DOP-C02 Test Questions Fee | DOP-C02 Materials

P.S. Free 2025 Amazon DOP-C02 dumps are available on Google Drive shared by NewPassLeader: https://drive.google.com/open?id=1vjVUtI6Eu3gXj5fUTEXD0SqT-McFPys7

At this moment, our company has been regarded as the best retailer of the DOP-C02 study materials. We are responsible for every customer. Your satisfactions on our DOP-C02 exam braindumps are our great motivation. In addition, all people have the right to enjoy our good pre-sale and after sale service on our DOP-C02 training guide. We warmly welcome every customer to select our DOP-C02 learning questions.

Amazon DOP-C02 Certification Exam is intended for experienced DevOps engineers, as well as other IT professionals who work in a DevOps environment. DOP-C02 exam is designed to be challenging, and candidates are advised to have at least two years of hands-on experience working in a DevOps role before attempting the certification. Candidates who pass the exam will be certified as AWS Certified DevOps Engineers - Professionals, and will have the skills and knowledge necessary to design, manage, and maintain DevOps systems on the AWS platform.

>> DOP-C02 Test Questions Fee <<

DOP-C02 Materials, Exam Questions DOP-C02 Vce

Our DOP-C02 study practice materials have so many advantages that basically meet all the requirements of the user. If you have a good comments or suggestions during the trial period, you can also give us feedback in a timely manner. Our DOP-C02 study materials will give you a benefit, we do it all for the benefits of the user. Our pass rate for DOP-C02 Training Material is as high as 99% to 100%, which is proved from our loayl customers, and you will be the next to benefit from it. Our DOP-C02 practice files look forward to your joining in.

The DOP-C02 Exam covers a broad range of topics related to DevOps, including continuous integration and delivery, infrastructure as code, monitoring and logging, security and compliance, and automation and optimization of AWS services. To pass the exam, candidates must demonstrate their ability to design and implement scalable, reliable, and secure DevOps solutions using AWS technologies and best practices. AWS Certified DevOps Engineer - Professional certification is highly valued by employers and can help DevOps professionals advance their careers and increase their earning potential.

Amazon AWS Certified DevOps Engineer - Professional Sample Questions (Q114-Q119):

NEW QUESTION # 114
The company's DevOps engineer discovered that the errors were caused by RDS connection limits being reached. The DevOps engineer also discovered that more than 90% of the API requests are GET requests that read from the DB cluster.
How should the DevOps engineer solve this problem with the LEAST development effort?

- A. Migrate from Amazon RDS to Amazon DynamoDB. Add an Amazon CloudFront distribution in front of the API Gateway REST API.
- B. Add a proxy from Amazon RDS Proxy in front of the RDS DB cluster. Enable API caching in API Gateway.
- C. Migrate from Amazon RDS to Amazon DynamoDB. Enable API caching in API Gateway.
- D. Add an Amazon RDS Proxy in front of the RDS database cluster. Provision an Amazon ElastiCache (Redis OSS) cluster.

Answer: B

Explanation:
The root cause of the problem is that the RDS PostgreSQL instance is exceeding its database connection limit due to high volumes of concurrent GET requests. The most efficient and lowest-effort remediation is to reduce the number of connections that reach the database and serve as many reads as possible from a cache layer. Option B accomplishes both goals with minimal changes to the existing architecture.
First, RDS Proxy manages database connections efficiently by pooling and reusing them. This reduces connection churn and prevents the Lambda function from opening excessive concurrent connections during traffic spikes. RDS Proxy is natively integrated with RDS and requires only minor configuration changes without altering application logic.
Second, enabling API Gateway caching allows caching GET responses directly at the API layer. Because over 90% of requests are GET operations, enabling caching drastically reduces traffic hitting both Lambda and RDS, significantly improving performance and lowering connection pressure.
Option C also improves read scaling, but adding ElastiCache introduces additional infrastructure and requires modifying application code to query Redis before querying the database. This is more complex than enabling API Gateway caching. Options A and D require full database migration, which is far more labor-intensive.
Option B is the simplest, most effective, and most aligned with AWS best practices for reducing RDS connection saturation.

NEW QUESTION # 115
A company is adopting AWS CodeDeploy to automate its application deployments for a Java-Apache Tomcat application with an Apache Webserver. The development team started with a proof of concept, created a deployment group for a developer environment, and performed functional tests within the application. After completion, the team will create additional deployment groups for staging and production.
The current log level is configured within the Apache settings, but the team wants to change this configuration dynamically when the deployment occurs, so that they can set different log level configurations depending on the deployment group without having a different application revision for each group.
How can these requirements be met with the LEAST management overhead and without requiring different script versions for each deployment group?

