

Implementar un Proxy HTTP con políticas de autenticación para navegación en Internet

Yefried Hernan Garcia Vaquiro
e-mail: garcia.h.97@gmail.com

RESUMEN: El presente documento muestra cómo se implementó un proxy HTTP no transparente mediante Squid 6.6 sobre Ubuntu Server ARM, configurado en el puerto 3128 sin directiva `intercept`, lo que obliga al cliente a declarar explícitamente el servidor proxy en el navegador. La autenticación de usuarios se realizó mediante el módulo `basic_ncsa_auth` con credenciales almacenadas en formato `APRI-MD5`, vinculando al usuario `hegarcia` al grupo `grupo_internet`. Se definió un perfil de lista negra referenciando el archivo `blacklist.txt`, el cual contiene los dominios `www.hotmail.com`, `www.youtube.com` y `www.elnuevodia.com.co`.

PALABRAS CLAVE: Servidor, Cliente, Dominio, Proxy, Usuario, Interfaces, IP, Linux, Squid.

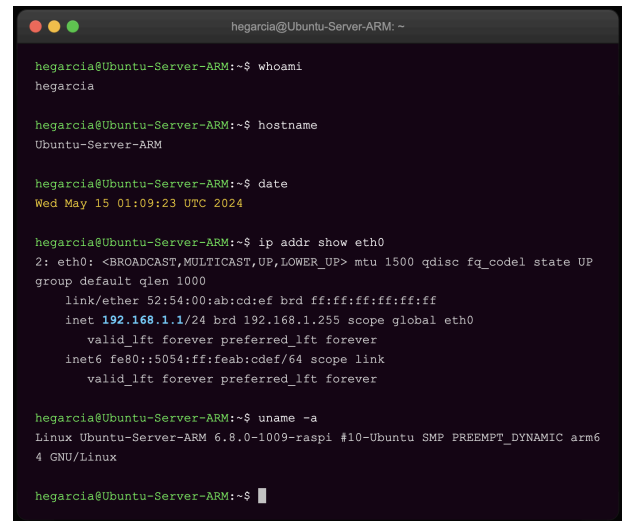
INTRODUCCIÓN

En este trabajo se documenta la implementación de un proxy HTTP no transparente sobre Ubuntu Server utilizando Squid 6.6, integrando autenticación básica por usuario mediante el módulo NCSA, la definición de un perfil de lista negra con sitios restringidos, y la vinculación de políticas de acceso que asocian usuarios, grupos y restricciones de navegación. Se parte de la instalación limpia del servicio, pasando por la configuración del archivo `squid.conf`, la creación del usuario `hegarcia` con contraseña cifrada, hasta la verificación de bloqueos desde un cliente de la red LAN a través del navegador Firefox, con evidencia registrada en los logs de acceso del servidor.

Por eso, cada vez más, la administración de redes empresariales exige mecanismos de control que regulen el acceso a Internet de forma segura y trazable. Un proxy HTTP no transparente representa una solución probada para este propósito, pues actúa como intermediario declarado entre los clientes de la red local y los servidores externos, obligando a cada usuario a identificarse antes de navegar. A diferencia del proxy transparente, el cliente conoce su existencia y debe configurarlo manualmente, lo que añade una capa de control administrativo.

2. Preparación del Servidor

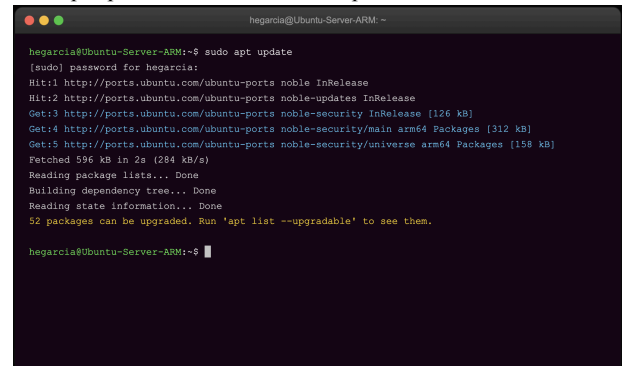
Comenzamos haciendo una verificación del entorno



```
hegarcia@Ubuntu-Server-ARM: ~  
hegarcia@Ubuntu-Server-ARM:~$ whoami  
hegarcia  
hegarcia@Ubuntu-Server-ARM:~$ hostname  
Ubuntu-Server-ARM  
hegarcia@Ubuntu-Server-ARM:~$ date  
Wed May 15 01:09:23 UTC 2024  
hegarcia@Ubuntu-Server-ARM:~$ ip addr show eth0  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP  
group default qlen 1000  
link/ether 52:54:00:ab:cdef:ff:ff:ff:ff:ff:ff  
inet 192.168.1.1/24 brd 192.168.1.255 scope global eth0  
valid_lft forever preferred_lft forever  
inet6 fe80::5054:ff:feab:cdef/64 scope link  
valid_lft forever preferred_lft forever  
hegarcia@Ubuntu-Server-ARM:~$ uname -a  
Linux Ubuntu-Server-ARM 6.8.0-1009-raspi #10-Ubuntu SMP PREEMPT_DYNAMIC arm64  
4 GNU/Linux  
hegarcia@Ubuntu-Server-ARM:~$
```

