

HTTP Linux Apache (I)

ÍNDICE

1.	Instalación.....	3
2.	Administración del servicio	3
3.	Ficheros de configuración.....	3
3.1.	Configuraciones globales	3
3.2.	Directorios de configuración de módulos.....	5
3.3.	Directorios de configuración de sitios o servidores virtuales (virtualhost).....	6
4.	Directivas	9
5.	Etiquetas de directorio	12
5.1.	Opciones de directorio.....	12
5.1.1.	Indexes.....	13
5.1.2.	FollowSymLinks.....	14
5.1.3.	MultiViews.....	15
5.1.4.	AllowOverride.....	17
5.2.	Acceso directorios.....	17
5.3.	Otros comandos o directivas.....	19
5.3.1.	DirectoryIndex.....	19
6.	Otras configuraciones básicas de interés.....	20
6.1.	Alias.....	20
6.2.	Redirect	22
6.3.	ErrorDocument	22
7.	VirtualHost	22

1. Instalación

La instalación del servidor Apache sobre Linux Debian Server puede hacerse desde el repositorio, por lo que ejecutaremos:

```
apt-get install apache2
```

2. Administración del servicio

Cada vez que cambien los ficheros de configuración hay que indicarle que lea de nuevo dichos ficheros para que se apliquen los cambios, por lo que deberíamos reiniciar el servicio.

```
service apache2 stop  
service apache2 start  
service apache2 restart
```

3. Ficheros de configuración

3.1. Configuraciones globales

El principal fichero de configuración es `/etc/apache2/apache2.conf` y contiene un conjunto de comentarios, aclaraciones y directivas que determinan el comportamiento del servidor y variables de entorno a aplicar.

Al igual que sucede en la configuración de otros servicios, pero que en Apache es algo **MUY IMPORTANTE**, es que el fichero `/etc/apache2/apache2.conf` contiene otros ficheros y directorios de configuración mediante la directiva `Include`, los cuales debemos conocer y que veremos en cada uno de los siguientes epígrafes.

```

GNU nano 2.2.4          Fichero: /etc/apache2/apache2.conf

# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn

# Include module configuration:
Include mods-enabled/*.load
Include mods-enabled/*.conf

# Include all the user configurations:
Include httpd.conf

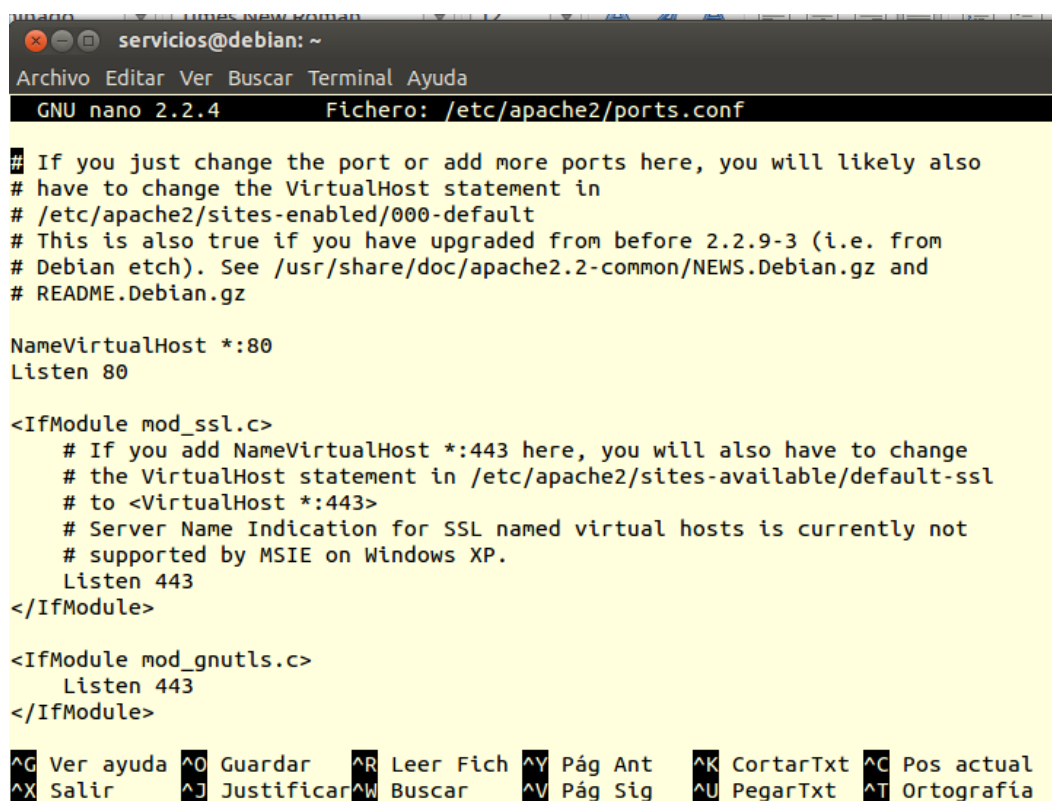
# Include ports listing
Include ports.conf

# Include generic snippets of statements
Include conf.d/

# Include the virtual host configurations:
Include sites-enabled/

```

❖ `/etc/apache2/ports.conf`. Define los puertos de escucha del servidor, que por defecto es el valor 80 para http y 443 para https.



```

servicios@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.4          Fichero: /etc/apache2/ports.conf

# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default
# This is also true if you have upgraded from before 2.2.9-3 (i.e. from
# Debian etch). See /usr/share/doc/apache2.2-common/NEWS.Debian.gz and
# README.Debian.gz

NameVirtualHost *:80
Listen 80

<IfModule mod_ssl.c>
    # If you add NameVirtualHost *:443 here, you will also have to change
    # the VirtualHost statement in /etc/apache2/sites-available/default-ssl
    # to <VirtualHost *:443>
    # Server Name Indication for SSL named virtual hosts is currently not
    # supported by MSIE on Windows XP.
    Listen 443
</IfModule>

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

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía

```

❖ **/etc/apache2/httpd.conf.** Fichero incluido por compatibilidad con otras versiones de Apache, pero en la versión actual se encuentra vacío.

