Comment se connecter aux machines des salles de TP par SSH avec X forwarding sous GNU/Linux

Introduction

`SSH` ("Secure SHell") permet d'obtenir un shell sur une machine à distance de façon sécurisée, voir https://fr.wikipedia.org/wiki/Secure_Shell

Afin de pouvoir utiliser les applications graphiques comme `marionnet`, on va téléporter l'affichage graphique de la machine distante vers la machine locale ("X forwarding").

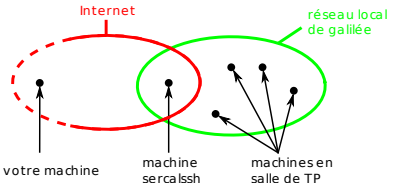
Les machines de TP ne sont pas accessibles depuis l'extérieur mais le sont depuis le réseau local (pour des raisons de sécurité et parce qu'elles n'ont pas d'adresse IP publique). Seule la machine `sercalssh` est accessible depuis l'extérieur. On va donc rebondir dessus, ou plutôt passer à travers elle.

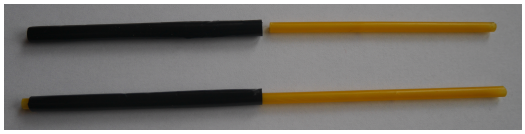
La méthode naïve consisterait à obtenir un shell sur `sercalssh`, et depuis `sercalssh` se connecter (avec la commande `ssh` disponible sur `sercalssh`) sur une machine des salles TP.

C'est tout à fait possible, mais ça n'est pas idéal :

- vous ne pourrez pas téléporter l'affichage graphique entre la machine des salles TP et la machine locale puisqu'aucune connexion directe n'est établie entre les deux.
- Lorsque vous vous connectez à la machine des salles TP, vous le faites depuis `sercalssh`, en particulier lors de la connexion SSH entre `sercalssh` et cette machine, vous devrez taper votre mot de passe depuis un shell sur `sercalssh`. Imaginez que cette machine soit compromise.
- De façon générale, gardez en mémoire qu'un bon chiffrement doit se faire d'extrémité à extrémité.

Ainsi, au lieu d'enchaîner une connexion de la machine locale à `sercalssh` puis de `sercalssh` à une machine de salle TP, on va établir un tunnel entre la machine locale et `sercalssh`, puis on va faire passer une connexion SSH entre la machine locale et une machine de salle TP à travers ce tunnel. Il y aura donc deux connexions partant de la machine locale, la seconde encapsulée dans la première.





[exercice : faites un dessin qui prenne en compte les deux images]

Configuration du client SSH

Si vous n'avez jamais utilisé `ssh`, vous devez créer un répertoire `.ssh` dans votre `HOME` de sorte à ce que seul votre user puisse y accéder :

```
$ mkdir ~/.ssh
$ chmod 700 ~/.ssh
```

Ensuite éditez le fichier `~/.ssh/config` et ajoutez-y les lignes suivantes, en remplaçant `<USERNAME>` (deux fois) par votre identifiant sur les machines distantes, en l'occurrence votre numéro étudiant :

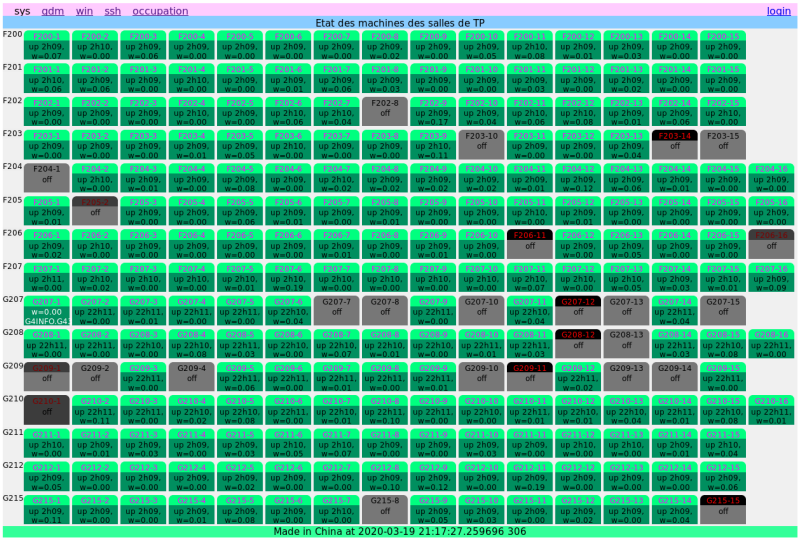
```
Host sercalssh
   Hostname sercalssh.ig-edu.univ-paris13.fr
   User <USERNAME>

Host f2* g2* F2* G2*
   ProxyJump sercalssh
   ForwardX11 yes
   User <USERNAME>
```

Selon votre distribution, vous devrez aussi rendre ce fichier inaccessible à autrui :

```
$ chmod 600 ~/.ssh/config
```

Sélection d'une machine en salle de TP

L'état des machines des salles de TP est disponible à l'adresse : https://si-galilee.univ-paris13.fr/salles/sys



Afin de répartir la charge, tirez une salle au hasard parmi les 15 et une machine au hasard parmi les 15 ou 16 de la salle. Si la machine est éteinte, retirez en une autre au hasard.