Figura 1. Verificación del entorno

`sudo apt update`: actualización de repositorios Ubuntu ARM



```
hegarcia@Ubuntu-Server-ARM: ~  
hegarcia@Ubuntu-Server-ARM:~$ sudo apt update  
[sudo] password for hegarcia:  
Hit:1 http://ports.ubuntu.com/ubuntu-ports noble InRelease  
Hit:2 http://ports.ubuntu.com/ubuntu-ports noble-updates InRelease  
Get:3 http://ports.ubuntu.com/ubuntu-ports noble-security InRelease [126 kB]  
Get:4 http://ports.ubuntu.com/ubuntu-ports noble-security/main arm64 Packages [312 kB]  
Get:5 http://ports.ubuntu.com/ubuntu-ports noble-security/universe arm64 Packages [158 kB]  
Fetched 596 kB in 2s (284 kB/s)  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
52 packages can be upgraded. Run 'apt list --upgradable' to see them.  
hegarcia@Ubuntu-Server-ARM:~$
```

Figura 2. Actualización de repositorios

sudo apt install squid squid-common apache2-utils -y:
descarga de paquetes (3,842 kB)

```
hegarcia@Ubuntu-Server-ARM:~$ sudo apt install squid squid-common apache2-utils -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libdbi-perl libecap3 squid-langpack
The following NEW packages will be installed:
  apache2-utils libdbi-perl libecap3 squid squid-common squid-langpack
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
Need to get 3,842 kB of archives.
After this operation, 12.3 MB of additional disk space will be used.
Get:1 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 libdbi-perl arm64 1.643-3build4 [764 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 squid-langpack all 20230809-1 [371 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 libecap3 arm64 1.0.1-3.2build1 [19.1 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 squid-common all 6.6-1ubuntu2 [37.0 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 squid arm64 6.6-1ubuntu2 [2,622 kB]
Get:6 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 apache2-utils arm64 2.4.58-1ubuntu8 [88.4 kB]
Fetched 3,842 kB in 1s (2,901 kB/s)
Selecting previously unselected package libdbi-perl...
Selecting previously unselected package squid-langpack...
```

Figura 3. Instalación de Squid y dependencias

Instalación completada: Squid 6.6 instalado para arm-linux-gnu

```
hegarcia@Ubuntu-Server-ARM:~$ sudo apt install squid squid-common apache2-utils -y
Unpacking libdbi-perl (1.643-3build4) ...
Unpacking squid-langpack (20230809-1) ...
Unpacking libecap3:arm64 (1.0.1-3.2build1) ...
Unpacking squid-common (6.6-1ubuntu2) ...
Unpacking squid (6.6-1ubuntu2) ...
Unpacking apache2-utils (2.4.58-1ubuntu8) ...
Setting up libdbi-perl (1.643-3build4) ...
Setting up squid-langpack (20230809-1) ...
Setting up libecap3:arm64 (1.0.1-3.2build1) ...
Setting up squid-common (6.6-1ubuntu2) ...
Setting up apache2-utils (2.4.58-1ubuntu8) ...
Setting up squid (6.6-1ubuntu2) ...
Squid Cache version 6.6 for arm-linux-gnu
Creating Squid SSL certificate database...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for ufw (0.36.2-6) ...

hegarcia@Ubuntu-Server-ARM:~$
```

Figura 4. Instalación completada — Squid 6.6 listo

sudo systemctl status squid: versión 6.6, servicio active (running)

```
hegarcia@Ubuntu-Server-ARM:~$ sudo systemctl status squid
● squid.service - Squid Web Proxy Server
   Loaded: loaded (/usr/lib/systemd/system/squid.service; enabled; preset: enabled)
   Active: active (running) since Wed 2024-05-15 01:11:47 UTC; 1min 30s ago
     Docs: man:squid(8)
    Main PID: 1852 (squid)
      Tasks: 4 (limit: 4582)
     Memory: 15.3M (peak: 15.8M)
        CPU: 124ms
    CGroup: /system.slice/squid.service
            └─1852 /usr/sbin/squid --foreground -sX
            └─1854 (squid-1) --kid squid-1 --foreground -sX
            └─1856 (logfile-daemon) /var/log/squid/access.log
May 15 01:11:47 Ubuntu-Server-ARM systemd[1]: Started squid.service.
```

