TP: SSH

Pierre Blondeau
pierre.blondeau@unicaen.fr

17/09/2012

1 Introduction

L'objectif de ce TP est de vous faire manipuler ssh et toutes les petites fonctionnalités qui font que ce logiciel est indispensable pour le tavail à distance.

ATTENTION 1 : Au cours de ce tp lorsque vous rencontrerez le caractère "\" dans un fichier de configuration ou une ligne de commande, il s'agit uniquement d'un symbole signifiant que la "phrase" est trop longue pour être mise sur une seul ligne dans ce sujet.

ATTENTION 2 : Après chaque modification des fichiers de configuration, il faudra redémarrer ou recharger la configuration des services modifiés. Par exemple pour ssh `/etc/init.d/ssh restart`.

2 VirtualBox

2.1 Lancement des machines virtuelles

Pour ce TP, nous allons utiliser VirtualBox. Pour exécuter le programme VirtualBox, aller dans le menu gnome :

```
 - Applications => Outils système => Oracle VM VirtualBox ou VirtualBox dans la console

Activation d'une carte virtuelle entre la première machine virtuelle Passerelle et la machine physique
 - Fichier
   - Préférences
     - Réseau
       - Ajouter une carte « vboxnet0 »
       - Cliquer sur « OK »

Pour créer une première machine virtuelle :
 - Faire « Nouvelle »
 - Cliquer « Suivant »
 - Remplir le champ « Nom » (Mettre Passerelle par exemple)
 - Type de l'OS
   - Système d'exploitation : Linux
   - Version : Debian
 - Cliquer « Suivant »
 - Quantité de mémoire vive : 512M
 - Cliquer « Suivant »
 - Disque virtuel
   - Sélectionner « Utiliser un disque dur existant »
   - Cliquer sur l'icône de dossier
 - Dans la nouvelle fenêtre « Gestionnaire de médias virtuels »
   - Cliquer sur « Ajouter »
   - Dans la boîte de sélection de fichier, ouvrir le fichier
     /export/VirtualBox/Debian_ssh_1.vdi
   - Sélectionner la ligne « Debian_ssh_1.vdi 8,00Gio ... » dans la liste
   - Cliquer sur « choisir »
 - Cliquer sur « suivant »
 - Cliquer sur Terminer
```

```
 - Clique DROIT sur la machine -> « Configuration »
  - Cliquer sur « Réseau »
   - Cliquer sur « Carte 1 »
    - Mode d'accès réseau : Réseau privé hôte
    - Nom : vboxnet0
   - Cliquer sur « Carte 2 »
    - Activer la carte réseau
    - Mode d'accès réseau : Réseau interne
    - Nom : intnet
  - Cliquer sur « OK »
Démarrez la machine

Pour créer une deuxième machine virtuelle :
 - Faire « Nouvelle »
 - Cliquer « Suivant »
 - Remplir le champ « Nom » (Mettre Web par exemple)
 - Type de l'OS
   - Système d'exploitation : Linux
   - Version : Debian
 - Cliquer « Suivant »
 - Quantité de mémoire vive : 512M
 - Cliquer « Suivant »
 - Disque virtuel
   - Sélectionner « Utiliser un disque dur existant »
   - Cliquer sur l'icône de dossier
 - Dans la nouvelle fenêtre « Gestionnaire de médias virtuels »
   - Cliquer sur « Ajouter »
   - Dans la boîte de sélection de fichier, ouvrir le fichier
     /export/VirtualBox/Debian_ssh_2.vdi
   - Sélectionner la ligne « Debian_ssh_2.vdi 8,00Gio ... » dans la liste
   - Cliquer sur « choisir »
 - Cliquer sur « suivant »
 - Cliquer sur Terminer
 - Clique DROIT sur la machine -> « Configuration »
  - Cliquer sur « Réseau »
   - Cliquer sur « Carte 1 »
    - Mode d'accès réseau : Réseau interne
    - Nom : intnet
  - Cliquer sur « OK »
Démarrez la machine
```

Vous pouvez maintenant diminuer toutes les fenètres virtualbox ou changer de bureau. Vous n'en aurez plus besoin grâce à SSH !

