

ADMINISTRACIÓN DE SERVIDORES WEB, APACHE2

Instalación de apache2 en debían ó Ubuntu.

Desde el terminal sudo apt-get update

Desde el terminal sudo apt-get install apache2.

Debian, Ubuntu : Start /Restart/ Stop Apache web server.

Arrancar apache en la última versión estable de debían la 9.

```
#systemctl start apache2
```

Parar apache:

```
#systemctl stop apache2
```

Comprobar estado de apache2

```
#systemctl status apache2
```

Reiniciar apache2

```
#systemctl restart apache2
```

Otra opción es utilizar apache2ctl start|stop|restart|graceful

Para ver la versión, desde el terminal: apache2 -v

Información sobre formas de poner en marcha apache o pararlo en el siguiente enlace.

<https://www.cyberciti.biz/faq/ubuntu-linux-start-restart-stop-apache-web-server/>

CONFIGURACIÓN DE APACHE 2.4 EN DEBIAN Ó UBUNTU

La 2.4 que tiene algunos cambios con respecto la anterior

<http://www.justgohome.co.uk/blog/2014/04/new-in-14-04-apache.html>)

Los archivos de configuración de apache2 se encuentran en la carpeta `/etc/apache2`

```
|— apache2.conf
|— conf-available
|   |— charset.conf
|   |— localized-error-pages.conf
|   |— other-vhosts-access-log.conf
|   |— security.conf
|   |— serve-cgi-bin.conf
|— conf-enabled
|   |— charset.conf -> ../conf-available/charset.conf
|   |— localized-error-pages.conf -> ../conf-available/localized-error-pages.conf
|   |— other-vhosts-access-log.conf -> ../conf-available/other-vhosts-access-log.conf
|   |— security.conf -> ../conf-available/security.conf
|   |— serve-cgi-bin.conf -> ../conf-available/serve-cgi-bin.conf
|— envvars
|— magic
|— mods-available
|— mods-enabled
|— ports.conf
|— sites-available
|   |— 000-default.conf
|   |— default-ssl.conf
|— sites-enabled
|   |— 000-default.conf -> ../sites-available/000-default.conf
```

apache2.conf	En este archivo se encuentra la configuración del servidor web apache.
envvars	Contiene la información del usuario, grupo Y PID del servicio de apache.
mods-available	Directorio en el que se depositan los módulos que se agregan al servidor apache, por defecto y que están disponibles para instalar.
mods-enabled	Directorio donde se encuentran los módulos activados y disponibles
ports.conf	Archivo de configuración en donde se especifica el puerto de escucha, por default es 80 (http) y 443 (https).
sites-available	Directorio en donde se encuentran los archivos de configuración de sitios o páginas web que configuremos.
sites-enabled	Directorio en donde se habilita los sitios web.
conf-available (anterior conf.d)	Directorio en donde están los archivos de configuración de apache.
magic	Configuración del módulo mod_mime_magic. Se encarga de transferir ficheros de determinados tipos.

Configuración global del servidor

El archivo de configuración (etc/apache2/apache2.conf) está compuesto por directivas que le indican al servidor cómo actuar. Cada vez que haces un cambio en el archivo de configuración es necesario reiniciar el servidor para que surta efecto.

CONTEXTO

<http://httpd.apache.org/docs/2.4/mod/directive-dict.html#Context>

La configuración de Apache se basa en una serie de directivas que tienen posibilidad de ser usadas dentro de un contexto, es decir, un ámbito en el que pueden ser aplicadas. Hay cuatro posibilidades que no son excluyentes:

- Configuración global del servidor (server config):
 - Reúne los aspectos globales del servidor. Por ejemplo el número máximo de clientes concurrentes, los timeouts, el directorio raíz del servidor, etc.
 - Agrupa las directivas que definen la forma de responder a todos los pedidos del servidor principal, o sea aquellos que no son para los hosts virtuales, de existir alguno definido. También reúne los aspectos por defecto de todos los hosts virtuales que se configuren más adelante.
- Secciones para configurar los host virtuales (virtual host):
Agrupa las directivas relacionadas con los hosts virtuales que se definan
- Secciones de configuración de directorios.
En este contexto se incluyen directivas dentro de los contenedores `<Directory>` , `<Location>` , `<Files>` , `<If>` , y `<Proxy>`
- Archivos .htaccess

Para cada directiva en la doc de apache se puede encontrar el contexto en el cual la directiva se puede utilizar.

