

ÍNDICE DE CONTENIDO

Información de Derechos reservados de esta publicación.....	2
Tema 5. Servidor Web Apache.....	3
5.1 Como empezó todo.....	4
5.2 Proceso de instalación del servidor web Apache.....	6
5.2.1 Instalando el servidor web apache.....	6
5.2.2 Archivos de configuración del servidor web Apache.....	6
5.2.2.1 Configuración del fichero httpd.conf.....	7
5.2.2.1.1 Directiva ServerTokens.....	7
5.2.2.1.2 Directiva ServerRoot.....	7
5.2.2.1.3 Directiva Timeout.....	8
5.2.2.1.4 Directiva KeepAlive.....	8
5.2.2.1.5 Directiva MaxKeepAliveRequests.....	8
5.2.2.1.6 Directiva KeepAliveTimeout.....	8
5.2.2.1.7 Directiva Listen.....	8
5.2.2.1.8 Directiva Include.....	8
5.2.2.1.9 Directiva LoadModule.....	9
5.2.2.1.10 Directiva User	9
5.2.2.1.11 Directiva Group.....	9
5.2.2.1.12 Directiva ServerAdmin.....	9
5.2.2.1.13 Directiva ServerName.....	9
5.2.2.1.12 Directiva UseCanonicalName	10
5.2.2.1.13 Directiva DocumentRoot.....	10
5.2.2.1.14 Directiva Options.....	10
5.2.2.1.15 Directiva AllowOverride	11
5.2.2.1.16 Directiva Allow.....	11
5.2.2.1.17 Directiva Deny.....	12
5.2.2.1.17 Directiva Order.....	12
5.2.2.1.18 Directiva Alias.....	12
5.2.2.1.18 Directiva ErrorLog.....	13
5.2.2.1.19 Directiva ErrorLevel.....	13
5.2.2.1.19 Directiva Redirect.....	13
5.2.3 Iniciar , detener o reiniciar el servidor web Apache.....	13
5.3 Creación de dominios virtuales en Apache.....	14
5.3.1 Paso 1.- Activando la directiva NameVirtualHost.....	14
5.3.2 Paso 2.- Estructura de directorios para dominios virtuales	14
5.3.3 Paso 3.- Creación y modificación de los ficheros de configuración de los dominios virtuales	15

Información de Derechos reservados de esta publicación.

Reconocimiento-NoComercial-CompartirIgual 2.1

Usted es libre de:

- Copiar, Distribuir y Comunicar públicamente la obra

Bajo las condiciones siguientes:



Reconocimiento. Debe reconocer y citar al autor original.



No comercial. No puede utilizar esta obra para fines comerciales.



Compartir bajo la misma licencia. Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

- Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
- alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor

Los derechos derivados de usos legítimos u otras limitaciones no se ven afectados por lo anterior.

