1) Install nginx and run nginx on port number 81

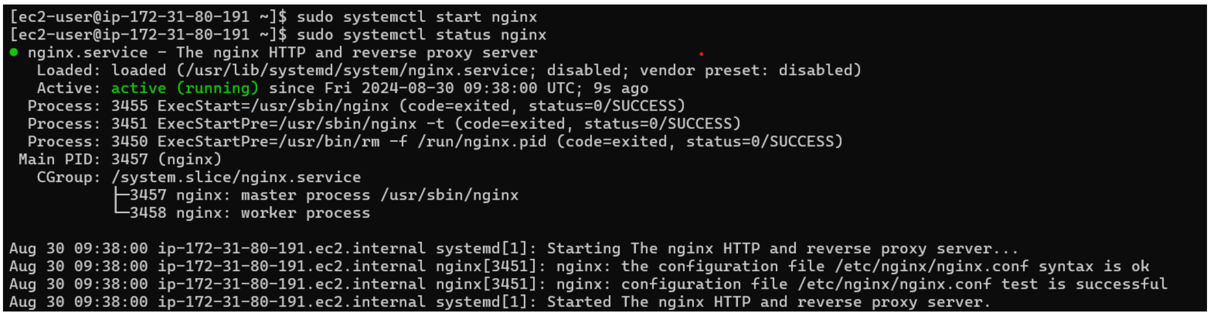
To install Nginx first we need to server .

Steps:

Connect any server and update the system.

To install nginx CMD ----- sudo yum –y install nginx

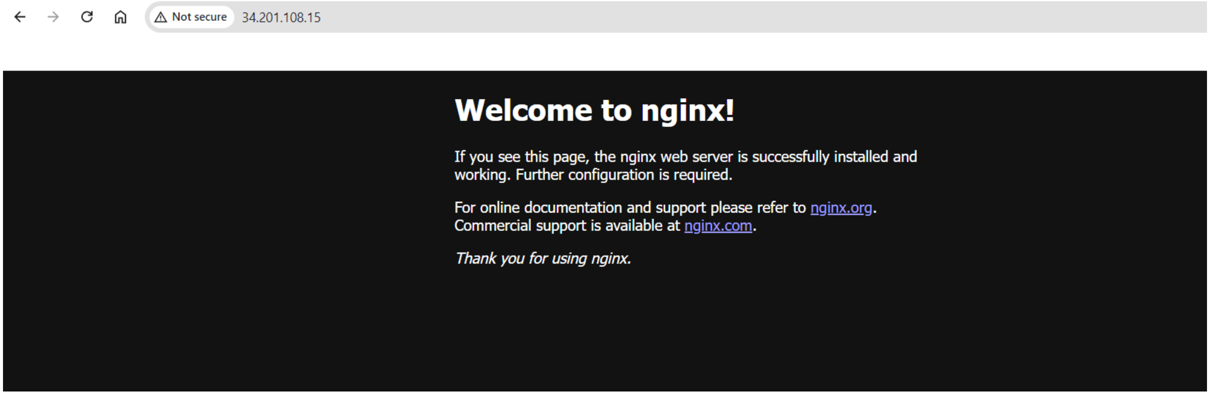
To start service CMD ---- sudo systemctl status



```
[ec2-user@ip-172-31-80-191 ~]$ sudo systemctl start nginx
[ec2-user@ip-172-31-80-191 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; vendor preset: disabled)
   Active: active (running) since Fri 2024-08-30 09:38:00 UTC; 9s ago
  Process: 3455 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
  Process: 3451 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
  Process: 3450 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
 Main PID: 3457 (nginx)
   CGroup: /system.slice/nginx.service
           ├─3457 nginx: master process /usr/sbin/nginx
           └─3458 nginx: worker process

Aug 30 09:38:00 ip-172-31-80-191.ec2.internal systemd[1]: Starting The nginx HTTP and reverse proxy server...
Aug 30 09:38:00 ip-172-31-80-191.ec2.internal nginx[3451]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Aug 30 09:38:00 ip-172-31-80-191.ec2.internal nginx[3451]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Aug 30 09:38:00 ip-172-31-80-191.ec2.internal systemd[1]: Started The nginx HTTP and reverse proxy server.
```

Nginx server is running Now. We can accesses with port 80 check one we are able to see content or Not.

Here we access the nginx page with port 80.



**Welcome to nginx!**

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

*Thank you for using nginx.*

Now I want to  change the port number as 81.

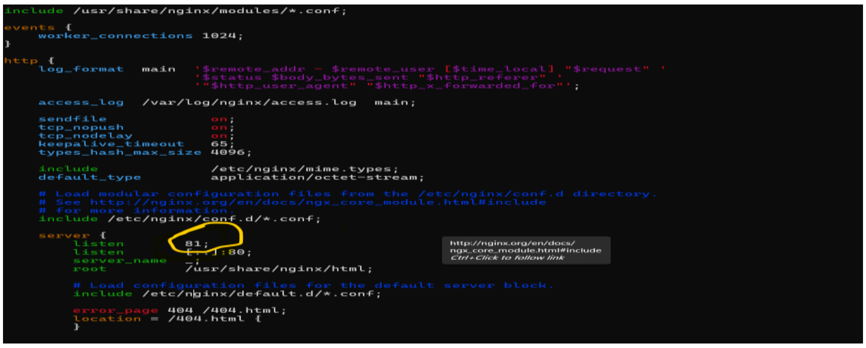
To change port we have to go these path  ----- /etc/nginx/

Stpes:

CMD : cd /etc/nginx

CMD : sudo vi nginx.conf

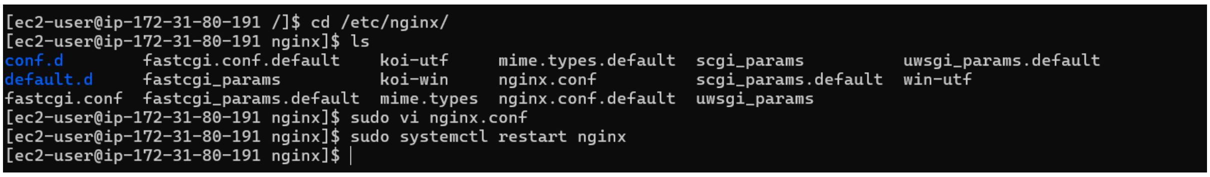
These file you need to change the port number 81



```
include /usr/share/nginx/modules/*.conf;

events {
    worker_connections 1024;
}

http {
    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile            on;
    tcp_nopush          on;
    tcp_nodelay         on;
    keepalive_timeout   65;
    types_hash_max_size 4096;

    include             /etc/nginx/mime.types;
    default_type        application/octet-stream;

    # Load modular configuration files from the /etc/nginx/conf.d directory.
    # See http://nginx.org/en/docs/ngx_core_module.html#include
    # for more information.
    include /etc/nginx/conf.d/*.conf;

    server {
        listen       81;
        listen       [::]:80;
        server_name  _;
        root         /usr/share/nginx/html;

        # Load configuration files for the default server block.
        include /etc/nginx/default.d/*.conf;

        error_page 404 /404.html;
        location = /404.html {
        }
```

