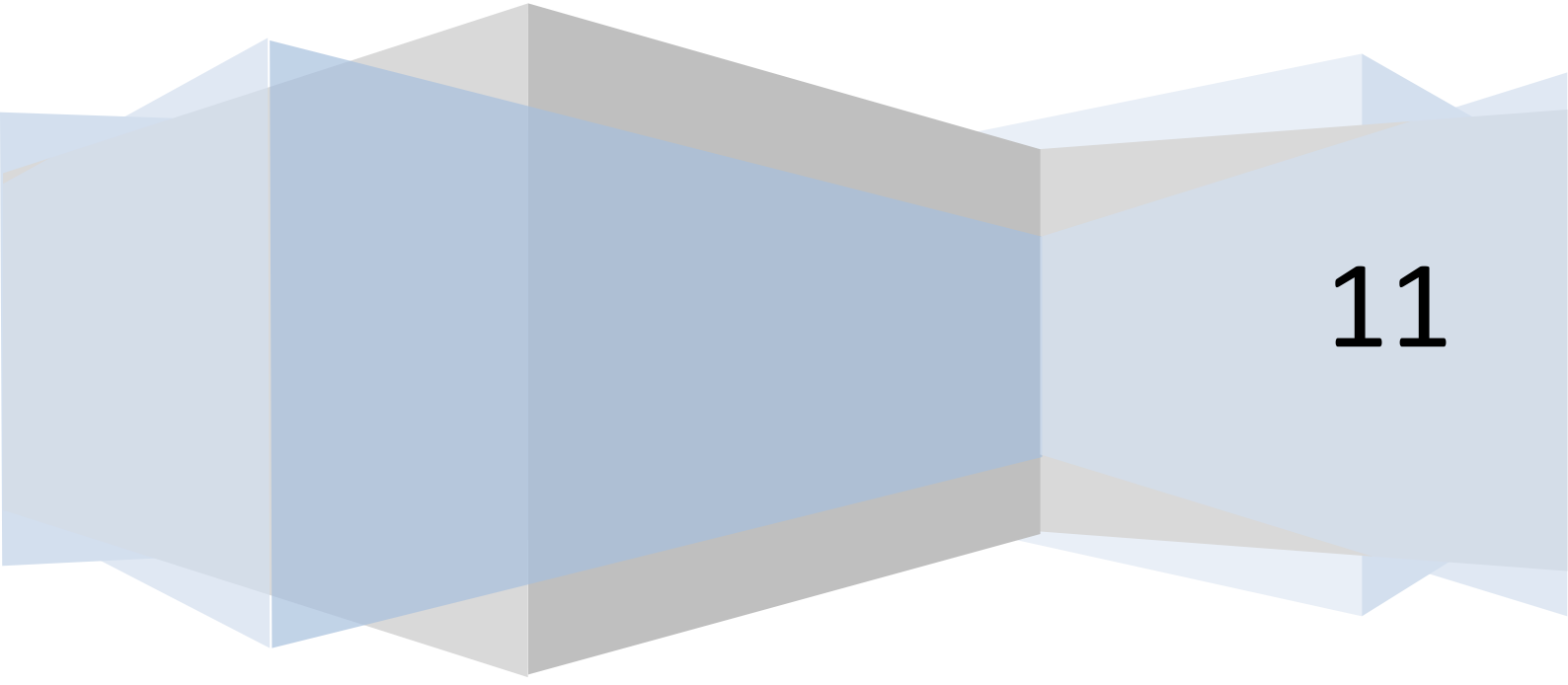
Société, Jean Miche la bidouille

Projet de mise en œuvre d'un serveur ftp sur serveur dédié

Etude de faisabilité



11

1. Contenu

2. Introduction

Louer un serveur dédié sur internet offre de nombreuses possibilités

- Peu de limitation de bande passante pour le téléchargement ou l'upload de fichiers
- Consommation électrique optimisée
- Taux de disponibilité élevé au regard d'un serveur hébergé chez soi
- Choix du système et des logiciels que l'on veut installer sur la machine (serveur ftp, web, forum, messagerie, messagerie instantanée, galerie photo, etc..)
- Ressources raisonnables (une centaine de Go de disque, processeur dédié uniquement au fonctionnement des services de l'utilisateur,..)

En contrepartie, il ne faut pas perdre de vue que la location d'un tel serveur nécessite des connaissances pour l'administrer. Ces serveurs sont souvent la cible de hackers qui profitent souvent de la négligence des administrateurs en laissant des services inutiles, en utilisant des mots de passe peu robuste, en ne chiffrant pas les accès au serveur, en n'installant pas régulièrement les correctifs du système d'exploitation.

3. Outils nécessaires

Les outils proposés sont gratuits et opensource

1. Pour le propriétaire du serveur :

Sous windows :

- Putty : client pour se connecter au serveur de manière sécurisé en ssh
  - http://the.earth.li/~sgtatham/putty/latest/x86/putty-0.61-installer.exe
- Winscp : Outil graphique pour échanger des fichiers à la manière d'un explorateur. La connection sera en ssh.
  - http://winscp.net/eng/docs/lang:fr
- Filezilla : pour se connecter en ftp sur le serveur.
  - http://www.filezilla.fr/

Sous linux :

- Openssh

1. Pour les clients:

- Filezilla : pour se connecter en ftp sur le serveur.
  - http://www.filezilla.fr/

2. Configurer les clients ssh

L'accès ssh sera réservé uniquement à l'administrateur du serveur.

1. Génération d'un trousseau de clés

L'utilitaire PuTTYgen installé lors de l'installation de Putty permet de gérer les trousseaux de clés utilisés par le protocole ssh.

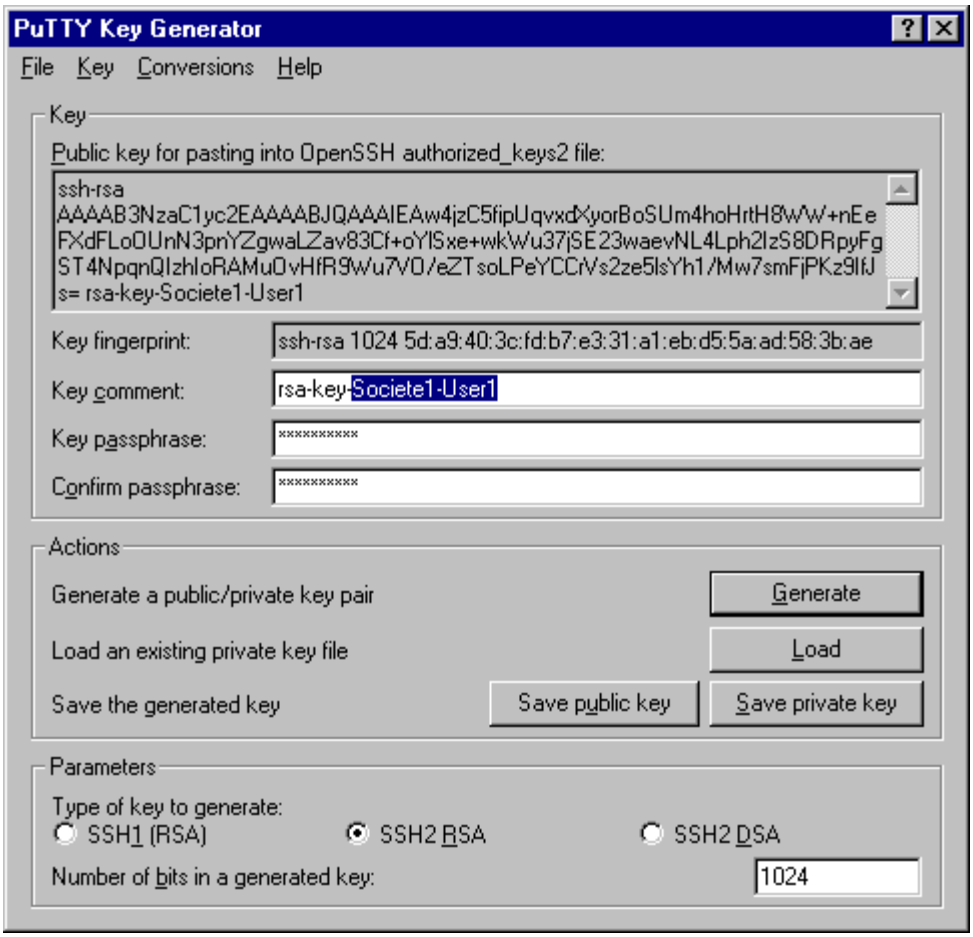
Voici la démarche de génération d'un trousseau de clés qui sera utilisé dans le cadre des connexions ssh.

Lancement de PuTTYgen

Sélection du type de clé à générer : Type de clé SSH2 RSA

Génération de la clé (bouton Generate)

NB : il faut déplacer la souris afin que la souris se génère

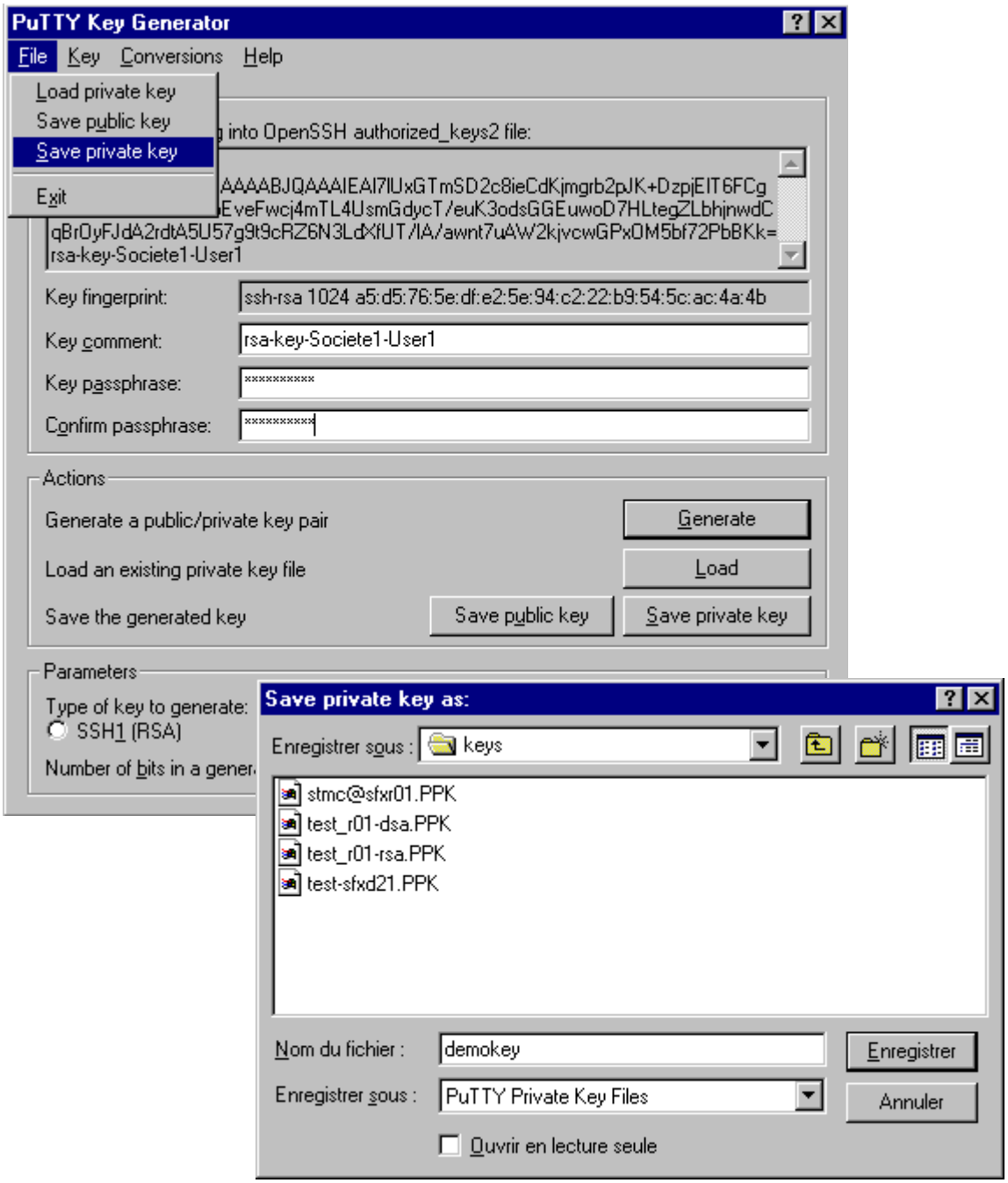


Saisie du nom de la clé (zone comment), et du mot de passe

NB : modifier le nom/commentaire associé à cette clé est très important car il permettra d'identifier précisément les différentes clés publiques présentes sur les serveurs.

Sauvegarde de la clé privée

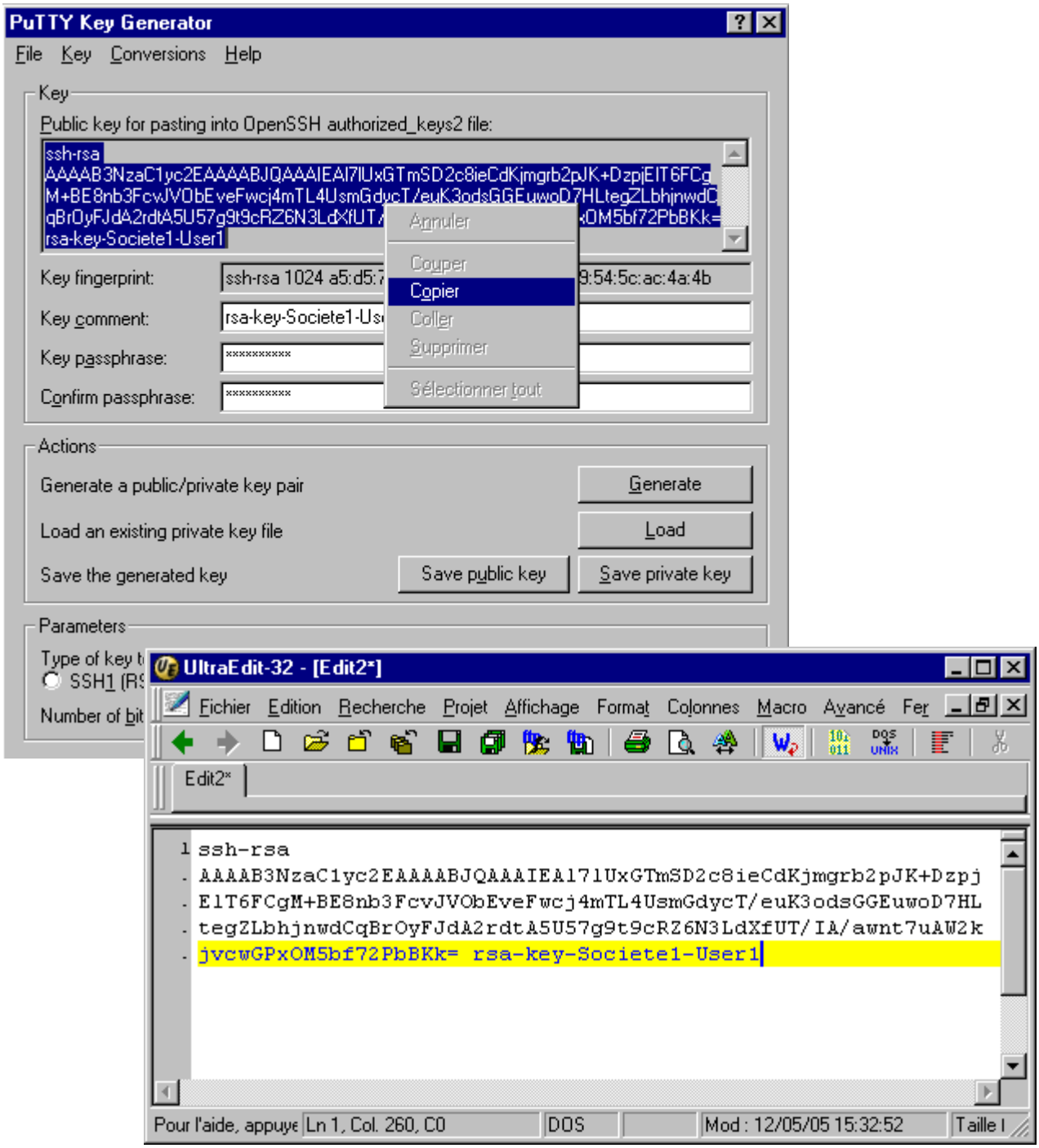
Clé qui servira pour le paramétrage des sessions ssh/sftp du poste vers les serveurs cibles



Il est également conseillé de faire une sauvegarde de la clé privé au format openssh si l'on veut se connecter depuis un client linux via un terminal. Sélectionnez conversion puis « export Openssh Key ».

