

## 1. INSTALL NGINX AND RUN NGINX ON PORT NUMBER 81.

```
[root@ip-172-31-33-4 ~]# yum install nginx
Last metadata expiration check: 3:48:02 ago on Tue Sep 2 10:52:24 2025.
Dependencies resolved.
=====
Package                        Architecture      Version           Repository        Size
-----
Installing:
nginx                         x86_64            1:1.28.0-1.amzn2023.0.2    amazonlinux        33 k
Installing dependencies:
gperf-tools-libs             x86_64            2.9.1-1.amzn2023.0.3      amazonlinux        308 k
liburwind                    x86_64            1.4.0-5.amzn2023.0.2      amazonlinux         66 k
nginx-core                    x86_64            1:1.28.0-1.amzn2023.0.2    amazonlinux        686 k
nginx-filesystem              noarch            1:1.28.0-1.amzn2023.0.2    amazonlinux         9.6 k
nginx-mimetypes                noarch            2.1.49-3.amzn2023.0.3      amazonlinux         21 k
Transaction Summary
-----
Install 6 Packages

Total download size: 1.1 M
Installed size: 3.7 M
Is this ok [y/N]: y
Downloading Packages:
(1/6): liburwind-1.4.0-5.amzn2023.0.2.x86_64.rpm                1.8 MB/s | 66 kB | 00:00
(2/6): nginx-1.28.0-1.amzn2023.0.2.x86_64.rpm                  850 kB/s | 33 kB | 00:00
(3/6): gperf-tools-libs-2.9.1-1.amzn2023.0.3.x86_64.rpm        6.3 MB/s | 308 kB | 00:00
(4/6): nginx-filesystem-1.28.0-1.amzn2023.0.2.noarch.rpm        492 kB/s | 9.6 kB | 00:00
(5/6): nginx-core-1.28.0-1.amzn2023.0.2.x86_64.rpm             20 MB/s | 686 kB | 00:00
(6/6): nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch.rpm         860 kB/s | 21 kB | 00:00
-----
Total                                                                9.9 MB/s | 1.1 MB | 00:00
Running transaction check
```

- install nginx service in linux machine : yum install nginx

```
[root@ip-172-31-33-4 ~]# systemctl start nginx
[root@ip-172-31-33-4 ~]# 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 Tue 2025-09-02 14:49:54 UTC; 3s ago
     Process: 38058 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
     Process: 38059 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
     Process: 38060 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
    Main PID: 38061 (nginx)
      Tasks: 3 (limit: 1057)
     Memory: 3.2M
        CPU: 59ms
    CGroup: /system.slice/nginx.service
            └─38061 "nginx: master process /usr/sbin/nginx"
              └─38062 "nginx: worker process"
                └─38063 "nginx: worker process"

Sep 02 14:49:54 ip-172-31-33-4.eu-north-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Sep 02 14:49:54 ip-172-31-33-4.eu-north-1.compute.internal nginx[38059]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Sep 02 14:49:54 ip-172-31-33-4.eu-north-1.compute.internal nginx[38059]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Sep 02 14:49:54 ip-172-31-33-4.eu-north-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[root@ip-172-31-33-4 ~]#
```

- start the nginx service : systemctl start nginx

```
[root@ip-172-31-33-4 ~]# find / -name nginx.conf
/etc/nginx/nginx.conf
/root/nginx.conf
[root@ip-172-31-33-4 ~]# ^C
[root@ip-172-31-33-4 ~]# vi nginx.conf
[root@ip-172-31-33-4 ~]# cd /etc/nginx/
[root@ip-172-31-33-4 nginx]# vi nginx.conf
```

- find the file : file / -name nginx.conf
- go to the directory : cd /etc/nginx/
- view the file contents : vi nginx.conf

```

root@ip-172-31-33-4:/etc/nginx
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log notice;
pid /run/nginx.pid;

# Load dynamic modules. See /usr/share/doc/nginx/README.dynamic.
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;
    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/nginx_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;
    }
}
-- INSERT --

```

- in nginx.conf change : LISTEN 80 to 81 by using ‘i’ – insert
- Enter : esc :wq! – to save and exit the file

```

[root@ip-172-31-33-4 nginx]# systemctl restart nginx
[root@ip-172-31-33-4 nginx]# 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 Tue 2025-09-02 15:02:43 UTC; 16s ago
     Process: 38445 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
     Process: 38447 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
     Process: 38448 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
    Main PID: 38449 (nginx)
       Tasks: 3 (limit: 1057)
      Memory: 3.2M
         CPU: 58ms
    CGroup: /system.slice/nginx.service
            └─38449 "nginx: master process /usr/sbin/nginx"
              └─38450 "nginx: worker process"
                └─38451 "nginx: worker process"

```

- Restart the service : systemctl restart nginx

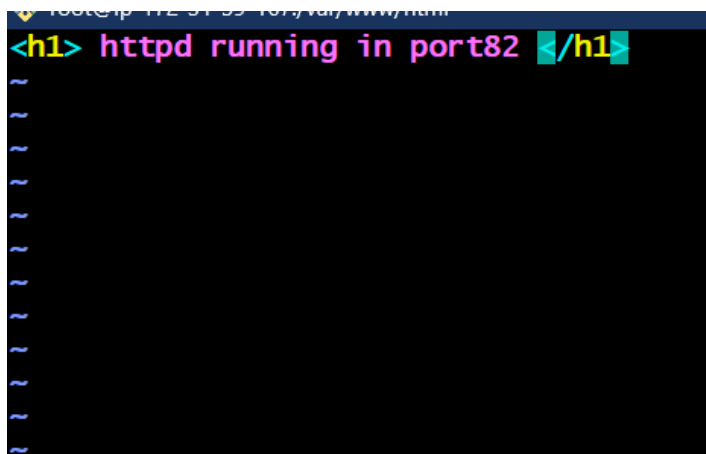


- Now got o web and enter the – “ipv 4 publicaddress: 81” (note : you can get the ip v4 adress in aws instance that you have launched before)
- You can see the nginx is running in port 81.

