

Práctica

Proxy Squid

Franco Larrea

2º SMR-A

(Prof. Fernando Albert González)

Instituto IES SAN VICENTE

Índice

| | |
|----------------------------|-----------|
| Teoría..... | Pag. 3 |
| Tareas y cuestiones..... | Pag. 4-18 |
| Problemas encontrados..... | Pag. 19 |
| Fuentes..... | Pag. 19 |
| Alumnos participantes..... | Pag. 19 |

Teoría

Para realizar esta practica es conveniente saber algunos conceptos:

Proxy

Un proxy es un servidor, programa o dispositivo, que hace de intermediario en las peticiones de recursos que realiza un cliente a otro servidor. Esta situación estratégica de punto intermedio le permite ofrecer diversas funcionalidades: control de acceso, registro del tráfico, restricción a determinados tipos de tráfico, mejora de rendimiento, anonimato de la comunicación, caché web, etc. Dependiendo del contexto, la intermediación que realiza el proxy puede ser considerada por los usuarios, administradores o proveedores como legítima o delictiva y su uso es frecuentemente discutido.

Squid

Squid es un servidor proxy para web con caché. Es una de las aplicaciones más populares y de referencia para esta función, es un desarrollo en software libre publicado bajo licencia GPL. Entre sus utilidades está la de mejorar el rendimiento de las conexiones de empresas y particulares a Internet guardando en caché peticiones recurrentes a servidores web y DNS, acelerar el acceso a un servidor web determinado o añadir seguridad realizando filtrados de tráfico.

Tareas y cuestiones

1. Crea un fichero con un grupo de ordenadores a los cuales no se les permita el acceso a Internet.
2. El resto de ordenadores podrá conectarse a Internet.
3. Comprueba que los puntos 1 y 2 funcionan. Puedes ir cambiando la IP de la máquina de Windows para hacer las pruebas. No olvides tomar capturas de pantalla.
4. Crea otro grupo de ordenadores a los cuales se les prohíba el acceso a Internet de 3 a 9 de la tarde los Lunes, miércoles y viernes.
5. Prueba con la máquina de Windows cambiando de nuevo la IP que este punto también funciona.
6. Crea otro grupo de equipos a los cuales no se les permita conectar con sitios web cuya URL contengan las palabras “sex” y “porn”.
7. Vuelve a comprobar que este punto funciona también.

DEFINICIONES


```
acl grupoNOinternet src "grupo1.txt" > 192.168.6.30-192.168.6.39
acl grupo2 src "grupo2.txt" > 192.168.6.40-192.168.6.49
acl horario time MWF 15:00-21:00
acl grupo3 src "grupo3.txt" > 192.168.6.50-192.168.6.59
acl palabras url_regex sex porn
```

REGLAS


1. http_access deny grupoNOinternet
2. http_access deny grupo2 horario
3. http_access deny grupo3 palabras
4. http_access allow

Esta práctica se ha realizado en dos maquinas virtuales con Ubuntu 18.04.3 LTS

Estas son las características de la maquina servidor:

| | |
|--|---|
| General | Previsualización |
| Nombre: Ubuntu servidor SQUID Sistema operativo: Ubuntu (64-bit) Grupos: 2SMRA |  |
| Sistema | |
| Memoria base: 2048 MB Procesadores: 2 Orden de arranque: Disquete, Óptica, Disco duro Aceleración: VT-x/AMD-V, Paginación anidada, Paravirtualización KVM | |
| Pantalla | |
| Memoria de vídeo: 128 MB Servidor de escritorio remoto: Inhabilitado Captura de vídeo: Inhabilitado | |
| Almacenamiento | |
| Controlador: IDE IDE secundario maestro: [Unidad óptica] Vacío Controlador: SATA Puerto SATA 0: Ubuntu servidor SQUID-disk1.vdi (Normal, 10,00 GB) | |
| Audio | |
| Controlador de anfitrión: PulseAudio Controlador: ICH AC97 | |
| Red | |
| Adaptador 1: Intel PRO/1000 MT Desktop (Adaptador puente, «en0») Adaptador 2: Intel PRO/1000 MT Desktop (Red NAT, «FMOLnetwork») | |
| USB | |
| Controlador USB: OHCI, EHCI Filtros de dispositivos: 0 (0 activo) | |

Y estas son las características de la maquina cliente:

| | |
|--|---|
| General | Previsualización |
| Nombre: Ubuntu Seguridad xd Sistema operativo: Ubuntu (64-bit) Grupos: 2SMRA |  |
| Sistema | |
| Memoria base: 2048 MB Procesadores: 2 Orden de arranque: Disquete, Óptica, Disco duro Aceleración: VT-x/AMD-V, Paginación anidada, Paravirtualización KVM | |
| Pantalla | |
| Memoria de vídeo: 128 MB Servidor de escritorio remoto: Inhabilitado Captura de vídeo: Inhabilitado | |
| Almacenamiento | |
| Controlador: IDE IDE secundario maestro: [Unidad óptica] Vacío Controlador: SATA Puerto SATA 0: Ubuntu Server.vdi (Normal, 10,00 GB) | |
| Audio | |
| Controlador de anfitrión: PulseAudio Controlador: ICH AC97 | |
| Red | |
| Adaptador 1: Intel PRO/1000 MT Desktop (Red NAT, «FMOLnetwork») | |
| USB | |
| Controlador USB: OHCI, EHCI Filtros de dispositivos: 0 (0 activo) | |

El servidor tendrá dos adaptadores de red, un adaptador en modo puente, con el que se conectará a internet y un adaptador de red NAT conectado a “FMOLnetwork” para tener conectividad con el equipo cliente.

El equipo cliente solo tendrá un adaptador de red NAT conectado también a “FMOLnetwork”.

Vamos a configurar las interfaces de red. Para ello editaremos el fichero `/etc/netplan/50-cloud-init.yaml`.

Editamos este fichero ya que no queremos que se cambie la IP al reiniciar la maquina.

