

LPIC-2 / Examen 208 - Servicios Web

208.3 Implementar un proxy de caché

Teoría

Un proxy de caché es un servidor que se sitúa entre los clientes (navegadores) y los servidores web. Cuando un cliente solicita un recurso (una página web, una imagen), la solicitud va primero al proxy. Si el proxy tiene una copia reciente de ese recurso en su caché, la sirve directamente al cliente (un "hit"). Si no la tiene (un "miss"), el proxy solicita el recurso al servidor web original, se lo entrega al cliente y guarda una copia en su caché para futuras solicitudes.

Beneficios de un Proxy de Caché:

- **Mejora del Rendimiento:** Las solicitudes satisfechas desde la caché son mucho más rápidas que ir al servidor original.
- **Ahorro de Ancho de Banda:** Reduce el tráfico hacia internet (si es un proxy de salida) o hacia los servidores web internos (si es un proxy inverso).
- **Control de Acceso y Filtrado:** Permite implementar políticas sobre a qué sitios web se puede acceder, filtrar contenido o autenticar usuarios.
- **Monitorización y Registro Centralizados:** Todos los accesos web pasan por el proxy y se registran en un solo lugar.
- **Seguridad:** Puede ofrecer cierta protección al enmascarar las IPs de los clientes internos.

Squid:

Squid es el software de proxy de caché más popular y ampliamente utilizado en Linux. Puede funcionar como proxy de salida (para redes internas que acceden a internet) o como proxy inverso (delante de servidores web para cachear contenido saliente). LPIC-2 se enfoca típicamente en Squid como proxy de salida básico.

Implementación Básica de Squid:

1. Instalación del Software:

- **Paquete:** squid (estándar en ambas ramas Debian/Red Hat).
- **Comando:** `sudo apt install squid` o `sudo dnf install squid`.

2. Gestión del Servicio:

- **Nombre del Servicio:** squid.service (estándar en ambas ramas).
- **Comandos Systemd:** `sudo systemctl enable squid`, `sudo systemctl start squid`, `sudo systemctl status squid`, `sudo systemctl restart squid`, `sudo systemctl stop squid`, `sudo systemctl reload squid` (para recargar configuración).

3. Archivo de Configuración:

- **Ubicación:** /etc/squid/squid.conf (estándar en ambas ramas). Este archivo es muy extenso y documentado con comentarios.

- **Estructura:** Contiene directivas de configuración global y, fundamentalmente, **Access Control Lists (ACLs)** y reglas `http_access` para controlar el acceso.

4. Directivas de Configuración Clave en `squid.conf`:

- `http_port <puerto> [opciones]`: Define el puerto(s) en el que Squid escuchará las solicitudes HTTP de los clientes. El puerto por defecto es 3128. Puede especificar IPs (`http_port 192.168.1.1:3128`). La opción `transparent` (o `tproxy`) se usa para proxies transparentes (requiere configuración de firewall/enrutamiento para redirigir tráfico al puerto de Squid).
- `cache_dir <tipo> <directorio> <tamaño_en_MB> <niveles_directorios_hash>`: Configura dónde se almacenará la caché en disco. Puedes tener varias líneas `cache_dir`.
 - `<tipo>`: `ufs` (filesystem universal), `aufs` (ufs asíncrono, mejor rendimiento), `diskd` (asíncrono, usa un demonio). `aufs` o `diskd` son comunes en producción.
 - `<directorio>`: Ruta al directorio de caché (ej: `/var/spool/squid/cache`).
 - `<tamaño_en_MB>`: Tamaño máximo de la caché en disco.
 - `<niveles_directorios_hash>`: Dos números que definen la estructura de subdirectorios para distribuir los objetos cacheado (ej: `cache_dir aufs /var/spool/squid 10000 16 256`).
- `cache_mem <tamaño_en_MB>`: Cantidad de RAM que Squid usa para cachear objetos pequeños, metadatos y buffers de red.
- `access_log <ruta_archivo> [formato]`: Ubicación del archivo de log de acceso. El formato por defecto es `squid`, que es legible.
- `http_access <allow/deny> <ACL_name> [<ACL_name>...]`: Define reglas de acceso. Se procesan en orden secuencial. La primera regla que coincide (`allow` o `deny`) determina el acceso. Si ninguna regla coincide, se aplica la acción por defecto (normalmente `http_access deny all`).
- `acl <ACL_name> <type> <data>`: Define una Access Control List.
 - `src <ip>/< mascara>`: Coincide con la IP o red de origen del cliente (ej: `acl localnet src 192.168.1.0/24`).
 - `dstdomain <dominio> [<dominio>...] /dstdom_regex <exp_regular>`: Coincide con el dominio de destino.
 - `url_regex <exp_regular>`: Coincide con la URL completa o parte de ella.
 - `port <puerto> [<puerto>...]`: Coincide con el puerto de destino.
 - `time <días> <horas>`: Coincide con el momento del día o día de la semana.
 - Otras ACLs: `browser`, `referer`, `user`, `proxy_auth`, etc.

- `visible_hostname <hostname>`: Establece el nombre que Squid reportará (útil si no tiene un FQDN resoluble).
5. **Inicialización de Directorios de Caché:** Antes de iniciar Squid por primera vez o después de modificar `cache_dir`, los directorios de caché deben ser creados con la estructura de directorios adecuada. El script de inicio de `systemd` a menudo lo hace automáticamente, pero puedes hacerlo manualmente: `sudo squid -z`.
 6. **Firewall:** Asegúrate de que el firewall permite el tráfico TCP entrante al puerto configurado en `http_port` (por defecto 3128) desde las IPs o redes de tus clientes.

Gestión de Caché y Configuración:

- `sudo systemctl reload squid` o `sudo squid -k reconfigure`: Recarga la configuración sin detener el servicio (si es posible).
- `sudo squid -k parse`: Verifica la sintaxis del archivo `squid.conf` sin aplicar los cambios. Crucial antes de recargar/reiniciar.
- `squidclient`: Utilidad para interactuar con el demonio Squid (ej: `squidclient mgr:info` para información del servidor, `squidclient -p <puerto> -h <host>` para usar el cliente en un puerto/host específico).
- Limpiar la caché: A menudo se hace eliminando el contenido del directorio `cache_dir` después de detener Squid y reiniciando el servicio (riesgoso y borra toda la caché).

Configuración del Cliente:

Los clientes (navegadores, aplicaciones) deben configurarse para usar el proxy en la IP y puerto donde escucha Squid (ej: `192.168.1.100:3128`). Para proxies transparentes, la configuración se hace a nivel de red/firewall sin necesidad de configurar cada cliente.

Diferencias Debian vs. Red Hat (Squid):

- El paquete `squid` es estándar.
- El nombre del servicio `squid` es estándar.
- La ubicación del archivo principal `/etc/squid/squid.conf` es estándar.
- El contenido por defecto de `/etc/squid/squid.conf` y la organización de archivos incluidos o ejemplos de ACLs pueden variar ligeramente.
- El directorio `cache_dir` por defecto es `/var/spool/squid/` en ambas, pero los permisos y el tipo de `cache_dir` predeterminado pueden diferir.