
RA5. Implantació de servidors proxy

Índex

0. Escenari de pràctiques	1
1. Tipus de proxy: característiques i funcions	1
1.1 Proxy directe (forward proxy)	1
1.2 Proxy invers (reverse proxy)	1
1.3 Proxy transparent	2
1.4 Proxy obert (open proxy)	2
1.5 Altres classificacions útils	2
2. Instal·lació d'un servidor proxy-cache (Squid)	2
2.1 Instal·lació	2
2.2 Configuració bàsica	3
2.3 Configuració de la memòria cau	3
3. Configuració de clients proxy	4
3.1 Configuració manual	4
3.2 Configuració automàtica (PAC/WPAD)	4
4. Autenticació al proxy	5
4.1 Autenticació Basic amb fitxer local	5
4.2 Autenticació integrada amb Active Directory (Kerberos/NTLM)	5
5. Proxy en mode transparent	6
5.1 Configuració de Squid	6
5.2 Redirecció amb iptables (si srv-proxy fa també de router/gateway)	6
5.3 Limitacions importants	7
6. Restriccions d'accés web (filtres i ACL)	7
6.1 ACL per domini o paraula clau	7
6.2 Llistes de dominis des de fitxer	7
6.3 Restriccions per horari	7
6.4 Restriccions per tipus de fitxer (MIME/extensió)	8
6.5 Filtratge avançat amb SquidGuard/e2guardian	8
7. Resolució de problemes d'accés client-proxy	8
7.1 Comprovar connectivitat bàsica	8
7.2 Comprovar que Squid escolta	8
7.3 Revisar logs	9
7.4 Errors típics i causes	9
7.5 Validar sintaxi abans de reiniciar	9
8. Proves de funcionament i monitoratge gràfic	10
8.1 Proves funcionals bàsiques	10
8.2 Eina de gestió squidclient	10
8.3 Monitoratge amb SARG (Squid Analysis Report Generator)	10
8.4 Monitoratge en temps real amb Lightsquid o Zabbix	10
9. Proxy en mode invers (reverse proxy)	11
9.1 Squid com a accelerator	11

9.2 Alternativa amb Nginx	11
9.3 Avantatges d'un reverse proxy	11
10. Documentació del servei	12
11. Resum d'ordres	12

Cicle formatiu: Administració de sistemes informàtics en xarxa (ASIX)

Mòdul: 0378. Seguretat i alta disponibilitat

0. Escenari de pràctiques

Farem servir un escenari típic de laboratori, semblant al de `thos.local`:

Màquina	IP	Rol
srv-proxy (Debian 12)	192.168.10.10	Servidor Squid
pc-client01	192.168.10.50	Client Windows/Linux
srv-web01	192.168.10.20	Servidor web intern (destinació del reverse proxy)
Xarxa interna	192.168.10.0/24	LAN de l'aula

NOTA

Adapteu les IP i noms d'interfície (`ens18`, `eth0...`) al vostre entorn real abans d'executar les ordres.

1. Tipus de proxy: característiques i funcions

Un **servidor proxy** és un intermediari entre un client i un servidor de destinació que intercepta, filtra o modifica les peticions i respostes. Segons la seva posició i funció, distingim:

1.1 Proxy directe (forward proxy)

- Es col·loca al costat del **client**.
- El client sap que hi ha un proxy i hi envia explícitament les peticions (configuració manual, PAC o WPAD).
- Funcions típiques: **cau** de continguts, **filtratge** de contingut, **control d'accés**, ocultació de l'origen (anonimat parcial), registre d'activitat.
- Exemple: Squid configurat perquè tots els PC de l'aula surtin a Internet a través seu.

1.2 Proxy invers (reverse proxy)

- Es col·loca al costat del **servidor**.
- El client no sap que existeix: creu que parla directament amb el servidor final.
- Funcions típiques: **balanceig de càrrega**, terminació SSL/TLS, cau de contingut estàtic, protecció del servidor real (oculta la seva IP i versió), publicació de diversos serveis interns sota un mateix domini/IP pública.
- Exemples: HAProxy, Nginx, Squid en mode `accele`.