2.2 Schéma

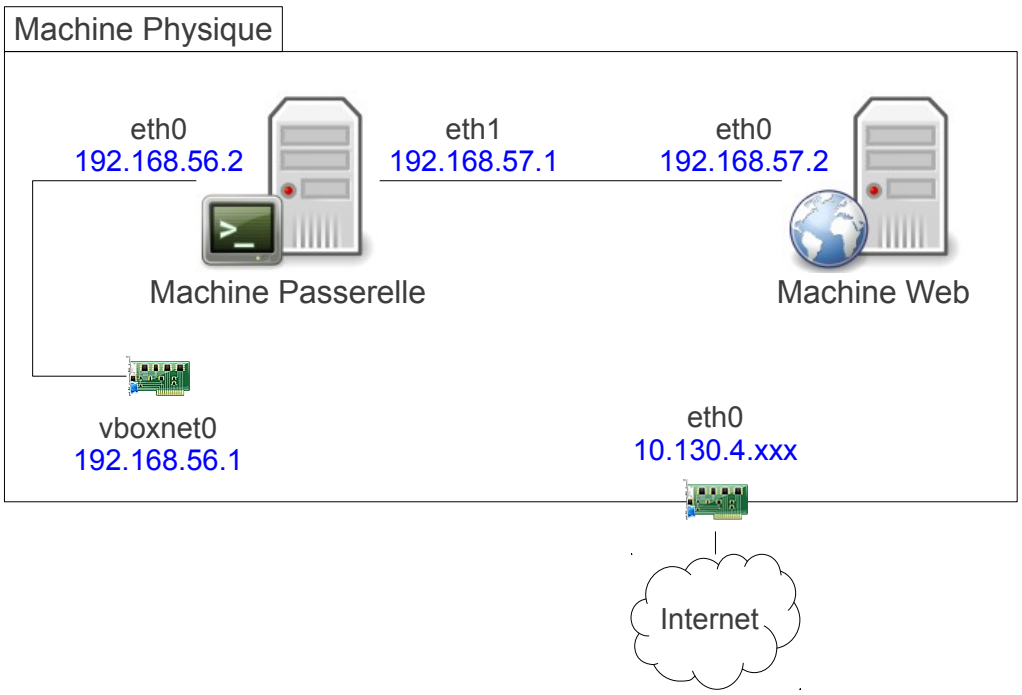

FIGURE 1 – Schema

Comme nous le montre la Figure 1, nous venons de créer avec VirtualBox cette architecture dans la marchine physique.

3 SSH

3.1 Installation et Configuration

Pour gagner du temps, les serveurs ssh sont déja installés et configurés sur les machines. Si vous avez besoin d'effectuer cette opération sur une autre machine :

```
# apt-get install ssh
# vi /etc/ssh/sshd_config
```

3.2 Test de connexion

Vous allez maintenant vous connecter en ssh sur la machine Passerelle en tant que root (mdp : root), tapez dans une console :

```
# ssh root@192.168.56.2
```

Le client ssh vous pose alors une question du genre :

```
The authenticity of host 'mike (2001:660:7101:1::9)' can't be established.
RSA key fingerprint is 7d:8a:f3:fc:f9:2d:db:77:18:e5:2e:cf:0b:1b:7d:b3.
Are you sure you want to continue connecting (yes/no)?
```

Lors de l'établissement de la connexion, des échanges de clés sont effectués pour la sécuriser. Votre client ssh demande donc si l'empreinte de la clé de la machine X (ici mike) est bien valide.

Si vous tapez yes, cette empreinte sera stockée dans le fichier *~/.ssh/known_hosts*, sinon la connexion sera refusée.

Cette procédure vous permet de vérifier que vous vous connectez sur le bon serveur et que son adresse ip n'a pas été usurpée.

Vous pouvez affichier le fichier *∼/.ssh/known_hosts*. Pourquoi est-il incompréhensible ?

