1 Introduction

- Développé à l'université d'Helsinki, SSH est un protocole réseau qui a pour but de remplacer :
  - telnet(1969), rlogin et rsh qui permettent d'établir une connexion TCP sur un hôte et transmettent les données en clair sur le réseau !
  - FTP si seules les fonctions d'ajout et de modification des fichiers sur un serveur sont nécessaires
  - Tunneling : sécurisation d'un protocole en l'intégrant dans ssh comme VNC par exemple.
- SSH offre des services encapsulés dans une connexion sécurisée basée sur une cryptographie à clé publique/clé privée. On retiendra la version 2 d'SSH.
- Le principe de cette cryptographie :
  - Tout ce qui est chiffré par la clé publique ne peut être déchiffré par la clé privée, et vice-versa.
  - Il est également impossible de dériver l'une à partir de l'autre.
  - La clé publique est destinée à être diffusée, tandis que la clé privée ne doit être connue que par le propriétaire.
  - Est utilisé dans la signature ou le chiffrage des E-Mails.

2 OpenSSH : http://doc.ubuntu-fr.org/ssh

- Vérifier le port utilisé par ssh dans /etc/services

- Vérifier que openssh-client soit installé sur le client : `dpkg --get-selections | grep openssh`

- Et sur le serveur, vérifier que openssh-server soit installé.
  Si par défaut il ne l'est pas, alors : `sudo apt-get install openssh-server`

- Pour se connecter du poste (client) sur le poste serveur, 2 choix possible :
  - Par mot de passe de l'utilisateur du poste serveur ET/OU
  - Par une passphrase avec un client authentifié par une clés publique/privée
    - Sur le poste client, créez une paire de clés privée/publique avec : ssh-keygen -t rsa . Voir le dossier caché ~/.ssh
    - Puis déposez la clé publique sur la machine cible (serveur) : ssh-copy-id
- Tester une connexion d'un client openssh sur un serveur openssh avec soit le mot de passe ou la passphrase :
  `ssh login@adrip`
  - quel est le rôle de l'option -X : ...
  - quel est le rôle de l'option -C : ...
  - quels sont les rôles des options -v, -vv et -vvv : …

- Depuis le poste client :
  - effectuer des commandes en mode console : ls, who, ifconfig
  - lancer une application graphique (ex:nautilus &) sur le serveur avec son affichage sur le client et une compression des données.

- Dans la cas où les clés sur le serveur sont regénérées, effectuer la commande suivante sur le client : `ssh-keygen -R <ip>`

- Vérifier sur le serveur ssh la connexion du client `:who`

- Se déconnecter du serveur : ...

- Copier un fichier de votre station (cliente) vers l'hôte (serveur) distants : scp ...
  - quel est le rôle de l'option -p : ...
  - quel est le rôle de l'option -r : ...
  - quel est le rôle de l'option -C : ...

- Utilisation du protocole SFTP pour un transfert de fichiers : sftp...
  - Vérifier le nouveau prompt
  - Vérifier les commandes : help, ls, cd, pwd, lls, lcd, lpwd

  - Vérifier les commandes : put, get

- Via sftp tester : nautilus sftp://log@serveurssh

- Que contient le répertoire /etc/ssh:

- Que contient le répertoire ~/.ssh:

3 SSH et les tubes

Il est possible de brancher la sortie de commandes locales sur des commandes distantes, et vice-versa. On exploitera les guillemets pour protéger les commandes.

- Exécuter la commande ls sur une machine distante et récupérer sa sortie en locale : ...

- Exécuter la commande suivante et expliquer son fonctionnement et donc son rôle :
```
tar -czf - repertoire | ssh user@host "cat >sauv.tar.gz"
```

- Exécuter la commande suivante et expliquer son fonctionnement et donc son rôle :
```
ssh user@host "cat sauv.tar.gz" | tar xzf
```

4 Tunnels SSH ou comment passer par Internet.

Supposons qu'il y ait un firewall (ne filtrant pas le port 22 de SSH, sous linux : ufw status) entre les 2 machines.
L'option -L de SSH permet de créer un tunnel permettant d'employer un port de la machine locale pour transporter des données à travers la connexion SSH et les rediriger où l'on veut à partir de la machine distante.
Le tunnel est également utilisable dans le sens inverse.

Machine_Locale(port 2500, SSH)
<-tunnel SSH->
Firewall
<-tunnel SSH->
Machine_Intermédiaire(SSH)
<-réseau local->
Serveur_POP3(port 110)

Ouvrir le tunnel sur Machine_Locale :
```
ssh -L 2500:Serveur_POP3:110   Machine_Intermédiaire
```

L'option -L demande à rediriger le port local 2500 vers le port 110 de Serveur_POP. Celui-ci est spécifié du point de vue Machine_Intermédiaire (Machine_Locale et Serveur_POP3 ne sont pas visibles entre elles).

Le client mail de Machine_Locale se connectera sur localhost:2500 ce qui équivaut à se connecter sur Serveur_POP3:110 avec une communication cryptée dans un tunell SSH.

http://www.linuxpedia.fr/doku.php/commande/ssh

http://www.tuteurs.ens.fr/internet/loin/ssh.html