

# Pré-requis

Pour valider les pré-requis nécessaires, avant d'aborder le TP, répondez aux questions ci-après :

- 1. Citer les différents type de comptes utilisateurs sous Unix.
- 2. Dans quel fichier sont renseignés les groupes secondaires d'un compte utilisateur ?
- 3. Comment ajouter le compte utilisateur **nicolas** sur le système sachant que le groupe formateur existant doit être son groupe principal ?

a.

```
useradd -g formateur nicolas
```

b.

```
useradd nicolas ; groupadd nicolas formateur
```

c.

```
useradd -G formateur nicolas
```

d.

```
useradd nicolas ; usermod -G formateur nicolas
```

4. Que fait la commande ?

```
find / -user 502
```

- a. Elle recherche le mot de passe de l'utilisateur ayant l'UID égal à 502.
  - b. Elle affiche le nom de tous les fichiers appartenant à l'utilisateur ayant l'UID égal à 502 dans l'arborescence du système.
  - c. elle retourne le nom de l'utilisateur ayant l'UID égal à 502.
  - d. Elle affiche le nom des fichiers ayant les droits positionnés à 502 en notation octale.
5. Quel droit signifie qu'un programme doit être lancé avec l'identifiant du propriétaire du fichier ?
- a. x
  - b. SUID
  - c. SGID
  - d. bit "sticky"

## Exercice 1 : Utilisateurs et groupes

- 1. Se connecter en tant que root sur une console texte.
- 2. Lister les comptes utilisateurs et les groupes existants sur le système.
- 3. Quels sont l'UID et le GID du compte **root** ?
- 4. Quelles sont les valeurs minimales des UID et GID utilisées par défaut lors de la création de nouveaux comptes utilisateurs ou groupes ? Faire de sorte que L'UID minimal soit 700 et que le GID minimal soit 800.
- 5. Créer les groupes **grp1**, **grp2** et **grp3**. Le GID du groupe **grp3** doit être 823.
- 6. Créer les comptes utilisateurs **util1**, **util2**, et **util3** ayant respectivement les groupes **grp1**, **grp2** et **grp3** comme groupe principal. Le compte **util2** doit aussi être membre des groupe **grp1** et **grp3**. Le pseudonyme **tux1** de l'utilisteur **util1** doit être renseigné dans la configuration.
- 7. Noter les UID et GID des comptes utilisateurs et ds groupes créés précédement.
- 8. Supprimer le groupe **grp3**. Cela est-il possible ? Pourquoi ?
- 9. Supprimer le compte utilisateur **util3** sans supprimer son répertoire personnel et supprimer le groupe **grp3**.
- 10. A qui appartient maintenant le répertoire **/home/util3** ? Pourquoi ?
- 11. Comment retrouver et supprimer tous les fichiers orphelins du système qui appartenaient au compte utilisateur **util3** ou au groupe **grp3** ?

## Exercice 2 : Droits d'accès

- 1. En tant que **util1**, créer le répertoire **rep1** dans **/tmp** puis créer le fichier vide **fic1** dans ce répertoire. Quels sont les droits associés à ces nouveaux fichiers ?
- 2. Modifier la valeur de umask de **util1** à 007.
- 3. Créer un nouveau répertoire **rep2** contenant le fichier vide **fic2** dans **/tmp**, et regarder les droits associés à ces nouveaux fichiers. Cela a-t-il changé les droits des fichiers déjà existant ?
- 4. Changer les droits du fichiers **fic2** afin que personne ne puisse le modifier.
- 5. Se connecter en tant que **util2** sur une console texte libre.
- 6. Essayer de supprimer le fichier **/tmp/rep2/fic2**. Est-ce possible ? Pourquoi ?

7. En tant que **root**, créer le programme C /tmp/droits.c suivant :

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

main()
{
    int uid = getuid();
    int euid = geteuid();
    int gid = getgid();
    int egid = getegid();
    printf("UID réel : %u\n", uid);
    printf("UID effectif : %u\n", euid);
    printf("GID réel : %u\n", gid);
    printf("GID effectif : %u\n", egid);
}
```

8. Compiler le code source précédent avec la commande :

```
gcc droits.c -o droits.exe
```

9. Exécuter successivement le programme droits.exe en tant que **root** et **util1**. Quel est le résultat affiché ? Qu'en déduire ?

10. Positionner le droit SUID sur ce fichier et exécuter de nouveau ce programme en tant que **root** et **util1**. Quelle est la différence ?

11. Changer le propriétaire et le groupe du fichier /tmp/droits.exe en **util2** et **grp2**.

12. Positionner le droit SGID sur ce fichier et exécuter de nouveau ce programme en tant que *root* et **util1**.

13. Modifier de nouveau le propriétaire et le groupe du fichier /tmp/droits.exe en **root** et **root**.

14. Positionner les droits d'accès à 0755 sur le fichier /tmp/droits.exe.

15 Ajouter la ligne suivante au fichier /etc/sudoers :

Ne pas éditer le fichier, mais utiliser la commande visudo

```
util1 ALL=(root) /tmp/droits.exe
```

16. Exécuter la commande :

```
sudo /tmp/droits.exe
```

successivement en tant que **util1** et **util2**. Est-ce possible ? Si oui, sous quelle identité ?

## Exercice 3 : Pour aller plus loin

1. Mettre en place un répertoire de travail commun pour les membres du **grp1** ; tout nouveau fichier créé à l'intérieur de celui-ci devra appartenir à ce groupe.