Travaux Pratiques : SSH

L'objectif de ce TP est d'appréhender les différentes fonctionnalités de SSH

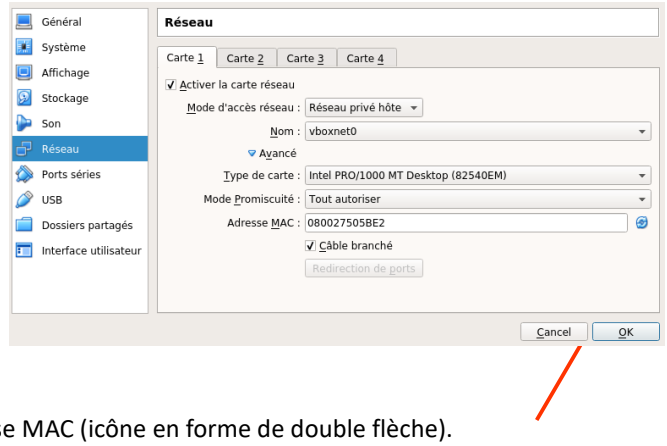
Présentation de SSH

Secure Shell (**SSH**) est à la fois un programme informatique et un protocole de communication sécurisé. Il existe deux version majeures de SSH (1.0 et 2.0), mais la seconde est la plus répandue et surtout la plus sûre (notamment d'un point de vue cryptographique). Elle possède également un protocole de transfert de fichiers : le SSH File Transfer Trotocol (**SFTP**). Le protocole utilise généralement le port TCP 22. Il est le plus souvent utilisé pour ouvrir un shell sur une machine distante (généralement de type UNIX), mais peut également réaliser des opérations de "port forwarding" en encapsulant d'autres flux au sein du tunnel sécurisé.

L'implémentation la plus fréquente est celle d'OpenSSH (mais il en existe d'autre). La configuration de la partie serveur s'effectue sous le dossier ***/etc/ssh*** sous ***Debian***. La partie cliente est appelée via la commande *ssh*.

Installation de la machine Virtuelle

Une machine virtuelle vous a été distribuée sous la forme d'un fichier .ova. Importez-la dans VirtualBox.
Editez sa configuration réseau afin d'avoir quelque-chose de similaire à ceci :



Pensez à réinitialiser l'adresse MAC (icône en forme de double flèche).
Démarrez-la et pensez à relever son adresse IP. Le login est ***root*** et le mot de passe ***toor***.

Gwenn Feunteun (**gwenn@acceis.fr**)

Connexion et authentification du serveur

1. Connectez-vous sur votre machine virtuelle depuis votre host. Vous recevez un message d'avertissement tel que ci-après. Pourquoi ?

```
The authenticity of host '192.168.0.192 (192.168.0.192)' can't be established.
ECDSA key fingerprint is SHA256:J3FXAh0vZSi1GK7sSQozO+BJj9GZUECDX6SaUZaIKsk.
Are you sure you want to continue connecting (yes/no)?
```

2. Comment vérifier que vous dialoguez avec le bon serveur (spoil : *ssh-keygen*) ?

3. Une fois l'authenticité de l'échange établie, acceptez la connexion. Vérifiez que l'empreinte du serveur a bien été ajoutée dans le fichier local *~/.ssh/known_hosts*.

4. Créez un répertoire old dans lequel vous déplacez les clé SSH actuelles.

5. Regénérez un ensemble de clés : rsa de 3072 bits, ecdsa de 512 bits.

6. Reconnectez-vous au serveur. Vous devriez avoir le message suivant. Pourquoi ?

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ECDSA key sent by the remote host is
SHA256:qrBv+fpD7sDBy5UgsW8k8OHys/8InFkmtVOlL4z8r8I.
Please contact your system administrator.
Add correct host key in /home/latliq/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /home/latliq/.ssh/known_hosts:65
ECDSA host key for 192.168.0.192 has changed and you have requested strict checking.
Host key verification failed.
```

Comment corriger le problème ?

Remplacement des mots de passe

Par défaut, SSH utilise les mots de passe du système pour authentifier les utilisateurs (quoique, pas toujours pour root). Il existe cependant une alternative : l'authentification par clé, voire par certificat.

1. Sur votre poste, générez un bi-clé RSA de 3072 bits via la commande *ssh-keygen*. La clé privée doit être protégée par une passphrase de votre choix.

    Où se situent et à quoi correspondent les fichiers générés ?
    Comment la clé privée est-elle protégée ?

2. Sur le serveur, ajoutez votre clé publique au fichier *~/.ssh/authorized_keys* (à créer s'il n'existe pas). Vous pouvez le faire à la main ou en utilisant la commande *ssh-copy-id*.

3. Connectez-vous au serveur au moyen de votre clé.

    Est-il encore possible de se connecter avec un mot de passe ?

4. Désactivez les connexions par mot de passe dans le fichier de configuration du serveur (directive *PasswordAuthentication*)

Renforcement de la configuration

Pour garantir la traçabilité des accès, il est recommandé de ne pas utiliser le compte root pour ce connecter, mais celui d'un utilisateur et d'élever ensuite ses privilèges.

1. Créez un nouvel utilisateur sur le serveur (adduser)

3. Ajoutez-le aux sudoers

4. Créez et déclarez un nouveau bi-clé pour qu'il puisse se connecter en SSH

5. Interdisez à root de se connecter (directive *PermitRootLogin*).

Transfert de fichiers

Les commandes scp et sftp permettent de transférer des fichiers par l'intermédiaire de ssh.

1. Essayez aussi de transférer un fichier entre votre machine et la machine virtuelle.

Gwenn Feunteun (**gwenn@acceis.fr**)

Port forwarding

1. Un service web écoute sur le port 80 de la machine virtuelle. Vérifiez-le en lançant la commande suivante :

$ netstat -laputen | grep apache2

Vous devriez obtenir quelque-chose comme cela :

```
tcp        0      0 127.0.0.1:80            0.0.0.0:*               LISTEN      0          17378       3138/apache2
```

2. Tentez de vous connecter au service depuis votre machine hôte. Cela marche-t-il ? Pourquoi ?

3. Etablissez une connexion SSH sur le serveur en redirigeant le port TCP 8080 de la machine hôte vers le port 80 de la machine virtuelle (option -L de la commande SSH). Connectez-vous ensuite avec votre naivgateur sur l'adresse *http://127.0.0.1:8080*. Cela marche-t-il ? Pourquoi ?

4. Recommancez, mais cette fois-ci avec le port local TCP 80. Cela marche-t-il ? Pourquoi ? Comment résoudre ce problème ?