

Instalar Squid, el proxy caché de Linux

En este manual intentamos enseñar cómo configurar Squid en modo consola. Se presume que el usuario tiene cierta formación en Linux, básica. De todas maneras, iré explicando paso a paso todo lo que se haga.

Información de Squid

Squid es un proxy caché. Eso quiere decir que funciona como intermediario en tus conexiones a Internet, con un caché que guarda las páginas para visualizarlas más rápido. El tamaño del caché y demás es configurable por el usuario. Squid, como buen programa de Linux, es completamente libre y disponible a través de Internet.

Advertencia: Squid no funciona para protocolos de correo, POP3 y SMTP. Para ese propósito necesitas algo como Dante.

Instalación de Squid

Lo haremos por consola. Abrimos un terminal y ponemos:

```
sudo apt-get squid
```

Se instalará y configurará. A veces, el programa da un error de host, que aparece con el nombre de `VISIBLE_HOST` (fallo en visible host). Simplemente ocurre que Squid no es capaz de determinar el host de la máquina. La solución es poner en la línea del `squid.conf` (el archivo donde se guarda la configuración de Squid) `visible_hostname` el nombre de tu máquina. Acto seguido Squid ya arrancará con una configuración básica.

Configuración de Squid

La configuración básica de Squid es muy simple y no es funcional. Explicamos una configuración que sirva para controlar el acceso en una red local a una serie de direcciones IP (páginas de Internet). Vamos a ojear rápidamente los parámetros que yo recomiendo configurar para empezar a comprender Squid.

```
http_port
cache_mem
ftp_user
cache_dir
ACL
http_access
httpd_accel_host
httpd_accel_port
httpd_accel_with_proxy
```

A continuación explicamos cada uno de ellos:

http_port:

Es el puerto que Squid utilizará para funcionar. O sea: cuando configuremos el ordenador A para acceder a Internet por el proxy, necesitamos configurar ese puerto en el menú del navegador. Veremos mas adelante como se hace esta sencilla operación. Por defecto, Squid funciona en el puerto 3128. Puedes configurar también otros puertos para funcionar. Como ejemplo, quedaría algo como ésto:

```
# # You may specify multiple socket addresses on multiple lines. # #
Default: http_port 3128 http_port 3128 :::::>>>> Puerto por defecto
http_port 8080 :::::>>>> Puerto que nosotros hemos elegido para
funcionar.
```

cache_mem:

Este parámetro no determina la cantidad máxima de memoria del proceso Squid. Limita la cantidad máxima de memoria que Squid utilizará para almacenar objetos.

`In-Transit objects Hot Objects Negative-Cached objects`

Por defecto está configurada en 8 MB. Yo la he puesto en 16, como parámetro recomendado, si tienes un servidor con al menos 128 megas de RAM. Ésto da idea de lo poquito que Squid necesita para funcionar. Si tienes en la oficina algún ordenador viejo que ya no tenga uso, siempre podrá reutilizarse como proxy. No voy a aburrir con detalles técnicos. Si alguien desea mas información puede acceder a la web y buscar toda la información que sea necesaria.

ACL:

Una ACL es ni más ni menos que una lista de control de acceso, creo que la principal característica de Squid. Con ésta opción determinamos quien accederá a Internet a través del proxy y quien no. Esto significa que por lo menos hemos de decidir poner una lista, si queremos que funcione. Por ejemplo, decidamos que los ordenadores 192.168.1.6 y 192.1.168.1.7 accedan a través del proxy. La sintaxis de la instrucción para definir una lista acl es:

`acl [nombre de la lista] src [lo que compone a la lista] o [ruta].`

Si queremos definir toda una red local, bastaría con hacer:

`acl milistapermitidos 192.168.1.0/255.255.255.0`

Llamamos a nuestra lista milistapermitidos y la situamos en /etc/squid/milistapermitidos Ponemos:

`acl milistapermitidos src /etc/Squid/milistapermitidos`

Previamente hemos creado una lista en nuestro disco duro, en la ubicación indicada, donde estarían las direcciones 192.168.1.6 y 192.168.1.7, en nuestro ejemplo. Con esto le estaríamos indicando al servidor proxy que esas direcciones podrán acceder al Squid. Luego, mediante control podremos decidir a que páginas se conectará cada una.

ftp_user:

Determina el usuario que accederá al ftp de forma anónima. Por defecto, Squid envía como contraseña @Squid. La puedes cambiar por la que quieras.

