

41311 Laboratorio de Interfaces de Redes

Práctica 2 - Configuración del Apache

1. Objetivos

En esta práctica vamos a familiarizarnos con la configuración del servidor web Apache. Este programa dispone de un gran número de opciones de configuración. Veremos algunas de ellas y a medida que otras hagan falta se presentarán en las siguientes prácticas.

2. Introducción

En esta práctica les será muy útil el manual del Apache que pueden encontrar en la [página web](#) de la asignatura o en su propia instalación del Apache. Las referencias a secciones de la documentación se harán con URLs a la página de la asignatura pero pueden visitar cualquier otra copia de la documentación.

3. Ficheros de configuración

Apache se configura colocando *directrices* en ficheros de configuración, que están formados por simple texto. El fichero principal de configuración se suele llamar `httpd.conf` y su localización se puede especificar al compilar (opciones del script `configure`). También se puede ejecutar el servidor `httpd` con la opción `-f` lo cual permite especificarle el fichero de configuración a emplear. Además, se pueden añadir otros ficheros empleando la directriz `include`. Esta directriz se coloca en el fichero de configuración especificando el nombre de otro fichero de configuración cuyo contenido deseamos que se incluya ahí. Es muy importante tener en cuenta que Apache solo lee el fichero de configuración cuando es lanzado, por lo que cualquier cambio realizado en él requiere pararlo y volverlo a lanzar para que los cambios surtan efecto.

Podemos incluir comentarios en el fichero de configuración comenzando la línea con el carácter `#`. También podemos comprobar la sintaxis del fichero de configuración lanzando el `httpd` con la opción `-t` o empleando `apachectl configtest`

4. Algunas directrices básicas

Estudie el significado de las siguientes directrices:

- *ServerRoot*
- *Listen*
- *ServerAdmin*
- *ServerName*
- *DocumentRoot*
- *DirectoryIndex*

Puede encontrar todas las directrices en:

http://www.tlm.unavarra.es/~daniel/docencia/lir/lir03_04/manuales/httpd/mod/directives.html

A continuación:

- Adapte la configuración de todas esas directrices en su fichero `httpd.conf`
- Haga que la dirección de correo de administrador sea la de uno de los miembros del grupo de prácticas
- Que el servidor espere conexiones al puerto 3128 y al 8080
- Que sirva los ficheros que se coloquen en `/opt3/lir/lirXY/webs`
- Que cuando se solicite un directorio intente primero servir un fichero `index.php` que esté en el directorio y si no lo encuentra busque un `index.html` y si no un `index.htm`

Compruebe que todo esto funciona.

Checkpoint 1

Haga una demostración al profesor de prácticas de que todos esos cambios le funcionan

5. Alcance de las directrices

Las directrices colocadas directamente en el fichero de configuración se aplican a todo el servidor. Sin embargo, se pueden colocar directrices dentro del ámbito de otra, de forma que solo apliquen dentro del alcance de la segunda. Por ejemplo, las directrices colocadas entre un `<Directory>` y su correspondiente `</Directory>` se aplican solo para el directorio indicado. Ejemplo:

```
<Directory /home/miweb>  
    AllowOverride All  
</Directory>
```

Pero hay algunas directrices que solo tienen sentido de forma global y por lo tanto no se pueden colocar dentro de estas directrices de bloque.

6. Directrices de bloque

Estudie el significado de las siguientes directrices de bloque:

- *Directory*
- *Files*
- *Location*

Encontrará una buena explicación de las diferencias entre ellas en:

http://www.tlm.unavarra.es/~daniel/docencia/lir/lir03_04/manuales/httpd/sections.html

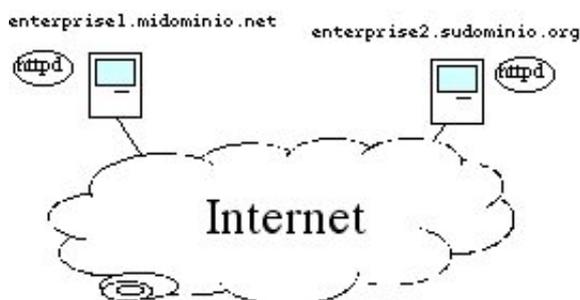
Cuestiones:

El sistema de ficheros `ext3` empleado en los linux del laboratorio distingue mayúsculas de minúsculas en los nombres de ficheros. Sin embargo, por ejemplo el sistema de ficheros `HFS+` no las distingue, para él el fichero `MiFichero` y `Mifichero` son el mismo. Qué consecuencias puede tener esto a la hora de limitar el acceso a ciertos directorios del servidor web?

7. VirtualHost

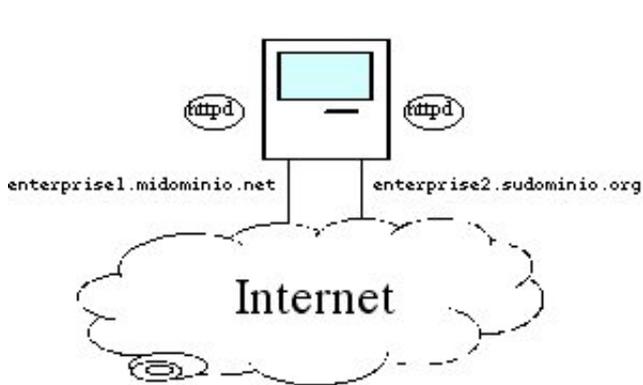
Existen más directrices de bloque. Una de ellas es `<virtualHost>` la cual nos permite servir más de un sitio Web con el mismo servidor Apache.

Suponga que queremos instalar el servidor web para dos empresas independientes. Una posibilidad es tener dos máquinas distintas, cada una con su interfaz de red, su dirección IP, su nombre DNS y ejecutando Apache, una contiene las páginas de la empresa 1 y se llama `enterprise1.midominio.net` y la otra contiene las páginas de la empresa 2 y se llama `enterprise2.sudominio.org`.

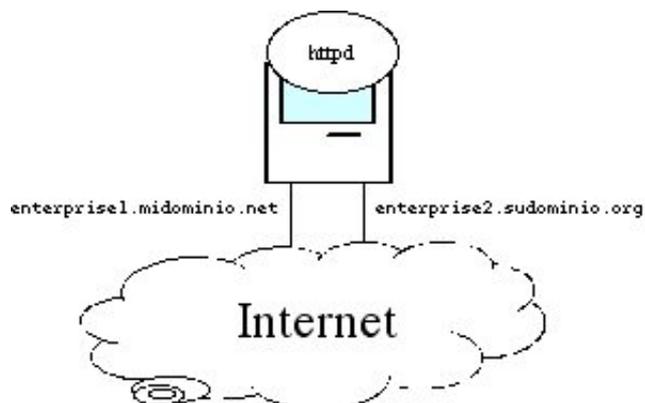


Dos servidores independientes

Perfecto, pero hay más formas de hacerlo. Por ejemplo, podríamos tener una sola máquina pero con dos tarjetas de red, cada una con una dirección IP. Para cada IP tenemos un nombre de dominio (DNS) diferente. Entonces podemos ejecutar dos copias del programa Apache simultáneamente, una de ellas atendiendo a peticiones que vengan a una de las direcciones IP y el otro atendiendo a las peticiones que vayan a la otra. Cada programa Apache corriendo posee un fichero de configuración diferente y un directorio con páginas web a servir diferente. Estudie cómo debería configurar y lanzar cada copia de Apache.



Un servidor con 2 interfaces y 2 programas Apache corriendo

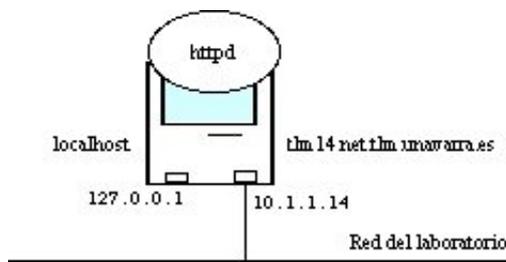


Un servidor con 2 interfaces y 1 programa Apache corriendo

Una tercera posibilidad es tener dos direcciones IP pero solo ejecutar un servidor Apache, no dos. Ese servidor tendrá que servir las páginas de una u otra empresa según a qué interfaz se dirijan las peticiones.

