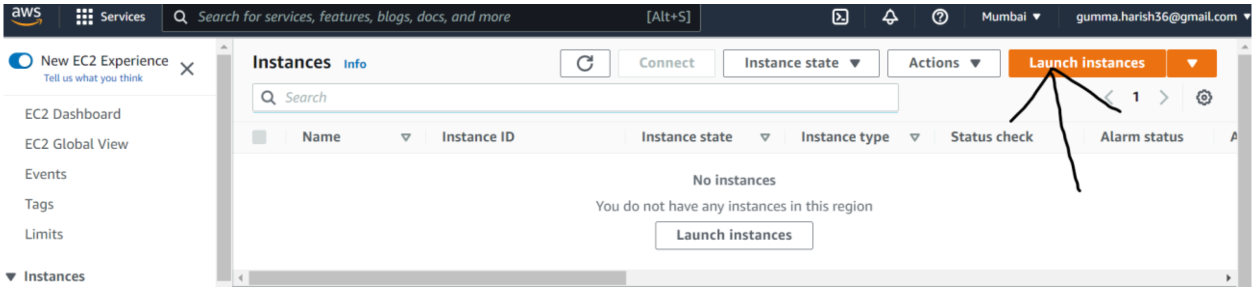
How install the apache directory

Step-1

Launch the instances



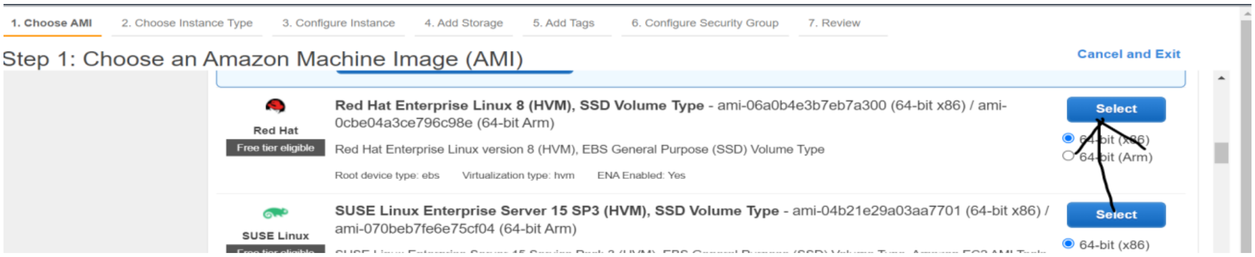
No instances

You do not have any instances in this region

Launch instances

Step-2

Choose an Amazon Machine Image (AMI)



Step 1: Choose an Amazon Machine Image (AMI)

Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-06a0b4e3b7eb7a300 (64-bit x86) / ami-0cbe04a3ce796c98e (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

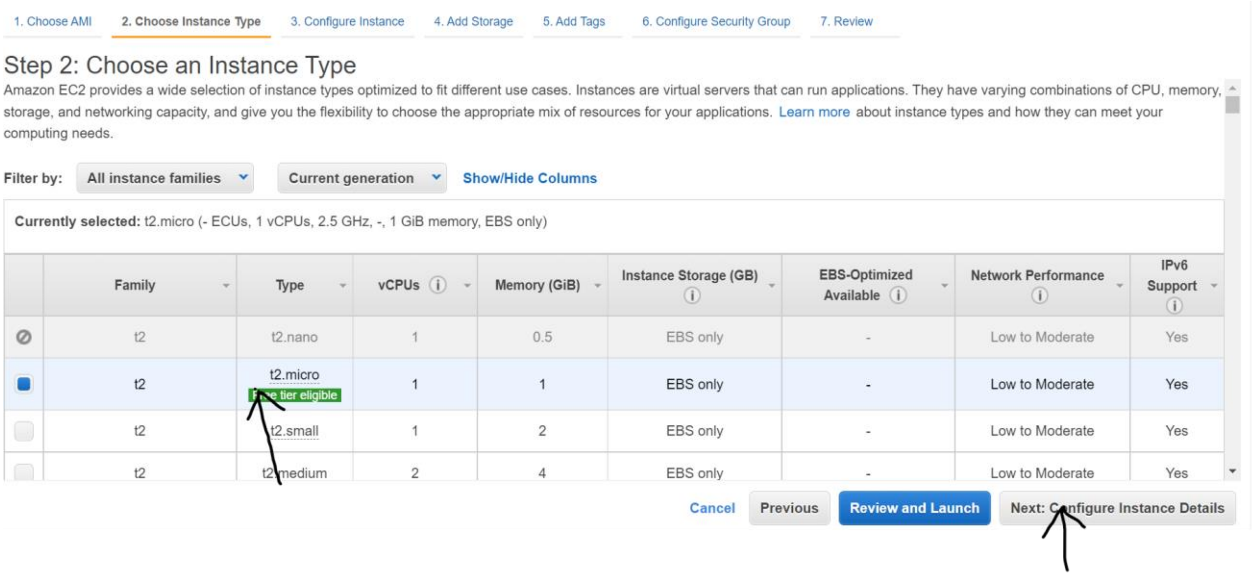
● 64-bit (x86)
○ 64-bit (Arm)

SUSE Linux Enterprise Server 15 SP3 (HVM), SSD Volume Type - ami-04b21e29a03aa7701 (64-bit x86) / ami-070beb7fe6e75cf04 (64-bit Arm)

● 64-bit (x86)

Step-3

Choose an instance type



Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

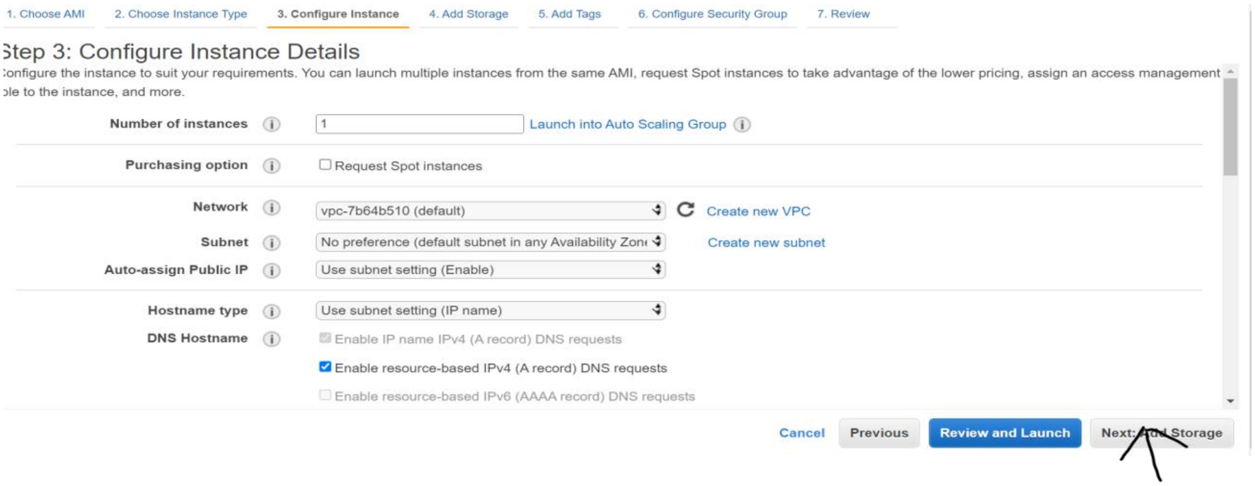
Filter by: All instance families Current generation Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
⊘	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
□	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
□	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes

Step-4

Configure instance details (keep the defaults setting only)



Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 — Launch into Auto Scaling Group

Purchasing option: □ Request Spot instances

Network: vpc-7b64b510 (default) — Create new VPC

Subnet: No preference (default subnet in any Availability Zone) — Create new subnet

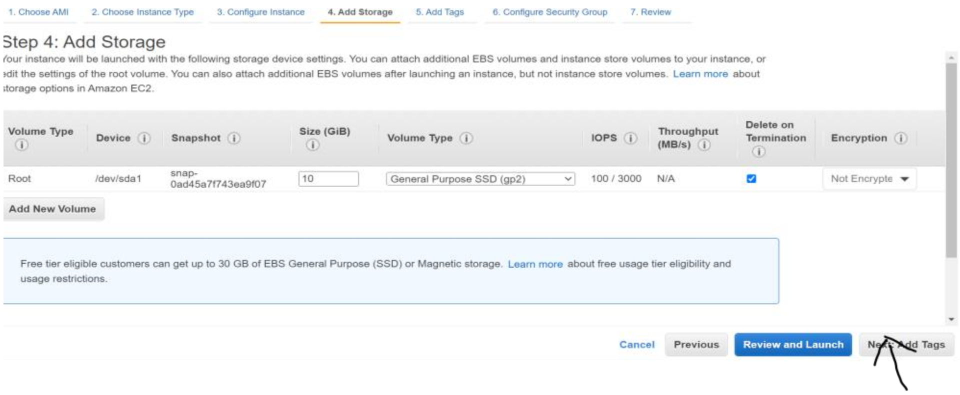
Auto-assign Public IP: Use subnet setting (Enable)

Hostname type: Use subnet setting (IP name)

DNS Hostname:
- ☑ Enable IP name IPv4 (A record) DNS requests
- ☑ Enable resource-based IPv4 (A record) DNS requests
- □ Enable resource-based IPv6 (AAAA record) DNS requests

Step-5

Add storage (on need to add keep the default storage only)



Step 4: Add Storage

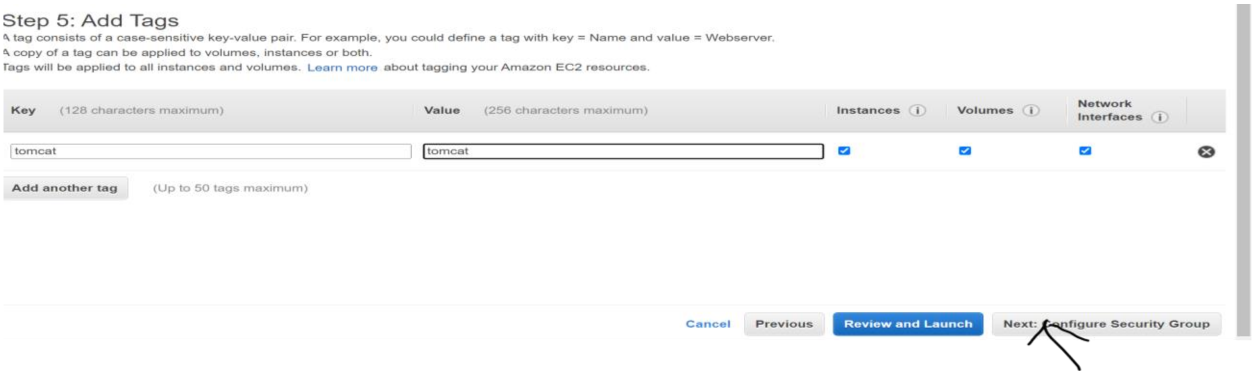
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-0ad45a7f743ea9f07	10	General Purpose SSD (gp2)	100 / 3000	N/A	☑	Not Encrypte

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

Step-6

Add tags



Step 5: Add Tags

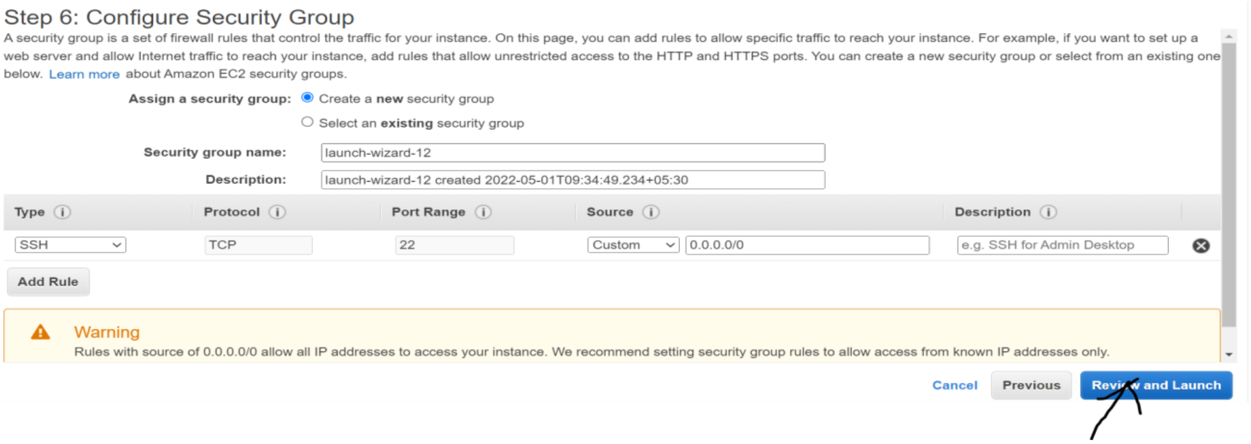
A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes	Network Interfaces
tomcat	tomcat	☑	☑	☑

(Up to 50 tags maximum)

Step-7

Configure security group (keep the defaults security group only don't change any thing)



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group
○ Select an **existing** security group

Security group name: launch-wizard-12

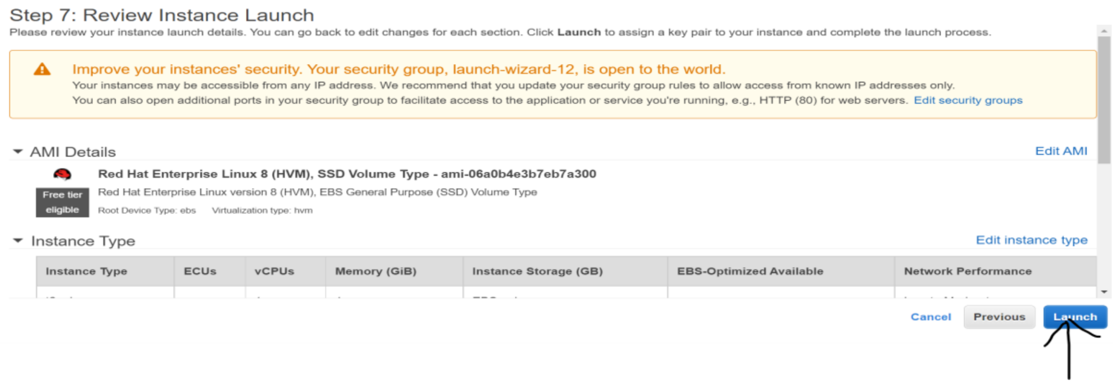
Description: launch-wizard-12 created 2022-05-01T09:34:49.234+05:30

