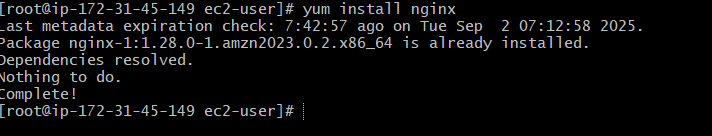
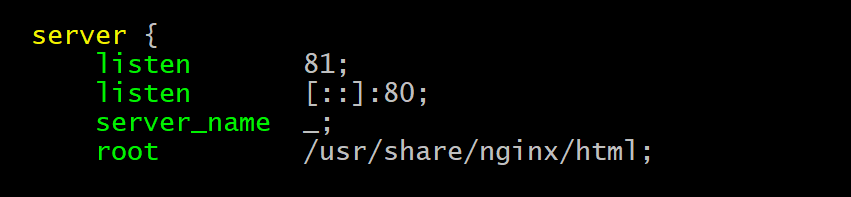
1. Install nginx and run nginx on port number 81.

Step 1:Use **yum install nginx** to install nginx



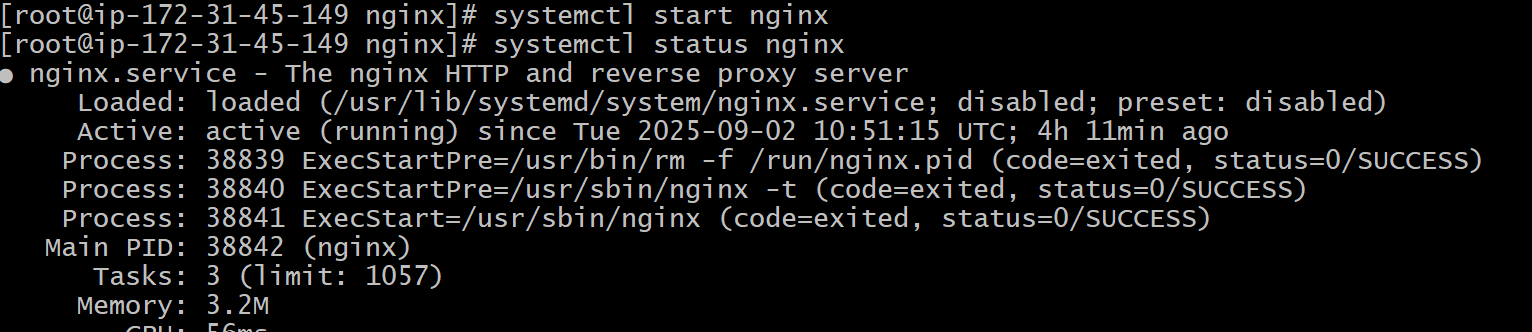
Step 2: find **nginx.conf** file and change port to 81

