Amazon.DOP-C02-KR.v2026-05-20.q183

□□□□:	DOP-C02-KR
□□□□:	AWS Certified DevOps Engineer - Professional (DOP-C02 Korean Version)
□□□:	Amazon
□□ □□ □□□:	183
□□:	v2026-05-20
# □□ □:	166
# □□ □□□:	1830
https://www.krdump.com/Amazon.DOP-C02-KR.v2026-05-20.q183.html	

NEW QUESTION: 1
□ □□□ AWS Organizations, AWS Control Tower, AWS Config, Terraform□ □□□□ AWS □□□ □□□□ □□□□□. □□□ □□□ □ □□ AWS □□□ VPC□ □□□ AWS Lambda □□□ □□□□□ □□ □□□.
□□ □□□□ □□□□ □□ □□□□ □□□ □□ □□□ □□□ □ □□□□?
A. AWS Control Tower□ □□□□ □□ □□□ □□(□□□□)□ □□□□□. □□ OU□□ Lambda□ AWS CloudFormation □□□ □□ □ □□□ □□□ □□□□□□.
B. □□□□ □□ □□ □□:VpcIds □□ □□ □□□□ □ SCP□ □□□□.
C. VPC□ □□□□ □□ Lambda □□□ □□□□ □□ □□□ □□ AWS Config □□□ □□□□.
D. lambda:VpcIds□ null□ □□ Lambda □□□ □□□□ □□□□□ □ SCP□ □□□□.
Answer: (SHOW ANSWER)
Use a Service Control Policy (SCP) with a Null condition on lambda:VpcIds to deny Lambda function creation or update when not VPC-attached. This enforces compliance across all accounts automatically without manual remediation, aligning with AWS Control Tower governance recommendations.

NEW QUESTION: 2
DevOps □□□□□ AWS Lambda □□□ □□□□□□. Lambda □□□ □□ CloudFormation □□□ □□ □□□□ □□ □□□□□ AWS CloudFormation □□□□ □□ □□□ □□□□□. □□ □□ Lambda □□□ □□ □□□ □□□□□. DevOps □□□□□ Lambda □□□ □□□□ Amazon EventBrdge □□ □□□ □□□□□□. □□□. Amazon Simple □□ □□□(Amazon SNS) □□□ AWS □□□ □□ □□□□□. DevOps □□□□□ □□□ □□ □□ SNS □□□ □□□□□□. DevOps □□□□□ □ □□ □□ □□□□ □□□□□ □□□□ □□□ □ □□ □□□ □□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. SNS □□□ □□□□ □□□□□ □□ EventBridge □□□ □□□□□. Cloud Formation □□□ □□□□□ SNS □□ □□ □□□ □ □□□□. SNS tomc□ □□ □□ □□□ □□□□□.
B. □ □□ Lambda □□□ □□□□ □□□ □□□□ □□ □□□ CloudFormation API□ □□□□□. SNS □□□ □□□□ □□□□□ □ □□ Lambda □□□ □□□□□. □□□□□ □□□□ □ □□ Lambda□ □□□□ □□□□□ □□ EventBridge □□□ □□□□□. □□

C. □□ CloudFormation □□□ □□ □□□□ □□ □□□ □□ □□□□ Amazon GuardDuty□ □□□□□. □□ CloudFormation □□□ □□ GuardDuty □□□□ □□ □□□ □□□ □□□□ □ □□ EventBndge □□□ □□□□□. SNS □□□ □ □□ EventBridge □□□ □□□□ □□□□□.
D. □□□□ AWS Config□ □□□□□. cloudformation-stack-drift-Detection-check □□□ □□□ □□□□□. CloudFormaUon □□□ □ □ □□ □□ □□ □□□□ □□□□ □ □□ EventBndge □□□ □□□□□. SNS topc□ □ □□ EventBridge □□□ □□□□ □□□□ □.
Answer: D (LEAVE A REPLY)
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

NEW QUESTION: 3
□ □□□□ □□ □□ EC2 □□□□□ □□□□ □□□, □□□ □□□ □□□□□ □□/□□□□□. □□□ □□ □□ □□ □□ □□□ □□ □□□□ □□□□ □□□□ □□□. □□ □□□□ □□□□, □□□□ □□ □□□□□□ □□□□ □□□□□ □□□.
□□ □□□□ □ □□ □□□ □□□□□?
A. RunInstances API □□□ □□ EventBridge □□, Lambda□ □□□□ □□ □□□□ □□□□□.
B. ec2-instance-profile-attached □□ □□□ □□□ AWS Config, □□□□ □□□□ □□ Systems Manager Automation □□□ □□□□ □□ □□.
C. StartInstances API □□□ □□ EventBridge □□, □□□□ □□□□ □□ Systems Manager Automation □□□ □□□□□.
D. AWS Config iam-role-managed-policy-check □□ □□, Lambda□ □□□□ □□□□ □□□□ □□□□ □□□□□.
Answer: (SHOW ANSWER)
* AWS Config's ec2-instance-profile-attached managed rule checks for attached instance profiles.
* Config supports automatic remediation via Systems Manager Automation runbooks.
* This provides continuous compliance with minimal operational overhead.
* EventBridge and Lambda (A) require custom coding and risk missing existing instances.
* StartInstances (C) does not cover RunInstances and new instances.

* IAM-role managed policy check (D) does not check instance profile attachments.
References:
AWS Config Managed Rules
Config Automatic Remediation

NEW QUESTION: 4
□□ □□ □□□ □□ □□□□ □□ □□ □□□□ □□□□. □□ □□□ □□□ Amazon API Gateway□ □□ □□□□ Amazon Simple Queue Service(Amazon SQS) □□ □□□□ □□ □□□□□. □□□ □□ □□□□□□□ □□□□ □□□□ □□□□ □□ □□□□ □ □□□□.
□□□ □□□□ □□□□ □□□□ □□ □□□□□□□ □□□ □□□□ □□□ □ □□□□. □□□ □□ □□□□ SQS □□□□ □□ □□□□. DevOps □□□□□ □□□ □□□□ □□□□ □□ □ □□ □□□ □□ □□□□ □□□ □ □□□ □□ □□□□ □□□□ □ □□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. SQS □□□□ □□□□ Lambda □□□ □□□□ □□□ □□□□ □□□□ □□□□□ AWS Lambda□ □□□□□. □□□ □□□□ □□□□ □□□□ □□□□ □□□ □ □□□ □□□□ □□ □□□ □□□ Amazon S3 □□□□ □□□□. □□□□ □□□□ □□□ □ □□□ □□ Lambda □□□ □□□□ SQS □□□□ □□□□ □□□□□.
B. SQS □□ □□□□ SQS FIFO □□□□ □□□□□. Amazon EventBridge □□□ □□□□ 10□□□ SQS □□□□ □□□□□ AWS Lambda□ □□□□□. Lambda □□□ □□□□ 5□□□ □□□ SentTimestamp □□ □□ □□□□ □□□□ □□□□ □□□□□□□ □□ □□□ □□□ □□□ □□□□ □□□□□ □□□□ □□□□□.
C. SQS □□ □□ □□ □□□□ □□□□□. Maximum Receives □□□ 1□ □□□□ □□ □□ □□ □□□ ARN□ □□ □□□ □□□ □ ARN□□ □□□□ □□□□□ □□□ □□□□ □□ □□□□ □□□□□. □□□□ □□ □□□□ □□□□ □□ □□ □□ □□ □□ □□ □□□□□ □□□□□□ □□□□□. □□□ □ □□□□ □□ □□□□□□.
D. □ □□□ □□ □□□ □□□ □□ SQS □□ □□□□ □□□□ □□□□ API Gateway□ □□□□□. □□□ □ □□ □□ □□□□ □□ □ □□ □□ □□□□□ □□□□□□□ □□□□□□ □□□□ □ □□ □□ □□□□. □□□□□□ □□ □□□□ □□□□ □□ □□ □□ □□□□ □□□□□ □□□□□. □□□ □ □□□□ □□ □□□□□□.
Answer: C (LEAVE A REPLY)
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.

NEW QUESTION: 5
□□□□□ □□ AWS CodePipeline□ □□□□ Java □□□□□□ □□□ □□□□□□ □□□□□ □□□□. □□□□□□ □□ □□, □□ □□, □□ □□□ □□□□□. □ □□□□ runOrder □□ 1□ □□ □□□ □□□□ □□□□.
□□ □□ □□□□ □□ □□□ □□□□□□ □□□□□ □□□. □□□ □□ □□ □□□□ □□□ □□ □□ □□□ □□□□ □□□□ □□□□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □□ □□□ □□□□□. runOrder □□ 1□ □□□ □□□ □□□□□. AWS CodeDeploy□ □□ □□□□ □□□□ □□ □□□□ □ □□□□.
B. □□ □□ □□ runOrder □□ 2□ □□□ □□ □□ AWS CodeBuild□ □□ □□□□ □□□□ □□ □□□ □□
C. □□ □□ □□ runOrder □□ 1□ □□□ □□ □□ AWS CodeDeploy□ □□ □□□□ □□□□ □□ □□□ □□

D. □□ □□ □□ runOrder □□ 2□ □□□ □□ □□ AWS CodeBuild□ □□ □□□□ □□□□ □□ □□□ □□

Answer: B (LEAVE A REPLY)

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.

References:

AWS CodePipeline

AWS CodeBuild

Using CodeBuild with CodePipeline

NEW QUESTION: 6
□ □□□□□ □□□ □□□ □□□□ □□□□□ □□ AWS□ □□□□□□. □ □□□ DevOps □□ AWS Control Tower □□ □ □□ □ □□□□□□. DevOps □□ AWS IAM Identity Center(AWS Single Sign-On) □□ □□ □□ □□□□ □□ □□ □□ □□□(IdP)□ □ □□□ SAML 2.0□ □□□□□□.
DevOps □□ □□ □□ □□□ □□□□ □□□ □□ □□□ □□□□. □□□ □□ □□□ □□□□ □□□□ □□□ □ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (3□□ □□□□□.)
A. □□□ □□□ □□□□ IAM □□□ □□□□. aws:PrincipalTag □□ □□ □□□□□.
B. □□ □□□ □□□□. □□□ □□□ □□□□ aws:PrincipalTag □□ □□ □□□□ □□ □□□ □□□□ □□□ □□□ □□□□□.
C. IdP□ □□□ □□□□□. □□□□ □□□ □□□□□. IAM Identity Center□ □□ □ □□ □□□ □□□ □□□□□.
D. IdP□ □□□ □□□□□. □□□□ □□□ □□□□□. □□□ OU □ IAM □□□ □□□□□.
E. IAM Identity Center□□ □□□ □□□ □□ □□□ □□□□□□. □□□□□ □□□ □□□□□. □□□ □-□ □□□ □□□□□.
F. IAM Identity Center□□ □□□ □□□ □□ □□□ □□□□□□. IdP□ □□□ □-□ □□□ □□□□□.
Answer: (SHOW ANSWER)
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

NEW QUESTION: 7
□□□□ □□□□□□□ CI/CD □□□□□□ □□□□. CI/CD □□□□□□ AWS CodePipeline □□□□□□ AWS CodeBuild □□□□ □ □□□□□. CodeBuild □□□□□ □□ □□□□□ □□□ □□□□□□□ □□ □□□□ □□□□ □□□ □□□□ □□□□□. □□ □ □□□ □□□□ 90□ □□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. CodeBuild □□□□□ □□□ □□ □□ CodePipeline □□□□□□ □ □□□ □□□□□. □□□□ □□□ Amazon S3 □□□ □□□ □□. □□□□ □□ □□□ □□□ □□□ □□□□ □ CodePipeline □□□□ S3 □□ □□ □□□ □□□□□.
B. □□□□ □□ □□□ □□□ □□□ □□□□ CodeBuild □□□□ buildspec □□□ □□□ □□□ □□□□□. □□□□ □□□ Amazon S3 □□□ □□□□□. □□□ □□□□ S3 □□□ □□□□ □□□□ □□ AWS Lambda □□□ □□□□ Amazon EventBridge □□□ □□□□□. 90□ □□ □□□ □□□□□ S3 □□ □□ □□□ □□□□□.
C. CodePipeline □□□□□□ □ □□□ □□□□□. □□□□ □□ □□□ □□□ □□□□ □□□ □□ □□□ □□□□□. CodeBuild □□□□ buildspec □□□□ □□□ □□ □□□ 90□□ □□□□□.
D. □□□□ □□ □□□ □□□ □□□ □□□□ CodeBuild □□□□ buildspec □□□ □□□ □□□ □□□□□. □□□□ □□□ Amazon S3 □□□ □□□□□. CodeBuild □□□□ buildspec □□□□ □□□ □□□ □□□□□ □□□□□. S3 □□□ □□□□ □□ □□ □□□□□. □□ □□□ 90□□ □□□□□.
Answer: B (LEAVE A REPLY)
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 intervention1.
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.

NEW QUESTION: 8
□□□□ □□□□□(SaaS) □□□ □□□□□□ □□ □□□(ALB) □□ □□ Amazon Elastic Container Service(Amazon ECS) □□□□ □ □□□□ □□□□ □□□ □□ □□□□ □□□□□. □ □□□ AWS CodePipeline□ AWS CodeDeploy□ □□□□ □□□□ □□ □ □/□□ □□ □□□□□ □□□□ □□□□.
□□ □□□ □□ □□ □□ □□□ □ □□ □□ □□ □□□□ □□□□ □□□□ □□□□□ □□ □□□□□ □□□□. □□ □□□□ □□□□□ □□□□ □□ □□□ □□□ □□□ □□□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□□□ ECS □□□□ □□□□ □ □□□□ CodeDeploy □□□□□□□ appspec.yaml □□□□ TrafficRoutingConfig □□□□□ TimeBasedLinear□ □□□□□. linearPercentage □□□□□ linearInterval □□□□□ □□ □□□□□.
B. □□□□ ECS □□□□ □□□□ □ □□□□ CodeDeploy □□□□□□□ appspec.yaml □□□ □□ TrafficRoutingConfig □□□□ □ AllAtOnce □□□□ □□□□□□□.
C. □□ □□ □□□ □□□□□. TrafficRoutingConfig □□□□□ TimeBasedCanary □□□□ □□□□□. □□□ □□□□ □□ □□ □ □□□ □□□□ □□□□□ ALB□□ □□□ □□□ □□□□□.
D. CodeDeploy□□ □□ □□□ □□□□□. □□ □□ ALB□□ □□□ □□□□ □□□□□.
Answer: A (LEAVE A REPLY)
Comprehensive and Detailed Explanation From Exact Extract of DevOps Engineer documents only:
The requirement is to shift traffic automatically in equal increments over a defined total deployment time with no manual intervention. In AWS CodeDeploy blue/green deployments for Amazon ECS, the TimeBasedLinear traffic routing option is specifically designed for this purpose.
Option A is correct because:
* TimeBasedLinear shifts traffic in equal percentages.
* The deployment proceeds automatically at each interval.
* linearPercentage defines how much traffic is shifted each time.
* linearInterval defines how often the traffic shift happens.
* This supports gradual traffic shifting and zero-downtime blue/green deployment behavior.
Why the other options are incorrect:
B). AllAtOnce shifts all traffic immediately, not in equal increments over time.
C). TimeBasedCanary shifts traffic in an initial portion and then the remainder later. That is not equal incremental shifting across the full deployment period.
D). This option is too generic and does not specifically identify the CodeDeploy configuration that provides equal incremental automatic traffic shifting. The exact feature required is TimeBasedLinear.

NEW QUESTION: 9
□ □□□ □ □□ AWS □□□ □□□ □□□□□□□ □□□□□□. □□□□□□□ □ □□ □□□□ ALB(Application Load Balancer) □ □□□□□. □□□ □ ALB□ □□ Amazon Route 53 □□ DNS □□□□ □□□□ □□□□.

□ □□□ Amazon Route 53 □□□□□□ □□ □□□□□ □□□□ □□□□□□□ □ □□ □□ □□ □□□ □□□ □ □□□ □□□ □□. Route 53 ARC □□□□ □ □□ □□□ □□ □□□ □□□ □□□□ □□□□. □ □□□ Route 53 ARC□ □□□□ □□□ □□ □ □(DR) □□□□ □□□□□.

□□ □□□ DR □□□□□ DevOps □□□□□ □□□ □ □□□ □□□ □□ □ □□□□□. □□□□□ □□ □□ □□□ □□□ □□ □ □□ □□□ □□□□ □□□.

□□ □□□□ □□□ □□ □□□ □□□□□?

A. Route 53 ARC□□. □□□ □□□ □□ □□□ □□□□. □ □□ □□□ □□□ □□□ □□ □□□ □□□□□. □□□□ 1□ ATLEAST □□□□ □□□ □□□□□.

B. Route 53 ARC□□ □□□ □□□ □□ □□□ □□□□. □ □□ □□□ □□□ □□□ □□ □□□ □□□□□. □□□□ 1□ OR □□ □□ □□□ □□□□□.

C. Route 53 ARC□□ □ □□□ □□□ □□□□□. AWS: Route53: HealthCheck □□□ □□□□ □□□ □□□ □□□□□. □ □□□ □□□ ARN□ □□ □□□□ □□□□□. □□□ □□□ □□ □ □□ □□□ □□□□.

D. Route 53 ARC□□ □ □□□ □□□ □□□□□. AWS: Route53RecoveryReadiness: DNSTargetResource □□□ □□□□ □□□ □ □□ □□□□□. □ Route 53 □□ DNS □□□□ □□□ □□□ □□ □□□□ □□□□□. □□□ □□□ □□ □ □□ □□□ □□□□.

Answer: (SHOW ANSWER)

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. References:

* 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

NEW QUESTION: 10

□ □□□□ AWS□□ □□□□□ □□□□ □ □□□□□□□ □□□□□□. □□□□□□□ Amazon S3□ □□□□□. Amazon API Gateway, □□ AWS Lambda □□ □ Amazon RDS for MySQL □□□□□□. □ □□□ AWS CodeCommit□ □□□□ □□ □□□ □□ □□ □□□□. □□ □□□ AWS Serverless Application Model(AWS SAM) □□□□ Python □□□ □□□□□.

□□ □□ □ □□ □□□□□ □□□□□□ □□□ □□ □□□ □□□ □□□ CodeCommit □□□□□ □□ □□□□□□ □□□ □□□ □□□. DevOps □□□□□ □□□□□ □□□ □□□□ □□□□ □□□□ □□□□ □□□□ □□□.

□□□ □□ □□□ □□□□ □□ □□□ □□□□ □□□□□?

A. Amazon CodeGuru □□□□□□ □□□□□□. □□□ □□□ @with_lambda_profiler()□ □□□□□. □□ □□ □□□□ □□□□ □□□□□. □□□ □□ □□□□ AWS Systems Manager Parameter Store□ □□□. SAM □□□□ Python □□□ □□□□□□ Parameter Store□□ □□□ □□□□□.
B. CodeCommit □□□□□□ Amazon CodeGuru □□□□ □□□□□. □□ □□□ □□ □□ □□□ □□□□ □□□□□□. □□□ □ □□□ □□□ □□□□□. SAM □□□□ Python □□□ □□□□□□ AWS Secrets Manager□□ □□□ □□□□□.
C. Amazon CodeGuru □□□□□□ □□□□□□. □□□ □□□ @with_lambda_profiler()□ □□□□□. □□ □□ □□□□ □□□□ □□□□□. □□□ □□□□ □□□ □□□□□. SAM □□□□ Python □□□ □□□□□□ AWS Secrets Manager□□ □□□ □□□ □□.
D. CodeCommit □□□□□□ Amazon CodeGuru □□□□ □□□□□. □□ □□□ □□ □□ □□□ □□□□ □□□□□□. AWS Systems Manager Parameter Store□ □□□ □□□□ □□□. SAM □□□□ Python □□□ □□□□□□ Parameter Store□□ □□□ □□□□□.
Answer: (SHOW ANSWER)
https://docs.aws.amazon.com/codecommit/latest/userguide/how-to-amazon-codeguru-reviewer.html

NEW QUESTION: 11
□ □□□ AWS Organizations□□ □□ AWS □□□ □□□□ □□□□. □□□ □□ □□□ □□□ □□ □□□ AWS □□ □□ □□□ □□ □□□ □□□□□ □ □□□. □□□ □□ □□□ □□ □□□ □□ □□□ □□□□□□□.
□□ □□□ □□□ □□□ □□ □□□ □□ □□□□ Amazon EC2 □□□□□ □□□□□□. DevOps □□□□□ □□□ □□ □□□□ SCP(□□□ □□ □□)□ □□□□ □□ □□□ □□ □□□□ AWS □□□ API□ □□□□ □□□□ □□□□ □□□.
□□ SCP□ □□□ □□ □□□ □□□□□?

A.

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

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

B.

C.

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

Answer: B (LEAVE A REPLY)

NEW QUESTION: 12

□ □□□ □ □□□□□□ □□ □□ □□□□□□□ □□□□. □ □□□ Amazon CloudFront, □□□□□□ □□ □□□(ALB), □□□ Amazon EC2 □□□□□ Auto Scaling □□□□ □□□□, □□□ 3□□□. □□ □□ □□□ AMI□ □□□□□. □□□□□□□ 1□ □□ □□□□□. □ □□□□□□□ Amazon OpenSearch □□□ □□□□□ □□□□, □□ □□ □□□ □□□□□. □□ □□ □□ □□□□ □□ □□□□□□□ □□□ AWS □□□ □□ □□(DR) □□□ □□□□ □□□. □ □□□ DR □□□ □□□□ □□□ □□□□□ □□ 30□ □□□ RTO(□□ □□ □□)□ RPO(□□ □□ □□)□ □□□□□. □□□ DR □□□ ALB□ □□□□□□. □□□ □□ □□□ □ □□□ □□□□ □□□□□?

A. CloudFront □□□ □ ALB□ □□□□□ □□□□□. □□□ □□ □□ □□□ □□□□□. AMI□ DR □□□ □□□□□. DR □□□ □□ □□□□ □□□ □□□ 3□ Auto Scaling □□□ □□□□□. DR □□□ □ OpenSearch □□□ □□□□□ □□□□□. □□□□□ □□ □□□□ □ □□□ □□□□□.

B. CloudFront □□□ □ ALB□ □□□□□ □□□□□. □□□ □□ □□ □□□ □□□□□. AMI□ DR □□□ □□□□□. DR □□□ □□ □□□□ □□□ □□□ 0□ □□ □□ □□□ □□□□□. DR □□□ □ OpenSearch □□□ □□□□□ □□□□□. □□□□□ □ □ □□□□ □ □□□ □□□□□.

C. DR □□□ □ CloudFront □□□ □□□□ □ ALB□ □□□□□ □□□□□. □□ □□ □□(failover)□□ Amazon Route 53 DNS□ □ □□□□. AMI□ DR □□□ □□□□□. DR □□□ □□ □□□□ □□□ □□□ 0□ Auto Scaling □□□ □□□□□. OpenSearch □□ □ □□□□□ □□ AZ(Standby □□ □□)□ □□□□□□. □□ □□□ DR □□□ □□□ □□□□□.

D. DR □□□ □ CloudFront □□□ □□□□ □ ALB□ □□□□□ □□□□□. □□ □□ □□(failover)□□ Amazon Route 53 DNS□ □ □□□□. AMI□ DR □□□ □□□□□. DR □□□ □□ □□□□ □□□ □□(3)□ □□ Auto Scaling □□□ □□□□□. OpenSearch □□□ □□□□□ □□ □□□ □□□□ □□ AZ□ □□□□□□. □□ □□□ DR □□□ □□□ □□□□□.

Answer: B (LEAVE A REPLY)

NEW QUESTION: 13

□ □□□ □□□ □□ □□□□□□ □□□ □□□ □□□□ □ □□□□□□□ □□□□ □□□□. □ □□□ □□□ □□ □□ □□□ □ □□□□. □□□ JSON □□□□□. □ □□□□□□□ Amazon CloudWatch Logs □□ □□□ □□□ □□□□□.

□□□ □□□□ □□ □□□ □□□□ □□□□□ □□□ □□□□□.

□□ □□□□ □□ □□□□□ □□□□□□ □ □□ □□□ □□□ □ □□□□?

A. Amazon OpenSearch □□□ □□□□□ OpenSearch □□□ □□ □□□ □□□□ □□ □□ □□□□ □□□□□ □□□□□.

OpenSearch □□□ □□□□□ □□□□ □□ □□ □□ □□□ □□□□□ □□ □□ □□□ □□□□ □□□□□ □□□□□.

B. □□□□ □□ □□□ □□□□ □□ □□□ □□ □□ □□□□ □□ □□□ □□ CloudWatch □□□ □□□ □□□□□. □ □□□□ □□□□ □□□ □□□□ □□□□ CloudWatch □□□□□ □□□□□.

C. AWS Lambda □□□ □□□□ □□ □□□ □□ CloudWatch □□ □□□ □□□□.

Lambda □□□ □□□□ JSON □□□ □□ □□□□ □□ □□□ □□□□□ □□ □□□ □□ □□□ CloudWatch□ □□□□□. □□ □ □□ □□□ □□□□ □□ □□□□ □□□□ CloudWatch □□□□□ □□□□□.

D. □□ □□□ □□□□ Amazon Kinesis □□□ □□□□ □□□□□. □□ □□□ □□□□ □□ □□ □□□□ □□□□□ □□□□ □ □□ AWS Lambda □□□ □□□□□ □□□ □□□□ □□□□□. Lambda □□□ □□□ □□ □□□ CloudWatch□ □□□□□ □□ □□□. □□□ □□ □□□ □□□□ □□ □□□□ □□□□ CloudWatch □□□□□ □□□□□.

Answer: B (LEAVE A REPLY)

NEW QUESTION: 14

□ □□□□ □□ □□□□□□□ AWS CodeDeploy □□□ □□□ □□□□□□. □□ □□□□□□□ Amazon EC2 □□□□□ □□□ □ □□□□. DevOps □□□□□ □□□ □□□ □□□□ □□□ □□ □□□ □□□□ □□□.

□□ □□□□ □□ □□□ □□□□ □ □□□ □□□ □□□ □□□□□?

A. □□□□ □□□□ □□□□□□□ VPC □□ □□□ □□□□□. Amazon Inspector□ □□□□ □□□□ □□□ □□□□ □□ □□□ □□□□□. Amazon Detective□ □□□□ □□□ □□□□□.

B. EC2 □□□□□ □□ □□ □□□□□ □□□□□□. AWS Systems Manager Run Command□ □□□□ □□ EC2 □□□□□□ □□ □ □□ □□ □□□□□ □□□□□. AWS CloudTrail □□□□ □□□ □□□□□.

C. Amazon CloudWatch Logs□ □□□□ □□□□□□ □□□ □□□□□. EC2 □□□□□ /opt/codedeploy-agent/deployment-root/ □□ □□□□ CodeDeploy □□ □□□ □□□□□. AWS X-Ray□ □□□□ □□□□□□ □□ □□□ □□□ □□□□□.

D. CodeDeploy □□□ □□ AWS Trusted Advisor □□ □□□ □□□□□. AWS Health Dashboard□ □□□□ □□□□□□ □□□ □□ □□□□□. Amazon CloudWatch □□□□□□ □□ □□□ □□□□□.

Answer: C (LEAVE A REPLY)
CodeDeploy deployment failures must be diagnosed using deployment-specific logs and related application logs. CodeDeploy writes detailed logs on the target EC2 instances under /opt/codedeploy-agent/deployment- root/ (for example in logs and deployment-logs subdirectories). These logs include information about lifecycle events (BeforeInstall, AfterInstall, ApplicationStart, etc.), script execution failures, permissions issues, and any exit codes returned by deployment hooks.
Option C correctly points to CodeDeploy agent logs on the EC2 instances as the primary source for root cause analysis. Additionally, application-level logs streamed to Amazon CloudWatch Logs help validate whether the application itself is working correctly after deployment. For complex, multi-tier applications, AWS X-Ray can trace requests end to end, helping determine if issues are caused by downstream dependencies instead of the deployment process.
Options A, B, and D focus on network flow, generic performance, or high-level checks, none of which provide sufficient, deployment-level detail for CodeDeploy. Trusted Advisor and AWS Health do not surface step-by-step deployment log data. Therefore, the combination of CodeDeploy logs on the instances, CloudWatch Logs, and (optionally) X-Ray is the correct troubleshooting approach.

NEW QUESTION: 15
□□□ Amazon EC2□ □□ □□□ □□□□□ □□□□□. DevOps □□ □□□ EC2 □□□□□ □□□□ EC2 □□□□□ □□□ □□ □□□□□ □□□□□□ □□□□□ □□□.
□□ □□□□ □□ □□□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □ □□□□□□ AWS □□□ □□□ □□ AWS □□□ □□□ □□□□ □□ □□□ □□□ □□□ □□□ □□□□ □□□□ Amazon S3 □□□ □□□ □□□□□ □□□□□. □ □□□ S3 □□□ □□□ □ □□□□ AWS Lambda □□□ □□□□□. . □□□ □□□□□ □□ □□□□□ Lambda □□□ □□□□□.
B. □ □□□□□□ AWS Systems Manager □□ Systems Manager □□□□ □□ Systems Manager □□□□□ □□ □□□ □□□□□ □ □□□ □□□□□□□ □□□□ AWS Config □□□ □□□□□.
C. □ □□□□□ AWS Systems Manager□ □□□□□. □□□ □□□ □□□□□ □□□□□□. AWS CloudTrail□□ Systems Manager □□□□ □□□□ □□ □□□ □□□□□ □□□ □□□□□□□ □□□□□.
D. Amazon CloudWatch Logs□ □□ □□□□□□ □□□□□ □□ □□□□ □□□□□. □□ □□□□□□ □□□□ □□□□□ □□ □□ □□□ □□□□□□□ □□□□□ □□□□□. □□□ CloudWatch Logs□ □□□□□ □□□□□ □□□□□. □□ □□□ □□□ □ CloudWatch □□ □□ □□□ □□□□□□□ □□□□ □□ □□ □□□□ □□□□□.
Answer: A (LEAVE A REPLY)
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.
References:
AWS Systems Manager Inventory
AWS Systems Manager Resource Data Sync
S3 Event Notifications
AWS Lambda

NEW QUESTION: 16
□□□ Auto Scaling □□□ □□ Amazon EC2 □□□□ □□ Application Load Balancer□ □□□□ □□□□□□□ □□□□□. □□□ □□□□ □□ □□□□□□. Auto Scaling □□□ □□□ □□ □□□ □□□ □□ AMI□ □□□□□. EC2 □□□□□□ □□□ □□ □ □□□□□ □□□□□ □□□□.
Auto Scaling □□□□ □□□□ AMI□ □□ □□□□□□□. AMI ID□ □□□□ □□ □□□ Auto Scaling □□□ □□ □□□ □□□ □ □□□□.
DevOps □□□□□ □□□ □□ □□□ □□□□□ □□ □□ □□□ □□□□ □□□? (3□□ □□□□□.)
A. □ AMI□ □□□□ □ □□ □□□□ □□□□□.
B. □ □□ □□□□ □□□□□ Auto Scaling □□□ □□□□□□□.
C. Auto Scaling □□□ □□□ □□□ O□ □□□□.
D. Auto Scaling □□□ □□□ □□□ I□□ □□□□.
E. Auto Scaling □□□□ □□ □□ EC2 □□□□□□ □ AMI□ □□□□□.
F. EC2 □□□□□ □□□□ □□ □□□ □□ □□□ AMI□ □□□□ □ AMI□ □□□□□.
Answer: A,B,F (LEAVE A REPLY)

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.

NEW QUESTION: 17
DevOps □□□□□ AWS CodeDeploy□ AWS CloudFormation□ □□□□ □□□□□□□ □□ Amazon Elastic Container Service(Amazon ECS) □□/□□ □□□ □□□□ □□□□. □□ □□ □□ □□□□□□□ □□□□□□□ □□ CodeDeploy□ □□ □ □□□ □□□ □□□ □□ □□□□ 10%□ □□□□□□□ □ □□□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ DevOps □□□□□ CloudFormation □□□□ □□ □□□ □□□□ □□□?
A. CodeDeployDefault.ECSLineaMOPercentEverylMinutes □□ □□□ □□□□ AppSpec □□□ □□□□□.
B. CodeDeployDefault.ECSLinear10PercentEvery1 Minutes □□ □□□□ AWS::CodeDeployBlueGreen □□□ AWS::CodeDeploy::BlueGreen □□ □□□□□ □□□□□.
C. ECSCanary10Percent5Minutes □□ □□□ □□□□ AppSpec □□□ □□□□□.
D. ECSCanary10Percent5Minutes □□ □□□□ AWS::CodeDeployBlueGroen □□□ AWS::CodeDeploy::BlueGreen □□ □□□□□ □□□□□.
Answer: (SHOW ANSWER)
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.

NEW QUESTION: 18
□□□ AWS Organizations□ □□□ AWS □□□ □□□ □□□□. AWS Config□ □ AWS □□□□ □□□□ □□□□□. □□□ □□□ □□ □□□ □□ AWS Config□ □□□□ □□□□ □□□□ □□□□ □□□. □ □□□□ □□ □□□ □□ □□□ □□ □□□ □□□ □ □□□.

□□□ □□ □□□ □□□□ □□ DevOps □□□□□ □□□□ □□ □□ □□□ □□□□□? (□ □□□ □□□□□.)
A. AWS Config□ □□ □□□ □□□ □□□ □□□□□. □□□□ AWS Config□ □□ □□□ □ □□ □□□□ □□□□□□.
B. AWS Config□ □□ □□□ □□□ □□□ □□□□□. □□□ □□□ □□□□ AWS Config□ □□ □□□ □□ □□□ □□□□□.
C. AWS CloudFormation □□□□ □□□□ AWS Config □□□□ □□□□□. □□□ □□ □□□ □□□□ □□□□□ CloudFormation □□ □□□ □□□□□.
D. □□□ □□□ □□□□ AWS Config □□ □□□□ □□□□□. □□□ □□ AWS □□ □ □□ AWS □□□□ □□□ □□□ □□□□ □.
E. □□□ □□□ □□□□ AWS Config □□ □□□□ □□□□□. □□□ □□ AWS □□ □ □□ AWS □□□□ □□□ □□□ □□□□ □.
Answer: A,E (LEAVE A REPLY)
https://aws.amazon.com/blogs/mt/org-aggregator-delegated-admin/ https://docs.aws.amazon.com /organizations/latest/userguide/services-that-can-integrate-config.html

NEW QUESTION: 19
□ □□□ □□ AWS □□□ □□□□ □□ □□ □□□ □□□□ AWS Organizations □□□ □□□□ □□□□. □ □□□ □□ □□□ AWS Config□ □□□□□□□. □□□□□ □□ □□ □□ □□□□□□□ □□□ □□□□□ □□ □□□ AWS □□□ AWS CloudFormation □□□ □□□□ □□□.
□□ □□□ □□□□□ □□□□□□□ □□□ Amazon EC2 □□□□ □□□ □□□□□ □□□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. EC2 □□□□ □□ □□□ □□□ □□□□ □□ □□□ □□ □□□ □ SUCCESS□ □□□□ AWS Lambda □□□ □□□□□. □ AWS □□□□ CloudFormation Guard Hook□ □□□□□ Lambda □□□ □□□□□.
B. □ AWS □□□□ desired-instance-type □□□ □□□□ AWS Config □□□ □□□□□. □□ □□□ □□□ □□□□ □□ □□□ □ □□□□. AWS-StopEC2Instance □□ □□□ □□□□ AWS Config □□□ □□ □□ □□□ □□□□□.
C. ec2:InstanceType □□□ □□□ □□□□ □□ □□□ □□ □□□□ □□ □ ec2:RunInstances□ □□ □□ □□□ □□□□ SCP□ □□□□□. □□ SCP□ □□□ □□□ □□□□□.
D. EC2 □□□□ □□□ □□□ □□□□ □□ □□□ □□ □□□□□ □□□□ CloudFormation Guard □□□ □□□□□. □ AWS □□ □□ Guard Hook□ □□□□□ Guard □□□ □□□□□.
Answer: D (LEAVE A REPLY)
The key requirement is to prevent non-approved EC2 instance types from being created at stack creation time, specifically when developers deploy AWS CloudFormation stacks. This is a pre-deployment enforcement requirement, not a detection or remediation requirement after resources already exist.
CloudFormation Guard Hooks are purpose-built for this use case. They allow organizations to define policy- as-code rules that validate CloudFormation templates before resources are created or updated. By writing a CloudFormation Guard rule that explicitly checks the InstanceType property against an approved allow list, the stack operation will fail immediately if a developer attempts to use a disallowed instance type. This enforces compliance early in the deployment lifecycle and avoids post-deployment cleanup.
Option B uses AWS Config, which only detects noncompliance after resources are created. Even with remediation, the instance would still be launched briefly, which violates the requirement. Option C uses an SCP, which applies broadly to all EC2 launches in the organization and is not limited to CloudFormation stacks; this is overly restrictive and could unintentionally block other valid use cases. Option A incorrectly combines Lambda with Guard Hooks-Guard Hooks natively evaluate Guard rules and do not invoke Lambda functions.
Therefore, using a CloudFormation Guard rule with a Guard Hook is the correct and AWS-recommended solution for enforcing approved EC2 instance types during CloudFormation deployments.

NEW QUESTION: 20
□ □□□ Amazon S3 □□□ □□ □□□□ □□□□ □□□□□□□ □□□□ □□□□. □□□□ □□□□□ □ □□ □□□□□□□ □□□□ □□□□□ □□□. □□□□□□□ □□□□ □□□□ □ □□□□ □ □□□□□□□□ □□□□ □□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □ □□□□□□□ □□ S3 □□□ □□□□□. □□ □□□□ S3 □□□□ □ □□□□□□□ S3 □□□□ S3 □□ □□ □□(SRR)□ □□□□□. □□ S3 □□□ □□□□ □□□□□ □ □□□□□□□ □□□□□.
B. Amazon Kinesis □□□ □□□□ □□□□□. □□ □□□□ S3 □□□□ □□ □□ □□□□ □□ □□□□ AWS Lambda □□□ □□ □□□. □ □□□□□□□ □□□□ □□□□□ Lambda □□□ □□□□□□□□. Kinesis □□□ □□□□ □□□□ □□□□□.
Kinesis □□□ □□□□ □□□□ □□□□□ □ □□□□□□□ □□□□□.
C. □ □□□□□□□ □□ □□ □□□□ S3 □□□ □□□□ □□□□ S3 □□□ □□□□ □□□□□. □□ □□□□ S3 □□□□ □□ □□ □□□□ □□ □□□□ AWS Lambda □□□ □□□□□. □ □□□□□□□ □□□□ □□□□□ Lambda □□□ □□□□□□□ □. □ □□□□□□□ S3 □□□ □□□□ □□□□ □□□□□. □□ S3 □□□ □□□□ □□□□ □□□□□ □ □□□□□□□ □□ □□□.
D. □□ □□□□ S3 □□□ □□□□ □□□□ S3 □□□ □□□□ □□□□□. □ □□□□□□□ □□ S3 □□□ □□□□ □□□□ S3 □□ Lambda □□□ □□□□ □□□□□. □□□ □□□ □ □□□□ □□□□□ □ S3 □□ Lambda □□□ □□□□ □□ AWS Lambda □□□ □□□□□. □□ S3 □□ Lambda □□□ □□□□ □□□□ □□□□□ □ □□□□□□□ □□□□□.
Answer: D (LEAVE A REPLY)
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.
1: Amazon S3 Features | Object Lambda | AWS 2: Transforming objects with S3 Object Lambda - Amazon Simple Storage Service

NEW QUESTION: 21
□□□ AWS Organizations □ AWS Control Tower□ □□□□ □□□ □□ AWS □□□ □□□□□. □□□ Enterprise Support □□□ □ □□□□.
DevOps □□□□□ AFT(Account Factory for Terraform)□ □□□□ □ □□□ □□□□□□□□. □ □□□ □□□□□□□ DevOps □ □□□□ □ □□□ □□ □□ □□□ □□ □□ □□□□ □□□□□□ □□□□□. DevOps □□□□□ □□□□□□ □□ □□□□ □ □□□ □□□□□□□ □□□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. AWS Config □□□ □□ □□□□ account-part-of-organizations AWS Config □□□ □□□□ □□□ □□□ □□□□ □□□□□.
B. Enterprise Support □□□ □□□ □□□□ □□ AWS Support□ □□ □□□ □□□□ AWS Lambda □□□ □□□□□. Lambda □ □□ support:ResolveCase □□□ □□□□□.
C. control_tower_parameters □□□ □□ □□ □□□□ AWSEnterpriseSupport □□□□□ □□□ □□□ □□ □□□ □□□□□.
D. AFT □□ □□ □□□□ aft_feature_enterprise_support □□ □□□□ True□ □□□□□. AFT□ □□□□□ □□ □□□ □□□□□.
Answer: (SHOW ANSWER)
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

NEW QUESTION: 22
□□□□ □□ □□□ □□□□□□□ □□□□ □ □□□ □□□ AWS □□□ □□□□. □□□□ □□□□□□□ □□□ AWS CloudTrail □□ □□□ □□ □□ Amazon S3 □□□ □□□ □□□□. □□□ □□ □□□ □□ Amazon Macie□ □□□□□□□. DevOps □□□□□ □□ □□□ □□□□□ □□ □□□ Macie □□□ □□□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. CloudTrail □□□ □□□ S3 □□□ □□ □□□□ □□□□□.
B. □□ □□□□ □□ □□ □□□ □□□ □□ S3 □□□ □□□□□.
C. □□□ □□ S3 □□□ □□ □□□ □□ □□ □□□ □□□□□.
D. □□□□□ □□□ □□□ □□ S3 □□□ □□□□□ □□ □□□ □□□□□.
E. □□□□ □□□□ □□□ □□□ S3 □□□ □□□□□ □□ □□□ □□□□□.
Answer: (SHOW ANSWER)
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.

NEW QUESTION: 23
DevOps □□□□□ □□□□ □□ □□□ □□□ □□ Auto Scaling □□□ Amazon EC2 □□□□□ □□ □□□□ □□□□ □□□□ □ □□.
□□ □□□□ □□□□ □□ □□□□ □□□ □□ □□□ □□□□□?
A. Amazon Route 53□ □□□□ □□ □□□ Amazon CloudWatch Logs□ □□ □□□□□. 5□□□ □□□□ □□□□ □□□ □□□ □ □□□ □□ □□□□ □□□□ AWS Lambda □□□ □□□□□. □□ □□□□ □□□□ □□□ EC2 □□□□□ □□□□□.
B. VPC □□ □□□ □□□□ Amazon S3 □□□□ □□ □□□ □□□□□. 5□□□ □□□□ Amazon Athena □□□ □□□□ □□□ □ □□□ □□□ IP □□□ □□ AWS Lambda □□□ □□□□□. IP □□□ □□□□ □□ EC2 □□□□□ □□□□□.
C. Amazon GuardDuty□ □□□□□□. EC2 □□□□□ □□□□□□ □□ □□□ □□□□□. GuardDuty□ □□□ □□□ □□□□ Amazon EventBridge □□□ □□□□□. EventBridge □□□ □□ □□□□□ AWS Lambda □□□ □□□□□. Lambda □□□ □□□ □ □□ □□□□ □□□ EC2 □□□□□ □□□□□ □□□□□.
D. AWS Security Hub□ □□□□□□. EC2 □□□□□□ □□□ □□□ □□□□□□□. Security Hub□ □□□ □□□ □□□□ Amazon EventBridge □□□ □□□□□. EventBridge □□□ □□ □□□□□ AWS Lambda □□□ □□□□□. Lambda □□□ □□□ □ □□ □□□□ □□□ EC2 □□□□□ □□□□□ □□□□□.
Answer: C (LEAVE A REPLY)
The requirement is to immediately detect and terminate EC2 instances involved in cryptocurrency mining with the least development effort. Amazon GuardDuty is the AWS-native service specifically designed to detect malicious activities such as crypto-mining by continuously analyzing CloudTrail events, VPC Flow Logs, and DNS logs. GuardDuty includes managed threat intelligence and predefined findings like CryptoCurrency:EC2/BitcoinTool.B!DNS and CryptoCurrency:EC2/BitcoinTool.B!IP, which directly identify mining behavior without custom detection logic.

Option C leverages this built-in capability. Once GuardDuty is enabled, findings are automatically generated when mining activity is detected. These findings are sent to Amazon EventBridge in near real time. An EventBridge rule can filter for cryptocurrency-related findings and trigger an AWS Lambda function . The Lambda function can then identify the affected EC2 instance and terminate it or adjust the Auto Scaling group to replace it. This approach requires minimal custom code and no log parsing, scheduled jobs, or analytics pipelines.
Options A and B rely on custom log analysis, periodic execution, and maintaining lists of mining domains or IPs, which significantly increases complexity and response time. Option D uses AWS Security Hub, which aggregates findings from GuardDuty and other services but is not intended for immediate, low-latency remediation.
Therefore, Option C provides the fastest detection, immediate response, and lowest development overhead using AWS-managed threat detection services.