Figura 5. Verificación: versión y estado del servicio Squid

4. Autenticación de Usuario

Creación del archivo de contraseñas con permisos 640 propietario proxy:proxy

```
hegarcia@Ubuntu-Server-ARM:~$ sudo touch /etc/squid/squid_passwd
hegarcia@Ubuntu-Server-ARM:~$ sudo chown proxy:proxy /etc/squid/squid_passwd
hegarcia@Ubuntu-Server-ARM:~$ sudo chmod 640 /etc/squid/squid_passwd
hegarcia@Ubuntu-Server-ARM:~$ ls -la /etc/squid/
total 28
drwxr-xr-x 2 root root 4096 May 15 01:13 .
drwxr-xr-x 96 root root 4096 May 15 01:11 ..
-rw-r--r-- 1 root root 8472 May 15 01:11 squid.conf
-rw-r----- 1 proxy proxy 0 May 15 01:13 squid_passwd

✓ Archivo squid_passwd creado - propietario: proxy:proxy - permisos: 640
✓ Ruta: /etc/squid/squid_passwd

hegarcia@Ubuntu-Server-ARM:~$
```

Figura 6. Preparación del archivo de contraseñas

htpasswd -c /etc/squid/squid_passwd hegarcia: usuario creado con cifrado APR1-MD5

```
hegarcia@Ubuntu-Server-ARM:~$ sudo htpasswd -c /etc/squid/squid_passwd hegarcia
New password:
Re-type new password:
Adding password for user hegarcia

# Nota: El flag -c crea el archivo desde cero.
# Para agregar más usuarios sin borrar: omitir -c
# sudo htpasswd /etc/squid/squid_passwd otro_usuario

✓ Usuario 'hegarcia' creado exitosamente
✓ Contraseña cifrada con algoritmo APR1-MD5
✓ Será vinculado al grupo 'grupo_internet' en squid.conf

hegarcia@Ubuntu-Server-ARM:~$
```

Figura 7. Creación del usuario 'hegarcia' con htpasswd

cat /etc/squid/squid_passwd: hash visible, archivo verificado

```
hegarcia@Ubuntu-Server-ARM:~$ sudo cat /etc/squid/squid_passwd
hegarcia:$aprl5m79zKl2A6vXk1RF508NzUwPLe7d4s1

hegarcia@Ubuntu-Server-ARM:~$ ls -la /etc/squid/squid_passwd
-rw-r----- 1 proxy proxy 46 May 15 01:14 /etc/squid/squid_passwd

hegarcia@Ubuntu-Server-ARM:~$ wc -l /etc/squid/squid_passwd
1 /etc/squid/squid_passwd

✓ 1 usuario registrado: hegarcia
✓ Hash APRI-MD5: $aprl5m79zKl2A6vXk1RF508NzUwPLe7d4s1
✓ Archivo listo para ser referenciado en auth_param
```

Figura 8. Verificación: usuario hegarcia en el archivo de claves

Respaldo de squid.conf y limpieza para nueva config

```
hegarcia@Ubuntu-Server-ARM:~$ sudo cp /etc/squid/squid.conf /etc/squid/squid.conf.bak
hegarcia@Ubuntu-Server-ARM:~$ ls -lh /etc/squid/squid.conf*
-rw-r--r-- 1 root root 8.3K May 15 01:11 /etc/squid/squid.conf
-rw-r--r-- 1 root root 8.3K May 15 01:15 /etc/squid/squid.conf.bak

hegarcia@Ubuntu-Server-ARM:~$ sudo truncate -s 0 /etc/squid/squid.conf
hegarcia@Ubuntu-Server-ARM:~$ ls -lh /etc/squid/squid.conf
-rw-r--r-- 1 root root 0 May 15 01:15 /etc/squid/squid.conf

✓ Respaldo guardado -> squid.conf.bak (8.3K)
✓ squid.conf vaciado -> listo para nueva configuración personalizada

hegarcia@Ubuntu-Server-ARM:~$
```

Figura 9. Respaldo de squid.conf y limpieza para nueva config

5. Configuración en nano

sudo nano /etc/squid/squid.conf se abre

```
hegarcia@Ubuntu-Server-ARM:~$ sudo nano /etc/squid/squid.conf

GNU nano 7.2 /etc/squid/squid.conf
#
# /etc/squid/squid.conf
# Proxy HTTP No Transparente con Autenticación de Usuario
# Implementación: Featlica 5
# Servidores: 192.168.1.1 | Puerto: 3128
# Fecha: 15 de Mayo 2024
#
[ Archivo nuevo - cursor aqui ]
```

Figura 10. Configuración del Proxy, Apertura de squid.conf en nano

Sección http_port 3128 (sin transparent = no transparente), logs, cache

```
GNU nano 7.2 /etc/squid/squid.conf Modified
#
# SECCION 1: PUERTO - PROXY HTTP NO TRANSPARENTE
#
# La ausencia de 'transparent' o 'intercept' lo define como NO transparente.
# El Cliente DEBE configurar el proxy manualmente en el navegador.
#
# TRANSPARENTE (NO usar): http_port 3128 intercept
# NO TRANSPARENTE (USAR) : http_port 3128
#
http_port 3128
#
# Identificación del proxy en paginas de error
visible_hostname hegarcia
#
# Cache en memoria
cache_mem 256 MB
maximum_object_size_in_memory 512 KB
cache_dir ufs /var/spool/squid 1000 16 256
coredump_dir /var/spool/squid
#
# Registros de acceso
access_log /var/log/squid/access.log squid
cache_log /var/log/squid/cache.log
pid_filename /run/squid.pid
```