Reconocimiento-NoComercial-CompartirIgual 2.1

Autor del manual: Javier Eduardo Duarte

Tema 5. Servidor Web Apache

APACHE
HTTP SERVER



5.1 Como empezó todo

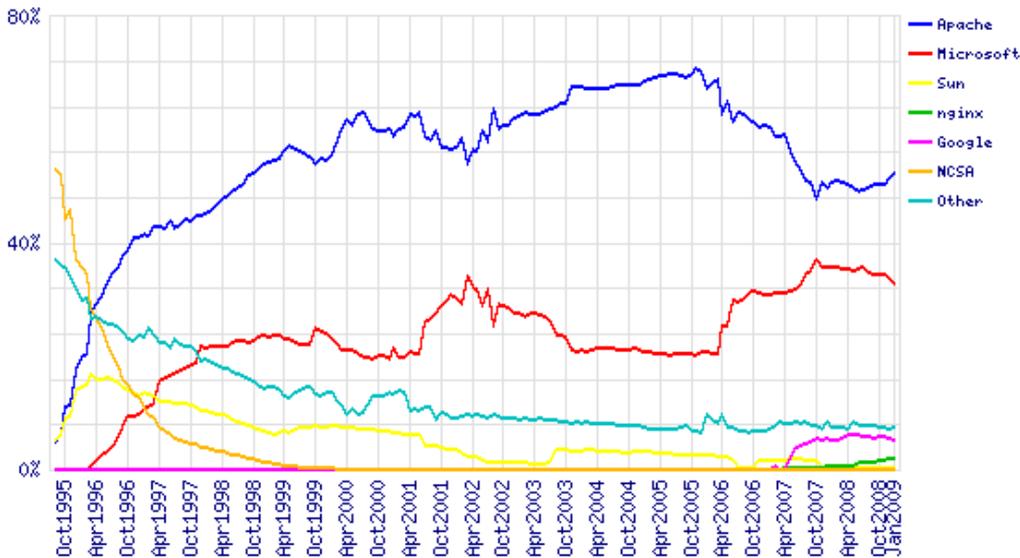
El nombre del servidor web apache proviene de la palabra en ingles patchy server que en español se puede entender como servidor parchado, ¿Tal vez te preguntaras, porque parchado?, la explicación es sencilla, el servidor web apache fue conformado por diversos parches del servidor web usado en ese momento, nos referimos al servidor web NCSA el cual era desarrollado en ese entonces por el National Center Supercomputing.

El desarrollo del servidor web apache se remonta al lejano año de 1995, dicho desarrollo dio como resultado una especie de versión beta de lo que llegaría a convertirse en la primera versión de apache ya que estaba compuesto en su totalidad por una colección de parches del servidor web NCSA. Fue hasta el año de 1996 cuando fue lanzada la primera versión estable de Apache la cual tenia entre sus principales características la reescritura por completo de su código base, también incluía la carga de módulos en tiempo de ejecución. Meses mas tarde fue lanzada la versión 1.1 la cual tenia como novedad la inclusión de un modulo de autentificación contra bases de datos. La versión 1.3 de apache vio la luz en el año de 1998 y esta incluía como principal característica soporte para plataformas Windows.

Actualmente el servidor web apache se encuentra en su versión 2 e incluye notables mejoras con respecto a versiones anteriores, algunas de ellas son:

- Modo Híbrido
- Nuevo sistema de configuración y compilación
- Soporte nativo para Ipv6
- Mensajes de error en diferentes idiomas
- Mejoras adicionales.

Como dato adicional, cabe menciona que apache es el servidor web numero uno a nivel mundial el cual abarca cerca de un 52.26 % del mercado total de Internet desbancando a servidores web como el IIS (Internet Information Server) de Microsoft.



Estos cifras pueden ser verificadas visitando el portal de Netcraft

<http://news.netcraft.com/>

Existe también una fundación dedicada a dar soporte legal y financiero al desarrollo de los proyectos relacionados con Apache , el nombre de esta fundación es Apache Software Foundation, la cual actualmente esta conformada por una comunidad de desarrolladores los cuales día a día contribuyen a la expansión y mejora de proyectos.

Entre los proyectos mas destacados de esta fundación podemos encontrar los siguientes:

- Http Server.- Servidor Web Apache
<http://www.apache.org/>
- Jakarta.-Proyectos en el lado del servidor basados en Java
<http://tomcat.apache.org/>
- mod_perl.- Modulo de apache para la programación dinámica en Perl
<http://perl.apache.org/>
- SpamAssin.-Sistema de detección de Spam
<http://spamassassin.apache.org/>

5.2 Proceso de instalación del servidor web Apache

5.2.1 Instalando el servidor web apache

La instalación del servidor web apache es relativamente sencilla , solo debe teclear en terminal el siguiente comando.

```
[root@ localhost ~]# yum install -y httpd
```

Recuerde que este comando se debe ejecutar como root

5.2.2 Archivos de configuración del servidor web Apache

La configuración del servidor web apache se realizara sobre dos ficheros distintos, uno de configuración general del servidor web apache y otro para indicarle al servidor apache los dominios virtuales que deben ser cargados al sistema.

