



DESPLIEGUE DE APLICACIONES WEB

# INSTALACIÓN Y CONFIGURACIÓN DE NGINX Y DE SFTP

# ÍNDICE

<b>SERVIDOR WEB NGINX .....</b>	<b>3</b>
A. INTRODUCCIÓN .....	3
B. INSTALACIÓN SERVIDOR WEB NGINX .....	3
C. CONFIGURACIÓN DE SERVIDOR WEB NGINX.....	7
D. ARCHIVOS .log .....	10
<b>PROTOCOLO DE TRANSFERENCIA DE FICHEROS (FTP) .....</b>	<b>11</b>
A. INTRODUCCIÓN .....	11
B. INSTALACIÓN .....	12
C. CONFIGURACIÓN DEL FTP .....	13
D. INSTALACIÓN Y USO DE FILEZILLA .....	15
<b>EXTRACCIÓN DEL ZIP SUBIDO A TRAVÉS DE FTP .....</b>	<b>17</b>
<b>DESPLIEGUE DEL SITIO WEB CONTENIDO EN EL .ZIP.....</b>	<b>18</b>

# SERVIDOR WEB NGINX

## A. INTRODUCCIÓN

En esta práctica vamos a realizar una instalación del servidor web nginx en una máquina virtual con el SO Debian 12 instalado:



## B. INSTALACIÓN SERVIDOR WEB NGINX

Para instalar nginx en la máquina virtual primero vamos a realizar una actualización de los repositorios con el comando:

`sudo apt update`

```
albertom-servidor@servidor-debian:~$ sudo apt update
[sudo] contraseña para albertom-servidor:
Obj:1 http://deb.debian.org/debian bookworm InRelease
Des:2 http://deb.debian.org/debian bookworm-updates InRelease [52,1 kB]
Obj:3 http://security.debian.org/debian-security bookworm-security InRelease
Descargados 52,1 kB en 1s (71,4 kB/s)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Todos los paquetes están actualizados.
```

A continuación, vamos a instalar el paquete que corresponde a nginx, para ello usaremos el comando:

`sudo apt install nginx`

```
albertom-servidor@servidor-debian:~$ sudo apt install nginx
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  nginx-common
Paquetes sugeridos:
  fcgiwrap nginx-doc
Se instalarán los siguientes paquetes NUEVOS:
```

Podemos comprobar el estado de nginx con el comando correspondiente de systemctl:

systemctl status nginx

```
albertom-servidor@servidor-debian:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Fri 2023-10-06 19:07:09 CEST; 1min 10s ago
     Docs: man:nginx(8)
  Process: 1561 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Process: 1562 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
 Main PID: 1588 (nginx)
    Tasks: 3 (limit: 2284)
   Memory: 2.5M
      CPU: 26ms
   CGroup: /system.slice/nginx.service
           └─1588 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─1590 "nginx: worker process"
               └─1591 "nginx: worker process"
```

Podemos visualizar el puerto de escucha del servidor con el comando:

netstat -ant

```
albertom-servidor@servidor-debian:~$ netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 :::1:53                 :::*                     LISTEN
tcp6       0      0 :::22                   :::*                     LISTEN
tcp6       0      0 :::80                   :::*                     LISTEN
tcp6       0      0 :::1:631                :::*                     LISTEN
```

- Protocolo: Si es TCP o UDP.
- Dispositivo local (local address): Identifica el equipo, está compuesto por una dirección IP y un número de puerto.
- Dispositivo externo (foreign address): Identifica el equipo de destino, al igual que el local se compone de una dirección IP y un número de puerto.

Si el número de puerto tiene 5 cifras se refiere a un cliente. Los números de puerto de los servidores tiene 2, 3 ó 4 cifras. Por ejemplo, HTTP tiene el 80, el HTTPS tiene el 443, MySQL tiene el 3306...

En este caso tenemos una dirección asociada a un puerto 80, por lo tanto, con esta máquina estamos actuando de servidor. Además de ese puerto de escucha tenemos el 22 para SSH y el 53 para DNS.

El 0.0.0.0 es lo mismo que si pusiera \*.\*.\*, es decir cualquier IP. El \* en el puerto, significa por cualquier número de puerto.

Podemos comprobar si nuestro servidor funciona, utilizando localhost en nuestro navegador.



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

Thank you for using nginx.

```
albertom-servidor@servidor-debian:~$ netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp        0      0 192.168.0.9:51708       34.117.65.55:443        ESTABLISHED
tcp        0      0 192.168.0.9:50404       34.117.237.239:443      ESTABLISHED
tcp        0      0 192.168.0.9:59936       142.250.178.163:443     ESTABLISHED
tcp        0      0 192.168.0.9:36990       34.149.100.209:443      ESTABLISHED
tcp        0      0 192.168.0.9:52518       34.160.144.191:443      ESTABLISHED
tcp        0      0 192.168.0.9:54116       34.117.121.53:443       ESTABLISHED
tcp        0      0 127.0.0.1:80            127.0.0.1:55204         ESTABLISHED
tcp        0      0 127.0.0.1:55204         127.0.0.1:80            ESTABLISHED
tcp        0      0 192.168.0.9:36994       34.149.100.209:443      ESTABLISHED
tcp6       0      0 :::1:53                 :::*                     LISTEN
tcp6       0      0 :::22                   :::*                     LISTEN
tcp6       0      0 :::80                   :::*                     LISTEN
tcp6       0      0 :::1:631                :::*                     LISTEN
```

Nginx tiene un ejecutable (nginxd) que se encarga de observar el directorio /var/www/ que guarda los elementos de la web.

Cuando recibe una petición GET desde un cliente del index.html y lo devuelve como datos. Este directorio es configurable en los archivos de configuración del servidor.

```
albertom-servidor@servidor-debian:~$ ls -la /var/www
total 12
drwxr-xr-x  3 root root 4096 oct  6 19:07 .
drwxr-xr-x 12 root root 4096 oct  6 19:07 ..
drwxr-xr-x  2 root root 4096 oct  6 19:07 html
albertom-servidor@servidor-debian:~$ ls -la /var/www/html
total 12
drwxr-xr-x  2 root root 4096 oct  6 19:07 .
drwxr-xr-x  3 root root 4096 oct  6 19:07 ..
-rw-r--r--  1 root root  615 oct  6 19:07 index.nginx-debian.html
```

Para llevar a cabo esta práctica vamos a utilizar un sitio web que se encuentra en un repositorio de GitHub.

