

Planeta Linux Argentina

Acelerando la navegacion en Internet usando Proxies

por [Walter Omar](#)

Cuando se navega por Internet una de las principales quejas de los navegantes es la lentitud en la carga de algunas páginas, la que depende del momento del día en que nos conectemos a Internet (por el tráfico), de la calidad de nuestro proveedor de acceso a Internet y de qué tan pesada es la página o sitio en cuestión (cantidad de gráficos, applets, etc.) ¿Qué podemos hacer al respecto?

Una primera solución: Configurar el Navegador

En un principio, las empresas desarrolladoras de navegadores se aproximaron a la solución de este problema utilizando el concepto de caché o depósito de archivos en los que se guardan todos los elementos que constituyen cada página web que se visita. De esta manera, la próxima vez que se visite la misma página el navegador revisará si en su área de caché se encuentra guardada la página que se visita y si así es se compara la fecha de la página guardada con la de la existente en el sitio a visitar. Si las fechas coinciden el navegador carga directamente los elementos residentes en la caché y sólo recupera de Internet los elementos que no pueden almacenarse en esa área (como aquellas propagandas o banners que cambian constantemente mientras se visualiza una página).

Esta solución trabaja bastante bien siempre que se configure adecuadamente el navegador y se establezca una caché de un tamaño respetable.

En el caso del navegador más popular para Linux (Netscape Navigator) la configuración se realiza mediante el menú Edit -> Preferences -> Advanced -> Cache. Una de las opciones a configurar es la rotulada Disk Cache que determina la cantidad de espacio en disco que se destinará al área de caché.

Como es de suponer, una caché de 5 ó más megabytes permitirá almacenar una buena cantidad de páginas con sus elementos. Pero si acostumbramos a visitar páginas con abundantes imágenes de gran tamaño puede que esos 5 MB resulten insuficientes. Para tener una idea, una imagen de 800x600 pixels en formato jpg puede alcanzar a tener unos 300 KB y a veces más. Con diez de esas imágenes que se almacenen en la caché ya habremos hecho uso de un 60% de ella. Esto nos lleva a la necesidad de disponer de un espacio de más de 5 MB, digamos 20 ó 30 MB.

El segundo ítem a modificar en la configuración del navegador es la dada por las opciones rotuladas Document in cache is compared to document on network (Comparar los documentos de la caché con los existentes en la red). De entre las tres disponibles elegimos a Every time (siempre).

Estas consideraciones anteriores también puede aplicarse a otros navegadores como Mozilla, Galeon o Konqueror.

Desventajas de esta primera solución

Aunque la solución anterior funciona bastante bien para la mayoría de los usuarios, adolece de algunos inconvenientes: existe una caché para cada usuario de Linux que utilice el Netscape Navigator, haciendo que se destinen 20 ó 30 MB por cada usuario y, tal vez, para almacenar las mismas páginas. Ello se puede comprobar mirando los directorios /home/usuario/.netscape/cache.

Si disponemos de una red de computadoras hogareña en la que por algún medio se comparte el acceso a Internet, existirá una zona de caché para cada usuario que acceda a Internet, entendiéndose por usuario también a los perfiles con que cuenta el Netscape Navigator en su versión para Windows y a los perfiles del Microsoft Internet Explorer, haciendo que se empleen unos 20 ó 30 MB para las cachés individuales de esos usuarios.

Una segunda solución: Un Proxy con caché propio

Ya sea que contemos con una red hogareña con acceso compartido a Internet o que sólo queramos mejorar la navegación en nuestra máquina con Linux, podemos recurrir a algún programa que preste los servicios de Proxy y de caché. La combinación de ambos permite que un mismo caché se encuentre disponible para cuantos navegadores se utilicen en la computadora y, a la vez, para otros navegadores en otras máquinas de una red permitiendo el acceso compartido a Internet.

Existen varios programas que prestan estas funciones. En forma personal trabajé con Squid y con wwwoffle. Ambos son altamente configurables y prestan básicamente los mismos servicios, como se muestra en la tabla siguiente:

Característica

- ¿Elimina la necesidad de cachés individuales por usuario y por programas?
Squid: Sí
wwwoffle: Sí
- ¿Se lo puede utilizar para compartir el acceso a Internet para una red al mismo tiempo que se comparte su caché (servidor proxy)?
Squid: Sí
wwwoffle: Sí
- ¿Fácil configuración. Se pueden utilizar programas de terceros, algunos de ellos con interfaz gráfica?
Squid: Sí
wwwoffle: Sí
- ¿Poseen opciones para configurar aspectos de seguridad como listas de acceso (usuarios y máquinas permitidos), nombres virtuales, etc.?
Squid: Sí
wwwoffle: Sí
- ¿Se puede monitorear el funcionamiento utilizando un navegador?
Squid: Sí
wwwoffle: Sí
- ¿Funciona como acelerador del sitio local?
Squid: Sí
wwwoffle: No
- ¿Permite la navegación off-line (desconectado de Internet)?
Squid: No
wwwoffle: Sí
- ¿Permite la navegación off-line estando conectado a Internet?
Squid: No
wwwoffle: Sí
- ¿Puede limitar la navegación a las páginas guardadas aún estando conectado a Internet?
Squid: No
wwwoffle: Sí
- ¿Puede permitir navegar libremente a algunos usuarios y a otros limitarlos a las páginas guardadas en la zona de caché?
Squid: No
wwwoffle: Sí

Como puede apreciarse, ambos programas presentan importantes características. La decisión por uno o por otro depende exclusivamente de nuestros fines y las necesidades.

Recurrir a programas de Proxy constituye una posibilidad cierta de acelerar la navegación y de ahorrar espacio en disco de quienes navegamos en Internet, ya sea en un uso personal o en una red.

Consiguiendo Squid y wwwoffle

Como siempre, hay varias maneras de conseguir programas para Linux. Si se dispone de una distribución de Linux más o menos moderna (Red Hat, SuSE, etc.) es posible que se encuentren en el CD destinado a los programas de terceros (o contribuciones). Otra posibilidad es buscarlos en los sitios de software para Linux, como Linuxberg.com, Freshmeat.net, etc.

En el caso de Squid, la forma directa será visitando el sitio de sus desarrolladores: www.squid-cache.org, en donde encontraremos también documentación, respuestas a las preguntas frecuentes, etc. Pero tal vez sea importante incluir aquí el link a su última versión estable disponible al momento de la redacción de este documento:

<http://www.squid-cache.org/Versions/v2/2.5/>

que recomiendo que sea guardado en `/usr/local/src`, al menos provisoriamente, para lo que necesitaremos ser el usuario root.

Con respecto a wwwoffle, se lo puede bajar del siguiente enlace:

<http://www.gedanken.demon.co.uk/wwwoffle/>

Compilación e instalación

Terminada la descarga, debe procederse a descomprimir los fuentes para su posterior compilación. La descompresión puede efectuarse por varios medios pero aquí detallo los pasos necesarios para hacerlo desde la consola o una terminal:

```
root # cd /usr/local/src
root # tar xvfz squid-2.5.STABLE1.tar.gz
root # cd squid-2.5.STABLE1
```

Ahora sólo resta leer el archivo README para enterarnos de recomendaciones de los autores o simplemente ejecutar las siguiente órdenes para compilar e instalar Squid:

```
root # ./configure
root # make all
root # make install
```

Para el caso de wwwoffle, los pasos son similares:

```
root # cd /usr/local/src
root # tar xvfz wwwoffle-2.7d.tgz
root # cd wwwoffle-2.7d
```

El archivo INSTALL contiene las instrucciones de compilación, las que difieren ligeramente de las de Squid, al eliminarse el primer paso /configure

```
root # make all
root # make install
```

Squid: Configuración básica y arranque (¡quiero que ande ya!)

Una vez efectuada la instalación es necesario efectuar algunas tareas previas a la configuración. Para su funcionamiento Squid demanda la creación de una serie de directorios, típicamente en /usr/local/squid/cache que contendrán la llamada zona de caché . Para hacerlo debemos verificar que el dueño del directorio /usr/local/squid y todos los directorios por debajo de él sea el usuario root, para luego ejecutar la orden:

```
root # /usr/local/squid/bin/squid -z
```

Si da algún mensaje de error dejando sin crear el directorio /usr/local/squid/cache alegando que no tiene permisos, deberemos ejecutar

```
root # mkdir /usr/local/squid/cache
root # chmod -R 777 /usr/local/squid/cache y
root # /usr/local/squid/bin/squid -z
```

Hecho esto, ya estamos en condiciones de leer el archivo de configuración de Squid para efectuar algunos cambios mínimos. El archivo de configuración se llama squid.conf y se encuentra en /usr/local/squid/etc. Como se puede constatar tiene un mar de opciones de configuración, pero en esta etapa sólo modificaremos una o dos cosas, nada más. Por ejemplo, ubicamos la línea que dice:

```
#cache_dir ufs /usr/local/squid/cache 100 16 256
```

y que seguramente se encontrará comentada con el símbolo #. Esa línea especifica en qué directorio se creará la zona de caché y el tamaño máximo de la misma. Por omisión se asume un tamaño máximo de 100 MB, por lo que si se desea aumentar o reducir la caché deberá descomentarse la línea en cuestión y escribir el tamaño adecuado. Si, por ejemplo, se desea una caché de 120 MB, la línea debería lucir de esta manera:

```
cache_dir ufs /usr/local/squid/cache 120 16 256
```

