ETTORI Bastien	BTS SIO 2ème année
22 Mai 2017	Année scolaire : 2016/2017
Option : SISR	Version 2

SERVEUR PROXY SQUID DEBIAN

SOMMAIRE :

I) Objectif

Dans cette procédure, nous allons montrer comment installer et configurer un serveur **Proxy SQUID** sous Debian.

II) Prérequis

Pour réaliser cette procédure, nous avons besoin des éléments suivants :

OS	Distribution	Version	C/S
Debian Jessie	Linux	8.5	**S**

Nom du serveur SQUID	Adresse IP du serveur Proxy SQUID	Adresse IP du poste client
SQUID	192.168.1.132	192.168.1.74

III) Définition

Un serveur **Proxy** est un composant logiciel informatique qui permet de surveiller les échanges entre 2 hôtes ainsi que de mettre en cache et filtrer des données.

IV) Installation du service « squid3 »

- Tout d'abord, nous mettons à jour les paquets :

```
root@SQUID:~# apt-get update
```

- Nous installons le service « **squid3** » :

```
root@SQUID:~# apt-get install squid3
```

- Pour vérifier le port d'écoute par défaut de **SQUID**, nous nous nous rendons dans le fichier de configuration « **/etc/squid3/squid.conf** » (ici, le port est **3128**) :

```
# Squid normally listens to port 3128
http_port 3128
```

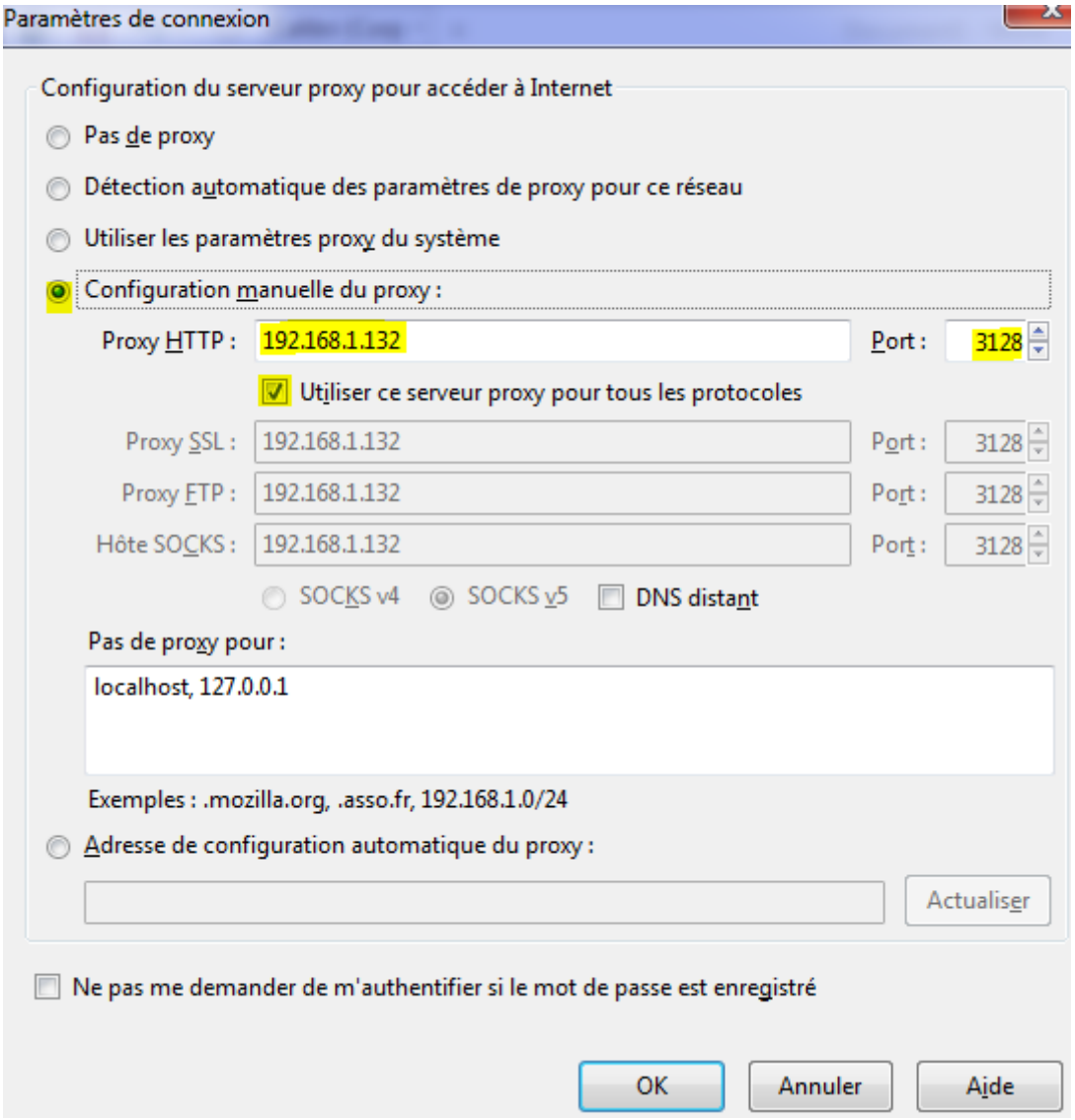
- Pour vérifier que l'utilisateur « **proxy** » appartient au groupe « **proxy** » créé, nous tapons les commandes suivantes et constatons que c'est le cas :

```
root@SQUID:~# cat /etc/passwd | grep proxy
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
```

```
root@SQUID:~# cat /etc/group | grep proxy
proxy:x:13:
```