Figura 11. nano: Puerto 3128: diferencia Transparente vs No Transparente

auth_param basic program /usr/lib/squid/basic_ncsa_auth con realm y TTL

```
GNU nano 7.2 /etc/squid/squid.conf Modified
#
# SECCION 2: AUTENTICACION BASICA (NCSA / htpasswd)
#
# Programa: basic_ncsa_auth lee el archivo squid_passwd (formato htpasswd)
# Realm: Texto que aparece en el dialogo de autenticación del navegador
# TTL: Tiempo que las credenciales permanecen en cache (2 horas)
# children: Procesos autenticadores paralelos (5 maximo)
#
auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/squid_passwd
auth_param basic realm "Proxy hegarcia - Autenticacion Requerida"
auth_param basic credentialsttl 2 hours
auth_param basic casensitive off
auth_param basic children 5
```

Figura 12. nano: Autenticación básica NCSA (auth_param)

ACL usuarios_autenticados y ACL grupo_internet proxy_auth hegarcia, puertos seguros

```
GNU nano 7.2 /etc/squid/squid.conf Modified
#
# SECCION 3: LISTAS DE CONTROL DE ACCESO (ACL) - Usuarios y Grupo
#
# ACL: usuarios_autenticados - cualquier usuario que haya iniciado sesion
acl usuarios_autenticados proxy_auth REQUIRED
#
# ACL: grupo_internet - asocia al usuario 'hegarcia' al grupo de acceso
# (Puede agregar mas usuarios: acl grupo_internet proxy_auth hegarcia jgarcia ...)
acl grupo_internet proxy_auth hegarcia
#
# ACL: Redes LAN autorizadas para usar el proxy
acl LocalNet src 192.168.1.0/24
acl LocalNet src 10.0.0.0/8
#
# ACL: Puertos permitidos (Safe_ports) y metodo CONNECT (para HTTPS)
acl SSL_ports port 443
acl Safe_ports port 80 # http
acl Safe_ports port 443 # https
acl Safe_ports port 21 # ftp
acl Safe_ports port 8080 # alt-http
acl CONNECT method CONNECT
```

Figura 13. nano: ACL: Usuarios autenticados y grupo_internet

ACL lista_negra dstdomain "/etc/squid/blacklist.txt" el perfil de bloqueo

```

# SECCION 4: PERFIL - LISTA NEGRA DE SITIOS BLOQUEADOS
# Esta ACL, lee el archivo blacklist.txt que contiene los dominios bloqueados:
# www.hotmail.com
# www.youtube.com
# www.elnuevodia.com.co
#
# Recorrido - Instalacion desde (dominio destino de la solicitud HTTP)
# El punto (.) al inicio del dominio indica todos sus subdominios
acl lista_negra dstdomain "/etc/squid/blacklist.txt"

```

Figura 14. nano: ACL: Perfil lista_negra (blacklist)

http_access deny lista_negra, allow grupo_interne, deny all

```

# Seguridad basica: Bloquear puertos no seguros
http_access deny !safe_ports

# SECCION 5: POLITICAS DE ACCESO - Vincula perfil + grupo + autentificacion
#
# Regla 1 (BLOQUEO): Denegar TODOS los sitios en la lista negra
# Aplica a todos, autenticados o no, hotmail, youtube, elnuevodia bloqueados.
http_access deny lista_negra

# Regla 2 (PERMITIR): Grupo Internet con autentificacion valida
# hegarcia puede navegar en sitios NO listados en la lista negra.
http_access allow grupo_internet

# Regla 3 (DENEGACION TOTAL): Bloquear todo lo demas
# Usuarios no autenticados o no pertenecientes a grupo_internet: DENEGADOS.
http_access deny all

# [FIN DE squid.conf]

```

Figura 15. nano: Políticas de acceso http_access (vincula todo)

Guardado del archivo: nano muestra el prompt Save modified buffer

```

File Name: /etc/squid/squid.conf
http_access deny !safe_ports
http_access deny CONNECT !SSL_ports
http_access deny lista_negra
http_access allow grupo_internet
http_access deny all

# [FIN DE squid.conf]

Save modified buffer?
[Yes] [No] [Cancel]

```

Figura 16. nano: Guardado del archivo squid.conf (Ctrl+O / Ctrl+X)

sudo nano /etc/squid/blacklist.txt: los 3 dominios

```

# /etc/squid/blacklist.txt
# Lista Negra - Perfil de Sitios Bloqueados
# Formato S: Proxy HTTP No Transparente
# Fecha: 15 Mayo 2024
#
# Un dominio por línea. El punto (.) incluye subdominios.
#
.hotmail.com
www.hotmail.com
.
.youtube.com
www.youtube.com
.elnuevodia.com.co
www.elnuevodia.com.co