## 2. .DEPLOYING THE SAMPLE HTML FILE IN NGINX PORT 82

```
[root@ip-172-31-39-167 ~]# find / -name nginx.conf
/etc/nginx/nginx.conf
[root@ip-172-31-39-167 ~]# cd /etc/nginx/
[root@ip-172-31-39-167 nginx]# vi nginx.conf
[root@ip-172-31-39-167 nginx]# cd ~
[root@ip-172-31-39-167 ~]# cd /var/www/html
[root@ip-172-31-39-167 html]# vi index.html
```

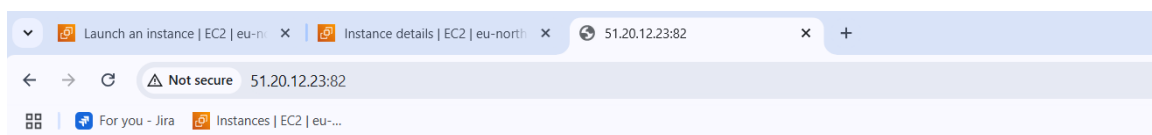
- Go to the html directory : `cd /var/www/html/`
- View the file : `vi index.html`



- Enter the text or content you want to view on web page .
- Save by entering esc button and enter :wq! – to save and exit

```
[root@ip-172-31-39-167 html]# vi index.html
[root@ip-172-31-39-167 html]# systemctl restart nginx
[root@ip-172-31-39-167 html]#
```

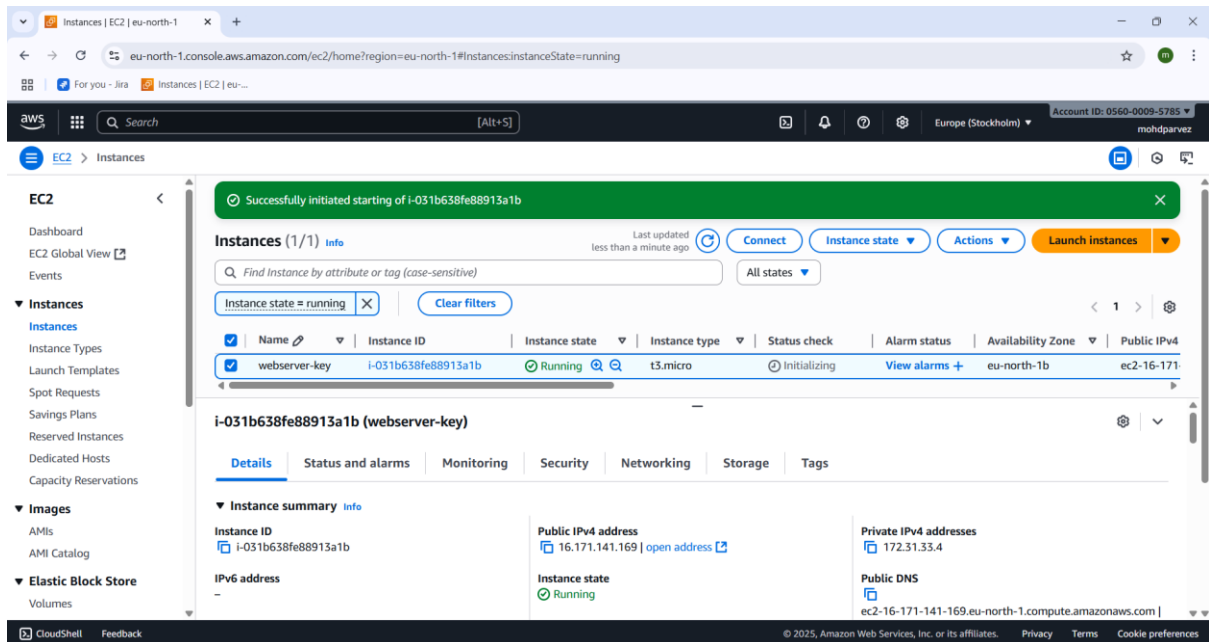
- Restart the service – `systemctl restart nginx`



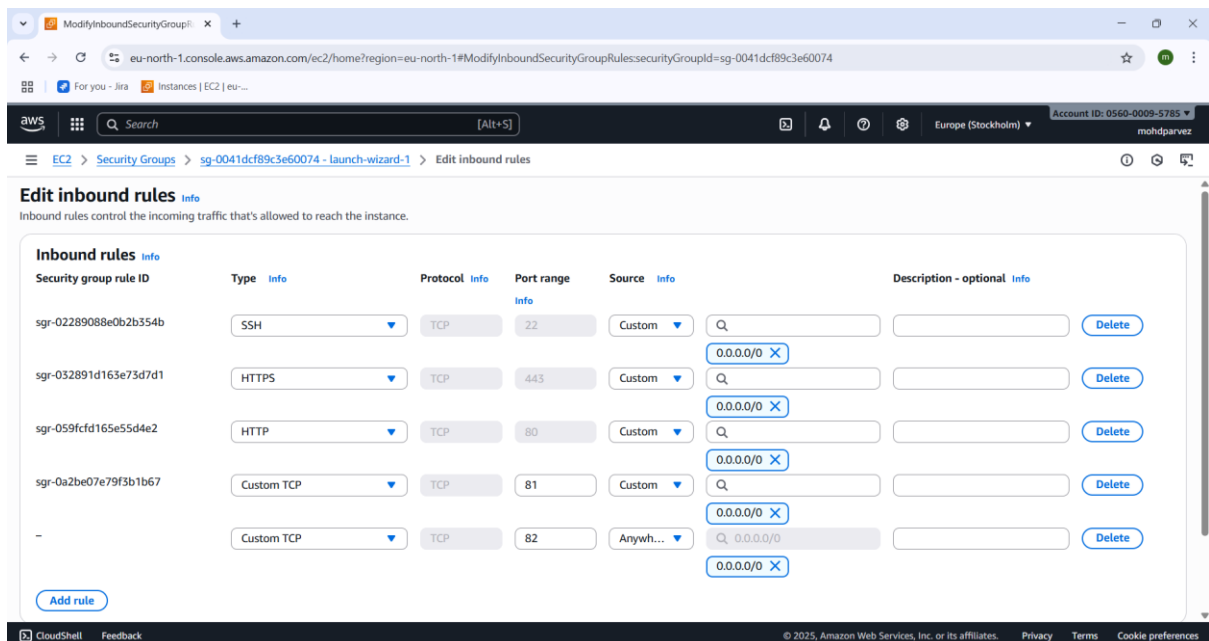
**httpd running in port82**

- Got to the web BROWSER and enter “ipv 4 publicaddress: 81”
- You see the deployed html file is reflecting in the web page.

