

COMANDOS

1. Ver el estado de un servicio

```
sudo systemctl status [servicio]
```

2. Reiniciar un servicio

```
sudo systemctl restart [servicio]
```

3. Actualizar la lista de paquetes

```
sudo apt update
```

```
sudo apt upgrade
```

4. Instalar un complemento

```
sudo apt install [nombre]
```

ejemplo: `sudo apt install php libapache2-mod-php php-mysql`

5. Listar archivos en la ruta actual

a. Normal -> `ls`

b. Archivos ocultos -> `ls -a`

c. Listar con detalles -> `ls -l`

d. Ver permisos de un archivo -> `ls -l /ruta/del/archivo`

6. Crear una carpeta

```
mkdir -p carpeta_principal/subcarpeta1/subcarpeta2
```

ejemplo -> crear carpeta para el document root

```
sudo mkdir -p /var/www/www.sitio.com/html
```

7. Cambiar los permisos de un archivo

```
chmod -R 755 /ruta/directorio
```

8. Editar un archivo

```
sudo nano [ruta]
```

Archivos de configuración

a. Apache

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

```
sudo nano /etc/apache2/sites-available/ejemplo.com.conf
```

b. Nginx

```
sudo nano /etc/nginx/sites-available/default
```

```
sudo nano /etc/nginx/sites-available/www.sitio.com
```

9. Mover un archivo

```
sudo mv /var/www/html/archivo.html /var/www/html/subcarpeta/
```

a. Verificar permisos después de mover el archivo

apache

```
sudo chown www-data:www-data /var/www/html/subcarpeta/archivo.html
```

```
sudo chmod 644 /var/www/html/subcarpeta/archivo.html
```

nginx

```
sudo chown nginx:nginx /usr/share/nginx/html/subcarpeta/archivo.html
```

```
sudo chmod 644 /usr/share/nginx/html/subcarpeta/archivo.html
```

10. Descomprimir archivo

a. `sudo apt-get install unzip`

b. `sudo unzip [archivo].zip`

11. Comandos de apache

a. Habilitar un sitio web

```
sudo a2ensite nombre-sitio.conf
```

b. Deshabilitar un sitio web

```
sudo a2dissite nombre-sitio.conf
```

c. Habilitar y deshabilitar un modulo de apache

```
sudo a2enmod rewrite
```

```
sudo a2dismod rewrite
```

12. Comandos de nginx

a. Habilitar el host virtual

```
sudo ln -s /etc/nginx/sites-available/www.sitio.com /etc/nginx/sites-enabled/
```

b. Probar la configuración

```
sudo nginx -t
```

c. Ver archivos de registro

```
sudo tail -f /var/log/nginx/access.log
```

```
sudo tail -f /var/log/nginx/error.log
```

13. Firewall (uncomplicated firewall)

a. Ver el estado

```
ufw status
```

b. Instalarlo

```
sudo apt install ufw
```

c. Lista de aplicaciones del firewall

```
sudo ufw app list
```

d. Activar aplicaciones

```
sudo ufw allow [app]
```

```
sudo ufw enable
```

14. Conectarse a la instancia EC2

```
ssh -i "labsuser.pem" ubuntu@[IPpublica]
```

15. Subir un archivo a la VPC

```
scp -i "labsuser.pem" [archivo].zip Ubuntu@[IPpublica]:[ruta]
```

UBUNTU

CREAR HOST VIRTUAL NGINX

1. Crear su directorio raíz

```
sudo mkdir -p /var/www/www.mi-sitio.com/html
```

2. Asignar permisos al directorio para que nginx lo pueda leer

```
sudo chown -R $USER:$USER /var/www/www.mi-sitio.com/html
```

```
sudo chmod -R 755 /var/www/www.mi-sitio.com
```

3. Crear un index en el directorio raíz

```
sudo nano /var/www/www.mi-sitio.com/html/index.html
```

```
<!DOCTYPE html>
```

```
<html lang="es">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Bienvenido a Nuestro Sitio Web</title>
```

```
</head>
```

```
<body>
```

```
<header>
```

```
  <h1>Bienvenido a Nuestro Sitio Web</h1>
```

```
</header>
```

