SAÉ SCR – BUT 1 – S2

Installer des paquets sur ArchLinux

```
[root@VM ~] # pacman -Sy <nom des paquets à installer>
```

`pacman` est le gestionnaire de paquet sur ArchLinux, il permet d'installer des programmes, la liste des programmes est disponible sur archlinux.org/packages
Lorsqu'on installe plusieurs paquets d'un coup, on sépare le nom des paquets par des espaces.
Il est nécessaire d'avoir les permissions administrateur (`root`) afin d'installer un programme avec le gestionnaire de paquets.

Mise à jour du système ArchLinux

```
[root@VM ~] # pacman -Sy archlinux-keyring
[root@VM ~] # pacman -Syu
```

La première ligne met à jour les clés PGP (clé publique de chiffrement) nécessaires à la validation de l'intégrité des paquets. La seconde ligne met à jour le système.

Partage de fichier Windows

Sous Windows avec le compte Albatros

- Étape 1 : Ouvrez l'Explorateur Windows depuis la barre des tâches
- Étape 2 : Cliquez sur Réseau
- Étape 3 : Authentifiez-vous avec le compte administrateur afin de valider la modification
- Étape 4 : Faites un clic droit sur Réseau puis cliquez sur Propriété et cliquer sur Afficher l'intégralité du réseau pour obtenir une carte du réseau
- Étape 5 : Repérez la machine (`VM-FreSel1`) sur laquelle se trouve le partage réseau sur la carte, puis en passant la souris sur la machine, notez son adresse IPv4 (`192.168.2.247`)
- Étape 6 : Fermez la fenêtre et revenez sur l'Explorateur Windows et ouvrez la machine sur laquelle se trouve le partage réseau en double cliquant sur son nom (trouvé à l'étape précédente)
- Étape 7 : Explorez les fichiers pour trouver le nom d'utilisateur du compte partageant ses fichiers, dans le dossier `Users` nous pouvons voir que nous pouvons accéder au dossier utilisateur du compte `Diamant`, et en continuant d'explorer le fichier nous trouvons un dossier `WIN-SHARE` qui semble être le dossier partagé, notez donc le nom d'utilisateur (Diamant) ainsi que le chemin d'accès au dossier (`\Users\Diamant\Documents\WIN-SHARE`)

Sous Linux

- Étape 1 : Se connecter sur le compte root
- Étape 2 : Ouvrez le terminal et tapez la commande `mkdir /mnt/share` pour créer le point de montage
- Étape 3 : Installez le paquet `cifs-utils`
- Étape 4 : Redémarrer le système à l'aide de la commande `reboot`
- Étape 5 : Se reconnecter sur root
- Étape 6 : On va modifier la configuration de `fstab`, c'est l'outil qui permet de gérer les montages. Ouvrez le fichier `/etc/fstab` avec un éditeur tel que `vim` et ajoutez cette ligne :

```
//192.168.2.247/Users/Diamant/Documents/WIN-SHARE /mnt/share cifs username=Diamant,password=Diamant,rw,user,noauto 0 0
```

- Étape 7 : Déconnectez-vous du compte root en cliquant sur le nom du compte en haut à droite de l'écran puis sur Déconnexion puis à nouveau sur Déconnexion et se connecter sur le compte d'Alice
- Étape 8 : Monter le partage avec la commande `mount /mnt/share`

- Étape 9 : Testez l'écriture de fichier sur le partage réseau à l'aide de la commande `echo "test" > /mnt/share/test.txt`
- Étape 10 : Testez la lecture du fichier que vous venez de créer à l'aide de la commande `cat /mnt/share/test.txt`
- Étape 11 : Supprimez le fichier à l'aide de la commande `rm /mnt/share/test.txt`
- Étape 12 : Tapez la commande `umount /mnt/share` afin de démonter le partage réseau

