

Pratique des Machines

TP

Halim Djerroud (hdd@ai.univ-paris8.fr)
Mehdy Tounsi (mehdy.tounsi02@univ-paris8.fr)
Stephane Malolepsza (stephane.malolepsza@univ-paris8.fr)

Le TP s'effectuera sur une machine virtuelle Debian , plus précisément un container Docker. Vous serez connectés sur ce container en root. Les containers Docker ne permettent pas de sauvegarder les changements lorsque l'on se déconnecte. Si vous vous déconnectez de la machine virtuelle ,au prochain lancement vos modifications n'auront pas été sauvegardés et vous n'aurez pas l'historique des commandes .Pensez à noter ce que vous faites.

Exercice 1 : Se connecter, installer un paquet

1. Ouvrez un terminal,lancez la commande "`sudo docker run -it debian bash`"
Si la commande s'est déroulée correctement vous devriez voir un nouveau prompt
2. A partir de ce prompt désignez l'utilisateur, le nom de la machine , quels sont vos privilèges ?
3. Confirmez à l'aide de la commande "`id`"
4. Tentez d'installer le paquet nano. Quelles est la commande ? Quel est le retour de la commande ?
5. Quelle commande lancer pour y remédier ? Lancez la commande puis tentez de réinstaller le paquet précédemment cité.

Exercice 2 : Créer un utilisateur manuellement :

nano est un éditeur de texte en mode console, il faudra l'utiliser pour éditer des fichiers. Si vous n'êtes pas parvenu à l'installer demander de l'aide à l'enseignant.

1. Éditez le fichier `/etc/passwd` avec nano, ajoutez à la fin du fichier cette ligne :

```
usr1:x:1000:1000:usr1,,,:/home/usr1:/bin/bash
```


Puis enregistrez les modifications et quittez l'éditeur nano.
2. Vous venez de créer un nouvel utilisateur `usr1`. Connectez vous avec cet utilisateur à l'aide le commande `login`
3. Y parvenez vous ? Quel est le problème ?

4. Résolvez le problème à l'aide de la commande adéquate, puis tentez une nouvelle connection avec l'utilisateur `usr1`.
5. Une fois connecté avec cet utilisateur , créez un fichier vide `test` sans changer de répertoire. Y parvenez vous ? Quel est le problème
6. Comment résoudre le problème.
7. Une fois le problème résolu , reprenez question 5.
8. Si a nouveau il y a un problème, identifiez le et résolvez le.

Exercice 3 : Partage de fichiers

1. Créez deux nouveaux utilisateurs `usr2` et `usr3` à l'aide de la commande `adduser`
2. Créez un groupe `dev` à l'aide de la commande `addgroup`.
3. Créez un répertoire `/opt/dev`. Changez son groupe d'appartenance, nouveau group `"dev"`
4. Changez les permissions de ce répertoire et/ou changez l'appartenance aux groupes des utilisateurs de telle sorte que :
 - `usr1` ne peut pas voir le contenu de ce répertoire
 - `usr2` peut voir et modifier le contenu de ce répertoire
 - `usr3` quand a lui ne peut simplement que voir le contenu du répertoire sans pouvoir faire de modification.
5. Faites tous les tests nécessaires pour vérifier que les contraintes citées ci dessus sont respectées.

Exercice 4 :Supprimez des utilisateurs

1. Supprimez les utilisateurs créés , supprimez les groupes , supprimez le répertoire `/opt/dev`
2. Vérifiez le contenu des fichiers `/etc/passwd`, `/etc/group`
3. Vérifiez le contenu du répertoire `/home`

Exercice 5 :Pour aller plus loin

Vous êtes connectés sur un container docker en root et avez donc la possibilité d'expérimenter sans crainte. N'hésitez pas

1. Installez le paquet `net-tools`.
2. Récupérez l'adresse IP du container à l'aide de la commande `ifconfig` (commande qui fait partie du paquet précédemment installé)
3. Installez le paquet `openssh-server`
4. Créez un utilisateur `user`
5. Ouvrez un autre onglet sur votre terminal (raccourci clavier CTRL+SHIFT+t)
6. Vérifiez que vous êtes bien sur la machine octopussy. Si oui lancez la commande `ssh user@[Ip]`

7. Ça ne fonctionne pas, retournez sur l'onglet de votre container et lancez le service ssh "`service ssh start`", re-essayez la connection ssh.

L^AT_EX