V) Configuration de base du serveur Proxy

- Nous ouvrons un navigateur et paramétrons le **Proxy** dans les paramètres avancés d'Internet. Pour ce faire, nous allons dans « **Options Internet** », « **Paramètres réseau** », cochons la case « **Configuration manuelle du proxy** », saisissons l'adresse IP du serveur et son port et cochons la case « **Utiliser ce serveur proxy pour tous les protocoles** » :



- Maintenant, nous constatons que nous ne pouvons plus naviguer sur Internet :

- Nous consultons le fichier de log « **/var/log/squid3/access.log** » et remarquons que l'accès à Internet est interdit :

```
  GNU nano 2.2.6       Fichier : /var/log/squid3/access.log

1473664043.485      0 192.168.1.74 TCP_DENIED/403 3610 CONNECT www.google.fr
1473664043.488      0 192.168.1.74 TCP_DENIED/403 3610 CONNECT www.google.fr
1473664043.488      0 192.168.1.74 TCP_DENIED/403 3610 CONNECT www.google.fr
1473664043.488      0 192.168.1.74 TCP_DENIED/403 3610 CONNECT www.google.fr
```

- Nous créons une copie du fichier de configuration de **SQUID** avant de le modifier pour s'assurer du bon fonctionnement du **Proxy** pour la suite en cas d'erreurs :

```
root@SQUID:/etc/squid3# cp squid.conf squid.conf.back
root@SQUID:/etc/squid3# _
```

- Ensuite, nous enlevons toutes les lignes de commentaires du fichier :

```
root@SQUID:/etc/squid3# cat squid.conf.back | grep -v ^# | grep -v ^$ > squid.co
nf
root@SQUID:/etc/squid3# _
```

  - Le premier « **-v** » permet d'enlever les lignes de commentaire.
  - Le second « **-v** » permet d'enlever les lignes vides.
  - Cette partie de la commande « **> squid.conf** » permet de sauvegarder le résultat dans le fichier de configuration de **SQUID**.

- Nous ajoutons ces 4 lignes à la fin du fichier qui permettent à l'utilisateur « **proxy** » de lancer des requêtes sur le serveur, créer un emplacement de stockage et régler les niveaux :

```
cache_effective_user proxy
cache_effective_group proxy
cache_mem 16 Mb
cache_dir ufs /var/spool/squid3 120 16 128
```

- Si nous consultons à nouveau le fichier de logs de **SQUID**, nous constatons que nous avons toujours la même erreur :

```
  GNU nano 2.2.6       Fichier : /var/log/squid3/access.log

1473665186.795      0 192.168.1.74 TCP_DENIED/403 3610 CONNECT www.google.fr
1473665186.796      0 192.168.1.74 TCP_DENIED/403 3610 CONNECT www.google.fr
```

- Pour vérifier que le port du **Proxy** est bien en écoute, nous pouvons saisir les commandes suivantes et constatons que c'est le cas :

```
root@SQUID:~# lsof -i:3128
COMMAND  PID  USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
squid3  1467 proxy   11u  IPv6  13209      0t0  TCP *:3128 (LISTEN)
root@SQUID:~# _
```

```
root@SQUID:~# netstat -ltp
Connexions Internet actives (seulement serveurs)
Proto Recv-Q Send-Q Adresse locale          Adresse distante        Etat
PID/Program name
tcp        0      0 *:ssh                   *:*                     LISTEN
468/sshd
tcp        0      0 localhost:smtp          *:*                     LISTEN
734/exim4
tcp        0      0 *:36260                 *:*                     LISTEN
453/rpc.statd
tcp        0      0 *:sunrpc                *:*                     LISTEN
444/rpcbind
tcp6       0      0 [::]:ssh                [::]:*                  LISTEN
468/sshd
tcp6       0      0 [::]:3128               [::]:*                  LISTEN
1467/(squid-1)
```

VI) Contrôles d'accès

- Nous allons utiliser les **ACL** qui permettent de contrôler les permissions afin de vérifier que le noyau du serveur supporte les **ACL** (« **y** » =yes) :

```
root@SQUID:~# cat /boot/config-3.16.0-4-amd64 | grep ACL
CONFIG_EXT4_FS_POSIX_ACL=y
CONFIG_REISERFS_FS_POSIX_ACL=y
CONFIG_JFS_POSIX_ACL=y
CONFIG_XFS_POSIX_ACL=y
CONFIG_BTRFS_FS_POSIX_ACL=y
CONFIG_FS_POSIX_ACL=y
CONFIG_TMPFS_POSIX_ACL=y
# CONFIG_HFSPLUS_FS_POSIX_ACL is not set
CONFIG_JFFS2_FS_POSIX_ACL=y
CONFIG_F2FS_FS_POSIX_ACL=y
CONFIG_NFS_V3_ACL=y
CONFIG_NFSD_V2_ACL=y
CONFIG_NFSD_V3_ACL=y
CONFIG_NFS_ACL_SUPPORT=m
CONFIG_CEPH_FS_POSIX_ACL=y
CONFIG_CIFS_ACL=y
CONFIG_9P_FS_POSIX_ACL=y
root@SQUID:~#
```

