L’authentification via l’Active Directory

Le protocole NTLM est un protocole d’authentification propriétaire de Microsoft. Ce procédé d’identification permet aux utilisateurs d’accéder au proxy sans remplir de formulaire d’authentification. Pour reconnaitre les requêtes web émises par un utilisateur, ce procédé se base sur les informations d’ouverture de session. C’est le principe de l’authentification transparente.

1. Le principe de fonctionnement

Précédemment nous avons vu avec la méthode d’authentification Basic que les noms d’utilisateur et les mots de passe étaient transmis dans les échanges lors du processus d’identification. Par le biais d’une authentification NTLM, les utilisateurs doivent prouver au serveur proxy leur identité sans transmettre leurs informations d’identification. Pour cela un mécanisme basé sur un principe d’interrogation/réponse (challenge/response) est mis en œuvre. Ce mécanisme met en œuvre plusieurs échanges entre le client et le proxy et celui-ci doit alors prendre en charge les connexions permanentes pour réussir l’authentification.

Les échanges entre le client et le proxy sont les suivants :

- Le client émet une requête web de type GET vers le proxy.
- Le proxy, par une réponse de type 407, informe l’émetteur qu’il est nécessaire de s’authentifier via NTLM, comme le montre l’en-tête de paquet HTTP ci-dessous :

```
HTTP/1.1 407 Proxy Authentication Required
Server: squid/3.2.1
Mime-Version: 1.0
Date: Wed, 17 Oct 2012 09:12:54 GMT
Content-Type: text/html
Content-Length: 3749
X-Squid-Error: ERR_CACHE_ACCESS_DENIED 0
Vary: Accept-Language
Content-Language: fr
Proxy-Authenticate: NTLM
Proxy-Authenticate: Basic realm="proxy_du_domaine_camp-facsalle.fr"
X-Cache: MISS from srv-proxy
Via: 1.1 srv-proxy (squid/3.2.1)
Connection: close
```

- Le client réémet sa requête GET en fournissant en plus ses informations d’identification NTLM sous forme hashée et non réversible (voir la partie NTLM de l’en-tête de paquet ci-dessous). Le processus d’authentification NTLM (Negotiate) est mis en route.

```
GET http://www.google.fr/ HTTP/1.1\r\n
       [Expert Info (Chat/Sequence): GET http://www.google.fr/ HTTP/1.1\r\n]
            [Message: GET http://www.google.fr/ HTTP/1.1\r\n]
            [Severity level: Chat]
            [Group: Sequence]
        Request Method: GET
        Request URI: http://www.google.fr/
        Request Version: HTTP/1.1
    Accept: */*\r\n
    Accept-Language: fr\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1;
```

```
WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729;
.NET CLR 3.0.30729; Media Center PC 6.0)\r\n
    Accept-Encoding: gzip, deflate\r\n
    Proxy-Connection: Keep-Alive\r\n
    Proxy-Authorization:NTLM RMTVNTUAABAAAAB4IIogAAAAAAAAGAbAdAAAADw==\r\n
        NTLM Secure Service Provider
            NTLMSSP identifier: NTLMSSP
            NTLM Message Type: NTLMSSP_NEGOTIATE (0x00000001)
            Flags: 0xa2088207
            Calling workstation domain: NULL
            Calling workstation name: NULL
            Version 6.1 (Build 7600); NTLM Current Revision 15
    Host: www.google.fr\r\n
    \r\n
    [Full request URI: http://www.google.frhttp://www.google.fr/]
```

- Le serveur proxy renvoie une nouvelle fois une réponse de type 407 et transmet ses informations concernant l'authentification NTLM, aussi sous forme hashée. C'est le challenge du processus NTLM.