ATTENTION : si vous réinstallé votre serveur ou si il y a un problème avec l'adresse ip, un message dans le genre du suivant s'affichera :

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    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 the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
7d:8a:f3:fc:f9:2d:db:77:18:e5:2e:cf:0b:1b:7d:b3.
Please contact your system administrator.
Add correct host key in /home/pierre/.ssh/known_hosts to get rid of this message.
Offending key in /home/pierre/.ssh/known_hosts:837
RSA host key for mike has changed and you have requested strict checking.
Host key verification failed.
```

Si la machine a été réinstallée, vous devez supprimer les anciennes clés de votre fichier *∼/.ssh/known_hosts*. Pour cela, il y a deux solutions :
– Supprimer les références de la machine avec les outils de ssh "ssh-keygen -R {machine}"
– Supprimer la ligne "Offending key" (ici 837) dans votre fichier *∼/.ssh/known_hosts*

3.3 Génération et copie de clés ssh

Comme vous pouvez le constater avec la Figure 1, vous ne pouvez pas vous connecter directement sur la machine WEB depuis la machine physique. Donc une fois connecté sur la machine Passerelle, connectez vous sur la machine WEB en tans que root (mdp : root) :

```
# ssh root@192.168.57.2
```

Il n'est pas très sécurisé d'utiliser des mots de passe pour se connecter. Il sont plus faciles à trouver qu'une suite de 2048 bits aléatoires lors d'attaque par brute force. C'est pourquoi, nous allons utiliser les clés ssh.

Déconnectez vous de la machine WEB, mais restez sur la machine Passerelle. Générez une paire de clé ssh :

```
# ssh-keygen -t rsa
```

Laissez le chemin de fichier par défaut et indiquez une passphrase que seul vous connaissez.

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/pierre/.ssh/id_rsa): -> Entrée
Enter passphrase (empty for no passphrase):  -> Votre PassPhrase
Enter same passphrase again:  -> Votre PassPhrase
Your identification has been saved in /tmp/id_rsa.
Your public key has been saved in /tmp/id_rsa.pub.
The key fingerprint is:
df:92:f0:27:7d:ca:64:55:7e:ec:26:36:cd:bb:4f:7f pierre@mike
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|                 |
|                .|
|               + |
|        S     . +|
|         + + . +.|
|          * * = *|
|           B + =E|
|            o  o*|
+-----------------+
```

Vous venez de générer une paire de clé ssh RSA. Vous avez maintenant une clé privée chiffrée par votre passphrase dans le fichier *~/.ssh/id_rsa* et une clé publique dans le fichier *~/.ssh/id_rsa.pub*.

C'est cette clé publique que vous allez copier sur le serveur WEB :

```
# ssh-copy-id root@192.168.57.2
```

Connectez vous maintenant sur le serveur WEB :

```
# ssh root@192.168.57.2
Enter passphrase for key '/root/.ssh/id_rsa':
```

Insérez maintenant votre passphrase.

Remarque : si vous n'insérez pas votre passphrase, le client ssh va vous demander le mot de passe de root.

3.4 Petite sécurisation du serveur ssh

Comme il n'est pas très sécurisé de se connecter avec un mot de passe, nous allons interdire les connexions de root par mot de passe sur le serveur WEB. Pour cela, vous allez modifier le fichier */etc/ssh/sshd_config* de la machine Web et plus particulièrement la directive *PermitRootLogin*. Le manuel de sshd_config devrait vous aider (man sshd_config).

N'oubliez pas de redémarrer le service ssh.

Testez la connexion sans clé ssh et avec clé ssh.

3.5 Agent ssh

Si vous tentez de vous connecter plusieurs fois entre la machine Passerelle et la machines WEB, vous constater que ssh vous demande à chaque fois votre passphrase. C'est très pénible. Il existe des "agents ssh" qui permettent de stocker votre clé ssh privée dans la RAM. Ainsi vous pouvez vous reconnecter sur les machines sans avoir à taper à chaque fois votre passphrase.
Pour cela, depuis la machine Passerelle, il faut démarrer un agent ssh et éxécuter les informations qu'il vous renvoie :

```
# ssh-agent
SSH_AUTH_SOCK=/tmp/ssh-jGoIb18346/agent.18346; export SSH_AUTH_SOCK;
SSH_AGENT_PID=18347; export SSH_AGENT_PID;
echo Agent pid 18347;
# SSH_AUTH_SOCK=/tmp/ssh-jGoIb18346/agent.18346; export SSH_AUTH_SOCK;
# SSH_AGENT_PID=18347; export SSH_AGENT_PID;
# echo Agent pid 18347;
Agent pid 18347
```

Il est également possible de faire :

```
# eval `ssh-agent`
```

Ensuite, il faut ajouter votre clé privée à l'agent ssh :

```
# ssh-add
Enter passphrase for /root/.ssh/id_rsa:
Identity added: /root/.ssh/id_rsa (/root/.ssh/id_rsa)
```

Retestez plusieurs connexions vers le serveur WEB, il ne devrait plus vous demander votre passphrase.

