

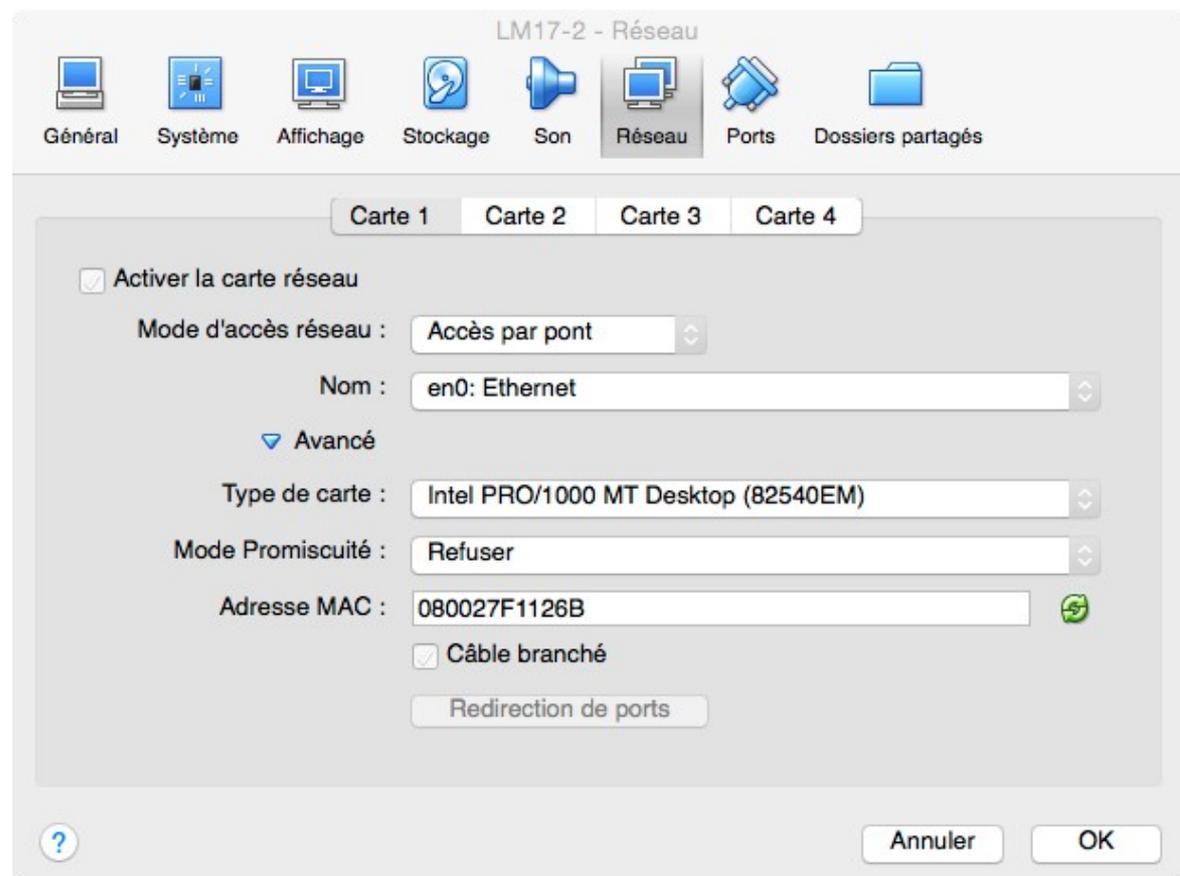
TP Service sur Réseaux N°2 - MMI 2

Ce TP aborde la gestion des comptes utilisateurs sous linux, introduit la notion de privilège et par conséquent la notion de sécurité. Ce TP fait beaucoup appel aux commandes vues précédemment.

Introduction

Ouvrez VirtualBox.

Modifiez les paramètres réseaux de la VM, sélectionnez un accès par pont. Régénérez l'adresse MAC (petit icône vert à coté de l'adresse).



et assurez vous que vous allez démarrer la machine LM17 à partir du snapshot 'Fresh Install'.

Une fois la VM démarrée, ouvrez le menu et lancez Konsole.

Ajouter un utilisateur

La commande `adduser` permet d'ajouter un utilisateur. Exécutez la commande suivante afin de créer un utilisateur appelé `toto`

```
adduser toto
```

Attention



Il existe également une commande qui s'appelle `useradd`. Ne les confondez pas : `useradd` permet aussi de créer un utilisateur mais sans mot de passe, sans home ...