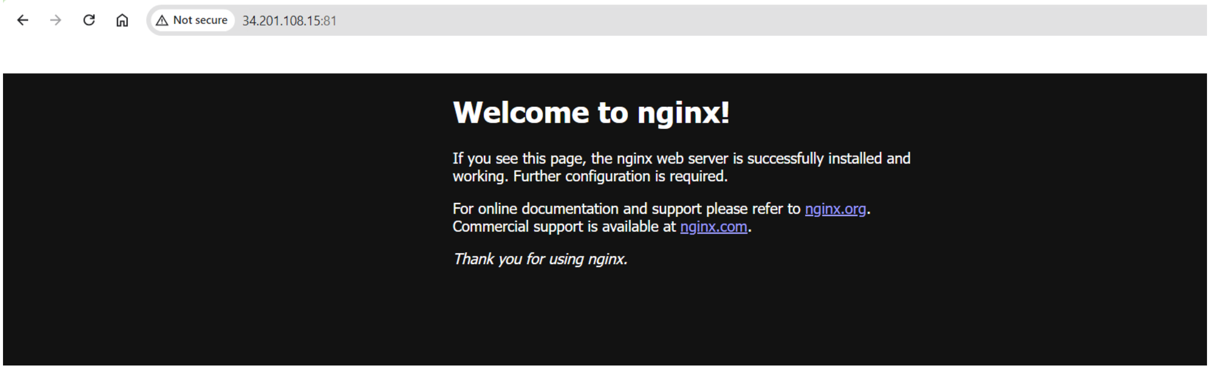
http://nginx.org/en/docs/
ngx_core_module.html#include
Ctrl+Click to follow link

After you need to restart the service .



```
[ec2-user@ip-172-31-80-191 /]$ cd /etc/nginx/
[ec2-user@ip-172-31-80-191 nginx]$ ls
conf.d        fastcgi.conf.default    koi-utf     mime.types.default  scgi_params           uwsgi_params.default
default.d     fastcgi_params          koi-win     nginx.conf          scgi_params.default   win-utf
fastcgi.conf  fastcgi_params.default  mime.types  nginx.conf.default  uwsgi_params
[ec2-user@ip-172-31-80-191 nginx]$ sudo vi nginx.conf
[ec2-user@ip-172-31-80-191 nginx]$ sudo systemctl restart nginx
[ec2-user@ip-172-31-80-191 nginx]$ |
```

You can search in google with port 81.



**Welcome to nginx!**

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

*Thank you for using nginx.*

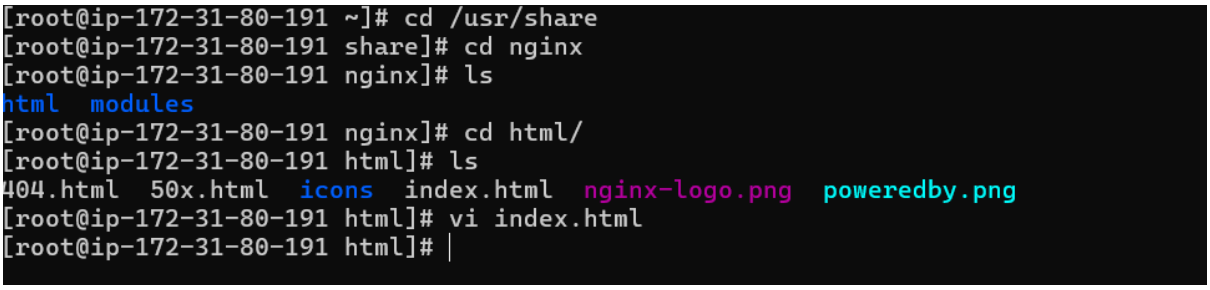
2) Deploy a sample index.html file on nginx

We have to go these directory to deploye sample on nginx

Cd ----- /usr/share/nginx/html

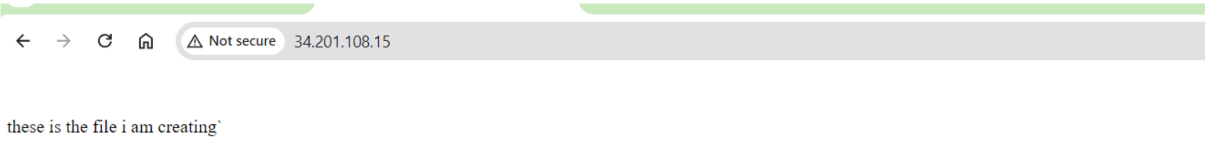
CMD --- vi index.html

Edit the data what you want .



```
[root@ip-172-31-80-191 ~]# cd /usr/share
[root@ip-172-31-80-191 share]# cd nginx
[root@ip-172-31-80-191 nginx]# ls
html  modules
[root@ip-172-31-80-191 nginx]# cd html/
[root@ip-172-31-80-191 html]# ls
404.html  50x.html  icons  index.html  nginx-logo.png  poweredby.png
[root@ip-172-31-80-191 html]# vi index.html
[root@ip-172-31-80-191 html]# |
```

Search in google with Ip address and port number.



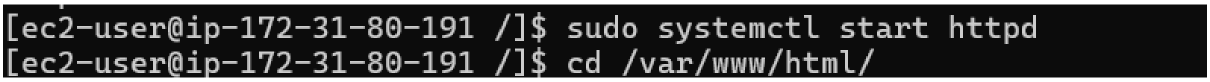
these is the file i am creating`

3) Install Apache and run Apache on port number 82

To install httpd CMD ---- sudo yum –y install httpd

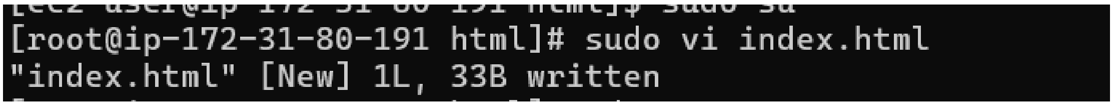
To start httpd CMD---- sudo systemctl start httpd

Go to the below location



```
[ec2-user@ip-172-31-80-191 /]$ sudo systemctl start httpd
[ec2-user@ip-172-31-80-191 /]$ cd /var/www/html/
```

Create a file



```
[root@ip-172-31-80-191 html]# sudo vi index.html
"index.html" [New] 1L, 33B written
```

Now we want to change the port number of the httpd server.

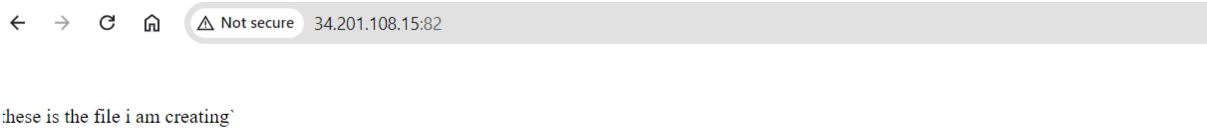
To do that we have to go these dir ------------→ /etc/httpd/conf

Here we edit the file --→ httpd.conf

To edit CMD --- sudo vi httpd.conf

After chnge the port number you need to restart the service .