La primera interfaz (`enp0s3`) tendrá una IP dinámica, mientras que la segunda interfaz tendrá la IP estática `192.168.6.1`.

```
GNU nano 2.9.3 50-cloud-init.yaml
# This file is generated from information provided by
# the datasource.  Changes to it will not persist across an instance.
# To disable cloud-init's network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      addresses: [192.168.6.1/24, ]
      gateway4: 192.168.0.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4, 192.169.0.1]

[ 16 líneas leídas ]
^G Ver ayuda  ^O Guardar    ^W Buscar     ^K Cortar Text ^J Justificar
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar txt   ^T Ortografía
```

Para que estos cambios tengan efecto ejecutaremos el comando `sudo netplan apply`.

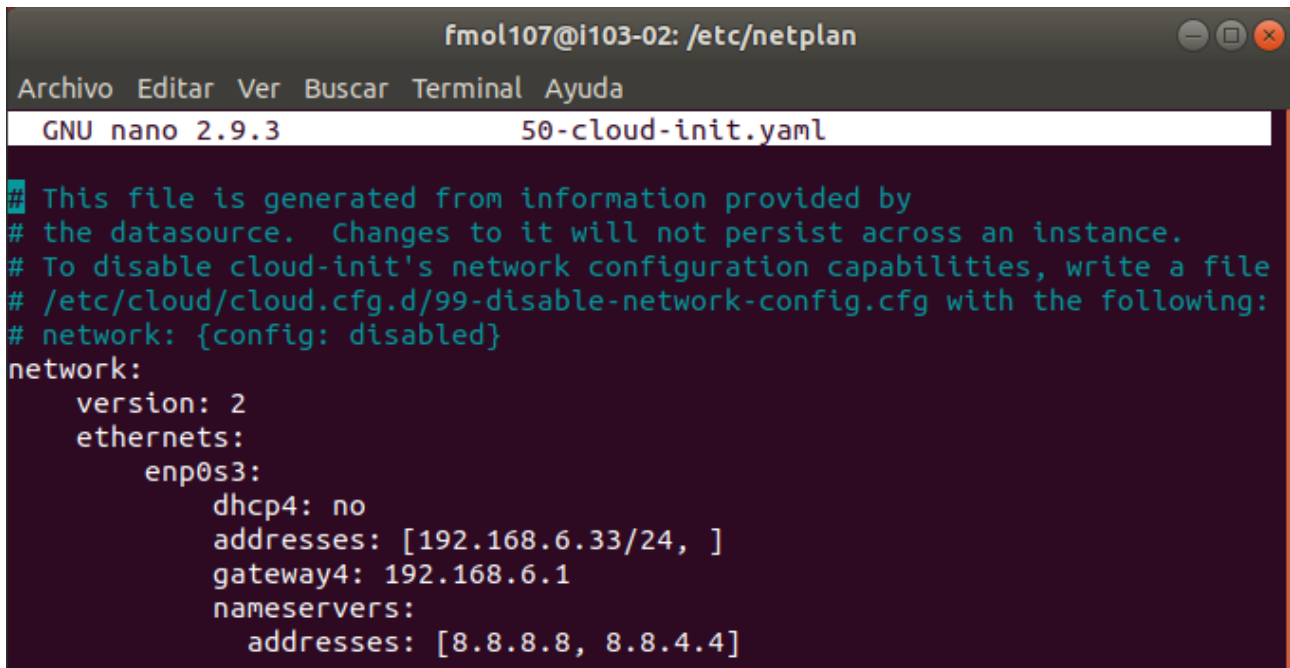
```
fmol107@i103-02:/etc/netplan$ sudo netplan apply
```

Tras ejecutar este comando comprobamos que se han efectuado los cambios con el comando `ip address`.

```
fmol107@i103-02:/etc/netplan$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP group default qlen 1000
    link/ether 08:00:27:86:41:bb brd ff:ff:ff:ff:ff:ff
    inet 192.168.43.177/24 brd 192.168.43.255 scope global dynamic enp0s3
        valid_lft 3561sec preferred_lft 3561sec
    inet6 fe80::a00:27ff:fe86:41bb/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP group default qlen 1000
    link/ether 08:00:27:3c:5e:99 brd ff:ff:ff:ff:ff:ff
    inet 192.168.6.1/24 brd 192.168.6.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe3c:5e99/64 scope link
        valid_lft forever preferred_lft forever
fmol107@i103-02:/etc/netplan$
```

De la misma forma vamos a configurar la interfaz del equipo cliente. En este caso le asignaremos la IP 192.168.6.33.

Cada vez que cambie la IP del equipo cliente lo haré de esta forma.



```
fmol107@i103-02: /etc/netplan
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 50-cloud-init.yaml

# This file is generated from information provided by
# the datasource.  Changes to it will not persist across an instance.
# To disable cloud-init's network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  ethernets:
    enp0s3:
      dhcp4: no
      addresses: [192.168.6.33/24, ]
      gateway4: 192.168.6.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
```

Para que estos cambios tengan efecto ejecutaremos el comando **sudo netplan apply**.

```
fmol107@i103-02:/etc/netplan$ sudo netplan apply
```

Tras ejecutar este comando comprobamos que se han efectuado los cambios con el comando **ip address**.

Antes de continuar vamos a comprobar con un **ping** que hay conectividad entre la maquina cliente y la maquina servidor.

```
fmol107@i103-02:/etc/netplan$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP group default qlen 1000
    link/ether 08:00:27:d5:71:ed brd ff:ff:ff:ff:ff:ff
    inet 192.168.6.33/24 brd 192.168.6.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fed5:71ed/64 scope link
        valid_lft forever preferred_lft forever
fmol107@i103-02:/etc/netplan$ ping -c 2 192.168.6.1
PING 192.168.6.1 (192.168.6.1) 56(84) bytes of data.
64 bytes from 192.168.6.1: icmp_seq=1 ttl=64 time=0.271 ms
64 bytes from 192.168.6.1: icmp_seq=2 ttl=64 time=0.265 ms

--- 192.168.6.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1012ms
rtt min/avg/max/mdev = 0.265/0.268/0.271/0.003 ms
fmol107@i103-02:/etc/netplan$
```

