

PRÁCTICA

EL SERVIDOR WEB APACHE (LINUX)

Servicio Web, Apache Web Server	1
Arranque y parada del servidor	2
Configuración del servidor Apache	3
Directivas en la primera sección "Global environment"	3
ServerRoot	3
PidFile	3
Timeout.....	4
Listen.....	4
LoadModule.....	4
Directivas en la segunda sección "Main server configuration"	5
User y Group.....	5
ServerAdmin	5
DocumentRoot	5
TypesConfig	6
AddDefaultCharset	6
DefaultType	6
Directivas de directorio.....	6
DirectoryIndex	8
ErrorLog	8
LogLevel	8
CustomLog	8
Otras directivas.....	9
Redirect.....	9
ErrorDocument.....	9
Cuestiones de seguridad en el acceso a las páginas	10
ESCENARIO	12
EJERCICIOS.....	13

Servicio Web, Apache Web Server

El servidor Apache (www.apache.org) es actualmente el más utilizado a la hora de configurar un servicio web en Internet. Todas las distribuciones de Linux incluyen una versión de este servidor, que se distribuye bajo la licencia de código abierto.

Además de la funcionalidad básica ofrecida por el programa, existen una serie de módulos adicionales que permiten obtener otros tipos de funcionalidades, desde las más básicas a las más avanzadas, como la capacidad para encriptar los datos de las transmisiones, ejecutar código para generar páginas dinámicas o con acceso a bases de datos en el servidor (php, perl), etc. Todos estos módulos se pueden utilizar o no. La instalación del servidor Apache debería ser aquella que proporcione el funcionamiento deseado, no instalando módulos que

no se vayan a necesitar y que podrían suponer un riesgo ante la existencia de agujeros de seguridad.

Se puede acceder a la documentación sobre Apache una vez instalado éste desde la URL <http://127.0.0.1/manual> y también en la dirección <http://httpd.apache.org/docs>.

Desinstalación e instalación del servidor

La máquina virtual del laboratorio tiene preinstalada una versión del servidor Apache HTTP Server. Lo primero que vamos a hacer es desinstalarlo mediante el siguiente comando:

sudo yum remove httpd

Y una vez se haya borrado, volvemos a instalarlo:

sudo yum install httpd

Arranque y parada del servidor

El demonio encargado de escuchar las peticiones para el servidor Apache se llama httpd y se encuentra en el directorio /etc/rc.d/init.d el cual tiene un enlace simbólico hecho (acceso directo) en la ruta /etc/init.d. Para iniciar, parar o forzar una parada y un posterior arranque del servicio habrá que utilizar las siguientes instrucciones:

/etc/rc.d/init.d/httpd start o /etc/init.d/httpd start

/etc/rc.d/init.d/httpd stop o /etc/init.d/httpd stop

/etc/rc.d/init.d/httpd restart o /etc/init.d/httpd restart

Si lo que se quiere es que la configuración se actualice después de cambiar el archivo httpd.conf sin tener que parar y posteriormente arrancar el servicio habrá que utilizar la instrucción:

/etc/rc.d/init.d/httpd reload o /etc/init.d/httpd reload

También se pueden invocar estas opciones de forma más sencilla y directa mediante la instrucción:

service httpd start|stop|restart|reload

(Cuando esté activo el funcionamiento del servidor seguro se pedirá la contraseña cada vez que se arranque el servicio)

Configuración del servidor Apache

El archivo principal de configuración del servidor Apache es `/etc/httpd/conf/httpd.conf`. Este archivo tiene una configuración por defecto que en muchos casos será suficiente sin tener que modificarla.

Si se producen errores en el funcionamiento del servicio se podrán consultar los mismos en el archivo de registro (log) `/var/log/httpd/error_log`

El archivo de configuración `httpd.conf` tiene tres secciones principales.

- La primera sección está dedicada a características (llamadas directivas) genéricas que controlan aspectos globales del servidor.
- La segunda sección contiene directivas que controlan aspectos del host no-virtual, es decir, el host por defecto de Apache.
- La tercera sección es la que contiene las directivas para diferentes hosts virtuales.

Seguidamente se comentan las principales directivas que se encuentran en el archivo de configuración. En esta práctica nos centraremos en las dos primeras secciones, dejando la tercera para una práctica posterior.