Lo primero que necesitamos es instalar git en nuestro equipo, para ello usamos el comando:

sudo apt-get install git

```
albertom-servidor@servidor-debian:~$ sudo apt-get install git
[sudo] contraseña para albertom-servidor:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  git-man liberror-perl
Paquetes sugeridos:
```

A continuación, vamos a realizar una clonación del repositorio, para ello usamos el comando:

git clone URL\_repositorio

```
albertom-servidor@servidor-debian:~$ git clone https://github.com/cloudacademy/static-website-example
Clonando en 'static-website-example'...
remote: Enumerating objects: 69, done.
remote: Counting objects: 100% (69/69), done.
remote: Compressing objects: 100% (59/59), done.
remote: Total 69 (delta 9), reused 53 (delta 9), pack-reused 0
Recibiendo objetos: 100% (69/69), 668.08 KiB | 2.03 MiB/s, listo.
Resolviendo deltas: 100% (9/9), listo.
```

Con ls -la podemos comprobar que se ha copiado el repositorio (en la imagen es el directorio static-website-example).

```
albertom-servidor@servidor-debian:~$ ls -la
total 120
drwxr-xr-x 16 albertom-servidor albertom-servidor 4096 oct 9 12:10 .
drwxr-xr-x  3 root               root             4096 oct 6 15:28 ..
-rw-r--r--  1 albertom-servidor albertom-servidor 202 oct 6 21:08 .bash_history
-rw-r--r--  1 albertom-servidor albertom-servidor 220 oct 6 15:28 .bash_logout
-rw-r--r--  1 albertom-servidor albertom-servidor 3526 oct 6 15:28 .bashrc
drwxr-xr-x  9 albertom-servidor albertom-servidor 4096 oct 9 11:51 .cache
drwxr-xr-x 12 albertom-servidor albertom-servidor 4096 oct 6 16:07 .config
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Descargas
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Desktop
-rw-r--r--  1 albertom-servidor albertom-servidor  35 oct 6 15:39 .dmrc
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Documentos
-rw-r--r--  1 albertom-servidor albertom-servidor 5290 oct 6 15:28 .face
lrwxrwxrwx  1 albertom-servidor albertom-servidor   5 oct 6 15:28 .face.icon -> .face
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Imágenes
drwxr-xr-x  3 albertom-servidor albertom-servidor 4096 oct 6 15:39 .local
drwxr-xr-x  4 albertom-servidor albertom-servidor 4096 oct 9 11:51 .mozilla
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Música
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Plantillas
-rw-r--r--  1 albertom-servidor albertom-servidor 807 oct 6 15:28 .profile
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Público
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 18:41 .ssh
drwxr-xr-x  6 albertom-servidor albertom-servidor 4096 oct 9 12:10 static-website-example
drwxr-xr-x  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-clipboard.pid
-rw-r--r--  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-display-svga-x11.pid
drwxr-xr-x  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-draganddrop.pid
-rw-r--r--  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-seamless.pid
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Videos
-rw-r--r--  1 albertom-servidor albertom-servidor  60 oct 9 11:36 .Xauthority
-rw-r--r--  1 albertom-servidor albertom-servidor 841 oct 9 11:36 .xsession-errors
-rw-r--r--  1 albertom-servidor albertom-servidor 841 oct 6 21:04 .xsession-errors.old
```

Para facilitar la referencia de este directorio en futuros pasos, vamos a renombrar el directorio a web\_estatica.

mv static-website-example/ web\_estatica

```
albertom-servidor@servidor-debian:~$ mv static-website-example/ web_estatica
albertom-servidor@servidor-debian:~$ ls -la
total 120
drwxr-xr-x 16 albertom-servidor albertom-servidor 4096 oct 9 12:13 .
drwxr-xr-x  3 root               root             4096 oct 6 15:28 ..
-rw-r--r--  1 albertom-servidor albertom-servidor 202 oct 6 21:08 .bash_history
-rw-r--r--  1 albertom-servidor albertom-servidor 220 oct 6 15:28 .bash_logout
-rw-r--r--  1 albertom-servidor albertom-servidor 3526 oct 6 15:28 .bashrc
drwxr-xr-x  9 albertom-servidor albertom-servidor 4096 oct 9 11:51 .cache
drwxr-xr-x 12 albertom-servidor albertom-servidor 4096 oct 6 16:07 .config
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Descargas
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Desktop
-rw-r--r--  1 albertom-servidor albertom-servidor  35 oct 6 15:39 .dmrc
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Documentos
-rw-r--r--  1 albertom-servidor albertom-servidor 5290 oct 6 15:28 .face
lrwxrwxrwx  1 albertom-servidor albertom-servidor   5 oct 6 15:28 .face.icon -> .face
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Imágenes
drwxr-xr-x  3 albertom-servidor albertom-servidor 4096 oct 6 15:39 .local
drwxr-xr-x  4 albertom-servidor albertom-servidor 4096 oct 9 11:51 .mozilla
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Música
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Plantillas
-rw-r--r--  1 albertom-servidor albertom-servidor 807 oct 6 15:28 .profile
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Público
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 18:41 .ssh
-rw-r--r--  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-clipboard.pid
-rw-r--r--  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-display-svga-x11.pid
-rw-r--r--  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-draganddrop.pid
-rw-r--r--  1 albertom-servidor albertom-servidor   5 oct 9 11:36 .vboxclient-seamless.pid
drwxr-xr-x  2 albertom-servidor albertom-servidor 4096 oct 6 15:39 Videos
drwxr-xr-x  6 albertom-servidor albertom-servidor 4096 oct 9 12:10 web_estatica
-rw-r--r--  1 albertom-servidor albertom-servidor  60 oct 9 11:36 .Xauthority
-rw-r--r--  1 albertom-servidor albertom-servidor 841 oct 9 11:36 .xsession-errors
-rw-r--r--  1 albertom-servidor albertom-servidor 841 oct 6 21:04 .xsession-errors.old
```

Lo que debemos hacer ahora es copiar el directorio que hemos descargado dentro de /var/www/ (esto será nuestro DocumentRoot). Para ello utilizamos este comando:

```
sudo mv web_estatica/ /var/www/
```

```
albertom-servidor@servidor-debian:~$ sudo mv web_estatica/ /var/www/
```

Comprobamos que se ha copiado de forma correcta:

```
albertom-servidor@servidor-debian:~$ cd /var/www
albertom-servidor@servidor-debian:/var/www$ ls -la web_estatica/
total 64
drwxr-xr-x 6 albertom-servidor albertom-servidor 4096 oct  9 12:10 .
drwxr-xr-x 4 root              root              4096 oct  9 12:14 ..
drwxr-xr-x 6 albertom-servidor albertom-servidor 4096 oct  9 12:10 assets
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 oct  9 12:10 error
drwxr-xr-x 8 albertom-servidor albertom-servidor 4096 oct  9 12:10 .git
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 oct  9 12:10 images
-rw-r--r-- 1 albertom-servidor albertom-servidor 14522 oct  9 12:10 index.html
-rw-r--r-- 1 albertom-servidor albertom-servidor 17128 oct  9 12:10 LICENSE.MD
-rw-r--r-- 1 albertom-servidor albertom-servidor  648 oct  9 12:10 README.MD
```

