

Assignment Sheet

Assignment 1: How to upload HTML web pages on Apache2 web server in EC-2 Instance? Please justify with step-by-step answers.

→ **Ans:**

We can install the Apache webserver in EC2 instance with the help of following command and can check the status as well.

Installation command: `sudo apt-get install apache2`

```
root@ip-172-31-87-5:~# sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/97.8 kB of archives.
After this operation, 546 kB of additional disk space will be used.
Selecting previously unselected package apache2.
(Reading database ... 64382 files and directories currently installed.)
Preparing to unpack .../apache2_2.4.52-1ubuntu4.1_amd64.deb ...
Unpacking apache2 (2.4.52-1ubuntu4.1) ...
Setting up apache2 (2.4.52-1ubuntu4.1) ...
apache-htcacheclean.service is a disabled or a static unit not running, not starting it.
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for ufw (0.36.1-4build1) ...
Rules updated for profile 'Apache Full'

Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.
```

Apache2 status check command: `systemctl status apache2`

```
root@ip-172-31-88-233:~# sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-11-11 14:37:19 UTC; 6min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 1875 (apache2)
    Tasks: 55 (limit: 1143)
   Memory: 5.0M
      CPU: 52ms
  CGroup: /system.slice/apache2.service
          └─1875 /usr/sbin/apache2 -k start
             └─1877 /usr/sbin/apache2 -k start
                └─1878 /usr/sbin/apache2 -k start

Nov 11 14:37:19 ip-172-31-88-233 systemd[1]: Starting The Apache HTTP Server...
Nov 11 14:37:19 ip-172-31-88-233 systemd[1]: Started The Apache HTTP Server.
```

Before testing Apache, it is necessary to modify the firewall settings to allow outside access to the default web ports. By default, we would have a UFW firewall configured to restrict access to your server. During installation, Apache registers itself with UFW to provide a few application profiles that can be used to enable or disable access to Apache through the firewall.

List the ufw application profiles by running the following command:

```
sudo ufw app list
```

As indicated by the below output, there are three profiles available for Apache:

Apache: This profile opens only port 80 (normal, unencrypted web traffic)

Apache Full: This profile opens both port 80 (normal, unencrypted web traffic) and port 443 (TLS/SSL encrypted traffic)

Apache Secure: This profile opens only port 443 (TLS/SSL encrypted traffic)

```
root@ip-172-31-88-233:~# sudo ufw app list
Available applications:
  Apache
  Apache Full
  Apache Secure
  OpenSSH
root@ip-172-31-88-233:~# sudo ufw allow 'Apache'
Rules updated
Rules updated (v6)
```

It is recommended that you enable the most restrictive profile that will still allow the traffic we have configured. Since we have not configured SSL for our server yet in this guide, we will only need to allow traffic on port 80:

```
sudo ufw allow 'Apache'
```

```
ubuntu@ip-172-31-87-5:/var/www/html$ ls -ltr; pwd; date
total 12
-rw-r--r-- 1 root root 10671 Nov 11 13:59 index.html
/var/www/html
Fri Nov 11 14:09:17 UTC 2022
ubuntu@ip-172-31-87-5:/var/www/html$
```

