ZIMBRA Collaboration Suite – SAMBA – OPENVPN

Introduction

La suite de travail collaborative éditée par Zimbra propose une solution open source très complète composée du meilleur du monde libre. Zimbra permet de mettre en place un serveur de messagerie électronique, un agenda partagée, un carnet d'adresses partagé, un wiki partagé, et autant de fonctionnalités supplémentaires que de Zimlets (Extentions pour Zimbra) existants. De plus toute la suite est accessible à tout moment depuis un accès internet avec un navigateur Internet comme Internet Explorer ou Mozilla Firefox permettant une mobilité accrue des collaborateurs. Cette mobilité n'est pas faite au détriment de la sécurité puisque tous les échanges sont réalisés par l'intermédiaire de communications sécurisées par TLS/SSL/AES.

Pour obtenir la liste exhaustive des fonctionnalités de Zimbra, suivez ce lien.

Pour renforcer notre solution de travail collaboratif, elle se doit de disposer d'un serveur de fichiers performant et à la fois simple pour les utilisateurs (un simple lecteur réseau dans leur poste de travail). Samba est un serveur de fichiers qui a fait ses preuves faces à l'environnement Active Directory de Microsoft et va donc parfaitement répondre à nos besoins. De plus, nous allons le configurer pour utiliser l'annuaire des utilisateurs présent dans Zimbra (basé sur openldap) pour centraliser toutes les informations d'administration.

L'accès au serveur de fichiers sera totalement sécurisé par le service OpenVPN. Il assure le chiffrement des communications entre le client et le serveur mais aussi l'authentification par certificats des utilisateurs qui vont établir une connexion. Il permettra aussi aux administrateurs du parc machines de prodiguer une assistance à distance grâce à VNC (ultravnc ou realvnc) rapidement et simplement.

Pré requis

Les performances minimales du serveur doivent se situer dans la moyenne actuelle à savoir, un processeur 1.8Ghz, un disque dur de 80 Go (RAID préférable), 512 Mo de mémoire, et disposer d'une connexion de 100 Mbits. La distribution utilisée est DEBIAN ETCH 4.0.

Pour ce manuel je vais partir d'un serveur disposant du système de base seulement avec un accès SSH opérationnel avec Putty comme client SSH pour permettre de copier/coller la plupart des informations rapidement et WinSCP pour permettre de transférer des fichiers par SSH simplement.

Etapes à suivre

1- Configuration du serveur DNS (BIND9)
2- Préparation de l'environnement à Zimbra
3- Installation de Zimbra
4- Préparation de l'installation de Samba
5- Installation de Samba
6- Configuration des services d'authentification
7- Installation et configuration de OpenVPN sur le serveur
8- Installation et configuration de OpenVPN sur le client

Par Julien « MightyDucks » SIMON - Julien.SIMON at Mighty-Studio.net

ZIMBRA Collaboration Suite – SAMBA – OPENVPN

Première Etape : Configuration du serveur DNS (BIND9)

Installation

Debian 4.0 ETCH propose une Base de données de paquets précompilés qui permet de déployer simplement tous les services dont nous avons besoin.

Pour installer le serveur de nom de domaine, taper la commande suivante :

```
aptitude install bind9
```

Le système de paquets synaptic se charge alors de récupérer tous les éléments pour permettre l'installation du service demandé.

Configuration

Maintenant que le serveur de nom de domaine est installé, il faut le configuré pour qu'il reflète notre propre nom de domaine. Tout au long de ce manuel j'utiliserai le domaine mighty-studio.net pour simplifier les explications. A vous d'adapter ma configuration à vos paramètres.

Nous allons commencer par définir toutes les informations de notre domaine. Créer un fichier « domaine.tld.hosts » dans le répertoire « /etc/bind/ ». Techniquement, on parlera de définition de zone primaire pour le domaine.

```
Exemple : pico /etc/bind/mighty-studio.net.hosts
```

Copier dedans mon modèle de fichier de configuration modifié avec votre propre nom de domaine.

```
$ttl 38400
mighty-studio.net.          IN    SOA    mighty-studio.net.  admin@ mighty-studio.net. (
                                          2007060810
                                          10800
                                          3600
                                          604800
                                          38400 )
mighty-studio.net.          IN    NS     mighty-studio.net.
mighty-studio.net.          IN    NS     sdns1.ovh.net
www.mighty-studio.net.      IN    A      91.121.xxx.xxx
ftp.mighty-studio.net.      IN    A      91.121.xxx.xxx
mail.mighty-studio.net.     IN    A      91.121.xxx.xxx
mighty-studio.net.          IN    A      91.121.xxx.xxx
mighty-studio.net.          IN    MX     10  mighty-studio.net.
mail.mighty-studio.net.     IN    MX     10  mighty-studio.net.
```


Par Julien « MightyDucks » SIMON - Julien.SIMON at Mighty-Studio.net

Pour fermer « pico » appuyer sur « CTRL+X ». Dans ce fichier, on déclare donc les informations de la zone, les alias sur le domaine, les informations de contact, pour la messagerie, … Pour plus d'info, recherchez des informations pour chaque type d'enregistrement (NS SOA A CNAME MX).