DirectoryIndex Directive	
Description:	List of resources to look for when the client requests a directory
Syntax:	DirectoryIndex disabled local-url [local-url] ...
Default:	DirectoryIndex index.html
Context:	server config, virtual host, directory, .htaccess
Override:	Indexes
Status:	Base
Module:	mod_dir

CONFIGURACIÓN /etc/apache/apache2.conf

→ Guía rápida de referencia de directivas

<http://httpd.apache.org/docs/2.4/mod/directives.html>

Se describen algunas directivas del server.

ServerRoot: Esta directiva define el directorio donde se ubica toda la información de configuración y registro que necesita el servidor para su correcto funcionamiento

PidFile; La directiva PidFile especifica la ruta del archivo PID (proceso ID)

Directivas relacionadas con el tiempo de espera de las conexiones con los clientes.

Timeout define, en segundos, el tiempo que el servidor esperará para recibir y enviar peticiones durante la comunicación, tras los cuales el servidor cierra la conexión. Está configurado por defecto a 300, lo cual es apropiado para la mayoría de las situaciones.

KeepAlive: Esta directiva se utiliza para indicar si se activarán las conexiones persistentes; es decir. el poder hacer más de una petición por conexión.

Por ejemplo, supongamos que hacemos una petición al servidor de una página web que contiene tres imágenes, entonces tendremos que hacer 4 peticiones, una para la página y una por cada imagen. El tener activadas las conexiones persistentes nos permite el poder hacer todas las peticiones a través de la misma conexión, y no tener que negociar nuevas conexiones. La respuesta del servidor será más rápida y tendremos un mejor rendimiento.

MaxKeepAliveRequests: Esta directiva establece el máximo número de peticiones que se pueden realizar en una conexión persistente. Las conexiones persistentes tienen que estar activadas, obviamente. Hay que tener en cuenta el ancho de banda de salida de nuestro servidor, por el cual deberá ser enviada toda la información. Si se establece un valor muy grande respecto al ancho de banda, el tiempo de respuesta se verá incrementado para cada usuario. El valor predeterminado de la directiva MaxKeepAliveRequests es de 100, que debería bastar en la mayoría de los casos.

KeepAliveTimeout: La directiva KeepAliveTimeout establece el número de segundos que el servidor esperará a la siguiente petición, tras haber dado servicio a una, antes de cerrar la conexión. Una vez recibida la petición, aplica la directiva Timeout en su lugar.

MaxClients: Permite especificar la cantidad máxima de clientes conectados simultáneamente al servidor. Por defecto es 150. Ejemplo:

MaxRequestsPerChild: Indica la cantidad de pedidos que puede atender un proceso servidor por hijo antes de que muera. Si se especifica cero el número será ilimitado. Poner límites a este número permite liberar la memoria asociada al proceso, en caso de que ésta se acumule accidentalmente, además de disminuir el número de procesos activos, cuando la carga del servidor disminuye.

Por defecto es 100. Este valor no incluye la cantidad de los pedidos en una conexión persistente, los cuales se cuentan como uno sólo. Sintaxis: MaxRequestsPerChild cantidad.
Ejemplo

Usuario y grupo

```
User ${APACHE_RUN_USER}  
Group ${APACHE_RUN_GROUP}
```

Directivas de log:

HostnameLookups: Se utiliza en los ficheros de registro. Por defecto, cuando se produce un acceso se guarda simplemente su número IP. Si esta directiva se encuentra en On, el servidor buscará la correspondencia de ese número IP con su nombre y lo almacenará. Establecer esta configuración en ON provocará que por lo menos se tenga que hacer una petición al servidor de nombres por cada una de las peticiones de usuario, por lo que el rendimiento de la máquina se puede ver decrementado. Esta directiva se puede encontrar dentro de una sección o fuera de cualquier otra.

ErrorLog: Especifica la ubicación del fichero que contiene el registro de errores. Por defecto en la carpeta logs. Esta directiva sólo se puede encontrar fuera de cualquier sección.

LogLevel: Especifica el tipo de mensajes que se guardarán en el fichero de registro de errores. Dependiendo de los valores especificados, se guardarán mas o menos. Esta directiva sólo se puede encontrar fuera de cualquier sección.

Valor de más a menos son: debug, info, notice, warn, error, crit, alert, emerg.

