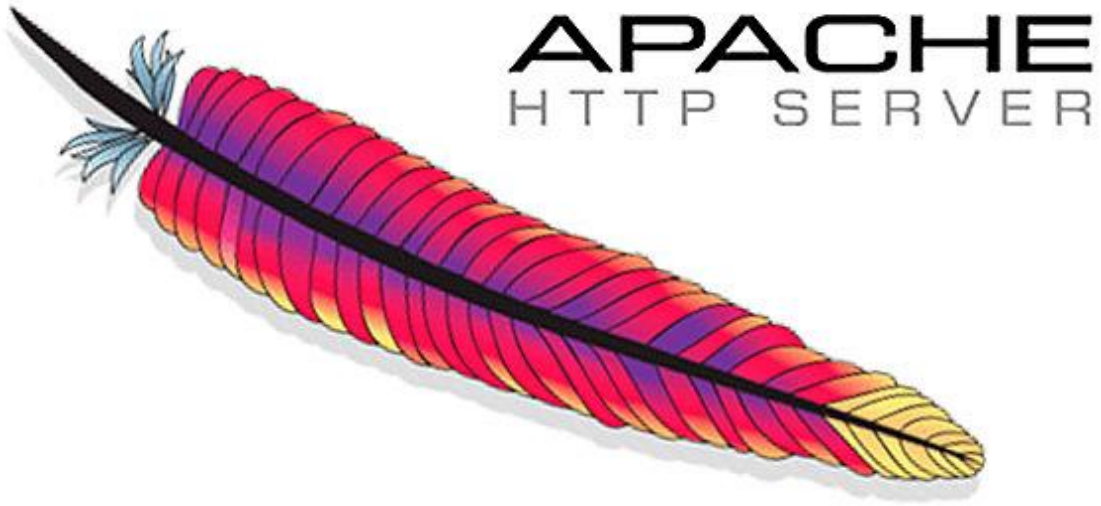
IES Valle Inclán

SERVIDOR WEB



Carlos González Martín

Contenido

1. Cambiamos el nombre a la maquina

```
root@debian-12:~# hostnamectl set-hostname apache2
root@debian-12:~# exit_
```

```
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
root@apache2:~#
```

2. Instalamos servicios

Ahora instalamos el servicio apache2

```
root@apache2:~# apt update ; apt install apache2
Des:1 http://security.debian.org/debian-security bookworm-security InRelease [48,0 kB]
Des:2 http://deb.debian.org/debian bookworm InRelease [151 kB]
Des:3 http://deb.debian.org/debian bookworm-updates InRelease [55,4 kB]
Des:4 http://security.debian.org/debian-security bookworm-security/main Sources [139 kB]
Des:5 http://security.debian.org/debian-security bookworm-security/main amd64 Packages [242 kB]
Des:6 http://security.debian.org/debian-security bookworm-security/main Translation-en [143 kB]
Des:7 http://deb.debian.org/debian bookworm/main Sources [9.496 kB]
```