Now we will see our content ip address:82



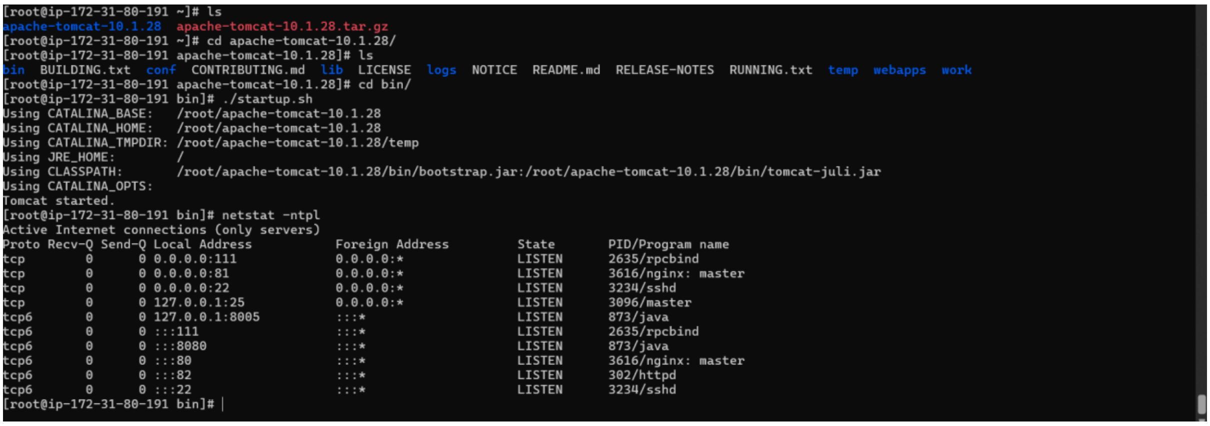
hese is the file i am creating`

4) install Apache tomcat on port number 8082

- To install Apache tomcat use these link the below Link

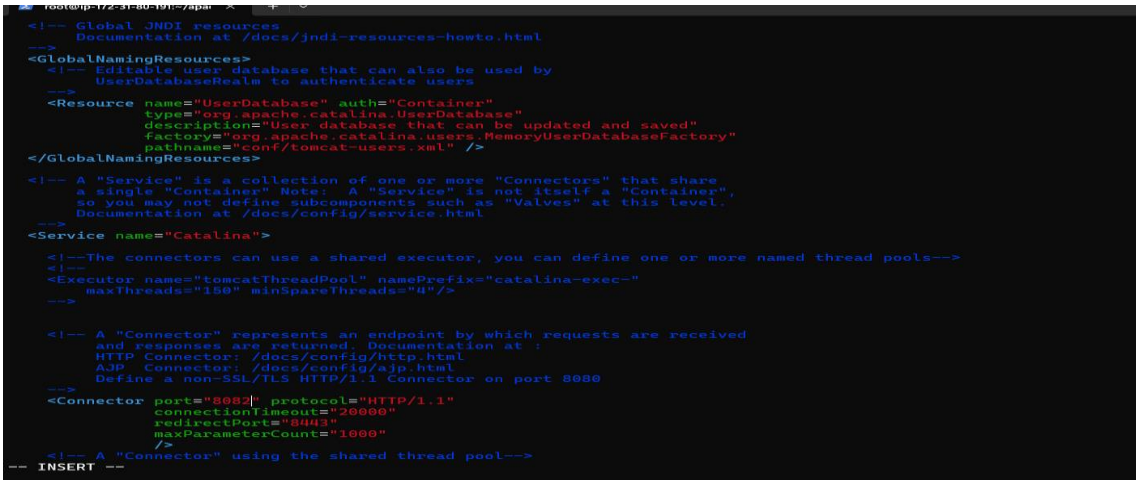
https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz

- you need to download the java also . because it's build by java.
- After that extract the tar file ---→ tar -xvf apache-tomcat-10.1.28.tar.gz
- Go to the bin dir and start the tomcat.



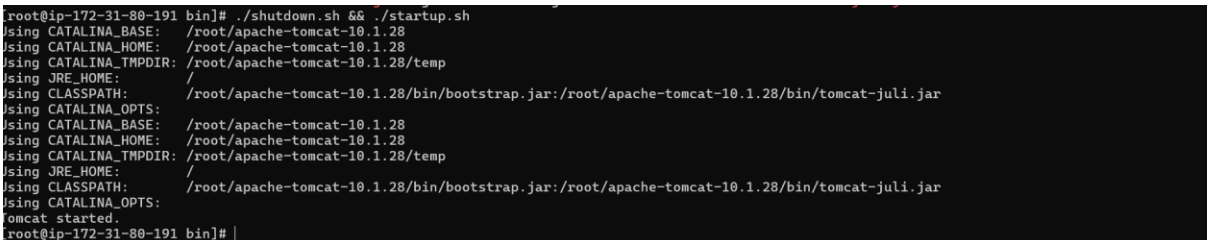
```
[root@ip-172-31-80-191 ~]# ls
apache-tomcat-10.1.28  apache-tomcat-10.1.28.tar.gz
[root@ip-172-31-80-191 ~]# cd apache-tomcat-10.1.28/
[root@ip-172-31-80-191 apache-tomcat-10.1.28]# ls
bin  BUILDING.txt  conf  CONTRIBUTING.md  lib  LICENSE  logs  NOTICE  README.md  RELEASE-NOTES  RUNNING.txt  temp  webapps  work
[root@ip-172-31-80-191 apache-tomcat-10.1.28]# cd bin/
[root@ip-172-31-80-191 bin]# ./startup.sh
Using CATALINA_BASE:   /root/apache-tomcat-10.1.28
Using CATALINA_HOME:   /root/apache-tomcat-10.1.28
Using CATALINA_TMPDIR: /root/apache-tomcat-10.1.28/temp
Using JRE_HOME:        /
Using CLASSPATH:       /root/apache-tomcat-10.1.28/bin/bootstrap.jar:/root/apache-tomcat-10.1.28/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@ip-172-31-80-191 bin]# netstat -ntpl
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      2635/rpcbind
tcp        0      0 0.0.0.0:81              0.0.0.0:*               LISTEN      3616/nginx: master
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      3234/sshd
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      3096/master
tcp6       0      0 127.0.0.1:8005          :::*                    LISTEN      873/java
tcp6       0      0 :::111                  :::*                    LISTEN      2635/rpcbind
tcp6       0      0 :::8080                 :::*                    LISTEN      873/java
tcp6       0      0 :::80                   :::*                    LISTEN      3616/nginx: master
tcp6       0      0 :::82                   :::*                    LISTEN      302/httpd
tcp6       0      0 :::22                   :::*                    LISTEN      3234/sshd
[root@ip-172-31-80-191 bin]# |
```

- Now we want to change the port number 8082.
- To do that we have to go to the CMD------ cd/conf
- There you edit the file server.xml file .so CMD-→ vi server.xml
- Here you need to change the port number 8082.



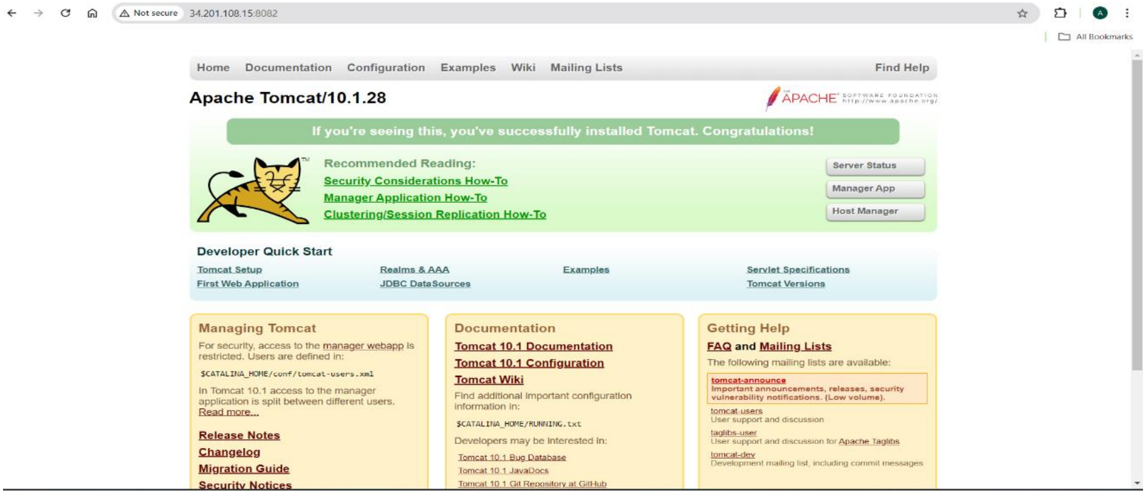
```
  <!-- Global JNDI resources
       Documentation at /docs/jndi-resources-howto.html
  -->
  <GlobalNamingResources>
    <!-- Editable user database that can also be used by
         UserDatabaseRealm to authenticate users
    -->
    <Resource name="UserDatabase" auth="Container"
              type="org.apache.catalina.UserDatabase"
              description="User database that can be updated and saved"
              factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
              pathname="conf/tomcat-users.xml" />
  </GlobalNamingResources>

  <!-- A "Service" is a collection of one or more "Connectors" that share
       a single "Container" Note:  A "Service" is not itself a "Container",
       so you may not define subcomponents such as "Valves" at this level.
       Documentation at /docs/config/service.html
   -->
  <Service name="Catalina">

    <!--The connectors can use a shared executor, you can define one or more named thread pools-->
    <!--
    <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
        maxThreads="150" minSpareThreads="4"/>
    -->


    <!-- A "Connector" represents an endpoint by which requests are received
         and responses are returned. Documentation at :
         HTTP Connector: /docs/config/http.html
         AJP  Connector: /docs/config/ajp.html
         Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
    -->
    <Connector port="8082" protocol="HTTP/1.1"
               connectionTimeout="20000"
               redirectPort="8443"
               maxParameterCount="1000"
               />
    <!-- A "Connector" using the shared thread pool-->
-- INSERT --
```