Type	Protocol	Port Range	Source		Description
SSH	TCP	22	Custom	0.0.0.0/0	e.g. SSH for Admin Desktop

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Step-8

Review instance launch



Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

Improve your instances' security. Your security group, launch-wizard-12, is open to the world.
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.
You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. Edit security groups

AMI Details — Edit AMI

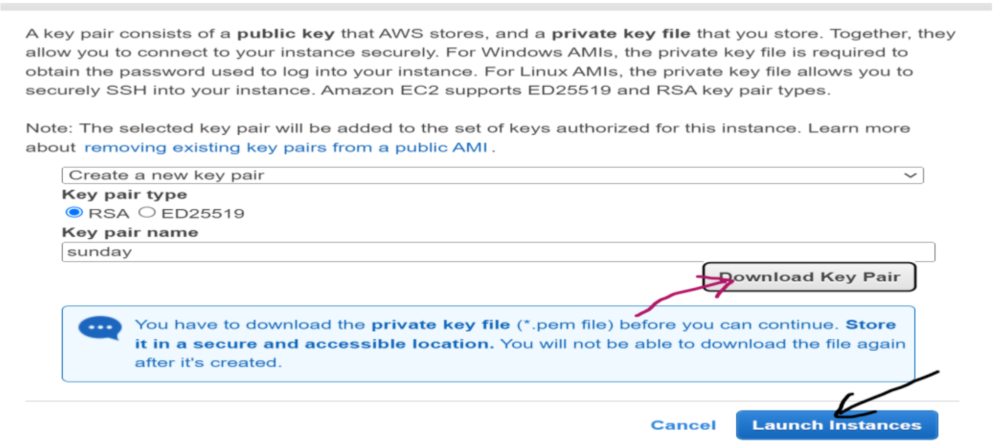
Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-06a0b4e3b7eb7a300
Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type
Root Device Type: ebs Virtualization type: hvm

Instance Type — Edit instance type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance

Step-9

Creating a new key pair (Download the key pair) and the Launch the instance



A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance. Amazon EC2 supports ED25519 and RSA key pair types.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI .

Create a new key pair

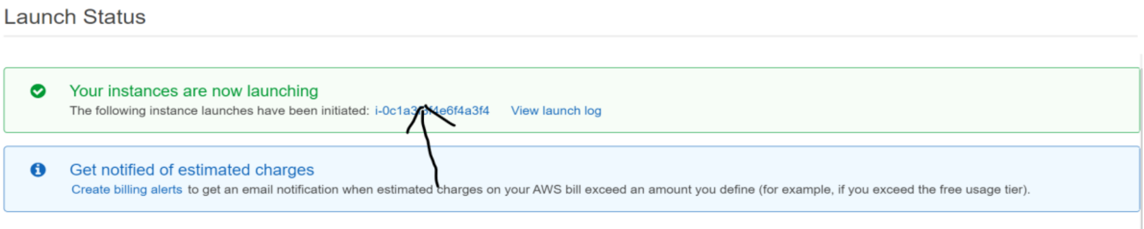
Key pair type
● RSA ○ ED25519

Key pair name
sunday

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Step-10

Launch status



Launch Status

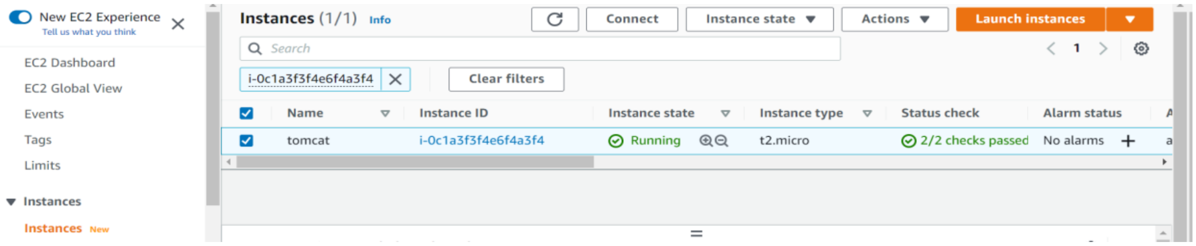
Your instances are now launching

The following instance launches have been initiated: i-0c1a3f3f4e6f4a3f4 View launch log

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

Finally we launched the instance

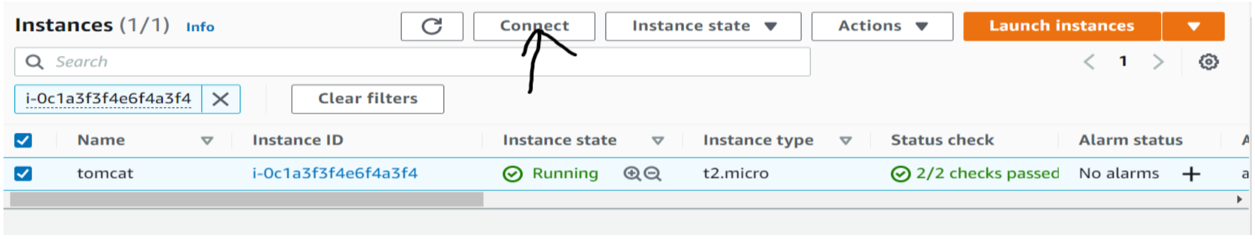


	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
☑	tomcat	i-0c1a3f3f4e6f4a3f4	Running	t2.micro	2/2 checks passed	No alarms

1.After launching the instance we need to convert the pem file into ppk file to access the putty terminal

Step-11

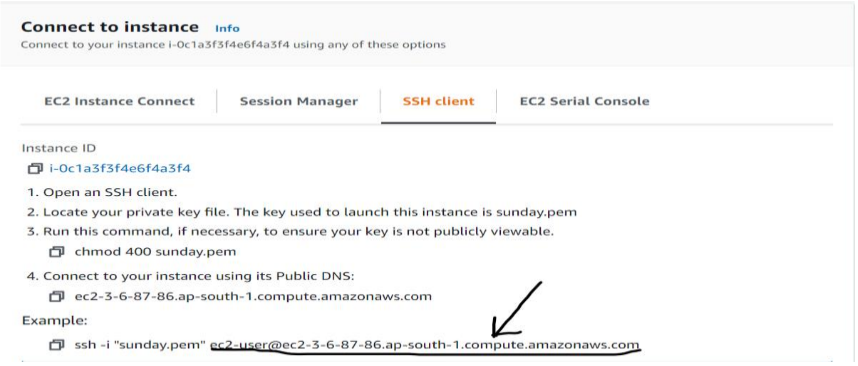
Press connect



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status
☑	tomcat	i-0c1a3f3f4e6f4a3f4	Running	t2.micro	2/2 checks passed	No alarms

Step-12

Copy the SSH id



Connect to instance Info

Connect to your instance i-0c1a3f3f4e6f4a3f4 using any of these options

Instance ID

i-0c1a3f3f4e6f4a3f4

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is sunday.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.

```
chmod 400 sunday.pem
```

4. Connect to your instance using its Public DNS:

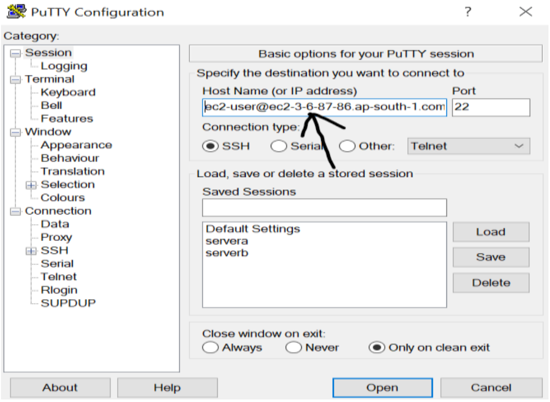
```
ec2-3-6-87-86.ap-south-1.compute.amazonaws.com
```

Example:

```
ssh -i "sunday.pem" ec2-user@ec2-3-6-87-86.ap-south-1.compute.amazonaws.com
```

Step-13

Open the putty (paste the ssh id inside the host name or IP address box)



Specify the destination you want to connect to

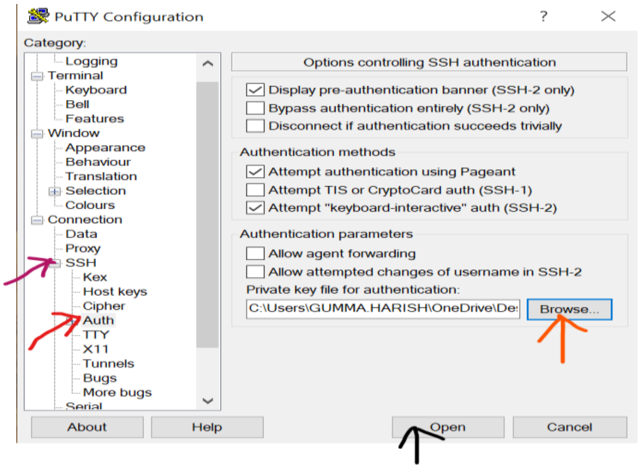
Host Name (or IP address): ec2-user@ec2-3-6-87-86.ap-south-1.com Port: 22

Connection type: ● SSH ○ Serial ○ Other: Telnet

Load, save or delete a stored session

Saved Sessions: Default Settings, servera, serverb

Close window on exit: ○ Always ○ Never ● Only on clean exit



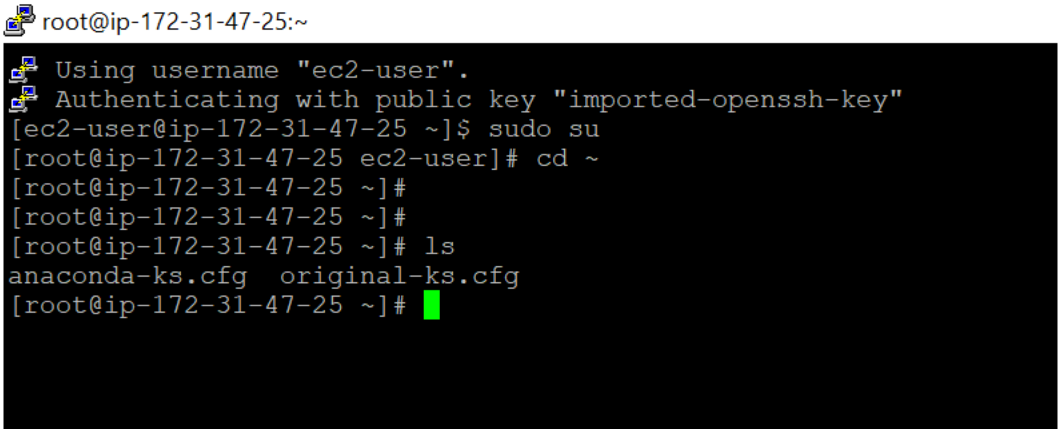
☑ Display pre-authentication banner (SSH-2 only)
□ Bypass authentication entirely (SSH-2 only)
□ Disconnect if authentication succeeds trivially

Authentication methods
☑ Attempt authentication using Pageant
□ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters
□ Allow agent forwarding
□ Allow attempted changes of username in SSH-2

Private key file for authentication: C:\Users\GUMMA.HARISH\OneDrive\De:

1. Click on SSH
2. Click on Auth
3. Browse for the ppk file
4. Open

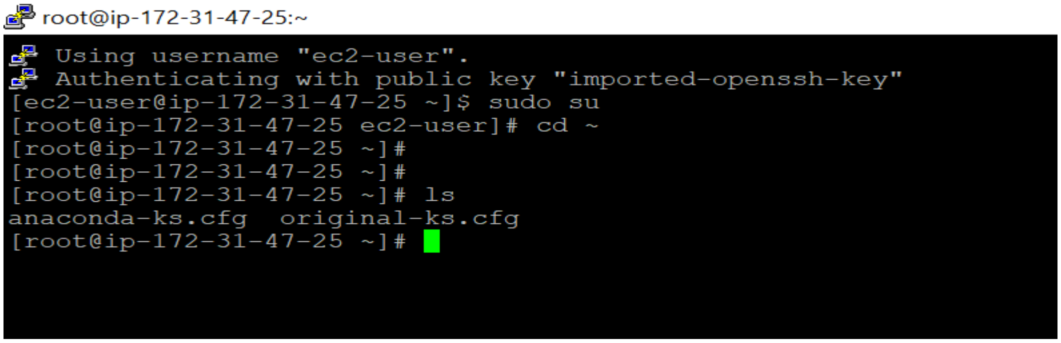


```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
[ec2-user@ip-172-31-47-25 ~]$ sudo su
[root@ip-172-31-47-25 ec2-user]# cd ~
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]# ls
anaconda-ks.cfg  original-ks.cfg
[root@ip-172-31-47-25 ~]#
```

You will get like this terminal

Step-14

We need to switch to root



root@ip-172-31-47-25:~

```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
[ec2-user@ip-172-31-47-25 ~]$ sudo su
[root@ip-172-31-47-25 ec2-user]# cd ~
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]# ls
anaconda-ks.cfg  original-ks.cfg
[root@ip-172-31-47-25 ~]#
```