</html>

4. Crear el archivo de configuración del host virtual

```
sudo nano /etc/nginx/sites-available/www.mi-sitio.com
```

```
server {  
    listen 80;  
    listen [::]:80;  
    server_name www.mi-sitio.com;  
  
    root /var/www/mi-sitio.com/html;  
    index index.html;  
    access_log /var/log/nginx/mi-sitio.com-access.log;  
    error_log /var/log/nginx/mi-sitio.com-error.log;  
  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```

5. Habilitar el host virtual

```
sudo ln -s /etc/nginx/sites-available/www.mi-sitio.com /etc/nginx/sites-enabled/
```

6. Añadir el sitio al archivo /etc/hosts

- a. `sudo nano /etc/hosts`
- b. **añadir la línea** -> `127.0.0.1 www.mi-sitio.com`

7. Verificar la configuración de nginx

```
sudo nginx -t
```

8. Reiniciar nginx

```
sudo systemctl restart nginx
```

9. Probar sitio en el navegador

```
http://www.mi-sitio.com
```

CREAR HOST VIRTUAL EN APACHE

1. Crear su directorio raíz

```
sudo mkdir -p /var/www/mi_sitio.com
```

2. Asignar permisos al directorio

```
sudo chown -R $USER:$USER /var/www/mi_sitio.com
```

```
sudo chmod -R 755 /var/www/mi_sitio.com
```

3. Crear un index en el directorio raíz

```
sudo nano /var/www.mi-sitio.com/index.html
```

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Bienvenido a Nuestro Sitio Web</title>
</head>
<body>
<header>
  <h1>Bienvenido a Nuestro Sitio Web</h1>
</header>
</html>
```

4. Crear el archivo de configuración del host virtual

```
sudo nano /etc/apache2/sites-available/mi_sitio.com.conf
```

```
<VirtualHost *:80>
    ServerAdmin webmaster@mi_sitio.com
    ServerName www.mi_sitio.com
    DocumentRoot /var/www/mi_sitio.com
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    <Directory /var/www/mi_sitio.com >
        AllowOverride
        All Require all granted
    </Directory>
</VirtualHost>
```

5. Habilitar el host virtual

```
sudo a2ensite mi_sitio.com.conf
```

6. Añadir el sitio al archivo /etc/hosts

- a. `sudo nano /etc/hosts`
- b. **añadir la línea** -> `127.0.0.1 www.mi-sitio.com`

7. Verificar la configuración de apache

```
sudo apache2ctl configtest
```

8. Reiniciar apache

`sudo systemctl restart apache2`

9. Probar sitio en el navegador

`http://www.mi-sitio.com`

SOLICITAR AUTENTIFICACION NGINX

1. Instalar htpasswd

`sudo apt install apache2-utils`

2. Crear el archivo de contraseñas

`sudo htpasswd -c /etc/nginx/.htpasswd [nombreUsuario] (-c la primera vez)`

3. Configurar sitio para usar la autentificacion básica

`sudo nano /etc/nginx/sites-available/www.mi-sitio.com`

modificar el bloque location:

```
location / {  
    try_files $uri $uri/ =404;  
  
    auth_basic "Restricted Access";  
    auth_basic_user_file /etc/nginx/.htpasswd;  
  
}
```

4. Verificar la configuración de nginx

`sudo nginx -t`

5. Reiniciar nginx

`sudo systemctl restart nginx`

6. Probar sitio en el navegador

`http://www.mi-sitio.com`

SOLICITAR AUTENTIFICACION APACHE

1. Instalar htpasswd

`sudo apt install apache2-utils`

2. Crear el archivo de contraseñas

`sudo htpasswd -c /etc/apache2/.htpasswd [nombreUsuario]`

3. Configurar sitio para usar la autentificación básica

```
sudo nano /etc/apache2/sites-available/mi_sitio.com.conf
```