- Now check it once it will be running on port 8082 or not.
- It will not running because off after changing the port number we haven't restart.
- To restart the the tomcat service like these we have to do.



```
[root@ip-172-31-80-191 bin]# ./shutdown.sh && ./startup.sh
Jsing CATALINA_BASE:   /root/apache-tomcat-10.1.28
Jsing CATALINA_HOME:   /root/apache-tomcat-10.1.28
Jsing CATALINA_TMPDIR: /root/apache-tomcat-10.1.28/temp
Jsing JRE_HOME:        /
Jsing CLASSPATH:       /root/apache-tomcat-10.1.28/bin/bootstrap.jar:/root/apache-tomcat-10.1.28/bin/tomcat-juli.jar
Jsing CATALINA_OPTS:
Jsing CATALINA_BASE:   /root/apache-tomcat-10.1.28
Jsing CATALINA_HOME:   /root/apache-tomcat-10.1.28
Jsing CATALINA_TMPDIR: /root/apache-tomcat-10.1.28/temp
Jsing JRE_HOME:        /
Jsing CLASSPATH:       /root/apache-tomcat-10.1.28/bin/bootstrap.jar:/root/apache-tomcat-10.1.28/bin/tomcat-juli.jar
Jsing CATALINA_OPTS:
omcat started.
[root@ip-172-31-80-191 bin]# |
```

Now the application we can see in browser.



Apache Tomcat/10.1.28

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:
Security Considerations How-To
Manager Application How-To
Clustering/Session Replication How-To

Developer Quick Start

Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specifications
Tomcat Versions

Managing Tomcat

For security, access to the manager webapp is restricted. Users are defined in:

$CATALINA_HOME/conf/tomcat-users.xml

In Tomcat 10.1 access to the manager application is split between different users. Read more...

Release Notes
Changelog
Migration Guide
Security Notices

Documentation

Tomcat 10.1 Documentation
Tomcat 10.1 Configuration
Tomcat Wiki

Find additional important configuration information in:

$CATALINA_HOME/RUNNING.txt

Developers may be interested in:

Tomcat 10.1 Bug Database
Tomcat 10.1 JavaDocs
Tomcat 10.1 Git Repository at GitHub

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).

tomcat-users
User support and discussion

taglibs-user
User support and discussion for Apache Taglibs

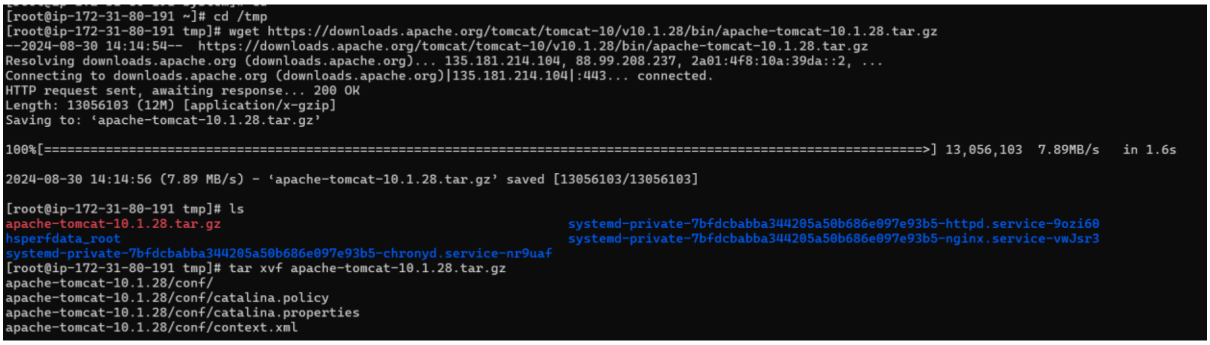
tomcat-dev
Development mailing list, including commit messages

7) Create a tomcat.service file for tomcat.

- Here I am creating the tomcat user
- CMD -→ # Create a Tomcat user and group --→ useradd -r -m tomcat

```
[root@ip-172-31-80-191 ~]# useradd -r -m tomcat
```

- After I went to these dir   CMD -----    cd /tmp
- I am downloaded CMD --→ wget https://downloads.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
- Next I am extract file CMD ----> tar xvf apache-tomcat-10.1.28.tar.gz



```
[root@ip-172-31-80-191 ~]# cd /tmp
[root@ip-172-31-80-191 tmp]# wget https://downloads.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
--2024-08-30 14:14:54--  https://downloads.apache.org/tomcat/tomcat-10/v10.1.28/bin/apache-tomcat-10.1.28.tar.gz
Resolving downloads.apache.org (downloads.apache.org)... 135.181.214.104, 88.99.208.237, 2a01:4f8:10a:39da::2, ...
Connecting to downloads.apache.org (downloads.apache.org)|135.181.214.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13056103 (12M) [application/x-gzip]
Saving to: ‘apache-tomcat-10.1.28.tar.gz’

100%[===================================================================================================>] 13,056,103  7.89MB/s   in 1.6s

2024-08-30 14:14:56 (7.89 MB/s) - ‘apache-tomcat-10.1.28.tar.gz’ saved [13056103/13056103]

[root@ip-172-31-80-191 tmp]# ls
apache-tomcat-10.1.28.tar.gz                                          systemd-private-7bfdcbabba344205a50b686e097e93b5-httpd.service-9ozi60
hsperfdata_root                                                       systemd-private-7bfdcbabba344205a50b686e097e93b5-nginx.service-vwJsr3
systemd-private-7bfdcbabba344205a50b686e097e93b5-chronyd.service-nr9uaf
[root@ip-172-31-80-191 tmp]# tar xvf apache-tomcat-10.1.28.tar.gz
apache-tomcat-10.1.28/conf/
apache-tomcat-10.1.28/conf/catalina.policy
apache-tomcat-10.1.28/conf/catalina.properties
apache-tomcat-10.1.28/conf/context.xml
```

Next I am move these file to  the below location.

