Reverse ssh tunneling

Author : pierrehirel

Une connexion ssh est généralement utilisée depuis un client pour se connecter à un serveur. Dans ce schéma, le port ssh (par défaut 22) du serveur doit être accessible (i.e. ouvert et correctement routé) -si le serveur est situé derrière un pare-feu non configuré alors il n'est pas possible d'y accéder.

1. Quel est l'intérêt d'un tunnel ssh inversé ?

Un tunnel ssh inversé permet de se connecter (en ssh) à une machine située derrière un pare-feu ou un routeur. Et ce, sans avoir à les configurer ni à les hacker.

Mise en situation : vous êtes chez vous, et vous voulez aider votre tante, Mme Michu, qui a des problèmes sur son PC. Mais vous la connaissez Mme Michu, si vous commencez à lui parler de router les ports de sa box, elle va péter un câble. Et puis configurer un routeur (et donc ouvrir des ports, rebooter le routeur…) pour se connecter une seule fois sur une machine n'est pas idéal. En revanche mettre en place un tunnel de chez elle vers *votre* serveur ne requiert aucune configuration de routeur, et pourrait être beaucoup plus simple.

Autre exemple : vous êtes chez vous et vous voulez vous connecter à votre boulot. Seulement les PC du boulot sont derrière un pare-feu qui empêche toute connexion depuis l'extérieur. Qu'à cela ne tienne : vous pouvez mettre en place un tunnel inversé depuis le boulot vers votre serveur à la maison. Cela vous permettra ensuite de vous connecter au boulot depuis chez vous.

Mise en garde sur ce dernier exemple : ce genre de pratique est souvent interdite dans les entreprises, vérifiez donc votre charte informatique avant de risquer votre job en violant la sécurité de l'entreprise…

2. Mise en place du tunnel inversé

Je ne reviendrai pas sur ce qu'est un tunnel ssh, l'Internet regorge de définitions à ce sujet. J'avais d'ailleurs illustré dans un précédent billet comment utiliser un tel tunnel pour faire transiter une connexion VNC. Nous attaquerons donc ici directement le cœur du sujet.

Supposons que nous disposions des machines suivantes :

- **un serveur** (celui de chez vous dont vous avez le contrôle) dont le port ssh est bien accessible ; par défaut il s'agit du port 22, mais pour corser les choses supposons que ce soit le port 2222.
- **un client** (le PC de Mme Michu ou du boulot), ne bénéficiant d'aucune redirection de port, situé derrière un pare-feu. La seule chose requise est d'avoir un serveur ssh installé sur ce client (dans le pire des cas où vous n'avez même pas les droits d'admin sur la machine vous pouvez simplement en télécharger un ne requérant pas d'installation).

Le tunnel se met en place côté client. Puisqu'on suppose qu'aucun port n'est routé vers le client et/ou qu'il est situé derrière un pare-feu, il est nécessaire d'accéder physiquement à la machine. La syntaxe est similaire à un tunnel ordinaire, à l'exception de l'option -L qui est ici remplacée par -R :

```
client:~$ ssh -Nf -R 26000:127.0.0.1:22 -p 2222 login@server.org
```

- **-N** indique au client de ne pas se connecter mais de se mettre en attente ; en l'occurence il va écouter le port 22 ;
- **-f** détache le client de la console, il continuera donc de tourner en arrière-plan ;
- **-R** pour « reverse » indique qu'on met en place un tunnel inversé ;
- **26000:127.0.0.1:22** indique au serveur de rediriger son port 26000 vers le port 22 de la machine locale (localhost ou 127.0.0.1) ; le port du serveur (26000) peut être remplacé par n'importe quel port disponible ; le port du client (22) est à remplacer par le port sur lequel le serveur ssh du client écoute.
- **-p 2222** indique au client ssh de communiquer vers le port 2222 du serveur (au lieu du 22 par défaut) ;
- la dernière partie est bien sûr à remplacer par un login et une adresse de serveur valides.

3. Accès au client depuis le serveur

Tout naturellement avec la redirection que nous avons effectuée plus haut, sur le serveur nous établissons une connexion ssh vers le port 26000 local :

server:~$ ssh -p 26000 127.0.0.1

…ce qui nous connectera au serveur ssh du client par le truchement du tunnel.