

INSTALLATION DE SQUID SOUS RASPBERRY PI

Raspberry Pi – Debian 12
Configuration de base

Tutoriel **SQUID** - RASPBERRY PI
Décembre 2024

David GOÏTRÉ

Table des matières

Introduction	1
1. Pré requis	1
2. Paramétrage de connexion au serveur	1
3. Paramétrage du serveur	2
4. Installation de Squid.....	3
5. Configuration l'authentification basée sur IP	3
6. Configuration de Squid pour anonymiser le trafic.....	4
7. Intégrité de la configuration de squid.....	5
8. Mise à jour de Squid.....	5
9. Désinstallation de Squid.....	5
10. Paramétrage du proxy dans le navigateur	5
11. Liens annexes	6
12. Commandes RaspberryPi	6
13. Conclusion	6

Introduction

Un serveur **Squid** est un serveur mandataire (proxy) et un **mandataire inverse** conçu pour relayer les protocoles FTP, HTTP, Gopher, et HTTPS.

Contrairement aux serveurs proxy classiques, un serveur Squid gère toutes les requêtes en un seul processus d'entrée/sortie asynchrone. C'est un logiciel libre distribué sous licence GNU GPL.

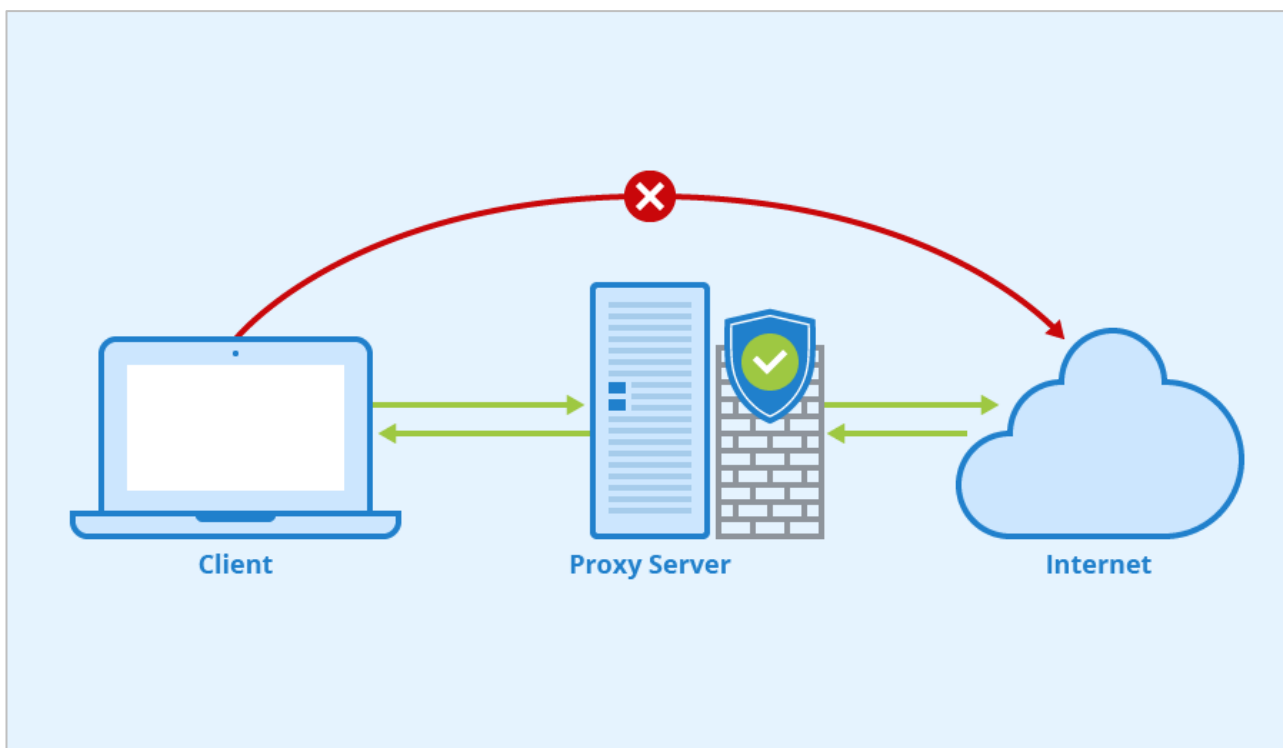
Squid garde les métadonnées et plus particulièrement les données les plus fréquemment utilisées en mémoire. Il conserve aussi en mémoire les requêtes DNS, ainsi que les requêtes ayant échoué. Les requêtes DNS sont non bloquantes.

1. Pré requis

On a besoin des différents matériels et logiciels pour la création d'un Serveur Squid avec un RaspberryPi.