CMD ---- mv apache-tomcat-10.1.28 /opt/tomcat

CMD --- chown -R tomcat:tomcat /opt/tomcat

CMD ---- chmod -R 755 /opt/tomcat

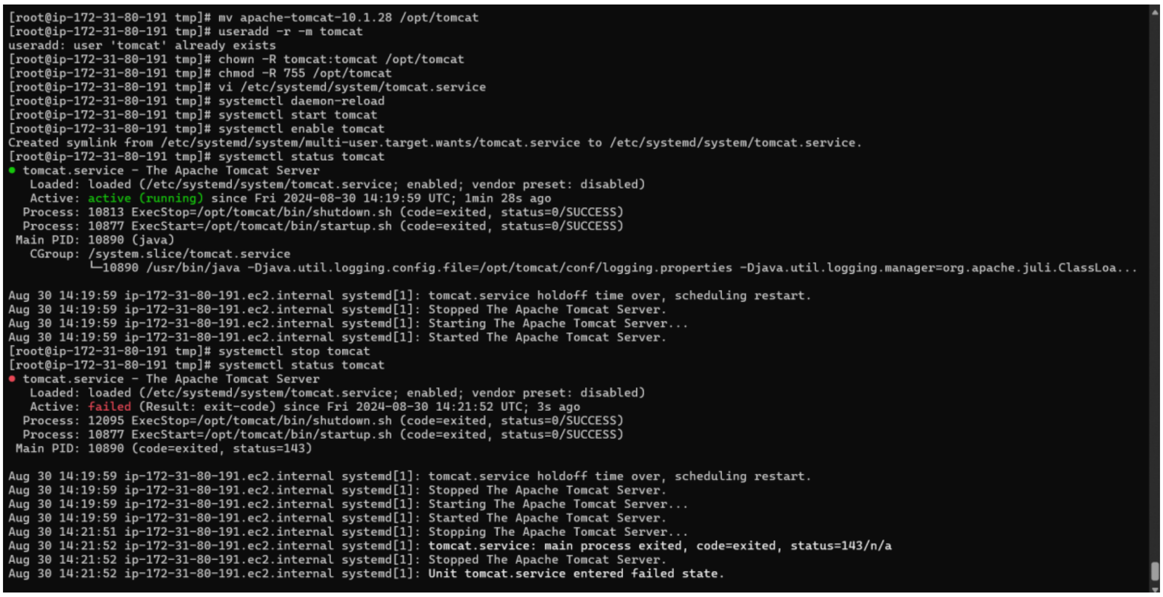
CMD ----- chmod -R 755 /opt/tomcat

After performing the all command I went to a  file and I paste some code .

These is the code :

```
[Unit]

Description=The Apache Tomcat Server

After=network.target


[Service]

Type=forking

User=tomcat

Group=tomcat

Environment="CATALINA_HOME=/opt/tomcat"

ExecStart=/opt/tomcat/bin/startup.sh

ExecStop=/opt/tomcat/bin/shutdown.sh

Restart=always


[Install]

WantedBy=multi-user.target
```

- Then save the file and perform the below commands.
- CMD---- systemctl daemon-reload
- CMD---- systemctl start tomcat
- CMD ---- systemctl enable tomcat
- Using these command we start the tomcat and we stop the tomcat.
- Without going the bin directory and all perform the task simple way to setup the tomcat server.



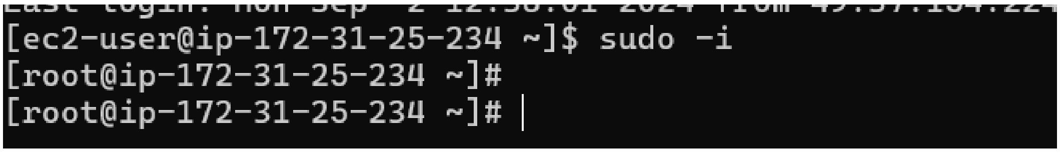
```
[root@ip-172-31-80-191 tmp]# mv apache-tomcat-10.1.28 /opt/tomcat
[root@ip-172-31-80-191 tmp]# useradd -r -m tomcat
useradd: user 'tomcat' already exists
[root@ip-172-31-80-191 tmp]# chown -R tomcat:tomcat /opt/tomcat
[root@ip-172-31-80-191 tmp]# chmod -R 755 /opt/tomcat
[root@ip-172-31-80-191 tmp]# vi /etc/systemd/system/tomcat.service
[root@ip-172-31-80-191 tmp]# systemctl daemon-reload
[root@ip-172-31-80-191 tmp]# systemctl start tomcat
[root@ip-172-31-80-191 tmp]# systemctl enable tomcat
Created symlink from /etc/systemd/system/multi-user.target.wants/tomcat.service to /etc/systemd/system/tomcat.service.
[root@ip-172-31-80-191 tmp]# systemctl status tomcat
● tomcat.service - The Apache Tomcat Server
   Loaded: loaded (/etc/systemd/system/tomcat.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2024-08-30 14:19:59 UTC; 1min 28s ago
  Process: 10813 ExecStop=/opt/tomcat/bin/shutdown.sh (code=exited, status=0/SUCCESS)
  Process: 10877 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
 Main PID: 10890 (java)
   CGroup: /system.slice/tomcat.service
           └─10890 /usr/bin/java -Djava.util.logging.config.file=/opt/tomcat/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoa...

Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: tomcat.service holdoff time over, scheduling restart.
Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: Stopped The Apache Tomcat Server.
Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: Starting The Apache Tomcat Server...
Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: Started The Apache Tomcat Server.
[root@ip-172-31-80-191 tmp]# systemctl stop tomcat
[root@ip-172-31-80-191 tmp]# systemctl status tomcat
● tomcat.service - The Apache Tomcat Server
   Loaded: loaded (/etc/systemd/system/tomcat.service; enabled; vendor preset: disabled)
   Active: failed (Result: exit-code) since Fri 2024-08-30 14:21:52 UTC; 3s ago
  Process: 12095 ExecStop=/opt/tomcat/bin/shutdown.sh (code=exited, status=0/SUCCESS)
  Process: 10877 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
 Main PID: 10890 (code=exited, status=143)

Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: tomcat.service holdoff time over, scheduling restart.
Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: Stopped The Apache Tomcat Server.
Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: Starting The Apache Tomcat Server...
Aug 30 14:19:59 ip-172-31-80-191.ec2.internal systemd[1]: Started The Apache Tomcat Server.
Aug 30 14:21:51 ip-172-31-80-191.ec2.internal systemd[1]: Stopping The Apache Tomcat Server...
Aug 30 14:21:52 ip-172-31-80-191.ec2.internal systemd[1]: tomcat.service: main process exited, code=exited, status=143/n/a
Aug 30 14:21:52 ip-172-31-80-191.ec2.internal systemd[1]: Stopped The Apache Tomcat Server.
Aug 30 14:21:52 ip-172-31-80-191.ec2.internal systemd[1]: Unit tomcat.service entered failed state.
```

8) Configure HA Proxy server:

HAProxy is a powerful, open-source tool that acts as a load balancer, distributing incoming network traffic across multiple servers to ensure no single server is overwhelmed. This helps maintain high availability, reliability, and performance for web applications, making it essential for handling large-scale traffic efficiently.

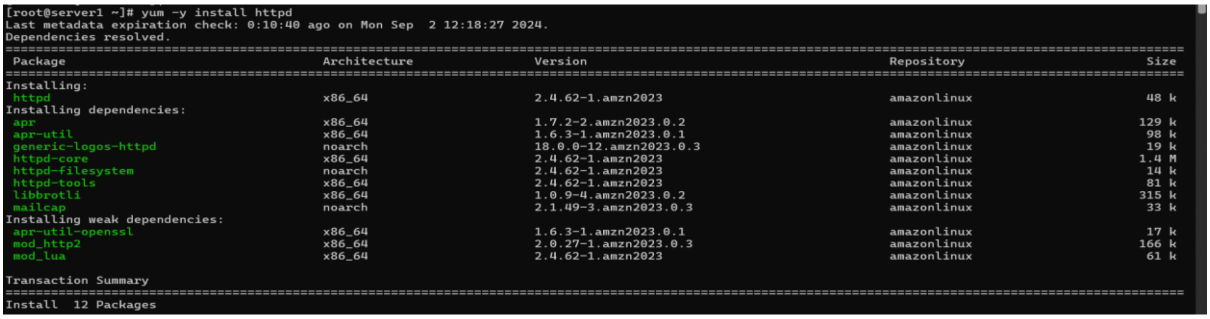
**Steps for Lab:**

- Setup 3 servers: -----→ Server_1, Server_2, HAproxy
- Connect to the Server_1
- Perform the below commands.
- sudo –i ---- To change the root.



```
[ec2-user@ip-172-31-25-234 ~]$ sudo -i
[root@ip-172-31-25-234 ~]#
[root@ip-172-31-25-234 ~]# |
```

