Cloud Infrastructure Automation – Web Server Deployment and Management

Task Overview

Objective: Deploy a static website on a cloud setup and configure supporting infrastructure for maintainability and automation.

1. Launch and Configure Web Server

Goal: Deploy a static website on a cloud setup.

Steps:

- Launch an EC2 instance using Ubuntu.

- Install Apache2 web server.

- Deploy the static website using FTP.

- Open the website in a browser using the public IP address.

- Set up a Bastion Host to securely manage EC2 instances.

- Route all administrative (SSH) access through a single, controlled, and monitored entry point.

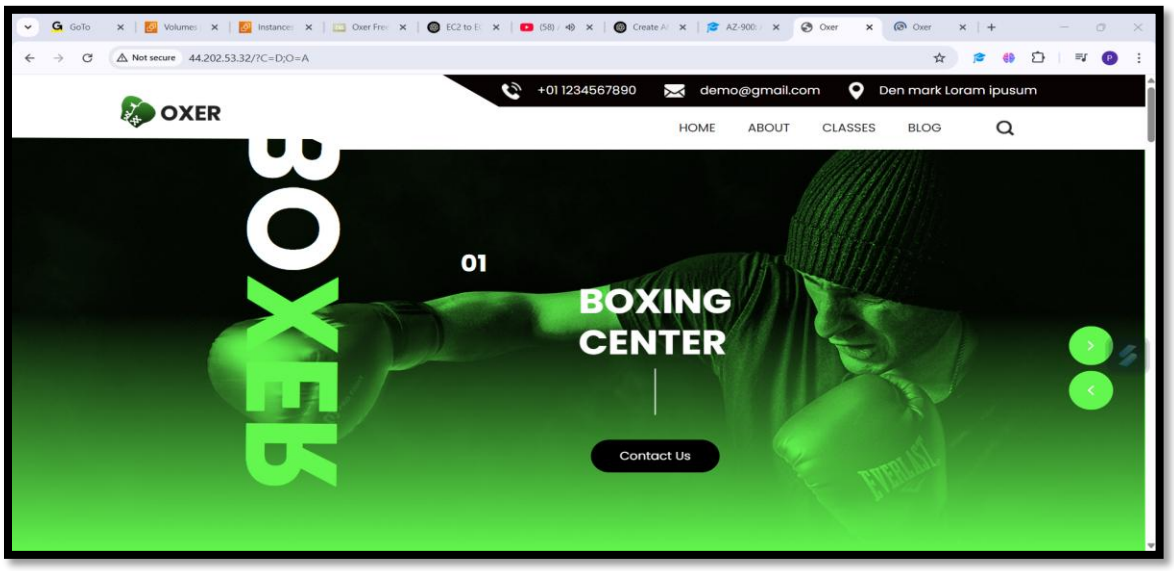


Fig. 1: Launch EC2 Instance

This image shows the configuration page for launching an EC2 instance using Ubuntu.

2. Create a Custom AMI

Goal: Save the current EC2 setup as a reusable Amazon Machine Image (AMI).