Lanzamos el administrador de tareas de Linux (ps -ef) y filtramos por los procesos nginx:

```
ps -ef | grep nginx
```

```
albertom-servidor@servidor-debian:/var/www$ ps -ef | grep nginx
root          909      1  0 11:36 ?        00:00:00 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
www-data      910      909  0 11:36 ?        00:00:00 nginx: worker process
www-data      911      909  0 11:36 ?        00:00:00 nginx: worker process
alberto+     1980     1253  0 12:16 pts/0    00:00:00 grep nginx
```

El usuario www-data es el usuario de ejecución de un servidor web. Por tanto, este usuario debe disponer de permisos en el directorio que hemos creado, para ello usamos estos dos comandos:

```
sudo chown -R www-data:www-data /var/www/web_estatica/
```

```
sudo chmod -R 755 /var/www/web_estatica/
```

```
albertom-servidor@servidor-debian:/var/www$ sudo chown -R www-data:www-data /var/www/web_estatica/
albertom-servidor@servidor-debian:/var/www$ sudo chmod -R 755 /var/www/web_estatica/
```

El 755 implica que el propietario tendrá todos los permisos (lectura-escritura-ejecución), el grupo principal tendrá los permisos de lectura y ejecución y el resto de grupos tendrán estos mismos permisos.

## C. CONFIGURACIÓN DE SERVIDOR WEB NGINX

Vamos a trabajar en /etc/nginx/ dentro de este directorio tenemos:

- sites-available: Directorio donde se guardan los sitios.
- sites-enabled: Directorio que tiene los sitios operativos, los que se encuentran en "on".

```
albertom-servidor@servidor-debian:~$ sudo ls -la /etc/nginx
total 76
drwxr-xr-x  8 root root 4096 oct  6 19:07 .
drwxr-xr-x 120 root root 4096 oct  9 12:09 ..
drwxr-xr-x  2 root root 4096 mar 14 2023 conf.d
-rw-r--r--  1 root root 1125 mar 14 2023 fastcgi.conf
-rw-r--r--  1 root root 1055 mar 14 2023 fastcgi_params
-rw-r--r--  1 root root 2837 mar 14 2023 koi-utf
-rw-r--r--  1 root root 2223 mar 14 2023 koi-win
-rw-r--r--  1 root root 4338 mar 14 2023 mime.types
drwxr-xr-x  2 root root 4096 mar 14 2023 modules-available
drwxr-xr-x  2 root root 4096 mar 14 2023 modules-enabled
-rw-r--r--  1 root root 1446 mar 14 2023 nginx.conf
-rw-r--r--  1 root root 180  mar 14 2023 proxy_params
-rw-r--r--  1 root root 636  mar 14 2023 scgi_params
drwxr-xr-x  2 root root 4096 oct  6 19:07 sites-available
drwxr-xr-x  2 root root 4096 oct  6 19:07 sites-enabled
drwxr-xr-x  2 root root 4096 oct  6 19:07 snippets
-rw-r--r--  1 root root 664  mar 14 2023 uwsgi_params
-rw-r--r--  1 root root 3071 mar 14 2023 win-utf
```

Si hacemos un `ls -la` de `/sites-available` podemos ver que tenemos un archivo `default`, este archivo es el archivo de configuración del sitio web.

```
albertom-servidor@servidor-debian:/etc/nginx/sites-available$ ls -la
total 12
drwxr-xr-x 2 root root 4096 oct  6 19:07 .
drwxr-xr-x 8 root root 4096 oct  6 19:07 ..
-rw-r--r-- 1 root root 2412 mar 14 2023 default
```

Vamos a hacer una copia de este archivo, pero con el nombre de nuestra web, por ejemplo, `www.estatica.es`, para ello se utiliza el comando:

```
sudo cp default www.estatica.es
```

```
albertom-servidor@servidor-debian:/etc/nginx/sites-available$ sudo cp default www.estatica.es
```

Vamos a modificar el archivo, para ello utilizamos `nano` con el comando:

```
sudo nano www.estatica.es
```

```
albertom-servidor@servidor-debian:/etc/nginx/sites-available$ sudo nano www.estatica.es
```

El puerto de escucha (`listen` y `listen [::]`, este último utilizado para las IPv6) debe ser el 80.

El `root` indica el directorio en el que está guardada nuestra web, en este caso `/var/www/web_estatica/`.

El `server_name` es el dominio con el que va a ser encontrada nuestra web, en este caso, decimos usar `www.estatica.es`.



```

GNU nano 7.2                                www.estatica.es *
# Default server configuration
#
server {
    listen 80;
    listen [::]:80;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/web_estatica/;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name www.estatica.es;

    location / {
        # First attempt to serve request as file, then
        # as directory, then fall back to displaying a 404.
        try_files $uri $uri/ =404;
    }

    # pass PHP scripts to FastCGI server
    #
    #location ~ \.php$ {
    #    include snippets/fastcgi-php.conf;
    #
    #    # With php-fpm (or other unix sockets):
    #    fastcgi_pass unix:/run/php/php7.4-fpm.sock;

```

Como no hemos comprado un dominio y no disponemos de un DNS, debemos modificar el archivo `/etc/hosts` para que nuestra IP se asocie al dominio que hemos especificado en el archivo.

Para ello lo primero que debemos hacer es comprobar cuál es la ip de nuestra máquina con el comando `ip a`:

```

albertom-servidor@servidor-debian:/etc/nginx/sites-available$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,DYNAMIC,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
    link/ether 08:00:27:8d:a3:39 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.6/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 533sec preferred_lft 533sec
    inet6 fe80::a00:27ff:fe8d:a339/64 scope link
        valid_lft forever preferred_lft forever

```

A continuación, vamos a modificar el archivo, para ello usamos el comando `sudo nano /etc/hosts` e incluimos la pareja ip – dominio.

```

GNU nano 7.2                                /etc/hosts *
127.0.0.1    localhost
127.0.1.1    servidor-debian
10.0.2.6     www.estatica.es

```

Para comprobar que se ha modificado de forma correcta realizamos un ping a la dirección web.

```

albertom-servidor@servidor-debian:/etc/nginx/sites-available$ ping www.estatica.es
PING www.estatica.es (10.0.2.6) 56(84) bytes of data:
64 bytes from www.estatica.es (10.0.2.6): icmp_seq=1 ttl=64 time=0.016 ms
64 bytes from www.estatica.es (10.0.2.6): icmp_seq=2 ttl=64 time=0.038 ms
64 bytes from www.estatica.es (10.0.2.6): icmp_seq=3 ttl=64 time=0.090 ms
64 bytes from www.estatica.es (10.0.2.6): icmp_seq=4 ttl=64 time=0.047 ms
64 bytes from www.estatica.es (10.0.2.6): icmp_seq=5 ttl=64 time=0.073 ms
^C
--- www.estatica.es ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4095ms
rtt min/avg/max/mdev = 0.016/0.052/0.090/0.026 ms

