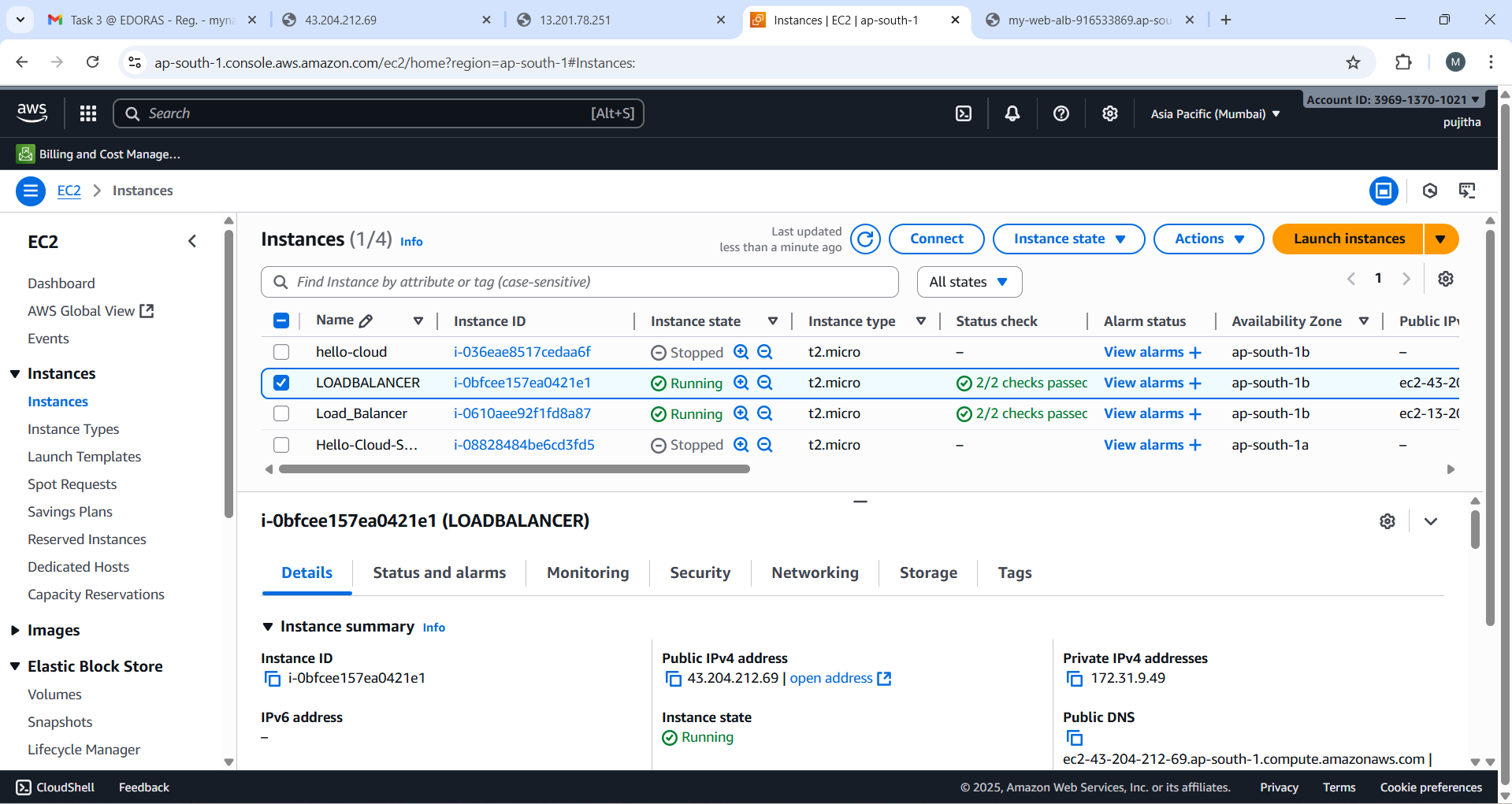
**Documentation**

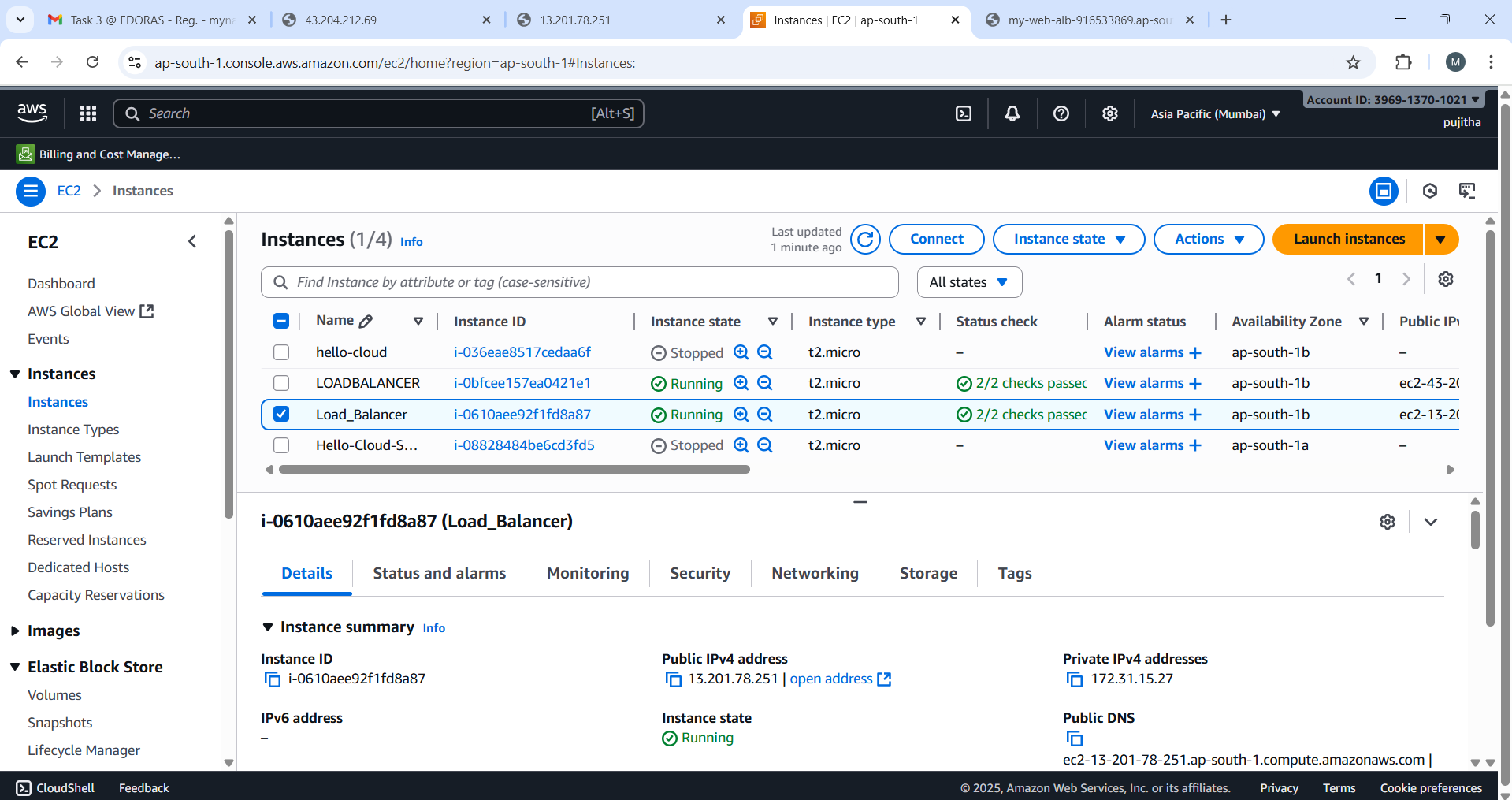
**Below of Screenshots:**

**1.EC2 launch page for both instances.**

* Both instances are shown in the EC2 dashboard