LogFormat: La directiva permite definir el formato que se utilizará para almacenar los registros. A cada formato se le puede asignar un nombre, utilizándolo luego para crear distintos tipos de ficheros de registro. Pueden existir varios LogFormat distintos. Sintaxis: LogFormat configuraciónError nombre

CustomLog: La directiva se utiliza para especificar la ubicación y el tipo de formato que se utilizará en un fichero de registro. Pueden existir varios ficheros de registro distintos con configuraciones distintas. Para hacer esto, simplemente hay que poner varias líneas customlog. Sintaxis: customLog fichero formato

http://httpd.apache.org/docs/2.4/mod/mod_log_config.html

DirectoryIndex: Especifica el fichero por defecto que buscará en cada directorio en caso de que no se especifique ninguno. Por defecto es index.html. Es decir, que si por ejemplo se pone en el navegador

El orden con el que se especifica el nombre de fichero determinará la prioridad a la hora de decidir que fichero es el que se muestra. Ejemplo:

```
DirectoryIndex index.html index.htm index.php
```

Directory: Esta opción se usa para configurar cómo se comportará y qué se permitirá en cada directorio al que tiene acceso el servidor Apache. Esta configuración se aplica a un directorio y los subdirectorios que contiene si no se sobrescribe en otra definición sobre un directorio más concreto.

Ejemplo:

Nos encontramos TRES veces esta etiqueta. La primera hace referencia al directorio raíz y se configura siempre con opciones muy restrictivas. Los subdirectorios heredan la configuración de los directorios padres y sobrescriben su configuración.

```
<Directory />
  Options FollowSymLinks
  AllowOverride None
  Require all denied
</Directory>

<Directory /usr/share>
  AllowOverride None
  Require all granted
</Directory>

<Directory /var/www/>
  Options Indexes FollowSymLinks
  AllowOverride None
  Require all granted
</Directory>
```

Opciones sobre directorios:

<Directory /directorio> se usa para configurar ciertos directorios del servidor
Por defecto, para el directorio raíz `<Directory />` (*fíjate que la barra invertida señala el directorio raíz*), no hay características permitidas salvo las siguientes:

Options La directiva *Options* controla características del servidor indicando con el atributo

FollowSymLinks: *FollowSymLinks* que el servidor puede seguir enlaces simbólicos en el directorio raíz.

Indexes: permite al servidor generar un listado del directorio.

AllowOverride: Establece qué directivas *Options* puede obviar un archivo `.htaccess`.
Por defecto, el directorio raíz está configurado para no permitir la prevalencia del fichero `.htaccess` (*None*). Es decir el fichero `htaccess` podría sobrescribir valores establecidos por defecto en el servidor si está activado, con el valor `All`. Con `None` no se puede sobrescribir ningún valor.

Cualquier directorio del sistema que necesite valores más permisivos ha de ser configurado explícitamente como ocurre con el `<Directory/var/www>`.

Por defecto, en la en el directorio `/var/www`, *Options* está configurado para incluir los comandos *Indexes*, *FollowSymLinks*, *Indexes* como ya se ha comentado anteriormente *FollowSymLinks* permite al servidor seguir enlaces simbólicos en ese directorio. Si utilizamos , por ejemplo, *Options -FollowSymLinks* y no utilizamos la directiva *Indexes*, no veremos nada, nos mostrará acceso prohibido “Forbidden” .

Ejemplo de uso:

<http://eltallerdelbit.com/followsymlinks-apache-options>

<http://httpd.apache.org/docs/2.4/es/mod/core.html#options>

Normally, if multiple *Options* could apply to a directory, then the most specific one is used and others are ignored; the options are not merged. (See [how sections are merged](#).) However if all the options on the *Options* directive are preceded by a + or - symbol, the options are merged. Any options preceded by a + are added to the options currently in force, and any options preceded by a - are removed from the options currently in force.

Warning

Mixing *Options* with a + or - with those without is not valid syntax, and is likely to cause unexpected results.

For example, without any + and - symbols:

```
<Directory /web/docs>
  Options Indexes FollowSymLinks
</Directory>

<Directory /web/docs/spec>
  Options Includes
</Directory>
```

then only *Includes* will be set for the `/web/docs/spec` directory. However if the second *Options* directive uses the + and - symbols:

```
<Directory /web/docs>
  Options Indexes FollowSymLinks
</Directory>

<Directory /web/docs/spec>
  Options +Includes -Indexes
</Directory>
```

then the options *FollowSymLinks* and *Includes* are set for the `/web/docs/spec` directory.