1.3 Proxy transparent

- És un proxy directe, però el client **no el configura**: el trànsit s'intercepta a escala de xarxa (tallafocs/router) i es redirigeix cap al proxy sense que l'usuari en sigui conscient.
- Avantatge: no cal tocar la configuració de cada client.
- Inconvenient: no es pot fer autenticació d'usuari fàcilment (el client no sap que hi ha un proxy) i no funciona de forma nativa amb HTTPS sense una capa addicional (SSL bump).

1.4 Proxy obert (open proxy)

- Accepta connexions de qualsevol origen sense restricció. **No recomanable** en producció: es converteix en un vector d'abús (spam, atacs, evasió de filtres).

1.5 Altres classificacions útils

Criteri	Tipus
Anonimat	Transparent / Anònim / Elit (no revela ni que hi ha proxy)
Protocol	HTTP/HTTPS proxy, SOCKS proxy, DNS proxy
Funció principal	Proxy-cache, proxy de filtratge de contingut, proxy d'autenticació, WAF/reverse proxy

2. Instal·lació d'un servidor proxy-cache (Squid)

Farem servir **Squid**, el proxy-cache de referència en entorns Linux.

2.1 Instal·lació

```
sudo apt update
sudo apt install -y squid squid-common
```

Serveis i fitxers principals:

Element	Ruta
Configuració principal	/etc/squid/squid.conf
Cau en disc	/var/spool/squid
Logs d'accés	/var/log/squid/access.log
Logs de cau	/var/log/squid/cache.log
Binari de gestió	squid -k reconfigure / systemctl

2.2 Configuració bàsica

Fem una còpia de seguretat abans de tocar res:

```
sudo cp /etc/squid/squid.conf /etc/squid/squid.conf.orig
```

Configuració mínima funcional (/etc/squid/squid.conf):

```
# Port d'escolta del proxy
http_port 3128

# Xarxes permeses
acl xarxa_local src 192.168.10.0/24
http_access allow xarxa_local
http_access deny all

# Identificació del cache
visible_hostname srv-proxy.thos.local
cache_mgr admin@thos.local
```

IMPORTANT

L'ordre de les regles `http_access` importa: Squid avalua de dalt a baix i s'atura a la primera que coincideix. Per això `deny all` sempre ha d'anar **al final**.

2.3 Configuració de la memòria cau

```
# Cau en memòria RAM (objectes petits, molt consultats)
cache_mem 256 MB

# Cau en disc: tipus ufs, ruta, mida total (MB), nivells de directoris
cache_dir ufs /var/spool/squid 10000 16 256

# Mida màxima/mínima d'objecte que es guarda a la cau
maximum_object_size 100 MB
minimum_object_size 0 KB

# Temps que es manté un objecte encara que el servidor no doni pistes
↳ de caducitat
refresh_pattern ^ftp:          1440      20%      10080
refresh_pattern ^gopher:       1440      0%       1440
refresh_pattern -i (/cgi-bin/|\?) 0        0%        0
refresh_pattern \.(deb|rpm|iso)$ 10080    90%     43200
refresh_pattern .               0        20%     4320
```

Inicialitzem l'estructura de directoris de la cau (obligatori la primera vegada):

```
sudo squid -z
sudo systemctl restart squid
```

```
sudo systemctl enable squid
```

Comprovació d'estat:

```
sudo systemctl status squid  
sudo tail -f /var/log/squid/cache.log
```

3. Configuració de clients proxy

3.1 Configuració manual

- **Linux (variables d'entorn):**

```
export http_proxy="http://192.168.10.10:3128"  
export https_proxy="http://192.168.10.10:3128"
```

- **Navegador (Firefox/Chrome):** Configuració manual del proxy → HTTP Proxy 192.168.10.10, port 3128.
- **Windows:** *Configuració* → *Xarxa i Internet* → *Proxy* → activar "Utilitza un servidor proxy" i indicar IP i port, o bé via GPO al domini THOS.LOCAL.

