Project - Echo Request Application on AWS

Table of Contents:

Executive Overview

The purpose of this project is to prepare a hardened, logging configured & highly available server that hosts a simple application to echo back sent requests

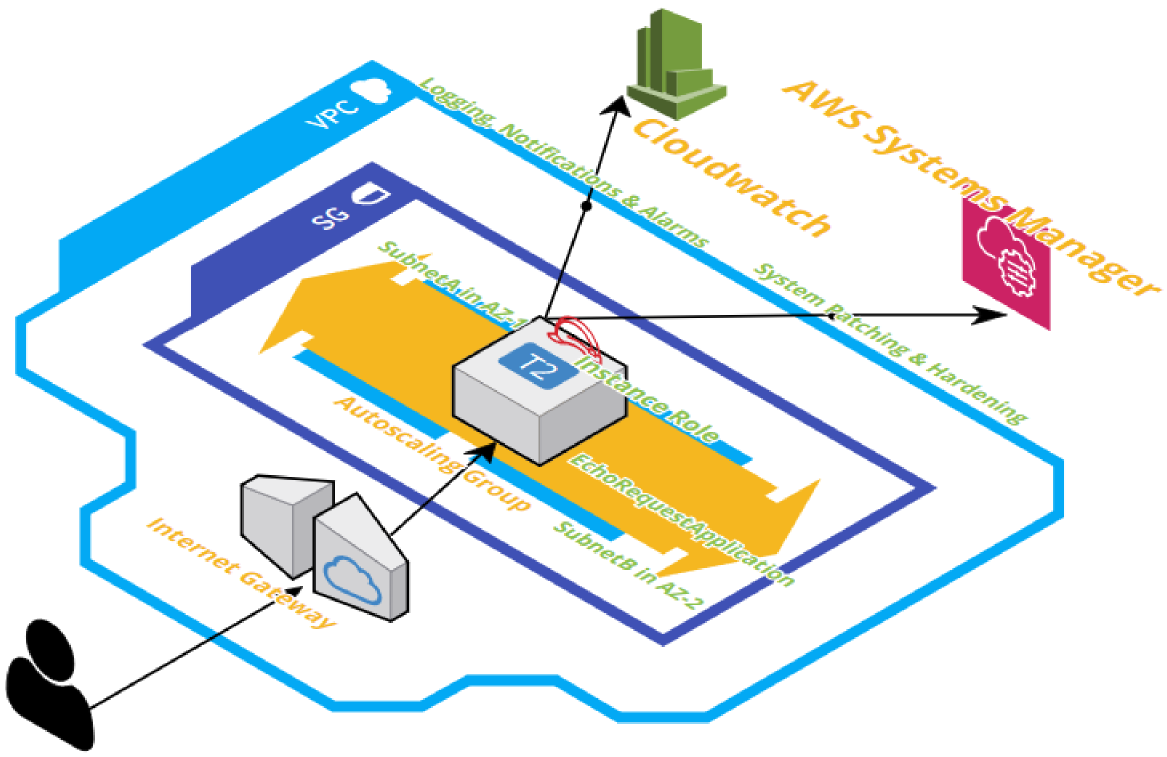
Project Scope:

Task Name	Task Deliverable
High Level Architecture	Diagram, details and
Basic Infrastructure & VM Setup	Working CloudFormation template that deploys the basic infrastructure required to setup the instance
Application Setup	Application Code and setup options
Logging Setup	Script to setup logging mechanism on the system
Hardening	Script that hardens the OS to be resistant to general cyber attacks
Final Output	CloudFormation Script Implementation Plan & Documentation

High Level Architecture:

The diagram and details can be found below:

- **Diagram:**



- **Details:**

  Basic architecture consists of:

  - 1 Region
  - 1 VPC
  - 2 Availability Zones
  - 2 Public Subnets
  - 1 Internet Gateway
  - 1 AutoScaling Group

  The single autoscaling group spans across 2 Availability Zones (AZ), 1 subnet in each availability zone, so the loss of 1 AZ/datacenter does not affect the **availability** of the solution and the application remains up.
  Logging will be configured for all critical logs and logs will be stored in CloudWatch.
  OS shall also be **hardened** to offer resistance to cyber threats.

Pros:

  - Application is highly available and resistant to failure of 1 availability zone
  - High & low traffic load cause the autoscaling group to scale accordingly and thus keep the application available at all times
  - Hardened underlying OS so that it is resistant to cyber attacks

- Critical logging configuration on the application server

Cons:

- Automatic traffic redirection not available since no loadbalancer has been set up. Therefore, one has to know the IP/DNS address of the instance before sending traffic to it. Please note that the IP and DNS are always new for every new instance. IP & DNS from the previous instance can't be shared with a new EC2 instance.

Application Setup:

A Python based Flask application has been setup on the instance that is responsible for echoing back the request sent by a user.
Application has been setup to automatically restart upon system reboot.

Logging:

CloudWatch agent to monitor system stats has been setup on the instance.
Collection of following metrics & logs has been setup:

- **Metrics:**
  - CPU
  - Memory
  - Disk
- **Logs:**
  - SSH
  - Syslog
  - Application logs
    - Access
    - Error
  - CloudWatch agent
  - IP ban logs

Metrics and logs, both can be viewed directly from the CloudWatch logs page in the specific region.

Hardening:

Ubuntu 18.04 OS has been hardened as follows:

- Encryption has been setup on the underlying volume
- Custom admin user has been created
- Custom restricted user has also been created
- IP banning system has been implemented to thwart relevant cyber attacks on the system via SSH and Application Ports
- System packages also get upgraded every month

High Availability:

An autoscaling group has been created which scales instance count based on the system CPU utilization metric.

Testing:

Testing for this script was conducted in Ireland region, ‘eu-west-1’ with t2.micro type instances.
Infrastructure got deployed as explained.
Application was up and running as expected.
Logging and hardening was also in-place.

Machine took about 10-15 minutes to setup with all the required scheme (logging, hardening etc.)