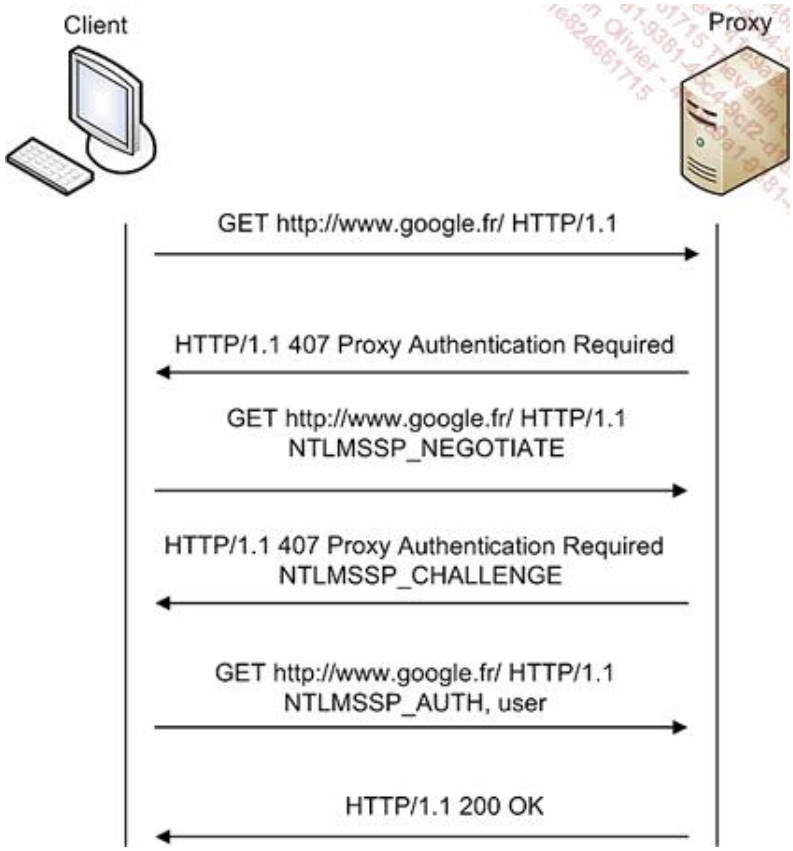
```
HTTP/1.1 407 Proxy Authentication Required\r\n
        [Expert Info (Chat/Sequence): HTTP/1.1 407 Proxy
Authentication Required\r\n]
           [Message: HTTP/1.1 407 Proxy Authentication Required\r\n]
           [Severity level: Chat]
           [Group: Sequence]
        Request Version: HTTP/1.1
        Status Code: 407
        Response Phrase: Proxy Authentication Required
    Server: squid/3.2.1\r\n
    Mime-Version: 1.0\r\n
    Date: Wed, 17 Oct 2012 09:12:54 GMT\r\n
    Content-Type: text/html\r\n
    Content-Length: 3847\r\n
    X-Squid-Error: ERR_CACHE_ACCESS_DENIED 0\r\n
    Vary: Accept-Language\r\n
    Content-Language: fr\r\n
    [truncated]
Proxy-Authenticate:NTLM
TlRMTVNTUAACAAAAGgAaADAAAAAFgomiK9BO6D5H8o8AAAAAAAAAAIAAgABKAAAAQ
wBBAE0AUAAtAEYAQQBDAFMAQQBMAEwARQACABoAQwBBAE0AUAAtAEYAQQBDAFMAQQ
BMAEwARQABABIAUwBSAFYALQBQAFIATwBYAFkABAAWAGwAbwBjAGEAbABkAG8AbQB
hAGkAbg
        NTLM Secure Service Provider
            NTLMSSP identifier: NTLMSSP
            NTLM Message Type: NTLMSSP_CHALLENGE (0x00000002)
            Target Name: CAMP-FACSALLE
            Flags: 0xa2898205
            NTLM Server Challenge: 2bd04ee83e47f28f
            Reserved: 0000000000000000
            Target Info
    X-Cache: MISS from srv-proxy\r\n
    Via: 1.1 srv-proxy (squid/3.2.1)\r\n
    Connection: keep-alive\r\n
```

- Le navigateur client crypte, via la fonction hachage, le mot de passe saisi lors de l'ouverture de session et renvoie une

clé secrète correspondant au mot de passe hashé. C'est la réponse au challenge NTLM émis par le proxy précédemment.

```
GET http://www.google.fr/ HTTP/1.1\r\n
       [Expert Info (Chat/Sequence): GET http://www.google.fr/ HTTP/1.1\r\n]
            [Message: GET http://www.google.fr/ HTTP/1.1\r\n]
            [Severity level: Chat]
            [Group: Sequence]
        Request Method: GET
        Request URI: http://www.google.fr/
        Request Version: HTTP/1.1
    Accept: */*\r\n
    Accept-Language: fr\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1;
WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729;
.NET CLR 3.0.30729; Media Center PC 6.0)\r\n
    Accept-Encoding: gzip, deflate\r\n
    Proxy-Connection: Keep-Alive\r\n
    [truncated]
    Host: www.google.fr\r\n
    [truncated]
Proxy-Authorization:NTLM
TlRMTVNTUAADAAAAGAAYAJwAAAAaARoBtAAAABoAGgBYAAAADgAOAHIAAAAcABwAg
AAAAAAAAADOAQAABYKIogYBsB0AAAAP2tjUKaZOgoEI9a5OW5lRSEMAQQBNAFAALQ
BGAEEAQwBTAEEATABMAEUAcABiAGEAcgByAGUAdABTAFQAQQBUAEkATwBOAC0AQwB
IAEkAT
       NTLM Secure Service Provider
        NTLMSSP identifier: NTLMSSP
        NTLM Message Type: NTLMSSP_AUTH (0x00000003)
        Lan Manager Response: 00000000000000000000000000000000000000000
            NTLM Client Challenge: 0000000000000000
            NTLM Response: 873849bf96f4060f241694f0695cd4e5010100000...
            NTLMv2 Response: 873849bf96f4060f241694f0695cd4e50101000...
            NTLM Client Challenge: d8ebd5224141d6a5
            Domain name: CAMP-FACSALLE
            User name: pbarret
            Host name: STATION-CHIMIE
            Session Key: Empty
            Flags: 0xa2888205
            Version 6.1 (Build 7600); NTLM Current Revision 15
            MIC: dad8d429a64e828108f5ae4e5b995148     \r\n
    [Full request URI: http://www.google.frhttp://www.google.fr/]
```