Sauvegarde de la clé publique au format Openssh

Clé qui sera mise en place sur les serveurs cibles afin d'autoriser les ouvertures de sessions présentant la clé privée correspondante – pour cela il faut copier dans un fichier texte la zone « Public key for pasting into Openssh authorized_keys file »

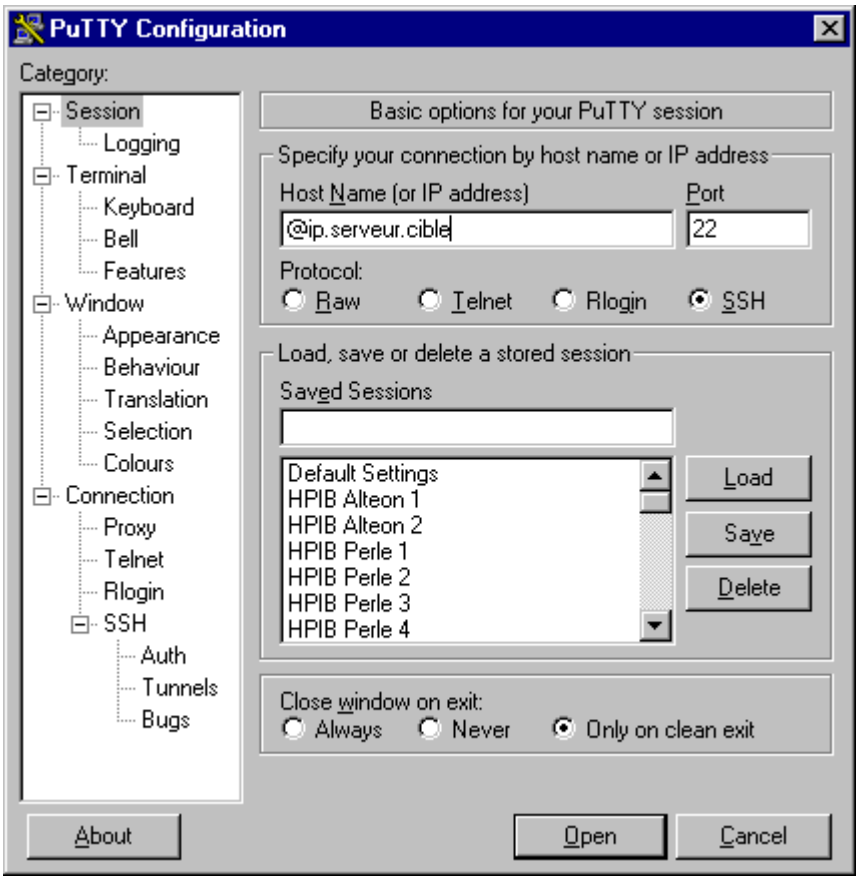


2. Utilisation PuTTY / WinSCP

