

Tarea 2: Instalación de Apache en linux

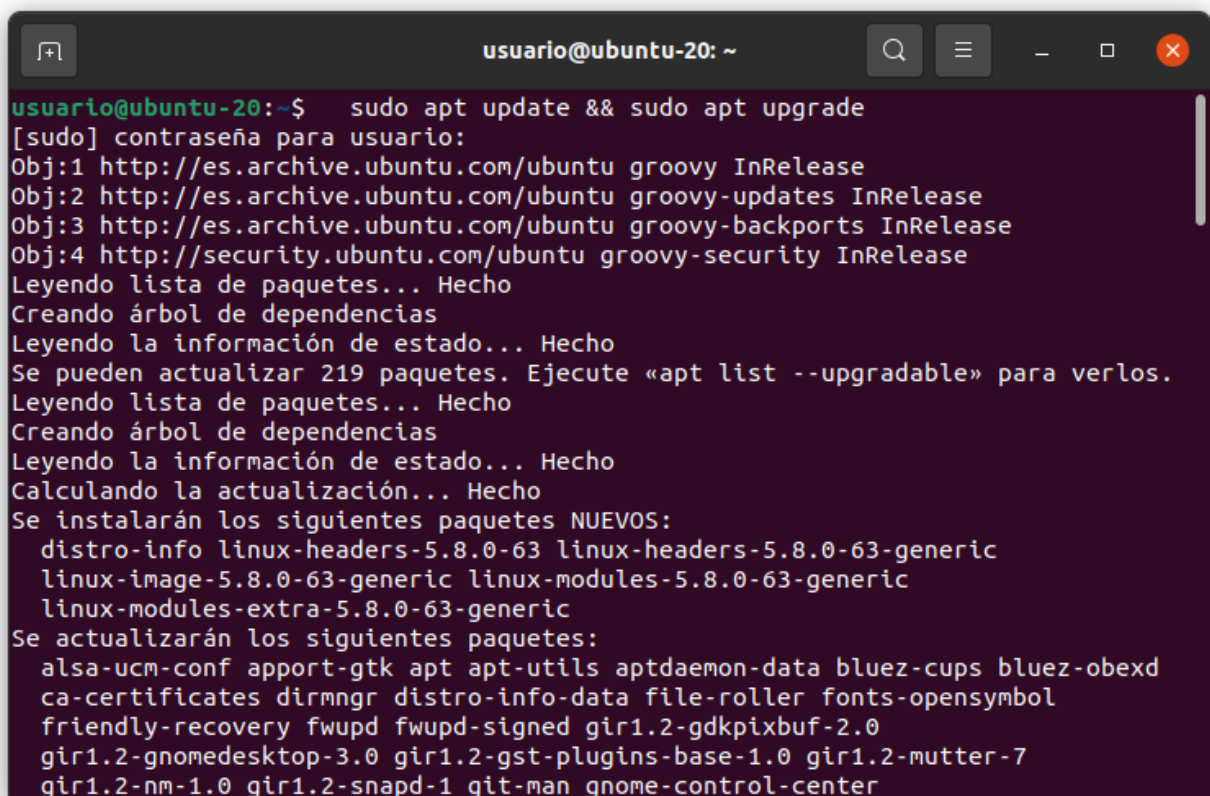


Índice

1. Actualización de repositorios
2. Instalacion de Apache
3. Acceso

1. Actualización de repositorios

Primero vamos a actualizar el repositorio y el sistema operativo para que la instalación de apache sea mas segura.



```
usuario@ubuntu-20: ~  
usuario@ubuntu-20:~$ sudo apt update && sudo apt upgrade  
[sudo] contraseña para usuario:  
Obj:1 http://es.archive.ubuntu.com/ubuntu groovy InRelease  
Obj:2 http://es.archive.ubuntu.com/ubuntu groovy-updates InRelease  
Obj:3 http://es.archive.ubuntu.com/ubuntu groovy-backports InRelease  
Obj:4 http://security.ubuntu.com/ubuntu groovy-security InRelease  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Se pueden actualizar 219 paquetes. Ejecute «apt list --upgradable» para verlos.  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Calculando la actualización... Hecho  
Se instalarán los siguientes paquetes NUEVOS:  
  distro-info linux-headers-5.8.0-63 linux-headers-5.8.0-63-generic  
  linux-image-5.8.0-63-generic linux-modules-5.8.0-63-generic  
  linux-modules-extra-5.8.0-63-generic  
Se actualizarán los siguientes paquetes:  
  alsa-ucm-conf apport-gtk apt apt-utils aptdaemon-data bluez-cups bluez-obexd  
  ca-certificates dirmngr distro-info-data file-roller fonts-opensymbol  
  friendly-recovery fwupd fwupd-signed gir1.2-gdkpixbuf-2.0  
  gir1.2-gnomedesktop-3.0 gir1.2-gst-plugins-base-1.0 gir1.2-mutter-7  
  gir1.2-nm-1.0 gir1.2-snapd-1 git-man gnome-control-center
```

2. Instalación de apache

Ahora vamos a instalar apache

```

usuario@ubuntu-20: ~
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
usuario@ubuntu-20:~$ sudo apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no
son necesarios.
  linux-headers-5.8.0-25 linux-headers-5.8.0-25-generic
  linux-image-5.8.0-25-generic linux-modules-5.8.0-25-generic
