1. As an AWS Cloud Engineer for a growing e-commerce platform, you must ensure high availability, data durability, and cost efficiency for EC2 instances and EBS volumes.

**Requirements**:

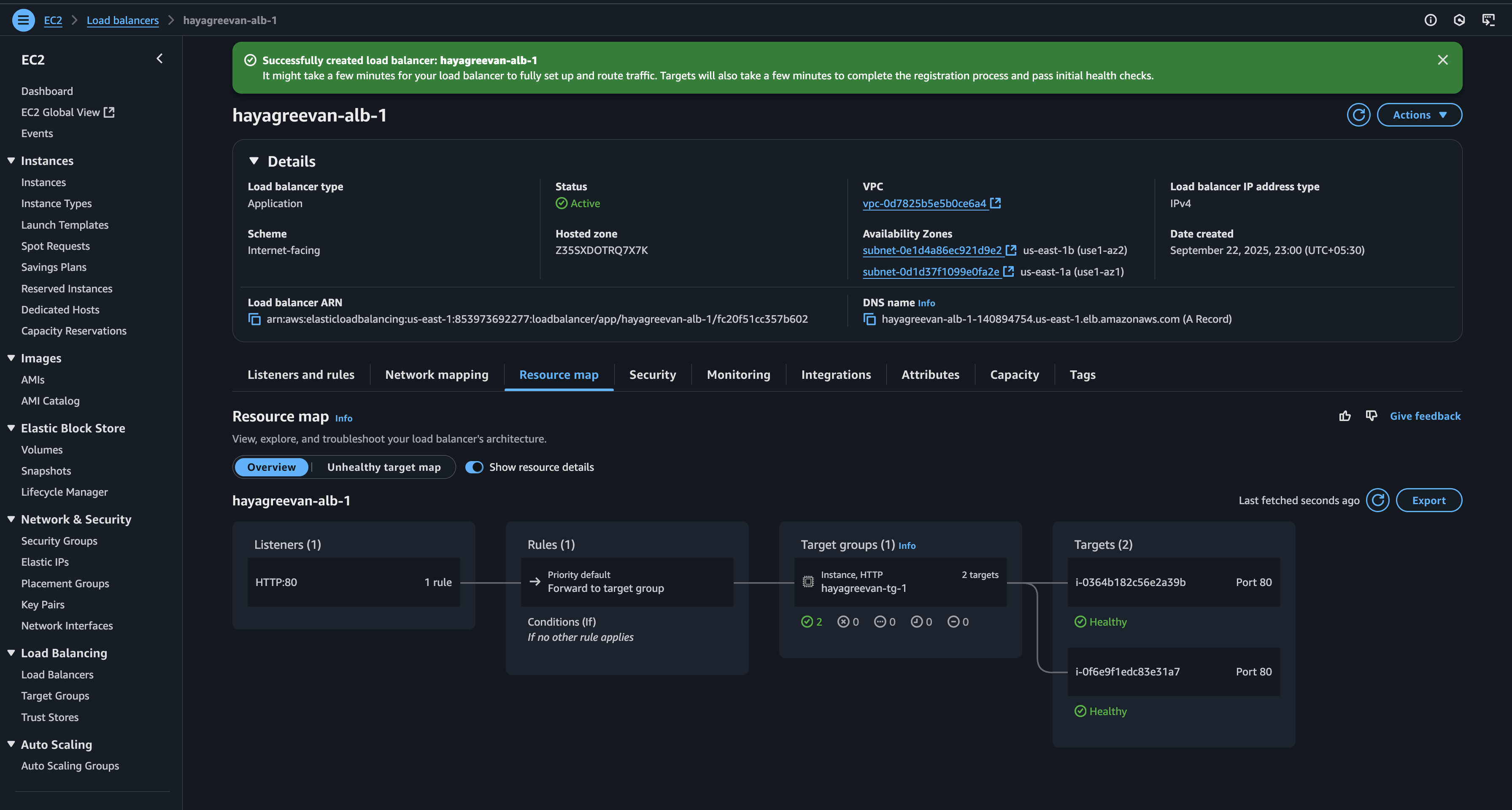
* Deploy EC2 instances across multiple Availability Zones (AZs) for high availability.
* Automate regular backups of EBS volumes to ensure data recovery.
* Implement cost-control mechanisms to prevent excessive spending on EC2 and EBS services.