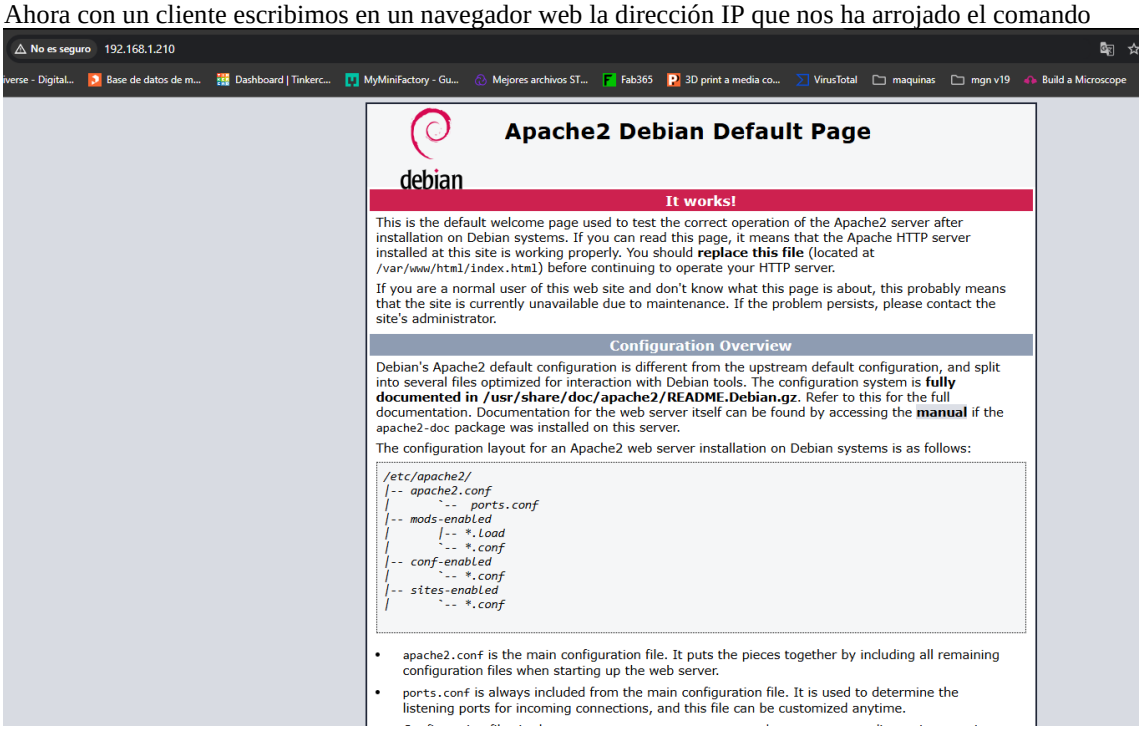
Comprobación básica de apache2

Ahora lo que haremos será comprobar si está el servidor apache con el navegador de algún cliente, en mi caso la maquina real

```
root@apache2:~# service apache2 status
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-01-26 19:26:24 CET; 1min 1s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1445 (apache2)
      Tasks: 55 (limit: 1098)
     Memory: 9.1M
        CPU: 46ms
     CGroup: /system.slice/apache2.service
             ├─1445 /usr/sbin/apache2 -k start
             ├─1447 /usr/sbin/apache2 -k start
             └─1448 /usr/sbin/apache2 -k start

ene 26 19:26:24 apache2 systemd[1]: Starting apache2.service - The Apache HTTP Server...
ene 26 19:26:24 apache2 apachectl[1444]: AH00557: apache2: apr_sockaddr_info_get() failed for apache2
ene 26 19:26:24 apache2 apachectl[1444]: AH00558: apache2: Could not reliably determine the server's f
ene 26 19:26:24 apache2 systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-17/17 (END)
^C
root@apache2:~# ip -c 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,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:11:c7:ba brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.210/24 brd 192.168.1.255 scope global dynamic enp0s3
```

3. Comprobación inicial

Ahora con un cliente escribimos en un navegador web la dirección IP que nos ha arrojado el comando



4. Crearemos un segundo sitio web

Ahora creamos un segundo sitio web y también cambiaremos el nombre a la carpeta por defecto

```
root@apache2:/var/www# ls
html
root@apache2:/var/www# mv html sitio1
root@apache2:/var/www# ls
sitio1
root@apache2:/var/www# cp sitio1 sitio2
cp: -r not specified; omitting directory 'sitio1'
root@apache2:/var/www# cp -r sitio1 sitio2
root@apache2:/var/www# ls -la
total 16
drwxr-xr-x  4 root root 4096 ene 26 20:21 .
drwxr-xr-x 12 root root 4096 ene 26 19:26 ..
drwxr-xr-x  2 root root 4096 ene 26 19:26 sitio1
drwxr-xr-x  2 root root 4096 ene 26 20:21 sitio2
```

Ahora nos iremos al sitio1 y cambiaremos un poco el index.html

```
root@apache2:/var/www# cd sitio1
root@apache2:/var/www/sitio1# nano index.html
```

```
GNU nano 7.2                                                    index.html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-trans
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Sitio 1</title>
    <style type="text/css" media="screen">
  * {
    margin: 0px 0px 0px 0px;
    padding: 0px 0px 0px 0px;
  }
```

Ahora nos cambiaremos el sitio2 y cambiamos el index

```
root@apache2:/var/www/sitio1# cd ../sitio2
root@apache2:/var/www/sitio2# nano index.html_
```

```
GNU nano 7.2                                                    index.html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtm
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Sitio 2</title>
    <style type="text/css" media="screen">
  * {
```