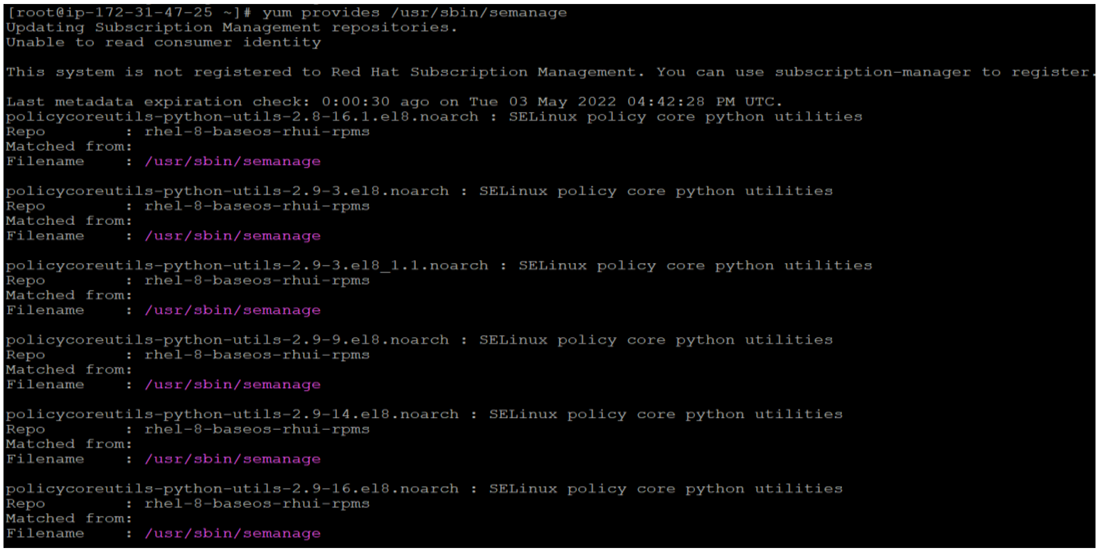
Step-15

You need to use the yum command to find out which package provides file called /usr/sbin/semanage. Type the following command:

```
# yum provides /usr/sbin/semanage
```

Link : https://www.cyberciti.biz/faq/redhat-install-semanage-selinux-command-rpm/ to install the semanage

Sample output

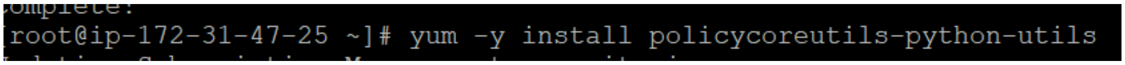


```
[root@ip-172-31-47-25 ~]# yum provides /usr/sbin/semanage
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Last metadata expiration check: 0:00:30 ago on Tue 03 May 2022 04:42:28 PM UTC.
policycoreutils-python-utils-2.8-16.1.el8.noarch : SELinux policy core python utilities
Repo        : rhel-8-baseos-rhui-rpms
Matched from:
Filename    : /usr/sbin/semanage

policycoreutils-python-utils-2.9-3.el8.noarch : SELinux policy core python utilities
Repo        : rhel-8-baseos-rhui-rpms
Matched from:
Filename    : /usr/sbin/semanage

policycoreutils-python-utils-2.9-3.el8_1.1.noarch : SELinux policy core python utilities
Repo        : rhel-8-baseos-rhui-rpms
Matched from:
Filename    : /usr/sbin/semanage

policycoreutils-python-utils-2.9-9.el8.noarch : SELinux policy core python utilities
Repo        : rhel-8-baseos-rhui-rpms
Matched from:
Filename    : /usr/sbin/semanage

policycoreutils-python-utils-2.9-14.el8.noarch : SELinux policy core python utilities
Repo        : rhel-8-baseos-rhui-rpms
Matched from:
Filename    : /usr/sbin/semanage

policycoreutils-python-utils-2.9-16.el8.noarch : SELinux policy core python utilities
Repo        : rhel-8-baseos-rhui-rpms
Matched from:
Filename    : /usr/sbin/semanage
```

Step-16

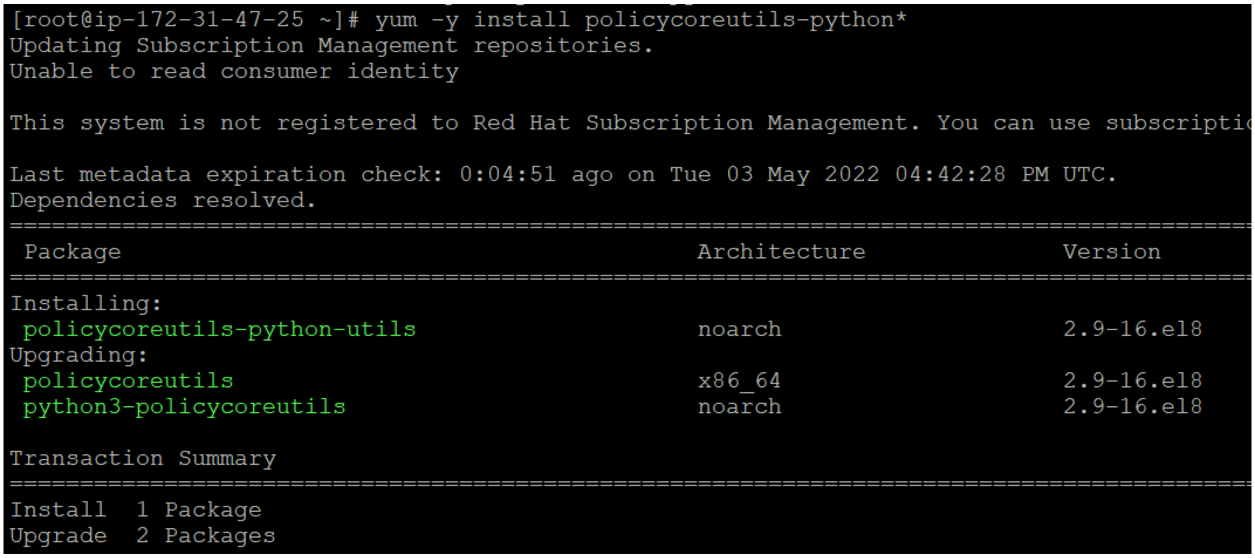
How to install semanage command in RHEL 6/7/8

# yum install policycoreutils-python-utils -y



```
omplete:
[root@ip-172-31-47-25 ~]# yum -y install policycoreutils-python-utils
```

Sample output



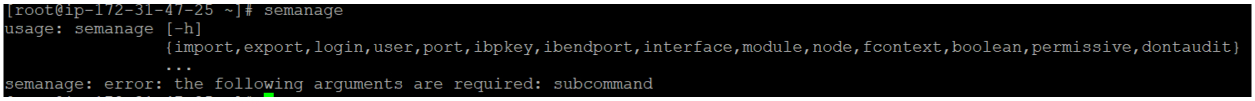
```
[root@ip-172-31-47-25 ~]# yum -y install policycoreutils-python*
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscripti

Last metadata expiration check: 0:04:51 ago on Tue 03 May 2022 04:42:28 PM UTC.
Dependencies resolved.
==========================================================================================
 Package                                        Architecture              Version
==========================================================================================
Installing:
 policycoreutils-python-utils                   noarch                    2.9-16.el8
Upgrading:
 policycoreutils                                x86_64                    2.9-16.el8
 python3-policycoreutils                        noarch                    2.9-16.el8

Transaction Summary
==========================================================================================
Install   1 Package
Upgrade   2 Packages
```

Step-17

How do I use the semanage command

Now, you can use semanage command

semanage



```
[root@ip-172-31-47-25 ~]# semanage
usage: semanage [-h]
                {import,export,login,user,port,ibpkey,ibendport,interface,module,node,fcontext,boolean,permissive,dontaudit}
                ...
semanage: error: the following arguments are required: subcommand
```

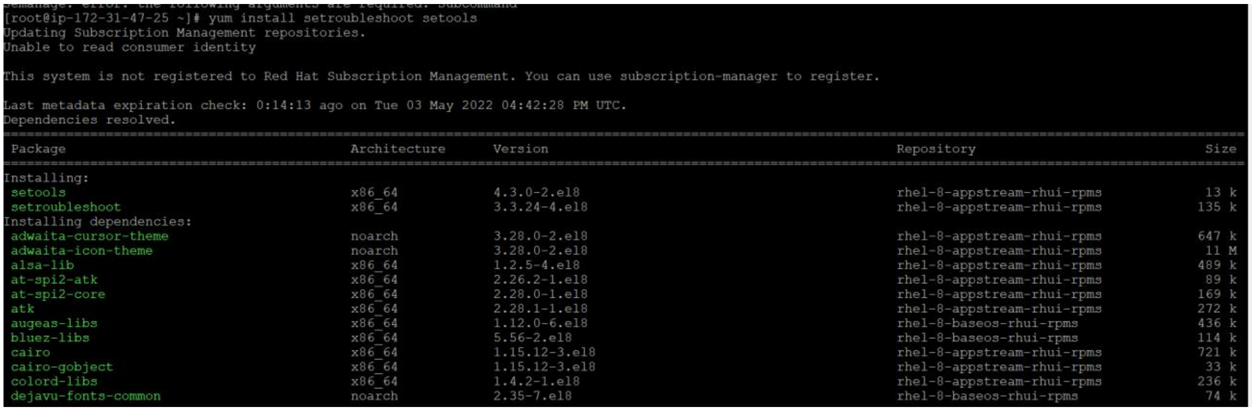
Step-18

Installing Setools and Setroubleshoot

Install setroubleshoot packages using Yum

Link : https://www.serverlab.ca/tutorials/linux/administration-linux/troubleshooting-selinux-centos-red-hat/ to install the setools and setroubleshoot

yum install setroubleshoot setools



```
semanage: error: the following arguments are required: subcommand
[root@ip-172-31-47-25 ~]# yum install setroubleshoot setools
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Last metadata expiration check: 0:14:13 ago on Tue 03 May 2022 04:42:28 PM UTC.
Dependencies resolved.
=================================================================================================================================
 Package                            Architecture      Version                                Repository                    Size
=================================================================================================================================
Installing:
 setools                            x86_64            4.3.0-2.el8                            rhel-8-appstream-rhui-rpms    13 k
 setroubleshoot                     x86_64            3.3.24-4.el8                           rhel-8-appstream-rhui-rpms   135 k
Installing dependencies:
 adwaita-cursor-theme               noarch            3.28.0-2.el8                           rhel-8-appstream-rhui-rpms   647 k
 adwaita-icon-theme                 noarch            3.28.0-2.el8                           rhel-8-appstream-rhui-rpms    11 M
 alsa-lib                           x86_64            1.2.5-4.el8                            rhel-8-appstream-rhui-rpms   489 k
 at-spi2-atk                        x86_64            2.26.2-1.el8                           rhel-8-appstream-rhui-rpms    89 k
 at-spi2-core                       x86_64            2.28.0-1.el8                           rhel-8-appstream-rhui-rpms   169 k
 atk                                x86_64            2.28.1-1.el8                           rhel-8-appstream-rhui-rpms   272 k
 augeas-libs                        x86_64            1.12.0-6.el8                           rhel-8-baseos-rhui-rpms      436 k
 bluez-libs                         x86_64            5.56-2.el8                             rhel-8-baseos-rhui-rpms      114 k
 cairo                              x86_64            1.15.12-3.el8                          rhel-8-appstream-rhui-rpms   721 k
 cairo-gobject                      x86_64            1.15.12-3.el8                          rhel-8-appstream-rhui-rpms    33 k
 colord-libs                        x86_64            1.4.2-1.el8                            rhel-8-appstream-rhui-rpms   236 k
 dejavu-fonts-common                noarch            2.35-7.el8                             rhel-8-baseos-rhui-rpms       74 k
```

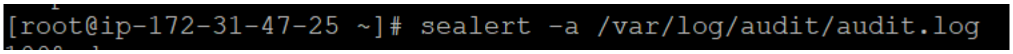
Step-19

SELinux Alerts

We now have a tool called sealert that analyzes the audit log used by SELinux. Sealert will scan the log file and report and will then generate a report containing all discovered SELinux issues.

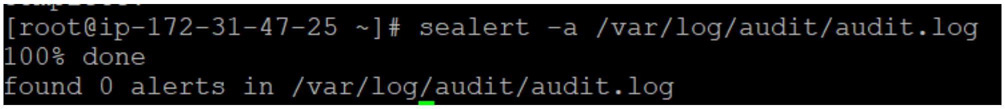
To run sealert from the command-line, we need to point it to the SELinux audit log.

sealert -a /var/log/audit/audit.log



```
[root@ip-172-31-47-25 ~]# sealert -a /var/log/audit/audit.log
```

Sample output

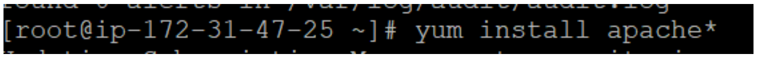


```
[root@ip-172-31-47-25 ~]# sealert -a /var/log/audit/audit.log
100% done
found 0 alerts in /var/log/audit/audit.log
```