- Nous vérifions la disponibilité des commandes avec les commandes (en jaune) avec leur description :

```
root@SQUID:~# setfacl -h
setfacl 2.2.52 -- définir les listes de contrôle d'accès des fichiers (ACL)
Utilisation : setfacl [-bkndRLP] { -m|-M|-x|-X ... } file ...
  -m, --modify=acl           modifier l'ACL(s) actuel de fichier(s)
  -M, --modify-file=fichier  lire l'entrée ACL à modifier du fichier
  -x, --remove=acl           supprimer les entrées de l'ACL des fichier
  -X, --remove-file=fichier  lire les entrées ACL à supprimer du fichier
  -b, --remove-all           supprimer toutes les entrées ACL étendues
  -k, --remove-default       supprimer l'ACL par défaut
      --set=acl           set the ACL of file(s), replacing the current ACL
      --set-file=file     read ACL entries to set from file
      --mask              do recalculate the effective rights mask
  -n, --no-mask           ne pas recalculer les masques de droits en vigueur
  -d, --default           les opérations s'appliquent à l'ACL par défaut
  -R, --recursive         parcourir récursivement les sous-répertoires
  -L, --logical           suivre les liens symboliques
  -P, --physical          ne pas suivre les liens symboliques
      --restore=fichier   restaurer les ACL (inverse de « getfacl -R »)
      --test              mode test (les ACL ne sont pas modifiés)
  -v, --version           print version and exit
  -h, --help              this help text
root@SQUID:~# _
```

```
root@SQUID:~# getfacl -h
getfacl 2.2.52 -- obtenir les listes de contrôle d'accès du fichier
Utilisation : getfacl [-aceEsRLPtpndvh] fichier...
  -a,  --access           display the file access control list only
  -d, --default           display the default access control list only
  -c, --omit-header       do not display the comment header
  -e, --all-effective     print all effective rights
  -E, --no-effective      print no effective rights
  -s, --skip-base         skip files that only have the base entries
  -R, --recursive         recurse into subdirectories
  -L, --logical           logical walk, follow symbolic links
  -P, --physical          physical walk, do not follow symbolic links
  -t, --tabular           use tabular output format
  -n, --numeric           print numeric user/group identifiers
  -p, --absolute-names    don't strip leading '/' in pathnames
  -v, --version           print version and exit
  -h, --help              this help text
root@SQUID:~# _
```

a) Autorisation d'accès au réseau local

- Nous éditons à nouveau le fichier de configuration et ajoutons les 2 lignes (en jaune) pour l'autorisation d'accès et de test à l'utilisateur du **Proxy** sur le réseau local :

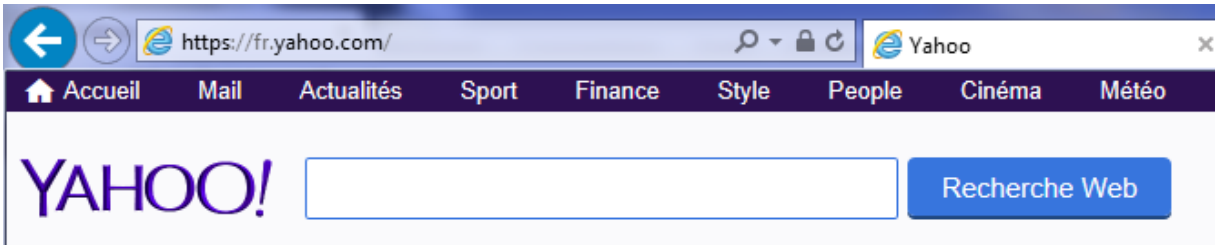
```
 GNU nano 2.2.6          Fichier : /etc/squid3/squid.conf

acl lan src 192.168.1.0/24
acl SSL_ports port 443
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443         # https
acl Safe_ports port 70          # gopher
acl Safe_ports port 210         # wais
acl Safe_ports port 1025-65535  # unregistered ports
acl Safe_ports port 280         # http-mgmt
acl Safe_ports port 488         # gss-http
acl Safe_ports port 591         # filemaker
acl Safe_ports port 777         # multiling http
acl CONNECT method CONNECT
http_access allow lan
http_access deny !Safe_ports
```

- Nous redémarrons le service « **squid3** » pour prendre en compte les modifications :

```
root@SQUID:~# systemctl restart squid3.service
root@SQUID:~# _
```

- Nous testons à nouveau la navigation d'Internet sur un site (par exemple : « https://fr.yahoo.com/ ») et constatons que l'accès est disponible :