El fichero de configuración principal de apache lo encontramos en la siguiente ruta

```
/etc/httpd/conf/
```

La carpeta donde deberán ser añadidos los ficheros de configuración de los dominios virtuales sera en la siguiente ruta:

```
/etc/httpd/conf.d/
```

5.2.2.1 Configuración del fichero httpd.conf

La ubicación de este fichero lo encontramos en:

```
/etc/httpd/conf/ -----> httpd.conf
```

El contenido del fichero "httpd.conf" esta compuesto por un gran numero de secciones es por ello que solo describiremos las mas relevantes del mismo, usted podrá habilitar o deshabilitar cada una de las funciones que describiremos según su necesidad.

5.2.2.1.1 Directiva ServerTokens

Esta directiva limita la cantidad de información que sera mostrada por nuestro servidor web apache como puede ser, la version del servidor web apache que tenemos instalado o los servicios que corren paralelamente con apache como php o MySQL.

Para delimitar la cantidad de información mostrada por el sistema existen cuatro opciones

ServerTokens ProductOnly	Solo mostrara el nombre del servidor web instalado. Ejemplo: Server Apache
ServerTokens Minimal	Muestra el nombre asi como la version de apache instalada. Ejemplo: Server Apache 2.1
ServerTokens OS	Mostrara el nombre, version y sistema operativo sobre el cual se encuentra montado: Ejemplo: Server Apache 1.3/(Linux)
ServerTokens Full	Mostrara nombre, version, sistema operativo asi como los servicios que hacen uso del servidor web. Ejemplo: Apache 1.3/(Linux)/PHP3/MySQL

5.2.2.1.2 Directiva ServerRoot

Esta directiva le indica al servidor web la ubicación donde se almacenan los ficheros de configuración de apache.

El valor por defecto es:

```
ServerRoot "/etc/httpd"
```

Si usted quisiera ubicar estos ficheros en otra ruta diferente solo deberá especificarla, aunque no es recomendable

5.2.2.1.3 Directiva Timeout

Esta directiva indica el número de segundos antes de que se cancele un conexión por falta de respuesta. Su valor por defecto es 120

```
Timeout 120
```

5.2.2.1.4 Directiva KeepAlive

Esta directiva indica si se permiten o no las conexiones persistentes, es decir más de una petición por conexión. Puede tomar los valores de "On" u "Off".

```
KeepAlive On|Off
```

5.2.2.1.5 Directiva MaxKeepAliveRequests

Esta directiva indica el máximo número de peticiones que se permiten en conexiones persistentes. Un valor 0 permite un número ilimitado. Se recomienda dejar esta valor elevado para obtener un mayor rendimiento. Por ejemplo 100

```
MaxKeepAliveRequests 100
```

5.2.2.1.6 Directiva KeepAliveTimeout

Esta directiva indica el número de segundos de espera para la siguiente petición del mismo cliente con la misma conexión. Por ejemplo 15

```
KeepAliveTimeout 15
```

5.2.2.1.7 Directiva Listen

Listen permite asociar Apache a una dirección y/o puerto específico además del predeterminado.

Ejemplo:

```
Listen 192.168.1.1:8080  
Listen 80
```

5.2.2.1.8 Directiva Include

```
Include conf.d/*.conf
```

Esta directiva indica al servidor web la ruta en donde se encuentran almacenados los ficheros de configuración adicionales de apache como por ejemplo los dominios virtuales.