**Questions**:

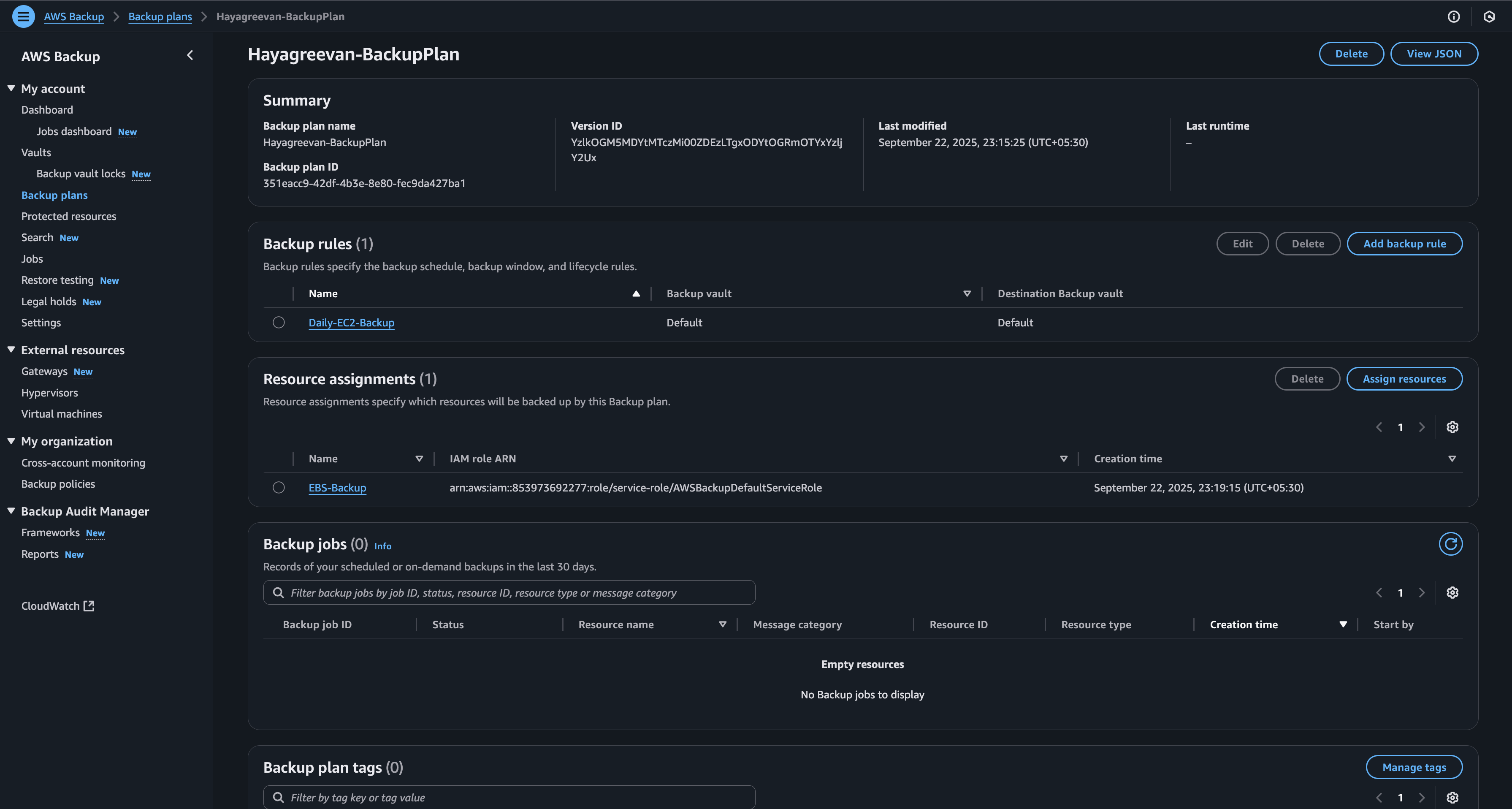
* How would you configure EC2 instances across multiple AZs for high availability?
* What steps would you take to automate regular backups of EBS volumes?
* How would you design a cost-control strategy to optimize EC2 and EBS usage?