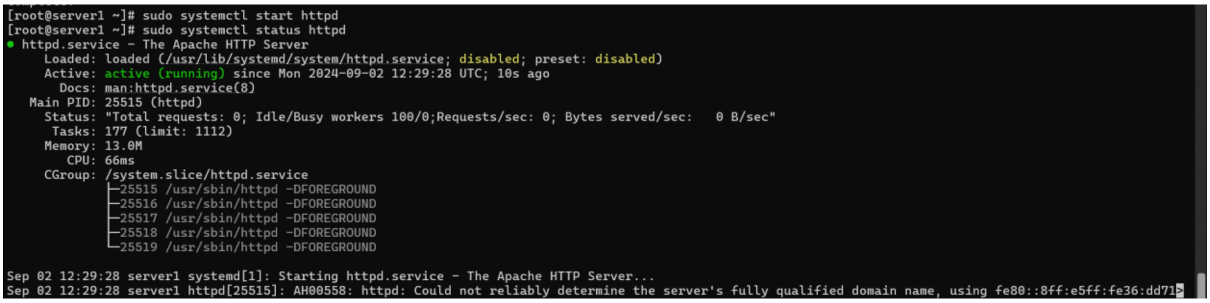
- Yum –y install httpd --- To install the httpd sever.



```
[root@server1 ~]# yum -y install httpd
Last metadata expiration check: 0:10:40 ago on Mon Sep  2 12:18:27 2024.
Dependencies resolved.
================================================================================================================================================
 Package                              Architecture               Version                                     Repository                   Size
================================================================================================================================================
Installing:
 httpd                                x86_64                     2.4.62-1.amzn2023                           amazonlinux                  48 k
Installing dependencies:
 apr                                  x86_64                     1.7.2-2.amzn2023.0.2                        amazonlinux                 129 k
 apr-util                             x86_64                     1.6.3-1.amzn2023.0.1                        amazonlinux                  98 k
 generic-logos-httpd                  noarch                     18.0.0-12.amzn2023.0.3                      amazonlinux                  19 k
 httpd-core                           x86_64                     2.4.62-1.amzn2023                           amazonlinux                 1.4 M
 httpd-filesystem                     noarch                     2.4.62-1.amzn2023                           amazonlinux                  14 k
 httpd-tools                          x86_64                     2.4.62-1.amzn2023                           amazonlinux                  81 k
 libbrotli                            x86_64                     1.0.9-4.amzn2023.0.2                        amazonlinux                 315 k
 mailcap                              noarch                     2.1.49-3.amzn2023.0.3                       amazonlinux                  33 k
Installing weak dependencies:
 apr-util-openssl                     x86_64                     1.6.3-1.amzn2023.0.1                        amazonlinux                  17 k
 mod_http2                            x86_64                     2.0.27-1.amzn2023.0.3                       amazonlinux                 166 k
 mod_lua                              x86_64                     2.4.62-1.amzn2023                           amazonlinux                  61 k

Transaction Summary
================================================================================================================================================
Install  12 Packages
```

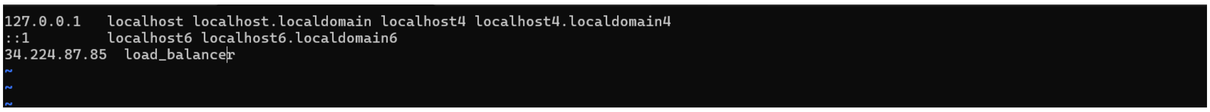
systemctl start httpd----- to start the httpd sever.

systemctl status httpd---- to check the status.



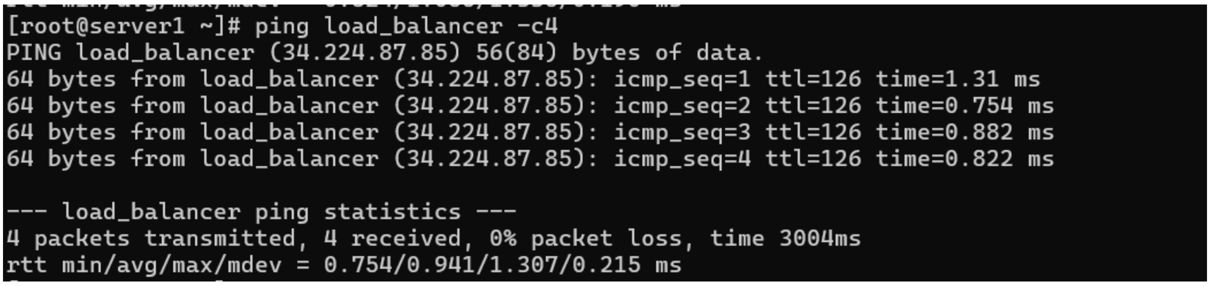
```
[root@server1 ~]# sudo systemctl start httpd
[root@server1 ~]# sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
     Active: active (running) since Mon 2024-09-02 12:29:28 UTC; 10s ago
       Docs: man:httpd.service(8)
   Main PID: 25515 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
      Tasks: 177 (limit: 1112)
     Memory: 13.0M
        CPU: 66ms
     CGroup: /system.slice/httpd.service
             ├─25515 /usr/sbin/httpd -DFOREGROUND
             ├─25516 /usr/sbin/httpd -DFOREGROUND
             ├─25517 /usr/sbin/httpd -DFOREGROUND
             ├─25518 /usr/sbin/httpd -DFOREGROUND
             └─25519 /usr/sbin/httpd -DFOREGROUND

Sep 02 12:29:28 server1 systemd[1]: Starting httpd.service - The Apache HTTP Server...
Sep 02 12:29:28 server1 httpd[25515]: AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using fe80::8ff:e5ff:fe36:dd71>
```

vi /etc/hosts ----- Go to these this location and update the ip address of the haproxy and also give tag.



```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost6 localhost6.localdomain6
34.224.87.85  load_balancer
~
~
~
```

Enter the command ---- ping load_balancer -c4

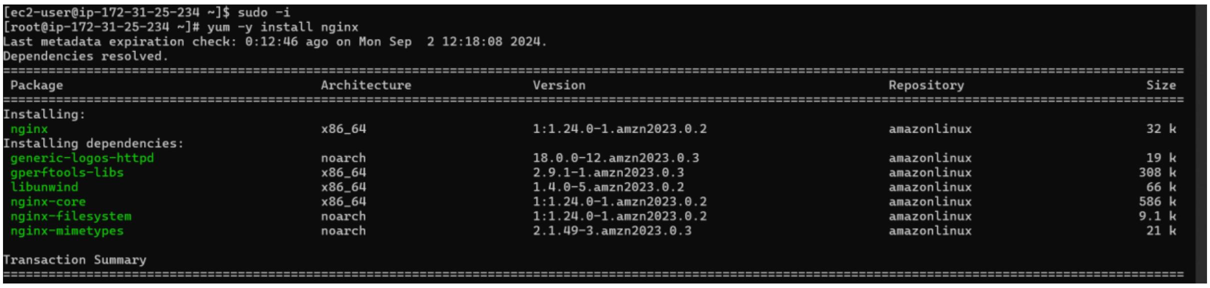


```
[root@server1 ~]# ping load_balancer -c4
PING load_balancer (34.224.87.85) 56(84) bytes of data.
64 bytes from load_balancer (34.224.87.85): icmp_seq=1 ttl=126 time=1.31 ms
64 bytes from load_balancer (34.224.87.85): icmp_seq=2 ttl=126 time=0.754 ms
64 bytes from load_balancer (34.224.87.85): icmp_seq=3 ttl=126 time=0.882 ms
64 bytes from load_balancer (34.224.87.85): icmp_seq=4 ttl=126 time=0.822 ms

--- load_balancer ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.754/0.941/1.307/0.215 ms
```

Next step is Go to the second server.

