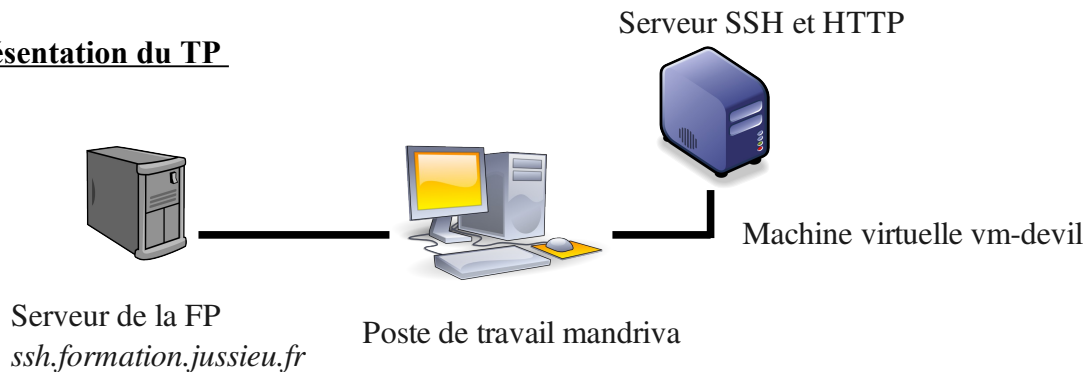


## TP SSH

### I Présentation du TP



On se propose d'utiliser pour ce TP une machine VMWARE sous DEVIL-LINUX comme serveur SSH et HTTP. C'est la même machine que le TP GnuPG.

Cette machine filtre en entrée tous les ports sauf les services SSH et FTP. Le service HTTP est par conséquent filtré, et donc inaccessible.

### I Préparation du serveur SSH

Comptes sur la machine virtuelle vm-devil

Administrateur:

login **root**

mot de passe: **tototo**

Utilisateur:

login **fpadm**

mot de passe : **azerty**

Pour travailler, il vous faudra trois terminaux :

un sur la machine virtuelle : devil

un sur le poste de travail : mandriva

un sur serveur.formation.jussieu.fr

## II Paramétrage du serveur SSH (machine virtuelle)

1) - Connectez-vous sur le serveur SSH en tant que **root**. Repérer le répertoire des fichiers de configuration de ssh. Que représentent ces fichiers de ce répertoire ?

Après chaque modification dans le fichier de configuration du serveur sshd, il sera nécessaire de redémarrer le serveur ssh.

2) - Editez le fichier de configuration du serveur sshd, et paramétrez le comme suit :

- le serveur ne doit accepter que des connexions SSH en protocole V2
- On doit bloquer les comptes sans mot de passe pour les connexions ssh
- on veut utiliser uniquement les algorithmes asymétrique RSA V2 ou DSA (on ne veut pas utiliser les algorithmes RSA1 pour l'authentification forte). On vérifiera son bon fonctionnement dans la partie IV
- on veut le transfert d'applications graphiques via ssh

Relancer le serveur ssh.

Vérifier : Que l'accès en SSH V1 est bien impossible

3) – Utiliser maintenant le compte utilisateur **fpadm**

- Bloquez l'accès au compte root pour toutes connexions ssh. Vérifiez via des connexions ssh que l'administrateur root ne peut plus se loguer en ssh sur la machine virtuelle.

- Connectez-vous maintenant via le login **fpadm**. Passez **root** via la commande su. Que se passe-t'il ? Comment autoriser l'utilisateur **fpadm** à passer **root** ?

**Pour autoriser l'utilisateur fpadm à passer root sur la devil, éditer le fichier /etc/group et ajouter l'utilisateur fpadm au groupe wheel**

**wheel:x:15:fpadm**

**L'utilisateur fpadm peut alors passer root**

## III Du côté client ! (machine Mandriva)

1 - Créer une configuration optimisée qui permet (sans créer d'alias shell !):

- de ne plus donner par défaut le login **fpadm** à chaque connexion ssh (il doit être implicite)
- de ne plus se servir de l'adresse Ip de la machine virtuelle pour les connexions mais en utilisant un nom du type : **devil**

Une connexion depuis la mandriva vers la machine virtuelle s'effectuera alors simplement avec la syntaxe : **ssh devil**

- Créer un répertoire **bin** dans le compte **fpadm** en utilisant ssh par une commande de création distante (et non via une connexion). De la même manière afficher l'ensemble des fichiers de **fpadm**.
- Créer un fichier texte, et effectuer le transfert du fichier vers la machine virtuelle
  - par scp dans le répertoire /tmp
  - par sftp dans le répertoire bin de **fpadm**

## IV – Authentification forte

On cherche maintenant à se connecter sur *devil* avec le compte *fpadm* sans utiliser le système d'authentification classique par mot de passe. On va alors se servir du système de gestion de l'authentification de ssh par clé publique/privée.