A continuación vamos a instalar Squid en la maquina servidor.
Con el siguiente comando instalamos squid:

```
fmo107@i103-02:~$ sudo apt-get install squid
```

Al finalizar la instalación comprobaremos el estado de squid con el comando **sudo systemctl status squid**.

```
fmo107@i103-02:~$ sudo systemctl status squid
● squid.service - LSB: Squid HTTP Proxy version 3.x
   Loaded: loaded (/etc/init.d/squid; generated)
   Active: active (running) since Mon 2020-02-10 12:30:59 CET; 1min 5s ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 4 (limit: 2313)
   CGroup: /system.slice/squid.service
           └─19634 /usr/sbin/squid -YC -f /etc/squid/squid.conf
             └─19636 (squid-1) -YC -f /etc/squid/squid.conf
               └─19637 (logfile-daemon) /var/log/squid/access.log
                 └─19638 (pinger)

feb 10 12:30:59 i103-02 systemd[1]: Starting LSB: Squid HTTP Proxy version 3.x..
feb 10 12:30:59 i103-02 squid[19578]: * Starting Squid HTTP Proxy squid
feb 10 12:30:59 i103-02 squid[19634]: Squid Parent: will start 1 kids
feb 10 12:30:59 i103-02 squid[19578]: ...done.
feb 10 12:30:59 i103-02 systemd[1]: Started LSB: Squid HTTP Proxy version 3.x.
feb 10 12:30:59 i103-02 squid[19634]: Squid Parent: (squid-1) process 19636 star
lines 1-17/17 (END)
```

Podemos observar que el servicio esta activo.

Ahora vamos a configurar en la maquina cliente el proxy de la red.

Esta configuración la podemos encontrar en *Configuración > Red > Proxy de la red*

Seleccionaremos la casilla “Manual”. Como solo vamos a utilizar servicios web para esta práctica solo configuraré el proxy para HTTP y HTTPS.

Como IP pondré en ambos casos la IP del adaptador de red NAT del servidor(192.168.6.1). Para el puerto pondré el que emplea Squid(3128).



Una vez configurado esto, el proxy ya estaría funcionando. Podemos comprobarlo intentando acceder a alguna pagina web desde el navegador.

Por defecto, la configuración de squid deniega todo, así que no deberíamos poder acceder a ninguna pagina web.



Ahora en el servidor, en la ruta donde están los ficheros de configuración de squid `/etc/squid/` haremos una copia de seguridad del archivo de configuración que tenemos que editar.

Esto lo haremos con el siguiente comando:

```
fmol107@i103-02:/etc/squid$ sudo cp ./squid.conf ./copiasquid.conf
```

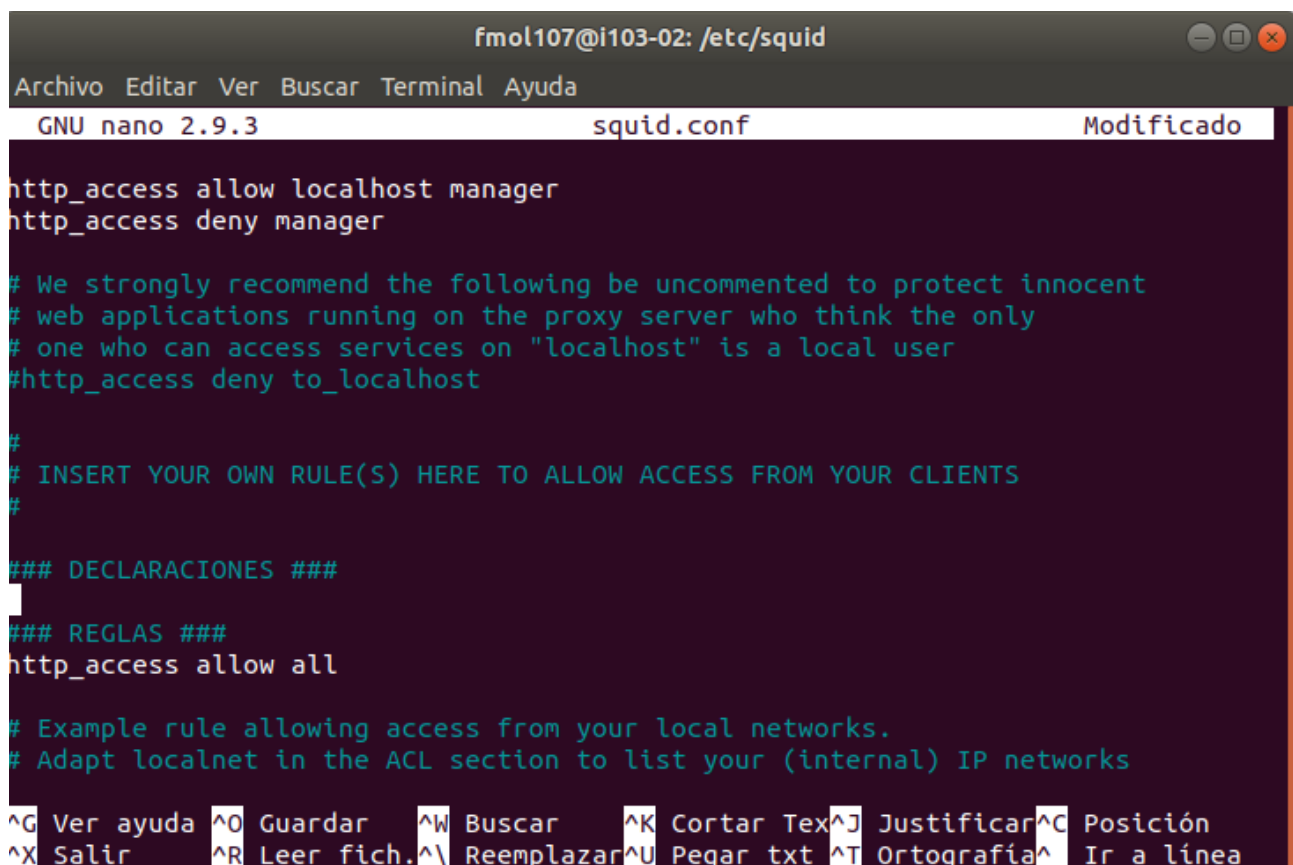
Una vez hecha una copia del archivo por si ocurriera alguna incidencia, procedemos a editar el mismo.