Guardamos ahora los cambios realizados y ya se está en condiciones de arrancar a Squid. El paquete trae dos scripts que se encargan de hacerlo por nosotros y se encuentran ubicados en /usr/local/squid/bin. Para nuestro caso sencillo, basta con impartir la orden:

```
root # /usr/local/squid/bin/RunCache &
```

Hecho esto debemos consultar el archivo /usr/local/squid/squid.out para ver si el arranque fue normal o si se produjeron errores de alguna naturaleza. Si todo anduvo bien, sólo queda conectarnos a Internet y configurar el navegador para que utilice a Squid como proxy, lo que se detalla en Configuración de los clientes.

wwwoffle: Configuración básica y arranque (¡quiero que ande ya!)

wwwoffle requiere de menos pasos para ser configurado y no se me presentaron problemas durante la instalación, configuración y puesta en marcha. Algo para tener en cuenta es que wwwoffle utiliza una árbol de directorios basado en /var/spool/wwwoffle en donde creará su zona de caché y también alojará a su

archivo de configuración `/var/spool/wwwoffle/wwwoffle.conf`. También es ese árbol existen directorios que retienen una lista de las últimas páginas visitadas para su posterior revisión.

La configuración rápida de `wwwoffle` se hace editando el archivo de configuración mencionado `/var/spool/wwwoffle/wwwoffle.conf` y revisando el tamaño de caché asignado. Por omisión, `wwwoffle` no tiene límite por lo que la caché podrá crecer ilimitadamente. Para evitar eso ubicamos la sección `# Purge method and maximum ages specification` y en ella a la línea

```
max-size = 100
```

en donde colocamos el tamaño de caché máximo que deseamos. En el ejemplo es de 100 MB.

La puesta en marcha de `wwwoffle` se hace mediante la orden

```
root # /usr/local/sbin/wwwoffled -c /var/spool/wwwoffle/wwwoffle.conf
```

Si bien la configuración y la puesta en marcha no representan problema alguno hay algo para tener en cuenta: es necesario indicarle a `wwwoffle` cuándo se está conectado a Internet y cuándo no se está conectado. Por ello, cada vez que nos conectemos a Internet deberemos impartir la orden

```
root # /usr/local/bin/wwwoffle -online
```

y cada vez que nos desconectemos:

```
root # /usr/local/bin/wwwoffle -offline
```

Por otro lado, cuando no estemos conectados a Internet y nos encontremos navegando off-line aprovechando esa prestación de `wwwoffle`, seguramente intentaremos acceder a algún enlace o imagen que no se grabó en la caché. En ese momento `wwwoffle` mostrará un página diciendo que dicho enlace ha sido anotado para ser descargado de la red la próxima vez que nos conectemos a Internet. El proceso de bajar esos enlaces anotados NO es automático por lo que estando conectados a Internet deberemos dar la orden

```
root # /usr/local/bin/wwwoffle -fetch
```

si impartimos esa orden desde una terminal o xterminal podremos ver los enlaces que se van descargando de la red.

Configuración de los clientes

Para que los navegadores hagan uso de cualquiera de los dos proxies es necesario configurarlos de manera similar a los detallado en la primera sección de este artículo. Una primera solución: Configurar el navegador especificando un tamaño de caché chico y, en la misma sección `Advanced`, accediendo a `Proxies -> Manual proxy configuration -> View`.

Para `wwwoffle`:

```
Type: HTTP / Address of proxy server to use: 127.0.0.1 / Port: 8080
Type: FTP / Address of proxy server to use: 127.0.0.1 / Port: 8080
Type: Security / Address of proxy server to use: 127.0.0.1 / Port: 8080
```

Para `Squid`:

```
Type: HTTP / Address of proxy server to use: 127.0.0.1 / Port: 3128
Type: FTP / Address of proxy server to use: 127.0.0.1 / Port: 3128
Type: Security / Address of proxy server to use: 127.0.0.1 / Port: 3128
```

En cambio, para las máquinas de la red, la dirección `127.0.0.1` deber ser cambiada por la dirección de la placa de red de la máquina que está corriendo Linux con `Squid` o `wwwoffle`. Para redes locales, es típico que esa dirección sea algo así como `192.168.0.2`

Configuración avanzada

La configuración avanzada, que implica restricción de accesos, monitoreado de los proxies, etc., queda para ser descripta en un documento futuro.

Walter Omar
alias «runlevel4»
woautalan@planetlinux.com.ar

URL: <http://www.planetalinux.com.ar/article.php?aid=69>