

## Paso 1: Instalar Apache

Apache está disponible en los repositorios de software predeterminados de Ubuntu, lo que permite instalarlo con las herramientas convencionales de administración de paquetes.

Comencemos actualizando el índice de paquetes locales para que reflejen los últimos cambios anteriores:

**sudo apt update**

A continuación, instale el paquete apache2:

**sudo apt install apache2**

Una vez confirmada la instalación, apt instalará Apache y todas las dependencias necesarias.

## Paso 2: Ajustar el firewall

Antes de probar Apache, es necesario modificar los ajustes de firewall para permitir el acceso externo a los puertos web predeterminados. Suponiendo que siguió las instrucciones de los requisitos previos, debería tener un firewall UFW configurado para que restrinja el acceso a su servidor.

Durante la instalación, Apache se registra con UFW para proporcionar algunos perfiles de aplicación que pueden utilizarse para habilitar o deshabilitar el acceso a Apache a través del firewall.

Enumere los perfiles de aplicación ufw escribiendo lo siguiente:

**sudo ufw app list**

Obtendrá una lista de los perfiles de aplicación:

### Output

**Available applications:**

## **Apache**

### **Apache Full**

### **Apache Secure**

### **OpenSSH**

Como lo indica el resultado, hay tres perfiles disponibles para Apache:

Apache: este perfil abre solo el puerto 80 (tráfico web normal no cifrado)

Apache Full: este perfil abre el puerto 80 (tráfico web normal no cifrado) y el puerto 443 (tráfico TLS/SSL cifrado)

Apache Secure: este perfil abre solo el puerto 443 (tráfico TLS/SSL cifrado)

Se recomienda habilitar el perfil más restrictivo, que de todos modos permitirá el tráfico que configuró. Debido a que en esta guía aún no configuramos SSL para nuestro servidor, solo deberemos permitir el tráfico en el puerto 80:

**sudo ufw allow 'Apache'**

Puede verificar el cambio escribiendo lo siguiente:

**sudo ufw status**

El resultado proporcionará una lista del tráfico de HTTP que se permite:

## **Output**

**Status: active**

<b>To</b>	<b>Action</b>	<b>From</b>
<b>--</b>	<b>-----</b>	<b>----</b>
<b>OpenSSH</b>	<b>ALLOW</b>	<b>Anywhere</b>
<b>Apache</b>	<b>ALLOW</b>	<b>Anywhere</b>
<b>OpenSSH (v6)</b>	<b>ALLOW</b>	<b>Anywhere (v6)</b>
<b>Apache (v6)</b>	<b>ALLOW</b>	<b>Anywhere (v6)</b>

Como lo indica el resultado, el perfil se activó para permitir el acceso al servidor web Apache.

### Paso 3: Comprobar su servidor web

Al final del proceso de instalación, Ubuntu 20.04 inicia Apache. El servidor web ya debería estar activo.

Realice una verificación con el sistema `init systemd` para saber si se encuentra en ejecución el servicio escribiendo lo siguiente:

```
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

Como lo confirma este resultado, el servicio se inició correctamente. Sin embargo, la mejor forma de comprobarlo es solicitar una página de Apache.

Puede acceder a la página de destino predeterminada de Apache para confirmar que el software funcione correctamente mediante su dirección IP: Si no conoce la dirección IP de su servidor, puede obtenerla de varias formas desde la línea de comandos.

Intente escribir esto en la línea de comandos de su servidor:

**hostname -I**

Obtendrá algunas direcciones separadas por espacios. Puede probar cada uno en el navegador web para determinar si funcionan.

Otra opción es utilizar la herramienta Icanhazip, que debería proporcionarle su dirección IP pública como aparece en otra ubicación en Internet:

**curl -4 icanhazip.com**

Cuando tenga la dirección IP de su servidor, introdúzcala en la barra de direcciones de su navegador:

[http://your\\_server\\_ip](http://your_server_ip)

Debería ver la página web predeterminada de Apache en Ubuntu 20.04:



# Apache2 Ubuntu Default Page

## It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

## Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

## Document Roots

By default, Ubuntu does not allow access through the web browser to *any* file apart of those located in `/var/www`, **public\_html** directories (when enabled) and `/usr/share` (for web applications). If your site is using a web document root located elsewhere (such as in `/srv`) you may need to whitelist your document root directory in `/etc/apache2/apache2.conf`.

The default Ubuntu document root is `/var/www/html`. You can make your own virtual hosts under `/var/www`. This is different to previous releases which provides better security out of the box.

## Reporting Problems

Please use the `ubuntu-bug` tool to report bugs in the Apache2 package with Ubuntu. However, check **existing bug reports** before reporting a new bug.

Please report bugs specific to modules (such as PHP and others) to respective packages, not to the web server itself.