NEW QUESTION: 24
□□ □□ AWS CodeCommit□ □□□□ □□□□□□ □□□ □□□ □□□□ AWS CodePipeline□ □□□□ □□□□□ □□□ □□□ □ □□□□. □□ □□ □□ □□□ □□□□ □□□□□□ □□□□ □□ □□ □□□ □□□□□ □□□□□□. □□□□ CodeCommit □□□□□□ □□ □□ □□□ □□□□□ □□□□□□ 10□ □□□ □□□ □□□ □□□□□□.
□ □□□ □□□□□ □□ □ □□ □□□ □□□ □□□?
A. □□ □□□□ □□□□□□ □□□□□□ Amazon EventBridge □□□ □□□□□□ □□□□□.
B. CodePipeline □□□ □□□ CodeCommit □□□□□□ □□□□ □ □□ □□□ □□□ □□□□□□.
C. □□□□ IAM □□□ CodeCommit □□□□□□ □□□ □ □□ □□□ □□□ □□□□□□.
D. Amazon CloudWatch Logs□ CodeCommit □□□ □□ □□□□□□ □□□□ □□□□ □□□□□.
Answer: (SHOW ANSWER)
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

NEW QUESTION: 25
DevOps □□□□□ AWS Cloud Formation □□□ □□ □□□□ □□□□ AD Connector□ □□□□□□. AWS Lambda □□□ □□□ □ AD □□□□ □□□□□ Cloud Formation□ CREATE_IN_PROGRESS□□ CREATE_COMPLETE□ □□□□ □□□□.
□□□□□ □ □□□ □□□□ □□ □□ □□□ □□□ □□□?
A. Lambda □□ □□□ □□□□□ □□□□□□ □□□□□□.
B. Lambda □□ IAM □□□ AWS □□□ □□ ds ConnectDirectory □□□ □□□ □□□□□.
C. Lambda □□ IAM □□□ □□ ARN□ □□ cloudformation UpdateStack □□□ □□□ □□□□□.
D. Lambda □□ □□□ □□ □□□ URL□ □□ □□□ □□□□□ □□□□□.
Answer: (SHOW ANSWER)

NEW QUESTION: 26
□□□□ AWS □□□ □□□, □ □□□□ □□ AWS □□□ □□□ □□ OU□ □□□□.
□□ □□□□ □□ □□ □□□ AWS □□□ □□□□.
□□□□□ AWS CloudFormation □□□□ □□□□□ □□□□□□ □□ Amazon Elastic Block Store(Amazon EBS) □□□ Amazon Simple Queue Service(Amazon SQS) □□□□ □□ □□ □ □□□□ □□□□ □□ □ OU□ □□□ □□□□□.
□ OU □□□□ CloudFormation □□ □□□ □□□□ □□ □□ □□□□ □ □□□ □□□□□?
A. CloudFormation StackSets□ □□ □□□ □ □□ □□□□ □□□□□□. EBS □□□ SQS □□□□ □□ □ □□□□ □□□□ CloudFormation Hook□ □□□□□. StackSets□ □□□□ OU □ □□ □□□ Hook□ □□□□□.
B. OU □ □□ □□□ AWS Config□ □□□□□. AWS Systems Manager□ □□□□ OU □ □□ □□□□ EBS □□ □ SQS □□□□ □□ □□ □ □□□□ □□□□ AWS Config □□□ □□□□□.
C. EBS □□□ SQS □□ □□ □ □□□□ □□□□ □□□ EBS □□□ SQS □ □□□ □□□□ SCP□ □□□□□. SCP□ OU□ □□ □□□.
D. □□□ □□□ □□□□ EBS □□ □ SQS □□□□ □□ □ □□□□ □□□□□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□□ OU □ □□□ □□ □□□ □□□ □□□□ IAM □□□ □□□□□.
Answer: A (LEAVE A REPLY)
The requirement specifies enforcing encryption before CloudFormation creates or updates resources. This is key because preventive enforcement must occur during the provisioning workflow, not after resources already exist. AWS provides CloudFormation Hooks specifically for this purpose. A Hook allows an organization to intercept a CloudFormation stack operation and validate resource configurations before provisioning occurs.
This feature is recommended by AWS for pre-deployment governance such as enforcing encryption policies, tag compliance, or security restrictions.

By enabling trusted access between CloudFormation StackSets and AWS Organizations, the Hook can be deployed centrally from the delegated administrator account across all accounts in the specified OU. Any attempt to create or update EBS volumes or SQS queues through CloudFormation is validated first by the Hook. If encryption is not configured, the operation fails immediately.
Option C (SCP) blocks API calls globally, but SCPs cannot perform conditional logic based on resource properties passed by CloudFormation prior to creation. Option B (AWS Config) detects violations after resources already exist, which does not satisfy "before stack operation." Option D (Lambda remediation) also occurs after the resource is created.
Thus, CloudFormation Hooks distributed via StackSets provide the only solution that enforces compliance before the provisioning lifecycle

NEW QUESTION: 27
DevOps □□□□□ Amazon EKS□□ □□□□ □□□ □□□□ □□ □□ □□□ □□□□ □□□□.
□□ □□□□□ □□□□ □□□ □□□□ □□□□□□. DevOps □□□□□ □□□ □□□□ □□□□ □□ □□ □□□ □□□□□.
□□ □□□□ □ □□ □□□ □□□ □ □□□□?
A. □□ EKS □□□ □□□□ □□□□ CloudWatch □□□□□ □□□□□.
B. EKS □□□□□ □□ AWS CloudTrail□ □□□□□.
C. EKS □□□□□ □□ CloudTrail Insights□ □□□□□.
D. CloudWatch Observability □□ □□□ □□□□□ EKS □□□□□ □□ Amazon CloudWatch Container Insights□ □□□□□.
Answer: D (LEAVE A REPLY)
Container restart failures require detailed observability into pod-level, node-level, and container-level metrics and logs. CloudWatch Container Insights is purpose-built for Kubernetes operational diagnostics and provides granular visibility into CPU, memory, network I/O, disk I/O, container restarts, OOM kills, throttling, pod lifecycle issues, and Kubernetes control plane behaviors.
The CloudWatch Observability add-on deploys Fluent Bit and the CloudWatch Agent directly into the EKS cluster as DaemonSets. These components automatically collect:
Container logs
Pod metrics
Node metrics
Cluster events
OOM errors
CrashLoopBackOff restart cycles
Control plane request anomalies
With this data, the DevOps engineer can easily identify misconfigurations, resource bottlenecks, unhealthy nodes, failing containers, or image pull issues.
Option A (dashboards only) lacks per-container diagnostic data.
Option B (CloudTrail) only logs API calls - not useful for restart debugging.
Option C (CloudTrail Insights) only detects anomalous API usage, not container failures.
Therefore, CloudWatch Container Insights is the correct and AWS-recommended solution for diagnosing container restart failures in EKS.

NEW QUESTION: 28
□□ □□ □□□ AWS □□□□ □□□ □□ □□□ □□□□ □□ AWS CloudTrail□ □□□□□. DevOps □□□□□ AWS □□□□ □ □□ CloudTrail□ □□□□ □□□□ □□□□ □□□□□.
CloudTrail □□ □□□ □□ □□□□ □□□□□ □□□□ □□□□ □□□□□?

A. CloudTrail StopLogging □□□□ □□ Amazon EventBridge □□□ □□□□□. AWS Lambda(AWS SDK□ □□□□ StopLogging□ □□□ □□□□ ARN□□ StartLogging□ □□□□ □□)□ □□□□□. Lambda □□ ARN□ EventBridge □□□ □□ □□□□ □□□ □□.
B. 1□□□ □□□□ □□□□ AWS □□□ CloudTrail □□ AWS Config □□ □□□ □□□□□. AWS Config □□ □□ □□□ □□ Amazon EventBridge □□□ □□□□□. AWS SDK□ □□□□ StopLoggmg□ □□□ □□□□ ARN□□ StartLogging□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□ ARN□ EventBridge □□□ □□□□ □□□□□.
C. 5□□□ □□□ □□□□ □□ Amazon EventBridge □□□ □□□□□. AWS SDK□ □□□□ AWS □□□ CloudTrail □□□□□ StartLogging□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□ ARN□ EventBridge □□□ □□□□ □□□□□.
D. AWS SDK□ □□□□ □□ □□□□ CloudTrail□ □□□□ □□□□□ 5□□□ □□□□ t2 nano □□□□□ □□□□□. CloudTrail □□□ □□□□□ □□ □□□□□□ □□□ □□ □□□□□□□.
Answer: A (LEAVE A REPLY)
https://aws.amazon.com/blogs/mt/monitor-changes-and-auto-enable-logging-in-aws-cloudtrail/

NEW QUESTION: 29
□ SaaS □□□ ALB(□□□□□□ □□ □□□) □□ ECS(Fargate)□ □□□□, CodePipeline□ CodeDeploy□ □□□□ □□/□□ □□ □ □□□□□. □□□ □□□□ □□ □□ □□□ □□ □□□□ □□□□ □□□□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. AllAtOnce □□ □□□ □□□□□.
B. appspec.yaml□□ □□□□ □□□ □□□□ TimeBasedLinear□ □□□□□□.
C. TimeBasedCanary□ □□□□□.
D. ALB□□ □□□ □□□□ □□□□ □□□□□.
Answer: B (LEAVE A REPLY)

NEW QUESTION: 30
□ □□□ □□ Amazon EC2 □□□□□ □□□□ Amazon Elastic Kubernetes Service(Amazon EKS) □□□□□ □□□□□□□ □□□ □□□□□□□. □ □□□ CPU □□□□ □□ □□□□ □□□□□ □□□□□□□ □□□□□□.
□ □□□□□□□ □□□ □□ □ □□□ □□□ □□□□□. □□ □ □□□□□□□ □□□ □□□ □□□ □□ □□□ □□□□ □□□ □. □ □□□ □□□ □□□ □□ □□□□□□□ □□□ □□□□ □□□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (□ □□□ □□□□□□.)
A. □□□□□ □□□□ 1AM □□□□ □□□□ Cloud WatchAgentServer □□□□ □□□□ 1AM □□□ □□□□□.
B. Cloud WatchAgentServer □□ □□ 1AM □□□ □□□□□ □□□ □□ □□□ □□□□□.
C. □□□□□ □□ EC2 □□□□□ □□ Amazon CloudWatch □□□□□ □□□□ □□ □□□ □□□□□. □□□□□ □□□ □□ □ EC2 □□□□□ □□ AMI□ □□□□□ □□□□□.
D. AWS Distro for OpenTelemetry □□□□ DaemonSet□□ □□□□ □□ □□□ □□□□□.
E. Service □□□ □□□□ ContainerInsights □□□□□□□ pod_memory_utilization Amazon CloudWatch □□□□ □□□□□.
F. ClusterName □□□ □□□□ ContainerInsights □□□□□□□ node_memory_utilization Amazon CloudWatch □□□□ □□□□□.
Answer: (SHOW ANSWER)
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 InstancesTo 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 InsightsAfter 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.

NEW QUESTION: 31
□□□ Amazon EC2 □□□□□□ □□□□□□□ □□□□□ □□ □□ AWS □□□ □□□□□. □□□ AWS □□□□ AWS Config□ □□ limited-ssh AWS Config □□ □□□ □□□□□□□.
□□□ □□□ □□ □□□ □□□ SSH □□□ □□□□ □□ □□ □□□□□ □□□ □□□ □□□□ □□□□ □□□□ □□□□ □□ □□□. □□□ □□ □□□□ □□□ □□ □□□ □□□ ID□ □□□□□ □□□.
DevOps □□□□□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□ □□□ □□□□ □□□□□.
DevOps □□□□□ □□□ □□ □□□ □□□□ □□ □□□ □□□ □□ □□□?
A. limited-ssh □□□ □□ NON_COMPLIANT□ AWS Config □□ □□□ □□□□ Amazon EventBridge □□□ □□□□□. EventBridge □□□ □□ □□ □□□ □□ SNS □□□ □□□ □□□□□ EventBridge □□□ □□□□□.
B. limited-ssh □□□ □□ □□ □□ □□□ SNS □□□ □□□□ AWS Config□ □□□□□. □□□ NON_COMPLIANT □□□□ □□ □ □□□ □□□□□ □□□□ SNS □□□ □□ □□ □□□ □□□□□.
C. limited-ssh □□□ □□ NON_COMPLIANT□ AWS Config □□ □□□ □□□□ Amazon EventBridge □□ □□ SNS □□□□ AWS Systems Manager Run Command□ □□□□ □□□ □□□ □□□□ □□□ SNS □□
D. NON_COMPLIANT□ □□ AWS Config □□ □□□ □□□□ Amazon EventBridge □□□ □□□□□. limited-ssh □□□ □□ □□ □□□□ □□□□□. SNS □□□ □□□ □□□□□ EventBridge □□□ □□□□□.
Answer: A (LEAVE A REPLY)
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.

NEW QUESTION: 32
□ □□□□ API□ □□ □□ □□□ □□□□ AWS Lambda □□□ □□□□□□. □ □□□ AWS CodeDeploy□ □□□□ CI/CD □□□ □□□ □□ □□□ Lambda □□□ □□□□ □□□□.
DevOps □□□□□ □□ □ □ □ □□ □□ API□ □□□□ □□□ □□□ □□□□□□. □□ □□ □ DevOps □□□□□ □□□ Lambda □□□ □□□□ □□ □□□ □□□□ □□ □□□□□□ □□□□ □□ □□□□ □□□□□. DevOps □□□□□ □□ □□□ □□□□ □□□?
A. □□□□ □ □□□ Lambda □□□ □□□ □□ □□□ □□□□□□ □□□ □□□□□ □□□□ AppSpec □□□ BeforeAllowTraffic □□□ □□□□□.
B. AppSpec □□□ AfterAllowTraffic □□□ □□□□ □ □□□ Lambda □□□ □□□□□ □□□□ □□ □□□□ □□ □□ □□□□ □□ □□□ □□□□□ □□□.
C. □ □□□ Lambda □□□ □□□□ □□ □□□ □□□□□□ □□□ □□□□□ □□□□ AppSpec □□□ BeforeAllowTraffic □□□ □□□□□.
D. □□ □□□□ □□□□ □□□□□□□ □□ □□ □□□□ □□ □□□□ □□ □□ □□□□□ □□□□ validateService □□□ AppSpec □□□ □□□□□.
Answer: A (LEAVE A REPLY)
https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure-hooks.html#appspec-hooks-lambda

NEW QUESTION: 33
□ □□□ □□□□□□ □□ □□□□□□ □□□ □□ AWS CodeCommit □□□□□□ □□□□□. □ □□□□□□ □□□ □□ AWS □□□ □□□□□□ □□□□ □□□, □□ □□□ AWS Organizations □□ □□□ □□ □□□□.
□ □□□□□□ □□ □□ IdP(ID □□□)□ □□□ AWS IAM Identity Center(AWS Single Sign-On)□ □□□□ □□□ IAM □□□ □□ □□□. □ □□□ □□□ □□ □□□□□□ □□ Git□ □□□□ □□□□□□ □□□ □□□ □ □□□□.
□□ □□ □□ □□□□□□ □□ □□ □□□□ □□ □□□□ □□□ □ □□ □□□ □□□□□□. DevOps □□□□□ □□□□□□ □□ □□□ □□□□ □□□□ □□ □□□□ □□□ □ □□□ □□ □□□□ □□□□ □□□.
□□ □□□□□ □□□□ □□ □□□ □□□□□? (□ □□□ □□□□□□.)
A. SAML □□□□ □□□□□□ □□□□ □ □□□ □□□□□ □□□. IAM □□□ □□ □□□ □□□□□□ □ □□□ □□□ access-team □□ □□□ □□□□□.

B. □□ □□ □□□ □ □□ □□ □□ □□ □□□□ □□□□□. □□□□ □□ □□□□ □□□□□. □□□ □□ ARN□ □□□□ □□ □□□.
C. □ □□□ □□ □□ □□ □□□□ □□□□□. □□ □□□□ □□□□ □□□□□. □□□□□.
" aws:ResourceTag/access-team " : " $;{aws:PrincipalTag/access-team} " □□ □□ □□□□ □□ □□□□□.
D. □ CodeCommit □□□□ □□ □□□ □□ □□□□ □□ □□□ access-team □□□ □□□□□.
E. □□□ SCP□ □□□□□. □□ □□□ □□□□□.
F. □ □□□ IAM □□ □□□ □□□□□. □□ □□□ □□□□□.
Answer: A,D,F (LEAVE A REPLY)
Short Explanation: 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.
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 membership1.
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 identity-based 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.

NEW QUESTION: 34
DevOps □□□□□ □□□□ □□ □□ □□□□ □□ □□□ □□ □□□□ □□□ □□□□ □□□□. □□□□□ □□□ □ □□ AWS □□□ □□□□, □□□□ □□□ □□□□ □□□ □□□□ □□ □□□ □ □□ □□□ □□□□□ □□ □□□. □□(US) □□□ □□ □ □□□□□ □□□□ □□□□ □□□□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. □□ □□ □□□ □□ □□□ □□□□ □□ □□□□ □□□□ AWS Organizations SCP□ □□□□□. □□□ □□□ □□□ □□□ □□.
B. □□□ Amazon CloudWatch Logs□ □□□□ □□ □□□□ □□□□□□ AWS CloudTrail□ □□□□□. CloudWatch Logs □□ □□ □ □□□□ □□ □□ □□□ □□ □□□ □□□ □□ □□□ □□□□.

C. AWS □□□ □□□ □□□□ □□ □□ □□□ □□□□ AWS Lambda □□□ □□□□□. □□□ Lambda □□□ □□□□ □□ □□ □□□□ □□□ □□□□ □□□ □□□ Amazon EventBridge □□□ □□□□□.
D. AWS Lambda □□□ □□□□ Amazon Inspector□ □□□□ □□ □ □□□ □□□ □□□ □□ □□□ □□□□ □□□ □□□□.
E. □□ □□□ □□ □□□□ □□□□ aws: RequestedRegion □□ □□ □□□□ SCP□ □□□□□. □□ □□□, □□, □□□ □□ □ □

Answer: A,B (LEAVE A REPLY)

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.

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.

NEW QUESTION: 35

DevOps □□ AWS CodePipeline, AWS CodeBuild □ AWS CodeDeploy□ □□□□ □□□□□□□ □□□□□. □□□□□□□ AWS Lambda □□□ Amazon API Gateway□ □□□□ REST API□□□. □□ □□□ □□ □□ □□□□ □□□ □□□ □□□ □□□□□□. DevOps □□□ □□□ □□□□ □□□□□□□ □□ □□□ □□□□ □□□□ □□□□ □□□□ □□□□□. □□□□ □□□ □ □□ □□ □□□□ □□□ □□□ □□□.

□□ □□□□ □□ □□□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?

A. CodeDeploy□□ LambdaAllAtOnce□ □□ □□□ □□□□□. □□ □□□□ □□ □□□ □□□□□. API □□□□□□□ HTTP □ □□ □□□□□ □□□ □□□□ Amazon CloudWatch □□□ □□□□□. □□ □□ □□ □□□□ □□□ □ □□□□□ □□ □□□ □ □□□□.

B. CodeDeploy□ □□ □□□ LambdaCanary10Percent10Minutes□ □□□□□. □□ □□□□ □□ □□ □□ API □□□□□□□ HTTP □□□ □□□□□ □□□ □□□□ Amazon CloudWatch □□ □□ □□ □□ □□ □□□□ □□□ □ □□□□□ □□ □□ □□

C. CodeDeploy□□ □□ □□□ LambdaAllAtOnce□ □□□□□. □□ □□□ □□ □□ □□□ □□□□□. □□□□ □□ □□□ □□□ □□□□ Amazon Simple □□ □□□(Amazon SNS) Topc□ □□□□□. □□ □□□ □□□□ □□ □□□ □□□ □□□ □□□□ □□ □ Lambda □□□ □□□□□ SNS topc□ □□□□□.

D. CodeDeploy□□ □□ □□□ LambdaCanaryIOPercentIOMinutes□ □□□□□. □□ □□□□ □□ □□□ □□□□□. API Gateway □ Amazon CloudWatch □□ □□□ □□ □□□ □□□□ HTTP □□□ □□□□□ □□□ □□□□□□□. □□ □□□ □□□□ □□ □□□ □□□ □□□ □□□□ □□□ Lambda □□□ □□□□□ □□ □□□ □□□□□.

Answer: B (LEAVE A REPLY)

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

NEW QUESTION: 36
□ □□□ CloudTrail □□ □□□ □□□ □□□□ AWS Organizations□ □□□□ □□□□. □□ □□ □ □□□ □□□□ □□ □□□ □ □ □ □□□□□ □□, □□□ □□ □□ □ □□□ □□□ □□□□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. CloudTrail □□□ □□ □□□ S3□ □□□□□. □□□ □□ □□□□ □□ EventBridge □□□ □□□□ SNS□ □□ □□□ □□□ □.
B. □□ □□□ □□□ □□□□ Athena + Lambda□ □□□□ 5□□□ □□□□□.
C. □□ □□□ S3□ □□□□ □□ □□□ CloudWatch □□ □□□ □□□□ □□□ □□ □ □□□ □□□ □□□□□.
SNS.
D. Kinesis # Flink # SNS□ □□□□.
Answer: A (LEAVE A REPLY)
Using an organization trail with logs centralized in the audit account's S3 bucket ensures compliance and isolation. An EventBridge rule in the audit account triggers on failed login events (ConsoleLogin failed) and sends SNS notifications in near real time.

NEW QUESTION: 37
□ □□□ □□□ □□ □□□ □□□ □□ □□□ □□□□□ □□□. □□□ DevOps □□ □□ □□□□ □□ □□□□□□ □□□□ □ □□. □□, □□□□ □□□ □□□ □□□□ □□□. □□□□ □□ □□ □□□□□□ □□□□ □□, □□ □□□□□□ □□ □□□□ □□ □□□□ □□□. □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □ AWS CodeArtifact □□□□ □□□□□. □□□□□ □□ □□ □□□□ □□□ □□□□ □□□□ AWS Lambda □□□ □□□□ □. □□□ □□□ □□□ □□□□ □□ □□□ □□□□ □□□□□ AWS CodeDeploy□ □□□□□.

B. AWS CodeArtifact □□□□ □□□□ □□□□ □□□□ □□□□□. □□ □□□□□ □□□□ □□□ □□□□ AWS CodeBuild □□ □□□ □□□□□. CodeArtifact □□□□ □□ □ □□□□ □□□□□ □□□□ □□□□□ □□□□□ □□□□□.
C. AWS CodeDeploy□ □□□□ □□□ □□□□□ □□ □□□ Amazon Elastic File System(Amazon EFS) □□ □□□□ □□□□□□.
D. AWS CodeBuild □□□□□ □□□□ □□□ □□□□ □□□ □□□□□ □□□□ □□□ AWS Artifact□ □□□□□. CodeBuild □ □□□□□ □□ □□□ □□□□□.
Answer: B (LEAVE A REPLY)

NEW QUESTION: 38
DevOps □□□□□ □□□ REST API□ □□□□ □□□□□□□ □□□□ □□□□. □□□□□□□□ Amazon Kinesis □□□ □□□ □□ □□□□ □□ AWS Lambda □□□ □□□□. Lambda □□□ □□□□ □□□ REST API□ □□□□.
Lambda □□□ Kinesis □□□ □□□□□ □□□ □□□□ □ 10%□□ □□□ □□□ □□□ □□□□ □□□□ □□□. Lambda □□ □□□ □□ □□□□ □□ □ □□□□ Amazon Simple Queue Service(Amazon SQS) □□ □□ □□ □□□□ □□□□. DevOps □□□ □□ □□□□ □□□ □□□□□ Lambda □□□ □□□□ □□ □ □□□□ □□□□□□.
□□□ □□ DevOps □□□□□ □□ □ □ □□ □□□□ □□□ □□□ □□ □□□ REST API□□ □□ □□□ □□□□ □□ □□□□ □□□ □□□□□. DevOps □□□□□ □□ □ □ □□ □□□□ □□□□ □□ □□ □□□ □□ □□□ □□ Lambda □□□ □□□ □ □ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □□□ □□ □□
B. □□ □□ □ □□□ □□□□□ □□ □□
C. □□□ □□ □□ □□
D. □□ □□ □□ □□
Answer: B (LEAVE A REPLY)
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

NEW QUESTION: 39
□□□ AWS Organizations□ □□□□ □□ □□□ □□□□□. □□ □□ □□□ □□ □□□□□ □□ □□ Amazon EBS □□□ □□□ □ □□□□□ □□□. DevOps □□□□□ □□□□ □□□□ □□□□ □ □□ □□ □□□ □□ □□□□□ □□□□ □□□.
□□ □□□□ □□ □□□□□?
A. EBS □□□□ □□□□□□□ □□□□ □□ AWS Inspector □□□ □□□□ AWS CloudFormation □□□□ □□□□□. □□ □ □ □ □□□ □□□ Amazon S3 □□□ □□□□ □□□□□. Amazon S3□ CloudFormation □□□□ □□□□ □□ □□ □□□□□ □□ □□□□□.
B. AWS Config □□ □□□ □□□□ EBS □□□□ □□□□□□□ □□□□ AWS CLI□ □□□□ □□□ □□□□□. □□ □□□□ AWS Config □□ □ □□□ □□□□ SCP□ □□□□ □□□□□.
C. □□□□ SCP□ □□□□□. □□□□ □□□□ EBS □□□□ □□□ □□ Amazon EC2 □□□□□ □□□□ □□□□ □□□ □□ □□□. SCP□ □□ AWS □□□ □□□□□. Amazon Athena□ □□□□ AWS CloudTrail □□□ □□□□ ec2:RunInstances □□□ □ □□□ □□□□ □□□□.

D. □□□ □ □□ □□ □□□ □□ □□□ IAM □□□ □□□□□. IAM □□□ □□ AWS Lambda□ □□□ □□ AWS CodePipeline□□ □□□□□□ □□□□ □□□ □□ EBS □□□ □□□□□. Amazon S3□ □□□□ □□□□□.

Answer: (SHOW ANSWER)

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

NEW QUESTION: 40

□□□ □□ □□□□□□□ □□ □□ □□□ □□□□ □□□. □□□□□□□□ AWS □□□ Amazon CloudFront □□□ □□□ ALB(Application Load Balancer)□ □□□□ □□□□. □□□ ALB□ □□□ □□ □□□□ □□□□□□.

□□ □□ □□□□□□ □□ □□ □□□ 0□ RTO□ □□□□. □□□ □ □□ □□□□ □□ □□□ □□□□□□□ □□□□□.

DevOps □□□□□ HTTP GET □□□ □□□ RTO□ □□□□□ □□ □□□ □□ □□□□□□ □□ □□□ □□□□□ □□□.

□□ □□□□ □□□ □□ □□□ □□□□□?

A. □□ ALB□ □□ □□□□□ □□□□ □ □□ CloudFront □□□ □□□□□. □ CloudFront □□ □□□ □□ □□ □□ □□ □ □□ □□ □□□ □□ □□□ Amazon Route 53 □□ □□□□ □□□□□. □ □□□ □□□ □□□□□ □□□□□□□ □□□□□□□.

B. □□ ALB□ □□ □□□□ □ □□□□ □□□□□. □ □□ □□□ □□□□. □□ ALB□ □□ □□□□ □□□□□. HTTP 5xx □□ □□□ □□ □□ □□□ □□□□□ □□ □□□ □□□□□.

□□ □□□ □□□□□ □□ □□□ □□□□□□□.

C. □ ALB□ □□ □□ □□ □□ □ □□ □□ □□□ □□ □□□ Amazon Route 53 □□ □□□□ □□□□□. □ □□□□ TTL□ 0□□ □□□□□. □ □□□ □□□ □□□□□ □□ □□□ □□□□□□□.

D. HTTP 5xx □□ □□□ □□□□ CloudFront □□□ □□□□□. □□□ 5xx □□ □□□ □□□□ □□ □□ ALB□ 307 □□ □□□□ □□ □□□ □□□□□ □□□ □□□□□. □□ □□□ □□□ □□□□ □□□ □□ □□□ □□□□□□□.

Answer: B (LEAVE A REPLY)

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 failure1. 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.

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

NEW QUESTION: 41

DevOps □□□□□ AWS Lambda □□□ □□□□ □□□□ □□□□□□□ □□ □□□□ □□ □□□□□□ □□□□ □□□□. □□ □ □□□ □□□ □□□□ □□□ □□□ □□□□ □□□. □□□ □□ □□□ □□□□□□□ □□□□.

□□□ □□ □□□ □□□□ □□ □□ □□□ □□□□□?

A. AWS Serverless Application Model(AWS SAM) □□□□ □□□□ □□□□ □□□□□□□ □□□□□. AWS CodeDeploy□ □□□ □ Canary10Percent15Minutes □□ □□ □□ □□□□ Lambda □□□ □□□□□. Amazon CloudWatch □□□ □□□□ □□□ □□□ □□□□□□□□.

B. AWS CloudFormation□ □□□□ □ □□ □□□□□ □□□□ □□ □□□□ Amazon CloudWatch □□□ □□□□□. □□□□ AWS CloudFormation □□ □□□ □□□□ □□□□□ AWS CodePipeline □□ □□□ □□□□□.

C. AWS CloudFormation□ □□□□ □□ □□ □□□□□ □ □□□ □□□□ □□ □□□□ Amazon CloudWatch □□□ □□□□□. AWS::Lambda::Alias □□□□ RoutingConfig □□□ □□□□ □□ □□□□ □□ □□□ □□□□ □□□□□□□.

D. AWS CodeBuild□ □□□□ □□□□ □□ □□□ □□□□□ Lambda □□□ □□□□□. □□□ □ □□□ □□□□ Amazon CloudWatch □□□ □□□□□. □ □□□ □□□□□ □□□□ □□□ □□□□□□□. □□□ ALARM □□□ □ □□□□□ □□□ □ □□□□.

Answer: D (LEAVE A REPLY)

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.

NEW QUESTION: 42

□ □□□ AWS Organizations□ □□□□ □□□ □□□ □□ □□ □□□ □□□□ □□□□ □□□ □□□□□ □□□.
AWS □□□ □□□ □□ □□□ □□ □□□ □ □□□□ □□□□□.
AWS □□□□ □□ □□□ □□ □□ □□□ □□□ □□□□ □□□□□.
□□□ Active Directory□ □□ □□□□□.
□□ □□□ □□□□ □□□□ □□ □□□□ □□□□□.
□□ □□□□ □□□ □□ □□□ □□□□□?

A. □□ □□□ □□ □□□ □□□ □□ □□(OU)□ □□□□ □□ □ □□□ □□□□ □□□□□. AWS CloudFormation StackSet□ □ □□□ □ □□□ □□ □□□ □□ □□□ □□□□□□□, □ □□□ IAM ID □□□ □□□ □□ IAM □□ □□□ □□□□□.

B. □□ □□□ □□ □□□ □□□□ □□ □ □□□ □□□□ □□□□□. AWS CloudFormation StackSet□ □□□□ □ □□□ □□ □ □□ □□ □□□ □□□□□□□, □ □□□□ IAM ID □□□ □□□ □□ IAM □□ □□□ □□□□□.

C. □□ □□□ □□□ □□ □□□ □□□□ □□ □ □□□ □□□□ □□□□□. AWS Resource Access Manager(AWS RAM)□ □□□ □ □ □□□ □□ □□□ □□ □□ □□ □□□ □□□□, □ □□□ □□□ □□ AWS IAM Identity Center□ □□□□□.

D. □□ □□□ □□□ □□ □□(SCP)□ □□□□ □□ □ □□□ □□□□ □□□□□. AWS CloudFormation StackSet□ □□□□ □ □ □□ □□ □□□ □□ □□□ □□□□□□□, □ □□□□ IAM ID □□□ □□□ □□ IAM □□ □□□ □□□□□.

Answer: D (LEAVE A REPLY)
This is a classic AWS Organizations governance design:
* Restrict Regions and allowed services across all accounts
* The correct Organizations mechanism for account-wide guardrails is a Service Control Policy (SCP) .
* SCPs set the maximum permissions that accounts/OUs can use, making them the right tool to enforce "only these Regions" and "only these services" consistently across the org.
* Authentication from Active Directory + consistent job-function roles in every account
* With AD as the identity source, the standard approach is federation to AWS using an IAM identity provider (SAML) and role assumption based on job function.
* AWS CloudFormation StackSets is the operationally efficient way to deploy identical IAM roles /policies into multiple accounts/OUs so job-function permissions are consistent everywhere.
* The roles' trust policies can be configured to allow federated identities (from the AD-backed IdP) to assume the correct job role in each account.
Why the other options don't fit:
* A "OU with group policies" isn't an AWS Organizations control mechanism for restricting Regions /services. The AWS-native control for this is SCPs , not "group policies."
* B Permissions boundaries are useful to limit what an IAM principal/role can do within an account , but they are not the org-wide enforcement mechanism for "all accounts must stay in these Regions / use only these services." That's what SCPs do.
* C AWS RAM is for sharing resources (like subnets, Transit Gateway, etc.), not for "sharing roles" across accounts as a governance baseline. Also, the question emphasizes identical permissions in each account, which is best achieved by provisioning roles in each account (StackSets), not attempting to "share" roles.

NEW QUESTION: 43
DevOps □□□□□ AWS Lambda □□□ □□□□ Amazon Aurora MySQL DB □□□□□ □□□□ □□□□□□□ □□□□ □□□□.
Lambda □□□ □□ □□□ □□□□□. Amazon EventBridge □□□□ Lambda □□□ □□□□□.
□ □□ □□□□ □□ Lambda □□□ □□□□□ □□□□□□□ □□ □□□ □□□□ □□□□□□□ □□□□ □□□□□. DevOps □□□□□ □□□□□□□ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (3□□ □□□□□.)
A. Amazon RDS Proxy□ □□□□ □□□□ □□□□□. □□□□ Aurora □□□□ □□ □□□□□□ □□□□□. □□□□□ □□ □□ □□□ □□□□□.
B. Lambda □□ □□ □□□□□□ □□ □□□ □□□□□. □□□□□□ □□ □□ □□ □□ □□ □□□□□.
C. Lambda □□□ □□□ □□ □□□□ □□□□□□ □□ □□□ □□□□□.
D. Lambda □□□ □□□ □□ □□□ □□□□□□ □□ □□ □ □□□ □□□□□.
E. Lambda □□□□ □□□ □□□□□□ □□□□□.
F. Lambda □□□□ Aurora □□□□ □□□□□□ □□□□□.
Answer: A,C,E (LEAVE A REPLY)
Short Explanation: 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. 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.

NEW QUESTION: 44
DevOps □□□□□ ALB(Application Load Balancer) □□ Amazon EC2 □□□□□□ □□□□ □ □□□□□□□ □□□□□. □□□□ □ □□ □□ □□□ □□ EC2 Auto Scaling □□□□ □□□□□. □□□□□ □□□ □□ □□ □□□ □□□□ □□□.
□□ □□□ □□□ □□□□ □□□□□ □ □□ □□□ □□□□□.
□ □□ □□□ □□□□ □□ □□ □□□ □□□□ □□ □□□□□.
□ □□ □□□ □□□ □□□□ □□□□ □ □□ □□□□ □□□□□.
□□ □ 1□□□ □□□ □□ □□□ □□□□ □□□□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. ALB□ □□ □□ □□□ 1□□□□ □□□ AWS CloudFormation □□□□ □□□□□. □ ALB□ □□□□□ Amazon Route 53 □□□ □ □□□□□□□.
B. □ □□ AWS Elastic Beanstalk □□□ □□□□ □□ □□□□ □ □□□□ □□/□□ □□□ □□□□□. 1□□ □□ □□ □□□ □□ □□ □□□□□□ □□ □□ □□ □□□ □□□□.
C. □□/□□ □□ □□□□ □□□ □□ □□□ □□ AWS CodeDeploy □□ 1□□□ □□ □□□□ □□ □□□□ □□ □□□□ □□ □ □□ □□□□□.
D. □□ □□□□□ □□□ □□□□ AWS Elastic Beanstalk□ □□□□□. ALB□ □□ □□□ 1□□□□ □□□□ □□□ □□ □□□ □ .ebextension□ □□□□ □□□□□□□ □□□□□.
Answer: C (LEAVE A REPLY)
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

NEW QUESTION: 45
□ □□□ AWS CodePipeline□ □□□□□□ □□□□ AWS CloudFormation □□□□ Amazon S3 □□□ □□□□□□. □ □□□□□ □ □□ □□□□ □□□□ □□□ □□□ □□□□ CloudFormation □□□ □□□□□.
□□ □□□ □□□□ □□ □□□□ □□□□□ □ □ □□□ □□□□□□. □□□ □□□ □□□□ □□ □□□ □□ □□□ □□ □□ □□□ □□□□ □□ □□□ □□ □□□ □□□ □ □□ □□□ □□□ □□□.
□□ □□□□ □□ □□□ □□□□□□ □□□ □□ □□□ □□□□□?
A. AWS CodeConnections□□ Git □□□□□□ □□ □□□ □□□□□. □□□□ Git □□□□□□ □□□□□. □□□ □□ □□□ □ □□□ □ □□□□ □□□□□ □□□□□. □□□ □□ AWS CloudFormation Git □□□□ □□□□□.
B. □□ □□ □□ □□□ □□ □□ □□□ □□□□ □□ □□□□□□ □□ □□ □□□ □□□□□.
C. □□ □□ □□ □□□ □□ □□□ □□□□ □□ AWS Lambda □□□ □□□□□ □□□□□□ □□□□□□□.
D. AWS CodeConnections□□ Git □□□□□□ □□ □□□ □□□□□. □□□□ Git □□□□□□ □□□□□. □□□ □□□□ □□ □□□ □□□□□□ □□□□□ □□□□□. □□ □□ □□ □□□ □□ □□□ □□□□ □□ □□□□□□ □□ □□ □□□ □□□□ □.
Answer: (SHOW ANSWER)
The requirement is simply: review changes before production deployment , with the least operational overhead .
* B is the lightest change: adding a Manual approval (review) action in CodePipeline creates a controlled gate before the deploy stage. It requires no new repositories, no new services, and no custom code-just pipeline configuration.
Why not the others:
* A introduces additional moving parts (Git repo integration, PR workflow management, and CloudFormation Git sync). That's useful, but it's more operational overhead than necessary to satisfy
"review before deploy."
* C requires custom Lambda logic to inspect templates and decide whether to proceed-more code to write, run, secure, and maintain.
* D adds both Git integration and a manual approval step-again more overhead than just adding the approval gate.
So B best meets the requirement with the least operational effort: a simple manual approval stage in the pipeline before production deployment.

NEW QUESTION: 46
□ □□□ □□ □□□□□ □□ □□ □□ □□□ □□□□ □□□ □□□□□ □□□. □□□□ Amazon EC2 □□□□□ □□□ □ □□ □ □□□ Amazon RDS □□□□□ □□□ □ □□□ □□□. □□□□□□ □□□□ Amazon RDS □□□□□ □□□ □ □□□ □□□.
□□ □□□ □□□ IAM □□□□ □□□ □□□□. □□□ □ □□□□□□ □□□ □□□ □□ □□ IAM □□□ □□ □□□□ □□□□.
□□ AWS □□□□□ □□□ □□□□ □□□ □□ □□□□ □□□□. □□ IAM □□□□□□ □□□ □□□□ □ □□ □□□ □□□□ □□□□.
□□□ □ □□□ □□□ □□□□ □□□□□□ □□□ □ □□□ □□□□ □□□.
□□ □□ □□□ □□□□ □□□□ □□□□□? (□ □□□ □□□□□□.)
A. □ □□□□□ □□ □□□□ IAM □□ □□□ □□□□□□ □□□□ IAM □□ □□□ □□□□□. □□ □□□□ □ □□□ □□ IAM □□□ □□□ □□□□□.
B. □□ IAM □□□ □□□□ □□□□□ ResourceTag□ PrincipalTag □□ □□□□ StringEquals □□□ □□□□□. □□□ □□□ □ □□□ IAM □□□ □□□□□.
C. □□□□ □□ □ □□ □□□ □□ ResourceTag □□ PrincipalTag □□ □□□□ □□□□ □□□□ □□□ □□□ □ □□□ □□ IAM □□□ □□□□□. □ □□□ □□ IAM □□□□□ □□□□□.

D. □□□□ □□□ □□ □□□ □□ ResourceTag □□ PrincipalTag □□ □□□□ □□□□ □□□□ □□□ □□□ □ □□□ □□ IAM □□□ □□□□□. □ □□□ □ □□□ IAM □□□ □□□□□.
E. □□ IAM □□□ □□□ □□□□ □ □□ □□□ □□□□□. □□□ □□□ □ □□ □□□ □□ IAM □□□ □□□□□.
F. □ □□□□□ □□ □□□□ IAM □□ □□□ □□□□□□ □□□□ IAM □□ □□□ □□□□□. □□□□ □□□ IAM □□□ □□ □ □ □□□ □□ □□□□ □ □□□ □□□□□.
Answer: A,B,C (LEAVE A REPLY)
The requirements describe a classic attribute-based access control (ABAC) model using AWS IAM. The company already has consistent tagging across users and resources, which is ideal for ABAC and least- privilege enforcement at scale.
First, Option A is required because IAM roles should represent job functions (developer vs. database administrator) and be scoped per project. Creating separate roles per project and job function allows permissions to be isolated cleanly and prevents cross-project access.
Second, Option B enforces project-level isolation at the policy level . By modifying the existing job-function policies to include a StringEquals condition that compares aws:ResourceTag/Project with aws:PrincipalTag
/Project, access is automatically limited to only those resources that belong to the same project as the user.
This is an AWS-recommended ABAC pattern and avoids policy sprawl.
Third, Option C is required so users can assume roles only when their tags match the role's tags . An IAM policy attached to users that allows sts:AssumeRole only if both project and job function tags match ensures users cannot assume roles outside their assigned scope.
Options D, E, and F either misapply policies to the wrong entities, misuse tagging on policies (which IAM does not evaluate for authorization), or introduce unnecessary group-based management that conflicts with the ABAC design.
Therefore, A, B, and C together provide least privilege, project isolation, and scalable access control with minimal ongoing administration.

NEW QUESTION: 47
□ □□□ Amazon Elastic Kubernetes Service(Amazon EKS)□□ □□□□□□□ □□□□□□□ □□□□ □□□□. □□ □□□□□ □ □ □□ □□□ □□□□ □ □□□ □□□ □□□□□ □□□□□, □□ □□□□ □□ □□□□ □□□ □□□□□□ □□□.
DevOps □□□□□ Amazon CloudWatch □□□ □□□ □□□□ □□□ □□□□□□. □□□ EKS □□□ CPU □ □□□ □□□□ □ □□□ □□□□□□. DevOps □□□□□ □□□□□□□ □□□□ □□□ □□□ □□□ □□□ □□□□ □□□□□.
DevOps □□□□□ □□□ □□□□ □□□ □□□ □□□□ □□ □□□□□□□ □□□□ □□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. AWS X-Ray □□□ EKS □□□□□ DaemonSet□□ □□□□□. X-Ray SDK□ □□□□ □□□□□□ □□□ □□□□□. □□□□ □□□ □□ □□□□□.
B. EKS □□□□□ □□ CloudWatch Container Insights□ □□□□□□. Container Insights □□□□ □□□□ □□□ □□□□□.
C. □□ CloudWatch □□□ □□□□ □□□ □□□□□. □□□ □□□ □□□□ □□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□□□.
D. □□□□ □□□ □□ □□□□□□ □□□ □□ □□ □□□ □□ □□□ □□□ □□□ □ □□ □□□ □□□□□.
Answer: A (LEAVE A REPLY)

AWS X-Ray provides distributed tracing for microservice-based applications. Deploying the X-Ray daemon as a DaemonSet in the EKS cluster and instrumenting the application with the X-Ray SDK enables end-to-end tracing across microservices, helping identify performance bottlenecks. This method is documented in "Using AWS X-Ray with Amazon EKS" (AWS Observability Guide).

NEW QUESTION: 48
□□□□ □□□□ □□□□□□□ □□ □□ □ □□ □□□ □□ □□□ □□□□□. □□□□□□□ MySQL □□□□□□□ Amazon EC2 □□□□□ □□□□□. □□□ □□ □□□ □ □□□□□□□ □□ □□ 2□□□ RPO□ □□ 10□□ RTO□ □□□□□.
□□□ □□ □□□ □□□□ □□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. □□ AWS □□□ Amazon Aurora □□ AZ □□□□□ □□□ □□□□ □□□□□. □□ □□ □ Aurora□ □□ □□ □□□ □□□□ □.
B. □ AWS □□□ Amazon Aurora □□□ □□□□□□□ □□□ □□□□ □□□□□. □□□ □□□□ □□ □□ □□□ □□□□□□ □ □□ □□□□ □□□□□. □□ □□□□ Aurora □□□□ □□□□□□ □□□□□ □□□□□□□ □□□□□□□.
C. □□ AWS □□□ Amazon Aurora □□□□□ □□□ □□□□ □□□□□. Network Load Balancer□ □□□□ □□ □□□ □□□□ □□ □□□ □□□ □□□□□.
D. □ □□ AWS □□□ □□□□□□□ □□□□□. □ □□ □□□□ Application Load Balancer□ □□□□ Amazon Route 53 □□ □□ □□□□ □□□□□. □ □□□□ □□ □□ □ Auto Scaling □□□ □□□□□.
E. □ □□ AWS □□□ □□□□□□□ □□□□□. □ □□ □□□□ Application Load Balancer(ALB)□ □□□□□ AWS Global Accelerator□ □□□□□. □ ALB□ □□ □□□□□ □□□ □□□□□. □ □□□□ □□ □□ □ Auto Scaling □□□ □□□□□.
Answer: (SHOW ANSWER)
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 application1. 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.