with clear identification each has a unique instance ID to avoid Confusion between resources**.**

****

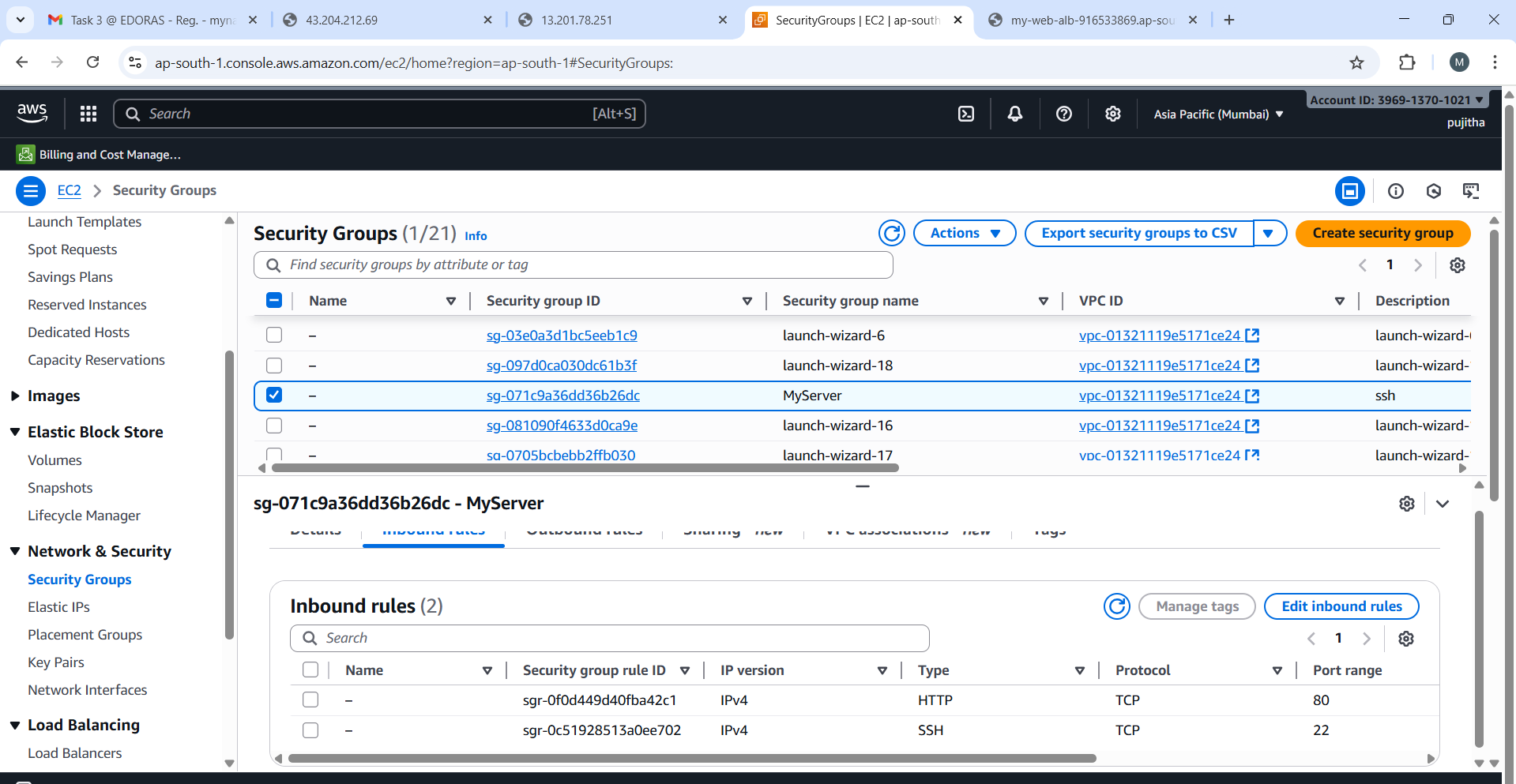
* Displays public and private IP addresses for SSH access and networking.