En el archivo `squid.conf` podremos definir acl y establecer reglas.

Ahora solo vamos a establecer una regla para comprobar que todo funciona correctamente.

- `httpaccess allow all`

Con esta regla permitimos todo el acceso.



```
fmol107@i103-02: /etc/squid
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 squid.conf Modificado

http_access allow localhost manager
http_access deny manager

# We strongly recommend the following be uncommented to protect innocent
# web applications running on the proxy server who think the only
# one who can access services on "localhost" is a local user
#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

### DECLARACIONES ###

### REGLAS ###
http_access allow all

# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks

^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Tex ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía ^_ Ir a línea
```

Guardamos los cambios y comprobamos si hemos cometido algún error de sintaxis con el siguiente comando:

```
fmol107@i103-02:/etc/squid$ sudo squid -k reconfigure
```

Para aplicar los cambios reiniciamos el servicio squid.

```
fmol107@i103-02:/etc/squid$ sudo systemctl restart squid
```

Ahora en el cliente deberíamos poder acceder a paginas web.

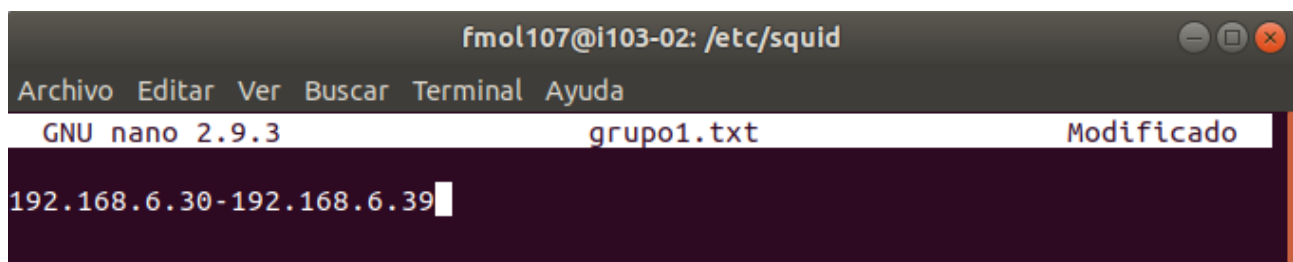


Con esto hemos comprobado que el proxy funciona correctamente. Ahora podemos empezar con las tareas.

- Crea un fichero con un grupo de ordenadores a los cuales no se les permita el acceso a Internet.
- El resto de ordenadores podrá conectarse a Internet.
- Comprueba que los puntos 1 y 2 funcionan. Puedes ir cambiando la IP de la máquina de Windows para hacer las pruebas. No olvides tomar capturas de pantalla.

Siempre que modifiquemos configuración del proxy será desde el servidor y en la carpeta */etc/squid/*

Crearemos un archivo llamado “grupo1.txt”. Dentro de este estará el siguiente rango de IPs: 192.168.6.30-192.168.6.39 .



Ahora editaremos el fichero *squid.conf*. Tendremos que añadir una ACL y una regla.

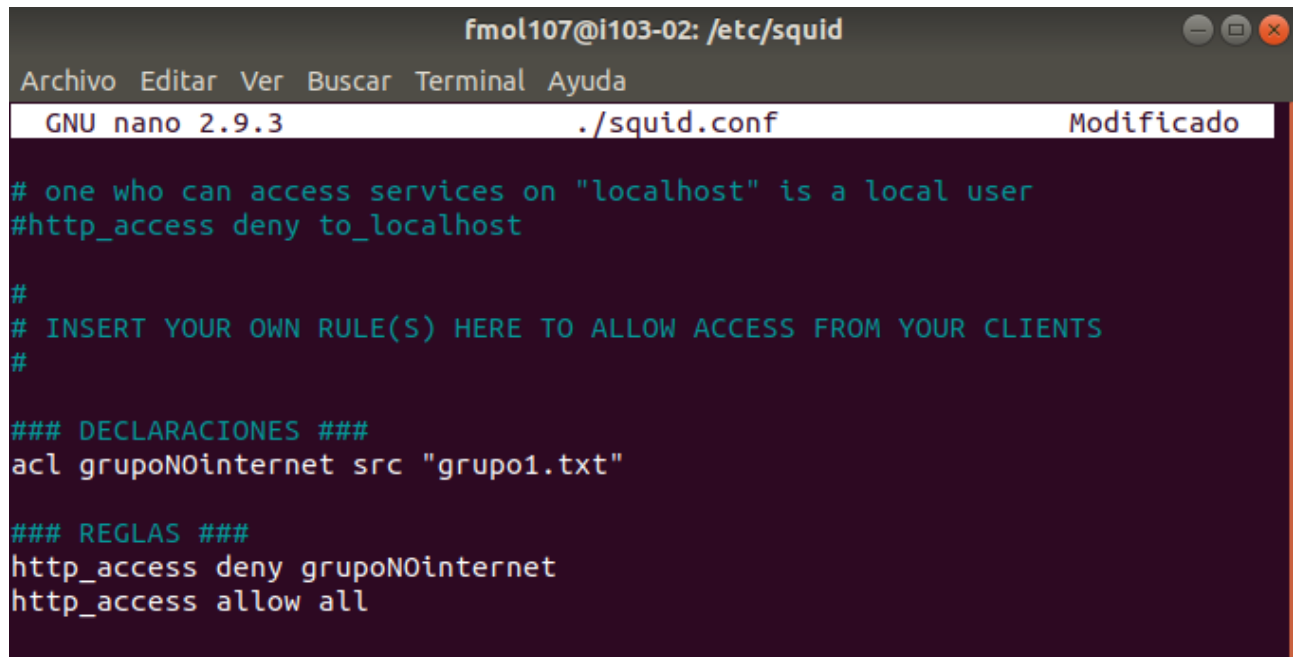
Definiremos la siguiente ACL:

- `acl grupoNOinternet src “grupo1.txt”`

Y la siguiente regla:

- `http_access deny grupoNOinternet`

Con esto denegamos el acceso a internet a un grupo de ordenadores y se lo permitiremos al resto.



```
fmol107@i103-02: /etc/squid
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 ./squid.conf Modificado

# one who can access services on "localhost" is a local user
#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

### DECLARACIONES ###
acl grupoNOinternet src "grupo1.txt"

### REGLAS ###
http_access deny grupoNOinternet
http_access allow all
```

Guardamos los cambios y comprobamos si hemos cometido algún error de sintaxis con el siguiente comando:

```
fmol107@i103-02:/etc/squid$ sudo squid -k reconfigure
```

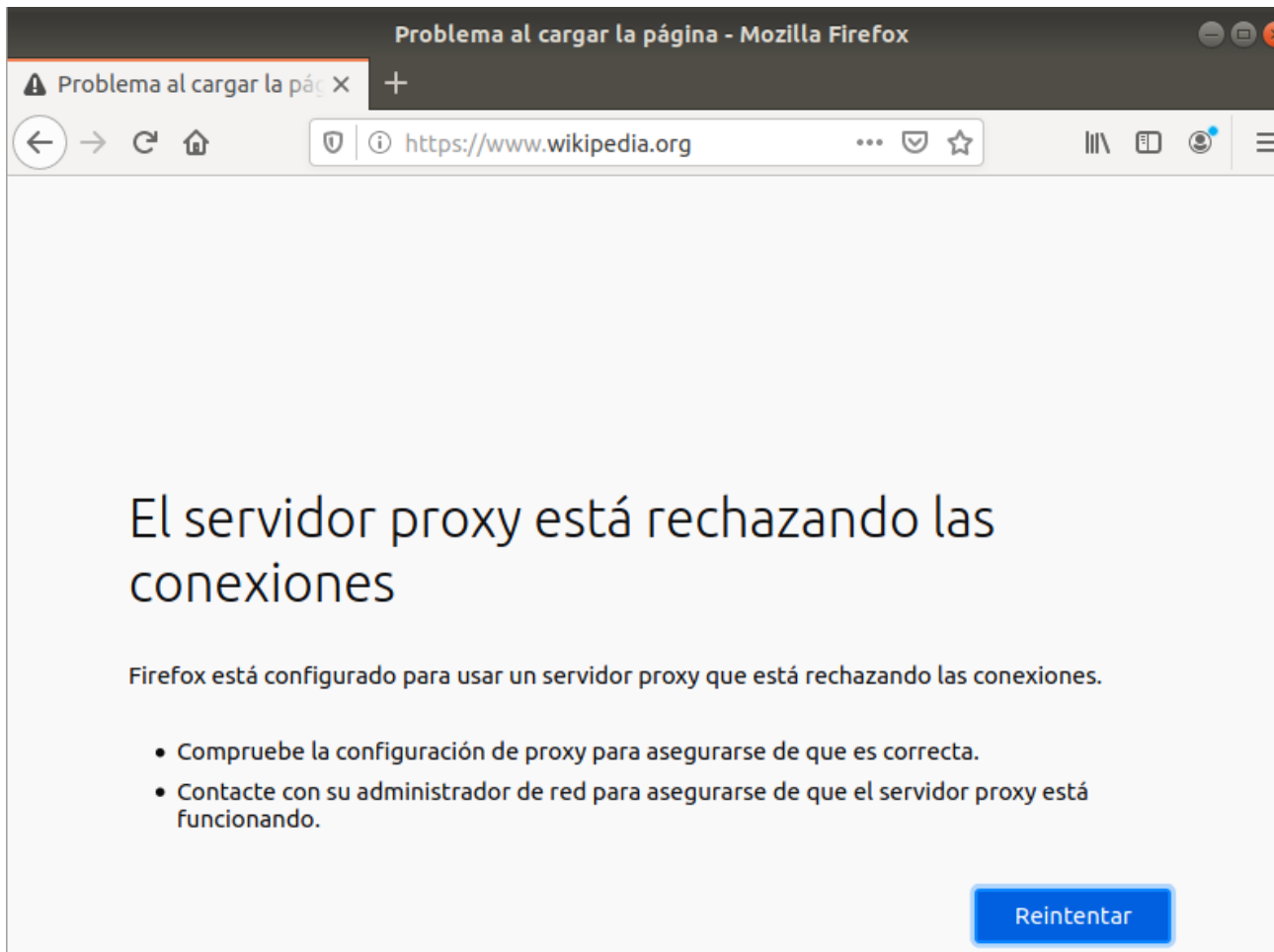
Para aplicar los cambios reiniciamos el servicio squid.

```
fmol107@i103-02:/etc/squid$ sudo systemctl restart squid
```

Ahora comprobaremos las nuevas reglas en el cliente. Podemos ver que la IP actual del cliente esta dentro del rango de los equipos que tienen prohibido el acceso a internet.

```
fmol107@i103-02:/etc/netplan$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP group default qlen 1000
    link/ether 08:00:27:d5:71:ed brd ff:ff:ff:ff:ff:ff
    inet 192.168.6.33/24 brd 192.168.6.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fed5:71ed/64 scope link
        valid_lft forever preferred_lft forever
```

Si lo comprobamos, no tenemos acceso a internet.



Ahora cambiaremos la IP del cliente por una que no este en el rango de los equipos que tienen prohibido el acceso a internet.

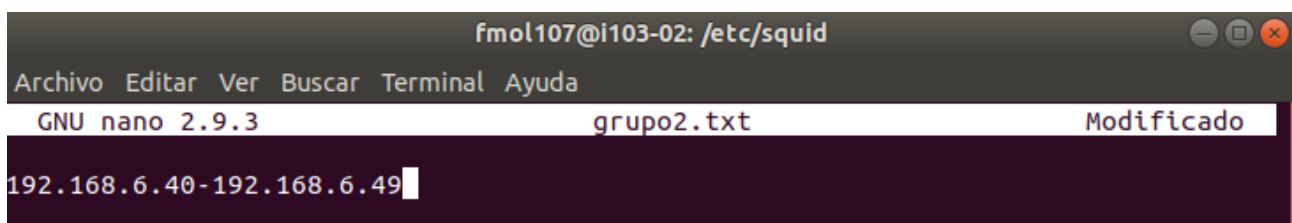
```
fmol107@i103-02:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:f8:dd:06 brd ff:ff:ff:ff:ff:ff
    inet 192.168.6.20/24 brd 192.168.6.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fef8:dd06/64 scope link
        valid_lft forever preferred_lft forever
```

Si lo comprobamos, si tenemos acceso a internet.