Step-20

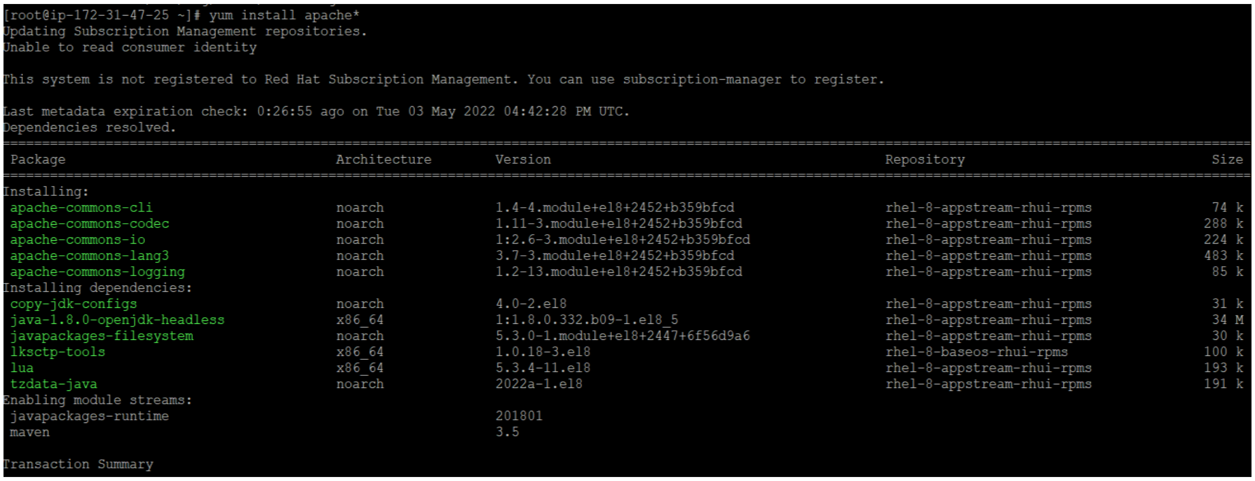
Install the apache

yum install apache*



```
[root@ip-172-31-47-25 ~]# yum install apache*
```

Sample output

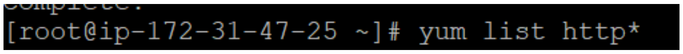


```
[root@ip-172-31-47-25 ~]# yum install apache*
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Last metadata expiration check: 0:26:55 ago on Tue 03 May 2022 04:42:28 PM UTC.
Dependencies resolved.
=============================================================================================================================================================
 Package                                  Architecture        Version                                          Repository                                Size
=============================================================================================================================================================
Installing:
 apache-commons-cli                       noarch              1.4-4.module+el8+2452+b359bfcd                   rhel-8-appstream-rhui-rpms                74 k
 apache-commons-codec                     noarch              1.11-3.module+el8+2452+b359bfcd                  rhel-8-appstream-rhui-rpms               288 k
 apache-commons-io                        noarch              1:2.6-3.module+el8+2452+b359bfcd                 rhel-8-appstream-rhui-rpms               224 k
 apache-commons-lang3                     noarch              3.7-3.module+el8+2452+b359bfcd                   rhel-8-appstream-rhui-rpms               483 k
 apache-commons-logging                   noarch              1.2-13.module+el8+2452+b359bfcd                  rhel-8-appstream-rhui-rpms                85 k
Installing dependencies:
 copy-jdk-configs                         noarch              4.0-2.el8                                        rhel-8-appstream-rhui-rpms                31 k
 java-1.8.0-openjdk-headless              x86_64              1:1.8.0.332.b09-1.el8_5                          rhel-8-appstream-rhui-rpms                34 M
 javapackages-filesystem                  noarch              5.3.0-1.module+el8+2447+6f56d9a6                 rhel-8-appstream-rhui-rpms                30 k
 lksctp-tools                             x86_64              1.0.18-3.el8                                     rhel-8-baseos-rhui-rpms                  100 k
 lua                                      x86_64              5.3.4-11.el8                                     rhel-8-appstream-rhui-rpms               193 k
 tzdata-java                              noarch              2022a-1.el8                                      rhel-8-appstream-rhui-rpms               191 k
Enabling module streams:
 javapackages-runtime                                         201801
 maven                                                        3.5

Transaction Summary
```

Step-21

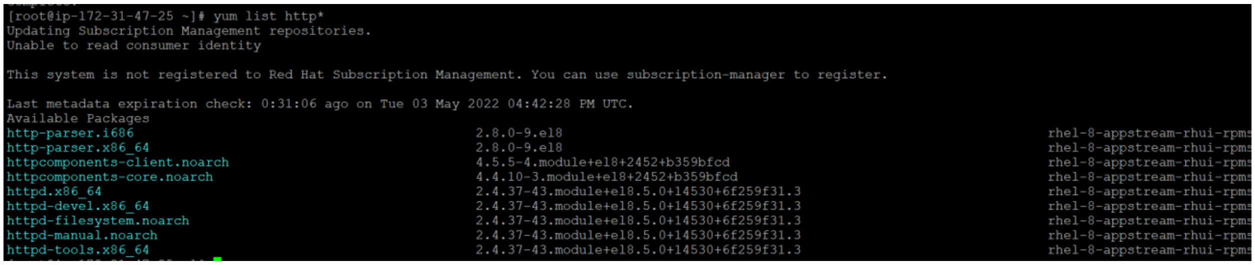
Install httpd

yum list http*



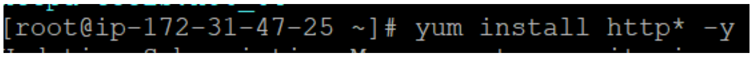
```
[root@ip-172-31-47-25 ~]# yum list http*
```

Sample output



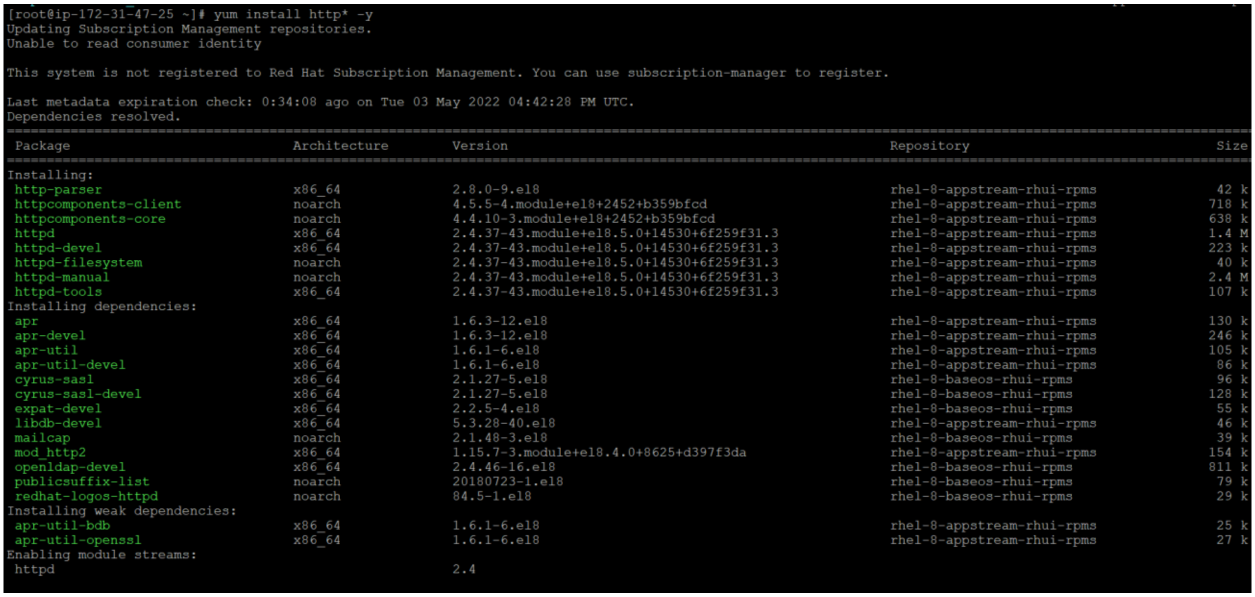
```
[root@ip-172-31-47-25 ~]# yum list http*
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Last metadata expiration check: 0:31:06 ago on Tue 03 May 2022 04:42:28 PM UTC.
Available Packages
http-parser.i686                                           2.8.0-9.el8                                             rhel-8-appstream-rhui-rpms
http-parser.x86_64                                         2.8.0-9.el8                                             rhel-8-appstream-rhui-rpms
httpcomponents-client.noarch                               4.5.5-4.module+el8+2452+b359bfcd                        rhel-8-appstream-rhui-rpms
httpcomponents-core.noarch                                 4.4.10-3.module+el8+2452+b359bfcd                       rhel-8-appstream-rhui-rpms
httpd.x86_64                                               2.4.37-43.module+el8.5.0+14530+6f259f31.3               rhel-8-appstream-rhui-rpms
httpd-devel.x86_64                                         2.4.37-43.module+el8.5.0+14530+6f259f31.3               rhel-8-appstream-rhui-rpms
httpd-filesystem.noarch                                    2.4.37-43.module+el8.5.0+14530+6f259f31.3               rhel-8-appstream-rhui-rpms
httpd-manual.noarch                                        2.4.37-43.module+el8.5.0+14530+6f259f31.3               rhel-8-appstream-rhui-rpms
httpd-tools.x86_64                                         2.4.37-43.module+el8.5.0+14530+6f259f31.3               rhel-8-appstream-rhui-rpms
```

yum install http* -y



```
[root@ip-172-31-47-25 ~]# yum install http* -y
```

Sample output

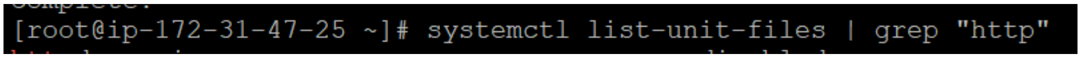


```
[root@ip-172-31-47-25 ~]# yum install http* -y
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Last metadata expiration check: 0:34:08 ago on Tue 03 May 2022 04:42:28 PM UTC.
Dependencies resolved.
================================================================================================================================================
 Package                            Architecture        Version                                                Repository                   Size
================================================================================================================================================
Installing:
 http-parser                        x86_64              2.8.0-9.el8                                            rhel-8-appstream-rhui-rpms   42 k
 httpcomponents-client              noarch              4.5.5-4.module+el8+2452+b359bfcd                       rhel-8-appstream-rhui-rpms  718 k
 httpcomponents-core                noarch              4.4.10-3.module+el8+2452+b359bfcd                      rhel-8-appstream-rhui-rpms  638 k
 httpd                              x86_64              2.4.37-43.module+el8.5.0+14530+6f259f31.3              rhel-8-appstream-rhui-rpms  1.4 M
 httpd-devel                        x86_64              2.4.37-43.module+el8.5.0+14530+6f259f31.3              rhel-8-appstream-rhui-rpms  223 k
 httpd-filesystem                   noarch              2.4.37-43.module+el8.5.0+14530+6f259f31.3              rhel-8-appstream-rhui-rpms   40 k
 httpd-manual                       noarch              2.4.37-43.module+el8.5.0+14530+6f259f31.3              rhel-8-appstream-rhui-rpms  2.4 M
 httpd-tools                        x86_64              2.4.37-43.module+el8.5.0+14530+6f259f31.3              rhel-8-appstream-rhui-rpms  107 k
Installing dependencies:
 apr                                x86_64              1.6.3-12.el8                                           rhel-8-appstream-rhui-rpms  130 k
 apr-devel                          x86_64              1.6.3-12.el8                                           rhel-8-appstream-rhui-rpms  246 k
 apr-util                           x86_64              1.6.1-6.el8                                            rhel-8-appstream-rhui-rpms  105 k
 apr-util-devel                     x86_64              1.6.1-6.el8                                            rhel-8-appstream-rhui-rpms   86 k
 cyrus-sasl                         x86_64              2.1.27-5.el8                                           rhel-8-baseos-rhui-rpms      96 k
 cyrus-sasl-devel                   x86_64              2.1.27-5.el8                                           rhel-8-baseos-rhui-rpms     128 k
 expat-devel                        x86_64              2.2.5-4.el8                                            rhel-8-baseos-rhui-rpms      55 k
 libdb-devel                        x86_64              5.3.28-40.el8                                          rhel-8-appstream-rhui-rpms   46 k
 mailcap                            noarch              2.1.48-3.el8                                           rhel-8-baseos-rhui-rpms      39 k
 mod_http2                          x86_64              1.15.7-3.module+el8.4.0+8625+d397f3da                  rhel-8-appstream-rhui-rpms  154 k
 openldap-devel                     x86_64              2.4.46-16.el8                                          rhel-8-baseos-rhui-rpms     811 k
 publicsuffix-list                  noarch              20180723-1.el8                                         rhel-8-baseos-rhui-rpms      79 k
 redhat-logos-httpd                 noarch              84.5-1.el8                                             rhel-8-baseos-rhui-rpms      29 k
Installing weak dependencies:
 apr-util-bdb                       x86_64              1.6.1-6.el8                                            rhel-8-appstream-rhui-rpms   25 k
 apr-util-openssl                   x86_64              1.6.1-6.el8                                            rhel-8-appstream-rhui-rpms   27 k
Enabling module streams:
 httpd                                                  2.4
```

Step-22

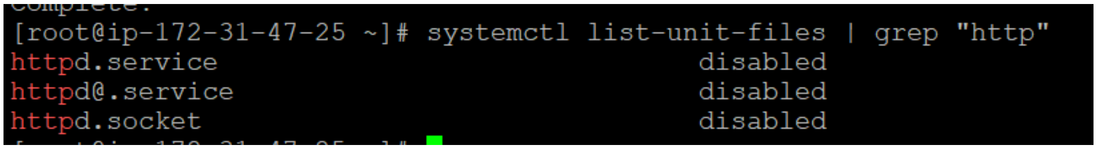
Check the daemon service start are not

# systemctl list-unit-files | grep “http”



```
[root@ip-172-31-47-25 ~]# systemctl list-unit-files | grep "http"
```

Sample output

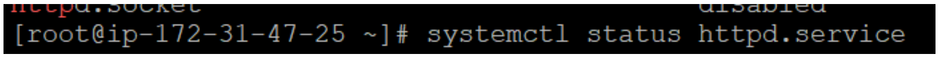


```
Complete:
[root@ip-172-31-47-25 ~]# systemctl list-unit-files | grep "http"
httpd.service                              disabled
httpd@.service                             disabled
httpd.socket                               disabled
```

Step-23

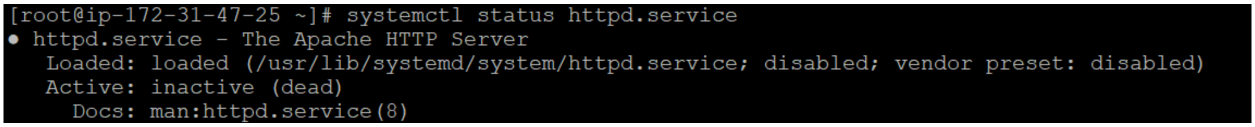
Check the status ,start ,enable,restart,of httpd.service

systemctl status httpd.service



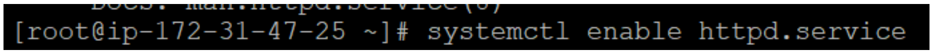
```
[root@ip-172-31-47-25 ~]# systemctl status httpd.service
```

