that ensures that if any part of the deployment process fails, the environment is rolled back to the original state automatically4.

D) Configure an Amazon EventBridge rule to monitor AWS Health events. Use an Amazon Simple Notification Service (Amazon SNS) topic as a target to alert the application team. This setup allows for automated monitoring and alerting of the application team in case of deployment failures or other health events56.

Reference:

AWS Elastic Beanstalk documentation on deploying Ruby applications1.

AWS documentation on application auto-scaling7.

AWS documentation on automated deployment strategies with automatic rollbacks and alerts456.

Question: 186

A company is using AWS CodePipeline to deploy an application. According to a new guideline, a member of the company's security team must sign off on any application changes before the changes are deployed into production. The approval must be recorded and retained.

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

A. Configure CodePipeline to write actions to Amazon CloudWatch Logs.

B. Configure CodePipeline to write actions to an Amazon S3 bucket at the end of each pipeline stage.

C. Create an AWS CloudTrail trail to deliver logs to Amazon S3.

D. Create a CodePipeline custom action to invoke an AWS Lambda function for approval. Create a policy that gives the security team access to manage CodePipeline custom actions.

E. Create a CodePipeline manual approval action before the deployment step. Create a policy that grants the security team access to approve manual approval stages.

Answer: CE

Explanation:

To meet the new guideline for application deployment, the company can use a combination of AWS CodePipeline and AWS CloudTrail. A manual approval action in CodePipeline allows the security team to review and approve changes before they are deployed. This action can be configured to pause the pipeline until approval is granted, ensuring that no changes move to production without the necessary sign-off. Additionally, by creating an AWS CloudTrail trail, all actions taken within CodePipeline, including approvals, are recorded and delivered to an Amazon S3 bucket. This provides an audit trail that can be retained for compliance and review purposes.

Reference:

AWS CodePipeline's manual approval action provides a way to ensure that a member of the security team can review and approve changes before they are deployed!.

AWS CloudTrail integration with CodePipeline allows for the recording and retention of all pipeline actions, including approvals, which can be stored in Amazon S3 for record-keeping2.

Question: 187

A company requires its internal business teams to launch resources through pre-approved AWS CloudFormation templates only. The security team requires automated monitoring when resources drift from their expected state.

Which strategy should be used to meet these requirements?

A. Allow users to deploy CloudFormation stacks using a CloudFormation service role only. Use CloudFormation drift detection to detect when resources have drifted from their expected state.

B. Allow users to deploy CloudFormation stacks using a CloudFormation service role only. Use AWS Config rules to detect when resources have drifted from their expected state.

C. Allow users to deploy CloudFormation stacks using AWS Service Catalog only. Enforce the use of a launch constraint. Use AWS Config rules to detect when resources have drifted from their expected state.

D. Allow users to deploy CloudFormation stacks using AWS Service Catalog only. Enforce the use of a template constraint. Use Amazon EventBridge notifications to detect when resources have drifted from their expected state.

Answer: C

Explanation:

The correct answer is C, Allowing users to deploy CloudFormation stacks using AWS Service Catalog only and enforcing the use of a launch constraint is the best way to ensure that the internal business teams launch resources through pre-approved CloudFormation

templates only. AWS Service Catalog is a service that enables organizations to create and manage catalogs of IT services that are approved for use on AWS. A launch constraint is a rule that specifies the role that AWS Service Catalog assumes when launching a product. By using a launch constraint, the DevOps engineer can control the permissions that the users have when launching a product. Using AWS Config rules to detect when resources have drifted from their expected state is the best way to automate the monitoring of the resources. AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources. AWS Config rules are custom or managed rules that AWS Config uses to evaluate whether your AWS resources comply with your desired configurations. By using AWS Config rules, the DevOps engineer can track the changes in the resources and identify any non-compliant resources.

Option A is incorrect because allowing users to deploy CloudFormation stacks using a CloudFormation service role only is not the best way to ensure that the internal business teams launch resources through pre-approved CloudFormation templates only. A CloudFormation service role is an IAM role that CloudFormation assumes to create, update, or delete the stack resources. By using a CloudFormation service role, the DevOps engineer can control the permissions that CloudFormation has when acting on the resources, but not the permissions that the users have when launching a stack. Therefore, option A does not prevent the users from launching resources that are not approved by the company. Using CloudFormation drift detection to detect when resources have drifted from their expected state is a valid way to monitor the resources, but it is not as automated and scalable as using AWS Config rules. CloudFormation drift detection is a feature that enables you to detect whether a stack's actual configuration differs, or has drifted, from its expected configuration. To use this feature, the DevOps engineer would need to manually initiate a drift detection operation on the stack or the stack resources, and then view the drift status and details in the CloudFormation console or API.

Option B is incorrect because allowing users to deploy CloudFormation stacks using a CloudFormation service role only is not the best way to ensure that the internal business teams launch resources through pre-approved CloudFormation templates only, as explained in option A. Using AWS Config rules to detect when resources have drifted from their expected state is a valid way to monitor the resources, as explained in option C.

Option D is incorrect because enforcing the use of a template constraint is not the best way to ensure that the internal business teams launch resources through pre-approved CloudFormation templates only. A template constraint is a rule that defines the values or properties that users can specify when launching a product. By using a template constraint, the DevOps engineer can control the parameters that the users can provide when launching a product, but not the permissions that the users have when launching a product. Therefore, option D does not prevent the users from launching resources that are not approved by the company. Using Amazon EventBridge notifications to detect when resources have drifted from their expected state is a less reliable and consistent solution than using AWS Config rules. Amazon EventBridge is a service that enables you to connect your applications with data from a variety of sources. Amazon EventBridge can deliver a stream of real-time data from event sources, such as AWS services, and route that data to targets, such as AWS Lambda functions. However, to use this solution, the DevOps engineer would need to configure the event source, the event bus, the event rule, and the event target for each resource type that needs to be monitored, which is more complex and error-prone than using AWS Config rules.

Question: 188

A DevOps engineer is setting up a container-based architecture. The engineer has decided to use AWS CloudFormation to automatically

provision an Amazon ECS cluster and an Amazon EC2 Auto Scaling group to launch the EC2 container instances. After successfully creating the CloudFormation stack, the engineer noticed that, even though the ECS cluster and the EC2 instances were created successfully and the stack finished the creation, the EC2 instances were associating with a different cluster.

How should the DevOps engineer update the CloudFormation template to resolve this issue?

A. Reference the EC2 instances in the AWS: ECS: Cluster resource and reference the ECS cluster in the AWS: ECS: Service resource.

B. Reference the ECS cluster in the AWS: AutoScaling: LaunchConfiguration resource of the UserData property.

C. Reference the ECS cluster in the AWS:EC2: Instance resource of the UserData property.

D. Reference the ECS cluster in the AWS: CloudFormation: CustomResource resource to trigger an AWS Lambda function that registers the EC2 instances with the appropriate ECS cluster.

Answer: B

Explanation:

The UserData property of the AWS: AutoScaling: LaunchConfiguration resource can be used to specify a script that runs when the EC2 instances are launched. This script can include the ECS cluster name as an environment variable for the ECS agent running on the EC2 instances. This way, the EC2

instances will register with the correct ECS cluster. Option A is incorrect because the AWS: ECS: Cluster resource does not have a property to reference the EC2 instances. Option C is incorrect because the EC2 instances are launched by the Auto Scaling group, not by the AWS: EC2: Instance resource. Option D is incorrect because using a custom resource and a Lambda function is unnecessary and overly complex for this scenario. Reference: AWS::AutoScaling::LaunchConfiguration, Amazon ECS Container Agent Configuration

Question: 189

A DevOps engineer is implementing governance controls for a company that requires its infrastructure to be housed within the United States. The engineer must restrict which AWS Regions can be used, and ensure an alert is sent as soon as possible if any activity outside the governance policy takes place. The controls should be automatically enabled on any new Region outside the United States (US).

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

A. Create an AWS Organizations SCP that denies access to all non-global services in non-US Regions. Attach the policy to the root

of the organization.

B. Configure AWS CloudTrail to send logs to Amazon CloudWatch Logs and enable it for all Regions. Use a CloudWatch Logs metric filter to send an alert on any service activity in non-US Regions.

C. Use an AWS Lambda function that checks for AWS service activity and deploy it to all Regions. Write an Amazon EventBridge rule that runs the Lambda function every hour, sending an alert if activity is found in a non-US Region.

D. Use an AWS Lambda function to query Amazon Inspector to look for service activity in non-US Regions and send alerts if any activity is found.

E. Write an SCP using the aws: RequestedRegion condition key limiting access to US Regions. Apply the policy to all users, groups, and roles

---

Answer: AB

Explanation:

To implement governance controls that restrict AWS service usage to within the United States and ensure alerts for any activity outside the governance policy, the following actions will meet the requirements:

A. Create an AWS Organizations SCP that denies access to all non-global services in non-US Regions. Attach the policy to the root of the organization. This action will effectively prevent users and roles in all accounts within the organization from accessing services in non-US Regions12.

B. Configure AWS CloudTrail to send logs to Amazon CloudWatch Logs and enable it for all Regions. Use a CloudWatch Logs metric filter to send an alert on any service activity in non-US Regions. This action will allow monitoring of all AWS Regions and will trigger alerts if any activity is detected in non-US Regions, ensuring that the governance team is notified as soon as possible3.

Reference:

AWS Documentation on Service Control Policies (SCPs) and how they can be used to manage permissions and restrict access based on Regions12.

AWS Documentation on monitoring CloudTrail log files with Amazon CloudWatch Logs to set up alerts for specific activities3.

Question: 190

A company is using AWS to run digital workloads. Each application team in the company has its own AWS account for application hosting. The accounts are consolidated in an organization in AWS Organizations.

The company wants to enforce security standards across the entire organization. To avoid noncompliance because of security misconfiguration, the company has enforced the use of AWS CloudFormation. A production support team can modify resources in the production environment by using the AWS Management Console to troubleshoot and resolve application-related issues.

A DevOps engineer must implement a solution to identify in near real time any AWS service misconfiguration that results in noncompliance. The solution must automatically remediate the issue within 15 minutes of identification. The solution also must track noncompliant resources and events in a centralized dashboard with accurate timestamps.

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

A. Use CloudFormation drift detection to identify noncompliant resources. Use drift detection events from CloudFormation to invoke an AWS Lambda function for remediation. Configure the Lambda function to publish logs to an Amazon CloudWatch Logs log group. Configure an Amazon CloudWatch dashboard to use the log group for tracking.

B. Turn on AWS CloudTrail in the AWS accounts. Analyze CloudTrail logs by using Amazon Athena to identify noncompliant resources. Use AWS Step Functions to track query results on Athena for drift detection and to invoke an AWS Lambda function for remediation. For tracking, set up an Amazon QuickSight dashboard that uses Athena as the data source.

C. Turn on the configuration recorder in AWS Config in all the AWS accounts to identify noncompliant resources. Enable AWS Security Hub with the ~no-enable-default-standards option in all the AWS accounts. Set up AWS Config managed rules and custom rules. Set up automatic remediation by using AWS Config conformance packs. For tracking, set up a dashboard on Security Hub in a designated Security Hub administrator account.

D. Turn on AWS CloudTrail in the AWS accounts. Analyze CloudTrail logs by using Amazon CloudWatch Logs to identify noncompliant resources. Use CloudWatch Logs filters for drift detection. Use Amazon EventBridge to invoke the Lambda function for remediation. Stream filtered CloudWatch logs to Amazon OpenSearch Service. Set up a dashboard on OpenSearch Service for tracking.

Answer: C

Explanation:

The best solution is to use AWS Config and AWS Security Hub to identify and remediate noncompliant resources across multiple AWS accounts. AWS Config enables continuous monitoring of the configuration of AWS resources and evaluates them against desired configurations. AWS Config can also automatically remediate noncompliant resources by using conformance packs, which are a collection of AWS Config rules and remediation actions that can be deployed as a single entity. AWS Security Hub provides a comprehensive view of the security posture of AWS accounts and resources. AWS Security Hub can aggregate and normalize the findings from AWS Config and other AWS services, as well as from partner solutions. AWS Security Hub can also be used to create a dashboard for tracking noncompliant resources and events in a centralized location.

The other options are not optimal because they either require more development overhead, do not provide near real time detection and remediation, or do not provide a centralized dashboard for tracking.

Option A is not optimal because CloudFormation drift detection is not a near real time solution. Drift detection has to be manually initiated on each stack or resource, or scheduled using a cron expression. Drift detection also does not provide remediation actions, so a custom Lambda function has to be developed and invoked. CloudWatch Logs and dashboard can be used for tracking, but they do not provide a comprehensive view of the security posture of the AWS accounts and resources. Option B is not optimal because CloudTrail logs analysis using Athena is not a near real time solution. Athena queries have to be manually run or scheduled using a cron expression. Athena also does not provide remediation actions, so a custom Lambda function has to be developed and invoked. Step Functions can be used to orchestrate the query and remediation workflow, but it adds more complexity and cost. QuickSight dashboard can be used for tracking, but it does not provide a comprehensive view of the security posture of the AWS accounts and resources.

Option D is not optimal because CloudTrail logs analysis using CloudWatch Logs is not a near real time solution. CloudWatch Logs filters have to be manually created or updated for each resource type and configuration change. CloudWatch Logs also does not provide remediation actions, so a custom Lambda function has to be developed and invoked. EventBridge can be used to trigger the Lambda function, but it adds more complexity and cost. OpenSearch Service dashboard can be used for tracking, but it does not provide a comprehensive view of the security posture of the AWS accounts and resources.

Reference:

AWS Config conformance packs

Introducing AWS Config conformance packs

Managing conformance packs across all accounts in your organization

Question: 191

A company is building a web and mobile application that uses a serverless architecture powered by AWS Lambda and Amazon API Gateway The company wants to fully automate the backend Lambda deployment based on code that is pushed to the appropriate environment branch in an AWS CodeCommit repository

The deployment must have the following:

- Separate environment pipelines for testing and production
- Automatic deployment that occurs for test environments only

Which steps should be taken to meet these requirements'?

A. Configure a new AWS CodePipelme service Create a CodeCommit repository for each environment Set up CodePipeline to retrieve the source code from the appropriate repository Set up the deployment step to deploy the Lambda functions with AWS CloudFormation.

B. Create two AWS CodePipeline configurations for test and production environments Configure the production pipeline to have a manual approval step Create a CodeCommit repository for each environment Set up each CodePipeline to retrieve the source code from the appropriate repository Set up the deployment step to deploy the Lambda functions with AWS CloudFormation.

C. Create two AWS CodePipeline configurations for test and production environments Configure the production pipeline to have a manual approval step. Create one CodeCommit repository with a branch for each environment Set up each CodePipeline to retrieve the source code from the appropriate branch in the repository. Set up the deployment step to deploy the Lambda functions with AWS

CloudFormation

D. Create an AWS CodeBuild configuration for test and production environments Configure the production pipeline to have a manual approval step. Create one CodeCommit repository with a branch for each environment Push the Lambda function code to an Amazon S3 bucket Set up the deployment step to deploy the Lambda functions from the S3 bucket.

Answer: C

Explanation:

The correct approach to meet the requirements for separate environment pipelines and automatic deployment for test environments is to create two AWS CodePipeline configurations, one for each environment. The production pipeline should have a manual approval step to ensure that changes are reviewed before being deployed to production. A single AWS CodeCommit repository with separate branches for each environment allows for organized and efficient code management. Each CodePipeline retrieves the source code from the appropriate branch in the repository. The deployment step utilizes AWS CloudFormation to deploy the Lambda functions, ensuring that the infrastructure as code is maintained and version-controlled.

Reference:

AWS Lambda with Amazon API Gateway: Using AWS Lambda with Amazon API Gateway Tutorial on using Lambda with API Gateway: Tutorial: Using Lambda with API Gateway AWS CodePipeline automatic deployment: Set Up a Continuous Deployment Pipeline Using AWS CodePipeline

Building a pipeline for test and production stacks: Walkthrough: Building a pipeline for test and production stacks

Question: 192

A DevOps engineer is using AWS CodeDeploy across a fleet of Amazon EC2 instances in an EC2 Auto Scaling group. The associated CodeDeploy deployment group, which is integrated with EC2 Auto Scaling, is configured to perform in-place deployments with codeDeployDefault.oneAtATime During an ongoing new deployment, the engineer discovers that, although the overall deployment finished successfully, two out of five instances have the previous application revision deployed. The other three instances have the newest application revision

What is likely causing this issue?

A. The two affected instances failed to fetch the new deployment.

B. A failed Afterinstall lifecycle event hook caused the CodeDeploy agent to roll back to the previous version on the affected instances

C. The CodeDeploy agent was not installed in two affected instances.

D. EC2 Auto Scaling launched two new instances while the new deployment had not yet finished, causing the previous version to be deployed on the affected instances.

