Apache

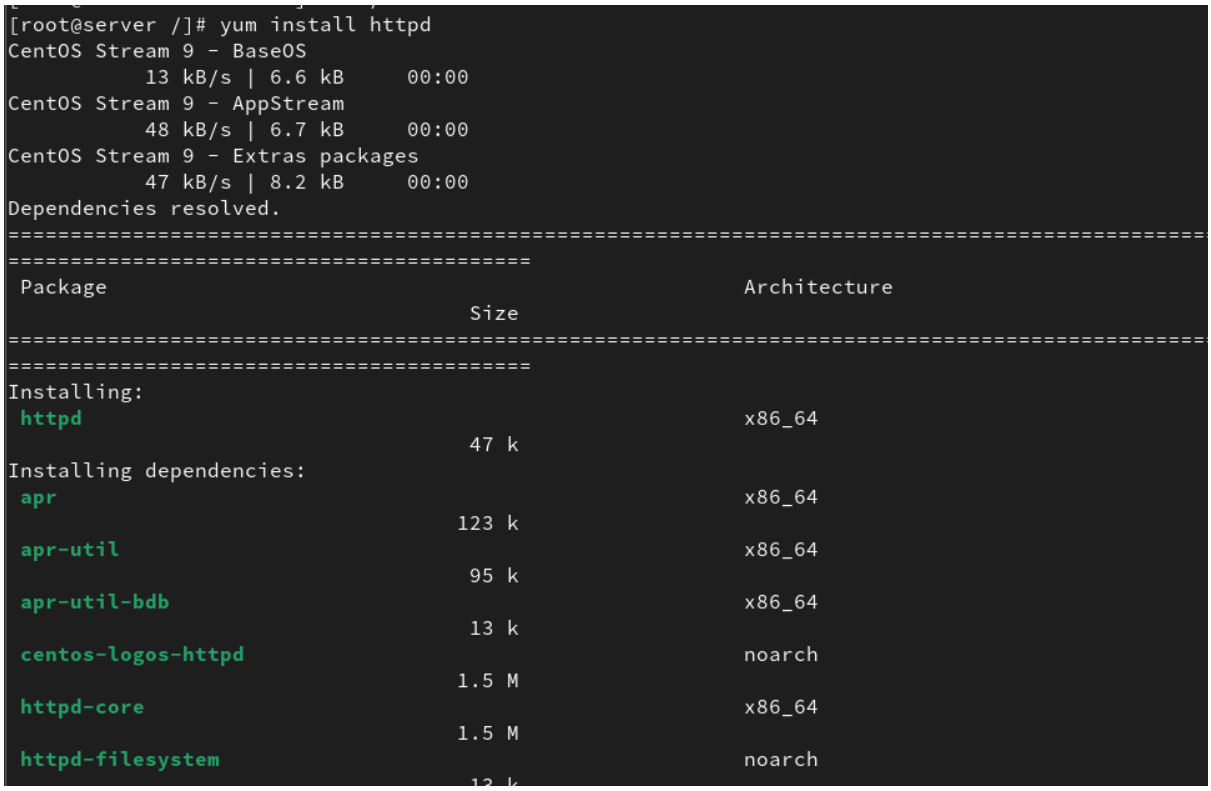
- It is web server
- http port no. – 80
- https (SSL) port no. – 443

Steps to follow:

To configure Apache:

Step-1: Install httpd package

yum install httpd



Step-2: Start and enable the service

systemctl start httpd

systemctl enable httpd

Step-3: create a new config file in httpd directory

cd /etc/httpd/conf.d

touch web.conf

```
[root@server conf.d]# touch web.conf
```

Step-4: Find and copy the virtual Host conf lines to web.conf

rpm -qa | grep httpd

rpm -ql httpd-core-2.4….. | grep vhosts

vi /usr/share/doc/httpd-core/httpd-vhosts.conf

```
[root@server conf.d]# rpm -qa | grep httpd
httpd-tools-2.4.62-1.el9.x86_64
httpd-filesystem-2.4.62-1.el9.noarch
httpd-core-2.4.62-1.el9.x86_64
centos-logos-httpd-90.8-1.el9.noarch
httpd-2.4.62-1.el9.x86_64
httpd-devel-2.4.62-1.el9.x86_64
httpd-manual-2.4.62-1.el9.noarch
[root@server conf.d]# rpm -ql httpd-core-2.4.62-1.el9.x86_64 | grep host
/usr/lib64/httpd/modules/mod_authz_host.so
/usr/lib64/httpd/modules/mod_vhost_alias.so
/usr/share/doc/httpd-core/httpd-vhosts.conf
[root@server conf.d]# vi /usr/share/doc/httpd-core/httpd-vhosts.conf
```

