R101 TP 6

Mon premier client / serveur

L'objectif de ce TP est d'une part de se familiariser avec la gestion des utilisateurs sur une machine ainsi que de découvrir les systèmes de droits, et d'autre part de réaliser un client / serveur d'authentification sous linux. Pour la deuxième partie de ce TP il faudra obligatoirement travailler avec 2 machines dont 1 jouera le rôle de serveur et l'autre de client.

1. Gestion des comptes locaux

Il s'agit dans un premier temps de manipuler les comptes des utilisateurs présents localement sur votre machine. Un des rôles de l'administrateur réseau est de gérer cela pour des raisons évidentes de sécurité : tout le monde n'a pas le droit de se connecter sur les machines de votre entreprise.

1. Comptes locaux et groupes

Les comptes locaux sur la machine sont stockés dans le fichier `/etc/passwd`.

1. Visualiser le contenu du fichier /etc/passwd. Quelles informations peut on tirer concernant le compte « test » ?
2. A quoi servent les groupes définis sur le système d'exploitation ?

Les groupes sont stockés dans le fichier `/etc/group`.

3. Quels sont les groupes correspondants à des utilisateurs.
4. A l'aide de la commande groups déterminez la liste des groupes du compte « test ». Afficher à l'aide du fichier /etc/group la liste des membre du groupe « audio ».

Les mots de passe des utilisateurs sont quant à eux enregistrés dans le fichier `/etc/shadow`.

5. Afficher le contenu du fichier /etc/shadow. Est-ce que l'utilisateur test peut le faire ? Pourquoi ?
6. Comparer le mot de passe de test avec ce qui est enregistré dans le fichier /etc/shadow. Quel est l'état du mot de passe ?
7. Visualiser le manuel de la commande mkpasswd. Essayer de crypter le mot de passe du compte test (test) avec les différentes méthodes possibles. Comparer le résultat avec ce qui est présent dans le fichier /etc/shadow. Quelle méthode de cryptage a été utilisée ? Cryptez plusieurs fois de suite le mot test, le résultat est il toujours le même ?

2. Création d'utilisateurs / groupes

Maintenant que nous avons rapidement vu les propriétés d'un utilisateur, nous allons en rajouter d'autres. Dans un premier temps en local.

1. Pour ajouter un utilisateur, on utilise la commande **`useradd`** . A l'aide du manuel de la fonction ajoutez un utilisateur titi dont l'UID est 1500 et lui ajoutant un répertoire de base et personnel /home/titi et qu'il fasse partie des groupes titi (par défaut) ainsi que wireshark.
2. On va maintenant vérifier si Linux est bien un système d'exploitation multi-user. Ouvrir le terminal 1 (CTRL + ALT + F1). Essayer de se connecter avec le compte de titi. Que se passe-t-il ?
3. Se connecter en tant que root. Afficher le contenu du fichier /etc/passwd. Trouvez vous les informations sur le compte de titi ? Quelle information peut on avoir sur le mot de passe ?
4. Visualiser le contenu de /etc/shadow. Quel est le mot de passe de titi ?
5. On va modifier maintenant le mot de passe de titi à l'aide de la commande **`passwd`**. Attribuer un mot de passe au compte de titi (si possible différent de vos voisins).
6. Déconnecter le root et essayer de se connecter en tant que titi. Que se passe t'il ? Peut on dire que le système est multi-user ? Est ce que test est toujours connecté sur la session graphique (on bascule en mode graphique à l'aide de CTRL + ALT + F7).
7. Dans une console utiliser la commande **`who`** qui indique les personnes connectées sur la machine. Cela corrobore-t-il vos suppositions ?
8. Quel est le répertoire courant au login de titi (commande **`pwd`**) ?
9. Afficher les droits des fichiers contenus dans le répertoire /home. Quels sont ceux attribués

aux répertoires de titi et test ?

10. Est-ce que titi peut se rendre dans le répertoire de test ? Peut-il lire son contenu ? Peut-il créer un document dedans ? Est-ce que tout ça est compatible avec les droits que vous avez vu ?
11. Est-ce que l'administrateur peut créer un fichier dans le répertoire de titi ? Est ce normal ? Quel pourrait être le problème si l'administrateur n'avait pas les droits sur les fichiers ?
12. Essayer de vous connecter sur la machine de votre voisin avec votre compte titi ? Est-ce que cette solution est viable pour un grand parc informatique ? Justifier votre réponse : quelles sont les limites ?

2. Gestion des comptes réseau

Nous avons vu que les comptes gérés localement sont utiles dans le cas d'une machine simple utilisateur. Par contre cela devient vite compliqué en réseau avec de multiples utilisateurs se connectant sur une même machine : homogénéité et mise à jour des mots de passe, contenu des répertoires, etc. Nous ne nous intéresserons pas pour l'instant à l'aspect des répertoires mais juste à la gestion centralisée des mots de passe. Pour cela il faut installer un serveur d'authentification. Les machines seront des clients de ce serveur pour permettre au personnes de se connecter.