Ahora cambiamos el usuario y grupo de las carpetas creadas anteriormente

```
root@apache2:/var/www# chown www-data:www-data sitio1
root@apache2:/var/www# chown www-data:www-data sitio2
root@apache2:/var/www# ls -la
total 16
drwxr-xr-x  4 root     root     4096 ene 26 20:21 .
drwxr-xr-x 12 root     root     4096 ene 26 19:26 ..
drwxr-xr-x  2 www-data www-data 4096 ene 26 20:40 sitio1
drwxr-xr-x  2 www-data www-data 4096 ene 26 20:47 sitio2
root@apache2:/var/www#
```

Ahora nos iremos a habilitar los sitios

```
root@apache2:/etc/apache2/sites-available# cp 000-default.conf sitio1.conf
root@apache2:/etc/apache2/sites-available# cp 000-default.conf sitio2.conf
root@apache2:/etc/apache2/sites-available# ls -la
total 28
drwxr-xr-x 2 root root 4096 ene 26 20:53 .
drwxr-xr-x 8 root root 4096 ene 26 19:26 ..
-rw-r--r-- 1 root root 1286 sep 29 20:51 000-default.conf
-rw-r--r-- 1 root root 6195 oct  4 17:16 default-ssl.conf
-rw-r--r-- 1 root root 1286 ene 26 20:53 sitio1.conf
-rw-r--r-- 1 root root 1286 ene 26 20:53 sitio2.conf
root@apache2:/etc/apache2/sites-available#
```

Una vez creados los archivos cambiaremos el sitio1.conf

```
 GNU nano 7.2                                                 sitio1.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/sitio1_

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

Ahora modificamos el sitio2

```
 GNU nano 7.2                                                 sitio2.conf
<VirtualHost *:8080>
        # 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/sitio2

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

Ahora deshabilitamos el sitio por defecto y habilitamos el sitio1 y el sitio2

```
root@apache2:/etc/apache2/sites-available# a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache2:/etc/apache2/sites-available# a2ensite sitio1.conf
Enabling site sitio1.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache2:/etc/apache2/sites-available# a2ensite sitio2.conf
Enabling site sitio2.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache2:/etc/apache2/sites-available#
```

Ahora modificamos los puertos de escucha de apache en el siguiente directorio

```
root@apache2:/etc/apache2/sites-available# cd ..
root@apache2:/etc/apache2# ls
apache2.conf  conf-available  conf-enabled  envvars  magic  mods-available  mods-enabled  ports.conf
root@apache2:/etc/apache2# nano ports.conf
```

Añadimos la línea 6 que es el puerto de escucha del segundo sitio web

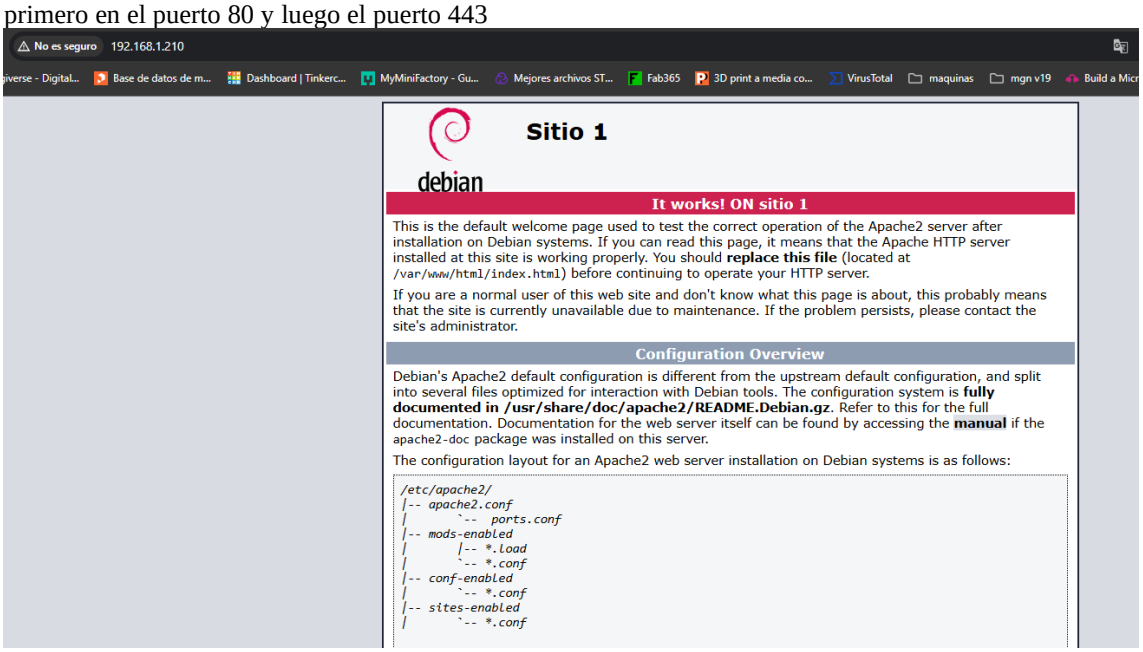
```
  GNU nano 7.2                                                    ports.conf
 1 # If you just change the port or add more ports here, you will likely also
 2 # have to change the VirtualHost statement in
 3 # /etc/apache2/sites-enabled/000-default.conf
 4
 5 Listen 80
 6 Listen 8080
 7
 8 <IfModule ssl_module>
 9         Listen 443
10 </IfModule>
11
12 <IfModule mod_gnutls.c>
13         Listen 443
14 </IfModule>
15
```