Directivas en la primera sección "Global environment"

ServerRoot

Esta directiva especifica el directorio en el que ha sido instalado el servidor. Normalmente contendrá los subdirectorios `conf/` y `logs/`. Por defecto su valor será `/etc/httpd`:

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

A todo directorio que se nombre de forma no absoluta en otras directivas le será añadido este valor.

Es importante no confundir este directorio raíz con el directorio raíz de los documentos que servirá el servidor web. En este último caso, se debe utilizar la directiva `DocumentRoot`, vista más adelante.

PidFile

Esta directiva especifica el fichero en el que el servidor guarda el ID del proceso demonio de escucha (daemon). Si el nombre del fichero

especificado no es una ruta absoluta, entonces se asume que es relativa al directorio especificado en ServerRoot.

PidFile run/httpd.pid

(esta directiva puede omitirse)

Timeout

Esta directiva indica, en segundos, la cantidad de tiempo que el servidor esperará por recepciones y transmisiones durante la comunicación. Por ejemplo:

(esta directiva puede omitirse)

Timeout 300

Listen

La directiva Listen sirve para indicar el puerto en el que se escucharán las peticiones de servicio web. El valor por defecto es 80:

Listen 80

Esta directiva se puede utilizar para indicar también la dirección o direcciones IP particulares en las que se aceptarán conexiones, en el caso de que se tenga configurada más de una IP.

LoadModule

Se usa para cargar módulos compartidos de forma dinámica. Por cada módulo que se desee cargar aparecerán un par de líneas que comienzan por LoadModule y AddModule respectivamente. El orden en que se cargan los módulos es importante. No cambiar de orden las líneas que aparecen por defecto.

<IfModule nombremodulo>.....</IfModule>

Estas etiquetas de apertura y cierre encierran directivas que se aplicarán sólo si el módulo indicado en nombremodulo está cargado en el servidor Apache.

Los módulos sirven para complementar el funcionamiento básico de Apache, añadiéndole nuevas funcionalidades. Muchos de los módulos que se cargan en el archivo por defecto no se necesitan para la presente práctica pero por facilidad de configuración los dejaremos todos tal como están en el archivo httpd.conf por defecto.

Directivas en la segunda sección "Main server configuration"

Estas directivas serán utilizadas para el servidor "principal" que funcionará en ese ordenador, es decir, serán las directivas aplicadas a todas las peticiones que no sean recogidas por ningún "host virtual" (visto más adelante). Estas directivas también serán las que se utilicen por defecto en el caso de no indicar nada en las secciones correspondientes a cada host virtual (dentro de las etiquetas <VirtualHost>.....</VirtualHost>).

User y Group

Estas directivas se utilizan para indicar la cuenta de usuario que servirá para contestar las peticiones del servidor. Cualquier archivo inaccesible para el usuario indicado lo será para cualquier usuario que se conecte al servidor. El usuario indicado mediante esta directiva sólo debería tener privilegios para acceder a archivos que se supone que se quiere que sean visibles para todo el mundo.

Por defecto, existen tanto el usuario "apache" como el grupo "apache". Este usuario y grupo se crean de forma automática al instalar el servicio. Su uso se indicaría del siguiente modo:

User apache
Group apache

ServerAdmin

Esta directiva se utiliza para indicar la dirección del correo electrónico del administrador del web (webmaster). El valor de esta directiva será automáticamente utilizado al generar mensajes de error, de forma que los clientes tengan una dirección de correo a la que notificar problemas.

(esta directiva puede omitirse)

DocumentRoot

La directiva DocumentRoot indica el directorio que contiene los archivos HTML que se van a servir ante las peticiones de los clientes. Por defecto, el valor de esta directiva es "/var/www/html". Por ejemplo, si el ordenador se llama labf2.intracampus.unileon.es, al poner:

<http://labf2.intracampus.unileon.es/principal.html>

el servidor trataría de servir el archivo:

/var/www/html/principal.html

TypesConfig

La directiva TypesConfig indica el archivo donde están almacenados los diferentes tipos de archivos que el servidor podrá servir. Estos tipos de archivo se definen mediante el estándar MIME y pueden ser: texto plano, imagen jpg, texto html, archivo pdf, etc. El valor de la directiva será el archivo que contiene esta información, por ejemplo:

TypesConfig /etc/mime.types

AddDefaultCharset

Otra directiva importante a la hora de visualizar correctamente las páginas web en un navegador es AddDefaultCharset. Las páginas web se envían a los navegadores utilizando una codificación determinada. Dependiendo del tipo de código utilizado se podrán representar más o menos caracteres. Por ejemplo, hay codificaciones que no pueden representar las vocales acentuadas del idioma español, otros que no pueden representar caracteres arábigos, etc.