- Une fois ces cinq échanges effectués, le client est alors authentifié et le proxy peut donc lui renvoyer la page demandée via la requête GET. Le diagramme d'échange ci-dessous récapitule les échanges mis en œuvre.



L'implémentation de ce schéma d'authentification nécessite de mettre en œuvre des modules supplémentaires. Ils vont permettre au serveur proxy d'être intégré à un domaine mais aussi de pouvoir interroger un annuaire.

2. Les modules des pré-requis

L'implémentation de ce type de contrôle nécessite d'installer Samba et Winbind sur le serveur, ces deux modules permettent à Squid d'interroger l'annuaire Active Directory pour vérifier l'identité des clients.

Depuis l'avènement de Windows 2000, les contrôleurs de domaines Windows n'utilisent plus les authentifications via le protocole NTLM ou NTLMv2. Il est possible de modifier ce paramètre via les stratégies de sécurité de domaine (GPO) mais ce n'est pas conseillé. Lors d'une ouverture de session sur un domaine Windows de type 2000, 2003 ou 2008, l'authentification est réalisée via le protocole Kerberos.

Afin de ne pas modifier les stratégies de sécurité, il faut aussi mettre en œuvre Kerberos sur le proxy. Il ne sera pas chargé de l'authentification comme pour une ouverture de session mais permettra à Squid d'interroger l'annuaire de façon sécurisée. Ce protocole ne fonctionne pas à l'aide de noms d'utilisateur et de mots de passe, il se base sur un mécanisme de cryptographie à clefs secrètes et sur des tickets qui permettent d'authentifier des utilisateurs et donc dans notre cas d'authentifier les requêtes qui seront émises vers le serveur proxy.

Plusieurs modules sont donc nécessaires pour implémenter cette stratégie d'authentification :

- Posséder un service DNS fonctionnel (Windows ou Linux)
- Sur le serveur Windows :
  - Active Directory : c'est l'annuaire qui permet de créer les comptes utilisateurs et les groupes sur un domaine Windows.
- Sur le proxy :

- Samba : module qui permet d'intégrer le serveur proxy au domaine. On utilisera aussi le module d'authentification fourni par Samba pour interroger l'annuaire.

- Winbind : module qui permet de récupérer les comptes utilisateurs et les groupes présents dans l'Active Directory. Il permet aussi de mettre en œuvre un mécanisme d'authentification via un module PAM (*Pluggable Authentication Modules*).

- Kerberos : service qui permet d'interroger l'Active Directory de façon sécurisée afin de vérifier les utilisateurs contenus dans celui-ci.

- NTP : serveur de temps qui permet au serveur proxy de synchroniser son horloge sur une référence commune.

Installation des paquets pré-requis

Installez les modules nécessaires sur le serveur proxy :

```
# yum install samba samba-winbind krb5-server krb5-workstation ntp
Loaded plugins: fastestmirror, refresh-packagekit, security
Loading mirror speeds from cached hostfile
 * base: centos.cict.fr
 * extras: centos.cict.fr
 * updates: centos.bio.lmu.de
base                          | 3.7 kB     00:00
extras                        | 3.0 kB     00:00
updates                       | 3.5 kB     00:00
Setting up Install Process
Package krb5-workstation-1.9-33.el6_3.3.x86_64 already installed and
latest version
Package ntp-4.2.4p8-2.el6.centos.x86_64 already installed and
latest version
Resolving Dependencies
--> Running transaction check
---> Package krb5-server.x86_64 0:1.9-33.el6_3.3 will be installed
---> Package samba.x86_64 0:3.5.10-125.el6 will be installed
---> Package samba-winbind.x86_64 0:3.5.10-125.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
================================================================================
Package            Arch      Version       Repository    Size
================================================================================
================================================================================
Installing:
 krb5-server      x86_64   1.9-33.el6_3.3   updates      947 k
 samba            x86_64   3.5.10-125.el6   base         5.0 M
 samba-winbind    x86_64   3.5.10-125.el6   base         3.5 M

Transaction Summary
================================================================================
================================================================================
Install       3 Package(s)
```

```
Total download size: 9.4 M
Installed size: 31 M
Is this ok [y/N]:y
```

Vérification de l'installation des paquets

Une fois que les paquets sont correctement installés, vérifiez que les fichiers de configuration nécessaires sont présents sur le système.

Fichier de configuration de Samba :

```
# ls -l /etc/samba/smb.conf
-rw-r--r--. 1 root root 9778 22 juin  14:18 /etc/samba/smb.conf
```

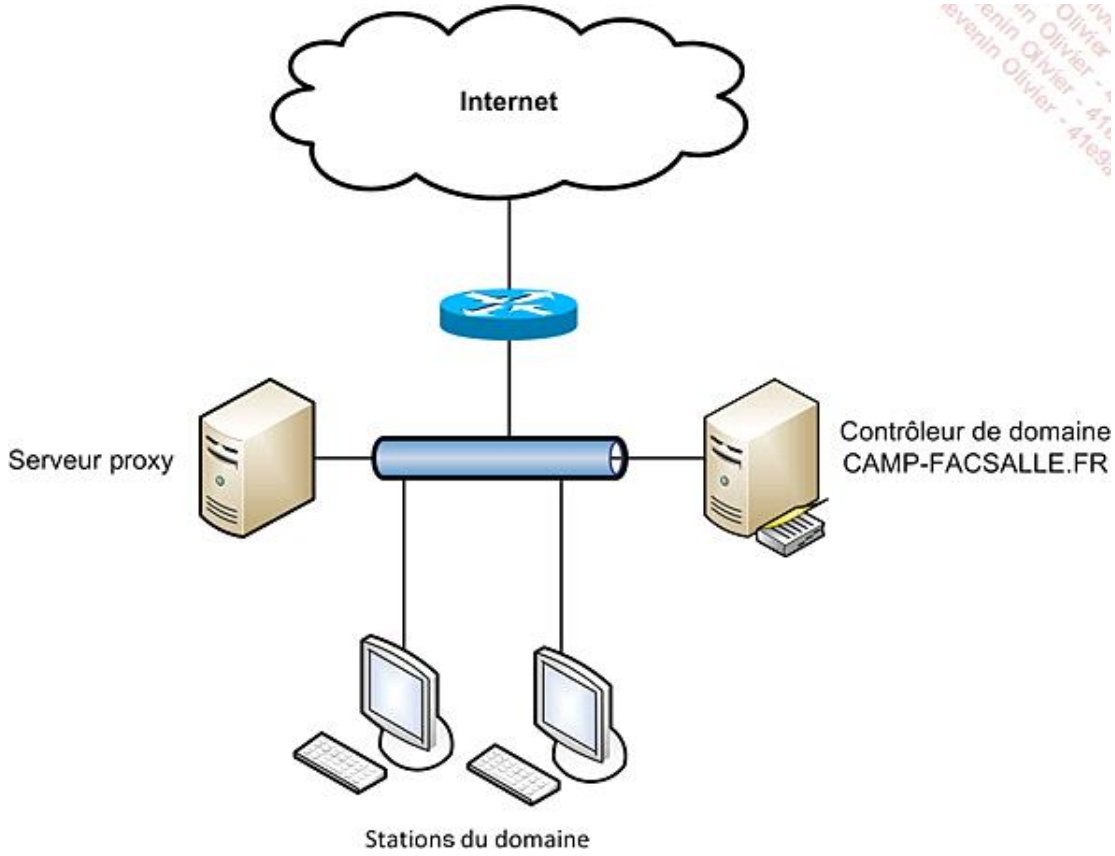
Fichier de configuration de Kerberos :

```
# ls -l /etc/krb5.conf
-rw-r--r--. 1 root root 449 11 sept. 03:12 /etc/krb5.conf
```

3. Mise en œuvre de NTLM

Cette section va permettre aux utilisateurs présents dans l'annuaire du domaine camp-facsalle.fr de s'authentifier sur le serveur proxy (srv-proxy). Le domaine est installé à partir d'une architecture Microsoft en se basant sur Windows Serveur W2008 (srv-2008). On utilisera sur le proxy le processus d'identification fourni par Samba `ntlm_auth` qui se trouve sous */usr/bin/*.

L'exemple d'authentification via un processus d'authentification NTLM s'appuie sur la synoptique du réseau suivante :



L'annuaire Active Directory comporte plusieurs OU (*Organizational Unit*) dont une appelée Pole-Chimie. Dans ce conteneur, on retrouve trois comptes *ldupont*, *hferry* et *pbarret*.

