

Task-1: Setting up an Apache web server

Step-1: Installing Apache

```
mustakin@DESKTOP-V9TQVMG:/mnt$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64
  liblua5.4-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser ufw
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  libaprutil1t64 liblua5.4-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 193 not upgraded.
Need to get 2086 kB of archives.
After this operation, 8090 kB of additional disk space will be used.
```

Step-2: Adjusting firewall

i. Install ufw in the system

```
mustakin@DESKTOP-V9TQVMG:/mnt$ sudo apt install ufw -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  iptables libip4tc2 libip6tc2 libnetfilter-contrack3 libnfnetwork0 libnftables1 libnftnl11 nftables
Suggested packages:
  firewalld
The following NEW packages will be installed:
  iptables libip4tc2 libip6tc2 libnetfilter-contrack3 libnfnetwork0 libnftables1 libnftnl11 nftables ufw
0 upgraded, 9 newly installed, 0 to remove and 193 not upgraded.
Need to get 1152 kB of archives.
After this operation, 5144 kB of additional disk space will be used.
```

ii. List the ufw application profiles by typing: `sudo ufw app list`

```
mustakin@DESKTOP-V9TQVMG:/mnt$ sudo ufw app list
Available applications:
  Apache
  Apache Full
  Apache Secure
```

iii. Allow traffic on port 80 to configure ssl in our server

```
mustakin@DESKTOP-V9TQVMG:/mnt$ sudo ufw allow 'Apache'
Rule added
Rule added (v6)
```

iv. Now check the ufw status. Now We can see HTTP traffic are allowed

```
mustakin@DESKTOP-V9TQVMG:/mnt$ sudo ufw status
Status: active
```

To	Action	From
--	-----	-----
Apache Full	ALLOW	Anywhere
Apache	ALLOW	Anywhere
Apache Full (v6)	ALLOW	Anywhere (v6)
Apache (v6)	ALLOW	Anywhere (v6)

Step-3: Checking Web Server

- i. Check if apache2 has been installed or not.

```
mustakin@DESKTOP-V9TQVMG:/mnt$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
  Active: active (running) since Fri 2025-11-07 23:27:43 +06; 10min ago
    Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1297 (apache2)
     Tasks: 55 (limit: 8201)
    Memory: 5.4M (peak: 7.1M)
      CPU: 115ms
     CGroup: /system.slice/apache2.service
             ├─1297 /usr/sbin/apache2 -k start
             ├─1299 /usr/sbin/apache2 -k start
             └─1300 /usr/sbin/apache2 -k start

Nov 07 23:27:43 DESKTOP-V9TQVMG systemd[1]: Starting apache2.service - The Apache HTTP Server...
Nov 07 23:27:43 DESKTOP-V9TQVMG systemd[1]: Started apache2.service - The Apache HTTP Server.
```



Task-2: Setting up virtual hosts

Step 1 — Manage Apache

Useful commands:

```
sudo systemctl stop apache2  
sudo systemctl start apache2  
sudo systemctl restart apache2  
sudo systemctl reload apache2  
sudo systemctl enable apache2  
sudo systemctl disable apache2
```

Step 2 — Create First Virtual Host (`example.com`)

1. Create directory structure

```
sudo mkdir -p /var/www/example.com/html  
sudo chown -R $USER:$USER /var/www/example.com/html  
sudo chmod -R 755 /var/www/example.com
```

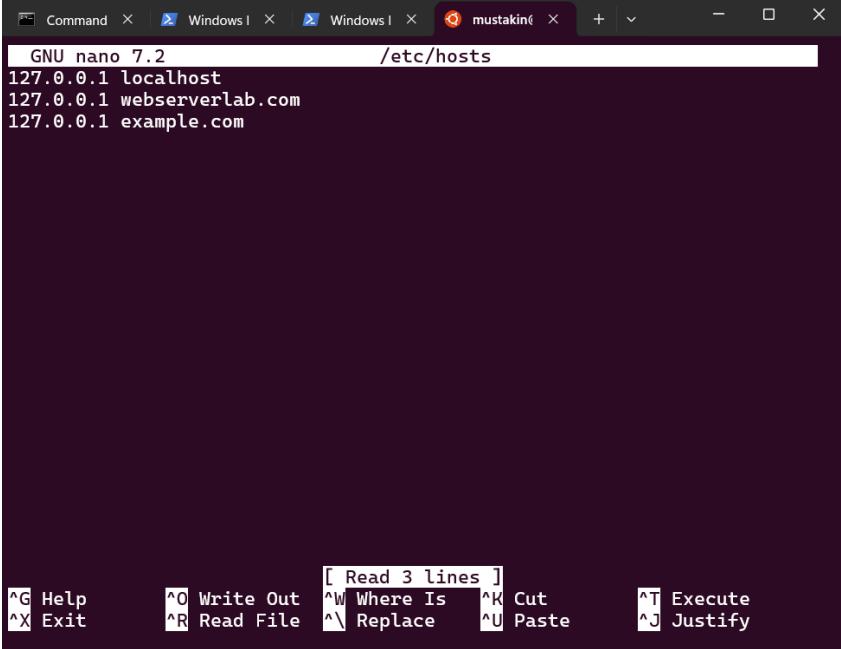
2. Create web page

Command: `nano /var/www/example.com/html/index.html`

```
html>
```

```
<head><title>Welcome to Example.com!</title></head>  
<body><h1>Success! The example.com virtual host is working!</h1></body>  
</html>
```

3. Add domain mapping



```
GNU nano 7.2          /etc/hosts
127.0.0.1 localhost
127.0.0.1 webserverlab.com
127.0.0.1 example.com

^G Help      ^O Write Out  [ Read 3 lines ]  ^W Where Is  ^K Cut      ^T Execute
^X Exit      ^R Read File   ^\ Replace    ^U Paste    ^J Justify
```

```
mustakin@DESKTOP-V9TQVMG:~$ cat /etc/hosts
127.0.0.1 localhost
127.0.0.1 webserverlab.com
127.0.0.1 example.com
mustakin@DESKTOP-V9TQVMG:~$ ping example.com
PING example.com (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.223 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.143 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.120 ms
```

Checking if it's mapped or not

4. Going to the domain: <http://example.com> and <http://webserverlab.com>

```
mustakin@DESKTOP-V9TQVMG:~$ curl http://example.com
<html>
  <head><title>Welcome to Example.com!</title></head>
  <body><h1>Success! The example.com virtual host is working!</h1></body>
</html>
```

```
mustakin@DESKTOP-V9TQVMG:~$ curl http://webserverlab.com
<html>
  <head><title>Welcome to Example.com!</title></head>
  <body><h1>Success! The example.com virtual host is working!</h1></body>
</html>
```