### 3. HOW TO INSTALL AND RUN HTTPD IN PORT 82

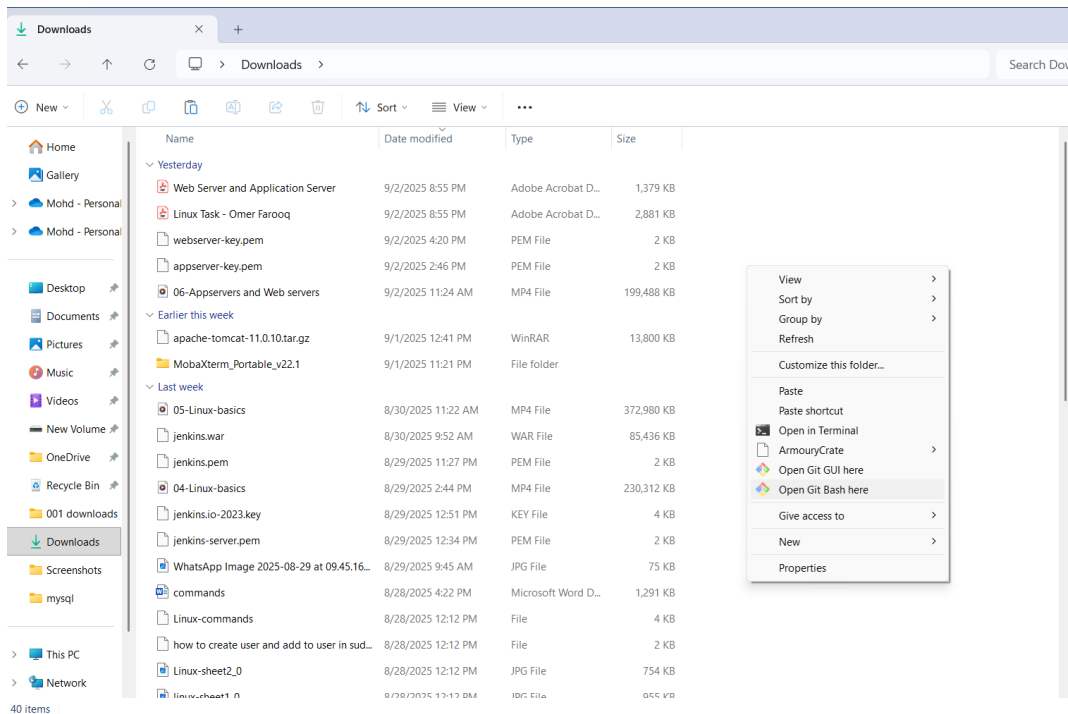


→ Launch aws ec2 instance .



→ Go to security groups and click on edit inbound rules

→ Add security group with custom tcp and give port no 82 and save.



→ Go to the location where keypair.pem is downloaded and right click and select open git bash here.

```

ec2-user@ip-172-31-33-4:~
Mohd Parvez@LAPTOP-NT4KDSHS MINGW64 /e/downloads
$ ssh -i webserver-key.pem ec2-user@16.171.141.169
bash: ssh: command not found

Mohd Parvez@LAPTOP-NT4KDSHS MINGW64 /e/downloads
$ ssh -i web
Web Server and Application Server.pdf webserver-key.pem

Mohd Parvez@LAPTOP-NT4KDSHS MINGW64 /e/downloads
$ ssh -i webserver-key.pem ec2-user@16.171.141.169
The authenticity of host '16.171.141.169 (16.171.141.169)' can't be established.
ED25519 key fingerprint is SHA256:+5JcZDJAohcxB7UcbHJLUU/fRQ04ShZugUBTUn7Nncw.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:10: 16.171.148.159
  ~/.ssh/known_hosts:12: 51.21.196.65
  ~/.ssh/known_hosts:13: 13.60.83.249
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '16.171.141.169' (ED25519) to the list of known hosts
.
#_
~\_####_
~\_#####\
~\_###|

Amazon Linux 2023

```

→ Connect to the remote machine by using : ssh -i “keypair.pem” ec2-user@ “ip\_v4\_public address”.





```
root@ip-172-31-33-4:~  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: d  
   Active: active (running) since Wed 2025-09-03 10:43:32 UTC; 3s ago  
     Docs: man:httpd.service(8)  
  Main PID: 2467 (httpd)  
    Status: "Started, listening on: port 80"  
    Tasks: 177 (limit: 1057)  
   Memory: 18.0M  
      CPU: 67ms  
   CGroup: /system.slice/httpd.service  
           └─2467 /usr/sbin/httpd -DFOREGROUND  
             └─2468 /usr/sbin/httpd -DFOREGROUND  
               └─2469 /usr/sbin/httpd -DFOREGROUND  
                 └─2472 /usr/sbin/httpd -DFOREGROUND  
                   └─2490 /usr/sbin/httpd -DFOREGROUND  
  
Sep 03 10:43:32 ip-172-31-33-4.eu-north-1.compute.internal systemd[1]: Starting  
Sep 03 10:43:32 ip-172-31-33-4.eu-north-1.compute.internal systemd[1]: Started  
Sep 03 10:43:32 ip-172-31-33-4.eu-north-1.compute.internal httpd[2467]: Server  
[root@ip-172-31-33-4 ~]# find / -name httpd.conf  
/etc/httpd/conf/httpd.conf  
/usr/lib/tmpfiles.d/httpd.conf  
/usr/lib/sysusers.d/httpd.conf  
[root@ip-172-31-33-4 ~]#
```

- Find the configuration file using : `find / -name httpd.conf`
- Go to the directory where the httpd. Conf file is located using `cd` command

```
root@ip-172-31-33-4:/usr/lib/tmpfiles.d  
Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: d  
Active: active (running) since Wed 2025-09-03 10:43:32 UTC; 3s ago  
   Docs: man:httpd.service(8)  
 Main PID: 2467 (httpd)  
  Status: "Started, listening on: port 80"  
  Tasks: 177 (limit: 1057)  
 Memory: 18.0M  
    CPU: 67ms  
   CGroup: /system.slice/httpd.service  
           └─2467 /usr/sbin/httpd -DFOREGROUND  
             └─2468 /usr/sbin/httpd -DFOREGROUND  
               └─2469 /usr/sbin/httpd -DFOREGROUND  
                 └─2472 /usr/sbin/httpd -DFOREGROUND  
                   └─2490 /usr/sbin/httpd -DFOREGROUND  
  
Sep 03 10:43:32 ip-172-31-33-4.eu-north-1.compute.internal systemd[1]: Starting  
Sep 03 10:43:32 ip-172-31-33-4.eu-north-1.compute.internal systemd[1]: Started  
Sep 03 10:43:32 ip-172-31-33-4.eu-north-1.compute.internal httpd[2467]: Server  
[root@ip-172-31-33-4 ~]# find / -name httpd.conf  
/etc/httpd/conf/httpd.conf  
/usr/lib/tmpfiles.d/httpd.conf  
/usr/lib/sysusers.d/httpd.conf  
[root@ip-172-31-33-4 ~]# cd /usr/lib/tmpfiles.d/  
[root@ip-172-31-33-4 tmpfiles.d]# vi httpd.conf
```

- `Cd /usr/lib/tmpfiles.d/` : going ton the directory where the httpd.conf file is present
- See the content in the file : `vi "filename".conf`

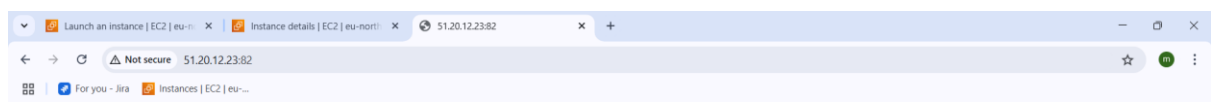
```
root@ip-172-31-39-167:/etc/httpd/conf
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for win32), the
# server will use that explicit path.  If the filenames do "not" begin
# with "/", the value of ServerRoot is prepended -- so 'log/access_log'
# with ServerRoot set to '/www' will be interpreted by the
# server as '/www/log/access_log', where as '/log/access_log' will be
# interpreted as '/log/access_log'.

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path.  If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used.  If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default.  See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts.  See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 82

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
-- INSERT --
```

- After entering in to file look LISTEN 80 and the change 80 to 82 by using “i” command
- Save it by entering esc button :wq!
- Run command : systemctl restart httpd
- 



**It works!**

- Copy the ipv4 adress and enter : “ipv4 public address” :82
- You can see the the httpd is running on port 82.

