Proxy transparente con Squid mini−HOWTO

Daniel Kiracofe
v1.0, 01 April 2000

Traducción:

Julián Espinosa de los Monteros, del Equipo AEBIUS, http://www.aebius.com

Este documento proporciona información sobre cómo configurar un servidor proxy HTTP de una manera transparente usando sólo Linux y squid.

Tabla de contenidos

1. Introducción

1.1 Comentarios
1.2 Copyright y Trademarks
1.3 #include <disclaimer.h>

2. Apreciación global de un proxy transparente

2.1 Motivación
2.2 Alcance de este documento

3. Configurando el Kernel

4. Preparando squid

5. Preparando ipchains

6. Juntándolo todo

7. Ir más allá de los Recursos

---

1. Introducción

1.1. Comentarios

Los comentarios y la mejora general de este mini HOWTO son bienvenidas y pueden dirigirse a su autor, Daniel Kiracofe, a drk@unxsoft.com.

1.2. Copyrights y Trademarks

Copyright 2000 by UnxSoft Ltd (www.unxsoft.com)

Este manual puede reproducirse en todo o en parte, sin retribuir, sujeto a las restricciones siguientes:

El aviso de copyright anterior y este aviso de permiso debe conservarse completo en todos las copias completas o parciales

Cualquier traducción o trabajo derivado debe ser aprobado por escrito por el autor antes de la distribución.

Si usted distribuye este trabajo en parte, deben ser incluidas instrucciones para obtener la versión completa de este manual, y los medios para proporcionar la obtención de una versión completa.

Pequeñas partes pueden reproducirse como ilustraciones para revisiones o citas en otros trabajos sin este aviso del permiso si se da la cita apropiada.

Pueden concederse excepciones a estas reglas para propósitos académicos:

Escriba al autor y pregunte. Estas restricciones están aquí para protegernos como autores, no para restringir a estudiantes y educadores. Cualquier código fuente (aparte del SGML en el que este documento está escrito) en este documento se pone bajo la GNU General Public License, disponible vía FTP anónimo del archivo GNU.

1.3. #include <disclaimer.h>

Ninguna garantía, expresa o implícita, etc, etc, etc...

2. Apreciación global de un proxy transparente

2.1. Motivación

En ˆ un proxy ordinario˜, el cliente especifica el hostname y número del puerto de un proxy en su software del navegador web. El navegador hace entonces las peticiones al proxy, y el proxy los remite a los servidores de origen. Esto es todo elegante y bueno, pero a veces una de las siguientes situaciones surge.

Que usted quiera obligar a los clientes en su red usar el proxy, quieran ellos o no.

Usted quiere que los clientes usen un proxy, pero no quiere que ellos sepan que lo están usando.

Usted quiere que los clientes usen el proxy, pero no quiere hacer todo el trabajo de actualizar los ambientes en centenares o miles de navegadores web.

Esto es donde el proxy transparente aparece. Una petición web puede interceptarse por el proxy, transparentemente. Es decir, hasta donde el software del cliente sabe, está comunicándose con el propio servidor de origen, cuando realmente lo hace con el servidor proxy.

Los routers Cisco soportan el proxy transparente. Pero, (muy sorprendentemente) Linux puede actuar como un router, y puede realizar el proxy transparente remitiendo las conexiones de TCP a los puertos locales. Sin embargo, nosotros también necesitamos hacer a nuestro web proxy consciente del efecto de la redirección, para que pueda hacer las conexiones a los servidores de origen apropiados. Hay generalmente dos formas de hacerlo:

El primero es cuando su web proxy no es consciente de serlo. Usted puede usar que un pequeño e ingenioso demonio llamado transproxy que se instala ante su web proxy y cuida de todos los detalles complicados por usted. transproxy ha sido escrito por John Saunders, y esta disponible en ftp://ftp.nlc.net.au/pub/linux/www/ o en su mirror local metalab. Transproxy no se avanzará en su discusión en este documento.

Una solución más limpia es conseguir un web proxy que sea consciente que es transparente. El que nosotros vamos a estudiar aquí es squid. Squid es Open Source que guarda un servidor proxy para los sistemas Unix. Está disponible en www.squid−cache.org

2.2. Alcance de este documento

Este documento tratará de la versión 2.3 de squid y el kernel de linux versión 2.2, las actualizaciones más estables en el momento de realizar este escrito (Marzo del 2.000). Creo que también funcionará con versiones de squid anteriores a la 2.0 y versiones del kernel de Linux anteriores a la 2.1. Si usted necesita información de versiones anteriores, puede buscar documentación en www.unxsoft.com.

Si usted quiere usar linux 2.3, usted tendrá que usar algo llamado netfilter en lugar del ipchains. Sin embargo, se asume que si usted está ejecutando un kernel de desarrollo, usted puede configurar netfilter solo de la documentación proporcionada. Si no, usted realmente no debe estar ejecutando un kernel de desarrollo (créame). Una vez linux 2.4 se lance, este documento se pondrá al día para cubrir el netfilter

.

3. Configurando el Kernel

Primero, necesitamos asegurarnos que todas las opciones apropiadas están configuradas en el

kernel. Si usted está usando un kernel normal de su distribución, el proxy transparente puede o no puede habilitarse (IIRC, está en RH 6.1, pero no me dice eso). Si usted no está, la mejor manera de saberlo es simplemente saltarse esta sección, y si las órdenes en la próxima sección le dan errores raros, probablemente es porque el kernel no fue configurado debidamente.

Si su kernel no se configura para el proxy transparente, usted necesitará recompilarlo. Recompilar un kernel es un proceso complejo (por lo menos al principio), y está más allá del alcance de este documento. Si usted necesita compilar un kernel de ayuda, por favor vea

<http://metalab.unc.edu/pub/Linux/do cs/HOWTO/Kernel−HOWTO˜>The Kernel HOWTO.