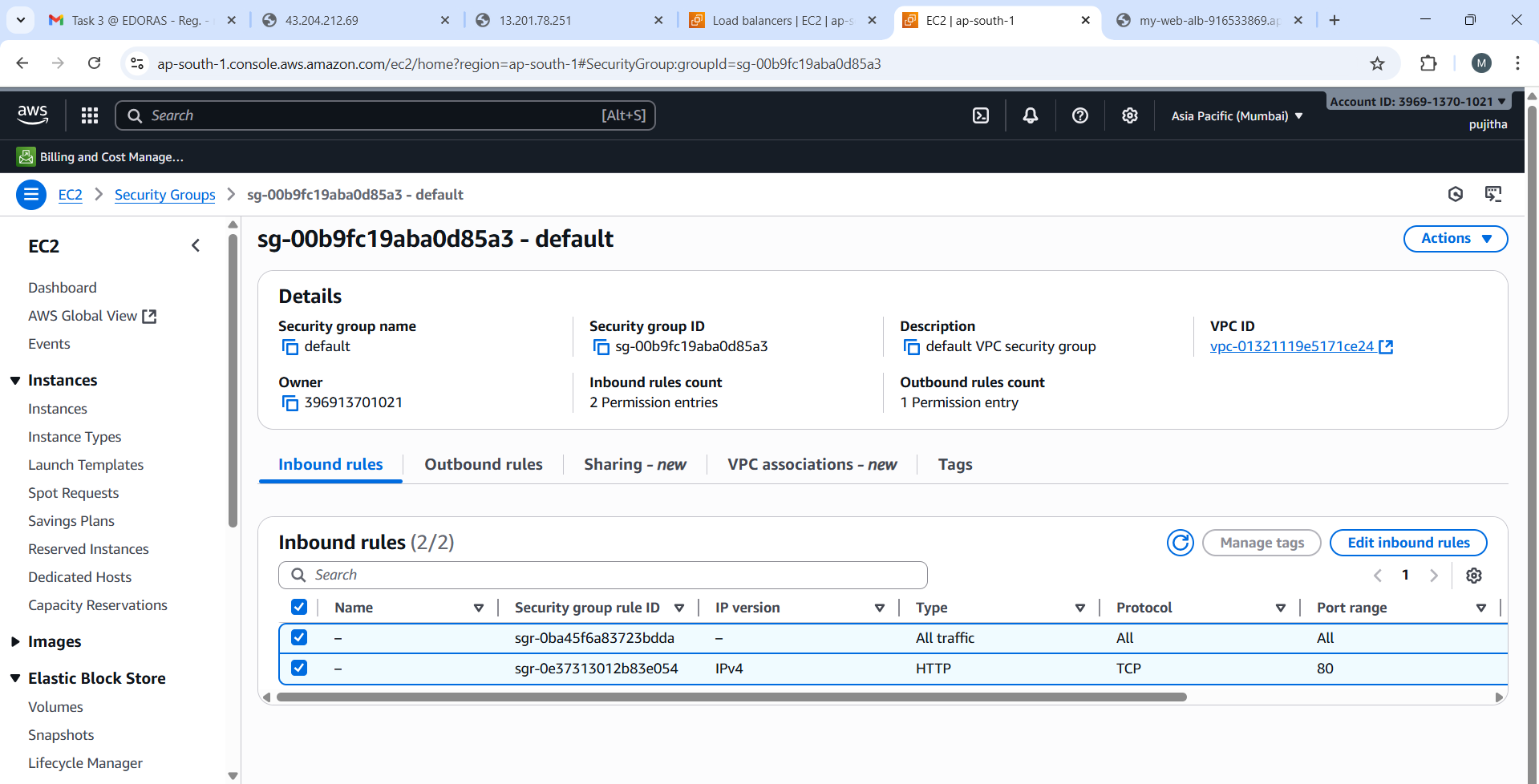
* Includes monitoring and status check tabs for health and performance.
* Provides Key configuration information for quick reference

**2. Security group rules:**

* Show the security group assigned to the EC2 instance and the load balancer.
* Highlight inbound rules that allow traffic (e.g., HTTP/HTTPS for web, SSH for management.



