

Apache: micro-COMO  
GarZa - garzalin@worldonline.es  
27 de Octubre de 1999

Este micro-COMO cubre la configuración básica del servidor http Apache

---

## Introducción

Lo que pretendo es instalar un servidor web en el servidor "pinero" y que los usuarios publiquen y gestionen sus páginas web. Estoy utilizando la distribución RedHat 6.0 con el kernel 2.2.5 y el paquete apache-1.3.6-7. Lo tengo instalado en /home/httpd/ y los ficheros de configuración están en /etc/httpd/conf, aquí se encuentran los ficheros httpd.conf donde se especifica la configuración general, srm.conf para configurar recursos y access.conf para configurar permisos, como sólo es necesario httpd.conf y me resulta más cómodo manejar un fichero que tres, los fusiono con:

```
# cat srm.conf >> httpd.conf
# cat access.conf >> httpd.conf
```

## Configuración

Quiero tener la siguiente estructura de páginas web, en http://www.pinero/index.html estará la página principal y en http://www.pinero/~nombre\_usuario, estarán las páginas de los usuarios:

```
/home/httpd/html/index.html
|___ garza/index.html
|___ paco/index.html
|___ lolo/index.html
```

En /home/httpd/html/ creo un subdirectorio por cada usuario que va a publicar páginas web, con permisos 755 y como propietario del directorio el usuario. El umask de cada usuario debe ser 022, así los directorios se crearán con permisos 755 y los archivos con 644 y todos podremos ver todas las páginas.

En httpd.conf modifico lo siguiente:

```
User nobody
Group nobody
ServerAdmin garza@pinero -- email del usuario administrador de apache
ServerName pinero -- nombre del servidor
ServerRoot /etc/httpd -- directorio donde están los ficheros de configuración y logs
```

Por motivos de seguridad es conveniente asignar como usuario y grupo a nobody, ya que si atacasen nuestro sistema, aprovechando cualquier bug de apache, el atacante entraría en el sistema como nobody, que se trata de un usuario con pocos privilegios. Cuando un usuario navega por las páginas del servidor web, lo hace con los privilegios de nobody.

Especifico donde voy a alojar las páginas:

```
DocumentRoot /home/httpd/html -- lugar donde se van a almacenar las páginas, corresponde a http://pinero/index.html
UserDir public_html /home/httpd/html -- páginas de usuarios, corresponde a http://pinero/~login_usuario
```

Configuro la directiva:

```
<Directory /home/httpd/html>
Options Indexes Includes FollowSymLinks
AllowOverride None
order allow,deny
allow from all
</Directory>
```

Y con esto ya puede funcionar el servidor web, el resto de la configuración lo dejo con las opciones (directivas) que por defecto a instalado apache, no obstante es importante revisar que el puerto predeterminado para la web es el 80 (Port 80).

El servidor web puede atender simultáneamente numerosas peticiones, pues dependiendo del número de peticiones de clientes que tenga el servidor será necesario modificar las directivas MinSpareServer y MaxSpareServers para que el rendimiento del equipo sea óptimo. Revisar también el número de servidores con los que se inicia el servidor: StartServers.

Terminada la configuración, con "httpd -t" se chequea la sintaxis de los ficheros de configuración.

Para iniciar httpd lee el punto 5 de este documento.

## Como publicar páginas web.

3.1.- Para poder publicar las páginas web en el servidor utilizando la opción de menú de Publicar de netscape u otros navegadores:

- Modificar httpd.conf

```
<Directory /home/httpd/html>
```

```
.....
```

```
Script PUT /cgi-bin/put.cgi
```

```
</Directory>
```

```
<Directory /home/httpd/cgi-bin>
```

```
AllowOverride None
```

```
Options ExecCGI
```

```
AuthType Basic
```

```
AuthName "Authorised PUT Publishers"
```

```
AuthUserFile /home/httpd/htpasswd-putusers
```

```
</Directory>
```

- Copiar el fichero put.cgi (ver <http://www.apacheweek.com/features/put>) en /home/httpd/cgi-bin/

- Para que un usuario pueda publicar hay que darlo de alta en el fichero de password /home/httpd/htpasswd-putusers con

```
# htpasswd htpasswd-putusers nombre_usuario
```

Este método presenta algunos problemas, que yo no he podido solucionar, en primer lugar los usuarios no tienen forma de borrar sus páginas web, y cuando un usuario publica una página, ésta queda con propietario nobody y permisos 644, lo cual puede causar algún que otro problema, ya que los directorios de usuarios deben tener permisos que permita a nobody escribir en ellos, y por tanto un usuario podría publicar páginas en el sitio web de otro usuario. Tendré que seguir estudiando el tema.