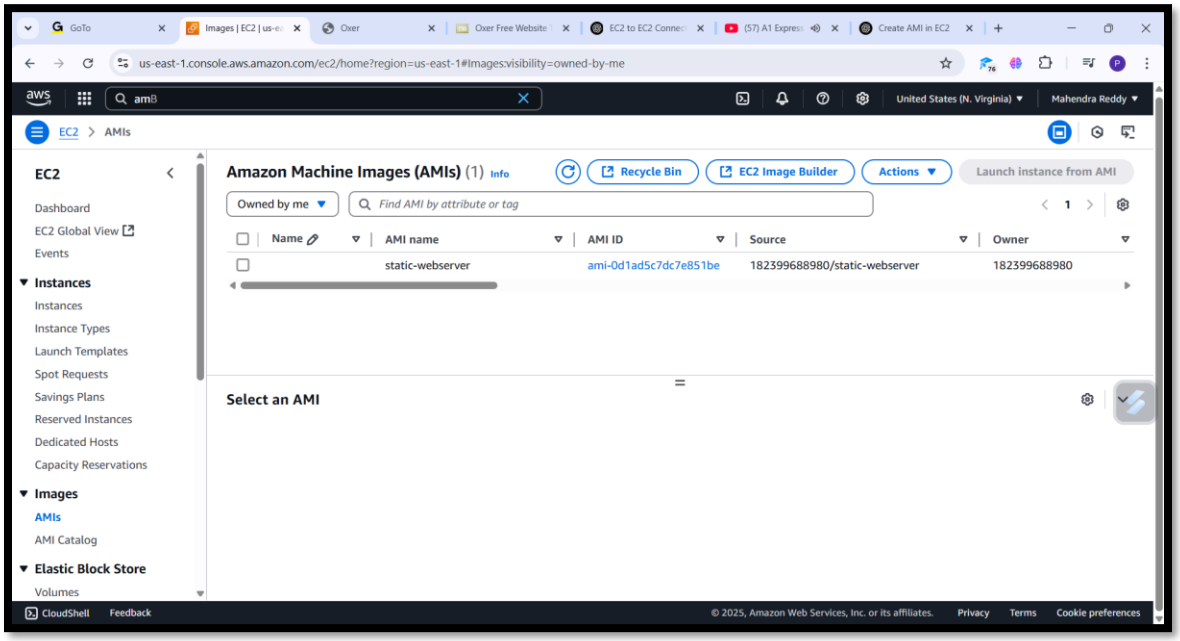


Fig. 2: FTP Deployment

Deploying the static website files to the EC2 instance using an FTP client.

3. Automate Regular Backups

Goal: Create automated EBS volume snapshots using AWS Lifecycle Manager.

Steps:

- Create a snapshot policy to take backups every 24 hours.

- Retain snapshots for 2 days.