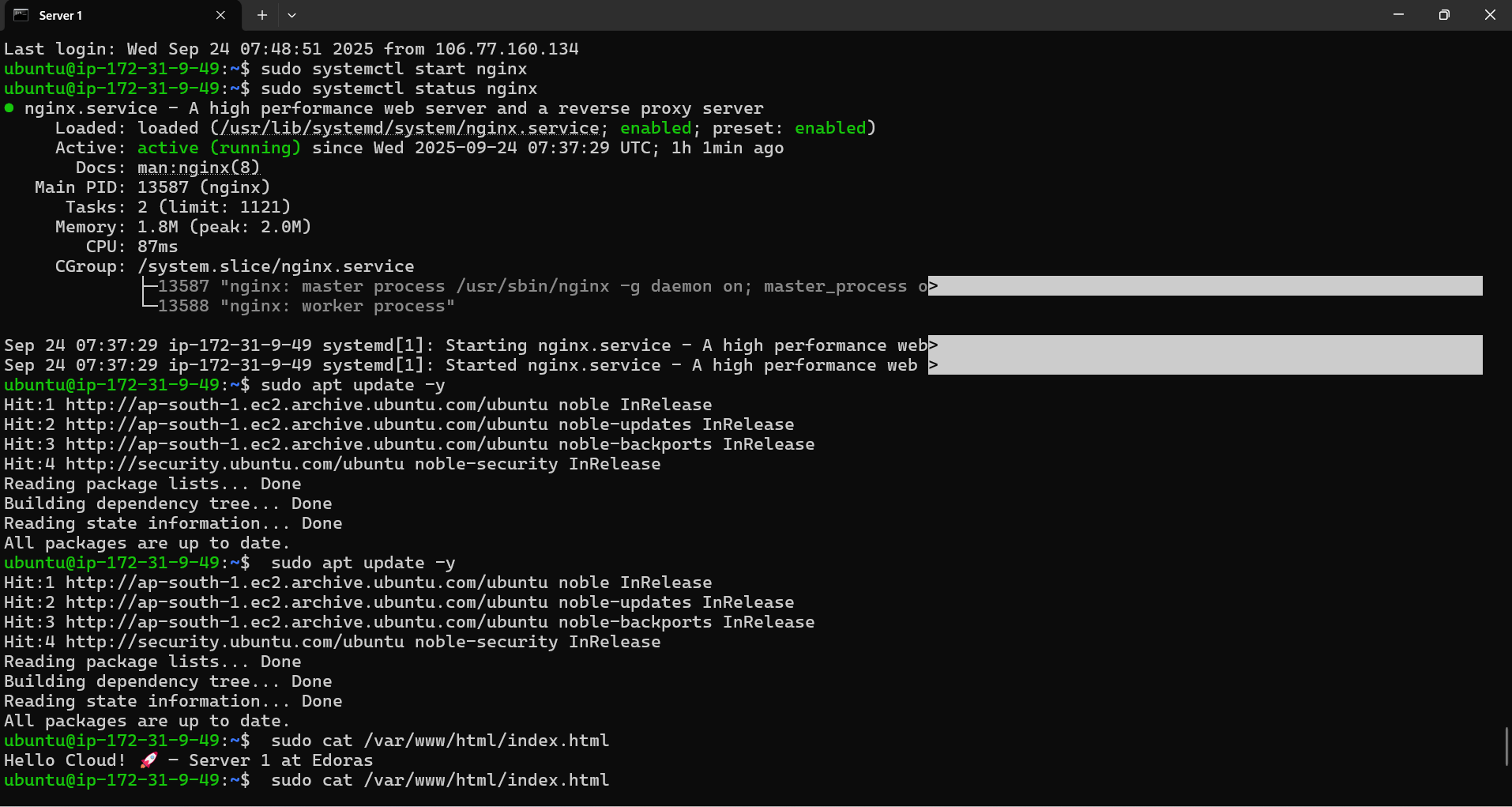
* Show that the load balancer’s security group allows inbound traffic from the internet.



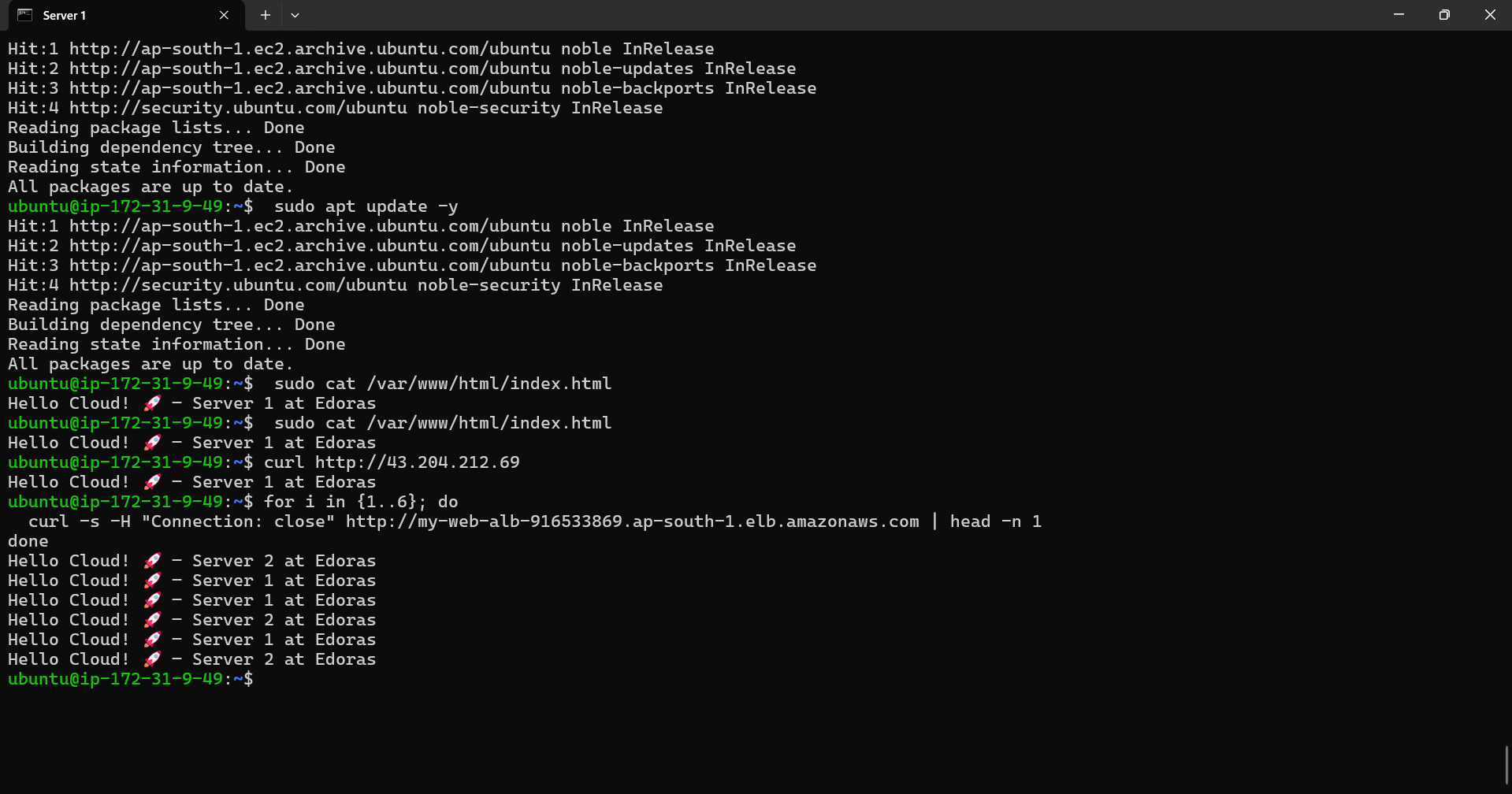
* Show EC2’s security group allows inbound traffic only from the load balancer’s security group.
* llustrate how security groups act like virtual firewalls controlling access.