Y reiniciamos el servicio

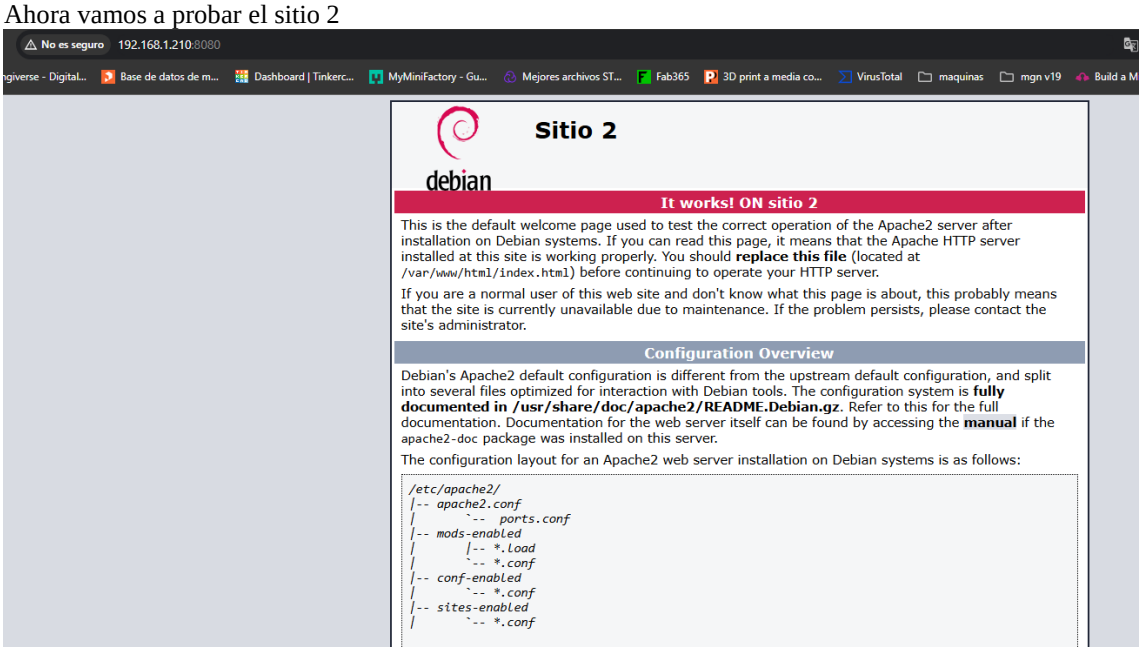
```
root@apache2:/etc/apache2# service apache2 restart
root@apache2:/etc/apache2# service apache2 status
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-01-26 21:24:05 CET; 3s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 1884 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 1888 (apache2)
      Tasks: 55 (limit: 1098)
     Memory: 8.6M
        CPU: 31ms
     CGroup: /system.slice/apache2.service
             ├─1888 /usr/sbin/apache2 -k start
             ├─1889 /usr/sbin/apache2 -k start
             └─1890 /usr/sbin/apache2 -k start

ene 26 21:24:05 apache2 systemd[1]: Starting apache2.service - The Apache HTTP Server...
ene 26 21:24:05 apache2 apachectl[1887]: AH00557: apache2: apr_sockaddr_info_get() failed for apache2
ene 26 21:24:05 apache2 apachectl[1887]: AH00558: apache2: Could not reliably determine the server's fu
ene 26 21:24:05 apache2 systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-18/18 (END)
```