Quel message est renvoyé par le SE ?

-

Super utilisateur

Seul le super utilisateur `root` peut créer un utilisateur dans le système. Afin d'exécuter une commande « comme si » vous étiez le super utilisateur, relancez la commande précédente comme ceci :

```
sudo adduser toto
```

Vous êtes invités à taper votre mot de passe (celui qui a permis d'ouvrir la session).

Note

`sudo` signifie 'Super User Do'

Et répondez aux questions demandées. Maintenant, comment vérifier si l'utilisateur a été effectivement créé ? Il suffit de consulter le fichier `/etc/passwd`, qui est le fichier définissant les utilisateurs du système, afin de voir si il contient le user `toto`.

Chaque ligne de ce fichier est de la forme :

```
toto:x:1001:1001:,,,:/home/toto:/bin/bash
```

Description des champs :

- user (encore appelé nom d'utilisateur ou login)
- ancienne place du mot de passe crypté
- numéro d'utilisateur **uid**, sa valeur est le véritable identifiant pour le système Linux; l'uid de **root** est toujours 0.
- numéro de groupe **gid**, dans lequel se trouve l'utilisateur par défaut; le **gid** de **root** est toujours 0.
- nom complet, il peut être suivi d'une liste de renseignements personnels
- home directory
- l'interpréteur de commandes (shell) à utiliser pour ce user

Compte tenu des commandes que vous connaissez déjà, comment vérifier si le fichier sus-nommé contient la chaîne de caractère `toto` ?

-

Changer d'identité

Vous venez de créer un nouvel utilisateur. Ouvrez à présent une nouvelle fenêtre de terminal grâce aux commandes CTRL+Shift+N.

Vous pouvez vous identifier en tant que *toto* grâce à la commande suivante :

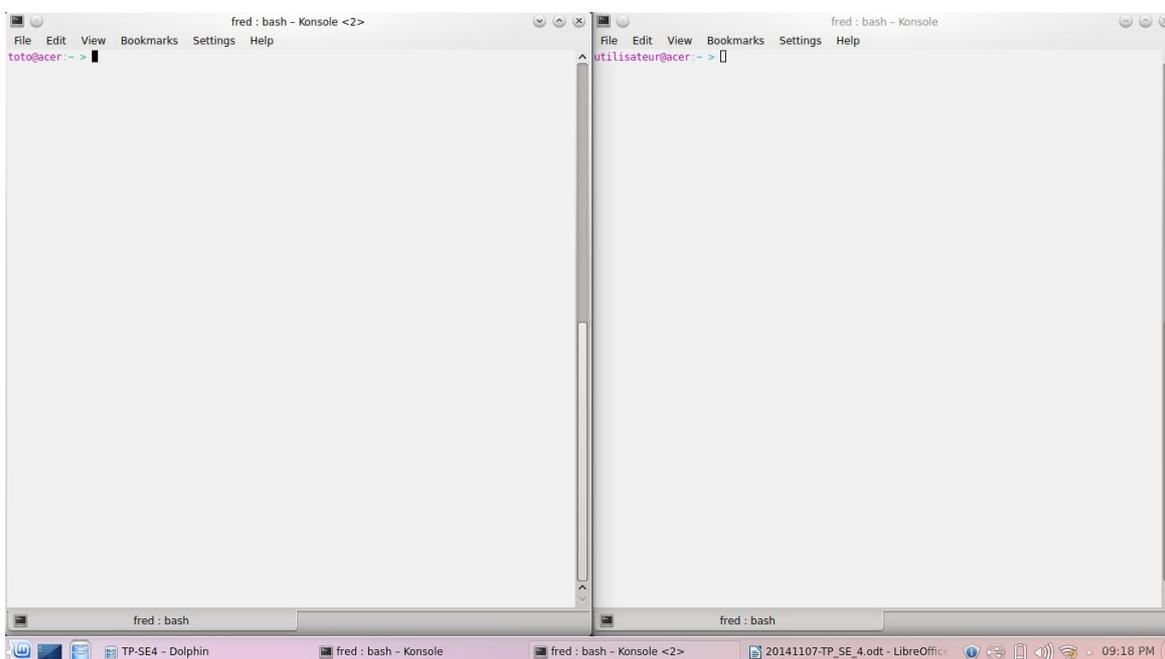
```
su - toto
```

Attention



Ne confondez pas la commande *sudo* qui permet de lancer une autre commande 'en tant que' *root*, et la commande *su*, qui permet de changer d'identité.

Le prompt a changé, il vous indique qu'à présent vous êtes loggés en tant que *toto*. Vous avez donc une fenêtre où vous êtes *utilisateur* et une autre où vous êtes *toto*. Comme ceci :



Dans la fenêtre où vous êtes *toto*, créez un fichier appelé *fichier.txt* contenant le texte « Vive les MMI ! ». Indiquez ci-dessous la commande utilisée :

-

Quel est le chemin absolu du fichier en question ?

-

Quels sont les droits de *fichier.txt* ?

-

Depuis la fenêtre où vous êtes *utilisateur*, tentez d'afficher le contenu de *fichier.txt*. Indiquez la commande utilisée :

-

Comment retirer le droit au user *utilisateur* d'afficher le contenu de *fichier.txt* ?

-

Tentez à nouveau d'afficher *fichier.txt* en tant que *utilisateur*, afin de vérifier l'efficacité de votre commande précédente.

Variables d'environnement

Dans votre fenêtre où vous êtes utilisateur, tapez la commande suivante :

echo \$HOME

Que représente la variable d'environnement \$HOME ?

-

Refaites la même opération dans la fenêtre où vous êtes *toto*. Le résultat est différent. Qu'en concluez vous ?

-

Gestion des groupes

La commande *groupadd*, permet de créer un nouveau groupe. Voici un exemple, qui permet de créer un groupe appelé *MMI* :

```
sudo groupadd MMI
```

Exécutez cette commande à partir de la fenêtre où vous êtes identifié en tant que *utilisateur*.

Le fichier qui définit les groupes est */etc/group*. Il est de la forme :

```
sambashare:x:111:user1, user2, ..... usern
```

Description des champs :

- nom du groupe
- champ non utilisé
- numéro du groupe, c'est à dire l'identifiant **gid**
- liste d'utilisateur faisant partie du groupe

A présent, notez le numéro du groupe MMI :

-

La commande `id` vous donne des informations pertinentes concernant l'appartenance à votre groupe primaire, et les autres. Exemple :

```
utilisateur@acer:~ > id
uid=1000(utilisateur) gid=1000(utilisateur)
groups=1000(utilisateur),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),108(lpadmin),111(sambashare),129(wireshark)
```

Ici le user `utilisateur` fait partie d'un groupe primaire appelé `utilisateur` mais il fait également partie des groupes `adm`, `cdrom`, `sudo` ... etc.

Note

Par défaut sur les distributions linux modernes, chaque utilisateur fait partie d'un groupe du même nom. En revanche sur Mac le groupe par défaut est le même pour chaque utilisateur. Son nom est `staff`.

Nous allons rassembler ces 2 users (`utilisateur` et `toto`) au sein du même groupe MMI.

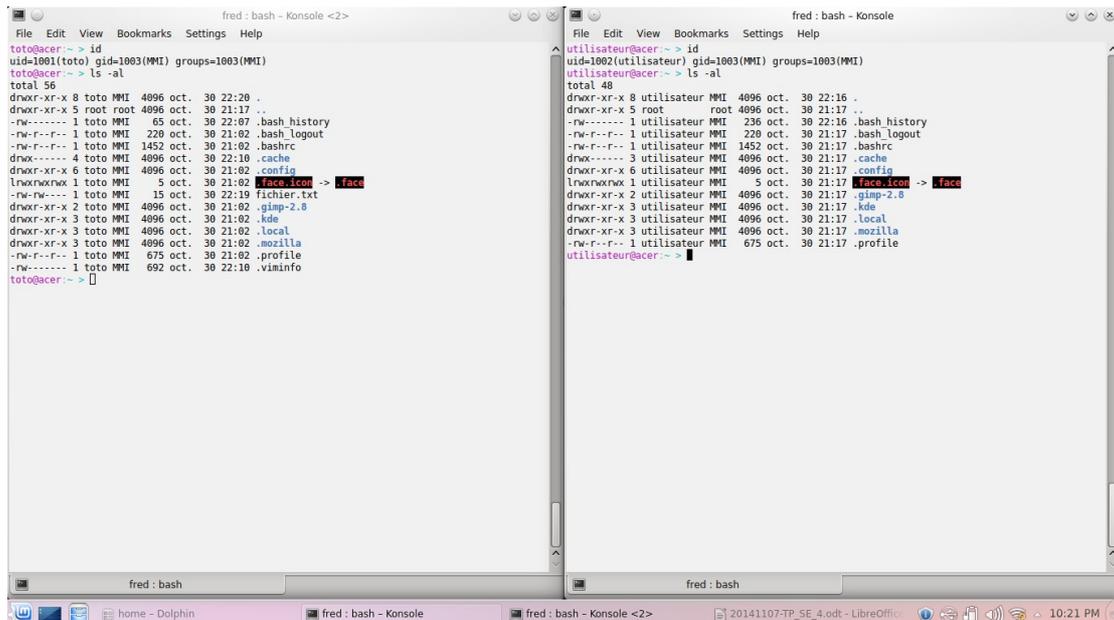
Afin de changer de groupe primaire le user `utilisateur`, utilisez la commande `usermod` en spécifiant le nouveau numéro de groupe :

```
sudo usermod utilisateur -g <gid>
```