`ftp_user proxy@su-dominio.net sería un ejemplo.`

cache_dir:

Determina el tamaño de la caché en el disco duro que utilizará Squid para almacenar información. La cantidad se determina en función de lo que el usuario necesite, siempre con un poco de sentido común, por supuesto. Si se especifica un tamaño mas grande que el disponible, Squid se bloqueará. La sintaxis es:

`cache_dir ufs /var/spool/Squid [tamaño] 16 256`

Por defecto, Squid utiliza 100 megas. Claro está que se puede usar mucho más en cualquier ordenador un poco moderno.

http_acces:

Este parámetro determina las reglas de control de acceso de nuestra red. En el ejemplo, las direcciones 192.168.1.6 y 192.168.1.7 podrán acceder según se

determine por el parámetro `http_acces`. La sintaxis es

```
http_acces allow o http_acces deny según se quiera permitir o denegar
acceso respectivamente.
```

Las dos palabras clave, ACL y `http_acces` permiten mucho juego combinándolas entre si. Podemos, por ejemplo, decidir que la ip 192.168.1.6 acceda a páginas para profesores y la 192.168.1.7 a páginas para alumnos, por ejemplo. ¿Cómo se hace?: Para explicarlo definamos primero la instrucción `dst_domain`. La susodicha define un dominio de Internet. O sea: `dst_domain .google.es` permitiría acceder a google. Hagamos lo siguiente: definimos una lista `acl`:

```
acl red1 dst_domain .gmail.com
acl red1 dst_domain .paginasamarillas.es
acl red2 dst_domain .paginabonita.es
acl red2 dst_domain .adslzone.net (cómo no).
```

Y ahora, decidimos qué ip accederá a qué páginas.

```
http_access allow milistapermitidos red1
http_access allow milistapermitidos2 red2
```

O sea: la ip 192.168.1.6 estará en `milistapermitidos` y la ip 192.168.1.7 en la `milistapermitidos2`. La primera accederá a gmail y páginas amarillas, y la segunda a `paginabonita` y `adslzone`. También se podría hacer:

```
http_access allow !milistapermitidos, lo que significaría que se dejaría acceder a
lo contrario que estuviera en la lista. O sea: todas las direcciones menos la
192.1.168.6. Repito: daos cuenta de la potencia de Squid para definir y depurar
el control de la red hacia Internet. Con lo explicado hasta ahora, podemos darle
a Squid un buen uso, aprovechando las características de listas de acceso y
reglas de control. Ahora, aprovechemos las de rapidez. Squid como acelerador. Si
un usuario accede a una página en Internet, y ésta no ha sufrido cambios desde
ese acceso, Squid mostrará el que está guardado en la caché, en lugar de volver a
descargarlo. Eso quiere decir que si siempre se accede a las mismas páginas se
gana velocidad. Para aprovechar esta característica, es necesario habilitarla
dentro del Squid. Se busca la línea:
```

```
HTTPD-ACCELERATOR OPTIONS
```

y se escribe:

```
httpd_accel_host virtual
httpd_accel_port 0
Httpd_accel_with_proxy on
```

Esos parámetros serán necesarios si no utilizamos proxy transparente. En ese caso, que yo no necesité, se utiliza:

```
httpd_accel_host virtual
httpd_accel_port 80
httpd_accel_with_proxy on
httpd_accel_uses_host_header on
```

Estos parámetro están sacados de Internet. No están comprobados por mí, así que no puedo asegurar el correcto funcionamiento.

