Apache Web Server
Hosting Multiple Web SItes

Install Apache Web Server

To install apache web server on a Redhat Based system use following command.

sudo yum install httpd -y

```
[admin@demosrv ~]$ sudo yum install httpd -y
```

Once the apache web server is installed, its default configuration file httpd.conf is stored in the /etc/httpd/conf directory.

Here we have to configure Apache web server to host multiple web sites on the same server.
The websites will use the same IP address and same port 80. This is possible only with virtual host configuration of Apache server. Thus we need a third parameter which will identify each web site uniquely. This parameter is the hostname which is the name of the website like www.abc.lab . The Apache web server will identify user request based on the name provided by the user. Thus here the websites should be accessed using names only and not the IP address.

Configure Virtual Hosting

Here we are going to host 2 web sites. The first website will be accessed by name www.demo.lab. and the second website will be accessed by name www.example.lab .

The default configuration specified in the /etc/httpd/conf/httpd.conf file makes the Apache web server to consider the configuration files in the /etc/httpd/conf.d directory. The extension of the file should be .conf. Thus you will create a file by name virtualhosts.conf in this directory.

sudo touch /etc/httpd/conf.d/virtualhosts.conf

```
[admin@demosrv html]$ sudo touch /etc/httpd/conf.d/virtualhosts.conf
```

Now we will create the directories for these web sites. These directories will contain the web pages for these web sites. We will create the directories based on the name of the websites. This makes easy to identify and manage the web site data.

sudo mkdir /var/www/demo.lab

sudo mkdir /var/www/example.lab

```
[admin@demosrv html]$ sudo mkdir /var/www/demo.lab
[sudo] password for admin:
[admin@demosrv html]$ sudo mkdir /var/www/example.lab
[admin@demosrv html]$ _
```

Now we will go to these directories and create index.html page.

cd /var/www/demo.lab

sudo vi index.html

Type "**Welcome to demo.lab website!**"

Save the file.

Similarly

cd /var/www/example.lab

sudo vi index.html

Type "**Welcome to example.lab website**!"

Save the file.

Now you will configure virtual hosting so that Apache server will host these 2 websites.

You need to edit the /etc/httpd/conf.d/virtualhosts.conf file and add the configuration.

sudo vi /etc/httpd/conf.d/virtualhosts.conf

Add following configuration

<VirtualHost *:80>
ServerName www.demo.lab
ServerAdmin root@demo.lab
DocumentRoot /var/www/demo.lab/
ErrorLog /var/log/httpd/demo.lab-error.log
CustomLog /var/log/httpd/demo.lab-custom.log combined
</VirtualHost>

<VirtualHost *:80>
ServerName www.example.lab
ServerAdmin admin@example.lab
DocumentRoot /var/www/example.lab/
ErrorLog /var/log/httpd/example.lab-error.log
CustomLog /var/log/httpd/example.lab-custom.log combined
</VirtualHost>

```
<VirtualHost  *:80>
ServerName    www.demo.lab
ServerAdmin   root@demo.lab
DocumentRoot  /var/www/demo.lab/
ErrorLog      /var/log/httpd/demo.lab-error.log
CustomLog     /var/log/httpd/demo.lab-custom.log combined
</VirtualHost>

<VirtualHost  *:80>
ServerName    www.example.lab
ServerAdmin   admin@example.lab
DocumentRoot  /var/www/example.lab/
ErrorLog      /var/log/httpd/exaple.lab-error.log
CustomLog     /var/log/httpd/example.lab-custom.log combined
</VirtualHost>
~
```

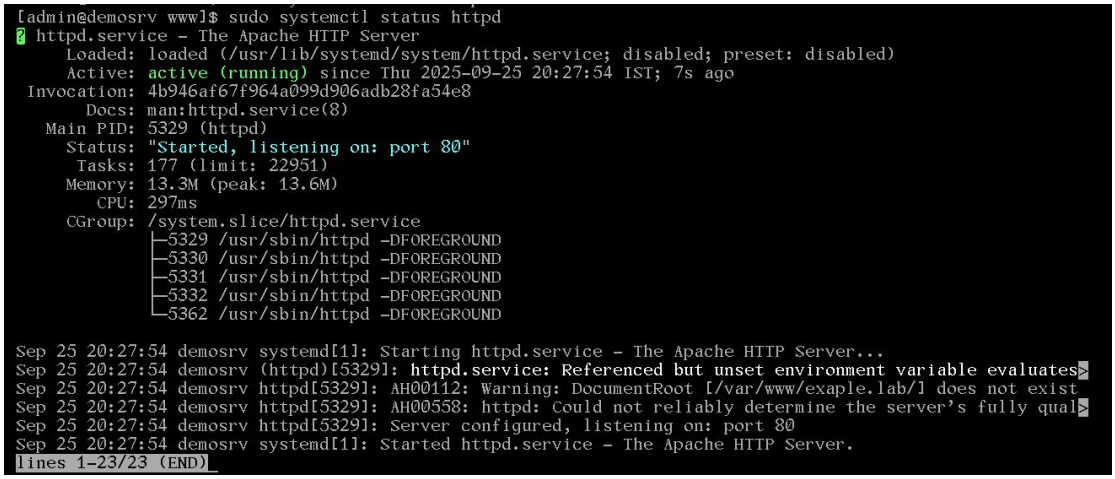
Save the file.

Now start/restart the httpd service.
sudo systemctl start httpd

If the service is already running then restart it.
sudo systemctl restart httpd

Verify that the service started successfully.

sudo systemctl status httpd



In case if there is any syntax error, the service will not start. Give following command to find out the error.

sudo journalctl -xeu httpd

Read the output carefully.

Once the service start successfully, you need to check if the websites are accessible.

However you know that the websites need to be accessed using there names www.demo.lab and www.example.lab.

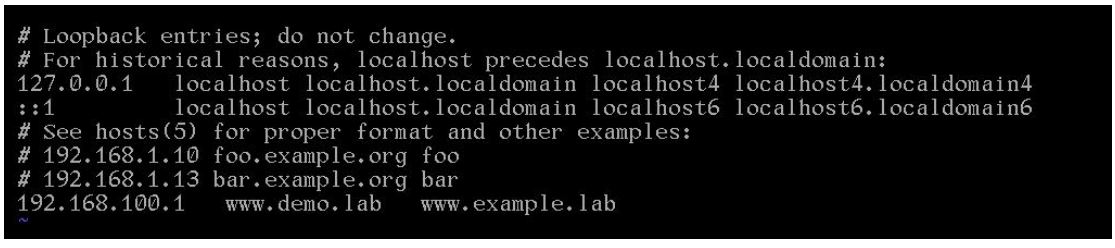
But the computer requires the IP address to access the web server. Here we will use the /etc/hosts file as the DNS server.

sudo vi /etc/hosts

Add following line at the end of the file.

192.168.100.1 www.demo.lab www.example.com

The file will look like this



Save the file.

Now check if the site opens using

curl **www.demo.lab**

```
[admin@demosrv www]$ curl www.demo.lab
Welcome to demo.lab website!
[admin@demosrv www]$
```

curl **www.example.com**

```
[admin@demosrv www]$ curl www.example.lab
Welcome to example.com website!
[admin@demosrv www]$ _
```

If you get this output for both the commands it means the web sites are working.

Great Job!!!

You are a Web Hosting Service Provider…….. If you add one more web site which will be accessed by name **www.hack.lab** !!!