CLI niveau 3 : Gestion des permissions et droits.

🎯 Objectif : comprendre l'attribution des droits et la modification des permissions pour comprendre la sécurité des fichiers sous Linux.

Table des matières

1. Cours : Gestion des permissions et droits sous Linux....1
a. Les permissions sous Linux....1
b. Modifier les permissions....2
c. Changer le propriétaire ou groupe attaché à un fichier/dossier....2
d. Vérifier le nom de l'utilisateur courant....2
e. Exécuter en admin (root) une commande ; seul root a tous les droits....2
f. Changer son mot de passe....2
2. TP Découverte : Gestion des permissions....3
Objectif général :....3
Mise en place....3
Partie A : Observation et compréhension....3
Partie B : Manipulation des droits....3
Exercice 1 : Ajouter des droits d'exécution....3
Exercice 2 : Retirer des droits....3
Exercice 3 : Utiliser la notation numérique....3
Partie C : Propriétaires et groupes....4
Exercice 4 : Changer de propriétaire....4
Partie D : Commandes système liées....4
Exercice 5 : Savoir qui on est....4
Exercice 6 : Modifier son mot de passe....4
3. Série d'exercices supplémentaires (progressifs)....4
Exercice 7 : Script pratique (Premier script!)....4
Exercice 8 : Gestion complète....4
Exercice 9 : Challenge....4
Exercice 10 : Simulation admin....4

1. Cours : Gestion des permissions et droits sous Linux

a. Les permissions sous Linux

- Les permissions sur un fichier/dossier sont de trois types :
  - **r** (read) → autorisation de lecture
  - **w** (write) → autorisation d'écriture
  - **x** (execute) → autorisation d'exécution
- Les permissions sont en général distincts en fonction de l'utilisateur :
  - **u** (user) → propriétaire (si l'utilisateur est le propriétaire du fichier)
  - **g** (group) → groupe (si l'utilisateur appartient à ce groupe)
  - **o** (others) → autres (si l'utilisateur n'est ni propriétaire, ni membre du groupe)

remarque essentielle : l'utilisateur root a tous les droits sur tous les éléments du système.

Exemple d'un `ls -l` :

-rw-r--r-- 1 phil users 1234 août 25 10:20 fichier.txt

- `-rw-r--r--` → type + permissions
- `1` → nombre de liens (nombre de fichiers qui pointent vers le même contenu)
- `phil` → propriétaire
- `users` → groupe
- `1234` → taille en octets
- `août 25 10:20` → date de dernière modification
- `fichier.txt` → nom

---

b. Modifier les permissions

- **chmod** (symbolique) : on utilise des symboles pour modifier les permissions

chmod u+x fichier.txt # ajoute exécution au propriétaire

chmod g-w fichier.txt # retire écriture au groupe

- **chmod** (numérique) : on utilise un codage numérique pour modifier les permissions

chmod 755 fichier.txt # rwxr-xr-x

chmod 644 fichier.txt # rw-r--r--

Table : r=4, w=2, x=1 → somme par catégorie.

---

c. Changer le propriétaire ou groupe attaché à un fichier/dossier

sudo chown nouvel_utilisateur fichier.txt

sudo chown user:group fichier.txt

---

d. Vérifier le nom de l'utilisateur courant

whoami

---

e. Exécuter en admin (root) une commande ; seul root a tous les droits

sudo commande

---

f. Changer son mot de passe

passwd

2. TP Découverte : Gestion des permissions

Objectif général :

Expérimenter la création de fichiers, l'attribution des droits et la modification des permissions pour comprendre la sécurité des fichiers.

Mise en place

Créer un dossier de travail :
mkdir ~/tp_linux && cd ~/tp_linux
mkdir ./tp_permissions && cd ./tp_permissions
touch fichier1.txt fichier2.sh fichier3.conf
`mkdir DossierBidon`

Partie A : Observation et compréhension

1. Dans le dossier tp_permissionss, affiche les permissions des fichiers avec la commande `ls -l`
2. Note la différence entre `-rw-r--r--` et `drwxr-xr-x`.
   Peux-tu expliquer la signification de ces lettres ?

Partie B : Manipulation des droits

Exercice 1 : Ajouter des droits d'exécution

- Ajoute le droit d'exécution pour **l'utilisateur** sur `fichier2.sh` :

chmod u+x fichier2.sh

- Vérifie avec `ls -l`.

Exercice 2 : Retirer des droits

- Retire le droit d'écriture pour les autres sur `fichier1.txt` :

chmod o-w fichier1.txt

- Vérifie avec `ls -l`.

Exercice 3 : Utiliser la notation numérique

- Passe `fichier3.conf` en `rw-r-----` :

chmod 640 fichier3.conf

- Vérifie le résultat.

Partie C : Propriétaires et groupes

Exercice 4 : Changer de propriétaire

- Simule le changement (avec sudo si nécessaire) :

sudo chown root fichier1.txt

- Vérifie avec `ls -l`.

Partie D : Commandes système liées

Exercice 5 : Savoir qui on est

whoami

Exercice 6 : Modifier son mot de passe

passwd

(Ne le fais que si c'est ton système perso pour tester mais rétablis ensuite le mot de passe d'origine)

3. Série d'exercices supplémentaires (progressifs)

Exercice 7 : Script pratique (Premier script!)

Crée un fichier `mon_script.sh` avec pour contenu les deux lignes suivantes :

#!/bin/bash

echo "Bonjour $(whoami)"

1. Donne ensuite le droit d'exécution à ce fichier.
2. Exécute-le avec la commande `./mon_script.sh`.

Exercice 8 : Gestion complète

1. Crée `secret.txt` et mets `r--` pour toi, rien pour les autres.
2. Crée `public.txt` avec tous les droits pour tout le monde.
3. Vérifie avec `ls -l`.

Exercice 9 : Challenge

Crée un fichier `challenge.txt`

- Donne **rwx** au propriétaire, **rx** au groupe, **r** aux autres avec une seule commande numérique.

Exercice 10 : Simulation admin

1. Crée un fichier `admin.txt`.
2. Change son propriétaire pour `root` (avec sudo).
3. Vérifie que seul root peut le modifier.
