acloudGuru Sim-3

Thursday, June 04, 2020 10:13 PM

QUESTION 5

A transport company uses a mobile GPS application to track the location of each of their 60 vehicles. The application records each vehicle's location to a DynamoDB table every 6 seconds. Each transmission is just under 1KB and throughput is spread evenly within that minute. How many units of write capacity should you specify for this table?

- ✖ **60** SELECTED
- ✔ 10
- ✖ 600
- ✖ 100

EXPLANATION:

Writing to the database every six seconds, there are 10 writes/minute/vehicle. There are sixty vehicles in the fleet, so there are 600 writes/minute overall. 600/60 seconds = 10 writes/second.

RESOURCES

- **Throughput Capacity for Reads and Writes**

STATS

- You spent **02:51** on this question

QUESTION 25

You are developing an application in API Gateway, and need to categorize your APIs based on their status as: sandbox, test, or prod. You want to use a name-value pair system to label and manage your APIs. What feature of API Gateway would you use to accomplish this task?

- ✔ Use stage variables based on the API deployment stage to interact with different backend endpoints.
- ✖ Use the API Gateway console to create a canary release deployment.
- ✖ Use environment variables based on the API deployment stage to interact with different backend endpoints.
- ✖ **Use tags based on stages. The tag can be set directly on the stage of the API.** SELECTED

EXPLANATION:

Stage variables are name-value pairs that you can define as configuration attributes associated with a deployment stage of a REST API. They act like environment variables and can be used in your API setup and mapping templates. With deployment stages in API Gateway, you can manage multiple release stages for each API, such as: alpha, beta, and production. Using stage variables you can configure an API deployment stage to interact with different backend endpoints. Environment variables apply to AWS Lambda. Canary release is a software development strategy in which a new version of an API (as well as other software) is deployed as a canary release for testing purposes, and the base version remains deployed as a production release for normal operations on the same stage. (This would be appropriate when your application is live and you'd want to reduce the risk inherent in a new software version release.) A tag is a metadata label that you assign or that AWS assigns to an AWS resource and would not impact the functionality of your APIs.

RESOURCES

- **Set up Stage Variables for a REST API Deployment**

STATS

- You spent **01:09** on this question
- You flagged this question

QUESTION 31

Where should the appspec.yml be stored?

- ✖ In the config directory in your application source directory
- ✖ In /opt
- ✔ In the root of your application source directory
- ✖ **In the .ebextentions folder** SELECTED

EXPLANATION:

The AppSpec file (appspec.yml) must always be in the root or your application source directory otherwise the deployment will not work. The .ebextensions folder is used to set custom environment variables in Elastic Beanstalk, not CodeDeploy.

RESOURCES

- **CodeDeploy AppSpec File**

STATS

- You spent **00:53** on this question
- You flagged this question

QUESTION 57

An organization wishes to use CodeDeploy to automate its application deployments. The organization has asked a developer to advise on which of their services can integrate with CodeDeploy. Which of the following services can the developer advise are compatible with CodeDeploy managed deployments?

- ✔ Fargate
- ✔ **Lambda** SELECTED
- ✔ On-premises servers
- ✔ **EC2** SELECTED
- ✖ **Elastic Kubernetes Service pods** SELECTED
- ✖ **S3 Static website hosting** SELECTED

EXPLANATION:

CodeDeploy supports EC2, ECS (both EC2 and Fargate), Lambda, and on-premise servers.

RESOURCES

CodeDeploy - supported compute platforms

STATS

- You spent **01:04** on this question

QUESTION 62

You are developing a serverless retail application which includes a mobile app. All your product data is stored in DynamoDB, whilst the application itself runs on Lambda. The product catalogue is updated once every 6 months, to reflect seasonal stock and price updates. Each database read is 3KB in size and the application performs around 20 reads per second. Which of the following DynamoDB settings would you recommend?

- ✖ Configure the table to use high performance reads
- ✖ Use strongly consistent reads
- ✖ **Configure the table with 20 read capacity units** SELECTED
- ✖ Configure your application to use a query rather than a scan
- ✔ Configure the table with 10 read capacity units
- ✔ **Use eventually consistent reads** SELECTED

EXPLANATION:

A read capacity unit represents one strongly consistent read per second, or two eventually consistent reads per second, for an item up to 4 KB in size. Eventually consistent reads provide greater throughput than strongly consistent. In this case the data changes infrequently, so eventually consistent is a good option.