Modificar el bloque directory

```
<Directory /var/www/mi_sitio.com >  
    AllowOverride  
  
    AuthType Basic  
    AuthName "Restricted Access"  
    AuthUserFile /etc/apache2/.htpasswd  
    Require valid-user  
  
</Directory>
```

4. Verificar la configuración de apache

```
sudo apache2ctl configtest
```

5. Reiniciar apache

```
sudo systemctl restart apache2
```

6. Probar sitio en el navegador

```
http://www.mi-sitio.com
```

ARCHIVO PERSONALIZADO PARA ERORES NGINX

1. Crear el archivo html en el directorio raíz (root) del host virtual

```
sudo nano /var/www/www.mi-sitio.com/html/404.html
```

```
<!DOCTYPE html>  
<html lang="es">  
  <head>  
    <meta charset="UTF-8">  
  </head>  
  <body>  
    <h1>Error 404 - Página no encontrada</h1>  
    <p>Lo sentimos, la página que buscas no existe.</p>  
    <p><a href="/">Volver al inicio</a></p>  
  </body>  
</html>
```

2. Configurar el sitio para usar el archivo de error personalizado

```
sudo nano /etc/nginx/sites-available/www.mi-sitio.com
```

añadir las siguientes líneas

```
error_page 404 /404.html;
```

```
error_page 401 /401.html;
```

```
location = /404.html {  
    root /var/www/www.mi-sitio.com/html;  
    internal;  
  
}
```

3. Verificar la configuración de nginx

```
sudo nginx -t
```

4. Reiniciar nginx

```
sudo systemctl restart nginx
```

5. Forzar error para probar

404 -> <http://www.mi-sitio.com/pagina-inexistente>

401 -> utilizar usuario incorrecto

ARCHIVO PERSONALIZADO PARA ERORES APACHE

1. Crear el archivo html en el directorio raíz (root) del host virtual

```
sudo nano /var/www/www.mi-sitio.com/404.html
```

```
<!DOCTYPE html>  
<html lang="es">  
  <head>  
    <meta charset="UTF-8">  
  </head>  
  <body>  
    <h1>Error 404 - Página no encontrada</h1>  
    <p>Lo sentimos, la página que buscas no existe.</p>  
    <p><a href="/">Volver al inicio</a></p>  
  </body>  
</html>
```


2. Configurar el sitio para usar el archivo de error personalizado

```
sudo nano /etc/apache2/sites-available/mi_sitio.com.conf
```

añadir dentro del bloque virtualhost, debajo de <directory >:

```
ErrorDocument 404 /404.html
```

```
<Location /404.html>
```

```
    SetHandler default-handler
```

```
</Location>
```

3. Verificar la configuración de apache

```
sudo apache2ctl configtest
```

4. Reiniciar apache

```
sudo systemctl restart apache2
```

5. Probar sitio en el navegador

```
http://www.mi-sitio.com
```

REDIRECCION HTTP A HTTPS NGINX

1. Crear carpeta ssl

```
sudo mkdir -p /etc/nginx/ssl
```

2. Crear certificado

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/nginx/ssl/mi-sitio.key -out /etc/nginx/ssl/mi-sitio.crt
```

3. Modificar el archivo de configuración del host virtual

```
sudo nano /etc/nginx/sites-available/www.mi-sitio.com
```

modificar el bloque server del puerto 80:

```
server {  
    listen 80;  
    server_name www.mi-sitio.com;  
  
    return 301 https://$host$request_uri;  
}
```

Añadir bloque server del puerto 443

```
server {  
    listen 443 ssl;  
    server_name www.mi-sitio.com;  
  
    ssl_certificate /etc/nginx/ssl/mi-sitio.crt;  
    ssl_certificate_key /etc/nginx/ssl/mi-sitio.key;  
  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```