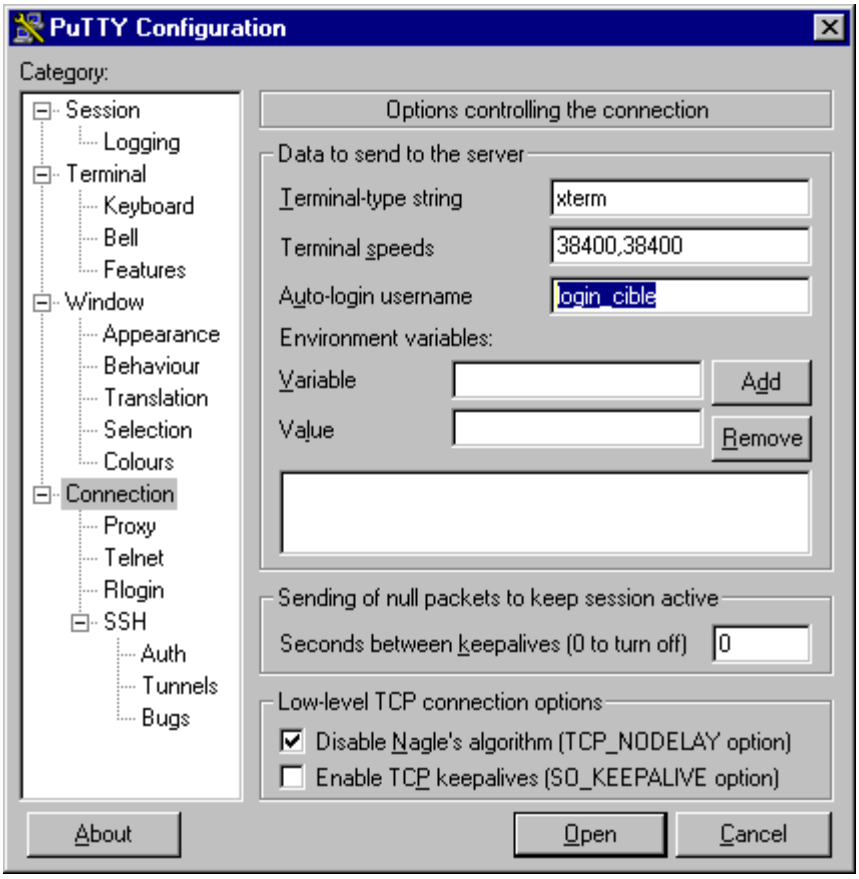
Ouverture de session ssh via PuTTY

- Lancer PuTTY

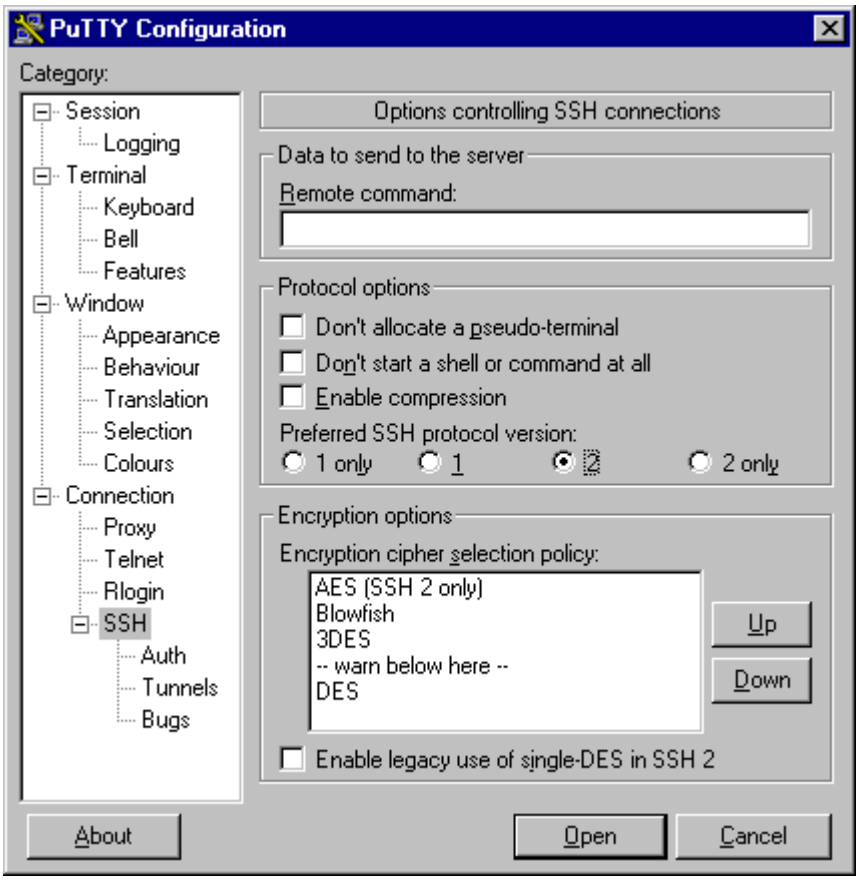
- Paramétrer la catégorie « session » :
  - Saisir l'adresse du serveur
  - Cocher le protocole ssh



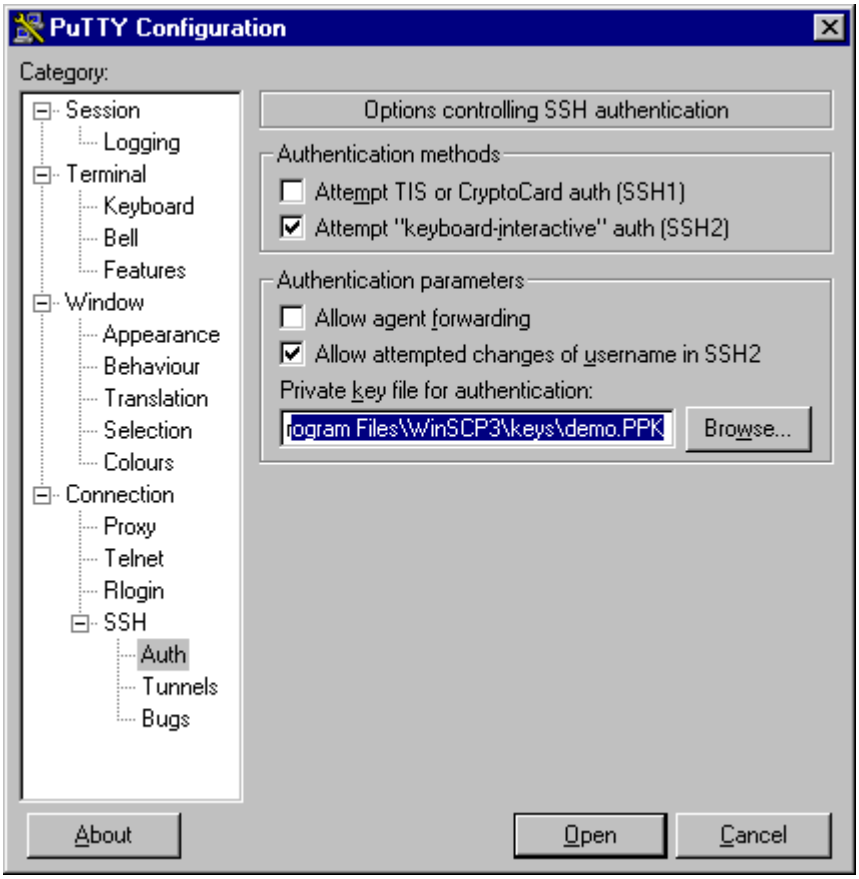
- Paramétrer la catégorie « connexion » :
  - Saisir le login unix sur le serveur cible dans la zone « auto-login username »