```

El siguiente paso consiste en crear un enlace simbólico en el directorio sites-enabled, para ello se utiliza el siguiente comando:

```
sudo ln -s /etc/nginx/sites-available/www.estatica.es /etc/nginx/sites-enabled/
```

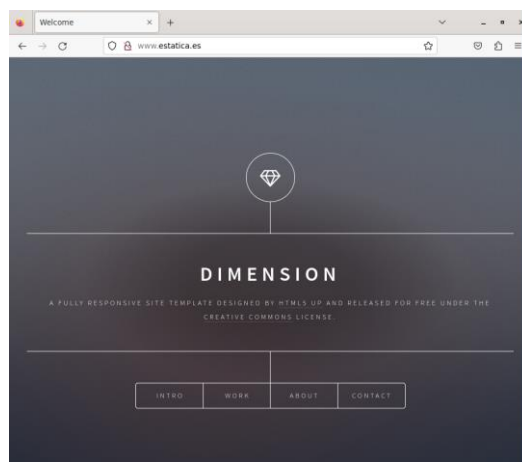
```
albertom-servidor@servidor-debian:/etc/nginx/sites-available$ sudo ln -s /etc/nginx/sites-available/www.estatica.es /etc/nginx/sites-enabled/
```

Para comprobar que se ha realizado bien vamos a reiniciar el servicio nginx, para ello utilizamos la opción restart del comando systemctl:

```
sudo systemctl restart nginx
```

```
albertom-servidor@servidor-debian:/etc/nginx/sites-available$ sudo systemctl restart nginx
```

Si ahora accedemos a la dirección [www.estatica.es](http://www.estatica.es) desde el navegador web, accedemos al sitio.



## D. ARCHIVOS .log

Los logs del sistema se van a guardar en el directorio /var/log

```
albertom-servidor@servidor-debian:~$ cd /var/log
albertom-servidor@servidor-debian:/var/log$ ls -la
total 1000
drwxr-xr-x 11 root      root      4096 oct 16 11:38 .
drwxr-xr-x 12 root      root      4096 oct  6 19:07 ..
-rw-r--r--  1 root      root     51121 oct  6 15:41 alternatives.log
drwxr-xr-x  2 root      root      4096 oct  9 16:51 apt
-rw-r----- 1 root      root      3060 oct 16 11:38 boot.log
-rw-r----- 1 root      root     22151 oct 16 11:38 boot.log.1
-rw-r----- 1 root      root     2866 oct 10 07:13 boot.log.2
-rw-r----- 1 root      root     53696 oct  9 16:48 boot.log.3
-rw-rw-r--  1 root      utmp      2304 oct  6 19:05 btmp
drwxr-xr-x  2 root      root      4096 oct 16 11:38 cups
-rw-r--r--  1 root      root     71167 oct  9 16:51 dpkg.log
-rw-r--r--  1 root      root         0 oct  6 15:09 faillog
-rw-r--r--  1 root      root      5635 oct  6 15:23 fontconfig.log
drwxr-xr-x  3 root      root      4096 oct  6 15:37 installer
drwxr-sr-x+ 3 root     systemd-journal 4096 oct  6 15:38 journal
-rw-rw-r--  1 root      utmp    292292 oct  6 18:55 lastlog
```

En este directorio hay múltiples ficheros si queremos organizar por última actualización podemos usar la opción -t.

```
albertom-servidor@servidor-debian:/var/log$ ls -lat
total 1000
-rw-rw-r-- 1 root      utmp           29568 oct 16 11:39 wtmp
drwx--x--x 2 root      root            4096 oct 16 11:38 lightdm
-rw-r--r-- 1 root      root            23571 oct 16 11:38 Xorg.0.log
drwxr-xr-x 11 root     root            4096 oct 16 11:38 .
-rw----- 1 root      root            3060 oct 16 11:38 boot.log
drwxr-xr-x 2 root     adm             4096 oct 16 11:38 nginx
drwxr-xr-x 2 root     root            4096 oct 16 11:38 cups
-rw----- 1 root      root            22151 oct 16 11:38 boot.log.1
-rw-r--r-- 1 root      root            25480 oct 10 10:14 Xorg.0.log.old
-rw----- 1 root      root            2866 oct 10 07:13 boot.log.2
-rw-r--r-- 1 root      root            711067 oct 9 16:51 dpkg.log
drwxr-xr-x 2 root     root            4096 oct 9 16:51 apt
-rw----- 1 root      root            53696 oct 9 16:48 boot.log.3
drwxr-xr-x 12 root     root            4096 oct 6 19:07 ..
-rw-rw---- 1 root      utmp           2304 oct 6 19:05 btmp
```

Dentro de `./nginx/` vamos a encontrar los archivos de log correspondientes a nginx, tanto de acceso como de errores:

```
albertom-servidor@servidor-debian:/var/log$ cd ./nginx/
albertom-servidor@servidor-debian:/var/log/nginx$ ls -lat
total 20
drwxr-xr-x 11 root      root      4096 oct 16 11:38 ..
drwxr-xr-x 2 root     adm       4096 oct 16 11:38 .
-rw-r----- 1 www-data adm        0 oct 16 11:38 access.log
-rw-r----- 1 www-data adm    2791 oct 10 10:48 access.log.1
-rw-r----- 1 www-data adm     410 oct 9 17:06 access.log.2.gz
-rw-r----- 1 www-data adm        0 oct 9 16:48 error.log
-rw-r----- 1 www-data adm      76 oct 6 19:07 error.log.1
```

- **Acceso al servidor:** En este log se van a almacenar los diferentes accesos que se hayan producido al servidor. Si queremos acceder a este log hacemos un `cat` o un `tail` de `access.log.1`.

```
albertom-servidor@servidor-debian:/var/log/nginx$ sudo tail -200 access.log.1
[sudo] contraseña para albertom-servidor:
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /assets/css/main.css HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /assets/js/jquery.min.js HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /images/pic03.jpg HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /images/pic02.jpg HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /images/pic01.jpg HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /assets/js/skel.min.js HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /assets/js/rutill.js HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
10.0.2.4 - - [10/Oct/2023:10:14:24 +0200] "GET /assets/js/main.js HTTP/1.1" 304 0 "http://www.estatica.es/" "Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0"
```

En este documento podemos ver las siguientes columnas:

- o Columna 1: IP desde donde se produjo la conexión.
- o Columna 2: fecha en la que se produjo la conexión.
- o Columna 3: petición que se realizó al servidor.
- **Errores:** En este log se almacenan los diferentes errores que se hayan podido producir en el servidor. Para acceder a él podemos hacerlo a través de un `cat` o un `tail` al archivo `error.log.1`.

```
albertom-servidor@servidor-debian:/var/log/nginx$ sudo tail -100 error.log.1
2023/10/06 19:07:09 [notice] 1588#1588: using inherited sockets from "5;6;"
```

## PROTOCOLO DE TRANSFERENCIA DE FICHEROS (FTP)

### A. INTRODUCCIÓN

El FTP es un protocolo de transferencia de ficheros entre sistemas conectados a una red TCP, es decir, definir una zona donde el usuario va

a poder dejar sus cosas. Es un protocolo que es inseguro y tiene una versión que sí aporta seguridad que es el SFTP, este protocolo añade una capa SSH haciéndolo más seguro y privado.