NEW QUESTION: 49
Auto Scaling □□□ Amazon EC2 □□□□ □□□□ □□□□ □□□□□□□ □□□□□ □□ □□□ □□□□□. □□□□□ AWS CloudFormation□□ □□ □ □□ □□□□□. DevOps □□□□□ □□□□□ □□□ □ □□ □□ □□□ □□□ □□□ □□ □□□ □ □ □□ □□□ CloudFormation □□□□ □□□□□ □ □□□□ □□□□ □□ □□□□□ □□□□□ □□□□. □□ □□□ □□ □□ □□□□ □□ □□□ □□ □□□□ AWS □□□ □□ □□□ □□ □□ □□□□□ □□□.
□□ □□□□ □□ □□□□□?
A. CloudFormaiion □□□□□ AWS Config □□□ □□□□□. □□□ InputParameters □□□ □□ □□ □□□□ □□□□ Scope □□ □ EC2 Auto Scaling □□□□ □□□□□. □□□□ AWS Systems Manager □□□ □□□ □□□ □□□□ □□□□ □□ □□□□□ □□□□□.
B. CloudFormation □□□□□ EC2 □□ □□□ □□□□ □□□□□. □□ □□□□ □□ □□ □□□□ □□□□□. □□□□□ □□□ □ □□□□□ cfn-mit □□□□□ □□□□ □□ □□□□□ □□□□□ cfn-hup □□□□□ □□□□□.
C. CloudFormation □□□□□ EC2 □□ □□□ □□□□ □□□□□. □□ □□□□ □□ □□ □□□□ □□□□□. □□□□ AWS Systems Manager □□□ □□□ □□□ □□□□ □□□□ □□ □□□□□ □□□□□.
D. CloudFormation □□□□□ CloudFormation imt □□□□□□ □□□□□. □□ □□ □□□□ □□□□□□ □□□□□. □□□□□ □□□ □ □□□□□ cfn-init □□□□□ □□□□ □□ □□□□□ □□□□□ cfn-hup □□□□□ □□□□□.
Answer: D (LEAVE A REPLY)
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

NEW QUESTION: 50
□ □□□ □ □□□□□□□ □□ □□□ CI/CD □□□□□□ AWS CodeBuild□ □□□□ □□□□. □ □□□ □□ □□ □□□□□□ □□ □□□□□ □□□□ □□□. □□ □ □□□□□ □□ □□□□ □□□□□□ □□□ □ □□□ □□□. □□□□□ □□ □□ □ □ □□ □□ Amazon S3 □□□ □□□□□ □□□. □□□ □□□□ □□□□ □□□ Git □□□□ □□□□ □□□. □ □□□ □□ □□□ □□□□□ □□□. □□□ □□ □□□ □□ □□□□□ □□□□ □□□□ □□□□□?
A. buildspec.yml □□□ □□□□ □□ □□ □□ □□□ □□ □□ □□□□□ □□□□□. □□ □□ □□□ □□□□ □□ □□□□□ □□ □□□ □□□□□□. □□ □□□ □□□□ □□ ID□ □□□□ □□□□□ □□□ □□□□ □□□□□.
B. □ □□□□ □□□□ □□□ CodeBuild □□□□□ □□□□□. Docker □□ □□ □□□ □□□□ □□ □□□□□ □□ □□□ □□ □□□□. □ □□□□□ □□ ID□ □□□□ □□ □□ □□□ □□□□ □□ □□□□□ S3 □□□ □□□□□ □□□□□. AWS Step Functions□ □□□□ □□□□□ □□□ □□□□□.
C. CodeBuild□ □□□□ □□ □□□ □□□□ □□ ZIP □□□□□ □□□□□. □□ □□□□□ S3 □□□ □□□□□□. AWS CodePipeline□□ □□□ □□ □□□ □□□□ □□□ □□□□ S3 □□□ □□ □□□□□ □□□□□□. □□ □□□ □□□□ □□ □ □□ □□□□ □□□□□ □□□ □□ □□□ □□□□□.
D. □□ □□□ □□ □□□□□ □□□□□ buildspec.yml □□□ □□□□□. □□□ □□ □□ □□□ □□□□ □□ □□□□□ □□ □ □□ □□□□□□. CodeBuild □□ □ □□□□ AWS Lambda □□□ □□□□□. Lambda □□□ □□□□□□□ □□□□□ □□ □□ □ □□□□ □□ ID□ □□□□ □□ □□ □□□ □□□□ S3 □□□ □□□□□□.

Answer: A (LEAVE A REPLY)

NEW QUESTION: 51
□ □□□ □□□ AWS □□ □□□ □□ □□□ □□ Amazon Elastic Kubernetes Service(Amazon EKS) □□□□□□ □□□□□□□ □ □□□□. □ □□□ Auto Scaling □□□ □□□□ □ EKS □□□□□ □□ □□□ □□ □□ □□□ □□□□□. □ EKS □□□□ □□ □□□□ □□□□ □□□□ □□ Application Load Balancer(ALB)□ □□□□□.
□□ □□□ □□ □□□□ □□□ □□□□□□ □□□□□□□ □□□□□ □□□. □□ □□□□ □□ □□□ □□ □□ □□□ □□□ □□□□ □□□□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □ □□□□□□□ □ □□ EKS □□□□□ □□□□□. □ ALB□ □□ □□ □□□ □□□ Amazon Route 53 □□□□ □□□□□. □ □ □□ □□□ □□□ □□□□□. Kubernetes □□ □□ □ □□ □□ □□□□ □□□□□.
B. □ □□□□□□□ □□ □□□ EKS □□□□□ □□□□□. □□ □□ ALB□ □□ □□ □□□ □□□ Amazon Route 53 □□□□ □ □□□□. □□□ □□□ □□□ □□□□□.
C. □ □□□□□□□ □ EKS □□□□ □□□ □□□□□. □ ALB □□ □□□□ □□□□ □□□□ □□□ □□□ □□□ □□□□ Amazon Route 53 □□□□ □□□□□. Kubernetes □□ □□ □ □□ □□ □□□□ □□□□□.
D. □□ □□□ EKS □□□□□ □ □□□□□□□ □□□□□. □□ □□ ALB□ □□ □□ □□□ □□□ Amazon Route 53 □□□□ □ □□□□. □□ □□ □□□ □□□ □□□□□.
Answer: C (LEAVE A REPLY)
The requirement is to deploy a stateless application with multi-Region fault tolerance, ensuring high availability even if an entire AWS Region becomes unavailable. For this design, traffic must be actively served from both Regions, not only during a failure event.
Option C correctly implements an active-active, multi-Region architecture. By deploying the application to both EKS clusters, each Region is capable of serving traffic independently. Using Amazon Route 53 weighted routing, traffic is distributed across both Application Load Balancers, allowing both Regions to handle requests simultaneously. If one Region becomes unhealthy, Route 53 health checks can stop routing traffic to that Region, maintaining availability.
Implementing Kubernetes readiness and liveness probes ensures that traffic is only sent to healthy pods within each cluster. This provides fault tolerance at both the container level (pod health) and the Regional level (Route 53 routing).
Option A uses a failover routing policy, which results in an active-passive design. While fault tolerant, it does not utilize both Regions simultaneously and provides slower recovery during Region failure. Options B and D deploy the application only in the primary Region, which does not meet multi-Region fault tolerance requirements.
Therefore, Option C delivers the most resilient, highly available, and AWS-recommended architecture for a stateless, multi-Region EKS application.

NEW QUESTION: 52
□ □□□ Amazon CloudFront □□□ □□□□ □□ □□□□ □□□□ □□□□. □ □□□ IP □□ □□□ □□□ □□ □□□ □□□□ □□□ □□□ □ □□□ □□ □□□. AWS WAF □ ACL□ □□□ □□□□ □□□, □□ □□□ 'Count'□ □□□□ □□□□.
□□ □□□□ □□ □□□ □□□□□□ □□□ □□ □□□ □□□ □ □□□□?
A. □ □□□ □□ □□□ □□□□□. □□□ □□ □□□ □ □□ □□□ □□□□□. □□ □□□ '□□'□□ □□□ □ □ ACL□ □□□ □□. □ ACL□ CloudFront □□□ □□□□□. □ □□ □□□ □□ □□□□ □□□□ □□□ □□□□□.
B. □□ □□□ IP □□ □□□ □□□□ AWS WAF IP □□ □□□ □□□□□. □□ □□□ □□□□ □□□ □ □ ACL□ □□□□□. □ ACL□ CloudFront □□□ □□□□□. IP □□ □□□□ □□□□ □□□□ □□□□ □□□ □□□□□.

C. □ □□□ □□ □□□ □□□□. □□□ □□ □□□ □ □□ □□□ □□□□□. □□ □ ACL□ □□ □□□ □□□□ □□□□□. □ □□ □□ □□□ □□ □□□□ □□□□ □□□□ 0 □□□ □□□□□.
D. □□ □□□ IP □□ □□□ □□□□ WAF IP □□ □□□ □□□□□. □□ □ ACL□ □□ □□□ '□□'□□ □□□□□. IP □□ □□ □□ □□□□ □□□□ □□□□ □□□□ 0 □□□ □□□□□.
Answer: D (LEAVE A REPLY)
To restrict access to CloudFront to a specific IP address range:
* Create an AWS WAF IP address set with the corporate office IPs.
* Modify the existing WebACL ' s default action to Block (deny all except explicitly allowed).
* Add a high-priority rule that allows traffic from the IP address set (the corporate IPs). This way, only requests from the corporate IPs are allowed; all others are blocked. Regex pattern sets are not necessary for IP-based restrictions and add complexity. Setting default action to Allow with exceptions is less secure and more complex to manage.
References:
AWS WAF IP Set Examples
Restricting Access by IP Address

NEW QUESTION: 53
□□□ □□□ AWS □□□ □□ □□ Amazon EC2 □□□□□ □□□ □□□□. □□□ □ □□□□□□ □ □□□ □□□□□ □□ Slack□ □□ □□□ □□ □□□□ □□□□ □□□□. DevOps □□□□□ □□ AWS □□ EC2 □□ □□ □□□ Slack □□ □ □□ □ □ □□□□□ □□□ □□□. □□□□□ □□□□□ □□ □ □□□ □□□ □□□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. AWS Trusted Advisor□ AWS Config□ □□□□□. □□□ □□ AWS Config □□□ □□□□ AWS Lambda □□□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□ □□□□□. Slack □□ □□□□□ □ □□ □□ □□□□ □□□ □□□□□.
B. Amazon EventBridge□ □□□□ AWS □□ □□□□ □□□□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□ □□ □□ □□ □□□□ □□□□□. SNS □□□ □□ AWS Lambda □□□ □□□□ Slack □□ □ □□ □□□□ □□□ □□□□.
C. EC2 □□ □□ □□□ Slack □□ □ □□ □□ □□□□□ □□□ AWS Lambda □□□ □□□□□. Amazon CloudWatch □□□ □□ □□ EC2 □□ □□□□ □□□□□□□. □□ □□ □□□ □□□□ Lambda □□□ □□□□ CloudWatch □□□ □□□□□.
D. AWS CloudTrail□ AWS □□ □□ □□ CloudTrail □□ □□□□ □□□□ EC2 □□ □□ □□□ Amazon Simple Notification Service(Amazon SNS)□ □□□□ □□ AWS Lambda □□□ □□□□□. Slack □□ □ □□ □□ □□□□ □□□□ □□□□□ Amazon SNS□ □□□□□.
Answer: B (LEAVE A REPLY)
https://docs.aws.amazon.com/health/latest/ug/cloudwatch-events-health.html

NEW QUESTION: 54
□□ □□ □□ □□□ □□□□ □ □□□ Amazon Linux □□ □□□ □□□□ Amazon EC2 □□□□□ □□□□ □□□□. □ □□□ EC2 □□□□□ □□ □□□□ □□ □□□ □□□□ □□ □□ □□□ □□□□□□ □□□ □□□□□ □□□□ □□□ □□□□ □□ □. □ □□□ □□□□□□ □□□ □□□□ □□ □□□ □□□□ □□□□□.
DevOps □□□□□ □□ □□ □□□ □□□□□□ □□□ □□□ □□□□□ □□□.
DevOps □□□□□ □□ □□ □□ □□ □□□□ □□□ □□ □□ □□□□ □□ □□□□□ □□□.
□□ □□□□ □□ □□ □□□□ □□□ □□ □□□ □□□ □ □□□□?

A. AWS Systems Manager□ □□□□ □□ □□ □□ □□□□□□ □□□ □□ □□□□□□ □□□□ □□□ □□□ □□ □□ □□□ □□□□□. Run □□□ □□□□ AWS-RunPatchBaseline □□□ □□□□ □□□ □□ □ □□□□□. BaselineOverride API□ □□□□ □□□ □□□ □□ □□ □□□ □□□□□.
B. AWS Direct Connect□ □□□□ □□□ □□ □□□□□□ EC2 □□□□□ □□□□□. Amazon EventBridge □□□□ □□□□ □□ □ □□□□□.
C. yum-config-manager □□□ □□□□ □□□ □□ □□□□ /etc/yum.repos.d □□□ □□□□□. yum-config-manager-enable □□□ □□□□ □ □□□□ □□□□□□.
D. AWS Systems Manager□ □□□□ □□ □□ □□ □□□□ □□ □□ □□□□ □□□ □□ □□□□ □□ □ □□ □□ □□□□ □ □□□□. Run □□□ □□□□ AWS-RunPatchBaseline □□□ □□□□ □□□ □□□□ □□□□□. BaselineOverride API□ □□□□ □□ □□ □□□□ □□□ □□ □□ □□□ □□□□□.
Answer: C (LEAVE A REPLY)
The operating system (Amazon Linux) uses yum for package management. To use both the default and a custom repository, the simplest method is to add the custom repository configuration file under /etc/yum.repos.
d/ using the yum-config-manager CLI, and enable it. This allows the system to access both repositories during patching automatically.
This approach requires minimal setup and effort and leverages the existing yum patching mechanisms.
Option A and D involve managing multiple patch baselines in Systems Manager Patch Manager, which does not natively support custom repositories - it relies on the underlying OS package manager.
Option B (Direct Connect) is not necessary unless network connectivity is an issue.
Thus, option C meets the requirement with the least effort.
Reference:
Managing Custom Repositories on Amazon Linux: " To add a custom yum repository, add a repo file in /etc /yum.repos.d/ and enable it using yum-config-manager. " (Amazon Linux Repository Management) AWS Systems Manager Patch Manager: " Patch Manager uses the OS package manager, so custom repositories must be configured at the OS level. " (AWS Patch Manager)

NEW QUESTION: 55
□□□ Amazon EC2□□ □□□□□ □□□ □□ □ □□□□□□□ □□□□ □□□□□ □□□□□□□□□□ □□□. □ □□□ □□ EC2 □□□□□ □□□ Auto Scaling □□□ □□□ Elastic Load Balancer□ □□□ □□ Amazon Route 53 □□ □□□ □□□ □□□□ □□□□ □□□□□□ □□ □□□ □□□□□□□ □□□□□.
□□□ □□□□ □□□□□□□ □□ □□□ Amazon API Gateway □ AWS Lambd□ □□□ □□□□□. □□□ □ □□□□□□□ □□ □□□□□ □□ □□□□□ □□□□□□ □□□. □□ □□ □□□ □□□ □□□ □□□□ □□ □□□ □□□□ □□ □□□ □□□ □ □□□ □ □□ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□?
A. AWS CDK□ □□□□ API Gateway □ Lambda □□□ □□□□□. □□□ □□□□ □□ □□ AWS CloudFormation □□□ □□□□ □□ □ □□□ API □ Lambda □□□ □□□□□. □□□□ □□□ □□□□ Route 53 □□ □□ □□□ □□□ □□□□□□.
B. AWS CloudFormation□ □□□□ Lambda □□ □□□ □□□□ API Gateway □ Lambda □□□ □□□□□. □□□ □□□□ □□ □ □ □ Lambda □□□ CloudFormation □□□ □□□□□□ □□□□ □□□ □□□ □□□□ API □□□ □□□□□□□. □□□□ □□ □□ □ □□□ □□□□□□.
C. AWS Elastic Beanstalk□ □□□□ API Gateway □ Lambda □□□ □□□□□. □□□ □□□□ □□ □□ □ □□□ API □ Lambda □□□ □□□□□. Elastic Beanstalk □□/□□ □□□ □□□□ □□□□□ □□□□ □□□□□.

D. AWS OpsWorks□ □□□□ □□□ □□□ API Gateway□ □□□□ □□□ □□ □□□ Lambda □□□ □□□□□. □□□ □□□□ □□ □□ OpsWorks□ □□□□ □□/□□ □□□ □□□□ □□□□ □□□□□ □□□□□.

Answer: B (LEAVE A REPLY)

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

NEW QUESTION: 56

□□□□□ □□□□□ □□ AWS CodeDeploy□ □□□□ □ □□□ □□□□□□□ □□□□ □□□□. □□ □□ □ □□□ □□□ □ □ □□□□□ □□ □□□□ □□ □□□□ □□□ □□□□□ □□□.

□□ □□□□ □□ □□□ □□ □□□□□□□ □□□□ □□□□□. □□ □□□ □□□□ □□□ □□□□ □□ □□□ □□□□□.

□□ □□□□ □□□ □□ □□□ □□□□□□□?

A. □□ □□□□ □□□□□□ □□□ □□□ □□□□□.

B. AWS CloudFormation□ □□□□ □□ □□□ □□□□□□□.

C. Amazon CloudWatch□□ □□□ □□□□ □□□ □□□□□.

D. AWS CodeDeploy□□ □□ □□ □□□ □□□□□.

Answer: (SHOW ANSWER)

AWS CodeDeploy supports automatic rollback capabilities, which allow the deployment to automatically revert to the last known good revision if a failure occurs during the deployment or if health checks fail.

* You can enable automatic rollback on CodeDeploy deployments by configuring rollback settings in the deployment group.

* Automatic rollback triggers can include failed deployments and alarms failing during the deployment lifecycle, which means if the health check fails, CodeDeploy will automatically roll back without manual intervention.

* Lifecycle event hooks (Option A) provide custom scripting capabilities during deployment but do not provide built-in rollback automation.

* CloudFormation (Option B) can deploy and manage infrastructure but does not control CodeDeploy rollback.

* CloudWatch alarms (Option C) can trigger notifications but require manual or additional automation to trigger rollback. Hence, enabling CodeDeploy's automatic rollback settings is the simplest and most reliable method for automatic rollback upon health check failure.

Reference from AWS Official Documentation and Study Guide:

* AWS CodeDeploy Automatic Rollback: " You can configure automatic rollback in AWS CodeDeploy to roll back a deployment if it fails or if specified CloudWatch alarms are triggered. " (AWS CodeDeploy User Guide - Automatic Rollbacks)

* AWS Certified DevOps Engineer Professional Study Guide: " Leverage CodeDeploy automatic rollback features to reduce downtime and ensure application stability during deployments. "

NEW QUESTION: 57

□ □□□□ □□□ □□□□□□□ □□ □□□□. □□□□□□□ □□□ □□□ AWS Lambda □□□ □□□□□. □□□ Lambda □□ □ □□□□□ □□□□ □□□ □□□□ □□□. □□□ □□□□ □□ □□□ □□□□ □□□.

□□□ DevOps □□ □ □□□□ □□ IaC(Infrastructure as Code)□ CI/CD □□□□□□ □□□□ □□□.

□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (3□□ □□□□□.)

A. □□□□□□□ AWS CloudFormation □□□□ □□□□□. AWS::Lambda::Function □□□ □□□ □□□□ □□□□□ □ Lambda □□□ □□□□□. □□□□ AWS::Lambda::Version □□□ □□□ □□□□ Lambda □□ □□□ □□□□□. CodeSha256 □□□ □□ □□□. □□ □□□ Lambda □□□ □□□□ AWS::Lambda::Alias □□□□ □□□□□.

B. □□□□□□□ AWS Serverless Application Model(AWS SAM) □□□□ □□□□□. AWS::Serverless::Function □□□ □□□ □□□ □ □□□□□ □ Lambda □□□ □□□□□. □ □□□ □□ AutoPublishAlias □□ □ □□Preference □□□ □□ □□□ □□□□□. □ □ □□ □□□ LambdaCanary10Percent10Minutes□ □□□□□.

C. AWS CodeCommit □□□□□□ □□□□□. AWS CodePipeline □□□□□□ □□□□□. □□□□□□ □□□□ □ □□ □□□□ CodeCommit □□□□□□ □□□□□. AWS Serverless Application Model(AWS SAM) □□□□ □□□□ □□ AWS CodeBuild □□□ □□ □□□□□. □□□□ □□ □□□ CodeCommit □□□□□□ □□□□□□. CodeCommit □□□□□□□ SAM □□□□□□□ □ □□□ □□□□ □□ □□□ □□□ buildspec.yml □□□ □□□□□.
D. AWS CodeCommit □□□□□□ □□□□□. AWS CodePipeline □□□□□□ □□□□□. □□□□□□ □□□□ □ □□ □□□□ CodeCommit □□□□□□ □□□□□. Canary10Percent10Minutes□ □□ □□ □□ □□□ □□□□ □□□□ □□□□□ □□□ AWS CodeDeploy □□ □□□ □□□□□. AWS CloudFormation □□□□ □□ □□□ CodeCommit □□□□□□ □□□□□□. CodeCommit □□□□□□□ CloudFormation □□□□ □□□□ □□ □□□ □□□ appspec.yml □□□ □□□□□.
E. □□ Lambda □□□ □□ Amazon CloudWatch □□ □□□ □□□□□. Lambda□ □□ □□ □□□ □□□ □□□□□. □□□ □□ □□□ □□□□ □□□ □□ ALARM □□□ □□□□□ □□□ □□□□□.
F. □ Lambda □□□ □□ Amazon CloudWatch □□□ □□□□□. □□□ □□□□ ALARM □□□ □□□□□ □□□ □□□□□. □□ □□□□ □□ □□, □ Lambda □□ □ □□□ □□, □□□□□□□ AWS/Lambda□ □□□□□.
Answer: B,C,F (LEAVE A REPLY)
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 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.

NEW QUESTION: 58
□ □□□ CI/CD □□□□□□ □□□□ ap-southeast-2 □□□ □□□□□ □□□□□. □ □□□ □□□□ □□ □□□(NLB)□ □□ □ □□□ □□□□ Amazon ECS □□□□□ AWS Fargate □□□□ □□□□ □□□□□. Amazon ECR □□□□ □□□□ Docker □□□

□ □□□□□. □ □□□ DNS□ □□ Route 53□ □□□□□. □□□□ S3 □□□ □□□□ □□□□□□ DynamoDB□ □□□□□. □ □□□ □□ □□□□ □□□□ □□ eu-west-2 □□□□ □□□□□ □□□.
□□ □□ □□ □□□□ □□□ □□ □□□ □□□ □ □□ □□ □□□ □□□□□? (3□ □□)
A. □□□□□ eu-west-2□ ECR □□□ □□□□□. eu-west-2□ □□ ECS □□□□□ Fargate □□□□ □□□□ NLB□ □□□□□. Route 53□ □ □□□□□ □□ □□ □□ □□□ □□□ □□□□□.
B. DynamoDB □□□□ eu-west-2□ □□□□ □□ □□□ □□□□ □□□□□. Fargate □□□ ap-southeast-2□ DynamoDB □□□□ □□ □□□□□ □□□□□.
C. DynamoDB □□□□ eu-west-2□ □□□□ □□ □□□ □□□□ □□□□□. Fargate □□□ □□□□ □□□ □□□ □□□ DynamoDB □□□□ □□ □□□□□ □□□□□.
D. eu-west-2□ □ S3 □□□ □□□□□. ap-southeast-2□ S3 □□□ eu-west-2□ S3 □□ □□ □□□ □□□ □□□□□. Fargate □□ □ □□□□ □□□ □□□ □□□ S3 □□□ □□□□ S3 PUT □ GET □□□ □□□□□ □□□□□.
E. ap-southeast-2□ S3 □□□ eu-west-2□ □ S3 □□□ □□ S3 □□ □□ □□□ □□□□ □□□□□. S3 □□□ □□□ □□□ □□ □□□. S3 PUT □ GET □□□ □□ □□ □□□ □□□□ □□□□□ □□□□□ □□□□□.
F. □ □□ □□□ ECR □□□□ □□□□□ CI/CD □□□□□□ □□□□□. eu-west-2□ □□ ECS □□□□□ Fargate □□□□ □□ □□ NLB□ □□□□□. Route 53□ □ □□□□□ □□ □□ □□ □□□ □□□ □□□□□.
Answer: A,C,D (LEAVE A REPLY)
* (A) UseECR replicationto keep images synchronized between Regions, minimizing CI/CD pipeline changes.
* (C) DynamoDB global tables allow multi-Region replication and provide local read/write access, so tasks should interact with the DynamoDB replica in their Region.
* (D) Use S3 cross-Region replication or separate buckets with replication; tasks access the bucket in the same Region for latency and data sovereignty.
* (B) Using DynamoDB global table but pointing tasks to only one Region reduces resilience.
* (E) S3 Multi-Region Access Points are newer but add complexity and are not required for minimal changes.
* (F) Managing CI/CD pipeline for multiple Regions and failover routing adds complexity beyond minimal changes.
References:
Amazon ECR Cross-Region Replication
DynamoDB Global Tables
Amazon S3 Cross-Region Replication

NEW QUESTION: 59
□□□□ AWS Organizations□ □□□ □□□ □□ □□ □□□ □□□□. □□ □□ □□ Amazon EC2 □□ □□□ □□ □□□□ □ □ □□□□ □□□ □□□□ □□□. □□ □□ □□□ □□□ □□□□ AWS Lambda □□□ □□□□ □□ □□□□ □□□□□ □□□□ □ □□□ □□□□□ □□□.
□□□ □□ □□□ □□□□ □□□ □□ □□□ □□□□□? (3□□ □□□□□.)
A. □□ □□□ □□□□ □□□ □□ IAM □□□ □□ □ □□□ □□ □□□ □□□□□.
B. □□□ □□□ □□□□ □□ □□□ IAM □□□ □□ □ □□□ □□ □□□ □□□□□.
C. AmazonEC2ReadOnlyAccess □□□ □□□ □□ □□□ □□□ □□ □ □□ □□□□ IAM □□□ □□□□□.
D. □ □□ □□□□ IAM □□□ □□□□ □□□ □□ IAM □□□ ARN□ □□ sts:AssumeRole □□□ □□□□□.
E. □□ □□ IAM □□□ ARN□ □□ sts:AssumeRole □□□ □□□□ □□□ □□□□ I AM □□□ □□□□□.
F. AmazonEC2ReadOnlyAccess □□□ □□□ □□ □□□ □□□ □□ □□□ □□□□ IAM □□□ □□□□□.
Answer: (SHOW ANSWER)

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

NEW QUESTION: 60
DevOps □□ Amazon API Gateway □□□□□□ □□□ □□□ □□ □□□□□□□ □□□□ API□ □□□□□. □□□ □□ □□ □□ □□□ □□ □□□ □□□ □□ □□ Amazon CloudWatch□ API □□□□□ □□ □□ □□□ □□□□ □□ □□□□□□. □□□ □□ □□ □□ □□ □□ □□ □□ □□ □□ □□ □□□□ □□□□ □□□.
□□ □□□□ □□ □□ □□□ □□□ □□□? (□ □□□ □□□□□.)
A. CloudWatch □□□□ □□ □□ □□□□ □□ □□□ CloudWatch□ □□□□□□ □□□□□ □□□□□.
B. API Gateway□□ AWS X-Ray □□□ □□□□□, □□ □□□□□ □□□□□ □□□□□□□ □□□□, □ □□ □□ □□ □□□□ □ X-Ray□ □□□□□□.
C. API Gateway□□ AWS X-Ray □□□ □□□□□, □□ □□□□□ □□□□□ □□□□□□□ □□□□, X-Ray □□□ □□□□ X-Ray□ □□□□□ □□□□□□.
D. □□□ □□ □□□ □□ □□□ API Gateway□ □□ □□□□ □□□□□ □□□□□□□ □□□□□.
E. API □□□ □□□ □□□ □□ □□□□ □□□□ CloudWatch □□□□ □□□□□□ □□□□□ □□□□□□□ □□□□□.
Answer: (SHOW ANSWER)
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

NEW QUESTION: 61
□□□ Docker □□□ □□□□ AWS CodeBuild □□□□□□ □□□□ □□□□ □□□□□. □□□□ □□□□ □□□ □ CodeBuild □ □□□□ □□□□ □□□□ Amazon S3 □□□ □□□□□□. CodeBuild □□□□□□ S3 □□□ □□□□ □ □□ □□□ □□ 1AM □ □□ □□□ □□□□.
DevOps □□□□□ S3 □□□ Amazon Elastic Container Registry(Amazon ECR) □□□□□□ □□□□ □□□□ □□□□ □□□□ □ □□. DevOps □□□□□ CodeBuild □□□□□ □□□ AWS □□□ ECR □□□□ □□□ □□□□□□ □□□□□. DevOps □□□□ □ □ ECR □□□□□ □□□ □□□ □□□□ 1AM □□□ □□□ □□□□□. □□ DevOps □□□□□ buildspec.yml □□□□ □□□ □ docker build □□ □ docker push □□□ □ □□□□□ □□□ □□□□□.
CodeBuild □□□□□ □□ □□□ □□□ □ □□□ ECR □□□□□□ □□□□□□ □□ □□□ □□□□□.
ECR □□□□□□ □□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □□ □□□ □□ □□ aws ecr get-login-password AWS CLI □□□ □□□□ ECR □□□□□□ □□□□□□ buildspec.yml □□□ □□□□□□□. □□ □□□ □□□□ ECR □□□□□□ □□□□□□ docker login □□□ □□□□□□□.
B. SECRETS_MANAGER □□□ □□ □□□ CodeBuild □□□□□ □□□□□. □□ □□□ CodeBuild □□□□□ IAM □□□ □□□ ARN□ □□□□□. □ □□ □□□ □□□□ docker login □□□□ □□□□□ ECR □□□□□□ □□□□□□ buildspec.yml □□□ □ □□□□□□.
C. ECR □□□□ □□ □□□ □□□□ □□□□□□□. 1AM □□□ □□□ □□□□ □ □□□ □□□□ ECR □□□□□ □□□ □□□ □□.
D. AWS CLI□ □□□□ ECR □□□ □□ 1AM □□□ □□□ □□□□□ buildspec.yml □□□ □□□□□□□. 1AM □□□ □□□ □□ □□ □ □□□ □□□□ ECR □□□□□ □□□ □□□□□.
Answer: A (LEAVE A REPLY)

(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.

NEW QUESTION: 62
□□□□ □□ AWS □□□ □□□□ VPC□ □□□□□□□ □□□□□□. □ □□ □□□□□ □□□ □□□ □□□□ □□□□. □□□ □□ □□ AWS WAF□ □□ □□ □□□ □□□□□□ □□□ □□□□□ □□□. □□□ □□□□□□□ □□ □□□□ □□□ □□□□ □□ □□ □□ □□□ □□□□□ □□ □□□ □ □□ □ □□□ □□□□□ □□□ □□□□.
□□ □□ □□□□□□ □□ □ □□□ □□ □□□ □□ IP □□□ □□ □□□□□□ □ □ □□□ □□□□□. □□ □□ DevOps □□ □□□□ □□ □□□ □□□□□. □□ □□□ □□ IP □□ □ □□□□ □□□□□□□ □□□□□ □□ □□ □□ □□ □□ □□□□□ □□ □□ □□□ □□□□ □□□□ □ □□□ □□ □□□□□ □□□□ □□□ □□□ □□□□. DevOps □□□□□ □□□□ VPC□ □□ VPC □□ □□□ □□□□□.
□□□ □□ □□□ □□ □□ □□□□□ □□□□ □□ DevOps □□□□□ □□□□ □□ □□ □□□ □□□□□?
A. Amazon CloudWatch Logs□□ □□ □□□ □□□□□. □□□ □□□□ □□□□ □□□□ □□ □□□□ □□□□ VPC □□ □□□ □□□□□. □□ □□□ IP □□□ □□ Amazon CloudWatch □□ □□□ □□□□□. □□ □□□ □□□□ □□□□ CloudWatch □□ □ □□□□□. □□□ 5□□□ □□□□ □□□ □□□ □□□□ 1□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□ □ □□□□ □□ □□□ □□ □□ □□□□.
B. □□ □□□ Amazon S3 □□□ □□□□□. □□□ □□□□ □□□□ □□□□ S3 □□□□ □□□□□ VPC □□ □□□ □□□□ □. □□ □□□ □□ Amazon OpenSearch Service □□□□ □ □□□□ □□□□□. S3 □□□□ □□□ □□□□, □□ □□□ □□□ □, OpenSearch Service □□□□□ □□□ □□□□ AWS Lambda □□□ □□□□□. 5□□□ □□□□□ Lambda □□□ □□□□□. □□ □□□ IP □□□□ □□□□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□ □□ □□ □□□ □□□□ OpenSearch Service□□ □□ □ □□□ □□□□□.
C. Amazon CloudWatch Logs□□ □□ □□□ □□□□□. □□ □□□ □□□ Amazon S3 □□□ □□□□□. □□ □□□□ □□□□ □□□□ □□ □□□□ □□□□ VPC □□ □□□ □□□□□. AWS Lambda□□ Amazon Athena CloudWatch □□□□ □□□□□. □ □□□ □□ □□□ □□□□□. □□ □□□ IP □□□□ □□□ □□ □□□□ □□□□□ □□□□ □□□ S3 □□□ □□□□□ Athena□ □□□□□. □ □□□ S3 □□□ □□□ □ Amazon Simple Notification Service(Amazon SNS) □□□ □□ □□ □□ □□□□ □□□□ S3 □□□ □□□ □□□□□.
D. □□ □□□ Amazon S3 □□□ □□□□□. □□ □□□□ □□□□ □□□□ S3 □□□□ □□□□□ VPC □□ □□□ □□□□□. □□ □□□ IP □□□ □□ S3 □□□ □□ □□ □□□ □□□□□ Amazon Athena□ □□□□□. Athena□□ □□□□ □□□□ □□ □□ □□□□ □ □□ □□□□□ □□□□ □□□□□ Amazon QuickSight□ □□□□□. □□□□ □□□□ □□ □□□ □□ 1□ □□ □□. □□ □□□□ □□□□ □□□ □ □□ □□ □□ □□□□ □□□□ □□□ □□□□□.
Answer: (SHOW ANSWER)

NEW QUESTION: 63
DevOps □□□□□ Auto Scaling □□□ Application Load Balancer □□□ □□□□ □□ Amazon EC2 □□□□□ □□□ □□□ □□□ □ □□□ □□ □□□□□□. EC2 □□□□□ □□ □□ HTTP □□ □□□ □□□□□. □□ □□ □□□□□ EC2 □□□□□□ □□□ □□□ □□□□□ □□□□ □□ □□□ □□□□□□. □□□ □□□ □□□ □□ □□□□ □□□ □□ □□□□. □□□□□ □□□□ □□□□□□ □□□ □□□ □□□□ □□ □□□□□□□ □□□□ □□□□ □□□. □□□ □□ □ □□□□□ □□□□ □ □□□ □ □□□□ □□□. □□ □□ □□□ □□□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. □□ □□□□ □□ □□□ □□□ □□ □□□□□ □□□□□ Auto Scaling □□□ □□□□□.
B. □□ □□ □□ □□ HealthCheckIntervalSeconds □□□□□ □□□□ □□ □□ □□□ □□□□.
C. □□ □□ □□ □□□ HTTP□□ TCP□ □□□□ □□□□□□□ □□ □□ □□ □□□ □□□ □ □□□ □□□□□.
D. □□ Auto Scaling □□□ □□ Amazon CloudWatch □□□□ □□□ □□ □□□ □□□ □□ □□□ □□□□□□. □□□ □□□□ □□ □ □□□ □□□□□. □□□ □□ □ □□□ □□□□ Amazon SNS □□□ □□□ □□□□□.
E. Amazon CloudWatch □□□□□ □□□□ Auto Scaling □□□ □□ EC2 □□□□□ □□□ □□□□ □□□□□. □□□ □□□□ □ □ □ □□□ □□□□ Amazon SNS □□□ □□□□ □□□ □□□□.
Answer: A,E (LEAVE A REPLY)
https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/metrics-collected-by-CloudWatch-agent.
html

NEW QUESTION: 64
DevOps □□□□□ AWS Control Tower□ □□□□ □□ □ □□□□, □□ □ □□ □□□ □□□□ □□ □□ AWS □□□ □□□□□. □□ □□ DevOps □□□□□□ □□□ □□□□ AWS Control Tower □□□□□ □□□□ □ □□ □□□□□. □ □□□□□ AWS Organizations□□ □□ □□□ OU□ □□ □□ □□□ □□□□□ □□□.
□□□□ □□ □□ □□□ □□ □□□□ □□□□□. □□□□ □□□ □□ □□□ □□□□ □□□ □ □□□ □□□. □□□□ □□□ □□(OU) □□□ □□□□ □□□ □□□□.
□□□□ □□□□ □□ □□□ □□□ □□□□□ □□□□□ □□□. □□ □□□□ □□□ □□□ □□ □□□ □□ □□□□□ □□ □ □□□□.
□□ □□□□ □□ □□ □□ □□□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□□ □ □□□□□ □□ AWS CloudFormation □□□□ □□□□□. □□□□ AWS CodeConnections□ □□□□ Git □□□□□□ □□□□□. □□□ □ OU□ □□ □□□□ AWS::ControlTower::EnableControl □□ □□□□ □□□□□. Git □□□□□□ □□□□ □ □□□ □□□□ AWS CodeBuild □□□□□ □□□□□.
B. □□□ □ □□□□□ □□ □□ AWS CloudFormation □□□□ □□□□□. □□□□ AWS CodeConnections□ □□□□ Git □□□ □□□ □□□□□. □□ □ □ □□□ □□ □□□□ AWS::ControlTower::EnableControl □□ □□□□ □□□□□. □□ □ □□□ AWS CodePipeline□□ □□□□□□ □□□□□. □□ □□ □□□ □□□ □ □□□□□□ □□□□ □□□□ □□□□ □□□□□ □□□□ □ □□□.
C. □□□ □□□□□ □□ □□ AWS CloudFormation □□□□ □□□□□. □□□□ AWS CodeConnections□ □□□□ Git □□□□□ □ □□□□□. □□□ □ OU□ □□ □□□□ AWS::ControlTower::EnableControl □□ □□□□ □□□□□. □□ □ □□□ AWS CodePipeline□□ □□□□□□ □□□□□. □□ □□ Git □□□□□□ □□□ □□□ □ □□□□□□ □□□□□ Amazon EventBridge □□□ □□□□□.

D. □□ □ □□□ AWS CodePipeline□ □□□□□□ □□□□□. □□□□□□ Amazon S3 □□□ PutObject □□□□ □□□□□ Amazon EventBridge □□□ □□□□□. □□□ □ □□□□□ □□ □□ AWS CloudFormation □□□□ □□□□□. □□□□ S3 □□ □ □□□□□. □□□ □ OU□ □□ □□□□ AWS::ControlTower::EnableControl □□ □□□□ □□□□□.

Answer: (SHOW ANSWER)

The security team's requirements emphasize governance, approval control, versioning, and automation with minimal operational overhead. AWS Control Tower natively supports guardrail enablement through the AWS::ControlTower::EnableControl CloudFormation resource, which can target entire organizational units.

Managing these resources as CloudFormation templates stored in a Git repository provides built-in version control, change review, and rollback capabilities through standard Git workflows.

Option C implements a GitOps-style approach. Each guardrail is defined declaratively in CloudFormation, scoped at the OU level, ensuring consistent enforcement across all accounts in that OU. Storing the templates in a CodeConnections-compatible Git repository allows the security team to review, approve, and audit changes through pull requests. By configuring AWS CodePipeline in the security team's account and triggering it automatically via an Amazon EventBridge rule on repository merges, only approved changes are deployed. This ensures that new guardrails are enabled only after explicit approval and that the deployment process is fully automated.

Option A lacks pipeline orchestration and approval workflows. Option B requires manual pipeline invocation and per-account configuration, increasing operational overhead. Option D relies on S3 as the source of truth, which lacks native version control and approval mechanisms compared to Git.

Therefore, Option C provides the most operationally efficient, secure, and auditable solution aligned with AWS Control Tower automation best practices.

NEW QUESTION: 65

□ □□□ □□ □□□ □□□□□□□ □□□□□□□ AWS□ □□□□□□□□□□. □□□□□□□ Amazon API Gateway □□ □□ Application Load Balancer □□ □□ Amazon EC2 □□□□□□ □□□□□□. □□□ □□□□ □ □□□ □□ □□□□□ □□□□ □ □ □□□□ □□□ □□□□□ □□□□. □□□ □□ □□□ □□ □□ □□□□□ □□□□ □□□ □ □□□ □□□□.

□□ □□□□□ □□ □□□□ □□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?

A. □□ □□□ □□ □□□ □□□□ □□□ □□□□ □□ □□□□ □□□ □□□ □□□□ □□□ □□□□ □□ □□ □□□ □ □□□ □ □□□□ API Gateway□ □□□□□.

B. □□ □□□ □□□ □□□ □□□□ □□ □□□ □□□□□. □ □□□ □□□□□ □□□□□□□ DNS □□ □□□□ □□□□□□ □.

C. □□ Application Load Balancer □□ □□□ □□ □□□□ □□ □□ □□ □□□□□ □□□ □□□□ □ □□ □□□□ □□□□□□ API Gateway□ □□□□□.

D. □□ □□□ □□ Application Load Balancer □□ □□□ □□ □□□□ □□ □□ □□□□ Application Load Balancer□ □□□□ □□ □ □□ □□□□ □□□□ □□□□□ API Gateway□ □□□□□.

Answer: A (LEAVE A REPLY)

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.

NEW QUESTION: 66

□□□□ □□□□ □□ □□□□□□ □□ □□□ □□□□ □□ □□□□ □□□□. □□□□ □ □□□□□ Amazon S3 □□□ □□ □ □□ □□□□ □□□□. □□□ □□ □ □□ □□□□ □□□□ □□□ □□ □□□ □□ □□□ □□□□ □□□□. □□□ □□ □□□ □ □□ □□ □□□ □□□ □□□□ □□□ □□□ □ □□□□ □ □□□ □□□□ □□□ □ □□□ □□□ □□□ □□□□□ □□□. □□□.

□□□ □□ □□□ □□□□ □□ □□□□ □□□ □□□□□?

A. □ □□□□□ □□ AWS CodeCommit □□□□□□ □□□□, □□□□ □□□ mam □□□□ □□□□, □□□□ □□□ □□□ □ □ □□□ □□□□ □□□□□. □□ □□□□ □□□□ □□□ □□□ □□□□ □□□ □□□ □ □□ □□□ □□□□ □□ □ □□□ □□□□□. □□.

B. □□ □□□□ □□ □ □□□□□ □□ □ □□ S3 □□□ □□□□ AWS Lambda □□□ □□□□ □□□ □□□ □□□□ □□ □□ □□ □□□ □□□□□. □□ □□□□ □□ □□□ □□□□□ □□ □□□ □□□□□.

C. □ □□□□□ □□ AWS CodeCommit □□□□□□ □□□□ □ □□□ □□ □□ □□ □□ □□□□□□ □□ □□□□ □ □□□ □□□ □□ □□□□ □□□□□. □□ □□□□ □□□□ □□□ □□□ □□□□□.

D. □ S3 □□□□ □□ □□ □ □□□ □□□□□, □□□□ □□□ □□ □□ □□□ □□□□, □□□□ □□□ □□□ □□ □□□ □ □□ □□□□□. □□□□ □ □□□□ □□□ □□ □ □□□ □□□□□ □□□.

Answer: A (LEAVE A REPLY)

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. References:

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

NEW QUESTION: 67

□ □□□ □□ AWS □□□□ □□ □□□ □□ □□□□□□□ □□□□□. AWS CodePipeline □□□□□□ □□□ Amazon Elastic Container Service(Amazon ECS) □□□□□ □□□□ Amazon Elastic Container Registry(Amazon ECR) □□□□□□ □□□□ □□□ □□□. □□ □□ □□□□□□ □□ □□□□ □□□□ ECS □□□□□ □□□□□□.

□□ □□□ □□□□ □□□□□ □□□ AWS □□ □□ □□ AWS □□□□ □□□□ □□□.

□□ □□□□□ □□ □□□ □□ □□□□ □□□□□ □ □□□ □□□.

□□ □□□□ □□□ □□ □□□ □□□□□?

A. Amazon ECR VPC □□□□□□ Amazon S3 □□□□□ □□□□□□ □□□□□. □□□ AWS □□□ ECR □□□□□□ □□□□ □. □□□□ ECS □□□□ □□ AWS □□□□ □□□□ □□□ □ □□□ □□□□□ □□□ □□□□□. □□□□ ECS □□□ □□ □ □□ ECR □□□□□□□ □□□□ □□□□□ □ □□ □□□ □□□□□.

B. □□ AWS □□□ □□□□ ECR □□□□□□ □□□ □□□□□. □□□ AWS □□□ □□ □□□□ ECS □□□□ □□ □□□□ □ □□□ □□□ □ □□□ □□□□□ □□□ □□□□□. □□□□ ECS □□□ □□ □□□ ECR □□□□□□□ □□□□ □□□□□ □ □□ □□□ □□□□□.

C. □□ AWS □□□□ ECR □□□□ □□□ □□□ □□□□□. □□ □ □□□ □□□□□□.
□□□ AWS □□□ □□ □□ ID□ □□□□□.
D. Amazon ECR VPC □□□□□□ Amazon S3 □□□□□ □□□□□□ □□□□□. □□ AWS □□□ ECR □□□□□□ □□□□□ □□□ □□□□□. □□□ AWS □□□ □□ □□□□ ECS □□□□ □□ □□□□ □□□□ □□□ □ □□□ □□□□□ □□□ □□ □□□. □□□□ ECS □□□ □□ □□□ ECR □□□□□□□ □□□□ □□□□□ □ □□ □□□ □□□□□.
Answer: (SHOW ANSWER)
Comprehensive and Detailed Explanation From Exact Extract of DevOps Engineer documents only:
The correct solution is to allow the production ECS cluster in the separate AWS account to pull images directly from the Amazon ECR repository in the main AWS account by using a cross-account repository policy and the required private networking endpoints.
Option D satisfies both requirements:
* Cross-account image access is provided by the repository policy on the Amazon ECR repository in the main AWS account.
* Private image download is provided by Amazon ECR VPC endpoints together with the Amazon S3 gateway endpoint, which are required for private image pulls without using the public internet.
* The ECS task execution role in the production account also needs permission to pull the image.
Why the other options are incorrect:
A). Creating an ECR repository in the separate account is unnecessary for this requirement, and the option does not correctly describe the image pull model from the source repository.
B). This option addresses cross-account access, but it does not include the VPC endpoints and S3 gateway endpoint required for private connectivity.
C). Cross-account replication copies images to another account, but this option does not address the requirement for the production cluster to download images over a private connection.