Es habitual dejar el fichero de configuración con las características globales que no se tienen que modificar en el fichero principal e incluir los ficheros que pueden estar sujetos a modificación en el directorio

```
"/etc/httpd/conf.d"
```

De esta forma para añadir o quitar algún fichero de configuración de apache sólo tenemos que borrarlo del directorio `/etc/httpd/conf.d`

5.2.2.1.9 Directiva LoadModule

Esta directiva corresponde al soporte de Dynamic Shared Object (Objetos Dinámicos Compartidos). Son módulos que incorporan ciertas funcionalidades que se le incorporan al servidor Apache. Para que un módulo sea funcional tienen que estar construido como un DSO e incorporar la correspondiente directiva `'LoadModule'` antes de que se a utilizada. Los módulos compilados de forma estática no es necesario incluirlos.

Ejemplo:

```
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule auth_digest_module modules/mod_auth_digest.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authn_alias_module modules/mod_authn_alias.so
LoadModule authn_anon_module modules/mod_authn_anon.so
```

5.2.2.1.10 Directiva User

Esta directiva especifica qué usuario es el que ejecuta los procesos del servidor web y en consecuencia los permisos de lectura y escritura que se aplican sobre los recursos.

```
User apache
```

5.2.2.1.11 Directiva Group

Esta directiva especifica qué grupo es el que ejecuta los procesos del servidor web y en consecuencia los permisos de lectura y escritura que se aplican sobre los recursos.

```
Group apache
```

5.2.2.1.12 Directiva ServerAdmin

Esta directiva especifica la persona a la que se le debe notificar los problemas referentes al portal web , esto a través de su cuenta de correo.

Ejemplo:

```
ServerAdmin administrador@tuDominio.net
```

5.2.2.1.13 Directiva ServerName

Esta directiva especifica el nombre y puerto que el servidor utiliza para identificarse. Con una correcta configuración, este valor se puede determinar automáticamente, pero es recomendable especificarlo explícitamente para evitar problemas durante el arranque.

```
ServerName www.tuDominio.net:80
```

5.2.2.1.12 Directiva UseCanonicalName

UseCanonicalName determina como Apache construye las autoreferencias de URL y las variables SERVER_NAME y SERVER_PORT.

Cuando está directiva esta como "Off" apache usa los valores suministrados por el cliente. Cuando está como "On" , apache usa la directiva ServerName.

```
UseCanonicalName On|Off
```

5.2.2.1.13 Directiva DocumentRoot

Esta directiva indica al servidor web la ruta en donde se encuentran almacenados los ficheros web de tu sitio principal

```
DocumentRoot "/var/www/html"
```

NOTA:Esta directiva cambia cuando se implementan sitios virtuales

5.2.2.1.14 Directiva Options

La directiva Options indica varias posibles opciones de comportamiento y estas pueden ser aplicadas a un directorio concreto. Un claro ejemplo de aplicación de estas directiva se puede observar en el siguiente cuadro:

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

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

Las opciones que podemos observar se explican con mas detalle en el siguiente cuadro:

All	todas las opciones salvo MultiViews
ExecCGI	Se permite la ejecución de scripts CGI.
FollowSymLinks	el servidor seguirá los enlaces simbólicos. Tener esta opción activa aumenta el rendimiento ya que el servidor no comprueba si un fichero o directorio es un enlace simbólico y es más rápido, pero en algunos casos puede presentar problemas de inseguridad.
Includes	Se permiten incluir Server-side.
Indexes	Si una URL solicita un directorio y no existe DirectoryIndex (v.g., index.html) en ese directorio, el servidor devolverá una lista del contenido del directorio.
MultiViews	Se permite mostrar contenido negociado en función de diversos valores.
SymLinksIfOwnerMatch	Se sigue un enlace simbólico sólo si los propietarios del enlace y del destino coinciden.