4. Verificar configuración y reiniciar nginx

REDIRECCION HTTP A HTTPS APACHE

1. Crear carpeta ssl

```
sudo mkdir -p /etc/apache2/ssl
```

2. Crear certificado

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/apache2/ssl/mi-sitio.key -out /etc/apache2/ssl/mi-sitio.crt
```

3. Habilitar los modulos necesarios

```
sudo a2enmod ssl  
sudo a2enmod rewrite
```

4. Modificar el archivo de configuración del host virtual

```
sudo nano /etc/apache2/sites-available/mi-sitio.com.conf
```

modificar el bloque del puerto 80:

```
<VirtualHost *:80>  
    ServerAdmin webmaster@mi-sitio.com  
    ServerName www.mi-sitio.com  
  
    RewriteEngine On  
    RewriteCond %{HTTPS} off  
    RewriteRule ^ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]  
</VirtualHost>
```

Añadir bloque del puerto 443

```
<VirtualHost *:443>
    ServerAdmin webmaster@mi_sitio.com
    ServerName www.mi_sitio.com

    SSLEngine on
    SSLCertificateFile /etc/apache2/ssl/mi-sitio.crt
    SSLCertificateKeyFile /etc/apache2/ssl/mi-sitio.key

    DocumentRoot /var/www/mi_sitio.com

    <Directory /var/www/mi_sitio.com >
        AllowOverride
        All Require all granted
    </Directory>
</VirtualHost>
```

5. Habilitar el sitio

```
sudo a2ensite mi-sitio.com.conf
```

6. Comprobar errores y reiniciar apache

** errores de certificados

1. Verificar si el certificado y la clave coinciden, si están emparejados correctamente

a. Comparar la clave privada con el certificado ssl

```
openssl x509 -noout -modulus -in /etc/apache2/ssl/mi-sitio.crt | openssl md5
openssl rsa -noout -modulus -in /etc/apache2/ssl/mi-sitio.key | openssl md5
```

b. Comparar la clave privada con la solicitud de firma de certificado

```
openssl req -noout -modulus -in /etc/apache2/ssl/mi-sitio.csr | openssl md5
openssl rsa -noout -modulus -in /etc/apache2/ssl/mi-sitio.key | openssl md5
```

2. Si no coinciden

a. La clave privada con el certificado ssl

Generar una nueva clave privada

```
sudo openssl genpkey -algorithm RSA -out /etc/apache2/ssl/mi-sitio.key
```

Generar una nueva solicitud de firma de certificado

```
sudo openssl req -new -key /etc/apache2/ssl/mi-sitio.key -out
/etc/apache2/ssl/mi-sitio.csr
```

Generar un certificado autofirmado

```
sudo openssl x509 -req -days 365 -in /etc/apache2/ssl/mi-sitio.csr -signkey  
/etc/apache2/ssl/mi-sitio.key -out /etc/apache2/ssl/mi-sitio.crt
```

b. La clave privada con la solicitud de firma de certificado

Genera una nueva CSR con la clave privada que tienes

```
sudo openssl req -new -key /etc/apache2/ssl/mi-sitio.key -out  
/etc/apache2/ssl/mi-sitio-new.csr
```

si no hay clave privada o no se puede generar

Generar una nueva clave privada:

```
sudo openssl genpkey -algorithm RSA -out /etc/apache2/ssl/mi-sitio.key
```

Generar una nueva CSR con la nueva clave privada

```
sudo openssl req -new -key /etc/apache2/ssl/mi-sitio.key -out  
/etc/apache2/ssl/mi-sitio.csr
```

Obtener un nuevo certificado