Ans :

* Created EC2 Instances of Web server in Two different Availability Zones. Created a load balancer for these instances – High Availability



* Created AWS Backup plan for EBS Volumes



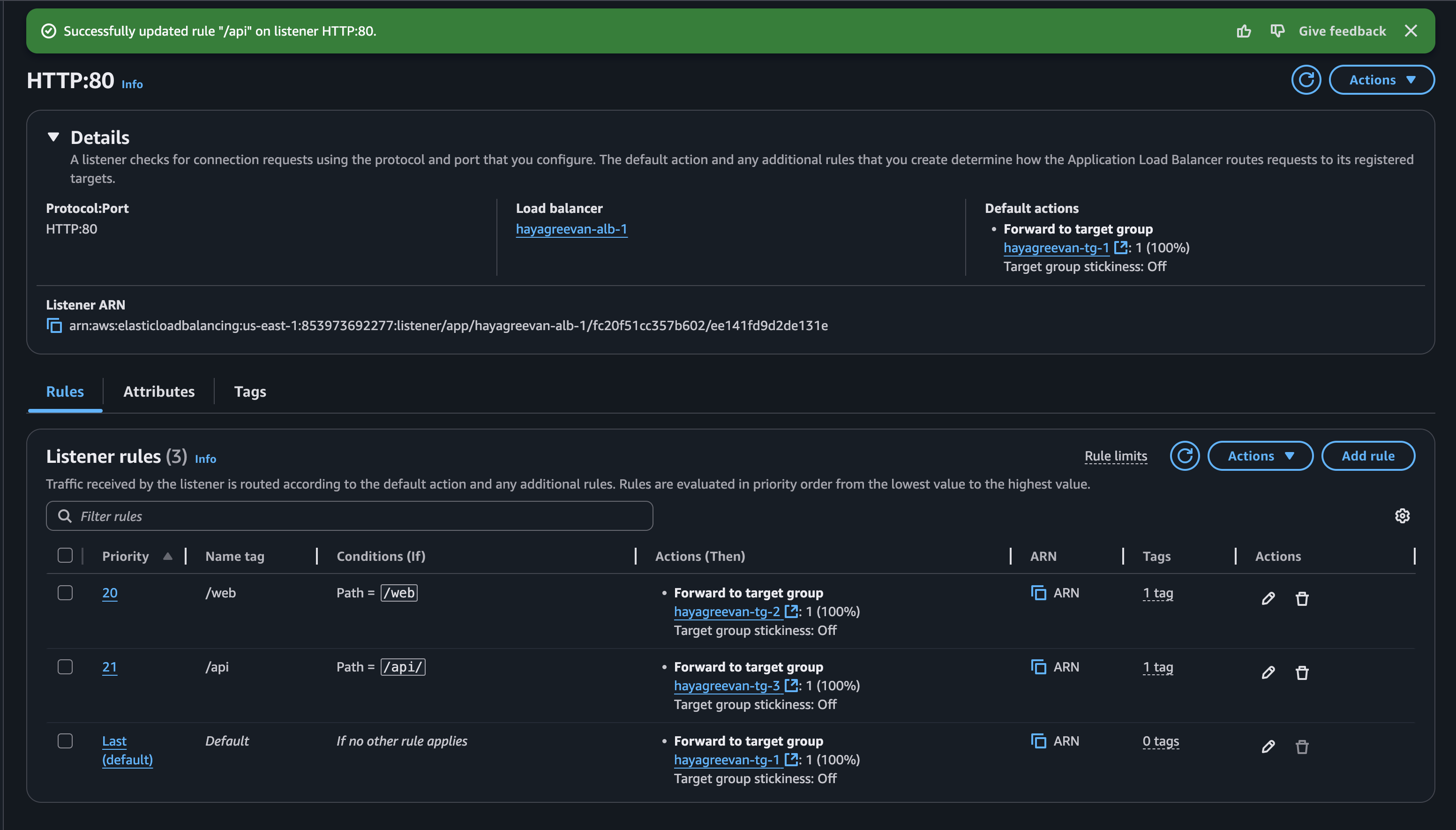
* For cost control mechanism, we can setup ASG with scale-in functionality.

1. You are tasked with deploying a scalable, highly available web application on EC2 instances to handle variable traffic efficiently using a load balancer.

