

Instalación y configuración de un proxy Squid en Ubuntu 20.04

Vamos a instalar un servidor Proxy Squid en nuestra máquina Ubuntu. Desde otra máquina windows cliente realizaremos las pruebas. Para ello la máquina Ubuntu debe tener la siguiente configuración de red en Virtualbox:

Máquina Ubuntu. Sobre ella instalamos el cliente. Tendrá dos interfaces de red. Una NAT, la que se conectará a la red. Y otra red Nat interna que será la que se comuniquen con la máquina cliente.

Máquina Windows.Red Nat interna Cliente.

Empezamos con la instalación de squid en la máquina Ubuntu.

Lo primero

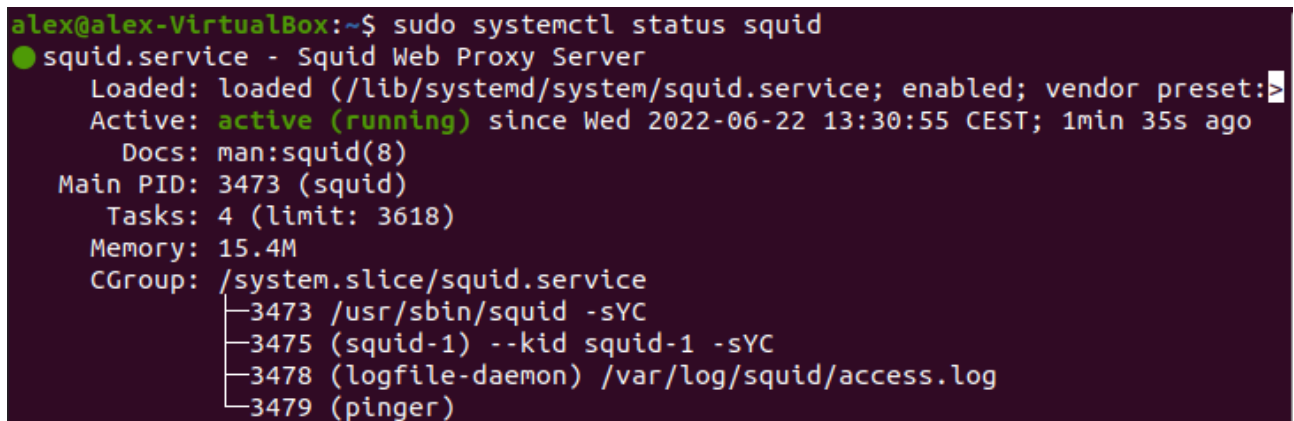
```
sudo apt-get update
```

seguidamente

```
sudo apt-get install squid3
```

Una vez que se complete la instalación, el servicio Squid se iniciará automáticamente. Para verificar esto, verifique el estado del servicio:

```
sudo systemctl status squid
```



```
alex@alex-VirtualBox:~$ sudo systemctl status squid
● squid.service - Squid Web Proxy Server
   Loaded: loaded (/lib/systemd/system/squid.service; enabled; vendor preset:
   Active: active (running) since Wed 2022-06-22 13:30:55 CEST; 1min 35s ago
     Docs: man:squid(8)
    Main PID: 3473 (squid)
      Tasks: 4 (limit: 3618)
    Memory: 15.4M
    CGroup: /system.slice/squid.service
            └─3473 /usr/sbin/squid -sYC
               └─3475 (squid-1) --kid squid-1 -sYC
                  └─3478 (logfile-daemon) /var/log/squid/access.log
                     └─3479 (pinger)
```

Configuramos ahora el navegador del cliente para que navegue a través del proxy.

En este caso el que tenemos instalado es el Firefox.

