

Análisis del Proxy-Cache y Reverse-Proxy

Alberto Pesquera Martín

Documentación en progreso

Historial de revisiones

Revisión 0.1

22-Feb-2003

Revisado por: apm

Creación

Tabla de contenidos

- [1. Introducción](#)
- [2. Tipos de Proxy](#)
 - [2.1. Reverse-Proxy \(Surrogate\)](#)
- [3. Cómo los Proxies Obtienen el Tráfico](#)
 - [3.1. Modificar el cliente](#)
 - [3.2. Modificar la red](#)
 - [3.3. Modificar el DNS](#)
 - [3.4. Modificar el servidor web](#)
- [4. Ventajas y Desventajas de un Sistema Proxy-Cache](#)
- [5. Notas Generales para saltarse un Proxy](#)

1. Introducción

Un servidor proxy es un tipo de buffer (memoria intermedia) entre tu ordenador y los recursos de Internet a los que estas accediendo. Éstos acumulan y guardan los archivos que mas amenudo son solicitados por los usuarios de Internet en una base de datos especial, llamada *cache*.

Por eso, los servidores proxy pueden incrementar la velocidad de tu conexión a Internet. La cache de un servidor proxy puede contener la información que tu necesitas cuando la solicitas, haciendo posible que el proxy te la sirva inmediatamente. Por lo que el incremento de rendimiento puede ser muy alto.

2. Tipos de Proxy

Filtro de contenidos: Se suele emplear en centros educativos para bloquear el acceso a contenido para adultos.

Control de Acceso a contenidos: Implementan una estrategia de control de acceso uniforme para un conjunto de servidores y recursos web, proporcionando un medio de auditoría. Es bastante usado en conjuntos de grandes corporaciones u otros centros burocraticos.

Cortafuegos de seguridad: Esta mejora de seguridad es restrictiva con los protocolos a nivel de aplicación en una organización con un flujo de entrada y salida, en un punto seguro de la red.

Web Cache: Mantienen copias locales de los archivos mas solicitados y los sirven bajo demanda, reduciendo la baja velocidad y coste en la comunicación con Internet.

Reverse-Proxy (Surrogate): Este tipo de proxy puede enmascarar a los servidores web. Reciben las peticiones a los servidores web reales, pero, a diferencia de los servidores web, pueden iniciar la comunicación con otros servidores para localizar el contenido solicitado bajo demanda.

Encaminador de contenido: Guian las peticiones a servidores web basados en las condiciones de tráfico de Internet y el tipo de contenido.

Transcoder: Pueden modificar el formato del cuerpo del contenido antes de enviarlo a los clientes. Traducciones de lenguajes, transformación de formatos de datos...

Anónimo: Proveen de un aumento en la privacidad y el anonimato, mediante una activa eliminación de características identificativas (dirección IP del cliente, cabeceras From y Referer, cookies, identificadores de sesion URI...).

2.1. Reverse-Proxy (Surrogate)

Se colocan al borde de la red, en frente de los servidores web, donde ellos puedan devolver todas las peticiones directamente al servidor web y preguntar a éste por los recursos cuando solamente sea necesario. Puede añadir características de seguridad a los servidores web o mejorar el rendimiento como una cache rápida del servidor web.

El *Reverse-Proxy* normalmente asume el nombre y la dirección IP del servidor web directamente, por eso todas las peticiones iran dirigidas al proxy en lugar de al servidor.

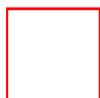
3. Cómo los Proxies Obtienen el Tráfico

Como normalmente los clientes se comunican directamente con los servidores web. Hay varias formas de especificar al tráfico HTTP la existencia de un proxy.

3.1. Modificar el cliente

La mayoría de los clientes soportan una configuración manual y automatica. Si el cliente está configurado para usar el servidor proxy, el cliente mandará las peticiones HTTP directamente e intencionadamente al proxy, en lugar de al servidor original.

Figura 1. Cliente configurado para usar proxy

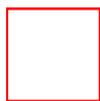


3.2. Modificar la red

Hay varias tecnicas donde la infraestructura de red intercepta y redirige el tráfico al proxy sin conocimiento o participación del cliente. Esta interceptción se basa en dispositivos de commutación (*switch*) o encaminamiento (*router*) que observan el tráfico HTTP, interceptandolo, y desviandolo al

proxy, sin conocimiento del cliente.

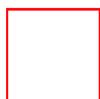
Figura 2. La red intercepta y redirecciona el tráfico al proxy



3.3. Modificar el DNS

Surrogates, que ubica el servidor proxy antes del servidor web, asumiendo el nombre y la dirección IP del servidor web directamente. Por eso, todas las peticiones llegan al proxy en lugar de al servidor. Esto puede ser organizado manualmente editando las tablas de nombres del DNS o usando un sistema especial de servidores DNS dinámicos que calculan el proxy apropiado o servidor de uso bajo demanda.

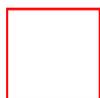
Figura 3. El proxy suplanta al servidor



3.4. Modificar el servidor web

Algunos servidores web pueden ser configurados para redireccionar las peticiones del cliente a un proxy mediante el comando de redirección (código de respuesta 305) se le devuelve al cliente. Y cuando el cliente recibe la redirección, éste se comunica con el proxy.

Figura 4. El servidor redirecciona las peticiones al proxy



4. Ventajas y Desventajas de un Sistema Proxy-Cache

Las ventajas de la tecnología caché son:

- Velocidad de navegación mayor para el usuario final
- Menor carga para el proveedor de contenidos
- Menor tráfico y congestión para Internet

Pero también tiene algunos problemas:

- Los contenidos pueden ser visualizados por más gente de la que podría parecer. Pudiéndose poner en peligro la privacidad y la seguridad de las personas.
- Cuando un usuario realiza una petición sobre un cierto contenido, éste puede no estar actualizado; recibiendo información falseada.

5. Notas Generales para saltarse un Proxy

Etiquetas HTML META. Probar las meta etiquetas `http-equiv="Last-Modified"`, `http-equiv="Expires" content="-1" o "Cache-control"`:

public: Se puede realizar el proceso de caché, aunque el objetivo normal fuese por regla general "no-cache" o para almacenar exclusivamente en cachés no compartidas.

private: El objeto o parte de sus cabeceras no se debe almacenar en una caché compartida (por ejemplo, no deben almacenarse las cookies, pero sí el objeto propiamente dicho).

no-cache: El objeto o parte de sus cabeceras no deben emplearse para responder peticiones subsiguientes, sin una revalidación completa.

no-store: Evita el almacenamiento del objeto en un disco.

no-transformation: No debe transformarse el objeto (por ejemplo, comprimirlo).

must-revalidate: La caché debe revalidar el objeto si no considera que sea "fresco", aunque se haya configurado para trabajar con objetos expirados. Es decir, si el objeto se considera expirado en función de "Expires" o "max-age", la caché debe revalidarlo, sin hacer un cículo "implícito" de frescura.

proxy-revalidate: Actua de la misma forma que "must-revalidate" para cachés compartidas.

max-age: Similar a la cabecera "Expires"

s-maxage: Actua de la misma forma que "max-age" para cachés compartidas.



Los sistemas HTTP/1.0 no disponen de la directiva "Cache-control" por lo que deben usar: "Pragma: no-cache". De forma que la caché deba recargar el objeto del servidor original.

Configuración del Servidor Web Apache.

```
ExpiresActive On
ExpiresDefault "modification plus 1 day"
<Files ^*\.(php|xml)>
Header append Cache-Control
"no cache"
</Files>
```

Configuración del Servidor Web AOLserver. AOLserver se puede adaptar a través de los módulos Tcl. Un módulo Tcl puede ser usado para crear las cabeceras HTTP por defecto para un sitio web. El webmaster especifica un directorio donde residen los módulos Tcl, y cuando arranca el servidor web, los módulos ubicados en ese directorio también son inicializados. Por defecto este directorio se crea en `/servers/servername/modules/tcl`. Para crear por defecto una cabecera HTTP P3P, se crea un archivo en el directorio de módulos Tcl. La extensión del archivo debe ser `.tcl`. Y en el archivo se debe incluir la siguiente línea:

```
# Cabecera P3P
ns_set put [ns_conn outputheaders] "P3P"
"policyref="http://www.mydomain.com/path/to/p3p.xml", CP="NON CURa TIA"
```

```
# Cabecera Normal HTTP
```

```
ns_ser put [ns_conn outputheaders] "Cache-control" "private"
```

```
ns_ser put [ns_conn outputheaders] "pragma" "no-cache"
```

Modificación del código de todas las páginas. Indicar que todo el contenido sea llamado como un componente dinámico. Incluyendo una "?" tras las extensiones de cada archivo:

- Enlaces: ``

- Imágenes: `<a img src="directorios/imagen.png?">`

También se puede seguir este método para archivos de descarga y scripts.