❖ **/etc/apache2/conf.d.** Directorio para configuraciones adicionales de errores y otras aplicaciones.

```
root@debian:~# ls -l /etc/apache2/conf.d
total 16
-rw-r--r-- 1 root root 269 mar 22 2011 charset
-rw-r--r-- 1 root root 3296 mar 22 2011 localized-error-pages
-rw-r--r-- 1 root root 143 mar 22 2011 other-vhosts-access-log
-rw-r--r-- 1 root root 1424 mar 22 2011 security
root@debian:~# █
```

3.2. Directorios de configuración de módulos

❖ **/etc/apache2/mods-enabled/.** Directorio con los módulos de configuración habilitados o activos. Son enlaces simbólicos a los ficheros de `/etc/apache2/mods-available`, contenedor de los módulos disponibles. Los módulos son una serie de programas u opciones complementarias que el servidor web Apache dispone para añadir funcionalidades adicionales mediante su utilización integrada con Apache. Dichos módulos se localizan en los dos directorios anteriores, dividiéndose, en la mayoría de los casos en un fichero con la instrucción de carga (`*.load`) y otro con los comandos de configuración (`*.conf`), aunque este último puede no existir para impedir la configuración de algún módulo obligando a realizar un uso determinado.

```
root@debian:/etc/apache2/mods-enabled# ls
alias.conf      authz_host.load  deflate.conf    mime.load       setenvif.load
alias.load      authz_user.load  deflate.load    negotiation.conf status.conf
auth_basic.load autoindex.conf  dir.conf        negotiation.load status.load
authn_file.load autoindex.load  dir.load        reqtimeout.conf reqtimeout.load
authz_default.load cgid.conf       env.load        reqtimeout.conf setenvif.conf
authz_groupfile.load cgid.load       mime.conf       setenvif.conf
root@debian:/etc/apache2/mods-enabled# █
```

Por ejemplo, podemos acceder a `dir.load` y observar que el módulo para el control por defecto de *DirectoryIndex* es `/usr/lib/apache2/modules/mod_dir.so`:

```
root@debian:/etc/apache2/mods-enabled# cat dir.load
LoadModule dir_module /usr/lib/apache2/modules/mod_dir.so
```

Y si accedemos a su fichero de configuración (`dir.conf`) podríamos modificar dicho valor por defecto al valor deseado.

```
root@debian:/etc/apache2/mods-enabled# cat dir.conf
<IfModule mod_dir.c>

    DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm

</IfModule>
```

❖ **`/etc/apache2/mods-available/`**. Directorio de configuración de módulos disponibles que contiene los ficheros de extensión `.load` y `.conf`.

3.3. Directorios de configuración de sitios o servidores virtuales (virtualhost)

❖ **`/etc/apache2/sites-available/`**. Directorio de configuración de los sitios virtuales. Contiene los ficheros de configuración de los sitios virtuales disponibles y, por defecto, está creado el fichero `000-default.conf` con la configuración del denominado servidor virtual por defecto y el fichero `default-ssl.conf` para conexiones http seguras por defecto.

❖ **`/etc/apache2/sites-enabled`**. Directorio con los ficheros de configuración de los sitios, hosts o servidores virtuales activos. Contiene el fichero `000-default`, que es un enlace simbólico al fichero *default* de `/etc/apache2/sites-available`, con la configuración del servidor virtual por defecto. El sitio o servidor principal se dedica a atender las peticiones si no se configuran servidores virtuales. No es nada más que un grupo de páginas web enlazadas y que se ofrecerán a la red por Apache.

Los servidores virtuales nos permiten tener distintos sitios o alojamientos web en un mismo servidor o equipo. Apache soporta sitios o servidores virtuales basados en IP, nombres y puertos, configurándose bajo las etiquetas:

```
<VirtualHost> ... </VirtualHost>
```

La instalación por defecto de Apache sobre Linux tiene habilitado un servidor virtual por defecto (*default*).

```
GNU nano 2.2.4      Fichero: /etc/apache2/sites-enabled/000-default
<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
    <Directory "/usr/lib/cgi-bin">
        AllowOverride None
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
        Order allow,deny
        Allow from all
    </Directory>
```

Respecto a las variables de entorno, el fichero de configuración principal de Apache, entre otras, describe la que controla el PID (`${APACHE_PID_FILE}`) y las que identifican el usuario (`${APACHE_RUN_USER}`) y grupo (`${APACHE_RUN_GROUP}`) de sus procesos, así como otras variables importantes para el tratamiento de errores. Para los casos anteriores se describen dichas variables en el fichero de variables de entorno `/etc/apache2/envvars`.

