How To Install the Apache Web Server on Ubuntu

Step 1 — Installing Apache

Apache is available within Ubuntu's default software repositories, making it possible to install it using conventional package management tools.

Let's begin by updating the local package index to reflect the latest upstream changes:

```
sudo apt update
```

Then, install the `apache2` package:

```
sudo apt install apache2
```

After confirming the installation, `apt` will install Apache and all required dependencies.

Step 2 — Adjusting the Firewall

Before testing Apache, it's necessary to modify the firewall settings to allow outside access to the default web ports. Assuming that you followed the instructions in the prerequisites, you should 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 typing:

```
sudo ufw app list
```

You will receive a list of the application profiles:

```
OutputAvailable applications:
  Apache
  Apache Full
  Apache Secure
  OpenSSH
```

As indicated by the 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)

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

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

You can verify the change by typing:

```
sudo ufw status
```

The output will provide a list of allowed HTTP traffic:

```
OutputStatus: active

To                         Action      From
--                         ------      ----
OpenSSH                    ALLOW       Anywhere
Apache                     ALLOW       Anywhere
OpenSSH (v6)               ALLOW       Anywhere (v6)
Apache (v6)                ALLOW       Anywhere (v6)
```

As indicated by the output, the profile has been activated to allow access to the Apache web server.

Step 3 — Checking your Web Server

At the end of the installation process, Ubuntu 20.04 starts Apache. The web server should already be up and running.

Check with the `systemd` init system to make sure the service is running by typing:

```
sudo systemctl status apache2
```

```
Output apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled;
vendor preset: enabled)
     Active: active (running) since Thu 2020-04-23 22:36:30 UTC; 20h ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 29435 (apache2)
      Tasks: 55 (limit: 1137)
     Memory: 8.0M
     CGroup: /system.slice/apache2.service
             29435 /usr/sbin/apache2 -k start
             29437 /usr/sbin/apache2 -k start
             29438 /usr/sbin/apache2 -k start
```

As confirmed by this output, the service has started successfully. However, the best way to test this is to request a page from Apache.

You can access the default Apache landing page to confirm that the software is running properly through your IP address. If you do not know your server's IP address, you can get it a few different ways from the command line.

Try typing this at your server's command prompt:

```
hostname -I
```

You will get back a few addresses separated by spaces. You can try each in your web browser to determine if they work.

Another option is to use the Icanhazip tool, which should give you your public IP address as read from another location on the internet:

```
curl -4 icanhazip.com
```

When you have your server's IP address, enter it into your browser's address bar:

```
http://your_server_ip
```

You should see the default Ubuntu 20.04 Apache web page: