DEPLOY - Secure SHell

Déploiement SSH

Pourquoi arrêter de taper son mot de passe ?

Le protocole SSH (Secure Shell) est la colonne vertébrale de l'administration système. Par défaut, il utilise un mot de passe. Le problème avec les mots de passe, c'est qu'ils sont souvent faibles ("admin123" ou "P@ssw0rd" , sérieusement ?) et qu'ils sont vulnérables aux attaques par force brute.

Nous allons utiliser la **cryptographie asymétrique** (Paire de clés) :

La Clé Privée (id_rsa)

id-rsa, qui contient la clé privée de la paire. Ce fichier ne doit jamais être partagé ou copié sur un système distant, sauf s'il s'agit d'un système contrôlé par le détenteur de la paire de clés.

C'est votre identité. Elle reste sur VOTRE machine. Si vous la perdez ou la partagez, vous êtes compromis. C'est l'équivalent de la clé physique de votre maison.

La Clé Publique (id_rsa.pub)

id-rsa.pub, qui contient la clé publique de la paire. Ce fichier peut être partagé et copié sur d'autres systèmes, en particulier lorsque l'utilisateur prévoit de se connecter à ces systèmes à l'aide de SSH.

C'est ce que vous déposez sur le serveur distant. C'est l'équivalent de la serrure que vous installez sur la porte. N'importe qui peut voir la serrure, mais seul celui qui a la clé privée peut l'ouvrir.

Le fichier authorized_keys

Côté serveur, votre clé publique est stockée dans un fichier spécifique : ~/.ssh/authorized_keys. Quand le serveur reçoit une connexion, il vérifie si la clé privée de l'utilisateur correspond à l'une des clés publiques listées dans ce fichier.

Le fichier config (Côté Client)

Si le serveur a son registre (authorized_keys), votre machine locale a son propre carnet d'adresses intelligent : le fichier ~/.ssh/config.

Sans ce fichier, pour vous connecter, vous devez taper des commandes à rallonge du type :

```
ssh -i ~/.ssh/id_rsa mon_user_complique@192.168.1.50
```

Le fichier config nous permet de mapper toute cette complexité vers un simple **alias**. C'est un gain de temps critique pour tout administrateur qui se respecte. Nous y définissons :

- **Host :** Le surnom que nous utiliserons (ex: srv-prod).
- **HostName :** La véritable adresse IP ou nom de domaine.
- **User :** L'utilisateur distant.
- **IdentityFile :** Le chemin précis de la clé privée à utiliser.
- **Et d'autres options par la suite ...**

TD (Travaux Dirigés) : Votre première connexion sans mot de passe

Prérequis :

- Une machine locale (votre PC Linux).
- Une machine distante (VM ou Serveur) avec un accès SSH par mot de passe fonctionnel.

Étape 1 : Générer la paire de clés

Sur votre machine **locale**, ouvrez un terminal. Nous allons générer une paire de clés RSA standard.

-t rsa : Type de chiffrement

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

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/tuto/.ssh/id_rsa): # <-- ENTRER
Created directory '/tuto/seed/.ssh'.
Enter passphrase for "/home/tuto/.ssh/id_rsa" (empty for no passphrase): # <-- ENTRER
Enter same passphrase again: # <-- ENTRER
Your identification has been saved in /home/tuto/.ssh/id_rsa
Your public key has been saved in /home/tuto/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:4sgeNZsngksY0fsULrpdoU4Bdj92MiXla38SNgc6xgs tuto@laptop
The key's randomart image is:
+---[RSA 3072]----+
|                 |
|     .           |
|    + + +        |
|   . = O o       |
|    E / S o      |
|   o X ^ X       |
|  o B O * .      |
| . = * + o       |
|  . o o          |
+----[SHA256]-----+
```

*Appuyez sur **Entrée** pour accepter l'emplacement par défaut (/home/user/.ssh/id_rsa). Pour ce niveau débutant, laissez la "passphrase" vide (appuyez deux fois sur Entrée).*

Étape 2 : Déployer la clé (La méthode propre)

Inutile de copier-coller le texte de la clé à la main (c'est le meilleur moyen de casser le format). Utilisez l'outil dédié.

Remplacez 'root' et '192.168.200.252' par vos infos réelles

```
ssh-copy-id main@192.168.200.252
```

```
The authenticity of host 'server (192.168.200.252)' can't be established.
RSA key fingerprint is 4c:87:08:f7:34:31:b6:2d:66:4d:19:bd:06:b7:6d:77.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'server,192.168.200.252' (RSA) to the list of known hosts.
main@server's password: # <-- ENTRER LE PASSWORD de l'utilsateur (ici ROOT)
Now try logging into the machine, with "ssh 'main@server'", and check in:

  .ssh/authorized_keys

to make sure we haven't added extra keys that you weren't expecting.
```

Le système vous demandera votre mot de passe utilisateur une dernière fois pour installer la clé.

Étape 3 : La connexion

Testez la connexion. Le serveur ne doit plus vous demander de mot de passe.

```
ssh main@192.168.200.252
```

```
Last login: Mon Dec 23 01:42:26 2013
```

Étape 4 : Simplifier la vie avec config

Les sysadmins paresseux sont les meilleurs sysadmins. Au lieu de taper l'IP et l'utilisateur à chaque fois, nous allons créer un alias.

Créez ou modifiez le fichier ~/.ssh/config sur votre machine locale :

```
nano ~/.ssh/config
```

```
Host mon-serveur
    HostName 192.168.200.252
    User main
    IdentityFile ~/.ssh/id_rsa
```

Sauvegardez (Ctrl+O, Entrée, Ctrl+X).

Maintenant, connectez-vous simplement avec :

```
ssh mon-serveur
```