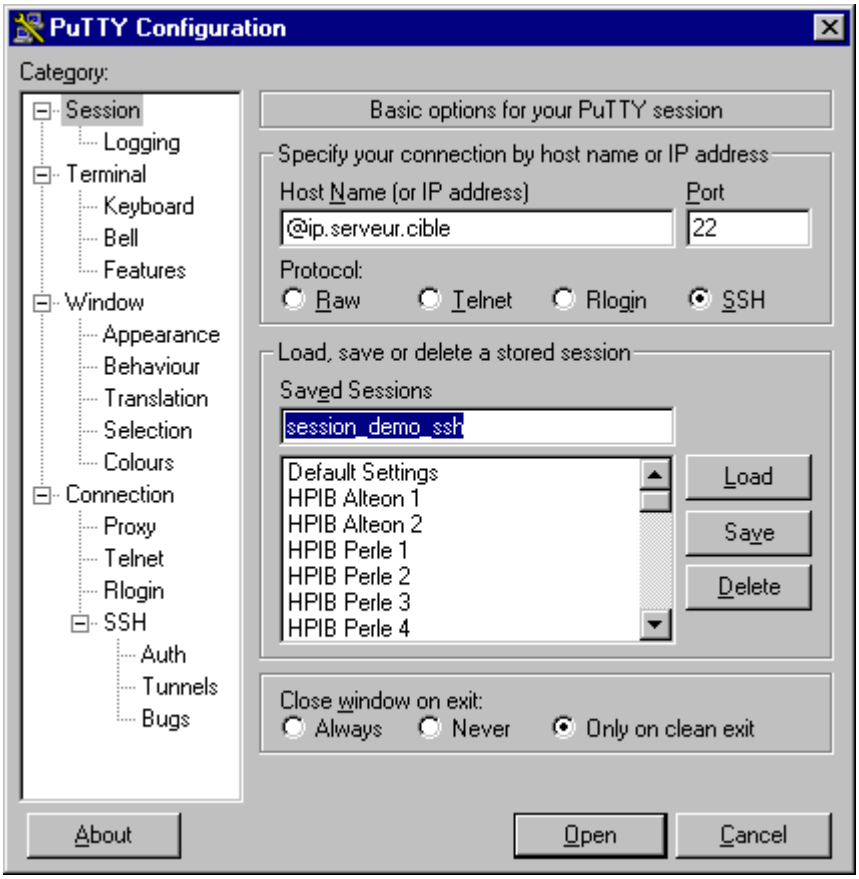
- Paramétrer la catégorie « connexion → ssh » :
  - Cocher la version 2 du protocole ssh



- Paramétrer la catégorie « connexion → ssh → auth » :
  - Cocher « allow attempted changes of username in SSH2 »
  - Positionner le fichier contenant la clé privée adéquate



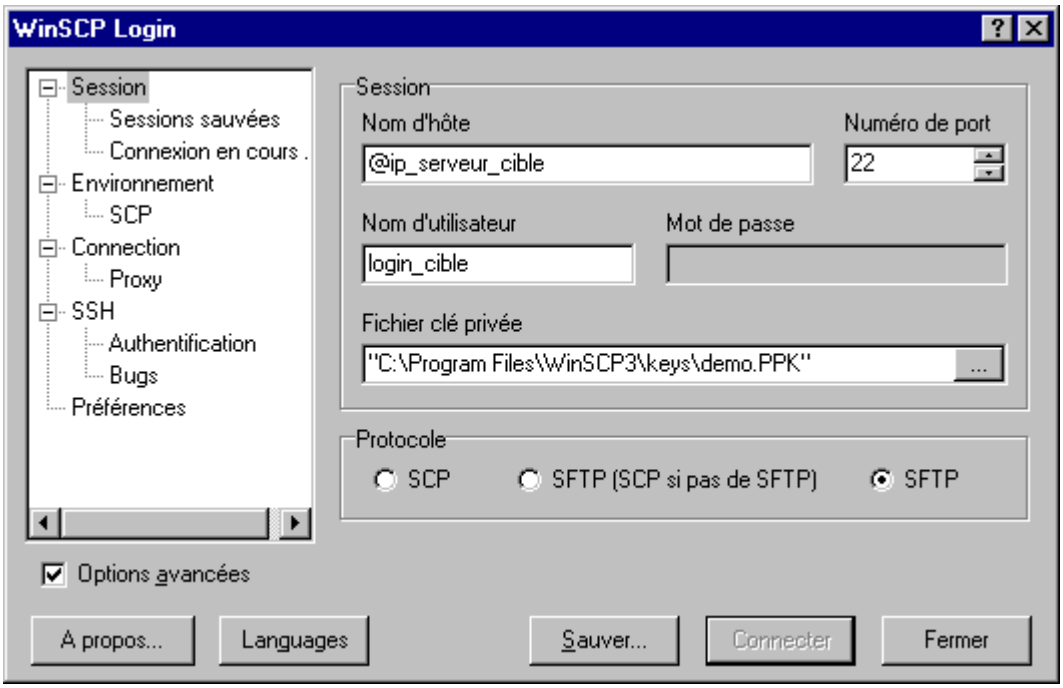
- Sauvegarder puis lancer la session :



3. Transfert en sftp via WinSCP

- Lancer WinSCP

- Paramétrer une nouvelle session :
  - Saisir l'adresse ip du serveur cible dans la zone « nom session »
  - Saisir le login cible dans la zone « nom utilisateur »
  - Positionner le fichier de clé privé dans la zone « fichier clé privée »
  - Sélectionner le protocole sftp



- Sauvegarder puis lancer la session

NB: la connexion sera possible seulement lorsque les administrateurs des serveurs cibles aurons reçu et traité votre clé publique.

4. Configuration du serveur ssh

- Installer le paquet openssh-server

  Apt-get install openssh-server

- Copier la clé publique générée via puttygen dans le fichier /home/utilisateur/.ssh/authorized_keys
- Tester la connection ssh via putty (voir chapitre suivant). S'il n'y a pas de message d'erreur, le système doit demander de saisir la passphrase.

Si la connection fonctionne:

- Editer le fichier « sudo vi /etc/ssh/sshd_config »
- Modifier *#PasswordAuthentication yes* à "*PasswordAuthentication no*" pour que les accès se fasse uniquement avec via clé ssh, les connections sur la machine ne pourront pas être effectués sans cela.
- Avoir UsePAM à no
- PermitRootLogin à no pour empêcher la connexion ssh via root
- *AllowUsers jmv* pour autoriser le seul utilisateur jmv à se connecter via ssh. Nous conseillons de déclarer ici uniquement l'administrateur du système
- Décommenter la ligne « *#MaxStartups 10 :30 :60* » pour limiter le nombre de tentative
- Redémarrer le service: « *sudo /etc/init.d/ssh restart* »

3. Serveur ftp : vsftpd

Vsftpd est un serveur ftp robuste qui offre de nombreuses fonctionnalités. Les options de configuration proposées ont été choisies afin de simplifier l'administration du serveur ftp.