Dans la suite, on suppose que vous avez tiré la machine `F209-11`. Cette machine n'existe pas, c'est pour l'exemple et pour éviter que tout le monde se retrouve sur la même machine.

Connexion SSH

Pour vous connecter à la machine `F209-11` à travers le tunnel entre chez vous et `sercalssh`, tapez simplement :

```
$ ssh F209-11
```

Cette commande va d'abord établir le tunnel entre la machine locale et `sercalssh`, et pour cela vous devrez fournir votre mot de passe au serveur SSH de `sercalssh`. Ensuite, elle va connecter la machine locale à la machine `F209-11`, et pour cela vous devrez fournir votre mot de passe au serveur SSH de `F209-11`.

Lors de la première connexion, un message vous alerte que la clef publique RSA du serveur SSH `sercalssh` est inconnue et vous demande si vous voulez continuer. Vérifiez que le fingerprint de cette clef est : `SHA256:dN1Wp0xEsZi6joi7KffCwt/VTmxnt/p2iakDe6Qj4IM` et tapez `yes`.

Lors de la première connexion à une machine de TP, un message vous alerte que la clef publique ECDSA (ou ED25519 ou RSA selon le choix que fait votre client SSH) du serveur SSH de la machine est inconnue et vous demande si vous voulez continuer. Vérifiez que le fingerprint est

- `SHA256:9MtY4yZiAsWS6GJReDO3lFBW3lWIcRwJZG2lzEv2kag` (pour la clef ECDSA)
- `SHA256:56WJXRCh/SClrwDQG4BL36rMT1U3f1eyNvjOXzgF7YA` (pour la clef ED25519)
- `SHA256:2QqQbLRLnR0ViXRC2gXwk+tKSX2RB6kuZjVnpOf7fxs` (pour la clef RSA)

et tapez `yes` si tout va bien.

Fabrication et utilisation de clefs SSH

Afin d'éviter de devoir taper son mot de passe à chaque connexion (et même deux fois par connexion puisqu'il faut d'abord s'authentifier sur `sercalssh` puis sur la machine de TP), nous allons créer une paire de clefs SSH (une clef "publique" et une clef "privée" ou "secrète") sur la machine locale. La clef publique sera envoyée sur les serveurs SSH, et lorsque vous voudrez vous y connecter, le serveur SSH utilisera votre clef publique pour verifier l'authenticité de votre machine en lui proposant un challenge que seules les machines possédant votre clef secrète peuvent résoudre.

La clef secrète est secrète et ne doit être donnée à personne, ni aux serveurs SSH des machines des salles de TP, ni à l'enseignant·e, ni à qui que ce soit, car elle sert à authentifier votre machine lors de vos prochaines connexions SSH. Si vous avez plusieurs machines personnelles, ne recopiez pas la clef secrète d'une machine à l'autre, générez plutôt une paire de clef par machine cliente.

Pour générer votre paire de clefs :

```
$ ssh-keygen
```

Le programme de génération de clefs vous propose de choisir le fichier dans lequel sauvegarder la clef, appuyez sur `[ENTER]` pour garder le choix par défaut (qui correspond en général à `~/.ssh/id_rsa`).

Puis il vous demande un mot de passe. Ce mot de passe sert à protéger votre clef secrète des fois qu'on vous vole votre ordinateur. Il n'a rien à voir avec le mot de passe pour vous authentifier sur les machines des salles TP, ni avec un quelconque autre mot de passe. Si vous entrez un mot de passe, il vous sera demandé lors de vos prochaines connexions SSH. Vous pouvez aussi appuyer sur `[ENTER]` sans donner de mot de passe (à vos risques et périls).

La commande `ssh-keygen` a crée une paire de clefs : une clef secrète `~/.ssh/id_rsa` et une clef publique `~/.ssh/id_rsa.pub`. Vous pouvez vérifier leur présence en tapant :

```
$ ls -l ~/.ssh/
```

Le serveur SSH des machines de salles TP ont juste besoin de connaître votre clef publique pour vérifier l'authenticité de votre machine. Pour envoyer votre clef publique au serveurs SSH :

```
$ ssh-copy-id sercalssh
```

Lors de votre prochaine connexion, vous pourrez vérifier que le contenu de votre clef publique a été copié dans le fichier `~/.ssh/authorized_keys` de la machine distante. Il n'est pas nécessaire de répéter l'opération sur les machines des salles de TP puisque c'est le même `HOME` qui est monté via NFS sur toutes les machines.