

TP n°2

Droits d'accès

Pour cet exercice, vous créerez un utilisateur banalisé

- Vous est-il possible de faire une copie du fichier `/etc/passwd`? Vous est-il possible de supprimer ou de modifier le fichier `/etc/passwd`? Répéter ces tentatives en tant qu'utilisateur root. Expliquer la situation à l'aide de la commande `ls -l`.
 - A l'aide de la commande `id`, vérifier votre identité et le(s) groupe(s) au(x)quel(s) vous appartenez.
 - Créer un petit fichier texte (de contenu quelconque), qui soit lisible par tout le monde, mais non modifiable (même pas par vous).
 - Créer un répertoire nommé "**Secret**", dont le contenu est visible uniquement par vous même. Les fichiers placés dans ce répertoire sont-ils lisibles par d'autres membres de votre groupe?
 - Créer un répertoire nommé "**Connaisseurs**" tel que les autres utilisateurs ne puissent pas lister son contenu mais puissent lire les fichiers qui y sont placés.
- On obtiendra:

```
ls Connaisseurs
```

```
ls : Connaisseurs: Permission denied
```

```
cat Connaisseurs/toto
```

```
<...le contenu du fichier toto (s'il existe)...>
```

- Chercher dans le répertoire `/usr/bin` des exemples de commandes ayant la permission **SUID**. De quelle genre de commande s'agit-il?

Les utilisateurs

- Votre compte d'utilisateur est-il défini dans le fichier `/etc/passwd`? Pourquoi? Il y a-t-il d'autres alternatives?
- Quel est le répertoire de connexion de l'utilisateur root?
- Quel est le shell de l'utilisateur root?
- Quelle est la particularité de l'utilisateur **nobody**? Et de l'utilisateur **shutdown**?
- Quels sont les utilisateurs définis dans `/etc/passwd` qui font partie du même groupe que l'administrateur?

Redirections, méta-caractères

Le répertoire `/usr/include` contient les fichiers d'entête standards en langage C (`stdlib.h`, ...).

- Créer un répertoire nommé `inc` dans votre répertoire de connexion (`HOME`). En utilisant une seule commande, y copier les fichiers du répertoire `/usr/include` dont le nom commence par `std`.
- Afficher la liste des fichiers de `/usr/include` dont le nom commence par `a`, `b` ou `c`.
- Modifier la commande de la question précédente pour qu'au lieu d'afficher le résultat, celui-ci soit placé dans un fichier nommé `"Abc.list"` de votre répertoire de connexion.

- Afficher le contenu de ce fichier en utilisant la commande `cat`.
Copier avec `cat` son contenu dans un nouveau fichier nommé "Copie".
- Toujours avec `cat`, créer un nouveau fichier nommé "Double" formé par la mise bout à bout (concaténation) des fichiers "Abc.list" et "Copie".
Vérifier que le nombre de lignes a bien doublé à l'aide de la commande `wc`.
- Créer un fichier nommé "Temp" contenant une ligne de texte.
- Avec `cat`, ajouter la ligne "*The end*" à la fin du fichier "Temp".
- En une seule ligne de commande, faire afficher le nombre de fichiers de `/usr/include` dont le nom contient la lettre `t`.

find

- Afficher la liste des fichiers `.h` situés sous le répertoire `/usr/include`.
- Afficher la liste des fichiers plus vieux que 3 jours situés sous votre répertoire de connexion.

head, tail, tubes

- Afficher les 5 premières, puis les 5 dernières lignes du fichier `/etc/passwd`.
- Afficher la 7ième ligne de ce fichier (et elle seule), en une seule ligne de commande.

tr, more, tubes

- Afficher le fichier `/etc/passwd` en remplaçant les caractères `/` par des `X`.
- Obtenir le résultat précédent page par page.

Contrôles d'accès

Un administrateur désire s'assurer chaque matin que tous les fichiers placés sous `$REPERTOIRE` sont lisibles par tout le monde, mais non modifiables excepté par leur propriétaire.

- Quel doit être le mode de ces fichiers et répertoires?
- Écrire un script permettant d'afficher la liste des fichiers d'un répertoire et le type de chacun (utilisation de `file` et `sort`) trié par type.

Flux d'E/S standards

Étudier la documentation de la commande `tee`.

- Écrire une commande qui affiche "*Bonjour*" à l'écran et en même temps crée un fichier nommé "Salutation.txt" dont le contenu est le même message.
- Écrire un script qui lit une ligne de caractères sur son entrée standard et l'écrit sur sa sortie, en passant tous les caractères en majuscules.
- Modifiez votre script pour qu'il affiche le nombre de caractères traités sur sa sortie d'erreur.
- Créer un fichier texte quelconque. Écrire une seule commande shell qui recopie le contenu passé en majuscule de ce fichier dans un nouveau fichier nommé "Titre.txt" et affiche à l'écran le nombre de caractères traités.

grep, cut, uniq, sort et tubes

Étudier la documentation des commandes `cut` et `uniq`.

- Afficher la liste des répertoires de connexion des utilisateurs déclarés dans le fichier `/etc/passwd`.

- On rappelle qu'à chaque utilisateur est associé un interpréteur de commandes (shell) lancé lors de son login. La commande correspondante est indiquée dans le 7ième champ du fichier /etc/passwd.

Afficher en une ligne de commande le *nombre* d'interpréteurs de commandes différents mentionnés dans /etc/passwd.

- On dispose d'un fichier texte [telephone.txt](#) contenant un petit carnet d'adresses. Chaque ligne est de la forme "nom prenom numerotelephone". Les champs sont séparés par des tabulations.

Répondre aux questions suivantes en utilisant à chaque fois une ligne de commande shell:

- Afficher le carnet d'adresse trié par ordre alphabétique de noms.
- Afficher le nombre de personnes dans le répertoire.
- Afficher toutes les lignes concernant les "Dupond".
- Afficher toutes les lignes ne concernant pas les "Dupond".
- Afficher le numéro de téléphone (sans le nom) du premier "Dupond" apparaissant dans le répertoire.
- Afficher le numéro de téléphone (sans le nom) du premier "Dupond" dans l'ordre alphabétique (ordre basé sur les prénoms).