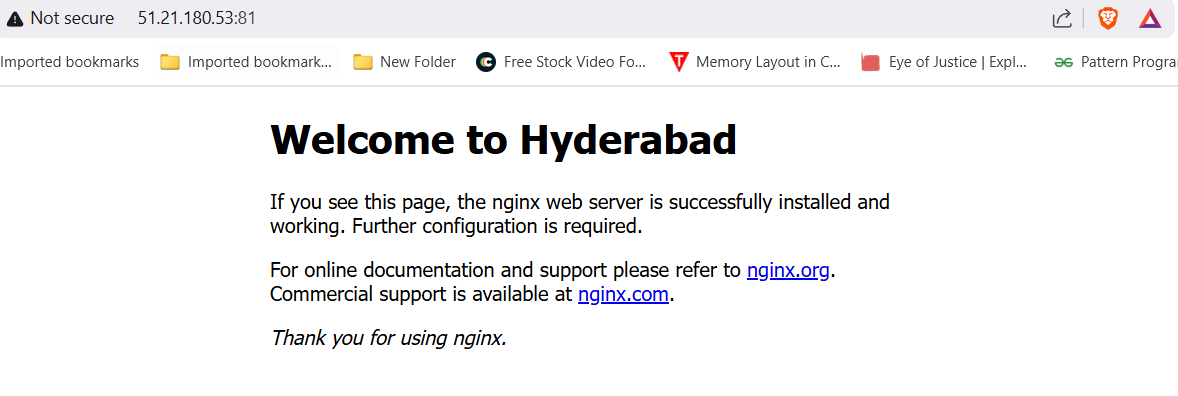


Step 3: Add port 81 in security group of AWS Ec2 instance



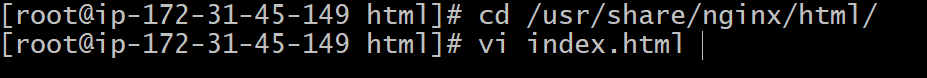
Step 4: Use **systemctl start nginx** to start

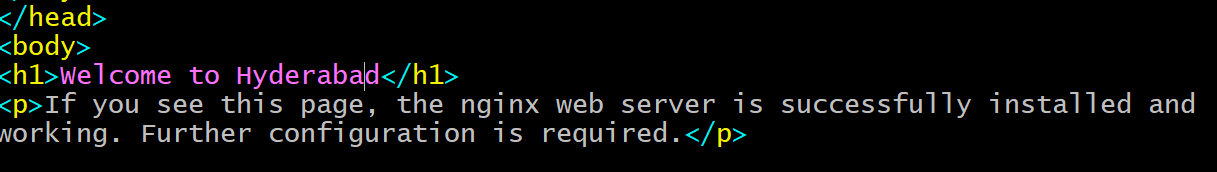


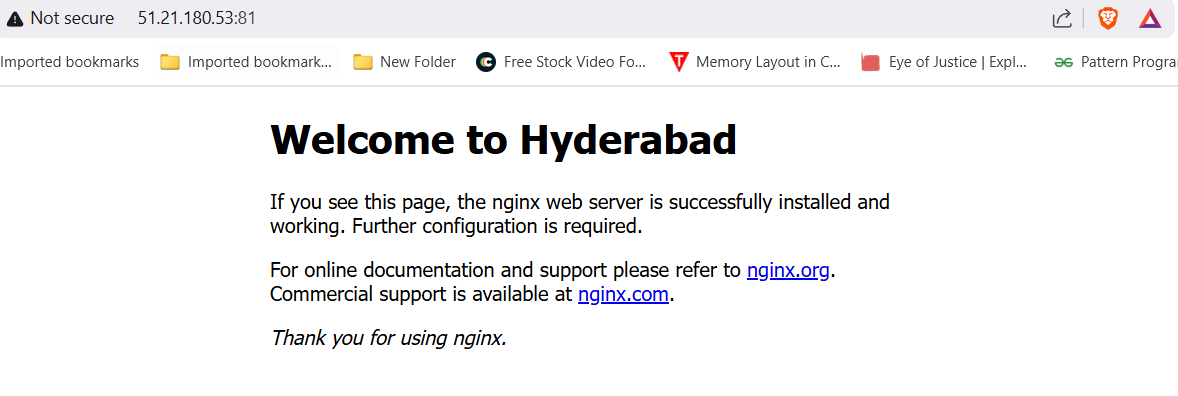


1. Deploy a sample index.html file on nginx.

Step 1: go to **/usr/share/nginx/html/** and create a file named **index.html** and enter the data and restart it.