**3.Apache/Nginx setup with unique server messages.**

* I Installed and configured Apache/Nginx on two EC2 instances.
* I Started and enabled the service to run on boot, then modified the default index.

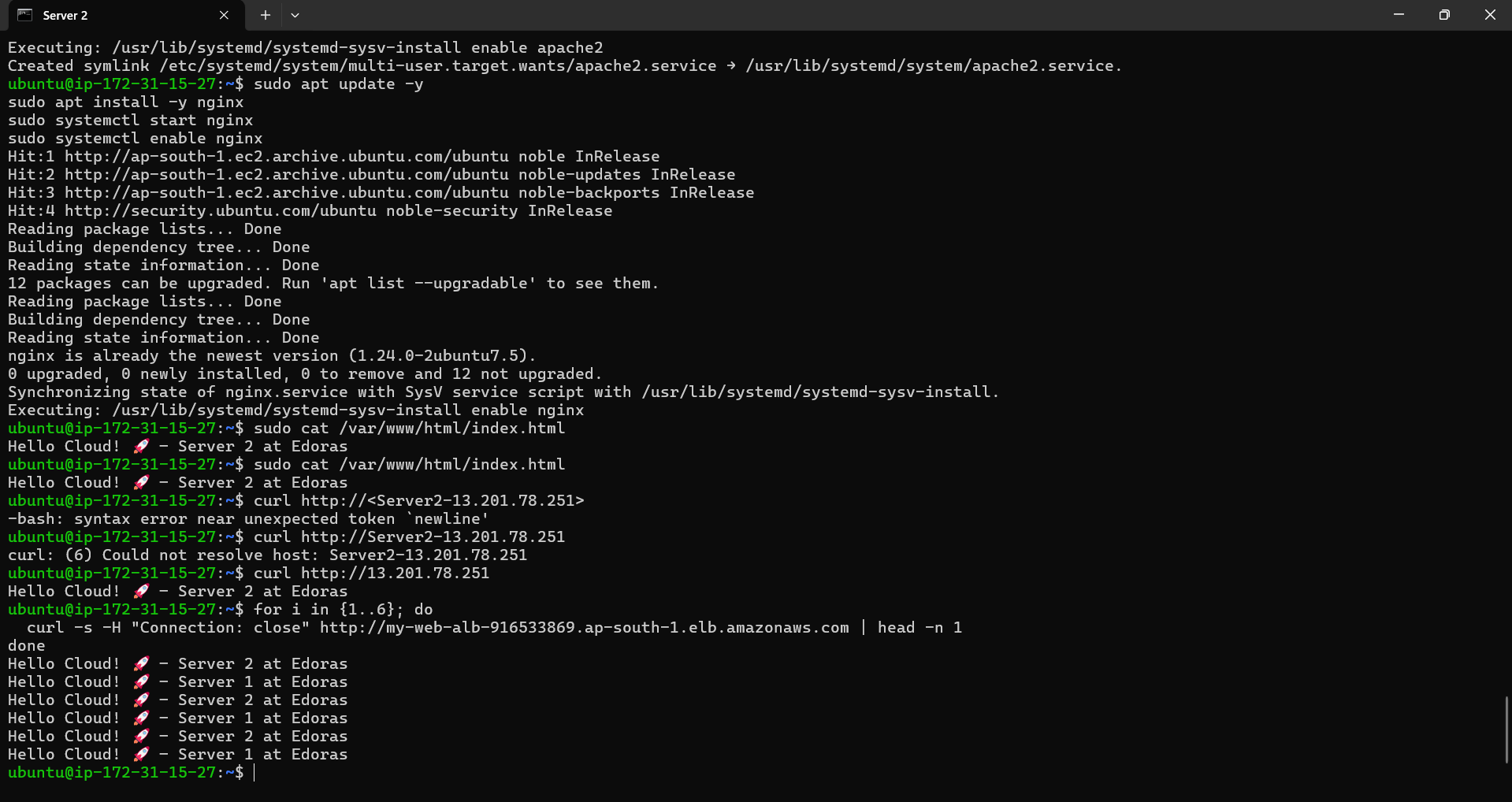
****

* Html file to display unique message was *‘Hello Cloud -Server 1 at Edoras,*’ and for Server 2, it was *‘Hello Cloud!-Server 2 at Edoras’.*