Remarque : A partir de ce moment il faut travailler en binôme. Une des machine du binôme fera office de serveur l'autre de client.

1. Installation du service (sur le serveur)

Le programme permettant de partager les informations de connexion s'appelle nis (Network Information Service). Il était autrefois appelé yp (comme Yellow Page … et oui les pages jaunes …) et on retrouve ce nom dans les fichiers de configuration et maintenant dans les services également.

1. Installer le paquet nis.
2. Après l'installation configurer le nom du domaine nis. Choisir un nom avec votre nom de binôme et créer un fichier `/etc/defaultdomain` contenant ce nom. Vous utiliserez également la commande `domainname <votreNomDeDomaine>` pour finir la configuration. La configuration peut être longue car le service est lancé et tente de se connecter au serveur qui n'est pas configuré ...
3. Il faut modifier le fichier `/etc/default/nis` et déclarer la machine comme serveur maître. On laissera les autres options inchangées
4. On configure le fichier `/etc/yp.conf` qui renseigne l'adresse ip du serveur. (on pourra mettre soit l'adresse ip du serveur soit localhost sous réserve que localhost soit définit dans le fichier /etc/hosts qui assure la conversion locale entre l'adresse symbolique et l'adresse ip (par défaut c'est le cas).
5. On lance le serveur à l'aide de la commande systemctl restart ypserv.service (là encore cela peut prendre du temps)
6. On doit maintenant configurer les base de données du service NIS (utilisateurs, machines, etc). Ces bases sont stockées dans le répertoire `/var/yp`. Pour cela on commence par modifier le fichier **Makefile** qui contient quelques directives. On modifiera la ligne gérant les UID et GID min pour les faire commencer à 2000.
7. Quel est l'intérêt de les changer de 1000 la valeur par défaut ?
8. Une fois que le serveur tourne, on initialise le serveur en lançant le script /usr/lib/yp/ypinit. Quelle option faut il utiliser pour lancer ce script sur le serveur ? (voir man avant de lancer le script). On laissera les options non modifiées (CRTRL + D) et on répondra oui à la question posée.
9. Étudier le manuel de la commande ypcat afin de savoir quelles sont les noms exacts des listes exportées. Quel est le nom de la table qui contient la liste des comptes exportés par le service NIS ?
10. Ajouter un nouvel utilisateur dont l'UID est supérieur à 2000 (voir les options de useradd).
11. Réutiliser la commande ypcat pour connaître la liste des utilisateurs. Que constatez vous ?

12. Pour actualiser la liste des utilisateurs sur le serveur, il faut lancer **`make`** dans le répertoire `/var/yp`. Faites le et afficher la liste des utilisateurs exportée par le serveur.
    Les serveur est prêt, il faut maintenant configurer le client.

2. Installation du service sur le client

Maintenant sur le client on doit également installer le service nis.

1. Installer le paquet nis et configurer le même nom de domaine que sur le serveur.
2. Utiliser la commande **`domainname`** pour afficher le domaine par défaut de la machine.
3. Configurer le fichier /etc/yp.conf en renseignant l'adresse ip du serveur.
4. Dans le fichier `/etc/nsswitch.conf` on rajoutera les termes **files nis** sur les lignes de passwd, group et shadow.

> *Le **Name Service Switch** (**NSS**) autorise le remplacement des traditionnels fichiers Unix de configuration (par exemple /etc/passwd, /etc/group, /etc/hosts) par une ou plusieurs bases de données centralisées (wikipedia).*

5. Lancer le service nis sur le client à l'aide de `systemctl start ypbind.service nscd.service` et afficher la liste des utilisateurs disponibles via le NIS à l'aide de la commande ypcat passwd.byname. Que constatez vous ? Visualiser la liste des comptes locaux sur le client. Retrouve-t-on les comptes NIS.
6. Essayer de se logger avec l'utilisateur définit sur le serveur soit sur une autre interface (CTRL+ALT+F2).

> Attention le fait d'être en root fait que le mot de passe n'est pas demandé quand vous changez d'utilisateur.

7. Se déconnecter et lancer sur le client wireshark afin de capturer les trames lors de la connexion. Une fois wireshark actif sur la bonne interface connecter l'utilisateur nis. Arrêter la capture de trame et observer les trames relatives à la connexion (on pourra filtrer par protocole YPSERV). Le mot de passe de l'utilisateur passe-t-il lors de la demande de connexion ? Est-il en clair ?
8. Que signifie le terme **bind** en anglais ?
9. Essayer d'éteindre le serveur (ou de le débrancher du réseau) et de se connecter sur le client à l'aide d'un compte NIS. Que se passe-t-il ?
10. Quel peut être à votre avis le risque de ne pas mettre le mot clef **files** dans la configuration de nsswitch ? Penser au cas où le serveur est éteint et où le client se connecte.

3. Remise en état des postes

Une fois le TP terminer il faut remettre les machines en l'état, c'est-à-dire supprimer les installation réalisées. On utilisera la commande `apt autoremove --purge nis`.

FIN