Sample output



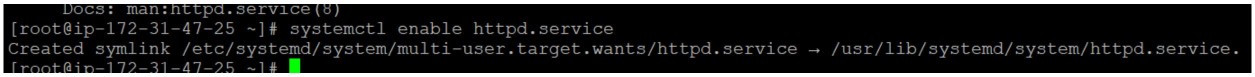
```
[root@ip-172-31-47-25 ~]# systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
```

systemctl enable httpd.service



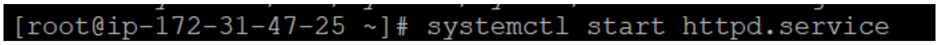
```
[root@ip-172-31-47-25 ~]# systemctl enable httpd.service
```

Sample output



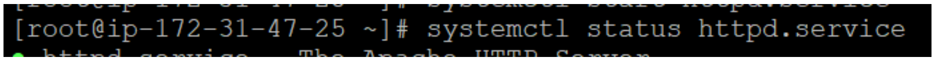
```
     Docs: man:httpd.service(8)
[root@ip-172-31-47-25 ~]# systemctl enable httpd.service
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-47-25 ~]#
```

systemctl start httpd.service



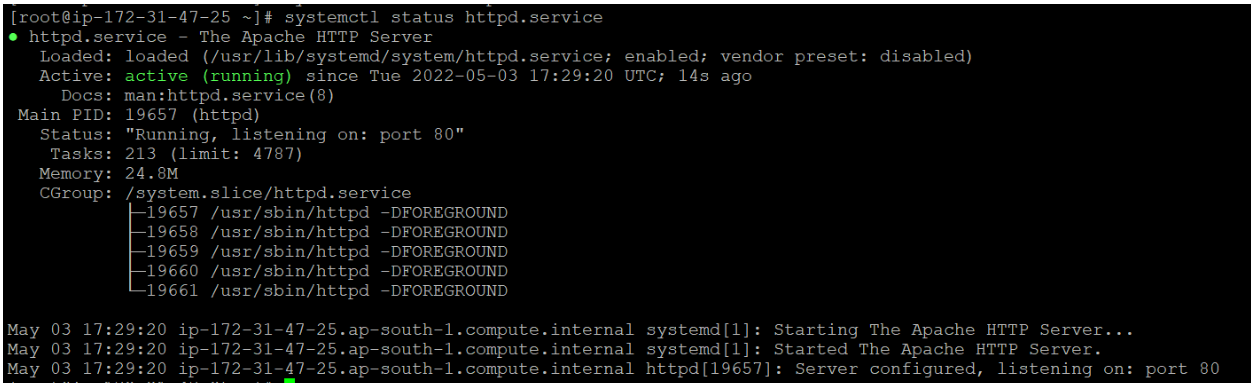
```
[root@ip-172-31-47-25 ~]# systemctl start httpd.service
```

systemctl status httpd.service



```
[root@ip-172-31-47-25 ~]# systemctl status httpd.service
```

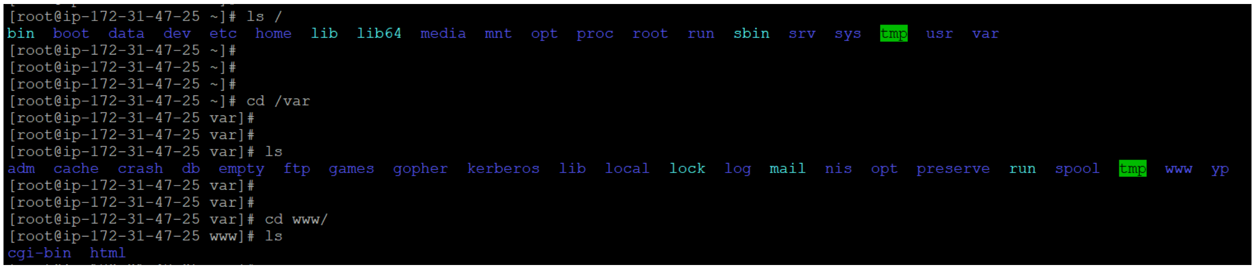
Sample output



```
[root@ip-172-31-47-25 ~]# systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2022-05-03 17:29:20 UTC; 14s ago
     Docs: man:httpd.service(8)
 Main PID: 19657 (httpd)
   Status: "Running, listening on: port 80"
    Tasks: 213 (limit: 4787)
   Memory: 24.8M
   CGroup: /system.slice/httpd.service
           ├─19657 /usr/sbin/httpd -DFOREGROUND
           ├─19658 /usr/sbin/httpd -DFOREGROUND
           ├─19659 /usr/sbin/httpd -DFOREGROUND
           ├─19660 /usr/sbin/httpd -DFOREGROUND
           └─19661 /usr/sbin/httpd -DFOREGROUND

May 03 17:29:20 ip-172-31-47-25.ap-south-1.compute.internal systemd[1]: Starting The Apache HTTP Server...
May 03 17:29:20 ip-172-31-47-25.ap-south-1.compute.internal systemd[1]: Started The Apache HTTP Server.
May 03 17:29:20 ip-172-31-47-25.ap-south-1.compute.internal httpd[19657]: Server configured, listening on: port 80
```

Check once in var directory

Sample output

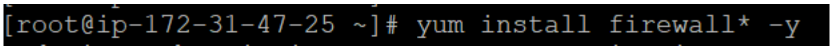


```
[root@ip-172-31-47-25 ~]# ls /
bin  boot  data  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]#
[root@ip-172-31-47-25 ~]# cd /var
[root@ip-172-31-47-25 var]#
[root@ip-172-31-47-25 var]#
[root@ip-172-31-47-25 var]# ls
adm  cache  crash  db  empty  ftp  games  gopher  kerberos  lib  local  lock  log  mail  nis  opt  preserve  run  spool  tmp  www  yp
[root@ip-172-31-47-25 var]#
[root@ip-172-31-47-25 var]#
[root@ip-172-31-47-25 var]# cd www/
[root@ip-172-31-47-25 www]# ls
cgi-bin  html
```

Step-24

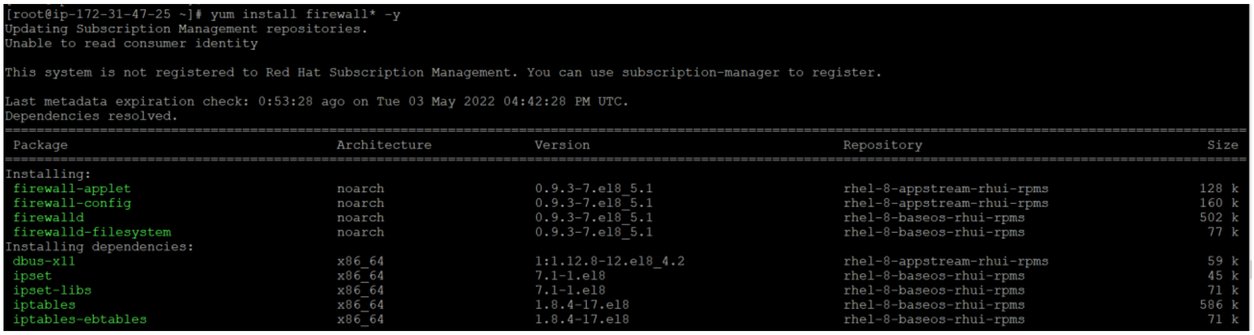
Check firewalld.services and ports (In aws ec2 instance we need to install this service)

yum install firewall* -y



```
[root@ip-172-31-47-25 ~]# yum install firewall* -y
```

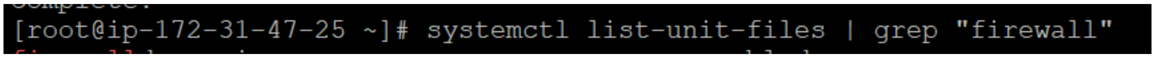
Sample output



```
[root@ip-172-31-47-25 ~]# yum install firewall* -y
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered to Red Hat Subscription Management. You can use subscription-manager to register.

Last metadata expiration check: 0:53:28 ago on Tue 03 May 2022 04:42:28 PM UTC.
Dependencies resolved.
==================================================================================================================================
 Package                        Architecture         Version                          Repository                            Size
==================================================================================================================================
Installing:
 firewall-applet                noarch               0.9.3-7.el8_5.1                  rhel-8-appstream-rhui-rpms           128 k
 firewall-config                noarch               0.9.3-7.el8_5.1                  rhel-8-appstream-rhui-rpms           160 k
 firewalld                      noarch               0.9.3-7.el8_5.1                  rhel-8-baseos-rhui-rpms              502 k
 firewalld-filesystem           noarch               0.9.3-7.el8_5.1                  rhel-8-baseos-rhui-rpms               77 k
Installing dependencies:
 dbus-x11                       x86_64               1:1.12.8-12.el8_4.2              rhel-8-appstream-rhui-rpms            59 k
 ipset                          x86_64               7.1-1.el8                        rhel-8-baseos-rhui-rpms               45 k
 ipset-libs                     x86_64               7.1-1.el8                        rhel-8-baseos-rhui-rpms               71 k
 iptables                       x86_64               1.8.4-17.el8                     rhel-8-baseos-rhui-rpms              586 k
 iptables-ebtables              x86_64               1.8.4-17.el8                     rhel-8-baseos-rhui-rpms               71 k
```

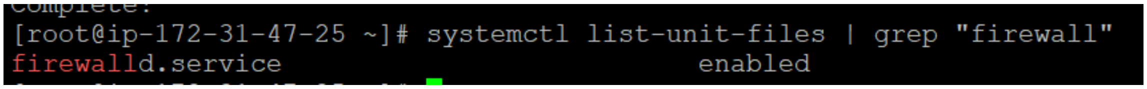
Check the firewall daemon service

systemctl list-unit-files | grep “firewall”



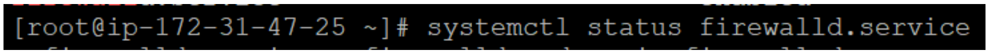
```
[root@ip-172-31-47-25 ~]# systemctl list-unit-files | grep "firewall"
```

Sample output



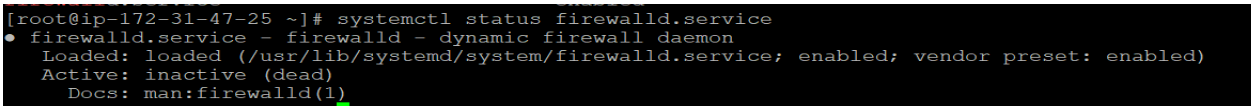
```
[root@ip-172-31-47-25 ~]# systemctl list-unit-files | grep "firewall"
firewalld.service                          enabled
```

systemctl status firewalld.service



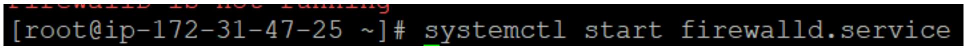
```
[root@ip-172-31-47-25 ~]# systemctl status firewalld.service
```

Sample output



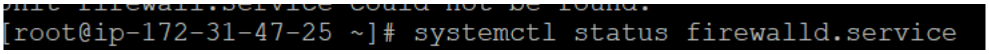
```
[root@ip-172-31-47-25 ~]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
     Docs: man:firewalld(1)
```

systemctl start firewalld.service



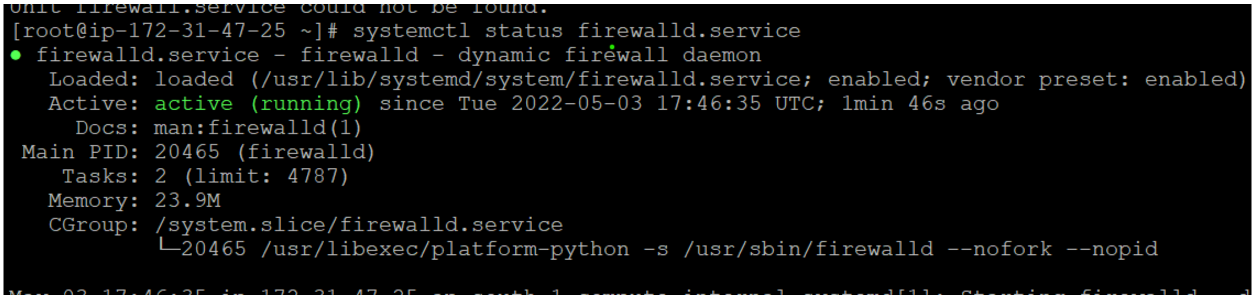
```
[root@ip-172-31-47-25 ~]# systemctl start firewalld.service
```

#systemctl status firewalld.service



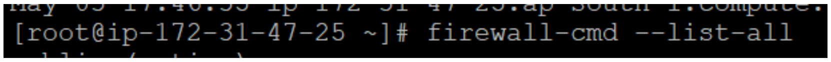
```
Unit firewall.service could not be found.
[root@ip-172-31-47-25 ~]# systemctl status firewalld.service
```