**



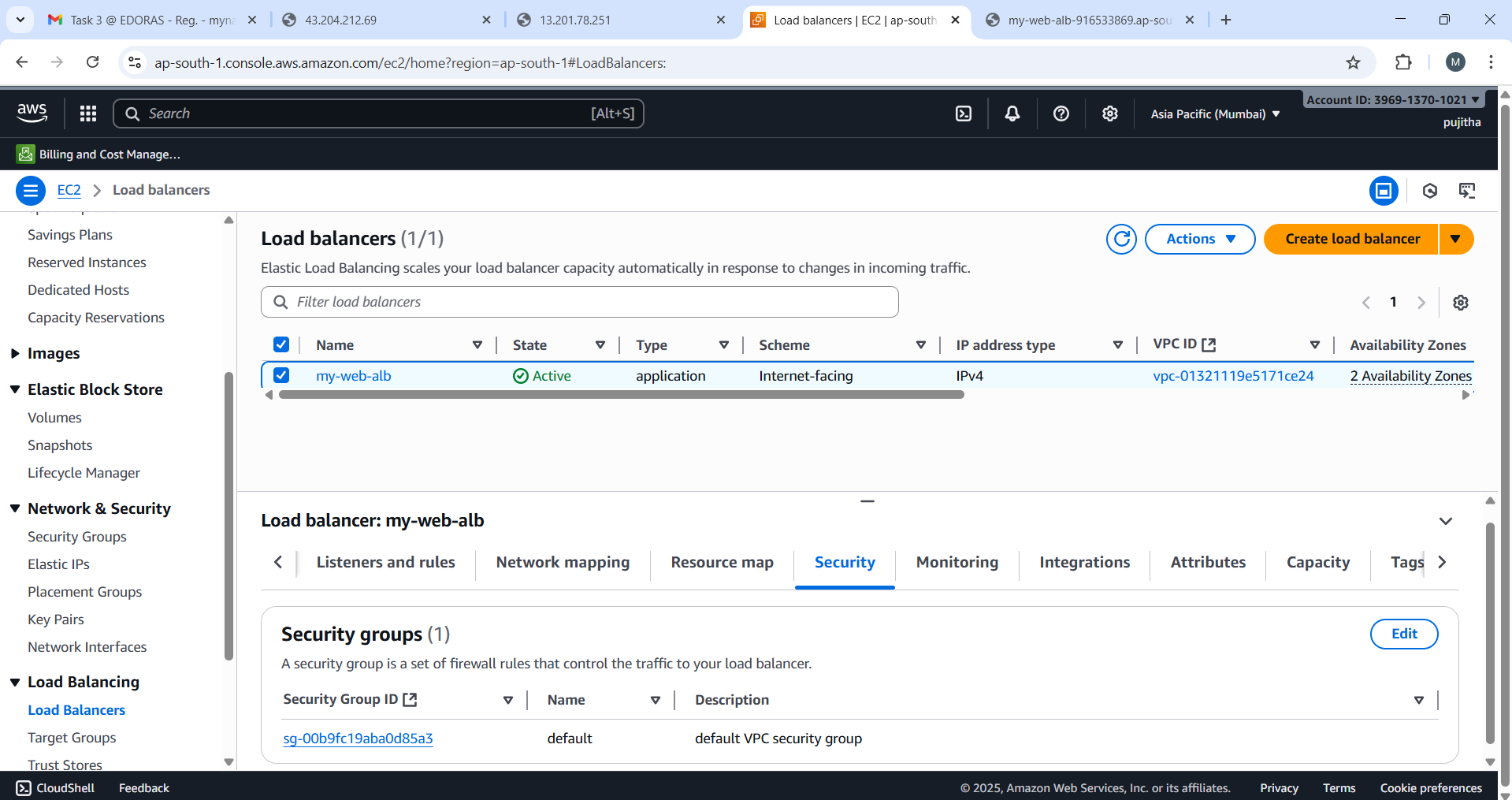
* I verified the setup by accessing the EC2 public Ips in the browser and confirmed each instance returned the correct unique message.