Esta tercera posibilidad es la que vamos a intentar configurar a continuación. Las máquinas del laboratorio no tienen dos tarjetas de red pero sí tienen dos interfaces de red y por tanto dos

direcciones IP (lo cual puede comprobar ejecutando el programa `/sbin/ifconfig`). Esto se debe a que normalmente todas las máquinas que emplean TCP/IP tienen un interfaz que se llama de *Loopback* que no corresponde a una tarjeta de red sino a software dentro del sistema operativo (podríamos tener este interfaz sin tener tarjeta de red). Este interfaz tiene configurada la dirección IP `127.0.0.1` y todo el tráfico que se dirija a esa IP se queda dentro de la máquina.



Un servidor con 2 interfaces (uno es localhost) y 1 programa Apache corriendo

Estudie la sintaxis de la directriz `<virtualHost>`. Por ejemplo lea:

http://www.tlm.unavarra.es/~daniel/docencia/lir/lir03_04/manuales/httpd/mod/core.html#virtualhost

Configure su servidor Apache para que:

- Atienda a conexiones dirigidas al puerto 3128
- Cuando se le solicite la página `localhost:3128/presentacion.html` sirva el fichero `/opt3/lir/lirXY/webs/presentacion.html`
- y cuando se le solicite la página `t1mAB.net.tlm.unavarra.es:3128/presentacion.html` sirva el fichero `/opt3/lir/lirXY/www/htdocs/presentacion.html` donde AB debe substituirse por lo que corresponda a la máquina donde tenga corriendo el Apache
- Cree unos ficheros HTML simples:
`/opt3/lir/lirXY/webs/presentacion.html` y
`/opt3/lir/lirXY/www/htdocs/presentacion.html` para probarlo.

Checkpoint 2

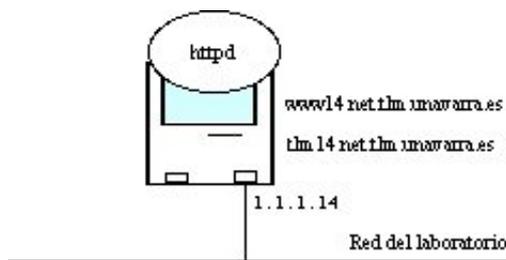
Muestre al profesor de prácticas que le funciona esta última configuración y explique cómo lo ha hecho.

Ahora supongamos que nuestra máquina tiene un solo interfaz de red (ignoremos la IP `127.0.0.1` dado que nadie de fuera de nuestra máquina puede en realidad comunicarse con ella!). Podemos llegar a una solución similar a la anterior con la ayuda de los nombres de dominio (DNS). Las máquinas del laboratorio tienen como nombre `t1mAB.net.tlm.unavarra.es` el cual se convierte en la dirección IP `10.1.1.AB`. Hemos configurado el servidor de DNS del laboratorio para que ADEMÁS, el nombre `wwwAB.net.tlm.unavarra.es` también se resuelva como la dirección `10.1.1.AB`. Es decir, podemos dirigirnos a nuestras máquinas por cualquiera de los dos nombres y ambos se convertirán en la misma dirección IP antes de mandar los paquetes IP (recuerde que en los paquetes IP se ponen direcciones IP, no nombres, si tenemos un nombre, antes de poder enviar un paquete IP a esa máquina debemos averiguar a qué dirección IP corresponde ese nombre).