- Nous consultons à nouveau les logs et constatons que « **TCP_DENIED** » a été modifié par « **TCP_MISS** » :

```
root@SQUID:~# tail /var/log/squid3/access.log
1473666947.431    265 192.168.1.74 TCP_MISS/200 4996 CONNECT ir2.beap.gemini.yah
oo.com:443 - HIER_DIRECT/188.125.66.82 -
1473666947.701   6264 192.168.1.74 TCP_MISS/200 599288 CONNECT s.yimg.com:443 -
HIER_DIRECT/66.196.65.111 -
1473666947.956    259 192.168.1.74 TCP_MISS/200 7042 CONNECT beap-bc.yahoo.com:4
43 - HIER_DIRECT/66.196.66.212 -
1473666948.622    921 192.168.1.74 TCP_MISS/200 7042 CONNECT beap-bc.yahoo.com:4
43 - HIER_DIRECT/66.196.66.212 -
1473666948.692   1602 192.168.1.74 TCP_MISS/200 6406 CONNECT geo.query.yahoo.com
:443 - HIER_DIRECT/98.138.243.53 -
1473666948.693    736 192.168.1.74 TCP_MISS/200 701 CONNECT pagead2.googlesyndic
ation.com:443 - HIER_DIRECT/216.58.198.226 -
1473666950.685   5518 192.168.1.74 TCP_MISS/200 3786 CONNECT ssp.adriver.ru:443
- HIER_DIRECT/195.209.111.7 -
1473666953.884  10477 192.168.1.74 TCP_MISS/200 3065 CONNECT secure-ams.adnxs.co
m:443 - HIER_DIRECT/37.252.163.218 -
1473666953.895  10488 192.168.1.74 TCP_MISS/200 3065 CONNECT secure-ams.adnxs.co
m:443 - HIER_DIRECT/37.252.163.218 -
1473666953.915  10509 192.168.1.74 TCP_MISS/200 3065 CONNECT secure-ams.adnxs.co
m:443 - HIER_DIRECT/37.252.163.218 -
root@SQUID:~# _
```

b) Accès horaire

- Pour restreindre les horaires d'accès aux clients, nous devons ajouter dans le fichier de configuration, l'adresse IP du (ou des) client(s), la plage horaire et la permission d'accès (soient les lignes en jaune) :

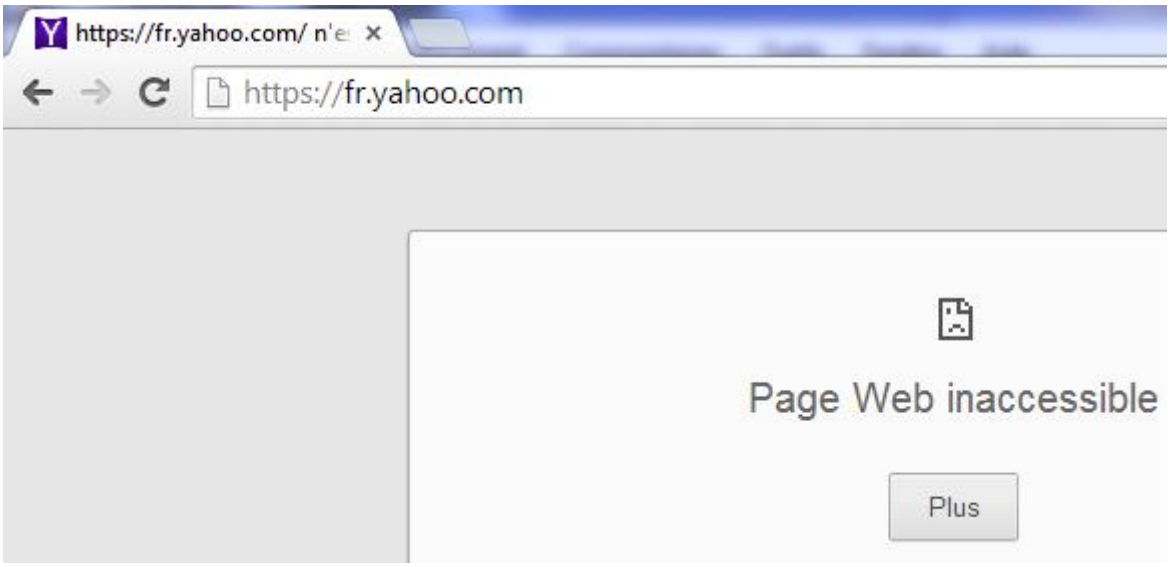
```
  GNU nano 2.2.6         Fichier : /etc/squid3/squid.conf

acl allowed_hosts src 192.168.1.74
acl lan time 11:15-12:00
#acl lan src 192.168.1.0/24
acl SSL_ports port 443
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443         # https
acl Safe_ports port 70          # gopher
acl Safe_ports port 210         # wais
acl Safe_ports port 1025-65535  # unregistered ports
acl Safe_ports port 280         # http-mgmt
acl Safe_ports port 488         # gss-http
acl Safe_ports port 591         # filemaker
acl Safe_ports port 777         # multiling http
acl CONNECT method CONNECT
http_access allow lan
http_access allow allowed_hosts lan
http_access deny !Safe_ports
```

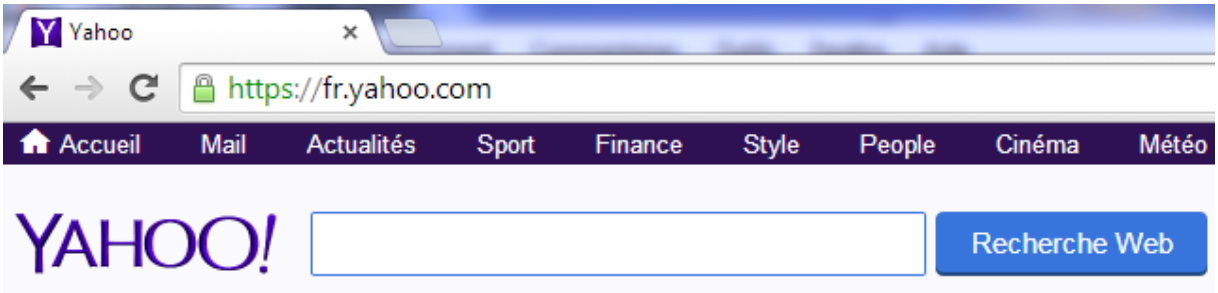
- Nous redémarrons le service « **squid3** » pour prendre en compte les modifications :

```
root@SQUID:~# systemctl restart squid3.service
root@SQUID:~# _
```