5. Comprobaciones

Ahora vamos a probar con el primer sitio web, no tenemos que poner el puerto ya que por defecto prueba primero en el puerto 80 y luego el puerto 443



Ahora vamos a probar el sitio 2



6. Openssl

Ahora que los dos funcionan, vamos a configurar el sitio 2 con SSL o certificados
Vamos a instalar el paquete Openssl

```
root@apache2:/etc/apache2# apt update ; apt install openssl
Obj:1 http://security.debian.org/debian-security bookworm-security InRelease
Obj:2 http://deb.debian.org/debian bookworm InRelease
Obj:3 http://deb.debian.org/debian bookworm-updates InRelease
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se pueden actualizar 47 paquetes. Ejecute «apt list --upgradable» para verlos.
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libssl3
Se actualizarán los siguientes paquetes:
  libssl3 openssl
2 actualizados, 0 nuevos se instalarán, 0 para eliminar y 45 no actualizados.
Se necesita descargar 3.452 kB de archivos.
Se utilizarán 1.024 B de espacio de disco adicional después de esta operación.
```

Ahora creamos la clave privada

```
root@apache2:~# openssl genpkey -algorithm RSA -out /etc/ssl/private/server.key
.+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.....+.....
...+...+..+.....+...+.....+..+.........+.........+..+....+.....+.+.............+....+.........+......+......+.....+...+.....+......+.............
....+...........+..+.+..+.........+.....+.........+.............+..+.........+........+....+...........+.......+..+....+...........+......+.......+..+...+
.......+...+..+.....+.....+.....+.....+.....+...+.....+...+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
....+...+........+.....+..+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.....+++++++++++++++++++++++++++++++++++++++++++++++++
++++++*.....+..+.............+.+..+....+...+..+......+.............+.........+..+.............+..+....+++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++
root@apache2:~# _
```

Ahora generamos el certificado
Creamos el directorio csr

```
root@apache2:~# openssl req -new -key /etc/ssl/private/server.key -out /etc/ssl/csr/server.csr
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,
If you enter '.', 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) []:alcala
Organization Name (eg, company) [Internet Widgits Pty Ltd]:google
Organizational Unit Name (eg, section) []:patata
Common Name (e.g. server FQDN or YOUR name) []:patata
Email Address []:patata@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:patata
An optional company name []:patata
root@apache2:~# _
```

```
root@apache2:~# openssl x509 -req -in /etc/ssl/csr/server.csr -signkey /etc/ssl/private/server.key -out /etc/ssl/certs/server.crt
Certificate request self-signature ok
subject=C = ES, ST = madrid, L = alcala, O = google, OU = patata, CN = patata, emailAddress = patata@gmail.com
root@apache2:~#
```

Una vez que tenemos los siguientes archivos vamos a habilitar el módulo SSL

```
root@apache2:~# a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
root@apache2:~# service apche2 restart
Failed to restart apche2.service: Unit apche2.service not found.
root@apache2:~# service apache2 restart
root@apache2:~# _
```

Vamos a configurar el sitio de apache, primero deshabilitando el sitio2

```
root@apache2:~# a2dissite sitio2.conf
Site sitio2 disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache2:~# systemctl apache2 restart
Unknown command verb apache2.
root@apache2:~# service apache2 restart
root@apache2:~# _
```