NEW QUESTION: 68
□ □□□□ Amazon DynamoDB□□ □□□□ □□ AWS Lambda□□ □□□□ □□□□□□□□ □□□□. Lambda □□□ □□□□ □□□ □ □□□□ □□□□ □□□□□. □□□ □□ □□□□ □□□ □□□□□ □□□□□□ □□□□ □□□. □□ □ □□□□ □□ □□ □ □□□ □□ □□□□ □□ □ □□□ □□□ □□ □□□□□ □□□□ □□□.
□□ □□ □□□ □□□□ MOST □□□ □□□ □□□□ □□□□ □□□□□?
A. AWS CodePipelme □□□ □□□□ □□ □ □□□ □□□□ □□□ □□ □ □□□□□□ □□□□□□. AWS CodeDeploy□ □□□ □ □□□ □□□ □□□ □□□□ Canary □□ □□□ □□□□□.
B. □□□ □□□ □□□ □ □□□□□ AWS CodeBuild □□□ □□□□□. AWS CloudFormation□ □□□□ □ Lambda □□□ □□□ □ □□□ □□ □□ □ □□□ □□□□ AWS CodePipelme □□□ □□□□□□.
C. AWS CodePipelme □□□ □□□□ □□□ □□□ □ □□□□ □□ □□ □□□ □□□□□. AWS CodeBuild□ □□□□ □□□□ □ □□□□ □□ □□□ □□□□□. □□□ AWS CodeDeploy □□□ □□□ □□ CodeDeployDefault.LambdaLinearlDPercentEvery3Minut.es □□□ □□□□□.
D. AWS CLI□ □□□□ □□□□ □□□ □ Amazon S3 □□□ □□□ □□□□□ □□ □ □□□ □□□□□. □ □□□ □□□□ Lambda □□□ □□□□ S3 □□□ □□□□ □□□□□. Lambda □□□ □□□ □□□□ □□□ □□□ □□ □□□ □□□ □□□ □ □□□□.
Answer: (SHOW ANSWER)
https://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-configurations.html

NEW QUESTION: 69

□□□ EC2 □□□ □□ □□ □□□ □□ □□ □□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. AWS Config □□ □□ □□ □ □□□ □□□ □□ Systems Manager Patch Manager□ □□□□□.
B. □ □□□□□ □□□□ SSH□ □□□□□.
C. □□ AMI□ Auto Scaling □□□ □□□□□ □□ □□□□□.
D. □□ □□□ CloudFormation □□□□ □□□□□.
Answer: A (LEAVE A REPLY)
AWS Systems Manager Patch Manager automates patch scanning and installation. Integrating with AWS Config enables compliance reporting and auto-remediation. This provides centralized patch compliance management per AWS Ops best practices.

NEW QUESTION: 70
□ □□□□ AWS□ □□□□ □□□ □□□□□ □□□□ □□□□. □□□ □ □□□□□□ □□□ □□□□□□ □□□□ □□ □□ AWS □□□ □□□□. □□□ AWS Organizations□ □□□ □□□□□.
□□□ □□ □□□ □□ □□□ □□□□□ □□□□. □□□ □□ □□□□ □□ □□ □□□ □□□□ □□ □□□□□ AWS CloudFormation □□□ □□□□□□. □□□□ □□ □□ AWS Management Console□ □□□□ □□□□□□ □□ □□□ □□□□□ □ □□□□ □□□ □□□□ □□□ □ □□□□.
DevOps □□□□□ □□ □□□ □□□□ □□ AWS □□□ □□ □□□ □□ □□□□□ □□□□ □□□□ □□□□ □□□. □□□□ □□ □□ □ 15□ □□□ □□□□ □□□ □□□□ □□□. □□ □□□□ □□□ □□□□□□ □□□□ □□ □□□ □□□□□□ □□ □ □□□□ □□□□ □□□□ □□□.
□□□□ □□ □□□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. CloudFormation □□□□ □□□ □□□□ □□□ □□□□ □□□□□. CloudFormation□ □□□□ □□ □□□□ □□□□ □□ □ □□ □□ AWS Lambda □□□ □□□□□. Amazon CloudWatch Logs □□ □□□ □□□ □□□□□ Lambda □□□ □□□□□. □□ □ □□ □□ □□□ □□□□□ Amazon CloudWatch □□□□□ □□□□□.
B. AWS □□□□ AWS CloudTrail□ □□□□□□. Amazon Athena□ □□□□ CloudTrail □□□ □□□□ □□□ □□□□ □□□□□. AWS Step Functions□ □□□□ □□□□ □□□ □□ Athena□□ □□ □□□ □□□□ □□ □□□ □□ AWS Lambda □□□ □□□ □□. □□□ □□ Athena□ □□□ □□□ □□□□ Amazon QuickSight □□□□□ □□□□□.
C. □□ AWS □□□ AWS Config□□ □□ □□□□ □□ □□□ □□□□ □□□□□. □□ AWS □□□□ ~no-enable-default-standards □□□ □□□□ AWS Security Hub□ □□□□□□. AWS Config □□□ □□□ □□□ □□ □□□ □□□□□. AWS Config □□□ □ □ □□□□ □□ □□□ □□□□□. □□□ □□ □□□ Security Hub □□□ □□□□ Security Hub□ □□□□□ □□□□□.
D. AWS □□□□ AWS CloudTrail□ □□□□□□. Amazon CloudWatch Logs□ □□□□ CloudTrail □□□ □□□□ □□□ □□□□ □□□□□. □□□□ □□□ □□ CloudWatch Logs □□□ □□□□□. Amazon EventBridge□ □□□□ □□ □□□ □□ Lambda □□ □ □□□□□. □□□□ CloudWatch □□□ Amazon OpenSearch Service□ □□□□□□□. □□□ □□ OpenSearch □□□□ □□□ □□ □□□□□.
Answer: C (LEAVE A REPLY)
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.
AWS Config conformance packs
Introducing AWS Config conformance packs
Managing conformance packs across all accounts in your organization

NEW QUESTION: 71
□□□□ AWS□ □ □□□□□□□ □□□□ □□□□. □□□□□□□ Application Load Balancer □□ Amazon EC2 □□□□□□ □□ □□□. □□□□□ □□ □□ □□□ Auto Scaling □□□□ □□□□□. □□□□□□□ Oracle DB □□□□ □ Amazon DynamoDB□ Amazon RDS□ □□□□ □□□□□. □□ □□□□ □□□□□□ □□□ □□□ □□□□.
□□ □□ □□ □□ □□□ □□ □ □□ □□ □□□□ □□□ □□□ □□□□□?
A. AWS □□□□ □□□□□ □□ AWS Systems Manager securestring □□□□□□ □□□ □□ □□□□□. Systems Manager SecureString □□□□□□ □□□□□□ □□ □□□ □□□□□.
B. EC2 1AM □□□ EC2 □□□□□ □□□□ AWS □□□□ □□□□□□. AWS Secrets Manager□□ □□□□□□ □□ □□□ □□ □□□.
C. AWS □□□□ □□□□□ □□ AWS Systems Manager □□ □□□ □□□□□□ □□□ □□ □□□□□. Systems Manager SecureString □□□□□□ □□□□□□ □□ □□□ □□□□□.
D. EC2 1AM □□□ EC2 □□□□□ □□□□ AWS □□□□ □□□□□□. □□□□□□ □□□□□ □□ □□□□ □□ □□□ □□□ □□□ □□□ □□□□□.
Answer: B (LEAVE A REPLY)
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/

NEW QUESTION: 72
□□□ AWS CloudFormation □□□ □□□□ □□□□□□□ □□□□□ □□□□□. □□□ □□ □□ □□□□ □□□□□. □□□□ □ AWS Auto Scaling □□, Amazon EC2 □□□□, Application Load Balancer(ALB) □ □□ □□□ □□□□ □□ □□□□ □ □□□ □ □ □□□□ □□□□□. CloudFormation □□ □□□□ □□□ □□□□□□ □□□□ □□ □□□ □□□□ □□□□.
□ □□□ □□ AWS CLI□ □□□□ □□□□□□ □□ □□□□□ □□□□□□. □□ □□□□□ □□□□ □□ □□ □□□□ □□□ □□□□. □□□ UPDATE_ROLLBACK_FAILED □□□ □□□□□. * □ □□□ □□□□ □□□□ □□□□□?
A. ALB□ □□ □□□ □□□ □□□ □□□□□□□. aws cloudformation update-stack-set AWS CLI □□□ □□□□□.
B. □□□ □□□□□□ □ □□□ □□□ □□□□ 1AM □□□ □□□□□□□. aws cloudformation continue-update-rollback AWS CLI □□□ □□□□□.
C. □□□ EC2 □□□□ □□ □□ □□□ □□ □□□ □□□□□. aws cloudformation cancel-update-stack AWS CLI □□□ □□□□□.
D. Auto Scaling □□ □□□□ □□□□□. aws cloudformation rollback-stack AWS CLI □□□ □□□□□.
Answer: B (LEAVE A REPLY)
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.

NEW QUESTION: 73
DevOps □□□□□ S3 □□ □□ □□ □□□ □□□□ □□□□ □□ □□□□ S3 □□ □□ □□□ □□□ Amazon S3 □□□ □□□□ □□□. □□□ □□ AWS □□ □ □□□ □□ □□□ □□□□ □□□.
□ □□□ □□□□□□ □□ □□ □□□ □□□□ □□□? (3□□ □□□□□.)
A. □□ □□□□ □□ IAM □□ □□
B. □□ □□□□ □□ I AM □□□ □□□□□.
C. □□ IAM □□□ □□□ □□□□□ □□□□ □□ □□ □□□ □□ □□□□□.
D. □□ IAM □□□ □□□ □□□□□ □□□□ □□ □□ □□□ □□ □□□□□.
E. □□ □□□ □□ □□□ □□□□ □□□ □□□□□□.
F. □□ □□□ □□ □□□ □□□□ □□□ □□□□□□.
Answer: A,D,E (LEAVE A REPLY)
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

NEW QUESTION: 74
□ □□□□ AWS CodeDeploy□ □□□□ Amazon EC2 □□□□ □□□ □□□□□□□ □□□□ □□□□. □□ □□ □□□□ □□ EC2 □□□□□ □□□□□ □□□□□□. DevOps □□□□□ □□□ □□ □□□ □□□□ □□□ □□□ □□ □□ □□ □□ □□□□ □□□□ □□□.
□□ □□□ □□ □□□ □□ □□□ □□□□□ □□□□ □□ □□ □□□□ □□□ □□□□□?
A. □□□□□ □□□ □ EC2 □□□□□ SSH□ □□□□ □□□□□. CodeDeploy □□□□□ □□□ □□□□.

B. AWS Systems Manager □□ □□□□ □□□□ □□□□□ □□□ □ EC2 □□□□□ □□□□□. CodeDeploy □□□□□ □□□ □ □□□.
C. CodeDeploy □□□ □□□ Amazon S3 □□□ □□□□□. appspec.yml □□□ □□□□□□ □□□ S3 □□□ □□□□□. Amazon Athena□ □□□□ S3 □□□ □□□□□.
D. CloudWatch □□□□□ □□□□ CodeDeploy □□□□ □□□ Amazon CloudWatch Logs□ □□□□□. CloudWatch Logs Insights□ □□□□ □□□ □□□□□.
Answer: D (LEAVE A REPLY)
AWS recommends centralizing CodeDeploy agent logs in Amazon CloudWatch Logs for efficient analysis.
By configuring the CloudWatch agent or using the CodeDeploy console integration, you can stream logs in near-real time and query them with CloudWatch Logs Insights. This enables filtering by deployment lifecycle events, saving time versus manually connecting to instances. Refer to AWS documentation section
"Monitoring and troubleshooting deployments with CodeDeploy and CloudWatch."

NEW QUESTION: 75
□ □□□ □□□ Amazon S3□ □□□□ us-east-1 □□□ □□ □□□□ □□□□□ □□□□□ □□□□. □ □□□ □□ □□□ □□□ □ □□ □□□□ □□□ □□□□ □□□ □□□ □□□ □□□ □□□. □□ Amazon CloudFront□ □□□□ □□□□ □□□ □□ □□ □□ □□□□ □□□. □□ □□□ □□□□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. eu-west-1 □□□□ Origin Shield□ □□□□ Cl oudFront□ □□□□□□. □□□ □□ □□□ □□□□□. AWS Global Accelerator□ □□□□ □□□ □□ □□□ □□□ □□ □□□□□□ □□□□□□. us-east-1 □□□ S3 □□□ □□ □□□ S3 □□ □ S3 □□□ □ □□□□□.
B. us-east-1□ S3 □□□ □□ □□□ S3 □□ □□ S3 □□□ □□□□□□.
C. us-east-1□□ Origin Shield□ □□ CloudFront□ □□□□□□. □□□ □□ □□□□□ □□□□□. □□ □□□□ □□ □□□□ □□ □□□ TTL□ □□□□ □□ □□□ □□□□□. □□ □□□ S3 □□□□ □□□ □□ □□□ □□□□□. us-east-1□ S3 □□□ □□ □ □□ S3 □□ □□ S3 □□□ □□□□□□.
D. □□ CloudFront □□□ □□□□□. □ □□□ □□ □□□ □□ S3 □□□ □□□□□ □□□□□. Amazon Route 53 □□ □□ □□ □□□□ □□□□ □□□□ □□ □□□ □□□ □□□□□.
E. us-east-1□□ Origin Shield□ □□□□ CloudFront□ □□□□□□. □□ □□□ Amazon ElastiCache □□□□□ □□□□ CloudFront□ S3 □□□ □□ □□ □□ □□ □□□ □□□ □□□. us-east-1□ S3 □□□ □□ □□□ S3 □□□ □□□□□ □□ □□ □□□ □□□□□. Amazon EventBridge□ □□□□ □□□□□ □□□ □ □ □□□□□ □□□□□.
Answer: (SHOW ANSWER)

NEW QUESTION: 76
□ □□□ □□ AMI□ □□ □□□□□□ □□□□□□ □□□□□ □□□. □□□ Amazon EC2 □□□□□ □□ □□□ □□□□□ □□ □□□ □□□□□ AMI□ □□□□□□ □□□. □ □ AMI□ AWS Organizations□ □□ □□ □□□ □□ AWS □□□□ □□□ □ □□ □ □□□.
□□□ AMI□ □□□□ □□ □□□□ □□□□□□ □□□□ □□□.
□□ □□□□ □□ □□ □□ □□□□□ □□□ □□ □□□ □□□ □ □□□□?
A. AWS CodeBuild□ □□□□ AWS CodePipeline □□□□□□ □□□□□. □□ □□□□□□ □□□□ AWS Lambda □□□ □□□□ □. AWS CloudFormation □□□□ □□□□□. □□□□ □□□ □□ AWS □□□ □□□□□.

B. EC2 Image Builder□ □□□□ AMI □□□□□□ □□□□□. □□□□□□ □□□□ □□□ AWS □□□ AMI□ □□□□□. □□□ □□□ □□ □□□□□ □□□□□.
C. AMI□ □□□□ □□ AWS Lambda □□□ □□□□ AWS CodePipeline □□□□□□ □□□□□. □□□ AWS □□□ AMI□ □□□ □□ □□□□□□ □□□□□. Amazon EventBridge Scheduler□ □□ □□□□□□ □□□□□ □□□□□.
D. AWS Systems Manager Automation □□□ □□□□□. □□□ □□ AWS □□□□ □□□□ □□□□□ □□□□□. □□ □□□□ □□□□ AWS Lambda □□□ □□□□□.
Answer: (SHOW ANSWER)
EC2 Image Builder is the recommended AWS-native service for automating the creation, validation, and distribution of AMIs. It integrates directly with AWS Organizations, allowing AMIs to be automatically shared with all accounts. The service also supports scheduled builds (e.g., monthly), significantly reducing operational overhead. This design is referenced in AWS documentation: "Automate OS image creation and distribution securely across multiple accounts using EC2 Image Builder."

NEW QUESTION: 77
DevOps □□□□□ Application Load Balancer(ALB) □□ □□ Amazon Elastic Container Service(Amazon ECS) □□□□ □□□ □ □□ □□□□□ □□ □□□□□.
□□ □□□ □□□□□ □□ □□□□□□ □□□ □□□ □□□□ □□□□□□□ □□□□□ □□□□□□□.
DevOps □□□□□ □□□□ □□□ □□□ □□□□ □□ □□□ □□□□ □□□□ □□□□□ □□□□ □ □□ □□□□ □□□□ □ □□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. AWS CodePipeline□□ □ □□□□□□ □□□□□. □□□□□□ AWS CodeConnections□ □□□□ Git □□□□□□ □□□□□. ECRBuildAndPublish □□□ Amazon ECS □□ □□ □□□ □□□□ □□□□□□ □□□□□. ECS □□□□ □□ □□□□ □□ □□ □□ □□□□□.
B. ECS □□□□ AWS CodeDeploy□ □□ □□□□□ □□□□□ □□□□□□□. ALB□ □ □□ □□ □□□ □□□□□. CodeDeploy □□□□□□□ □□ □□□□□□□ □□ □□□ □□□□□. CodeDeployDefault.ECSCanary10Percent15Minutes □□ □□□ □□□□ CodeDeploy□ □□□□□ □□□□□.
C. AWS Lambda □□□□ □□□□ ECS □□ □□□ □□□□□ □ □□ □□□ □□ IAM □□□ □□□□ □ AWS CodeBuild □□□□ □ □□□□□. □□□ □□ Amazon ECS RegisterTaskDefinition API□ □□□□□ □□□□ □□ □□ □□□ □□□□□□□. ECS □□ □□ □□ □□□□ □□ □□□ □□□□□ □□□□□.
D. □ AWS CodeBuild □□□□□ □□□□□. □□□□□ AWS CodeConnections□ □□□□ Git □□□□□□ □□□□□. Git □□□□ □□ Amazon ECS UpdateService API□ □□□□ □□□ □□□ buildspec.yml □□□ □□□□□. ALB □□ □□□ □□□ □□□ □□ □ 2□□□ □□□□□.
Answer: (SHOW ANSWER)

The key requirement is to shift a small percentage of traffic to a new version first to validate changes before rolling out to the entire ECS service, thereby reducing downtime and deployment risk. For ECS services behind an Application Load Balancer, the AWS-recommended and most reliable approach is to use AWS CodeDeploy with blue/green or canary deployments .
Option B directly addresses this requirement. By configuring the ECS service to use CodeDeploy as the deployment controller , CodeDeploy can manage traffic shifting between two ALB target groups: one for the current (stable) version and one for the new version. The CodeDeployDefault.
ECSCanary10Percent15Minutes deployment configuration initially routes 10% of production traffic to the new task set, monitors the deployment for issues, and then shifts the remaining traffic after the defined interval. This enables early detection of application errors while limiting user impact.
Option A and C rely on rolling updates , which replace tasks incrementally but do not provide precise traffic control or automated rollback based on health checks. Option D's slow start setting only affects how quickly targets receive traffic after registration and does not implement true canary testing.
Therefore, using AWS CodeDeploy with ECS canary deployments is the most robust, AWS-supported solution to test changes safely before full rollout.

NEW QUESTION: 78
DevOps □□□□□ AWS Cloud Formation □□□ □□ □□□□ □□□□ AD Connector□ □□□□□□. AWS Lambda □□□ □□□ □ AD □□□□ □□□□□ Cloud Formation□ CREATE_IN_PROGRESS□□ CREATE_COMPLETE□ □□□□ □□□□.
□□□□□ □ □□□ □□□□ □□ □□ □□□ □□□ □□□?
A. Lambda □□ □□□ □□□□□ □□□□□□ □□□□□□.
B. Lambda □□ □□□ □□ □□□ URL□ □□ □□□ □□□□□ □□□□□.
C. Lambda □□ IAM □□□ □□ ARN□ □□ cloudformation UpdateStack □□□ □□□ □□□□□.
D. Lambda □□ IAM □□□ AWS □□□ □□ ds ConnectDirectory □□□ □□□ □□□□□.
Answer: (SHOW ANSWER)
Reference: https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/crpg-ref-responses.html

NEW QUESTION: 79
□ □□□ PipelineAccount□□ AWS □□□ □□□□. □ □□□ AWS CodePipeline□□ □□□□□□ □□□□□. □ □□□ CodePipeline_Service_Role□□□ IAM □□□ □□□□ Amazon S3 □□□ □□□□ □□□□□ □□□□□. □ □□□ □□ □□□ AWS KMS □□ □□□□ S3 □□□ □□□ □□□□□□.
DevOps □□□□□ CodeDeployAccount□□ AWS □□□□ AWS CodeDeploy □□□□□□□ □□□□ □□□ □□□□□ □□□□□ □□□□□□ □□□□□ □□□.
DevOps □□□□□ CodeDeployAccount □□□ □ □□ □□□ □□□□ □□ KMS □ □□□ □□□□□□□. DevOps □□□□□ □□ □□□□ □□□ CodeDeploy □□□□ □□□□ □ □□ DevOps_Role□□□ □□□ IAM □□□ CodeDeployAccount □□□ □□□□ □. DevOps □□□□□ CodeDeployAccount □□ □□□ □□□□ Amazon EC2 □□□□ □□□ □□□□□□ S3 □□□ PipelineAccount □□□ □□ KMS □□ □□□□ □ □□□ □□□.
□□□ □□ □□□ □□□□□ □□ □□ □□□ □□□□□?
A. S3 □□ □□□ □□□□□□ CodeDeployAccount □□□ S3 □□□ □□ □□□ □□□ □□□□□.
PipelineAccount □□□ DevOps_Role □□□ □□□ □ □□□ □□□□ IAM □□ □□□ □□□ IAM □□□ □□□□□.
CodePipeline_Service_Role IAM □□□ □□□□□□ DevOps_Role □□□ □□□ □ □□ □□□ □□□□□.
B. S3 □□ □□□ □□□□□□ CodeDeployAccount □□□ S3 □□□ □□ □□□ □□□ □□□□□.

PipelineAccount □□□ □□ □□□ □□ □ □□□ □□□□ IAM □□ □□□ □□□ DevOps_Role IAM □□□ □□□□□.
CodePipeline_Service_Role □□□ □□ □ □□ □□□ □□□□□ DevOps_Role IAM □□□ □□□□□□□.
C. S3 □□ □□□ □□□□□□ PipelineAccount □□□ S3 □□ □□□ □□□ □□□□□. DevOps_Role IAM □□□ PipelineAccount □□□ □□ □□□ □□ □ □□□ □□□□ IAM □□ □□□ □□□□□. CodePipeline_Service_Role IAM□ □□□□□□ DevOps_Role □□□ □□ □ □□ □□□ □□□□□.
D. S3 □□ □□□ □□□□□□ CodeDeployAccount □□□ S3 □□ □□□ □□□ □□□□□. DevOps_Role IAM □□□ CodeDeployAccount □□□ □□ □□□ □□ □ □□□ □□□□ IAM □□ □□□ □□□□□. CodePipeline_Service_Role IAM □□□ □□□□□□ DevOps_Role □□□ □□ □ □□ □□□ □□□□□.
Answer: A (LEAVE A REPLY)
For cross-account deployments, CodePipeline must assume a role in the target account. Configure S3 bucket policy to allow the target account (CodeDeployAccount) access, trust relationship in DevOps_Role to allow assumption by the pipeline's account (PipelineAccount), and update CodePipeline_Service_Role with sts:
AssumeRole permissions. This cross-account pipeline setup is documented in "Cross-Account Access for AWS CodePipeline."

NEW QUESTION: 80
□ □□□ □□□□□ Windows □□□□□□□ Linux □□□□□□□ AWS□ □□□□□□□□ □□□□. □□□ □□□□ □□ Amazon EC2 □□□□□ □□□□ □□□□□ □□□ □□□□ □□□□. □□□□□□□ □□□□□□□□ Windows□ SMB□ Linux□ NFS□ □□□□ □□ □□□□□ □□ □□□□ □□□□□.
□□ □ □□□ □□ AWS □□□□ □□□ □□ □□(DR) □□□ □□□□□ □□□□ □□□□. □□□ □□□□ □□□□ DR □□□□ EC2 □□□□□ □□□□ □□□ □□□□. □□□ □□□□□ DR □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□ □□□□□?
A. □□□□□□ □□□□□ Amazon S3□ □□□□□. □□ □□□ S3 □□□ □□□□ DR □□□ S3 □□□ □□□□□. □□ □□□□ DR □□□□ S3 □□ □□ □□(CRR)□ □□□□□.
B. □□□□□□ □□□□□ Amazon Elastic Block Store(Amazon EBS)□ □□□□□. □□ □□□ □□ EBS □□□ □□□□ □□□□ □□ □□□□ DR □□□ □□□□ □□ □□□ AWS Backup□□ □□□□□.
C. □□□□□□ □□□□□ AWS Storage Gateway□ □□ □□□□□□ □□□□□. □□ □□□□ DR □□□□ □□ □□□□□□ CRR(□□ □□ □□)□ □□□□□.
D. □□□□□□ □□□□□ NetApp ONTAP□ Amazon FSx□ □□□□□. DR □□□ ONTAP□ FSx □□□□□ □□□□□. □□ □□ □□ DR □□□□ NetApp SnapMirror □□□ □□□□□.
Answer: D (LEAVE A REPLY)
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.
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

NEW QUESTION: 81
DevOps □□□□□ AWS CodePipeline□ □□□□ □□□ □□□□□□ □□□□ □□□□□□□ □□, □□, □□□□, □□□ □ □□ □□□. □□□ □□□ □□ □□ □□□ □□ □□ □□□ □□□□□. □□ □□ □□ □□□ □□□ □□□ □□□ □□□ □□ □□□ □ □ □ □□ □□□ □□□□□.
DevOps □□□□□ □□ □□□ □□□ □□□□□ □□ □ □□ □□□ □□ □□ □□□□□ □□□ □□□□ □□□?
A. CodePipeline □□□□□ □□ □□ □□□ □□□□□ Amazon CloudWatch Logs □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□ □□□□ □□□□□. SNS □□□ □□ □□ □□ URL□ □□□□ □□ □□□ □□□ □□□□□.
B. AWS CloudTrail □□□□ □□ □□□□ AWS Lambda □□□ □□□□□. CodePipeline □□□□□ □□ □□ □□ □□□□ □□□ □ □□□ □□ □□□ □□ □□□ URL□ □□□□.
C. CodePipeline □□□□□ □□ □□ □□□ □□□□□ Amazon EventBridge □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□□□. □□□ □□ □□□ □□ □□□ URL□ □□□ AWS Lambda □□□ □□□□□. SNS □□□ □□ □□□ □□□□□□.
D. □ □□ □□ □ □□ □□ URL□ □□□ □□□ □□□□ □□□□□ □□□ □□□□□. □ □□□□□□ □□□□□ □□□ □□ □ □ URL□ □□ □ □□□ URL□ □□□□□□□□.
Answer: (SHOW ANSWER)
https://aws.amazon.com/premiumsupport/knowledge-center/sns-lambda-webhooks-chime-slack-teams/

NEW QUESTION: 82
□ □□□ □□ □□□ AWS Systems Manager □□□□ □□□□ □□ □□ □□ □□□□□ □□□□□ □□□□ □□□□. □□ □□ □□ 60□□□ □□□□ □□□□□ □□□.
DevOps □□□□□ □□□□□□□ Amazon ElastiCache(Redis OSS) □□□□□ □□□□ □ □□□ □ □□ □□ □□□□ □□□.
□□ □□□□ □□ □□□ □□□□□□ □□□ □□ □□□ □□□□□?
A. AWS Secrets Manager□□ □□□□ □□□□□. □□□ □□□□□□. □□ □□□ 60□□ □□□□□.
□□□□ □□□□□ □□ □□□ □□□□ □□□□□□□ □□ □□ □□□□□ □□□□□□.
B. □□□□ □□□□ □□ □□ □□□□□. □□ □□□ □□□□□□. □□ □□□ 60□□ □□□□□.
□□□□ □□□□□ □□ □□□ □□□□ □□□□□□□ □□ □□ □□□□□ □□□□□□.

C. □□□□ □□□□ □□□□ □□□□□. □□□□ □□□□□ AWS Lambda □□□ □□□□□. 60□□□ Lambda □□□ □□□□□ Amazon EventBridge □□□□ □□□□□. □□□□□□□ □□□□ □□□□□ □□□□ □□□ □ □□□ □□□ □□□ □□□ □□□ □□ □□□□□.

D. AWS Secrets Manager□□ □□□□ □□□□□. AWS Lambda □□□ □□□□ □□□ □□□ □□□□□□. Secrets Manager□□ □□□□ ElastiCache(Redis OSS) □□□ □□□□ □□□□□. □□ □□□ 60□□ □□□□□. Secrets Manager□□ □□ □□□ □□ □□□. □□□□□□□ □□□□ □□□ □ Parameter Store□ □□ □□ □□□ □□□□□ □□□□□.

Answer: D (LEAVE A REPLY)

Comprehensive and Detailed Explanation From Exact Extract of DevOps Engineer documents only:

The correct answer is D because AWS Secrets Manager supports automatic secret rotation and provides built- in rotation templates for supported services, including Amazon ElastiCache for Redis OSS. This gives the least operational overhead for rotating the secret every 60 days.

The requirement also states that employees and applications must access secrets and parameters through AWS Systems Manager Parameter Store. Parameter Store can reference Secrets Manager secrets by using a reserved path, which allows the application to retrieve the secret through Parameter Store while the secret is actually stored and rotated in Secrets Manager.

Why the other options are incorrect:

A). This option correctly uses Secrets Manager rotation, but it does not specifically use the reserved Parameter Store reference path approach that satisfies the access requirement through Parameter Store as clearly as option D.

B). Parameter Store does not provide native automatic rotation for secrets in the same way that Secrets Manager does.

C). This option is possible, but it requires custom Lambda and EventBridge automation, which creates more operational overhead than using native Secrets Manager rotation with the provided template.

NEW QUESTION: 83

DevOps □□□□□ AWS□□ □ □□□□□□□ □□□□ □□□□ AWS CodePipeline □□□□□□ □□□□□. □□□□□□ □□ □ □, □□ □□, □□ □□□ □□□□□. □□□□□ □□□ □ □□□□□□□ □□□□ URL□ □□□□ 200 OK HTTP □□ □□□ □□ □□□. CodePipeline□ □□□□□□□ □□□ □ □□ □□□□□□ 503 HTTP □□ □□□ □□□□□□□. DevOps □□□□□ □□□ □□□ □□□□ □□□□ □□□□ □□□. □□□□ □□□□ □□□□□□□ □□□ □ 200 OK HTTP □□ □□□ □□□□□ □□□ □ □□□. □□□ □□ □□ □□□ □□□ □□□□□□ □□□□ □□□. DevOps □□□□□ □□□□□□ □□ □□ □□□ CheckURL □□□ □□□□□□. DevOps □□□□□ □□□□ □□□□ □□□□ □□ □□□□ □□□ □□ □□□?

A. AWS CodeDeploy □□□ □□□□□ CheckURL □□□ □□□□□. □□□□□□ URL□ □□ □□□□□ □□□□ □□□ □□□□ CodePipeline□ □□ □□ □□□ □□□□□.

B. Amazon CloudWatch □□□ □□□□□ CheckURL □□□ □□□□□. □□□□□□ URL□ □□ □□□□ □□ □□□□ □□□ □ □□□ CodePipeline□ □□ □□ □□□ □□□□□ □□□ □□□□□.

C. URL□ □□ □□ □□□ □□□□ CodePipeline□ □□ □□ □□□ □□□□ AWS Lambda □□□ □□□□□. CheckURL □□□□ Lambda □□□ □□□□ □□□ □□□□□.

D. URL□ □□ □□ □□□ □□□□ CodePipeline□ □□ □□ □□□ □□□□ Amazon API Gateway HTTP API□ □□□□□. CheckURL □□□□ AWS Device Farm □□□ □□□ □□□□ API Gateway HTTP API□ □□ □□□□□ □□□□□ □□□□□.

Answer: (SHOW ANSWER)

NEW QUESTION: 84

□□ □□ □□□□ □□□ □□□□ □□□ Amazon EC2 □□□□□ □□□ IP □□□ □□□ □ □□□ □□ □□□□ □□□□. □□ □ □□□ □□□□□ □□□ IP □□□ □□□□ □□ □□□□□ □ □□□. □□□□ □□□ □□□ IP □□□ □□□ □□□□ □□ □□ □□□ □□□. □ □□□ □□□ □□□□ □ □□□□?

A. Amazon Athena□ □□□□ AWS CloudTrail □□□ □□□□ □□ □□ □□□ □□□□□. AWS Lambda □□□ □□□□ □□□□□ □ □□□ IP □□ □□□ □□□□ □□ □□ □□□□.

B. □□□□ □□ 1□ □□□ □□ 1□ □□□ □□□□ □□ □□ □□□ □□□□□. □□□ □□ AWS Config □□□ □□□□ □□□ IP □□□ □□□□□□ □□□ □□□ □□□□□ □□□□ □□□ □□□□ □□ □□ □□□□.

C. □□□□ □□□ □□ 1AM □□□ □□ □□ □□□ □□□ □□□□□.
□□□ AWS Lambda □□□ □□□□ □□□ IP □□□ □□□□□□ □□□ □□□ □□□□□ □□□□ □□□ □□□□ □□□□□ □ □□ IP □□□ □□□□ □□□ □□ □□ □□□□.

D. □□ □□□□ □□□□□ □□ □□ □□ □□□ □□□□ EC2 1AM □□□ □□□ □□□□ AWS Config □□□ □□□□□. □□□ □□ □□□ □□□ IP □□□ □□□ □□□□ □□□□□ □□□ □□ □□ □□ □□ □□□□. □□□ □□□ IP □□.

Answer: B (LEAVE A REPLY)

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. References: from Amazon DevOps sources

NEW QUESTION: 85

□ □□□ AWS Systems Manager□ □□□□ □□□□ AWS □□□□ □□□ Amazon EC2 Linux □□□□□ □□□□ □□□□. □□□ EC2 □□□□□ □□ □□ □□ □□ □□□ □□□□ □□□.

□□ DevOps □□□□□□ □□ □□□ □□□□ □□□□ □□ Chef □□□ □□□□□□. □□□ Chef □□ □ □□□□ □□□□ □□ □□ □□□□ □□□□.

DevOps □□□□□ Chef □ック□□ EC2 □□□□□□ □□□□□ □□□□□ □□□□ □□□.

□□ □□□□ □□□ □□ □□□ □□□□□?

A. □□□ □□□ □□ □□□ □□□ □□□□□. □□ EC2 □□□□□ AWS-ApplyChefRecipes □□□ □□□□□. □□ □□□□□ Chef □ック□□ □□□□ □□□ □□□□□ □□□ □□□□□.

B. Chef □□□□ □□ □□□□ Amazon S3 □□□ □□□□□. □□ □□□□□□ AWS-InstallApplication □□□ □□□□ □□ □□□ □□□□□ Systems Manager □□ □□□ □□□□□. □□ □□□ □□□ □□□□□ □□□□□.

C. □□ EC2 □□□□□ AWS-RefreshAssociation □□□ □□□□ Systems Manager □□ □□□ □□□ □□□□□. □□□ □□□ □ □□□□ □□□□□.

D. Systems Manager □□ □□□ □□□□ □□□ □□ □ □□ □□□ □□□□□. EC2 □□□□□ □□ □□ □□□□ □□□□□. □□ □□□ □□□ □□□ □□ □□□□ IAM □□□□ □□□□□.

Answer: A (LEAVE A REPLY)

Option A directly matches all requirements with the least extra infrastructure:

* State Manager is the Systems Manager capability for defining and maintaining a desired configuration by running associations on a schedule (for example, hourly). That satisfies the "run periodically" requirement in a managed way across a large fleet.
* The AWS-ApplyChefRecipes Systems Manager document is specifically intended to run Chef recipes

/cookbooks on managed instances without requiring you to run your own Chef server infrastructure . You can point it at a cookbook source (such as an artifact/repo location) and have Systems Manager handle execution on the instances.
Why the other options aren't correct:
* B is not the right mechanism for periodic Chef cookbook execution. Installing an application package and running a "repair action" via Run Command is not the purpose-built Chef cookbook runner, and it's more brittle/DIY than State Manager associations.
* C (AWS-RefreshAssociation) is used to refresh/reevaluate association metadata/targets; it does not itself execute Chef cookbooks as the compliance mechanism.
* D is for OS patching compliance (Patch Manager/patch policies), not for applying Chef cookbooks that detect and remediate configuration drift.

NEW QUESTION: 86
DevOps □□□□□ AWS WAF□ □□□□ AWS □□ □□□ □ ACL□ □□□□□. DevOps □□□□□ □□ □ □□ □□□□□□ □□ □□□(ALB)□ AWS WAF□ □□□□□ □□□ □□□□ □□□. DevOps □□□□□ AWS CloudFormation □□□□ □□□□ □ □□□ □□□ □□□ □□ □□□□□ □□□ □□ ALB□ AWS WAF□ □□□□□. ALB □□ □ AWS WAF□ ALB□□ □□□□ AWS WAF□ ALB□ □□□□ □□□□□ □□□.
□□ □□□□ □□ □□ □□ □□□□□ □□□ □□ □□□ □□□ □ □□□□?
A. AWS Config□ □□□□□□. alb-waf-enabled □□□ □□□ □□□□□. AWS Systems Manager Automation □□□ □□□□ ALB□ AWS WAF□ □□□□□. □□□ □□□□ □□□□ □□□□□. Systems Manager Automation □□□ □□ □□□□ □□□□□.
B. AWS Config□ □□□□□□. alb-waf-enabled □□□ □□□ □□□□□. □□ AWS Config ConfigurationItemChangeNotification □□ □□□ AWS Lambda □□□ □□□□ Amazon EventBridge □□□ □□□□□. Lambda □□□ □□ □□□□ AWS Config start-resource-evaluation API□ □□□□□ □□□□□.
C. CloudFormation □□□□□ detect-stack-drift API□ □□□□ AWS Lambda □□□ □□□□□ □□□□□ Amazon EventBridge □□ □ □□□□□. AWS::WAFv2::WebACLAssociation □□□□ 'drifted' □□□ □□□□ □□, Lambda □□□ waf.fail_open.enabled□ true □ □□□□ ALB □□□ □□□□□ □□□□□.
D. CloudFormation □□□□□ detect-stack-drift API□ □□□□ AWS Lambda □□□ □□□□□ □□□□□ Amazon EventBridge □□ □ □□□□□. AWS::WAFv2::WebACLAssociation □□□□ 'drifted' □□□ □□□□ CloudFormation □□□ □□□□ □□ □□□□□ Lambda □□□ □□□□□.
Answer: A (LEAVE A REPLY)
AWS Config has a managed rule called alb-waf-enabled that checks whether AWS WAF is enabled on ALBs.
AWS Config supports automatic remediation actions that can be triggered when noncompliance is detected.
By creating a Systems Manager Automation document that adds AWS WAF to the ALB and associating it as the remediation action for the AWS Config rule, the system can automatically detect and remediate any removal of AWS WAF from ALBs without manual intervention.
This is the most operationally efficient and reliable approach to ensure continuous compliance.
Option B lacks automatic remediation. Options C and D rely on drift detection and Lambda, which add complexity and risk downtime during stack replacement.
Reference:
AWS Config Managed Rules: " The alb-waf-enabled rule checks for AWS WAF association with ALBs and supports automatic remediation using Systems Manager Automation. " (AWS Config Managed Rules) AWS Config Remediation: " AWS Config automatic remediation can invoke Systems Manager Automation documents to remediate noncompliance. " (AWS Config Remediation)

NEW QUESTION: 87

□ □□□ AWS Organizations□ □□□ □□□□ □□ AWS □□□ □□□□□. □□□ □□□ □□ □□□ □□□□□□□. AWS CloudFormation StackSets□ □□□□ □□□ □□□ □□□□□. □□ AWS Config□ □□□□ Amazon S3 □□□ □□□□□□□. □□ □ S3 □□□ □□□□□ □□ □□□ AWS Key Management Service(AWS KMS) □□□□ □□□□ □□□. □□□ □□ □□□ □□□ □ □□□□ □□□□□?

A. s3:createBucket □□□ □□ □□ □□□□ s3:x-amz-server-side-encryption□ aws:kms□ □□ □□□□ □□□□ SCP□ □□□□□. SCP□ □□□ □□□ □□□□□.

B. s3:putObject □□□ □□ □□ □□□ s3:x-amz-server-side-encryption□ aws:kms□ □□ □□□ □□□□ SCP□ □□□□□. SCP□ □□□ □□□ □□□□□.

C. s3-bucket-server-side-encryption-enabled □□□ □□□□ AWS Config □□□ □□ □□□□□. □□□ □□ □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□ □□□□□.

D. □□□ S3 □□□ □□□□ □□□ □ □□□ AWS CloudTrail □□□□ □□□□□ AWS CloudFormation □□ □□□ □□□□□. □ □ □□□□ AWS KMS □□□□ □□□□ □□ S3 PutObject □□□□ □□□□ Amazon EventBridge □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□ □□□□□.

Answer: B (LEAVE A REPLY)

NEW QUESTION: 88

□ □□□ AWS □□□□ □ □□□□□□□ □□□□□ □□□□. □□ □□□ □□ □ □□ □□□ □□ □□□□ □□□□ □□□□. □ □ □□ □□ □□□ □□□ □□ □□□□□ □□ □□ □□ □□□□ □□□□ □□ □□□ □□□ □□□□□ □□□□ □□ □□□ 1% □ □□ □□□□ □□□□□ □□□. □□ □□ □□□□ □□□ □□□ □□□□ □□□□ □□□□ □□ □□□□ □□□□□□ □□ □ □ □□□ □□ □□□□ □□□ □ □□□ □□□ □ □□□ □□□.

DevOps □□□□□ □□□ □□ □□□ □□□ □□□□ □□□?

A. □ □□□□ Amazon API Gateway□ □□ □□□□ AWS Lambda□ □□□□□□□ □□□□ □□ □□□□ □□ □□ □□ □□□ □ □ PostgreSQL□ Amazon RDS□ □□□□□. API □□□□□□ □□ □□□□ □□ □□□□□□ □□□ □□ □ □□□□□□□ □□□ □□.

B. □ □□□□ Auto Scaling □□□ □□□□□□□ □□□□ □□ □□□□ DynamoDB□ □□□□□. □□ □□□ □□ Route 53 □□ □□ □□□ □□□ □□□□ □□ □□□ □□□□ □□□□□.

C. □ □□□□ AWS Elastic Beanstalk□ □□□□□□□ □□□□ □□ □□□□ Amazon DynamoDB □□ □□□□ □□□□□. □□ □ □□ □□ Amazon Route 53 □□ □□□ □□□ □□□□ □□ □□□ □□□□ □□□□□.

D. □ □□□□ Auto Scaling □□□ □□□□□□□ □□□□ □□ □□□□ DynamoDB □□ □□□□ □□□□□. □□ □□□□ Amazon CloudFront □□ □□□ □□□□□□. CloudFront □□□□ Amazon Route 53 DNS □□□□ □□□□□.

Answer: D (LEAVE A REPLY)

NEW QUESTION: 89

□□□ □□□□□□□ □□ □□ AWS □□□ □□□□ □□□□. □□□ □□□ □□ □□□ □ □□□□ □□□□□. □ □□□□ □□ □□ □ □□ □□□ □□□□ □□□□. □ □□□ □□□□□□□ ALB(Application Load Balancer) □□ Amazon EC2 □□□□□□ □□ □□ Amazon DynamoDB□ □□□□□□ □□□□ □□□□□. □□□□□ □□ □□ □□□ □□ EC2 Auto Scaling □□□□ □□□□ □. DevOps □□□□□ □□□□□□ □□ □□□ □□□□□ □ □□□□ □□□□ □□□□ □□□□□ □□□ □□ □□□□.

□□ □□ □□□ □□□□□ □□ □□□ □□□ □□□ □□□? (3□□ □□□□□.)

A. □□ □ □□□ □□□□ □ □□□□ □ DynamoDB □□□□ □□□□□.

B. □ ALB □ Auto Scaling □□ □□□ □□□□ □□□□ □□□□ □ Auto Scaling □□□□ □□□□□ □ ALB□ □□□□□.

C. □ □□□□ □ ALB □ Auto Scaling □□ □□□□ □□□□ □□□□ □ Auto Scaling □□□□ □□□□ □ ALB□ □□□□□.
D. ALB□ □□□□ Amazon Route 53 □□□, □□ □□ □ □□ □□ □□ □□□ □□□ □□□□□.
E. ALB□ □□□□ Amazon Route 53 □□, □□ □□ □ □□ □□ □□□ □□□ □□□□□.
F. DynamoDB □□□□ □□ □□□□ □□□□□.
Answer: C,D,F (LEAVE A REPLY)
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

NEW QUESTION: 90
□ □□ □□□ □□□ □□ □□□ □□□□□□ □□ □□□□□□□ □□□□□ □□□□ □□ □□□ □□□□ □□□□. □□□ □□ □□□□□ Amazon EC2 □□ □□□□□□ □□□□□ □□□□ □□ □□ □□□□□ □□□□□ □□□. DevOps □□□□□ □□ □ □□ □□□□ □□ □□□□□□□ □□□□ □□□□□ □□□□ □□□.
□□□□ □□ □□□□□ □ □□ □□□ □□□□□ □□□□□ □□ □□□ □□□□ □□□?
A. AWS Systems Manager □□ □□ □□□ □□□□□. put-compliance-items API □□□ □□ □□□ □□□□ □□□ □□ □□□ □□ □□ □□□ EC2 □□□□□ □□□□□□□ □□□□ □□□□□. □□ □□□□ □□ Amazon DynamoDB □□□□ □□□□ □□□ □ □□□ ID□ □□□□□. list-compliance-summaries API □□□ □□□□ Systems Manager□ □□ □□□□ □□□□□.
B. EC2 □□□□□□ □□□□ □□□ □□ Java □□□ □□□□□. □□□ □□□□ □□ □□ □□□□□ □□ EC2 Auto Scaling□ □ □□□□. □□□ EC2 □□□□ ID □□□ Amazon SQS □□□□ □□□□. SQS □□□□ □□□□ ID□ □□□□ □□ Amazon DynamoDB□ □□□ □□ □□□ □□□□□ □□□□□. AWS Lambda □□□ □□□□ □□□□□ □□ □□□ □□□□ ID□ □□□ □ □□□ □□ Amazon SNS □□□ □□□ □□□□.
C. AWS Config□ □□□□□. □□ □□□ □□ Amazon EC2 □□□□ □□ □□ □□□ □□□□□ □□□ □□ EC2 □□□□□ □□□ □□. "config-rule-change-triggered" □□□□□□ □□□□ AWS Lambda □□□ □□□□□ □□□ □□ AWS Config □□□ □□□□ □. □□□□□ EC2 □□ □□□□□ □□□□ □□ □□ NON_COMPLIANT □□□ □□□□□ □□□ □□□ □□□□□ Lambda □□ □□() □□□ □□□□□. □□□ □□□□□ □□□□□ AWS Config □□□□ □□□□□□.
D. AWS CloudTrail□ □□□□□. EC2 RunCommand API □□□ □□ □□ □□□ □□□□ □□□ □□ EC2 □□□□□ □□□□□. □ □□□□ □□□ □□□ □□□□ AWS Lambda □□□ □□□□□. □□□ □□□□ EC2 □□□□ ID□ MySQL□ Amazon RDS DB □ □□□□ □□□□□. RDS □□□□□ □□□□ □□ □□□ CSV □□□ □□□ □□□ □□□□ □□□□□.
Answer: (SHOW ANSWER)
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.