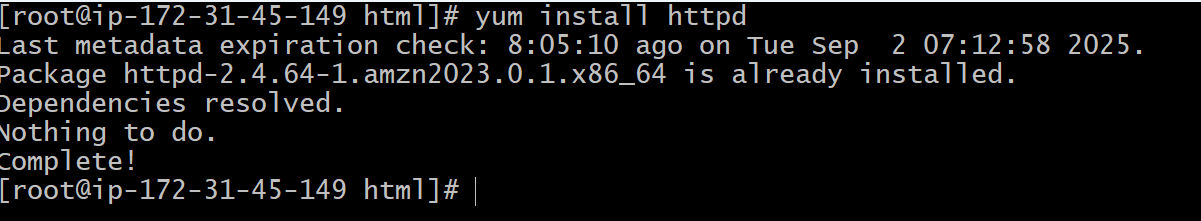




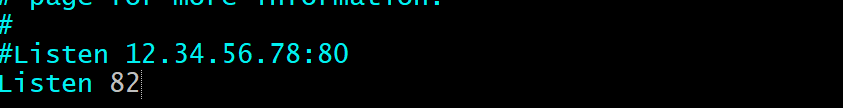


1. Install Apache and run Apache on port number 82

Step 1: use **yum install httpd** to install apache



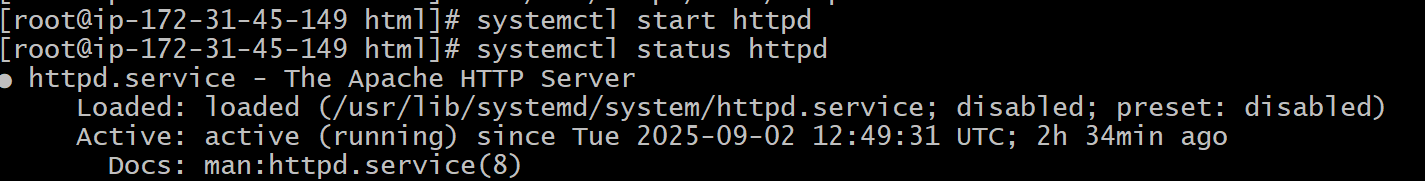
Step 2: find **httpd.conf** file and change port to 81

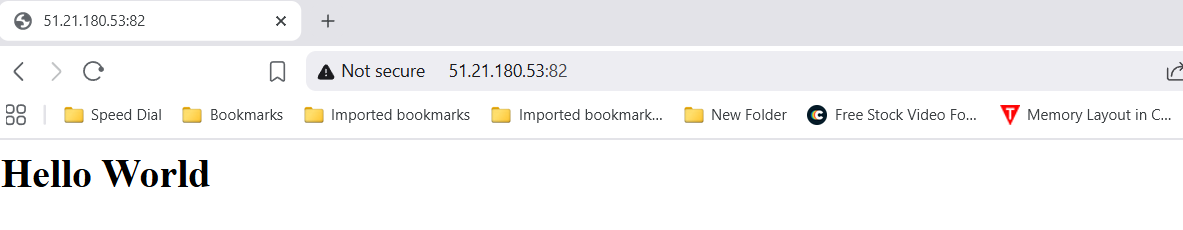


Step 3: Add port 82 in security group of AWS Ec2 instance



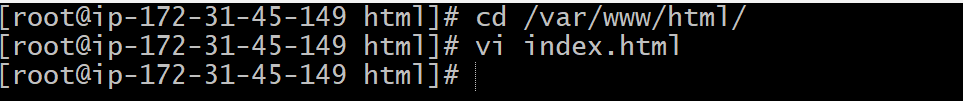
Step 3: use **systemctl start httpd** to start apache





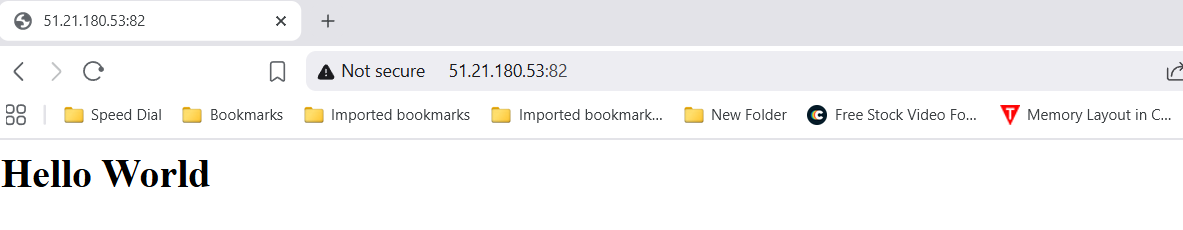
4.Deploy a sample index.html file on Apache.