**Requirements**:

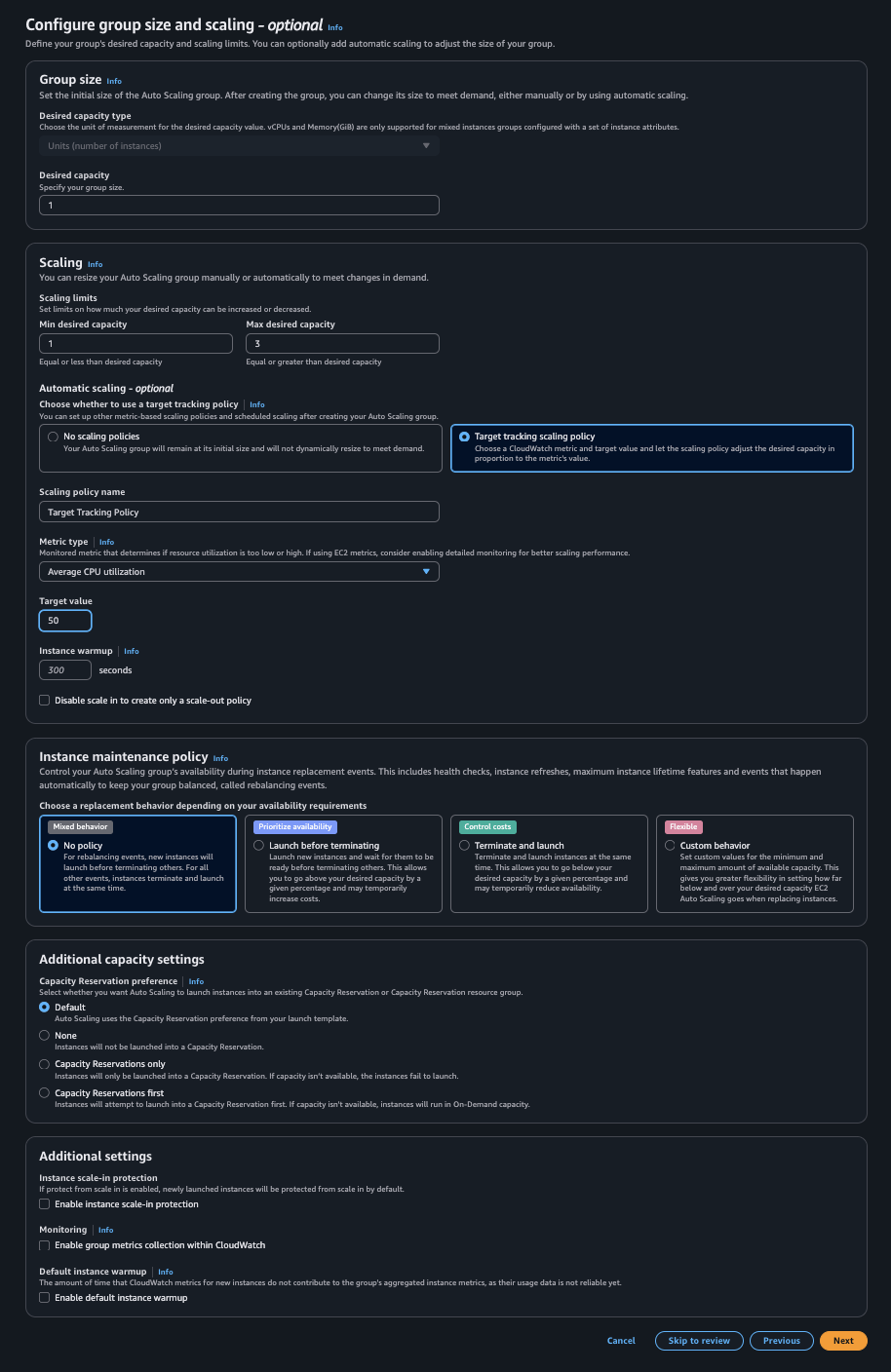
* Distribute traffic evenly across EC2 instances in multiple AZs.
* Redirect HTTP requests to HTTPS for secure communication.
* Monitor instance health and stop traffic to unhealthy instances.
* How would you configure an Application Load Balancer (ALB) to balance traffic across EC2 instances?
* Implement path-based routing to direct traffic to specific application components or services hosted on different servers (e.g., /api to one server and /web to another)
* How would you set up health checks to route traffic only to healthy instances?
* Configure Path based traffic both servers.
* Integrate Auto Scaling with the load balancer to automatically adjust the number of EC2 instances based on traffic and performance metrics (e.g., CPU utilization, latency).