NEW QUESTION: 91
□ □□□□ □ □□□□ □□□ □□□□ □□ □□□□ □□□ Amazon S3 □□□ □□□□ □□□□□ □□□□. □□ □□□□ S3 □ □□ □□□□ □ □□□□.
□ □□□□ □□ ID□ □□□□ □□□□. □□□□□ □□□□ □□ □□□ □□ ID□ □□ UUID□ □□□□□. □□ ID□ 12□□□□.
□□□□□ □□□□ □□ □□□ ID□ □□ P□ □□□□□.
□□□□□ □□□□□ □□□□ □□ □□□ □□ □□□ □□□□ □□ S3 □□□□ □□□□ □□ □□□ □□□□.
□□ □□□□ □□□ □□ □□□ □□□□□□?
A. □ □□□□ S3 □□□ □□□□□. □ □□□□ □□□ □□ □□□ □□□□□. □ □□□□ □□ UUID□ □□□□ □□ ID□ □□□ □ □□ Amazon Macie □□□ □□ □□□ □□□□ □□□□□.
□□□ □□ □□□ □□□□ □□□□ Amazon Macie □□ □□□ □□ □□□ □□□□□. Macie □□ □□□ □□ □□ □□□ □□□□ S3 □□□□ □□□□□.
B. □□□□ □□□□ □□ UUID□ □□□□ □□ ID□ □□□□ □□ Amazon Macie □□□ □□ □□□ □□□□ □□□□□. □□□ □ □ □□□ □□□□ □□□□ Amazon Macie □□ □□□ □□ □□□ □□□□□. □□□□ S3 □□□□ Macie □□□ □□ □□ □□□ □□□□□.
C. □ □□□□ S3 □□□ □□□□□. □ □□□□ □□□ □□ □□□ □□□□□. □□□ □□□ □□□□ □□□□ Amazon Macie □□ □□□ □□ □□□ □□□□□. Macie □□ □□□ □□ □□ □□□ □□□□ S3 □□□ □□□□□.
D. □□□ □□□ □□□□ □□□□ Amazon Macie □□ □□□ □□ □□□ □□□□□. Macie□□ □□□ □□ □□□ □□□□ S3 □ □□□ □□□□□.
Answer: (SHOW ANSWER)
The requirement is to prevent any product files containing unannounced product IDs (prefixed with a specific UUID) from being stored in the production S3 bucket that users can access.
To achieve this, a best practice is to use a staging bucket as a control point before files go to production, combined with Amazon Macie's data classification capabilities.

* Creating a custom data identifier in Amazon Macie allows precise detection of product IDs starting with the specific UUID, which default managed identifiers will not detect.
* By running a Macie sensitive data discovery job on the staging bucket, you can identify files containing these sensitive product IDs.
* Only files without findings (i.e., files that do not contain unannounced product IDs) are copied to the production bucket, ensuring no sensitive information is exposed. This approach aligns with AWS best practices for data classification and staged deployment workflows, maximizing control and reducing risk. Using Macie on the production bucket directly (options B and D) risks exposing sensitive data before detection and deletion. Option C uses managed data identifiers, which will likely not detect the custom UUID prefix pattern.
Reference from AWS Official Documentation and Study Guide:
* Amazon Macie Custom Data Identifiers: " You can create custom data identifiers in Amazon Macie to find sensitive data that is unique to your organization. " (Amazon Macie User Guide)
* Data Security Best Practices: " Use staging environments to inspect and sanitize data before moving it to production to reduce exposure risks. " (AWS Security Best Practices)

NEW QUESTION: 92
□□□□ □□□ □□ □□ □□□□ AMI□□ □□ Amazon EC2 □□□□□ □□□□ □□□ □□□ □□□□. □□ □□ □□ □□ □□ AMI□ □□□ □□□ □□□□ □□ □□ □□ □□□□.
□□ □□ AWS CloudFormation□ □□□□ □□□□□□□ □□□□□. □□□□ □ □□□□ □□□□ □□ □□□□ □□ □□ AMI□ □□ □□□□ □□□□ □□□. DevOps □□□□□ □□ □□ □□ □□ AMI ID□ □□□□ □ □□□□ □□□□□ □□□□□□ □□ □.
□□□ □□ □□□ □□□□ □□ □□ □□□ □□□□ □□□□□?
A. □□ □□ CloudFormation□ □□□□ □ □□□ AMI□ □□□□ □□□□□ □□□□□□! □□ □□ □□□ □□□ □□□□ Amazon S3 □□□ AMI ARN □□□□□ □□ □□ □□□ □□□□ □□□□ S3 □□□ □□□□ □□ AMI ARN□ □□□□□ □□□□□.
B. □□ □□ CloudFormation □□□ □□□□ □□□ AMI□ □□□□ □□ AMI ARN□ □□□□□ □□□ □□□ □□□□ Amazon S3 □□□ □□□□ AWS CodePipeline □□□□□□ □□□□□ □□□□□□. □□□□□ □□ □□□ □□□□□ □□□□□□. □□ CloudFormation □□□ □□□ □□□□ S3 □□ □□ □ □□ AMI ARN□ □□□□□.
C. □□ □□ Amazon EC2 Image Builder□ □□□□ □ AMI□ □□□□ AMI ARN□ AWS Systems Manager Parameter Store□ □□□□ □ □□□□□ □□□□□□. Parameter Store□ □□ AMI ARN.
D. □□ □□ □□ □□□ □□ □ □□□ □□ □□ Amazon EC2 □□□ □□□ □□□□ □ AMI□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□□□ □□□□□□. □□ □□ □□□ □□□ □□ □□□ AMI ARN□□ CloudFormation □□□ □□ □□□ AWS Lambda □□□ □□□□□ □□□□□.
Answer: C (LEAVE A REPLY)
https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/dynamic-references.html

NEW QUESTION: 93

□ □□□□ □□ □□ Amazon EC2 □□□□□ □□□□□ □□□□□ □□□. □□□ □□ □□□□ □□□□ Auto Scaling □□□ EC2 □□□□□ □□□□□□□□.
□□□□□ Amazon S3 □□□□ □□□ □□□ □□□□ □□□□ □□□ □□ S3 □□□ □□□□□. EC2 □□□□□□ □□ □□ □□ □ □□□ □□ □□ □□ □□ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. S3 □□□ □□ □□□ □□□ □□ IAM □□□ □□□□□. □□□□ □□□□ IAM □□□ □□□□□.
B. IAM □□□□ □□□□ □□□□□ □□ □□□□ □□□□□□□.
C. Amazon S3□ □□ □□□ □□□ □□ IAM □□□□ □□□□□. □□ □□ □□□ □□□□□.
D. □□□□ □□□ □□□□ □□□□□. IAM □□□ □□□□ □□□□□.
E. □□ □□□□ □□□□□□□. □ □□ □□ □□□ □□□□□ □□□ □□□□ □□□□□.
Answer: (SHOW ANSWER)
To meet the requirements of deploying a workload on several hundred EC2 instances with least-privilege 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.
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

NEW QUESTION: 94
□ □□□□□ □□□□ □□ □□ □□□□ □□ □□ □□ □□□ □□□□ □□□ □□□ □□ □□□□. □□ □□ □□□□ □□□ □ □□□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□□ Amazon Simple Queue Service(Amazon SQS) □□□□ □□ □□□ □ □□□□ □□□ □□□ Amazon DynamoDB □□□□ □□□□□. DynamoDB □□□□□ □□ □ □□ □□□ □□ Auto Scaling□ □ □□□□ □□□□.
DevOps □□□□□ □ □□□ □□□□ □□ □□ □□□ □□□ □□□? (□ □□□ □□□□□.)
A. SQS □□□□ □□ ApproximateAgeOfOldestMessage □□□ □□□□□□. Lambda □□ □□□ □□□ □□□□.
B. SQS □□□□ □□ ApproximateAgeOfOldestMessage □□□ □□□□□□. SQS □□□□ □□□□□ □□□ □□□□□□.
C. SQS □□□□ □□ NumberOfMessagesSent □□□ □□□□□□. SQS □□□ □□□ □□ □□□ □□□□.
D. DynamoDB □□□□ □□ WriteThrottleEvents □□□ □□□□□. □□□□ □□ □□□ □□ □□ □□ □□ □□(WCU)□ □□□□.
E. Lambda □□□ □□ Throttles □□□ □□□□□. Lambda □□ □□ □□□ □□□□.
Answer: (SHOW ANSWER)
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

NEW QUESTION: 95
DevOps □□□□□ AWS CodeBuild□ □□□□ □□□□□ □□□□ □□ □□□□□. CodeBuild □□□□□ DevOps □□□□□ □□ □□□□ □□□ □ □□ □□□ Docker □□□□ □□□□□. DevOps □□□□□ □□ □□□ □□□□ □□□ □□□□□□ □□□. □ □ □□□□ □□□ □□ □□□ □□□□□?
A. Docker □□□□ Amazon Elastic Container Registry(Amazon ECR) □□□□ □□□□□.
CodeBuild□ □□ □□ Docker □□ □□□ □□□□□.
B. □□ □□ □□□□□ □□ □□□ Amazon S3 □□□ Docker □□□□ □□□□□. S3 Lifecycle □□□ □□□□ □□□ □□□□□.
C. Docker □□□□ Amazon Elastic Container Registry(Amazon ECR) □□□□ □□□□□. CodeBuild □□□□ □□□ □□□ □□□□ □□ □□ □□□ □□□ □□□□□.
D. □□□ Docker □□□□ □□□□ □□□ □□ AMI□ □□□□. CodeBuild □□□□ □□□ □□ AMI□□ Amazon EC2 □□□□□ □ □□□□.
Answer: A (LEAVE A REPLY)
Step 1: Storing Docker Images in Amazon ECRDocker 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 CodeBuildDocker 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.

NEW QUESTION: 96
□□□ □□□ AWS Control Tower□ □□□□ □□ AWS □□□ □□□□□. □□□ □□ □□□□□□□ □□ □□□□□□□ □□□□ □□.
□□□ □ □□□□□□ □□□ □□□□□□ □□□□ □□ □□ AWS □□□ □□□□. □□□ AWS Organizations□ □□□ □□□□ □. AWS Control Tower □□ □□ □ □□□ □□□□□□ □□ □□ □□ AWS □□□ □□□□□□□ □□□□ □ □□□□ CI/CD □□ □□□□ □□ □□ □□□ DevOps □□ □□□ □□□. □□□ □□ 1AM □□□ □□ □□□ DevOps □□□ □□□□.
□□□□□□ □□ □□□□□□ AWS □□□ Amazon Elastic Kubernetes Service(Amazon EKS) □□□□□ □□□□□□□ □□□□□ □□□. □□□□□□ AWS □□□ □□□ □□ 1AM □□□ □□□□. □□□ □□ □□□ DevOps □□□ □□□ AWS CodeBuild □□□ □□ □□ □□□□□□. CodeBuild □□□□□ CodeBuild□ □□ □□ 1□ □□□ □□□ □□□□□. CodeBuild□□ □□ □□ EKS □ □□□□ □□□ □□□□ □□ □□ □□□ □□ □□□ □□□□□?
A. □□ □□□ DevOps □□□ □□ □□□ □□□ □□□□□□ □□□ □□ 1AM □□□ □□□□□. sts:AssumeRole □□□ □□□□ □ □□ □□□ □□□□□. EKS □□□□□ □□ □□ □□□ □□□ □□□ □□□□□□ □□□ □□ 1AM □□□ □□□□□. □□□ □□□ □□□ □□□ □□□□□ EKS □□□□ aws-auth ConfigMap□ □□□□□.
B. □□□□□□ □□□ □□ □□□ □□□ □□ □□□ DevOps □□□ □□ I AM □□□ □□□□□. sts:AssumeRole □□□ □□□□ □ □□ □□□ □□□□□. CodeBuild□ □□ □□ □□□□ □□□□□ □□ □□□ DevOps □□□ □□ □□ 1□ □□□ □□□□□.
C. □□□□□□ □□□ □□ □□□ □□□ □□ □□□ DevOps □□□ □□ 1AM □□□ □□□□□. sts:AssumeRoleWithSAML □□□ □□□□□ □□ □□□ □□□□□. CodeBuild□ □□ □□ □□□□ □□□□□ □□ □□□ DevOps □□□ □□ □□ 1□ □□□ □□ □□□.
D. AWS Control Tower □□ □□□ □□ □□□ □□□ □□□□□□ □□□ □□ 1AM □□□ □□□□□. sts:AssumeRole □□□ □□ □□□ □□ □□□ □□□□□. EKS □□□□□ □□ □□ □□□ □□□ □□□ □□□□□□ □□□ □□ 1AM □□□ □□□□□. □ □□ □□□ □□□ □□□ □□□□□ EKS □□□□ aws-auth ConfigMap□ □□□□□.
Answer: A (LEAVE A REPLY)
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.

NEW QUESTION: 97

□ □□□ AWS CodePipeline□ □□□□□□ □□□□ □□□□□□□ □□□□□. □ □□□ □□□□□□□ □□□□ □□□□□ □□ AWS Fault Injection Service(AWS FIS) □□ □□□□ □□□□□□. DevOps □□□□□ □ □□□ □□□□□□ □□□□ □□□.
□□ □□□□ □ □□ □□□ □□□□□□?

A. AWS FIS □□ □□□□ Amazon S3 □□□□ □□□□□. AWS FIS □□□ □□□□ □□ □□□ □□□ AWS CodeBuild □□ □□□ □□□□□ □□□□□. CodeBuild □□□□□ □□□ □□□ □ □□ □□□ □□□□□. CodeBuild □□ □□□ □□□□□ □□□□ □ □□ □□□ □□□□□□ □ □□□ □□□□□.

B. Amazon EventBridge □□□□□ □□□□□. □□□□□ AWS FIS □□□ □□□ □ □□ □□□ □□□□□. □□□□□□ EventBridge □□□□□ □□□□ □□□ □□□ □ □□□ □□□□□.

C. AWS FIS □□□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□□ □□ □□ □□□ □□□□□. □□□□□□ Lambda □□ □ □□ □ □□□ □□□□□. Lambda □□□ □□□□□ □□□ □□□□□.

D. AWS FIS □□□ □□□□ □□□□□□ □ □□□ □□□□□. AWS FIS □□ □□□□ □□□□□ □□□ □□□□□. □□□□□□ □□□ □□□ □ □□ □□□ □□□□□.

Answer: C (LEAVE A REPLY)

NEW QUESTION: 98

□ □□□ Amazon EKS□□ □□□□□□□ □□□□□□□ □□□□ □□□□. □□□□□ □□□□ □□ □□□□ □□ □□ □□ □□ □ □□□□□ □□□□□. CloudWatch □□□ □□□ EKS □□□ CPU □ □□□ □□□□ □□□□ □□□□□. DevOps □□□□□ □ □□□□□□ □□□ □□□ □□□□ □□□ □□□ □ □□□□.
□□ □□□□ □□□ □□ □□□ □□□□□□?

A. AWS X-Ray □□□ EKS □□□□□ DaemonSet□□ □□□□□. X-Ray SDK□ □□□□ □□□□□□ □□□ □□□□□. □□□□ □□□ □□ □□□□□.

B. EKS □□□□□ □□ CloudWatch Container Insights□ □□□□□□. Container Insights □□□□ □□□□ □□□ □□□□□.

C. □□ CloudWatch □□□ □□□□ □□□ □□□□□. SNS □□□ □□□ □□□□□.

D. □□□□ □□□ □□ □□□□□□ □□□ □□ □□ □□□ □□□□.

Answer: A (LEAVE A REPLY)

* AWS X-Ray provides distributed tracing, which allows visualization of latencies and errors within microservices, pinpointing bottlenecks or delays.
* Instrumenting application code with the X-Ray SDK and running the X-Ray daemon as a DaemonSet in EKS ensures tracing data is collected cluster-wide.
* Container Insights (Option B) provides resource-level metrics but not detailed request tracing.
* CloudWatch alarms and alerts (Option C) detect symptoms but don't provide root cause tracing.
* Increasing timeouts (Option D) only masks the issue and does not diagnose it.

References:
AWS X-Ray for Amazon EKS
Distributed Tracing in Microservices

NEW QUESTION: 99

DevOps □□□□□ AWS CDK(AWS Cloud Development Kit)□ □□□□ □□□□□□□ □□ □□□□□□□ □□□ □□□(IaC)□ □ □□□□ □□□. DevOps □□□□□ □□ □□□ □□□ □□ □□□ □□□ □□ □□□ □□□□, □□ □□□□□□□□ □□□ □□ □□ □□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?

A. □□ □□□ □□□ □□□□ □□□ □□ CDK □□ □□□□□□ □□□□. CDK □□ □□□□. TagManager □□□□ □□□□ □□ □□ □□ □□ □□□ □□□□□. □□□ □□ CDK □□ □□□□□□ □□□□ □□ □□□□□□□□ □□□□ □□ □□ □□□ □□ □□□. □□□□□□□□ □□□ □□□ □□□□ □□ CDK □□□□ □□□□□.
B. □□ □□□ □□□ □□□□ □□□ □□ CDK □□ □□□□□□ □□□□. CDK □□ □□□□. Tags □□□□ □□□□ □□ □□ □ □ □□ □□□ □□□□□. □□□ □□ CDK □□ □□□□□□ □□□□ □ □□□□□□□□ □□ □□ □□ □□□ □□□□□. □□ □□□□□□ □□□ □□□ □□□□ □□□ CDK □□□□ □□□□□.
C. □□ □□□ □□ □□□ □□□□ AWS □□□ □□□□ □□□ □□□□□. CDK □□ □□□□□. TagManager □□□□ □□□□ □□ □□ □□ □□ □□□ □□□□□. □□□ □□□□ □□□ □□□□ □□ □□□□□□□□ □□□□ □□ □□ □□□ □□□□□. □□□□□□□□ □□□ □□□ □□□□ □□ CDK □□□□ □□□□□.
D. □□ □□□ □□ □□□ □□□□ AWS □□□ □□□□ □□□ □□□□□. CDK □□ □□□□□. Tags □□□□ □□□□ □□ □□ □□ □□ □□□ □□□□□. □□□ □□□□ □□□ □□□□ □ □□□□□□□□ □□ □□ □□ □□□ □□□□□. □□□□□□□ □ □□□ □□□ □□□□ □□□ CDK □□□□ □□□□□.

Answer: (SHOW ANSWER)

Use custom constructs to encapsulate reusable patterns and deploy separate stacks per microservice for better isolation and environment-specific params. Apply tags programmatically at the app/stack/construct scope with Tags so all resources inherit them. This is the recommended CDK pattern to minimize duplication and maximize reusability.

NEW QUESTION: 100

□ □□□ AWS □□ □□□ □□□□ □□ □□□ □□□□ □□□□. □ □□□□ AWS Organizations□ □□□ □□□□. □□□ □□□ □□ □□ □ □□ □□□ □□□ IAM □□□ □□□ □□□□ □□□. □□□ □□ □□□ □□ □□□□□ □□□□□ □□ □□□□ □ □□□ □□□□ □□□□□?

A. □□ □□□□ AWS CodePipeline □□□□□□ □□□□□. AWS CloudFormation□ □□□□ □□□ IAM □□ □ □□□ □□□ □□ □□□□ □□□□□□ □□□□□. CodePipeline□ □□ □□□ □□□□ □□□ □ □□□ □□ □ IAM □□□ □□□□□.

B. □□ □□□□ AWS Control Tower□ □□□□□□. □□ □□□ □□□ IAM □□□ □□□ □□□□□ AWS Control Tower Account Factory □□□ □□□ □□□□□.

C. □□□ AWS CloudFormation StackSets□ □□ □□□ □ □□ □□□□ □□□□□□. □□ □□□□ □□ □□□ □□□ IAM □□□ □□□ □□□ □ □□ □□□ □□ □□□ □□ □□ □□□ □□□□□.□□ □□□ □□ □□ □□□ □□□□□□.

D. □ □□ □□□□ □□□□ □□□□ □□□ □ □□ □□□ □□ IAM □□□ □□□□□. □□ □□□□ □□ □□□ □□□ IAM □□ □ □□□ □□□ □ □□ □□ □□ □□□ □□ AWS CloudFormation □□ □□□ □□□□□. □□ □□□ □□ □□ □□□ □□□□□ □.

Answer: (SHOW ANSWER)

NEW QUESTION: 101

□□□ □□□□□□ □□ □□ Linux □□ Amazon EC2 □□□□□ □□□ □□□□ □□□□□. □□□ □□□□ □□ □□□□ SSH □ □□□ □□□ □□ □□□ □□□ □□ □□ IP □□□ □□□□□. □□□ □□ □□ □□ IP □□□□ SSH □□□□ □□□□□ □□ □ □ □□□ □□□ □□ □□□ □□□ □□□□.

DevOps □□□□□ □ □□ □□□ □□□□ □□ □□□ □□ □□□?

A. □□□ aws.cloudtrail□□ □□□ □□□ AuthorizeSecurityGroupIngress□ Amazon EventBridge □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□□□.

B. Amazon GuardDuty□ □□□□□ AWS Security Hub□□ □□ □□□ □□ □□□ □□□□□. GuardDuty □□□□ NON_COMPLIANT □□□ □□□□□ □□□ □□ □□□□ Amazon EventBridge □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□□□.
C. limited-ssh □□□ □□□ □□□□ □□ □□□ □□ □□ □□□□ SSH □□□□ □□□□ □□□ □□□ □□□□ AWS Config □□ □ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□□□ □□ □□□ □□□□□.
D. Amazon Inspector□ □□□□□□. Common Vulnerabilities and Exposures-1.1 □□ □□□□ □□□□ □□□ □□□□ □□□ □□ □□□ □□□□□. Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□□□□ Amazon Inspector□ □□□□□.
Answer: A (LEAVE A REPLY)
https://aws.amazon.com/premiumsupport/knowledge-center/monitor-security-group-changes-ec2/

NEW QUESTION: 102
DevOps □□□□□ □□□□ □□□□ Amazon Elastic Container Registry(Amazon ECR) □□□□□□ □□□□ AWS CodePipeline □ □□□□□ □□□□ □□□. □ □□□□□□ □□ □□□ □□□ □□□ □□□□ □□, AWS CodeConnections□ □□□ Git □□□□□ □ □ Git □□□ □□□ □ □□□□□ □□□. □□ □□ □□□□□□ □ □□□□ □□□ □□ □ □□□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. QUEUED □□□ □□□□ CodePipeline V2 □□ □□□□□□ □□□□□. □□ □□□ □□□□ □□□ □□□ □□□□□ □□□ □ □□□□. □□□□ □□□ □□□ □□□□ Amazon EventBridge □□□ □□□□ □□ □□ □□□□□□ □□□□□.
B. SUPERSEDED □□□ □□□□ CodePipeline V2 □□ □□□□□□ □□□□□. □□ □□□□ □□□□ □□□ □□□ □□□□□ □□□ □□□□□. □□□□ □□□ □□□ □□□□ Amazon EventBridge □□□ □□□□ □□ □□ □□□□□□ □□□□□.
C. SUPERSEDED □□□ □□□□ CodePipeline V1 □□ □□□□□□ □□□□□. □□ □□□ □□□□ □□□ □□□ □□□□□ □ □□ □□□□□. □□ □□ □□□□□□ □□□□ □□□ □□□□□ □□ □□□□□.
D. QUEUED □□□ □□□□ CodePipeline V1 □□ □□□□□□ □□□□□. □□ □□□□ □□□□ □□□ □□□ □□□□□ □□□ □□□□□. □□ □□ □□□□□□ □□□□ □□□ □□□□□ □□ □□□□□.
Answer: (SHOW ANSWER)
The requirements clearly indicate the need for modern CodePipeline capabilities , strict execution ordering, Git tag-based triggers, and loose coupling between pipelines. CodePipeline V2 introduces execution modes such as QUEUED and SUPERSEDED , along with native support for advanced trigger filtering when using AWS CodeConnections.
The requirement that the pipeline must wait for the previous run to finish directly maps to QUEUED mode , which ensures that pipeline executions run sequentially rather than replacing in-progress executions.
SUPERSEDED mode would cancel the running execution, which violates the requirement.
Triggering the pipeline when new Git tags are pushed is supported through CodePipeline V2 trigger filters using refs/tags/*. Branch-based triggers would not satisfy this condition.
Finally, the requirement that an existing deployment pipeline runs in response to new container images is best met using Amazon EventBridge , which natively emits events for ECR image push actions . EventBridge allows decoupled, event-driven orchestration between pipelines without tight dependencies or custom scripting. This is the AWS-recommended approach for pipeline-to-pipeline coordination.
Options C and D rely on CodePipeline V1 , which lacks modern trigger filtering and execution control.
Option B incorrectly uses SUPERSEDED mode and branch-based triggers.
Therefore, Option A correctly combines CodePipeline V2 , QUEUED execution mode , tag-based triggers
, and EventBridge-driven pipeline chaining , meeting all requirements with best practices and minimal operational complexity.

NEW QUESTION: 103
□ □□□□ □□ □□ □□□ □□□□ □□□ □□□□ □□□ □□ AWS □□□□ □□□ □ □□ □□□ □□□□.
□□□ AWS CodeCommit□ □□ □□□ □□ □□ □□□ □□□□□.
DevOps □□□□□ □□□ □□ □□□□ □□□ □□□ □ □□ □□□ □□□□ □□□. □□□ □□ □□□ □□□□ □□ □□ □□□ □ □□ Git □□□ □□ □□ URL□ □□□ □ □□□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□ □□□ CodeCommit □□□□□□ □□□□□. AWS CodeBuild □□□□□ □□□□ □□ □□□ CodeCommit □□□□□□ □ □ □□□ CodeCommit □□□□□□ Git □□□□□ □□□ □□□□□. CodeBuild □□□□□ □□□□ AWS Lambda □□□ □□□□ □. □□ □□□ CodeCommit □□□□□□□ □□ □□□□ □□□□ Amazon EventBridge □□□ □□□□□. Lambda □□□ □□□□ □ EventBridge □□□ □□□□□.
B. □□ □□□ Amazon S3 □□□ □□□□□. □□ □□□ CodeCommit □□□□□□ □□ Git □□ □□□ □□□□ □□□ S3 □□□ □□□□ AWS Fargate □□□ □□□□□. Fargate □□□ □□□□ AWS Lambda □□□ □□□□□. CodeCommit □□□□□□ □□ □□□□ □□□□ Amazon EventBridge □□□ □□□□□. Lambda □□□ □□□□□ EventBridge □□□ □□□□□.
C. □□ □□□ AWS CodeArtifact □□□□□□ □□□□□. □□ □□□ □□ □□ □□□ CodeCommit □□□□□□ □□□□ AWS CodePipeline □□□□□□ □□□□□. CodeCommit □□□□□ □□□□ □□□□□ □ □□□ CodeCommit □□□□□□ □□□ □ CodeArtifact □□□□□□ □□□□ □□□□ □□□□□□□ □□ □□ □□□ □□□□□.
D. □□ □□□ AWS Cloud9 □□□ CodeCommit □□□□□□ □□□□□. □□ □□□ CodeCommit □□□□□□ AWS Cloud9 □□□ □□ □□□□□□ □□□□□. □□ □□□ CodeCommit □□□□□□ AWS Cloud9 □□□ □□□□□.
Answer: A (LEAVE A REPLY)
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.
AWS CodeCommit User Guide on resilience and disaster recovery1.
AWS Documentation on monitoring CodeCommit events in Amazon EventBridge and Amazon CloudWatch Events2.

NEW QUESTION: 104
□ □□□ □□□□□ □□□ Python □□□□ □□□□ □□ AWS CodeArtifact □□□□ □□□□□. DevOps □□□□□ AWS CodeDeploy□ □□□□ Amazon EC2 □□□□□ □□□□□□□ □□□□ □□□. □ □□□□□□□ CodeArtifact □□□□ □□□ Python □□□□ □□□□□. BeforeInstall □□ □□ □□□ □□□ □□□□ □□□□□.
DevOps □□□□□ EC2 □□□□□ CodeArtifact □□□□ □□ □□□ □□□ □□□□ □□□.
□□ □□□□ □ □□ □□□ □□□□□□?
A. CodeArtifact□ □□ □□□ □□ □□□ □□□□□. □ □□□ EC2 □□□□□ □□□□□. □□□□□□ aws codeartifact get-authorization-token CLI □□□ □□□□□.
B. EC2 □□□□ □□□ □□ Read-FromRepository □□□ □□□□ CodeArtifact □□□□ □□ □□□ □□ □□□ □□□□□.
C. EC2 □□□□□ Python □□□□ □□□□ □ □□□ CodeArtifact □□□□ ACL□ □□□□□.
D. CodeArtifact□ □□□□ □ □□ IAM □□□ □□□ □□□□ □□□□□ □□□□□. □□□□ □□□□□ EC2 □□□□□ □□□□ □. □□□□□□ aws codeartifact login CLI □□□ □□□□□.

Answer: D (LEAVE A REPLY)
To allow an EC2 instance to access CodeArtifact, an IAM role attached via an instance profile must be granted permissions to access the CodeArtifact repository. The EC2 instance assumes this role.
The instance then uses the AWS CLI command aws codeartifact login to authenticate and configure the package manager (e.g., pip) to use the CodeArtifact repository. This command obtains an authorization token and sets up repository credentials securely on the instance.
Service-linked roles (Option A) are managed by AWS services, not used for instance access. CodeArtifact does not support ACLs (Option C), and resource-based policies (Option B) do not grant access to EC2 instances by principal.
This method is standard for securely managing credentials and access to CodeArtifact in automated deployments.
Reference:
AWS CodeArtifact Access Control and Authentication: " Use IAM roles attached to compute resources and the aws codeartifact login command to authenticate to repositories. " (AWS CodeArtifact Developer Guide)

NEW QUESTION: 105
□□□□□ □□ □□□□ □□ □□ □□□ AWS CloudFormation □□□□ □□□□ □□□□ □□□□□ □□□□□. □□□□ □□□ □ □□ □□□□ □□□ □ □□□□ □□□□□ □□□□□.
□□□ □□ □□□ □□□□□ □□ □□□ □□□□ □□□?
A. □□□□ CloudFormation □□□ □□□ □□□□ CloudFormation □□□ □□□ □ □□□ □□□□□. CloudFormation □□□□ □ □□ □□□□ □□□□ □□ □□□□ □□□ □□□ □□□□□.
B. □□□□ CloudFormation □□□ □□□ □□□□ CloudFormation □□□ □□□ □ □□□ □□□□□. AWS Config □□□ □□□□ □□□□ □□ □□□□ □□□ □□□ □□□□□.
C. □□□□ AWS Service Catalog□□ □□□□ CloudFormation □□□ □□□□□ □□□□□. □□ □□ □□□ □□□□□ □□□□ □. AWS Config □□□ □□□□ □□□□ □□ □□□□ □□□ □□□ □□□□□.
D. □□□□ AWS Service Catalog□□ □□□□ CloudFormation □□□ □□□□□ □□□□□. □□□ □□ □□□ □□□□□ □□□ □□. Amazon EventBridge □□□ □□□□ □□□□ □□ □□□□ □□□ □□□ □□□□□.
Answer: C (LEAVE A REPLY)
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.

NEW QUESTION: 106

□ □□□ AWS Fargate □□ □□□ □□□□ Amazon Elastic Container Service(Amazon ECS) □□□□□ □□□□□□□ □□□□□.
□ □□□□□□□ Amazon Simple Queue Service(Amazon SQS) □□□□ □□□□ □□□□□. □□□□□□□ □□□□ □ □□□□ □□□□ □ □ □□ □□ □ □□□□.
□□□□□□□ □□□□ □□□ □, □□□□□□□ Amazon S3 □□□□ □□□ □□ □□ □□□ □□□□ □□□□□. □□□□□□ □ □□□ □□□ □ □□ S3 □□□ □□□□□. □□□ Amazon CloudWatch Logs□ □□□□ □□ □□□ □□□□□□ □□□□□□□ □□□□ □□□□□ □□□□□ □□□□□.
SQS □□□□ □□□□□ □□ □□ □□□□ □□□□□. □□□ □□ □□□□ □ □□ □□ □□□□ □□□□ □□□ □□□□.
DevOps □□□□□ □□□ □□□ □□ □□□ □□□□ □□ □□□□ □□□□ □□□.
□□ □□□□ □□ □□ □□□□ □□□□ □ □□ □□□ □□□ □ □□□□?
A. AWS Application Auto Scaling□ ECS □□□□ □□ □□□ □□□□ □□□□□. □□□ □□□ □□ □□□□ □□□□□ □□ □□ □□ □□□ □□□□□.
B. Amazon SQS□□ □□□□ □□ □□□ □□ □□ □□□□ □□ □□□□□. □ □□□ □□□□ □□□□ □□ □□□□ Amazon SQS□ □□ API □□ □□ □□□□ □ □□□□.
C. SQS □□□ □□ Amazon EventBridge □□□ □□□ □□□□ □□□□□. ECS □□□□ □□□□ □□□□ □□□□ ECS □□□ □□□□ EventBridge □□□□ □□□□□.
D. EC2 □□□□□ □□ □□ □□□ □□□□□. □□ □□ □□□ □□□□ ECS □□□□□ □□ □□□□ □□□□□. ECS □□□□ EC2 □□ □□□ □□□□□ □□□□□.
Answer: (SHOW ANSWER)
AWS recommends Application Auto Scaling for ECS services to dynamically adjust the number of running tasks based on Amazon SQS queue metrics such as ApproximateNumberOfMessagesVisible. By using target tracking scaling policies, ECS on Fargate scales

automatically when the queue backlog grows and scales down when traffic decreases - a fully managed, cost-efficient solution (see ECS Service Auto Scaling Developer Guide).

NEW QUESTION: 107
□ □□□ □□ 1□ □□□ □□□□ □□□□. DevOps □□□□□ □□□ □□ □ □□□ □□ □□□□□ □□□ □□□□□□□. □ □ □□ □□ □□ Environment: NonProducion □□□ □□□ Amazon EC2 □□□□□ □□ □□□ □□□ AWS Lambda □□□□ □□□□ □. □□ □□□ □□□ □□□□.
□□□□□ □□□□ □□ □□ □□□ □□□□ □□ □□ □□ □□□ □□□□ □□□? (□ □□□ □□□□□.)
A. □□ A
B. □□ B
C. □□ C
D. □□ D
E. □□ E
F. □□ F
Answer: A,B,D (LEAVE A REPLY)
The engineer should make the following changes to achieve a policy of least permission:
A: Add a condition to ensure that the principal making the request is an AWS Lambda function. This ensures that only Lambda functions can execute this policy.
B: Narrow down the resources by specifying the ARN of EC2 instances instead of allowing all resources. This ensures that the policy only affects EC2 instances.
D: Add a condition to ensure that this policy only applies to EC2 instances tagged with "Environment: NonProduction". This ensures that production environments are not affected by this policy.
AWS Identity and Access Management (IAM) - AWS Documentation
Certified DevOps Engineer - Professional (DOP-C02) Study Guide (page 179)

NEW QUESTION: 108
□□□□ AWS CodePipeline □□□□□□ □□□□ □□□□□□ □□□□ □□□□□□. □□□□ □□□□□□ □□, □□□ □ □□ □ □ □□□ □□□□□. □ □□□ □□□ AWS CodeBuild □□□□□ □□□□ □ □□□ □□ □□□□□ □□□□□□. □□□ □□□ □□ AWS CodeDeploy□ □□□□ □□□□□□ □□ □□□ □□□□□ □□□.
□□□ □□□□□□□ RPM □□□□ □□□□□□ □□□□□□□ Amazon EC2 □□□□ □□□ □□□□ □□□. EC2 □□□□□ EC2 Auto Scaling □□□ □□□ □□ AMI□□ □□□□□.
□□□ □□ □□□ □□□□ □□ DevOps □□□□□ □□□□ □□ □□ □□□ □□□□□? (□ □□□ □□□□□.)
A. CodeDeploy □□□□□ □□□ □□ AMI□ □ □□□ □□□□□. CodeDeploy□ □□ □□□□ □□□□□ EC2 □□□□□ IAM □□ □ □□□□□□□.

B. CodeDeploy □□□□□ □□□ □□ AMI□ □ □□□ □□□□□. □□□□□□ □□ □□□□□ □□□□ CodeDeploy□ □□ □□□ □□□ □□□□ AppSpec file□ □□□□□.
C. CodeDeploy□□ □□□□□□□ □□□□□. □□□□□ □□ □□□ □□□□□. Auto Scaling □□□ □□ □□□□ □□□□□. CodePipeline □□□□□□ □□□ □□□□ EC2 Image Builder□ □□□□ □ AMI□ □□□□□. □□ □□□ AMI□ □□□□□ CodeDeploy□ □□□□□.
D. CodeDeploy□□ □□□□□□□ □□□□□. □□□□□ □□ □□□ □□□□□. Auto Scaling □□□ □□ □□□□ □□□□□. CodeDeploy □□□ □□□□ □□□□□□□ □□□□□ CodePipeline □□□□□□ □□□□□□□.
E. CodeDeploy□□ □□□□□□□ □□□□□. □□□□□ □□ □□□ □□□□□. □□ AMI□□ □□□□ EC2 □□□□□ □□ □□ □□ □□□□□. CodeDeploy □□□ □□□□ □□□□□□□ □□□□□ CodePipeline □□□□□□ □□□□□□□.
Answer: (SHOW ANSWER)
https://docs.aws.amazon.com/codedeploy/latest/userguide/integrations-aws-auto-scaling.html

NEW QUESTION: 109
□□□ AWS □□□□ □□□□□□□ □□□□□□. □ □□□□□□□ □□ Amazon CloudWatch □□ □□□ □□□□□. □□□ CloudWatch □□ □□□ □□□□ □□□□. DevOps □□□□□ □□ □□□□□□□ □□ □□ □□□ □□□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. CloudWatch □□□ □□□□ □□ □□ □□□□ □□□□ CloudWatch □□ □□□ □□□□ □□□ □□ □□□□ □□□□□.
B. CloudWatch Logs Insights□ □□□□ □□□□□□ □□ □□□ □□ □□ □□□ □□□□ □□ □□ □□ □□□ □□ □□ □□□□ □.
C. AWS Cost Explorer□ □□□□ CloudWatch □□ □□□ □□□ □□□□ □□ □□□□ □□□□□.
D. AWS CloudTrail□ □□□□ □ □□□□□□□ □□ CreateLogStream □□□ □□□
Answer: C (LEAVE A REPLY)
The correct answer is C.
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

NEW QUESTION: 110
□ □□□ Amazon API Gateway REST API, AWS Lambda □□ □ Amazon DynamoDB □□□□ □□□□ □□□□□□□ □□□□ □□ □□. □ □□□□□□□ □□ □□ AWS □□□□ □□□□ □□□□. □□□ □□□□□□□ □ □□ □□□ □□ □□□□□□ □□□□ □ □□□. □□□ □□□□ □□ □□□ □□ □□ □□□□ □□□□□□ □□□.
□□ □□□□□ □□□□ □□ □□□ □□□□□? (□ □□□ □□□□□□.)
A. □ □□ □□□ DynamoDB □□□□ □□□ □□□ □□□□ □□□□□.
B. DynamoDB □□□□ □□ □□ □□ □□□□ □□□□□.
C. REST API□ Lambda □□□ □□□□ □ □□ □□□ □□□□□.
D. Amazon Route 53□□ □□ □□□ □□□□□. □□ □□ □□□ □□□ □□□ DNS □□□□ □□□□□.
E. Amazon Route 53□□ □□ □□□ □□□□□. □□ □□ □□□ □□□ □□□□ DNS □□□□ □□□□□.
F. □□□ □□ □□□ □□□ □□□□ Amazon Route 53□ DNS □□□□ □□□□□.
Answer: A,C,E (LEAVE A REPLY)
To achieve multi-Region high availability for a serverless API-based application, the architecture must be duplicated in the second Region and supported by a multi-Region data layer and appropriate DNS routing.
Option A provides the required multi-Region data layer using DynamoDB global tables , which replicate data automatically and asynchronously between Regions. This allows reads and writes in either Region and prevents cross-Region latency bottlenecks.
Option C ensures that the compute components-API Gateway REST API and AWS Lambda functions-are deployed in both Regions. Each Region operates independently, providing true active-active redundancy and allowing Route 53 to route traffic directly to either Region without dependency on the other.
Option E selects Route 53 latency-based routing , which directs each user to the Region with the lowest round- trip latency. This meets the requirement to route traffic based on the Region providing the best response time rather than only providing active-passive failover.
Option D (failover routing) supports active-standby topologies, not active-active. Option B creates a global secondary index, which does not provide multi-Region replication. Option F (multivalue routing) distributes traffic randomly and does not account for latency or Region placement.
Thus, combining DynamoDB global tables, duplicated serverless architecture, and latency-based routing fully satisfies the requirements.

NEW QUESTION: 111
□□□ □ □□ EC2 Linux □□□□□□ □□□□ □□□□ □□□□□ □□□□ □□□□ API □□ □ DB □□ □□□ □□ □□□ □□ □□□ □□□□□□ □□□.
□□ □□□□ □□ □□□□□?
A. □□□□□ CloudWatch □□□□□ □□□□, □□□ □□ □□□□ □□□□□ □□□□, □□□□ □□□□□ □□□□□ □□ □□ □□□.

B. Amazon Managed Service for Prometheus□ □□□□ □□□□ □□□□□□ CloudWatch □□□□□ □□□□ □□□□ CloudWatch □ □□□□□.
C. □ □□□□□□ DB□ □□□□, □□□□ □□□□, PutMetricData□ □□ CloudWatch□ □□□□ Lambda□ □□□□□.
D. □□□ □□□ □□ □□□ □□□□□. CloudWatch Logs Insights□ □□□□ □□□□ □□□□ □□□□□.
Answer: (SHOW ANSWER)
* Installing the CloudWatch agent and instrumenting the application to push custom metrics to the agent is the easiest and lowest overhead method.
* Prometheus (B) adds operational complexity.
* Lambda polling (C) introduces unnecessary complexity and latency.
* Using Logs Insights (D) requires extracting metrics from logs, which is less efficient.
References:
Custom Metrics with CloudWatch Agent
CloudWatch Agent User Guide

NEW QUESTION: 112
DevOps □□□□□ Ruby □□ □□□□□□□ □□□□□ □□□ □□□□□. □□□□□□□ MySQL□ Amazon RDS □□□□□□□ □□ □□□□ □□ □□ □□ □ □□□□□ □□□ □□□ □□□. □□□□□□□ □□□ □□□□ □□□□ □□□□□□ □□ □□□ □□□□ □□□□□ □□□.
DevOps □□□□□ □□ □□□ □□ □□□□□□□ □□ □□□□ □□ □□□ □□□□ □□□. □□ □□□□ □□□ □□□ □□ □ □□□□□ □□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (3□□ □□□□□.)
A. AWS Elastic Beanstalk□ □□□□□□□ □□□□□. Elastic Beanstalk □□□ □□□ MySQL DB □□□□□ Amazon RDS□ □□□ □□.
B. AWS Elastic Beanstalk□ □□□□□□□ □□□□□. Elastic Beanstalk □□□ □□□ MySQL DB □□□□□ Amazon RDS□ □□□ □□.
C. AWS Elastic Beanstalk □□□□ □□□□□□ □□ □□□□ □□ □□□ □□□ □□□□□.
D. AWS □□ □□□□ □□□□□□□ Amazon EventBridge □□□ □□□□□. Amazon Simple □□ □□□(Amazon SNS) □□□ □□ □□ □□□□ □□□□□□ □□ □□□□.
E. □ □□□□□□ □□□ □□□□□ □□ □□ □□□ □□□□□.
F. □□ □□ □□□ □□□□ □ □□□□□□ □□□ □□□□□.
Answer: (SHOW ANSWER)
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.
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.