- Un ou des PC client sous Windows
- Une Box (Free, Orange, Sfr...)
- Un Raspberry 3B+ avec [l'OS RaspbianOS](#) installé avec [Etcher](#)
- Le logiciel [Putty](#) pour se connecter en SSH au serveur
- Connaître l'interface réseau (eth0, br0, ens3...) via la commande : `ip a`
Pour notre test c'est **l'interface eth0** qui sera utilisée

Voici un exemple de ce que l'on doit obtenir une fois le serveur **Squid** mise en place



2. Paramétrage de connexion au serveur

a) Le **SSH** est activé par défaut sur le serveur

b) Ouvrir **Putty** et se connecter au serveur avec les identifiants (par défaut **pi/raspberrypi**)

c) Exécuter la commande suivante pour mettre à jour et mettre à niveau les packages du système

```
# apt update && apt upgrade
```

3. Paramétrage du serveur

Avant d'aller plus loin, il nous faut connaître l'interface réseau de notre serveur **RaspberryPI** et lui attribuer une adresse IP fixe.

a) Lister les interfaces

```
# ip a # liste les interfaces
# grep -i ethernet /var/run/dmesg.boot # liste les propriétés de l'interface
```

b) Définir une adresse IP fixe

```
# apt install nano # installe le logiciel nano
# nano /etc/network/interfaces # ouvre le fichier des interfaces
```

c) Copier le texte ci-dessous dans le fichier **/etc/network/interface**

```
# Interface reseau de bouclage
auto lo
iface lo inet loopback
# Interface reseau principale
allow-hotplug eth0
iface eth0 inet static
address 192.xxx.xxx.xxx
netmask 255.255.255.0
gateway 192.xxx.xxx.xxx
```

d) Vérifier les DNS

```
# cat /etc/resolv.conf # affiche le contenu du fichier
```

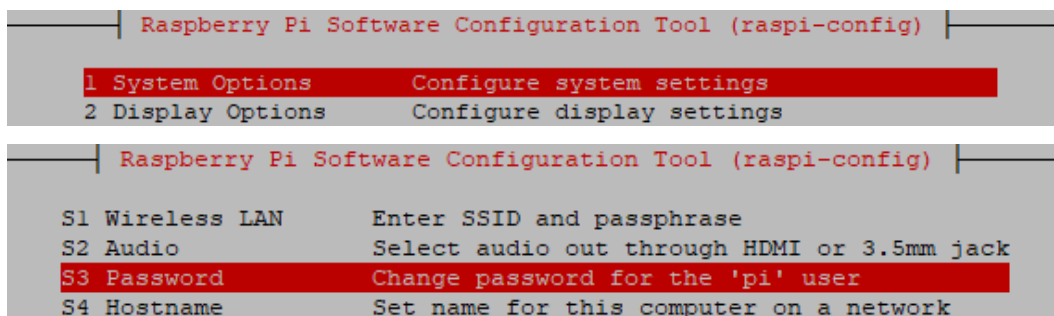
e) Redémarrer le serveur

```
# reboot
```

f) Modifier le mot de passe

```
$ raspi-config # ouvre l'utilitaire
```

Sélectionner le menu **System Options**, puis cliquer sur le menu **S3 Password** pour modifier le mot de passe et **S4 Hostname** pour modifier le nom du serveur.



```
Raspberry Pi Software Configuration Tool (raspi-config)
1 System Options          Configure system settings
2 Display Options         Configure display settings

Raspberry Pi Software Configuration Tool (raspi-config)
S1 Wireless LAN          Enter SSID and passphrase
S2 Audio                 Select audio out through HDMI or 3.5mm jack
S3 Password              Change password for the 'pi' user
S4 Hostname              Set name for this computer on a network
```

Sélectionner le menu **Localisation Options**. Dans la **liste déroulante** sélectionner **fr_FR UTF-8** pour ajouter les paramètres linguistiques français au système.

4. Installation de Squid

Par défaut, le paquet Squid n'est pas disponible dans le référentiel de Debian. Il faut ajouter le repository pour effectuer l'installation