```
sudo openssl x509 -req -days 365 -in /etc/apache2/ssl/mi-sitio.csr -signkey  
/etc/apache2/ssl/mi-sitio.key -out /etc/apache2/ssl/mi-sitio.crt
```

WINDOWS

CREAR HOST VIRTUAL EN WINDOWS

1. Crear la carpeta del sitio -> C:\xampp\htdocs\miproyecto
2. Abrir C:\xampp\apache\conf\extra\httpd-vhosts.conf (bloc de notas como administrador)
<VirtualHost *:80>
 DocumentRoot "C:/xampp/htdocs/miproyecto"
 ServerName miproyecto.local

 <Directory "C:/xampp/htdocs/miproyecto">
 Options Indexes FollowSymLinks
 AllowOverride All
 Require all granted
 </Directory>
</VirtualHost>
3. Abrir C:\Windows\System32\drivers\etc\hosts y añadir
127.0.0.1 miproyecto.local
4. Crear index.html en la carpeta C:\xampp\htdocs\miproyecto
5. Reiniciar apache

AUTENTIFICACION Y CONTROL DE ACCESO CON APACHE EN WINDOWS

1. Crear carpeta C:\xampp\apache\seguridad
2. Desde la Shell de xampp ejecutar
htpasswd -c C:\xampp\apache\seguridad\httpasswd usuario1
se puede ejecutar desde ahí porque
 - Incluye automáticamente la ruta a las herramientas de Apache (bin) en el entorno.
 - En cmd no puedes ejecutarlo a menos que agregues la ruta C:\xampp\apache\bin a las variables de entorno.
3. Configurar autenticación en httpd-vhosts.conf
<Directory "C:/xampp/htdocs/miproyecto">
 Options Indexes FollowSymLinks
 AllowOverride All
 AuthType Basic
 AuthName "Zona restringida"
 AuthUserFile "C:/xampp/apache/seguridad/.httpasswd"
 Require user usuario1
</Directory>

4. Para crear un segundo usuario

htpasswd C:\xampp\apache\seguridad\htpasswd usuario2

5. Reiniciar apache

ERRORES PERSONALIZADOS EN WINDOWS

1. Crear archivo C:\xampp\htdocs\miproyecto\error401.html

```
<!DOCTYPE html>
<html>
<head><title>Error 401</title></head>
<body>
<h1>Acceso no autorizado</h1>
<p>Debes iniciar sesión para acceder a esta página.</p>
</body>
</html>
```

2. Añadir en el archivo httpd-vhosts.conf en el bloque directory

ErrorDocument 401 /error401.html

PROTOCOLO HTTPS Y REDIRECCION, APACHE, WINDOWS

1. En C:\xampp\apache\conf\httpd.conf -> descomentar

```
LoadModule ssl_module modules/mod_ssl.so
Include conf/extra/httpd-ssl.conf
```

2. En httpd-ssl.conf

```
SSLEngine on
SSLCertificateFile "conf/ssl.crt/server.crt"
SSLCertificateKeyFile "conf/ssl.key/server.key"
```

3. Desde la Shell de xampp

```
openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyout
C:\xampp\apache\conf\ssl.key\clave.key -out
C:\xampp\apache\conf\ssl.crt\certificado.crt
```

4. Verificar que se han creado los archivos en conf/ssl.key y conf/ssl.crt

5. Añadir bloque 443 C:\xampp\apache\conf\extra\httpd-vhosts.conf

```
<VirtualHost *:443>
    ServerName miproyecto.local
    DocumentRoot "C:/xampp/htdocs/miproyecto"

    SSLEngine on
```

```
SSLCertificateFile "conf/ssl.crt/certificado.crt"  
SSLCertificateKeyFile "conf/ssl.key/clave.key"
```

```
<Directory "C:/xampp/htdocs/miproyecto">  
    Options Indexes FollowSymLinks  
    AllowOverride All  
    Require all granted  
</Directory>  
</VirtualHost>
```

6. Reiniciar apache

7. Redirigir de http a https

En el archivo httpd-vhosts.conf

```
<VirtualHost *:80>  
    ServerName miproyecto.local  
    DocumentRoot "C:/xampp/htdocs/miproyecto"
```

```
<Directory "C:/xampp/htdocs/miproyecto">  
    Options Indexes FollowSymLinks  
    AllowOverride All  
    Require all granted  
</Directory>
```

```
    Redirect permanent / https://miproyecto.local/  
</VirtualHost>
```

8. Reiniciar apache