1) - Créer un couple de clé Privée/Publique en RSA2 en les appelant : *key\_ssh\_devil* et *key\_ssh\_devil.pub* sur la Mandriva. Donner lui la passphrase : «Trop facile !»  
Changer la passphrase de la clé qui vient d'être générée et donnez lui votre propre passphrase.

2) - Mettre en place la le système de connexion par clé privée/publique en fonctionnement :

- Installez la clé publique sur le serveur *devil*
- Activez le client ssh pour prendre en compte cette méthode d'authentification.  
Le mot de passe ne doit plus être demandé à partir de ce moment, seul votre *passphrase* le sera !

3) – Mise en mémoire des clés. Afin de ne pas à avoir à donner à chaque connexion la passphrase, on se propose de se servir de la faculté de mise en mémoire des clés ssh.

- Lancer l'agent ssh afin qu'il crée l'environnement de gestion des clés dans un terminal shell
- Communiquez votre clé à l'agent via le même terminal

- Lancer toujours dans la même terminal une connexion ssh vers la machine *devil*. Il n'est alors plus demandé de passphrase. Lancer depuis ce terminal une xterm, et dans cette xterm lancer une connexion ssh vers *devil*. Que concluez-vous ?

4) – Mise en mémoire des clés au démarrage de la session KDE

Mettre cette méthode en place à l'ouverture d'une session graphique (KDE), et vérifier que quels que soient les shells utilisés, la connexion s'effectue alors systématiquement sans passphrase

- Créer le script ssh-add.sh nécessaire et placez-le dans le répertoire ~/.kde/Autostart, puis positionnez les droits en exécution au fichier.

```
#!/bin/bash
```

```
if [ -x /usr/bin/ssh-add ]; then
```

```
    /usr/bin/ssh-add $HOME/.ssh/key_ssh_devil
```

```
fi
```

- Lancer la commande keychain
- Quittez alors votre session KDE et reconnectez vous sur votre poste de travail
- Vérifier que la passphrase (via une boîte modale) est bien demandée au début de la session

## V Connexion par tunnel

1 – Comment ferriez-vous simplement pour vous connectez depuis le serveur de la formation permanente vers la machine virtuelle devil en une seule commande ssh ?

2 – Accédez au serveur Web de la devil à partir du poste de travail Mandriva:

- Vérifier l'accès au serveur Web de la devil avec un navigateur, il ne doit pas répondre.
- Mettre en place un tunnel SSH entre le poste de travail et le serveur devil
- Connectez-vous maintenant sur l'intranet

3 – On souhaite maintenant accéder au serveur web à partir du serveur de la formation permanente

(pour les test de requêtes web on utilisera wget).

Créer un double tunnel chiffré entre le serveur *serveur.formation.jussieu.fr* et la machine virtuelle *devil* (en passant par la Mandriva) en une seule commande ssh afin d'avoir accès au serveur web intranet de la devil depuis le serveur de la FP.

## **VI Forçage de commande**

Dans certain cas (lancement de tâches automatiques), il est nécessaire de créer des clés privé/public sans passphrase associées, ce qui rend le serveur très vulnérable. Pour renforcer la sécurité dans ce cas de figure, on peut forcer la commande utilisable par cette connexion (et seulement cette commande peut être exécutée) et l'on peut aussi l'adresse Ip source qui peut utiliser cette commande.

Mettre en place ce système :

- On se propose d'autoriser uniquement la commande *ls* par la connexion ssh entre le poste de travail et le serveur *devil*
- Créer une nouvelle clé (DSA) sans passphrase depuis le poste de travail et d'installer cette clé sur le serveur devil
- Configurer le forçage de la commande *ls* par la connexion ssh
- renforcer la sécurité en fixant *l'adresse IP source* autorisée (IP du poste de travail)