Faites la même chose pour le user `toto`, toujours à partir de la fenêtre où vous êtes loggés en tant que `utilisateur`. Indiquez la commande utilisée :

-

Vérifiez le résultat en visualisant le contenu des deux `home directory`.



The image shows two terminal windows side-by-side. The left window shows the output of the `id` command for user `toto`, indicating a primary group of `MMI` (gid=1003) and several secondary groups. The right window shows the output of the `id` command for user `utilisateur`, indicating a primary group of `MMI` (gid=1003) and several secondary groups. Both windows also show the output of the `ls -al` command, displaying the contents of the user's home directory, including files like `.bash_history`, `.bash_logout`, `.bashrc`, `.cache`, `.config`, `.face.icon`, `.gimp-2.8`, `.kde`, `.local`, `.mozilla`, `.profile`, and `.viminfo`.

A partir de la fenêtre où vous êtes identifié en tant que *toto* tentez de créer un groupe appelé *test*.

```
sudo groupadd test
```

Vous n'avez pas le droit. Pourquoi *toto* n'a pas ce droit mais, on l'a vu tout à l'heure, le user *utilisateur* le peut ?

La différence est toute simple : parce que *utilisateur* fait partie d'un groupe spécial, qui s'appelle ... *sudo*. Le fait d'appartenir à ce groupe lui permet d'obtenir le privilège de lancer des commandes en tant que *root*.

Trouvez les options à *usermod* qui permettent d'ajouter le user *toto* à ce groupe.

Afin que le système comprenne que désormais *toto* possède les droits nécessaires pour devenir *root*, il faut fermer la session shell (CTRL+D) et refaire *su - toto*.

Maintenant vous pouvez vous identifier en tant que *toto*, et exécuter *sudo groupadd test*.

Note

Pour retirer un utilisateur d'un groupe, le plus simple est d'éditer le fichier */etc/group*.

Connexion distante

A présent, nous allons tenter de nous connecter à la machine de votre voisin de droite.

Il existe plusieurs commande qui permettent une connexion distante (telnet, rlogin...) mais la plus répandue (et safe) est *ssh*. La syntaxe est la suivante :

```
ssh user@host
```

où *user* est le nom d'utilisateur de la machine distante, *host* est soit l'adresse IP soit le nom de ladite machine.

Exemple :

```
ssh utilisateur@INF-356
```

ou bien

```
ssh toto@10.40.1.224
```

Lors de la première connexion, il va vous être demandé si vous acceptez la clé publique, répondez 'y'.

A l'issue de la connexion, le prompt aura changé : le nom de la machine en particulier sera celui de votre voisin.

Créez maintenant un fichier vide, dans le home directory. Demandez à votre voisin de confirmer la création du fichier.

Fermez la connexion en tapant CTRL+D

Copie distante

Basé sur *ssh*, la commande *scp* permet de copier des fichiers ou des répertoires depuis une machine vers une autre. La syntaxe est la suivante :

```
scp [-12346BCpqr] [-c cipher] [-F ssh_config] [-i identity_file] [-l limit] [-o ssh_option] [-P port] [-S program] [[user@]host1:]file1 [[user@]host2:]file2
```

Note

Les arguments entre crochets sont optionnels

Trouvez la commande nécessaire permettant de copier un fichier (n'importe lequel, faites votre choix) depuis votre machine vers celle de votre voisin de droite, dans son home directory

-.

Suppression

Les commandes qui permettent de supprimer les utilisateurs ou les groupes sont respectivement *userdel* et *groupdel*.

Trouvez les commandes permettant de supprimer le user *toto* avec son home directory ainsi que le group *toto*.

-