_____ Answer: B

Explanation:

When AWS CodeDeploy performs an in-place deployment, it updates the instances with the new application revision one at a time, as specified by the deployment configuration codeDeployDefault.oneAtATime. If a lifecycle event hook, such as AfterInstall, fails during the deployment, CodeDeploy will attempt to roll back to the previous version on the affected instances. This is likely what happened with the two instances that still have the previous application revision deployed. The failure of the AfterInstall lifecycle event hook triggered the rollback mechanism, resulting in those instances reverting to the previous application revision.

Reference:

AWS CodeDeploy documentation on redeployment and rollback procedures!.

Stack Overflow discussions on re-deploying older revisions with AWS CodeDeploy2.

AWS CLI reference guide for deploying a revision2.

Question: 193

A company recently deployed its web application on AWS. The company is preparing for a large-scale sales event and must ensure that the web application can scale to meet the demand
The application's frontend infrastructure includes an Amazon CloudFront distribution that has an Amazon S3 bucket as an origin. The backend infrastructure includes an Amazon API Gateway API. several AWS Lambda functions, and an Amazon Aurora DB cluster
The company's DevOps engineer conducts a load test and identifies that the Lambda functions can fulfill the peak number of requests However, the DevOps engineer notices request latency during the initial burst of requests Most of the requests to the Lambda functions produce queries to the database A large portion of the invocation time is used to establish database connections Which combination of steps will provide the application with the required scalability? (Select TWO)

A. Configure a higher reserved concurrency for the Lambda functions.
B. Configure a higher provisioned concurrency for the Lambda functions
C. Convert the DB cluster to an Aurora global database Add additional Aurora Replicas in AWS Regions based on the locations of the company's customers.
D. Refactor the Lambda Functions Move the code blocks that initialize database connections into the function handlers.
E. Use Amazon RDS Proxy to create a proxy for the Aurora database Update the Lambda functions to use the proxy endpoints for database connections.

Answer: BE

Explanation:

The correct answer is B and E. Configuring a higher provisioned concurrency for the Lambda functions will ensure that the functions are ready to respond to the initial burst of requests without any cold start latency. Using Amazon RDS Proxy to create a proxy for the Aurora database will enable the Lambda functions to reuse existing database connections and reduce the overhead of establishing new ones. This will also improve the scalability and availability of the database by managing the connection pool size and handling failovers. Option A is incorrect because reserved concurrency only limits the number of concurrent executions for a function, not pre-warms them.

Option C is incorrect because converting the DB cluster to an Aurora global database will not address the issue of database connection latency, and may introduce additional costs and complexity. Option D is incorrect because moving the code blocks that initialize database connections into the function handlers will not improve the performance or scalability of the Lambda functions, and may actually worsen the cold start latency. Reference:

AWS Lambda Provisioned Concurrency
Using Amazon RDS Proxy with AWS Lambda
Certified DevOps Engineer - Professional (DOP-C02) Study Guide (page 173)

Question: 194

A DevOps engineer wants to find a solution to migrate an application from on premises to AWS The application is running on Linux and needs to run on specific versions of Apache Tomcat HAProxy and Varnish Cache to function properly. The application's operating system-level parameters require tuning The solution must include a way to automate the deployment of new application versions. The infrastructure should be scalable and faulty servers should be replaced automatically.
Which solution should the DevOps engineer use?

A. Upload the application as a Docker image that contains all the necessary software to Amazon ECR Create an Amazon ECS cluster using an AWS Fargate launch type and an Auto Scaling group. Create an AWS CodePipeline pipeline that uses Amazon ECR as a source and

Amazon ECS as a deployment provider

B. Upload the application code to an AWS CodeCommit repository with a saved configuration file to configure and install the software Create an AWS Elastic Beanstalk web server tier and a load balanced-type environment that uses the Tomcat solution stack Create an AWS CodePipeline pipeline that uses CodeCommit as a source and Elastic Beanstalk as a deployment provider

C. Upload the application code to an AWS CodeCommit repository with a set of ebextensions files to configure and install the software. Create an AWS Elastic Beanstalk worker tier environment that uses the Tomcat solution stack Create an AWS CodePipeline pipeline that uses CodeCommit as a source and Elastic Beanstalk as a deployment provider

D. Upload the application code to an AWS CodeCommit repository with an appspec.yml file to configure and install the necessary software. Create an AWS CodeDeploy deployment group associated with an Amazon EC2 Auto Scaling group Create an AWS CodePipeline pipeline that uses CodeCommit as a source and CodeDeploy as a deployment provider

---

Answer: D

Explanation:

The correct answer is D. The scenario requires a solution that can migrate an application from on premises to AWS, run on specific versions of Apache Tomcat, HAProxy, and Varnish Cache, tune the operating system-level parameters, automate the deployment of new application versions, and scale and replace faulty servers automatically. Option D meets all these requirements by using AWS CodeCommit, AWS CodeDeploy, AWS CodePipeline, and Amazon EC2 Auto Scaling. AWS CodeCommit is a fully managed source control service that hosts Git repositories and works with Git- based tools. AWS CodeDeploy is a fully managed deployment service that automates software deployments to a variety of compute services, including Amazon EC2, AWS Fargate, AWS Lambda, and on-premises servers. AWS CodePipeline is a fully managed continuous delivery service that helps automate the release pipelines for fast and reliable application updates. Amazon EC2 Auto

Scaling helps maintain application availability and allows scaling of Amazon EC2 capacity up or down automatically according to the defined conditions. By using these services together, the DevOps engineer can migrate the application code to AWS, configure and install the necessary software using the appspec.yml file, automate the deployment process using the pipeline, and scale and replace the servers using the Auto Scaling group.

Option A is incorrect because AWS Fargate is a serverless compute engine for containers that works with Amazon ECS and Amazon EKS. Fargate removes the need to provision and manage servers, but it also limits the ability to tune the operating system-level parameters, which is a requirement in the scenario. Moreover, Fargate does not support HAProxy and Varnish Cache as sidecar containers, which are needed to run the application properly.

Option B is incorrect because AWS Elastic Beanstalk is a fully managed service that automates the deployment and scaling of web applications and services using familiar servers such as Apache, Nginx, Passenger, and IIS. However, Elastic Beanstalk does not support HAProxy and Varnish Cache as part of the Tomcat solution stack, which are needed to run the application properly. Moreover, Elastic Beanstalk web server tier environments are designed to serve HTTP requests, not to process background tasks, which is the purpose of worker tier environments.

Option C is incorrect because AWS Elastic Beanstalk worker tier environments are designed to process background tasks using a daemon process that runs on each Amazon EC2 instance in the environment. Worker tier environments are not suitable for running web applications that serve HTTP requests, which is the case in the scenario. Moreover, Elastic Beanstalk does not support HAProxy and Varnish Cache as part of the Tomcat solution stack, which are needed to run the application properly.

Reference:

AWS CodeCommit

AWS CodeDeploy

AWS CodePipeline

Amazon EC2 Auto Scaling

AWS Fargate

AWS Elastic Beanstalk

www.atmicnetworks.com

Question: 195

A company runs a web application that extends across multiple Availability Zones. The company uses an Application Load Balancer (ALB) for routing. AWS Fargate (or the application and Amazon Aurora for the application data The company uses AWS CloudFormation templates to deploy the application The company stores all Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository in the same AWS account and AWS Region.

A DevOps engineer needs to establish a disaster recovery (DR) process in another Region. The solution must meet an RPO of 8 hours and an RTO of 2 hours The company sometimes needs more than 2 hours to build the Docker images from the Dockerfile

Which solution will meet the RTO and RPO requirements MOST cost-effectively?

A. Copy the CloudFormation templates and the Dockerfile to an Amazon S3 bucket in the DR Region Use AWS Backup to configure automated Aurora cross-Region hourly snapshots In case of DR, build the most recent Docker image and upload the Docker image to an ECR repository in the DR Region Use the CloudFormation template that has the most recent Aurora snapshot and the Docker image from the ECR repository to launch a new CloudFormation stack in the DR Region Update the application DNS records to point to the new ALB

B. Copy the CloudFormation templates to an Amazon S3 bucket in the DR Region Configure Aurora automated backup Cross-Region Replication Configure ECR Cross-Region Replication. In case of DR use the CloudFormation template with the most recent Aurora snapshot and the Docker image from the local ECR repository to launch a new CloudFormation stack in the DR Region Update the application DNS records to point to the new ALB

C. Copy the CloudFormation templates to an Amazon S3 bucket in the DR Region. Use Amazon EventBridge to schedule an AWS Lambda function to take an hourly snapshot of the Aurora database and of the most recent Docker image in the ECR repository. Copy the snapshot and the Docker image to the DR Region in case of DR, use the CloudFormation template with the most recent Aurora snapshot and the Docker image from the local ECR repository to launch a new CloudFormation stack in the DR Region

D. Copy the CloudFormation templates to an Amazon S3 bucket in the DR Region. Deploy a second application CloudFormation stack in the DR Region. Reconfigure Aurora to be a global database Update both CloudFormation stacks when a new application release in the current Region is needed. In case of DR. update, the application DNS records to point to the new ALB.

Answer: B

Explanation:

The most cost-effective solution to meet the RTO and RPO requirements is option B. This option involves copying the CloudFormation templates to an Amazon S3 bucket in the DR Region, configuring Aurora automated backup Cross-Region Replication, and configuring ECR Cross-Region Replication. In the event of a disaster, the CloudFormation template with the most recent Aurora snapshot and the Docker image from the local ECR repository can be used to launch a new CloudFormation stack in the DR Region. This approach avoids the need to build Docker images from the Dockerfile, which can sometimes take more than 2 hours, thus meeting the RTO requirement. Additionally, the use of automated backups and replication ensures that the RPO of 8 hours is met. Reference:

AWS Documentation on Disaster Recovery: Plan for Disaster Recovery (DR) - Reliability Pillar AWS Blog on Establishing RPO and RTO Targets: Establishing RPO and RTO Targets for Cloud Applications

AWS Documentation on ECR Cross-Region Replication: Amazon ECR Cross-Region Replication AWS Documentation on Aurora Cross-Region Replication: Replicating Amazon Aurora DB Clusters Across AWS Regions

Question: 196

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'?

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.

Question: 197

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?

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 Organizations! 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 nonadministrator 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: 198

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'?

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: 199

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

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[A]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

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 the Environment=Production tag and adding another single tag group that includes only the Name=ApplicationA tag, 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 the Department=Marketing and Name=ApplicationB tags.

Reference:

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: 200

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'?

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.

www.atmicnetworks.com

_____ 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 the associate-address permission 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: 201

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": "Lancda.amaoonaws.com"
```

B.

Change "Resource": ""to "Resource": "arn:aws:ec2:•:•:instance/-

C.

Add the following conditional expression "Condition": {

```
"StringNotEquals": (
    *'eo2:ResourceTag/Environraent": "Production"
J
```

D.

Add the following conditional expression

```
"Condition": {
  "StringEquals*: {
    "ec2:ResourceTag/Envir:nment": "NonProduction"
```

E.

Change "Action": "ec2:*" to "Action": "ec2:Stopinstances"

F.

Add the following conditional expression