Step 1: go to **/var/www/html/** and create **index.html** and add data



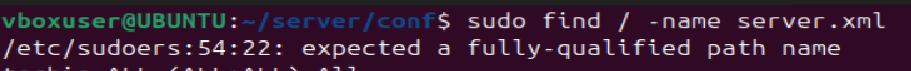


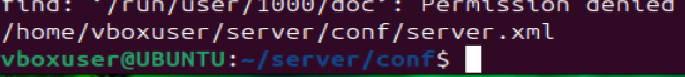
Step 2: start apache using **systemctl start httpd**



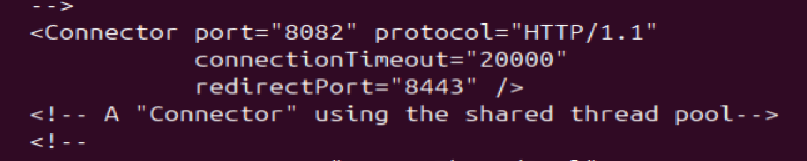
1. Install Apache tomcat on port number 8082

Step 1: Use **sudo find / -name server.xml** to find server.xml file



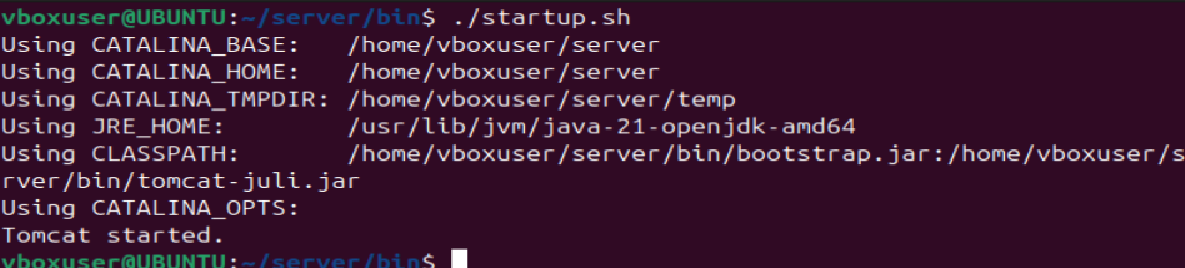


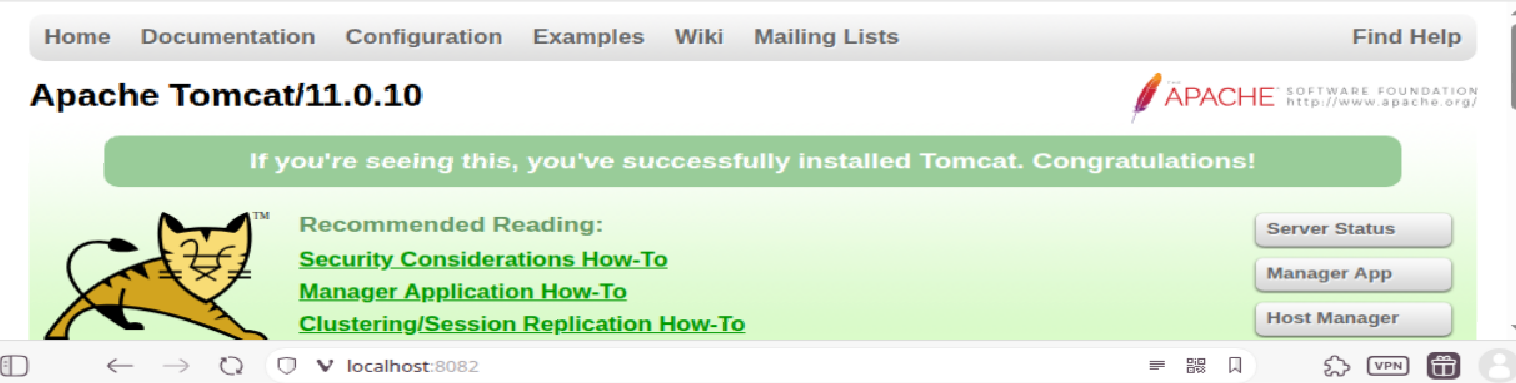
Step 2: Change port from 8080 to 8082 in server.xml file