Step-5: Copy the conf lines and paste to web.conf

vi web.conf

- restart the service

systemctl restart httpd

```
[root@server conf.d]# vi web.conf
```

```
<VirtualHost *:80>
    ServerAdmin server.com
    DocumentRoot "/var/www/web"
    ServerName web.server.com
    ErrorLog "/var/log/httpd/web.server.com-error_log"
    CustomLog "/var/log/httpd/web.server.com-access_log" common
</VirtualHost>
```

```
[root@server conf.d]# systemctl restart httpd
```

Step-6: Go to /var/www/ and create web directory and index.html within the web

cd /var/www

mkdir web

cd web

echo “This is html file” > index.com

```
[root@server /]# cd /var/www/
[root@server www]# ll
total 0
drwxr-xr-x. 2 root root 6 Aug 12 18:50 cgi-bin
drwxr-xr-x. 2 root root 6 Aug 12 18:50 html
[root@server www]# mkdir web
[root@server www]# cd web/
[root@server web]# ll
total 0
[root@server web]# echo "Hi this is html file" > index.html
[root@server web]# cat index.html
Hi this is html file
```

Step-7: add IP address to host

vi /etc/hosts

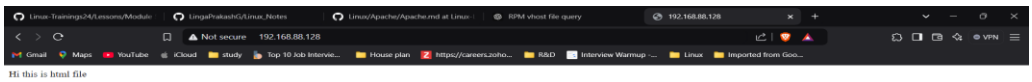
Add your IP address

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.88.128 server.com
```

Step-8: curl http:\\server.com

```
[root@server www]# curl http://server.com
Hi this is html file
```

Also try in the browser too.



NameBased Virtual Hosting

Change the conf file

vi /etc/httpd/conf.d/web.conf

Add the virtual host

```
<VirtualHost *:80>
    ServerAdmin server.com
    DocumentRoot "/var/www/web"
    ServerName web.server.com
    ErrorLog "/var/log/httpd/web.server.com-error_log"
    CustomLog "/var/log/httpd/web.server.com-access_log" common
</VirtualHost>
<VirtualHost *:80>
    ServerAdmin server1.com
    DocumentRoot "/var/www/web1"
    ServerName web1.server.com
    ErrorLog "/var/log/httpd/web1.server.com-error_log"
    CustomLog "/var/log/httpd/web1.server.com-access_log" common
</VirtualHost>
~
```

Add in hosts

vi /etc/hosts

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.88.128 server.com
192.168.88.128 server1.com
~
```

Restart the service

systemctl restart httpd

curl http://server1.com

curl http://server.com

```
[root@server web1]# curl http://server1.com
Hi this is html file
[root@server web1]# curl http://server.com
Hi this is html file
```

IP-Based virtual hosting

Step-1: Add the new network adopter in VM machine

Settings → Add → Network Adopter

ifconfig

nmcli device connect

```
[root@server /]# nmcli device connect ens36
Device 'ens36' successfully activated with '24c1b8c8-5e5b-4f9c-b9a0-152560120ded'.
```

```
ens36: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.88.129  netmask 255.255.255.0  broadcast 192.168.88.255
        inet6 fe80::bbbe:4951:2fc2:f90  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:cc:ff:8e  txqueuelen 1000  (Ethernet)
        RX packets 22  bytes 1888 (1.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 20  bytes 3665 (3.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Step-2: Edit the main configuration file httpd.conf

vi /etc/httpd/conf/httpd.conf

```
[root@server /]# vi /etc/httpd/conf/httpd.conf
```

Add the IP address, comment the unused ports

```
#Listen 80
Listen 192.168.88.128:80
Listen 192.168.88.129:80
```

Step-3: create a new config file in httpd directory

cd /etc/httpd/conf.d

```
[root@server /]# cd /etc/httpd/conf.d
```

touch ip_based.conf

```
[root@server conf.d]# touch ip_based.conf
```

Step-4: Find and copy the virtual Host conf lines to web.conf

rpm -qa | grep httpd

rpm -ql httpd-core-2.4….. | grep vhosts

vi /usr/share/doc/httpd-core/httpd-vhosts.conf

```
[root@server conf.d]# rpm -qa | grep httpd
httpd-tools-2.4.62-1.el9.x86_64
httpd-filesystem-2.4.62-1.el9.noarch
httpd-core-2.4.62-1.el9.x86_64
centos-logos-httpd-90.8-1.el9.noarch
httpd-2.4.62-1.el9.x86_64
httpd-devel-2.4.62-1.el9.x86_64
httpd-manual-2.4.62-1.el9.noarch
[root@server conf.d]# rpm -ql httpd-core-2.4.62-1.el9.x86_64 | grep host
/usr/lib64/httpd/modules/mod_authz_host.so
/usr/lib64/httpd/modules/mod_vhost_alias.so
/usr/share/doc/httpd-core/httpd-vhosts.conf
[root@server conf.d]# vi /usr/share/doc/httpd-core/httpd-vhosts.conf
```

Step-5: Copy the conf lines and paste to ip_based.conf

vi ip_based.conf