- Crea otro grupo de ordenadores a los cuales se les prohíba el acceso a Internet de 3 a 9 de la tarde los Lunes, miércoles y viernes.
- Prueba con la máquina de Windows cambiando de nuevo la IP que este punto también funciona.

Crearemos un archivo llamado “grupo2.txt”. Dentro de este estará el siguiente rango de IPs: 192.168.6.40-192.168.6.49 .



Ahora editaremos el fichero *squid.conf*. Tendremos que añadir dos ACL y una regla.

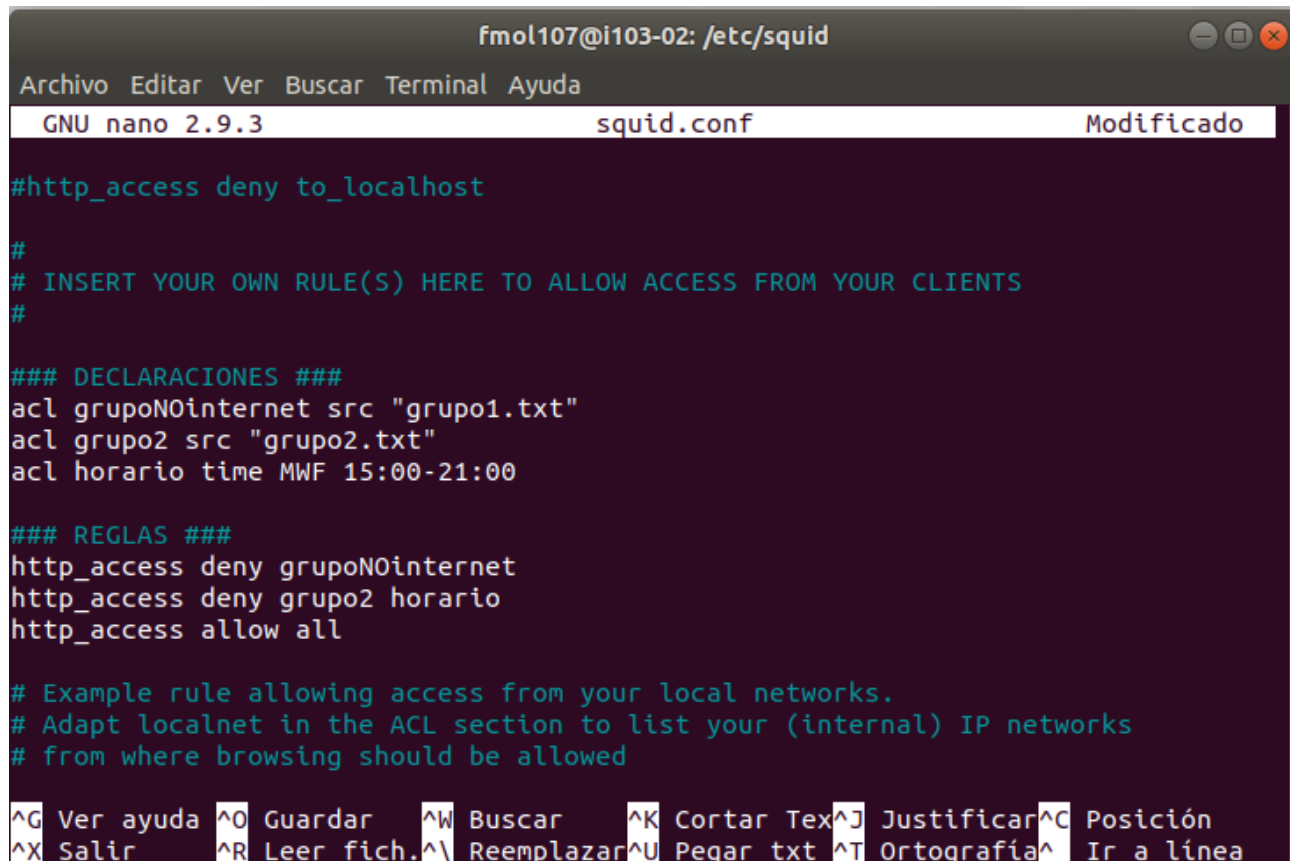
Definiremos la siguiente ACL:

- `acl grupo2 src “grupo2.txt”`
- `acl horario time MWF 15:00-21:00`

Y la siguiente regla:

- `http_access deny grupo2 horario`

Con esto denegamos el acceso a internet a un grupo de ordenadores en un horario definido, pero le permitimos el acceso fuera de ese horario.



```
fmol107@i103-02: /etc/squid
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 squid.conf Modificado

#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

### DECLARACIONES ###
acl grupoN0internet src "grupo1.txt"
acl grupo2 src "grupo2.txt"
acl horario time MWF 15:00-21:00

### REGLAS ###
http_access deny grupoN0internet
http_access deny grupo2 horario
http_access allow all

# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed

^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Tex ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^T Ortografía ^_ Ir a línea
```

Guardamos los cambios y comprobamos si hemos cometido algún error de sintaxis con el siguiente comando:

```
fmol107@i103-02:/etc/squid$ sudo squid -k reconfigure
```

Para aplicar los cambios reiniciamos el servicio squid.