5.2.2.1.15 Directiva AllowOverride

La directiva AllowOverride controla qué directivas se pueden situar en los ficheros .htaccess y estas pueden ser aplicadas igualmente a un directorio concreto. Un claro ejemplo de aplicación de esta directiva se puede observar en el siguiente cuadro:

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

Los valores de AllowOverride pueden ser "All", "None", o una combinación de:

AuthConfig	Permitir el uso de directivas de autorización (AuthDBMGroupFile, AuthDBMUserFile, AuthGroupFile, AuthName, AuthType, AuthUserFile, Require, etc).
FileInfo	Permitir el uso de directivas de control de tipo de documentos (DefaultType, ErrorDocument, ForceType, LanguagePriority, SetHandler, SetInputFilter, SetOutputFilter, etc).
Indexes	Permitir el uso de directivas que controlan los índices de directorios (AddDescription, AddIcon, AddIconByEncoding, AddIconByType, DefaultIcon, DirectoryIndex, FancyIndexing, HeaderName, IndexIgnore, IndexOptions, ReadmeName, etc).
Limit	Permitir el uso de directivas de acceso de hosts (Allow, Deny y Order).
Options	Permitir el uso de las opciones antes vistas en la directiva Options

5.2.2.1.16 Directiva Allow

La directiva Allow indica al sistema los equipos que pueden acceder a una determinada área del servidor. El acceso se puede controlar por nombre, IP, rango de IP, igualmente pueden ser aplicadas a un directorio concreto

El primer argumento de esta directiva es siempre **"from"**. Los siguientes argumentos pueden tener diferentes formas:

All permite el acceso a todos los equipos excepto los especificados en Deny y Order que se verá más adelante.

Para permitir el acceso a un dominio en específico solo se deberá especificar el antes mencionado

Ejemplo:

```
Allow from linuxparatodos.net
```

También puede aplicarse esa misma regla usando direcciones IP

Ejemplo:

```
Allow from 207.249.24.61
```

5.2.2.1.17 Directiva Deny

La directiva Deny indica al sistema los equipos que no podrán acceder al servidor web. El acceso se puede controlar por nombre, IP, rango de IP, igualmente pueden ser aplicadas a un directorio concreto

El primer argumento de esta directiva es siempre **"from"**. Los siguientes argumentos pueden tener diferentes formas:

Para denegar el acceso a un dominio en específico solo se deberá especificar el antes mencionado

Ejemplo:

```
Deny from microsoft.com
```

También puede aplicarse esa misma regla usando direcciones IP

Ejemplo:

```
Allow from 207.249.0.60
```

5.2.2.1.17 Directiva Order

Esta directiva trabaja en conjunto con las dos directivas anteriores así mismo se encarga de controlar el orden en que se ejecutan las antes descritas. Igualmente pueden ser aplicadas a un directorio concreto

Ejemplo 1:

```
Order Deny,Allow
```

En este ejemplo se evaluará primero Deny, de esta forma se permite el acceso a cualquier equipo que no este listado en Deny, de esta forma el acceso se garantiza por defecto.

Ejemplo 2:

```
Order Allow,Deny
```

En este ejemplo se evaluará primero Allow, de esta forma se niega el acceso a cualquier equipo que no este listado en Allow, de esta forma el acceso se niega por defecto.

5.2.2.1.18 Directiva Alias

La directiva Alias permite alojar ficheros fuera del directorio especificado en **DocumentRoot**, igualmente pueden ser aplicadas a un directorio concreto

La sintaxis necesaria para aplicar la directiva Alias es la siguiente:

```
Alias directorioAlternativo "/vaw/www/manual"
```

Ejemplo

```
Alias /home/gerencia "/var/www/gerencia"
```

5.2.2.1.18 Directiva ErrorLog

ErrorLog indica la ubicación del fichero de registro de errores en las consultas. Es conveniente especificar un fichero de registro en cada VirtualHost con el nombre asociado a ese servidor. De esta forma podemos separar los registros de los distintos dominios que tengamos albergados en el servidor web.

Ejemplo:

```
ErrorLog logs/error_log
```

5.2.2.1.19 Directiva ErrorLevel

LogLevel Controla el número de mensajes registrados en error_log.