RESOURCES

DynamoDB Provisioned throughput

STATS

- You spent **01:17** on this question

QUESTION 63

You are migrating a restaurant booking application from your own data center to AWS. The application currently runs on a number of virtual machines running web and application servers as well as a shared database server. The applications need to access a large number of shared images and documents containing drinks and food menus. Which of the following could you use as a shared storage solution for this application so that the application servers can still access the shared files?

- ✖ Store the files in DynamoDB
- ✖ **Generate a pre-signed URL to grant access** SELECTED
- ✖ Embed IAM credentials in the EC2 instance metadata
- ✖ Store the files in SQS
- ✔ Use an IAM instance role to grant access
- ✔ **Store the files in S3** SELECTED
- ✖ Store the files in ElastiCache

EXPLANATION:

ElastiCache is a temporary in memory data store, and is not for persisting shared files. DynamoDB is a noSQL database and not a suitable place to store images and text documents. SQS is a messaging system and not a data store. S3 is a storage solution suitable for images, documents and other files or objects which can be accessed by multiple users and services. The recommended way to enable EC2 instances to access S3 is by using an Instance Role.

RESOURCES

Using an IAM Role to Grant Permissions to Applications Running on Amazon EC2 Instances

STATS

- You spent **01:54** on this question

QUESTION 52

You can use X-Ray with applications running on which platforms?

- ✔ Elastic Beanstalk
- ✔ **EC2** SELECTED
- ✔ **ECS** SELECTED
- ✖ **S3** SELECTED
- ✔ **Lambda** SELECTED

EXPLANATION:

X-Ray works with Lambda, EC2, API Gateway, Elastic Beanstalk and ECS

RESOURCES

X-Ray FAQs

STATS

- You spent **00:25** on this question

QUESTION 58

You want to quickly deploy and manage an application in the AWS Cloud without having to learn about the infrastructure that runs the application. Elastic Beanstalk is the first service that comes to mind. You have written you application with C#. How would you launch your application on Elastic Beanstalk in the most efficient manner?

- ✖ **Use EC2 instance. Launch the application on an EC2 instance and use CloudFormation to automate infrastructure provisioning.** SELECTED
- ✖ Rewrite your application in Python as C# is not a supported programming language.
- ✔ Create your own Elastic Beanstalk platform using Packer. Use this platform for your application.
- ✖ Use AWS OpWorks instead to launch your application that will help automate deployment and configurations for your application.

EXPLANATION:

AWS Elastic Beanstalk supports custom platforms which lets you develop an entire new platform from scratch, customizing the operating system, additional software, and scripts that Elastic Beanstalk runs on platform instances. This flexibility enables you to build a platform for an application that uses a language or other infrastructure software, for which Elastic Beanstalk doesn't provide a managed platform. In addition, with custom platforms you use an automated, scripted way to create and maintain your customization, whereas with custom images you make the changes manually over a running instance. Rewriting your application would not be the most efficient way if you can create your own platform. Launching an EC2 instance would still require you to manage your own infrastructure. OpsWorks manages infrastructure deployment by organizing applications into layers to provision EC2 instances and resources for an application.

RESOURCES

- Elastic Beanstalk Custom Platforms

STATS

- You spent **01:52** on this question
- You flagged this question

QUESTION 61

You are developing a gaming website which scores all players scores in a DynamoDB table. You are using a Partition key of user_ID and a Sort Key of game_ID as well as storing the user_score which is the user's highest score for the game and also a timestamp. You need to find a way get the top scorers for each game, who have scored over 50,000 points. Which of the following will allow to to find this information in the most efficient way?

- ✖ **Use a global secondary index with a partition key of game_ID and a sort key of user_ID** SELECTED
- ✔ Use a local secondary index with a partition key of user_ID and a sort key of user_score
- ✖ Query the table using a partition key of user_ID and sort by game_ID
- ✖ Scan the table and order by score

EXPLANATION:

A scan operation would be less efficient than a query, so that is definitely not the most efficient way. The Query operation described won't help you find the top scorers for each game. A local secondary index is an index that has the same partition key as the base table, but a different sort key. A global secondary index is an index with a partition key and a sort key that can be different from those on the base table.

RESOURCES

- Queries and Scans
- DynamoDB Indexes

STATS

- You spent **02:03** on this question