Ahora eliminamos el sitio2 y creamos un nuevo sitio con el default-ssl.conf

```
root@apache2:/etc/apache2/sites-available# rm sitio2.conf
root@apache2:/etc/apache2/sites-available# cp default-ssl.conf sitio2.conf
root@apache2:/etc/apache2/sites-available# ls -la
total 32
drwxr-xr-x 2 root root 4096 ene 26 22:42 .
drwxr-xr-x 8 root root 4096 ene 26 21:23 ..
-rw-r--r-- 1 root root 1286 sep 29 20:51 000-default.conf
-rw-r--r-- 1 root root 6195 oct  4 17:16 default-ssl.conf
-rw-r--r-- 1 root root 1288 ene 26 20:54 sitio1.conf
-rw-r--r-- 1 root root 6195 ene 26 22:42 sitio2.conf
root@apache2:/etc/apache2/sites-available# _
```

Ahora modificamos el archivo sitio2.conf

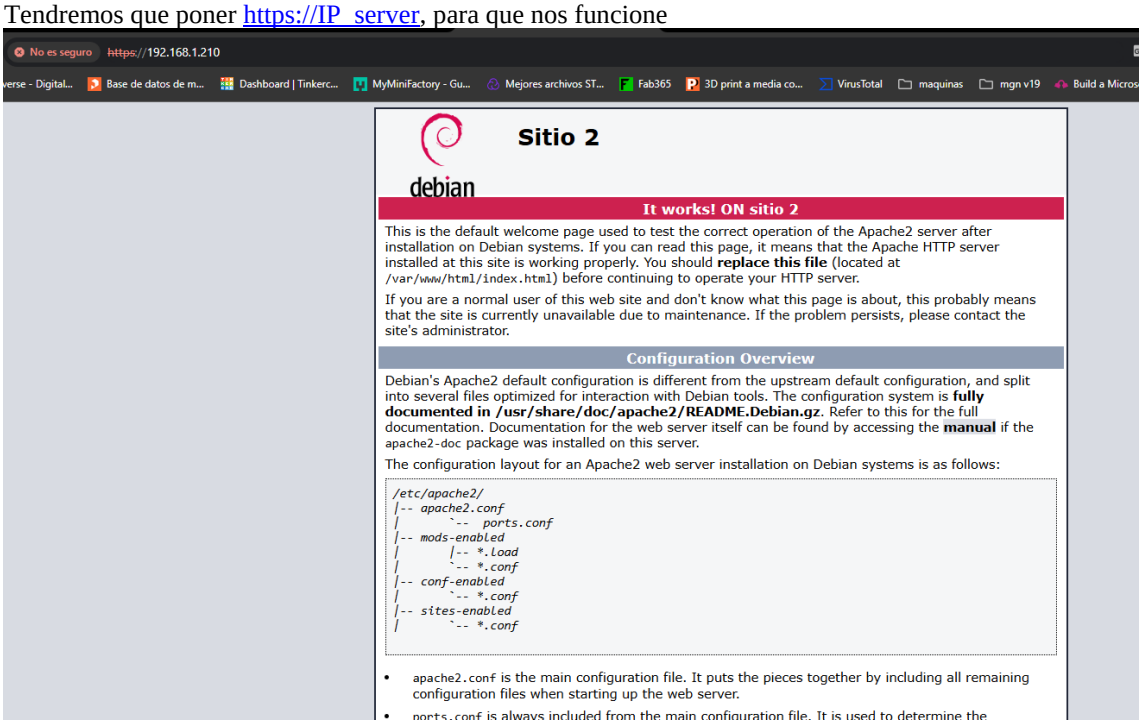
```
 GNU nano 7.2                                                  sitio2.conf
 1 <VirtualHost *:443>
 2         ServerAdmin webmaster@localhost
 3
 4         DocumentRoot /var/www/sitio2
 5
 6         # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
 7         # error, crit, alert, emerg.
 8         # It is also possible to configure the loglevel for particular
 9         # modules, e.g.
10         #LogLevel info ssl:warn
11
12         ErrorLog ${APACHE_LOG_DIR}/error.log
13         CustomLog ${APACHE_LOG_DIR}/access.log combined
14
15         # For most configuration files from conf-available/, which are
16         # enabled or disabled at a global level, it is possible to
17         # include a line for only one particular virtual host. For example the
18         # following line enables the CGI configuration for this host only
19         # after it has been globally disabled with "a2disconf".
20         #Include conf-available/serve-cgi-bin.conf
21
22         #   SSL Engine Switch:
23         #   Enable/Disable SSL for this virtual host.
24         SSLEngine on
25
26         #   A self-signed (snakeoil) certificate can be created by installing
27         #   the ssl-cert package. See
28         #   /usr/share/doc/apache2/README.Debian.gz for more info.
29         #   If both key and certificate are stored in the same file, only the
30         #   SSLCertificateFile directive is needed.
31         SSLCertificateFile      /etc/ssl/certs/server.crt
32         SSLCertificateKeyFile   /etc/ssl/private/server.key
33
34         #   Server Certificate Chain:
```