Partage de fichiers Samba

Sur Linux

- Étape 1 : Installez le paquet de `samba`
- Étape 2 : Créer le répertoire de partage avec la commande `mkdir -p /export/share`
- Étape 3 : On va créer le répertoire privé de Alice : `mkdir -p /export/home/alice` faites pareil pour bob
- Étape 4 : Définissez les permissions de `/export/share`, de sorte que tout le monde puisse lire, écrire et exécuter, utiliser la commande : `chmod 777 /export/share`
- Étape 5 : Définissez les permissions de `/export/home/alice`, de sorte que seulement Alice ai les droits, utiliser la commande : `chmod 700 /export/home/alice` pareil pour bob
  Étape 6 : Définissez le propriétaire du répertoire `/export/home/alice` avec la commande `chown alice:developpeurs /export/home/alice`
- Étape 7 : Définissez le propriétaire du répertoire `/export/home/bob` avec la commande `chown bob:commerciaux/export/home/alice`
- Étape 8 : Entrez la configuration suivante dans le fichier `/etc/samba/smb.conf` :

```
[global]
  workgroup = WORKGROUP
  server role = standalone server
  log file = /var/log/samba/%m.log
  max log size = 50
[export]
  path = /export
  read only = no
```

- Étape 9 : Dans le terminal, faire en sorte que le service samba s'active à chaque démarrage avec la commande `systemctl enable smb.service`
  Étape 10 : Lancer le service samba avec la commande `systemctl start smb.service`
- Étape 11 : Définir un mot de passe samba pour l'utilisateur bob avec la commande `smbpasswd -a bob` pareil pour alice

Depuis une autre machine pour faire des tests

- Étape 1 : Créer un répertoire test avec la commande `mkdir test`
- Étape 2 : monter le partage réseau avec la commande suivante

```
[root@VM ~] # mount.cifs //<Adresse IP>/export test -o user=bob,password=bob
```

- Étape 3 : Démonter le partage réseau avec la commande `umount test`
- Étape 4 : Supprimer le répertoire test avec la commande `rm -r test`

Sur Windows

- Étape 1 : Ouvrir l'explorateur Windows
- Étape 2 : Dans la barre d'adresse, taper `\\VM`
- Étape 3 : S'identifier avec le compte de bob

SSH

Installez le paquet `openssh`

On peut vérifier que l'authentification par mot de passe est activée en consultant le fichier de configuration situé dans `/etc/ssh/sshd_config` On peut faire ça à l'aide de la commande `grep` :

```
[root@VM ~] # grep PasswordAuthentication /etc/ssh/sshd_config
```

Les clés du serveur seront créées puis stockées dans le dossier `/etc/ssh` lorsque on démarrera le serveur, on peut consulter le contenu du dossier en utilisant la commande `ls`.

On doit maintenant activer et démarrer le serveur SSH :

```
[root@VM ~] # systemctl enable sshd.service
[root@VM ~] # systemctl start sshd.service
```

Pour vérifier que le serveur est actif on peut utiliser la commande ss qui indique les ports utilisés, on cherche le port SSH (22) :

```
[root@VM ~] # ss -na | grep :22
```

On peut désormais se connecter à distance à la machine en utilisant la commande suivante :

```
[root@VM ~] # ssh <nom d'utilisateur>@<adresse IP de la machine>
```

L'adresse IP de la machine peut être trouvée avec la commande `ip a`.

En consultant à nouveau le contenu du dossier `/etc/ssh`, on observe que les clés ont été créées.

SSH utilise des empreintes (fingerprint) pour assurer l'authenticité du serveur, elle est stockée chez les clients et une vérification est faite à chaque connexion, on utilisera la commande suivante pour créer l'empreinte à partir de la clé publique :

```
[root@VM ~] # ssh-keygen -lf /etc/ssh/ssh_host_rsa_key.pub
```

Pour se connecter plus facilement, on va créer des clés SSH, elles permettent de se connecter sans devoir entrer de mot de passe. Commençons par se connecter au compte d'Alice, la commande `su` permet de le faire directement dans le terminal :

```
[root@VM ~] # su alice
```

Cette commande conserve le dossier de travail actif (le dossier personnel du compte `root`), on utilisera alors la commande `cd` pour se placer dans le dossier personnel d'Alice.