****

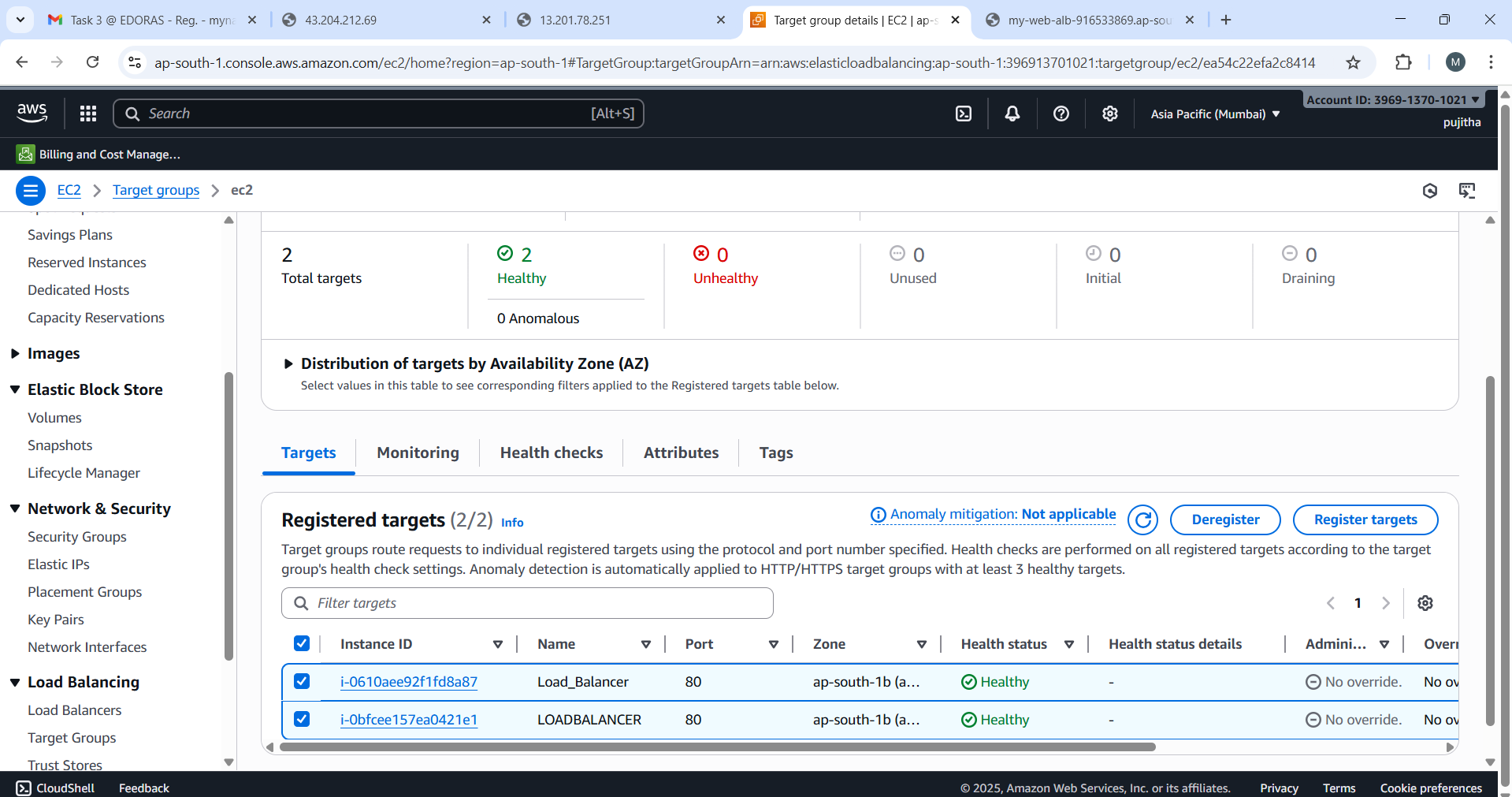
* Finally, I ensured HTTP traffic (port 80) was allowed in Security Group so the Load Balancer could route requests to both server.

**4.Load Balancer creation & health checks.**

* I Created an Application Load Balancer in AWS and configured it to listen on HTTP port 80.
* I then created a Target Group and registered my two EC2 instances in it.

****

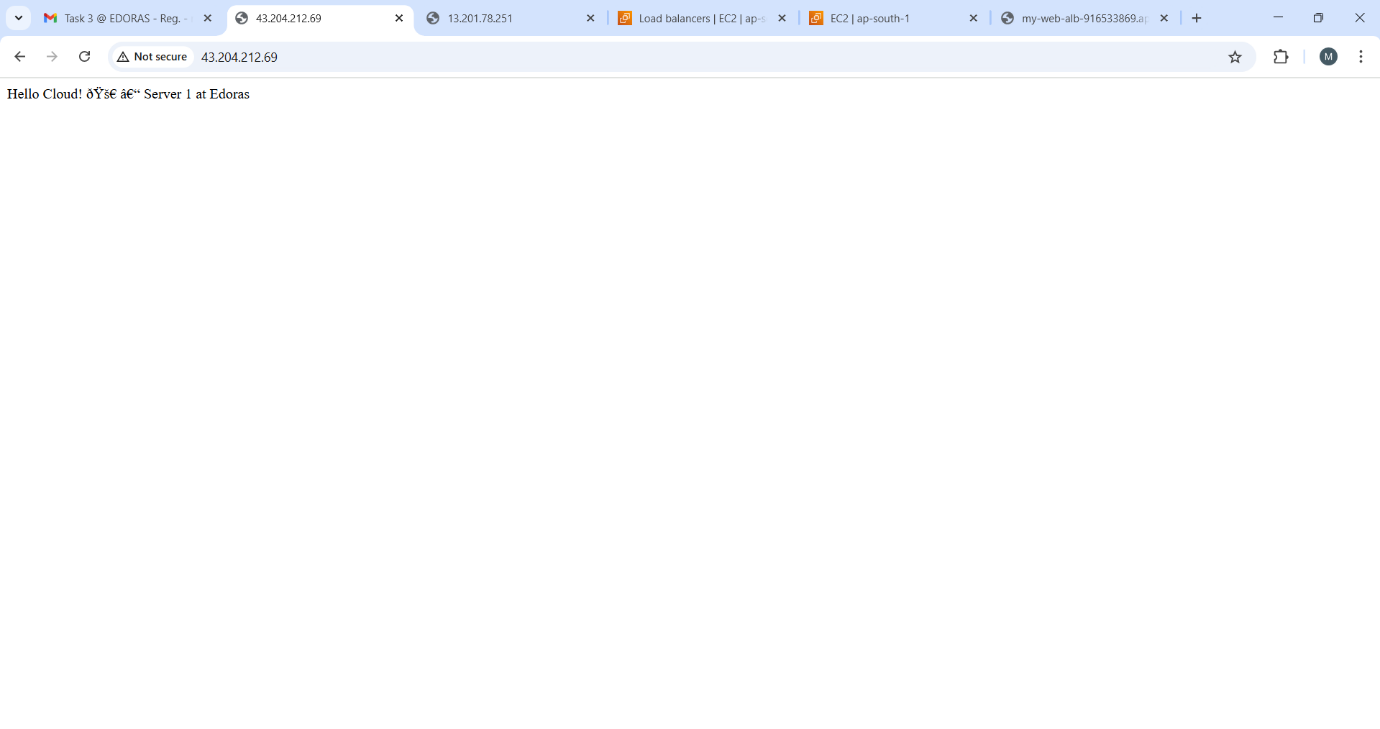
* I set up health checks on the /path with HTTP protocol to monitor instance availability.
* After ensuring both instances showed as healthy, I tested the ALB DNS name in my browser.