3.2 Configuració automàtica (PAC/WPAD)

Fitxer proxy.pac servit per un web intern:

```
function FindProxyForURL(url, host) {  
    if (isInNet(host, "192.168.10.0", "255.255.255.0")) {  
        return "DIRECT";  
    }  
    return "PROXY 192.168.10.10:3128; DIRECT";  
}
```

Amb **WPAD**, el client descobreix automàticament aquest fitxer mitjançant DNS (wpad.thos.local) o DHCP (opció 252), sense haver de distribuir l'URL manualment a cada equip.

4. Autenticació al proxy

Squid admet diversos *helpers* d'autenticació. Els més habituals:

Mètode	Helper	Ús típic
Basic	basic_ncsa_auth	Fitxer local d'usuaris/contrasenyes
Digest	digest_file_auth	Similar a Basic però sense enviar la contrasenya en clar
NTLM/Negotiate	ntlm_auth / negotiate_kerberos_auth	Integració amb Active Directory (SSO)

4.1 Autenticació Basic amb fitxer local

```
sudo apt install -y apache2-utils
sudo htpasswd -c /etc/squid/passwd alumne1
sudo htpasswd /etc/squid/passwd alumne2
```

A squid.conf:

```
auth_param basic program /usr/lib/squid/basic_ncsa_auth
↪ /etc/squid/passwd
auth_param basic realm Proxy THOS.LOCAL
auth_param basic credentialsttl 2 hours

acl usuarios_autenticats proxy_auth REQUIRED
http_access allow xarxa_local usuarios_autenticats
http_access deny all
```

IMPORTANT

Basic envia la contrasenya codificada en Base64, **no xifrada**. Cal combinar-lo amb HTTPS cap al proxy (via `https_port`) si es vol protegir les credencials, o utilitzar-lo només en xarxes internes de confiança.

4.2 Autenticació integrada amb Active Directory (Kerberos/NTLM)

Si `srv-proxy` és membre del domini `THOS.LOCAL` (via Samba/winbind o sssd):

```
sudo apt install -y winbind libnss-winbind libpam-winbind smbclient
```

```
auth_param negotiate program /usr/lib/squid/negotiate_wrapper_auth
↪ --ntlm /usr/lib/squid/ntlm_auth --kerberos
↪ /usr/lib/squid/negotiate_kerberos_auth
auth_param negotiate children 10
auth_param negotiate keep_alive on
```

```
acl usuarios_ad proxy_auth REQUIRED
http_access allow xarxa_local usuarios_ad
```

Amb aquesta configuració els usuaris del domini s'autentiquen de manera transparent des de navegadors configurats per a **Negotiate/Kerberos**, sense haver d'introduir cap contrasenya (SSO).

5. Proxy en mode transparent

En mode transparent, el trànsit HTTP es desvia cap al proxy a escala de tallafores, sense configurar res al client.

5.1 Configuració de Squid

```
http_port 3129 intercept

acl xarxa_local src 192.168.10.0/24
http_access allow xarxa_local
http_access deny all
```

NOTA

El mode intercept requereix que el paquet arribi a Squid **ja redirigit**, normalment mitjançant iptables/nftables a la mateixa màquina o a un router/tallafores intermedi.

5.2 Redirecció amb iptables (si srv-proxy fa també de router/gateway)

```
sudo sysctl -w net.ipv4.ip_forward=1

sudo iptables -t nat -A PREROUTING -i eth1 -p tcp --dport 80 \
-j REDIRECT --to-port 3129
```

Persistir la regla:

```
sudo apt install -y iptables-persistent
sudo netfilter-persistent save
```

5.3 Limitacions importants

- El mode transparent **només intercepta HTTP (port 80)** de forma senzilla; HTTPS (443) requereix desxifrar el trànsit (*SSL Bump*), la qual cosa implica desplegar un certificat propi de CA als clients --- una tècnica intrusiva que cal justificar i documentar bé en un entorn real.
- No es pot combinar de forma nativa amb autenticació d'usuari (el client no sap que hi ha un proxy, per tant, no li pot demanar credencials interactivament).