```
"Condition" :          {
    "DateGreaterThan" : {
        "aws:CurrentTome" : "S ;{aws:DateTime:Friday}"
    I /
    "DateLessThan":
        "aws:CurrentTime" : "$ ;{aws:Datelime:Monday■"
```

Answer: ABD

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. Reference:

AWS Identity and Access Management (IAM) - AWS Documentation

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

Question: 202

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?

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: 203

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?

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: 204

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.)

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: AE

Explanation:

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

Question: 205

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?

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: 206

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?

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: 207

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?

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: 208

A company has an application that runs on Amazon EC2 instances behind an Application Load Balancer (ALB) The EC2 Instances are in multiple Availability Zones The application was misconfigured in a single Availability Zone, which caused a partial outage of the application.
A DevOps engineer made changes to ensure that the unhealthy EC2 instances in one Availability Zone do not affect the healthy EC2 instances in the other Availability Zones. The DevOps engineer needs to test the application's failover and shift where the ALB sends traffic During failover. the ALB must avoid sending traffic to the Availability Zone where the failure has occurred.
Which solution will meet these requirements?

A. Turn off cross-zone load balancing on the ALB Use Amazon Route 53 Application Recovery Controller to start a zonal shift away from the Availability Zone

B. Turn off cross-zone load balancing on the ALB's target group Use Amazon Route 53 Application Recovery Controller to start a zonal shift away from the Availability Zone

C. Create an Amazon Route 53 Application Recovery Controller resource set that uses the DNS hostname of the ALB Start a zonal shift for the resource set away from the Availability Zone

D. Create an Amazon Route 53 Application Recovery Controller resource set that uses the ARN of the ALB's target group Create a readiness check that uses the ElbV2TargetGroupsCanServeTraffic rule

Answer: B

Explanation:

Turn off cross-zone load balancing on the ALB's target group:
Cross-zone load balancing distributes traffic evenly across all registered targets in all enabled Availability Zones. Turning this off will ensure that each target group only handles requests from its respective Availability Zone.
To disable cross-zone load balancing:
Go to the Amazon EC2 console.

Navigate to Load Balancers and select the ALB.

Choose the Target Groups tab, select the target group, and then select the Group details tab.

Click on Edit and turn off Cross-zone load balancing.

Use Amazon Route 53 Application Recovery Controller to start a zonal shift away from the Availability Zone:

Amazon Route 53 Application Recovery Controller provides the ability to control traffic flow to ensure high availability and disaster recovery.

By using Route 53 Application Recovery Controller, you can perform a zonal shift to redirect traffic away from the unhealthy Availability Zone.

To start a zonal shift:

Configure Route 53 Application Recovery Controller by creating a cluster and control panel. Create routing controls to manage traffic shifts between Availability Zones.

Use the routing control to shift traffic away from the affected Availability Zone.

Reference:

Disabling cross-zone load balancing

Route 53 Application Recovery Controller

Question: 209

A company has a fleet of Amazon EC2 instances that run Linux in a single AWS account. The company is using an AWS Systems Manager Automation task across the EC2 instances.

During the most recent patch cycle, several EC2 instances went into an error state because of insufficient available disk space. A DevOps engineer needs to ensure that the EC2 instances have sufficient available disk space during the patching process in the future.

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

A. Ensure that the Amazon CloudWatch agent is installed on all EC2 instances

B. Create a cron job that is installed on each EC2 instance to periodically delete temporary files.

C. Create an Amazon CloudWatch log group for the EC2 instances. Configure a cron job that is installed on each EC2 instance to write the available disk space to a CloudWatch log stream for the relevant EC2 instance.

D. Create an Amazon CloudWatch alarm to monitor available disk space on all EC2 instances Add the alarm as a safety control to the Systems Manager Automation task.

E. Create an AWS Lambda function to periodically check for sufficient available disk space on all EC2 instances by evaluating each EC2 instance's respective Amazon CloudWatch log stream.

Answer: A, D

Explanation:

Ensure that the Amazon CloudWatch agent is installed on all EC2 instances:

The Amazon CloudWatch agent collects and logs metrics and sends them to Amazon CloudWatch. To install the CloudWatch agent:

Download the CloudWatch agent package.

Install the agent on your EC2 instances.

Configure the agent to collect disk space metrics.

Create an Amazon CloudWatch alarm to monitor available disk space on all EC2 instances Add the alarm as a safety control to the Systems Manager Automation task:

Create CloudWatch alarms to monitor the available disk space and trigger notifications or actions when the disk space falls below a defined threshold.

Add the CloudWatch alarm to the Systems Manager Automation task to halt or fail the task if disk space is insufficient.

To create the alarm:

Navigate to the CloudWatch console and create a new alarm.
Set the metric to monitor (e.g., disk space utilization).
Define the threshold and notification actions.

Reference:

Amazon CloudWatch agent
Creating Amazon CloudWatch alarms

Question: 210

A company is migrating from its on-premises data center to AWS. The company currently uses a custom on-premises CI/CD pipeline solution to build and package software.
The company wants its software packages and dependent public repositories to be available in AWS CodeArtifact to facilitate the creation of application-specific pipelines.
Which combination of steps should the company take to update the CI/CD pipeline solution and to configure CodeArtifact with the LEAST operational overhead? (Select TWO.)

A. Update the CI/CD pipeline to create a VM image that contains newly packaged software Use AWS Import/Export to make the VM image available as an
Amazon EC2 AMI. Launch the AMI with an attached 1AM instance profile that allows CodeArtifact actions. Use AWS CLI commands to publish the packages to a CodeArtifact repository.
B. Create an AWS Identity and Access Management Roles Anywhere trust anchor Create an 1AM role that allows CodeArtifact actions and that has a trust relationship on the trust anchor. Update the onpremises CI/CD pipeline to assume the new 1AM role and to publish the packages to CodeArtifact.
C. Create a new Amazon S3 bucket. Generate a presigned URL that allows the PutObject request. Update the on-premises CI/CD pipeline to use the
presigned URL to publish the packages from the on-premises location to the S3 bucket. Create an AWS Lambda function that runs when packages are created in the bucket through a put command Configure the Lambda function to publish the packages to CodeArtifact
D. For each public repository, create a CodeArtifact repository that is configured with an external connection Configure the dependent repositories as upstream public repositories.
E. Create a CodeArtifact repository that is configured with a set of external connections to the public repositories. Configure the external connections to be downstream of the repository

Answer: B, D

Explanation:

Create an AWS Identity and Access Management Roles Anywhere trust anchor Create an IAM role that allows CodeArtifact actions and that has a trust relationship on the trust anchor. Update the onpremises CI/CD pipeline to assume the new IAM role and to publish the packages to CodeArtifact: Roles Anywhere allows on-premises servers to assume IAM roles, making it easier to integrate onpremises environments with AWS services.

Steps:

Create a trust anchor in IAM.
Create an IAM role with permissions for CodeArtifact actions (e.g., publishing packages).
Update the CI/CD pipeline to assume this role using the trust anchor.
Create a new Amazon S3 bucket. Generate a presigned URL that allows the PutObject request. Update the on-premises CI/CD pipeline to use the presigned URL to publish the packages from the on-premises location to the S3 bucket. Create an AWS Lambda function that runs when packages are created in the bucket through a put command Configure the Lambda function to publish the packages to CodeArtifact:

Using an S3 bucket as an intermediary, you can easily upload packages from on-premises systems. Steps:
Create an S3 bucket.

Generate presigned URLs to allow the CI/CD pipeline to upload packages.
Configure an AWS Lambda function to trigger on S3 PUT events and publish the packages to CodeArtifact.

Reference:

IAM Roles Anywhere
Amazon S3 presigned URLs
AWS Lambda function triggers

Question: 211

A company operates sensitive workloads across the AWS accounts that are in the company's organization in AWS Organizations The company uses an IP address range to delegate IP addresses for Amazon VPC CIDR blocks and all non-cloud hardware.
The company needs a solution that prevents principals that are outside the company's IP address range from performing AWS actions In the organization's accounts
Which solution will meet these requirements?

A. Configure AWS Firewall Manager for the organization. Create an AWS Network Firewall policy that allows only source traffic from the company's IP address range Set the policy scope to all accounts in the organization.

B. In Organizations, create an SCP that denies source IP addresses that are outside of the company s IP address range. Attach the SCP to the organization's root

C. Configure Amazon GuardDuty for the organization. Create a GuardDuty trusted IP address list for the company's IP range Activate the trusted IP list for the organization.

D. In Organizations, create an SCP that allows source IP addresses that are inside of the company s IP address range. Attach the SCP to the organization's root.

Answer: B

Explanation:

https://docs.aws.amazon.com/IAM/latest/UserGuide/reference policies examples aws deny- ip.html

Question: 212

A company uses AWS Organizations to manage its AWS accounts. The organization root has a child OU that is named Department. The Department OU has a child OU that is named Engineering. The default FullAWSAccess policy is attached to the root, the Department OU. and the Engineering OU. The company has many AWS accounts in the Engineering OU. Each account has an administrative 1AM role with the AdmmistratorAccess 1AM policy attached. The default FullAWSAccessPolicy is also attached to each account.
A DevOps engineer plans to remove the FullAWSAccess policy from the Department OU The DevOps engineer will replace the policy with a policy that contains an Allow statement for all Amazon EC2 API operations.
What will happen to the permissions of the administrative 1AM roles as a result of this change'?

A. All API actions on all resources will be allowed
B. All API actions on EC2 resources will be allowed. All other API actions will be denied.
C. All API actions on all resources will be denied
D. All API actions on EC2 resources will be denied. All other API actions will be allowed.

www.atmicnetworks.com

Answer: B

Explanation:

Impact of Removing FullAWSAccess and Adding Policy for EC2 Actions:

The FullAWSAccess policy allows all actions on all resources by default. Removing this policy from the Department OU will limit the permissions that accounts within this OU inherit from the parent OU. Adding a policy that allows only Amazon EC2 API operations will restrict the permissions to EC2 actions only.

Permissions of Administrative IAM Roles:

The administrative IAM roles in the Engineering OU have the AdministratorAccess policy attached, which grants full access to all AWS services and resources.

Since SCPs are restrictions that apply at the organizational level, removing FullAWSAccess and replacing it with a policy allowing only EC2 actions means that for all accounts in the Engineering OU: They will have full access to EC2 actions due to the new SCP.

They will be restricted in other actions that are not covered by the SCP, hence, non-EC2 API actions will be denied.

Conclusion:

All API actions on EC2 resources will be allowed.

All other API actions will be denied due to the absence of a broader allow policy.

Question: 213

A company is running a custom-built application that processes records. All the components run on Amazon EC2 instances that run in an Auto Scaling group. Each record's processing is a multistep sequential action that is compute-intensive. Each step is always completed in 5 minutes or less.

A limitation of the current system is that if any steps fail, the application has to reprocess the record from the beginning The company wants to update the architecture so that the application must reprocess only the failed steps.

What is the MOST operationally efficient solution that meets these requirements?

A. Create a web application to write records to Amazon S3 Use S3 Event Notifications to publish to an Amazon Simple Notification Service (Amazon SNS) topic Use an EC2 instance to poll Amazon SNS and start processing Save intermediate results to Amazon S3 to pass on to the next step

B. Perform the processing steps by using logic in the application. Convert the application code to run in a container. Use AWS Fargate to manage the container Instances. Configure the container to invoke itself to pass the state from one step to the next.

C. Create a web application to pass records to an Amazon Kinesis data stream. Decouple the processing by using the Kinesis data stream and AWS Lambda functions.

D. Create a web application to pass records to AWS Step Functions. Decouple the processing into Step Functions tasks and AWS Lambda functions.

_____ Answer: D

Explanation:

Use AWS Step Functions to Orchestrate Processing:

AWS Step Functions allow you to build distributed applications by combining AWS Lambda functions or other AWS services into workflows.

Decoupling the processing into Step Functions tasks enables you to retry individual steps without reprocessing the entire record.

Architectural Steps:

Create a web application to pass records to AWS Step Functions:

The web application can be a simple frontend that receives input and triggers the Step Functions workflow.

Define a Step Functions state machine:

Each step in the state machine represents a processing stage. If a step fails, Step Functions can retry the step based on defined conditions.

Use AWS Lambda functions:

Lambda functions can be used to handle each processing step. These functions can be stateless and handle specific tasks, reducing the complexity of error handling and reprocessing logic.

Operational Efficiency:

Using Step Functions and Lambda improves operational efficiency by providing built-in error handling, retries, and state management.

This architecture scales automatically and isolates failures to individual steps, ensuring only failed steps are retried.

Reference:

AWS Step Functions

Building Workflows with Step Functions

Question: 214

A company has an organization in AWS Organizations. A DevOps engineer needs to maintain multiple AWS accounts that belong to different OUs in the organization. All resources, including 1AM policies and Amazon S3 policies within an account, are deployed through AWS CloudFormation. All templates and code are maintained in an AWS CodeCommit repository Recently, some developers have not been able to access an S3 bucket from some accounts in the organization.

The following policy is attached to the S3 bucket.

```
"Statement": [
        (
            "Action": [
                "s 3:Putobj ect•"
            J ,
            "Principal": ( "AWS": "arn:aws:sts:^assumed-role/developer-role/access-session" ), "Resource": [
                "arn:aws:s3:::DOC-£XAMPLE-BUCKET",
                "arn: aws: s3::: DOC-EXAMPLE-BUCKET/ •"
]
```

What should the DevOps engineer do to resolve this access issue?

A. Modify the S3 bucket policy Turn off the S3 Block Public Access setting on the S3 bucket In the S3 policy, add the awsSourceAccount condition. Add the AWS account IDs of all developers who are experiencing the issue.

B. Verify that no 1AM permissions boundaries are denying developers access to the S3 bucket Make the necessary changes to IAM permissions boundaries. Use an AWS Config recorder in the individual developer accounts that are experiencing the issue to revert any changes that are blocking access. Commit the fix back into the CodeCommit repository. Invoke deployment through Cloud Formation to apply the changes.

C. Configure an SCP that stops anyone from modifying 1AM resources in developer OUs. In the S3 policy, add the awsSourceAccount condition. Add the AWS account IDs of all developers who are experiencing the issue Commit the fix back into the CodeCommit repository Invoke deployment through CloudFormation to apply the changes

D. Ensure that no SCP is blocking access for developers to the S3 bucket Ensure that no 1AM policy permissions boundaries are denying access to developer 1AM users Make the necessary changes to the SCP and 1AM policy permissions boundaries in the CodeCommit repository Invoke deployment through CloudFormation to apply the changes

Answer: D

Explanation:

Verify No SCP Blocking Access:
Ensure that no Service Control Policy (SCP) is blocking access for developers to the S3 bucket. SCPs are applied at the organization or organizational unit (OU) level in AWS Organizations and can restrict what actions users and roles in the affected accounts can perform.
Verify No IAM Policy Permissions Boundaries Blocking Access:
IAM permissions boundaries can limit the maximum permissions that a user or role can have. Verify that these boundaries are not restricting access to the S3 bucket.
Make Necessary Changes to SCP and IAM Policy Permissions Boundaries:
Adjust the SCPs and IAM permissions boundaries if they are found to be the cause of the access issue. Make sure these changes are reflected in the code maintained in the AWS CodeCommit repository.
Invoke Deployment Through CloudFormation:
Commit the updated policies to the CodeCommit repository.
Use AWS CloudFormation to deploy the changes across the relevant accounts and resources to ensure that the updated permissions are applied consistently.
By ensuring no SCPs or IAM policy permissions boundaries are blocking access and making necessary changes if they are, the DevOps engineer can resolve the access issue for developers trying to access the S3 bucket.

Reference:
AWS SCPs
IAM Permissions Boundaries
Deploying CloudFormation Templates

Question: 215

A company is developing a web application's infrastructure using AWS CloudFormation The database engineering team maintains the database resources in a Cloud Formation template, and the software development team maintains the web application resources in a separate CloudFormation template. As the scope of the application grows, the software development team needs to use resources maintained by the database engineering team However, both teams have their own review and lifecycle management processes that they want to keep. Both teams also require resource-level change-set reviews. The software development team would like to deploy changes to this template using their CI/CD pipeline.
Which solution will meet these requirements?

A. Create a stack export from the database CloudFormation template and import those references into the web application CloudFormation template
B. Create a CloudFormation nested stack to make cross-stack resource references and parameters available in both stacks.
C. Create a CloudFormation stack set to make cross-stack resource references and parameters available in both stacks.
D. Create input parameters in the web application CloudFormation template and pass resource names and IDs from the database stack.

_____ Answer: A

Explanation:

Stack Export and Import:
Use the Export feature in CloudFormation to share outputs from one stack (e.g., database resources) and use them as inputs in another stack (e.g., web application resources).
Steps to Create Stack Export:

Define the resources in the database CloudFormation template and use the Outputs section to export necessary values.

```
Outputs:
DBInstanceEndpoint:
  Value: !GetAtt DBInstance.Endpoint.Address
  Export:
    Name: DBInstanceEndpoint
```

Steps to Import into Web Application Stack:

In the web application CloudFormation template, use the ImportValue function to import these exported values.

```
Resources:
MyResource:
  Type: "AWS::SomeResourceType"
  Properties:
    SomeProperty: !ImportValue DBInstanceEndpoint
```

Resource-Level Change-Set Reviews:

Both teams can continue using their respective review processes, as changes to each stack are managed independently.
Use CloudFormation change sets to preview changes before deploying.

By exporting resources from the database stack and importing them into the web application stack, both teams can maintain their separate review and lifecycle management processes while sharing necessary resources.

Reference:

AWS CloudFormation Export
AWS CloudFormation ImportValue

Question: 216

A company uses Amazon RDS for all databases in Its AWS accounts The company uses AWS Control Tower to build a landing zone that has an audit and logging account All databases must be encrypted at rest for compliance reasons. The company's security engineer needs to receive notification about any noncompliant databases that are in the company's accounts
Which solution will meet these requirements with the MOST operational efficiency?

A. Use AWS Control Tower to activate the optional detective control (guardrail) to determine whether the RDS storage is encrypted Create an Amazon Simple Notification Service (Amazon SNS) topic in the company's audit account. Create an Amazon EventBridge rule to filter noncompliant events from the AWS Control Tower control (guardrail) to notify the SNS topic. Subscribe the security engineer's email address to the SNS topic

B. Use AWS Cloud Formation StackSets to deploy AWS Lambda functions to every account. Write the Lambda function code to determine whether the RDS storage is encrypted in the account the function is deployed to Send the findings as an Amazon CloudWatch metric to the management account Create an Amazon Simple Notification Service (Amazon SNS) topic. Create a CloudWatch alarm that notifies the SNS topic when metric thresholds are met. Subscribe the security engineer's email address to the SNS topic.

C. Create a custom AWS Config rule in every account to determine whether the RDS storage is encrypted Create an Amazon Simple Notification Service (Amazon SNS) topic in the audit account Create an Amazon EventBridge rule to filter noncompliant events from the AWS Control Tower control (guardrail) to notify the SNS topic. Subscribe the security engineer's email address to the SNS topic

D. Launch an Amazon EC2 instance. Run an hourly cron job by using the AWS CLI to determine whether the RDS storage is encrypted in each AWS account Store the results in an RDS database. Notify the security engineer by sending email messages from the EC2 instance when noncompliance is detected

Answer: A

Explanation:

Activate AWS Control Tower Guardrail:

Use AWS Control Tower to activate a detective guardrail that checks whether RDS storage is encrypted.

Create SNS Topic for Notifications:

Set up an Amazon Simple Notification Service (SNS) topic in the audit account to receive notifications about non-compliant databases.

Create EventBridge Rule to Filter Non-compliant Events:

Create an Amazon EventBridge rule that filters events related to the guardrail's findings on non- compliant RDS instances. Configure the rule to send notifications to the SNS topic when non-compliant events are detected.

Subscribe Security Engineer's Email to SNS Topic:

Subscribe the security engineer's email address to the SNS topic to receive notifications when non- compliant databases are detected.

By using AWS Control Tower to activate a detective guardrail and setting up SNS notifications for non- compliant events, the company can efficiently monitor and ensure that all RDS databases are encrypted at rest.

Reference:

AWS Control Tower Guardrails

Amazon SNS

Amazon EventBridge

Question: 217

A company has an AWS Control Tower landing zone. The company's DevOps team creates a workload OU. A development OU and a production OU are nested under the workload OU. The company grants users full access to the company's AWS accounts to deploy applications.

The DevOps team needs to allow only a specific management 1AM role to manage the 1AM roles and policies of any AWS accounts In only the production OU.

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

A. Create an SCP that denies full access with a condition to exclude the management 1AM role for the organization root.

B. Ensure that the FullAWSAccess SCP is applied at the organization root

C. Create an SCP that allows IAM related actions Attach the SCP to the development OU

D. Create an SCP that denies IAM related actions with a condition to exclude the management I AM role Attach the SCP to the workload OU

E. Create an SCP that denies IAM related actions with a condition to exclude the management 1AM role Attach the SCP to the production OU

Answer: B, E

Explanation:

You need to understand how SCP inheritance works in AWS. The way it works for Deny policies is different that allow policies.

Allow polices are passing down to children ONLY if they don't have an allow policy.

Deny policies always pass down to children.

That's why there is always an SCP set to the Root to allow everything by default. If you limit this policy, the whole organization will be limited, not matter what other policies are saying for the other OUs. So it's not A. It's not D because it restricts the wrong OU.

Question: 218

A company has set up AWS CodeArtifact repositories with public upstream repositories The company's development team consumes open source dependencies from the repositories in the company's internal network.

The company's security team recently discovered a critical vulnerability in the most recent version of a package that the development team consumes. The security team has produced a patched version to fix the vulnerability. The company needs to prevent the vulnerable version from being downloaded. The company also needs to allow the security team to publish the patched version.

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

A. Update the status of the affected CodeArtifact package version tounlisted
B. Update the status of the affected CodeArtifact package version todeleted
C. Update the status of the affected CodeArtifact package version toarchived.
D. Update the CodeArtifact package origin control settings to allow direct publishing and to block upstream operations
E. Update the CodeArtifact package origin control settings to block direct publishing and to allow upstream operations.

Answer: B, D

Explanation:

Update the status of the affected CodeArtifact package version to deleted:

Deleting the vulnerable package version prevents it from being available for download by any users or systems, ensuring that the compromised version is not consumed.

Update the CodeArtifact package origin control settings to allow direct publishing and to block upstream operations:

By allowing direct publishing, the security team can publish the patched version of the package directly to the CodeArtifact repository.

Blocking upstream operations prevents the repository from automatically fetching and serving the vulnerable package version from upstream public repositories.

By deleting the vulnerable version and configuring the origin control settings to allow direct publishing and block upstream operations, the company ensures that only the patched version is available and the vulnerable version cannot be downloaded.

Reference:

Managing Package Versions in CodeArtifact

Package Origin Controls in CodeArtifact

Question: 219

A company deploys an application to Amazon EC2 instances. The application runs Amazon Linux 2 and uses AWS CodeDeploy. The application has the following file structure for its code repository:

appspec.yml
config/config.txt
applicat ion/web

The appspec.yml file has the following contents in the files section:

destination: /usr/local/src/config.tx

www.atmicnetworks.com

ztestiricit icn i /V3r/www/h£nLl

What will the result be for the deployment of the config.txt file?

A. The config.txt file will be deployed to only /var/www/html/config/config txt

B. The config.txt file will be deployed to /usr/local/src/config.txt and to /var/www/html/config/config txt.

C. The config.txt file will be deployed to only /usr/local/src/config txt

D. The config txt file will be deployed to /usr/local/src/config.txt and to /var/www/html/application/web/config txt

Answer: C

Explanation:

Deployment of config.txt file based on the appspec.yml:
The appspec.yml file specifies that config/config.txt should be copied to /usr/local/src/config.txt.
The source: / directive in the appspec.yml indicates that the entire directory structure starting from the root of the application source should be copied to the specified destination, which is /var/www/html.
Result of the Deployment:
The config.txt file will be specifically deployed to /usr/local/src/config.txt as per the explicit file mapping.
The entire directory structure including application/web will be copied to /var/www/html, but this does not include config/config.txt since it has a specific destination defined.
Thus, the config.txt file will be deployed only to /usr/local/src/config.txt.
Therefore, the correct answer is:
C. The config.txt file will be deployed to only /usr/local/src/config.txt.
Reference:
AWS CodeDeploy AppSpec File Reference
AWS CodeDeploy Deployment Process

Question: 220

A company gives its employees limited rights to AWS DevOps engineers have the ability to assume an administrator role. For tracking purposes, the security team wants to receive a near-real-time notification when the administrator role is assumed.
How should this be accomplished?

A. Configure AWS Config to publish logs to an Amazon S3 bucket Use Amazon Athena to query the logs and send a notification to the security team when the administrator role is assumed

B. Configure Amazon GuardDuty to monitor when the administrator role is assumed and send a notification to the security team

C. Create an Amazon EventBridge event rule using an AWS Management Console sign-in events event pattern that publishes a message to an Amazon SNS topic if the administrator role is assumed

D. Create an Amazon EventBridge events rule using an AWS API call that uses an AWS CloudTrail event pattern to invoke an AWS Lambda function that publishes a message to an Amazon SNS topic if the administrator role is assumed.

Answer: D

Explanation:
Create an Amazon EventBridge Rule Using an AWS CloudTrail Event Pattern:
AWS CloudTrail logs API calls made in your account, including actions performed by roles.
Create an EventBridge rule that matches CloudTrail events where the AssumeRole API call is made to assume the administrator role.
Invoke an AWS Lambda Function:

Configure the EventBridge rule to trigger a Lambda function whenever the rule's conditions are met. The Lambda function will handle the logic to send a notification.

Publish a Message to an Amazon SNS Topic:

The Lambda function will publish a message to an SNS topic to notify the security team.

Subscribe the security team's email address to this SNS topic to receive real-time notifications. Example EventBridge rule pattern:

```
{
"source": ["aws.cloudtrail"],
"detail-type": ["AWS API Call via CloudTrail"],
"detail": {
  "eventSource": ["sts.amazonaws.com"],
  "eventName": ["AssumeRole"],
  "requestParameters": {
    "roleArn": ["arn:aws:iam::<account-id>:role/AdministratorRole"]
  }
}
}
```

Example Lambda function (Node.js) to publish to SNS:

```
const AWS = require('aws-sdk');
const sns = new AWS.SNS();

exports.handler = async (event) => {
  const params = {
    Message: 'Administrator role assumed: ${JSON.stringify(event.detail)}',
    TopicArn: 'arn:aws:sns:<region>:<account-id>:<sns-topic>'
  };
  await sns.publish(params).promise();
};
```

Reference:

Creating EventBridge Rules

Using AWS Lambda with Amazon SNS

Question: 221

A DevOps engineer has created an AWS CloudFormation template that deploys an application on Amazon EC2 instances The EC2 instances run Amazon Linux The application is deployed to the EC2 instances by using shell scripts that contain user dat a. The EC2 instances have an 1AM instance profile that has an 1AM role with the AmazonSSMManagedlnstanceCore managed policy attached

The DevOps engineer has modified the user data in the CloudFormation template to install a new version of the application. The engineer has also applied the stack update. However, the application was not updated on the running EC2 instances. The engineer needs to ensure that the changes to the application are installed on the running EC2 instances.

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

A. Configure the user data content to use the Multipurpose Internet Mail Extensions (MIME) multipart format. Set the scripts-user parameter to always in the text/cloud-config section.

B. Refactor the user data commands to use the cfn-init helper script. Update the user data to install and configure the cfn-hup and cfn-mit helper scripts to monitor and apply the metadata changes

C. Configure an EC2 launch template for the EC2 instances. Create a new EC2 Auto Scaling group. Associate the Auto Scaling group with the EC2 launch template Use the AutoScalingScheduledAction update policy for the Auto Scaling group.

D. Refactor the user data commands to use an AWS Systems Manager document (SSM document). Add an AWS CLI command in the

user data to use Systems Manager Run Command to apply the SSM document to the EC2 instances

E. Refactor the user data command to use an AWS Systems Manager document (SSM document) Use Systems Manager State Manager to create an association between the SSM document and the EC2 instances.

Answer: B, E

Explanation:

Refactor User Data to Use cfn-init and cfn-hup:

cfn-init helps to bootstrap the instance, installing packages and starting services.

cfn-hup is a daemon that can monitor metadata changes and re-apply configurations when necessary.

Example user data script with cfn-init:

```
#!/bin/bash
yum update -y
yum install -y aws-cfn-bootstrap
/opt/aws/bin/cfn-init -v --stack ${AWS::StackName} --resource WebServer --region ${AWS::Region} /opt/aws/bin/cfn-hup
```

Use Systems Manager State Manager:

State Manager can automatically apply an AWS Systems Manager document to instances at regular intervals, ensuring configurations are kept up-to-date.

Steps:

Create an SSM document that installs and configures your application.

Use State Manager to associate this document with your EC2 instances.

Example SSM document:

```
{
"schemaVersion": "2.2",
"description": "Install My Application",
"mainSteps": [
  {
    "action": "aws:runShellScript",
    "name": "installApplication",
    "inputs": {
      "runCommand": [
        "yum install -y my-application"
      ]
    }
  }
]
}
```

Create State Manager association:

```
aws ssm create-association --name "InstallMyApplication" --instance-id <instance-id> --documentversion "\$LATEST"
```

Reference:

Using cfn-init and cfn-hup

AWS Systems Manager State Manager

www.atmicnetworks.com

Question: 222

A company has an application that stores data that includes personally Identifiable Information (Pll) In an Amazon S3 bucket All data Is encrypted with AWS Key Management Service (AWS KMS) customer managed keys. All AWS resources are deployed from an AWS Cloud Formation template. A DevOps engineer needs to set up a development environment for the application in a different AWS account The data in the development environment's S3 bucket needs to be updated once a week from the production environment's S3 bucket.

The company must not move Pll from the production environment without anonymizmg the Pll first The data in each environment must be encrypted with different KMS customer managed keys. Which combination of steps should the DevOps engineer take to meet these requirements? (Select TWO)

A. Activate Amazon Macie on the S3 bucket In the production account Create an AWS Step Functions state machine to initiate a discovery job and redact all Pll before copying files to the S3 bucket in the development account. Give the state machine tasks decrypt permissions on the KMS key in the production account. Give the state machine tasks encrypt permissions on the KMS key in the development account

B. Set up S3 replication between the production S3 bucket and the development S3 bucket Activate Amazon Macie on the development S3 bucket Create an AWS Step Functions state machine to initiate a discovery job and redact all Pll as the files are copied to the development S3 bucket. Give the state machine tasks encrypt and decrypt permissions on the KMS key in the development account.

C. Set up an S3 Batch Operations job to copy files from the production S3 bucket to the development S3 bucket. In the development account, configure an AWS Lambda function to redact all Pll. Configure S3 Object Lambda to use the Lambda function for S3 GET requests Give the Lambda function's 1AM role encrypt and decrypt permissions on the KMS key in the development account.

D. Create a development environment from the CloudFormatlon template in the development account. Schedule an Amazon EventBridge rule to start the AWS Step Functions state machine once a week

E. Create a development environment from the CloudFormation template in the development account. Schedule a cron job on an Amazon EC2 instance to run once a week to start the S3 Batch Operations job.

Answer: AD

Explanation:

Activate Amazon Macie on the Production S3 Bucket:
Macie can identify and protect sensitive data such as PII.

Create a Step Functions state machine to automate data discovery and redaction before copying it to the development environment.

Example Step Functions state machine:

```
{
"Comment": "Anonymize PII and copy data",
"StartAt": "MacieDiscoveryJob",
"States": {
  "MacieDiscoveryJob": {
  "Type": "Task",
    "Resource": "arn:aws:states:::macie:startClassificationJob",
    "End": true
  }
}
}
```

Create a Development Environment from CloudFormation Template:

Deploy the development environment in a new account using the existing CloudFormation template. Schedule an EventBridge rule to start the Step Functions state machine on a weekly basis. EventBridge rule example:

{
"ScheduleExpression": "rate(7 days)",
"StateMachineArn": "arn:aws:states:<region>:<account-id>:stateMachine:AnonymizeAndCopyData" }

By using Macie for data anonymization and Step Functions for automation, you ensure PII is properly handled before data transfer between environments.

Reference:

Amazon Macie

AWS Step Functions

AWS CloudFormation Templates

Question: 223

A DevOps engineer needs to implement integration tests into an existing AWS CodePipelme CI/CD workflow for an Amazon Elastic Container Service (Amazon ECS) service. The CI/CD workflow retrieves new application code from an AWS CodeCommit repository and builds a container image. The CI/CD workflow then uploads the container image to Amazon Elastic Container Registry (Amazon ECR) with a new image tag version.

The integration tests must ensure that new versions of the service endpoint are reachable and that vanous API methods return successful response data The DevOps engineer has already created an ECS cluster to test the service

Which combination of steps will meet these requirements with the LEAST management overhead?

(Select THREE.)

A. Add a deploy stage to the pipeline Configure Amazon ECS as the action provider

B. Add a deploy stage to the pipeline Configure AWS CodeDeploy as the action provider

C. Add an appspec.yml file to the CodeCommit repository

D. Update the image build pipeline stage to output an imagedefinitions json file that references the new image tag.

E. Create an AWS Lambda function that runs connectivity checks and API calls against the service. Integrate the Lambda function with CodePipeline by using aLambda action stage

F. Write a script that runs integration tests against the service. Upload the script to an Amazon S3 bucket. Integrate the script in the S3 bucket with CodePipeline by using an S3 action stage.

Answer: A, D, E

Explanation:

Add a Deploy Stage to the Pipeline, Configure Amazon ECS as the Action Provider:

By adding a deploy stage to the pipeline and configuring Amazon ECS as the action provider, the pipeline can automatically deploy the new container image to the ECS cluster.

This ensures that the service is updated with the new image tag, making the new version of the service endpoint reachable.

Reference: AWS CodePipeline with ECS

Update the Image Build Pipeline Stage to Output an imagedefinitions.json File that Reference the New Image Tag:

The imagedefinitions.json file provides the necessary information about the container images and their tags for the ECS task definitions.

Updating the pipeline to output this file ensures that the correct image version is deployed. Example imagedefinitions.json

[

{

www.atmicnetworks.com

```
  "name": "container-name",
  "imageUri": "123456789012.dkr.ecr.region.amazonaws.com/my-repo:my-tag"
}
]
```

Reference: CodePipeline ECS Deployment

Create an AWS Lambda Function that Runs Connectivity Checks and API Calls against the Service. Integrate the Lambda Function with CodePipeline by Using a Lambda Action Stage:

The Lambda function can perform the necessary integration tests by making connectivity checks and API calls to the deployed service endpoint.

Integrating this Lambda function into CodePipeline ensures that these tests are run automatically after deployment, providing near-real-time feedback on the new deployment's health.

Example Lambda function integration:

```
actions:
- name: TestService
  actionTypeId:
  category: Test
    owner: AWS
    provider: Lambda
    version: 1
  runOrder: 2
  configuration:
    FunctionName: testServiceFunction
```

Reference: Integrating Lambda with CodePipeline

These steps ensure that the CI/CD workflow deploys the new container image to ECS, updates the image references, and performs integration tests, meeting the requirements with minimal management overhead.

Question: 224

A company wants to use AWS Systems Manager documents to bootstrap physical laptops for developers The bootstrap code Is stored in GitHub A DevOps engineer has already created a Systems Manager activation, installed the Systems Manager agent with the registration code, and installed an activation ID on all the laptops.

Which set of steps should be taken next?

A. Configure the Systems Manager document to use the AWS-RunShellScnpt command to copy the files from GitHub to Amazon S3, then use the aws-downloadContent plugin with a sourceType of S3

B. Configure the Systems Manager document to use the aws-configurePackage plugin with an install action and point to the Git repository

C. Configure the Systems Manager document to use the aws-downloadContent plugin with a sourceType of GitHub and sourcelnfo with the repository details.

D. Configure the Systems Manager document to use the aws:softwarelnventory plugin and run the script from the Git repository

Answer: C

Explanation:

Configure the Systems Manager Document to Use the aws-downloadContent Plugin with a sourceType of GitHub and sourceInfo with the Repository Details:

The aws-downloadContent plugin can download content from various sources, including GitHub, which is necessary for bootstrapping the laptops with the code stored in the GitHub repository. schemaVersion: '2.2' description: "Download and run bootstrap script from GitHub" mainSteps:

```
- action: aws:downloadContent
  name: downloadBootstrapScript
  inputs:
    sourceType: GitHub
    sourceInfo: '{"owner":"my-org","repository":"my- repo","path":"scripts/bootstrap.sh","getOptions":"branch:main"}'
    destinationPath: /tmp/bootstrap.sh
- action: aws:runShellScript
  name: runBootstrapScript
  inputs:
    runCommand:
      - chmod +x /tmp/bootstrap.sh
      - /tmp/bootstrap.sh
```

This setup ensures that the bootstrap code is downloaded from GitHub and executed on the laptops using Systems Manager.

Reference:

AWS Systems Manager aws-downloadContent Plugin

Running Commands Using Systems Manager

Question: 225

A company hired a penetration tester to simulate an internal security breach The tester performed port scans on the company's Amazon EC2 instances. The company's security measures did not detect the port scans.

The company needs a solution that automatically provides notification when port scans are performed on EC2 instances. The company creates and subscribes to an Amazon Simple Notification Service (Amazon SNS) topic.

What should the company do next to meet the requirement?

A. Ensure that Amazon GuardDuty is enabled Create an Amazon CloudWatch alarm for detected EC2 and port scan findings. Connect the alarm to the SNS topic.

B. Ensure that Amazon Inspector is enabled Create an Amazon EventBridge event for detected network reachability findings that indicate port scans Connect the event to the SNS topic.

C. Ensure that Amazon Inspector is enabled. Create an Amazon EventBridge event for detected CVEs that cause open port vulnerabilities. Connect the event to the SNS topic

D. Ensure that AWS CloudTrail is enabled Create an AWS Lambda function to analyze the CloudTrail logs for unusual amounts of traffic from an IP address range Connect the Lambda function to the SNS topic.

Answer: A

Explanation:

Ensure that Amazon GuardDuty is Enabled:

Amazon GuardDuty is a threat detection service that continuously monitors for malicious activity and unauthorized behavior. It can detect port scans and generate findings for these events.

Create an Amazon CloudWatch Alarm for Detected EC2 and Port Scan Findings:

Configure GuardDuty to monitor for port scans and other threats.

Create a CloudWatch alarm that triggers when GuardDuty detects port scan activities.

Connect the Alarm to the SNS Topic:

The CloudWatch alarm should be configured to send notifications to the SNS topic subscribed by the security team.

This setup ensures that the security team receives near-real-time notifications when a port scan is detected on the EC2 instances.

Example configuration steps:

Enable GuardDuty and ensure it is monitoring the relevant AWS accounts.

Create a CloudWatch alarm:

```
{
"AlarmName": "GuardDutyPortScanAlarm", "MetricName": "ThreatIntelIndicator",
"Namespace": "AWS/GuardDuty",
"Statistic": "Sum",
"Dimensions": [
  {
    "Name": "FindingType",
    "Value": "Recon:EC2/Portscan"
  }
],
"Period": 300,
"EvaluationPeriods": 1,
"Threshold": 1,
"ComparisonOperator": "GreaterThanOrEqualToThreshold", "AlarmActions": ["arn:aws:sns:region:account-id:SecurityAlerts"]
}
```

Reference:

Amazon GuardDuty

Creating CloudWatch Alarms for GuardDuty Findings

Question: 226

A company's DevOps team manages a set of AWS accounts that are in an organization in AWS Organizations

The company needs a solution that ensures that all Amazon EC2 instances use approved AMIs that the DevOps team manages. The solution also must remediate the usage of AMIs that are not approved The individual account administrators must not be able to remove the restriction to use approved AMIs.

Which solution will meet these requirements?

A. Use AWS CloudFormation StackSets to deploy an Amazon EventBridge rule to each account. Configure the rule to react to AWS CloudTrail events for Amazon EC2 and to send a notification to an Amazon Simple Notification Service (Amazon SNS) topic. Subscribe the DevOps team to the SNS topic

B. Use AWS CloudFormation StackSets to deploy the approved-amis-by-id AWS Config managed rule to each account. Configure the rule with the list of approved AMIs. Configure the rule to run the the AWS-StopEC2Instance AWS Systems Manager Automation runbook for the noncompliant EC2 instances.

C. Create an AWS Lambda function that processes AWS CloudTrail events for Amazon EC2 Configure the Lambda function to send a notification to an Amazon Simple Notification Service (Amazon SNS) topic. Subscribe the DevOps team to the SNS topic. Deploy the Lambda function in each account in the organization Create an Amazon EventBridge rule in each account Configure the EventBridge rules to react to AWS CloudTrail events for Amazon EC2 and to invoke the Lambda function.

D. Enable AWS Config across the organization Create a conformance pack that uses the approved - amis-by-id AWS Config managed rule with the list of approved AMIs. Deploy the conformance pack across the organization. Configure the rule to run the AWS-StopEC2Instance AWS Systems Manager

Automation runbook for the noncompliant EC2 instances.

Answer: D

Explanation:

Enable AWS Config Across the Organization:

AWS Config provides a detailed view of the configuration of AWS resources in your AWS account. It can be used to assess, audit, and evaluate the configurations of your resources.

Enabling AWS Config across the organization ensures that all accounts are monitored for compliance. Create a Conformance Pack Using the approved-amis-by-id AWS Config Managed Rule:

A conformance pack is a collection of AWS Config rules and remediation actions that can be easily deployed across an organization.

The approved-amis-by-id managed rule checks whether running instances are using approved AMIs. Deploy the Conformance Pack Across the Organization:

Deploying the conformance pack across the organization ensures that all accounts adhere to the policy of using only approved AMIs.

The conformance pack can be deployed via the AWS Management Console, CLI, or SDKs.

Configure the Rule to Run the AWS-StopEC2Instance AWS Systems Manager Automation Runbook for Non-Compliant EC2 Instances:

The AWS-StopEC2Instance runbook can be configured to automatically stop any EC2 instances that are found to be non-compliant (i.e., not using approved AMIs).

This remediation action ensures that any unauthorized instances are promptly stopped, enforcing the policy without manual intervention.

By following these steps, the solution ensures that all EC2 instances across the organization use approved AMIs, and any non-compliant instances are remediated automatically.

Reference:

AWS Config Conformance Packs

AWS Config Managed Rules

AWS Systems Manager Automation Runbooks

Question: 227

A company uses containers for its applications The company learns that some container Images are missing required security configurations

A DevOps engineer needs to implement a solution to create a standard base image The solution must publish the base image weekly to the us-west-2 Region, us-east-2 Region, and eu-central-1 Region.

Which solution will meet these requirements?

A. Create an EC2 Image Builder pipeline that uses a container recipe to build the image. Configure the pipeline to distribute the image to an Amazon Elastic Container Registry (Amazon ECR) repository in us-west-2. Configure ECR replication from us-west-2 to us-east-2 and from us-east-2 to eu-central- 1 Configure the pipeline to run weekly

B. Create an AWS CodePipeline pipeline that uses an AWS CodeBuild project to build the image Use AWS CodeOeploy to publish the image to an Amazon Elastic Container Registry (Amazon ECR) repository in us-west-2 Configure ECR replication from us-west-2 to us-east-2 and from us-east-2 to eu-central-1 Configure the pipeline to run weekly

C. Create an EC2 Image Builder pipeline that uses a container recipe to build the Image Configure the pipeline to distribute the image to Amazon Elastic Container Registry (Amazon ECR) repositories in all three Regions. Configure the pipeline to run weekly.

D. Create an AWS CodePipeline pipeline that uses an AWS CodeBuild project to build the image Use AWS CodeDeploy to publish the image to Amazon Elastic Container Registry (Amazon ECR) repositories in all three Regions. Configure the pipeline to run weekly.

Answer: C

Explanation:

Create an EC2 Image Builder Pipeline that Uses a Container Recipe to Build the Image:

EC2 Image Builder simplifies the creation, maintenance, validation, and sharing of container images. By using a container recipe, you can define the base image, components, and validation tests for your container image.

Configure the Pipeline to Distribute the Image to Amazon Elastic Container Registry (Amazon ECR) Repositories in All Three Regions:

Amazon ECR provides a secure, scalable, and reliable container registry.

Configuring the pipeline to distribute the image to ECR repositories in us-west-2, us-east-2, and eu- central-1 ensures that the image is available in all required regions.

Configure the Pipeline to Run Weekly:

Setting the pipeline to run on a weekly schedule ensures that the base image is regularly updated and published, incorporating any new security configurations or updates.

By using EC2 Image Builder to automate the creation and distribution of the container image, the solution ensures that the base image is consistently maintained and available across multiple regions with minimal management overhead.

Reference:

EC2 Image Builder

Amazon ECR

Setting Up EC2 Image Builder Pipelines

Question: 228

A software team is using AWS CodePipeline to automate its Java application release pipeline The pipeline consists of a source stage, then a build stage, and then a deploy stage. Each stage contains a single action that has a runOrder value of 1.

The team wants to integrate unit tests into the existing release pipeline. The team needs a solution that deploys only the code changes that pass all unit tests.

Which solution will meet these requirements?

A. Modify the build stage. Add a test action that has a runOrder value of 1. Use AWS CodeDeploy as the action provider to run unit tests.

B. Modify the build stage Add a test action that has a runOrder value of 2 Use AWS CodeBuild as the action provider to run unit tests

C. Modify the deploy stage Add a test action that has a runOrder value of 1 Use AWS CodeDeploy as the action provider to run unit tests

D. Modify the deploy stage Add a test action that has a runOrder value of 2 Use AWS CodeBuild as the action provider to run unit tests

Answer: B

Explanation:

Modify the Build Stage to Add a Test Action with a RunOrder Value of 2:

The build stage in AWS CodePipeline can have multiple actions. By adding a test action with a runOrder value of 2, the test action will execute after the initial build action completes.

Use AWS CodeBuild as the Action Provider to Run Unit Tests:

AWS CodeBuild is a fully managed build service that compiles source code, runs tests, and produces software packages.

Using CodeBuild to run unit tests ensures that the tests are executed in a controlled environment and that only the code changes that pass the unit tests proceed to the deploy stage.

Example configuration in CodePipeline:

```
{
"name": "BuildStage",
"actions": [
  {
    "name": "Build",
    "actionTypeId": {
    "category": "Build",
      "owner": "AWS",
       "provider": "CodeBuild",
      "version": "1"
    },
    "runOrder": 1
  },
  {
    "name": "Test",
    "actionTypeId": {
    "category": "Test",
      "owner": "AWS",
       "provider": "CodeBuild",
      "version": "1"
    },
    "runOrder": 2
  }
]
}
```

By integrating the unit tests into the build stage and ensuring they run after the build process, the pipeline guarantees that only code changes passing all unit tests are deployed.

Reference:

AWS CodePipeline

AWS CodeBuild

Using CodeBuild with CodePipeline

Question: 229

A company runs applications on Windows and Linux Amazon EC2 instances The instances run across multiple Availability Zones In an AWS Region. The company uses Auto Scaling groups for each application.

The company needs a durable storage solution for the instances. The solution must use SMB for Windows and must use NFS for Linux. The solution must also have sub-millisecond latencies. All instances will read and write the data.

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

A. Create an Amazon Elastic File System (Amazon EFS) file system that has targets in multiple Availability Zones
B. Create an Amazon FSx for NetApp ONTAP Multi-AZ file system.
C. Create a General Purpose SSD (gp3) Amazon Elastic Block Store (Amazon EBS) volume to use for shared storage.
D. Update the user data for each application's launch template to mount the file system
E. Perform an instance refresh on each Auto Scaling group.
F. Update the EC2 instances for each application to mount the file system when new instances are launched

Answer: ABD

Explanation:

Create an Amazon Elastic File System (Amazon EFS) File System with Targets in Multiple Availability Zones:
Amazon EFS provides a scalable and highly available network file system that supports the NFS protocol. EFS is ideal for Linux instances as it allows multiple instances to read and write data concurrently.

Setting up EFS with targets in multiple Availability Zones ensures high availability and durability. Reference: Amazon EFS Overview
Create an Amazon FSx for NetApp ONTAP Multi-AZ File System:

Amazon FSx for NetApp ONTAP offers a fully managed file storage solution that supports both SMB for Windows and NFS for Linux.
The Multi-AZ deployment ensures high availability and durability, providing sub-millisecond latencies suitable for the application's performance requirements.
Reference: Amazon FSx for NetApp ONTAP
Update the User Data for Each Application's Launch Template to Mount the File System:
Updating the user data in the launch template ensures that every new instance launched by the Auto Scaling group will automatically mount the appropriate file system.

This step is necessary to ensure that all instances can access the shared storage without manual intervention.
Example user data for mounting EFS (Linux)

```
#!/bin/bash
sudo yum install -y amazon-efs-utils
sudo mount -t efs fs-12345678:/ /mnt/efs
```

Example user data for mounting FSx (Windows):

By implementing these steps, the company can provide a durable storage solution with submillisecond latencies that supports both SMB and NFS protocols, meeting the requirements for both Windows and Linux instances.

Reference:
Mounting EFS File Systems
Mounting Amazon FSx File Systems

Question: 230

A company uses Amazon EC2 as its primary compute platform. A DevOps team wants to audit the company's EC2 instances to check whether any prohibited applications have been installed on the EC2 instances.
Which solution will meet these requirements with the MOST operational efficiency?

A. Configure AWS Systems Manager on each instance Use AWS Systems Manager Inventory Use Systems Manager resource data sync to synchronize and store findings in an Amazon S3 bucket Create an AWS Lambda function that runs when new objects are added to the S3 bucket. Configure the Lambda function to identify prohibited applications.
B. Configure AWS Systems Manager on each instance Use Systems Manager Inventory Create AWS Config rules that monitor changes from Systems Manager Inventory to identify prohibited applications.
C. Configure AWS Systems Manager on each instance. Use Systems Manager Inventory. Filter a trail in AWS CloudTrail for Systems Manager Inventory events to identify prohibited applications.
D. Designate Amazon CloudWatch Logs as the log destination for all application instances Run an automated script across all instances to create an inventory of installed applications Configure the script to forward the results to CloudWatch Logs Create a CloudWatch alarm that uses filter patterns to search log data to identify prohibited applications.

www.atmicnetworks.com

Answer: A

Explanation:

Configure AWS Systems Manager on Each Instance:
AWS Systems Manager provides a unified interface for managing AWS resources. Install the Systems Manager agent on each EC2 instance to enable inventory management and other features.
Use AWS Systems Manager Inventory:
Systems Manager Inventory collects metadata about your instances and the software installed on them. This data includes information about applications, network configurations, and more.
Enable Systems Manager Inventory on all EC2 instances to gather detailed information about installed applications.
Use Systems Manager Resource Data Sync to Synchronize and Store Findings in an Amazon S3 Bucket:
Resource Data Sync aggregates inventory data from multiple accounts and regions into a single S3 bucket, making it easier to query and analyze the data.
Configure Resource Data Sync to automatically transfer inventory data to an S3 bucket for centralized storage.
Create an AWS Lambda Function that Runs When New Objects are Added to the S3 Bucket:
Use an S3 event to trigger a Lambda function whenever new inventory data is added to the S3 bucket.
The Lambda function can parse the inventory data and check for the presence of prohibited applications.
Configure the Lambda Function to Identify Prohibited Applications:
The Lambda function should be programmed to scan the inventory data for any known prohibited applications and generate alerts or take appropriate actions if such applications are found. Example Lambda function in Python import json import boto3

```
def lambda_handler(event, context):
    s3 = boto3.client('s3')
    bucket = event['Records'][0]['s3']['bucket']['name']
    key = event['Records'][0]['s3']['object']['key']

    response = s3.get_object(Bucket=bucket, Key=key)
    inventory_data = json.loads(response['Body'].read().decode('utf-8'))

    prohibited_apps = ['app1', 'app2']
    for instance in inventory_data['Instances']:
        for app in instance['Applications']:
            if app['Name'] in prohibited_apps:
                # Send notification or take action print(f"Prohibited application found: {app['Name']} on instance {instance['InstanceId']}")