El tipo de codificación que más caracteres puede representar y que por tanto es el más recomendable para codificar las páginas es el UTF-8. Esto se indicaría mediante la directiva:

AddDefaultCharset UTF-8

DefaultType

Esta directiva se utiliza para indicar al servidor el tipo de archivo que debe suponer cuando no es capaz de averiguarlo a través de la extensión del mismo. Por ejemplo, para que los archivos que no se puedan reconocer se envíen por defecto como texto plano se indicaría:

DefaultType text/plain

Una vez definidas estas directivas, es hora de ir definiendo las restricciones de acceso y de uso de cada directorio del sitio web, ya que puede haber directorios donde sólo haya archivos de lectura, otros de lectura/escritura, otros con acceso restringido, etc.

Directivas de directorio

Todo lo relativo a cada directorio se encierra entre etiquetas <Directory nombredirectorio>.....</Directory>

Todo lo que se especifique para un directorio será aplicado a todos los subdirectorios que queden dentro de él. Por esta razón, lo habitual es aplicar las opciones más restrictivas posibles al directorio raíz (en este caso "/" representa al directorio raíz del sistema de archivos de Linux):

```
<Directory />
    AllowOverride None
    Order allow,deny
    Deny from all
</Directory>
```

Una vez indicado este grupo de directivas, si se quiere dar acceso menos restringido a cualquier directorio del sistema se debe hacer explícitamente. En concreto, lo primero sería dar las opciones para el directorio especificado en DocumentRoot:

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

La directiva Options controla qué características del servidor están disponibles en cierto directorio. El valor "Indexes" sirve para que, ante la falta de un documento por defecto que servir al usuario, se permita el listado del contenido del directorio.

La directiva AllowOverride dice si se puede o no se puede sustituir alguna de las opciones indicadas en Options por medio de declaraciones incluidas en un archivo .htaccess (visto más adelante). En los ejemplos, tanto el directorio raíz como el DocumentRoot no permiten que lo indicado en Options pueda ser "puenteado".

La directiva Allow especifica qué cliente puede acceder a un determinado directorio. El cliente se puede indicar mediante la palabra all (para todos), un nombre de dominio, una dirección IP, una par red/máscara, etc. En el ejemplo, el directorio DocumentRoot está configurado para permitir acceso de cualquier cliente (Allow from all).

De forma similar, la directiva Deny se utiliza para indicar qué cliente no puede acceder a un determinado directorio.

La directiva Order sirve para controlar el orden en el que se procesan las directivas Allow y Deny. Por ejemplo, en el directorio

DocumentRoot está configurado evaluar las directivas allow antes que las deny.

Estas directivas se verán de nuevo al hablar de la seguridad.

DirectoryIndex

Esta directiva especifica el archivo (página web típicamente) que se servirá ante una petición que sólo indica un directorio y no un archivo determinado. Se pueden indicar más de un documento, en cuyo caso se separarán por medio de un espacio. Por defecto (en el archivo que viene predeterminado) esta directiva tiene los valores: index.html index.htm index.shtml index.php index.php4 index.php3 index.cgi. El servidor tratará de encontrar cualquiera de estos archivos y devolverá el primero que encuentre. Si no se encuentra ninguno de estos documentos y la opción Option Indexes está activa para este directorio, el servidor devolverá un listado del contenido del directorio, en formato HTML, en el que se reflejarán los archivos y directorios que el directorio contiene.

Ejemplo:

DirectoryIndex inicio.html

Archivos de registro (log)

El servidor Apache puede ser configurado para que guarde en archivos de registro muchas de las actividades del servidor, accesos de los usuarios, errores producidos, etc. A continuación, se verán directivas que se utilizan para este contenido.