Ampliación de conocimientos de sistemas operativos

Cuando se inicia Apache se registra su número identificador de proceso o **PID** (Process IDentifier), que es un número entero usado por el kernel del sistema operativo Linux para identificar un proceso de forma única. Para asignar el PID, el kernel utiliza internamente una variable global que se va incrementando con cada nuevo proceso creado.

```
GNU nano 2.2.4          Fichero: /etc/apache2/apache2.conf

# Do NOT add a slash at the end of the directory path.
#
#ServerRoot "/etc/apache2"

#
# The accept serialization lock file MUST BE STORED ON A LOCAL DISK.
#
LockFile ${APACHE_LOCK_DIR}/accept.lock

#
# PidFile: The file in which the server should record its process
# identification number when it starts.
# This needs to be set in /etc/apache2/envvars
#
PidFile ${APACHE_PID_FILE}

#
# Timeout: The number of seconds before receives and sends time out.
#

# These need to be set in /etc/apache2/envvars
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}
```

```
GNU nano 2.2.4          Fichero: /etc/apache2/envvars

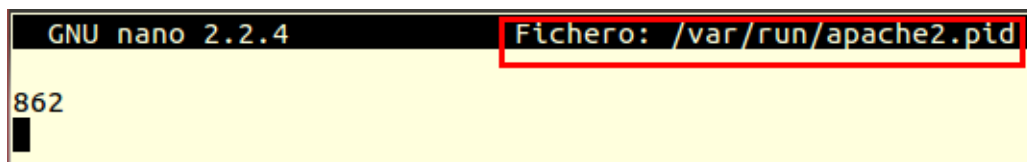
# envvars - default environment variables for apache2ctl

# this won't be correct after changing uid
unset HOME

# for supporting multiple apache2 instances
if [ "${APACHE_CONFDIR##/etc/apache2-}" != "${APACHE_CONFDIR}" ] ; then
    SUFFIX="-${APACHE_CONFDIR##/etc/apache2-}"
else
    SUFFIX=
fi

# Since there is no sane way to get the parsed apache2 config in scripts, some
# settings are defined via environment variables and then used in apache2ctl,
# /etc/init.d/apache2, /etc/logrotate.d/apache2, etc.
export APACHE_RUN_USER=www-data
export APACHE_RUN_GROUP=www-data
export APACHE_PID_FILE=/var/run/apache2${SUFFIX}.pid
export APACHE_RUN_DIR=/var/run/apache2${SUFFIX}
export APACHE_LOCK_DIR=/var/lock/apache2${SUFFIX}
# Only /var/log/apache2 is handled by /etc/logrotate.d/apache2.
export APACHE_LOG_DIR=/var/log/apache2${SUFFIX}
```


En la imagen anterior podemos apreciar que el valor de grupo y usuario utilizados por los procesos de Apache es `www-data` y el fichero contenedor del PID es `/var/run/apache2.pid`, donde el valor concreto del apache en ejecución utilizado para realizar las capturas de pantalla es:



```
GNU nano 2.2.4 Fichero: /var/run/apache2.pid
862
█
```

4. Directivas

Las directivas de Apache nos permiten realizar la configuración de distintos aspectos de funcionamiento, es decir, son una serie de comandos con los que realizar la configuración personalizada del servidor web Apache.

Las directivas se definen en los ficheros de configuración y cuando una directiva no aparece de forma explícita se tomará su valor por defecto en la configuración del servidor web.

Dentro del proyecto Apache podemos encontrar información sobre sus directivas en los enlaces:

<http://httpd.apache.org/docs/2.2/mod/directives.html>

<http://httpd.apache.org/docs/2.2/mod/quickreference.html>

Algunas de las directivas más importantes son:

<code>ServerRoot</code>	<code><Directory...> ... </Directory></code>
<code>Listen</code>	<code>LoadModule</code>
<code>DocumentRoot</code>	<code><IfModule...> ... </IfModule></code>
<code>DirectoryIndex</code>	<code>ServerName</code>
<code>Options</code>	<code>Indexes</code>

Podemos observar el valor de alguna de las directivas como:

```
servicios@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.4      Fichero: /etc/apache2/apache2.conf      Modificado

#
# NOTE!  If you intend to place this on an NFS (or otherwise network)
# mounted filesystem then please read the LockFile ocumentation (available
# at <URL:http://httpd.apache.org/docs/2.2/mod/mpm_common.html#lockfile>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory path.
#
#ServerRoot "/etc/apache2"
```

Como indicamos antes, también podemos buscar información respecto a la configuración del almacenamiento de avisos de errores, ya que la directiva que lo gestiona es `ErrorLog` y el fichero que los contiene es `/var/log/apache2/error.log`.

```
servicios@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.4      Fichero: /etc/apache2/apache2.conf      Modificado

# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off

# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog ${APACHE_LOG_DIR}/error.log
```

Para conocer la ruta de almacenamiento indicada por la variable de entorno deberemos consultar el fichero `/etc/apache2/envvars`:

```
GNU nano 2.2.4      Fichero: /etc/apache2/envvars

# Only /var/log/apache2 is handled by /etc/logrotate.d/apache2.
export APACHE_LOG_DIR=/var/log/apache2$SUFFIX
```

Si accedemos a su contenido, veremos algo similar a lo siguiente:

```
servicios@debian: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@debian:~#  
root@debian:~# cat /var/log/apache2/error.log | more  
[Sat Dec 15 17:46:34 2012] [notice] Apache/2.2.16 (Debian) configured -- resumin  
g normal operations  
[Sat Dec 15 17:57:32 2012] [error] [client 192.168.56.1] File does not exist: /v  
ar/www/favicon.ico  
[Sat Dec 15 17:57:32 2012] [error] [client 192.168.56.1] File does not exist: /v  
ar/www/favicon.ico  
[Sat Dec 15 18:36:16 2012] [error] [client 172.26.25.101] File does not exist: /  
var/www/favicon.ico  
[Sat Dec 15 18:36:19 2012] [error] [client 172.26.25.101] File does not exist: /  
var/www/favicon.ico  
[Sat Dec 15 18:37:13 2012] [error] [client 172.26.25.101] File does not exist: /  
var/www/favicon.ico  
[Sat Dec 15 18:46:16 2012] [notice] caught SIGTERM, shutting down  
[Sat Dec 15 18:55:08 2012] [notice] Apache/2.2.16 (Debian) configured -- resumin  
g normal operations  
[Sat Dec 15 18:59:58 2012] [notice] caught SIGTERM, shutting down  
[Sun Dec 16 10:43:26 2012] [notice] Apache/2.2.16 (Debian) configured -- resumin
```