L'exemple de configuration d'authentification en utilisant le schéma NTLM doit permettre à ces comptes de se connecter au serveur proxy via une authentification transparente. Si ces utilisateurs utilisent des stations qui ne sont pas intégrées au domaine, ils devront se connecter à Internet par le biais du proxy via une authentification Basic.

Avant de procéder à la mise en œuvre de l'authentification, il est nécessaire de synchroniser l'heure du serveur proxy sur le contrôleur de domaine. Un décalage de temps supérieur à cinq minutes entre les deux équipements rendra impossible l'authentification.

Synchronisation de l'heure du proxy sur le contrôleur de domaine :

```
#ntpdate srv-2008.camp-facsalle.fr
```

Cette commande a un effet immédiat mais pas sur le long terme. Pour que ce paramètre soit pérenne, éditez le fichier */etc/ntp.conf* en lui ajoutant la ligne suivante :

```
server srv-2008.camp-facsalle.fr prefer
```

a. Configuration du fichier de configuration Samba

La première étape consiste à configurer Samba, afin que celui-ci puisse récupérer les informations de l'Active Directory.

Éditez le fichier de configuration */etc/samba/smb.conf* et modifiez-le en fonction des données ci-dessous. La section `global` du fichier permet de définir les paramètres globaux des données réseau à utiliser.

```
[global]
workgroup = CAMP-FACSALLE
netbios name = srv-proxy
server string = srv-proxy
load printers = no
log file = /var/log/samba/log.%m
max log size = 500
password server = srv-2008.camp-facsalle.fr
realm = camp-facsalle.fr
security = ADS
winbind separator = /
encrypt passwords = yes
winbind cache time = 15
winbind enum users = yes
winbind enum groups = yes
winbind use default domain = yes
idmap uid = 10000-20000
idmap gid = 10000-20000
local master = no
os level = 233
domain master = no
preferred master = no
```

Explication du rôle des directives du fichier *smb.conf* :

- `workgroup` : définit le nom de domaine à utiliser.
- `realm` : définit la zone Kerberos à utiliser que l'on appelle aussi royaume.
- `server string` : nom du serveur sur le domaine.
- `security` : ici l'option ADS spécifie que le serveur est intégré à l'Active Directory, il devient donc serveur membre du domaine. Ce paramètre définit aussi qu'il doit utiliser le protocole Kerberos pour s'authentifier auprès du serveur.
- `password server` : nom du serveur du royaume Kerberos.
- `winbind separator` : spécifie le caractère à utiliser pour séparer le nom de domaine et le nom d'utilisateur sur celui-ci.

Vous pouvez vérifier votre fichier de configuration à l'aide de la commande `testparm`.

```
#testparm /etc/samba/smb.conf
Load smb config files from /etc/samba/smb.conf
rlimit_max: increasing rlimit_max (1024) to minimum Windows limit (16384)
Loaded services file OK.
Server role: ROLE_DOMAIN_MEMBER
Press enter to see a dump of your service definitions

[global]
        workgroup = CAMP-FACSALLE
```

```
        realm = CAMP-FACSALLE.FR
        server string = srv-proxy
        security = ADS
        password server = 172.16.1.11
        log file = /var/log/samba/log.%m
        max log size = 500
        load printers = No
        os level = 233
        local master = No
        domain master = No
        idmap uid = 10000-20000
        idmap gid = 10000-20000
        winbind separator = /
        winbind cache time = 15
        winbind enum users = Yes
        winbind enum groups = Yes
        winbind use default domain = Yes
```

Lors de la vérification, il est important de bien s'assurer que Samba est déclaré en tant que ROLE_DOMAIN_MEMBER.

b. Configuration de Kerberos

Le protocole Kerberos est nécessaire pour que Samba, via le module Winbind, puisse interroger l'Active Directory de manière sécurisée. Pour configurer les informations nécessaires, éditez le fichier *krb5.conf* qui se trouve sous */etc* et modifiez-le en vous basant sur les données ci-dessous :

```
[logging]
 default = FILE:/var/log/krb5libs.log
 kdc = FILE:/var/log/krb5kdc.log
 admin_server = FILE:/var/log/kadmind.log

[libdefaults]
 default_realm = CAMP-FACSALLE.FR
 dns_lookup_realm = false
 dns_lookup_kdc = false
 ticket_lifetime = 24h
 renew_lifetime = 7d
 forwardable = true

[realms]
 CAMP-FACSALLE.FR = {
  kdc = srv-2008.camp-facsalle.fr:88
  admin_server = srv-2008.camp-facsalle.fr:789
  default_domain=camp-facsalle.fr
 }

[domain_realm]
 .camp-facsalle.fr = CAMP-FACSALLE.FR
 camp-facsalle.fr = CAMP-FACSALLE.FR
```

La configuration du fichier *krb5.conf* ci-dessus définit au serveur proxy qu'il doit utiliser le royaume Kerberos *camp-*

facsalle.fr et que le serveur qui fournit les tickets nécessaires à l'authentification est *srv-2008.camp-facsalle.fr*.

Une fois que les modules Samba et Kerberos sont correctement configurés, il faut indiquer au serveur proxy qu'il est nécessaire d'utiliser Winbind pour trouver les noms d'utilisateurs et les groupes du domaine *camp-facsalle.fr*. Pour cela il faut modifier le fichier *nsswitch.conf*. C'est le fichier de configuration des bases de données et des services de résolution de noms du serveur. Ce fichier se trouve sous */etc*.

Éditez le fichier *nsswitch.conf* et rajoutez l'information `winbind` au niveau des directives définissant les comptes et les groupes à utiliser.

```
passwd:     files winbind
shadow:     files
group:      files winbind
```

Par défaut, le mécanisme d'authentification `ntlm_auth` de Samba s'exécute en tant que *root* pour s'authentifier auprès de l'Active Directory via Winbind. Pour cela, il s'appuie sur le dossier pour le processus pipe *winbindd_privileged*. Ce dossier appartient au groupe *root* et seuls les membres de ce groupe peuvent donc appeler ce processus pour utiliser l'authentificateur `ntlm_auth`. Afin que l'utilisateur `squid` puisse utiliser ce processus, il faut définir que le groupe `squid` est propriétaire de ce dossier.

Modifiez les permissions du dossier *winbindd_privileged* :

```
#chgrp squid /var/lib/samba/winbindd_privileged
```

À chaque redémarrage du serveur, les permissions de ce dossier sont réaffectées à l'utilisateur *root*. Il est judicieux de modifier le script de démarrage du service Winbind ou de rajouter la ligne de commande ci-dessus dans le fichier *rc.local*.

Une fois que les différents fichiers de configuration des services nécessaires à l'authentification sont modifiés, vous devez redémarrer les services Samba et Winbind.

```
# service winbind stop
# service smb restart
# service winbind start
```

c. Intégration du proxy au domaine

Pour intégrer le serveur au domaine Microsoft et qu'il devienne ainsi serveur membre du domaine, vous devez initialiser le module Kerberos. Pour cela il faut utiliser la commande `kinit`, qui permet d'obtenir un ticket Kerberos. Elle permet d'obtenir un ticket d'authentification (TGT) et met celui-ci en cache.

La commande `kinit` permet donc de vérifier que l'authentification Kerberos est possible auprès du serveur.

Lancez la commande `kinit`, comme ci-dessous, pour obtenir un ticket d'authentification auprès du contrôleur de domaine.

```
# kinit administrateur
Using default cache: /tmp/krb5cc_0
Using principal: Administrateur@CAMP-FACSALLE.FR
```

```
Password for Administrateur@CAMP-FACSALLE.FR:
Authenticated to Kerberos v5
```

Utilisez la commande `klist` pour visualiser le ticket d'authentification du service Kerberos obtenu à l'aide de la commande `kinit`.

```
# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: Administrateur@CAMP-FACSALLE.FR

Valid starting     Expires            Service principal
10/12/12 15:56:13  10/13/12 01:56:15  krbtgt/CAMP-FACSALLE.FR@CAMP-FACSALLE.FR
        renew until 10/19/12 15:56:13
```

Après avoir vérifié que vous obtenez un ticket Kerberos auprès du contrôleur de domaine, il faut intégrer votre serveur proxy au domaine *camp-facsalle.fr*. Pour cela, utilisez la commande `net`. Elle permet de se connecter au serveur Windows 2008 et d'intégrer le proxy au domaine.

```
# net ADS join -U Administrateur
Enter Administrateur's password:
Using short domain name - CAMP-FACSALLE
Joined 'SRV-PROXY' to realm 'camp-facsalle.fr'
```

Une fois que votre serveur est bien intégré au domaine, vous pouvez vérifier que votre serveur peut lire les informations présentes dans l'Active Directory. Pour cela, utilisez la commande `wbinfo`.

La commande `wbinfo -u` permet de lister les utilisateurs présents dans l'annuaire :

```
# wbinfo -u
administrateur
invité
krbtgt
ldupont
hferry
pbarret
```

On retrouve bien les trois utilisateurs créés dans l'OU Pole-Chimie.