3.6 Transfert X11

Votre deuxième machine s'appel WEB car il y a un serveur http apache qui tourne. Sur la machine Passerelle tapez (utilisez la touche q pour quitter) :

```
# lynx http://192.168.57.2
```

Vous avez alors un navigateur ligne de commande qui s'ouvre et qui vous permet de visualiser de contenu de la page, mais ce n'est pas très pratique pour visualiser des sites plus évolués avec du javascript ou des animations par exemple.

Déconnectez vous de la machine Passerelle et reconnectez vous avec :

```
# ssh -X root@192.168.56.2
```

Et tapez :

```
# iceweasel http://192.168.57.2
```

Un firefox se lance alors avec la même page que lynx.

Comme vous ne pouvez pas accéder directement au serveur WEB, vous venez de lancer un firefox sur la machine Passerelle. Le client ssh effectue alors un transfert des instruction X11 (environement graphique) vers votre machine locale pour vous permettre d'afficher iceweasel.

3.7 Port Forwarding

Le transfert X11 fonctionne bien sur des connexions très haut-débit (réseau local), mais à travers internet, c'est souvent inutilisable. Il éxiste alors une autre solution qui permet de transférer un port local vers une machine distant à travers une autre machine passerelle :

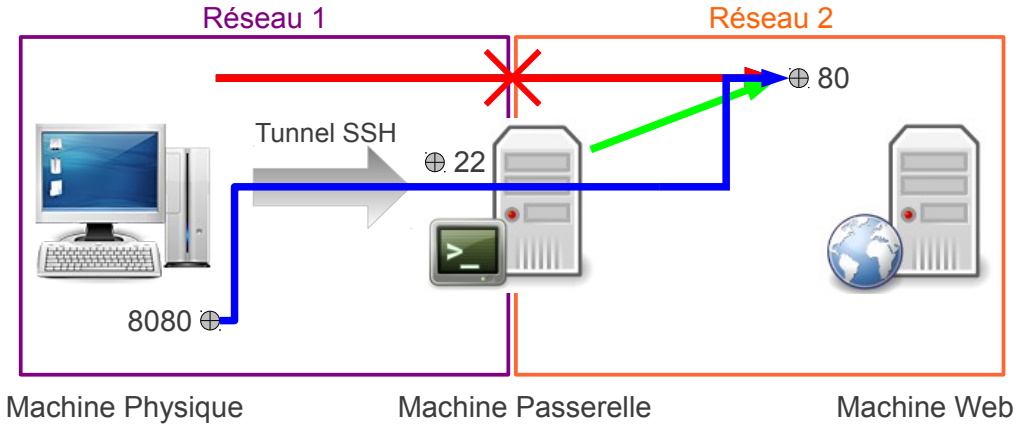

FIGURE 2 – Port Forwarding

Fermez iceweasel et votre connexion vers la machine Passerelle, puis reconnectez vous avec :

```
# ssh -L 8080:192.168.57.2:80 root@192.168.56.2
```

Lancez un firefox sur la machine Physique et allez sur l'url : http ://localhost :8080
La même page devrait alors s'afficher. Grâce au port forwarding de ssh, les paquets envoyés sur le port 8080 de votre machine Physique sont transférés à la machine WEB à travers la machine Passerelle.

La syntaxe de la commande ssh est :

```
# ssh -L {port local}:{machine destination}:{port destination} {user}@{machine passerelle}
```

Dans la plupart des cas, cette commande est lancée avec un utilisateur non privilégié (!= root). Dans ce cas, le port ouvert en local doit être supérieur à 1024.

3.8 Port Forwarding Bis et Déport d'agent ssh

3.8.1 Port Forwarding Bis

Il est également possible de faire du port forwarding dans l'autre sens, c'est à dire que vous ouvrez un port sur la machine distante qui va permetre à celle-ci d'atteindre un service local à travers votre machine passerelle.