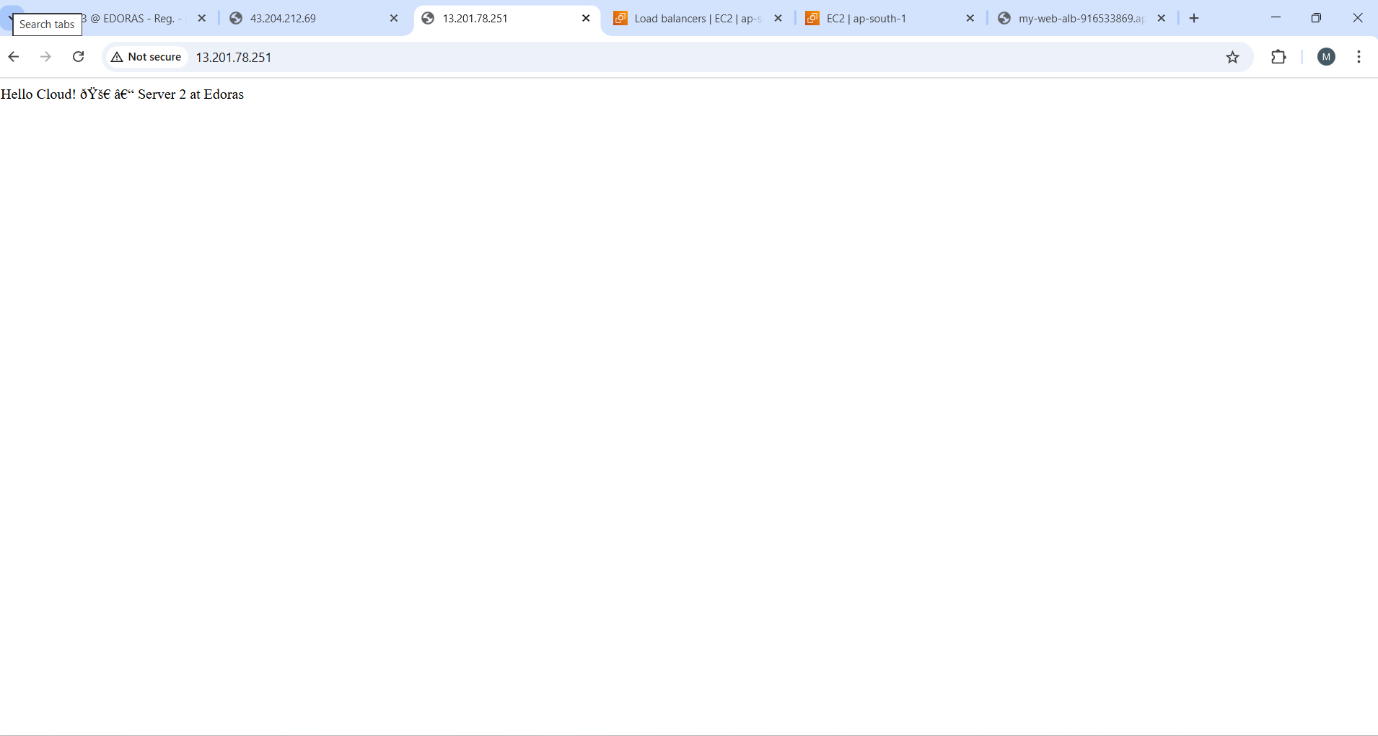
****

* Refreshing the page alternated responses between ‘Server 1’ and ‘Server 2’.
* I ALB only routes traffic to the healthy server, ensuring high availability.

**5.Browser output showing Server 1 / Server 2 responses when refreshing.**

* After completing the setup, I accessed the Application Load Balancer DNS name in my browser.
* When I refreshed the page multiple times, I observed alternating responses: one refresh showed the unique message from Server 1 (‘Hello Cloud! – Server 1 at Edoras’), and the text next showed the message from Server 2 (‘Hello Cloud! – Server 1 at Edoras’).
* 

****

****This confirmed that the load balancer was successfully distributing traffic between the two instance in a round-robin manner, ensuring high availability.