- Selon la plage horaire définie (« **acl lan time** » dans le fichier de configuration « **/etc/squid3/squid.conf** »), si l'accès n'est pas dans la plage horaire, l'utilisateur ne peut pas naviguer sur Internet :



- Sinon, il peut naviguer :



VII) Authentification des utilisateurs

- Nous allons créer 2 utilisateurs dans le fichier « **/etc/squid3/squidusers** » :

```
root@SQUID:/etc/squid3# touch squidusers
root@SQUID:/etc/squid3# _
```

- Si le paquet « **htpasswd** » n'est pas installé sur le serveur, nous devons installer le paquet « **apache2-utils** » pour pouvoir ainsi attribuer un mot de passe aux utilisateurs :

```
root@SQUID:/etc/squid3# apt-get install apache2-utils
```

- Nous créons 2 utilisateurs et leur ajoutons un mot de passe chacun :

Utilisateurs	Mots de passe
« **tintin** »	« **reporter** »
« **milou** »	« **chien** »

```
root@SQUID:/etc/squid3# htpasswd -b squidusers tintin reporter
Adding password for user tintin
root@SQUID:/etc/squid3# htpasswd -b squidusers milou chien
Adding password for user milou
root@SQUID:/etc/squid3# _
```

- Pour visualiser les mots de passe cryptés des utilisateurs, nous allons dans le fichier « **squidusers** » :

```
root@SQUID:~# nano /etc/squid3/squidusers
```

```
  GNU nano 2.2.6          Fichier : /etc/squid3/squidusers

tintin:$apr1$JPDkM4Rf$hNvirJ8KMbTWFq7tnqaal0
milou:$apr1$GRSd8uJ9$eFA6rpGgdh0wnBS.VEgwZ/
```

- Nous éditons à nouveau le fichier de configuration et ajoutons les lignes suivantes au tout début du fichier :

```
  GNU nano 2.2.6              Fichier : squid.conf                        Modifié

auth_param basic program /usr/lib/squid3/basic_ncsa_auth /etc/squid3/squidusers
auth_param basic children 5
auth_param basic realm Squid proxy 2A
authenticate_ttl 1 hour
authenticate_ip_ttl 60 seconds
```

- Et, nous ajoutons ces 2 lignes supplémentaires (la première avant « **acl lan** » et la seconde avant tous les « **http_access** ») :

    - La ligne « **acl utilisateurs proxy_auth REQUIRED** » signifie qu'une authentification pour le (ou les) utilisateur(s) est demandée.

```
acl utilisateurs proxy_auth REQUIRED
acl lan src 192.168.1.0/24
acl SSL_ports port 443
```

    - La ligne « **http_access allow utilisateurs** » permet l'autorisation d'accès via la connexion et aux sites.

```
acl CONNECT method CONNECT
http_access allow utilisateurs
http_access allow lan
```

- Nous modifions les droits sur le fichier « **basic_ncsa_auth** » :

```
root@SQUID:~# chown proxy:shadow /usr/lib/squid3/basic_ncsa_auth
root@SQUID:~# chmod 2750 /usr/lib/squid3/basic_ncsa_auth
root@SQUID:~# _
```

La commande « **chmod 2750** », le numéro « **2** » représente le bit **SUID** et le nombre « **750** » concerne tous les droits pour le propriétaire, pas de lecture pour le groupe « **proxy** » et aucun droit pour le public.

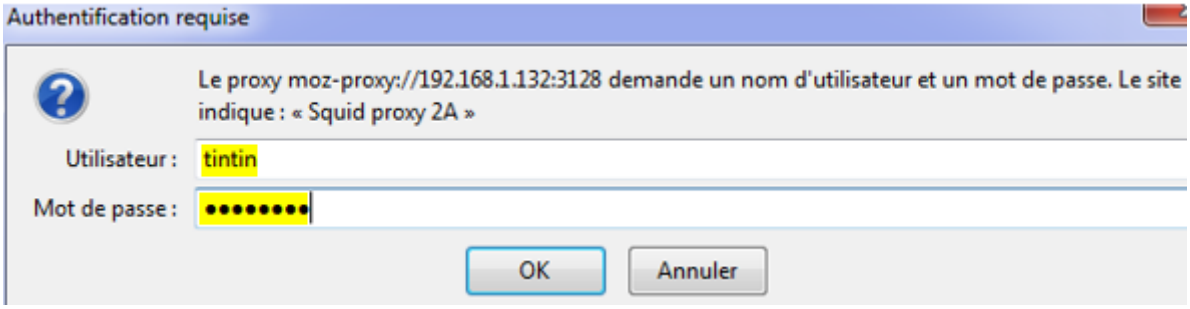
- Nous vérifions les droits de ce fichier pour remarquer le droit « **s** » pour **SUID** permettant le transfert de droits aux utilisateurs :

```
root@SQUID:~# ls -l /usr/lib/squid3/basic_ncsa_auth
-rwxr-s--- 1 proxy shadow 22496 juil. 21 14:20 /usr/lib/squid3/basic_ncsa_auth
root@SQUID:~# _
```