- A. Tag the Amazon EC2 instances depending on the deployment group. Then place a script into the application revision that calls the metadata service and the EC2 API to identify which deployment group the instance is part of. Use this information to configure the log level settings. Reference the script as part of the AfterInstall lifecycle hook in the appspec.yml file.
- B. Create a script that uses the CodeDeploy environment variable DEPLOYMENT_GROUP_ NAME to identify which deployment group the instance is part of. Use this information to configure the log level settings. Reference this script as part of the BeforeInstall lifecycle hook in the appspec.yml file.
- C. Create a CodeDeploy custom environment variable for each environment. Then place a script into the application revision that checks this environment variable to identify which deployment group the instance is part of. Use this information to configure the log level settings. Reference this script as part of the ValidateService lifecycle hook in the appspec.yml file.
- D. Create a script that uses the CodeDeploy environment variable DEPLOYMENT_GROUP_ID to identify which

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

Answer: B

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

NEW QUESTION # 116
A company has chosen AWS to host a new application. The company needs to implement a multi-account strategy. A DevOps engineer creates a new AWS account and an organization in AWS Organizations. The DevOps engineer also creates the OU structure for the organization and sets up a landing zone by using AWS Control Tower.
The DevOps engineer must implement a solution that automatically deploys resources for new accounts that users create through AWS Control Tower Account Factory. When a user creates a new account, the solution must apply AWS CloudFormation templates and SCPs that are customized for the OU or the account to automatically deploy all the resources that are attached to the account. All the OUs are enrolled in AWS Control Tower.
Which solution will meet these requirements in the MOST automated way?

- A. Deploy CloudFormation stack sets by using the required templates. Enable automatic deployment.
  Deploy stack instances to the required accounts. Deploy a CloudFormation stack set to the organization's management account to deploy SCPs.
- B. Create an Amazon EventBridge rule to detect the CreateManagedAccount event. Configure AWS Service Catalog as the target to deploy resources to any new accounts. Deploy SCPs by using the AWS CLI and JSON documents.
- C. Deploy the Customizations for AWS Control Tower (CfCT) solution. Use an AWS CodeCommit repository as the source. In the repository, create a custom package that includes the CloudFormation templates and the SCP JSON documents.
- D. Use AWS Service Catalog with AWS Control Tower. Create portfolios and products in AWS Service Catalog. Grant granular permissions to provision these resources. Deploy SCPs by using the AWS CLI and JSON documents.

Answer: C

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

NEW QUESTION # 117
A DevOps engineer is setting up an Amazon Elastic Container Service (Amazon ECS) blue/green deployment for an application by using AWS CodeDeploy and AWS CloudFormation. During the deployment window, the application must be highly available and

CodeDeploy must shift 10% of traffic to a new version of the application every minute until all traffic is shifted.
Which configuration should the DevOps engineer add in the CloudFormation template to meet these requirements?

- A. Add the AWS::CodeDeployBlueGreen transform and the AWS::CodeDeploy::BlueGreen hook parameter with the CodeDeployDefault.ECSLinear10PercentEvery1 Minutes deployment configuration.
- B. Add an AppSpec file with the CodeDeployDefault.ECSLineaMOPercentEverylMinutes deployment configuration.
- C. Add the AWS::CodeDeployBlueGroen transform and the AWS::CodeDeploy::BlueGreen hook parameter with the ECSCanary10Percent5Minutes deployment configuration.
- D. Add an AppSpec file with the ECSCanary10Percent5Minutes deployment configuration.

Answer: A

Explanation:
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:
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 # 118
A DevOps engineer is building an application that uses an AWS Lambda function to query an Amazon Aurora MySQL DB cluster. The Lambda function performs only read queries. Amazon EventBridge events invoke the Lambda function.
As more events invoke the Lambda function each second, the database's latency increases and the database's throughput decreases. The DevOps engineer needs to improve the performance of the application.
Which combination of steps will meet these requirements? (Select THREE.)

- A. Implement the database connection opening and closing inside the Lambda event handler code.
- B. Use Amazon RDS Proxy to create a proxy. Connect the proxy to the Aurora cluster reader endpoint. Set a maximum connections percentage on the proxy.
- C. Implement database connection pooling inside the Lambda code. Set a maximum number of connections on the database connection pool.
- D. Implement the database connection opening outside the Lambda event handler code.
- E. Connect to the Aurora cluster endpoint from the Lambda function.
- F. Connect to the proxy endpoint from the Lambda function.

Answer: B,D,F

Explanation:
Explanation
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.
References:
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 # 119
......

DOP-C02 Materials: https://www.newpassleader.com/Amazon/DOP-C02-exam-preparation-materials.html