Sample output



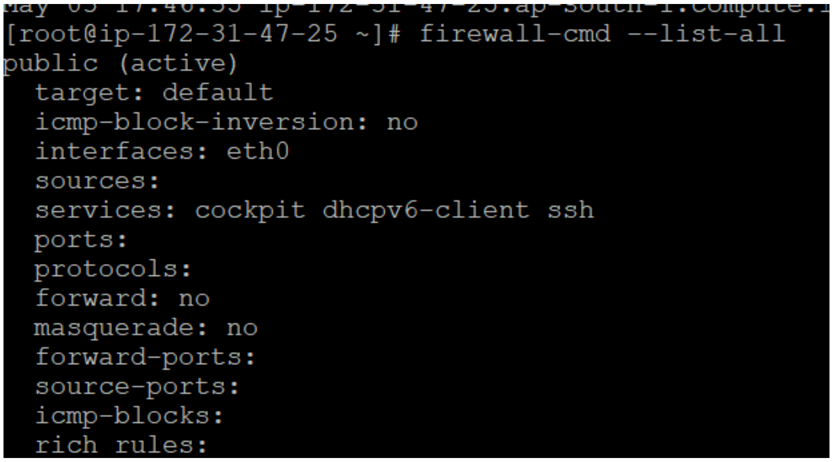
```
Unit firewall.service could not be found.
[root@ip-172-31-47-25 ~]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2022-05-03 17:46:35 UTC; 1min 46s ago
     Docs: man:firewalld(1)
 Main PID: 20465 (firewalld)
    Tasks: 2 (limit: 4787)
   Memory: 23.9M
   CGroup: /system.slice/firewalld.service
           └─20465 /usr/libexec/platform-python -s /usr/sbin/firewalld --nofork --nopid
```

firewall-cmd --list-all



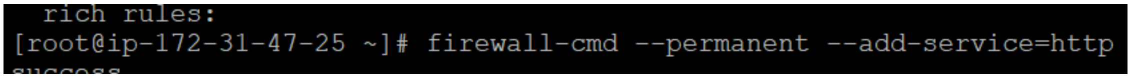
```
[root@ip-172-31-47-25 ~]# firewall-cmd --list-all
```

Sample output



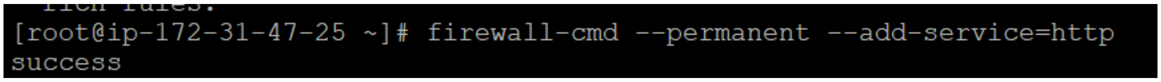
```
[root@ip-172-31-47-25 ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: eth0
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  protocols:
  forward: no
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

firewall-cmd --permanent --add-service=http



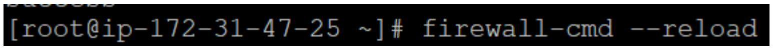
```
  rich rules:
[root@ip-172-31-47-25 ~]# firewall-cmd --permanent --add-service=http
```

Sample output



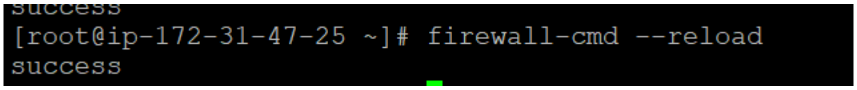
```
[root@ip-172-31-47-25 ~]# firewall-cmd --permanent --add-service=http
success
```

firewall-cmd --reload



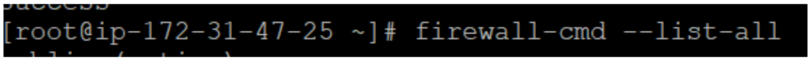
```
[root@ip-172-31-47-25 ~]# firewall-cmd --reload
```

Sample output



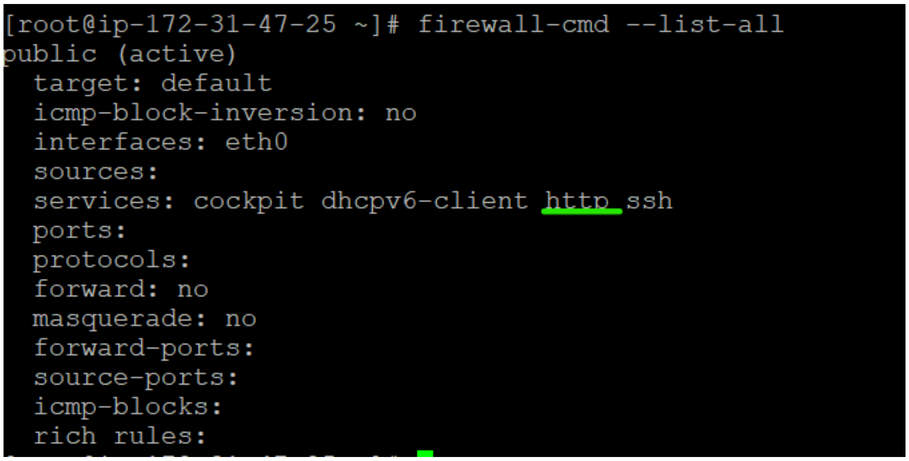
```
success
[root@ip-172-31-47-25 ~]# firewall-cmd --reload
success
```

firewall-cmd --list-all



```
[root@ip-172-31-47-25 ~]# firewall-cmd --list-all
```

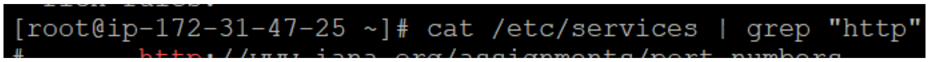
Sample output



```
[root@ip-172-31-47-25 ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: eth0
  sources:
  services: cockpit dhcpv6-client http ssh
  ports:
  protocols:
  forward: no
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

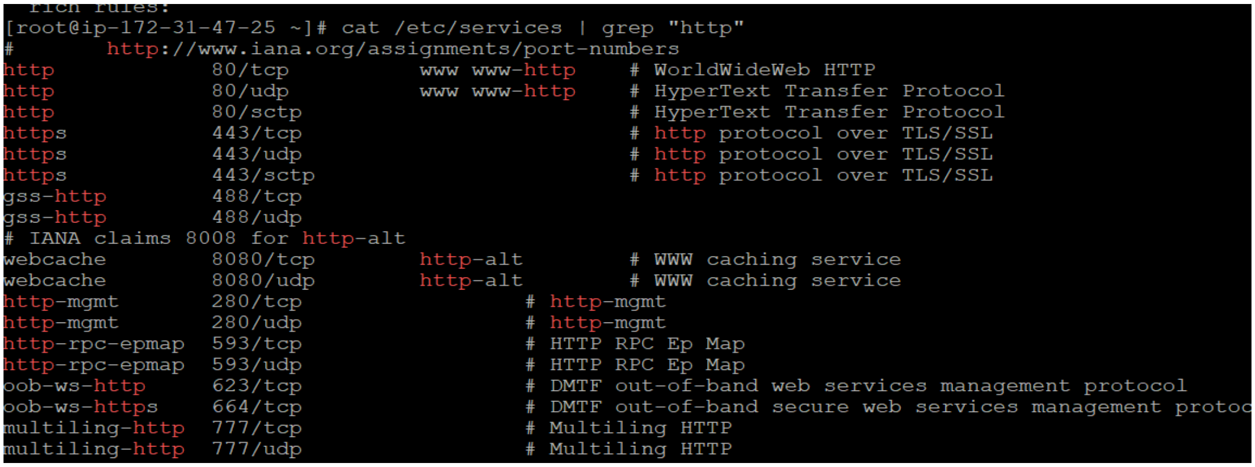
Check the http ports are listening port

cat /etc/services | grep “http”



```
[root@ip-172-31-47-25 ~]# cat /etc/services | grep "http"
```

Sample output



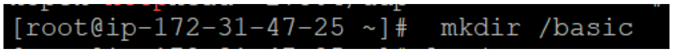
```
  rich rules:
[root@ip-172-31-47-25 ~]# cat /etc/services | grep "http"
#       http://www.iana.org/assignments/port-numbers
http            80/tcp          www www-http    # WorldWideWeb HTTP
http            80/udp          www www-http    # HyperText Transfer Protocol
http            80/sctp                         # HyperText Transfer Protocol
https           443/tcp                         # http protocol over TLS/SSL
https           443/udp                         # http protocol over TLS/SSL
https           443/sctp                        # http protocol over TLS/SSL
gss-http        488/tcp
gss-http        488/udp
# IANA claims 8008 for http-alt
webcache        8080/tcp        http-alt        # WWW caching service
webcache        8080/udp        http-alt        # WWW caching service
http-mgmt       280/tcp                 # http-mgmt
http-mgmt       280/udp                 # http-mgmt
http-rpc-epmap  593/tcp                 # HTTP RPC Ep Map
http-rpc-epmap  593/udp                 # HTTP RPC Ep Map
oob-ws-http     623/tcp                 # DMTF out-of-band web services management protocol
oob-ws-https    664/tcp                 # DMTF out-of-band secure web services management protoc
multiling-http  777/tcp                 # Multiling HTTP
multiling-http  777/udp                 # Multiling HTTP
```

Step-25

Apache document root

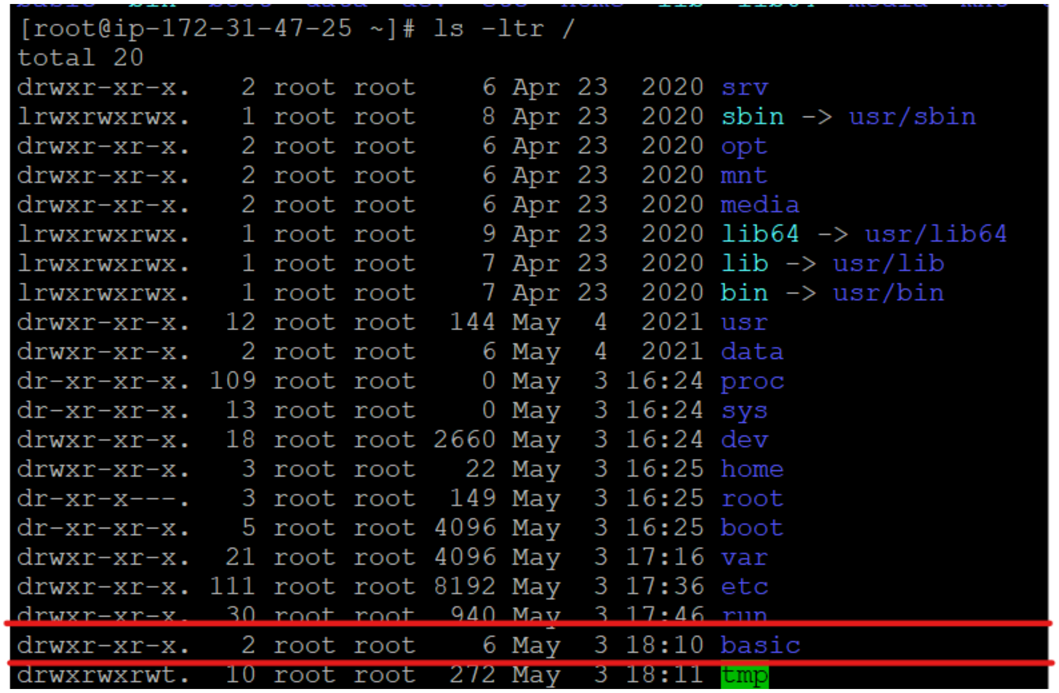
Create a one new directory

mkdir /basic



```
[root@ip-172-31-47-25 ~]#  mkdir /basic
```

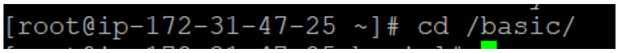
Sample output



```
[root@ip-172-31-47-25 ~]# ls -ltr /
total 20
drwxr-xr-x.   2 root root    6 Apr 23  2020 srv
lrwxrwxrwx.   1 root root    8 Apr 23  2020 sbin -> usr/sbin
drwxr-xr-x.   2 root root    6 Apr 23  2020 opt
drwxr-xr-x.   2 root root    6 Apr 23  2020 mnt
drwxr-xr-x.   2 root root    6 Apr 23  2020 media
lrwxrwxrwx.   1 root root    9 Apr 23  2020 lib64 -> usr/lib64
lrwxrwxrwx.   1 root root    7 Apr 23  2020 lib -> usr/lib
lrwxrwxrwx.   1 root root    7 Apr 23  2020 bin -> usr/bin
drwxr-xr-x.  12 root root  144 May  4  2021 usr
drwxr-xr-x.   2 root root    6 May  4  2021 data
dr-xr-xr-x. 109 root root    0 May  3 16:24 proc
dr-xr-xr-x.  13 root root    0 May  3 16:24 sys
drwxr-xr-x.  18 root root 2660 May  3 16:24 dev
drwxr-xr-x.   3 root root   22 May  3 16:25 home
dr-xr-x---.   3 root root  149 May  3 16:25 root
dr-xr-xr-x.   5 root root 4096 May  3 16:25 boot
drwxr-xr-x.  21 root root 4096 May  3 17:16 var
drwxr-xr-x. 111 root root 8192 May  3 17:36 etc
drwxr-xr-x.  30 root root  940 May  3 17:46 run
drwxr-xr-x.   2 root root    6 May  3 18:10 basic
drwxrwxrwt.  10 root root  272 May  3 18:11 tmp
```

Now change the directory

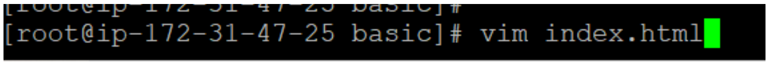
cd /basic



```
[root@ip-172-31-47-25 ~]# cd /basic/
```

Create index.html file inside the basic directory

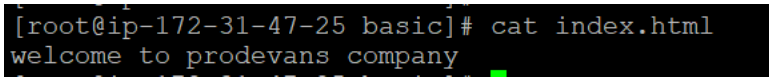
vim index.html



```
[root@ip-172-31-47-25 basic]# vim index.html
```

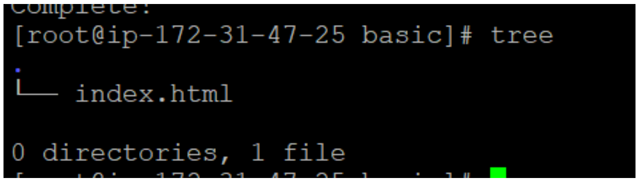
(In AWS in ec2 terminal we install vim)

Sample output



```
[root@ip-172-31-47-25 basic]# cat index.html
welcome to prodevans company
```

tree



```
Complete!
[root@ip-172-31-47-25 basic]# tree
.
└── index.html

0 directories, 1 file
```