NEW QUESTION: 113
□ □□□ □□ □□□□□□□ □□□□□ □□□ □□□□ □□□□ Amazon EC2 □□□□□ □□□□ □□□□. □□□ EC2 □□□□ □ □□□□□□□ □□□□ □□ □□ □ □□□ □□□□ □□ □□ □□ □□□ □□□□□ □□□. □□□ EC2 □□□□ □□□ □□ □□ □□ □□ □□□ □□□□□□ □□□ □ □□ □□□□ □□□□ □□□□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□□ □□ □□□ □ □□ □□ □□□ □□□□ AWS Systems Manager □□ □□□□ AWS Config□ □□□□ Systems Manager □ □□ □□□ □□□□□.
B. SSH □□ □□□□ □ EC2 □□□□□ □□□□□□. □□□ □□□□ □□□□ □□ □□□□□ □□□□ □□□□□. □□□ □□□ □□.
C. Amazon CloudWatch □□□□ □□□□ □□ □□□□ □□□ □□ Auto Scaling □□□ □□□□□. □□□ □□ □□□ □□□ □□ AMI□ □□□□ □ □□□□□ □□□□ Auto Scaling □□ □□□□ □□□□□.
D. AWS CloudFormation□ □□□□ □ □□□ □□□ □□□ □□ AMI□ EC2 □□□□□ □□ □□□□□. AWS CloudTrail □□□ □□ □□ □□ □□ □□□ □□□□□□ □□□ □□□□ □□ □□□□□ □□ □□□ □□□□.
Answer: A (LEAVE A REPLY)
Option A is the most correct because it provides both : (1) automated patching and (2) compliance monitoring /enforcement across a fleet, using AWS-native services built for exactly this purpose.
* AWS Systems Manager Patch Manager is designed to automate patching of managed instances using patch baselines , maintenance windows (or on-demand), and it produces compliance status for patching. It's the standard AWS service to apply OS/security patches at scale without SSH'ing into instances.
* AWS Config can be used to evaluate and track compliance over time against defined rules, giving centralized visibility and continuous compliance assessment. With remediation , Config can invoke Systems Manager Automation documents to correct non-compliant resources or trigger patch actions (depending on the rule/remediation design). This meets the "monitor and enforce" requirement.
Why the other options don't meet requirements as well:
* B is manual, doesn't scale well, and increases operational risk (key management, human error). It's not "automated monitoring and enforcement."
* C (replacing instances with new AMIs) can be part of an immutable infrastructure strategy, but by itself it does not provide compliance monitoring across the current fleet, and scaling policies based on CloudWatch metrics are unrelated to patch compliance. Also, patch cadence would depend on AMI pipelines and instance rotation rather than direct compliance enforcement.
* D is operationally heavy and mismatched: CloudTrail records API activity; it does not natively provide "patch compliance" status for instance OS packages. Recreating instances via CloudFormation for every patch is not an efficient or standard enforcement mechanism for patch compliance.

NEW QUESTION: 114
□□□ AWS Organizations□ □□□□ □ □□□ □□ □□□ AWS □□□ □□□□ □□□□. □□□ □□ □□□ □□□□□ □□□.

* □□□□□ □□□ □□□ Linux AMI□ □□□□□□ □□ □□□□ □□□□□.
* □□ □□□□ Chef □□□□□ □ □□ □□ □□
* □□ □□□ AMI□ □□ □□□ □□

□□□□ □□ □□□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?

A. □□ □□ □□□□□ Amazon EC2 □□□□□ □□□□ □□□□ □□ □□ □□□□ □□ □ □□□ Chef □□□□□ □□□□ □ □ □ □□□□ □□□ □□ □ □□ □□□□ □□□□□ AMI □□□ □□□□□. □□□ □□□ □□□ □□□□□□.

B. Amazon EC2 Image Builder□ □□□□ □□ Linux AMI□ Chef □□□□□ □□□□ □□ □□□ □□□ □□□ □□□□□□ □□□ □□. AWS Resource Access Manager□ □□□□ EC2 Image Builder □□□□ □□ □□□ □□□□□.

C. AWS Systems Manager Automation Runbook□ □□□□ □□ □□□□ □□□□ Linux AMI□ □□□□□□□. Chef □□□□□ □□ □□□ □□□□□ □□ URL□ □□□□□. AWS Organizations□ □□□□ □□ □□□ □□ □□ □□□□ □□□□□.

D. Amazon EC2 Image Builder□ □□□□ □□ Linux AMI□ Chef □□□□□ □□□ □□ □□□ □□□ □□□ □□□□□□ □□□□ □. AWS Systems Manager Parameter Store□□ □□□□□ □□□□ □□□ □ □□ □ AMI ID□ □□□□□. □□□ □□

Answer: B (LEAVE A REPLY)

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. References:

* Amazon EC2 Image Builder
* Sharing EC2 Image Builder images

NEW QUESTION: 115

□□□ DevOps □□□□□ □□ □□ □□□□ □□□□ □□□□. □ □□□ AWS Transit Gateway□ □□□□ □□□□ □□ □□□ □ □ □ □□ □□□□□ □□□□ □□□□□□. □□□□ □□ □□□□ □□ □□ □□□□ □□□□ □□□ □□□□□□ □□□□ □□ □ □□ □□ □□□ □□□□□□□ □□□□□.

□□□ □□□□□□□ Amazon CloudWatch Logs□ □□□ □□□□ CRITICAL, HIGH, MEDIUM, LOW □ INFO□ □□□ 70□ □□□□ □. □□ □□ CRITICAL □□□□ □□□ □□ □□□ □□□ □□□□.

DevOps □□□□□ □□□ □□ □□□ □□□□ □□ □□□ □□ □□□?

A. Amazon CloudWatch Synthetics □□□□□ □□□□ □□□ □□□ □□□□□□□. □□□□ CRITICAL □□□ □□□□□ CRITICAL □□□□ □□□□ □□ CloudWatch □□□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□ □□□ □□□□□. □□ □□ □□□ □□□ □□□ □□□□□.

B. CRITICAL □□□ □□□ □□□□ Amazon CloudWatch □□ □□□ □□□□□. □□□ □□ □□□ □□ □□□ □□□□□. □□□ □□ □□□ □□□□ □□ CloudWatch □□□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□ □□□□□. □ □ □□ □□□ □□□ □□□ □□□□□□.

C. □□□□ □□ □□□□ Amazon GuardDuty□ □□□□□□. □□ □□□ □□□□□□□ GuardDuty □□ □□ GuardDuty □□□□ □□ □□□□ Amazon EventBridge □□□ □□ □□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□ □□ □□ □□□ □□□ □□□ □□□□□.

D. AWS Firewall Manager□ □□□□ □□ □□□ □□□ □□□ □□□□□□. □□□□ □□□□. □□ Firewall Manager □□□□ □□ □□□□ EventBridge □□□ □□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□□ □□ □□ □□ □□□ □□□ □□ □ □□□□□□.

Answer: (SHOW ANSWER)

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

NEW QUESTION: 116

□ □□□ AWS Organizations□ □□□□ □□ AWS □□□ □□□□ □□□□. □ □□□ □□□□ □□□ AWS □□□ □□□ □□□ □ □□□ □□□□□.

□ □□□ AWS CloudFormation□ □□□□ □□□□ □□□ □□ □□□□ □□□□□ □□□. □□□ AWS □□□□ □□□□ □□□ □ □□ □□□□ □□□□ □□□ □ □□□ □□□. □□□□ □□□□ AWS CodeConnections □□ Git □□□□ □□□ CloudFormation □□□□ □□□□ □□□□ □ □□□□ □□□ □ □□□ □□ □□□.

□□ □□ □□□ □□ □□ □□ □□□□ □□□ □□ □□□ □□□ □ □□□□? (3□□ □□)

A. AWS □□□ □□ □□□□ □□□□□ CloudFormation IaC(Infrastructure as Code) □□□□ □□□□□. □□□ □□□□ □□□□ CloudFormation □□□□ □□□□□. CloudFormation □□□□ □ CloudFormation □□□□ □□□□□.

B. AWS Config□ □□□□ AWS □□□ □□ □□□□ □□□□□. □□□ □□□□ □□□□ CloudFormation □□□□ □□□□□. CloudFormation □□□□ □ CloudFormation □□□□ □□□□□.

C. CodeConnections□ □□□□ Git □□□□ AWS CodePipeline □□ □□□ □□□□□. CloudFormation □□□□ Git □□□□ □□□□. CodePipeline□□ Git □□□□ □□□ □□□ CloudFormation □□□ □□□□ □□□□□□ □□□□□.

D. CodeConnections□ □□□□ Git □□□□ CloudFormation □□ □□□ □□□□□. CloudFormation □□□□ Git □□□□ □□□□ □. Git □□□□ CloudFormation □□□ □□□□□□.

E. IAM □□□ □□□□ CloudFormation□ □□ □□□ □□□□□. IAM □□□ □□ □□□□ □□□ □ □□ □□□ □□□□□. IAM □ □□ □□□ □□ □□ □□□ □□ □□□ □□□□ SCP□ □□□□□. SCP□ □□□□ OU□ □□□□□.

F. IAM □□□ □□□□ CloudFormation□ □□ □□□ □□□□□. IAM □□□ □□ □□□□ □□□ □ □□ □□□ □□□□□. □□ □ □□ IAM □□□□ □□□□ SCP□ □□□□□. SCP□ □□□□ OU□ □□□□□.

Answer: A,C,E (LEAVE A REPLY)

* Step A: Using a tool like CloudFormation resource import or IaC generator to scan and create a template from existing resources is efficient to bring current infrastructure under management.
* Step C: Using CodeConnections (AWS ' s solution to connect Git repositories) with AWS CodePipeline ensures any changes to CloudFormation templates in the Git repo automatically deploy infrastructure changes, enforcing infrastructure as code workflows.
* Step E: Creating an IAM role with CloudFormation as the principal ensures CloudFormation has permissions to manage resources. Using an SCP to deny all actions except by this role enforces strict control, preventing manual changes outside the pipeline. Option B uses AWS Config which is more for compliance and auditing, not direct resource import. Option D is invalid because CloudFormation does not natively sync with Git; CodePipeline does. Option F is less secure than denying all except the IAM role.

Reference:

AWS CloudFormation Resource Import: " Import existing resources into CloudFormation stacks for management. " (CloudFormation Resource Import) AWS CodePipeline and CodeConnections Integration: " Use CodeConnections to connect Git providers with AWS CodePipeline for continuous deployment. " (AWS CodePipeline Git Integration) AWS Organizations SCP and IAM Role Best Practices: " Use SCPs to restrict actions and IAM roles with limited principals to enforce secure management. " (AWS Organizations Best Practices)