Step 3: start Apache by using **./startup.sh** and enter **localhost:8082**

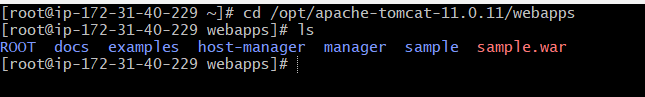
in a browser



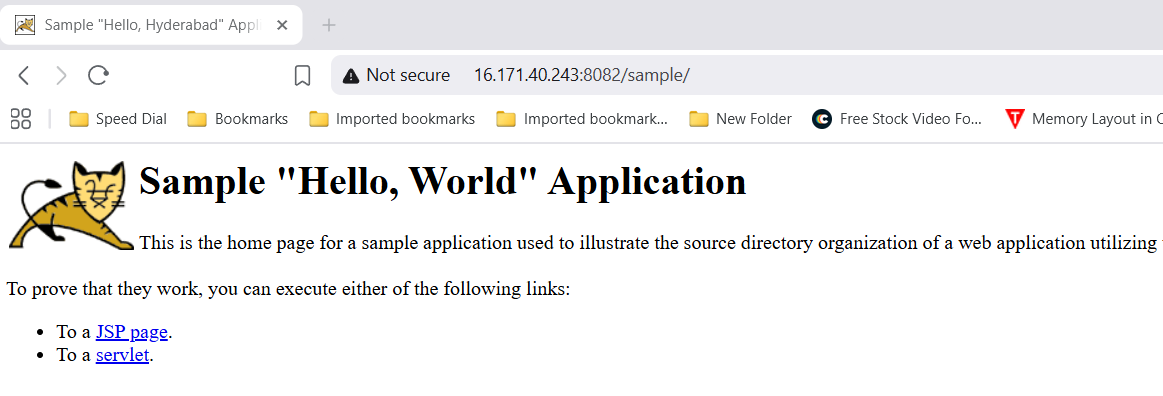


1. Deploy a sample app on webapps

Step 1: use command **cd /opt/apache-tomcat-11.0.11/webapps** to webapps directory and use **wget sample.war\_url** to download sample file and extract it