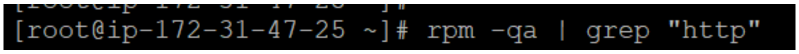
Step-26

Configuration file edit

We need the remove the /var/www/html path.add the new directory path which we created /basic

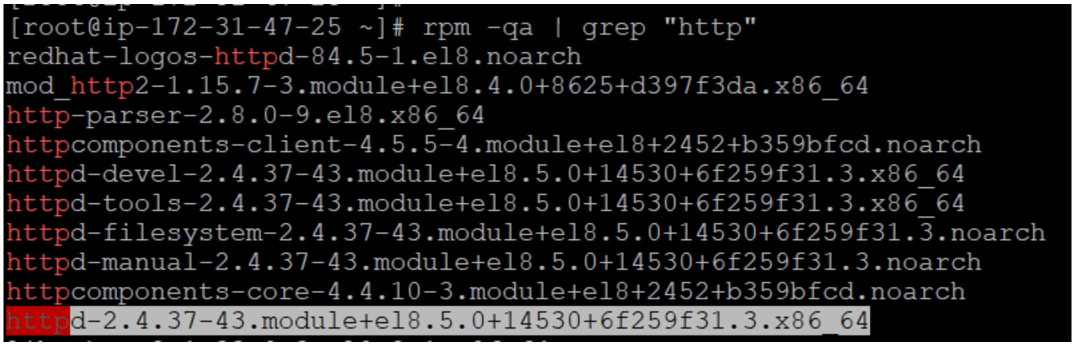
First find the httpd configuration file by using this commands

rpm -qa | grep "http"



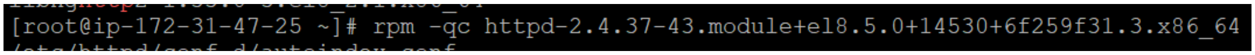
```
[root@ip-172-31-47-25 ~]# rpm -qa | grep "http"
```

Sample output



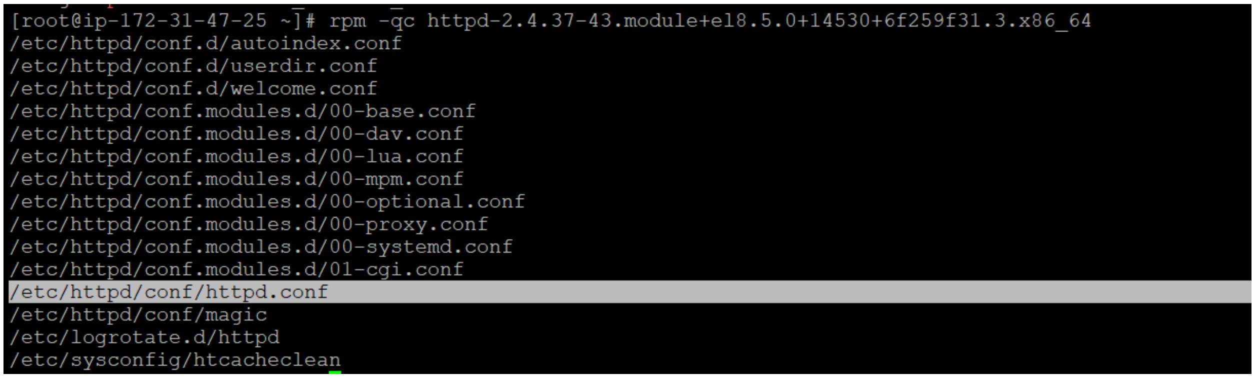
```
[root@ip-172-31-47-25 ~]# rpm -qa | grep "http"
redhat-logos-httpd-84.5-1.el8.noarch
mod_http2-1.15.7-3.module+el8.4.0+8625+d397f3da.x86_64
http-parser-2.8.0-9.el8.x86_64
httpcomponents-client-4.5.5-4.module+el8+2452+b359bfcd.noarch
httpd-devel-2.4.37-43.module+el8.5.0+14530+6f259f31.3.x86_64
httpd-tools-2.4.37-43.module+el8.5.0+14530+6f259f31.3.x86_64
httpd-filesystem-2.4.37-43.module+el8.5.0+14530+6f259f31.3.noarch
httpd-manual-2.4.37-43.module+el8.5.0+14530+6f259f31.3.noarch
httpcomponents-core-4.4.10-3.module+el8+2452+b359bfcd.noarch
httpd-2.4.37-43.module+el8.5.0+14530+6f259f31.3.x86_64
```

rpm -qc httpd-2.4.37-43.module+el8.5.0+14530+6f259f31.3.x86_64



```
[root@ip-172-31-47-25 ~]# rpm -qc httpd-2.4.37-43.module+el8.5.0+14530+6f259f31.3.x86_64
```

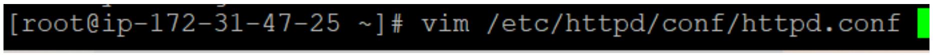
Sample output



```
[root@ip-172-31-47-25 ~]# rpm -qc httpd-2.4.37-43.module+el8.5.0+14530+6f259f31.3.x86_64
/etc/httpd/conf.d/autoindex.conf
/etc/httpd/conf.d/userdir.conf
/etc/httpd/conf.d/welcome.conf
/etc/httpd/conf.modules.d/00-base.conf
/etc/httpd/conf.modules.d/00-dav.conf
/etc/httpd/conf.modules.d/00-lua.conf
/etc/httpd/conf.modules.d/00-mpm.conf
/etc/httpd/conf.modules.d/00-optional.conf
/etc/httpd/conf.modules.d/00-proxy.conf
/etc/httpd/conf.modules.d/00-systemd.conf
/etc/httpd/conf.modules.d/01-cgi.conf
/etc/httpd/conf/httpd.conf
/etc/httpd/conf/magic
/etc/logrotate.d/httpd
/etc/sysconfig/htcacheclean
```

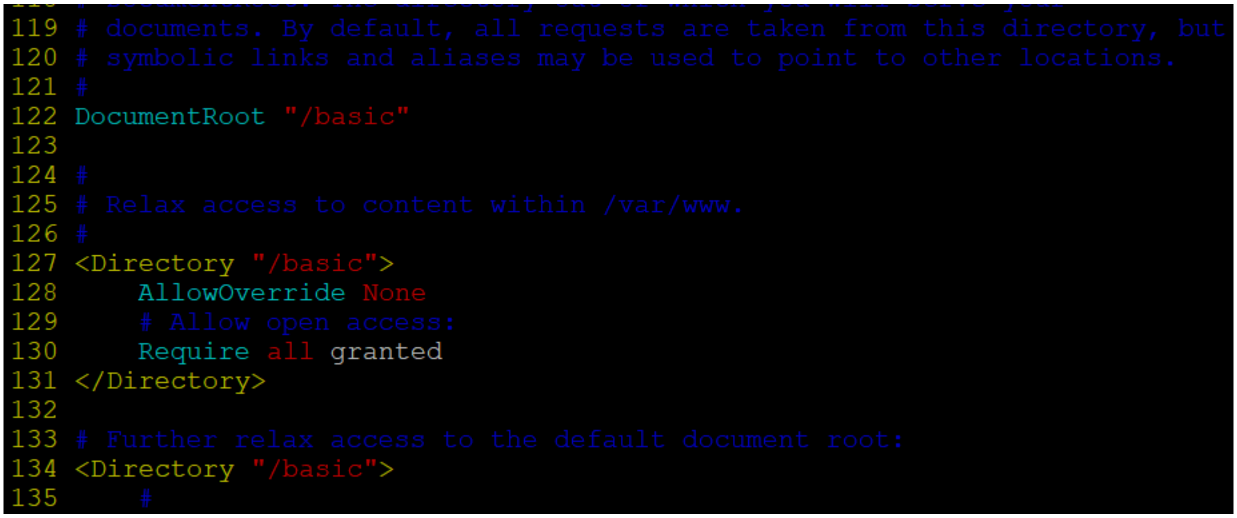
Now open the configuration file and add the basic directory

vim /etc/httpd/conf/httpd.conf



```
[root@ip-172-31-47-25 ~]# vim /etc/httpd/conf/httpd.conf
```

Edit area inside the configuration file



```
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/basic"

#
# Relax access to content within /var/www.
#
<Directory "/basic">
    AllowOverride None
    # Allow open access:
    Require all granted
</Directory>

# Further relax access to the default document root:
<Directory "/basic">
    #
```

After editing the configuration file we need to restart the service than only it will get update

# systemctl restart httpd.service

Now take the ipaddress and open in the web browser .it wont open because of selinux.

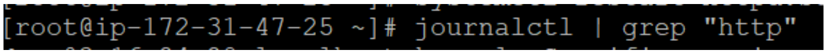
Selinux is the another layer of the security to the application

Now we need to find out the error log by using the journalctl

Use this command for logs

# jounalctl | grep “http”

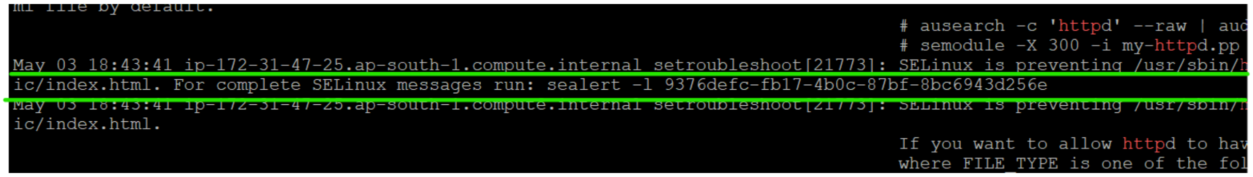
After running this command we will find some id with selinux



```
[root@ip-172-31-47-25 ~]# journalctl | grep "http"
```

Some where in middle u will find the sealert -l with some id copy that id and check

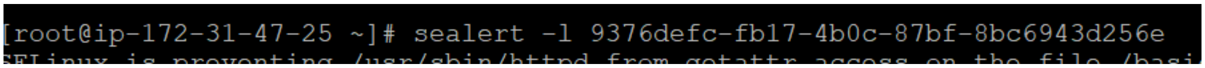
Sample output



```
mi file by default.
                                                                           # ausearch -c 'httpd' --raw | auc
                                                                           # semodule -X 300 -i my-httpd.pp
May 03 18:43:41 ip-172-31-47-25.ap-south-1.compute.internal setroubleshoot[21773]: SELinux is preventing /usr/sbin/h
ic/index.html. For complete SELinux messages run: sealert -l 9376defc-fb17-4b0c-87bf-8bc6943d256e
May 03 18:43:41 ip-172-31-47-25.ap-south-1.compute.internal setroubleshoot[21773]: SELinux is preventing /usr/sbin/h
ic/index.html.
                                                                           If you want to allow httpd to hav
                                                                           where FILE_TYPE is one of the fol
```

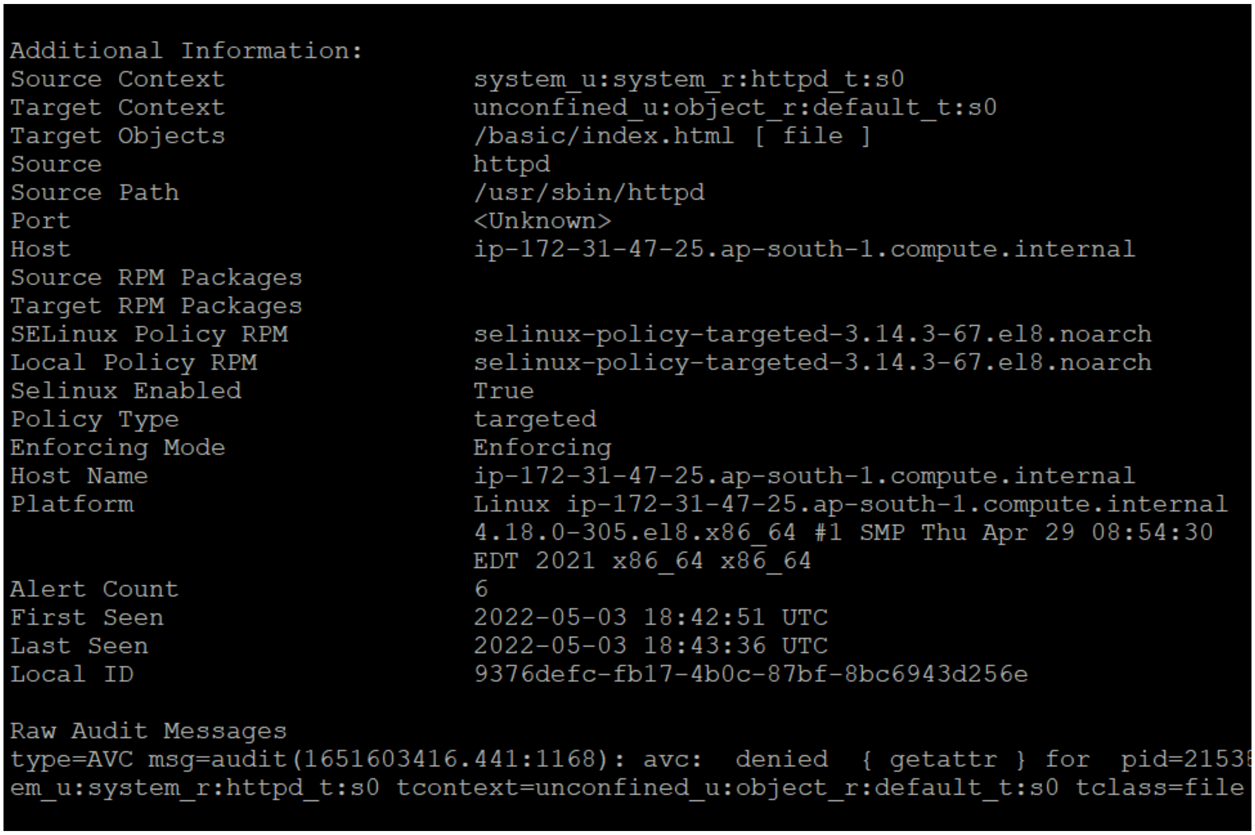
Now check

# sealert -l 9376defc-fb17-4b0c-87bf-8bc6943d256e



```
[root@ip-172-31-47-25 ~]# sealert -l 9376defc-fb17-4b0c-87bf-8bc6943d256e
```