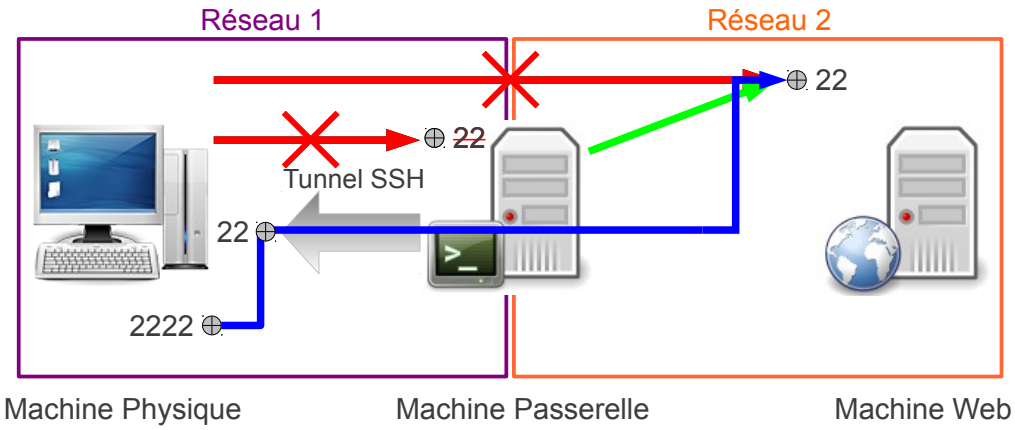

FIGURE 3 – Port Forwarding Bis

Depuis la machine passerelle, lancez cette commande en remplacant user par votre nom de login :

```
# ssh -R 2222:192.168.57.2:22 {user}@192.168.56.1
```

Une fois la connexion établie, vous êtes connecté sur la machine Physique. Vous pouvez maintenant vous connecter directement sur la machine Web. Testez la connexion vers Web avec la commande :

```
# ssh root@localhost -p 2222
```

Pourquoi le mot de passe de root ne fonctionne pas ?
La syntaxe de la commande ssh est :

```
# ssh -R {port machine distante}:{machine destination}:{port destination} {user}@{machine distante}
```

3.8.2 Déport d'agent ssh

Comme vous ne pouvez pas vous connecter sur la machine Web avec un mot de passe, il faut utiliser une clé ssh.

Une des options de ssh permet de "déporter" un agent ssh de votre machine sur une machine distante lors de la connexion. Comme l'agent ssh possède la clé privée dans la RAM, ssh transmet cette clé vers un agent sur la machine où l'on se connecte. Ainsi, on peut utiliser ce nouvel agent pour se connecter sur une autre machine.

Ainsi déconnectz vous de la machine Physique. Lancez un nouvel agent ssh et reconnectez vous avec :

```
# eval `ssh-agent`
# ssh-add
Enter passphrase for /root/.ssh/id_rsa:
Identity added: /root/.ssh/id_rsa (/root/.ssh/id_rsa)
# ssh -A -R 2222:192.168.57.2:22 {user}@192.168.56.1
```

Grâce au déport de l'agent ssh, il est maintenant possible de se connecter directement de la machine phyique vers la machine Web :

```
# ssh root@localhost -p 2222
```

3.9 Vol d'agent ssh

Les agents ssh communiquent avec les clients par le biais de socket unix qui appartient à l'utilisateur qui a lancé l'agent. Si vous pouvez accéder à cette socket, vous pouvez utiliser l'agent ssh que cet utilisateur utilise.

Pour voler un agent ssh, il suffit d'exporter la variable "SSH_AUTH_SOCK". Trouvez comment faire en analysant ce que vous demande de faire un agent ssh lors de son lancement.

Dans une nouvelle console, essayez de vous volez l'agent ssh que vous venez de déporter sur la machine Physique. Quand vous l'aurez volé, essayez de vous connecter sur la machine WEB.

4 Information

4.1 Gnome-Keyring

Sur les nouvelles versions de Gnome, il existe se qu'on appelle un gnome-keyring. Celui-ci permet d'enregistrer des mots de passe et des passphrases dans un conteneur chiffré qui est déverrouillé par votre mot de passe de session. Ce gnome-keyring possède également un agent ssh.