    return {'statusCode': 200, 'body': json.dumps('Check completed')}
```

By leveraging AWS Systems Manager Inventory, Resource Data Sync, and Lambda, this solution provides an efficient and automated way to audit EC2 instances for prohibited applications. Reference:
AWS Systems Manager Inventory
AWS Systems Manager Resource Data Sync
S3 Event Notifications
AWS Lambda

www.atmicnetworks.com

Question: 231

A company is refactoring applications to use AWS. The company identifies an internal web application that needs to make Amazon S3 API calls in a specific AWS account.

The company wants to use its existing identity provider (IdP) auth.company.com for authentication. The IdP supports only OpenID Connect (OIDC). A DevOps engineer needs to secure the web application's access to the AWS account.

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

A. Configure AWS 1AM Identity Center. Configure an IdP. Upload the IdP metadata from the existing IdP.

B. Create an 1AM IdP by using the provider URL, audience, and signature from the existing IdP.

C. Create an 1AM role that has a policy that allows the necessary S3 actions. Configure the role's trust policy to allow the OIDC IdP to assume the role if the sts.amazon.conraud context key is appid from idp.

D. Create an 1AM role that has a policy that allows the necessary S3 actions. Configure the role's trust policy to allow the OIDC IdP to assume the role if the auth.company.com:aud context key is appid_from_idp.

E. Configure the web application lo use the AssumeRoleWith Web Identity API operation to retrieve temporary credentials. Use the temporary credentials to make the S3 API calls.

F. Configure the web application to use the GetFederationToken API operation to retrieve temporary credentials Use the temporary credentials to make the S3 API calls.

Answer: B, D, E

Step 1: Creating an Identity Provider in IAM

You first need to configure AWS to trust the external identity provider (IdP), which in this case supports OpenID Connect (OIDC). The IdP will handle the authentication, and AWS will handle the authorization based on the IdP's token.

Action: Create an IAM Identity Provider (IdP) in AWS using the existing provider's URL, audience, and signature. This step is essential for establishing trust between AWS and the external IdP.

Why: This allows AWS to accept tokens from your external IdP (auth.company.com) for authentication.

Reference: AWS documentation on IAM Identity Providers.

So, this corresponds to Option B: Create an IAM IdP by using the provider URL, audience, and signature from the existing IdP.

Step 2: Creating an IAM Role with Specific Permissions

Next, you need to create an IAM role with a trust policy that allows the external IdP to assume it when certain conditions are met. Specifically, the trust policy needs to allow the role to be assumed based on the context key auth.company.com:aud (audience claim in the token).

Action: Create an IAM role that has the necessary permissions (e.g., Amazon S3 access). The role's trust policy should specify the OIDC IdP as the trusted entity and validate the audience claim (auth.company.com:aud), which comes from the token provided by the IdP.

Why: This step ensures that only the specified web application authenticated via OIDC can assume the IAM role to make API calls.

Reference: AWS documentation on OIDC and Role Assumption.

This corresponds to Option D: Create an IAM role that has a policy that allows the necessary S3 actions. Configure the role's trust policy to allow the OIDC IdP to assume the role if the auth.company.com:aud context key is appid_from_idp.

Step 3: Using Temporary Credentials via AssumeRoleWithWebIdentity API

To securely make Amazon S3 API calls, the web application will need temporary credentials. The web application can use the AssumeRoleWithWebIdentity API call to assume the IAM role configured in the previous step and obtain temporary AWS credentials. These credentials can then be used to interact with Amazon S3.

Action: The web application must be configured to call the AssumeRoleWithWebIdentity API operation, passing the OIDC token from the IdP to obtain temporary credentials.

Why: This allows the web application to authenticate via the external IdP and then authorize access to AWS resources securely using short-lived credentials.

Reference: AWS documentation on AssumeRoleWithWebIdentity.

This corresponds to Option E: Configure the web application to use the AssumeRoleWithWebIdentity API operation to retrieve temporary credentials. Use the temporary credentials to make the S3 API calls.

Summary of Selected Answers:

B: Create an IAM IdP by using the provider URL, audience, and signature from the existing IdP.

D: Create an IAM role that has a policy that allows the necessary S3 actions. Configure the role's trust policy to allow the OIDC IdP to assume the role if the auth.company.com:aud context key is appid_from_idp.

E: Configure the web application to use the AssumeRoleWithWebIdentity API operation to retrieve temporary credentials. Use the temporary credentials to make the S3 API calls.

This setup enables the web application to use OpenID Connect (OIDC) for authentication and securely interact with Amazon S3 in a specific AWS account using short-lived credentials obtained through AWS Security Token Service (STS).

Question: 232

A company uses an organization in AWS Organizations to manage several AWS accounts that the company's developers use. The company requires all data to be encrypted in transit.

Multiple Amazon S3 buckets that were created in developer accounts allow unencrypted connections. A DevOps engineer must enforce encryption of data in transit for all existing S3 buckets that are created in accounts in the organization.

Which solution will meet these requirements?

A. Use AWS Cloud Formation StackSets to deploy an AWS Network Firewall firewall to each account. Route all outbound requests from the AWS environment through the firewall. Deploy a policy to block access to all outbound requests on port 80.

B. Use AWS CloudFormation StackSets to deploy an AWS Network Firewall firewall to each account. Route all inbound requests to the AWS environment through the firewall. Deploy a policy to block access to all inbound requests on port 80.

C. Turn on AWS Config for the organization. Deploy a conformance pack that uses the s3-bucket-ssi- requests-only managed rule and an AWS Systems Manager Automation runbook. Use a runbook that adds a bucket policy statement to deny access to an S3 bucket when the value of the aws:SecureTransport condition key is false.

D. Turn on AWS Config for the organization. Deploy a conformance pack that uses the s3-buckot-ssl- requests-only managed rule and an AWS Systems Manager Automation runbook. Use a runbook that adds a bucket policy statement to deny access to an S3 bucket when the value of the s3:x-amz- server-side-encryption-aws-kms-key-id condition key is null.

Answer: C

Step 1: Enabling AWS Config for the Organization

The first step is to enable AWS Config across the AWS Organization. AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources. By enabling AWS Config, you can ensure that all S3 buckets within the organization are tracked and evaluated according to compliance rules.

Action: Turn on AWS Config for all AWS accounts in the organization.

Why: AWS Config will help monitor all resources (like S3 buckets) in real time to detect whether they are compliant with security policies.

Reference: AWS documentation on Configuring AWS Config.

Step 2: Deploying a Conformance Pack with Managed Rules

After AWS Config is enabled, you need to deploy a conformance pack that contains the s3-bucket-ssi- requests-only managed rule. This rule enforces that all S3 buckets only allow requests using Secure Socket Layer (SSL) connections (HTTPS).

Action: Deploy a conformance pack that uses the s3-bucket-ssi-requests-only rule. This rule ensures that only SSL connections (for encrypted data in transit) are allowed when accessing S3.

Why: This rule guarantees that data is encrypted in transit by enforcing SSL connections to the S3 buckets.

Reference: AWS documentation on Conformance Packs.

Step 3: Using an AWS Systems Manager Automation Runbook

To automatically remediate the compliance issues, such as S3 buckets allowing non-SSL requests, a Systems Manager Automation runbook is deployed. The runbook will automatically add a bucket policy that denies access to any requests that do not use SSL.

Action: Use a Systems Manager Automation runbook that adds a bucket policy statement to deny access when the aws:SecureTransport condition key is false.

Why: This ensures that all S3 buckets across the organization comply with the policy of enforcing encrypted data in transit.

Reference: AWS documentation on AWS Config Managed Rules.

This corresponds to Option C: Turn on AWS Config for the organization. Deploy a conformance pack that uses the s3-bucket-ssi-requests-only managed rule and an AWS Systems Manager Automation runbook. Use a runbook that adds a bucket policy statement to deny access to an S3 bucket when the value of the aws:SecureTransport condition key is false.

Question: 233

A company is using AWS CodeDeploy to automate software deployment. The deployment must meet these requirements:

- A number of instances must be available to serve traffic during the deployment Traffic must be balanced across those instances, and the instances must automatically heal in the event of failure.
- A new fleet of instances must be launched for deploying a new revision automatically, with no manual provisioning.
- Traffic must be rerouted to the new environment to half of the new instances at a time. The deployment should succeed if traffic is rerouted to at least half of the instances; otherwise, it should fail.
- Before routing traffic to the new fleet of instances, the temporary files generated during the deployment process must be deleted.
- At the end of a successful deployment, the original instances in the deployment group must be deleted immediately to reduce costs.

How can a DevOps engineer meet these requirements?

A. Use an Application Load Balancer and an in-place deployment. Associate the Auto Scaling group with the deployment group. Use the Automatically copy Auto Scaling group option. and use CodeDeployDefault.OneAtAtime as the deployment configuration. Instruct AWS CodeDeploy to terminate the original instances in the deployment group, and use the AllowTraffic hook within appspec.yml to delete the temporary files.

B. Use an Application Load Balancer and a blue/green deployment. Associate the Auto Scaling group and Application Load Balancer target group with the deployment group. Use the Automatically copy Auto Scaling group option, create a custom deployment configuration with minimum healthy hosts defined as 50%. and assign the configuration to the deployment group. Instruct AWS CodeDeploy to terminate the original instances in the deployment group, and use the BeforeBlockTraffic hook within appspec.yml to delete the temporary files.

C. Use an Application Load Balancer and a blue/green deployment. Associate the Auto Scaling group and the Application Load Balancer target group with the deployment group. Use the Automatically copy Auto scaling group option, and use CodeDeployDefault.HalfAtAtime as the deployment configuration. Instruct AWSCodeDeploy to terminate the original instances in the deployment group, and use the BeforeAllowTraffic hook within appspec.yml to delete the temporary tiles.

D. Use an Application Load Balancer and an in-place deployment. Associate the Auto Scaling group and Application Load Balancer target group with the deployment group. Use the Automatically copy Auto Scaling group option, and use CodeDeployDefaulLAIlatOnce as a deployment configuration. Instruct AWS CodeDeploy to terminate the original instances in the deployment group, and use the BlockTraffic hook within appspec.yml to delete the temporary files.

Answer: C

Step 1: Use a Blue/Green Deployment Strategy

A blue/green deployment strategy is necessary to meet the requirement of launching a new fleet of instances for each deployment and

ensuring availability. In a blue/green deployment, the new version (green environment) is deployed to a separate set of instances, while the old version (blue environment) remains active. After testing the new version, traffic can be gradually shifted. Action: Use AWS CodeDeploy's blue/green deployment configuration.

Why: Blue/green deployment minimizes downtime and ensures that traffic is shifted only to healthy instances.

Reference: AWS documentation on Blue/Green Deployment.

Step 2: Use an Application Load Balancer and Auto Scaling Group

The Application Load Balancer (ALB) is essential to balance traffic across multiple instances, and Auto Scaling ensures the deployment scales automatically to meet demand.

Action: Associate the Auto Scaling group and Application Load Balancer target group with the deployment group.

Why: This configuration ensures that traffic is evenly distributed and that instances automatically scale based on traffic load.

Reference: AWS documentation on Deploying with Auto Scaling and Load Balancing.

Step 3: Use Custom Deployment Configuration

The company requires that traffic be rerouted to at least half of the instances to succeed. AWS CodeDeploy allows you to configure custom deployment settings with specific thresholds for healthy hosts.

Action: Create a custom deployment configuration where 50% of the instances must be healthy. Why: This ensures that the deployment continues only if at least 50% of the new instances are healthy.

Reference: AWS documentation on Deployment Configurations.

Step 4: Clean Temporary Files Using Hooks

Before routing traffic to the new environment, the temporary files generated during the deployment must be deleted. This can be achieved using the BeforeAllowTraffic hook in the appspec.yml file.

Action: Use the BeforeAllowTraffic lifecycle event hook to clean up temporary files before routing traffic to the new environment.

Why: This ensures that the environment is clean before the new instances start serving traffic. Reference: AWS documentation on AppSpec File Hooks.

Step 5: Terminate Original Instances After Deployment

After a successful deployment, AWS CodeDeploy can automatically terminate the original instances (blue environment) to save costs.

Action: Instruct AWS CodeDeploy to terminate the original instances after the new instances are healthy.

Why: This helps in cost reduction by removing unused instances after the deployment. Reference: AWS documentation on Terminate Old Instances.

This corresponds to Option C: Use an Application Load Balancer and a blue/green deployment. Associate the Auto Scaling group and the Application Load Balancer target group with the deployment group. Use the Automatically copy Auto Scaling group option, and use CodeDeployDefault.HalfAtATime as the deployment configuration. Instruct AWS CodeDeploy to terminate the original instances in the deployment group, and use the BeforeAllowTraffic hook within appspec.yml to delete the temporary files.

Question: 234

A company has deployed a new platform that runs on Amazon Elastic Kubernetes Service (Amazon EKS). The new platform hosts web applications that users frequently update. The application developers build the Docker images for the applications and deploy the Docker images manually to the platform.

The platform usage has increased to more than 500 users every day. Frequent updates, building the updated Docker images for the applications, and deploying the Docker images on the platform manually have all become difficult to manage.

The company needs to receive an Amazon Simple Notification Service (Amazon SNS) notification if Docker image scanning returns any HIGH or CRITICAL findings for operating system or programming language package vulnerabilities.

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

A. Create an AWS CodeCommit repository to store the Dockerfile and Kubernetes deployment files. Create a pipeline in AWS CodePipeline. Use an Amazon S3 event to invoke the pipeline when a newer version of the Dockerfile is committed. Add a stop to the

pipeline to initiate the AWS CodeBuild project.

B. Create an AWS CodeCommit repository to store the Dockerfile and Kubernetes deployment files. Create a pipeline in AWS CodePipeline. Use an Amazon EvenlBridge event to invoke the pipeline when a newer version of the Dockerfile is committed. Add a step to the pipeline to initiate the AWS CodeBuild project.

C. Create an AWS CodeBuild project that builds the Docker images and stores the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository. Turn on basic scanning for the ECR repository. Create an Amazon EventBridge rule that monitors Amazon GuardDuty events. Configure the EventBridge rule to send an event to an SNS topic when the finding-severity-counts parameter is more than 0 at a CRITICAL or HIGH level.

D. Create an AWS CodeBuild project that builds the Docker images and stores the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository. Turn on enhanced scanning for the ECR repository. Create an Amazon EventBridge rule that monitors ECR image scan events. Configure the EventBridge rule to send an event to an SNS topic when the finding-severity-counts parameter is more than 0 at a CRITICAL or HIGH level.

E. Create an AWS CodeBuild project that scans the Dockerfile. Configure the project to build the Docker images and store the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository if the scan is successful. Configure an SNS topic to provide notification if the scan returns any vulnerabilities.

Answer: B, D

Step 1: Automate Docker Image Deployment using AWS CodePipeline

The first challenge is the manual process of building and deploying Docker images. To address this, you can use AWS CodePipeline to automate the process. AWS CodePipeline integrates with CodeCommit (for source code and Dockerfile storage) and CodeBuild (to build Docker images and store them in Amazon Elastic Container Registry (ECR)).

Action: Create an AWS CodeCommit repository to store the Dockerfile and Kubernetes deployment files. Then, create a pipeline in AWS CodePipeline that triggers on new commits via an Amazon EventBridge event.

Why: This automation significantly reduces the manual effort of building and deploying Docker images when updates are made to the codebase.

Reference: AWS documentation on AWS CodePipeline and CodeCommit Integration.

This corresponds to Option B: Create an AWS CodeCommit repository to store the Dockerfile and Kubernetes deployment files. Create a pipeline in AWS CodePipeline. Use an Amazon EventBridge event to invoke the pipeline when a newer version of the Dockerfile is committed. Add a step to the pipeline to initiate the AWS CodeBuild project.

Step 2: Enabling Enhanced Scanning on Amazon ECR and Monitoring Vulnerabilities

To scan for vulnerabilities in Docker images, Amazon ECR provides both basic and enhanced scanning options. Enhanced scanning offers deeper and more frequent scans, and integrates with Amazon EventBridge to send notifications based on findings.

Action: Turn on enhanced scanning for the Amazon ECR repository where the Docker images are stored. Use Amazon EventBridge to monitor image scan events and trigger an Amazon SNS notification if any HIGH or CRITICAL vulnerabilities are found.

Why: Enhanced scanning provides a detailed analysis of operating system and programming language package vulnerabilities, which can trigger notifications in real-time.

Reference: AWS documentation on Enhanced Scanning for Amazon ECR.

This corresponds to Option D: Create an AWS CodeBuild project that builds the Docker images and stores the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository. Turn on enhanced scanning for the ECR repository. Create an Amazon EventBridge rule that monitors ECR image scan events. Configure the EventBridge rule to send an event to an SNS topic when the finding-severity-counts parameter is more than 0 at a CRITICAL or HIGH level.

Question: 235

A company uses an organization in AWS Organizations to manage multiple AWS accounts The company needs an automated process across all AWS accounts to isolate any compromised Amazon EC2 instances when the instances receive a specific tag.

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

A. Use AWS Cloud Formation StackSets to deploy the Cloud Formation stacks in all AWS accounts.

B. Create an SCP that has a Deny statement for the ec2:" action with a condition of "aws:RequestTag/isolation": false.

C. Attach the SCP to the root of the organization.

D. Create an AWS Cloud Formation template that creates an EC2 instance rote that has no 1AM policies attached. Configure the template to have a security group that has an explicit Deny rule on all traffic. Use the Cloud Formation template to create an AWS Lambda function that attaches the 1AM role to instances. Configure the Lambda function to add a network ACL. Sot up an Amazon EventBridge rule to invoke the Lambda function when a specific tag is applied to a compromised EC2 instance.

E. Create an AWS Cloud Formation template that creates an EC2 instance role that has no 1AM policies attached. Configure the template to have a security group that has no inbound rules or outbound rules. Use the CloudFormation template to create an AWS Lambda function that attaches the 1AM role to instances. Configure the Lambda function to replace any existing security groups with the new security group. Set up an Amazon EventBridge rule to invoke the Lambda function when a specific tag is applied to a compromised EC2 instance.

Answer: AE

Step 1: Deploy the Automation Solution using CloudFormation StackSets

To automate the process across multiple AWS accounts within an organization, you can use AWS CloudFormation StackSets. StackSets allow you to deploy CloudFormation templates to multiple accounts within an organization, ensuring consistent infrastructure and automation.

Action: Use AWS CloudFormation StackSets to deploy the necessary resources across all AWS accounts. This includes deploying the Lambda function and security groups that will isolate compromised EC2 instances.