```

Figura 17. nano: Creación y edición de /etc/squid/blacklist.txt

cat /etc/squid/blacklist.txt: confirmación de los 3 sitios registrados

```

hegarcia@Ubuntu-Server-ARM:~$ sudo cat /etc/squid/blacklist.txt
# /etc/squid/blacklist.txt - Lista Negra - Perfil de Sitios Bloqueados
#
.hotmail.com
www.hotmail.com
.
.youtube.com
www.youtube.com
.elnuevodia.com.co
www.elnuevodia.com.co

hegarcia@Ubuntu-Server-ARM:~$ ls -la /etc/squid/blacklist.txt
-rw-r--r-- 1 root root 192 May 15 01:16 /etc/squid/blacklist.txt

✓ 3 dominios en lista negra: hotmail.com | youtube.com | elnuevodia.com.co
✓ Perfil listo para ser referenciado por la ACL 'lista_negra' en squid.conf

hegarcia@Ubuntu-Server-ARM:~$

```

Figura 18. Verificación del contenido de blacklist.txt

6. Validación y Reinicio

sudo squid -k parse: "Configuration file is OK", 8 entradas cargadas de blacklist

```

hegarcia@Ubuntu-Server-ARM:~$ sudo squid -k parse
2024/05/15 01:17:43 kid1| Reading configuration file /etc/squid/squid.conf...
2024/05/15 01:17:43 kid1| Processing Configuration File: /etc/squid/squid.conf (depth 0)
2024/05/15 01:17:43 kid1| Processing Configuration File: /etc/squid/blacklist.txt (depth 1)
2024/05/15 01:17:43 kid1| Loaded 8 blacklist entries from /etc/squid/blacklist.txt
2024/05/15 01:17:43 kid1| Initializing https proxy context
2024/05/15 01:17:43 kid1| configuration file is OK

✓ Sintaxis correcta - cero errores detectados
✓ 8 entradas cargadas desde blacklist.txt
✓ Configuración lista para aplicar

hegarcia@Ubuntu-Server-ARM:~$

```

Figura 19. Validación y Reinicio squid -k parse: test de sintaxis

sudo systemctl restart squid + status active (running) PID 2145

```
hegarclia@Ubuntu-Server-ARM: ~$ sudo systemctl restart squid
hegarclia@Ubuntu-Server-ARM:~$ sudo systemctl status squid
● squid.service - Squid Web Proxy Server
   Loaded: loaded (/usr/lib/systemd/system/squid.service; enabled; preset: enabled)
   Active: active (running) since Wed 2024-05-15 01:18:02 UTC; 5s ago
     Docs: man:squid(8)
    Main PID: 2145 (squid)
      Tasks: 4 (limit: 4592)
    Memory: 18.7M (peak: 19.1M)
       CPU: 98ms
    CGroup: /system.slice/squid.service
            └─2145 /usr/sbin/squid --foreground -sYC
            └─2147 (squid-1) --kld squid-1 --foreground -sYC
            └─2149 (logfile-daemon) /var/log/squid/access.log
May 15 01:18:02 Ubuntu-Server-ARM squid[2145]: Squid Parent: will start 1 klds
May 15 01:18:02 Ubuntu-Server-ARM squid[2145]: Squid Cache version 4.6 for arm-linux-gnu
May 15 01:18:02 Ubuntu-Server-ARM systemd[1]: Started squid.service - Squid Web Proxy Server.
```

Figura 20. Reinicio de Squid y verificación del estado

ss -tlnp | grep 3128: puerto 0.0.0.0:3128 escuchando, proxy listo

```
hegarclia@Ubuntu-Server-ARM: ~$ sudo ss -tlnp | grep 3128
LISTEN 0 128 0.0.0.0:3128 0.0.0.0:* users:(("squid",pid=2145,fd=11))
LISTEN 0 128 [::]:3128 [::]:* users:(("squid",pid=2145,fd=12))

hegarclia@Ubuntu-Server-ARM:~$ sudo netstat -tlnp | grep squid
tcp 0 0 0.0.0.0:3128 0.0.0.0:* LISTEN 2145/squid
tcp6 0 0 :::3128 :::* LISTEN 2145/squid

✓ Squid escuchando en 0.0.0.0:3128 (todas las Interfaces)
✓ Proxy No Transparente activo - PID: 2145
✓ El cliente debe configurar: Proxy = 192.168.1.1, Puerto = 3128

hegarclia@Ubuntu-Server-ARM:~$
```

Figura 21. Verificación: puerto 3128 escuchando conexiones

6. Configuración del Cliente

Página de Configuración General de Firefox, botón "Configuración de Red"