3.2.- La solución sería instalar un servidor ftp y los usuarios con un cliente ftp administrarían su sitio web. Para instalar un servidor ftp ver ftp-anonimo-como.

## Proxy.

También podemos hacer que el servidor apache haga de proxy con caché, para ello descomenta la directiva

```
ProxyRequests On
```

y si deseas una cache para acelerar los accesos descomenta las directivas:

```
CacheRoot /var/cache/httpd
```

```
CacheSize 5
```

```
CacheGcInterval 4
```

```
CacheMaxExpire 24
```

```
CacheLastModifiedFactor 0.1
```

```
CacheDefaultExpire 1
```

```
#NoCache a_domain.com another_domain.edu joes.garage_sale.com -- No deseo cache para los dominios especificados.
```

Si deseas sabes más cosas sobre proxys mira el documento ipchainsYsquid-micro-como.

## Script de inicio

La directriz ServerType puede tener dos valores, "standalone" si httpd se va a iniciar mediante un script de inicio o "inetd" si se va a iniciar con inetd, así que modifica httpd.conf para iniciar de un modo u otro. Parece ser que si el servidor web va a tener muchas peticiones mejor iniciarlo con standalone, pero para una red local es suficiente hacerlo con inetd y además puede ser más seguro.

En el primer caso incluir en /etc/rc.d/rcX.d el siguiente script de inicio, que en mi sistema tiene el nombre S85httpd, ejecutar con el parámetro "start", probarlo y suerte.

```
#!/bin/sh
```

```

#
# Startup script for the Apache Web Server
#
# chkconfig: 345 85 15
# description: Apache is a World Wide Web server. It is used to serve \
#             HTML files and CGI.
# processname: httpd
# pidfile: /var/run/httpd.pid
# config: /etc/httpd/conf/access.conf
# config: /etc/httpd/conf/httpd.conf
# config: /etc/httpd/conf/srm.conf
# Source function library.
./etc/rc.d/init.d/functions
# See how we were called.
case "$1" in
  start)
    echo -n "Starting httpd: "
    daemon httpd
    echo
    touch /var/lock/subsys/httpd
    ;;
  stop)
    echo -n "Shutting down http: "
    killproc httpd
    echo
    rm -f /var/lock/subsys/httpd
    rm -f /var/run/httpd.pid
    ;;
  status)
    status httpd
    ;;
  restart)
    $0 stop
    $0 start
    ;;
  reload)
    echo -n "Reloading httpd: "
    killproc httpd -HUP
    echo
    ;;
  *)
    echo "Usage: $0 {start/stop/restart/reload/status}"
    exit 1
esac
exit 0

```

Para iniciarlo con inetd, y aplicarle las reglas de seguridad de tcp\_wrappers (ver seguro-micro-como), añade en el fichero /etc/inetd.conf:

```
httpd stream tcp nowait root /usr/sbin/tcpd /usr/sbin/httpd -f /etc/httpd/conf/httpd.conf
```

comprueba que en /etc/services existe la entrada:

```
httpd 80/tcp httpd
```

y reiniciar inetd:

```
# killall -HUP inetd
```

**Más información** en el documento servidor-intranet-como de INSFLUG y en la documentación de apache.

nota de derechos en todas las copias. Todas las traducciones, trabajos derivados o adicionales que incorporen alguno de nuestros documentos o parte de su contenido deben ser cubiertos bajo esta nota de derechos y de cualquier trabajo derivado de éste documento no se pueden imponer restricciones a su distribución gratuita