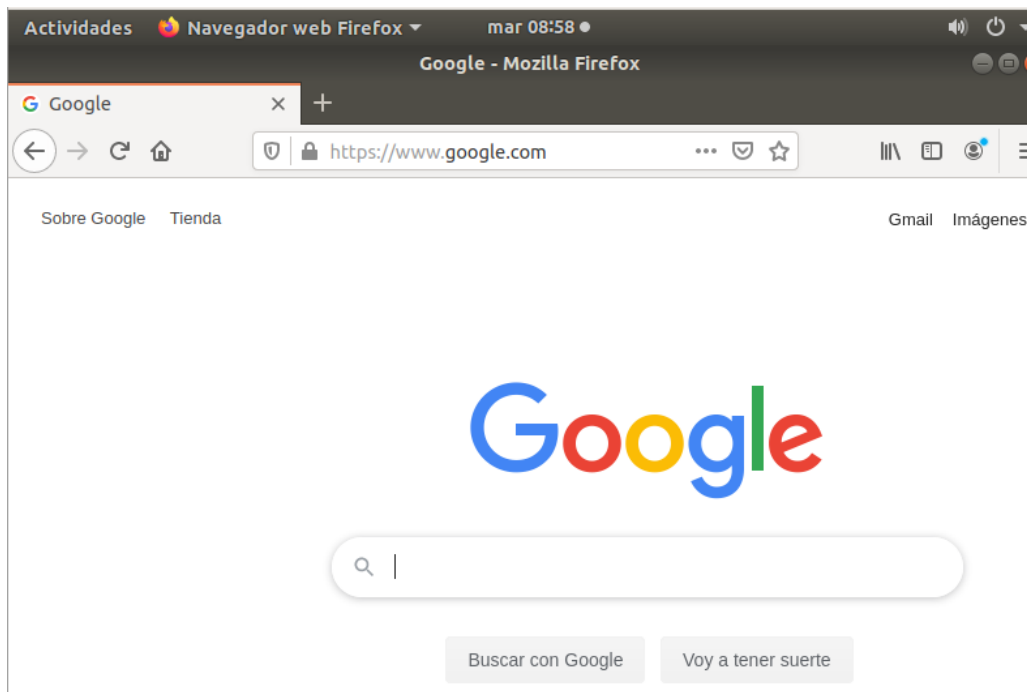
```
fmol107@i103-02:/etc/squid$ sudo systemctl restart squid
```

Vamos a comprobar los cambios en la configuración del proxy. Para esto iremos al equipo cliente y pondremos una IP de los ordenadores que no se pueden conectar en el horario definido.

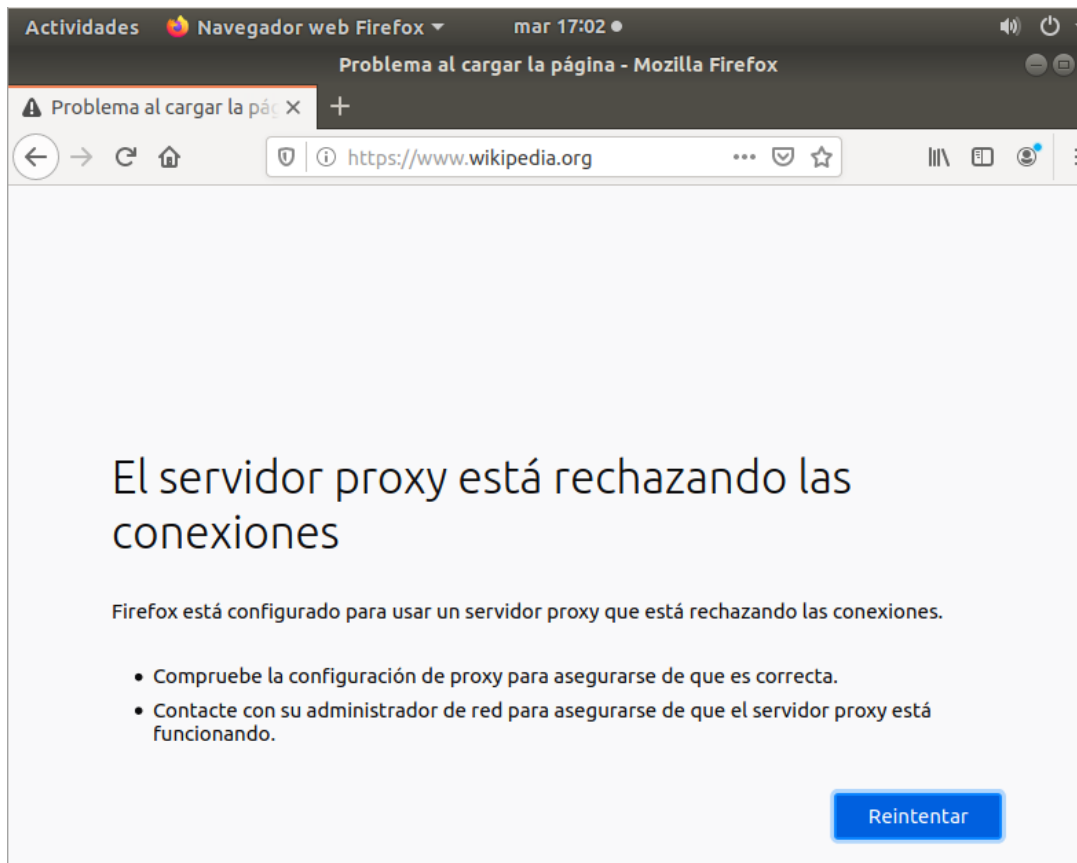
```
fmo1107@i103-02:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:f8:dd:06 brd ff:ff:ff:ff:ff:ff
    inet 192.168.6.45/24 brd 192.168.6.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe80:dd06/64 scope link
        valid_lft forever preferred_lft forever
fmo1107@i103-02:~$
```

Por ultimo intentaremos acceder a alguna web.

Como son las 08:58 el proxy nos permite el acceso.