Entramos en ajustes y configuramos el proxy:

Configurar acceso proxy a Internet

- ☐ Sin proxy
- ☐ Autodetectar configuración del proxy para esta red
- ☐ Usar la configuración del proxy del sistema
- ☒ Configuración manual del proxy

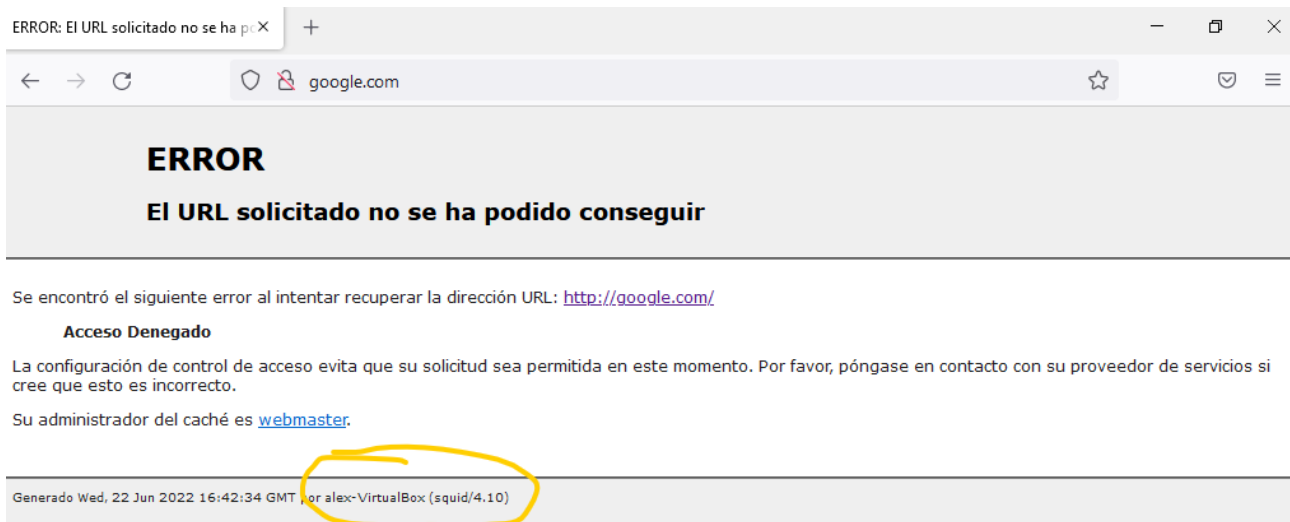
Proxy HTTP 10.0.2.6

Puerto 3128

☒ Usar también este proxy para HTTPS

Proxy HTTPS 10.0.2.6

Puerto 3128



Ahora el proxy nos impide navegar. Para ello debemos pasar a la configuración de este.

El mensaje de error lo podemos editar desde:

```
alex@alex-VirtualBox:~$ cd /usr/share/squid/errors/es-es/
alex@alex-VirtualBox:/usr/share/squid/errors/es-es$ ls -la
total 192
drwxr-xr-x  2 root root  4096 jun 22 13:30 .
drwxr-xr-x 48 root root 12288 jun 22 13:30 ..
-rw-r--r--  1 root root  1114 nov 10  2019 ERR_ACCESS_DENIED
```

```
</head><body id="%c">
<div id="titles">
<h1>ERROR</h1>
<h2>El URL solicitado no se ha podido conseguir. Los siento Alex</h2>
</div>
<hr>
```



Para ver que todos los cambios que vamos realizando van funcionando podemos consultar los distintos logs desde:

```
alex@alex-VirtualBox:/usr/share/squid/errors/es-es$ ls /var/log/squid/
access.log  cache.log
alex@alex-VirtualBox:/usr/share/squid/errors/es-es$
```

Nuestro objetivo ahora es configurar squid para que nuestro cliente sí pueda navegar a través de internet.

Tendremos que tocar el archivo de configuración que se encuentra en /etc/squid/squid.conf

```
alex@alex-VirtualBox:/usr/share/squid/errors/es-es$ cd /etc/squid
alex@alex-VirtualBox:/etc/squid$ ls
conf.d  errorpage.css  squid.conf
alex@alex-VirtualBox:/etc/squid$
```

entramos en squid.conf con sudo y nano.