Maintenant, nous allons déclarer notre zone au serveur DNS. Pour cela, ouvrez le fichier « named.conf.local » dans « /etc/bind/ ».

```
Exemple : pico /etc/bind/named.conf
```

Ajouter les informations suivantes en les adaptant :

```
zone "mighty-studio.net" {
        type master;
        file "/etc/bind/mighty-studio.net.hosts" ;
        };
```

Sauvegardez le contenu du fichier. Nous allons redémarrer le serveur DNS pour que la nouvelle configuration soit prise en compte.

```
/etc/init.d/bind9 restart
```

La propagation des informations que vous venez de donner peut prendre jusqu'à 48H suivant votre fournisseur d'accès à Internet. En effet, les modifications DNS ne sont pas instantanées. Pour savoir si les informations sont répliquées sur Internet, lancez un ping sur votre nom de domaine. Dés que vous recevez une réponse cela signifie que les informations ont bien été propagées. Vérifiez aussi que les informations que vous avez saisies sont correctes en tapant :

```
dig mighty-studio.net any
```

Cette commande effectue une requête DNS pour récupérer tous les enregistrements « IN » que vous avez déclaré. Cela doit donc vous retourner quelque chose de similaire à la capture ci-dessous :

```
; <<>> DiG 9.3.4 <<>> mighty-studio.net any
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 44177
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;mighty-studio.net.                IN      ANY
;; ANSWER SECTION:
mighty-studio.net.          38400   IN      A       91.121.xxx.xxx
mighty-studio.net.          38400   IN      MX      10 mighty-studio.net.
mighty-studio.net.          38400   IN      SOA     mighty-studio.net. admin@mighty-studio.net. 200706
0810 10800 3600 604800 38400
mighty-studio.net.          38400   IN      NS      sdns1.ovh.net.
mighty-studio.net.          38400   IN      NS      mighty-studio.net.
```


Par Julien « MightyDucks » SIMON - Julien.SIMON at Mighty-Studio.net

```
;; Query time: 12 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sat Jun  9 15:12:03 2007
;; MSG SIZE  rcvd: 142
```

Maintenant, on va modifier le nom de machine du serveur (host). Ouvrez le fichier « /etc/hosts »

```
pico /etc/hosts
```

Celui-ci doit refléter la configuration des domaines présents sur votre machine. Voici les 4 premières lignes de mon fichier « hosts » (le reste concerne IPv6 mais ne nous intéresse pas encore, dans un futur éloigné oui ☺)

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       localhost.localdomain localhost
91.121.xxx.xxx   mighty-studio.net domaine.tld autredomaine.tld
```

Ensuite taper la commande « hostname » suivie de votre nom de domaine.

```
Exemple : hostname mighty-studio.net
```

Voilà la configuration complète de vos DNS est terminée. On fait un redémarrage de sa machine pour valider et charger toute la nouvelle configuration correctement.

```
reboot
```

Une fois redémarré, l'invite du serveur via SSH doit correspondre à votre nom de domaine.

```
Exemple : mighty-studio:~#
```

L'extension du domaine n'est pas affichée, c'est normal.

Nous allons maintenant passer à la préparation du système pour installer Zimbra.

Deuxième Etape : Préparation de l'environnement pour Zimbra

Zimbra n'a pas encore été complètement porté pour Debian 4.0 ETCH. Nous devons donc utiliser des ruses pour permettre son installation sans problèmes.

Pour commencer, nous allons faire croire à Zimbra que notre version de Debian est la 3.1 Sarge en entrant la commande suivante :

```
echo "3.1" > /etc/debian_version
```

Installez les bibliothèques de composants suivantes qui sont requises par Zimbra :

```
aptitude  install sudo curl fetchmail libgmp3c2 libssl0.9.7 libdb3 libstdc++5 libexpat1  libpcre3
```

Si des questions durant l'installation vous sont posées, laissez les réponses par défaut.

On modifie les droits sur notre répertoire « /tmp »

```
chmod a+rwx /tmp
```

On se place dans le répertoire « /tmp » et on récupére Zimbra

```
wget http://ovh.dl.sourceforge.net/sourceforge/zimbra/zcs-4.5.5_GA_838.DEBIAN3.1.tgz
```

On décompresse l'archive obtenue

```
tar xzf zcs-4.5.5_GA_838.DEBIAN3.1.tgz
```

On entre dans le répertoire ainsi extrait

```
cd zcs
```

On édite le script d'installation pour lui faire reconnaître correctement notre système

```
pico /tmp/zcs/util/utilfunc.sh
```

A l'aide de « CTRL+W », trouvez la ligne

```
if [ $PLATFORM = "UBUNTU6" ]; then
```

que vous remplacez par

```
if [ $PLATFORM = "UBUNTU6" -o $PLATFORM = "DEBIAN3.1" ]; then
```

Voilà nous sommes prêts à lancer l'installation de Zimbra.

Troisième Etape : Installation de Zimbra

On lance l'installation de la suite Zimbra avec la commande

```
sh install.sh
```

Vérifiez de bien être dans le répertoire zcs (/tmp/zcs)

Suivez le processus d'installation en répondant « yes » aux questions posées.

```
Select the packages to install
Install zimbra-ldap [Y]
Install zimbra-logger [Y]
Install zimbra-mta [Y]
Install zimbra-snmp [Y]
Install zimbra-store [Y]
Install zimbra-spell [Y]
Installing:
   zimbra-core
   zimbra-ldap
```


Par Julien « MightyDucks » SIMON - Julien.SIMON at Mighty-Studio.net

```
    zimbra-logger
    zimbra-mta
    zimbra-snmp
    zimbra-store
    zimbra-apache
    zimbra-spell
The system will be modified.  Continue? [N] Y
```

Si tout va bien, votre installation doit déboucher sur un menu affichant tous les paramètres crées.

```
Main menu
  1) Hostname:                          mighty-studio.net
  2) Ldap master host:                  mighty-studio.net
  3) Ldap port:                         389
  4) Ldap password:                     set
  5) zimbra-ldap:                       Enabled
  6) zimbra-store:                      Enabled
       +Create Admin User:              yes
       +Admin user to create:           admin@mighty-studio.net
******* +Admin Password                 UNSET
       +Enable automated spam training:  yes
       +Spam training user:              spam.onq3rxcag@mighty-studio.net
       +Non-spam(Ham) training user:     ham.tg0rjjov9_@mighty-studio.net
       +Global Documents Account:        wiki@mighty-studio.net
       +SMTP host:                       mighty-studio.net
       +Web server HTTP port:            80
       +Web server HTTPS port:           443
       +Web server mode:                 http
       +Enable POP/IMAP proxy:           no
       +IMAP server port:                143
       +IMAP server SSL port:            993
       +POP server port:                 110
       +POP server SSL port:             995
       +Use spell check server:          yes
       +Spell server URL:                http://mighty-studio.net:7780/aspell.php
  7) zimbra-mta:                        Enabled
  8) zimbra-snmp:                       Enabled
  9) zimbra-logger:                     Enabled
 10) zimbra-spell:                      Enabled
  r) Start servers after configuration  yes
  s) Save config to file
  x) Expand menu
  q) Quit

Address unconfigured (**) items  (? - help)
```


Par Julien « MightyDucks » SIMON - Julien.SIMON at Mighty-Studio.net

Il faut cependant définir le mot de passe du compte administrateur. Pour le définir, tapez 6 puis 4 pour accéder à la bonne section. Entrez votre mot de passe administrateur.

Pensez aussi à récupérer le mot de passe administrateur de l'annuaire LDAP. Il faut se rendre dans le menu principal et sélectionner la 4ème option « Ldap Password ». Un mot de passe s'affiche entre crochet. Copier le quelque part, valider puis tapez « r » pour revenir au menu principal.

Note : Si vous le souhaitez, vous pouvez redéfinir plus tard le mot de passe LDAP, en utilisant les commandes suivantes,

```
su – zimbra
zmldappasswd --root NouveauMdpLDAP
exit
```

On peut maintenant appliquer la configuration que nous venons de définir en appuyant sur la touche « a »

Le processus d'installation se termine. Rendez vous avec votre navigateur Internet à l'url suivante pour tester votre suite : http://domaine.tld/ dans mon cas http://mighty-studio.net ; et entrez le pseudo « admin » ainsi que le mot de passe saisi lors de l'installation. Bienvenue sur l'interface utilisateur de base ☺.

Pour accéder à la console d'administration, pointez votre navigateur à l'URL suivante :

https://domaine.tld:7071/zimbraAdmin

Soit dans mon cas :

https://mighty-studio.net:7071/zimbraAdmin

Si vous ne parvenez pas à accéder à ces URL, il est fort probable que votre configuration soit incorrecte quelque part et je vous encourage à reprendre les différentes étapes pour vérifier vos paramètres.

Note : Pour désinstaller Zimbra lors d'un disfonctionnement, allez dans le répertoire « /tmp/zcs » puis tapez la commande

```
sh install.sh –u
```

Note : Si l'installation s'est bien déroulée, vous pouvez remettre le fichier « /etc/debian_version » à « 4.0 », seulement pour l'instant pour chaque mise à jour vous serez obligé de remettre à « 3.1 ».

DURANT TOUTE LA FIN DE CE MANUEL, VEUILLEZ NE PAS VOUS DECONNECTEZ DE L'INTERFACE D'ADMINISTRATION POUR MENER A BIEN TOUS LES OPERATIONS QUI VONT SUIVRE.

ZIMBRA Collaboration Suite – SAMBA – OPENVPN

Quatrième Etape : Préparation à l'installation de Samba

Les parties qui suivent proviennent du wiki officiel de Zimbra enrichies et modifiées par mes soins : Version Originale

Pour permettre à samba de communiquer correctement avec Zimbra, quelques modifications sont nécessaires. Une collection de classes java doit être ajoutée pour permettre à Zimbra un certain nombre d'opérations sur l'annuaire LDAP.

Pour cela, créez tout d'abord le répertoire suivant,

```
mkdir /opt/zimbra/lib/ext/zimbraldaputils
```

Puis placez vous à l'intérieur,

```
cd /opt/zimbra/lib/ext/zimbraldaputils/
```

Téléchargez la collection de classes java (fichier jar),

```
wget http://gallery.zimbra.com/files/uploads/aff24/zimbraldaputils.jar
```

Et enfin lui attribuer les bons droits.

```
chmod 0755 zimbraldaputils.jar
```

On redémarre Zimbra

Je l'explique une fois pour toutes, pour redémarrer Zimbra, voici la procédure à suivre, tapez,

```
su – zimbra
zmcontrol stop
zmcontrol start
exit
```

A chaque fois que vous demanderez de redémarrer Zimbra, suivez cette procédure.

Télécharger sur votre ordinateur personnel (différent du serveur) les deux extensions suivantes :

ZimbraPosixAccount
ZimbraSamba

Désarchivez les sur votre bureau, puis éditez les fichiers

```
zimbra_posixaccount/config_template.xml
zimbra_samba/config_template.xml
```

Modifiez la ligne,

```
dc=gregzimbra1,dc=zimbra,dc=com
```

par,

```
dc=domaine,dc=tld
```

dans mon exemple,

```
dc=mighty-studio,dc=net
```

Placez-vous **à l'intérieur** du répertoire « zimbra_posixaccount », selectionnez tous les fichiers et archivez les en .zip (zimbra_posixaccount.zip). Répétez l'opération pour le répertoire « zimbra_samba » avec le nom d'archive « zimbra_samba.zip »

Connectez-vous à l'interface d'administration, http://domaine.tld:7071/zimbraAdmin, puis allez dans le menu Admin Extentions. Cliquez sur le bouton « Deploy » et sélectionnez dans la fenêtre « zimbra_posixaccount.zip ». Recommencer l'opération pour « zimbra_samba.zip ».

On redémarre Zimbra suivant la procédure donnée plus haut.

Aucune erreur ne doit apparaître !

Maintenant, nous allons charger les schémas de données pour que Zimbra les chargent dans l'annuaire LDAP.

On récupère la documentation Samba

```
aptitude install samba-doc
```

On désarchive le schéma de données Samba et on l'envoie dans le répertoire de configuration des schémas de l'annuaire

```
gunzip -c /usr/share/doc/samba-doc/examples/LDAP/samba.schema.gz >
/opt/zimbra/openldap/etc/openldap/schema/samba.schema
```

Note : La commande doit être entrée sur une seule ligne !

On configure le service OPENLDAP qui gère annuaire pour qu'il charge les schémas

```
pico /opt/zimbra/conf/slapd.conf.in
```

En dessous de la ligne,

```
include         "/opt/zimbra/lib/conf/zimbra-ext.schema"
```

ajoutez,

```
include         "/opt/zimbra/openldap/etc/openldap/schema/nis.schema"
include         "/opt/zimbra/openldap/etc/openldap/schema/samba.schema"
```

et ajoutez à la fin du fichier,

```
index uidNumber           eq
index gidNumber           eq
index memberUID           eq
index sambaSID            eq
index sambaPrimaryGroupSID  eq
index sambaDomainName       eq
```

On redémarre Zimbra.

Entrez les deux commandes suivantes

```
zmprov mcf +zimbraAccountExtraObjectClass posixAccount
zmprov mcf +zimbraAccountExtraObjectClass sambaSamAccount
```

Les étapes de configuration pré requises à l'installation de Samba sont maintenant terminées.

Cinquième Etape : Installation de Samba

Nous allons maintenant installer le serveur de fichiers Samba. Pour lancer l'installation, taper la commande suivante

```
aptitude install samba
```

Laissez par défaut toutes les réponses aux questions posées durant l'installation.

On arrête samba avant toute chose

```
/etc/init.d/samba stop
```

On renomme le fichier de configuration par défaut

```
mv /etc/samba/smb.conf /etc/samba/smb.conf.orginal
```

On crée notre propre fichier de configuration

```
pico /etc/samba/smb.conf
```

Collez puis modifiez la configuration suivante

```
[global]
  workgroup = mighty-studio
  netbios name = mighty-studio
```

```
  os level = 33
  preferred master = yes
  enable privileges = yes
  server string = Serveur de fichiers mighty-studio
  wins support =yes
  dns proxy = no
  name resolve order = wins bcast hosts
  log file = /var/log/samba/log.%m
  log level = 3
  max log size = 1000
  syslog only = no
  syslog = 3
  panic action = /usr/share/samba/panic-action %d
  security = user
  encrypt passwords = true
  ldap passwd sync = yes
  passdb backend = ldapsam:ldap://mighty-studio.net/ ; A MODIFIER
  ldap admin dn = "uid=zimbra,cn=admins,cn=zimbra" ; A NE PAS MODIFIER
  ldap suffix = dc=mighty-studio,dc=net ; A MODIFIER
  ldap group suffix = ou=groups
  ldap user suffix = ou=people
  ldap machine suffix = ou=machines
  obey pam restrictions = no
  passwd program = /usr/bin/passwd %u
  passwd chat = *Enter\snew\sUNIX\spassword:* %n\n *Retype\snew\sUNIX\spassword:* %n\n
*password\supdated\ssuccessfully* .
  domain logons = yes
  logon path = \\mighty-studio.fr\%U\profile ; A MODIFIER
  logon home = \\mighty-studio.fr\%U ; A MODIFIER
  add user script = /usr/sbin/adduser --quiet --disabled-password --gecos "" %u
  add machine script = /usr/sbin/adduser --shell /bin/false --disabled-password --quiet --gecos
"machine account" --force-badname %u
  socket options = TCP_NODELAY
  domain master = yes
  local master = yes
  hosts allow = 10.8. 127.

[homes]
  comment = Home Directories
  browseable =yes
  read only = No
  valid users = %S
```

```
[partage]
  comment = Repertoire commun
  browseable = Yes
  read only = No
  writeable = yes
  path = /home/partage

[netlogon]
  comment = Network Logon Service
  path = /var/lib/samba/netlogon
  guest ok = yes
  locking = no

[profiles]
  comment = Users profiles
  path = /var/lib/samba/profiles
  read only = No

[profdata]
  comment = Profile Data Share
  path = /var/lib/samba/profdata
  read only = No
  profile acls = Yes
```

On enregistre le mot de passe de l'administrateur de l'annuaire LDAP (celui récupéré lors de l'installation. Si vous l'avez loupé vous pouvez chercher sur zimbra comment le changer)

```
smbpasswd -w test123
```

Pensez à créer les répertoires pour le partage, le netlogon, etc. (« mkdir /home/partage » par exemple) Je vous conseille fortement un chmod 077 dessus puisque c'est samba qui s'occupe des droits d'accès (plus de détails en consultant la doc samba).

On peut maintenant lancer samba,

```
/etc/init.d/samba start
```

On vérifie qu'il tourne réellement !! Pour cela, tapez la commande

```
ps x | grep smbd
```

Cela doit vous renvoyer au moins deux lignes !

```
6501 ?       Ss     0:00 /usr/sbin/smbd –D
6506 ?       S      0:00 /usr/sbin/smbd –D
6616 pts/0   S+     0:00 grep smbd
```

Je vous rassure tout de suite, le plus dur est fait !

Passons maintenant à la configuration du serveur OpenVPN.

Sixième Etape : Configuration des services d'authentification

Pour pouvoir utiliser les mêmes couples identifiant/mot de passe dans Zimbra et Samba, il reste à installer les bibliothèques adéquates, libnss-ldap et libpam-ldap, pour permettre la communication entre nos deux services. Pour les installer, tapez la commande suivante,

```
aptitude install libnss-ldap libpam-ldap
```

Laissez les réponses par défaut, nous allons éditer directement les fichiers de configuration pour simplifier mon explication.

Nous allons éditer le fichier de configuration de la bibliothèque NSS LDAP, « libnss-ldap.conf », pour tapez la commande,

```
pico /etc/libnss-ldap.conf
```

Recherchez les lignes suivantes,

```
#host 127.0.0.1
base dc=example,dc=net
uri ldap://127.0.0.1/
#binddn cn=proxyuser,dc=padl,dc=com
#bindpw secret
#rootbinddn cn=manager,dc=padl,dc=com
```

Que vous allez modifier selon le modèle suivant,

```
host cerbx.fr
base dc=mighty-studio,dc=net # A modifier
#uri ldap://127.0.0.1/
binddn uid=zimbra,cn=admins,cn=zimbra
bindpw VotreMDPLdap # A modifier
rootbinddn uid=zimbra,cn=admins,cn=zimbra
```

Il faut créer un fichier contenant le mot de passe pour la derniére ligne ci-dessus, tapez la commande suivante pour procéder à sa saisie,

```
echo "VotreMdpLDAP" > /etc/libnss-ldap.secret
```

Maintenant nous allons copier notre configuration de NSS vers PAM puisque leurs fichiers de configuration sont sensiblement identiques, pour cela, tapez les commandes suivantes,

```
cp /etc/libnss-ldap.conf  /etc/pam_ldap.conf
cp /etc/libnss-ldap.secret /etc/pam_ldap.secret
```

Revenez dans l'interface Web d'administration de Zimbra pour créer le groupe POSIX (Linux) qui contiendra vos utilisateurs. Dans Posix Groups, cliquez sur « New » et remplissez les différents champs sans modifier celui indiquant un numéro de groupe. Enregistrez le groupe puis repassez sur votre client SSH pour tester la bonne communication de PAM avec l'annuaire LDAP en tapant la commande suivante,

```
getent group
```

Si votre configuration est bonne, vous devriez voir le groupe que vous venez de créer.

Nous allons maintenant crée un utilisateur dans Zimbra toujours pour tester nos paramètres. Pour cela, dans Zimbra, allez dans « Accounts » et cliquez « New ». Saisissez les informations relatives à l'utilisateur et suivez l'assistant jusqu'à la fenêtre « Posix Account ». Complétez les champs requis et cliquez sur « Next » pour aller dans la fenêtre « Samba Account ». Complétez à nouveau les champs requis et cliquez « Finish ». Pour tester la bonne lecture par PAM des informations relatives aux utilisateurs, tapez la commande suivante,

```
getent passwd
```

Vous devriez voir apparaître le ou les comptes Zimbra que vous venez de créer. Si tel est le cas, vous pouvez passer à la configuration du service OpenVPN afin de sécuriser les échanges avec votre serveur de fichiers.

Septième Etape : Installation et configuration de OpenVPN

OpenVPN est un service permettant de créer un tunnel chiffré entre deux machines. Il repose sur une authentification à base de certificats et ou de clés pré partagées et utilise divers protocole de chiffrement comme BlowFish, AES, etc.

Ce manuel est grandement inspiré de celui présent sur le site de Coagul.org

Pour l'installer, tapez la commande suivante,

```
aptitude install openvpn
```

en laissant les réglages par défaut.

Nous allons maintenant créer tous les certificats nécessaires au serveur et à au moins un client. Pour cela, rendez vous dans le répertoire suivant,

```
cd /usr/share/doc/openvpn/examples/easy-rsa/
```

On édite les scripts pour qu'ils reflètent les couleurs françaises :

```
pico vars
```

Modifiez en fonction de vos envies ☺

```
export KEY_COUNTRY=FR
export KEY_PROVINCE=France
export KEY_CITY=BORDEAUX
export KEY_ORG="Mighty-Studio.net"
export KEY_EMAIL="mightyducks@mighty-studio.net"
```

On applique nos changements (A faire avant chaque création de certificats clients)

```
. ./vars
```

On réinitialise le répertoire « keys » pour permettre la création du nouveau jeu de certificats

```
./clean-all
```

Le script suivant permet de créer dans « keys » le certificat principal du serveur « ca.crt » et la clé correspondante « ca.key » :

```
./build-ca
```

ATTENTION : Pour les questions, la plupart des champs sont renseignés par défaut sauf le « Common Name » qu'il faut renseigner manuellement. Exemple « mighty-studio.net ».

Le script suivant permet de créer dans « keys » le certificat « mighty-studio.net.crt » et la clé « mighty-studio.net.key » pour le serveur VPN nommé par exemple « mighty-studio.net » :

```
./build-key-server mighty-studio.net
```

ATTENTION : Pour les questions, tous les champs sont renseignés par défaut sauf le « Common Name » qu'il faut renseigner manuellement. Exemple «mighty-studio.net ». Personnellement, je n'ai pas renseigné le champ « password »

Le script suivant permet de créer dans « keys » le certificat « mightyducks.crt » et la clé « mightyducks.key » pour le client VPN nommé par exemple « mightyducks » :

```
./build-key mightyducks
```

ATTENTION : Il faudra renouveler cette opération pour chaque client. Pour les questions, tous les champs sont renseignés par défaut sauf le « Common Name » qu'il faut renseigner manuellement. Exemple « mightyducks». Chaque « Common Name » de chaque client doit être différent. Personnellement, je n'ai pas renseigné le champ « password »

Le script suivant permet de créer dans « keys » le fichier « dh1024.pem » :

```
./build-dh
```

Ce fichier sert lors de l'établissement de la connexion entre les deux parties pour s'échanger les clés au travers d'Internet (Qui est un cannal de connexion non sécurisé). Il permet d'établir une clé

symétrique dont la longueur est paramétrable en fonction des lois de l'Etat où est hébergé le serveur, qui sera utilisée pour toutes les opérations de chiffrement.

Maintenant que notre jeu de certificats est crée, le plus simple est de copier les 4 fichiers dans le dossier « /etc/openvpn »

```
cp ./keys/ca.crt /etc/openvpn/
cp ./keys/ca.key /etc/openvpn/
cp ./keys/mighty-studio.net.crt /etc/openvpn/
cp ./keys/mighty-studio.net.key /etc/openvpn/
cp ./keys/dh1024.pem /etc/openvpn/
```

Pour configurer le serveur, le plus simple est de partir du fichier d'exemple « server.conf.gz », qu'il faut donc décompresser et mettre en place dans « /etc/openvpn », en tapant les commandes suivantes,

```
cd /usr/share/doc/openvpn/examples/sample-config-files/
gunzip server.conf.gz
cp server.conf /etc/openvpn/
```

On édite le fichier de configuration que l'on vient de copier selon les besoins à satisfaire pour la sécurité du système (Rien ne sert d'employer le plus fort chiffrement si les données échangées ne sont que des photos. Renseignez vous sur le degré de sécurité assuré par chaque protocole pour ne pas utiliser des ressources inutilement)

```
pico /etc/openvpn/server.conf
```

A titre d'exemple, voici mon fichier de configuration,

```
;Protocole utilisé (Le protocole udp est plus sécurisé que le tcp)
proto udp
;Type d'interface réseau virtuelle créée
dev tun
;Nom des fichiers servant à l'authentification des clients via OpenSSL
ca ca.crt
cert mighty-studio.net.crt
key mighty-studio.net.key
dh dh1024.pem
;Adresse du réseau virtuel (Le serveur aura l'adresse 10.8.0.1)
server 10.8.0.0 255.255.255.0
 ;Permet que les utilisateurs se voient entre eux. Super pour l'administration à distance (VNC & Co)
client-to-client
 ;protocole de chiffrement à utiliser
cipher AES-128-CBC   # AES bon compromis performance/sécurité
 ;On réactive le tunnel pour éviter qu'il ne se coupe
keepalive 10 120
```

```
;Cette ligne active la compression
comp-lzo
;Ces lignes permettent de rendre persistante la connexion
persist-key
persist-tun
 ;fichier de log (inutile car tout est redirigé dans /var/log/syslog automatiquement)
status openvpn-status.log
;Cette ligne permet d'indiquer le niveau de log souhaité (de 1 à 9) pour déboguer les problèmes
verb 1
```

On applique la configuration du service OpenVPN avec la commande,

```
/etc/init.d/openvpn restart
```

Ne pas hésiter à regarder dans les logs que tout c'est bien passé,

```
tail -100 /var/log/syslog
```

Cette commande vous affiche les 100 dernières lignes du fichier de log centralisé de votre serveur.

Il ne reste plus qu'à configurer votre client pour qu'il se connecte au service OpenVPN.

Huitième Etape : Configuration du Client OpenVPN sur Windows

Rien de plus simple ! On commence par télécharger sur openvpn.se le fichier d'installation de OpenVPN GUI avec les pilotes pour créer une interface virtuelle TAP/TUN (Je n'explique pas à quoi cela sert mais c'est relativement simple. Ce sont des interfaces virtuelles qui seront utilisées pour établir la connexion avec le serveur. Bref des sous interfaces de votre carte réseau.)

L'installateur vous permet d'installer en tant que service la connexion et de cacher l'interface virtuelle comme bon vous semble ☺

Ensuite, il est nécessaire de copier dans le dossier « C:\Program Files\OpenVPN\config » les fichiers servant à l'authentification du client crée précédemment

```
ca.crt
mightyducks.crt
mightyducks.key
```

Ensuite, il faut modifier le fichier de configuration pour l'adapter à votre cas. Pour éditer le fichier, il est possible de faire un clic droit sur l'icône « OpenVPN » situé à gauche de l'heure et de choisir l'option « Edit config ».

Voici un exemple de fichier que j'utilise pour mes clients à modifier selon votre domaine :

```
client
dev tun
```

```
proto udp
remote mighty-studio.net 1194 # A modifier !
resolv-retry infinite
nobind
persist-key
persist-tun
ca ca.crt
cert mightyducks.crt # A modifier !
key mightyducks.key # A modifier !
cipher AES-128-CBC
comp-lzo
verb 1
```

Pour lancer la connexion, il suffit de faire un clic droit sur l'icône « OpenVPN » situé à gauche de l'heure et de choisir l'option « Connect ».

Si tout se passe bien, une fenêtre affichant les logs doit s'afficher et une fois la connexion effectuée, le réseau est opérationnel.

En cas de problème, et pour trouver l'origine de celui-ci il faut augmenter le niveau des logs en changeant le paramètre « verb » du fichier de configuration :

```
verb 3 #-> Suffisamment de logs dans la plupart des cas.
verb 9 #-> Énormément de logs.
```

Une fois la connexion établie, il doit être possible de pinguer le serveur soit sur son adresse virtuelle (ex : ping 10.8.0.1)

Pour accéder aux ressources partagées du serveur, rendez vous dans votre Poste de Travail et tapez

\\10.8.0.1

On vous demande de vous authentifier. Utiliser alors un couple nom d'utilisateur, mot de passe crée dans Zimbra (Si ce n'est pas fait un tour dans la console d'administration s'impose☺)

Vos répertoires partagés doivent maintenant s'afficher. Essayer de copier/coller un fichier depuis votre pc. En cas d'erreurs de permissions, vérifiez les droits d'accès à vos différents partages. (chmod 0777 /home/partage par exemple. A affiner par la suite selon les besoins)

Les petits trucs en plus …

Permettre l'installation d'une nouvelle instance Apache2 pour votre site Public

Sans hésiter, je sais qu'elle est la question qui vous hante à ce stade : Zimbra, c'est super, mais comment je mets mon site public sur mon serveur puisque Zimbra utilise le port 80 par défaut ?

En basculant toute la plateforme Zimbra sur le port 443 (HTTPS) libérant ainsi le port 80 pour vos sites. Maintenant, si votre site requiert lui aussi du HTTPS, lisez sur le wiki Zimbra comment mettre en place le mod proxy apache2 pour rediriger correctement les requêtes au bon endroit.

Commandes à entrer pour basculer du port 80 http vers le port 443 https :

```
su – zimbra
zmtlsctl https
tomcat restart
exit
```

Vous pouvez ainsi installer une nouvelle instance apache2 pour gérer votre site publique. Vous pouvez installez les langages de scripts que vous souhaitez, mais vous devrez réutiliser l'instance MySQL utilisée par Zimbra.

Pour cela, vous devez récupérer le mot de passe « root » de MySQL. Le mot de passe root MySQL vous permet par la suite de configurer PHPMyAdmin, par exemple ou tout autre de vos scripts nécessitants une base de données.

Commandes à utiliser pour récupérer le mot de passe « root » :

```
su – zimbra
zmlocalconfig -s | grep mysql | grep password
```

Qui vous renvoie :

```
mysql_logger_root_password = AWHZ60JYaBw8_hVkA9NDVGh0irmp7xVz
mysql_root_password = lkAd7vkYI.Q_VeWt8uyL9kj0
zimbra_logger_mysql_password = 2iiyAVj3GeH0akkCe6M1o_HvY
zimbra_mysql_password = uMv4EsNqPZdK5htERx97VY5m
```

mysql_root_password vous ouvre les portes de votre instance MySQL et vous permet de l'utiliser en parallèle.

Passage de l'interface en Français

Un passage par le forum de la communauté Française s'impose. Laissez-vous guider par les modérateurs pour trouver le pack Français pour Zimbra ainsi que la procédure d'installation

C'EST FINI !

Tout fonctionne ? Bien joué ! Vous disposez maintenant d'une suite de travail collaboratif ultra performante que vous avez rendu encore plus complète en assurant la gestion de vos fichiers de manière sécurisée !

Pour compléter votre architecture, il est possible d'ajouter un service de VoIP. En effet, le système des modules Zimbra (Zimlets) permet de faire communiquer les deux sans soucis mais il fera l'objet d'une documentation suplémentaire.

Voilà, je crois que tout est dit. A vous de jouer !

Juste pour information : L'architecture que nous venons de mettre en place vous sera facturée une fortune chez grosbillou at gros$oft.world et sera uniquement compatible avec leurs propres outils moyennant quelques centaines d'euro supplémentaires.

RESSOURCES DOCUMENTAIRES :

Installation de Zimbra sur Debian 4.0 ETCH
Utilisation de Zimbra LDAP pour Samba
MySQL ROOT Password
OpenVPN GUI
OpenVPN Server
Coagul
WIKI Zimbra
PDC SAMBA

Auteur : Julien « MightyDucks » SIMON, Julien.SIMON at Mighty-Studio.net
Etudiant à L'IUT de Mont de Marsans, Département Génie des Télécommunications et Réseaux
Licence Pro Réseaux et Télécommunications : Administration et sécurité des réseaux informatiques