6. Restriccions d'accés web (filtres i ACL)

6.1 ACL per domini o paraula clau

```
acl dominis_bloquejats dstdomain .facebook.com .instagram.com  
↪ .tiktok.com  
acl paraules_bloquejades url_regex -i "porn|casino"  
  
http_access deny dominis_bloquejats  
http_access deny paraules_bloquejades
```

6.2 Llistes de dominis des de fitxer

```
acl llista_negra dstdomain "/etc/squid/blacklist.txt"  
http_access deny llista_negra
```

```
sudo nano /etc/squid/blacklist.txt
```

```
.facebook.com  
.tiktok.com  
.betfair.com
```

6.3 Restriccions per horari

```
acl horari_lectiu time MTWHF 08:00-15:00  
acl xarxa_alumnes src 192.168.20.0/24  
  
http_access deny xarxa_alumnes !horari_lectiu
```

6.4 Restriccions per tipus de fitxer (MIME/extensió)

```
acl descarregues_prohibides urlpath_regex -i \.(exe|bat|msi)$  
http_access deny descarregues_prohibides
```

6.5 Filtratge avançat amb SquidGuard/e2guardian

Per a categories completes (adults, xarxes socials, jocs...) es recomana un motor especialitzat:

```
sudo apt install -y squidguard
```

SquidGuard es crida des de Squid com a redirector (`url_rewrite_program`) i treballa amb bases de dades de categories de dominis descarregables (blacklists públiques o comercials), oferint un manteniment molt més escalable que llistes manuals.

7. Resolució de problemes d'accés client-proxy

Metodologia de diagnòstic esglaonada:

7.1 Comprovar connectivitat bàsica

```
ping 192.168.10.10  
telnet 192.168.10.10 3128
```

Si `telnet` no connecta: problema de xarxa, tallafocs o Squid aturat.

7.2 Comprovar que Squid escolta

```
sudo ss -tlnp | grep squid  
sudo systemctl status squid
```

7.3 Revisar logs

```
sudo tail -f /var/log/squid/access.log
sudo tail -f /var/log/squid/cache.log
```

Codis de resultat habituals a access.log:

Codi	Significat
TCP_HIT	Servit des de cau
TCP_MISS	No estava a la cau, es va anar a buscar
TCP_DENIED	Bloquejat per una ACL
NONE/407	Falta autenticació

7.4 Errors típics i causes

Síntoma	Causa probable
"Access Denied" (TCP_DENIED)	ACL massa restrictiva o mal ordenada (http_access deny all abans d'hora)
El client no arriba mai al proxy	Configuració incorrecta al navegador, o WPAD no resol
Funciona en HTTP però no en HTTPS	Cal https_port amb SSL Bump, o el mode transparent no cobreix el 443
Autenticació demanada en bucle	Helper d'autenticació no arrenca (ps aux grep squid) o permisos incorrectes al fitxer passwd
Molt lent	Cau plena, disc ple (df -h /var/spool/squid), o refresh_pattern mal ajustat

7.5 Validar sintaxi abans de reiniciar

```
sudo squid -k parse
sudo squid -k reconfigure
```

reconfigure aplica canvis sense tallar les connexions actives; restart és més dràstic i s'ha de reservar per a canvis estructurals (com cache_dir).

8. Proves de funcionament i monitoratge gràfic

8.1 Proves funcionals bàsiques

```
curl -x http://192.168.10.10:3128 http://example.com
curl -x http://usuari:contrasenya@192.168.10.10:3128
  ↪ http://example.com
```

8.2 Eina de gestió squidclient

```
squidclient -h 192.168.10.10 mgr:info
squidclient -h 192.168.10.10 mgr:menu
```

8.3 Monitoratge amb SARG (Squid Analysis Report Generator)

```
sudo apt install -y sarg
```

Configuració mínima (/etc/sarg/sarg.conf):

```
access_log /var/log/squid/access.log
output_dir /var/www/html/sarg
```

Generació d'informes:

```
sudo sarg -x
```

Els informes es publiquen com a pàgines HTML amb gràfics de: usuaris amb més trànsit, dominis més visitats, hores de més ús, i percentatge de trànsit servit des de cau.