Para conocer el nivel de error (LogLevel warn) a controlar deberemos acceder a la configuración del sitio de defecto (/etc/apache2/sites-enabled/000-default), donde podemos apreciar también el fichero de almacenamiento de las visitas o accesos (/var/log/apache2/access.log).

```
servicios@debian: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.2.4 Fichero: /etc/apache2/sites-enabled/000-default  
  
        Allow from all  
    </Directory>  
  
    ErrorLog ${APACHE_LOG_DIR}/error.log  
  
    # Possible values include: debug, info, notice, warn, error, crit,  
    # alert, emerg.  
    LogLevel warn  
  
    CustomLog ${APACHE_LOG_DIR}/access.log combined
```

El contenido de los accesos es:

```
servicios@debian: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.2.4      Fichero: /var/log/apache2/access.log  
192.168.56.1 - - [15/Dec/2012:17:57:32 +0100] "GET / HTTP/1.1" 200 484 "-" "Moz$  
192.168.56.1 - - [15/Dec/2012:17:57:32 +0100] "GET /favicon.ico HTTP/1.1" 404 5$  
192.168.56.1 - - [15/Dec/2012:17:57:32 +0100] "GET /favicon.ico HTTP/1.1" 404 5$  
172.26.25.101 - - [15/Dec/2012:18:36:16 +0100] "GET / HTTP/1.1" 200 484 "-" "Mo$  
172.26.25.101 - - [15/Dec/2012:18:36:16 +0100] "GET /favicon.ico HTTP/1.1" 404 $  
172.26.25.101 - - [15/Dec/2012:18:36:19 +0100] "GET /favicon.ico HTTP/1.1" 404 $  
172.26.25.101 - - [15/Dec/2012:18:37:13 +0100] "GET /favicon.ico HTTP/1.1" 404 $
```

5. Etiquetas de directorio

Podemos especificar opciones de funcionamiento a aplicar sobre los directorios en los que Apache permite el acceso a los clientes web, así como el acceso a directorios y otros comandos o directivas.

5.1. Opciones de directorio

Para configurar las opciones de directorio utilizaremos las etiquetas:

```
<Directory path_directorio> y </Directory>
```

con lo que cualquier directiva incluida entre ellas se aplicará sobre el directorio especificado y sus subdirectorios.

Si quisiéramos aplicar distintas opciones a un directorio hijo que a su directorio padre, se deberá incluir una nueva etiqueta para dicho subdirectorio con nuevas opciones que sobrescriban a las del padre.

El comando `Options` permite especificar opciones sobre el funcionamiento del servidor web en el directorio. De entre ellas veremos las opciones por defecto para `/var/www/`, es decir: `Indexes`, `FollowSymLinks` y `MultiViews`, aunque existen otras más.

```
GNU nano 2.2.4  Fichero: /etc/apache2/sites-enabled/000-default

<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>

    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>
```

5.1.1. Indexes

En caso de no encontrar la página configurada por defecto (index.html), el contenido de la carpeta se mostrará como un listado de los distintos ficheros contenidos.



5.1.2. FollowSymLinks

Permite hacer un seguimiento de los enlaces simbólicos. Teniendo en cuenta esta opción podemos enlazar el servicio web Apache con cualquier carpeta. Por ejemplo, dado un usuario del sistema (*servicios*) crear una carpeta (con el contenido deseado) en su home que sea accesible desde el servidor Apache mediante su nombre *FQDN/nombre_de_carpeta* utilizando un enlace simbólico.

- 1) Lo primero que debemos hacer es crear la carpeta en el usuario del sistema y poner allí las páginas web a mostrar.

```
servicios@debian:~$ ls -l
total 4
drwxr-xr-x 2 servicios servicios 4096 dic 21 19:23 prueba
servicios@debian:~$ cd prueba
servicios@debian:~/prueba$ ls -l
total 4
-rw-r--r-- 1 servicios servicios 352 dic 21 19:24 index.html
servicios@debian:~/prueba$
```

Nota 1. La carpeta creada en el usuario y sus ficheros deben tener permisos de lectura para otros.

Nota 2. Crear directorio → `mkdir nom_dir`

Borrar directorio no vacío → `rm -r nom_direc`

- 2) Ahora debemos acceder a la carpeta `/var/www` y allí crear el enlace simbólico:

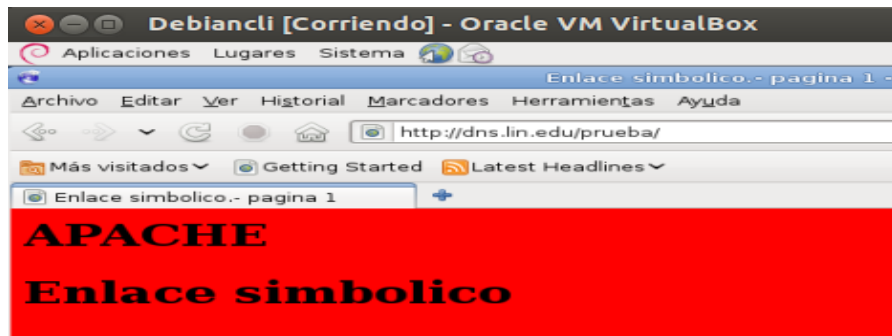
```
root@debian:/var/www# ls -l
total 8
-rw-r--r-- 1 root root 349 dic 21 18:39 inicio.html
-rw-r--r-- 1 root root 461 dic 21 18:40 pagina2.html
lrwxrwxrwx 1 root root 22 dic 21 19:08 prueba -> /home/servicios/prueba
root@debian:/var/www#
```

Nota 3. Crear enlace simbólico → `ln -s ruta_carpeta_enlazar nom_enlace_simbólico`

Ejemplo: `ln -s /home/servicios/prueba prueba`

Nota 4. : Borrar enlace simbólico → `unlink nombre_enlace_simbolico`

- 3) Ahora utilizaremos un cliente web para acceder al servidor web mediante su *FQDN/nombre_enlace_simbolico*, mostrándose así el contenido de la carpeta del home del usuario.