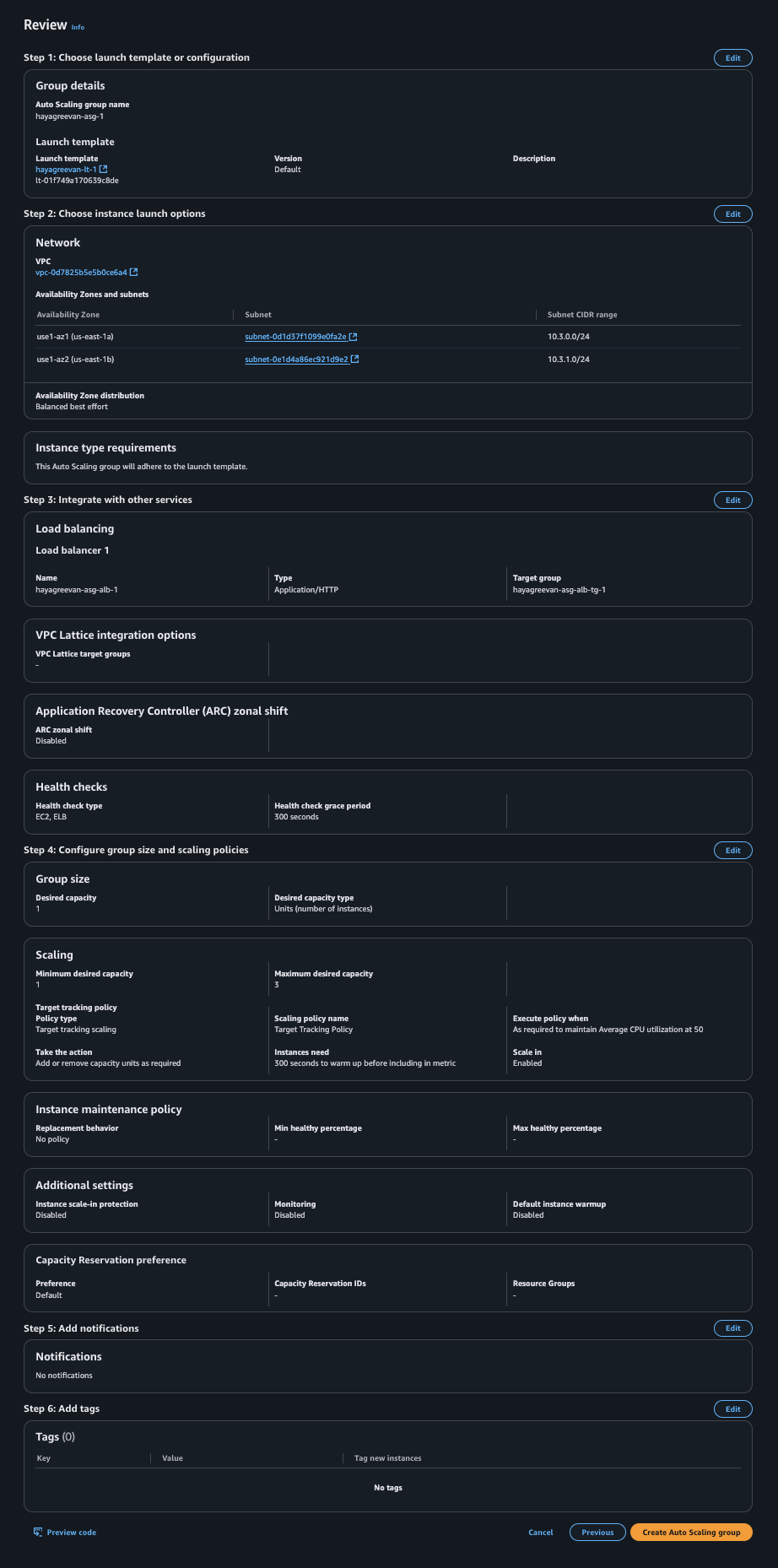
Path Based Routing in Application Load Balancer



Creating Auto Scale group :







3. Configure an Auto Scaling Group (ASG) that scales EC2 instances based on CPU utilization. Ensure each instance automatically installs necessary software packages upon launch using a bootstrap script.

**Requirements** :

* Operate across multiple Availability Zones for high availability.
* Set minimum, desired, and maximum instance counts (e.g., min: 1, desired: 2, max: 4).
* Implement Scaling Policies Based on CPU Utilization.
* Utilise bootstrap script for package Installation.

Bootstrap script (User data) :

#!/bin/bash

sudo apt update

sudo apt install apache2 -y

echo "Hello from $(hostname -f)" > /var/www/html/index.html

1. Use EC2 instance roles to manage AWS resources.

* Create an IAM role with policies to:
  + Read objects from a specific S3 bucket.
  + Write logs to CloudWatch.
* Launch an EC2 instance and attach the IAM role.
* Use the AWS CLI to:
  + List objects in the S3 bucket.
  + Upload a log file to CloudWatch.
* Test role-based permissions by disabling the role temporarily and re-enabling it.