- Nous redémarrons le service « **squid3** » pour prendre en compte les modifications :

```
root@SQUID:~# systemctl restart squid3.service
root@SQUID:~# _
```

- Maintenant, nous testons l'authentification avec un des 2 utilisateurs (Ici, « **tintin** ») :



VIII) Installation et configuration de SquidGuard

- Avant d'utiliser le service « **squidguard** », nous installons d'abord « **apache2** » :

```
root@SQUID:~# apt-get install apache2 squidguard_
```

- Nous vérifions l'accès au serveur Web « **apache2** » :

- Maintenant, nous devons créer 2 fichiers nommés « **black** » pour bloquer l'accès aux sites et « **white** » pour l'autoriser dans le dossier « **/etc/squid** » :

```
  GNU nano 2.2.6          Fichier : /etc/squid3/black

www.google.fr
www.youtube.com
fr.yahoo.com
```

Ici, ces URL représentent le blocage d'accès à ces sites.

```
  GNU nano 2.2.6          Fichier : /etc/squid3/white

www.scolinfo.net
www.lycee-sainte-ursule.fr
www.bing.com
```

Ici, ces URL représentent l'autorisation d'accès à ces sites.

- Maintenant, nous éditons le fichier « **/etc/squid3/squid.conf** » et ajoutons les lignes suivantes permettant le blocage (fichier « **/etc/squid3/black** ») et l'autorisation (fichier « **/etc/squid3/white** ») aux sites :

```
acl whitelist dstdomain "/etc/squid3/white"
acl blacklist dstdomain "/etc/squid3/black"

http_access deny blacklist
http_access allow whitelist
```

- Nous allons dans le répertoire « **/var/lib/squidguard/db** » et récupérons les sources de la liste noire « **blacklists** » :

```
root@SQUID:/var/lib/squidguard/db# wget http://cri.univ-tlse1.fr/blacklists/down
load/blacklists.tar.gz_
```

- Nous décompressons l'archive :

```
root@SQUID:/var/lib/squidguard/db# tar xvzf blacklists.tar.gz
```

- Maintenant, nous retournons dans le fichier « **/etc/squid3/squid.conf** » et ajoutons les lignes suivantes permettant la redirection de **SQUID** vers **SQUIDGUARD** et indiquant le nombre de processus engendré :

```
url_rewrite_program /usr/bin/squidGuard
url_rewrite_children 5
```

- Avant de modifier le fichier de configuration « **/etc/squidguard/squidGuard.conf** », nous faisons une copie de ce dernier pour garder une trace de la configuration :

```
root@SQUID:/etc/squidguard# cp squidGuard.conf squidGuard.conf.back
root@SQUID:/etc/squidguard# _
```

- Nous éditons le fichier « **/etc/squidguard/squidGuard.conf** » en définissant le réseau, une destination interdite et les ACL via le contenu suivant :

  - La ligne « **dbhome** » qui se réfère aux bases de données des **blacklists** et la ligne « **logdir** » qui concerne les logs de **SQUIDGUARD** :

```
dbhome /var/lib/squidguard/db/blacklists
logdir /var/log/squid3
```

  - La ligne « **src lan** » concerne les adresses IP des machines en réseau local qui peuvent accéder aux sites :

```
src lan {
        ip 192.168.1.1-192.168.1.100
}
```

- La ligne « **dest games** » permet de définir la liste noire concernant les domaines et les URL auxquels le (ou les) utilisateur(s) n'auront pas accès :

```
dest games {
        domainlist games/domains
        urllist games/urls
}
```

- Ce contenu décrit l'affichage d'un message d'interdiction d'accès aux sites concernant les jeux :

```
acl {

        lan {
                pass !games all
                redirect http://192.168.1.132/proxy.html
}
```

- Nous commentons la ligne « **redirect http** » avec un « **#** » du contenu « **default** » car la redirection se réalise via le serveur **Proxy** :

```
default {
        pass local none
        #redirect http:
}
```

- Ensuite, nous reconstruisons la base de la liste noire pour **SQUIDGUARD** :

```
root@SQUID:~# squidGuard -C all -d /var/lib/squidguard/db/blacklists
2016-09-13 11:15:31 [2357] INFO: New setting: dbhome: /var/lib/squidguard/db/bla
cklists
2016-09-13 11:15:31 [2357] INFO: New setting: logdir: /var/log/squid3/
2016-09-13 11:15:31 [2357] init domainlist /var/lib/squidguard/db/blacklists/gam
es/domains
2016-09-13 11:15:31 [2357] INFO: create new dbfile /var/lib/squidguard/db/blackl
ists/games/domains.db
2016-09-13 11:15:31 [2357] init urllist /var/lib/squidguard/db/blacklists/games/
urls
2016-09-13 11:15:31 [2357] INFO: create new dbfile /var/lib/squidguard/db/blackl
ists/games/urls.db
2016-09-13 11:15:31 [2357] destblock local missing active content, set inactive
2016-09-13 11:15:31 [2357] INFO: squidGuard 1.5 started (1473758131.837)
2016-09-13 11:15:31 [2357] INFO: db update done
2016-09-13 11:15:31 [2357] INFO: squidGuard stopped (1473758131.870)
root@SQUID:~# _
```