```
<VirtualHost 192.168.88.128:80>
    ServerAdmin server.com
    DocumentRoot "/var/www/ip1"
    ServerName web.server.com
    ErrorLog "/var/log/httpd/ip1.server.com-error_log"
    CustomLog "/var/log/httpd/ip1.server.com-access_log" common
</VirtualHost>
<VirtualHost 192.168.88.129:80>
    ServerAdmin server1.com
    DocumentRoot "/var/www/ip2"
    ServerName web1.server.com
    ErrorLog "/var/log/httpd/ip2.server.com-error_log"
    CustomLog "/var/log/httpd/ip2.server.com-access_log" common
</VirtualHost>
```

- restart the service

systemctl restart httpd

```
[root@server conf.d]# systemctl restart httpd
```

Step-6: Go to /var/www/ and create web directory and index.html within the ip1 and ip2

cd /var/www

mkdir ip1

```
[root@server www]# mkdir ip1 ip2
[root@server www]# cd ip1/
[root@server ip1]# echo "This is ip based virutal host IP1" > index.html
[root@server ip1]# cd ..
[root@server www]# cd ip2
[root@server ip2]# echo "This is ip based virutal host IP2" > index.html
```

Step-7: Check in browser or use curl

curl http://192.168.88.128

curl http://192.168.88.129

```
[root@server ip2]# curl http://192.168.88.128
This is ip based virutal host IP1
[root@server ip2]# curl http://192.168.88.129
This is ip based virutal host IP2
```

HTTPS (SSL)

Step-1: Install openssl package

yum install openssl

```
[root@server /]# yum install openssl
CentOS Stream 9 - BaseOS
CentOS Stream 9 - AppStream
CentOS Stream 9 - Extras packages
Package openssl-1:3.2.2-6.el9.x86_64 is already installed.
Dependencies resolved.
```

Step-2: Generate private key

openssl genpkey -algorithm RSA -out pri-key.key

```
[root@server /]# openssl genpkey -algorithm RSA -out prikey.key
```

Step-3: Generate certificate

openssl req -new -x509 -key pri-key.key -out ssl_certificate.crt -days 365

```
[root@server /]# openssl req -new -x509 -key pri-key.key -out ssl_certificate.crt -days 365
```

IN

KA

Banglore

#

server.com

Step-4: Change ownship permission

chmod 600 pri-key.key

```
-rw-------. 1 root root 1704 Nov 26 12:09 pri-key.key
```

Configure SSL to apache

Step-5: Install mod_ssl openssl package

yum install mod_ssl openssl

```
[root@server certs]# yum install mod_ssl openssl
Last metadata expiration check: 0:00:33 ago on Wednesday 27 November 2024 12:30:25 PM.
Package mod_ssl-1:2.4.62-1.el9.x86_64 is already installed.
Package openssl-1:3.2.2-6.el9.x86_64 is already installed.
```

Step-6: Set up the certificate

mv pri-key /etc/pki/tls/private

mv ssl_certificate.crt /etc/pki/tls/certs

```
[root@server private]# ll
total 8
-rw-------. 1 root root 3272 Nov 20 10:16 postfix.key
-rw-------. 1 root root 1704 Nov 26 12:09 pri-key.key
[root@server private]# pwd
/etc/pki/tls/private
```

```
[root@server certs]# ll
total 8
lrwxrwxrwx. 1 root root   49 Aug 19 19:16 ca-bundle.crt -> /etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem
lrwxrwxrwx. 1 root root   55 Aug 19 19:16 ca-bundle.trust.crt -> /etc/pki/ca-trust/extracted/openssl/ca-bundle.trust.crt
-rw-r--r--. 1 root root 2155 Nov 20 10:16 postfix.pem
-rw-r--r--. 1 root root 1281 Nov 26 12:12 ssl_certificate.crt
[root@server certs]# pwd
/etc/pki/tls/certs
```

Step-7: Configure apache to use ssl

vi /etc/httpd/conf.d/ssl.conf

Change the conf file

#SSLCertificateFile /etc/pki/tls/certs/ssl_certificate.crt

#SSLCertificateKeyFile /etc/pki/tls/private/pri-key.key

```
SSLCertificateFile /etc/pki/tls/certs/ssl_certificate.crt

#   Server Private Key:
#   If the key is not combined with the certificate, use this
#   directive to point at the key file.  Keep in mind that if
#   you've both a RSA and a DSA private key you can configure
#   both in parallel (to also allow the use of DSA ciphers, etc.)
#   ECC keys, when in use, can also be configured in parallel
SSLCertificateKeyFile /etc/pki/tls/private/pri-key.key
```

Step-8: Restart the service

```
# systemctl restart httpd
```

Step-9: Open a web browser and navigate to your server using the https

https://192.168.88.128