Ahí tenemos toda la información que podemos tocar, parámetros, comentarios que podemos descomentar, etc

Buscamos por insert

y encontramos este apartado:

```
#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#
include /etc/squid/conf.d/*

# 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
#http_access allow localnet
http_access allow localhost
```

ý aquí añadiremos nuestra regla para que permita navegar a todos:

```
#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#
include /etc/squid/conf.d/*

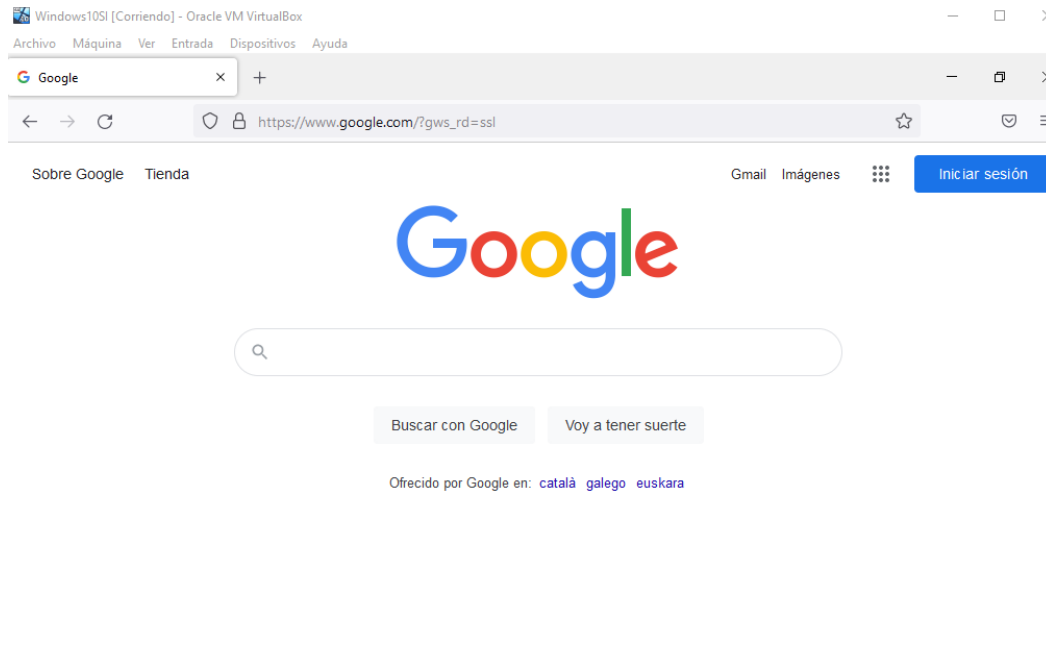
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
#http_access allow localnet
http_access allow localhost
#REGLA PARA DEJAR ACCEDER AL RESTO DE CLIENTES A TRAVÉS DEL PROXY
http_access allow all

# And finally deny all other access to this proxy
http_access deny all
```

guardamos el fichero.

sudo systemctl restart squid

y ahora el cliente ya podría navegar:



Ahora lo que nos interesa es que los clientes no puedan acceder a determinados dominios. Por ejemplo denegar redes sociales como twitter o instagram.

Para ello hacemos lo siguiente:

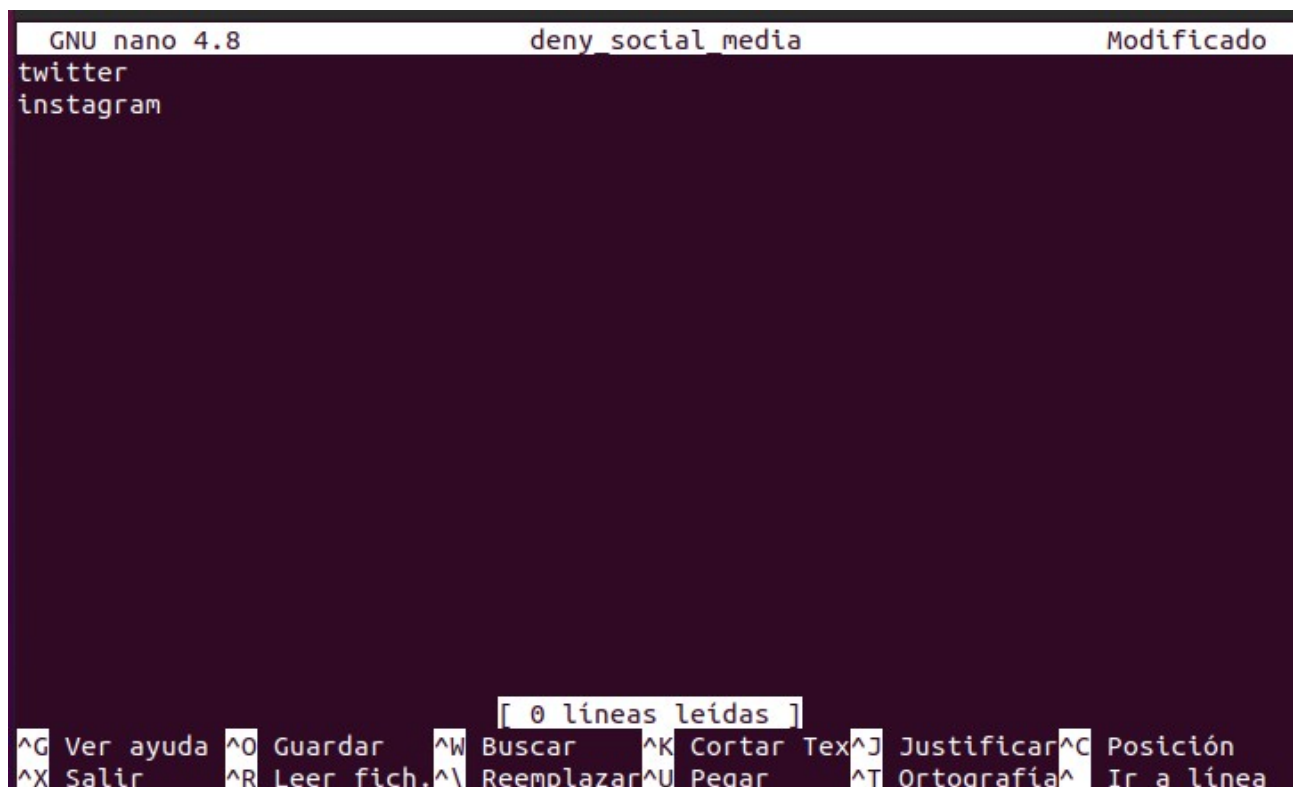
dentro de la ubicación en la que nos encontramos:

```
alex@alex-VirtualBox:/etc/squid$ ls
conf.d  errorpage.css  squid.conf
alex@alex-VirtualBox:/etc/squid$
```

creamos un fichero que llamaremos deny_social_media con touch

```
alex@alex-VirtualBox:/etc/squid$ sudo touch deny_social_media
alex@alex-VirtualBox:/etc/squid$
```

entro en el archivo creado con nano y escribiré twitter e instagram:



```
GNU nano 4.8 deny_social_media Modificado
twitter
instagram

[ 0 líneas leídas ]
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Text ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar ^T Ortografía ^_ Ir a línea
```

Ahora debo generar un ACL (Access Control List) . Lo haremos en el fichero .conf

lo hacemos sobre la regla que hemos creado anteriormente utilizando expresiones regulares de esta manera:

```
#http_access allow localnet
http_access allow localhost

#REGLA O ACL PARA DENEGACION DE ACCESO A REDES SOCIALES

acl deny_social_media url_regex -i "/etc/squid/deny_social_media"
http_access deny deny_social_media

#REGLA PARA DEJAR ACCEDER AL RESTO DE CLIENTES A TRAVÉS DEL PROXY
http_access allow all
```

Recargamos de nuevo el servicio

```
sudo systemctl restart squid
```

y ahora desde el cliente si intento entrar en twitter:



El servidor proxy está rechazando las conexiones

Ha ocurrido un error al conectar con twitter.com.