- Nous attribuons la propriété de l'ensemble des fichiers de la liste noire à l'utilisateur « **proxy** » et au groupe « **proxy** » :

```
root@SQUID:~# chown -Rf proxy:proxy /var/lib/squidguard/db/blacklists
root@SQUID:~# _
```

- Ensuite, nous créons une page **HTML** nommée « **proxy.html** » dans le dossier « **/var/www/html** » :

```
root@SQUID:~# nano /var/www/html/proxy.html_
```

- Nous saisissons du contenu signifiant que l'utilisateur ne peut pas avoir accès aux sites de jeux :

```
  GNU nano 2.2.6        Fichier : /var/www/html/proxy.html

<!DOCTYPE html>
<html>
<head>
        <title>Proxy</title>

</head>

<body>
        <h1>Vous n'avez pas l'autorisation d'acces!</h1>
</body>
</html>
```

- Nous retournons dans le répertoire « **/etc/squidguard** » et attribuons le fichier « **squidguard.conf** » à l'utilisateur « **proxy** » afin qu'il en soit le propriétaire :

```
root@SQUID:/etc/squidguard# chown proxy.proxy squidGuard.conf
root@SQUID:/etc/squidguard# _
```

- Nous redémarrons le service « **squid3** » pour prendre en compte les modifications :

```
root@SQUID:~# systemctl restart squid3.service
root@SQUID:~# _
```

- Nous testons l'accès au site « www.games.fr » et constatons que les utilisateurs ne sont pas autorisés à y accéder :

IX) Analyseur de log Lightsquid

- Nous devons installer la librairie « **libgd-gd2-perl** » avant d'utiliser « **Lightsquid** » pour permettre l'affiche du proxy :

```
root@SQUID:~# apt-get install libgd-gd2-perl
```

- Nous téléchargeons l'archive de « **Lightsquid** » dans le dossier « **/var/www/html** » :

```
root@SQUID:/var/www/html# wget http://sourceforge.net/projects/lightsquid/files/
lightsquid/1.8/lightsquid-1.8.tgz_
```

- Nous décompressons l'archive :

```
root@SQUID:/var/www/html# tar xvzf lightsquid-1.8.tgz
```

- Nous modifions son nom :

```
root@SQUID:/var/www/html# mv lightsquid-1.8/ lightsquid
root@SQUID:/var/www/html# _
```

- Nous rendons les scripts « **pl** » et « **cgi** » exécutables :

```
root@SQUID:/var/www/html# chmod -R ugo+x lightsquid/*.pl
root@SQUID:/var/www/html# chmod -R ugo+x lightsquid/*.cgi
root@SQUID:/var/www/html# _
```

- Nous modifions le propriétaire du dossier « **lightsquid** » par « **www-data** » :

```
root@SQUID:/var/www/html# chown -R www-data:www-data lightsquid
root@SQUID:/var/www/html# _
```

- Nous éditons le fichier « **/etc/apaches2/sites-available/000-default.conf** » :

```
root@SQUID:~# nano /etc/apache2/sites-available/000-default.conf
```

- <u>Nous ajoutons les lignes suivantes</u> :

  - La directive « **Addhandler cgi-script .cgi** » permet l'exécution de tous les programmes **CGI** possédant l'extention « **.cgi** » :
  - « **AllowOverride All** » signifie
  - « **DirectoryIndex index.cgi** » définit
  - « **Options +ExecCGI** » décrit l'exécution des programmes **CGI** permise depuis un répertoire particulier.

```
<Directory "/var/www/html/lightsquid">
        AddHandler cgi-script .cgi
        AllowOverride All
        DirectoryIndex index.cgi
        Options +ExecCGI
</Directory>
```

- Maintenant, nous éditons le fichier de configuration de **Lightsquid** nommé « **/var/www/html/lightsquid/lightsquid.cfg** » en modifiant les lignes suivantes :

  - Nous le personnalisons avec le nom du dossier des logs de **SQUID** :

```
#path to access.log
$logpath                ="/var/log/squid3";
```

  - Nous changeons la langue en français « **fr** » :

```
#language
#see `lang` folder (avaible: bg,eng,fr,hu,it,pt_br,ru,sp)
$lang                   ="fr";
```

- Nous testons l'installation :

```
root@SQUID:/var/www/html/lightsquid# ./check-setup.pl
LightSquid Config Checker, (c) 2005-9 Sergey Erokhin GNU GPL

LogPath    : /var/log/squid3
reportpath: /var/www/html/lightsquid/report
Lang       : /var/www/html/lightsquid/lang/fr
Template   : /var/www/html/lightsquid/tpl/base
Ip2Name    : /var/www/html/lightsquid/ip2name/ip2name.simple


all check passed, now try access to cgi part in browser
root@SQUID:/var/www/html/lightsquid# _
```

- Si nous n'avons pas d'erreur, nous pouvons parser le fichier de log de **SQUID** :

```
root@SQUID:/var/www/html/lightsquid# ./lightparser.pl
root@SQUID:/var/www/html/lightsquid# _
```

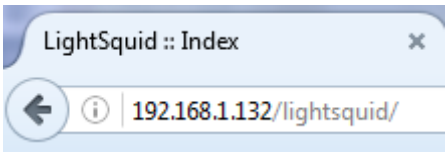
- Nous activons le module « **cgi** » représentant une interface utilisée par les serveurs Web :