ErrorLog

Sirve para indicar dónde se guardan los registros de error del servidor. Este archivo es donde habrá que mirar cuando se produzca algún error en el servidor.

LogLevel

Indica el nivel de detalle con que se describirán los mensajes de error en los archivos de registro. Los valores pueden ser (de menos a más información) emerg, alert, crit, error, warn, notice, info o debug.

CustomLog

Define qué archivos se utilizan y cómo son esos archivos para guardar los registros de acceso al sitio web.

Otras directivas

Redirect

Se utiliza para redirigir una petición de una página a otra. Se hace cuando una página es cambiada de sitio. Por ejemplo, si se pone:

Redirect /carpeta/archivo.html <http://www.elpais.es>

se estará redirigiendo las peticiones de la página archivo.html del directorio "carpeta" del sitio web a la página principal de el diario El País. Obviamente, este no es el comportamiento habitual que se busca, sólo se hace para demostrar el uso de la directiva.

Las directivas IndexOptions, AddIconByEnconding, AddIconByType, AddIcon, DefaultIcon, AddDescription, ReadmeName, HeaderName, IndexIgnore se utilizan para configurar la forma en la que se presentarán listados de directorios cuando no se encuentra una página por defecto en el directorio solicitado.

ErrorDocument

Esta directiva sirve para devolver una página web determinada cuando se produce un error en el servidor. Cada error que se produce en el servidor tiene asociado un código de tres números que lo identifica. Por ejemplo, cuando una página no se encuentra, se genera un error 404 "página no encontrada". Cuando se pide un documento para el que no se tienen permisos de acceso se produce un error 403 "acceso prohibido", y así sucesivamente.

Para cada código de error se puede definir un documento o página web a enviar al cliente mediante la directiva ErrorDocument. Por ejemplo, para enviar una página web que informe de una petición de una página que no se ha encontrado se haría:

ErrorDocument 404 /errores/paginanoencontrada.html

En este caso, la página a enviar estará en el subdirectorio "errores" del directorio principal del sitio web, que a su vez viene definido por la directiva DocumentRoot, por lo que la situación real de esa página sería:

/var/www/html/errores/paginanoencontrada.html

para el caso en que DocumentRoot tenga el valor /var/www/html

También es posible utilizar directorios fuera del sitio web para almacenar las páginas de errores. En este caso, para que el servidor pueda acceder a las páginas, habrá que crear un alias que represente al directorio donde están las páginas. Por ejemplo, si se quieren almacenar las páginas de error en el directorio /var/www/errores, que está fuera del sitio web, habría que indicar:

```
Alias /errores/ /var/www/errores/  
ErrorDocument 404 /errores/paginanoencontrada.html
```

Cuestiones de seguridad en el acceso a las páginas

A la hora de configurar un servidor web es muy importante tener en cuenta las cuestiones de seguridad, especialmente si se va a dar un servicio público (no sólo a clientes de la red local). Es necesario proteger no sólo las carpetas y documentos que se exponen en Internet sino también el resto de carpetas que se encuentran en el equipo.

Una forma básica y sencilla de configurar el acceso al servicio web es utilizar las directivas Allow, Deny y Order. Estas directivas se usan para permitir o denegar el acceso a las peticiones dependiendo del sitio de donde provengan, es decir, dependiendo de la IP o del dominio de donde llegue la petición.

Allow se utiliza para especificar qué sitios están autorizados a ver cierta información. Deny se utiliza para denegar el acceso a ciertos sitios. Order se utiliza para especificar en qué orden se deben procesar las directivas Allow y Deny. Se pueden indicar direcciones IP, nombres de equipos, nombres de dominios, direcciones IP incompletas, nombres de dominio incompletos o también "all".

Estas directivas suelen utilizarse dentro de las etiquetas <Directory nombredirectorio>.....</Directory> para especificar la seguridad de un determinado directorio y sus subdirectorios.

