Installez un serveur Linux et sécurisez son accès distant

Exemple de corrigé pour l'activité "À vous de jouer"

1. Test l'accès SSH au serveur

Depuis votre machine cliente, connectez-vous au serveur avec l'utilisateur `admin`.

```
None
ssh admin@192.168.1.50
```

Remplacez `192.168.1.50` par l'adresse IP de votre serveur.

Résultat attendu lors de la première connexion :

```
None
The authenticity of host '192.168.1.50' can't be
established.
ED25519 key fingerprint is SHA256:...
Are you sure you want to continue connecting
(yes/no/[fingerprint])?
```

Saisissez :

```
None
yes
```

Puis entrez le mot de passe de l'utilisateur `admin`.

Résultat attendu :

```
None
admin@serveur:~$
```

➜ La connexion SSH fonctionne. L'empreinte de la clé hôte du serveur a été acceptée et mémorisée par votre machine cliente.

2. Mise en place l'authentification par clé SSH

Sur votre machine cliente, générez une paire de clés SSH avec l'algorithme Ed25519.

```
None
ssh-keygen -t ed25519
```

Résultat attendu :

```
None
Generating public/private ed25519 key pair.
Enter file in which to save the key
(/home/votre_utilisateur/.ssh/id_ed25519):
```

Vous pouvez valider l'emplacement proposé par défaut avec `Entrée`.

Vérifiez ensuite la présence des fichiers dans le répertoire `.ssh` :

```
None
ls -l ~/.ssh
```

Résultat attendu :

```
None
-rw------- 1 user user ... id_ed25519
-rw-r--r-- 1 user user ... id_ed25519.pub
```

➜ Le fichier `id_ed25519` est la clé privée : elle ne doit jamais être partagée. Le fichier `id_ed25519.pub` est la clé publique : c'est elle qui sera copiée sur le serveur.

Copiez la clé publique vers le serveur :

```
None
ssh-copy-id admin@192.168.1.50
```

Résultat attendu :

```
None
Number of key(s) added: 1

Now try logging into the machine, with:
 "ssh 'admin@192.168.1.50'"
```

Puis testez une nouvelle connexion :

```
None
ssh admin@192.168.1.50
```

Résultat attendu :

```
None
admin@serveur:~$
```

➜ La connexion par clé fonctionne. Le serveur ne demande plus le mot de passe du compte `admin`. Si vous avez défini une phrase de passe pour la clé privée, elle peut encore être demandée localement : ce n'est pas le mot de passe du serveur.

3. Conservation de la session SSH actuelle ouverte

Avant de modifier la configuration du service SSH, gardez votre session SSH actuelle ouverte.

Résultat attendu :

```
None
admin@serveur:~$
```

➜ Cette session sert de filet de sécurité. Si une erreur de configuration empêche une nouvelle connexion SSH, vous pourrez encore utiliser cette session ouverte pour corriger le fichier de configuration.

4. Modification de la configuration du service SSH

Ouvrez le fichier de configuration du service SSH avec les privilèges administrateur.

```
None
sudo nano /etc/ssh/sshd_config
```

Configurez ou vérifiez les directives suivantes :

```
None
PermitRootLogin no
PubkeyAuthentication yes
PasswordAuthentication no
```

➜ La connexion directe au compte `root` est interdite. L'authentification par clé publique est autorisée. L'authentification SSH par mot de passe est désactivée, maintenant que la connexion par clé fonctionne.

5. Test de la syntaxe de la configuration SSH

Avant d'appliquer la nouvelle configuration, testez la syntaxe du fichier.

```
None
sudo sshd -t
```

Résultat attendu :

```
None
```

➜ Si la commande ne renvoie aucun message, la syntaxe est correcte. Si une erreur apparaît, ne rechargez pas encore le service SSH : corrigez d'abord le fichier `/etc/ssh/sshd_config`.

6. Recharge du service SSH

Rechargez le service SSH pour appliquer la configuration.

```
None
sudo systemctl reload ssh
```

Vérifiez ensuite que le service est toujours actif.

```
None
systemctl status ssh
```

Résultat attendu :

```
None
Active: active (running)
```

➜ Le service SSH est toujours actif. La nouvelle configuration a été prise en compte sans interrompre l'accès au serveur.

7. Vérification de l'accès dans un nouveau terminal

Sans fermer la session SSH initiale, ouvrez un nouveau terminal sur votre machine cliente et testez une nouvelle connexion.

```
None
ssh admin@192.168.1.50
```

Résultat attendu :

```
None
admin@serveur:~$
```

➜ La connexion SSH fonctionne toujours après la modification de la configuration du service SSH. L'utilisateur `admin` peut se connecter au serveur avec sa clé SSH, tandis que l'authentification par mot de passe est désormais désactivée.