Ce qui signifie que si vous générez une clé ssh, que vous enregistrez la passphrase dans votre gnome-keyring. Au prochain démarrage de votre session, elle va automatiquemenet lancer un agent ssh avec votre clé privée. Vous pouvez alors directement vous connecter sur les machines sans avoir besoin de taper votre passphrase.

4.2 Autres petits éléments de sécurité

4.2.1 AllowUsers

Dans la configuration du serveur ssh, il est possible de restreindre l'accès d'un utilisateur uniquement depuis certaines adresses ip avec l'option AllowUsers , par exemple :

```
AllowUsers root@10.130.0.1
AllowUsers root@2001:660:7101:1::1
```

4.2.2 *~/.ssh/authorized_keys*

Lorsque vous copiez votre clé publique sur une nouvelle machine, elle est enregistrée dans le fichier *~/.ssh/authorized_keys*. Il est possible de restreindre les accès d'un utilisateur dans ce fichier. Il est par exemple possible d'interdire les fonctions ssh que nous venons de voir avec les options :

```
no-port-forwarding,no-agent-forwarding,no-X11-forwarding
```

On peut restreindre les adresses ip qui peuvent utiliser cette clé :

```
from="10.130.0.1"
```

Quelle est la différence par rapport à AllowUsers ?

On peut également limiter les commandes que ssh peut éxécuter. Une des application est par exemple de pouvoir utiliser svn au desous de ssh :

```
command="/usr/bin/svnserve -t -r /home/svn/depot/ --tunnel-user=pierre"
```

Exemple de commande svn :

```
# svn checkout svn+ssh://{user}@{machine}/{chemin du depot} {répertoire de destination}
```

4.3 Outils de port forwarding

Il éxiste un outil graphique pour faire pour port forwarding qui s'appelle GSTM (Voir la démonstration).

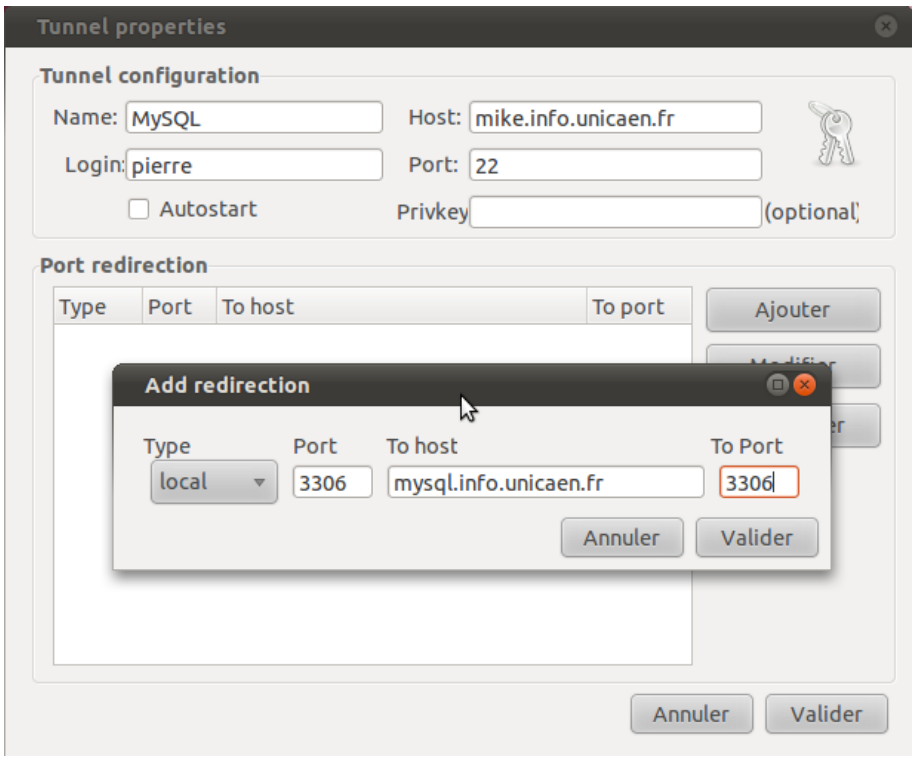

FIGURE 4 – GSTM