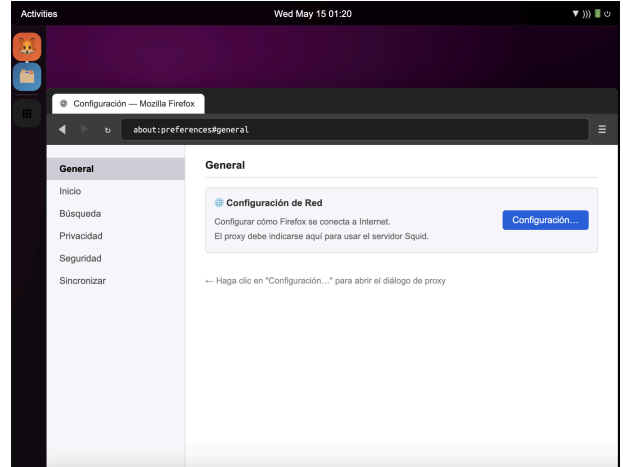


Figura 22. Firefox: Configuración General, sección Red. Diálogo de Red abierto, opción "Sin proxy" seleccionada (estado inicial)

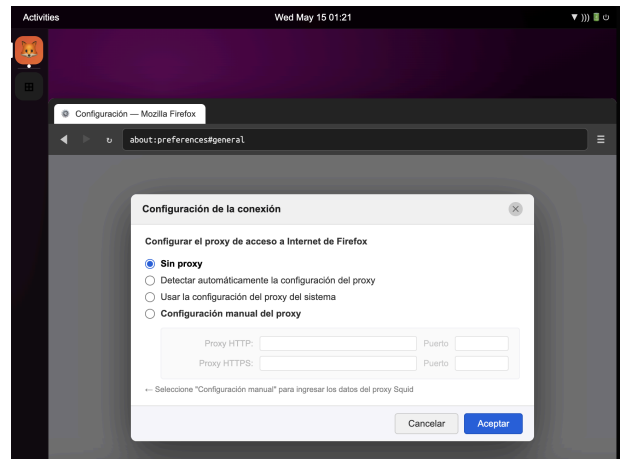


Figura 23. Firefox: Diálogo Configuración de Red (sin proxy aún)

Proxy manual configurado: 192.168.1.1 puerto 3128 aplicado

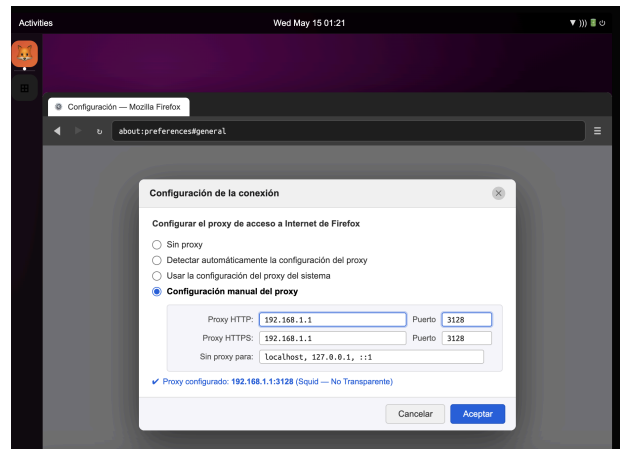


Figura 24. Firefox: Proxy Manual: 192.168.1.1:3128 configurado

6. Pruebas de la Lista Negra

URL bar con www.youtube.com, spinner de carga

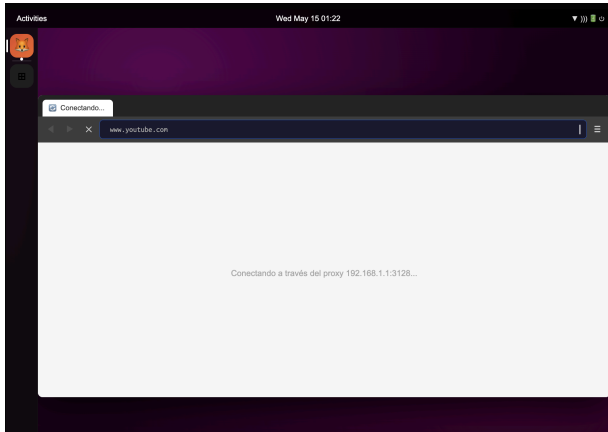


Figura 25. Ingresando www.youtube.com en la barra de URL

Página de error Squid: youtube.com, 403 Forbidden / lista_negra ACL

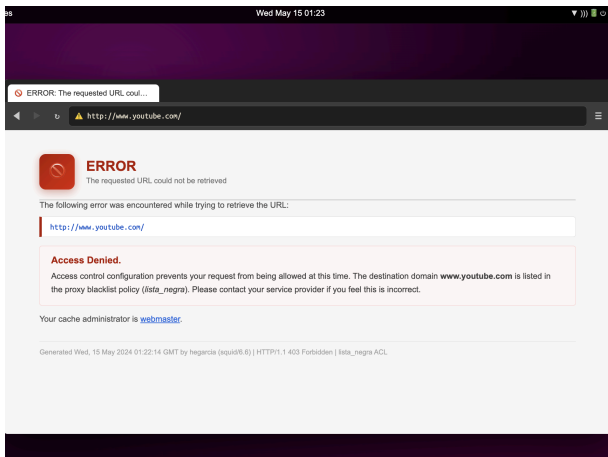


Figura 26. RESULTADO: www.youtube.com BLOQUEADO por Squid (403)