8.4 Monitoratge en temps real amb Lightsquid o Zabbix

- **Lightsquid**: alternativa lleugera a SARG, també basada en informes HTML generats periòdicament (cron).
- **Zabbix** (ja desplegat a `thos.local`): es pot afegir un *template* de Squid via SNMP o mitjançant un script que llegeixi `mgr:info` i `mgr:5min` i publiqui els valors com a *items* (peticions/segon, ràtio de cau, connexions actives), integrant el proxy als mateixos dashboards que la resta d'infraestructura.

9. Proxy en mode invers (reverse proxy)

9.1 Squid com a *accelerator*

```
http_port 80 accel defaultsite=intranet.thos.local

acl backend_web dstdomain intranet.thos.local
cache_peer 192.168.10.20 parent 8080 0 no-query originserver
  ↪ name=web01

http_access allow backend_web
```

Amb aquesta configuració, les peticions que arriben a Squid pel port 80 amb Host: intranet.thos.local es reenvien internament a 192.168.10.20:8080, mentre el client només veu la IP pública/del proxy.

9.2 Alternativa amb Nginx

Molt habitual en producció per la seva senzillesa i rendiment amb TLS:

```
server {
    listen 443 ssl;
    server_name intranet.thos.local;

    ssl_certificate      /etc/ssl/certs/intranet.crt;
    ssl_certificate_key  /etc/ssl/private/intranet.key;

    location / {
        proxy_pass http://192.168.10.20:8080;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }
}
```

9.3 Avantatges d'un reverse proxy

- Un únic punt d'entrada i de terminació TLS per a diversos serveis interns.
- El servidor real queda ocult (protecció addicional davant escanejos i atacs directes).
- Es pot afegir cau, compressió i limitació de peticions (*rate limiting*) sense tocar l'aplicació.
- Facilita el **balanceig de càrrega** si s'afegeixen diversos cache_peer/upstream.

10. Documentació del servei

Tota implantació de proxy s'ha d'acompanyar de documentació tècnica que permeti a un altre administrador mantenir-la. Com a mínim ha d'incloure:

1. **Esquema de xarxa:** ubicació del proxy, xarxes que serveix, ports oberts.
2. **Inventari de configuració:** versió de Squid, ruta dels fitxers de configuració, mida de la cau, mètode d'autenticació utilitzat.
3. **Polítiques aplicades:** llistat d'ACL amb la justificació de cada restricció (per què es bloqueja tal domini o horari).
4. **Procediment d'alta d'usuari:** com afegir un usuari nou a l'autenticació (Basic o AD).
5. **Procediment de resolució d'incidències:** els passos de l'apartat 7 d'aquest document, adaptats a l'entorn concret.
6. **Registre de canvis:** data, autor i motiu de cada modificació rellevant a `squid.conf`.
7. **Captures i informes de monitoratge:** exemple d'informe SARG/Zabbix, amb data de generació.

11. Resum d'ordres

```
# Instal·lació
sudo apt install -y squid squid-common apache2-utils sarg squidguard

# Inicialitzar cau
sudo squid -z

# Validar i aplicar configuració
sudo squid -k parse
sudo squid -k reconfigure
sudo systemctl restart squid

# Diagnòstic
sudo systemctl status squid
sudo ss -tlnp | grep squid
sudo tail -f /var/log/squid/access.log
squidclient -h <IP> mgr:info

# Usuaris d'autenticació Basic
sudo htpasswd -c /etc/squid/passwd usuari1
sudo htpasswd /etc/squid/passwd usuari2

# Informes de trànsit
sudo sarg -x
```

Versions d'aquest document

- [HTML](#)
- [PDF](#)
- [ODT](#)
- [MD](#)

[Domini Públic \(CC0\)](#)