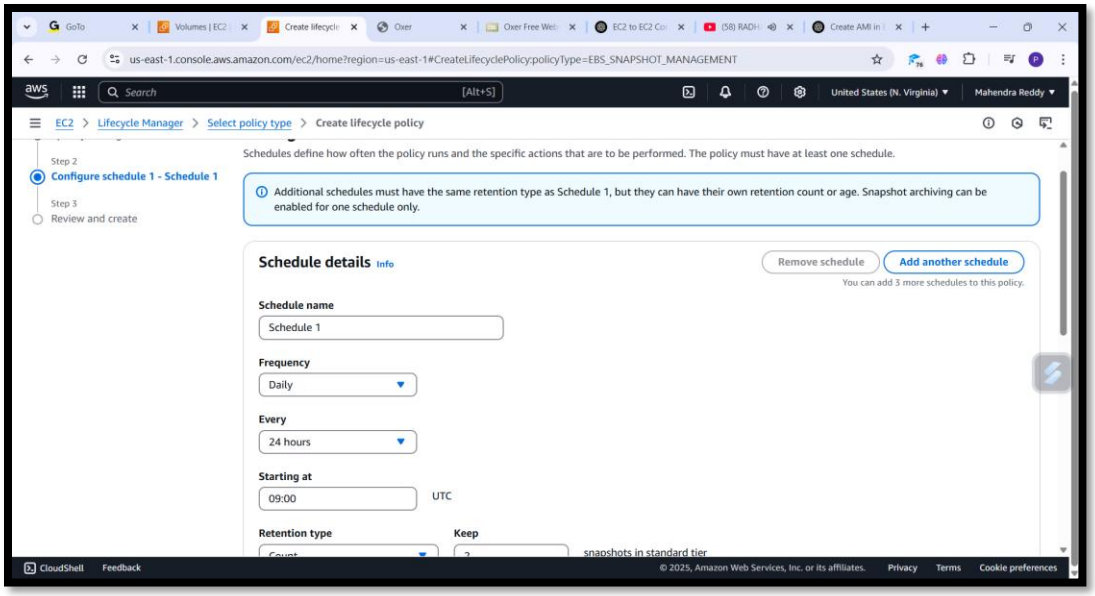
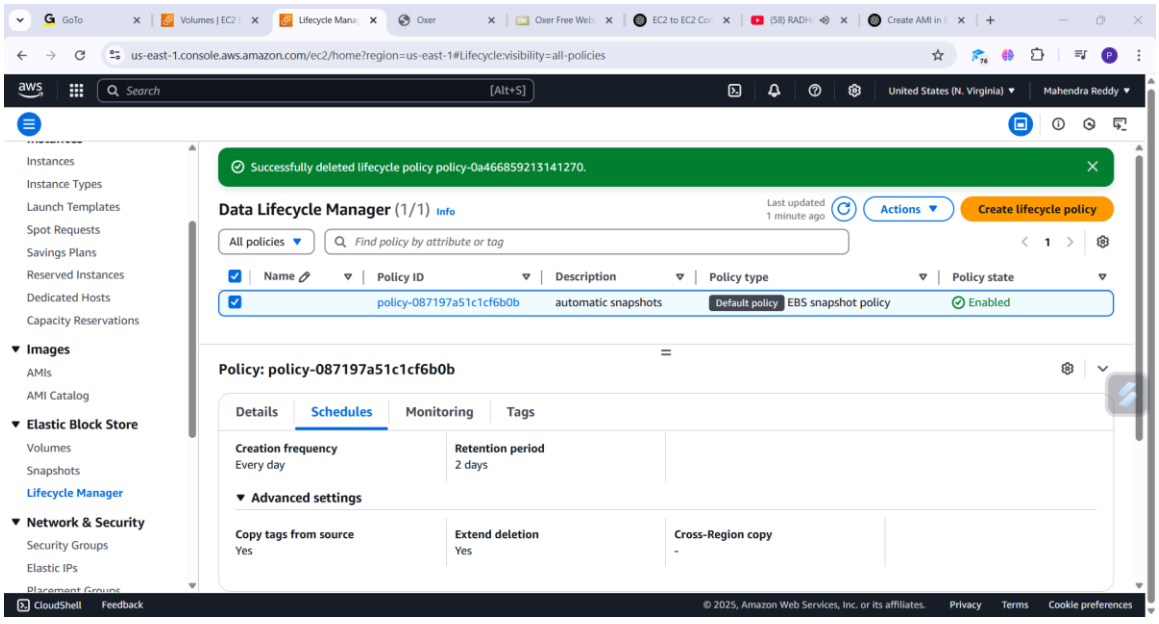


Fig. 3: Bastion Host Setup

Bastion host configured as a secure entry point for managing internal EC2 instances.