Why: StackSets make it easy to deploy and manage resources across multiple AWS accounts, reducing the operational overhead.

Reference: AWS documentation on CloudFormation StackSets.

This corresponds to Option A: Use AWS CloudFormation StackSets to deploy the CloudFormation stacks in all AWS accounts.

Step 2: Isolate EC2 Instances using Lambda and Security Groups

When an EC2 instance is compromised, it needs to be isolated from the network. This can be done by creating a security group with no inbound or outbound rules and attaching it to the instance. A Lambda function can handle this process and can be triggered automatically by an Amazon EventBridge rule when a specific tag (e.g., "isolation") is applied to the compromised instance. Action: Create a Lambda function that attaches an isolated security group (with no inbound or outbound rules) to the compromised EC2 instances. Set up an EventBridge rule to trigger the Lambda function when the "isolation" tag is applied to the instance.

Why: This automates the isolation process, ensuring that any compromised instances are immediately cut off from the network, reducing the potential damage from the compromise. Reference: AWS documentation on Tag-based Event Handling.

This corresponds to Option E: Create an AWS CloudFormation template that creates an EC2 instance role that has no IAM policies attached. Configure the template to have a security group that has no inbound rules or outbound rules. Use the CloudFormation template to create an AWS Lambda function that attaches the IAM role to instances. Configure the Lambda function to replace any existing security groups with the new security group. Set up an Amazon EventBridge rule to invoke the Lambda function when a specific tag is applied to a compromised EC2 instance.

Question: 236

A DevOps learn has created a Custom Lambda rule in AWS Config. The rule monitors Amazon Elastic Container Repository (Amazon ECR) policy statements for ecr:' actions. When a noncompliant repository is detected, Amazon EventBridge uses Amazon Simple Notification Service (Amazon SNS) to route the notification to a security team.

When the custom AWS Config rule is evaluated, the AWS Lambda function fails to run.

Which solution will resolve the issue?

A. Modify the Lambda function's resource policy to grant AWS Config permission to invoke the function.

B. Modify the SNS topic policy to include configuration changes for EventBridge to publish to the SNS topic.

C. Modify the Lambda function's execution role to include configuration changes for custom AWS Config rules.

D. Modify all the ECR repository policies to grant AWS Config access to the necessary ECR API actions.

Answer: A

Step 1: Understanding Lambda Permissions and AWS Config
The custom AWS Config rule evaluates resources and invokes an AWS Lambda function when a compliance check is triggered. For AWS Config to invoke the Lambda function, it requires permission to do so.
Issue: The Lambda function fails to execute because AWS Config doesn't have permission to invoke it. Action: Modify the resource-based policy of the Lambda function to grant AWS Config permission to invoke the Lambda function.
Why: Without this permission, AWS Config cannot trigger the Lambda function, which is why the evaluation fails.
Reference: AWS documentation on Lambda permissions and resource-based policies.
This corresponds to Option A: Modify the Lambda function's resource policy to grant AWS Config permission to invoke the function.

Question: 237

A DevOps engineer is setting up an Amazon Elastic Container Service (Amazon ECS) blue/green deployment for an application by using AWS CodeDeploy and AWS CloudFormation. During the deployment window, the application must be highly available and CodeDeploy must shift 10% of traffic to a new version of the application every minute until all traffic is shifted.
Which configuration should the DevOps engineer add in the CloudFormation template to meet these requirements?

A. Add an AppSpec file with the CodeDeployDefault.ECSLineaMOPercentEverylMinutes deployment configuration.
B. Add the AWS::CodeDeployBlueGreen transform and the AWS::CodeDeploy::BlueGreen hook parameter with the CodeDeployDefault.ECSLinear10PercentEvery1 Minutes deployment configuration.
C. Add an AppSpec file with the ECSCanary10Percent5Minutes deployment configuration.
D. Add the AWS::CodeDeployBlueGroen transform and the AWS::CodeDeploy::BlueGreen hook parameter with the ECSCanary10Percent5Minutes deployment configuration.

Answer: B

Step 1: Using AWS CloudFormation with ECS Blue/Green Deployments
The requirement is to implement an ECS blue/green deployment where traffic is shifted gradually. AWS CodeDeploy supports such blue/green deployments with predefined configurations, like ECSLinear10PercentEvery1Minute, which shifts 10% of traffic every minute.
Action: Use the AWS::CodeDeployBlueGreen transform and the appropriate hooks in the CloudFormation template. The ECSLinear10PercentEvery1Minute deployment configuration meets the requirement of shifting 10% of traffic every minute.
Why: The transform and hook parameters in CloudFormation are essential for configuring the blue/green deployment with the desired traffic-shifting behavior.
Reference: AWS documentation on ECS Blue/Green deployments.
This corresponds to Option B: Add the AWS::CodeDeployBlueGreen transform and the AWS::CodeDeploy::BlueGreen hook parameter with the CodeDeployDefault.ECSLinear10PercentEvery1Minutes deployment configuration.

Question: 238

A company is migrating its container-based workloads to an AWS Organizations multi-account environment. The environment consists of application workload accounts that the company uses to deploy and run the containerized workloads. The company has also provisioned

a shared services account tor shared workloads in the organization.

The company must follow strict compliance regulations. All container images must receive security scanning before they are deployed to any environment. Images can be consumed by downstream deployment mechanisms after the images pass a scan with no critical vulnerabilities. Pre-scan and post-scan images must be isolated from one another so that a deployment can never use pre-scan images.

A DevOps engineer needs to create a strategy to centralize this process.

Which combination of steps will meet these requirements with the LEAST administrative overhead? (Select TWO.)

A. Create Amazon Elastic Container Registry (Amazon ECR) repositories in the shared services account: one repository for each pre-scan image and one repository for each post-scan image. Configure Amazon ECR image scanning to run on new image pushes to the pre-scan repositories. Use resource-based policies to grant the organization write access to the pre-scan repositories and read access to the post-scan repositories.

B. Create pre-scan Amazon Elastic Container Registry (Amazon ECR) repositories in each account that publishes container images. Create repositories for post-scan images in the shared services account. Configure Amazon ECR image scanning to run on new image pushes to the pre-scan repositories. Use resource-based policies to grant the organization read access to the post-scan repositories.

C. Configure image replication for each image from the image's pre-scan repository to the image's post-scan repository.

D. Create a pipeline in AWS CodePipeline for each pre-scan repository. Create a source stage that runs when new images are pushed to the pre-scan repositories. Create a stage that uses AWS CodeBuild as the action provider. Write a buildspec.yaml definition that determines the image scanning status and pushes images without critical vulnerabilities lo the post-scan repositories.

E. Create an AWS Lambda function. Create an Amazon EventBridge rule that reacts to image scanning completed events and invokes the Lambda function. Write function code that determines the image scanning status and pushes images without critical vulnerabilities to the post-scan repositories.

Answer: A, C

Step 1: Centralizing Image Scanning in a Shared Services Account

The first requirement is to centralize the image scanning process, ensuring pre-scan and post-scan images are stored separately. This can be achieved by creating separate pre-scan and post-scan repositories in the shared services account, with the appropriate resource-based policies to control access.

Action: Create separate ECR repositories for pre-scan and post-scan images in the shared services account. Configure resource-based policies to allow write access to pre-scan repositories and read access to post-scan repositories.

Why: This ensures that images are isolated before and after the scan, following the compliance requirements.

Reference: AWS documentation on Amazon ECR and resource-based policies.

This corresponds to Option A: Create Amazon Elastic Container Registry (Amazon ECR) repositories in the shared services account: one repository for each pre-scan image and one repository for each postscan image. Configure Amazon ECR image scanning to run on new image pushes to the pre-scan repositories. Use resource-based policies to grant the organization write access to the pre-scan repositories and read access to the post-scan repositories.

Step 2: Replication between Pre-Scan and Post-Scan Repositories

To automate the transfer of images from the pre-scan repositories to the post-scan repositories (after they pass the security scan), you can configure image replication between the two repositories. Action: Set up image replication between the pre-scan and post-scan repositories to move images that have passed the security scan.

Why: Replication ensures that only scanned and compliant images are available for deployment, streamlining the process with minimal administrative overhead.

Reference: AWS documentation on Amazon ECR image replication.

This corresponds to Option C: Configure image replication for each image from the image's pre-scan repository to the image's post-scan repository.

Question: 239

A company has configured Amazon RDS storage autoscaling for its RDS DB instances. A DevOps team needs to visualize the autoscaling events on an Amazon CloudWatch dashboard. Which solution will meet this requirement?

A. Create an Amazon EventBridge rule that reacts to RDS storage autoscaling events from RDS events. Create an AWS Lambda function that publishes a CloudWatch custom metric. Configure the EventBridge rule to invoke the Lambda function. Visualize the custom metric by using the CloudWatch dashboard.

B. Create a trail by using AWS CloudTrail with management events configured. Configure the trail to send the management events to Amazon CloudWatch Logs. Create a metric filter in CloudWatch Logs to match the RDS storage autoscaling events. Visualize the metric filter by using the CloudWatch dashboard.

C. Create an Amazon EventBridge rule that reacts to RDS storage autoscaling events (rom the RDS events. Create a CloudWatch alarm. Configure the EventBridge rule to change the status of the CloudWatch alarm. Visualize the alarm status by using the CloudWatch dashboard.

D. Create a trail by using AWS CloudTrail with data events configured. Configure the trail to send the data events to Amazon CloudWatch Logs. Create a metric filter in CloudWatch Logs to match the RDS storage autoscaling events. Visualize the metric filter by using the CloudWatch dashboard.

Answer: A

Step 1: Reacting to RDS Storage Autoscaling Events Using Amazon EventBridge

Amazon RDS emits events when storage autoscaling occurs. To visualize these events in a CloudWatch dashboard, you can create an EventBridge rule that listens for these specific autoscaling events.

Action: Create an EventBridge rule that reacts to RDS storage autoscaling events from the RDS event stream.

Why: EventBridge allows you to listen to RDS events and route them to specific AWS services for processing.

Step 2: Creating a Custom CloudWatch Metric via Lambda

Once the EventBridge rule detects a storage autoscaling event, you can use a Lambda function to publish a custom metric to CloudWatch. This metric can then be visualized in a CloudWatch dashboard.

Action: Use a Lambda function to publish custom metrics to CloudWatch based on the RDS storage autoscaling events.

Why: Custom metrics allow you to track specific events like autoscaling and visualize them easily on a CloudWatch dashboard.

Reference: AWS documentation on Publishing Custom Metrics to CloudWatch.

This corresponds to Option A: Create an Amazon EventBridge rule that reacts to RDS storage autoscaling events from RDS events. Create an AWS Lambda function that publishes a CloudWatch custom metric. Configure the EventBridge rule to invoke the Lambda function. Visualize the custom metric by using the CloudWatch dashboard.

Question: 240

A company has an AWS Cloud Format ion slack that is deployed in a single AWS account. The company has configured the stack to send event notifications to an Amazon Simple Notification Service (Amazon SNS) topic.

A DevOps engineer must implement an automated solution that applies a tag to the specific Cloud Formation stack instance only after a successful stack update occurs. The DevOps engineer has created an AWS Lambda function that applies and updates this tag (or the specific slack instance. Which solution will meet these requirements?

A. Run the AWS-UpdateCloudfomationStack AWS Systems Manager Automation runbook when Systems Manager detects an UPDATE_COMPLETE event for the instance status of the Cloud Formation stack. Configure the runbook to invoke the Lambda function.

B. Create a custom AWS Config rule that produces a compliance change event if the CloudFormation stack has an UPDATE_COMPLETE instance status. Configure AWS Config to directly invoke the Lambda function to automatically remediate the change event.

C. Create an Amazon EventBridge rule that matches the UPDATE COMPLETE event pattern for the instance status of the

CloudFormation stack. Configure the rule to invoke the Lambda function.

D. Adjust the configuration of the CloudFormation stack to send notifications for only an UPDATE COMPLETE instance status event to the SNS topic. Subscribe the Lambda function to the SNS topic.

Answer: C

Step 1: Reacting to CloudFormation Stack Events with EventBridge

AWS CloudFormation emits events during the lifecycle of a stack, including the UPDATE_COMPLETE event after a successful stack update. You can use Amazon EventBridge to detect this event and trigger a specific action (such as invoking a Lambda function).

Action: Create an EventBridge rule that listens for the UPDATE_COMPLETE event for the CloudFormation stack.

Why: EventBridge allows you to automatically detect CloudFormation stack lifecycle events and take action based on them.

Step 2: Invoking the Lambda Function

After the EventBridge rule detects the UPDATE_COMPLETE event, it can invoke the pre-configured Lambda function to apply or update the tag on the specific CloudFormation stack instance.

Action: Configure the EventBridge rule to invoke the Lambda function when the UPDATE_COMPLETE event occurs.

Why: This automation ensures that the tag is applied immediately after a successful stack update without any manual intervention.

Reference: AWS documentation on EventBridge Rules.

This corresponds to Option C: Create an Amazon EventBridge rule that matches the UPDATE_COMPLETE event pattern for the instance status of the CloudFormation stack. Configure the rule to invoke the Lambda function.

Question: 241

A company's organization in AWS Organizations has a single OU. The company runs Amazon EC2 instances in the OU accounts. The company needs to limit the use of each EC2 instance's credentials to the specific EC2 instance that the credential is assigned to. A DevOps engineer must configure security for the EC2 instances.

Which solution will meet these requirements?

A. Create an SCP that specifies the VPC CIDR block. Configure the SCP to check whether the value of the aws:VpcSourceIp condition key is in the specified block. In the same SCP check, check whether the values of the aws:EC2InstanceSourcePrivateIPv4 and aws:SourceVpc condition keys are the same. Deny access if either condition is false. Apply the SCP to the OU.

B. Create an SCP that checks whether the values of the aws:EC2InstanceSourceVPC and aws:SourceVpc condition keys are the same. Deny access if the values are not the same. In the same SCP check, check whether the values of the aws:EC2InstanceSourcePrivateIPv4 and awsVpcSourceIp condition keys are the same. Deny access if the values are not the same. Apply the SCP to the OU.

C. Create an SCP that includes a list of acceptable VPC values and checks whether the value of the aws:SourceVpc condition key is in the list. In the same SCP check, define a list of acceptable IP address values and check whether the value of the aws:VpcSourceIp condition key is in the list. Deny access if either condition is false. Apply the SCP to each account in the organization.

D. Create an SCP that checks whether the values of the aws:EC2InstanceSourceVPC and aws:VpcSourceIp condition keys are the same. Deny access if the values are not the same. In the same SCP check, check whether the values of the aws:EC2InstanceSourcePrivateIPv4 and aws:SourceVpc condition keys are the same. Deny access if the values are not the same. Apply the SCP to each account in the organization.

Answer: B

Step 1: Using Service Control Policies (SCPs) for EC2 Security

To limit the use of EC2 instance credentials to the specific EC2 instance they are assigned to, you can create a Service Control Policy (SCP) that verifies specific conditions, such as whether the EC2 instance's source VPC and private IP match expected values.

Action: Create an SCP that checks whether the values of the aws:EC2InstanceSourceVPC and aws:SourceVpc condition keys are the same. Deny access if they are not.

Why: This ensures that credentials cannot be used outside the designated EC2 instance or VPC. Step 2: Further Validation with Private IPs

The SCP should also verify that the EC2 instance's private IP matches the IP range specified for the VPC. If the instance's private IP does not match, access should be denied.

Action: In the same SCP, check whether the values of the aws:EC2InstanceSourcePrivateIP and aws:VpcSourceIP condition keys are the same. Deny access if they are not.

Why: This ensures that the credentials are only used within the specific EC2 instance and its associated VPC.

Reference: AWS documentation on Service Control Policies (SCPs).

This corresponds to Option B: Create an SCP that checks whether the values of the aws:EC2InstanceSourceVPC and aws:SourceVpc condition keys are the same. Deny access if the values are not the same. In the same SCP check, check whether the values of the aws:EC2InstanceSourcePrivateIP and aws:VpcSourceIP condition keys are the same. Deny access if the values are not the same. Apply the SCP to the OU.

Question: 242

A DevOps engineer uses AWS CodeBuild to frequently produce software packages. The CodeBuild project builds large Docker images that the DevOps engineer can use across multiple builds. The DevOps engineer wants to improve build performance and minimize costs.

Which solution will meet these requirements?

A. Store the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository. Implement a local Docker layer cache for CodeBuild.

B. Cache the Docker images in an Amazon S3 bucket that is available across multiple build hosts. Expire the cache by using an S3 Lifecycle policy.

C. Store the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository. Modify the CodeBuild project runtime configuration to always use the most recent image version.

D. Create custom AMIs that contain the cached Docker images. In the CodeBuild build, launch Amazon EC2 instances from the custom AMIs.

Answer: A

Step 1: Storing Docker Images in Amazon ECR

Docker images can be large, and storing them in a centralized, scalable location can greatly reduce build times. Amazon Elastic Container Registry (ECR) is a fully managed container registry that stores, manages, and deploys Docker container images.

Action: Store the Docker images in an ECR repository.

Why: Storing Docker images in ECR ensures that Docker images can be reused across multiple builds, improving build performance by avoiding the need to rebuild the images from scratch.

Reference: AWS documentation on Amazon ECR.

Step 2: Implementing Docker Layer Caching in CodeBuild

Docker layer caching is essential for improving performance in continuous integration pipelines. CodeBuild supports local caching of Docker layers, which speeds up builds that reuse Docker images across multiple runs.

Action: Implement Docker layer caching within the CodeBuild project.

Why: This improves performance by allowing frequently used Docker layers to be cached locally, avoiding the need to pull or build the layers every time.

Reference: AWS documentation on Docker Layer Caching in CodeBuild.

This corresponds to Option A: Store the Docker images in an Amazon Elastic Container Registry (Amazon ECR) repository. Implement a local Docker layer cache for CodeBuild.

www.atmicnetworks.com

Question: 243

A company uses an Amazon Aurora PostgreSQL global database that has two secondary AWS Regions. A DevOps engineer has configured the database parameter group to guarantee an RPO of 60 seconds. Write operations on the primary cluster are occasionally blocked because of the RPO setting.

The DevOps engineer needs to reduce the frequency of blocked write operations.

Which solution will meet these requirements?

A. Add an additional secondary cluster to the global database.
B. Enable write forwarding for the global database.
C. Remove one of the secondary clusters from the global database.
D. Configure synchronous replication for the global database.

Answer: C

Step 1: Reducing Replication Lag in Aurora Global Databases

In Amazon Aurora global databases, write operations on the primary cluster can be delayed due to the time it takes to replicate to secondary clusters, especially when there are multiple secondary regions involved.

Issue: The write operations are occasionally blocked due to the RPO setting, which guarantees replication within 60 seconds.