Check the ufw status and if it is not active then enable it.

```
sudo ufw status
```

```
sudo ufw enable
```

```
root@ip-172-31-88-233:~# sudo ufw status
Status: inactive
root@ip-172-31-88-233:~# sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@ip-172-31-88-233:~# sudo ufw status
Status: active
```

```

root@ip-172-31-88-233:~# sudo ufw status
Status: active

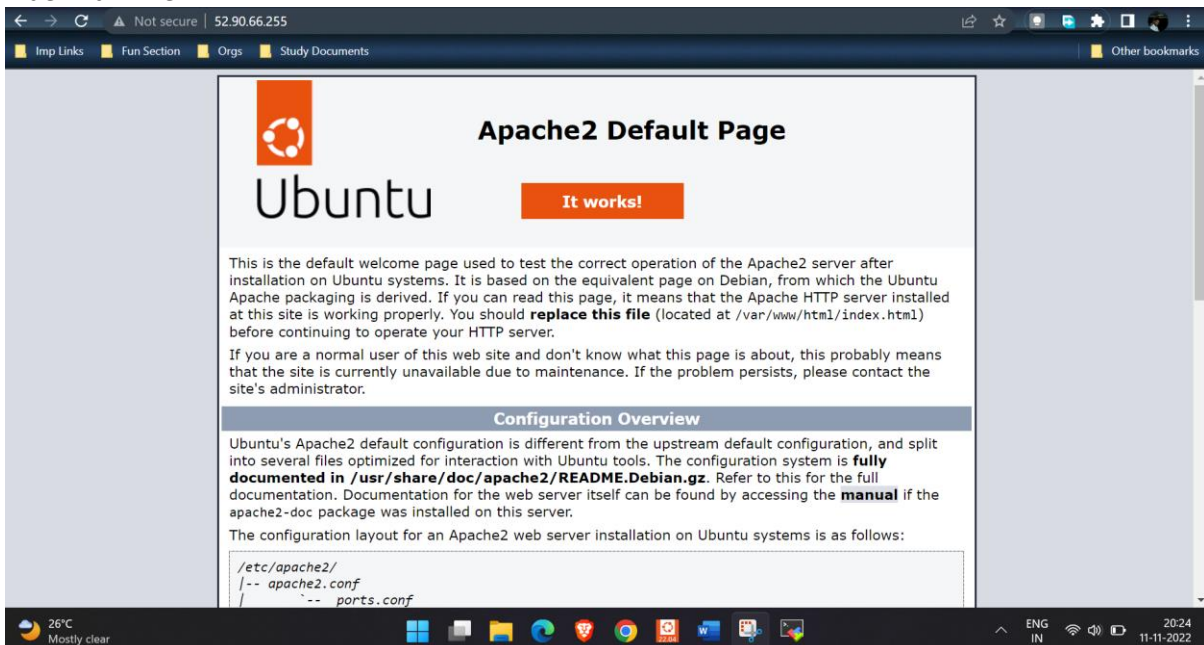
To Action From
--
Apache Full ALLOW Anywhere
Apache ALLOW Anywhere
Apache Full (v6) ALLOW Anywhere (v6)
Apache (v6) ALLOW Anywhere (v6)

root@ip-172-31-88-233:~# sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-11-11 14:37:19 UTC; 8min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 1875 (apache2)
    Tasks: 55 (limit: 1143)
   Memory: 5.0M
      CPU: 61ms
   CGroup: /system.slice/apache2.service
           └─1875 /usr/sbin/apache2 -k start
             └─1877 /usr/sbin/apache2 -k start
               └─1878 /usr/sbin/apache2 -k start

Nov 11 14:37:19 ip-172-31-88-233 systemd[1]: Starting The Apache HTTP Server...
Nov 11 14:37:19 ip-172-31-88-233 systemd[1]: Started The Apache HTTP Server.
root@ip-172-31-88-233:~# curl -4 icanhazip.com
52.90.66.255
root@ip-172-31-88-233:~#

```

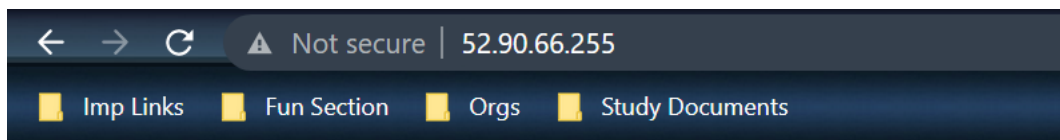
The default document root file path is `/var/www/html` from where it displays the content from `index.html` file.



