
CPE Lyon - 4ETI & 3IRC - Année 2018/19

Administration Système

TP 4 - Utilisateurs, groupes et permissions



Dans ce quatrième TP, nous allons voir comment gérer les utilisateurs, les groupes et les permissions sur notre système, ainsi que les quotas disque.

Dans votre compte-rendu de TP, vous n'oublierez pas d'indiquer, en plus des réponses aux questions, les **commandes** qui vont ont permis d'obtenir ces réponses. Essayez de donner les commandes qui répondent **exactement** à la question posée (càd qui n'affichent pas d'autres informations que celle attendue).

Exercice 1. Gestion des utilisateurs et des groupes

- Commencez par créer ensuite deux groupes **groupe1** et **groupe2**
- Créez ensuite 4 utilisateurs **u1**, **u2**, **u3**, **u4** avec leur dossier "home" et ayant **bash** pour shell
- Placez les utilisateurs dans les groupes :
 - **u1**, **u2**, **u4** dans **groupe1**
 - **u2**, **u3**, **u4** dans **groupe2**
- Donnez deux moyens d'afficher les membres de **groupe2**
- Faites de **groupe1** le groupe propriétaire de **/home/u1** et **/home/u2** et de **groupe2** le groupe propriétaire de **/home/u3** et **/home/u4**
- Remplacez le groupe primaire des utilisateurs :
 - **groupe1** pour **u1** et **u2**
 - **groupe2** pour **u3** et **u4**
- Créez deux répertoires **/home/groupe1** et **/home/groupe2** pour le contenu commun aux groupes, et mettez en place les permissions permettant aux membres de chaque groupe d'écrire dans le dossier associé.
- Comment faire pour que, dans ces dossiers, seul le propriétaire d'un fichier ait le droit de renommer ou supprimer ce fichier ?
- Pouvez-vous vous connecter en tant que **u1** ? Pourquoi ?
- Activez le compte de l'utilisateur **u1** et vérifiez que vous pouvez désormais vous connecter avec son compte.
- Quels sont l'uid et le gid de **u1** ?
- Quel utilisateur a pour uid 1003 ?
- Quel est l'id du groupe **groupe1** ?

- Quel groupe a pour `guid` 1002 ? (▲ Rien n'empêche d'avoir un groupe dont le **nom** serait 1002...)
- Retirez l'utilisateur `u3` du groupe `groupe2`. Que se passe-t-il ? Expliquez.
- Modifiez le compte de `u4` de sorte que :
 - il expire au **1^{er} juin 2019**
 - il faut changer de mot de passe avant 90 jours
 - il faut attendre 5 jours pour modifier un mot de passe
 - l'utilisateur est averti 14 jours avant l'expiration de son mot de passe
 - le compte sera bloqué 30 jours après expiration du mot de passe
- Quel est l'interpréteur de commandes (Shell) de l'utilisateur `root` ?
- à quoi correspond l'utilisateur `nobody` ?
- Par défaut, combien de temps la commande `sudo` conserve-t-elle votre mot de passe en mémoire ? Quelle commande permet de forcer `sudo` à oublier votre mot de passe ?

Exercice 2. Gestion des permissions

- Dans votre `$HOME`, créez un dossier `test`, et dans ce dossier un fichier `fichier1` contenant quelques lignes de texte. Quels sont les droits sur `test` et `fichier1` ?
- Retirez tous les droits sur ce fichier (même pour vous), puis essayez de le modifier et de l'afficher en tant que `root`. Conclusion ?
- Redonnez vous les droits en écriture et exécution sur `fichier` puis exécutez la commande `echo "echo Hello" > fichier`. On a vu lors des TP précédents que cette commande remplace le contenu d'un fichier s'il existe déjà. Que peut-on dire au sujet des droits ?
- Essayez d'exécuter le fichier. Est-ce que cela fonctionne ? Et avec `sudo` ? Expliquez.
- Placez-vous dans le répertoire `test`, et retirez-vous le droit en lecture pour ce répertoire. Listez le contenu du répertoire, puis exécutez ou affichez le contenu du fichier `essai`. Qu'en déduisez-vous ? Rétablissez le droit en lecture sur `test`
- Créez dans `test` un fichier `nouveau` ainsi qu'un répertoire `sstest`. Retirez au fichier `nouveau` et au répertoire `test` le droit en écriture. Tentez de modifier le fichier `nouveau`. Rétablissez ensuite le droit en écriture au répertoire `test`. Tentez de modifier le fichier `nouveau`, puis de le supprimer. Que pouvez-vous déduire de toutes ces manipulations ?
- Positionnez vous dans votre répertoire personnel, puis retirez le droit en exécution du répertoire `test`. Tentez de créer, supprimer, ou modifier un fichier dans le répertoire `test`, de vous y déplacer, d'en lister le contenu, etc...Qu'en déduisez vous quant au sens du droit en exécution pour les répertoires ?
- Rétablissez le droit en exécution du répertoire `test`. Positionnez vous dans ce répertoire et retirez lui à nouveau le droit d'exécution. Essayez de créer, supprimer et modifier un fichier dans le répertoire `test`, de vous déplacer dans `ssrep`, de lister son contenu. Qu'en concluez-vous quant à l'influence des droits que l'on possède sur le répertoire courant ? Peut-on retourner dans le répertoire parent avec `"cd .."` ? Pouvez-vous donner une explication ?

- Rétablissez le droit en exécution du répertoire `test`. Attribuez au fichier `essai` les droits suffisants pour qu'une autre personne de votre groupe puisse y accéder en lecture, mais pas en écriture.
- Définissez un umask très restrictif qui interdit à quiconque à part vous l'accès en lecture ou en écriture, ainsi que la traversée de vos répertoires. Testez sur un nouveau fichier et un nouveau répertoire.
- Définissez un umask très permissif qui autorise tout le monde à lire vos fichiers et traverser vos répertoires, mais n'autorise que vous à écrire. Testez sur un nouveau fichier et un nouveau répertoire.
- Définissez un umask équilibré qui vous autorise un accès complet et autorise un accès en lecture aux membres de votre groupe. Testez sur un nouveau fichier et un nouveau répertoire.
- Transcrivez les commandes suivantes de la notation classique à la notation octale ou vice-versa (vous pourrez vous aider de la commande `stat` pour valider vos réponses) :
 - `chmod u=rx,g=wx,o=r fic`
 - `chmod uo+w,g-rx fic` en sachant que les droits initiaux de `fic` sont `r--r-x---`
 - `chmod 653 fic` en sachant que les droits initiaux de `fic` sont `711`
 - `chmod u+x,g=w,o-r fic` en sachant que les droits initiaux de `fic` sont `r--r-x---`
- Affichez les droits sur le programme `passwd`. Que remarquez-vous ? En affichant les droits du fichier `/etc/passwd`, pouvez-vous justifier les permissions sur le programme `passwd` ?
- **Access Control Lists (ACL)** : suivez le tutoriel de cette page : <https://doc.ubuntu-fr.org/acl>.
- **Quotas disques** : suivez le tutoriel de cette page : <https://doc.ubuntu-fr.org/quota>.