NEW QUESTION: 117
□□□ □□□□ AWS DevOps □□□□□□ □□□ □□□ □□ □ □□ □□□ □□□ □□□□□. □□□ □□ □□ □□ □□□ □□□ □□□ □ □□ □□□□□ □□□ □□□ □□□□.
□□□ □□□ □□□□□ □□□?
A. Amazon S3 □□□ □□□ □□□□□ AWS Config □□ Amazon Athena□ □□□□ □□□ □□□□ □□□ □□□ □□□□ □□ □ □ □□□ □□□□.
B. □□□ □□□ □□□□ □□□ □□□□□□ □□ □□ □□□ □□□□ Amazon GuardDuty□ □□□□□.
C. □□□ □□□ □□□ □□ Amazon SNS □□□ □□□□ □□□□ AWS Management Console □□□ □□□ □□□ □□□ □□□□ Amazon EventBridge □□□ □□□ □□□□□.
D. □□□ □□□ □□□ □□ Amazon SNS □□□ □□□□ □□□□ AWS Lambda □□□ □□□□ □□ AWS CloudTrail □□□ □□ □ □□□□ AWS API □□□ □□□□ Amazon EventBridge □□□ □□□ □□□□□.
Answer: D (LEAVE A REPLY)
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();
```

};
References:
Creating EventBridge Rules
Using AWS Lambda with Amazon SNS

NEW QUESTION: 118
□ □□□ □□□□□ □□□ □□□□ □□□□□ □□□□ □□ □□□□ □□□□ □□□□. □□□ □□□□ 4□□ AWS □□□ □□ □□ □□□□□ □□□□□ □□ □□□ □□□ □□□ □ □□ □□□.
□□ □□□□ □□ Linux □□□□□□ □□□ □□□□□. □□□□□ □□□ □□□ NetApp ONTAP □□□□ □□□ □□□□□□□ □□□□□. □□□□ □□□ □□ □□□□□ VM□ □□ □□□□ □□□□□□. □□□ □□ □□□ □□□□□□□ □□□□ □ □ □ □□□□□□.
DevOps □□□□□ □□ □□□ □□□ □□□□ □□□□ □□□. DevOps □□□□□ □□ □□□ □□ □□□□ □□ □□ □□□ □□ □□□ □□ □□□. □□, □□□□ □□□□□ □□□□ □□□ □□□□ □□□□□ □ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□□□□ □□□ □□□ Amazon S3 □□ □□□□□□ □□□□□. □ □□□ S3 □□□ □□□□□.
□□ □□□□ S3 □□ □□□□□□ □□□□ □□□□ □□ □□□ □□□□□. □ □□□ S3 □□□ S3 □□□ □□ □□(CRR)□ □□ □□□.
B. □ □□□ Amazon FSx □□ □□□□□□ □□□□□. □ □□□ Amazon FSx for Windows □□ □□□ □□□□□. □□ □□□□ FSx □□ □□□□□□ □□□□ □□□□ □□ □□□ □□□□□.
C. □ □□□ □□ AZ Amazon FSx for NetApp ONTAP □□□□ □ □□□ □□□□□. □□□□□ □□□□ □□□ FSx for ONTAP □□ □□ □□ □□□ SnapMirror □□□ □□□□□.
D. □ □□□ Amazon Elastic File System(Amazon EFS) □□ □□□□ □□□□□. □□□□□ □□□ □□□ AWS DataSync □□□□□ □□□□□. DataSync□ □□ Amazon EFS□ □□□□ □□□□□ □□□ □□□□□.
Answer: C (LEAVE A REPLY)
Amazon FSx for NetApp ONTAP provides NetApp ONTAP features in AWS, including SnapMirror replication and storage efficiencies like deduplication and compression. Create FSx for ONTAP in each Region and use SnapMirror from on-prem ONTAP to each Region for efficient, incremental replication.
Regions can serve data independently of on-prem availability once replicated.

NEW QUESTION: 119
□□□□ □□□□ □□ □□□□□□□ Amazon EKS□ □□□□□□□□□ □□□□ □□□ □□□ □□□□□ □□□. □ □□□ □□ □ □□□□ □□□ EKS □□ □□□ □□□ □□ □□□ □□ □□□□. □ □□□□□ □□□□ □□ □□□□ □□□□ □□□ SNS □ □□ □□□□ □□□□ Amazon SNS □□□ AWS Lambda □□□ □□□□□.
□□□ □□ □□□ □□□□ □□ □□□□ □□□□□?
A. Amazon CloudWatch Logs□ □□□□□ EKS □□ □□□ □□□□□. Lambda□ □□ □□ □□□□ □□□□ □ □□ □□□ □□ CloudWatch □□ □□□ □□□□□.
B. Amazon CloudWatch Logs□ □□□□□ EKS □□ □□□ □□□□□. Lambda□ □□□□ Amazon EventBridge □□□□ □□□ CloudWatch Logs Insights □□□ □□□□□.
C. EKS □□ □□□ □□ Amazon S3 □□□ □□□□□□. Lambda□ □□ □□ □□□□ □□□□ □ □□ □□□ □□ Amazon CloudWatch □□ □□□ □□□□□.
D. EKS □□ □□□ □□ Amazon S3 □□□ □□□□□□. AWS Lambda□ □□□□ S3 PUT □□ □□□ □□□ □□□□□.

Answer: A (LEAVE A REPLY)
https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/SubscriptionFilters.
html#LambdaFunctionExample
https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/SubscriptionFilters.html

NEW QUESTION: 120
DevOps □□□□□ □□□□ □□ □□□□□ □□□□ □□□□. □□□□□ AWS CloudFormation□ □□□□ Amazon ECS □□□□ □ Amazon EC2 Auto Scaling □□□ □□□□ □□□□□□□ EC2 □□□□ □□□□□ □□□□□ □□□□□□. CloudFormation □□ □ □□□□□ □□□ □ □□□□□ ECS □□□□□ EC2 □□□□□ □□□□□ □□□□ □□□ □□□ □□□□□ EC2 □□□□□ □□ □□□□□ □□□□ □□□ □□□□□□.
□ □□□ □□□□□ DevOps □□□□□ CloudFormation □□□□ □□□ □□□□□□ □□□?
A. AWS: ECS: □□□□ □□□□□ EC2 □□□□□ □□□□ AWS: ECS: □□□ □□□□□ ECS □□□□□ □□□□□.
B. UserData □□□ AWS: AutoScaling: LaunchConfiguration □□□□□ ECS □□□□□ □□□□□.
C. AWS:EC2: UserData □□□ □□□□ □□□□□ ECS □□□□□ □□□□□.
D. AWS: CloudFormation: CustomResource □□□□□ ECS □□□□□ □□□□ EC2 □□□□□ □□□ ECS □□□□□ □□□□ AWS Lambda □□□ □□□□□□.
Answer: B (LEAVE A REPLY)
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. References: AWS::AutoScaling::LaunchConfiguration, Amazon ECS Container Agent Configuration

NEW QUESTION: 121
□ □□□□□ Microsoft SQL Server□ Amazon RDS□ □□ □□□□□□□ □□□□ RPO < 1□, RTO < 10□□□ □□ □ □□□□□ □□□□ □□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. RDS □□ AZ DB □□□□□ □□ □ □□ □□□□ □□□□□. Route 53□ □□ □□ □□□ □□□□□□.
B. □□□□ □□ □□□ □□□□ □□ AZ □□□□□ □□□□□.
C. □□ AZ RDS + DMS □□□ □□□ □□□□□.
D. □□ □ □□□ □□ AZ□ □□□□□.
Answer: (SHOW ANSWER)
RDS Multi-AZ cluster deployments with cross-Region read replicas support near real-time replication (<1 min RPO) and fast promotion (<10 min RTO). Route 53 CNAME failover redirects application traffic automatically. This is the AWS-recommended DR pattern.

NEW QUESTION: 122
□ □□□□ □□□□ □□□ □□ □□(DR) □□□ □□□ AWS □□□ □□□□ □□□ □□□□□□. □□ □□□□ □□□□□□□ Amazon EC2 □□□□□□ □□□□ AWS CloudFormation□ □□ □□□□□. □□□□□□□ Amazon FSx for NetApp ONTAP □□□ □□□□□□ □□□□□ □□□□□. EC2 □□□□□□ □□□□□□ □□□□ □□□□ □□□□. DevOps □□□□□ □□□ AMI□ □□□ DR □□□ □□□□□. □□ DevOps □□□□□ □□□ □□□□□ □□□□□ CloudFormation □□□ □□□□□□□. DR □□ □ □□□□□ 10□□ RPO□ □□□ □□□. □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □ □□ □□□ Amazon S3 □□□ □□□□□. S3 □□□ □□ S3 □□ □□ □□(CRR)□ □□□□□. FSx for ONTAP □□□ □ □□ □□ □□□□ □□□ S3 □□□□ □□□□ □□□ AWS Lambda □□□ □□□□□.
B. DR □□□ FSx for ONTAP □□□□□ □□□□□. □□ □□ NetApp SnapMirror□ □□□□ □□□ □□□ DR □□□□ □□□□□ 5□ □□□ □□□ □□□□□.
C. EC2 □□□□□ □□□ □□□□ □□□ □□□ □□□□ □□□□ AWS Lambda □□□ □□□□□. □□□□ DR □□□ □□□□ □ □ □□□□ □□□□□ Lambda □□□ □□□□□. 10□□□ Lambda □□□ □□□□ Amazon EventBridge □□ □□□ □□□□□.
D. AWS Backup□ □□□□ □□ □□□ 10□ □□□ □□□ □□ □□ □□□ □□□□□. □□ □□□□ DR □□□ □□□□□. □□□ □ □□□ EC2 □□□□□ □□ □□□ □□□□□.
Answer: B (LEAVE A REPLY)

NEW QUESTION: 123
□ □□□ RDS DB □□□□□ □□ Amazon RDS □□□□ □□ □□□ □□□□□□. DevOps □□ Amazon CloudWatch □□□□□□ □□ □□ □□□□ □□□□□ □□□. □□ □□□□ □ □□ □□□ □□□□□?
A. RDS □□□□□ □□□□ RDS □□□□ □□ □□ □□ □□□□ □□□□ Amazon EventBridge □□□ □□□□.
CloudWatch □□□ □□ □□□□ □□□□ AWS Lambda □□□ □□□□. Lambda □□□ □□□□□ EventBridge □□□ □□□□□.
CloudWatch □□□□□ □□□□ □□□ □□ □□□□ □□□□□□.
B. □□ □□□□ □□□ AWS CloudTrail□ □□□□ □□□□ □□□□. □□ □□□□ Amazon CloudWatch Logs□ □□□□ □□□□ □□□□□. RDS □□□□ □□ □□ □□□□ □□□□□ CloudWatch Logs□□ □□□ □□□ □□□□. CloudWatch □□□□□ □□ □□ □□□ □□□ □□□□□□.
C. RDS □□□□ □□ □□ □□□(RDS □□□)□ □□□□ Amazon EventBridge □□□ □□□□. CloudWatch □□□ □□□□.
CloudWatch □□□ □□□ □□□□□ EventBridge □□□ □□□□□. CloudWatch □□□□□ □□□□ □□ □□□ □□□□□□.
D. □□□ □□□□ □□□ AWS CloudTrail□ □□□□ □□□□ □□□□. □□□□ □□□□ □□□ □□□□ Amazon CloudWatch Logs□ □□□□. RDS □□□□ □□ □□ □□□□ □□□□□ CloudWatch Logs□□ □□□ □□□ □□□□. CloudWatch □□□□□ □□□□ □□□ □□□ □□□□□□.
Answer: A (LEAVE A REPLY)
Step 1: Reacting to RDS Storage Autoscaling Events Using Amazon EventBridgeAmazon 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 LambdaOnce 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.

NEW QUESTION: 124
□ □□□ Amazon Redshift□ □□□□□□□□ □□□□□ □□□□ □□□□. □ □□□ Redshift □□□ □□ □□□ □□□□ □□□ □ □□□ □□□ □ □□ □□□□□ □□□□ □□□.
□ □□ □□□ □□□□□ □□ □□ □□□ □□□□□? (□ □□□ □□□□□.)
A. Amazon CloudWatch □□ □□□ □□□□□. CloudWatch □□ □□□ □□□□ □□□□ AWS CloudTrail □□□□ □□□□□.
B. □ Amazon S3 □□□ □□□□□. Redshift □□□□□□ □□ □□ □□□ □□□□□. S3 □□□ □□□□ □□□□□.
C. Redshift □□□□ □□□□□□ □□ □□□ □□□ □□ □□□ □□□□□ □□□□□. Amazon CloudWatch□ □□□□ □□□□□.
D. □□ □□□ □□ Amazon CloudWatch □□□□□ □□□□. Redshift □□□ □□□ □□ □□□ □□□□□ □□□ □□□□□.
E. Amazon Athena□ □□□□ Redshift □□□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□□ □□□□ □□□ □□ □□ □ □□ □□ Amazon CloudWatch □□□□□ □□□□□.
Answer: (SHOW ANSWER)
Amazon Redshift supports audit logging which can be configured to log user activity, including queries and user changes. The logs can be delivered to an S3 bucket (Option B), which is the standard location for Redshift audit logs.
While Redshift can send some logs to CloudWatch, the native audit logging is typically routed to S3. Creating a CloudWatch dashboard with a log widget (Option D) allows visualization of user activities from logs ingested into CloudWatch Logs or through custom processing.
Option A is invalid because CloudTrail does not capture Redshift query logs directly. Option C incorrectly assumes Redshift audit logs can be delivered directly to CloudWatch. Option E adds complexity by requiring Lambda and Athena unnecessarily when direct visualization in CloudWatch is simpler.
References:
Amazon Redshift Audit Logging
Visualizing Logs with Amazon CloudWatch Dashboards

NEW QUESTION: 125
□ □□□□ Amazon EC2 □□□□□□ □□□□ □□□□□□□ □□□ □□□□ □□ □□ □□□□□□□ □□□□.
□□□ □□ EC2 □□□□□□ □□ □□□ □□□□ 60□□ □□ □□□□□□ □□□□ □□□□ □□□.
□□□□□□ □□□□ □□ □□ □ □□□□ □□□. □□□ □□ □□□□ □□ □□ □ □□□ □□□ □□□ □□□.
□□ □□□□ □□ □□ □□ □□□□ □□□ □□ □□□ □□□ □ □□□□?
A. AWS Systems Manager State Manager□ □□□□ □□ □□□ □□□ □□□ □□□ □□□□ □□□□□. run □□□ □□ □□ □□ □□□ □□□□□ □□□□□ □□□ □□□ □□□□□.
B. AWS Systems Manager State Manager□ □□□□ □□ □□□ □□□ □□□ □□□ □□□□ □□□□□. □□□ □□□ □□ □□ □□□ □□□□□ □□□□□ □□□□□. □□□ □□□ □□□ □□ □□□ □□□ □□□□ Amazon SES□ □□ □□ □□□ □□□ □□.

C. □□□ □□□ Lambda □□□ □□□□ EventBridge □□□ □□□□□. Lambda □□□ □□ □□□□□ □□□□□ □□□□ SNS□ □□ □□ □□□ □□□□□ □□□□□.
D. □□□ □□□ Lambda □□□ □□□□ EventBridge □□□ □□□□□. Lambda □□□ □□ □□□□□ □□□□□ □□□□ SES□ □□ □□ □□□ □□□□□ □□□□□.
Answer: (SHOW ANSWER)
* Systems Manager State Manager with Automation documents allows running scriptssequentially and reliablywith built-in retry and status tracking.
* Using EventBridge with SNS for notifications leverages managed services with minimal custom development.
* Using SES (Options B, D) requires more setup and custom logic for email formatting and sending.
* Lambda (Options C, D) can run scripts but might have limitations on execution time and complexity compared to Systems Manager Automation.
References:
AWS Systems Manager Automation
Monitoring Automation Execution and Notifications

NEW QUESTION: 126
□ □□□ CloudWatch Logs□ □□□ □□□□□□□. □□□ 30□ □□ □□ □□□□ □□, 90□ □□ □□ □□ □□□□ □□ □□□ □, 180□ □□□ □□□ □□□ □ □□□ □□□.
□□ □□□□ □□ □□ □□□□□□?
A. □□ □□ □□□ 30□□ □□□□ Kinesis Data Streams□ □□ S3□ □□□□□. S3 □□ □□ □□□ □□□□□.
B. 30□ □□ □□□ □□□□ Kinesis Data Firehose□ □□ S3□ □□□□□□□. S3 One Zone- IA □ Glacier Flexible Retrieval□ □□ □□□.
C. 30□ □□ □□□□□ □□□□ Kinesis Data Streams□ □□ □□□□□ □□ S3 Standard-IA □ Glacier Instant Retrieval□ □□□□ □.
D. 30□ □□ □□□ □□□□ Kinesis Data Firehose□ □□ S3□ □□□□□□□. S3 Standard-IA(90□) □ Glacier Deep Archive(180□) □ □□□□□.
Answer: D (LEAVE A REPLY)
Exporting logs using CloudWatch subscription filters + Firehose # S3 enables long-term archival. S3 lifecycle transitions optimize cost with Standard-IA (low-latency) and Deep Archive (long-term retention). This matches AWS best practice for cost-effective log retention.

NEW QUESTION: 127
□□□ AWS Organizations□ □□□ □□□□ □□ AWS □□□ □□□□□. □ □□□ □□ □□ □□□ □□ □□□ □□□□ □□ AWS Lambda □□□ □□□□□ □□□ VPC□ □□□ Lambda □□□ □□ □□□□□ □□ □□□□□□□ □□ □□□□ □□□□. □□ □ □ □□□ □□ A□□ Amazon Elastic Pile System(Amazon EFS)□ □□ □□□□□ □□□□ □□□□.
□□□ Lambda □□ □□□ □□ Amazon EPS□ □□ □□□□□ □□□□ □□ □□□□ □□□□□ □□ B□ □□□□ □□□.
DevOps □□□□□ Lambda □□□ □□ EPS □□□ □□□□ □□ □□□□ □□□□ □ □□□ □□ EFS □□ □□□□ □□□□□ □ □□.
DevOps □□□□□ □□□ □□ □□□ □□□□ □□ □□ □□ □□□ □□□□ □□□? (3□□ □□□□□.)
A. EFS □□ □□□ □□□ □□□□□□ □□ B□ □□ A□ EFS □□ □□□□ □□□□□ □ □ □□ □□□ □□□ □□□□□.
B. Amazon EFS□ □□ □□□□ □□□ □□ □□□□□ □□□□ SCP□ □□□□□.

C. □□ B□ □ EFS □□ □□□ □□ AWS DMS(AWS Database Migration Service)□ □□□□ □□ A□ □□ B□ □□□□ □□□□ □ □□ □□□□□.
D. VPC □ EFS □□ □□□□ □□□□ □ □□ □□□□ Lambda □□ □□□ □□□□□□□.
E. VPC □□□ □□□ □□□□ □□ A□ □□ B□ □□□□□.
F. □□ A□□ □□ IAM □□□ □□□ □□ B□□ Lambda □□□ □□□□□.
Answer: A,E,F (LEAVE A REPLY)
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.html
https://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": [
"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.

NEW QUESTION: 128
DevOps □□□□□ □ □□□□ □□□□ □□ AWS CloudFormation □□□□ □□□□ □□□□. □ □□□□ ALB(Application Load Balancer) □□ □□ □□□□ □□□□ Amazon EC2 □□□□□□ □□□□□. DevOps □□□□□ □□□□ IPv6 □□□ □□ □□□□ □□ □□□ □□□ □ □□□ □□□□ □□□.
DevOps □□□□□ IPv6 □□□□□□ □ □□□□ □□□□ □ □□□ CloudFormation □□□□□ □□□ □□ □□□?
A. VPC□ IPv6 CIDR □□□ □□□□ EC2 □□□□□ □□□□ □□□□ □□□□□. IPv6 □□□□□ □□ □□□ □□□ □□□ □□ □□, IPv6□ □□□□ EC2 □□□□ □□□ □□□□, □ EC2 □□□□□ IPv6 □□□ □□□□□.
B. □ EC2 □□□□□ IPv6 □□□ IP □□□ □□□□□. □□ □□□ □□□□ EC2 □□□□□ □□□□ □□□□□. ALB□ □□ 443□ □ □□□□ □□□ □□ □□□ ALB□ □□□□□.
C. ALB□ NLB(Network Load Balancer)□ □□□□□. IPv6 CIDR □□□ VPC□ NLB□ □□□□ □□□□ NLB□ IPv6 □□□ IP □□□ □□□□□.

D. IPv6 CIDR □□□ VPC□ ALB□ □□□□ □□□□□. □□ 443□□ □□□□ □□□□ ALB□□ □□ □□ IP □□ □□□ □□□□ □. □□ □□□ □□□□ EC2 □□□□□ □□□□ □□□□□. □□ □□□ ALB□ □□□□□.

Answer: D (LEAVE A REPLY)

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.

NEW QUESTION: 129

□□□□ □□□□ □□□□ □□□□ □□ Amazon EC2 □□□□□ 24□□ □□□ □□□□□ □□□ □□ □□□ □□□□. □□□□ □□□ □□ □□□□□□□ Amazon CloudWatch Logs □□□□□ □□□ Auto Scaling □□□ □□□□ □□□□.

□ □□□□□ □□□ □□□□ □ □□□□?

A. AWS Step Functions □□□□□□□ □□ CloudWatch Logs □□□ □□□□□. □□□ □□□□ □□□ EC2 □□□□□ □□□ □□ □□ □□□ □□□□□ □□□□□ AWS Lambda □□□ □□□□□. □ □□□ □□ □□ □□□□□ □□□□ □ □□ Lambda □□□ □□□ □ □ □□□□ Amazon EventBridge □□□ □□□□□.

B. □□□ □□□□ □□ □□□ Amazon CloudWatch □□□ □□□□□. □□□□ □□□□ Amazon Simple Notification Service(Amazon SNS) □□□ □□□ □□□ 24□□ □□□ EC2 □□□□□ □□□□□ □□□.

C. □□□ □□□□ □□ □□□ Amazon CloudWatch □□□ □□□□□. Amazon Simple Queue Service(Amazon SQS) □□□□ □□□ □ □□□ □□□□□. □□□ □□□□ □□□ □□□□ □□□□ □□□□ □□□ □□ □□□ Amazon EventBridge □□□ □□□□□.

D. AWS Lambda □□□ □□ CloudWatch Logs □□□ □□□□□. □□□ □□□□ □□□ EC2 □□□□□ □□□ □□□□ □□□ □ □□□□ □□□□□ □□□ □□□□□. □ □□□ □□ □□ □□□□□ □□□□ □□ Lambda □□□ □□□□ Amazon EventBridge □□□ □□□□□.

Answer: (SHOW ANSWER)

"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

NEW QUESTION: 130

□□□ AWS Lambda □□□ Amazon S3 □□□ □□□ □□□□□□. □□□ □ □□□ □□□□□ □□ □□□ □□□ □ □□□□ □□ Lambda □□□ □□□□□. □□ S3 □□□□ Lambda □□□ S3 □□ □□□ □□□□□ □□□ S3 □□□ □□□ □□ □□ □□ □□ □□ S3 □□ □ Lambda □□□ □□□ □□ □□□□ □□ □□□ □□□ Amazon DynamoDB □□□□ □□□□□.
Lambda □□□ □□ □□□□ S3 □□□□ □□ DynamoDB □□□□ □ □ □□ □□□ □□□□.□□□ □□ DevOps □□□□□ □□ □ S3 □□□ □□□□□ □□ □□□ □□□ □ Lambda □□□ □□□□ □□□□ □□ □□□□□□.
□ □□□ □□□□ □□□□ □□□□□?

A. Lambda □□□ □□□□ □□ S3 □□□ □□□ □□□ □□□ □ □□ □□□ □□□□□.

B. Amazon S3□ S3 □□□ □□ Lambda □□□ □□□ □ □□ □□□ □□□□ □□ Lambda □□□ □□ □□□ □□□ □□□□□.

C. Amazon Simple Queue Service(Amazon SQS) □□□□ Lambda □□□ OnFailure □□□□ □□□□□.

D. Lambda □□□ /tmp □□□ □□□ □□□□□□□ □□□ S3 □□□ □□□ □□□ □□□ □ □□ □□□ □□□□□.

Answer: B (LEAVE A REPLY)

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]

NEW QUESTION: 131
□□□□ □□□□ □□ □□□□□□ □□□□□ □□□. □□□ □□□ GitHub □□□□□□ □□□□□□ □□□ □□□□□. □□□ □□□□□□ □□ □□□ Amazon Elastic Container Service(Amazon ECS)□ □□□□ □□□. Amazon EC2 □ AWS Lambd □□□□□ □ □□ □□ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. Amazon ECS□ □□ AWS CodePipeline□ □□□□□. □□ □□□□□□ Amazon EC2 □ Lambda.
B. AWS CodeDeploy□ □□ AWS CodePipeline□ □□ □□□□ □□□□□.
C. AWS Elastic Beanstalk□ □□ AWS CodePipeline□ □□ □□□□ □□□□□.
D. GitHub □□□ □□ AWS CodeDeploy□ □□□□ □□□□□□□ □□□□□.
Answer: (SHOW ANSWER)
https://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-steps.html

NEW QUESTION: 132
□ □□□ Amazon ECR□ □□□ □□□ □□□ □□□ □□□□ □□ □□ □□□ Docker □□□□ □□ □□□□□. □ □□□ □□□□ □□□ □□□□□ □□□□□□ □□□□ □□ □□□□ □□□□□ □□□.
□□ □□□□ □ □□ □□□ □□□□□?
A. S3 □□ □□ □□□ □□□□□(□□ □□).
B. □□□ □□□□ □□□ □□ ECR □□ □□ □□□ □□□□□.

C. Lambda□ □□□□ □□□□□ □□□□□.
D. □□□ □□□ □□□ □□□□□ □□□□□.
Answer: B (LEAVE A REPLY)
ECR Lifecycle Policies automatically manage image retention based on tag status, age, or count. They execute natively within the ECR service, requiring no external management or scripts - the least overhead and AWS-recommended cleanup method.

NEW QUESTION: 133
□ □□□ Amazon S3 □□□ □□□□ □□ □□□□□ □□□□□□. □ □□□□□ AWS CloudFormation□ □□□□ □□□□□.
CloudFormation □□□□ S3 □□□ □□ □□□□ □□□□ □□□□ □□□□ □□□ □□ □□□□ □□□□□.
□□□ □□□□□ □□□ □□□ □□□ □□□ □□□□ □□□ □□ CloudFormation □□□ □□□□ □□ □□□□ □□□. □□□ CloudFormation□ □□□ □□□□ □□□ □ □□□ □□□□□.
□□ □□□ □□ □□□ □□□□ DevOps □□□□□ □ □□□ □□ □□□ □□□□□□ □ □□□ □□□ □□□ □ □□□□?
A. S3 □□□ □□ □□□□ □□□ □□ □□□ □□□ □□□□□□. Cloud Formation □□□□ □□□□ S3 □□ □□□□□ WebsiteConfiguration□ □□□ □□□□□.
B. S3 □□□ □□ □□ □□ □□□ □□□□□□. RequestType□ Delete□ □ □□□ □□□□□ □□□□ □□□ □□ □□□□ AWS Lambda □□ □□□ □□□□□.
C. □□□ □□ □□□□ □□ □□□ □□□□ □□ □□□ □□□□□□. RemoveOnDeletion □□ □□□□ □□□ □□ □□□ □□□ DeletionPolicy □□□ □□□□□.
D. S3 □□□ □□ □□ □□ □□□ □□□□□□. CloudFormation □□□□□ S3 □□ □□□□ □□□□ Empty □□ □□ DeletionPolicy □□□ □□□□□.
Answer: (SHOW ANSWER)
Step 1: Understanding the Deletion FailureThe 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 DeletionTo 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.

NEW QUESTION: 134
DevOps □□ Amazon Elastic Kubernetes Service(Amazon EKS) □□□□□ □□□□□□□ □□□□□□□□ □□□□ □□□□. □□ □□□□□ □□□ □□ □□□ □□□□□.
DevOps □□ □□ CPU □□□□ □□ □□□□□□□ Pod□ □□ □□□ □□□□□□ □□□. DevOps □□ □□ □□□□□ Kubernetes Metrics Server□ □□□□□□.
□□ □□□□ □□ □□ □□□□ □□□□ □□□ □□ □□□ □□□ □ □□□□?

A. EKS □□□□□ □□□ □□□ □□□ □□ □□ □□□ □□□□□. □□ □□ □□□ □□ CPU □□□□ □□ □□□ □□□□ □□ □□□ □□ □□ □□ □□□ □□□□□ □□ □□ □□□ □□□□□.
B. □□□□□ Kubernetes □□□ Pod □□ □□□(HPA)□ Kubernetes □□□ Pod □□ □□□(VPA)□ □□□□□. □□ CPU □□□□ □□ HPA□ □□□□□ □□□□□.
VPA□ □□ □□ □□□ □□□□□ □□□□□.
C. AWS Systems Manager AWS-UpdateEKSManagedNodeGroup Automation □□□ □□□□□.
NodeGroupDesiredSize, NodeGroupMaxSize, NodeGroupMinSize□ □□ □□□ □□ □□□ □□ □□□□ □□ □□□□□.
D. □□□□□ Kubernetes □□ □□ □□ □□□(HPA)□ Kubernetes □□□□ □□ □□□□ □□□□□. □□ CPU □□□□ □□ HPA □ □□□□□ □□□□□. □□□□ □□ □□□□ □□ □□ □□□ □□□□□ □□□□□.
Answer: D (LEAVE A REPLY)
To scale microservice Pods based on CPU utilization, the Kubernetes Horizontal Pod Autoscaler (HPA) uses the Kubernetes Metrics Server to monitor resource usage and automatically adjusts the number of Pods.
However, scaling Pods may require additional nodes if the current node capacity is insufficient.
* The Cluster Autoscaler works with EKS managed node groups to add or remove worker nodes based on pending Pod requirements and resource usage.
* By deploying both HPA and Cluster Autoscaler, the system can automatically scale Pods and add nodes as necessary, ensuring efficient resource utilization and availability.
* Configuring the Cluster Autoscaler with auto-discovery allows it to manage node groups without manual intervention, reducing operational effort.
* Option A only scales nodes based on node CPU utilization, not Pods.
* Option B uses VPA recommender mode, which only suggests resource changes and does not scale automatically.
* Option C involves manual updates and is not automated scaling. Therefore, option D provides the most operationally efficient, fully automated scaling solution.
Reference from AWS Official Documentation:
* Kubernetes Horizontal Pod Autoscaler: " HPA automatically scales the number of Pods based on observed CPU utilization or other metrics. " (Kubernetes HPA)
* Cluster Autoscaler on Amazon EKS: " The Cluster Autoscaler automatically adjusts the size of the Kubernetes cluster when there are Pods that fail to run due to insufficient resources or when nodes in the cluster are underutilized. " (AWS EKS Cluster Autoscaler)

NEW QUESTION: 135
□ □□□ AWS Organizations□ □□□□ AWS □□□ □□□□ □□□□ □□□□. □□□ AWS Cloud Formation StackSets□ □□□□ □ □□ □□□□ AWS Config□ □□□□□□□. □□□ □□□□ AWS Config□ □□ □□□ □ □□ □□□□ □□□□□ □□ □□□ AWS Config□ □□ □□□ □□□ □□□□ □□□□□□. DevOps □□□□□ □□□ □□□□ □□□□ □□□. □□ □□ □□□ □□ □ □□□ □□ AWS □□ □□□ □□ □□□□ □□□□ □□ □□□ □□□ AWS Config □□□ □□ □□□ □□□□□ □□□□ □□ □. □□ □□□ □□□□ □ □□ □□□□ □□ □□□□ □ □□ □□□ □□□ □ □□□ □□□. □ □□ □□□ □□□□ AWS Config □□□ □□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. AWS Config □□ □ □□ □□□ □□□ CloudFormation □□□□ □□□□□. CloudFormation StackSets□ □□□□ □□ □□□ □ □□□ □□□□ □□□□□.
B. AWS Config □□ □ □□ □□□ □□□ AWS Config □□□ □□ □□□□□. CloudFormation StackSets□ □□□□ □□ □□□ □□ □□ □□ □□□□□.

C. AWS Config □□ □ □□ □□□ □□□ CloudFormation □□□□ □□□□□. AWS Config□ □□□□ □□□ □□□ □□□□ □□ □□ □□□□□.
D. AWS Config □□ □ □□ □□□ □□□ AWS Config □□□ □□ □□□□□. AWS Config□ □□□□ □□□ □□□ □□□□ □□ □□□□□.
Answer: D (LEAVE A REPLY)
The correct answer is D. Creating an AWS Config conformance pack that contains the AWS Config rules and remediation actions and deploying it from the delegated administrator account by using AWS Config will meet the requirements. A conformance pack is a collection of AWS Config rules and remediation actions that can be easily deployed as a single entity in an account and a region or across an organization in AWS Organizations1. By using the delegated administrator account, the DevOps engineer can centrally manage the conformance pack and prevent non-administrator users from modifying it in the member accounts. Option A is incorrect because creating a CloudFormation template that contains the AWS Config rules and remediation actions and deploying it from the Organizations management account by using CloudFormation StackSets will not prevent non-administrator users from modifying the AWS Config rules in the member accounts.
Option B is incorrect because deploying the conformance pack from the Organizations management account by using CloudFormation StackSets will not use the trusted access feature of AWS Config and will require additional permissions and resources. Option C is incorrect because creating a CloudFormation template that contains the AWS Config rules and remediation actions and deploying it from the delegated administrator account by using AWS Config will not leverage the benefits of conformance packs, such as simplified deployment and management. References:
Conformance Packs - AWS Config
Certified DevOps Engineer - Professional (DOP-C02) Study Guide (page 176)

NEW QUESTION: 136
□□□□ □□□□□□□ □□□□ □□□□. □□□□□□□ □□□ AWS □□□□ □□□□ □□□. □□□□□□□ □□□□ □□□ □□□ □ 1□□ □ □□ □□□□ AWS Organizations OU□ □□□□□□□.
□□□ □□□ □□ □□ □□□ □□□□ □□□. □□ □□□□ □□ AWS □□□□ □□□□ □□□□ □□□□□ □□□□ □□□ SCP□ □□□ □□□.
□□□ □□□□□ AWS Identity and Access Management(IAM) □□□ □□□□ □□□ □□□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. IAM Access Analyser□ □□□□ □□□□ □□□□ SCP□ □□□□□. □□□ □□ OU□ □□□□. □□□ □ OU□ □□□□□. □ SCP□ □ OU□ □□□□□. □ OU□□ □□ FullAWSAccess SCP□ □□□□□.
B. IAM Access Analyser□ □□□□ □□□□ □□□□ SCP□ □□□□□. □□□ □□ OU□ □□□□. □□□ □□□ OIJ□ □□□□□. □ SCP□ □ OU□ □□□□□.
C. IAM Access Analyser□ □□□□ □□□□ □□□□ SCP□ □□□□□. □ SCP□ □□□ □□□ □□□□□.
D. IAM Access Analyser□ □□□□ □□□□ □□□□ SCP□ □□□□□. □□□ □□ OU□ □□□□. □□□ □ OU□ □□□□□. □ SCP□ □□□ □□□ □□□□□. □ OU□□ □□ FullAWSAccess SCP□ □□□□□.
Answer: A (LEAVE A REPLY)
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.
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

NEW QUESTION: 137
□□□ Microsoft Active Directory□ AWS Directory Service□ □□ □□ □□□(IdP)□ □□□□□. □□□□□ □□ □□□□ AWS CloudFormation□□ □□□□ □□□□ □□□.
DevOps □□□□□ □□□□□□□ □□□□□ □□ Windows □□ Amazon EC2 □□□□ □□□ □□□□ □□□. DevOps □□□□□ EC2 □□ □□□, IAM □□, EC2 □□ □□ □ EC2 Auto Scaling □□□ □□□□ CloudFormation □□□□ □□□□□□. DevOps □□ □□□ □□ EC2 □□□□□ AWS Managed Microsoft AD □□□□□ □□□□ □□□□ □□□□ □□□□ □□□.
□□ □□□□ □□ □□□□□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. CloudFormation □□□□□ □□ □□□□□ □□ □□□□□ □□□□ EC2 □□□□□ AWS Managed Microsoft AD □□□□ □□□ □ AWS::SSM::Document □□□□ □□□□□. □ SSM □□□ □□□□□ SSMAssociation □□□ □□□□□ □□ □□□□ □□□□ □□□. EC2 □□□□□ □□□□ IAM □□□ AmazonSSMManagedInstanceCore □ AmazonSSMDirectoryServiceAccess AWS □□□ □□□ □□□□□.

B. CloudFormation □□□□□ □□ □ □□□□ □□ □□□ □□□□□ □□ □□□□ □□□□□□□. AWS::SSM::Association □□□ □ □□□□ AWS-JoinDirectoryServiceDomain □□□ Runbook□ □□□ □□□ □□ EC2 □□□□□ □□□□□. AWS Managed Microsoft AD □□□□□ □□□□ □ □□□ □□□□□ □□□□□. EC2 □□□□□ □□□□ IAM □□□ AmazonSSMManagedInstanceCore □ AmazonSSMDirectoryServiceAccess AWS □□□ □□□ □□□□□.
C. □□ AWS Managed Microsoft AD □□□ □□ □□ □□□ AWS Secrets Manager□ □□□□□. CloudFormation □□□□□ AWS::SSM::Association □□□□ □□□□ AWS-CreateManagedWindowsInstanceWithApproval □□□ Runbook□ EC2 Auto Scaling □ □□ □□□□□. □□□□ □□□□□ Secrets Manager□ □□□□□ □□ ARN□ □□□□□. EC2 □□□□□ □□□□ IAM □□□ AmazonSSMDirectoryServiceAccess □ SecretsManagerReadWrite AWS □□□ □□□ □□□□□.
D. □□ AWS Managed Microsoft AD □□□ □□□ □□ □□□ AWS Secrets Manager□ □□□□□. CloudFormation □□□□□ □□ □ □□□□ □□□□□ EC2 □□ □□□□ □□□□□□□. Secrets Manager□□ □□□ □□ □□□ □□□□ AWS Managed Microsoft AD □□□□ □□□□□ □□□ □□□□ □□□□□. EC2 □□□□□ □□□□ IAM □□□ AmazonSSMManagedInstanceCore □ SecretsManagerReadWrite AWS □□□ □□□ □□□□□.
Answer: B (LEAVE A REPLY)
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.

NEW QUESTION: 138
□ □□□ □□□□□□□ Amazon Elastic Container Service(Amazon ECS)□ □□□□□□□ □□□□. □ □□□ □□ □□□□ us-west-2 □□□ Amazon CloudWatch□ □□□□□ □□□. □□□ □□ Amazon ECS□ CloudWatch □□ □□□ □□□□ □□□□.
ECS □□ □□□ □□□ □ □□□□ □□□ □□ □□ □□□ □□□□□.
"□□□ my-service-name□ □□ □□ □□□ □□□□ □□□□ □□□□□ □□□ □□□ □□□ □ □□□□□." ECS □□ □□□□ □ □□ □□ □□□□ □□ □□□ □□□□ □□□□.
"logConfiguration": {
"logDriver": "awslogs",
"options": {
"awslogs-create-group": "true",
"awslogs-group": "awslogs-mytask",
"awslogs-region": "us-west-2",
"awslogs-stream-prefix": "awslogs-mytask",
"mode": "non-blocking",
"max-buffer-size": "25m"
}
}

ECS □□□□□ Amazon EC2 Auto Scaling □□□ □□□□ □□□ □□□ □□□ □□□□□. EC2 □□□□□ Amazon ECS□ □□□ □ AMI□ □□□□□.
□□ □□□□ □□□ □□□□□?
A. ECS □□□ IAM □□□ □□□□ logs:CreateLogStream □ logs:PutLogEvents □□□ □□□□□.
B. ECS □□ □□□ □□ □□□ □□□□□ □□□□□.
C. ECS □□□□ □□□□□ IAM □□□ □□□□ logs:CreateLogStream □ logs:PutLogEvents □□□ □□□□□.
D. awslogs-create-group □□□ false□ □□□□ ECS □□ □□□ □□□□□.
Answer: C (LEAVE A REPLY)
When using the awslogs log driver with Amazon ECS on EC2, CloudWatch Logs permissions must be granted to the ECS container instance IAM role , not the task definition or infrastructure role. The ECS agent running on the EC2 instances is responsible for creating log streams and pushing log events to Amazon CloudWatch Logs on behalf of the containers.
In this scenario, the task definition is correctly configured to automatically create the log group (awslogs- create-group: true) and send logs to the specified Region. However, the error occurs because the EC2 container instances do not have sufficient IAM permissions to perform the required CloudWatch Logs API calls. As a result, ECS cannot place the task, and it reports that no container instance meets the requirements.
According to AWS documentation, the container instance IAM role must include the following permissions when using the awslogs driver:
* logs:CreateLogStream
* logs:PutLogEvents
* (and, when creating log groups automatically) logs:CreateLogGroup
Option C correctly addresses the root cause by updating the container instance IAM role . Option A is incorrect because the ECS infrastructure or service role is not used to write logs. Option B is unrelated to permissions and does not resolve the issue. Option D would fail because the log group does not already exist, causing task startup to fail.
Therefore, modifying the container instance IAM role is the correct solution.

NEW QUESTION: 139
DevOps □□ Amazon EC2 □□□□□□ □□□□ □□ □□□□ □□□□□. DevOps □□ Amazon Route 53□ □□□□ □□□ □□□ □□□□ □□ □□ □□□□ □□□ □□□ □□□□□. □ □□ □□□□ □□ □□□□□ □□□ □ □□ □□□ □□ Amazon Elastic File System(Amazon EFS)□ □□□□□. □□ □□□□ □□ □□ □□ □□ □□□□ □□□□ □□□□. DevOps □□ □□ □□ □□ (MTTR)□ □□□□ □□ □ EC2 □□□□□ □□ □□□□ □ □□□□□ □□□□□. DevOps □□ □□ □□□□ □□□□ EC2 □□□ □□ □□ □□ □□□□□ □□□□. □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □□□□□□ □□ □□□(ALB)□ □□□□□. □□ Route 53 □□□□ □□□□□□ ALB□ □□□□□ □□□. □ EC2 □□□□□ □ □ □□ □□□ □□□□□. □ □□ □□□ □□□□□□ □□ □□□ □□□□□. □ □□ □□□ □□□ ALB □□□□ □□□□□. □□ □□ □□□ □□□□ 99□ □□□□□. □□ □□ □□□ □□□□ 1□ □□□□□.
B. □□ Route 53 DNS □□□□ □□□ □□□ □□□ □□□□ □□□□□□□. □□ DNS □□□□ □□□□ 99□ □□□□□. □□□ □□□□ □□ □□ EC2 □□□□□ □□□□ □ DNS □□□□ □□□□□. □ DNS □□□□ □□□□ 1□ □□□□□. □ □□□□ □□ □□□□ □□ □□□ □□□□□.
C. □□□□□□ □□ □□□(ALB)□ □□□□□. □□ Route 53 □□□□ □□□□□□ ALB□ □□□□□ □□□. □ EC2 □□□□□ □ □ □□ □□□ □□□□□. □ □□ □□□ □□□□□□ □□ □□□ □□□□□. □ □□ □□□ □□□ ALB □□□□ □□□□□. □□ □□ □□□ □□□□ 100□□ □□□□□. □□ □□ □□□ □□□□ 0□□ □□□□□.

D. □□ Route 53 DNS □□□□ □□□ □□□ □□□ □□□□ □□□□□□□. □□ DNS □□□□ □□□□ 100□□ □□□□□. □□ □ □□□□ □□ □□ EC2 □□□□□ □□□□ □ DNS □□□□ □□□□□. □ DNS □□□□ □□□□ 0□□ □□□□□. □ □□□□ □□□□□□ □□ □□□ □□□□□.

Answer: D (LEAVE A REPLY)

NEW QUESTION: 140

□□ □□□ □□□ Amazon Elastic Block Store(Amazon EBS) □□ Amazon EC2 □□□□□ □□ □□□□□. □□ □□□□□□ □□ □□□ □□□ □□ DevOps □□□□□ EC2 □□□□ □□ □□□□ □□□ □ □□ □□ □□□ □□□□□ □□□ □□□□.
□□□ □□□ □□□□ □ □□□□?

A. □□□ Amazon EventBridge □□□ □□□□ EC2 □□□□□ □□□□ □ □ □□□□□ □□□□ □□□ □□□□ AWS □□□ □□ □ □□□ Runbook□ □□□□□. □□□□□ □□□□□ □□□ □□ Runbook□ □□□□□ □□ □□ □□□ □□□□□.

B. □□ □□□□□□ EC2Auto Recovery□ □□□□□□. □□ □□ □□ □□□□ □□□ □□□□□ □□□□ AWS Config □□□ □□ □□□.

C. □□ □□ □□ □□□ □□□ □□ □□ □□ □□ □□ EC2 □□□□□ □□□□□□. □□□□□ EC2 □□□□ □□ □□□ □□□ □□ □□□ □□□□ Amazon CloudWatch □□□ □□□□□.

D. □□ □□ □□□□ □□□ □ EC2 □□□□□ □□□□ □□□□ AWS Systems Manager Automation Runbook□ □□□□□ AWS □ □ Amazon EventBridge □□□ □□□□□.

Answer: D (LEAVE A REPLY)

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

NEW QUESTION: 141

□ □□□ AWS CodeConnections □□ Git □□□□ □□□□□□ □□□ □□□□ □□□□. □ □□□ □ □□□□□ □□ □ □□ □□ □□ □□□□□ □□□□□ □□□. □□, □□□□ □□□□ □ □□□□□ □□□ □□□ □□□□□ □□□ □□□. □□, □□□□ □ □□ □ □□□□□ □□□ □□ □□□ □□□ Amazon S3 □□□ □□□□□ □□□. □□□ □□ □□□ □□□□ □□□ □□□ □□□ □□? (□ □□ □□)

A. □□□ □□□ □□□□ □□ □□□ □□□□ □□□ □□□ □□ buildspec.yml □□□ □□□□.

B. AWS CodePipeline□ □□□ □□□ □□ □□□□□□ □□□□□. □ □□□□□ □□□□□ □□□□□ □ □□□□□□ □□□□ □□□□ □□□□□. □□□ □□□ □□□□ □□ □□□ □□□□□.

C. □□□ □□□ □□□□ □□□□□ □□□□ □□□□ □□□ □□ buildspec.yml □□□ □□□□.

D. □□□□ □□□□ □ □□□ □□□□ □□ □□□□ □□□□ IAM □□□ □□□ □□□□.

E. □□□□ □□□ AWS CodeBuild □□□□□ □□□□□. □ □□□□□ □□□□□ □□□□□ □ □□□□ □□□□□ □□ □□□□ □□□□□□. □□□ □□□ □□□□□ □□ □□ □□□ □□□□□□.

F. □□□ □□□ □□□□ □□ □□□ □□□□ □□ □□□ □□ appspec.yml □□□ □□□□.

Answer: C,D,E (LEAVE A REPLY)

NEW QUESTION: 142

□ □□□ Amazon S3 □□□ 100GB□ □□ □□□□ .csv □□□□ □□□□ □□□□. SQL □□□□ □ □□□□ □□□□ □□□□ □ □□□ □□□□□ □□□. □□ SQL □□□□ .csv □□□ □□□□□□ □□□□□ □□□□ □□□□ □□□□ □□□. □□□ □□ □ □□ □□ □□ □□□□ □□□ □ □□ □□ □□□ □□□□□? (□ □□ □□)

A. AWS X-Ray□ □□ □□□□ □□□□□ □□□□□□.
B. Amazon DynamoDB□ □□ □□□□□ □□□□ □□□□□.
C. Amazon QuickSight□ □□ □□□□ □□□□□ □□□□ □□□□□□.
D. AWS Glue □□□ □□□□□ □□ □□□□□ □□□□ □□□□□.
E. Amazon Athena□ □□□□ □□□□ □□□□□.
F. Amazon Redshift□ □□□□ □□□□□.
Answer: B,C,E (LEAVE A REPLY)

NEW QUESTION: 143
□□□□ □□ □□ □□(PII)□ □□□ □□□□ Amazon S3 □□□ □□□□ □□□□□□□ □□□□. □□ □□□□ AWS Key Management Service(AWS KMS) □□ □□□ □□ □□□□□□. □□ AWS □□□□ AWS Cloud Formation □□□□□ □□□□□.
DevOps □□□□□ □□ AWS □□□ □□□□□□□ □□ □□ □□□ □□□□ □□□. □□ □□□ S3 □□□ □□ □□□□ □□□□ □□□ S3 □□□□ □□□□ □ □□ □□□□□□□ □□□.
□□□ □□ PII□ □□□□□ □□ □□□□ □□□□ PII□ □□□□□ □ □□□. □ □□□ □□□□ □□ □□ KMS □□ □□ □□ □□ □□ □□□□□□ □□□.
DevOps □□□□□ □□□ □□ □□□ □□□□ □□ □□ □□ □□□ □□□□ □□□? (2□ □□)
A. □□□□ □□□ S3 □□□□ Amazon Macie□ □□□□□□. AWS Step Functions □□ □□□□ □□□□ □□ □□□ □□□□ □□ □□□ S3 □□□ □□□ □□□□ □□ □□ PII□ □□□□□. □□ □□□ □□□ □□□□ □□□ KMS □□ □□ □□ □□ □□□ □ □□□□. □□ □□ □□□ □□ □□□ KMS □□ □□ □□□ □□□ □□□□□.
B. □□□□ S3 □□□ □□ S3 □□ □□ S3 □□□ □□□□□. □□ S3 □□□□ Amazon Macie□ □□□□□□. AWS Step Functions □□ □□□□ □□□□ □□ □□□ □□□□ □□□ □□ PII□ □□□□□. □□ S3 □□. □□ □□□ □□□ □□ □□□ KMS □□ □□ □□□ □ □□ □□ □□□ □□□□□.
C. □□□□ S3 □□□□ □□ S3 □□□□ □□□ □□□□□ S3 □□ □□ □□□ □□□□□. □□ □□□□ □□ PII□ □□□□□ AWS Lambda □□□ □□□□□. S3 GET □□□ Lambda □□□ □□□□□ S3 □□ Lambda□ □□□□□. Lambda □□□ 1AM □□ □ □□ □□□ KMS □□ □□ □□□ □ □□ □□ □□□ □□□□□.
D. □□ □□□ CloudFormatlon □□□□□ □□ □□□ □□□□□. □□□□ □ □□ AWS Step Functions □□ □□□ □□□□□ Amazon EventBridge □□□ □□□□□.
E. □□ □□□ CloudFormation □□□□□ □□ □□□ □□□□□. □□□□ □ □ □□□□□ Amazon EC2 □□□□□□ □□ □□□ □□□□ S3 □□ □□ □□□ □□□□□.
Answer: A,D (LEAVE A REPLY)
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.
References:
Amazon Macie
AWS Step Functions
AWS CloudFormation Templates

NEW QUESTION: 144
□ □□□□ □□□ □□□□ □□□□□(SaaS) □□□□□□□ □□ □□ □□(Proof of Concept)□ □□□□ □□□□. □ □□□□□□ □ AWS Organizations □□ □□□ □□ □□ □□ AWS □□□ □□□□.
□□□□ □□ □□□ □□□ AWS □□□□ □□ □□□ □□ IAM □□□ □□□□ □□□. □□□□ □□□□□ □□□ □□ □□□ □ □□□ □□□ □ □□ □□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□□ □□ □□□ □□□□ IAM □□□□ □□□□□. □□□□ □□□ □ □□□ □□ □□□ □□ □□ □□□ □□□□□. □□ □ □ □□□ □□□ □□ □□□□ □□□□□.
B. □□□□ IAM □□□ □□□□□. PowerUserAccess □□ □□□ IAM □□□ □□□□□. □□□ □□□ □□□ □□(MFA)□ □□□□ □.
C. □□□ □□ □□□ SCP□ □□□□□. □□□□ □□□ □□□□ □□ iam:*□ □□ □□ □□□□ SCP□ □□□□□.
D. □□□□ □□ □ □□□ □□□ □ □□□ □□□ IAM □□□ □□□ □□ IAM □□□ □□□□□. □□□ □□ □□□ □□□□ □□ □□□. □□□□□ □□ □□ □□□ □□□□□.
Answer: (SHOW ANSWER)
To allow only creation/configuration of service-linked roles , you need a way to tightly scope what IAM actions the developer can perform.
The most AWS-appropriate mechanism for "delegate limited IAM admin" is a role with a permissions boundary :
* A permissions boundary sets the maximum permissions the role (and any roles it creates, depending on design) can ever have. This is a standard pattern to safely delegate IAM tasks without granting broad IAM administration.
* You can define the role's policy + boundary so the developer can call only the APIs required for service-linked roles (for example actions like creating service-linked roles and passing only allowed AWS service principals), while preventing general role/policy creation outside that scope.
Why the others don't meet "service-linked roles only":

* A is vague ("common services") and cross-account access doesn't inherently restrict to only service- linked roles. It's also more operational overhead than needed for a single dev account use case.
* B PowerUserAccess is far too broad and does not restrict to service-linked roles.
* C An SCP with Deny iam:* would block IAM entirely (including what the developer needs). Even if refined, SCPs set account-wide guardrails and are not the right tool to grant a single developer the precise ability to create only service-linked roles.

NEW QUESTION: 145
□□□□ MySQL □□ Amazon Aurora □□ AZ DB □□□□□ □□□□□□□ □□□□ □□□□□□□ □□□□. □□ □□□ □□ □ □ □ □□ □□□□ □□□□□□□. DevOps □□□□□ □□ □□ □ □□ □□□□□□ □□□□□ □□□ □□□ □□□ □□□□□ □ □□□.
□□ □□□□ □□ □□□□□?
A. □□ □□ □□ Amazon Route 53 CNAME□ □□ □□□ □□ □□□□ □□ □□□□□□ □□ □□□□□□ □□ □□□□□ □□ □. AWS CloudTrail□ Amazon RDS □□ □□□ □□ Amazon SNS □□□ □□□□ □□ □□□ □□□□ □□□ □□□□□ □□ □□ □□□ □□□□ AWS Lambda □□□ □□□□□.
B. □□ □□□□□□ □□□□□ □□□□□ Aurora □□□ □□ □□□□□□ □□□□□. □ □□□□□□ □□□□□ □□□□□□□ □□□□□. AWS Lambda □□□ □□□□ □□□ □□□□□ □□□□ □□ □□□ □□□□□ □□□□□ □□□ □□ □□□□□□ □□□□□ AWS CloudTrail□ □□□□□.
C. □□□□□□□ AWS CloudFormation □□□□ □□□□ □□□□ □□□□, □□□□ □□□□ □□□ □□□□□□, □□□□□□□ □□ □□□ □□□□□ □□□□□ AWS Lambda □□□ □□□□□. □□ □□□□ □□□ □ □ Lambda □□□ □□□□ Amazon CloudWatch □□□ □□□□□.
D. AWS Systems Manager Parameter Store□ Aurora □□□□□□ □□□□□. □□□□□□ □□□ □□□□ AWS Lambda □□□ □ □□□ □□□ □□□□□ □□□□ AWS Systems Manager Parameter Store□ □□□ □□□□□ URL□ □□□□□□ Amazon EventBridge □□□□ □□□□□. □□□□□□ □□□ □□□ □□ Parameter Store□□ □□□□□□ □□ □□□□□ □□□□□□□ □□□□□.
Answer: D (LEAVE A REPLY)
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/

NEW QUESTION: 146
□ □□□ □□ □□□ Amazon S3 □□□ □□□□ □□□□□. □ □□□ □□ □□ □□□□ □□□□ □□□ □□ □□ □□□ □□ □ □□ □□□□ □□□.
□□ □□□□ □□ □□ □□ □□□□ □□□ □□ □□□ □□□ □ □□□□?
A. Amazon S3 □□ □□□ □□□ □□□□□□. □□□ □□□ Amazon Aurora □□□□□□□ □□□□□. Aurora □□□□□□□□ SQL □□□ □□□□ □□□ □□□ □□□ □□□□□.
B. Amazon S3 □□ □□□ □□□ □□□□□□. Amazon Athena□ □□□□ □□□ □□□ □□□ □□ □□□□ □□□□□. Athena □ □□□□ SQL □□□ □□□□ □□□ □□□ □□□ □□□□□.
C. □□ S3 □□ □□□ □□□□ □□ AWS Lambda □□□ □□□□□. □□□ ID, S3 □□ ID, □□ □□ □□□ □□ □□□ □□□ Amazon Aurora □□□□□□□ □□□□□ Lambda □□□ □□□□□. Aurora □□□□□□□□ SQL □□□ □□□□ □□□ □□□ □□□ □□□□□.

D. □□ S3 □□ □□□ □□□□ □□ Amazon CloudWatch Logs□ □□ □□□□ □□□□□. CloudWatch Logs□ □□ □□□□ □□□ □ □□□ ID, S3 □□ ID, □□ □□ □□□ □□ □□□ □□□ Amazon Managed Service for Apache Flink □□□□□□□ □□□□□.
□□□ □□□ □□□ □□ □□□□ □□□ □□□ □□□□□.
Answer: (SHOW ANSWER)
Amazon S3 can generate server access logs that record detailed information about each request, including requester, bucket, key, operation, time, and status. These logs are written as objects to an S3 bucket. To analyze access patterns, the simplest and most serverless approach is to use Amazon Athena directly on those logs without building ingestion pipelines or databases.
Option B enables S3 server access logging and then creates an Athena external table over the log bucket.
AWS provides standard log formats and even example schemas for S3 access logs. The analytics team can run ad hoc SQL queries to count the number of accesses per object per time period, filter by user, and perform aggregations, all without provisioning compute or managing databases.
Option A requires ingesting logs into Aurora, which adds ETL complexity and ongoing database management. Option C requires a Lambda function for every access event plus DB writes, which is more complex and potentially expensive at scale. Option D uses CloudWatch Logs and Managed Flink, which is more suited for streaming analytics and is significantly more complex than necessary for monthly summary reports.
Therefore, Option B provides the required analysis with the least development and operational effort.

NEW QUESTION: 147
□□□ AWS □□□□ □□ □□ □□□□□□□ □□□□□□. □□ □□□□□□□ IAM □□□ □□□□ □□ AWS □□□ □□□□□ □□. □□ □□□ AWS Organizations□ □□□ □□□ □□□□.
□□ □□ □□□□ □□ □□ AWS □□□ □□□□ □□ □□□□□□□ IAM □□□ □□□□□ □□□ □ □□ □□□ □□□□□□.
□□□ □□□ □ □□ □□□ IAM □□ □□□ □□□□ □□ □□□□□□□ IAM □□□ □□□□ □□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □□ □□□□□□□ IAM □□ □□□ □□ Deny □□ □□□□ SCP□ □□□□□. □□□ □ □□ □□□ IAM □□□ □□□ □ □□ □ □□□□ □□□ □□□□□. □□□ □□□ SCP□ □□□□□.
B. □□□ □ □□ □□□ IAM □□□ □□ □□ □□□□□□□ IAM □□ □□□ □□ Allow □□ □□□□ SCP□ □□□□□. □□ □□ IAM □□ □□□ □□□ □□ Deny □□ □□□□□. □□ □□□□□□□ IAM □□□ □□ □ AWS □□□ IAM □□□□ SCP□ □□□ □□.
C. □□ □□□□□□□ IAM □□ □□□ □□ Deny □□ □□□□ IAM □□ □□□ □□□□□. □□□ □ □□ □□□ IAM □□□ □□ □ □ □□□ □□□□ □□□ □□□□□. □□□ AWS □□□ □□ □□□ □□□□□.
D. □□ □□□□□□□ IAM □□ □□□ □□ Deny □□ □□□□ IAM □□ □□□ □□□□□. □□□ □ □□ □□□ IAM □□□ □□ □ □ □□□ □□□□ □□□ □□□□□. AWS □□□□ □□ □□□□□□□ IAM □□□ □□ □□□ □□□□□.
Answer: A (LEAVE A REPLY)
https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_scps.html?icmpid=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.

NEW QUESTION: 148

DevOps □□□□□ □□□□ □□□□ □□□□, □□ ECR□ □□□□, □□□□ □□□□ □□□□, □□□□ □□ □□□ □□□□ □ □□ □□ □□□□ □□ □□□□ CI/CD □□□□□□ □□□□□.
□□ □□□□ □□ □□□□□?
A. □□ ECR □□□□ □□□, □□ □ □□□□ □□□□, □□ □□□ □□ □□□□ □□□□□ □□□□ □□□□□.
B. □□□□ □□□□ □□□□ □□□□ □□ □□ ECR □□□□ □□□, □□□□ □□□□ □□ □□□ □□□□ □□□, □□ □ □□ □□ □□□□□.
C. □□ ECR □□□□ □□□, □□□ □□ □□□ □□□□, □□□□ □□□ □□ □□□□ □□□, □□ □□□□ □□□□□□.
D. □□ ECR □□□□ □□□, □□ □□□□ □□□□□, □□□ □□ □□□ □□□□.
Answer: D (LEAVE A REPLY)
* ECR pull-through cache caches images from upstream repositories for resilience.
* Private repo with basic scanning ensures vulnerability detection on pushed images.
* Enabling pull-through caching on a private repo combines caching and vulnerability scanning seamlessly.
* Public repos do not support pull-through caching of upstream images.