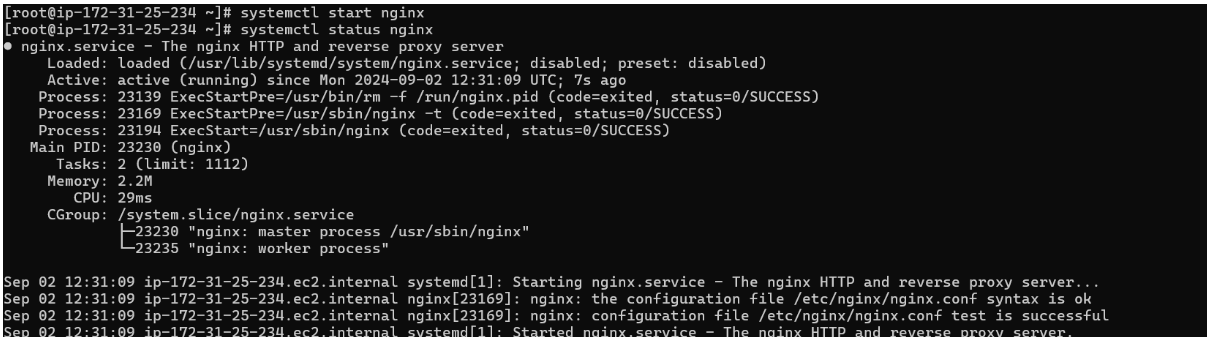
Steps:

- sudo –I
- Install nginx server ----yum –y install nginx



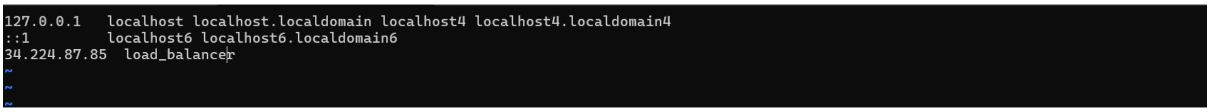
```
[ec2-user@ip-172-31-25-234 ~]$ sudo -i
[root@ip-172-31-25-234 ~]# yum -y install nginx
Last metadata expiration check: 0:12:46 ago on Mon Sep  2 12:18:08 2024.
Dependencies resolved.
================================================================================================================================
 Package                           Architecture              Version                                  Repository          Size
================================================================================================================================
Installing:
 nginx                             x86_64                    1:1.24.0-1.amzn2023.0.2                  amazonlinux         32 k
Installing dependencies:
 generic-logos-httpd               noarch                    18.0.0-12.amzn2023.0.3                   amazonlinux         19 k
 gperftools-libs                   x86_64                    2.9.1-1.amzn2023.0.3                     amazonlinux        308 k
 libunwind                         x86_64                    1.4.0-5.amzn2023.0.2                     amazonlinux         66 k
 nginx-core                        x86_64                    1:1.24.0-1.amzn2023.0.2                  amazonlinux        586 k
 nginx-filesystem                  noarch                    1:1.24.0-1.amzn2023.0.2                  amazonlinux        9.1 k
 nginx-mimetypes                   noarch                    2.1.49-3.amzn2023.0.3                    amazonlinux         21 k

Transaction Summary
================================================================================================================================
```

- systemctl start nginx----- to start the nginx sever.
- systemctl status nginx ---- to check the status.



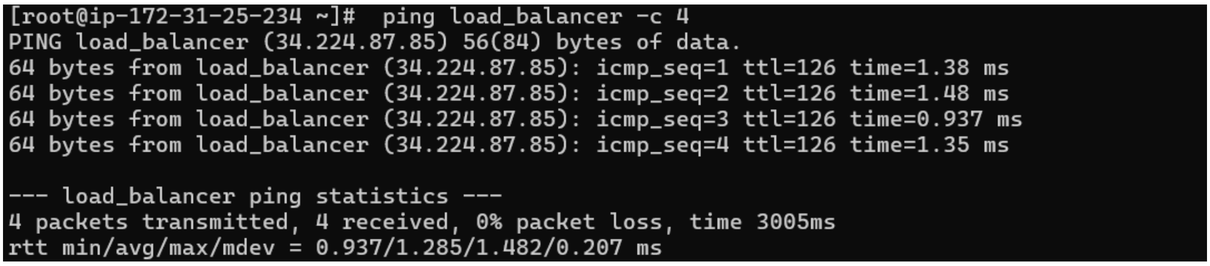
```
[root@ip-172-31-25-234 ~]# systemctl start nginx
[root@ip-172-31-25-234 ~]# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; preset: disabled)
     Active: active (running) since Mon 2024-09-02 12:31:09 UTC; 7s ago
    Process: 23139 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 23169 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 23194 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 23230 (nginx)
      Tasks: 2 (limit: 1112)
     Memory: 2.2M
        CPU: 29ms
     CGroup: /system.slice/nginx.service
             ├─23230 "nginx: master process /usr/sbin/nginx"
             └─23235 "nginx: worker process"

Sep 02 12:31:09 ip-172-31-25-234.ec2.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Sep 02 12:31:09 ip-172-31-25-234.ec2.internal nginx[23169]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Sep 02 12:31:09 ip-172-31-25-234.ec2.internal nginx[23169]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Sep 02 12:31:09 ip-172-31-25-234.ec2.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
```

vi /etc/hosts ----- Go to these this location and update the ip address of the haproxy and also give tag.



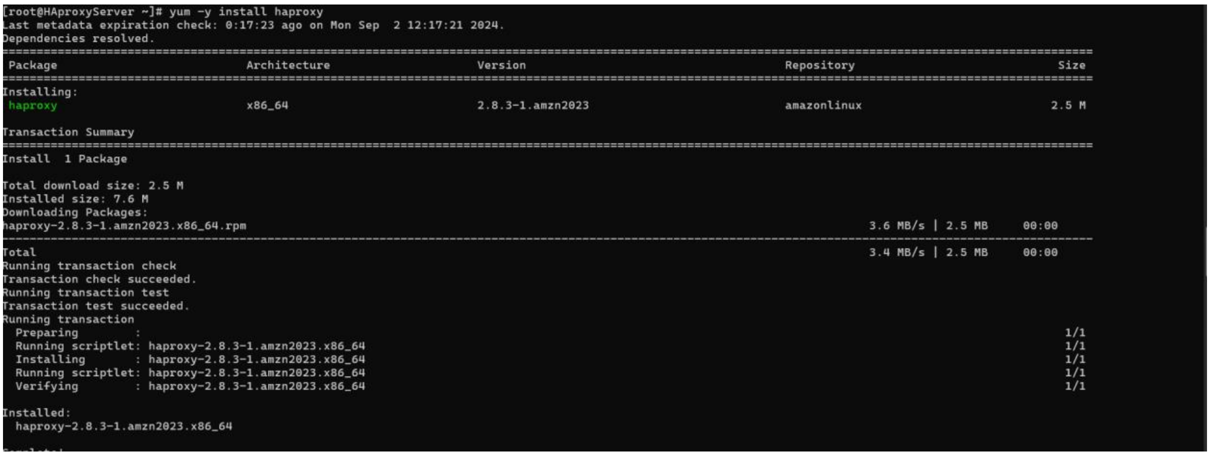
```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost6 localhost6.localdomain6
34.224.87.85  load_balancer
~
~
~
```