Order simplemente controla el orden en que *allow* y *deny* se evalúan. El servidor está configurado para evaluar *Allow* antes que *deny* para el directorio *DocumentRoot*.

Allow especifica qué cliente puede acceder un directorio dado. El cliente puede ser *all*, un nombre de dominio, una dirección IP, un par red/máscara de red, etc. El directorio *DocumentRoot* está configurado para permitir peticiones de *all* (cualquiera).

Deny funciona como *allow*, pero especifica a quién se niega el acceso. *DocumentRoot* no está configurado para rechazar peticiones de nadie.

Require all granted: Acceso permitido incondicionalmente. Ver:

http://httpd.apache.org/docs/2.4/mod/mod_authz_core.html#require

Require Sustituye a las anteriores directivas (*Order*, *Allow*, *Deny*) de apache 2.2

AccessFileName: Indica el nombre (“*.htaccess*”) que deberá tener el fichero local de un directorio en el que se almacenen directivas de acceso.

IncludeOptional: Incluye los ficheros de configuración de los directorios, en versiones anteriores a la 2.4 era **Include**.

Módulos

Recuerda que Apache es de los servidores Web más usados por su modularidad. Es decir se pueden agregar funcionalidades al servidor por medio de módulos. Apache por defecto incluirá todos los módulos existentes en el directorio “/etc/apache2/mods-enabled/”. Todos estos ficheros son funcionalidades extra del servidor.

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

Fíjate que los archivos que aparecen dentro de la carpeta “mods-enabled” son enlaces a la carpeta “mods-available”.

Puerto

Incluirá las directivas incluidas en el archivo “ports.conf”, que hacen referencia al puerto por el que se dará servicio (normalmente el puerto 80).

```
Include ports.conf
```

En ese fichero aparece la directiva Listen que especifica el puerto (y, opcionalmente, la dirección IP) por el que escuchará Apache2.

Si no se especifica la dirección IP, Apache2 escuchará por todas las direcciones IP asignadas a la máquina en la que se ejecute.

El valor predeterminado de la directiva Listen es 80. Cambiarlo a 127.0.0.1:80 provoca que Apache2 sólo escuche por su dispositivo loopback, de forma que no estará disponible para Internet.

Dominios virtuales

Incluye el directorio “/etc/apache2/sites-enabled”, que es el lugar donde se definirán los distintos dominios virtuales que deseemos ofrecer con nuestro servidor.

```
# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf
```

¿Qué dos sitios se han definido por defecto?

En la versión 2.4 de apache es necesario que los archivos acaben en .conf

Fíjate que los archivos que aparecen dentro de la carpeta “sites-enabled” son enlaces a la carpeta “sites-available”.

Ficheros de configuración

```
# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf
```

Otras directivas que se pueden configurar a nivel de servidor global(se recomienda configurarlas dentro del directorio conf-available en vez de en apache2.conf)

ServerName: Nombre del servidor. ServerName localhost

Apache lanza un warning si esta directiva no está configurada a nivel global del server

AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message

Para eliminar este warning tenemos dos opciones (mejor la 2):

1) Meter en apache2.conf la directiva

```
ServerName localhost
```

2) Utilizar el directorio conf-available creando el fichero "servername.conf" con dicha directiva y activar la configuración creando el link simbólico a conf-enabled con el comando

```
sudo a2enconf servername
```

ServerAdmin: Esta opción permite configurar la dirección del administrador del servidor web que se mostrará si el servidor genera una página de error.

Buscar directivas en un directorio

Si quieres buscar en qué ficheros está una directiva definida se puede usar grep -lir directiva directorio, por ejemplo:

```
grep -lir documentRoot "/etc/apache2/"
```