a) Installation de squid

```
# apt update
# apt install squid -y
```

b) Vérification l'état du service Squid

```
# systemctl status squid
```

Résultat de la sortie

```
kungen@raspberrypi:~ $ systemctl status squid
● squid.service - Squid Web Proxy Server
   Loaded: loaded (/lib/systemd/system/squid.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-12-28 12:26:07 GMT; 17min ago
     Docs: man:squid(8)
   Process: 7266 ExecStartPre=/usr/sbin/squid --foreground -z (code=exited, status=0/SUCCESS)
  Main PID: 7270 (squid)
    Tasks: 5 (limit: 763)
       CPU: 3.810s
   CGroup: /system.slice/squid.service
           └─7270 /usr/sbin/squid --foreground -sYC
             └─7272 "(squid-1)" --kid squid-1 --foreground -sYC
               └─7273 "(logfile-daemon)" /var/log/squid/access.log
                 └─7274 "(unlinkd)"
                   └─7275 "(pinger)"

Dec 28 12:26:07 raspberrypi squid[7272]:      0 Objects expired.
Dec 28 12:26:07 raspberrypi squid[7272]:      0 Objects cancelled.
Dec 28 12:26:07 raspberrypi squid[7272]:      0 Duplicate URLs purged.
Dec 28 12:26:07 raspberrypi squid[7272]:      0 Swapfile clashes avoided.
Dec 28 12:26:07 raspberrypi squid[7272]: Took 0.19 seconds (67978.84 objects/sec).
Dec 28 12:26:07 raspberrypi squid[7272]: Beginning Validation Procedure
Dec 28 12:26:07 raspberrypi squid[7272]: Completed Validation Procedure
Dec 28 12:26:07 raspberrypi squid[7272]: Validated 12579 Entries
Dec 28 12:26:07 raspberrypi squid[7272]: store_swap_size = 329672.00 KB
Dec 28 12:26:08 raspberrypi squid[7272]: storeLateRelease: released 0 objects
kungen@raspberrypi:~ $
```

c) Vérification du port d'écoute 3128 de Squid

```
# netstat -plunt | grep 3128
```

Résultat de la sortie

```
tcp60      0      :::3128      :::*      LISTEN      50017/(squid-1)
```

5. Configuration l'authentification basée sur IP

Il existe plusieurs façons de restreindre l'accès du client à Internet. Dans cette section, on va configurer Squid pour s'authentifier en fonction des clients.

a) Modifier le fichier de configuration `/etc/squid/squid.conf`

```
# nano /etc/squid/squid.conf
```

b) Ajout les lignes ci-dessous au début du fichier pour autoriser tout les clients du réseau local

```
acl localnet src 192.168.1.0/24
http_access allow localnet
```

c) Ajout les lignes ci-dessous au début du fichier pour autoriser les clients inscrits

```
acl client1 src 192.168.1.10
http_access allow 192.168.1.10
```

d) Redémarrer ensuite le service Squid pour appliquer les modifications

```
# systemctl restart squid
```

e) Démarrage automatique de Squid

```
# sudo systemctl enable squid
```

6. Configuration de Squid pour anonymiser le trafic

On doit ajouter des règles pour masquer les adresses IP des clients des serveurs qui reçoivent le trafic de notre proxy HTTP Squid.

a) Toujours en éditant le fichier **/etc/squid/squid.conf**

```
# nano /etc/squid/squid.conf
```

b) Ajout des lignes ci-dessous au début du fichier

```
forwarded_for off
request_header_access Allow allow all
request_header_access Authorization allow all
request_header_access WWW-Authenticate allow all
request_header_access Proxy-Authorization allow all
request_header_access Proxy-Authenticate allow all
request_header_access Cache-Control allow all
request_header_access Content-Encoding allow all
request_header_access Content-Length allow all
request_header_access Content-Type allow all
request_header_access Date allow all
request_header_access Expires allow all
request_header_access Host allow all
request_header_access If-Modified-Since allow all
request_header_access Last-Modified allow all
request_header_access Location allow all
request_header_access Pragma allow all
request_header_access Accept allow all
request_header_access Accept-Charset allow all
request_header_access Accept-Encoding allow all
request_header_access Accept-Language allow all
request_header_access Content-Language allow all
request_header_access Mime-Version allow all
request_header_access Retry-After allow all
request_header_access Title allow all
request_header_access Connection allow all
request_header_access Proxy-Connection allow all
request_header_access User-Agent allow all
request_header_access Cookie allow all
request_header_access All deny all
```

c) Redémarrer ensuite le service Squid pour appliquer les modifications

```
# systemctl restart squid
```

7. Intégrité de la configuration de squid

Vérifier l'intégrité du fichier de configuration

```
# systemctl status squid.service
```

8. Mise à jour de Squid

Effectuer la mise à jour manuellement

```
# apt update  
# apt upgrade
```

9. Désinstallation de Squid

Désinstaller Squid peut-être parfois nécessaire pour le réinstaller

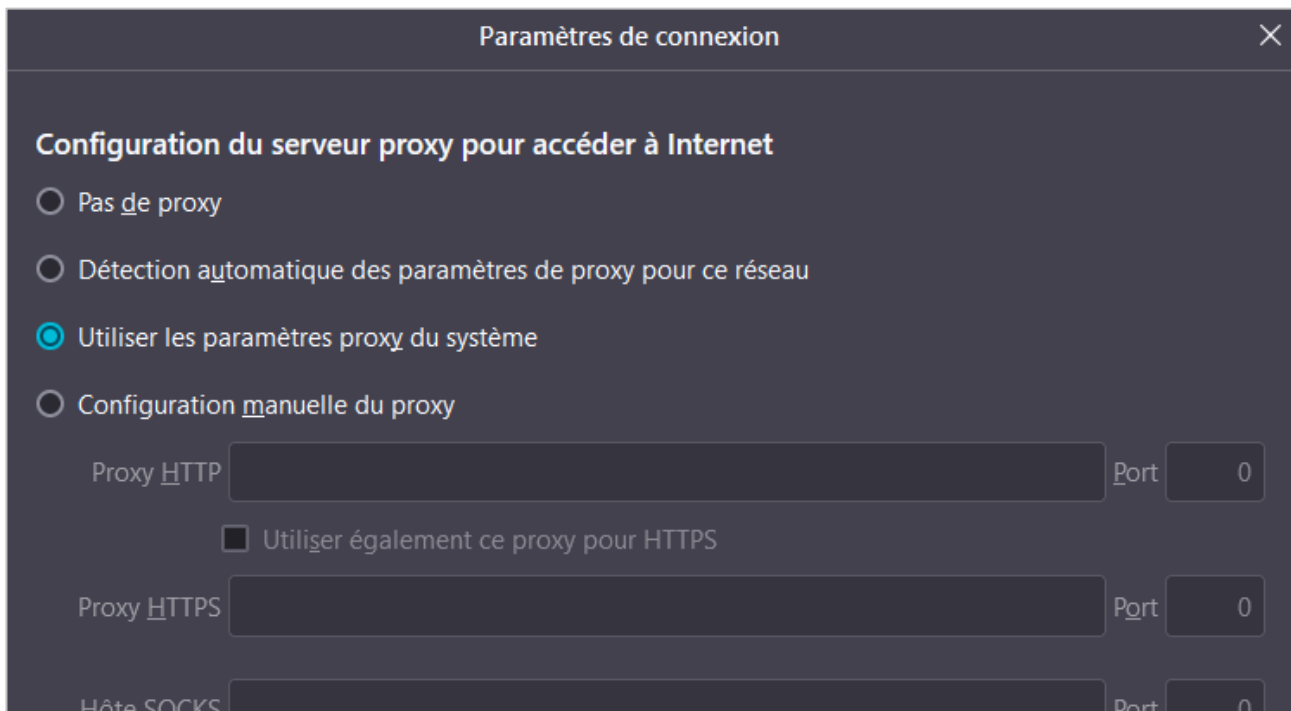
```
# systemctl stop squid  
# apt remove squid  
# reboot
```

10. Paramétrage du proxy dans le navigateur

a) Ouvrir le navigateur **Mozilla Firefox**

b) Sélectionner le menu **Modifier/Préférences**

c) En bas de page de la section **Paramètres réseaux**, cliquer sur le bouton **Paramètres...**



d) Si le proxy est **configuré dans Windows**, cocher la case **Utiliser les paramètres proxy du système**

e) Si le proxy n'est pas configuré dans Windows, cocher la case **Configuration manuelle du proxy** et saisir **l'@IP du serveur Squid** et le **port 3128** dans les 3 champs

11. Liens annexes

Liste de contenu à consulter ou à télécharger pour **Squid**

- Squid Ubuntu : <https://fr.linux-terminal.com>
- Squid Docker : <https://fariszr.com/squid-proxy>
- Documentation : <https://openclassrooms.com/fr/squid>

12. Commandes RaspberryPi

a) Liste des commandes basique à la gestion du serveur RaspberryPi

```
# shutdown -h now # éteint le serveur en toute sécurité
# shutdown -r now # redémarre le serveur en toute sécurité
# apt install xrdp # installe le bureau à distance RDP
# nano /etc/rc.conf -> service_enable=YES # active le service au démarrage
##### Désactive la mise en veille #####
# systemctl mask sleep.target suspend.target hibernate.target hybrid-sleep.target
```

13. Conclusion

Squid est installé et configuré avec succès sur le serveur **RaspberryPi Debian 12**. On peut désormais naviguer rapidement sur Internet.

Destiné au RaspberryPi (Raspbian), **Squid** fonctionne aussi parfaitement sur une distribution Ubuntu, CentOS, Windows, Docker...

Pour Linux CentOS : <https://www.linuxtricks.fr/wiki/centos>

Pour Linux RedHat : <https://docs.redhat.com/fr/documentation>