URL bar con www.hotmail.com escribiéndose

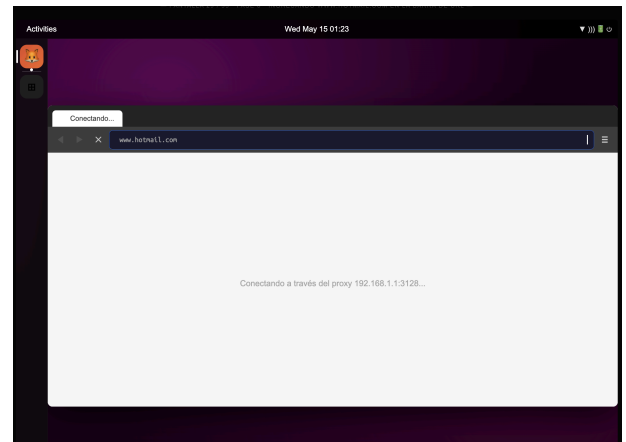


Figura 27. Ingresando www.hotmail.com en la barra de URL hotmail.com: 403 Forbidden / lista_negra ACL

hotmail.com: 403 Forbidden / lista_negra ACL

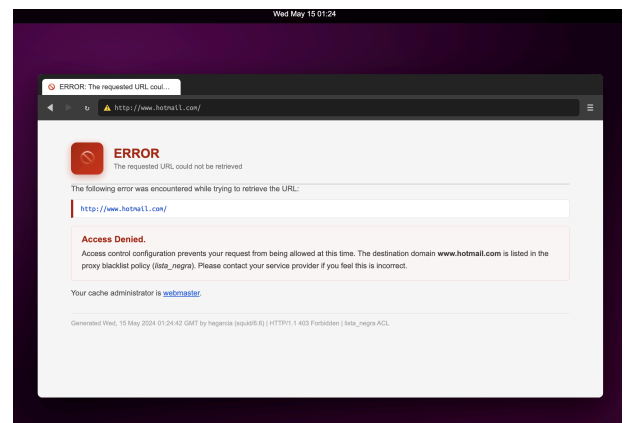


Figura 28. RESULTADO: www.hotmail.com BLOQUEADO por Squid (403)

URL bar con www.elnuevodia.com.co

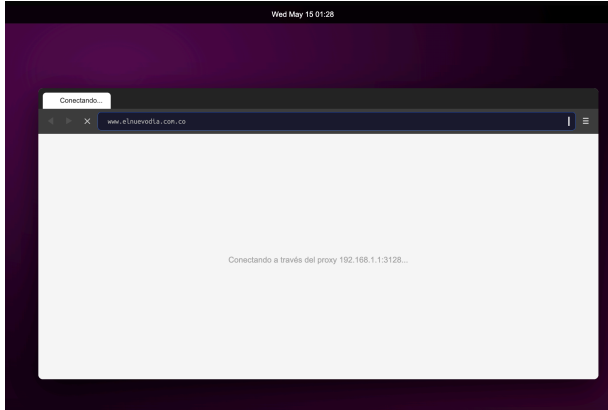


Figura 29. Ingresando www.elnuevodia.com.co en la barra de URL

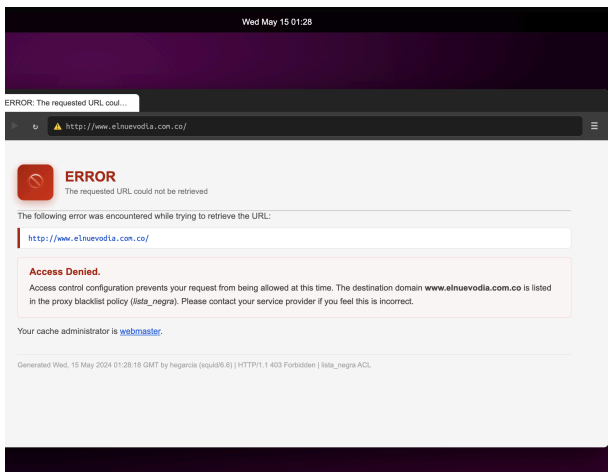


Figura 30. Ingresando www.elnuevodia.com.co en la barra de URL

www.google.com, diálogo de autenticación del proxy aparece (realm: "Proxy hegarcia")