Las opciones que usted necesita poner en su configuración son las siguientes (Nota: nada de esto se puede crear como módulo)

Sysctl support

TCP/IP networking

IP: firewalling

IP: always defragment

IP: transparent proxy support

/proc filesystem support

Una vez usted tiene su nuevo kernel funcionando, usted puede necesitar habilitar el IP forwarding. IP forwarding permite su ordenador actuar como un router. Ya que esto no es lo que el usuario medio suele hacer, no está cargado por defecto y debe habilitarse explícitamente en run−time. No obstante, su distribución ya puede hacer esto para usted. Para verificarlo, teclee ˆ cat /proc/sys/net/ipv4/ip_forward ˆ. Si usted ve ˆ1ˆ está bien. Si no es así, teclee:

cat ‘1’ > /proc/sys/net/ipv4/ip_forward

Usted necesitará entonces agregar ese mandato a su apropiado script de inicio en /etc/rc.d /.

4. Preparando squid

Nosotros necesitamos conseguir squid cargado y funcionando. Descargue el último tarball fuente de www.squid−cache.org. Asegúrese que usted consigue una versión ESTABLE, no una versión de DESARROLLO. El último a la fecha de escribir este trabajo era squid−2.3.STABLE1.tar.gz.

Ahora, desempaquete y descomprima el archivo (use ˆ tar −xzf <filename> ˆ). Ejecute el script de autoconfiguración (ˆ./configure˜), compile (ˆmake˜) y después instale (ˆmake install˜).

Ahora, nosotros necesitamos revisar el fichero squid.conf predefinido (instalado en /usr/local/squid / etc/squid.conf, a menos que usted cambie los valores por defecto). El archivo squid.conf se comenta ampliamente. De hecho, alguna de la mejor documentación disponible para squid está en el archivo squid.conf. Después que usted consiga ponerlo todo en marcha, usted debe regresar y debe releerlo todo de nuevo. Pero para ahora, simplemente hagamos el mínimo requerido. Encuentre las siguientes directivas, no las comentamos, y los cambia a los valores apropiados:

httpd_accel_host virtual

httpd_accel_port 80

httpd_accel_with_proxy on httpd_accel_uses_host_header on

Finalmente mire la directiva http_access. Normalmente por defecto está ˆhttp_access deny all˜. Esto impedirá a cualquiera acceder a squid. Para ahora, usted puede cambiar esto a ˆ http_access allow all˜, pero una vez está funcionando, usted querrá probablemente leer las direcciones ACL (Access Control Lists), y configurar el caché tal que sólo personas en su red local (o cualquiera) pueden acceder al caché. Esto puede parecer tonto, pero usted debe poner alguna clase de restricciones en el acceso a su caché. Gente detrás de cortafuegos de filtrado (como los filtros del porno, o filtros en naciones dónde la expresión no es muy libre) a menudo hacenˆhijack˜ hacia proxies abiertos y pueden comer su ancho de banda.

Inicialice los directorios de caché con ˆ squid −z ˆ (sí esto no es una nueva instalación de squid, usted debe saltar este paso).

Ahora, ejecute squid que usando el script RunCache script en el directorio /usr/local/squid/bin /. Si funciona, usted debe poder poner las características de proxy de su navegador de web al IP del puerto 3128 (a menos que usted cambie el número del puerto predefinido) y acceso squid como un proxy normal.

Para ayuda adicional e configuración, vea el squid FAQ en www.squid−cache.org.

5. Preparando ipchains

deben instalarse ipchains con la mas reciente distribución (cualquiera basada en el kernel 2.2). Sin embargo, si usted no tiene ipchains, usted puede bajarlo de ftp://ftp.rustcorp.com/ipchains/. ipchains es una herramienta muy poderosa, y nosotros arañaremos sólo la superficie aquí. Para más información, por favor vea http://www.rustcorp.com/linux/ipchains/HOWTO.html para el ipchains HOWTO.

Para configurar las reglas, usted necesitará saber dos cosas, las direcciones IP del apartado (yo usaré 192.168.1.1 por ejemplo) y el puerto squid que está ejecutándose por delante (yo usaré el valor por defecto de 3128 por ejemplo).

Primero, nosotros necesitamos autorizar paquetes destinados a cualquier servidor web a través de este apartado. Nosotros debemos configurar la interface del loopback y la interface ethernet. Usted no debe saltar este paso aun cuando no tenga ningún servidor web real en su apartado, la ausencia de estas reglas puede crear infinitos intentos de envío porque el proxy intente conectar consigo mismo. Utilice los siguientes comandos:

ipchains −A input −p TCP −d 127.0.0.1/32 www −j ACCEPT

ipchains −A input −p TCP −d 192.168.1.1/32 www −j ACCEPT

Ahora, las palabras mágicas del proxy transparente:

ipchains −A input −p TCP −d any/0 www −j REDIRECT 3128

Usted querrá agregar los órdenes anteriores a su apropiado script de inicio en /etc/rc.d /.

6. Juntándolo todo

Si todo ha ido bien hasta ahora, vaya a otra máquina, cambie la entrada a la IP de su nuevo apartado de squid, y adelante. Para asegurarse que realmente están remitiéndose las peticiones a través de su proxy en lugar de un extraño servidor origen, verifique el archivo log /usr/local/squid/logs/acces s.log

7. Ir mas allá de los recursos

Si usted todavía necesita ayuda, usted puede desear revisar el squid FAQ o el squid mailing list en www.squid−cache.org. Usted también puede mandarme un correo electrónico a drk@unxsoft.com, y yo intentaré contestar sus preguntas si el tiempo me lo permite (a veces hace, pero a veces no hace)