- Les utilisateurs seront ceux du système linux
- Seuls ceux autorisés et déclarés dans un fichier pourront se connecter au serveur via ftp
- Les utilisateurs seront chrooter dans le répertoire ce qui signifient que leur déplacement sera restreint uniquement aux répertoires de leur compte

Il est proposé de configurer le serveur ftp pour que la connexion soit chiffrée en SSL. Cela permettra d'éviter de voir passer en clair les mots de passe des utilisateurs sur le réseau, ainsi que le contenu des fichiers.

1. Installation

Installation du paquet vsftpd

Apt-get install vsftpd

Configurons maintenant le fichier de configuration /etc/vsftpd.conf

Vérifier si les variables suivantes sont décommentées et forcez les aux valeurs suivantes :

- *anonymous_enable=NO* pour empêcher les connections anonyme (anonymous)
- *local_enable=YES* pour autoriser la connection des utilisateurs déclarer sur le système
- *write_enable=YES* pour permettre aux clients de copier des fichiers sur le serveur ftp
- *chroot_local_user=YES* pour empêcher les utilisateurs d'accéder aux répertoires qui ne sont pas dans leur répertoire utilisateur

- *userlist_deny=NO* et *userlist_enable=YES* pour clairement definer quells utilisateurs peuvent accede au service frp en les declarant dans le fichier /etc/vsftpd.user_list
- *ftpd_banner*=Bienvenue sur le serveur des Marti Pour mettre à jour le message de bienvenu sur le serveur.

Les manipulations suivantes sont optionnelles. Elles permettent de mettre en œuvre des connections chiffrées en ftp. Elles sont fortement conseillées afin de ne pas faire transiter en clair les mots de passe.

Dans le fichier on ajoutera :

```
Rajouter le fichier :
```

 # Options for SSL
 # encrypted connections.
ssl_enable=YES
allow_anon_ssl=NO
force_local_data_ssl=NO
force_local_logins_ssl=YES
ssl_tlsv1=YES
ssl_sslv2=YES
ssl_sslv3=YES
rsa_cert_file=/etc/ssl/certs/vsftpd.pem

On créera le certificat SSL par l'exécution des commandes suivantes :

```
mkdir ~/SSL-cert-vsftpd && cd ~/SSL-cert-vsftpd

openssl req -x509 -nodes -days 730 -newkey rsa:1024 -keyout vsftpd.pem -out
vsftpd.pem

cp ~/SSL-cert-vsftpd/vsftpd.pem /etc/ssl/certs/
chown root:root /etc/ssl/certs/vsftpd.pem
chmod 600 /etc/ssl/certs/vsftpd.pem
```

Pour la prise en compte de ses changements, il faut redémarrer le service :

/etc/init.d/vsftpd restart

1. Ajouter un utilisateur

Nous nous appuierons sur un exemple. Nous allons créer l'utilisateur raymond

Création du compte

useradd -d /home/raymond -m raymond

Fixer un mot de passe à l'utilisateur

Passwd raymond

Attention, nous conseillons de mettre un mot passe comportant au moins 6 caractères avec des majuscules, minuscule, caractères spéciaux numériques. Il existe des logiciels qui savent sans problèmes forcer des mots de passe dès que ceci sont simples (exemple john the ripper, cf http://korben.info/comment-cracker-un-mot-de-passe-sous-linux.html)

Exemple de mot de passe adéquat : k12#P5f

Déclarer l'utilisateur dans vsftpd dans le fichier /etc/ vsftpd.user_list

Ajouter raymond dans le fichier /etc/ vsftpd.user_list

1. Supprimer un utilisateur

Pour supprimer l'utilisateur il faut l'enlever du système

« deluser raymond »