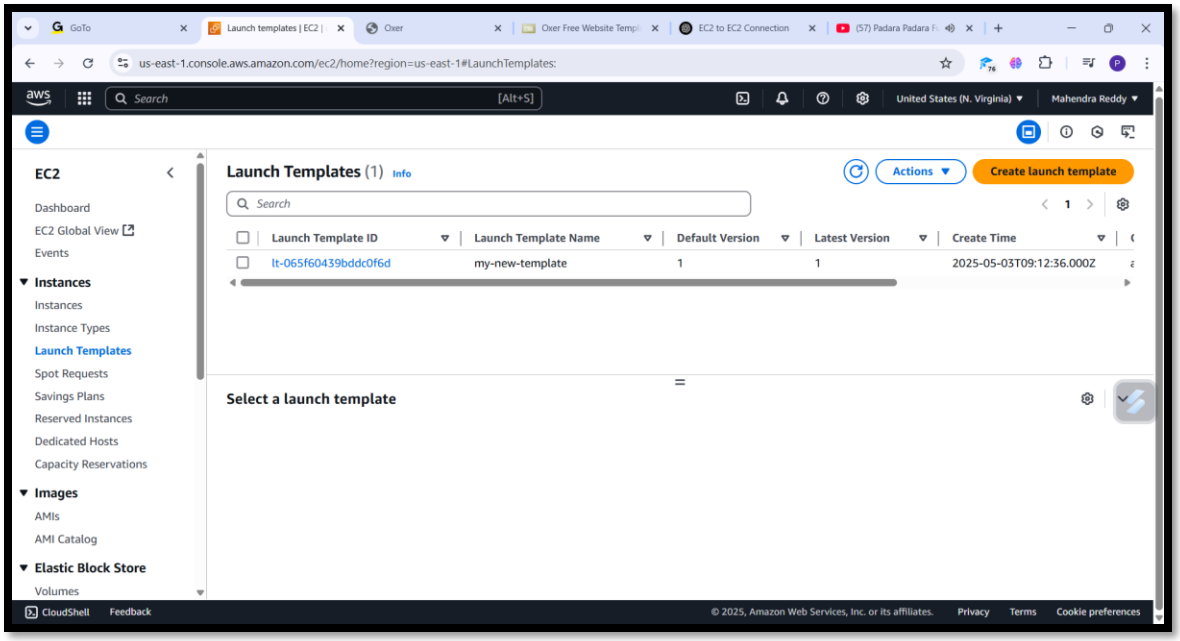


Fig. 4: Apache Installation

The terminal output showing Apache2 installation to serve the static website.

4. Create and Use Launch Template

Goal: Launch a new EC2 instance using a predefined configuration.

Expected Outcome:

- New instance launches with Apache and vsftpd installed.

- The static website is up and running.

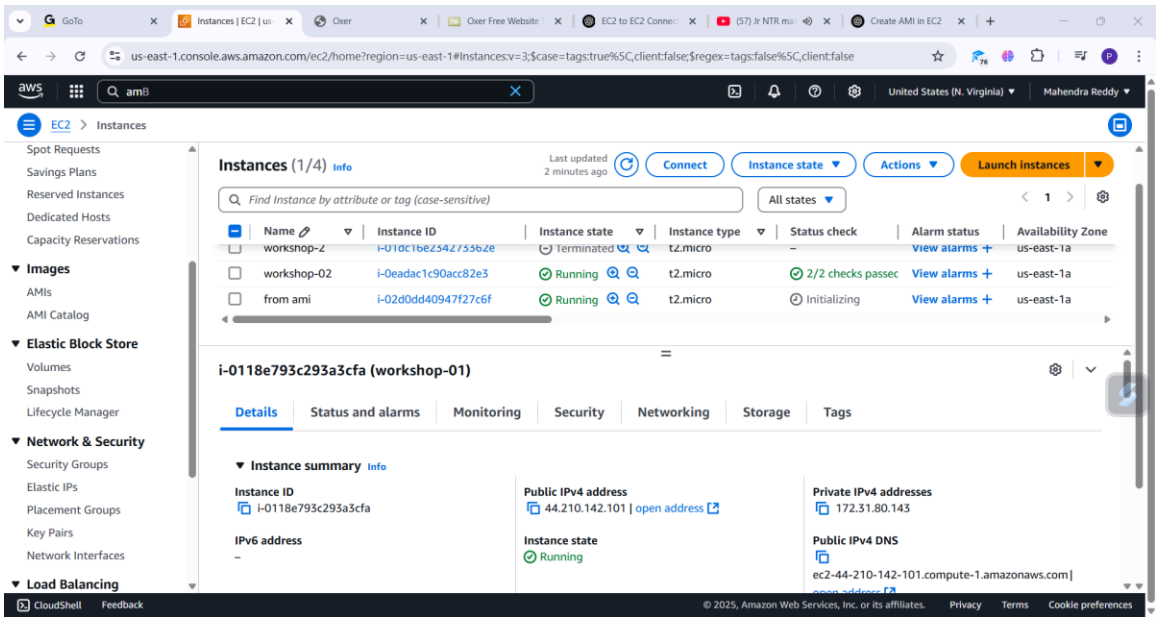


Fig. 5: Website Access

Accessing the hosted website through a public IP address in a browser.