CONFIGURACIÓN DEL HOST VIRTUAL POR DEFECTO

Si se consulta el fichero /etc/apache2/sites-enabled/000-default.conf se observa que básicamente se traduce a:

```
<VirtualHost *:80>
  ServerName www.example.com
  ServerAdmin webmaster@localhost
  DocumentRoot /var/www/html
  #LogLevel info ssl:warn
  ErrorLog ${APACHE_LOG_DIR}/error.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

A continuación se detallan algunas directivas que se pueden configurar a nivel de host virtual, localízalas si existen en el host virtual por defecto.

VirtualHost *:80 Esta directiva define el host virtual, indicando la dirección IP donde va a responder el servidor (podríamos tener una tarjeta de red con varias direcciones IP) y el puerto (80 es el puerto por defecto donde escucha Apache). Si queremos que escuche en todas las direcciones IP del servidor colocamos un asterisco quedaría *:80.

ServerName es el nombre del host virtual.

ServerAlias nombre del servidor alternativo.

DocumentRoot path donde están guardados los archivos de nuestra página, puede ser cualquier carpeta donde tengamos los ficheros (cuidado con la seguridad y los permisos de la carpeta).

ServerAdmin especifica la dirección de correo del administrador del servidor. El valor por omisión es `webmaster@localhost`. Esta dirección se utiliza para que si tu sitio web tiene algún problema, Apache2 muestre un mensaje de error en el que aparecerá esta dirección de correo para que la gente pueda enviar un informe del error.

DirectoryIndex página servida por defecto por el servidor cuando un usuario solicita el índice de un directorio añadiendo la barra de división (/) al final del nombre del directorio. Por ejemplo, cuando un usuario solicita la página <http://www.ejemplo.com/directorio/>, el servidor intentará buscar uno de los archivos listados en la directiva `DirectoryIndex` y devolverá el primero que encuentre, si no encuentra ninguno pero tiene especificada la opción *Indexes* devolverá un listado en formato HTML, de los subdirectorios y archivos del directorio, o una página Permiso Denegado si no se cumplen ninguna de las condiciones anteriores. En el módulo `dir.conf` está configurado que la directiva por defecto sea `index.html`.

```
<IfModule mod_dir.c>
DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>
```

CustomLog fichero log donde el servidor escribe los registros de las transferencias.

ErrorLog archivo log en el que se registrarán los errores.

Estos dos ficheros se mantienen separados para ayudar en la resolución de problemas.

OTRAS DIRECTIVAS

Códigos de error (ErrorDocument)

Códigos HTTP Apache: http://members.tripod.com/r_castaneda0/

Se pueden configurar los mensajes de los errores que lanza Apache, para indicar qué deben mostrar los errores a nivel global del servidor en:

`/etc/apache2/conf-available/localized-error-pages.conf`,

O también a nivel de Virtual host utilizando la siguiente sintaxis:

```
ErrorDocument <3-digit-code> <action>
```

Donde `action` puede ser:

1. Texto a mostrar. Ponga antes del texto que quiere que se muestre unas comillas ("). Lo que sea que siga a las comillas se mostrará. Nota: las comillas (") no se muestran.
2. Una URL local a la que se redireccionará la petición.
3. Una URL externa a la que se redireccionará la petición.

```
ErrorDocument 404 "Lo siento esa pagina no existe"
```

Codificación (problema con las tildes, ñ, etc)

Se puede configurar el tipo de codificación a usar a nivel global para que muestre bien las tildes en `/etc/apache2/conf-available/charset.conf`

```
AddDefaultCharset UTF-8
```

Solución buena:

<http://principlatecnologica.wordpress.com/2013/01/08/problemas-html-acentos-y-enes-charset-utf-8-iso-8859-1/>

Multilinguaje

Configurar browser

Preferencias/contenido/idioma: Configurarlos según las preferencias

Crear páginas en diferentes idiomas

`index.html.en` – English `index.html.de` – German

`index.html.es` - Español

Configurar server

Por ejemplo si lo queremos configurar a nivel de virtualHost

```
<VirtualHost *:80>
    ServerName www.example.com
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    #LogLevel info ssl:warn
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # Lenguajes soportados (no obligatorio porque ya es definido en
    /etc/apache2/mods-available/mime.conf)
    AddLanguage es .es
    AddLanguage en .en
    AddLanguage de .de
    LanguagePriority es es de

    <Directory /var/www/html> direc
        Options Indexes FollowSymLinks Multiviews
        AllowOverride None
        Require all granted
    </Directory>
</VirtualHost>
```

Las directivas de lenguajes soportados ya vienen configuradas por defecto en diferentes ficheros de configuración por lo que no es obligatorio configurarlo en el virtual host si queremos usar la configuración por defecto.

`/etc/apache2/mods-available/negotiation.conf`

`/etc/apache2/mods-available/mime.conf`