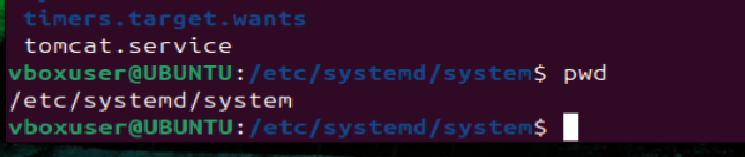


Step 2: Enter **publicip:8082/sample/** in browser to run sample webapp

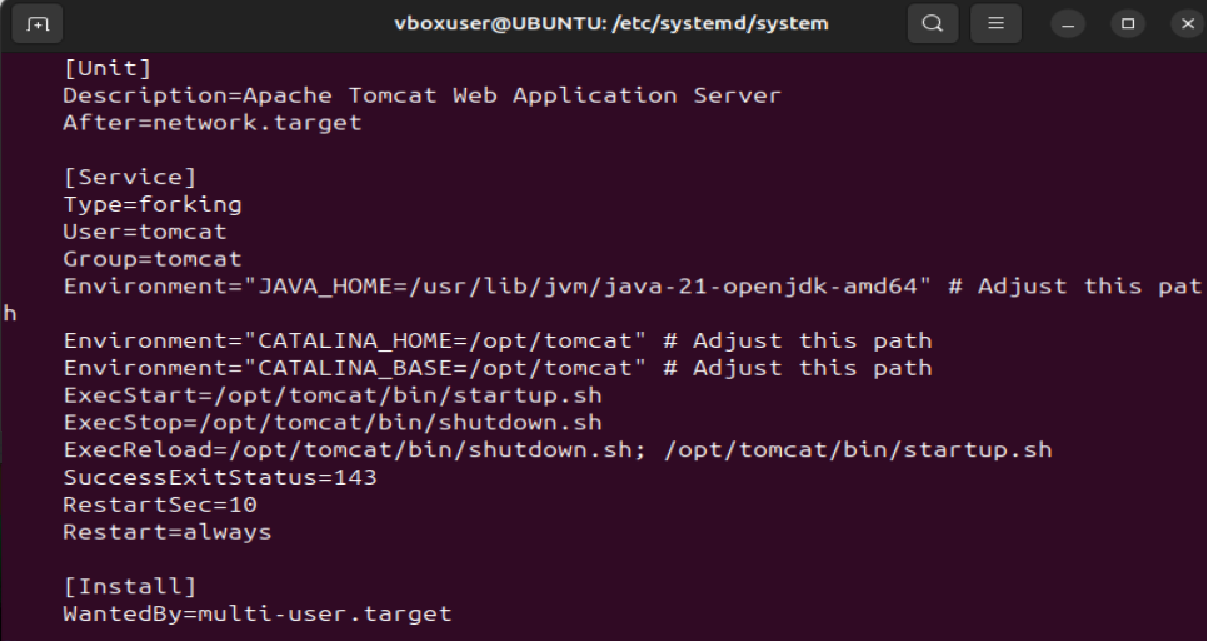


1. Create a tomcat.service file for tomcat.

Step 1: create **tomcat.service** in **/etc/systemd/system/**



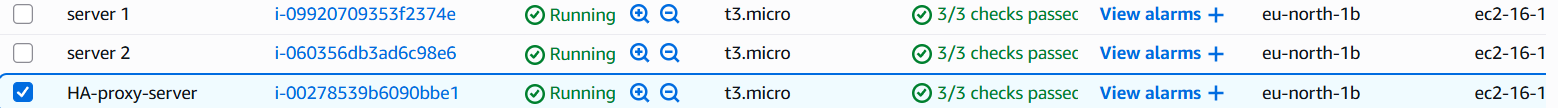
Step 2: Enter service configuration data in **tomcat.service** file



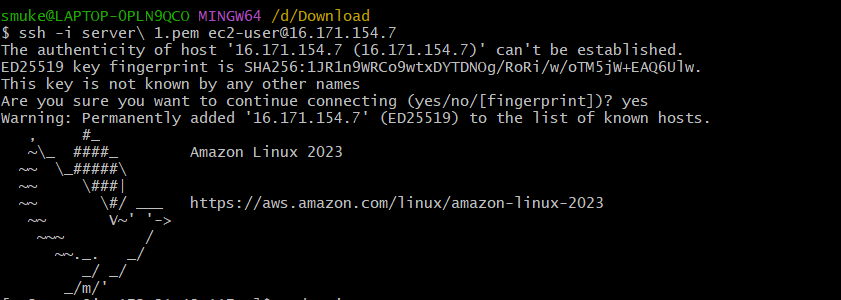
Step 3: save and reload by using **systemctl daemon-reload** command to recognize new service file

1. Configure HA proxy server

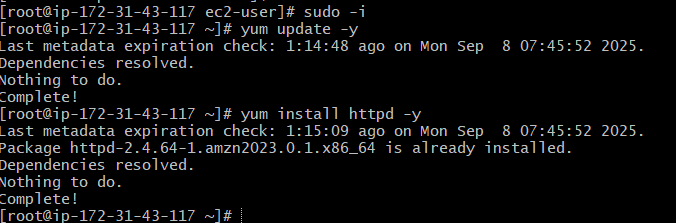
Step 1: Launch 3 AWS Ec2 instances named server1, server2, HA proxy.



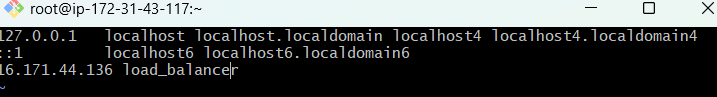
Step 2: Connect to server1 and run following commands



* Sudo -i
* Yum update -y
* Yum install httpd -y

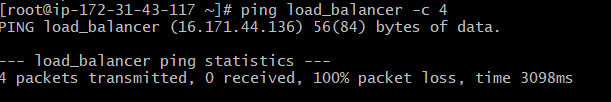


Step 3: Add public ip of HA-Proxy server in **vi /etc/hosts** and mention as load\_balancer



