Install and Configure Apache

Apache Web Server will serve as the software responsible for delivering web content to clients. It listens for requests from browsers and responds by sending back HTML, PHP, or other supported files. During installation steps, when prompted for sudo password enter your root password.

1. Install Apache

    a. To install Apache type the following command and press enter:

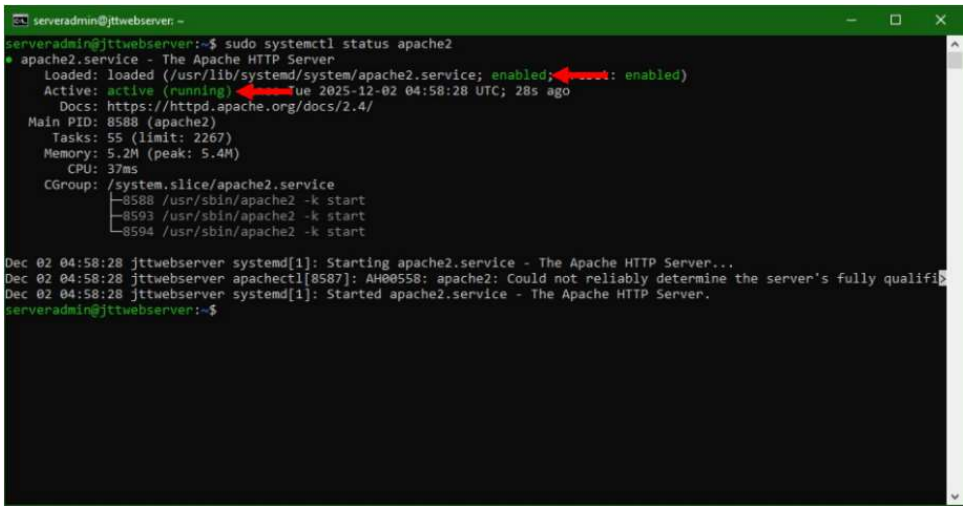
    ```
    sudo apt install apache2 -y
    ```

    This installs the Apache package and sets up the service to run on your Ubuntu Server.

    b. Verify the Apache Service by typing the following command and pressing enter:

    ```
    sudo systemctl status apache2
    ```

    Look for “enabled” and “active (running) in the output and press “q” to get out of the viewer.



If Apache is not enabled and running, you will have to start and enable the service manually using the following commands and pressing enter after each line:

```
sudo systemctl start apache2
sudo systemctl enable apache2
```

Check the status again to ensure the service is enabled and running.

2. Configure the firewall.

   a. Now we need to configure our firewall settings to ensure security of the server. First type the following command and press enter:

   ```
   sudo ufw status
   ```

   You will notice that it says inactive.

   b. Before we activate the firewall, let's see what applications are available for the firewall. Enter the following command and press enter to see a list of application profiles on the system:

   ```
   sudo ufw app list
   ```

   This will display a list of available applications. You will notice an OpenSSH profile and three Apache profiles: Apache (for HTTP traffic on Port 80), Apache Secure (for HTTPS traffic on Port 443), Apache Full (for both HTTP traffic on Port 80 and HTTPS traffic on Port 443).

   c. Now let's set some rules for the firewall by typing the following commands and pressing enter after each line:

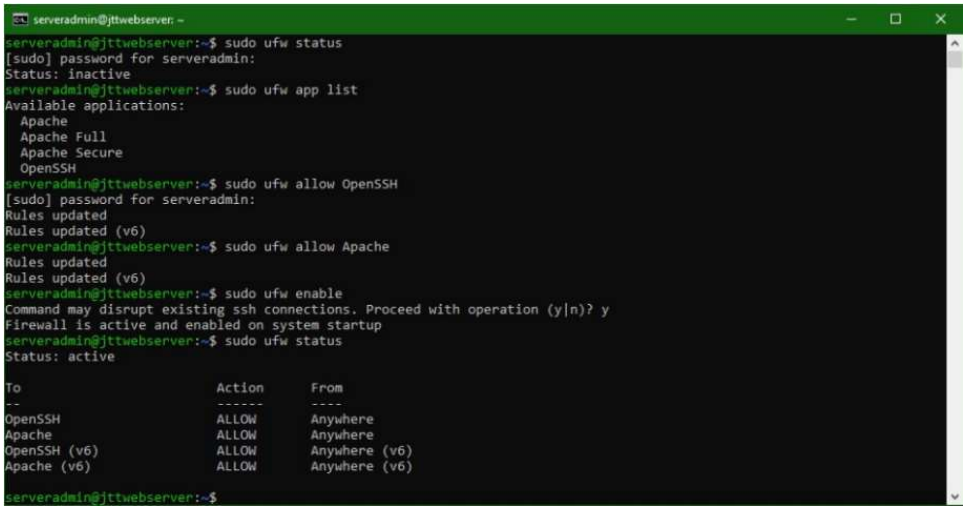
   ```
   sudo ufw allow OpenSSH
   sudo ufw allow Apache
   ```

   This will only allow OpenSSH traffic on Port 22 and HTTP traffic on Port 80. You can also allow either HTTPS traffic on Port 443 or both HTTP and HTTPS by replacing Apache with "Apache Secure" or "Apache Full" (be sure to add the quotations), but you would need to configure SSL/TSL certificates to establish secure connections to your server.

   d. Type the following commands to activate the firewall (you will get a message that the command may disrupt existing ssh connections, type y and press enter), and then view the status again and press enter after each line:

```
sudo ufw enable
sudo ufw status
```

You should see the status is now “active” along with a list of the rules that we have created.



3. Test the connection to the Apache Server by opening a web browser on your host machine and entering http://<your-ip-address> in the address bar replacing <your-ip-address> with the static IP address for the Ubuntu Server created earlier. It should load the Apache2 Default Page shown below:

You have now successfully installed and configured Apache Web Server, now let’s move on to the next layer of the stack – MySQL.