Puede ser: debug, info, notice, warn, error, crit, alert, emerg.

5.2.2.1.19 Directiva Redirect

La directiva Redirect permite indicar al cliente que un documento ha sido modificado o actualizado.

Ejemplo

```
Redirect permanent /portall http://www.ies-bezmiliana/portal2
```

5.2.3 Iniciar , detener o reiniciar el servidor web Apache

Para iniciar el servidor FTP por primera vez solo deberá teclear en terminal el siguiente comando:

```
[root@ localhost ~]# /etc/init.d/httpd start
```

Igualmente existen opciones ya sea para reiniciar, detener, recargar o conocer el status en el que se encuentra el servidor FTP. Estas opciones pueden ser consultadas en la siguiente tabla:

start	Inicia el servicio
stop	Detiene el servicio
restart	Reinicia el servicio.-La diferencia con reload radica en que al ejecutar un restart este mata todos los procesos relacionado con el servicio y los vuelve a generar de nueva cuenta
reload	Recarga el servicio.-La diferencia con restart radica en que al ejecutar un reload este solamente carga las actualizaciones hechas al fichero de configuración del servicio sin necesidad de matar los procesos relacionados con el mismo, por lo que podría entenderse que hace el cambio en caliente.
condrestart	Reinicio Condicional.- Solamente se inicia si el servicio se encuentra ejecutándose.
status	Da a conocer el estado en el que se encuentra el servicio

Como alternativa también podemos ocupar el siguiente comando para iniciar el servidor FTP

```
[root@ localhost ~]# service httpd start
```

Y de igual manera podemos usar las opciones antes descritas en la tabla anterior.

Recuerde que estos comandos se ejecutan como root.

5.3 Creación de dominios virtuales en Apache

La creación de dominios virtuales sobre un servidor web como apache tiene una vital importancia cuando se trata de dar hospedaje a varios sitios web dentro del mismo servidor.

Lograr implementar de manera correcta los dominios virtuales sobre el servidor web apache es tarea sencilla por lo que le recomendamos primero haber leído todas las directivas que pueden ser aplicadas al fichero

```
httpd.conf
```

A partir de este punto comenzaremos a crear los dominios virtuales, es por ello que pedimos tu total concentración y paciencia para que leas poco a poco estos puntos.

5.3.1 Paso 1.- Activando la directiva NameVirtualHost

El primer paso sera abrir el fichero

```
httpd.conf
```

el cual esta almacenado en la ruta:

```
/etc/httpd/conf/
```

En dicho fichero debemos localizar la siguiente linea y descomentarla si es que lo esta

```
NameVirtualHost *:80
```

La función de esta directiva sirve para indicar la dirección IP en la que se esta brindando el servicio o bien insertando un asterisco(*) para que esté activa en cualquier interfaz del servidor que es como nosotros lo debemos tener.

5.3.2 Paso 2.- Estructura de directorios para dominios virtuales

Lo siguiente sera crear la estructura que contendrá cada uno de los dominios virtuales que serán hospedados en nuestro servidor.

Ejemplo: Suponga que tenemos 5 nombres de dominio que serán hospedados en nuestro servidor web

```
www.turbolinux.com.mx  
www.comerciolinux.com  
www.escuelalinux.edu  
www.linuxunido.org  
www.linuxblogger.net
```

por cada dominio se deberá crear un directorio, dicho directorio sera nombrado de la misma forma que el dominio, solo omitiendo el “www”.

```
[root@ localhost ~]# mkdir turbolinux.com.mx
[root@ localhost ~]# mkdir comercioline.com
[root@ localhost ~]# mkdir escuelaalinux.edu
[root@ localhost ~]# mkdir linuxunido.org
[root@ localhost ~]# mkdir linuxblogger.net
```