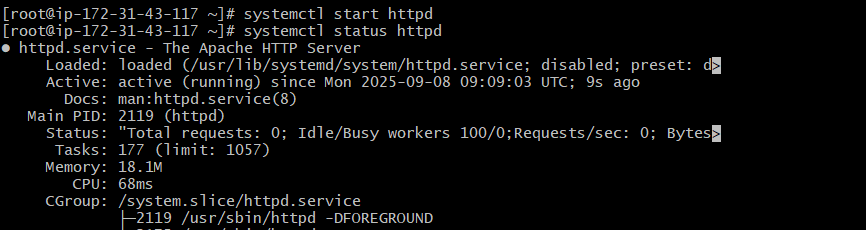
Step 4: Run following commands on server 1

**Ping load\_balancer -c 4**

****

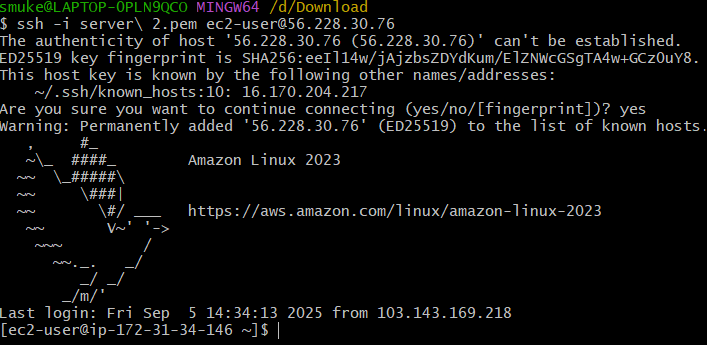
**Systemctl start httpd**

**Systemctl status httpd**

****

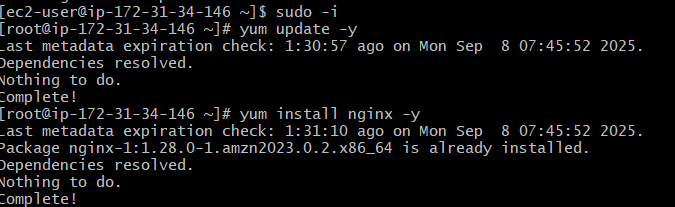
In your browser search for public ip of server 1 with port :80 **13.60.196.5:80**

Step 5: Parallely connect to server-2 using its public ip

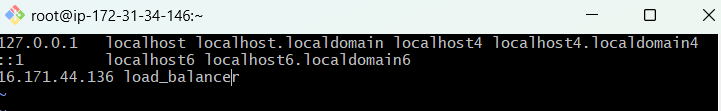


Step 6: Run following commands

* **sudo -i**
* **yum update -y**
* **yum install nginx -y**

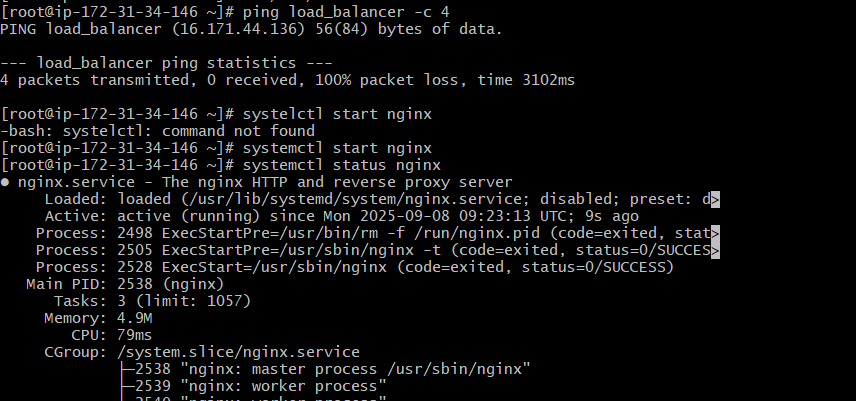


Step 7: Add public ip of HA-Proxy server in **vi /etc/hosts** and mention as load\_balancer