Ahora habilitamos el sitio SSL

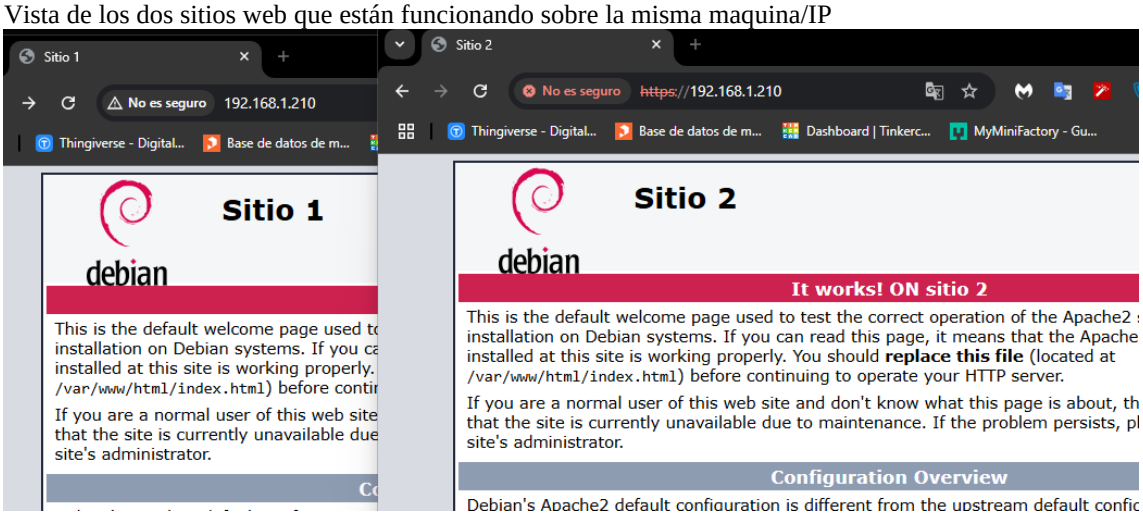
```
root@apache2:/etc/apache2/sites-available# a2ensite sitio2.conf
Enabling site sitio2.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@apache2:/etc/apache2/sites-available# service apache2 restart
root@apache2:/etc/apache2/sites-available# service apache2 status
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-01-26 22:46:10 CET; 5s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 2482 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 2487 (apache2)
      Tasks: 55 (limit: 1098)
     Memory: 9.8M
        CPU: 42ms
     CGroup: /system.slice/apache2.service
             ├─2487 /usr/sbin/apache2 -k start
             ├─2488 /usr/sbin/apache2 -k start
             └─2489 /usr/sbin/apache2 -k start

ene 26 22:46:10 apache2 systemd[1]: Starting apache2.service - The Apache HTTP Server...
ene 26 22:46:10 apache2 apachectl[2486]: AH00557: apache2: apr_sockaddr_info_get() failed for ap
ene 26 22:46:10 apache2 apachectl[2486]: AH00558: apache2: Could not reliably determine the serv
ene 26 22:46:10 apache2 systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-18/18 (END)
```

7. Comprobaciones

Tendremos que poner https://IP_server, para que nos funcione



Vista de los dos sitios web que están funcionando sobre la misma maquina/IP



8. Conclusión

Apache2 es un servidor web robusto y adaptable, ampliamente utilizado por su confiabilidad y capacidad para gestionar sitios web de alto tráfico. Su naturaleza de código abierto, multiplataforma y modular lo convierte en una opción versátil para diversos proyectos web. Además, su escalabilidad y características de seguridad lo hacen adecuado tanto para sitios web pequeños como para grandes plataformas en línea. Con una comunidad activa y documentación completa, Apache2 ofrece un entorno sólido y fácil de usar para implementar y administrar servidores web.