Importante. La opción `FollowSymLinks` puede presentar problemas de seguridad, ya que si tenemos la opción activa y un cracker consigue acceder hasta la carpeta `/var/www` puede crear enlaces simbólicos a la carpeta deseada del sistema y tener acceso a su contenido.

5.1.3. MultiViews

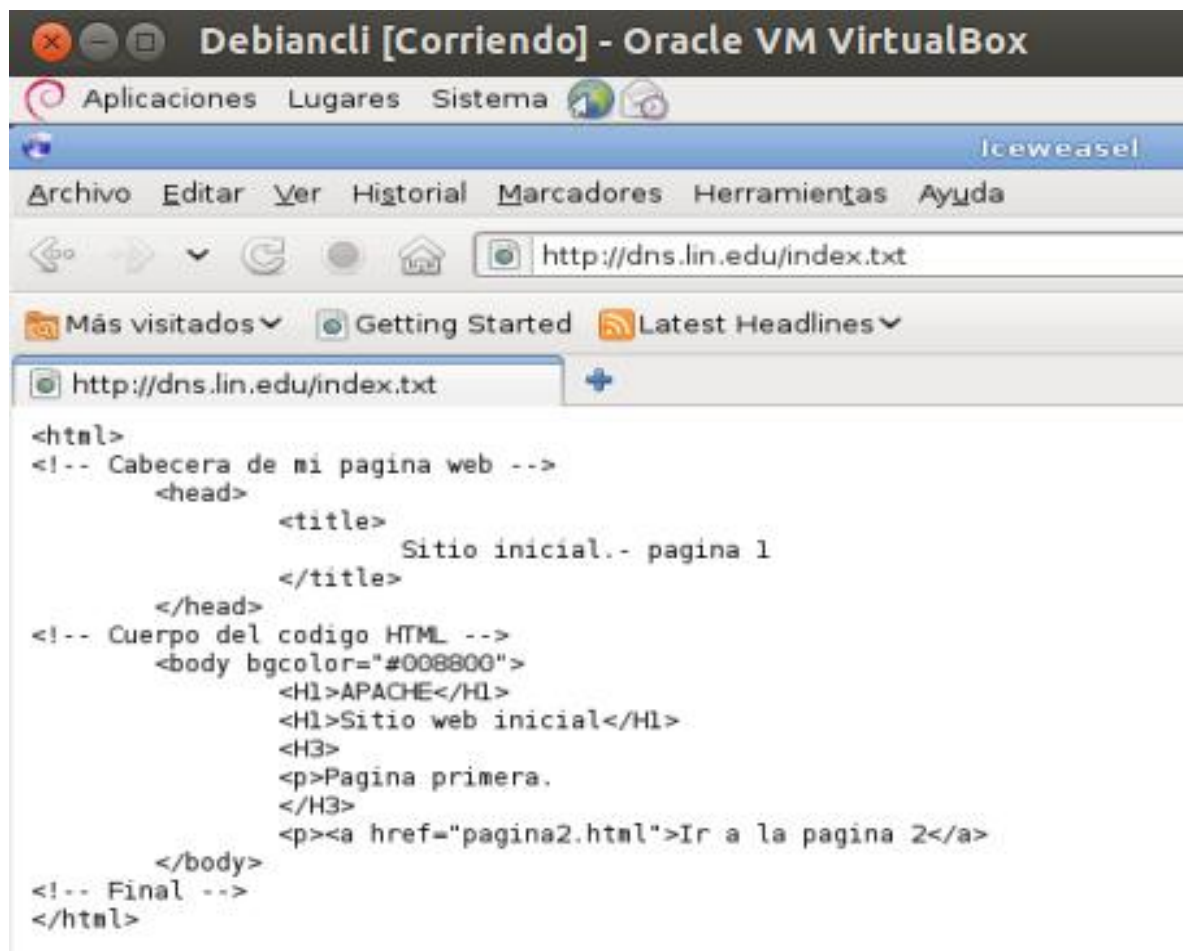
Está relacionada con el nombre del fichero solicitado por el cliente y el navegador donde se ejecuta. En una búsqueda con `Multiviews` activo, el servidor realiza una coincidencia implícita de patrón de nombre de archivo y elige entre los resultados según la configuración por defecto de variables internas de Apache. Por ejemplo, será capaz de mostrar el contenido del fichero sin necesidad de especificar su extensión u otras combinaciones.

```
root@debian:/var/www# ls -l
total 8
-rw-r--r-- 1 root root 349 dic 21 18:39 index.txt
-rw-r--r-- 1 root root 460 dic 21 19:47 pagina2.html
lrwxrwxrwx 1 root root 22 dic 21 19:08 prueba -> /home/servicios/prueba
root@debian:/var/www#
```


Al no especificar fichero buscará el de valor por defecto (*index.html*) y en caso de no encontrarlo buscará otro de igual nombre y distinta extensión y lo mostrará como si de código html se tratara.



Si solicitamos expresamente el fichero *index.txt*, al localizarlo será mostrado con el formato adecuado al tipo de fichero indicado, en este caso de texto plano.



5.1.4. AllowOverride

Existe otra opción muy importante que no aparece controlada por el comando `Options`, indicándose como otro comando a incluir. Se trata de **AllowOverride**.

`AllowOverride None` tiene la función de evitar que sea posible configurar el servidor mediante otro tipo de ficheros (medida de seguridad), no permitiendo sobrescribir las directivas del directorio. (Trabajaremos con esta opción cuando avancemos más en el tema).

5.2. Acceso directorios

Igual que indicamos las opciones de funcionamiento a aplicar sobre los directorios, también podemos configurar los permisos de acceso de los clientes web.