  linux-modules-extra-5.8.0-25-generic
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Paquetes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 1.844 kB de archivos.
Se utilizarán 8.012 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s

```

En el caso de que nos diera algún error accederíamos a su configuración y cambiaríamos los parámetros siguientes:

```
usuario@ubuntu-20:~$ sudo vi /etc/apache2/ports.conf
```

```

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

Cambiamos **Listen 80** a **Listen 8081**.

```

Listen 8081

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

También accedemos a:

```
usuario@ubuntu-20:~$ sudo vi /etc/apache2/sites-enabled/000-default.conf
```

```
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html
```

Cambiamos **<VirtualHost *:80>** a **<VirtualHost *:8081>**.

```
<VirtualHost *:8081>
```

Ahora debemos reiniciar los servicios para evitar conflictos entre apache y gitlab.

```
usuario@ubuntu-20:~$ sudo systemctl restart apache2
usuario@ubuntu-20:~$ sudo service apache2 restart
```

Luego vamos a cambiar la configuración del firewall. Para ello veremos la lista de perfiles.

```
usuario@ubuntu-20:~$ sudo ufw app list
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
```

Accederemos con el perfil de apache.

```
usuario@ubuntu-20:~$ sudo ufw allow 'Apache'
Reglas actualizadas
Reglas actualizadas (v6)
```

Verificamos los perfiles activos.

```
usuario@ubuntu-20:~$ sudo ufw status
Estado: inactivo
```

Como esta inactivo nosotros manual mente lo activaremos.

```
usuario@ubuntu-20:~$ sudo ufw enable
El cortafuegos está activo y habilitado en el arranque del sistema
```

Y ahora sí, podremos verificar el perfil activo.

```
usuario@ubuntu-20:~$ sudo ufw status
Estado: activo

Hasta          Acción         Desde
-----
Apache         ALLOW          Anywhere
Apache (v6)    ALLOW          Anywhere (v6)
```