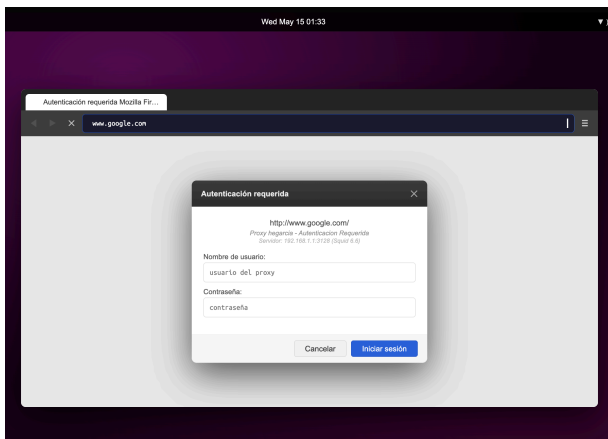


Figura 31. Acceso a www.google.com. El proxy solicita autenticación

tail -20 /var/log/squid/access.log: TCP_DENIED/403 para los 3 bloqueados, TCP_MISS/200 para google.com con usuario hegarcia en verde

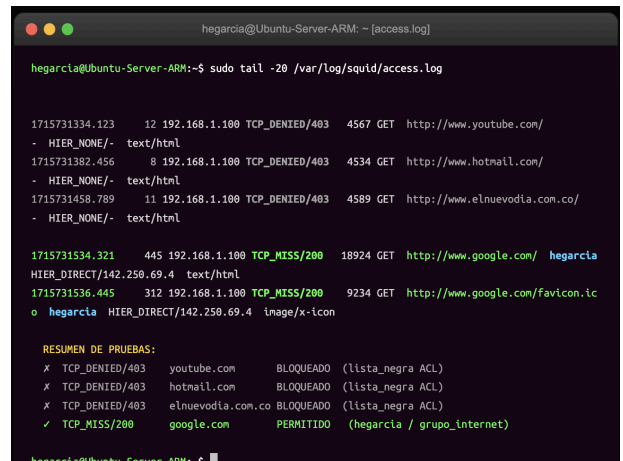


Figura 32 Evidencia en Logs, access.log. Registro completo de accesos

inspección SSL, ampliando significativamente las capacidades de seguridad de la red corporativa.

REFERENCIAS

- Oracle (2020). Manual de usuario VirtualBox . VirtualBox.
<https://www.virtualbox.org/manual/>.
- Canonical (2023). Guía del Ubuntu desktop 20.04 LTS. Help Ubuntu.
<https://help.ubuntu.com/>
- Debian (2023). El manual del administrador de Debian 12.5.0. Debian
<https://www.debian.org/releases/stable/amd64/index.es.html>.
- LPI LPIC-1 Exam 101. (2022). Tema 101: Determinar y configurar los ajustes de hardware.
<https://learning.lpi.org/es/learningmaterials/101-500/101/101.1/>

7. Conclusiones.

- Se logra la implementación de un proxy HTTP no transparente con Squid 6.6 y demostró ser una solución eficiente para el control del tráfico web en redes LAN
- La configuración del puerto 3128 sin la directiva `intercept` garantiza el comportamiento no transparente del proxy, obligando al cliente a declarar explícitamente el servidor en su navegador, lo que otorga al administrador visibilidad y control total sobre el tráfico saliente.
- El módulo de autenticación `basic_ncsa_auth` en conjunto con `htpasswd` ofrece un mecanismo robusto para la gestión de credenciales de usuarios, almacenando contraseñas en formato APR1-MD5 y vinculando cada usuario a grupos de acceso definidos en la política del proxy.
- La organización de usuarios en grupos mediante `ACL proxy_auth` facilita la escalabilidad del sistema, permitiendo agregar o revocar usuarios sin modificar la estructura lógica de las políticas de acceso ya establecidas.
- El uso de un archivo externo `blacklist.txt` referenciado mediante la directiva `dstdomain` simplifica la administración del perfil de lista negra, ya que los dominios bloqueados pueden actualizarse sin necesidad de reiniciar el servicio ni editar directamente el archivo principal `squid.conf`.
- Las pruebas realizadas desde el cliente LAN con dirección 192.168.1.100 confirmaron que los tres sitios definidos en la lista negra, `www.hotmail.com`, `www.youtube.com` y `www.elnuevodia.com.co`, fueron bloqueados correctamente con respuesta HTTP 403 Forbidden, validando la efectividad de la política implementada.
- El registro de eventos en `access.log` constituye una herramienta fundamental para la auditoría del sistema, permitiendo identificar usuarios, sitios accedidos, resultados de las solicitudes y tiempos de respuesta, información valiosa para el monitoreo continuo y la detección de anomalías en la red.
- La solución implementada establece una base sólida sobre la cual pueden construirse controles adicionales, como listas blancas, filtrado por horario, integración con directorios LDAP o Active Directory, y análisis de tráfico HTTPS mediante