Enter the command ---- ping load_balancer -c4



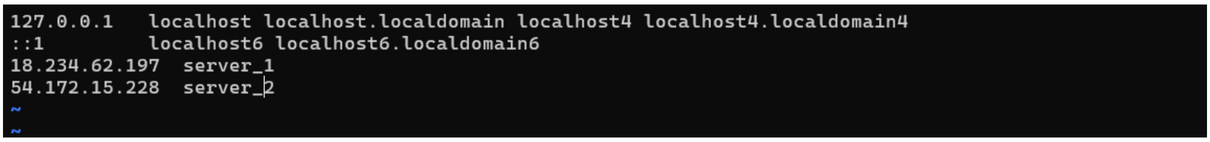
```
[root@ip-172-31-25-234 ~]#  ping load_balancer -c 4
PING load_balancer (34.224.87.85) 56(84) bytes of data.
64 bytes from load_balancer (34.224.87.85): icmp_seq=1 ttl=126 time=1.38 ms
64 bytes from load_balancer (34.224.87.85): icmp_seq=2 ttl=126 time=1.48 ms
64 bytes from load_balancer (34.224.87.85): icmp_seq=3 ttl=126 time=0.937 ms
64 bytes from load_balancer (34.224.87.85): icmp_seq=4 ttl=126 time=1.35 ms

--- load_balancer ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.937/1.285/1.482/0.207 ms
```

- Now GO to the HAproxy server.
- Install haproxy server------- yum –y install haproxy



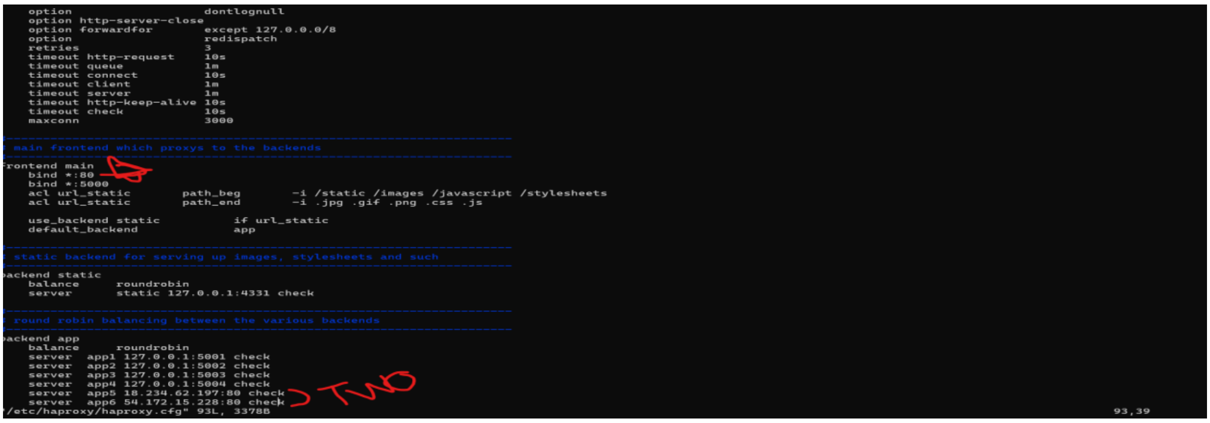
```
[root@HAproxyServer ~]# yum -y install haproxy
Last metadata expiration check: 0:17:23 ago on Mon Sep  2 12:17:21 2024.
Dependencies resolved.
================================================================================================================================================
 Package                          Architecture                  Version                                Repository                          Size
================================================================================================================================================
Installing:
 haproxy                          x86_64                        2.8.3-1.amzn2023                       amazonlinux                        2.5 M

Transaction Summary
================================================================================================================================================
Install  1 Package

Total download size: 2.5 M
Installed size: 7.6 M
Downloading Packages:
haproxy-2.8.3-1.amzn2023.x86_64.rpm                                                                         3.6 MB/s | 2.5 MB     00:00
------------------------------------------------------------------------------------------------------------------------------------------------
Total                                                                                                       3.4 MB/s | 2.5 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                                                         1/1
  Running scriptlet: haproxy-2.8.3-1.amzn2023.x86_64                                                                                         1/1
  Installing       : haproxy-2.8.3-1.amzn2023.x86_64                                                                                         1/1
  Running scriptlet: haproxy-2.8.3-1.amzn2023.x86_64                                                                                         1/1
  Verifying        : haproxy-2.8.3-1.amzn2023.x86_64                                                                                         1/1

Installed:
  haproxy-2.8.3-1.amzn2023.x86_64
```

Go to the vi /etc/hosts and give the sever_1 sever_2 ip addresses.



```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost6 localhost6.localdomain6
18.234.62.197  server_1
54.172.15.228  server_2
~
~
```

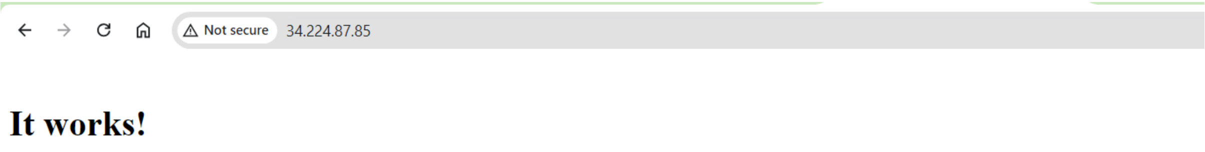
- GO to ---- vi /etc/haproxy/haproxy.cfg
- Add Server-1, Server-2 public IP's



```
    option                  dontlognull
    option http-server-close
    option forwardfor       except 127.0.0.0/8
    option                  redispatch
    retries                 3
    timeout http-request    10s
    timeout queue           1m
    timeout connect         10s
    timeout client          1m
    timeout server          1m
    timeout http-keep-alive 10s
    timeout check           10s
    maxconn                 3000

#---------------------------------------------------------------------
# main frontend which proxys to the backends
#---------------------------------------------------------------------
frontend main
    bind *:80
    bind *:5000
    acl url_static       path_beg       -i /static /images /javascript /stylesheets
    acl url_static       path_end       -i .jpg .gif .png .css .js

    use_backend static          if url_static
    default_backend             app

#---------------------------------------------------------------------
# static backend for serving up images, stylesheets and such
#---------------------------------------------------------------------
backend static
    balance     roundrobin
    server      static 127.0.0.1:4331 check

#---------------------------------------------------------------------
# round robin balancing between the various backends
#---------------------------------------------------------------------
backend app
    balance     roundrobin
    server  app1 127.0.0.1:5001 check
    server  app2 127.0.0.1:5002 check
    server  app3 127.0.0.1:5003 check
    server  app4 127.0.0.1:5004 check
    server  app5 18.234.62.197:80 check
    server  app6 54.172.15.228:80 check
"/etc/haproxy/haproxy.cfg" 93L, 3378B                                                                93,39
```

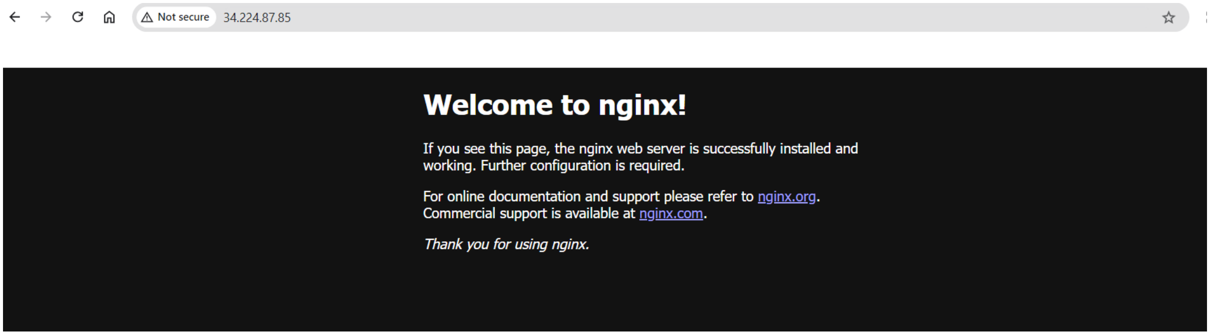
TWO

- now browse with HA-Proxy-Server PublicIP:80 it will distribute load to Server-1, Server-2.
- The below one is httpd server.



# It works!

- Again refresh you will see the nginx server.



# Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

*Thank you for using nginx.*