En este ejemplo:

```
<Directory /var/www/html/directorio>  
.....  
.....  
Order deny,allow  
Deny from all  
Allow from 10.30.48.172  
</Directory>
```

se permite el acceso sólo al equipo 10.30.48.172. Para conseguir esto primero (por lo indicado en la directiva Order) se deniega todo acceso y luego se permite sólo a un equipo. Si se hubiese puesto sólo la directiva allow, estaríamos permitiendo el acceso a todo el mundo y no sólo a la dirección 10.30.48.172.

Para asegurar mejor la protección del sistema es buena práctica indicar unos permisos de acceso muy restrictivos desde el directorio raíz del sistema y luego ser menos restrictivo en los directorios dedicados a las páginas web:

```
<Directory />
.....
.....
Order deny,allow
Deny from all
</Directory>

<Directory /var/www/html>
.....
.....
Order deny,allow
Allow from all
</Directory>
<Directory /var/www/html/directorioparausuariosinternos>
.....
.....
Order deny,allow
Deny from all
Allow from 10.30.48
</Directory>
```

En cada directorio del sitio web pueden existir archivos en los que se especifican directivas de configuración adicionales. Cuando existen estos archivos, los permisos asignados dentro de ellos prevalecen sobre los vistos antes en el fichero de configuración de Apache. Se dice que los permisos del fichero principal son sobrepasados o "puenteados" por los especificados en esos archivos, en inglés se expresa con la palabra **override**.

Para proteger los directorios más sensibles de esta posibilidad se debe impedir cualquier posibilidad de que se produzca este "override", utilizando para ello la directiva AllowOverride con un valor "None". Por ejemplo, en el caso del directorio raíz, debería escribirse:

```
<Directory />
AllowOverride None
Order deny,allow
```

```
Deny from all
</Directory>
```

La directiva AllowOverride también bloquea cualquier intento de cambiar lo especificado a nivel de archivo de configuración principal en las directivas Options.

Cuando se permite que existan los archivos y que prevalezcan sobre la configuración general se suele hacer para permitir a algunos usuarios controlar los permisos de acceso a sus páginas personales. Estos archivos suelen tener como nombre .htaccess (empieza por un punto, lo cual indica que es un archivo que no se listará ante una orden ls a no ser que se indique que se quieren ver también ese tipo de archivos).

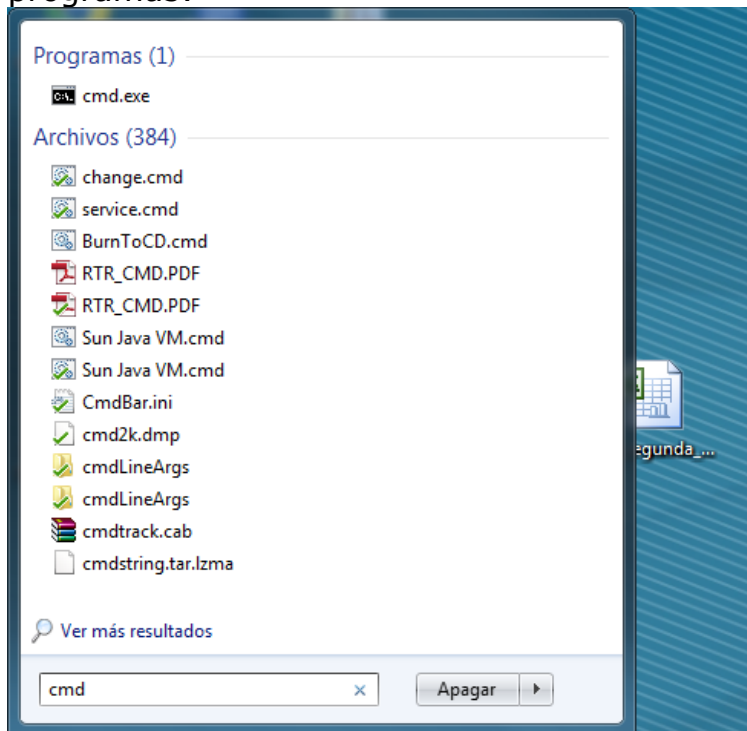
El nombre de los fichero donde se indican configuraciones adicionales se indica mediante la directiva AccessFileName . Por defecto:

AccessFileName .htaccess

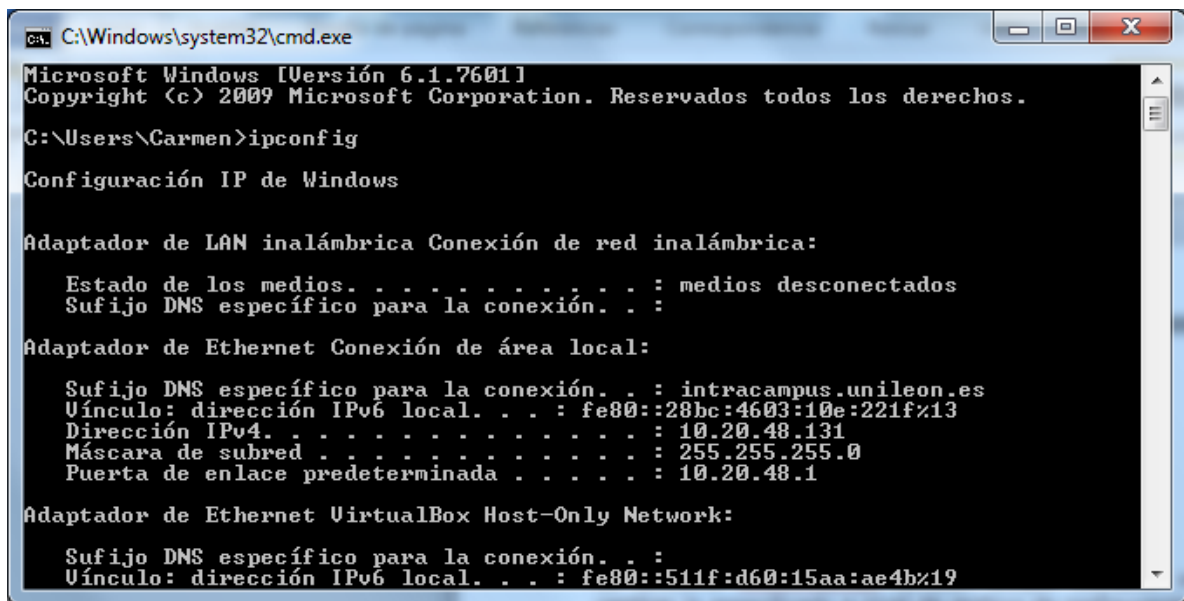
ESCENARIO

Se utilizará Apache sobre un Centos Virtualizado(huésped) sobre el Windows del ordenador del laboratorio (anfitrión).

En primer lugar, identificar la IP del Windows, para ello abrir un terminal ejecutando cmd.exe (botón de inicio de Windows), buscar programas:



Insertar el comando ipconfig y anotar la IP del adaptador Ethernet:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\Carmen>ipconfig

Configuración IP de Windows

Adaptador de LAN inalámbrica Conexión de red inalámbrica:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . :

Adaptador de Ethernet Conexión de área local:

    Sufijo DNS específico para la conexión. . : intracampus.unileon.es
    Vínculo: dirección IPv6 local. . . : fe80::28bc:4603:10e:221f%13
    Dirección IPv4. . . . . : 10.20.48.131
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . : 10.20.48.1

Adaptador de Ethernet VirtualBox Host-Only Network:

    Sufijo DNS específico para la conexión. . :
    Vínculo: dirección IPv6 local. . . : fe80::511f:d60:15aa:ae4b%19
```

En la figura de ejemplo sería la 10.20.48.131

En segundo lugar, identificar la IP del Centos, para ello abrir un terminal y ejecutar ifconfig, apuntar la IP. Esta será la IP del Servidor Web ya que Apache está instalado en el linux

EJERCICIOS

(Para la edición del archivo de configuración se recomienda utilizar el programa gedit por su mayor facilidad de uso y posibilidad de copiar-pegar).

0.- En el Centos abrir el navegador e introducir la dirección IP del servidor Web. ¿Qué obtienes? ¿Por qué?

1.- En el Centos ejecutar desde un terminal:

```
#service httpd start
```

Httpd es el demonio (proceso) encargado de escuchar las peticiones para el servidor Apache.

En el navegador introducir la dirección IP del servidor Web. ¿Qué obtienes?

Desde el Windows o host anfitrión, abrir un navegador e introducir la Ip del servidor Web. ¿Qué obtienes? ¿Por qué? ¿Puedes acceder a los servidores Web de tus compañeros?

2. Realizar una copia de seguridad del archivo de configuración actual. Borrar el contenido del archivo original de configuración y pegar el contenido que aparece en el fichero "módulos de apache" en ahora (básicamente contiene las directivas LoadModule del archivo httpd.conf original).

A continuación, añadir las directivas imprescindibles :

```
ServerName
ServerRoot
User
Group
Listen      80
TypesConfig
...
```

El sitio web por defecto estará alojado en el directorio:

`/var/www/webpredet`

Crear una página de inicio y probar que el servidor funciona. Una página de inicio sencilla puede ser la siguiente (guardar las siguientes líneas en un archivo de texto `index.htm` ó `index.html`):

```
<html>
<body bgcolor="#FFEE00">
Página principal del sitio web por defecto de NOMBRE DEL
ALUMNO
</body>
</html>
```

Nota: En la dirección URL del navegador se debe poner localhost o la dirección IP del ordenador.

3.- Solicita en el navegador una página que no exista, por ejemplo:
IP del servidor/paginaazul.html ¿Qué obtienes?

A continuación, crear una página para devolver cuando se produzca un error de "página no encontrada" (404), configurar el Apache para que sirva esta página al producirse este error. La página de error se guardará en el directorio:

`/var/www/webpredet/errores`

El código para la página puede ser el siguiente:

```
<html>
<body bgcolor="#FF33FF">
PÁGINA NO ENCONTRADA
<br>
</body>
</html>
```

Comprobar que al pedir una página que no existe se devuelve nuestra página de error y no la que presenta por defecto el navegador.

4.- Configurar la seguridad de acceso a los directorios del sitio web mediante las directivas **<Directory>**:

a) En primer lugar, cortar todo acceso a los subdirectorios por debajo del directorio raíz. Comprobar que ya no se puede acceder a ninguna de las páginas web del sitio.

b) Dar acceso a la IP del Windows. Comprobar que desde el navegador de Windows puedes acceder. Intenta acceder desde el Centos. ¿Puedes?

c) Seguidamente dar permiso de acceso a todo el mundo al directorio donde está nuestro sitio web, comprobando que se vuelve a tener acceso a las páginas web.

5.- Además de las directivas de seguridad y acceso a los directorios que se programan en Apache mediante las directivas **<Directory>**, también se siguen aplicando a los ficheros los permisos normales de archivos y directorios especificados a nivel de sistema operativo (es decir, aquellos que se pueden ver ejecutando el comando **ls -la** en un directorio).

Para comprobar esto se pide: Primero, hacer una copia de la página de inicio que se llame inicio2.html, cambia el color de fondo o el texto para diferenciarlas bien al acceder a ellas. Comprobar que se accede a ella desde la página web. Segundo: cambiar los permisos de ese fichero mediante el comando **chmod** de forma que no se den permisos de lectura para el mismo. Comprobar que ya no se puede ver esa página desde el navegador. ¿Qué permisos se han cambiado y por qué? ¿Sigues accediendo a index.html? Vuelve a modificar los permisos de inicio2.html para que se pueda acceder.

6.- Cambia el puerto en el que escucha el servidor al 8080. Comprueba si accedes a la página de inicio. ¿Cómo indicas en la dirección Web que quieres acceder a través del puerto 8080?

7.- Crea un fichero **info.php** dentro de la carpeta **/var/www/html** que contenga el siguiente código:

```
<?php phpinfo(); ?>
```

Comprueba con el navegador qué se observa:

<http://localhost/info.php>

8.- Instala el modulo php desde un terminal haciendo uso del comando **sudo yum install php**. Reinicia el servicio httpd y vuelve a acceder al enlace del ejercicio anterior <http://localhost/info.php>

Prueba a guardar el sitio web que ves desde el navegador y tras guardarlo ábrelo con el bloc de notas.

¿Ves el mismo código que habías escrito en el fichero info.php?

9.- Supón que deseas acceder a tu servidor tecleando www.miweb.com en el navegador ¿se te ocurre alguna forma de hacerlo posible? Indica cómo.