Sample output

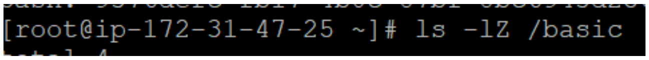


```
Additional Information:
Source Context                system_u:system_r:httpd_t:s0
Target Context                unconfined_u:object_r:default_t:s0
Target Objects                /basic/index.html [ file ]
Source                        httpd
Source Path                   /usr/sbin/httpd
Port                          <Unknown>
Host                          ip-172-31-47-25.ap-south-1.compute.internal
Source RPM Packages
Target RPM Packages
SELinux Policy RPM            selinux-policy-targeted-3.14.3-67.el8.noarch
Local Policy RPM              selinux-policy-targeted-3.14.3-67.el8.noarch
Selinux Enabled               True
Policy Type                   targeted
Enforcing Mode                Enforcing
Host Name                     ip-172-31-47-25.ap-south-1.compute.internal
Platform                      Linux ip-172-31-47-25.ap-south-1.compute.internal
                              4.18.0-305.el8.x86_64 #1 SMP Thu Apr 29 08:54:30
                              EDT 2021 x86_64 x86_64
Alert Count                   6
First Seen                    2022-05-03 18:42:51 UTC
Last Seen                     2022-05-03 18:43:36 UTC
Local ID                      9376defc-fb17-4b0c-87bf-8bc6943d256e

Raw Audit Messages
type=AVC msg=audit(1651603416.441:1168): avc:  denied  { getattr } for  pid=2153
em_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:default_t:s0 tclass=file
```

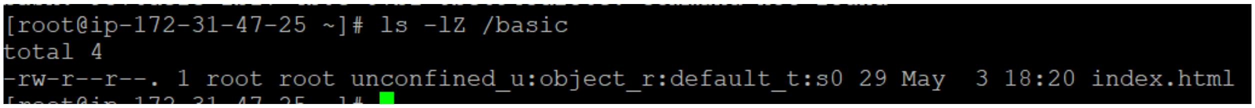
Step-27

Now check selinux context

ls -lZ /basic

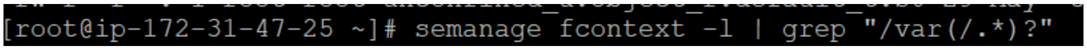


```
[root@ip-172-31-47-25 ~]# ls -lZ /basic
```



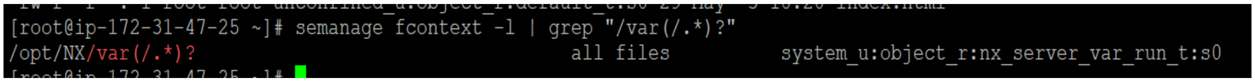
```
[root@ip-172-31-47-25 ~]# ls -lZ /basic
total 4
-rw-r--r--. 1 root root unconfined_u:object_r:default_t:s0 29 May  3 18:20 index.html
```

semanage fcontext -l | grep “/var(/.*)?”



```
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var(/.*)?"
```

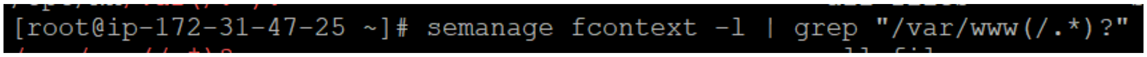
Sample output



```
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var(/.*)?"
/opt/NX/var(/.*)?                                  all files          system_u:object_r:nx_server_var_run_t:s0
```

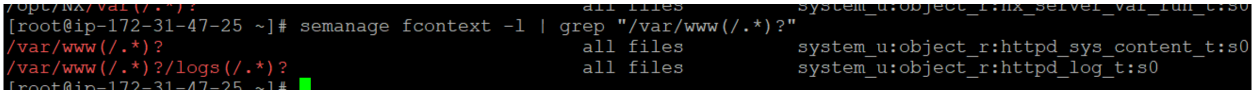
Directory + whatever will generate /create inside directory

semanage fcontext -l | grep “/var/www(/.*)?”



```
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var/www(/.*)?"
```

Sample output

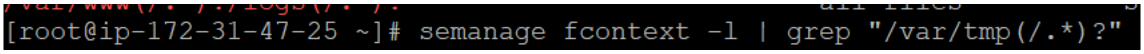


```
/opt/NX/var(/.*)?                                  all files          system_u:object_r:nx_server_var_run_t:s0
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var/www(/.*)?"
/var/www(/.*)?                                     all files          system_u:object_r:httpd_sys_content_t:s0
/var/www(/.*)?/logs(/.*)?                          all files          system_u:object_r:httpd_log_t:s0
[root@ip-172-31-47-25 ~]#
```

Step-28

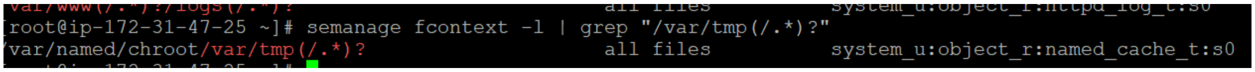
Before generate find out selinux Lable

semanage fcontext -l | grep “/var/tmp(/.*)?”



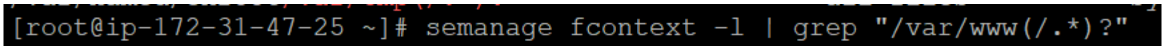
```
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var/tmp(/.*)?"
```

Sample output



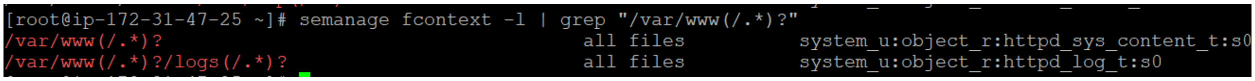
```
/var/www(/.*)?/logs(/.*)?                          all files          system_u:object_r:httpd_log_t:s0
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var/tmp(/.*)?"
/var/named/chroot/var/tmp(/.*)?                    all files          system_u:object_r:named_cache_t:s0
```

semanage fcontext -l | grep “/var/www(/.*)?”



```
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var/www(/.*)?"
```

Sample output

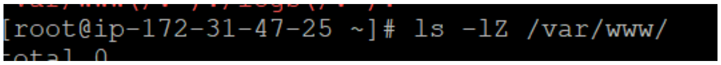


```
[root@ip-172-31-47-25 ~]# semanage fcontext -l | grep "/var/www(/.*)?"
/var/www(/.*)?                                     all files          system_u:object_r:httpd_sys_content_t:s0
/var/www(/.*)?/logs(/.*)?                          all files          system_u:object_r:httpd_log_t:s0
```

Step-29

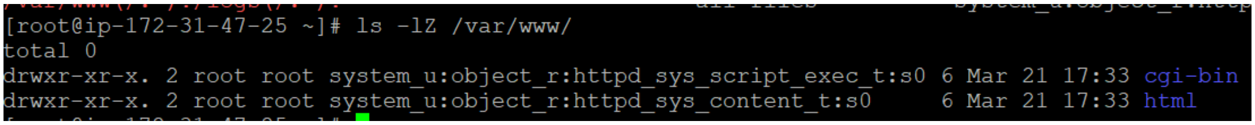
Acceptable content

ls -lZ /var/www/



```
[root@ip-172-31-47-25 ~]# ls -lZ /var/www/
```

Sample output

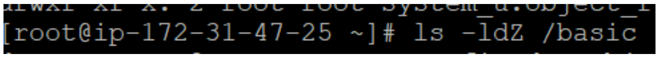


```
[root@ip-172-31-47-25 ~]# ls -lZ /var/www/
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Mar 21 17:33 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0      6 Mar 21 17:33 html
```

Step-30

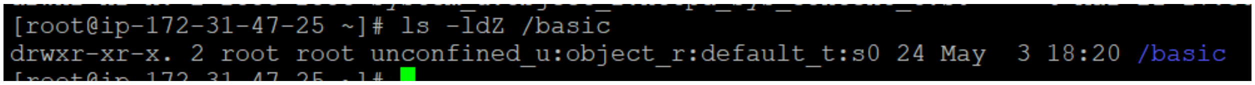
Basic content

# ls -ldZ /basic



```
[root@ip-172-31-47-25 ~]# ls -ldZ /basic
```

Sample output

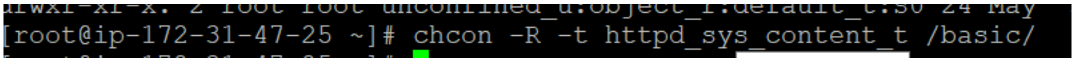


```
[root@ip-172-31-47-25 ~]# ls -ldZ /basic
drwxr-xr-x. 2 root root unconfined_u:object_r:default_t:s0 24 May  3 18:20 /basic
```

Step-31

Change the lable (or) context (this is for temporary)

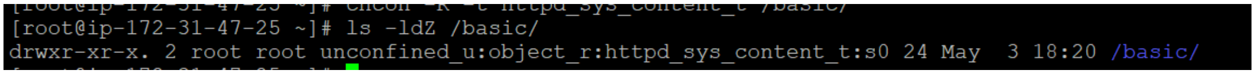
chcon -R -t httpd_sys_content_t /basic/



```
[root@ip-172-31-47-25 ~]# chcon -R -t httpd_sys_content_t /basic/
```

ls -ldZ /basic/

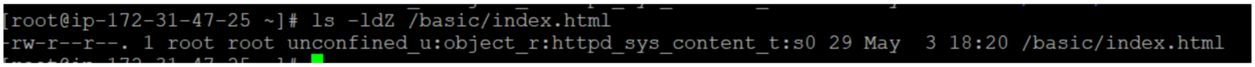
Sample output



```
[root@ip-172-31-47-25 ~]# ls -ldZ /basic/
drwxr-xr-x. 2 root root unconfined_u:object_r:httpd_sys_content_t:s0 24 May  3 18:20 /basic/
```

ls -ldZ /basic/index.html/

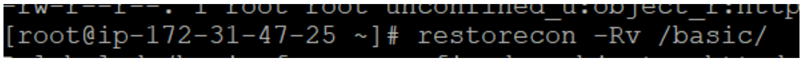
Sample output



```
[root@ip-172-31-47-25 ~]# ls -ldZ /basic/index.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 29 May  3 18:20 /basic/index.html
```

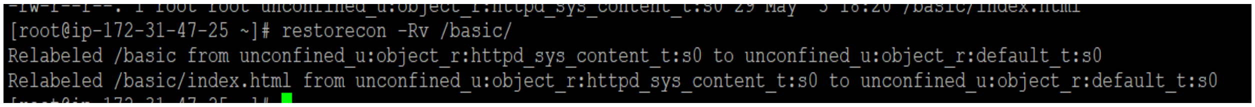
restorecon -Rv /basic/

This command will remove present httpd_sys_content and it will make to defaults one



```
[root@ip-172-31-47-25 ~]# restorecon -Rv /basic/
```

Sample output

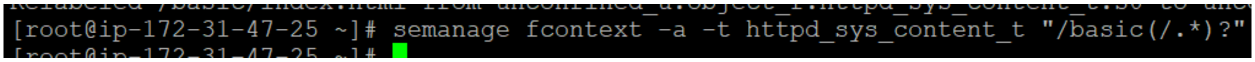


```
[root@ip-172-31-47-25 ~]# restorecon -Rv /basic/
Relabeled /basic from unconfined_u:object_r:httpd_sys_content_t:s0 to unconfined_u:object_r:default_t:s0
Relabeled /basic/index.html from unconfined_u:object_r:httpd_sys_content_t:s0 to unconfined_u:object_r:default_t:s0
```

Step-32

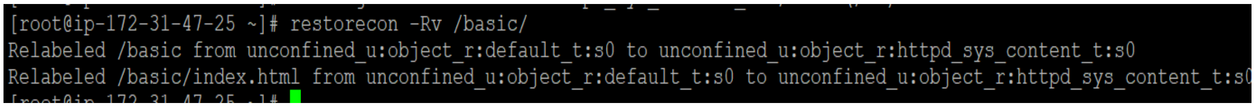
Make the lable (or)content persistent

semanage fcontext -a -t httpd_sys_content_t “/basic(/.*)?”



```
[root@ip-172-31-47-25 ~]# semanage fcontext -a -t httpd_sys_content_t "/basic(/.*)?"
```

restorecon -Rv /basic/

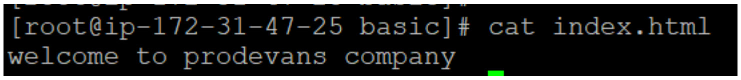


```
[root@ip-172-31-47-25 ~]# restorecon -Rv /basic/
Relabeled /basic from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0
Relabeled /basic/index.html from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0
[root@ip-172-31-47-25 ~]#
```

Step-33

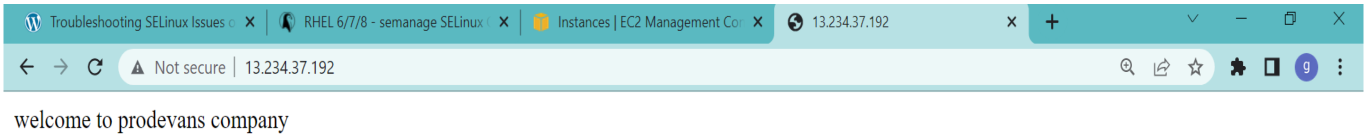
Final step

Copy the Ip address and paste in the web browser and try get what ever inside the index.html file



```
[root@ip-172-31-47-25 basic]# cat index.html
welcome to prodevans company
```

Now we got index.html content of the web page



Troubleshooting SELinux Issues o
RHEL 6/7/8 - semanage SELinux
Instances | EC2 Management Cor
13.234.37.192
Not secure
13.234.37.192

welcome to prodevans company