DEFINICIÓN DE REGLAS LÉXICAS: PATRONES:

Imaginemos que nuestra red es un poco mas complicada que la que hemos propuesto de ejemplo. Si queremos dejar acceso libre a todo Internet excepto las páginas pornográficas, por ejemplo (tristemente es algo que se ve mucho en las empresas). Sería una auténtica tarea agotadora buscar páginas pornográficas en Internet e incluirlas en nuestra lista. Además, siempre se encontraría una no definida y con

acceso. Para evitar eso, definimos patrones regulares que permitan contener el acceso. Un patrón regular es una serie de reglas que definen una palabra. Por ejemplo: si queremos definir todas las palabras que tengan en medio la palabra "sexo", caso que nos ocupa, se podría hacer: * sexo *. Existen muchos patrones regulares y formas de expresarlos. En nuestro caso, configurarlos para Squid, se hace así, con la ayuda de la instrucción: **url_regex** Ésta instrucción indica las palabras que denegaremos (o permitiremos según el caso). Definimos una lista de control con los patrones que identificaremos: `patronesdenegados`, situada en `etc/squid/patronesdenegados`. Con la siguiente información:

```
sexo
porn
massexo
masporn
porno
```

Paso a paso:

```
acl patronesdenegados url_regex src "/etc/Squid/patronesdenegados"
```

y ahora, la regla de control de acceso:

```
http_acces deny patronesdenegados
```

y listos. Como advertencia, es necesario que esta regla de control aparezca antes que cualquier otra que permita o niegue acceso a otras listas. Si fuera al revés el filtro no sería útil. De nuevo, imaginemos que queremos hacerlo al revés, permitir sólo un patrón regular. Lo hacemos así: Creamos la lista `patrones_no_negados`, con el contenido: `infosexo :::>>página necesaria y no de sexo`. `acl patrones_no_negados url_regex src "/etc/squid/patrones_no_negados"` y la regla de control de acceso:

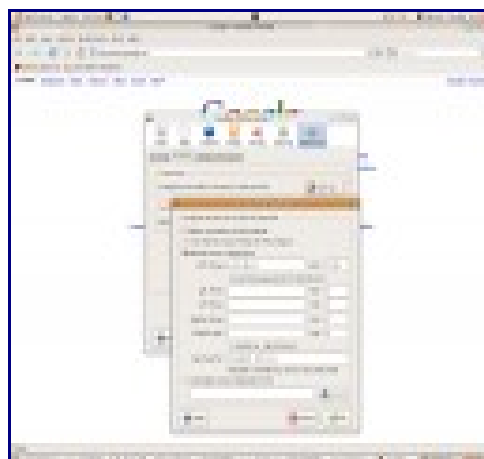
```
http_acces deny patronesdenegados !patrones_no_negados
```

Como antes, el signo de admiración indica que denegaremos el acceso a todo lo que no esté en esa lista. O sea: denegar acceso a `patronesdenegados` y denegar acceso a cualquier cosa que se salga de la lista de `patrones_no_negados`. En el ejemplo, negaríamos acceso a cualquier página que contuviera la palabra `sexo` pero no a `infosexo`.

CONFIGURACIÓN DE NAVEGADORES PARA USAR Squid:

Si no utilizamos un proxy transparente, será necesario configurar los clientes para que naveguen a través del proxy. Es una tarea rápida y fácil de llevar a cabo. Mozilla firefox:

Una pantalla para que quede mas claro:



Es muy simple: decidimos cual será la ip de la máquina donde estará funcionando Squid, y el puerto en el que funciona. Para la captura, está configurado en el puerto 3128 y en la dirección ip 192.168.1.1 Claro está, esto es prácticamente configurable por el administrador de la red.

CONCLUSIONES:

¿Sencillo, verdad? . El Squid tiene mucha más miga, mucha tela que cortar y muchas configuraciones posibles. El proxy transparente es una de las opciones útiles y más utilizadas. No obstante, recordar que Squid no funciona como proxy para correo, POP3 y SMTP. Usar Dante para esa tarea.