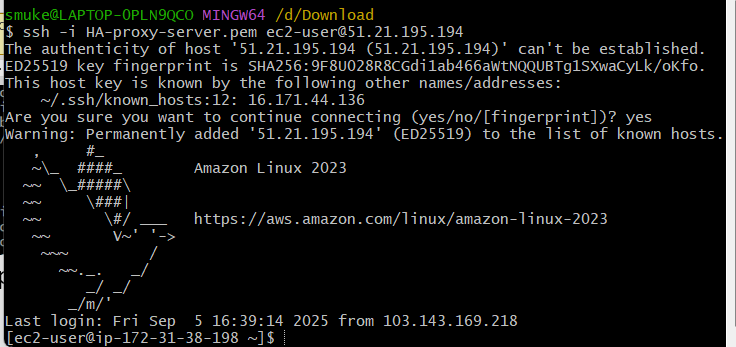
Step 8: Run following commands

* **ping load\_balancer -c 4**
* **systemctl start nginx**
* **systemctl status nginx**

****

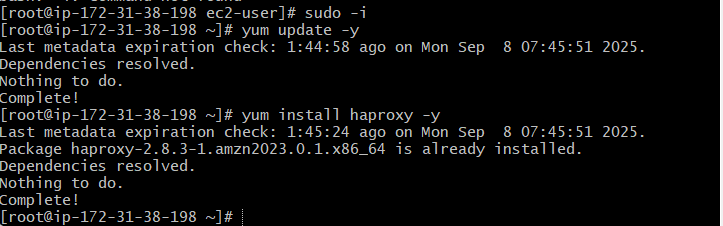
In your browser search for public ip of server 2 with port :80 **56.228.30.76:80**

Step 9: Parallely connect to HA-Proxy server using its public ip

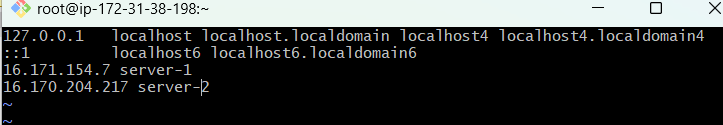
****

Step 10: Run following commands

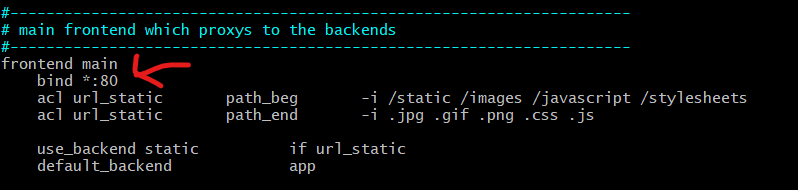
* **sudo -i**
* **yum update -y**
* **yum install haproxy -y**

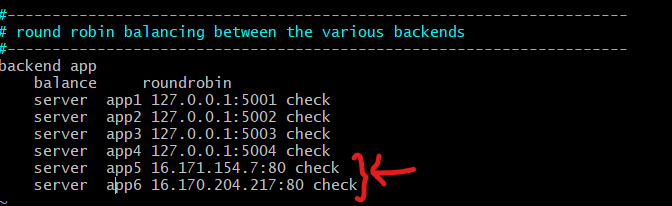


Step 11: Add server 1 and server 2’s public ip in **vi /etc/hosts**



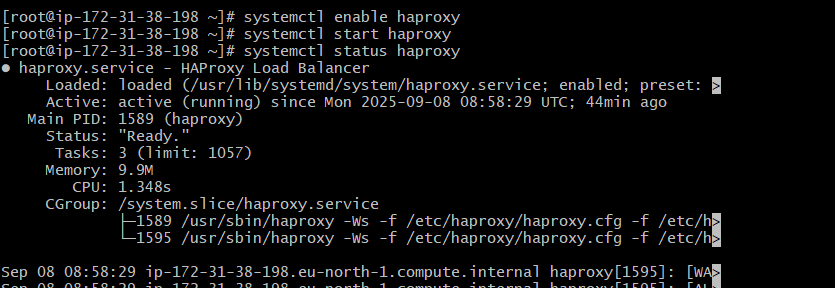
Step 12: Add server 1 and server 2’s public ip at the end of the page in **vi /etc/haproxy/haproxy.cfg** and change port number to **80**

****

****

Step 13: Run following commands

* **systemctl enable haproxy**
* **systemctl start haproxy**
* **systemctl status haproxy**



Now in your browser search with HA-Proxy server’s public ip with port :80 it will distribute load to server-1, server-2.

