|  |  |  |
| --- | --- | --- |
| Logging & Monitoring | * CloudWatch - Alarms, Logs, Metrics. * CloudWatch Events * API logs on CloudTrail * AWS Config | **Section  24 - (272 - 284)** |
| Serverless Lambda | * Lambda Overview * DynamoDB | **Section 19 - ( 214 - 226 )**  **Section 20 - ( 231 - 232, 234 )** |

1. Your company is launching a new web application and requires a highly available and scalable infrastructure on AWS. The goal is to automatically handle traffic spikes by scaling EC2 instances up or down, distribute incoming traffic evenly across instances, and monitor the system's performance. Additionally, the system should notify the operations team via email when instances are added or removed based on CPU utilisation. Requirements:

* Set up an EC2 instance running a web application that serves a "Hello World" page.
* Set up an Auto Scaling Group (ASG) with a minimum of 1 and a maximum of 3 instances to manage EC2 instances for web traffic.
* Create an Elastic Load Balancer (ELB) to evenly distribute incoming traffic.
* Configure CloudWatch Alarms to monitor CPU utilisation for scaling events.
  + Scale out when CPU utilisation exceeds 70% for 5 minutes.
  + Scale in when CPU utilization drops below 30% for 5 minutes.
* Ensure the infrastructure automatically scales to handle varying traffic loads.

1. Your company is developing a custom web application that requires hosting on AWS using EC2. To ensure effective monitoring and troubleshooting, the application needs to push its logs to CloudWatch. Additionally, you want to capture and monitor custom metrics from the application to gain insights into its performance.Your task is to create an EC2 instance running this custom application, configure it to push its logs to CloudWatch, and set up custom metrics for monitoring.

**Requirements**:

* Set up an EC2 instance running a custom web application.
* Install and configure the CloudWatch Agent to push application logs to CloudWatch.
* Modify the application to send custom metrics (e.g., Memory utilisation and errors) to CloudWatch.
* Set up CloudWatch Alarms to monitor custom metrics and trigger alerts for any issues or if it exceeds threshold.
* Optional (Implement SNS notifications to alert the team when alarms are triggered)
* Validate that logs and metrics are correctly displayed in CloudWatch.
* Make sure CloudWatch logs are encrypted & set to auto expire after 7 Days.

1. You're a cloud architect for a company that processes large volumes of text-based documents. The company uses Amazon S3 as its primary storage solution. Your task is to implement an automated system using S3 event-based triggers, Lambda functions, and DynamoDB to process and manage these documents efficiently.

**Requirements:**

1. **Text Processing:**
   * When a new .txt file is uploaded to the raw-documents/ prefix:
     + Trigger a Lambda function to count the words in the document.
     + Store the word count and metadata (file name, upload timestamp) in a DynamoDB table called DocumentMetadata.
     + Save a processed version(with the metadata added in the file at the top) of the file to the processed/ prefix.
2. **Archiving**:
   * When an object in the active-content/ prefix is tagged with archive, trigger a Lambda function to:
     + Move the object to the archived/ prefix.
     + Log the operation in DynamoDB, including the timestamp and user (passed via metadata).
3. **Deletion Notification:**
   * When an object is deleted, trigger a Lambda function to:
     + Send a notification to an SNS topic with details like file name, prefix, deletion time, and user (if available).
     + Log the deletion event in the DynamoDB table.