📁 LPIC-2 / 🌍 Examen 208 - Servicios Web

*Nota: Estos ejercicios implican instalar software y modificar archivos de configuración de red. Realízalos **SIEMPRE en una VM de prueba dedicada**. Asegúrate de que tu VM tiene acceso a internet para la instalación de paquetes. La configuración del firewall es crucial.*

Ejercicio 8.3.1: Instalando Squid y Gestionando el Servicio

- **Objetivo:** Instalar el software proxy y asegurarse de que el servicio base funciona.
- **Requisitos:** Acceso a la línea de comandos. Privilegios de superusuario (`sudo`). Conexión a internet.
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Instala Squid:** `sudo apt update && sudo apt install squid` (Debian/Ubuntu) o `sudo dnf install squid` (Red Hat/CentOS/Fedora).
    3. **Habilita e inicia el servicio:** `sudo systemctl enable squid && sudo systemctl start squid`.
    4. **Verifica el estado:** `systemctl status squid`. Debería estar `active (running)`.

Ejercicio 8.3.2: Verificando Reglas de Firewall para el Puerto de Squid

- **Objetivo:** Asegurarse de que los clientes pueden conectar al puerto donde escucha Squid.
- **Requisitos:** Privilegios de superusuario (`sudo`). Identificar la herramienta de firewall activa (Ej. 5.2.5). Saber el puerto por defecto de Squid (3128 TCP).
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Si usas `firewalld`:** Ejecuta `sudo firewall-cmd --zone=<zona> --list-ports`. Busca el puerto `3128/tcp`. Si no está, añádelo: `sudo firewall-cmd --zone=<zona> --add-port=3128/tcp --permanent` y `sudo firewall-cmd --reload`.
    3. **Si usas `ufw`:** Ejecuta `sudo ufw status`. Busca reglas para el puerto 3128 TCP. Si no están, añádelas: `sudo ufw allow 3128/tcp`.
    4. **Si usas `iptables` directamente:** Ejecuta `sudo iptables -L -v -n`. Busca reglas que permitan tráfico entrante a puerto 3128 TCP en la cadena `INPUT`.

Ejercicio 8.3.3: Localizando y Explorando el Archivo de Configuración Principal de Squid

- **Objetivo:** Encontrar y entender la estructura y directivas clave en `squid.conf`.
- **Requisitos:** Squid instalado. Privilegios de superusuario (`sudo`).
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Localiza el archivo:** `/etc/squid/squid.conf`.
    3. **Visualiza el contenido:** Ejecuta `sudo less /etc/squid/squid.conf`. Este archivo está muy comentado.

4. **Busca las directivas clave:**
    - `http_port`: Dónde escucha Squid (debería ser 3128 por defecto).
    - `cache_dir`: Dónde almacena la caché en disco (busca `/var/spool/squid`).
    - `cache_mem`: Uso de RAM para caché.
    - `access_log`: Ubicación del log de acceso.
    - `acl`: Definiciones de ACLs.
    - `http_access`: Reglas de acceso basadas en ACLs. Observa que las últimas líneas suelen ser `http_access allow localhost` seguido de `http_access deny all`.
    - `visible_hostname`: El nombre que reporta Squid.

Ejercicio 8.3.4: (Conceptual) Configurando Acceso Básico con ACLs

- **Objetivo:** Entender cómo permitir acceso al proxy solo a una red específica.
- *Requisitos:* Squid instalado. Privilegios de superusuario (`sudo`). Conocer la IP o red de tus clientes (ej: 192.168.1.0/24).
- **Desarrollo Paso a Paso (Conceptual):**
    1. Abre una terminal.
    2. **Edita el archivo de configuración de Squid:** `sudo vi /etc/squid/squid.conf`.
    3. **Busca la sección de ACLs y añade una para tu red (si no existe una adecuada):**

```
# Definir una ACL para mi red local 192.168.1.0/24
acl localnet src 192.168.1.0/24
```

    4. **Busca la sección `http_access` y ajusta las reglas:** Las reglas se procesan en orden. Asegúrate de que tu regla `allow` para la ACL de tu red esté *antes* de la regla `deny all`.

```
# Insertar regla para permitir a localnet ANTES de deny all
http_access allow localnet

# Reglas por defecto para localhost (suele venir)
# http_access allow localhost

# Regla final para denegar a todos los demas
http_access deny all
```

    5. **Guarda y sal.**
    6. **Verifica la sintaxis:** `sudo squid -k parse`.
    7. **Recarga la configuración:** `sudo systemctl reload squid` o `sudo squid -k reconfigure`.

Ejercicio 8.3.5: (Conceptual) Inicializando Directorios de Caché y Probando Acceso

- **Objetivo:** Asegurarse de que la caché está lista y probar que un cliente puede usar el proxy.

- *Requisitos:* Squid instalado. Configuración básica aplicada. Privilegios de superusuario (`sudo`). **VM de prueba con un cliente (navegador o `curl`).**
- **Desarrollo Paso a Paso (Conceptual):**
    1. Abre una terminal en el servidor Squid.
    2. **Inicializa los directorios de caché (si no se hizo automáticamente):** `sudo squid -z`. Puede que el servicio de systemd lo haga al iniciar si no existen. Verifica el directorio `cache_dir` (`/var/spool/squid/`) después de iniciar Squid; si contiene subdirectorios (`00`, `01`, etc.), la inicialización se realizó.
    3. **En el cliente (otra VM o tu propia máquina):**
        - **Configura el navegador o `curl` para usar el proxy:** En la configuración de red del navegador, especifica la IP del servidor Squid y el puerto 3128 (o el que hayas configurado en `http_port`).
        - **Prueba accediendo a una página web:** Navega a un sitio web.
        - **Usa `curl` con proxy:** `curl --proxy http://<IP_servidor_squid>:3128 http://example.com`.
    4. **En el servidor Squid, verifica los logs de acceso:** `sudo tail <ruta_log_acceso>`. Deberías ver una entrada registrando la solicitud del cliente a través del proxy. Si la solicitud fue servida desde la caché, puede que veas un estado como `TCP_HIT` o `TCP_MISS`.