Assessment 1

Tshwetso K Mokgatlhe

The Open Window Institute

DV_T200

Lecturer: Tsungai Katsuro

ASSESSMENT 1

Lets explore the differences between server-based (EC2) and serverless (Lambda) compute models across various dimensions.

1. Provisioning and Management:

EC2: In the EC2 model, you provision virtual machines (instances) and are responsible for configuring and managing the OS, applications, and security updates. You have more control over the environment but also need to handle scaling and maintenance.

Lambda: With Lambda, you don't manage servers directly. AWS handles the server provisioning, deployment, and scaling. You upload your code as a function and AWS triggers and executes it in response to events.

2. Resource Allocation and Scaling:

EC2: You need to manually configure and scale instances based on anticipated load. Over-provisioning can lead to wasted resources, while under-provisioning can result in performance issues during spikes.

Lambda: Automatically scales up or down based on the incoming event rate. You're billed based on the actual execution time of your functions.

3. Billing and Cost Management:

EC2: You're billed for the running instances, regardless of whether they are fully utilized or not. Pricing can be more predictable but may lead to higher costs if not managed properly.

Lambda: You're billed based on the number of invocations and the execution time of your functions. This can lead to cost savings since you only pay for actual usage.

4. Function Design and State Management:

EC2: Allows for more complex applications and stateful workloads as you have full control over the environment. You can store data locally or in external databases.

Lambda: Encourages stateless functions that are triggered by events. State can be managed using external storage services like DynamoDB, S3, or RDS.

5. Event-Driven Architecture:

EC2: Requires you to set up event handling and management systems manually.

Lambda: Inherently follows an event-driven model. It's particularly well-suited for microservices architectures and responding to events in real-time.

References

https://aws.amazon.com/getting-started/

For additional information on APA Style formatting, please consult the APA Style Manual, 7th Edition.