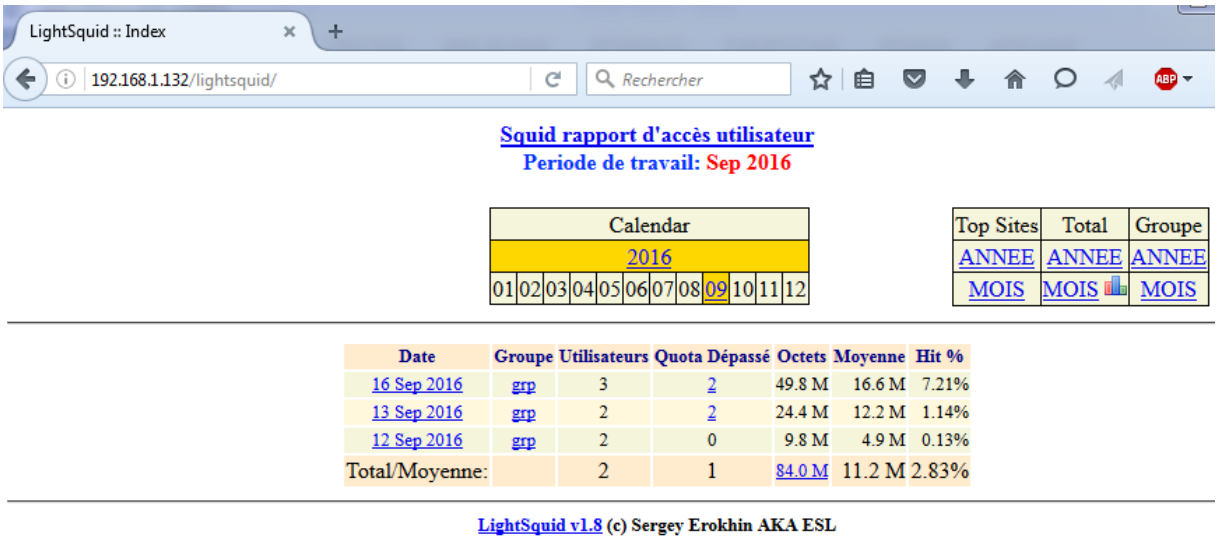
```
root@SQUID:/var/www/html/lightsquid# a2enmod cgi
AH00558: apache2: Could not reliably determine t
in name, using 127.0.1.1. Set the 'ServerName' d
is message
Your MPM seems to be threaded. Selecting cgid in
Enabling module cgid.
To activate the new configuration, you need to r
  service apache2 restart
root@SQUID:/var/www/html/lightsquid# _
```

- Nous redémarrons le service « **apache2** » pour prendre en considération les modifications :

```
root@SQUID:/var/www/html/lightsquid# systemctl restart apache2.service
root@SQUID:/var/www/html/lightsquid# _
```

- Enfin, nous testons via un navigateur en saisissant l'URL comme suit : « http://@IP_serveurSQUID/lightsquid/ » et constatons le résultat obtenu :





Nous pouvons constater que cela fonctionne car une interface Web de **Lightsquid** s'affiche pour mieux visualiser les sites visités.

X) Configuration d'un navigateur via un script

- Nous créons un script nommé « **proxy.pac** » dans le répertoire « **/var/www/html** » :

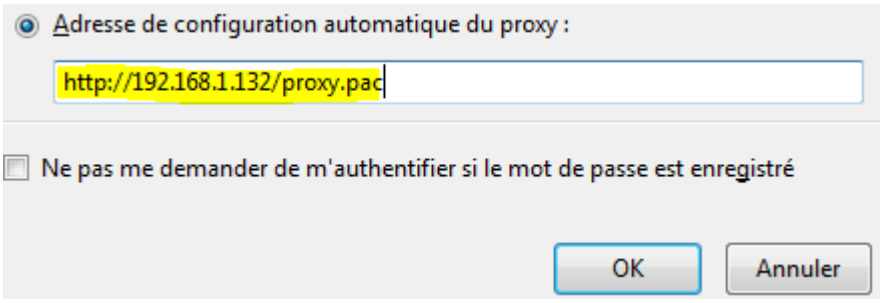
```
root@SQUID:/var/www/html# nano proxy.pac
```

- A l'intérieur de ce script, nous saisissons le contenu suivant qui permet de forcer les navigateurs à récupérer les pages Web sur le port **3128** du proxy :

```
  GNU nano 2.2.6                Fichier : proxy.pac

function FindProxyForURL(url,host)
{
return "PROXY 192.168.1.132:3128;DIRECT";
}
```

- Nous devons paramétrer le navigateur :



- Nous éditons le fichier de log « **/var/log/squid3/access.log** » :

```
root@SQUID:~# tail /var/log/squid3/access.log
```

Nous pouvons constater que nous nous sommes connectés sur le serveur **Proxy** avec l'utilisateur « **tintin** » et que cela a fonctionné grâce à l'instruction « **TCP_MISS/200** » :

```
1474037743.563  61176 192.168.1.74 TCP_MISS/200 3464 CONNECT incoming.telemetry.
mozilla.org:443 tintin HIER_DIRECT/54.69.68.55 -
root@SQUID:~#
```

XI) Conclusion

En conclusion, nous pouvons dire le serveur **Proxy SQUID** est fonctionnel car celui-ci permet de limiter l'accès à certains sites aux utilisateurs et faire du proxy-cache.