Para ello, igual que en el caso anterior, utilizaremos las etiquetas:

```
<Directory path_directorio> y </Directory>
```

incluyendo dentro de ellas los permisos deseados.

Se debe tener en cuenta que cuando no aparece directamente en un directorio su configuración de acceso para los clientes, dicho directorio tomará lo indicado en su directorio padre.

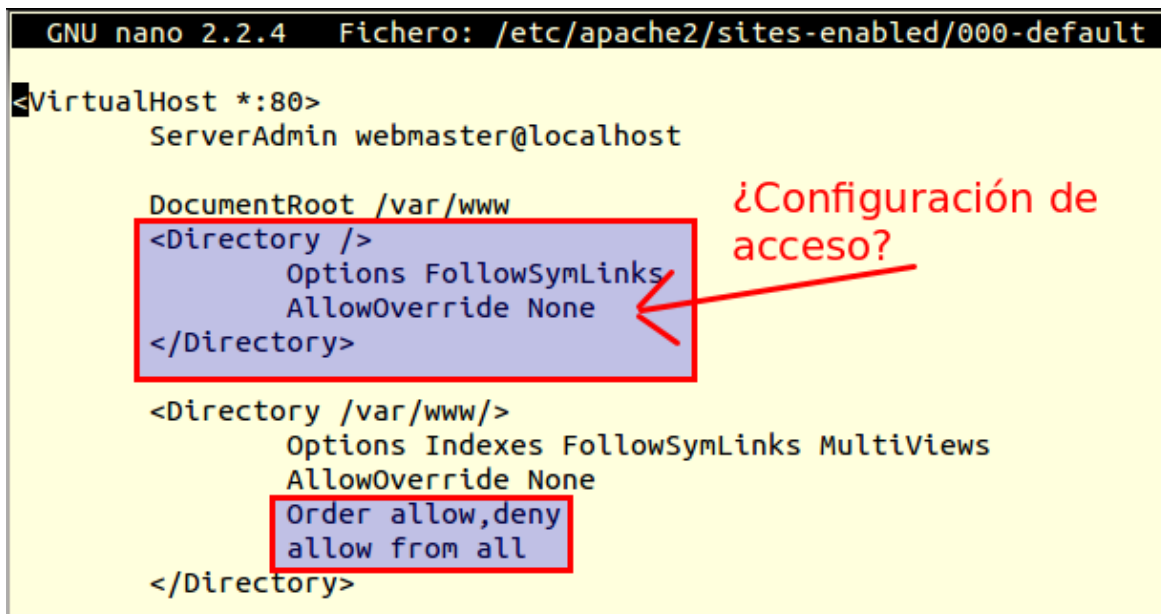
Si en el directorio padre de todos los directorios, es decir, el directorio raíz (/) no aparece configuración relativa a los acceso es porque se deniega el acceso a todos los clientes a dicho directorio.

La configuración del acceso consiste en especificar el orden en que se deben evaluar los comandos de permiso/denegación de acceso y posteriormente indicar los comandos relativos a dichos permisos de acceso/denegación.

```
GNU nano 2.2.4  Fichero: /etc/apache2/sites-enabled/000-default
<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>

    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>
```



Como podemos apreciar en la imagen anterior, para el directorio de trabajo por defecto `/var/www/`, el orden de evaluación de accesos consiste en evaluar primero la configuración de los clientes permitidos y después evaluar la configuración de los clientes denegados (`Order allow,deny`), teniendo en cuenta que a continuación se especifica acceso para todos (`allow from all`), es decir, se permitirá el acceso a todos los clientes.

¡Importante!. Es muy importante tener en cuenta que al escribir `allow,deny` o `deny,allow` no se debe especificar un espacio en blanco delante o detrás de la coma, ya que no funcionaría de forma correcta.

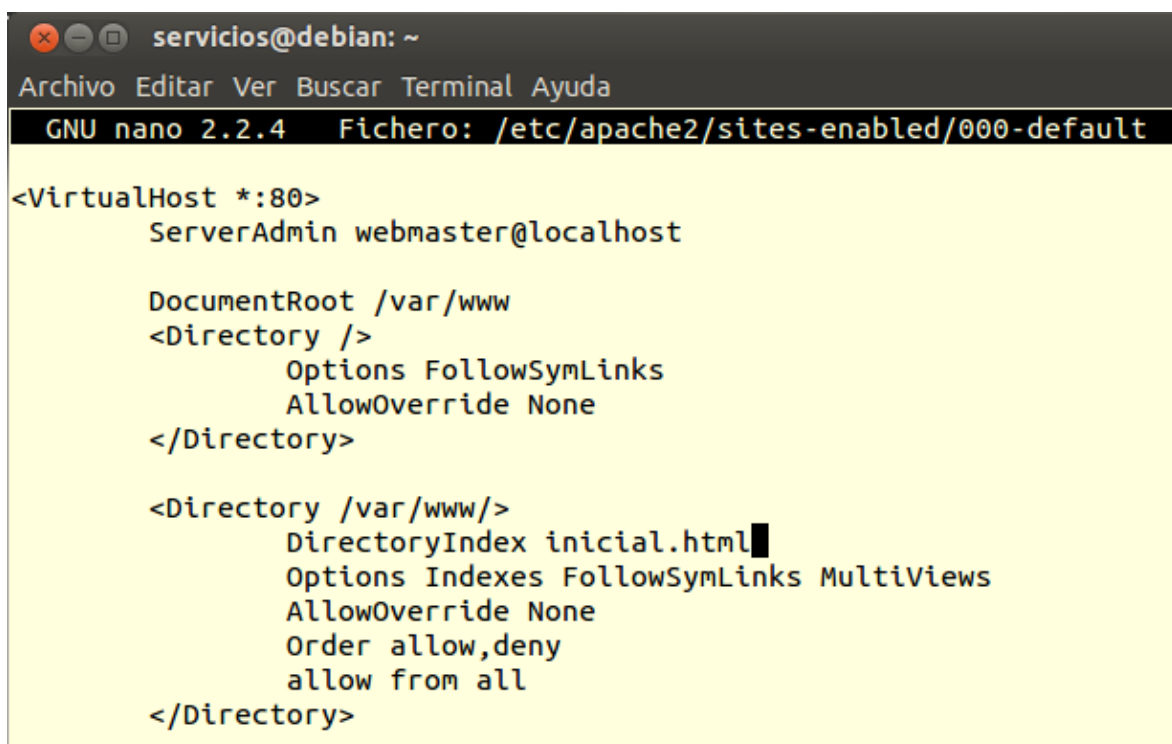
También se puede observar que en la imagen anterior la falta de configuración de accesos para los clientes en el directorio raíz, por lo que, por defecto, se le denegará el acceso a cualquier cliente.