## B. INSTALACIÓN

Lo primero que vamos a hacer es una instalación de ftp desde los repositorios, para lo que tras hacer un update de estos utilizaremos el comando de instalación:

`sudo apt-get install vsftpd`

```
albertom-servidor@servidor-debian:~$ sudo apt-get install vsftpd
[sudo] contraseña para albertom-servidor:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  vsftpd
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 142 kB de archivos.
Se utilizarán 351 kB de espacio de disco adicional después de esta operación.
Des:1 http://deb.debian.org/debian bookworm/main amd64 vsftpd amd64 3.0.3-13+b2 [142 kB]
Descargados 142 kB en 0s (1.215 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete vsftpd previamente no seleccionado.
Leyendo la base de datos ... 80%
```

Comprobamos el estado del servicio con el siguiente comando:

`systemctl status vsftpd`

```
albertom-servidor@servidor-debian:~$ systemctl status vsftpd
● vsftpd.service - vsftpd FTP server
   Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled; preset: enabled)
   Active: active (running) since Mon 2023-10-16 11:54:12 CEST; 3min 13s ago
     Process: 1486 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, status=0/SUCCESS)
    Main PID: 1487 (vsftpd)
       Tasks: 1 (limit: 2284)
      Memory: 880.0K
         CPU: 5ms
    CGroup: /system.slice/vsftpd.service
            └─1487 /usr/sbin/vsftpd /etc/vsftpd.conf
```

Si usamos el comando `netstat -ant` podemos ver los dos puertos de escucha del FTP, el 21 y el 22.

```
albertom-servidor@servidor-debian:~$ netstat -ant
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:21              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 :::80                   :::*                    LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::1:53                 :::*                    LISTEN
tcp6       0      0 :::1:631                 :::*                    LISTEN
```

Ahora vamos a crear un directorio que servirá como lugar del servidor donde subir los archivos a través del vsftpd. Además, como veremos después, nos servirá como localización de conexión inicial al iniciar sesión en el FTP:

```
mkdir /home/nombreUsuario/ftp
```

```
albertom-servidor@servidor-debian:~$ mkdir /home/albertom-servidor/ftp
```

Tenemos que crear los certificados de seguridad necesarios para aportar la capa de cifrado a nuestra conexión, para ello debemos crear un par de claves con el código x509, lo hacemos utilizando el comando:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
/etc/ssl/private/vsftpd.pem -out /etc/ssl/private/vsftpd.pem
```

```
albertom-servidor@servidor-debian:~$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/vsftpd.pem -out /etc/ssl/private/vsftpd.pem
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank.
For some fields there will be a default value.
For the 'country' field, the field will be left blank.
-----
Country Name (2 letter code) [AU]:ES
State or Province Name (full name) [Some-State]:Madrid
Locality Name (eg, city) []:Colindale
Organization Name (eg, company) [Internet Widgits Pty Ltd]:IES Luis Braille
Organizational Unit Name (eg, section) []:IESLB
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:albertom@ieslb.com
```

Para comprobar si se ha creado correctamente podemos usar el comando:

```
sudo ls -la /etc/ssl/private/vsftpd.pem
```

```
albertom-servidor@servidor-debian:~$ sudo ls -la /etc/ssl/private/vsftpd.pem
-rw----- 1 root root 3136 oct 16 12:22 /etc/ssl/private/vsftpd.pem
```

Esta clave que hemos creado va a servir para identificar al servidor SFTP como seguro cuando un cliente se conecte a él.

### C. CONFIGURACIÓN DEL FTP

Una vez realizado lo anterior hay que configurar el servicio vsftpd a través de su archivo de configuración vsftpd.conf. Para ello utilizaremos el comando:

```
sudo nano /etc/vsftpd.conf
```

```
albertom-servidor@servidor-debian:~$ sudo nano /etc/vsftpd.conf
```

Eliminamos las siguientes líneas del documento:

```
rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
ssl_enable=NO
```

Y las sustituimos por estas:

```
rsa_cert_file=/etc/ssl/private/vsftpd.pem
rsa_private_key_file=/etc/ssl/private/vsftpd.pem
ssl_enable=YES
allow_anon_ssl=NO
force_local_data_ssl=YES
force_local_logins_ssl=YES
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
require_ssl_reuse=NO
ssl_ciphers=HIGH

local_root=/home/albertom-servidor/ftp
```

- **rsa\_cert\_file:** Especifica la ruta del archivo que contiene el certificado X509 que hemos creado previamente.
- **rsa\_private\_key\_file:** Similar al anterior, pero en este caso para la clave privada.
- **ssl\_enable:** Activa (YES) o desactiva (NO) el uso de SSL/TLS en conexiones FTP.
- **allow\_anon\_ssl:** Permite (YES) o no (NO) el que los usuarios anónimos puedan utilizar conexiones SSL/TLS.
- **force\_local\_data\_ssl:** Fuerza (YES) o no (NO) el que todas las conexiones de datos se realicen a través de SSL/TLS.
- **ssl\_tlsv1:** Activa (YES) o desactiva (NO) el que se permitan conexiones que utilizan la versión 1 de TLS.
- **ssl\_sslvX:** Activa (YES) o desactiva (NO) el que se establezcan conexiones con las versiones 2 y 3 de SSL.
- **require\_ssl\_reuse:** Permite (YES) o no (NO) el que las conexiones SSL reutilicen las sesiones anteriores.
- **ssl\_ciphers:** Especifica los ciphersuites (conjuntos de algoritmos de cifrado) que se pueden utilizar en las conexiones SSL/TLS.
- **local\_root:** El directorio raíz para que los usuarios puedan conectarse al SFTP. En este caso elegiremos el directorio que hemos creado en pasos previos.

Guardamos el archivo, salimos y reiniciamos el servicio para que se cargue la nueva configuración con el comando:

```
sudo systemctl restart vsftpd
```

```
albertom-servidor@servidor-debian:~$ sudo systemctl restart vsftpd
```

## D. INSTALACIÓN Y USO DE FILEZILLA

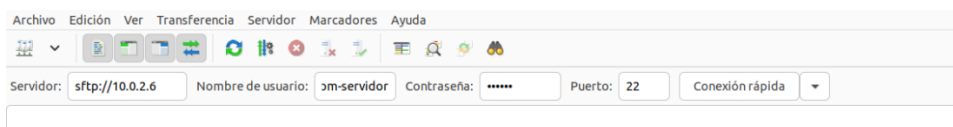
Una vez completados los pasos previos podremos acceder a nuestro servidor mediante un cliente FTP como por ejemplo FileZilla.

Para esta sección debemos utilizar la máquina cliente que creamos en la primera práctica e instalar filezilla en la máquina, para ello utilizaremos el comando:

```
sudo apt-get install filezilla
```

```
albertom-servidor@albertoMartinezPerez:~$ sudo apt-get install filezilla
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
libflashrom1 libftdi1-2 libllvm13
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
filezilla-common libfilezilla-common libfilezilla24 libpugixml1v5
libwxbase3.0-0v5 libwxgtk3.0-gtk3-0v5
Se instalarán los siguientes paquetes NUEVOS:
filezilla filezilla-common libfilezilla-common libfilezilla24 libpugixml1v5
libwxbase3.0-0v5 libwxgtk3.0-gtk3-0v5
0 actualizados, 7 nuevos se instalarán, 0 para eliminar y 24 no actualizados.
Se necesita descargar 10,1 MB de archivos.
Se utilizarán 36,9 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [s/n] s
```

Abrimos FileZilla y nos conectamos:

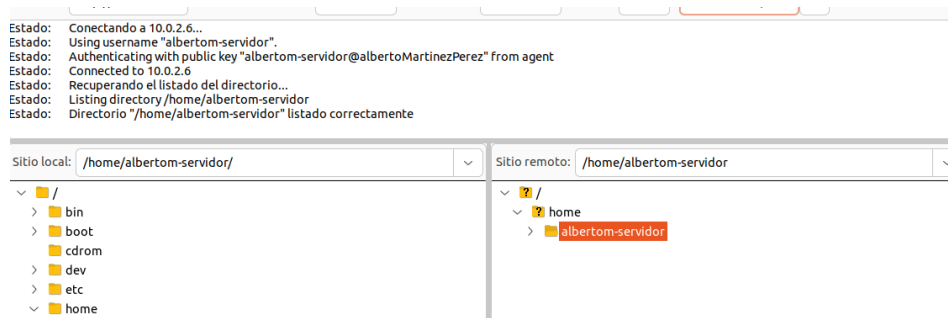


- El campo servidor identifica la IP de nuestra máquina virtual servidor. Si añadimos "sftp://" ya especificamos que nos queremos conectar a la IP mediante SFTP.
- El campo nombre de usuario el de nuestro usuario.
- El campo contraseña el de la contraseña asociada al usuario.
- El campo puerto introducimos 22. Es el puerto del SFTP dedicado al intercambio de datos mediante conexión SSH.

Una vez le demos a "conexión rápida", el servidor a través del SFTP nos devuelve una de las claves que se generaron en la práctica del SSH:

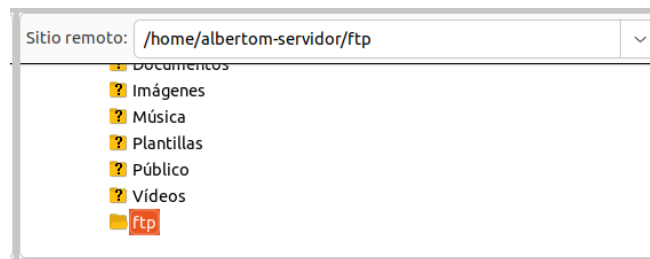


Nos conecta en el /home del usuario.

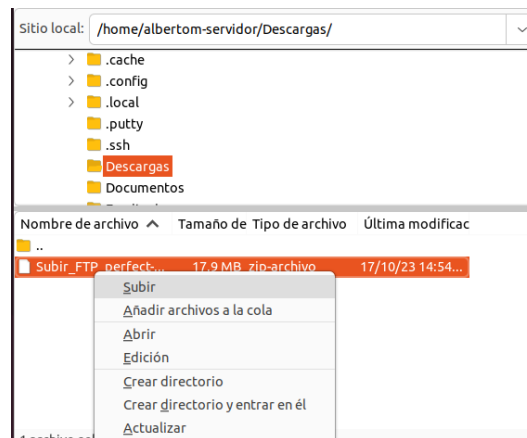


El sitio local es nuestra máquina cliente (Ubuntu) y el sitio remoto es nuestra máquina servidor (Debian).

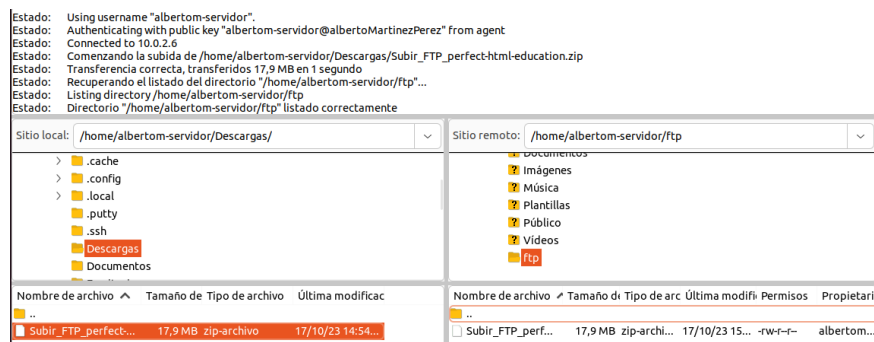
Para subir un archivo en primer lugar navegamos en el sitio remoto hasta la carpeta FTP:



Y subimos un archivo desde nuestro sitio local, para ello hacemos clic derecho en el archivo y elegimos la opción “subir”:



Con ello se habrá completado la subida del fichero:





## EXTRACCIÓN DEL ZIP SUBIDO A TRAVÉS DE FTP

Si ahora hacemos un `ls -la` para listar los archivos de nuestro directorio `/ftp` veremos que tenemos el archivo `.zip`:

```
sudo ls -la /home/albertom-servidor/ftp
```

```
albertom-servidor@servidor-debian:~$ sudo ls -la /home/albertom-servidor/ftp
total 17452
drwxr-xr-x  2 albertom-servidor albertom-servidor  4096 oct 17 15:01 .
drwx----- 17 albertom-servidor albertom-servidor  4096 oct 20 15:09 ..
-rw-r--r--  1 albertom-servidor albertom-servidor 17860176 oct 17 15:01 Subir_FTP_perfect-html-education.zip
```

Para poder descomprimir archivos `.zip` necesitamos tener instalado el `unzip`, para comprobar si lo tenemos instalado podemos usar el siguiente comando:

```
sudo apt-get update
```

```
sudo apt-get install unzip
```

```
albertom-servidor@servidor-debian:~$ sudo apt-get update
Des:1 http://security.debian.org/debian-security bookworm-security InRelease [48,0 kB]
Obj:2 http://deb.debian.org/debian bookworm InRelease
Des:3 http://deb.debian.org/debian bookworm-updates InRelease [52,1 kB]
Des:4 http://security.debian.org/debian-security bookworm-security/main Sources [51,5 kB]
Des:5 http://security.debian.org/debian-security bookworm-security/main amd64 Packages [86,2 kB]
Des:6 http://security.debian.org/debian-security bookworm-security/main Translation-en [48,8 kB]
Descargados 287 kB en 1s (338 kB/s)
Leyendo lista de paquetes... Hecho
albertom-servidor@servidor-debian:~$ sudo apt-get install unzip
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
unzip ya está en su versión más reciente (6.0-28).
fijado unzip como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 80 no actualizados.
```

Ahora utilizaremos el comando `unzip` para descomprimir el archivo, lo haremos en el propio directorio del FTP:

```
unzip /home/albertom-servidor/ftp/archivo.zip -d /home/albertom-servidor/ftp
```

```
albertom-servidor@servidor-debian:~$ unzip /home/albertom-servidor/ftp/Subir_FTP_perfect-html-education.zip -d /home/albertom-servidor/ftp
Archive: /home/albertom-servidor/ftp/Subir_FTP_perfect-html-education.zip
  creating: /home/albertom-servidor/ftp/html/
  inflating: /home/albertom-servidor/ftp/html/about.html
  inflating: /home/albertom-servidor/ftp/html/contact.html
  creating: /home/albertom-servidor/ftp/html/css/
  inflating: /home/albertom-servidor/ftp/html/css/animate.css
  inflating: /home/albertom-servidor/ftp/html/css/bootstrap.min.css
  inflating: /home/albertom-servidor/ftp/html/css/custom.css
  inflating: /home/albertom-servidor/ftp/html/css/flashy.min.css
  inflating: /home/albertom-servidor/ftp/html/css/flaticon.css
  inflating: /home/albertom-servidor/ftp/html/css/font-awesome.min.css
  inflating: /home/albertom-servidor/ftp/html/css/magnific-popup.css
  inflating: /home/albertom-servidor/ftp/html/css/pogo-slider.min.css
  inflating: /home/albertom-servidor/ftp/html/css/responsive.css
  inflating: /home/albertom-servidor/ftp/html/css/responsiveslides.css
  inflating: /home/albertom-servidor/ftp/html/css/style.css
  inflating: /home/albertom-servidor/ftp/html/css/timeline.css
  creating: /home/albertom-servidor/ftp/html/fonts/
  inflating: /home/albertom-servidor/ftp/html/fonts/Flaticon.eot
  inflating: /home/albertom-servidor/ftp/html/fonts/Flaticon.svg
  inflating: /home/albertom-servidor/ftp/html/fonts/Flaticon.ttf
  inflating: /home/albertom-servidor/ftp/html/fonts/Flaticon.woff
  inflating: /home/albertom-servidor/ftp/html/fonts/fontawesome-webfont.eot
  inflating: /home/albertom-servidor/ftp/html/fonts/fontawesome-webfont.svg
  inflating: /home/albertom-servidor/ftp/html/fonts/fontawesome-webfont.ttf
  inflating: /home/albertom-servidor/ftp/html/fonts/fontawesome-webfont.woff
  inflating: /home/albertom-servidor/ftp/html/fonts/fontawesome-webfont.woff2
  inflating: /home/albertom-servidor/ftp/html/fonts/FontAwesome.otf
```

Vamos a comprobar que se ha descomprimido de forma correcta, para ello haremos un `ls -la` del directorio, si todo ha salido correctamente debemos tener un directorio `/html`:

```
albertom-servidor@servidor-debian:~$ sudo ls -la /home/albertom-servidor/ftp/
total 17456
drwxr-xr-x  3 albertom-servidor albertom-servidor  4096 oct 20 15:18 .
drwx----- 17 albertom-servidor albertom-servidor  4096 oct 20 15:09 ..
drwxr-xr-x  7 albertom-servidor albertom-servidor  4096 mar 28 2019 html
-rw-r--r--  1 albertom-servidor albertom-servidor 17860176 oct 17 15:01 Subir_FTP_perfect-html-education.zip
```

En el interior de este directorio se encuentran todos los archivos del archivo zip:

```
albertom-servidor@servidor-debian:~$ sudo ls -la /home/albertom-servidor/ftp/html/
total 104
drwxr-xr-x 7 albertom-servidor albertom-servidor 4096 mar 28 2019 .
drwxr-xr-x 3 albertom-servidor albertom-servidor 4096 oct 20 15:18 ..
-rw-r--r-- 1 albertom-servidor albertom-servidor 12367 abr 2 2019 about.html
-rw-r--r-- 1 albertom-servidor albertom-servidor 11126 abr 2 2019 contact.html
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 css
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 fonts
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 abr 1 2019 images
-rw-r--r-- 1 albertom-servidor albertom-servidor 23289 abr 1 2019 index.html
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 js
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 php
-rw-r--r-- 1 albertom-servidor albertom-servidor 21427 mar 26 2019 services.html
```

## DESPLIEGUE DEL SITIO WEB CONTENIDO EN EL .ZIP

Lo primero que debemos hacer es mover el fichero /html al directorio /var/www creando para ello un nuevo directorio. En este caso en lugar de realizar un mkdir que genere este directorio, vamos a crear el directorio /fich\_academia junto a la orden mv, para ello usaremos este comando:

```
sudo mv /home/albertom-servidor/ftp/html /var/www/fich_academia/
```

```
albertom-servidor@servidor-debian:~$ sudo mv /home/albertom-servidor/ftp/html/ /var/www/fich_academia
albertom-servidor@servidor-debian:~$ sudo ls -la /var/www/fich_academia
total 104
drwxr-xr-x 7 albertom-servidor albertom-servidor 4096 mar 28 2019 .
drwxr-xr-x 5 root              root          4096 oct 20 15:23 ..
-rw-r--r-- 1 albertom-servidor albertom-servidor 12367 abr 2 2019 about.html
-rw-r--r-- 1 albertom-servidor albertom-servidor 11126 abr 2 2019 contact.html
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 css
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 fonts
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 abr 1 2019 images
-rw-r--r-- 1 albertom-servidor albertom-servidor 23289 abr 1 2019 index.html
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 js
drwxr-xr-x 2 albertom-servidor albertom-servidor 4096 mar 28 2019 php
-rw-r--r-- 1 albertom-servidor albertom-servidor 21427 mar 26 2019 services.html
```

Ahora debemos darle la propiedad y los permisos pertinentes al directorio /fich\_academia, el propietario debe ser www-data para que pueda acceder a este directorio y lanzar la web. Para esto utilizaremos los comandos:

```
sudo chown -R www-data:www-data /var/www/fich_academia
```

```
sudo chmod -R 755 /var/www/fich_academia
```

```
albertom-servidor@servidor-debian:~$ sudo chown -R www-data:www-data /var/www/fich_academia/
albertom-servidor@servidor-debian:~$ sudo chmod -R 755 /var/www/fich_academia/
albertom-servidor@servidor-debian:~$ sudo ls -la /var/www/
total 20
drwxr-xr-x 5 root      root      4096 oct 20 15:23 .
drwxr-xr-x 12 root     root      4096 oct 6 19:07 ..
drwxr-xr-x 7 www-data www-data 4096 mar 28 2019 fich_academia
drwxr-xr-x 2 root      root      4096 oct 6 19:07 html
drwxr-xr-x 6 www-data www-data 4096 oct 9 16:52 web_estatica
```

El siguiente paso será modificar la configuración del servidor web para este nuevo sitio web que vamos a lanzar. Para ello vamos a hacer una copia del fichero default contenido en /etc/nginx/sites-available para nuestro nuevo sitio:

```
sudo cp /etc/nginx/sites-available/default /etc/nginx/sites-  
available/www.academia.es
```

Debemos hacer a continuación un fichero de configuración de nuestro sitio web, para ello hacemos una copia de default:

```
albertom-servidor@servidor-debian:~$ sudo cp /etc/nginx/sites-available/default /etc/nginx/sites-available/www.academia.es  
albertom-servidor@servidor-debian:~$ ls -la /etc/nginx/sites-available/  
total 20  
drwxr-xr-x 2 root root 4096 oct 20 15:30 .  
drwxr-xr-x 8 root root 4096 oct 6 19:07 ..  
-rw-r--r-- 1 root root 2412 mar 14 2023 default  
-rw-r--r-- 1 root root 2412 oct 20 15:30 www.academia.es  
-rw-r--r-- 1 root root 2405 oct 9 16:57 www.estatica.es
```

Ahora habrá que configurar el archivo que hemos creado a través de nano:

```
albertom-servidor@servidor-debian:~$ sudo nano /etc/nginx/sites-available/www.academia.es
```

```
server {  
    listen 80;  
    listen [::]:80;  
  
    # SSL configuration  
    #  
    # listen 443 ssl default_server;  
    # listen [::]:443 ssl default_server;  
    #  
    # Note: You should disable gzip for SSL traffic.  
    # See: https://bugs.debian.org/773332  
    #  
    # Read up on ssl_ciphers to ensure a secure configuration.  
    # See: https://bugs.debian.org/765782  
    #  
    # Self signed certs generated by the ssl-cert package  
    # Don't use them in a production server!  
    #  
    # include snippets/snakeoil.conf;  
  
    root /var/www/fich_academia;  
  
    # Add index.php to the list if you are using PHP  
    index index.html index.htm index.nginx-debian.html;  
  
    server_name www.academia.es;  
  
    location / {  
        # First attempt to serve request as file, then  
        # as directory, then fall back to displaying a 404.  
        try_files $uri $uri/ =404;  
    }  
}
```

El listen será el puerto para escuchas de direcciones IPv4, el listen[::] para las escuchas de direcciones IPv6, por otra parte en root debemos poner el directorio donde se encuentran todos los ficheros de nuestro sitio web /var/www/fich\_academia. Por último, en server\_name pondremos el nombre de dominio que tendrá nuestra web.

Como último paso para la configuración de nuestro servidor Nginx será crear el enlace simbólico de este fichero en el directorio /etc/nginx/sites-enabled:

```
albertom-servidor@servidor-debian:~$ sudo ln -s /etc/nginx/sites-available/www.academia.es /etc/nginx/sites-enabled/  
albertom-servidor@servidor-debian:~$ ls -la /etc/nginx/sites-enabled/  
total 8  
drwxr-xr-x 2 root root 4096 oct 20 15:38 .  
drwxr-xr-x 8 root root 4096 oct 6 19:07 ..  
lrwxrwxrwx 1 root root 34 oct 6 19:07 default -> /etc/nginx/sites-available/default  
lrwxrwxrwx 1 root root 42 oct 20 15:38 www.academia.es -> /etc/nginx/sites-available/www.academia.es  
lrwxrwxrwx 1 root root 42 oct 9 17:05 www.estatica.es -> /etc/nginx/sites-available/www.estatica.es
```

Al haber cambiado la configuración del servicio Nginx debemos reiniciarlo para que se aplique la nueva configuración:

sudo systemctl restart nginx.service

```
albertom-servidor@servidor-debian:~$ sudo systemctl restart nginx.service
albertom-servidor@servidor-debian:~$ sudo systemctl status nginx.service
* nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Fri 2023-10-20 15:40:17 CEST; 12s ago
     Docs: man:nginx(8)
  Process: 1978 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Process: 1979 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
 Main PID: 1980 (nginx)
   Tasks: 3 (limit: 2284)
  Memory: 2.4M
    CPU: 16ms
  CGroup: /system.slice/nginx.service
          └─1980 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
              └─1981 "nginx: worker process"
                  └─1982 "nginx: worker process"

oct 20 15:40:17 servidor-debian systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
oct 20 15:40:17 servidor-debian systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
```

Si ahora intentáramos hacer ping a [www.academia.es](http://www.academia.es) recibiríamos el siguiente mensaje:

```
albertom-servidor@servidor-debian:~$ ping www.academia.es
ping: www.academia.es: Fallo temporal en la resolución del nombre
```

Esto se debe a que no tenemos contratado un dominio, para solucionarlo, modificamos el archivo /etc/hosts/ para hacer que nuestra IP esté asociada a este dominio:

```
GNU nano 7.2
127.0.0.1    localhost
127.0.1.1    servidor-debian
10.0.2.14    www.academia.es
# The following lines are desirable for IPv6
::1          localhost ip6-localhost ip6-loopback
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
```

Ahora podremos hacer ping al sitio y recibir la respuesta de que los paquetes han sido recibidos:

```
albertom-servidor@servidor-debian:~$ ping www.academia.es
PING www.academia.es (10.0.2.14) 56(84) bytes of data:
64 bytes from www.academia.es (10.0.2.14): icmp_seq=1 ttl=64 time=0.148 ms
64 bytes from www.academia.es (10.0.2.14): icmp_seq=2 ttl=64 time=0.039 ms
64 bytes from www.academia.es (10.0.2.14): icmp_seq=3 ttl=64 time=0.069 ms
64 bytes from www.academia.es (10.0.2.14): icmp_seq=4 ttl=64 time=0.033 ms
^C
--- www.academia.es ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3076ms
rtt min/avg/max/mdev = 0.033/0.072/0.148/0.045 ms
```

Si ahora accedemos al sitio desde un navegador web este sería el aspecto que tendría:

