

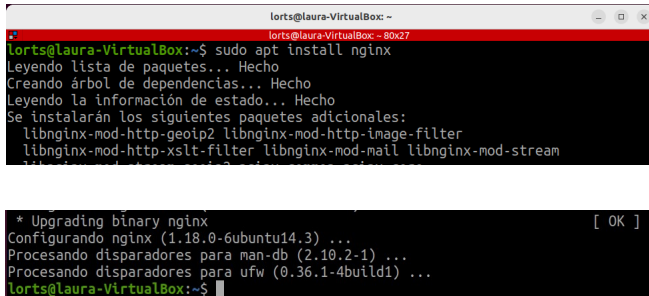
Nginx

Actividad 1

Instalar Nginx:

sudo apt update

sudo apt install nginx



```
lorts@laura-VirtualBox: ~  
lorts@laura-VirtualBox:~$ 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:  
  libnginx-mod-http-geoip2 libnginx-mod-http-image-filter  
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream  
* Upgrading binary nginx [ OK ]  
Configurando nginx (1.18.0-6ubuntu14.3) ...  
Procesando disparadores para man-db (2.10.2-1) ...  
Procesando disparadores para ufw (0.36.1-4build1) ...  
lorts@laura-VirtualBox:~$
```

Ahora podemos probar en el navegador que aparece al buscar localhost



1.1 ¿Cuál es el equivalente en Nginx a los host virtuales de apache? ¿Qué directiva se utiliza para configurar el dominio al que responderá cada VirtualHost en Nginx? y el DocumentRoot?

En Nginx, los equivalentes a los host virtuales de Apache se llaman "server blocks" o "virtual servers".

La directiva que se utiliza para configurar el dominio al que responderá cada server block es "server_name". La directiva que se utiliza para especificar el DocumentRoot de un server block es "root".

1.2 ¿Cuál es el archivo de configuración principal de nginx?

El archivo de configuración principal de Nginx se llama nginx.conf.

Actividad 2

2.1 ¿Qué permisos y propietario tiene asignado el fichero que representa el socket? ¿Por qué crees que tiene asignados dicho usuario y permisos?

Un socket suele tener asignados permisos de lectura y escritura para el usuario que lo ha creado y permisos de lectura para otros usuarios.

Esto se hace para permitir que el proceso que creó el socket pueda comunicarse a través del mismo, mientras que otros procesos pueden leer la información del socket para determinar qué proceso se está ejecutando en el servidor.

2.2 ¿Qué diferencia existe entre un socket Unix y un socket TCP?

La principal diferencia entre un socket Unix y un socket TCP es el protocolo de red que utilizan. Un socket Unix se comunica a través del sistema de archivos de Unix mediante el IPC. Por otro lado, un socket TCP es un tipo de socket que se utiliza para comunicarse a través de una red utilizando el protocolo TCP.

Actividad 3

Instala y configura un **servidor Nginx** con las siguientes especificaciones técnicas:

- Dispondrá de **1 bloque de servidor** que atenderá las peticiones dirigidas al dominio `app.ddaw.severoochoa.es`.
- Crea un documento `info.php` en el document Root. ¿Qué versión de php está ejecutando el servidor php-fpm? ¿Qué módulos/librerías tiene activas el intérprete? ¿Cuál es el archivo principal de configuración del servicio?
- Activa el `access_log` para el pool por defecto del **servicio php-fpm** y haz una captura del mismo tras atender una petición.

Instalamos php-fpm

```
laura@laura-VirtualBox: ~  
laura@laura-VirtualBox: ~ 80x24  
Configurando php8.1-opcache (8.1.2-1ubuntu2.9) ...  
Creating config file /etc/php/8.1/mods-available/opcache.ini with new version  
Configurando php8.1-cli (8.1.2-1ubuntu2.9) ...  
update-alternatives: utilizando /usr/bin/php8.1 para proveer /usr/bin/php (php)  
en modo automático  
update-alternatives: utilizando /usr/bin/phar8.1 para proveer /usr/bin/phar (pha  
r) en modo automático  
update-alternatives: utilizando /usr/bin/phar.phar8.1 para proveer /usr/bin/phar  
.phar (phar.phar) en modo automático  
Creating config file /etc/php/8.1/cli/php.ini with new version  
Configurando php8.1-fpm (8.1.2-1ubuntu2.9) ...  
Creating config file /etc/php/8.1/fpm/php.ini with new version  
Created symlink /etc/systemd/system/multi-user.target.wants/php8.1-fpm.service -  
/lib/systemd/system/php8.1-fpm.service.  
Configurando php-fpm (2:8.1+92ubuntu1) ...  
Procesando disparadores para man-db (2.10.2-1) ...  
Procesando disparadores para php8.1-cli (8.1.2-1ubuntu2.9) ...  
Procesando disparadores para php8.1-fpm (8.1.2-1ubuntu2.9) ...  
laura@laura-VirtualBox:~$ sudo apt install php-fpm
```

Y lo ejecutamos con el comando `systemctl start php8.1-fpm`

La instalación por defecto deja abierto un socket unix en la ruta `/run/php/php8.1.sock`.

Si ejecutamos el comando para ver el estatus (`systemctl status php8.1-fpm.service`) nos aparece lo siguiente

```
laura@laura-VirtualBox:/etc/nginx/sites-available$ systemctl status php8.1-fpm.service  
● php8.1-fpm.service - The PHP 8.1 FastCGI Process Manager  
   Loaded: loaded (/lib/systemd/system/php8.1-fpm.service; enabled; vendor preset: enabled)  
   Active: active (running) since Mon 2022-12-05 17:30:04 CET; 54min ago  
     Docs: man:php-fpm8.1(8)  
  Process: 4688 ExecStartPost=/usr/lib/php/php-fpm-socket-helper install /run/php/php-fpm.sock /etc/php/8.1/fpm/pool.d/www.conf  
 Main PID: 4685 (php-fpm8.1)  
   Status: "Processes active: 0, idle: 2, Requests: 1, slow: 0, Traffic: 0req/sec"  
    Tasks: 3 (limit: 4626)  
  Memory: 10.5M  
     CPU: 263ms  
   CGroup: /system.slice/php8.1-fpm.service  
           └─4685 "php-fpm: master process (/etc/php/8.1/fpm/php-fpm.conf)"  
             └─4686 "php-fpm: pool www"  
               └─4687 "php-fpm: pool www"  
dic 05 17:30:04 laura-VirtualBox systemd[1]: php8.1-fpm.service: Deactivated successfully.  
dic 05 17:30:04 laura-VirtualBox systemd[1]: Stopped The PHP 8.1 FastCGI Process Manager.  
dic 05 17:30:04 laura-VirtualBox systemd[1]: Starting The PHP 8.1 FastCGI Process Manager...  
dic 05 17:30:04 laura-VirtualBox systemd[1]: Started The PHP 8.1 FastCGI Process Manager.
```

Por lo que la instalación ha sido un éxito.

Para continuar crearemos una carpeta para ubicar los ficheros que vamos a servir en `/var/www/severoochoa`. Ubicaremos dentro un fichero, `index.php` con un `phpinfo()` como contenido.

Para el siguiente paso crearemos el archivo de configuración `/etc/nginx/sites-available/app.ddaw.severoochoa.es` y añadiremos la siguiente directiva.

```

GNU nano 6.2                                laura@laura-VirtualBox: /etc/nginx/sites-available 129x30
app.ddaw.severoochoa.es

server {
    listen 80;
    server_name app.ddaw.severoochoa.es www.app.ddaw.severoochoa.es;
    root /var/www/severoochoa;

    index index.html index.htm index.php;

    location / {
        try_files $uri $uri/ =404;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php8.1-fpm.sock;
    }

    location ~ /\.ht {
        deny all;
    }
}

[ El fichero «app.ddaw.severoochoa.es» no es de escritura ]
Ayuda  Guardar  Buscar  Cortar  Ejecutar  Ubicación  Deshacer  Poner marca
Salir  Leer fich.  Reemplazar  Pegar  Justificar  Ir a línea  Rehacer  Copiar

```

El siguiente paso sera enlazarlos con el comando `sudo ln -s /etc/nginx/sites-available/app.ddaw.severoochoa.es /etc/nginx/sites-enabled/`

Y por último añadiremos a `etc/hosts` nuestro nuevo dominio `app.ddaw.severoochoa.es`.

Y este es el resultado:

PHP Version 8.1.2-ubuntu2.9	
System	Linux laura-VirtualBox 5.15.0-56-generic #62-Ubuntu SMP Tue Nov 22 19:54:14 UTC 2022 x86_64
Build Date	Oct 19 2022 14:58:09
Build System	Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/fpm
Loaded Configuration File	/etc/php/8.1/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/fpm/conf.d
Additional .ini files parsed	/etc/php/8.1/fpm/conf.d/10-opcache.ini, /etc/php/8.1/fpm/conf.d/10-pdo.ini, /etc/php/8.1/fpm/conf.d/20-calendar.ini, /etc/php/8.1/fpm/conf.d/20-ctype.ini, /etc/php/8.1/fpm/conf.d/20-exif.ini, /etc/php/8.1/fpm/conf.d/20-fileinfo.ini, /etc/php/8.1/fpm/conf.d/20-ftp.ini, /etc/php/8.1/fpm/conf.d/20-gettext.ini, /etc/php/8.1/fpm/conf.d/20-iconv.ini, /etc/php/8.1/fpm/conf.d/20-phar.ini, /etc/php/8.1/fpm/conf.d/20-posix.ini, /etc/php/8.1/fpm/conf.d/20-readline.ini, /etc/php/8.1/fpm/conf.d/20-shmop.ini, /etc/php/8.1/fpm/conf.d/20-sockets.ini, /etc/php/8.1/fpm/conf.d/20-sysmsg.ini, /etc/php/8.1/fpm/conf.d/20-syssem.ini, /etc/php/8.1/fpm/conf.d/20-sysvshm.ini, /etc/php/8.1/fpm/conf.d/20-tokenizer.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API20210902.NTS
PHP Extension Build	API20210902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:

Como se puede ver estamos usando la versión 8.1 de PHP, el archivo de configuración es `/etc/php/8.1/fpm` y usamos como librerías: `ctype`, `date`, `exif`, `FFI`, `fileinfo`, `filter`, `ftp`, `gettext`, `hash`, `iconv`, `json`, `libxml`, `openssl`, `pcre`, `PDO`, `phar`, `pthreads`, `readline`, `reflection`, `session`, `shmop`, `sockets`, `sodium`, `spl`, `standard`, `sysvmsg`, `sysvsem`, `sysvshm`, `tokenizer`, `zendOPcache` y `zlib`.

Para la activación de `access log` vamos al fichero `/etc/php/8.1/fpm/pool.d` y añadimos las siguientes líneas:

```
laura@laura-VirtualBox: /etc/php/8.1/fpm/pool.d
laura@laura-VirtualBox: /etc/nginx/sites-available
laura@laura-VirtualBox: /etc/php/8.1/fpm/pool.d
laura@laura-VirtualBox: /etc
laura@laura-VirtualBox: /etc/php/8.1/fpm/pool.d 129x30
GNU nano 6.2 www.conf
; Note: The value must start with a leading slash (/). The value can be
; anything, but it may not be a good idea to use the .php extension or it
; may conflict with a real PHP file.
; Default Value: not set
;ping.path = /ping

; This directive may be used to customize the response of a ping request. The
; response is formatted as text/plain with a 200 response code.
; Default Value: pong
;ping.response = pong

; The access log file
; Default: not set
access.log = /var/log/$pool.access.log
access.format= "%R - %u %t \"%m %r%Q%q\""

; The access log format.
; The following syntax is allowed
; %: the '%' character
; %C: %CPU used by the request
; it can accept the following format:
; - %{user}C for user CPU only
; - %{system}C for system CPU only
; - %{total}C for user + system CPU (default)
; %d: time taken to serve the request
; it can accept the following format:

^G Ayuda      ^O Guardar   ^W Buscar    ^K Cortar    ^T Ejecutar  ^C Ubicación ^U Deshacer  ^A Poner marca
^X Salir      ^R Leer fich.^N Reemplazar ^U Pegar     ^J Justificar ^I Ir a línea ^E Rehacer   ^G Copiar
```

Cuando vamos a `/var/log/www.access.log` obtenemos lo siguientes