Et le supprimer du fichier **/etc/ vsftpd.user_list**

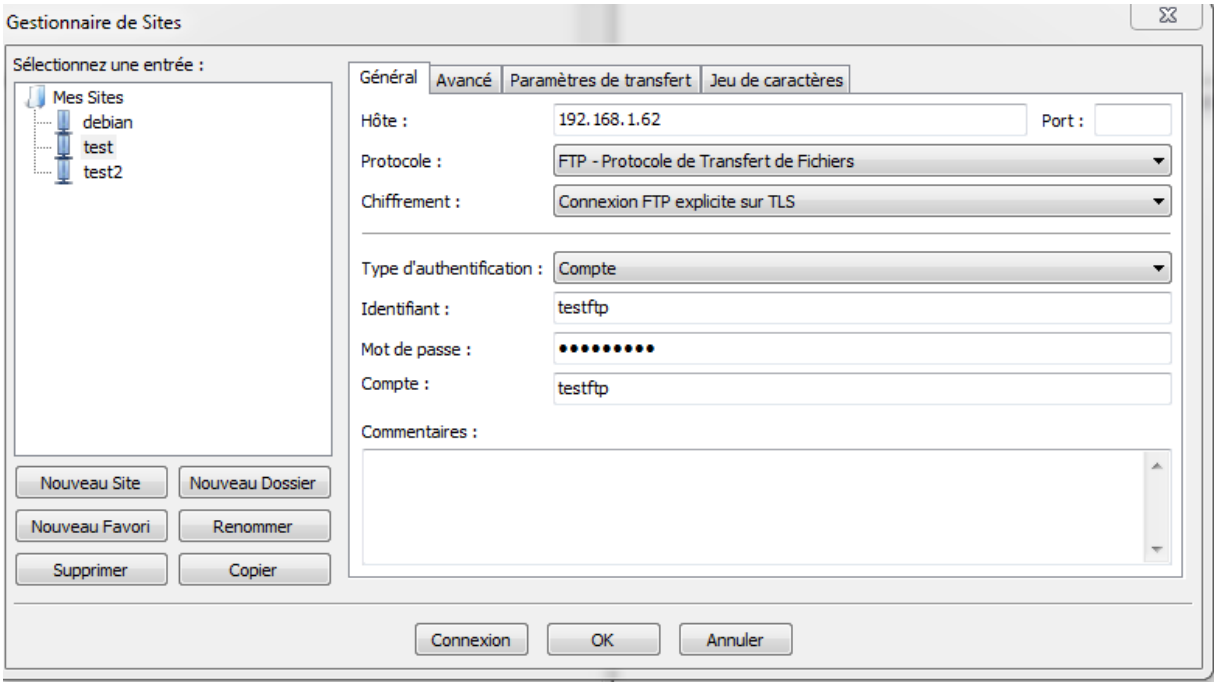
4. Quelques commandes utiles

Rebooter le serveur : *reboot*

Mettre à jour le serveur : *apt-get update && apt-get upgrade*

5. Configuration de filezilla pour le client

Exemple de configuration avec filezilla et un chiffrement ssl :



Hôte adresse ip du serveur

Pour chiffrement, choisir « Connexion FTP explicite sur TLS »

Type d'identification, il faut choisir compte

6. Annexe : Installation serveur lamp

1. Mysql :

```
aptitude install mysql-server mysql-client

Le mot de passe de l'utilisateur root est fixé au moment de l'installation
du paquet.
```

2. Apache :

```
aptitude install apache2 apache2-doc
```

3. php :

```
aptitude install php5 php5-mysql libapache2-mod-php5
```

4. phpmyadmin :

```
Installation de phpmyadmin : apt-get phpmyadmin
Puis redémarrez apache : # /etc/init.d/apache2 restart
```

7. Annexe : Installation d'un serveur webdav

1. Installation d'apache et des modules insdispensables

```
# apt-get install apache2 php5 libapache2-mod-php5
```

2. Ajout des modules davs au serveur web

```
# cd /etc/apache2/mods-enabled
# ln -s ../mods-available/dav* .
# ln -s /etc/apache2/mods-available/authn_anon.load
/etc/apache2/mods-enabled/authn_anon.load
```

3. Configuration de l'authentification et ajout d'utilisateurs à apache

```
# htpasswd -m -c /etc/apache2/.htpasswd USERNAME

Vous pouvez utiliser cette commande pour ajouter d'autres utilisateurs
ou changer leur mot de passe.
```

Remember to remove the **-c** option or you'll truncate this password database file every time!

4. Création du repertoire webdav du serveur

```
# mkdir /var/www/myWebDAV
# chown www-data:www-data /var/www/myWebDAV
```

5. Création des certificats pour le serveur

```
# mkdir /etc/apache2/ssl
# cd /etc/apache2/ssl
# /usr/lib/ssl/misc/CA.sh -newca
```

Suivez les instructions indiquées à l'écran

```
# /usr/lib/ssl/misc/CA.sh –newreq
```

Souvenez vous de renseigner le nom de votre serveur web **Common Name (eg, YOUR name) []:**, par exemple host.example.com (enter)

Ensuite signez vous-même la certifcation :

```
# /usr/lib/ssl/misc/CA.sh -sign
```

6. Tweak the certification files

```
# chmod 400 -R /etc/apache2/ssl/*
# cp newcert.pem host.example.com.pem
# openssl rsa -in /etc/apache2/ssl/newkey.pem -out
/etc/apache2/ssl/host.example.com.key
```

the last command is to remove the passphrase from the RSA private key which will stop to prompt for the passphrase when starting or restarting apache

7. Configuring Apache2 to serve HTTPS pages

```
# nano /etc/apache2/sites-available/testwebdav
```

edit the file with the following content, which not only enable the HTTPS but also setup the WebDAV folder

```
<VirtualHost _default_:443>
        Servername host.example.com
        DocumentRoot /var/www/myWebDAV
        CustomLog /var/log/apache2/ssl_access.log combined
        SSLCertificateFile    /etc/apache2/ssl/host.example.com.pem
        SSLCertificateKeyFile /etc/apache2/ssl/host.example.com.key
        <IfModule mod_ssl.c>
                SSLEngine on
                SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-
unclean-shutdown
        </IfModule>
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
```

```
        <Directory "/var/www/myWebDAV">
                Dav On
                AuthType Basic
                AuthName "USERNAME"
                AuthUserFile /etc/apache2/.htpasswd
                <Limit PUT POST DELETE PROPFIND PROPPATCH MKCOL COPY
MOVE LOCK UNLOCK>
                        Require user USERNAME
#                        Require valid-user
                </Limit>
#                Options Indexes FollowSymLinks MultiViews
                Options FollowSymLinks
                AllowOverride None
                Order allow,deny
                allow from all
        </Directory>
</VirtualHost>
```

now, enable the new site file and reload Apache2

```
# ln -s /etc/apache2/sites-available/testwebdav /etc/apache2/sites-
enabled/testwebdav
# /etc/init.d/apache2 reload
```

8. Verify the things

```
# netstat -anpt | grep apache2
```

you should find something like (the PID could be different, the port **80** and **443** do matter):

```
tcp6       0      0 :::80                   :::*
LISTEN     5395/apache2
tcp6       0      0 :::443                  :::*
LISTEN     5395/apache2
```

9. Now, it is time to configure your software to use the WebDAV service at **https://WEB_server_IP_address/**. Remember, the WebDAV setup here is for single user, which means one software can only have one version of data on the server!

Should be easy. Enjoy.