To upload the HTML web pages on Apache2 webserver in EC2 instance, we can connect to this server via MobaXterm software by providing the public IP address, user name ubuntu and key pair file `.pem`. Once you have connected the EC2 instance we can upload the HTML files in `/var/www/html` folder or we can customize the `index.html` file as per our need, sample given below.

```
52.90.66.255 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
/var/www/html/
Name
index.html
root@ip-172-31-88-233:/var/www/html# ls -ltr; pwd; date
total 12
-rw-r--r-- 1 root root 10671 Nov 11 14:37 index.html
/var/www/html
Fri Nov 11 15:00:26 UTC 2022
root@ip-172-31-88-233:/var/www/html#
```

```
root@ip-172-31-88-233:/var/www/html# ls -ltr; pwd; date
total 12
-rw-r--r-- 1 root root 10671 Nov 11 14:37 index.html
/var/www/html
Fri Nov 11 15:00:26 UTC 2022
root@ip-172-31-88-233:/var/www/html# sudo echo "Hello World from $(hostname -f)" > /var/www/html/index.html
root@ip-172-31-88-233:/var/www/html# cat index.html
"Hello World from ip-172-31-88-233.ec2.internal"
root@ip-172-31-88-233:/var/www/html#
```



â€œHello World from ip-172-31-88-233.ec2.internalâ€œ

Assignment 2: Create readfile.sh in which you can read the information of PWD like size, permission, date time etc.

➔ **Ans:** Please find below shell script and expected output in screenshot.
#!/bin/bash

```
Current_Path=$(pwd)
```

```
echo "Current Working directory is: "$Current_Path
printf "\n"
```

```
echo "$Current_Path size details are: "
```

```
echo $(df -h $Current_Path)
```

```
printf "\n"
```

```
echo "Permission details of all files from $Current_Path: "
```

```
ls -ltr $Current_Path
```

```
printf "\n"
```

```
echo "Current timestamp: "`date`
```

```

swapnil@Swapnil:~$ ls -ltr readfile.sh; pwd; date
-rwxr-xr-x 1 swapnil swapnil 306 Nov 11 18:25 readfile.sh
/home/swapnil
Fri Nov 11 06:25:33 PM IST 2022
swapnil@Swapnil:~$ cat readfile.sh
#!/bin/bash

Current_Path=$(pwd)

echo "Current Working directory is: "$Current_Path
printf "\n"
echo "$Current_Path size details are: "
echo $(df -h $Current_Path)
printf "\n"
echo "Permission details of all files from $Current_Path: "
ls -ltr $Current_Path
printf "\n"
echo "Current timestamp: "`date`
swapnil@Swapnil:~$ ./readfile.sh
Current Working directory is: /home/swapnil

/home/swapnil size details are:
Filesystem Size Used Avail Use% Mounted on /dev/sdb 251G 2.6G 236G 2% /

Permission details of all files from /home/swapnil:
total 40
-rw-r--r-- 1 swapnil swapnil 96 Aug 15 18:52 test.txt
-rw-r--r-- 1 swapnil swapnil 0 Sep 5 18:18 Photo2.jpg
-rw-r--r-- 1 swapnil swapnil 0 Sep 5 18:18 Photo3.jpg
-rw-r--r-- 1 swapnil swapnil 0 Sep 5 18:18 script.sh
-rw-r--r-- 1 swapnil swapnil 0 Sep 5 18:18 photo1.jpg
-rw-r--r-- 1 swapnil swapnil 0 Sep 5 18:18 Beach
drwxr-xr-x 2 swapnil swapnil 4096 Sep 5 18:23 backup
drwxr-xr-x 2 swapnil swapnil 4096 Sep 5 18:23 photo
drwxr-xr-x 2 swapnil swapnil 4096 Sep 5 18:23 dir
-rw-r--r-- 1 swapnil swapnil 79 Sep 20 19:41 test.py
-rw-r--r-- 1 swapnil swapnil 72 Sep 20 19:46 demo.py
-rw-r--r-- 1 swapnil swapnil 61 Sep 20 19:52 demo1.py
-rwxr-xr-x 1 swapnil swapnil 101 Nov 10 20:07 username.sh
-rwxr-xr-x 1 swapnil swapnil 340 Nov 11 18:00 balance_check.sh
-rwxr-xr-x 1 swapnil swapnil 306 Nov 11 18:25 readfile.sh

Current timestamp: Fri Nov 11 06:25:41 PM IST 2022

```

Assignment 3: Take an input of name from user and print Have a great day ahead {name}