#### **4. Deploying sample index.html on httpd:**

```
[root@ip-172-31-39-167 ~]# vi httpd.conf
[root@ip-172-31-39-167 ~]# systemctl stop httpd
[root@ip-172-31-39-167 ~]# systemctl start httpd
[root@ip-172-31-39-167 ~]# find / -name index.html
/usr/lib/python3.9/site-packages/awscli/customizations/sso/index.html
/usr/share/doc/oniguruma/index.html
/usr/share/doc/cyrus-sasl-lib/index.html
/usr/share/doc/python3-jinja2/html/index.html
/usr/share/httpd/noindex/index.html
[root@ip-172-31-39-167 ~]# find / -name html
/var/www/html
/usr/lib64/python3.9/html
/usr/share/doc/python3-jinja2/html
[root@ip-172-31-39-167 ~]# cd /var/www/html
[root@ip-172-31-39-167 html]#
```

- Go to the html directory where the index.html file is present
- Cd /var/www/html



```
[root@ip-172-31-39-167 html]# vi index.html
[root@ip-172-31-39-167 html]# |
```

→ Enter : vi index.html – to view the file contents

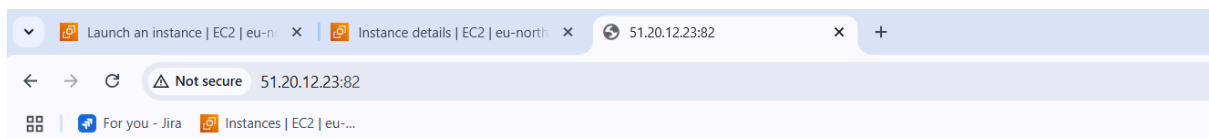
```
<h1> httpd running in port82 </h1>
~
~
~
~
~
~
~
~
~
~
```

→ In the file enter the output you want

→ And save it by entering esc button :wq!

```
[root@ip-172-31-39-167 html]# systemctl restart httpd
[root@ip-172-31-39-167 html]# |
```

→ Systemctl restart httpd

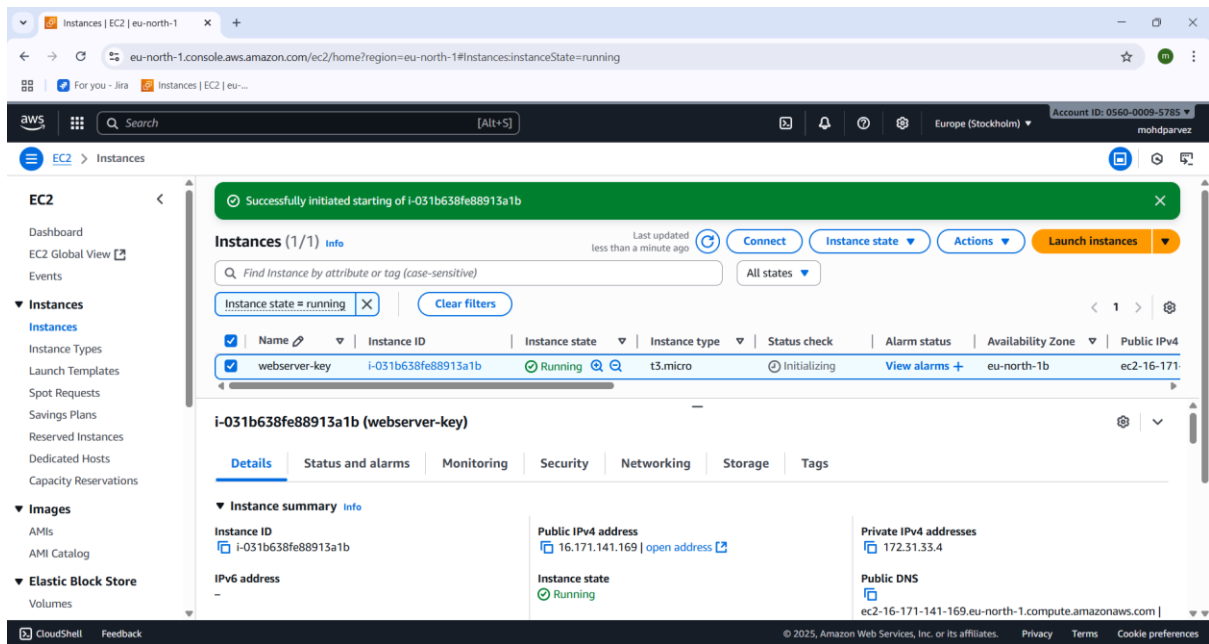


## httpd running in port82

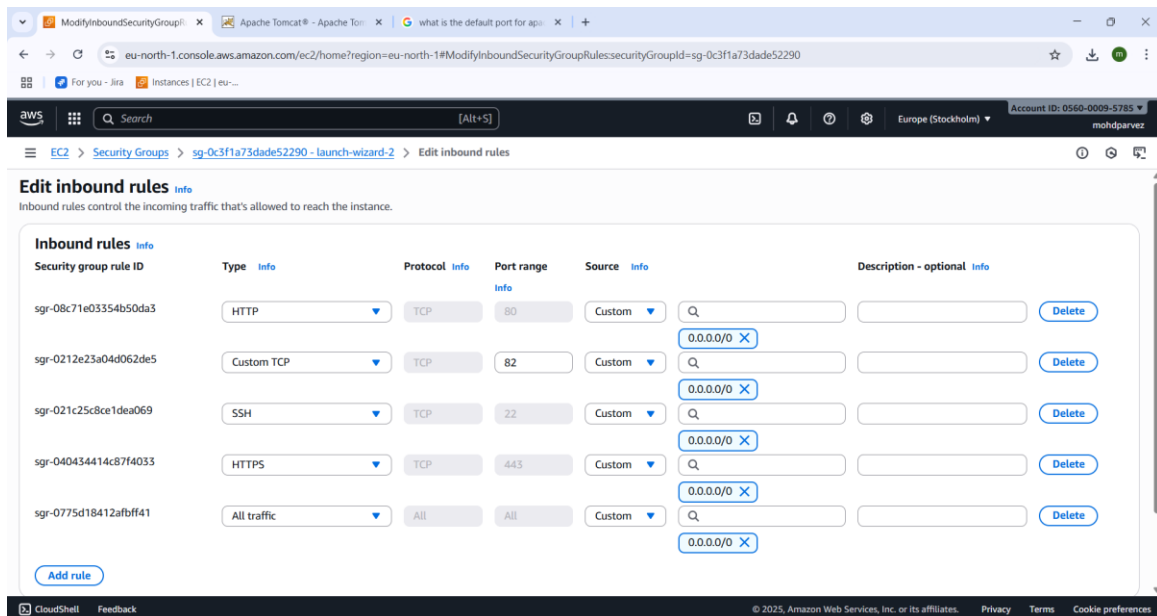
→ ENTER : AWS ipv4 address and enter port : “ipv4 public address” :82

→ You can see the page with the index.html I have provided.

## 5. APACHE TOMCAT ON PORT NUMBER 8082

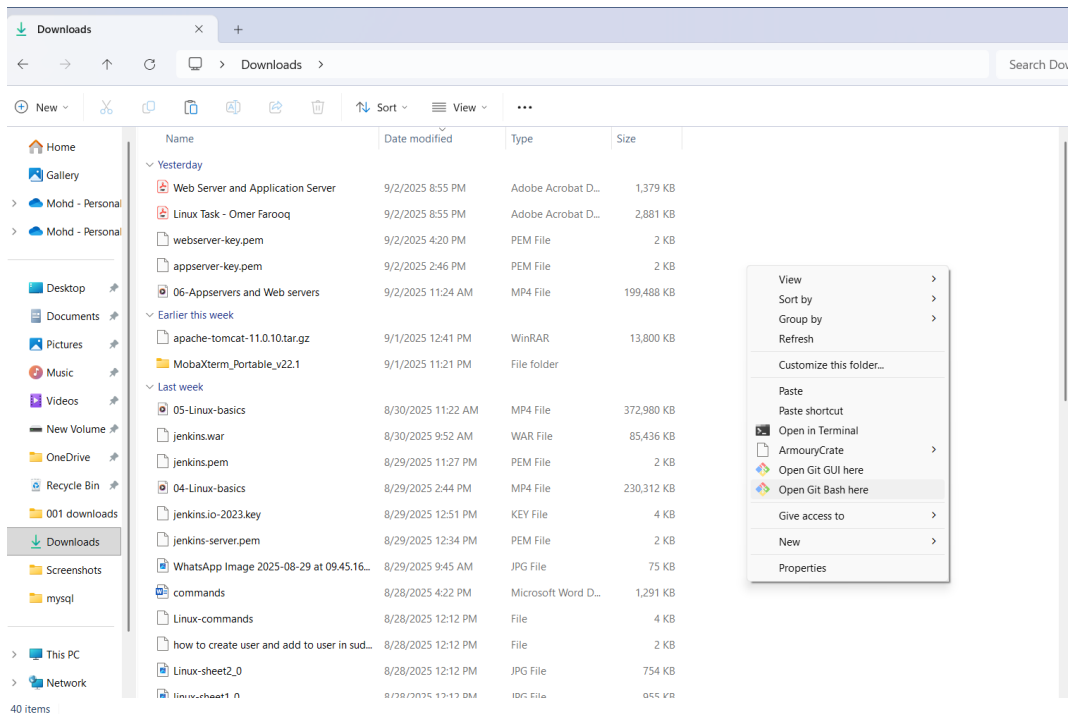


→ Lauch aws ec2 instance .



→ Go to security groups and click on edit inbound rules

→ Add security group with custom tcp and give port no all and save.



→ Go to the location where keypair.pem is downloaded and right click and select open git bash here.

```
ec2-user@ip-172-31-33-4:~
Mohd Parvez@LAPTOP-NT4KDSHS MINGW64 /e/downloads
$ ssh -i webserver-key.pem ec2-user@16.171.141.169
bash: ssh: command not found

Mohd Parvez@LAPTOP-NT4KDSHS MINGW64 /e/downloads
$ ssh -i web
Web Server and Application Server.pdf webserver-key.pem

Mohd Parvez@LAPTOP-NT4KDSHS MINGW64 /e/downloads
$ ssh -i webserver-key.pem ec2-user@16.171.141.169
The authenticity of host '16.171.141.169 (16.171.141.169)' can't be established.
ED25519 key fingerprint is SHA256:+5JcZDJAohcxB7UcbHJLUU/fRQ04ShZugUBTUn7Nncw.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:10: 16.171.148.159
  ~/.ssh/known_hosts:12: 51.21.196.65
  ~/.ssh/known_hosts:13: 13.60.83.249
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '16.171.141.169' (ED25519) to the list of known hosts
.
#_
~\_#####
~\_#####\
~\_###|

Amazon Linux 2023
```

→ Connect to the remote machine by using : ssh -i “keypair.pem” ec2-user@ “ip\_v4\_public address”.



The screenshot shows the Apache Tomcat download page at `tomcat.apache.org/download-90.cgi`. The left sidebar contains navigation links for documentation, problems, and getting involved. The main content area shows the current version as 9.0.108 and provides links for binary distributions. A right-click context menu is open over the `tar.gz` link, showing options like 'Open link in new tab', 'Save link as...', and 'Copy link address'.

Instances | EC2 | eu-north-1    Apache Tomcat® - Apache Tomcat

← → ↻ `tomcat.apache.org/download-90.cgi`

For you - Jira    Instances | EC2 | eu-north-1

Tomcat Migration Tool for Jakarta EE  
Tomcat Connectors  
Tomcat Native  
Taglibs  
Archives

**Documentation**  
Tomcat 11.0  
Tomcat 10.1  
Tomcat 9.0  
Upgrading  
Tomcat Connectors  
Tomcat Native 2  
Tomcat Native 1.3  
Wiki  
Migration Guide  
Presentations  
Specifications

**Problems?**  
Security Reports  
Find help  
FAQ  
Mailing Lists  
Bug Database  
IRC

**Get Involved**  
Overview  
Source code  
Buildbot  
Tools

**Media**  
Twitter

You are currently using **`https://d1cdn.apache.org/`**. If you encounter a problem with this mirror (or a complete list of mirrors) that should be available.

Other mirrors: `https://d1cdn.apache.org/`

**9.0.108**

Please see the [README](#) file for packaging information. It explains what every distribution contains.

**Binary Distributions**

- Core:
  - [zip](#) ([pgp](#), [sha512](#))
  - [tar.gz](#) ([pgp](#), [sha512](#))
  - [source code](#)
  - [binaries](#)
  - [binaries](#) ([pgp](#), [sha512](#))
- Full distribution
  - [tar.gz](#) ([pgp](#), [sha512](#))
- Deployable archives
  - [zip](#) ([pgp](#), [sha512](#))
  - [tar.gz](#) ([pgp](#), [sha512](#))
- Embedded archives
  - [zip](#) ([pgp](#), [sha512](#))
  - [tar.gz](#) ([pgp](#), [sha512](#))

**Source Code**

- [tar.gz](#) ([pgp](#), [sha512](#))
- [zip](#) ([pgp](#), [sha512](#))

- Every application is cannot be installed by “yum”
- Go to google and search for apache tomcat download select version copy the link of the download file that is zip or tar.

```
libxau-1.0.11-6.amzn2023.0.1.x86_64
libxext-1.3.6-1.amzn2023.0.1.x86_64
libxi-1.8.2-1.amzn2023.0.1.x86_64
libxinerama-1.1.5-6.amzn2023.0.1.x86_64
libxrandr-1.5.4-3.amzn2023.0.1.x86_64
libxrender-0.9.11-6.amzn2023.0.1.x86_64
libxt-1.3.0-3.amzn2023.0.1.x86_64
libxtst-1.2.5-1.amzn2023.0.1.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.1.x86_64
libpng-2:1.6.37-10.amzn2023.0.1.x86_64
libxcb-1.17.0-1.amzn2023.0.1.x86_64
pixman-0.43.4-1.amzn2023.0.1.x86_64
xml-common-0.6.3-56.amzn2023.0.1.x86_64

Complete!
[root@ip-172-31-33-4 ~]# wget https://downloads.apache.org/tomcat/tomcat-9/v9.0.108/bin/apache-tomcat-9.0.108.tar.gz.sha512
```

- Enter wget “url” that you copied and enter to download the tomcat.

```
[root@ip-172-31-33-4 opt]# wget https://d1cdn.apache.org/tomcat/tomcat-9/v9.0.108/bin/apache-tomcat-9.0.108.tar.gz
--2025-09-04 07:58:36-- https://d1cdn.apache.org/tomcat/tomcat-9/v9.0.108/bin/apache-tomcat-9.0.108.tar.gz
Resolving d1cdn.apache.org (d1cdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to d1cdn.apache.org (d1cdn.apache.org)|151.101.2.132|:443... connected
HTTP request sent, awaiting response... 200 OK
Length: 13028093 (12M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.108.tar.gz'

apache-tomcat-9.0.108.tar.gz 100%[=====>]
2025-09-04 07:58:36 (228 MB/s) - 'apache-tomcat-9.0.108.tar.gz' saved [13028093]

[root@ip-172-31-33-4 opt]# |
```

- here you can see the file has been downloaded.
- Now you need to extract the file

```
[root@ip-172-31-33-4 opt]# ll
total 12724
-rw-r--r--. 1 root root 13028093 Jul 31 18:33 apache-tomcat-9.0.108.tar.gz
drwxr-xr-x. 4 root root 4096 Aug 13 21:08 aws
[root@ip-172-31-33-4 opt]# tar xvf apache-tomcat-9.0.108.tar.gz
apache-tomcat-9.0.108/conf/
apache-tomcat-9.0.108/conf/catalina.policy
apache-tomcat-9.0.108/conf/catalina.properties
apache-tomcat-9.0.108/conf/context.xml
apache-tomcat-9.0.108/conf/localhost.xml
apache-tomcat-9.0.108/conf/server.xml
apache-tomcat-9.0.108/webapps/
```

- ll – to list the files
- tar xvf “file name” .gz : to extract the tomcat file.

```
[root@ip-172-31-33-4 opt]# cd apache-tomcat-9.0.108/
[root@ip-172-31-33-4 apache-tomcat-9.0.108]# ls
BUILDING.txt  LICENSE  README.md  RUNNING.txt  conf  logs  webapps
CONTRIBUTING.md  NOTICE  RELEASE-NOTES  bin  lib  temp  work
[root@ip-172-31-33-4 apache-tomcat-9.0.108]# cd bin/
[root@ip-172-31-33-4 bin]# ls
bootstrap.jar  commons-daemon-native.tar.gz  digest.sh  shutdown.sh  tool-wrapper.sh
catalina-tasks.xml  commons-daemon.jar  makebase.bat  startup.bat  version.bat
catalina.bat  configtest.bat  setclasspath.bat  tomcat-juli.jar  version.sh
catalina.sh  configtest.sh  setclasspath.sh  tomcat-native.tar.gz
ciphers.bat  daemon.sh  shutdown.bat  tool-wrapper.bat
ciphers.sh
[root@ip-172-31-33-4 bin]# |
```

- go to the tomcat directory where it is downloaded : cd “filename”/
- ls – to list the files you can find the configuration files of tomcat.
- Go to bin directory : cd bin/
- Ls - list the files you can find the startup.sh , sh- means shell & .bat is for windows



```
[root@ip-172-31-33-4 bin]# bash ./startup.sh
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.108
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.108
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.108/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.108/bin/
jar
Using CATALINA_OPTS:
Tomcat started.
[root@ip-172-31-33-4 bin]# |
```

- Start the tomcat : `bash ./startup.sh` – to start the service.
- Note : `systemctl start` cannot be used for every application

```
[root@ip-172-31-33-4 bin]# netstat -na | grep 8080
tcp6      0      0 :::8080 :::*           LISTEN
[root@ip-172-31-33-4 bin]# ps -ef | grep 8080
root      3311      1899  0 08:09 pts/1    00:00:00 grep --color=auto 8080
[root@ip-172-31-33-4 bin]# |
```

- Check the status using : `netstat -na | grep 8080`
- `Ps -ef | grep 8080` - you see the port 8080 is listening.
- Default log files for tomcat is `catlina.out` you can see logs.
- Go to web page and enter : “ip “ :8080 – you can see tomcat apache is running on 8080 port

```
[root@ip-172-31-39-167 ~]# find / -name server.xml
/etc/tomcat10/server.xml
[root@ip-172-31-39-167 ~]# cd /etc/tomcat10/
[root@ip-172-31-39-167 tomcat10]# vi server.xml
```

- Find the `server.xml` file
- Vi `server.xml` file

```

-->
<Connector port="8082" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443"
           maxParameterCount="1000"
           />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
           port="8082" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443"
           maxParameterCount="1000"
           />
-->
<!-- Define an SSL/TLS HTTP/1.1 Connector on port 8443 with HTTP/2
This connector uses the NIO implementation. The default
SSLImplementation will depend on the presence of the APR/native
library and the useOpenSSL attribute of the AprLifecycleListener
-->

```

- Change the port to 8082 and save it
- Restart the service and check status

```
[root@ip-172-31-39-167 tomcat10]# systemctl status tomcat10
● tomcat10.service - Apache Tomcat 10 Web Application Container
   Loaded: loaded (/usr/lib/systemd/system/tomcat10.service; disabled; preset: disabled)
   Active: active (running) since Wed 2025-09-03 13:33:15 UTC; 3min 19s ago
     Main PID: 32866 (java)
        Tasks: 30 (limit: 1057)
       Memory: 67.4M
          CPU: 4.422s
      CGroup: /system.slice/tomcat10.service
              └─32866 /usr/lib/jvm/jre/bin/java -Djavax.sql.DataSource.Factory=org.apache.com

Sep 03 13:33:16 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:16 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:16 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:16 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:16 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:16 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:17 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:17 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:17 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:17 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
Sep 03 13:33:17 ip-172-31-39-167.eu-north-1.compute.internal server[32866]: 03-Sep-2025 13:3
[root@ip-172-31-39-167 tomcat10]# cd ~
[root@ip-172-31-39-167 ~]#
[root@ip-172-31-39-167 ~]# |
```

- Got ot web page and enter : “ip address”:8082
- You can see the tomcat apache is running on port8082.



## 6. DEPLOY SAMPLE APPS ON WEB APPS

- Goto directory /opt/apache\_tomat9/webapps
- Download the sample.war that is sample app file
- sample.war file in the directory .
- restart the service : systemctl restart tomcat 9
- Then go to web page an enter : ipadress:80/sample.
- Then you can see the sample app hasbeen deployed.