5.3. Otros comandos o directivas

Dentro de la etiqueta `<Directory> ... </Directory>` podemos especificar otros valores de directivas o comandos para configurar opciones importantes para el directorio, como por ejemplo `DirectoryIndex`.

5.3.1. DirectoryIndex

Permite especificar una página a cargar por defecto distinta a la configurada a nivel general para Apache.



```
servicios@debian: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
GNU nano 2.2.4  Fichero: /etc/apache2/sites-enabled/000-default  
  
<VirtualHost *:80>  
    ServerAdmin webmaster@localhost  
  
    DocumentRoot /var/www  
    <Directory />  
        Options FollowSymLinks  
        AllowOverride None  
    </Directory>  
  
    <Directory /var/www/>  
        DirectoryIndex inicial.html  
        Options Indexes FollowSymLinks MultiViews  
        AllowOverride None  
        Order allow,deny  
        allow from all  
    </Directory>
```

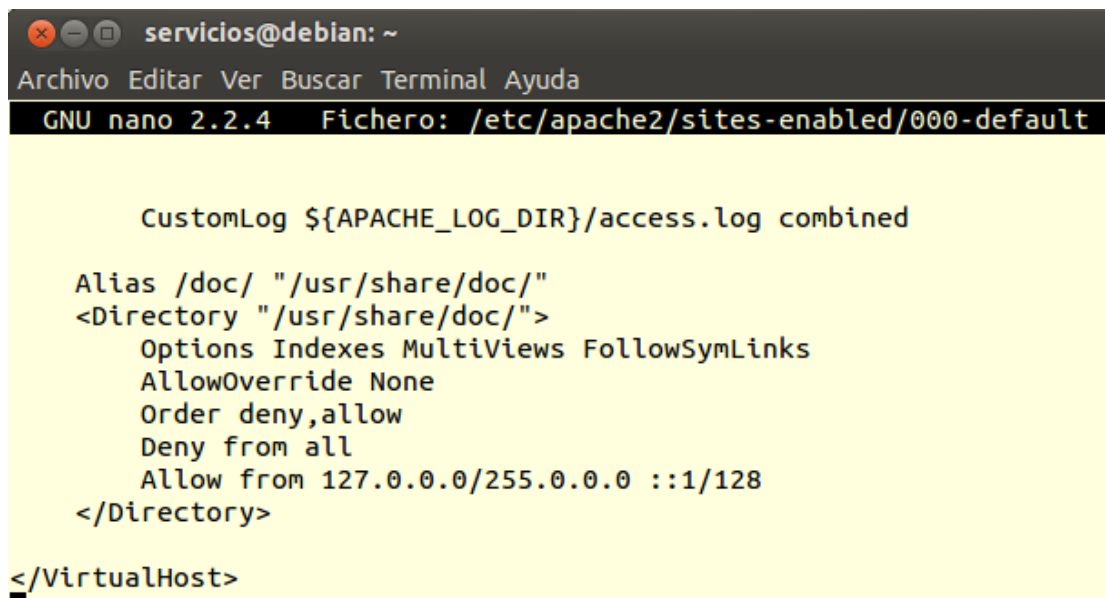


6. Otras configuraciones básicas de interés

Existen otras directivas muy interesantes que nos permiten configurar o personalizar un sitio web, entre ellas, destacan las siguientes.

6.1. Alias

Permite crear un alias de un directorio del sistema, es decir, podemos referenciar un directorio del sistema operativo utilizando otro nombre distinto al que tiene en el sistema. Como ejemplo, podemos observar la captura de pantalla siguiente, en la que podemos apreciar dentro de la configuración del sitio web por defecto la existencia de un alias apuntando a una carpeta (alias `/doc/` que apunta a la carpeta `/usr/share/doc/`), pero se deniega cualquier acceso desde un cliente externo al propio servidor.

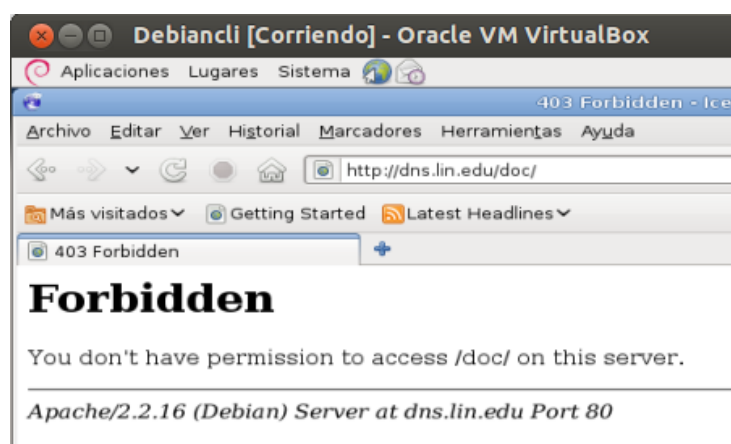


```
servicios@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.4 Fichero: /etc/apache2/sites-enabled/000-default

    CustomLog ${APACHE_LOG_DIR}/access.log combined

Alias /doc/ "/usr/share/doc/"
<Directory "/usr/share/doc/">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
    Order deny,allow
    Deny from all
    Allow from 127.0.0.0/255.0.0.0 ::1/128
</Directory>

</VirtualHost>
```