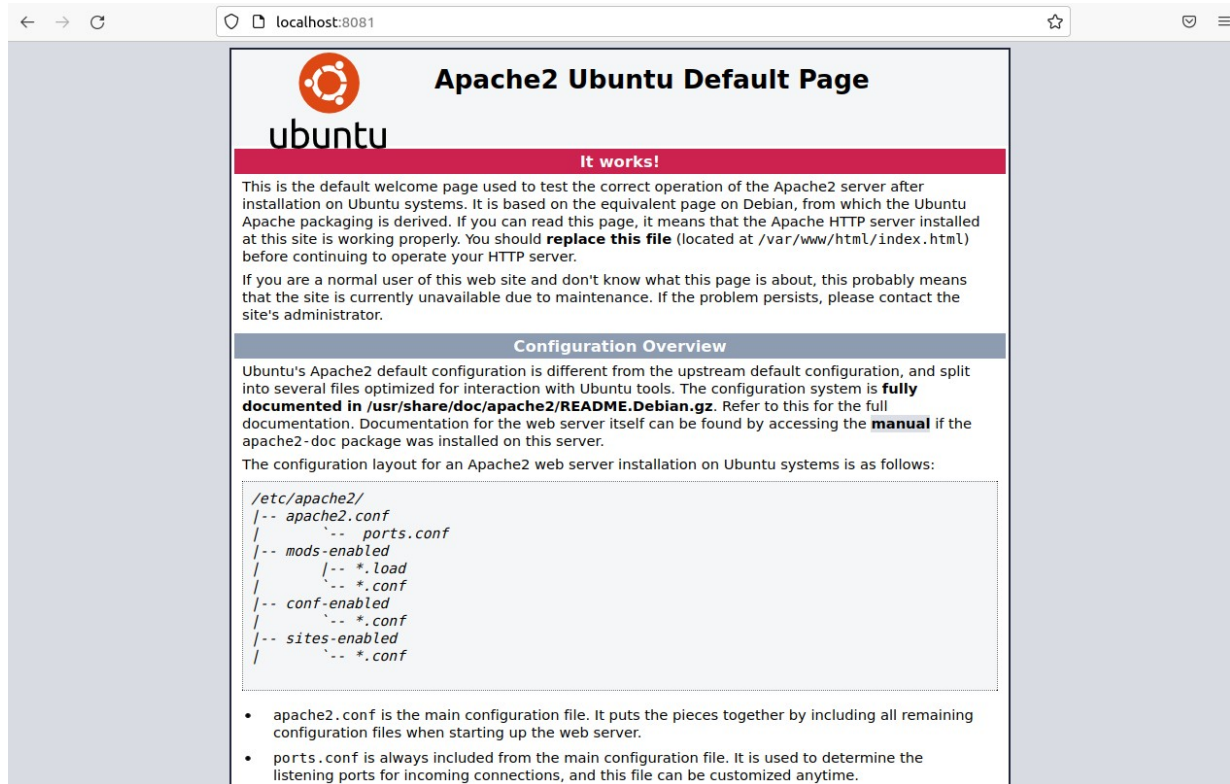
Y por ultimo utilizaremos el siguiente comando para luego acceder al servicio apache.

```
usuario@ubuntu-20:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese
   Active: active (running) since Tue 2021-10-05 22:17:33 CEST; 8min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 4802 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SU
  Main PID: 4806 (apache2)
    Tasks: 55 (limit: 2308)
   Memory: 5.4M
    CGroup: /system.slice/apache2.service
            └─4806 /usr/sbin/apache2 -k start
              └─4807 /usr/sbin/apache2 -k start
                └─4808 /usr/sbin/apache2 -k start

oct 05 22:17:33 ubuntu-20 systemd[1]: Starting The Apache HTTP Server...
oct 05 22:17:33 ubuntu-20 systemd[1]: Started The Apache HTTP Server.
```

3. Acceso

En este último paso vamos a acceder a localhost:8081.



The screenshot shows a web browser window with the address bar set to `localhost:8081`. The page title is "Apache2 Ubuntu Default Page". The page content includes the Ubuntu logo, a red banner saying "It works!", and a paragraph explaining that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It also mentions that the configuration system is fully documented in `/usr/share/doc/apache2/README.Debian.gz`. Below this, there is a section titled "Configuration Overview" which states that Ubuntu's Apache2 default configuration is different from the upstream default configuration and split into several files optimized for interaction with Ubuntu tools. It then lists the configuration layout for an Apache2 web server installation on Ubuntu systems as follows:

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

Below the code block, there are two bullet points:

- `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.