## 7. CREATE TOMCAT.SERVICE FILE

```
[root@ip-172-31-35-46 ~]# cd /etc/systemd/
[root@ip-172-31-35-46 systemd]# ls
coredump.conf  logind.conf  oomd.conf      sleep.conf     system.conf.d  user.conf
homed.conf     network      pstore.conf    system          timesyncd.conf
journald.conf  networkd.conf resolved.conf   systemd.conf    user
[root@ip-172-31-35-46 systemd]# cd system
[root@ip-172-31-35-46 system]# ls
basic.target.wants      nfs-idmapd.service.requires
chronyd.service.wants   nfs-mountd.service.requires
cloud-init.target.wants  nfs-server.service.requires
ctrl-alt-del.target     remote-fs.target.wants
dbus-org.freedesktop.home1.service  rpc-gssd.service.requires
dbus-org.freedesktop.network1.service  rpc-statd-notify.service.requires
dbus-org.freedesktop.resolve1.service  rpc-statd.service.requires
dbus.service            sockets.target.wants
getty.target.wants      sysinit.target.wants
multi-user.target.wants  sysstat.service.wants
network-online.target.wants  systemd-homed.service.wants
nfs-blkmap.service.requires  timers.target.wants
[root@ip-172-31-35-46 system]# vi tomcat.service
[root@ip-172-31-35-46 system]#
```

- Go to the default location of file that is system
- Add tomcat.service file : vi tomcat.service

Add configuration :

```
root@ip-172-31-35-46:/etc/systemd/system
[Unit]
Description=Apache Tomcat Web Application Container
After=network.target

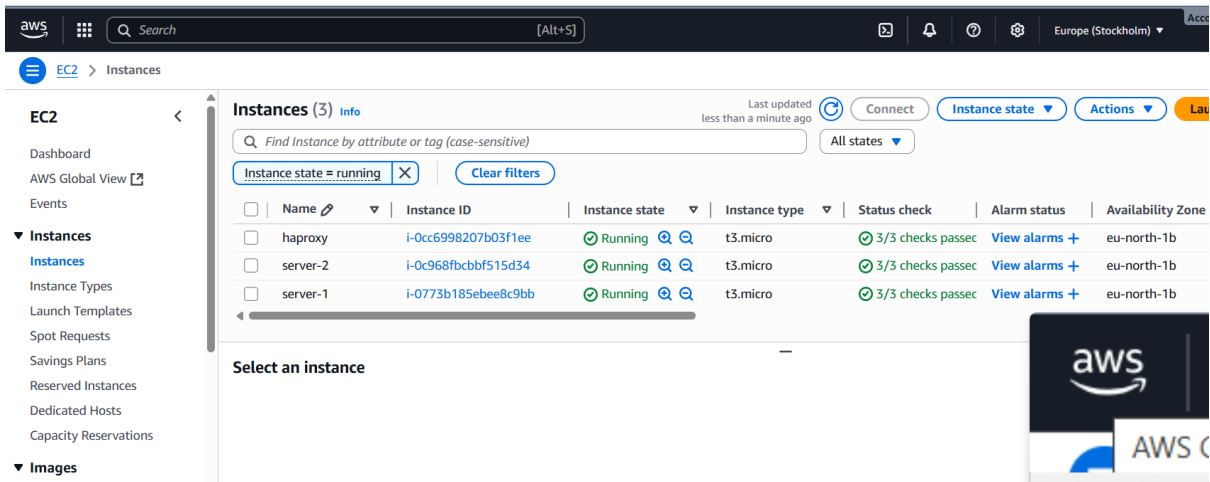
[Service]
Type=forking
Environment=JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
User=tomcat
Group=tomcat
RestartSec=10
Restart=always

[Install]
WantedBy=multi-user.target

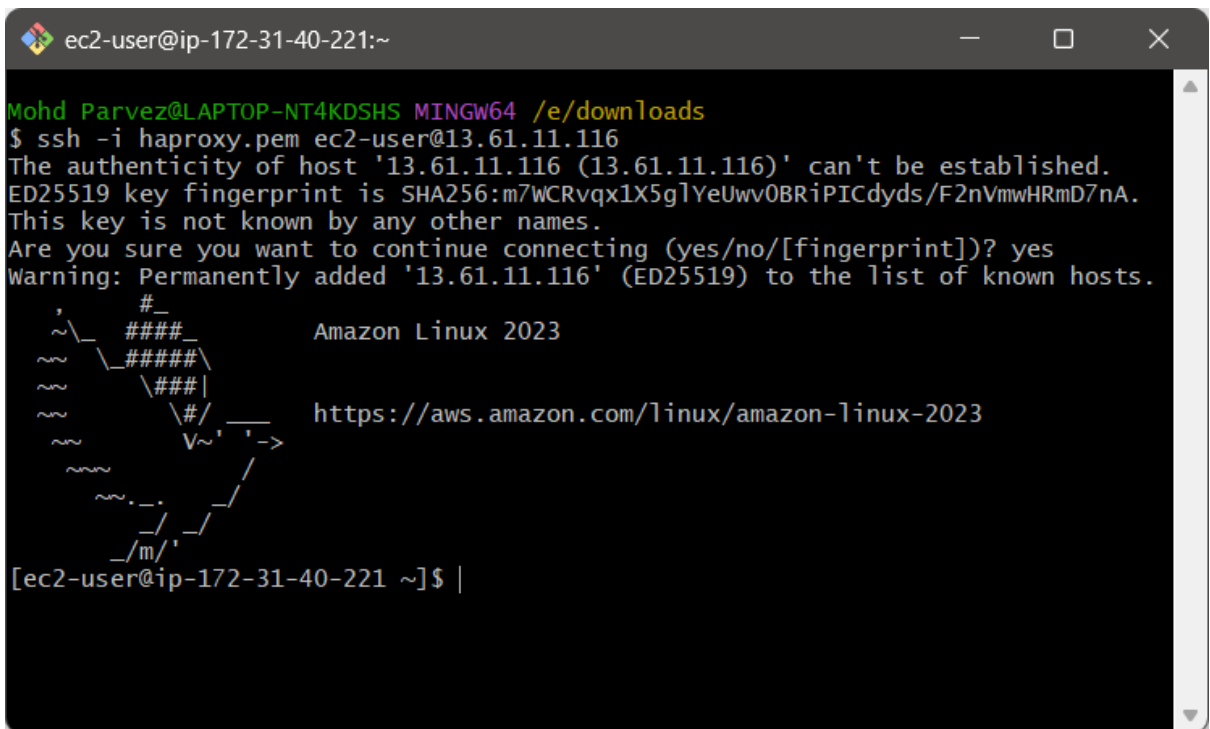
~
~
~
~
~
~
~
"tomcat.service" 19L, 456B                                     19,26  All
```

- save and exit
- restart the service
- make sure to create user with name tomcat and give access
- reload the daemon

## 8. CONFIGURE HAPROXY SERVER.



- Launch three ec2 instance



- Go to the location where `keypar.pem` is downloaded and right click and select open git bash here and connect to the `server-1` instance.



```
root@ip-172-31-40-221:~  
[root@ip-172-31-40-221 ~]# systemctl start httpd  
Failed to start httpd.service: Unit httpd.service not found.  
[root@ip-172-31-40-221 ~]# systemctl start httpd  
[root@ip-172-31-40-221 ~]# systemctl status httpd  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: d>  
   Active: active (running) since Sun 2025-09-07 15:36:17 UTC; 10s ago  
     Docs: man:httpd.service(8)  
  Main PID: 25991 (httpd)  
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes>  
    Tasks: 177 (limit: 1057)  
   Memory: 13.3M  
      CPU: 62ms  
   CGroup: /system.slice/httpd.service  
           └─25991 /usr/sbin/httpd -DFOREGROUND  
             └─25992 /usr/sbin/httpd -DFOREGROUND  
               └─25993 /usr/sbin/httpd -DFOREGROUND  
                 └─25994 /usr/sbin/httpd -DFOREGROUND  
                   └─25995 /usr/sbin/httpd -DFOREGROUND  
  
Sep 07 15:36:17 ip-172-31-40-221.eu-north-1.compute.internal systemd[1]: Starti>  
Sep 07 15:36:17 ip-172-31-40-221.eu-north-1.compute.internal systemd[1]: Starte>  
Sep 07 15:36:17 ip-172-31-40-221.eu-north-1.compute.internal httpd[25991]: Serv>  
[root@ip-172-31-40-221 ~]# |
```

- Check the status :systemctl status httpd

```
root@ip-172-31-40-221:~  
Failed to start httpd.service: Unit httpd.service not found.  
[root@ip-172-31-40-221 ~]# systemctl start httpd  
[root@ip-172-31-40-221 ~]# systemctl status httpd  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: d>  
   Active: active (running) since Sun 2025-09-07 15:36:17 UTC; 10s ago  
     Docs: man:httpd.service(8)  
  Main PID: 25991 (httpd)  
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes>  
    Tasks: 177 (limit: 1057)  
   Memory: 13.3M  
      CPU: 62ms  
   CGroup: /system.slice/httpd.service  
           └─25991 /usr/sbin/httpd -DFOREGROUND  
             └─25992 /usr/sbin/httpd -DFOREGROUND  
               └─25993 /usr/sbin/httpd -DFOREGROUND  
                 └─25994 /usr/sbin/httpd -DFOREGROUND  
                   └─25995 /usr/sbin/httpd -DFOREGROUND  
  
Sep 07 15:36:17 ip-172-31-40-221.eu-north-1.compute.internal systemd[1]: Starti>  
Sep 07 15:36:17 ip-172-31-40-221.eu-north-1.compute.internal systemd[1]: Starte>  
Sep 07 15:36:17 ip-172-31-40-221.eu-north-1.compute.internal httpd[25991]: Serv>  
[root@ip-172-31-40-221 ~]# vi /etc/hosts  
[root@ip-172-31-40-221 ~]#
```

- Go to : Vi /etc/hosts











```
root@ip-172-31-39-143:~  
Complete!  
[root@ip-172-31-39-143 ~]# systemctl start haproxy  
[root@ip-172-31-39-143 ~]# systemctl status haproxy  
● haproxy.service - HAProxy Load Balancer  
   Loaded: loaded (/usr/lib/systemd/system/haproxy.service; disabled; preset: enabled)  
   Active: active (running) since Sun 2025-09-07 16:22:19 UTC; 14s ago  
     Process: 27217 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -f $CFGDIR -c -q $>  
    Main PID: 27219 (haproxy)  
       Status: "Ready."  
        Tasks: 3 (limit: 1057)  
       Memory: 6.3M  
         CPU: 39ms  
      CGroup: /system.slice/haproxy.service  
              └─27219 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/>  
                └─27221 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/>  
  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27219]: [N>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal systemd[1]: Starte>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [A>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27219]: [N>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:20 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>
```

- start the haproxy service and check the status.

```
root@ip-172-31-39-143:~  
   Loaded: loaded (/usr/lib/systemd/system/haproxy.service; disabled; preset: enabled)  
   Active: active (running) since Sun 2025-09-07 16:22:19 UTC; 14s ago  
     Process: 27217 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -f $CFGDIR -c -q $>  
    Main PID: 27219 (haproxy)  
       Status: "Ready."  
        Tasks: 3 (limit: 1057)  
       Memory: 6.3M  
         CPU: 39ms  
      CGroup: /system.slice/haproxy.service  
              └─27219 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/>  
                └─27221 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/>  
  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27219]: [N>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal systemd[1]: Starte>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [A>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27219]: [N>  
Sep 07 16:22:19 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:20 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:20 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:21 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [w>  
Sep 07 16:22:21 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27221]: [A>  
[root@ip-172-31-39-143 ~]# vi /etc/hosts  
[root@ip-172-31-39-143 ~]# |
```

- go to the hosts file and add ip address of server-1 and server-2





```
root@ip-172-31-39-143:~  
timeout check      10s  
maxconn            3000  
  
#-----  
# main frontend which proxys to the backends  
#-----  
frontend main *:80  
  
    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  
  
#-----  
-- INSERT --                                     66,19      88%
```

```
root@ip-172-31-39-143:~  
    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 13.61.11.116:80 check  
    server app6 16.170.203.232:80 check|  
-- INSERT --                                     92,41      Bot
```

- Add server-1 and server-2 ip adrees and port to the file
- Save and exit

```

root@ip-172-31-39-143:~# systemctl restart haproxy
Job for haproxy.service failed because the control process exited with error cod
e.
See "systemctl status haproxy.service" and "journalctl -xeu haproxy.service" for
details.
root@ip-172-31-39-143:~# vi /etc/haproxy/haproxy.cfg
root@ip-172-31-39-143:~# systemctl restart haproxy
root@ip-172-31-39-143:~# systemctl status haproxy
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/usr/lib/systemd/system/haproxy.service; disabled; preset: disabled)
   Active: active (running) since Sun 2025-09-07 16:37:45 UTC; 14s ago
   Process: 27712 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -f $CFGDIR -c -q $OPTIONS (code=exited, status=0/
Main PID: 27724 (haproxy)
   Status: "Ready."
    Tasks: 3 (limit: 1057)
   Memory: 6.3M
     CPU: 43ms
    CGroup: /system.slice/haproxy.service
            └─27724 /usr/sbin/haproxy -ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.
              └─27728 /usr/sbin/haproxy -ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.

Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27724]: [ALERT] (27724) : config : par
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27724]: [NOTICE] (27724) : New worker (
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27728]: [WARNING] (27728) : Server stati
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27728]: [ALERT] (27728) : backend 'sta
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal systemd[1]: Started haproxy.service - HAProxy Loa
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27724]: [NOTICE] (27724) : Loading succ
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27728]: [WARNING] (27728) : Server app/a
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27728]: [WARNING] (27728) : Server app/a
Sep 07 16:37:45 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27728]: [WARNING] (27728) : Server app/a
Sep 07 16:37:46 ip-172-31-39-143.eu-north-1.compute.internal haproxy[27728]: [WARNING] (27728) : Server app/a
lines 1-23/23 (END)

```

- Restart and check the status of the service





**It works!**

- Go to browser and copy the haproxy server ip and enter : ip adress port number
- You can see first httpd service and refresh you see the nginx is running on the port 80