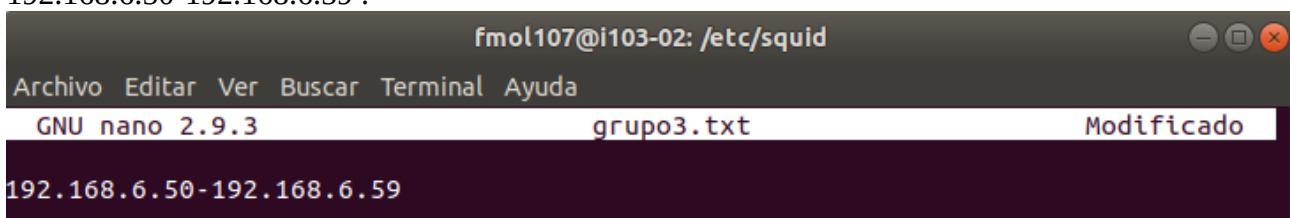


Pero si cambiamos hora a una dentro del rango horario, nos deniega el acceso.



- Crea otro grupo de equipos a los cuales no se les permita conectar con sitios web cuya URL contengan las palabras “sex” y “porn”.
- Vuelve a comprobar que este punto funciona también.

Crearemos un archivo llamado “grupo3.txt”. Dentro de este estará el siguiente rango de IPs: 192.168.6.50-192.168.6.59 .



Ahora editaremos el fichero *squid.conf*. Tendremos que añadir dos ACL y una regla.

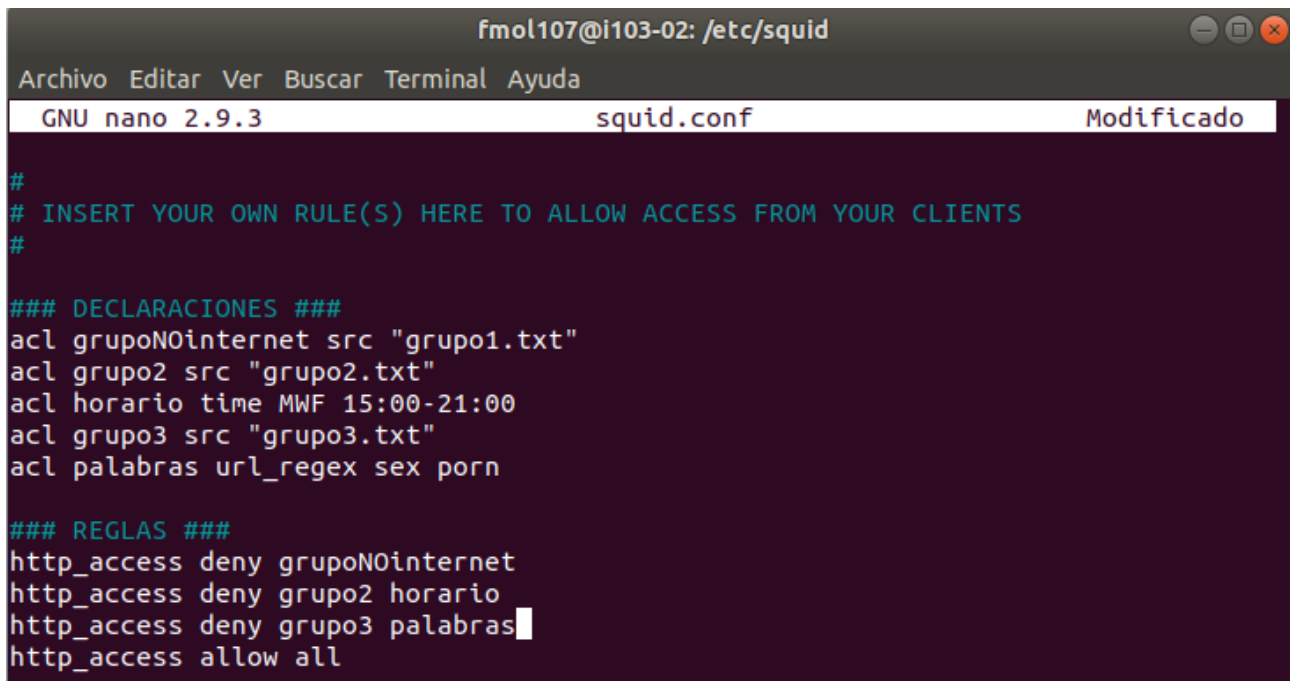
Definiremos la siguiente ACL:

- `acl grupo3 src “grupo3.txt”`
- `acl horario time MWF 15:00-21:00`

Y la siguiente regla:

- `http_access deny grupo3 palabras`

Con esto denegamos el acceso a sitios web cuya URL contengan las palabras “sex” y “porn” a un grupo de ordenadores, pero le permitimos el acceso al resto de sitios.



```
fmol107@i103-02: /etc/squid
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 2.9.3 squid.conf Modificado
#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#
### DECLARACIONES ###
acl grupoNOinternet src "grupo1.txt"
acl grupo2 src "grupo2.txt"
acl horario time MWF 15:00-21:00
acl grupo3 src "grupo3.txt"
acl palabras url_regex sex porn
### REGLAS ###
http_access deny grupoNOinternet
http_access deny grupo2 horario
http_access deny grupo3 palabras
http_access allow all
```

Guardamos los cambios y comprobamos si hemos cometido algún error de sintaxis con el siguiente comando:

```
fmol107@i103-02:/etc/squid$ sudo squid -k reconfigure
```

Para aplicar los cambios reiniciamos el servicio squid.

```
fmol107@i103-02:/etc/squid$ sudo systemctl restart squid
```

Falta comprobar el el equipo cliente que estas reglas funcionan, pero haciendo pruebas en clase comprobamos que los resultados no son concluyentes. Estas reglas no funcionan correctamente.

Problemas encontrados:

- Cada vez que quería que se aplicarían los cambios en la configuración del proxy, en el cliente tenía que reiniciar la maquina, sino no se efectuaban los cambios..
- La regla que deniega los sitios web cuya URL contengan las palabras “sex” y “porn” no funciona correctamente.

Fuentes:

- <https://aules.edu.gva.es/moodle/mod/resource/view.php?id=230906>
- https://es.wikipedia.org/wiki/Servidor_proxy
- [https://es.wikipedia.org/wiki/Squid_\(programa\)](https://es.wikipedia.org/wiki/Squid_(programa))

Alumnos participantes:

- Franco Matias Oscar Larrea