Pour créer une paire de clés SSH, on utilisera encore une fois la commande `ssh-keygen` mais cette fois-ci sans arguments, on laissera tous les paramètres par défaut en pressant la touche Entrer à chaque fois.
La clé publique se trouve dans le dossier personnel à l'emplacement suivant `.ssh/id_rsa.pub`, on va la transférer en SFTP à une autre machine via la commande `scp`, on doit enregistrer la clé publique dans le fichier `.ssh/authorized_keys` :

```
[alice@VM ~] # scp .ssh/id_rsa.pub <Adresse IP>:.ssh/authorized_keys
```

La paire de clés a été générée en utilisant l'algorithme RSA, mais on peut utiliser d'autres algorithmes tel que ECDSA en utilisant l'argument `-t`.

```
[alice@VM ~] # ssh-keygen -t ecdsa
```

On peut stocker plusieurs clés dans le fichier `.ssh/authorized_keys` en les mettant sur plusieurs lignes, on va donc envoyer la clé publique ECDSA dans un nouveau fichier temporaire, puis se connectera à la machine pour ajoutera le contenu de ce fichier dans `authorized_keys` :

```
[alice@VM ~] # scp .ssh/id_ecdsa.pub <Adresse IP>:id_ecdsa.pub
[alice@VM ~] # ssh <Adresse IP>
[alice@VM ~] # cat id_ecdsa.pub >> .ssh/authorized_keys
```

La commande `cat` permet d'afficher le contenu d'un fichier, et `>>` permet de rediriger la sortie du terminal vers un fichier l'ajoutant à la fin.

Pour finir, il faut modifier le fichier de configuration du serveur SSH pour qu'il permette la connexion par clé SSH, pour cela ouvrez le fichier `/etc/ssh/sshd_config` avec un éditeur tel que `vim`, cherchez la ligne contenant `PubkeyAuthentication`, décommentez là et remplacez `no` par `yes`.

Sous Windows

- Se connecter sous le compte d'Albatros
- Aller sur putty.org pour installer le logiciel, il nous permettra de nous connecter en SSH à une machine.
- Télécharger et installer `putty-64bits-0.78-installer.msi`
- Ensuite il va falloir lancer le programme PuTTYgen
- On va dès lors cliquer sur le bouton `generate` qui va générer une clef privée et une clef publique
- Enregistrer la clef publique en cliquant d'abord sur `Save public key`, ensuite on va l'enregistrer en tant que `id_rsa.pub` dans le dossier `Albatros`, comme ceci : `C:\Users\Albatros\id_rsa.pub`
- Enregistrer la clef publique en cliquant d'abord sur `Save private key`, ensuite on va l'enregistrer en tant que `id_rsa.ppk` dans le dossier `Albatros`, comme ceci : `C:\Users\Albatros\id_rsa.ppk`
- Après avoir fait cela il va falloir copier la clef publique qui est affichée
- Ouvrir le bloc note, coller la clef qu'on vient de copier et à la fin ajouter une ligne vide
- Enregistrer le fichier en tant que `C:\Users\Albatros\Documents\cle.dat`
- Ouvrir PSFTP
- Et dans le terminal qui vient de s'ouvrir marquer :

```
open alice@<Adresse IP>
put C:\Users\Albatros\Documents\cle.dat
quit
```

- Maintenant supprimer le fichier `C:\Users\Albatros\Documents\cle.dat`
- Ouvrir PuTTY
- Dans le champ Host Name mettre : `VM-2`
- Cliquer sur le bouton Connection, ensuite sur SSH, puis Auth et enfin sur Credentials
  Dans le champ : `Private key file for authentication` sélectionner le fichier `C:\Users\Albatros\id_rsa.ppk`
  Cliquer ensuite sur le bouton Open
  Dans le terminal qui vient tout juste de s'ouvrir faire les commandes suivantes :

```
[alice@VM ~] # cat cle.dat >> .ssh/authorized_keys
```

- Retournez sur PuTTYgen, mais cette fois-ci, utilisez l'algorithme ECDSA puis recommencez les étapes précédentes
- Vous pouvez maintenant vous connecter en utilisant vos clés SSH en les sélectionnant dans `Connexion > SSH > Auth > Credentials > Private key file for authentication`