➔ **Ans:** Please find below shell script and expected output in screenshot.

```
#!/bin/bash
```

```
read -p "Please enter the username: " username
```

```
echo "Have a great day ahead "$username
```

```

swapnil@Swapnil:~$ cat username.sh; pwd; date
#!/bin/bash

read -p "Please enter the username: " username

echo "Have a great day ahead "$username
/home/swapnil
Thu Nov 10 08:07:50 PM IST 2022
swapnil@Swapnil:~$
swapnil@Swapnil:~$ ls -ltr username.sh
-rwxr-xr-x 1 swapnil swapnil 101 Nov 10 20:07 username.sh
swapnil@Swapnil:~$ ./username.sh
Please enter the username: Swapnil
Have a great day ahead Swapnil
swapnil@Swapnil:~$

```

Assignment 4: Let us take a scenario of fintech app program in which we want to have three separate outputs for 3 different situations:

- The balance is less than zero
- The balance is zero
- The balance is above zero

For instance, in the following program, use the if, elif, else statements to display different outputs in different scenarios:

Use “if” condition to check if the balance is less than zero. If this condition evaluates to true, display the message using the echo command: “Balance is less than zero, please add more funds else you will be charged penalty”.

If the above condition does not match, then use “elif” condition to check if the balance is equal to zero. If it evaluates to true, display the message: Balance is zero, please add funds

If none of the above condition matches, use the “else” condition to display the: Your balance is above zero.

→ **Ans:** Please find below shell script and expected output in screenshot.

```
#!/bin/bash
```

```
read -p "Please enter the balance amount to check the status according to Bank laws: "
balance
```

```
if [ $balance -lt 0 ]
```

```
then
```

```
    echo "Balance is less than zero, please add more funds else you will be charged penalty"
```

```
elif [ $balance -eq 0 ]
```

```
then
```

```
    echo "Balance is zero, please add funds"
```

```
else
```

```
    echo "Your balance is above zero"
```

```
fi
```

```

swapnil@Swapnil:~$ ls -ltr balance_check.sh; pwd; date
-rwxr-xr-x 1 swapnil swapnil 340 Nov 11 18:00 balance_check.sh
/home/swapnil
Fri Nov 11 06:01:30 PM IST 2022
swapnil@Swapnil:~$ cat balance_check.sh
#!/bin/bash

read -p "Please enter the balance amount to check the status according to Bank laws: " balance

if [ $balance -lt 0 ]
then
    echo "Balance is less than zero, please add more funds else you will be charged penalty"
elif [ $balance -eq 0 ]
then
    echo "Balance is zero, please add funds"
else
    echo "Your balance is above zero"
fi
swapnil@Swapnil:~$ ./balance_check.sh
Please enter the balance amount to check the status according to Bank laws: -1
Balance is less than zero, please add more funds else you will be charged penalty
swapnil@Swapnil:~$ ./balance_check.sh
Please enter the balance amount to check the status according to Bank laws: 0
Balance is zero, please add funds
swapnil@Swapnil:~$ ./balance_check.sh
Please enter the balance amount to check the status according to Bank laws: 1
Your balance is above zero
swapnil@Swapnil:~$

```

Assignment 5: Debug and define briefly about the following program:

```

#!/bin/bash
# Print a message about disk usage.
space_free=$( df -h | awk '{ print $5 }' | sort -n | tail -n 1 | sed 's/%/' )
case $space_free in
[1-5]*)
echo Plenty of disk space available
[6-7]*)
echo There could be a problem in the near future
8*)
echo Maybe we should look at clearing out old files
9*)
echo We could have a serious problem on our hands soon
*)
echo Something is not quite right here
;;
esac

```

→ **Ans:**

Debug issues in above shell script:

Above script has syntax issues in case statement. There should be “;;” symbol after every echo command.

Brief description about the shell script:

Above program finds the biggest (size wise) used space mountpoint on system/server with the help of command "df -h | awk '{ print \$5 }' | sort -n | tail -n 1 | sed 's/%/'" and stores the numeric value on variable space_free.

Then with the help of case statement it prints the output,

if used space in between 10-50 then "Plenty of disk space available"

if used space in between 60-70 then "There could be a problem in the near future"

if used space in between 80-89 then "Maybe we should look at clearing out old files"

if used space in between 10-50 then "We could have a serious problem on our hands soon"

if used space in does not fall under above criteria then "Something is not quite right here"