- Compruebe la configuración de proxy para asegurarse de que es correcta.
- Contacte con su administrador de red para asegurarse de que el servidor proxy está funcionando.

Reintentar

lo mismo con instagram:

El servidor proxy está rechazando las conexiones

Ha ocurrido un error al conectar con www.instagram.com.

- Compruebe la configuración de proxy para asegurarse de que es correcta.
- Contacte con su administrador de red para asegurarse de que el servidor proxy está funcionando.

pero puedo comprobar que sí me deja acceder a otras webs.

Otras opciones sería obligar a un usuario a logearse para entrar en internet.

¿Cómo lo haríamos?

Entramos de nuevo en el archivo de configuración y volvemos a nuestras líneas configuradas.

Lo primero añadiremos estas líneas:

```
#http_access allow localnet
http_access allow localhost

#GESTIÓN DE USUARIOS AUTENTICADOS PROXY SQUID
#Utilidad para comprobar user/pass en el fichero /etc/squid/users

auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/users

#num de instancias activas que vamos a permitir
auth_param basic children 15
#mensaje de autenticacion de usuario
auth_param basic realm PROXY PRUEBA SAD
```

Indican el archivo de donde cogeremos los usuarios. Luego lo creamos.
Seguidamente el número de usuarios que pueden conectarse a la vez. En este caso 15
y el mensaje que queremos que salga al usuario al autenticarse.

Ahora vamos tras la ACL que hemos creado anteriormente:

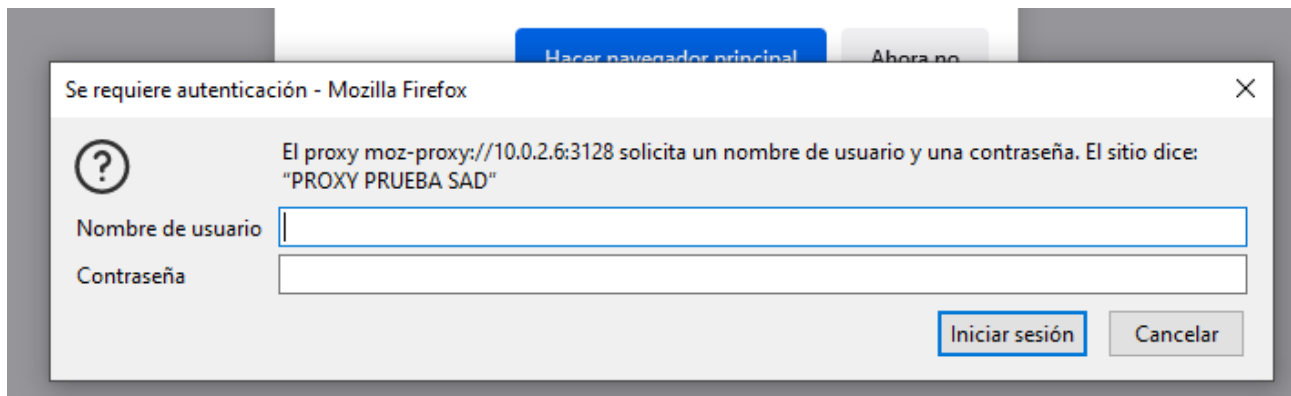
```
#REGLA O ACL PARA DENEGACION DE ACCESO A REDES SOCIALES

acl deny_social_media url_regex -i "/etc/squid/deny_social_media"
acl block_users proxy_auth REQUIRED
http_access deny deny_social_media
http_access allow block_users
```

Es importante seguir el orden a la hora de realizar la configuración ya que este es secuencial de arriba a abajo y de izquierda a derecha.