* Replication rules are for multi-Region replication, not upstream caching.
References:
Amazon ECR Pull Through Cache
Amazon ECR Image Scanning

NEW QUESTION: 149
□ □□□ AWS Organizations□ □□□ □□□□ AWS □□□ □□□□□. □□□ □□□ □□□□ □□□ AWS □□□ □□□□ □□□ □ CI/CD □□□□□□ □□□□ □□□□.
□□□□ □□ □ □□□ □□□□ □□□□ □□ □□□ □□ □□□□. □□□ □□ □□□ □□□ □□□ □□□□ □□□□□. □□□ □□ CreateAccount API □□□ □ □□□ □□□ □ □□□ □□□□ AWS CloudTrail □□□□ □□□□□ □□□.
□□□□□ □ CloudTrail □□□□ □□□ □□□ □□□ □□□□ □□□?
A. □□□ □□□ Amazon EventBridge □□□ □□□□ □□ □□ □□□□ □□□ □□□ □□ □□□ □□□ □□□□□. □□□ □□□ □□□□ □□□□□ □□□ □□□ □□ □□□ □□□ □□□□□□□.
B. □□□ □□□ □□□ □□ Amazon EventBridge □□□ □□□ □□□□□. □□□ □□□ □□□□ □□□□□ □□□ □□ □□□ □ □□ □□□□□□□. □□□ □□□ CloudTrail □□□□ □□ □□□□ EventBridge □□□ □□□ □□□ □□□□□.
C. □□□ □□□ □□□ □□□□ □□□ □□ Amazon EventBridge □□□ □□□ □□□□.
□□□ □□□ □□□ □□□ □□ □□□ □□ □□□ □□□ □□□□ EventBridge □□□ □□□ □□□□.
D. □□□ □□□ □□□ □□ Amazon EventBridge □□□ □□□ □□□□□. □□□ □□ □□□ □□□ □□□ □□□ □□ □□□ □ □□ □□□□ EventBridge □□□ □□□ □□□□□.
Answer: A (LEAVE A REPLY)
Use cross-account EventBridge by configuring a rule in the source (automation) account to send events to the target accounts' default event buses, and grant permissions on the target default event buses to accept events from the source account. This is the standard cross-account event routing model.

NEW QUESTION: 150
□ □□□ Amazon EKS □□□□□ □□□□ □□□ □□ □□□□ □□□ □□ □□□□ □□□ □□□□ □□□. □□, API □□□ □□ □□ □□□□ □□□ □□□□□□ □□□.
□□ □□□□ □□ □□□ □□□□□□ □□□ □□ □□□ □□□ □ □□□□?

A. □□ □□ □□□ □□ AWS CloudTrail□ □□□□□ □□□ Logstash□ □□□□□.
B. CloudWatch□ □□ □□ □□ □□□ □□□□□ □□ □ Pod □□□□ □□ CloudWatch Container Insights□ □□□□□.
C. S3□ API □□ □□□ □□□□□ Kubernetes □□□ □□□□□ □□□ □□□□□.
D. OpenTelemetry□ AWS Distro□ □□□□ Amazon Redshift□ □□□ □□□□□□□.
Answer: B (LEAVE A REPLY)
Enabling EKS control plane logs to CloudWatch captures API requests. CloudWatch Container Insights collects node and pod-level performance data with no additional infrastructure. AWS recommends this integrated observability solution for minimal management overhead.

NEW QUESTION: 151
SaaS □□□ □□/□□ □□□ □□ ALB□ CodePipeline + CodeDeploy□ □□□□ ECS(Fargate)□ □□□□□. □ □□□ □□□□ □□ □□□ □□ □□□□ □□□□ □□□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. □□□ □□□□ □□□ □□□□ appspec.yaml□□ TimeBasedLinear□ □□□□□.
B. AllAtOnce □□ □□□ □□□□□.
C. TimeBasedCanary□ □□□□□.
D. ALB□□ □□ □□□□ □□□□ □□□□□.
Answer: (SHOW ANSWER)
CodeDeploy supports TimeBasedLinear traffic shifting for ECS blue/green deployments. Traffic increments by linearPercentage every linearInterval until 100%. This provides zero-downtime gradual rollout - as per CodeDeploy ECS Blue/Green Traffic Shifting documentation.

NEW QUESTION: 152
□□□ □□□ □ □ □□□ □□ □□□ □□ 6□□ □□ □□□ □□□□ □□□ □□□ □□□□□. □ □□□ □□ Application Load Balancer □□ Amazon EC2 □□□□□□ □□□□□. □□□□□ □□ □□ □□□ □□ Amazon EC2 Auto Scaling □□□□ □□□□ □. □□ □□□□ Amazon Aurora □□□□□□ □□□□□ □□□□□.
□□□ □□ □□□ □□□□ □□ □□□ □□ □□□ □□ □□ □□ □□□□□ □□□□ □□ □□□ □□ □□ □□ □ □□□ □ □□ □ □□□□ □□□.
□□□ □□□□ □□□□□□ □□□□ □□□ □□ □□□ □□□ □□□□ □□□?
A. □□ □□□□□□ Amazon DynamoDB □□ □□□□ □□□□ □□ □□ □ □□□□ □□ □□□□ □□□□□.
B. □□ □□□□□□ Aurora□ □□□□ □□ □□ □ □□□□ Amazon DynamoDB □□ □□□□ □□□□□.
C. □□ □□□□□ □□ □□ □□□□ □□ Aurora□ □□□□ □□ □□ □ □□□ □□ □ □□□ □□ □□ Aurora □□□□□ □□□ □□.
D. □□ □□□□□ Amazon Redshift□ □□□□ □□ □□ □ □□□ Amazon DynamoDB □□□□ □□□□□.

Answer: C (LEAVE A REPLY)

NEW QUESTION: 153
DevOps □□□□ □□□ □□□□ □□□□ □□□ □□□□ □□□□ □□□□. □□□□ □□ □□□ □□ 15□□ □□ □□□□ □□ □□ □□□□ □□□□ □□ □□ □□□ □□□□ □□□. □□□ □□ □□□ □□□□□ □□ □□□□ □□ □□ □□□□ □□□□ □?
A. AWS CodeArtifact□ □□□□ □□□□□. CodeArtifact □□□□ □□□ □□□ □□□□ □□□□ □□□ □□ □□□ □□□ 15□ □ □ □□□□□ □□□.
B. Amazon Elastic Container Registry(Amazon ECR)□ □□□□ □□□□□. □□□□ □□ □□ □□□ □□□□ □□□ □□□□ □□ □□ 15□ □□ □□□□□ □□□.
C. EC2 Image Builder □□□□ □□□□ □□□□□. □□□ □□□□ □□□□□ 15□ □□ □□□□□ □□ □□ □□□ □□□□□.
D. Amazon S3□ □□□ □□□□□. □□□□ □□□ □□ □□□ □□□ 15□ □□ □□□□□ □□ □□ □□ □□□ □□□□□.
Answer: B (LEAVE A REPLY)

NEW QUESTION: 154
□ □□□ AWS CodeBuild □□□□□ □□□□ □□□□□□□ □□□□ □□□□□ □□□□.□□□ □□□□ □□ AWS □□□ □□ □□ □□ □□ Amazon S3 □□□ □□□□□.
buildspec.yml □□□□ □□ □□□ □□□□ □□□□.

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

DevOps □□□□□ AWS □□□ □□ □□□□□ □□□ □□□□□ □□□□□ □ □□□ □□□ □□□□□□.
DevOps □□□□□ □ □□□ □□□□ □□ □□ □□□ □□□ □□□?
A. post_build □□□□ □□□□ --acl public-read □□□ □□□□, □□ AWS □□□□ □□ □□□ □□□□ □□ □□□ □□□□□□.
B. □□ AWS □□□ □□□ □□□ □□□□ □□□□ □□ □□ □□□ □□□ □□□□ S3 □□□ □□ □□ ACL□ □□□□□.
C. □□ AWS □□□ □□ □□□ □□□ □□□□ □□ "*"□ □□ □□ □□□□ □□□□ S3 □□ □□□ □□□□□.
D. post_build □□□□ □□□□ --acl authenticated-read □□□ □□□□, □□ AWS □□□□ □□ □□□□ □□□□ □□ □□□ □□ □□□□.
Answer: D (LEAVE A REPLY)
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

NEW QUESTION: 155
□ □□□□□ □□□ □□ □□□ □□ □□□□ □□□ □□□□ □□□ □□□ □□ □□□□ □□□□.
□ □□□ CI/CD □□□□□□ AWS CodePipeline□ □□□□ □□□□. □□ □□□□ □□ □□□□ □□□ □□□ □□□ □ □□ □□ □□□ □□□ □□□□ □□□□ □□□. □□ □□□ □□□□□□ □□□ □□ □□□ □□□ □ □□ □□□□ □□□□□?
A. AWS Device Farm□ □□□□□□ □□□□ □□□□ □□□□ □□□ □□ □□□□□.
B. □□□ □□□ □□ □□□□□□ □□ □□ □□□ □□ Amazon EC2 □□□□□ □□□□ □□□□ □□□□□. EC2 □□□ □□□ □□□□ □□□ □□ AWS Lambda □□□ □□□□□.
C. □□ □□ □□□ □□ Amazon Elastic Container Service(Amazon ECS)□ □□□□ □□□□□□ □□□ □□□□ □□□□□. AWS Lambda □□□ □□□□ ECS □□□□□□ □□□ □□□ □□□□□ □□□□□□ □□□□□.
D. □□□ □□ □□□ □□□□□□ □□ AWS Lambda □□□ □□□□ □□□□ □□□□□. Lambda □□□ □□□□□□ □□□□□.
Answer: (SHOW ANSWER)
AWS Device Farm provides a fully managed environment for automated and manual app testing across real physical devices. It integrates directly with AWS CodePipeline and automatically scales based on test concurrency. AWS documentation recommends Device Farm for mobile app testing due to its scalability and zero infrastructure management.

NEW QUESTION: 156
□□□ □□□□□□□ Amazon EC2 □□□□□□ □□□□□. □□□□□□□ □□□□ □□□ □□, □□, □□ □ □□ IP □□□ □□ □□ □□ □□□ □□□□□. □□□ Amazon CloudWatch Logs□ □□ □□□ □□□□□. □□□ □□ □□□ □□□□ □□ □□ □□ □□□ □□□□ □□□□. □□□ □□ 7□ □□ □□ □□□□ □□□ □□□ □□□ □□□. □□ □□□□ □ □□□ □□□□□'?
A. □□ □□□ CloudWatch Logs □□ □□□ □□□□□. □□□ □□□ □□□□ □□ □□□ □□□□□.
□□ 7□ □□□ □□□ □□ □□□□ CloudWatch □□□ □□□□□.
B. □□ □□□ CloudWatch Logs □□ □□ □□□ □□□ □□□□ □□ □□ □□ □□ 7□ □□□ □□□ □□ □□□□ CloudWatch □ □ □□
C. □□ □□□ □□□□ □□ 7□ □□ □□□ □□□ □□ □□□ □□ □□□□ CloudWatch Logs Insights □□□ □□□□□. □□ □□ □ □□ □□ □□
D. CloudWatch □□□□□ □□□□□. □□ □□□□ □□ □□ 7□ □□ □□□ □□□ □□ □□□ □□□ □□□□ □□ □□□ □□ □□ □□□ □□□□□.
Answer: C (LEAVE A REPLY)
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. References: 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.

NEW QUESTION: 157
□ □□□ □□□□ □□□□ □□□□ □□□□ □□□□ □□□ □□□□ □□□. □ □□□ CI/CD □□□□□□□ □□□□□ □□ □ □ □□ □□□ □□□□□ □□ □□□ □□□ □□□ □□□□□ □□□. CI/CD □□□□□□ AWS CodeBuild □□□□, AWS CodeDeploy □□, □□□ Amazon Elastic Container Registry(Amazon ECR) □□□□ □□□□ AWS CodePipeline □□□□□□□□.

DevOps □□□□□ CI/CD □□□□□□ □□□ □□□ □□□□ □□□. CI/CD □□□□□□ CRITICAL □ HIGH □□□ □□ □□□□ □□□□ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□□ □□ □□ □□□ □□□□□? (□ □□□ □□□□□.)
A. Amazon ECR □□ □□□□ □□□□□.
B. Amazon ECR □□□ □□□□ □□□□□.
C. □□□ □□□□ CRITICAL □□ HIGH □□□ □□□□ CI/CD □□□□□□ □□□ □□□ □□□□□ Amazon ECR□ □□□□□.
D. □□□ □□□ □□□□ AWS Lambda □□□ □□□□□ Amazon EventBridge □□□ □□□□□. Lambda □□□ Amazon Inspector □□ □□□ □□□□ CI/CD □□□□□□ □□ □□ □□ □□□ □□□□□ □□□□□.
E. □□□ □□□ □□□□ AWS Lambda □□□ □□□□□ Amazon EventBridge □□□ □□□□□. Lambda □□□ Clair □□ □□□ □□□□ CI/CD □□□□□□ □□ □□ □□ □□□ □□□□□ □□□□□.
Answer: B,D (LEAVE A REPLY)
Amazon ECR supports enhanced scanning powered by Amazon Inspector, which provides deeper security scanning for container images including OS and programming language package vulnerabilities.
* Enabling enhanced scanning (Option B) allows detection of CRITICAL and HIGH vulnerabilities.
* Amazon ECR emits scan completion events via EventBridge, which can trigger Lambda functions. The Lambda function can process the scan results from Amazon Inspector and programmatically approve or reject the image in the CI/CD pipeline (Option D).
* Basic scanning (Option A) is limited and does not integrate with Inspector.
* Options C and E describe functionalities not natively supported (ECR does not automatically submit Rejected status; Clair is not used in AWS ECR scanning).
Reference:
Amazon ECR Enhanced Scanning: " Enhanced scanning powered by Amazon Inspector identifies vulnerabilities in container images. " (Amazon ECR Image Scanning) Using EventBridge and Lambda for Scan Status: " ECR emits scan events that can be used to trigger Lambda functions for custom approval workflows. " (Amazon ECR Scan EventBridge)

NEW QUESTION: 158
DevOps □□□□□ Amazon EC2□□ □□□□ □□□ □□ □ □□□□ □□□□□. □ □□□□ Amazon Kinesis Data Streams□ □□ □□ □ □□□ □□□□ □□□□□. DevOps □□□□□ Amazon EC2□□□ □□□□ Kinesis □□□ □□□□□□□ □□□□□.
□□□□ □□□ □□□□ Kinesis □□□ □□□□□□□ □□□□ □□□□ □□ □□ □□□□ Kinesis □□□ □□□□ □□□□ □□ □□□. DevOps □□□□□ □□□ □□□ □□□□ □□ □□□□ □□□□ □□□.
□□ □□ □□□□ □□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. Amazon S3□ □□□ □□□□□ □□□□□ Kinesis □□□ □□□□□□□ □□□□□. Amazon EMR□ □□□□ Amazon S3□□ □□ □□□□ □□□□ □□ □□□□ □□□□. □□□ Amazon S3□ □□□□□.
B. Amazon CloudWatch GetRecords IteratorAgeMilliseconds □□□ □□□□ □ □□ EC2 □□□□□ □□□□ Kinesis □□□ □□□□ □□□ □□□□ □□□□□. Kinesis □□□ □□□□ □□ □□□ □□□□.
C. AWS Lambda □□□ □□□□□ Kinesis □□□ □□□□□□□ □□□□□. □□□ □□□□ □□□□ □□ Lambda □□□ □□□ □□□ Kinesis □□□ □□□□ □□□□□.
D. □□□ □□□□□□□ □□□□ □ □□□ □□□ □ □□□ Kinesis □□□ □□□□ □□ □□ □□ □□ □□□□ □□□□.
Answer: B (LEAVE A REPLY)
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.

NEW QUESTION: 159
□□□ □□ AWS □□ □ □□ □□□ □□ AWS□ □□ □□□□ □□□□□. □□□□ Amazon EC2 □□□□□ □□□□ AWS IoT Greengrass □□□ □□□□□. □□□ □□□ □□ □□□□□ □□□ □□ □□□□ □□□□□.
□□□ □□□ □□ □□ □ □□□□□ □□□ □□□□□ □□□□ □□□. □ □□□ DevOps □□ AWS Systems Manager□ □□□□ □□□□ □□ □ □□ □□□ □□□ □□□□□. DevOps □□ □□□□ □□□ □□□□ Systems Manager□ □□□ □ □□□ □□□ □□. m Systems Manager□ □□ □□□ □□□□□ □□□ □ □□□□.
□□□ EC2 □□□□ IoT □□ □ □□□□□ □□□ □□□ □□ □□□□ □□ □ □□ □□□ □□□□ □□ DevOps □□ □□□□ □ □ □□ □□□ □□□□□? (3□□ □□□□□.)
A. □□ EC2 □□□□□ □□□ □□□□□. AWS IoT Greengrass □□□□ □ □□□□□ □□. Systems Manager Session Manager□ □□□□ □□□ □□□ □□ □□□ □□□ □□□□□.
B. Systems Manager Run Command□ □□□□ EC2 □□□□ AWS IoT Greengrass □□ □ □□□□□ □□□ □□ □□ □□□ □□□ □□.
C. Systems Manager □□ □□□□ □□□□ EC2 □□□□ AWS IoT Greengrass □□ □ □□□□□ □□□ Systems Manager □□ □ □ □□ □□□□ □□□□ □□□ □□□□□.
D. □□ □□ □□□□□ □□ Systems Manager Patch Manager□ □□□□□□□ Amazon EventBridge□ □□□□□. Systems Manager Run Command□ □□□□ □□□□ □□ EC2 □□□□ AWS IoT Greengrass □□□□ □ □□□□□ □□□ □□ □□ □□□ □□□□□.
E. Systems Manager□ IAM □□□□ □□□ □□ □□□□ □□□□ AWS □□□ □□ EC2 □□□□□ □□□□□. AWS IoT Greengrass □□□□ □ □□□□□ □□□ □□ Systems Manager□ □□ IAM □□□ □□□ □□□□□.
F. □□□ □□□□ □□□ □□ □□□ □□ □ □□□ ID□ □□□□ □□□□□ □□□ □ □□□ □□□ □□□ □□□□(SSM □□□ □) □□ AWS IoT Greengrass IAM □□ □□ □□ □□□□ □□□ □□□□ SSM □□ □□ IoT □□□ □□□□.
Answer: C,E,F (LEAVE A REPLY)
https://aws.amazon.com/blogs/mt/how-to-centrally-manage-aws-iot-greengrass-devices-using-aws-systems-manager/?force_isolation=true

NEW QUESTION: 160
□ □□□ AWS CodePipeline□ □□□□ □□□□□□□ □□□□ □□□□. □□□ □□□ □□□ □□□ □□ □ □□□□ □□ □□□ □□□□□ □□□□ □□ □□ □□□□□□ □□ □□□ □□□□ □□□. □□ □□□ □□□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. Amazon CloudWatch Logs□ □□□ □□□□□ CodePipeline□ □□□□□.
B. □ □□□□□ □□□ □□ □ Amazon S3 □□□ □□□ □□□ CodePipeline□ □□□□□.
C. Amazon S3□ □□□ □□□□ □□ AWS CloudTrail □□□ □□□□□.
D. □□□ □□ AWS Lambda □□□ □□□□ CodePipeline □□□ □□ □□□ □□□□□. CodePipeline □□□ □□ □□□ □□□□ □□ □□ □□ □□□ □□□ □□□□ □□□ □□□□□.
E. □□ □□ □□ CodePipeline □□ □□ □□□ □□□□□. □□ □□ □□□ □□□□ □□ □□ □□ □□□ □□□ □□□□ □□□ □□□□.
Answer: (SHOW ANSWER)

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.
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 deployed1.
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.

NEW QUESTION: 161
□ □□□ AWS CodeCommit □□□□□□ □□□□ □□ □□□ □□ □□ □□□□ □□□□□. □ □□□ □□□□□□ □□ □□□□ □□□ □□□ □ □□□□ AWS CodeBuild □□□□□ □□□□ AWS CodePipeline □□□□□□ □□□□□□.
□ □□□ CodeBuild □□□□□□ □□ □□□□ □□□□□ □□□□. □□ □□□□ □□□□ CodeBuild □□□□□ □□ □□□ □□ □ □□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□ □□□ CodeBuild □□□□□ □□□ □□□□ □□□?
A. CodeBuild □□□□□ □□□□ □□□□ Git□ □□□□ CodeCommit □□□□ □□□□□. □□□□□ □□□□ □□ □□□□ □□ □□□. □□□□□ □□□□ □□□□ Git□ □□□□ □□□ □□□ □□□ □□ □□□□ □□□□ Git □□□ □□□□ □□□□□.
B. CodeBuild □□□□□ □□□□ □□□□ Git□ □□□□ CodeCommit □□□□□□ □□□□□. □□□□□ □□□□ □□ □□□□ □□□□□. □□□ □□ □□□□□ □□□□ AWS CLI □□□ □□□□ □□□□□□ □ □□□□□ □□□ □□□□□ □□□□□ □□□ □□.
C. CodeCommit □□□□□□□ □□□ □□□□ □□ AWS CLI □□□ □□□□□ CodeBuild □□□□□ □□□□□. □□ □□□□ □ □□□□ □□□□□ □□□□□. □□□ □□ □□□□ □□□□ □□□□□□□ □ Git □□□ □□□□ □□ AWS CLI □□□ □□□□□ □□□□□ □□□□□.
D. CodeBuild □□□□□□□ □□□ □□□□ □□ AWS CLI □□□ □□□□□ CodeBuild □□□□□ □□□□□. □□ □□□□ □ □□□□ □□□□□ □□□□□. □□□ □□ □□□□ □□□□ □□□□□□□ □ □□□□□ □□□ □□□□ □□ AWS CLI □□□ □ □□□□ □□□□□ □□□□□.
Answer: (SHOW ANSWER)
Step 1: Using Native Git in CodeBuildTo 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 CommitOnce 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.

NEW QUESTION: 162
□ □□□ □□□□ □□□□ □□□□ □□ □□ □□ □□□□□□□ □□□□□□ □□□. □ □□□□□□□□ □□ □□ □□□□□ □□ □□□□, □□ □□ Amazon RDS DB □□□□□ □□□□ □□ □□□ □□ □□□ □□□□□. DevOps □□□□□ □□□□□□ □ □□□□□ □□□□ □□□.
□□ □□□□ □ □□ □□□ □□□□□□?
A. RDS DB □□□□□ □□ □□□□□ □□□□ □□ □□□ □□□ □□ □□□□ □□□□□.
□□ □□□ □□□ □□ □□□□ □□□□ □□ □□□□□□ □□ □□□□ □□□□□.
B. □□□□□□□ Amazon DynamoDB □□□ □□□□ □□□□□□□□□□. Amazon Route 53 □□ □□□ □□□□ AWS □□ □ □□ □□ □□□ □□□□□.
C. RDS DB □□□□□ □□ AZ □□□ □□□□□. □□ □□□□□ □□□ □ □□ □□ □□ □□□□□ □□ □□ □□(failover)□ □□ □□□ □□□□□.
D. □□□□□□□ □□ □□ □□□ Amazon EC2 □□□□□ □□□□□□□□□□. Amazon Elastic Block Store(Amazon EBS) □□ □□ □ □□□□ □□ □□□□□ □□ EBS □□□ □□□□□.
Answer: C (LEAVE A REPLY)

NEW QUESTION: 163
DevOps □□□□□ □□□□□□ □□□□ □□□□ □□□□. □□□□□□□ Amazon EC2 □□□□□ □□□□ Amazon Elastic Kubernetes Service(Amazon EKS) □□□□□□ □□□□□ □□□. EC2 □□□□□ Amazon Elastic File System(Amazon EFS) □□ □ □□□ □□□□ □□□□ □□□□ □□□. Amazon EFS □□□□ □□□□ □□□□□(CSI) □□□□□ EKS □□□□□ □□□□ □□ □□.
DevOps □□□□□ □□□□□□□ □□□□ EC2 □□□□□ EFS □□ □□□□ □□□□□ □□□□.
□□ □□□□ □□□ □□□ □ □□□□? (□ □□□ □□□□□.)
A. EKS □□□ Amazon EC2□□ AWS Fargate□ □□□□□.
B. EKS □□□□□□ NFS □□□□ □□□□ □□ EFS □□ □□□□ □□ □□□ □□□□ □□□ □□□□□.
C. Amazon EFS CSI □□□□□ □□ □□□□ □□ □□□ □ □□□ □□ IAM □□□ □□□□□.
D. EFS □□ □□□□ EKS □□ □ □□ □□□ □□□□ □□ AWS DataSync□ □□□□□.
E. EKS □□□ □□□□ □□ EFS □□ □□□□ □□ □□□ □□□ □□□□.
F. EFS □□ □□□□ □□□□ □□□□□□□.
Answer: (SHOW ANSWER)
Mounting EFS to EC2-backed EKS nodes requires:
* NFS (port 2049) open from nodes to EFS (Security Group rule).
* Mount targets in each subnet/AZ where nodes reside.
* IAM role for the EFS CSI driver with elasticfilesystem:ClientMount and ClientRootAccess permissions. These are the standard setup requirements in "Using the Amazon EFS CSI Driver with Amazon EKS."

NEW QUESTION: 164

□□□□ AWS □□ □□□ □□□□ □□ bash □□ □□□□□ □□□□□ □□□. □ □□□ □□ ALB(Application Load Balancer) □□ Amazon EC2 □□□□ □□□ LAMP □□□□□□□ □□□□ □□□□. □□□□ □□ □□ □□□ □□□ □□□□□□□ □□□□□, □□□□ □□ □ □□□□□, □□ □□□□ □□□□□ □□ □□ □ □□□□□, □□ □□□ □□□□□□□. □□□ AWS □□□ □□□ □□ □□□ □□ □□□ □□ □□□ □□□□□ □□□.

□□□ □□ □□□ □□□□ □□□□ □□□□□?

A. AWS CodeBuild□ □□□□ □□□□□□□ □□□□□□. AWS CodeDeploy□ appspec.yml □□□□ □□□ bash □□□□□ □□ □□ □□□□ □□ □□□□ ALB□ □□□□□ □□ □□ □ □□□□□. □□□ □□ □□□□ □□ □□ □□□ □□□□□□□ appspec.yml □□□ □□□□□□.

B. AWS CodePipeline□ □□□□ AWS CodeCommit □□□□□□ AWS CodeDeploy□ □□□□□□□ □□□□□. CodeDeploy□ □ □ □□□ □□□□ □□□□□□□ □□□□□ □□□□□ ALB□ □□ □□ □ □□□□□□. □□□□ □□ □□□□□. □□□ □□ □ □□□ □□ □□ □□□ □□□□□□□ appspec.yml □□□ □□□□□□.

C. AWS CodePipeline□ □□□□ AWS CodeCommit □□□□□□□ AWS CodeDeploy□ □□□□□□ □□ □□□ □□□□□. CodeDeploy□ □□□□ □□□□□□□ □□□□□□. CodeDeploy□ appspec.yml □□□ □□□□ □□□ □□ □□□□ □□ □□□□ □□ □□□□ □□□ □□□□□□□. AWS CodeBuild□ □□□□ ALB□ □□□□□ □□ □□□□ □□ □□□□□□.

D. AWS CodePipeline□ □□□□ AWS CodeBuild□ □□□□□ □□□□□□□ □□□□□□. AWS CodeDeploy□ appspec.yml □□□ □ □□□ bash □□□□□ □□□□ □□□□ □□ □□□□□. ALB□ □□□□ AWS CodeDeploy □□ □□□ □□□□□ □□ □□□ □ □□ □□□□□. □□□ □□ □□□□ □□ □□ □□□ □□□□□□□ appspec.yml □□□ □□□□□□□□.

Answer: (SHOW ANSWER)

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.

NEW QUESTION: 165

□□□ □□□ □□ Amazon Cloud Formation □□□□ □□□□ □□ □□ □□□□□□□ □□□□□. □□□ □□□□ □□ □□□ □ □□□ □□□. □□□ □□□ □□□□ □□□ □ □□ □□□□ □□□□ □□□ □ □□□□ □□ □□□□□ □□□□□ □□□□. □□ □□□ □□□□□ □□ □□□□ □□ □□ □□ □□□ □□□ □□□.

□□□ □□□ □□□□ □□ □□□ □□□ □□ □□□?

A. □□□ □□□ Cloud Formation □□□□ □□□□ AWS Lambda □□□ □□□□□. □□ □□□ □□□□ □□□ □□□□□ □□ □ □□□.

B. Amazon S3□□ Cloud Formation □□□ □□□ Amazon S3 □□□□ □□□□ CloudFormation □□□□ □ Amazon SNS □□□ □ □ □□□□□□.

C. CloudFormation StackSets□ □□□□ □□□□ □□□ □□□□ □□□ □□□□□ □□ □□ □□□□ □□□ □□□□□□.

D. CloudFormation □□ □□ □ □□ □□ □□(□□ □□ Amazon SNS□ □□□□ □□□ □□□□□ □□ □□□□.

Answer: (SHOW ANSWER)

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

NEW QUESTION: 166
□ □□□ AWS□ □□□□ □□□□ □□□□□□. □ □□□□ Amazon API Gateway REST API, □□ AWS Lambda □□, □□□ Amazon EventBridge□ □□□□□.
□□□ CI/CD □□□□□□□ □□□□ □□□ □□□ □□□ □□□ □□□□□ □□□. □□, □□□ □□□□□□□□ □□ □□□ □ □□ □ □□ □□□□ □□□□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. API Gateway REST API□ AWS WAF□ □□□□□□. AWS WAF ACL□ □□□□□. □□□ □□□ □□□ □□ □□□ □□ □□□ □□□□□.
B. Amazon GuardDuty□ □□□□□□. Lambda Protection□ □□□□□□. □□□ □□□ EventBridge□ □□□□□.
C. CI/CD □□□□□□ AWS CloudFormation Guard□ □□□□□. □□□ □□□ □□□□ □□□ □□□□□.
D. AWS □□□□ □□□□ □□□□ □□□□□. □□□ □□□□□. Emerging Threats □□ □□□ □□ □□□ □□□ □□□□□.
Answer: (SHOW ANSWER)
The question focuses on detecting supply chain attacks that bypass CI/CD controls and result in malicious runtime behavior in the deployed serverless application. This requires runtime threat detection , not just static scanning or infrastructure validation.
Amazon GuardDuty is AWS's managed threat detection service. GuardDuty now includes Lambda Protection
, which analyzes Lambda function behavior, including network calls, IAM API usage, and other signals to detect compromised functions, crypto-mining, data exfiltration, and other malicious activity. GuardDuty automatically ingests CloudTrail, VPC Flow Logs (if enabled), and other telemetry without requiring code changes or agents. Findings are published and can be routed to Amazon EventBridge for near real-time notifications or automated remediation.
Option A (AWS WAF) only inspects HTTP requests to API Gateway and is focused on known bad patterns, not deeper behavioral anomalies. Option C (CloudFormation Guard) is focused on infrastructure-as-code validation in the pipeline, which explicitly does not help once malicious code is deployed. Option D (Network Firewall with Emerging Threats rules) protects VPC traffic, but serverless components like Lambda and API Gateway may not all traverse that firewall, and it is not targeted at function-level behavior.
Hence, enabling GuardDuty with Lambda Protection and using EventBridge notifications is the correct solution.

NEW QUESTION: 167
DevOps □□□□□ □□ 3□□ □□□□□□□ □□ □□□ □□□ □□□□ □□□. □□□□□□□ Amazon EC2 □□□□□□ □□□ □ Amazon RDS □□□□□□□ □□□□□. EC2 □□□□□ Application Load Balancer(ALB) □□□ □□□□ Auto Scaling □□□ □□ □□.
DevOps □□□□□ □□ □□□ □□ □□ □□□□ Auto Scaling □□□ □□□□□□. DevOps □□□□□ □□□ □□□ □□ □□ □ □□□ Auto Scaling □□□ □□□□□□. □ Auto Scaling □□□ □□□□ □□□ □□ □□ □□ □□□ □□□□□. □□ □□ □□□ EC2 □□□□□ □□□□ □□□ □□ □□ □□□□□□ □□□□□. □□ □□ □□ □□ □□ □□ □□ □□ □□□□ □□□□ □□□ □ ALB□ □□□ □ □□□□. www example com□ □□ Amazon Route 53 □□□□ ALB□ □□□□□.

□□□ □□ □□□ EC2 □□□□□ □□ □□□□□□ □□ □□ EC2 □□□□□ □□ □□□ □□□□□ □□ □□□□ □ □□ □□ □ □□□ □□□. DevOps □□□□□ □□□ □□ □□□ □□□□ □□ □□□ □□ □□□?

A. □□ □□□ Auto Scaling □□□ □□ □□ □□□□ □□□□ □□ □□□ EC2 □□□□□ □ □□□□□□ □□□□□. □□ □□□ □ □□□□ AWS CLI □□□ □□□□ ALB□ □□□□□□ □□□□ □□□□ □□□□. □□□ □□ □□.

B. AWS CLI □□□ □□□□ ALB□ □□□□□□ □□ □□□ □□ □□□□ □□□□ □□□□. □□ □□ □□ □□□ □□ Auto Scaling □□□ □□ □□□□ □□□□ □□ □□□ EC2 □□□□□ □ □□□□□□ □□□□□.

C. □□ □□□□ □□□□□□ □□ □□□ EC2 □□□□□ □□ □□□ □□□□□□ □□□□□. □ □□□□ □□ □□□ Auto Scaling □□□ □□□□ □□ □□□□□. □□ □□□ EC2 □□□□□ □□□□□ □□ □□□□□.

D. □□ □□□ □□ Auto Scaling □□□ □□ □□□□ □□□□ □□ □□□ EC2 □□□□□ □ □□□□□□ □□ □□ □□□□ □□ □□ ALB□ □□ □□ □□□□□□ □□□□□ Route 53 DNS□ □□□□□□□. .

Answer: (SHOW ANSWER)

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.

NEW QUESTION: 168

□□□ □□□□ □□□□ AWS □□ □□□ □□ □□□ □□□ □□ □□□□□ □□□. □□ □□□ AWS Management Console□□ □ □□□ □□□□□. □□□□ □□ AWS CloudFormation□ □□□□ □□□□ □□□□ □□□□ Amazon VPC □ □□ □□□□ □□ □ □ □□ □□ □□□□□. □□ □□□□ Application Load Balancer, Amazon EC2 Auto Scaling □□, □□ □□ □ Amazon DynamoDB □ □□□ □□ □□ □□□ □□□□.

□□□ □□□□ □□ DevOps □□□□□ □□ □□ □□□ □□□□□□ □□□. □□□□□□□ □□□□ □ □□□ □□□□ □□□ □□□ □□□□ □□□ □□□ □□□□ □□ □□□□□ □ □□ □□□ □□□ □□□. CloudFormation□ □□ □□□ □□□□ □□□ □ □ □□□□□.

□□□ □□ □□□ □□□□ □□□□□ □□□ AWS □□□ □□□□ □□□□ □□ □□□ □□□□□?

A. □□□□ □□□ □□□□ Fn::ImportValue □□ □□□ □□□□ Virtual Private Cloud(VPC) □ □□□ □□ □□□□□. Count □□ □ □ □□□ □□□□ □□□ □□ □□ □□□□ CloudFormation StackSets□ □□ □□□ □□□□□□. UpdateStackSet □□□ □□□□ □□ □□ □□□ □□□□□□□.

B. □□ □□□ □□□□ □□ □□□ □□ □□□ □□□□□. □□□ □□ □□□□□□ TemplateURL□ □□□□ □□□□ □□ □□□ □ □□□□□□. Virtual Private Cloud(VPC) □ □□□ □□ □□□□□ □□ □□□□ □□□□ □□□□ Fn::ImportValue □□ □□□ □ □□□□□. CreateChangeSet □ ExecuteChangeSet □□□ □□□□ □□ □□ □□□ □□□□□□□.

C. □□ □□□ □□□□ □□ □□□ □□ □□□ □□□□□. Fn::ImportValue □□ □□□ □□ □□□ □□□□ □□ □□□□ Virtual Private Cloud(VPC) □ □□□ □□ □□□□□. CreateChangeSet □ ExecuteChangeSet □□□ □□□□ □□ □□ □□□ □□□□□□ □.

D. □□ □□□□ □□□□ □□□□ Fn::ImportValue □□ □□□ □□□□ Virtual Private Cloud(VPC) □ □□□ □□ □□□□□. CloudFormation □□ □□□□ □□□□ □□ □□□□ □□ □□□□ □□□□□. CreateChangeSet□ □□□□□. □ ExecuteChangeSet □□□ □□□□ □□ □□ □□□ □□□□□□□.

Answer: C (LEAVE A REPLY)
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

NEW QUESTION: 169
□□□ AWS Lambda □□□ □□□ □□□□□□□ □□□□. Lambda □□□ AWS CodeCommit □□□□□□ □□□ Python □□□ □ □□□□. □ □□□ □□ Python □□ □□□ □□ □□□□ □□□□ □□□ □□□□□□. □□□□□ □□□□ □□□ □□ □□□ □ □□□ □□ □□□□ □□ Lambda □□□ □□ □□ □□□□ □□□□□□.
□□□ DevOps □□ □□ □□□□ □□ AWS CodePipeline □□□□□□ □□□□ □□ □□□□ □□□□ □□□. □□□□ □□□ □ □ □□ □□ □□□□ □□ □□□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. CodeCommit □□□□□□ Amazon CodeGuru □□□□ □□□□□. □□□ AWS CodeBuild □□□□□ □□□□□. CodePipeline □ □□□□□□ □ CodeBuild □□□□□ □□□□ □□□ □□□ □□□□□. CodeCommit □□□□□□ buildspec.yml □□□ □□□□ □. buildspec.yml □□□□ CodeGuru □□□ □□□□ □□□ □□□□□.
B. □ AWS CodeBuild □□□□□ □□□□□. CodePipeline □□□□□□□ □ CodeBuild □□□□□ □□□□ □□□ □□□ □□□□ □. CodeBuild □□□ □□□ □□□□□. CodeCommit □□□□□□ buildspec.yml □□□ □□□□□. buildspec.yml □□□□ □□ □□ □□□ JUNITXML □□□□ □□ □□□□ □□□□ □□□ □□□□□. □ CodeBuild □□□ □□□ □□□□ □□□ □□□□ □□□□ □.
C. □□□ AWS CodeArtifact □□□□□□ □□□□□. □□□ AWS CodeBuild □□□□□ □□□□□. CodePipeline □□□□□□□ □ CodeBuild □□□□□ □□□□ □□□ □□□ □□□□□. □□ CodeCommit □□□□□□ appspec.yml □□□ □□□□□. appspec.yml □□□ □□ □□ □□□□ CUCUMBERJSON □□□□ □□ □□□□ □□□□ □□□ □□□□□. □ CodeArtifact □□□ □□□ □□□□□ □□□ □□□□ □□□□□.
D. □ AWS CodeBuild □□□□□ □□□□□. CodePipeline □□□□□□□ □ CodeBuild □□□□□ □□□□ □□□ □□□ □□□□ □. □ Amazon S3 □□□ □□□□□. CodeCommit □□□□□□ buildspec.yml □□□ □□□□□. buildspec.yml □□□ □□ □□□□ HTML □□□□ □□ □□□□ □□□□ □□□ □□□□□. □□□ □□□□ □□□ □□□□ S3 □□□ □□□□□□.
Answer: B (LEAVE A REPLY)
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 cost-effective 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.

NEW QUESTION: 170

□□□ Windows □ Linux Amazon EC2 □□□□□□ □□□□□□□ □□□□□. □□□□□ AWS □□□ □□ □□ □□□□ □□□□ □. □□□ □ □□□□□□□ □□ Auto Scaling □□□ □□□□□.

□□□□ □□□□□ □□ □□□ □□ □□□□ □□□□ □□□□□. □□□□ Windows□ SMB□ □□□□ □□ Linux□ NFS□ □□□ □ □□□. □□ □□□□ □□ □□□ □□□ □□□□□ □□□. □□ □□□□□ □□□□ □□ □□□.

□□□ □□ □□□ □□□□ □□ □□□ □□□□□? (3□□ □□□□□.)

A. □□ □□ □□□ □□□ □□ Amazon Elastic File System(Amazon EFS) □□ □□□□ □□□□□.

B. NetApp ONTAP □□ AZ □□ □□□□ Amazon FSx□ □□□□□.

C. □□ □□□□□ □□□ □□ SSD(gp3) Amazon Elastic Block Store(Amazon EBS) □□□ □□□□□.

D. □□ □□□□ □□□□ □□ □ □□□□□□□ □□ □□□□ □□ □□□ □□□□ □□□□□□□.

E. □ Auto Scaling □□□□ □□□□ □□ □□□ □□□□□.

F. □ □□□□□ □□□ □ □□ □□□□ □□□□□ □ □□□□□□□ EC2 □□□□□ □□□□□□□.

Answer: A,B,D (LEAVE A REPLY)

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 sub-millisecond latencies that supports both SMB and NFS protocols, meeting the requirements for both Windows and Linux instances.

References:

Mounting EFS File Systems

Mounting Amazon FSx File Systems

NEW QUESTION: 171
□ □□□ □□□□□□ □□ □□□(ALB) □□ Amazon EC2 □□□□□ □□□ Auto Scaling □□□□ □□□□□□□ □□□□□. EC2 □□□□□ □□□ EC2 □□□□□□ □□□□ MySQL □□□□□□□ □□□ □□□ Docker □□□□□ □□□□□. DevOps □□□□ □ □□□□ □□□□□ □□□□□ □□□□□□□ □□□□□□ □□□. □□ □□ □□□□ □ □□ □□□ □□□□ □□□□ □□□ □□?
A. EC2 □□□□□□ □□□□ □□□□□ AWS Fargate□ □□□□□. MySQL □□□□□□□ MySQL□ □□□□ Amazon Aurora Serverless v2 □□□□□□□ □□□□□.
B. EC2 □□□□□□ □□□□ □□□□□ AWS Fargate□ □□□□□. MySQL □□□□□□□ Amazon DynamoDB □□□□ □□□□ □.
C. EC2 □□□□□ ALB□□ □□□□ □□□□□ AWS Lambda □□□ □□□□□. MySQL □□□□□□□ MySQL□ □□□□ Amazon Aurora Serverless v2 □□□□□□□ □□□□□.
D. EC2 □□□□□ ALB□□ □□□□ □□□□□ AWS Lambda □□□ □□□□□. MySQL □□□□□□□ Amazon DynamoDB □□□ □ □□□□□.
Answer: A (LEAVE A REPLY)

NEW QUESTION: 172
□□□ Amazon S3 □□□□ □□ □ □□□□ □□□□□ □□□□. □□□□□ example.com□□ □□□□ □□□□□. □ □□□ TTL □ 1□□ Amazon Route 53 □□ □□□ □□□ □□□□□. □□□ □□□ □□ □□□□□ □□ □ □□□□□□□□ □□□□□ □□□ □□□. □□ □ □□□□□□□ Amazon EC2 □□□□ □□ □□□ Application Load Balancer(ALB)□ □□□□□.
□□□□ □□□ □□□□ □ □□□ □□□□ 255□□ TTL□ 1□□□ ALB□ □□□□ □□ Route 53 □□ DNS □□□ □□□ □□□□ □. □□ □ DevOps □□□□□ □□□ example.com□ □□□ □ □□□ □□□ □□ □ □□□□ □□□□□ □□□ □□ □□□.
DevOps □□□□□ □□□ example.com□ □□ □□ □□□□ □□□□□ □□□ □□□ □ □□□□?
A. □□ □ □□□ □□□□ □□□ S3 □□□□ □□ □□□ □□□ □□ □□□ □□□□□.
B. ALB□ □□□□□□ □□□ □□□ □□□□ S3 □□□□ □ □□□ □□□□ □□□ □□□□□.
C. S3 □□□ □□□□ □□ DNS □□□ □□□ □□□□□□□. □□□ 0□ □□□□□. □□ □□□ □□ □□□□□ □□□ □□□ □ □□ □□□□□.
D. example.com □□□ □□□□ S3 □□□ □□□□ □□ DNS □□□ □□□ □□□□□. DNS □□□ □□□ □□□ □□□□□□.
Answer: D (LEAVE A REPLY)

NEW QUESTION: 173
□□□□ □□□ □□□ □□ AWS CloudFormation□ □□□□□ □□□. □ □□□ □□□ □□ □ □□□ □□ □□□ □□□ □□□ □ □□□□ □□□□□ □□□. □□□□ □□□ □□□□□□□ □□ □□□ □□□□ □□□.
□□ □□□ □□ □□□□ □□□□□ □□□□ □□□□ □□□□□?
A. AWS Trusted Advisor □□□ □□□□ □□□□ □□ CloudFormation StackSets□ □□ □□□□□.
B. Cloud Formation □□□□ □□ □□□ □□□□ □□□□ □□ CloudFormation StackSets□ □□ □□□□□.
C. □□□ CloudFormation □□□□□ CloudFormation StackSets□ □□□□□.
D. □□□ CloudFormation □□□□ □□□□ AWS Service Catalog □□□ □□□□□.
Answer: (SHOW ANSWER)
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

NEW QUESTION: 174
□ □□□ Java □□ □□□□□□□ CodePipeline□ □□□□□. ECR□□ □□□□ □□□□ □□□□, □□□ □□□□ □□ □□□□ □□ □□ □□□□ □□□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□?
A. Inspector □□□ □□ EventBridge □□ □□□□ □□□ □□□□ □□□□ □□ □□□ □□□□ Lambda□ □□□□□.
B. □□□ □□□, Lambda□ SBOM□ □□ Inspector□ □□□□, S3□ □□□□, Athena□ SBOM□ □□□□, □□□ □□□ □□ □□ □□□ □□□□□.
C. □□□ □□□, EventBridge□ Detective □□ □□□ □□□□, Lambda□ □□□ □□□□ □□ □□ □□□ □□□□□.
D. □□□ □□□, EventBridge□ Inspector □□ □□□ □□□□, Lambda□ □□□ □□□□ □□ □□ □□□ □□□□□.
Answer: D (LEAVE A REPLY)
* Amazon ECR enhanced scanning uses Amazon Inspector for vulnerability detection.
* EventBridge can capture Inspector scan findings.
* Lambda can process scan findings and reject manual approval if critical vulnerabilities exist.
* Options A and C use incorrect or less integrated services (basic scanning or Detective).
* Option B adds unnecessary complexity with SBOM and Athena.
References:
Amazon ECR Image Scanning
Integrating ECR Scanning with CodePipeline

NEW QUESTION: 175
□□□ Docker □□□ □□□□ AWS CodeBuild □□□□□ □□□□ □□□□ □□□□□. □□□□ □□□□ □□□ □ CodeBuild □□ □□□ □□□□ □□□□ Amazon S3 □□□ □□□□□□. CodeBuild □□□□□□ S3 □□□ □□□□ □ □□ □□□ □□ IAM □□ □ □□□ □□□□.
DevOps □□□□□ □□□□ □□□□ □□□□ □□ S3 □□□ Amazon Elastic Container Registry(Amazon ECR) □□□□□□ □□□ □ □□□. DevOps □□□□□ CodeBuild □□□□□ □□□ AWS □□□ ECR □□□□ □□□ □□□□ □□□□□. DevOps □□□□ □ □ ECR □□□ □□□ □□□ □□□□ IAM □□□ □□□ □□□□□. DevOps □□□□□ □□ buildspec.yml □□□ □□□□ docker build □□□ docker push □□□ □ □□□ □□□ □□□□□.
CodeBuild □□□□□ □□ □□□ □□□ □ □□ □□□ ECR □□□□□□ □□□□□□ □□ □□□ □□□□□.
ECR □□□□ □□ □□□ □□ □□□ □□□□ □□□□ □□□□□?
A. □□ □□□ □□ □□ aws ecr get-login-password AWS CLI □□□ □□□□ ECR □□□□□□ □□□□□□ buildspec.yml □□□ □□□□□□□. □□ □□□ □□□□ ECR □□□□ □□□□□□ docker login □□□ □□□□□□□.
B. CodeBuild □□□□□ SECRETS_MANAGER □□□ □□ □□□ □□□□□. □□ □□□ CodeBuild □□□□□ IAM □□□ □□□ ARN□ □□□□□. □□□ □□ □□□ □□□□ docker login □□□□ □□□□□ ECR □□□□ □□□□□□ buildspec.yml □□□ □ □□□□□□.
C. ECR □□□□ □□ □□□ □□□□ □□□□□□□. IAM □□□ □□□ □□□□ □ □□□ □□□□ ECR □□□ □□□ □□□□□.
D. AWS CLI□ □□□□ ECR □□□ □□ IAM □□□ □□□ □□□ buildspec.yml □□□ □□□□□□□. IAM □□□ □□□ □□□□ □ □□□ □□□□ ECR □□□ □□□ □□□□□.
Answer: A (LEAVE A REPLY)

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.

NEW QUESTION: 176
□ □□□ AWS CodeArtifact□ □□□□ Python □□□□ □□□□ □□□□□. CodeArtifact □□□□□□ □□□ □□ □□□□□ □□ □□ □□□□ □□□□.
"□□": "2012-10-17",
"□□": [
{
"□□": [
"codeartifact:DescribePackageVersion", "codeartifact:DescribeRepository",
"codeartifact:GetPackageVersionReadme", "codeartifact:GetRepositoryEndpoint", "codeartifact:
ListPackageVersionAssets", "codeartifact: ListPackageVersionDependency", "codeartifact:
ListPackageVersions", "codeartifact :ListPackages",
"□□□□□□: ReadFromRepository"
],
"□□": "□□",
"□□□": "*",
"□□□":
"□□": {
"StringEquals": {
"aws:PrincipalOrgID": ["o-xxxxxxxxxxx"
]
}
}
}
]
□□□□ AWS Organizations□ □□□ □□ □□□□ □ □□□□□ □□□□ □□□□.
□□□□ □□ □ CodeArtifact □□□□ □□ □□□ Python □□□□□□ □□□□□ □□□. □□□□ AWS CodePipeline□ AWS CodeBuild□ □□□□ □ □□□□□□□ □□□□□. □□□□□□ □□□ □□□□ CodeBuild □□□ □□ □□ □□□□ □□ □□□ □□□ □□ VPC□□ □□□□□ □□□□ □□□□.
□□□□ CodeArtifact□ VPC □□□□□□ □□□□ CodeBuild □□ □□ YAML □□□ □□□□□□□□. □□□ □□□□ □□□□□ Python □□□□□□ □□□□□ □ □□□□.
DevOps □□□□□ □□□□ □□ □□□□□ □□□ □ □□□ □□ □□□□ □□□□ □□□□ □□□? (□ □□□ □□□□□.)
A. Amazon S3 □□□□□ □□□□□□ □□□□ CodeBuild □□□ □□ □□ □□□□ □□□ □□□□ □□□□□□□.
B. □□□□□ □□□ Principal □□ □□□□□□ CodeBuild □□□□□□ □□□□ □□□ ARN□ □□□□□□.
C. AWS Resource Access Manager(AWS RAM)□ □□□□ CodeArtifact □□□□□□ □□□ □□□□□.

D. CodeBuild □□□□□□ □□□□ □□□ □□□□□□ □□ □□□ CodeArtifact □□□□ □□□ □ □□ □□□ □□□ □□□ □□ □.
E. □□□□ CodeArtifact□ □□□ □□□□ □□□□ □□□□□ □□□ □□□□□.
Answer: A,D (LEAVE A REPLY)
" 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.

NEW QUESTION: 177
□□□□ □□□□□ □□□□ □□ Amazon Elastic Container Service(Amazon ECS) □□□□□ □□□□ □□□□. □□ □□□□□ □ □□□□□ □□ ECS □□□□ □□□□□. □ □□□□□ □□□ □□□ Application Load Balancer□ □□□□ □□ □□ □□□ □□□ □ □□□□ □□□□□□.
DevOps □□□□□ □□□□□□ □ □□□ □□□ □□□□ □□□. □□ □□ DevOps □□□□□ □□□□ □□□ □□□ □□ □□□ Amazon S3 □□□□ □□□ □□□.
□□□ □□ □□□ □□□□ □□ DevOps □□□□□ □□□□ □□ □□ □□□ □□□□□? (3□□ □□□□□.)
A. AWS□□ Amazon CloudWatch Logs □□□□ □□□□□ □□□□□□□. □ □□□□□ □□□□ □□□□□□. □□ □□□□ □□ □□□ □□□□□□ □□□ □□□ □□□□□□□.
B. ECS □□□□□ Amazon CloudWatch Logs □□□□□ □□□□□. ECS □□ □□□□ □□ □□□□□ awslogs□ □□□□□.
C. Amazon EventBridge□ □□□□ 60□□□ □□□□ Amazon CloudWatch Logs create-export-task □□□ □□□ AWS Lambda □□ □ □□□□□. □□ □□ □□□ □□ S3 □□□ □□□□□ □□□.
D. ALB□□ □□□ □□□ □□□□□□. □□ □□ ALB□ □□ S3 □□□ □□ □□□□□ □□□.
E. ECS □□□□ □□□□ □□ □□□□ □□□ □□□ □□□□□□. □□ □□ □□□ □□ S3 □□□□ □□ □□□□.
F. □□ S3 □□□ □□□ □□ Amazon Kinesis Data Firehose □□ □□□□ □□□□□. □□ □□ Kinesis Data Firehose□ □□ Amazon CloudWatch Logs □□ □□□ □□□□□.
Answer: B,D,F (LEAVE A REPLY)
https://docs.aws.amazon.com/AmazonECS/latest/developerguide/ecs-logging-monitoring.html

NEW QUESTION: 178
□ □□□ □ AWS □□□□ □ □□□□□□□ □□□□□. □ □□□□□□□□ Amazon Simple Queue Service(Amazon SOS) □□ □ □□□□ □□□□ □□□□ AWS Lambda □□□ □□□□ □□□□. Lambda □□□ □□ □□□□□ □□□ □□ Amazon S3 □□□ □□□ □□□□□. Lambda □□□ □□□□ □□□ □ □□ □□ □□ □□□□ □□□□ □□□. Lambda □□□ □□ □□□ 10□ □□ □□□ □□□ □□□ □□□□ □ □ □□ □□□□.
□□□□□□□ □ □□□ □□ □□□ □□□□ □□ □□□□ □□□□ Lambda □□□ □□□□ □□□ □ □□ □□□□ □ □□ □□ □ □□□□□. □□ □□□□ □□□ □□ □□□□□ □□□□.
□□□ □□□ □□□□ □□ □□ □□□□ □□□□ □□ □□□□ □□□□. □□□ □□□ □□□□ □□ □□□□ □□ □□□□ □□ □□□ □□□□ □□□.
□□ □□□□ □□□ □□ □□□ □□□□□□?
A. □□ □□□ □□ □□□ □□□□. SOS □□ □□ SOS FIFO □□ □□□□□.
AWS □□□□ Lambda □□□ □□□ □□□□□.

B. Lambda □□□ □□ □□□ □□□□. SOS □□□ □□□ □□□□ □□□□. AWS □□□□ Lambda □□□ □□□ □□□□□.
C. Lambda □□□ □□ □□□ □□□□. S3 □□□□ S3 Transfer Acceleration□ □□□□□.
SOS □□ □ □ □□ □□□□ □□□□□.
D. Lambda □□□ □□ □□□ □□□□ □□□□□. □□□ □□ □□□ □□□□□ Lambda □□□ □□□□□. SOS □□ □□ □□ □ □□□ □□□□□.

Answer: D (LEAVE A REPLY)

Step 1: Handling Invalid Data with Failed Batch ItemsThe 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 SizeKeeping 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.

NEW QUESTION: 179

□ □□□□ AWS CDK□ CodePipeline, □□□ CodeBuild□ □□□□ □□□□□□□ □□□□□. □ □□□ □□ □□ □□ □□□□ □ □□□, □□□□ □□□ □□□□ □□□ □□□□□ □□□.
□□ □□□□ □□ □□□□□? (□ □□□ □□□□□.)

A. □□□□ □□□ □□ □□□□□ CodeBuild □□ □□□ □□□□□□ OnFailure□ ABORT□ □□□□□.
B. □□□□ □□□ □ □□□□□ CodeBuild □□□ □□□□□□, cdk deploy□ --rollback true□ □□□□□.
C. □□□□ □□□ □□ □□□□□ CodeBuild □□□ □□□□□□ --require-approval any-change □□□□ □□□□□.
D. template.hasResourceProperties □□□□ □□□□ AWS CDK □□□ □□□ □□□□ □□□□.
E. cdk diff□ □□□□ □□□ □□□ □□□□ □□□□ □□□□ □□□□.

Answer: A,D (LEAVE A REPLY)

* Running unit tests in the build phase and aborting on failure (OnFailure=ABORT) prevents deployment if tests fail.
* AWS CDK assertions module provides programmatic unit tests against synthesized templates (Option D).
* The --rollback flag relates to CloudFormation stack rollback, not test gating.
* The --require-approval flag controls manual approvals, not test outcomes.

* cdk diff checks for changes but is not a unit test and may not catch logical errors.
References:
Testing AWS CDK Applications
CodeBuild Buildspec OnFailure

NEW QUESTION: 180
□ □□□ □□ AWS Organizations□ □ □□□ □ AWS Control Tower □□ □□ □□□□□□. □□ □□ Center to Internet Security(CIS) Benchmarks to AWS Foundations□ □□□□ □□□ □ □□□ □□□.
□□□ □□ □□ AWS Security Hub□ □□□□ □□ □□□ □□ □□□ □□□□□ □□□. □□ □□ □□□ Security Hub □□□ □ □ □□□□. □□ □□ □□□□ □□ □ □□□ □□□□ □□□ □ □ □□□ □□□. □□ □□□ □□□ □□□ □ Security Hub□ □□□ □□ □□□.
□□ □□□□ □□□□ □□□ □□ □□□ □□□□ □□ □□□ □□□□□? (3□□ □□□□□.)
A. □□□ □□□ □□□□ Security Hub□ □□ □□□ □ □□ □□□□ □□□. AWS Control Tower□ □□□□ □ □□ □□ □□ □ □ □ □□□ Security Hub□ □□ □□□ □□□ □□□□ □□□□□. □ □□ □□□□ □□□□□. AWS Foundations □□□ □□ CIS □ □□□□ □□□ Security Hub.
B. □□□ □□□ □□□□ Security Hub□ □□ □□□ □ □□ □□□□ □□□. □□□ □□□□ Security Hub□ AWS Foundations □□ □ □□ CIS □□□□□ □□□□□.
C. □□□ □□□ □□□ AWS IAM □□ □□ □□(AWS Single Sign-On) □□ □□ □□ CreateAccountAssignment API □□□ □□□□ □□ □ □□□□ □□ □□ □ □□□ □□ □□□ □□□□□.
D. □□ □□ □□□ □□ □□□□ Security Hub□ □□□□□ □□ □□□□□ □□□□ SCP□ □□□□□.
E. Security Hub□□ □□ □□□□ □□□.
F. □□□ □□□ □□□□ CreateManagedAccount □□□□ □□□□ Amazon EventBridge □□□ □□□□□. Security Hub CreateMembers API □□□ □□□□ Security Hub□ □ □□□ □□□□ AWS Lambda □□□ □□□□□. Lambda □□□ □□□□□ EventBridge □□□ □□□□□.
Answer: A,C,E (LEAVE A REPLY)
https://docs.aws.amazon.com/securityhub/latest/userguide/accounts-orgs-auto-enable.html

NEW QUESTION: 181
□□□ AWS CodePipeline □□□□□□ □□□□ □□□□ □□ □ Amazon API Gateway API□ □□ □□□□□ □□□□□. □□□□ □□□□□ □□□ □□□ API□□ API□ JavaScript SDK□ □□□□□. □□□□□ □□ □ SDK□ Amazon S3 □□□ □□□ □□□ S3 □□□ □□ □ □□□□□□ □□□□ Amazon CloudFront □□□ □□□ □□ CloudFront □□□ □□□□□□ □□□□ SDK□ □□□ □□□□. DevOps □□□□□ □ API □□ □□ □ SDK□ □□□□ □□□ □ □□□ □□□□ □□□□ □□□.
□□□ □□ □□□ □□□□ □□□□ □□□□□?
A. API □□ □□ □□□ CodePipeline □□□ □□□□□. AWS Lambda □□□ □□□□□ □□□ □□□□□. API Gateway□□ SDK□ □□□□□□ SDK□ S3 □□□ □□□□□ SDK □□□ □□ CloudFront □□□□ □□□□□ Lambda □□□ □□□□□.
B. API□ □□ □□ □□□ CodePipeline □□□ □□□□□. API□ CodePipelme □□□ □□□□□ □□□ □□□□□. SDK□ Amazon S3□ □□□□ □□ □□□□□ Amazon S3□□ CodePipeline □□□ □□□□ SDK □□□ □□ □□□ □□□□□ □ □□ □□□ □□ □□□.

C. aws apigateway□ UpdateStage □□□□ □□□□ Amazon EventBridge □□□ □□□□□. AWS Lambda □□□ □□□□ API Gateway□□ SDK□ □□□□□□□ □□□ □□□□□. SDK□ S3 □□□ □□□□□ CloudFront API□ □□□□ □□□ □□ □□□□ □□□□□. SDK □□.

D. Create□ □□□□ Amazon EventBridge □□□ □□□□□. aws apigateway□ □□ □□□. API□□ SDK□ □□□□□□ □□ AWS Lambda □□□ □□□□□ □□□ □□□□□. Gateway□ SDK□ S3 □□□ □□□□□ S3 API□ □□□□ SDK □□□ □□ □□□ □ □□□□□.

Answer: A (LEAVE A REPLY)

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.

NEW QUESTION: 182

DevOps □□□□□ □□□□ □□□□ Amazon ECR □□□□ □□□□ AWS CodePipeline □□□□□□ □□□□ □□□. □□□□□ □ □□ □□□ □□□ □□□ □□□□ □□, AWS CodeConnections□ □□□ Git □□□□ □□□ Git □□□ □□□ □ □□□□□ □□ □. □□ □□ □□□□□□ □□□ □□□□ □□□ □□□ □□□□ □□□□□ □□□.

□□ □□□□ □□□ □□ □□□ □□□□□□?

A. QUEUED □□□ □□□□ CodePipeline V2 □□ □□□□□□ □□□□□. □□ □□□ □□□□ □□□ □□□ □□□□□ □□□ □ □□□□. □□□□ □□□ □□□ □□□□ EventBridge □□□ □□□□ □□ □□ □□□□□□ □□□□□.

B. SUPERSEDED □□□ □□□□ CodePipeline V2 □□ □□□□□□ □□□□□. □□ □□□□ □□□□ □□□ □□□ □□□□□ □□□ □□□□□. □□□□ □□□ □□□ □□□□ EventBridge □□□ □□□□ □□ □□ □□□□□□ □□□□□.

C. SUPERSEDED □□□ □□□□ CodePipeline V1 □□ □□□□□□ □□□□□. □□ □□□ □□□□ □□□ □□□ □□□□□ □ □□ □□□□□. □□□□□ □□ □□ □□ □□□□□□ □□□□ □□□ □□□□□.

D. QUEUED □□□ □□□□ CodePipeline V1 □□ □□□□□□ □□□□□. □□ □□□□ □□□□ □□□ □□□ □□□□□ □□□ □□□□□. □□□□□ □□ □□ □□ □□□□□□ □□□□ □□□ □□□□□.

Answer: (SHOW ANSWER)

* CodePipeline V2 with QUEUED modeensures that new executions wait for the previous execution to finish, preventing overlapping runs.
* Adding a trigger filter to listen forGit tagstriggers the pipeline only on new tag pushes.
* An EventBridge rule can detect new image pushes to ECR and start the deployment pipeline, integrating the build and deploy pipelines effectively.
* SUPERSEDED mode cancels the previous run when a new one starts, which is not desired here.
* Using branches instead of tags would trigger on all commits, not just releases.

References:

AWS CodePipeline V2 Triggers
AWS CodePipeline Execution Modes

NEW QUESTION: 183
□□□ ALB(Application Load Balancer) □□ □□ Amazon EC2 □□□□□ □ □□□□□□□ □□□□□. □□□ AWS CodeCommit □ □□□□□ □□□□□□ □□□ □□□□□. □□□ □□ □□□□ □□□□ AWS Lambda □□□ AWS CodeBuild □□□□□ □□□□ □. CodeBuild □□□□□ □□□ □□□□□, □□□□ □□□ AWS CodeArtifact□ □□□□, AWS □□□ □□□ Run Command□ □□ □□ □□□□ □□□ EC2 □□□□□ □□□□□.
□□ □□□□□ □□, □□□□ □□□ □□ □□□ □□□□ □□ EC2 □□□□, □□□□ □ □□□□ □□□□□□.
□□ □□□□ □□ □□□□ □□□□□ DevOps □□□□□ □□□□ □□ □□ □□□ □□□□□? (2□□ □□□□□.)
A. AWS CodePipeline□□ CodeCommit □□□□□□ □□ □□□□ □□□□ □□□□□□ □□□□□. □□□□□□□ □□□□ □□ □□□ □□ CodeBuild □□□□□ □□□ □□□□ □□□□□ □□□ □□□□□. □□□□□□□ CodeBuild □□□□ □□ □□□□□ AWS CodeDeploy □□□ □□□□□.
B. AWS CodePipeline□□ CodeCommit □□□□□□ □□ □□□□ □□□□ □□□□□□ □□□□□. CodeBuild □□□□□ □□□□ □□□□□□□ □□□□ □□□□□ □□□ □□□□□ □□□ □□□□□. □□□□□□□ CodeBuild □□□□ □□ □□□□□ AWS CodeDeploy □□□ □□□□□.
C. AWS CodeDeploy □□□□□□□ □□ □□□ □□□□ □□□□ □□□ EC2 □□□□□ □□□□□. □□ □□□ □□ ALB□ □□ □□□.
D. Systems Manager □□ AWS CodeDeploy□ □□□□ □□, □□□ □ □□ □□□ □□□□ □□ Lambda □□□ □□□□□.
E. Amazon S3 □□□ □□□□□. CodeArtifact □□ S3 □□□ □□□□ □□□□□ CodeBuild □□□□□ □□□□□. CodeDeploy□ □□ □□□ □□□□ □□□□□ EC2 □□□□□ □□□□□.
Answer: A,C (LEAVE A REPLY)
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 cost-effective package management service that can store and share software packages for application development67

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