- Emplee `<virtualHost>` para que su servidor web ofrezca los ficheros en `/opt3/lir/lirXY/www/htdocs` cuando se soliciten con el formato

`http://t1mAB.net.t1m.unavarra.es:3128`

- además debe servir (con la misma configuración) del directorio `/opt3/lir/lirXY/webs` cuando se le soliciten con el nombre `http://wwwAB.net.t1m.unavarra.es:3128`
- No debe aceptar peticiones a ningún otro puerto.



Un servidor con 1 interfaz, dos nombres y 1 programa Apache corriendo

Cuestiones:

Cómo puede saber el servidor Web que se le está solicitando una página dirigiéndose a él con un nombre o con otro si en la cabecera de los paquetes IP solo aparecen direcciones IP y no nombres?

Checkpoint 3

Muestre al profesor de prácticas que le funciona esta última configuración y explique cómo lo ha hecho.

8. Trabajos opcionales para ampliar conocimientos sobre la configuración de Apache

A. Allow, Deny

Podemos controlar a qué clientes serviremos ciertas páginas y a cuáles no. Para ello disponemos de las directrices `Allow` y `Deny`. Puede encontrar una explicación de su funcionamiento en:

http://www.t1m.unavarra.es/~daniel/docencia/lir/lir03_04/manuales/httpd/mod/mod_access.html#allow

Continuando con la configuración de la sección anterior (se sirven 2 directorios diferentes según se le solicite a `wwwAB` o `t1mAB`):

- Modifique la configuración para que solo se sirvan las páginas de `wwwAB` a las máquinas de la `t1m11` a la `t1m31`
- y con la misma configuración las páginas de `t1mAB` solo a todas las demás.

B. Autenticación

Además de limitar las máquinas a las que serviremos ciertas páginas basándonos en su dirección IP o su nombre podemos también dejar acceder solo a ciertos *usuarios* donde nosotros definimos tales usuarios mediante un nombre de usuario y una password.

Las directrices típicas para esta funcionalidad son:

`AuthType`, `AuthName`, `AuthGroupFile`, `AuthUserFile` y `Require`

Encontrará explicado cómo funcionan en:

http://www.tlm.unavarra.es/~daniel/docencia/lir/lir03_04/manuales/httpd/howto/auth.html

- Modifique la configuración anterior para probar la autenticación, solo para solicitudes hechas a `tlmAB:3128`
- Para ello defina dos usuarios nuevos: `alumno` y `profesor` con passwords `aula315` y `d-1.31` respectivamente
- Cree un directorio con nombre `privado` en la raíz servida por el Apache para `tlmAB`
- Configure el servidor para que solicite password para todos los directorios de `tlmAB` de forma que `alumno` pueda acceder a todo menos al contenido de `privado` y que `profesor` pueda acceder a todo.

Cuestiones:

Qué haría falta para poder emplear el método de autenticación *Digest* en vez del *Basic*?

C. Ficheros `.htaccess`

Hasta ahora hemos visto directrices que hemos colocado en el fichero de configuración global `httpd.conf`. Sin embargo, podemos colocar directrices también en otros ficheros. Especialmente interesante son los ficheros que normalmente se llaman `.htaccess` que permiten cambiar la configuración que se aplica a los directorios en los que se encuentran. Todo lo que se puede hacer con ellos se puede hacer desde el fichero de configuración global pero hay situaciones en las que son útiles. Podemos controlar el tipo de directrices que se pueden colocar en estos ficheros mediante la directriz `AllowOverride`. Puede encontrar un tutorial respecto al empleo de estos ficheros en:

http://www.tlm.unavarra.es/~daniel/docencia/lir/lir03_04/manuales/httpd/howto/htaccess.html

Modifique la configuración de su servidor para que emplee ficheros `.htaccess` dentro de `/opt3/lir/lirXY/webs2` y configure mediante un fichero de este tipo que se solicite password para acceder a ese directorio, siendo válido solo el usuario `profesor` que creamos con anterioridad.

9. Conclusiones

Hemos aprendido a configurar Apache para que ofrezca varias webs diferentes. Esto nos servirá en las siguientes prácticas dado que el servidor Web va a ser la herramienta que dé el soporte principal a los scripts que creamos.