La commande `wbinfo -g` permet de lister les groupes utilisateurs présents dans l'annuaire.

```
#wbinfo -g
ordinateurs du domaine
controleurs de domaine
administrateurs du schéma
administrateurs de l'entreprise
editeurs de certificats
admins du domaine
utilisateurs du domaine
```

```
invites du domaine
proprietaires createurs de la strategie de groupe
serveurs ras et ias
groupe de replication dont le mot de passe rodc est autorise
groupe de replication dont le mot de passe rodc est refuse
controleurs de domaine en lecture seule
controleurs de domaine d'entreprise en lecture seule
dnsadmins
dnsupdateproxy
```

Une fois que le serveur proxy est intégré au domaine, il reste à configurer le fichier de configuration de Squid en déclarant les programmes externes à utiliser.

d. Configuration du fichier Squid

Pour rappel, précédemment nous avons défini que nous allons utiliser le service d'authentification `ntlm_auth` de Samba pour procéder à l'identification.

Avant de configurer le fichier *squid.conf*, on peut tester le fonctionnement de l'authentification du processus via la commande suivante :

```
# /usr/bin/ntlm_auth --username=ldupont --password=password

NT_STATUS_OK: Success (0x0)
```

Ici, l'authentification a été testée à l'aide du compte *ldupont* qui a comme mot de passe *password*. On constate que celle-ci a abouti. On peut donc passer à la configuration de Squid.

Comme pour l'authentification de type Basic, on appelle dans le fichier *squid.conf* un processus externe pour identifier les utilisateurs.

Éditez le fichier de configuration et modifiez la partie concernant l'authentification en vous basant sur l'exemple ci-dessous :

```
#####################################################################
#Authentification basée sur le programme NTLM
#
auth_param ntlm program /usr/bin/ntlm_auth
--helper-protocol=
squid-2.5-ntlmssp
auth_param ntlm children 10
auth_param ntlm max_challenge_reuses 0
auth_param ntlm max_challenge_lifetime 2 minutes

#####################################################################
#Authentification pour les utilisateurs dont la station n'est pas
#intégrée au domaine
#
auth_param basic program /usr/bin/ntlm_auth
--helper-protocol=
squid-2.5-basic
```

```
auth_param basic children 5
auth_param basic realm proxy du domaine camp-facsalle.fr
auth_param basic credentialsttl 2 hours
```

Pour mettre en œuvre l'authentification NTLM, Squid appelle le programme `ntlm_auth` situé sous */usr/bin* et se base sur le protocole `squid-2.5-ntlmssp`. Le nombre de processus d'authentification simultanés est de 10. Chaque authentification NTLM ne sera utilisée qu'une seule fois et aura une durée de vie de deux minutes. Ces informations sont renseignées par les paramètres `max_challenge_reuses` et `max_challenge_lifetime`. Pour définir que l'on utilise une seule fois une authentification NTLM, il suffit de fixer la valeur à 0.

Pour les utilisateurs n'utilisant pas de station intégrée au domaine, ou pour ceux utilisant leurs outils personnels, un formulaire de connexion leur permettra de rentrer leurs identifiants afin de pouvoir se connecter à Internet via le serveur proxy.

Il reste à définir dans la partie `acl` du fichier de configuration celle correspondant aux utilisateurs de l'annuaire, et à autoriser ces utilisateurs à accéder au proxy via la directive `http_access`.

```
acl auth_ntlm proxy_auth REQUIRED

http_access allow auth_ntlm
http_access allow localnet
http_access allow localhost
```

Une fois les modifications nécessaires apportées au fichier de Squid, rechargez le fichier de configuration pour appliquer les modifications effectuées au service.

```
#/usr/local/squid/sbin/squid -k reconfigure
```

Il est maintenant possible de tester l'authentification. Pour cela, ouvrez une session sur une station sur le domaine et lancez un navigateur web. Aucun formulaire de connexion ne doit vous être proposé.

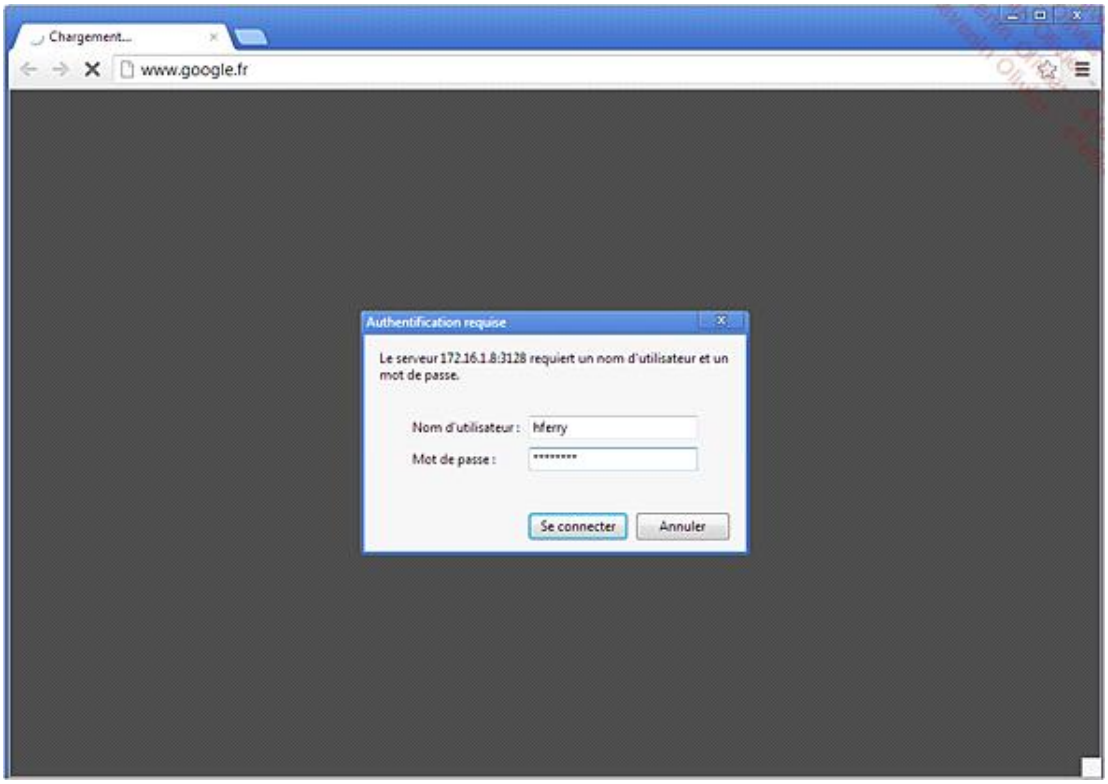
Afin de valider la configuration et de vérifier que les requêtes émises par les utilisateurs sont bien authentifiées, consultez le fichier journal *access.log*.

Vous constatez qu'au niveau des requêtes, un nom d'utilisateur, en plus de l'adresse IP, est renseigné.

```
1350460301.612    539 172.16.1.134 TCP_MISS/200 594 GET
http://www.google.fr/complete/search? pbarret
HIER_DIRECT/173.194.67.94 text/javascript

1350460302.374    146 172.16.1.134 TCP_MISS_ABORTED/000 0 GET
http://www.google.fr/complete/search? pbarret
HIER_DIRECT/173.194.67.94
```

Pour tester l'authentification à partir d'un poste non intégré au domaine, lancez un navigateur web et vérifiez qu'un formulaire de connexion vous est proposé comme ci-après.



Dans cet exemple, l'utilisateur *hferry* se connecte à Internet à partir d'une station qui ne fait pas partie du domaine. Lors du lancement du navigateur, un formulaire de connexion lui est proposé.

Après avoir entré ses identifiants, il peut accéder au web via le serveur proxy.

Pour vérifier que les requêtes sont bien identifiées dans ce cadre, ouvrez le fichier journal *access.log* et vérifiez que chaque entrée du fichier est bien associée à un nom d'utilisateur.

Dans les lignes ci-dessous, on constate que les accès effectués par l'utilisateur *hferry* sont identifiés.

```
232 172.16.1.134 TCP_MISS/200 860 GET http://www.google.fr/s?
hferry HIER_DIRECT/74.125.132.94 application/json

180 172.16.1.134 TCP_MISS/204 286 GET http://www.google.fr/
gen_204? hferry HIER_DIRECT/74.125.132.94 text/html
```

Si des formulaires d'authentification sont proposés aux utilisateurs du domaine, c'est que toutes les instances d'authentification sont utilisées. Si c'est le cas, c'est que vous n'en avez pas défini assez. Pour pallier ce problème, il vous faut augmenter le nombre de processus d'authentification simultanés que peut lancer le service via le paramètre `children`. La valeur maximale que l'on peut définir est 250.

Cette section vous a permis de vous familiariser avec le principe de l'authentification NTLM sous Squid. Cette méthode permet d'offrir une authentification transparente pour l'accès au serveur proxy. Elle n'est pas forcément des plus sécurisées et c'est pourquoi il est préférable aujourd'hui de définir l'authentification via le protocole Kerberos.

Il existe d'autres méthodes pour mettre en œuvre l'authentification sous Squid. Vous pouvez coupler Squid avec un annuaire LDAP (*Lightweight Directory Access Protocol*) par exemple. La configuration est très similaire à celle utilisée pour authentifier les utilisateurs via les informations de session de l'Active Directory. Si vous voulez en

savoir plus, n'hésitez pas à consulter le site officiel de Squid (http://www.squid-cache.org) qui propose de nombreux exemples.