Recargo el servicio: `sudo systemctl restart squid`

Ahora al abrir de nuevo el navegador desde el cliente requiere usuario y contraseña:



Ahora mismo no tengo ningún usuario creado. Necesitamos instalar Apache2-utils instalado para continuar. En mi caso ya lo tengo (apt install) voy a generar ahora el fichero con la aplicación htpasswd

```
alex@alex-VirtualBox:~$ sudo htpasswd -c /etc/squid/users alex
New password:
Re-type new password:
Adding password for user alex
alex@alex-VirtualBox:~$
```

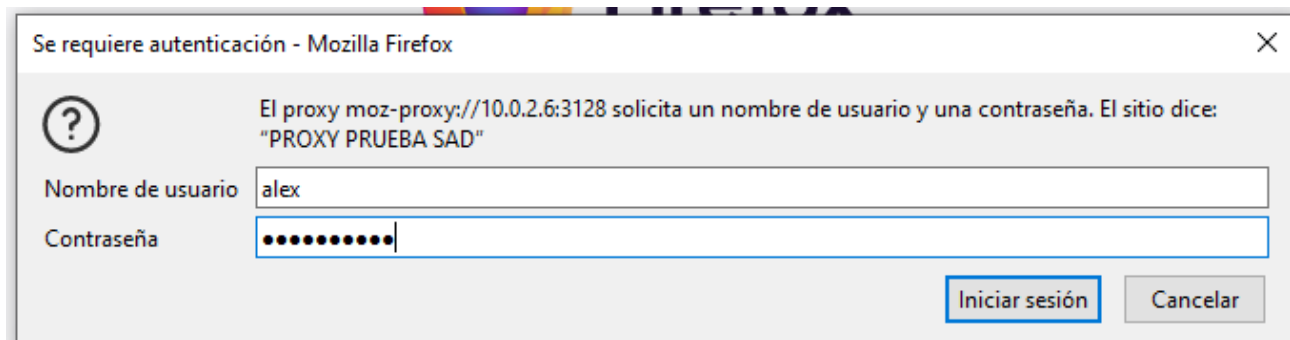
El resto de usuarios los creamos sin el parámetro -c para que no machaque el que ya tenía. Creo un par más

```
alex@alex-VirtualBox:~$ sudo htpasswd /etc/squid/users david
New password:
Re-type new password:
Adding password for user david
alex@alex-VirtualBox:~$ sudo htpasswd /etc/squid/users ana
New password:
Re-type new password:
Adding password for user ana
alex@alex-VirtualBox:~$
```

Podemos comprobar con un cat sobre el fichero. Veremos los usuarios con las claves encriptadas.

```
alex@alex-VirtualBox:~$ cat /etc/squid/users
alex:$apr1$d2dq2/e6$WHwDMMYcIFg3iYc8hmqnz.
david:$apr1$cwPKo84n$KYRkCaDYhGcjUPz06E5AN0
ana:$apr1$SwiLjiz9$UHP00YRUIhT3UlpSJXzSZ0
alex@alex-VirtualBox:~$
```

Ahora solo queda probar en el cliente que funciona...



The image shows a screenshot of a Mozilla Firefox authentication dialog box. The title bar reads "Se requiere autenticación - Mozilla Firefox". The main text area contains a question mark icon and the message: "El proxy moz-proxy://10.0.2.6:3128 solicita un nombre de usuario y una contraseña. El sitio dice: 'PROXY PRUEBA SAD'". Below this, there are two input fields: "Nombre de usuario" with the text "alex" and "Contraseña" with masked characters (dots). At the bottom right, there are two buttons: "Iniciar sesión" (highlighted with a blue border) and "Cancelar".

Y evidentemente ya puedo navegar!! con las reglas de proxy añadidas anteriormente.