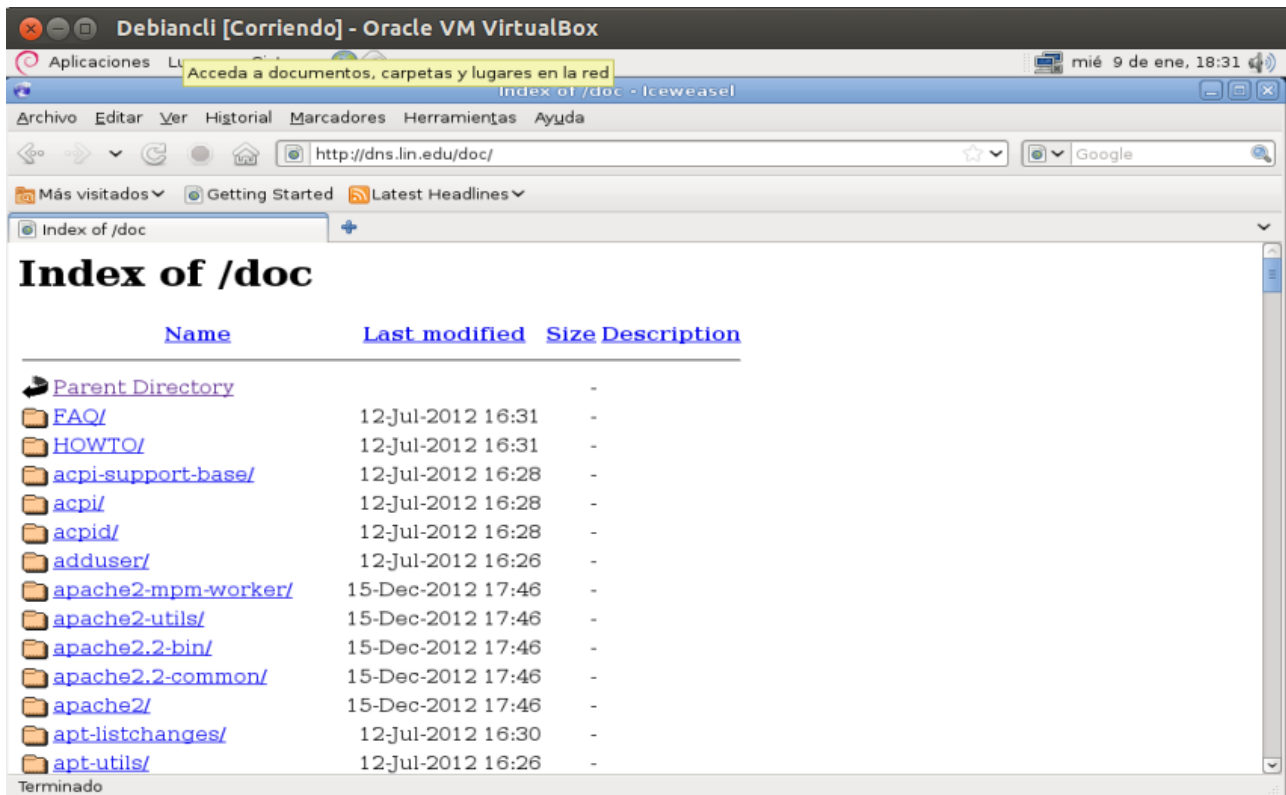
Para conseguir acceder al servidor desde cualquier cliente deberíamos modificar los parámetros relativos al acceso:

```
servicios@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.2.4 Fichero: /etc/apache2/sites-enabled/000-default

CustomLog ${APACHE_LOG_DIR}/access.log combined

Alias /doc/ "/usr/share/doc/"
<Directory "/usr/share/doc/">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
#    Order deny,allow
#    Deny from all
#    Allow from 127.0.0.0/255.0.0.0 ::1/128
    Order allow,deny
    Allow from all
</Directory>

</VirtualHost>
```



6.2. Redirect

Permite crear redirecciones, es decir, enviar al cliente que solicita una dirección a nuestro servidor web a otra dirección, del mismo o de otro servidor web.

6.3. ErrorDocument

Permite personalizar los mensajes de error, especificando un texto o página web a mostrar en caso de producirse un código de error concreto de Apache.

7. VirtualHost

Apache basa su funcionamiento en los VirtualHost, es decir, en la creación y configuración de sitios virtuales que contendrán las distintas páginas web. A continuación, vamos a ver la forma de hacer esto.

➤ Crear un directorio por cada VirtualHost donde se encontrará el código de las aplicaciones Web:

```
/var/www/html/dominio2.com
```

➤ Crear el VirtualHost. Para ello creamos un fichero de configuración (copia del virtualhost por defecto) para cada VirtualHost:

```
cp /etc/apache2/sites-available/default /etc/apache2/sites-available/dominio2.com.conf
```

➤ Configurar el contenido del fichero del VirtualHost (*/etc/apache2/sites-available/dominio2.com.conf*):

```
<VirtualHost *:80>
    ServerAdmin correo2@dominio2.com
    DocumentRoot /var/www/html/dominio2.com
    ServerName www.dominio2.com
    ServerAlias dominio2.com
</VirtualHost>
```

➤ Habilitar el nuevo VirtualHost:

```
a2ensite dominio2.com.conf
```

➤ Reiniciar el servicio.

```
service apache2 restart
```

➤ En caso de no tener servidor DNS, podemos modificar el fichero de */etc/hosts* con la resolución *IP-nombredominio*.