Estos directorios deberán ser creados dentro de la ruta

```
"/var/www/"
```

Al final estos directorios deberán quedar de la siguiente manera

```
/var/www/turbolinux.com.mx
/var/www/comercioline.com
/var/www/escuelaalinux.edu
/var/www/linuxunido.org
/var/www/linuxblogger.net
```

Si no están en la ruta antes descrita solo debe moverlos con el comando “mv”

Lo siguiente sera crear dentro de cada uno de estos directorios la estructura básica que debe llevar cada uno de estos dominios. Esta estructura estará conformada por cuatro directorios:

- html
- cgi-bin
- icons
- error

por lo que deberá crear estos cuatro directorios para cada uno de los directorios de dominio.

Ejemplo para el dominio turbolinux.com.mx

```
# mkdir /var/www/turbolinux.com.mx/html
# mkdir /var/www/turbolinux.com.mx/cgi-bin
# mkdir /var/www/turbolinux.com.mx/icons
# mkdir /var/www/turbolinux.com.mx/error
```

y así para los siguientes restantes dominios

5.3.3 Paso 3.- Creación y modificación de los ficheros de configuración de los dominios virtuales

Una vez creadas las carpetas de dominios así como también la estructura de cada uno pasaremos al último paso, crear los ficheros de configuración correspondientes a cada dominio.

Nuevamente por cada dominio se deberá crear un fichero de configuración, dicho fichero sera nombrado de la misma forma que el dominio, solo omitiendo el “www”.

```
[root@ localhost ~]# mkdir turbolinux.com.mx.conf
[root@ localhost ~]# mkdir comercioline.com.conf
[root@ localhost ~]# mkdir escuelaalinux.edu.conf
[root@ localhost ~]# mkdir linuxunido.org.conf
[root@ localhost ~]# mkdir linuxblogger.net.conf
```

Estos directorios deberán ser creados dentro de la ruta

```
"/etc/httpd/conf.d/"
```

Al final estos directorios deberán quedar de la siguiente manera

```
/etc/httpd/conf.d/turbolinux.com.mx.conf
/etc/httpd/conf.d/comercioline.com.conf
/etc/httpd/conf.d/escuelaalinux.edu.conf
/etc/httpd/conf.d/linuxunido.org.conf
/etc/httpd/conf.d/linuxblogger.net.conf
```

Si no están en la ruta antes descrita solo debe moverlos con el comando "mv"

Lo siguiente sera crear dentro de cada uno de estos ficheros la estructura básica que deben contener para que puedan ser leídos por el fichero principal de configuración de apache, nos referimos al fichero "httpd.conf" . Esta estructura estará conformada por la siguiente configuración básica:

Ejemplo de configuración para el dominio turbolinux.com.mx

```
<VirtualHost *:80>
  ServerAdmin administrador@tuDominio.net
  DocumentRoot "/var/www/turbolinux.com.mx/html"
  ServerName www.turbolinux.com.mx
  ServerAlias turbolinux.com.mx
</VirtualHost>
```

Los parámetros usados son descritos en la siguiente tabla:

VirtualHost	La función de esta directiva sirve para indicar la dirección IP en la que se esta brindando o bien insertando un asterisco(*) para que esté activa en cualquier interfaz del servidor que es como nosotros lo debemos tener.
ServerAdmin	Esta directiva especifica la persona a la que se le debe notificar los problemas referentes al portal web , esto a través de su cuenta de correo.
DocumentRoot	Esta directiva indica al servidor web la ruta en donde se encuentran almacenados los ficheros web de tu sitio principal
ServerName	Esta directiva especifica el nombre y puerto que el servidor utiliza para identificarse. Con una correcta configuración, este valor se puede determinar automáticamente, pero es recomendable especificarlo explícitamente para evitar problemas durante el arranque.
ServerAlias	Esta directiva sirve para que el mismo sitio web sea accesible desde distintos nombres de dominio. Ejemplo: turbolinux.com.mx ---> www.turbolinux.com.mx