Action: Remove one of the secondary clusters from the global database.

Why: Fewer secondary clusters will reduce the overall replication lag, improving write performance and reducing the frequency of blocked writes.

Reference: AWS documentation on Aurora Global Database.

This corresponds to Option C: Remove one of the secondary clusters from the global database.

Question: 244

A company uses AWS WAF to protect its cloud infrastructure. A DevOps engineer needs to give an operations team the ability to analyze log messages from AWS WAR. The operations team needs to be able to create alarms for specific patterns in the log output.

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

A. Create an Amazon CloudWatch Logs log group. Configure the appropriate AWS WAF web ACL to send log messages to the log group. Instruct the operations team to create CloudWatch metric filters.
B. Create an Amazon OpenSearch Service cluster and appropriate indexes. Configure an Amazon Kinesis Data Firehose delivery stream to stream log data to the indexes. Use OpenSearch Dashboards to create filters and widgets.
C. Create an Amazon S3 bucket for the log output. Configure AWS WAF to send log outputs to the S3 bucket. Instruct the operations team to create AWS Lambda functions that detect each desired log message pattern. Configure the Lambda functions to publish to an Amazon Simple Notification Service (Amazon SNS) topic.
D. Create an Amazon S3 bucket for the log output. Configure AWS WAF to send log outputs to the S3 bucket. Use Amazon Athena to create an external table definition that fits the log message pattern. Instruct the operations team to write SOL queries and to create Amazon CloudWatch metric filters for the Athena queries.

Answer: A

Step 1: Sending AWS WAF Logs to CloudWatch Logs

AWS WAF allows you to log requests that are evaluated against your web ACLs. These logs can be sent directly to CloudWatch Logs, which enables real-time monitoring and analysis.

Action: Configure the AWS WAF web ACL to send log messages to a CloudWatch Logs log group. Why: This allows the operations team to view the logs in real time and analyze patterns using CloudWatch metric filters.
Reference: AWS documentation on AWS WAF Logs to CloudWatch.
Step 2: Creating CloudWatch Metric Filters
CloudWatch metric filters can be used to search for specific patterns in log data. The operations team can create filters for certain log patterns and set up alarms based on these filters.
Action: Instruct the operations team to create CloudWatch metric filters to detect patterns in the WAF log output.
Why: Metric filters allow the team to trigger alarms based on specific patterns without needing to manually search through logs.
Reference: AWS documentation on Metric Filters in CloudWatch Logs.
This corresponds to Option A: Create an Amazon CloudWatch Logs log group. Configure the appropriate AWS WAF web ACL to send log messages to the log group. Instruct the operations team to create CloudWatch metric filters.

Question: 245

A company uses AWS Organizations to manage its AWS accounts. A DevOps engineer must ensure that all users who access the AWS Management Console are authenticated through the company's corporate identity provider (IdP).
Which combination of steps will meet these requirements? (Select TWO.)

A. Use Amazon GuardDuty with a delegated administrator account. Use GuardDuty to enforce denial of 1AM user logins
B. Use AWS 1AM Identity Center to configure identity federation with SAML 2.0.
C. Create a permissions boundary in AWS 1AM Identity Center to deny password logins for 1AM users.
D. Create 1AM groups in the Organizations management account to apply consistent permissions for all 1AM users.
E. Create an SCP in Organizations to deny password creation for 1AM users.

Answer: BE

Step 1: Using AWS IAM Identity Center for SAML-based Identity Federation
To ensure that all users accessing the AWS Management Console are authenticated via the corporate identity provider (IdP), the best approach is to set up identity federation with AWS IAM Identity Center (formerly AWS SSO) using SAML 2.0.
Action: Use AWS IAM Identity Center to configure identity federation with the corporate IdP that supports SAML 2.0.
Why: SAML 2.0 integration enables single sign-on (SSO) for users, allowing them to authenticate through the corporate IdP and gain access to AWS resources.
Reference: AWS documentation on IAM Identity Center and SAML Federation.
This corresponds to Option B: Use AWS IAM Identity Center to configure identity federation with SAML 2.0.
Step 2: Creating an SCP to Deny Password Logins for IAM Users
To enforce that IAM users do not create passwords or access the Management Console directly without going through the corporate IdP, you can create a Service Control Policy (SCP) in AWS Organizations that denies password creation for IAM users.
Action: Create an SCP that denies password creation for IAM users.
Why: This ensures that users cannot set passwords for their IAM user accounts, forcing them to use federated access through the corporate IdP for console login.
Reference: AWS documentation on Service Control Policies.
This corresponds to Option E: Create an SCP in Organizations to deny password creation for IAM users.

Question: 246

A company recently migrated its application to an Amazon Elastic Kubernetes Service (Amazon EKS) cluster that uses Amazon EC2 instances. The company configured the application to automatically scale based on CPU utilization.
The application produces memory errors when it experiences heavy loads. The application also does not scale out enough to handle the

increased load. The company needs to collect and analyze memory metrics for the application over time.
Which combination of steps will meet these requirements? (Select THREE.)

A. Attach the Cloud WatchAgentServer Pol icy managed 1AM policy to the 1AM instance profile that the cluster uses.
B. Attach the Cloud WatchAgentServer Pol icy managed 1AM policy to a service account role for the cluster.
C. Collect performance metrics by deploying the unified Amazon CloudWatch agent to the existing EC2 instances in the cluster. Add the agent to the AMI for any new EC2 instances that are added to the cluster.
D. Collect performance logs by deploying the AWS Distro for OpenTelemetry collector as a DaemonSet.
E. Analyze the pod_memory_utilization Amazon CloudWatch metric in the ContainerInsights namespace by using the Service dimension.
F. Analyze the node_memory_utilization Amazon CloudWatch metric in the ContainerInsights namespace by using the ClusterName dimension.

Answer: ACE

Step 1: Attaching the CloudWatchAgentServerPolicy to the IAM Role
The CloudWatch agent needs permissions to collect and send metrics, including memory metrics, to Amazon CloudWatch. You can attach the CloudWatchAgentServerPolicy managed IAM policy to the IAM instance profile or service account role to grant these permissions.
Action: Attach the CloudWatchAgentServerPolicy managed IAM policy to the IAM instance profile that the EKS cluster uses.
Why: This ensures the CloudWatch agent has the necessary permissions to collect memory metrics. Reference: AWS documentation on CloudWatch Agent Permissions.
This corresponds to Option A: Attach the CloudWatchAgentServerPolicy managed IAM policy to the IAM instance profile that the cluster uses.
Step 2: Deploying the CloudWatch Agent to EC2 Instances
To collect memory metrics from the EC2 instances running in the EKS cluster, the CloudWatch agent needs to be deployed on these instances. The agent collects system-level metrics, including memory usage.
Action: Deploy the unified Amazon CloudWatch agent to the existing EC2 instances in the EKS cluster. Update the Amazon Machine Image (AMI) for future instances to include the CloudWatch agent. Why: The CloudWatch agent allows you to collect detailed memory metrics from the EC2 instances, which is not enabled by default.
Reference: AWS documentation on Installing and Configuring the CloudWatch Agent.
This corresponds to Option C: Collect performance metrics by deploying the unified Amazon CloudWatch agent to the existing EC2 instances in the cluster. Add the agent to the AMI for any new EC2 instances that are added to the cluster.
Step 3: Analyzing Memory Metrics Using Container Insights
After collecting the memory metrics, you can analyze them using the pod_memory_utilization metric in Amazon CloudWatch Container Insights. This metric provides visibility into the memory usage of the containers (pods) in the EKS cluster.
Action: Analyze the pod_memory_utilization CloudWatch metric in the Container Insights namespace by using the Service dimension.
Why: This provides detailed insights into memory usage at the container level, which helps diagnose memory-related issues.
Reference: AWS documentation on CloudWatch Container Insights.
This corresponds to Option E: Analyze the pod_memory_utilization Amazon CloudWatch metric in the Container Insights namespace by using the Service dimension.

Question: 247

A company has developed a static website hosted on an Amazon S3 bucket. The website is deployed using AWS CloudFormation. The CloudFormation template defines an S3 bucket and a custom resource that copies content into the bucket from a source location.
The company has decided that it needs to move the website to a new location, so the existing CloudFormation stack must be deleted and re-created. However, CloudFormation reports that the stack could not be deleted cleanly.

What is the MOST likely cause and how can the DevOps engineer mitigate this problem for this and future versions of the website?

A. Deletion has failed because the S3 bucket has an active website configuration. Modify the Cloud Formation template to remove the WebsiteConfiguration properly from the S3 bucket resource.

B. Deletion has failed because the S3 bucket is not empty. Modify the custom resource's AWS Lambda function code to recursively empty the bucket when RequestType is Delete.

C. Deletion has failed because the custom resource does not define a deletion policy. Add a DeletionPolicy property to the custom resource definition with a value of RemoveOnDeletion.

D. Deletion has failed because the S3 bucket is not empty. Modify the S3 bucket resource in the CloudFormation template to add a DeletionPolicy property with a value of Empty.

Answer: B

Step 1: Understanding the Deletion Failure

The most likely reason why the CloudFormation stack failed to delete is that the S3 bucket was not empty. AWS CloudFormation cannot delete an S3 bucket that contains objects, so if the website files are still in the bucket, the deletion will fail.

Issue: The S3 bucket is not empty during deletion, preventing the stack from being deleted.

Step 2: Modifying the Custom Resource to Handle Deletion

To mitigate this issue, you can modify the Lambda function associated with the custom resource to automatically empty the S3 bucket when the stack is being deleted. By adding logic to handle the RequestType: Delete event, the function can recursively delete all objects in the bucket before allowing the stack to be deleted.

Action: Modify the Lambda function to recursively delete the objects in the S3 bucket when RequestType is set to Delete.

Why: This ensures that the S3 bucket is empty before CloudFormation tries to delete it, preventing the stack deletion failure.

Reference: AWS documentation on CloudFormation custom resources.

This corresponds to Option B: Deletion has failed because the S3 bucket is not empty. Modify the custom resource's AWS Lambda function code to recursively empty the bucket when RequestType is Delete.

Question: 248

A company uses an AWS CodeCommit repository to store its source code and corresponding unit tests. The company has configured an AWS CodePipeline pipeline that includes an AWS CodeBuild project that runs when code is merged to the main branch of the repository. The company wants the CodeBuild project to run the unit tests. If the unit tests pass, the CodeBuild project must tag the most recent commit.

How should the company configure the CodeBuild project to meet these requirements?

A. Configure the CodeBuild project to use native Git to clone the CodeCommit repository. Configure the project to run the unit tests. Configure the project to use native Git to create a tag and to push the Git tag to the repository if the code passes the unit tests.

B. Configure the CodeBuild project to use native Git to clone the CodeCommit repository. Configure the project to run the unit tests. Configure the project to use AWS CLI commands to create a new repository tag in the repository if the code passes the unit tests.

C. Configure the CodeBuild project to use AWS CLI commands to copy the code from the CodeCommit repository. Configure the project lo run the unit tests. Configure the project to use AWS CLI commands to create a new Git tag in the repository if the code passes the unit tests.

D. Configure the CodeBuild project to use AWS CLI commands to copy the code from the CodeCommit repository. Configure the project to run the unit tests. Configure the project to use AWS CLI commands to create a new repository tag in the repository if the code passes the unit tests.

Answer: A

Step 1: Using Native Git in CodeBuild

To meet the requirement of running unit tests and tagging the most recent commit if the tests pass, the CodeBuild project should be configured to use native Git to clone the CodeCommit repository. Native Git support allows full functionality for managing the repository, including the ability to create and push tags.

Action: Configure the CodeBuild project to use native Git to clone the repository and run the tests. Why: Using native Git provides flexibility for managing tags and other repository operations after the tests are successfully executed.

Step 2: Tagging the Most Recent Commit

Once the unit tests pass, the CodeBuild project can use native Git to create a tag for the most recent commit and push that tag to the repository. This ensures that the tagged commit is linked to the test results.

Action: Configure the project to use native Git to create and push a tag to the repository if the tests pass.

Why: This ensures the correct commit is tagged automatically, streamlining the workflow. Reference: AWS documentation on AWS CodeBuild and Git integration.

This corresponds to Option A: Configure the CodeBuild project to use native Git to clone the CodeCommit repository. Configure the project to run the unit tests. Configure the project to use native Git to create a tag and to push the Git tag to the repository if the code passes the unit tests.

Question: 249

A company has an organization in AWS Organizations for its multi-account environment. A DevOps engineer is developing an AWS CodeArtifact based strategy for application package management across the organization. Each application team at the company has its own account in the organization. Each application team also has limited access to a centralized shared services account. Each application team needs full access to download, publish, and grant access to its own packages. Some common library packages that the application teams use must also be shared with the entire organization.

Which combination of steps will meet these requirements with the LEAST administrative overhead? (Select THREE.)

A. Create a domain in each application team's account. Grant each application team's account lull read access and write access to the application team's domain

B. Create a domain in the shared services account Grant the organization read access and CreateRepository access.

C. Create a repository in each application team's account. Grant each application team's account lull read access and write access to its own repository.

D. Create a repository in the shared services account. Grant the organization read access to the repository in the shared services account. Set the repository as the upstream repository in each application team's repository.

E. For teams that require shared packages, create resource-based policies that allow read access to the repository from other application teams' accounts.

F. Set the other application teams' repositories as upstream repositories.

Answer: BDE

Step 1: Creating a Centralized Domain in the Shared Services Account

To manage application package dependencies across multiple accounts, the most efficient solution is to create a centralized domain in the shared services account. This allows all application teams to access and manage package repositories within the same domain, ensuring consistency and centralization.

Action: Create a domain in the shared services account.

Why: A single, centralized domain reduces the need for redundant management in each application team's account.

Reference: AWS documentation on AWS CodeArtifact domains and repositories.

This corresponds to Option B: Create a domain in the shared services account. Grant the organization read access and CreateRepository access.

Step 2: Sharing Repositories Across Teams with Upstream Configurations

To share common library packages across the organization, each application team's repository can point to the shared services repository as an upstream repository. This enables teams to access shared packages without managing them individually in each team's account.

Action: Create a repository in the shared services account and set it as the upstream repository for each application team.

Why: Upstream repositories allow package sharing while maintaining individual team repositories for managing their own packages.

Reference: AWS documentation on Upstream repositories in CodeArtifact.

This corresponds to Option D: Create a repository in the shared services account. Grant the organization read access to the repository in the shared services account. Set the repository as the upstream repository in each application team's repository.

Step 3: Using Resource-Based Policies for Cross-Account Access

For teams that need to share their packages with other application teams, resource-based policies can be applied to grant the necessary permissions. These policies allow cross-account access without having to manage permissions at the individual account level.

Action: Create resource-based policies that allow read access to the repositories across application teams.

Why: This simplifies management by centralizing permissions in the shared services account while allowing cross-team collaboration.

Reference: AWS documentation on CodeArtifact resource-based policies.

This corresponds to Option E: For teams that require shared packages, create resource-based policies that allow read access to the repository from other application teams' accounts.

Question: 250

A company releases a new application in a new AWS account. The application includes an AWS Lambda function that processes messages from an Amazon Simple Queue Service (Amazon SOS) standard queue. The Lambda function stores the results in an Amazon S3 bucket for further downstream processing. The Lambda function needs to process the messages within a specific period of time after the messages are published. The Lambda function has a batch size of 10 messages and takes a few seconds to process a batch of messages. As load increases on the application's first day of service, messages in the queue accumulate at a greater rate than the Lambda function can process the messages. Some messages miss the required processing timelines. The logs show that many messages in the queue have data that is not valid. The company needs to meet the timeline requirements for messages that have valid data.

Which solution will meet these requirements?

A. Increase the Lambda function's batch size. Change the SOS standard queue to an SOS FIFO queue. Request a Lambda concurrency increase in the AWS Region.

B. Reduce the Lambda function's batch size. Increase the SOS message throughput quota. Request a Lambda concurrency increase in the AWS Region.

C. Increase the Lambda function's batch size. Configure S3 Transfer Acceleration on the S3 bucket. Configure an SOS dead-letter queue.

D. Keep the Lambda function's batch size the same. Configure the Lambda function to report failed batch items. Configure an SOS dead-letter queue.

Answer: D

Step 1: Handling Invalid Data with Failed Batch Items

The Lambda function is processing batches of messages, and some messages contain invalid data, causing processing delays. Lambda provides the capability to report failed batch items, which allows valid messages to be processed while skipping invalid ones. This functionality ensures that the valid messages are processed within the required timeline.

Action: Keep the Lambda function's batch size the same and configure it to report failed batch items. Why: By reporting failed batch items, the Lambda function can skip invalid messages and continue processing valid ones, ensuring that they meet the processing timeline.

Reference: AWS documentation on Lambda batch item failure handling.

Step 2: Using an SQS Dead-Letter Queue (DLQ)

Configuring a dead-letter queue (DLQ) for SQS will ensure that messages with invalid data, or those that cannot be processed successfully, are moved to the DLQ. This prevents such messages from clogging the queue and allows the system to focus on processing valid messages.

Action: Configure an SQS dead-letter queue for the main queue.

Why: A DLQ helps isolate problematic messages, preventing them from continuously reappearing in the queue and causing processing delays for valid messages.

Reference: AWS documentation on Amazon SQS Dead-Letter Queues.

Step 3: Maintaining the Lambda Function's Batch Size

Keeping the current batch size allows the Lambda function to continue processing multiple messages at once. By addressing the failed items separately, there's no need to increase or reduce the batch size.

Action: Maintain the Lambda function's current batch size.

Why: Changing the batch size is unnecessary if the invalid messages are properly handled by reporting failed items and using a DLQ.

This corresponds to Option D: Keep the Lambda function's batch size the same. Configure the Lambda function to report failed batch items. Configure an SQS dead-letter queue.