



## INF2132 : SYSTÈMES D'EXPLOITATION

---

### TP7 - Gestion des Permissions et des Utilisateurs sous Linux

---

*Nom de l'enseignant :*  
Pr. Ilias Tougui

*Nom de l'assistant :*  
Pr. Yasser Aderghal

## Table des matières

<b>I</b>	<b>Objectifs du TP</b>	<b>2</b>
<b>II</b>	<b>Prérequis</b>	<b>2</b>
<b>1</b>	<b>Préparation de l'Environnement Linux</b>	<b>3</b>
<b>2</b>	<b>Exécution du script et analyse des utilisateurs</b>	<b>4</b>
<b>3</b>	<b>Analyse approfondie des permissions</b>	<b>5</b>
<b>A</b>	<b>Annexe</b>	<b>7</b>

Lisez attentivement cette page

## I Objectifs du TP

À la fin de ce TP, vous serez capable de :

- Comprendre et utiliser les commandes de base Linux de gestion des utilisateurs et groupes (`useradd`, `usermod`, `groupadd`)
- Configurer les permissions des fichiers et répertoires en mode symbolique et octal
- Analyser les risques de sécurité liés aux permissions inadéquates
- Démontrer la compréhension du système de permissions Linux et du rôle du `umask`
- Appliquer le principe du moindre privilège dans la configuration des accès
- Analyser et corriger les vulnérabilités de configuration système

## II Prérequis

Avant de commencer ce TP, vous devez :

- Maîtriser les commandes de base Linux : `cd`, `ls`, `mkdir`, `touch`, `cat`
- Comprendre les concepts de fichiers et répertoires
- Connaître les bases du shell bash
- Consulter **l'Annexe** pour les références sur les permissions Linux, et pour le tableau de conversion octal/symbolique

# 1 Préparation de l'Environnement Linux

Durée : 20 min, Note : 5 points

En utilisant les commandes de base de Linux, suivez les consignes ci-dessous pour créer l'environnement de travail structuré de ce TP.

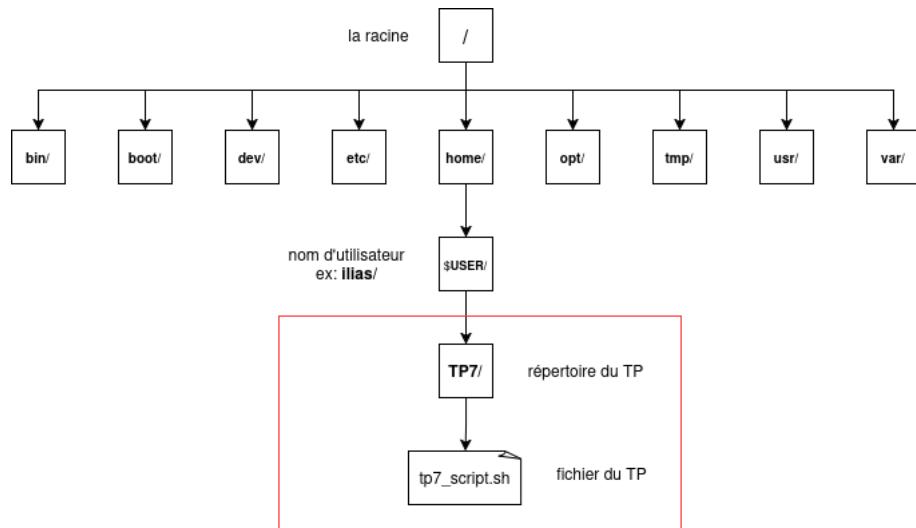


FIGURE 1 – Structure de l'environnement de travail.

1. Déplacez-vous dans le répertoire racine.
2. Déplacez-vous ensuite dans votre répertoire personnel.
3. Créez un nouveau répertoire nommé TP7.
4. Accédez au répertoire du TP.
5. Créez un fichier de script nommé `tp7_script.sh`.
6. Vérifiez la création de `tp7_script.sh` dans le répertoire TP7.
7. Remplissez `tp7_script.sh` avec son contenu fourni sur Connect UIR.
8. Affichez le contenu de `tp7_script.sh` page par page dans le terminal.
9. En utilisant le `pipe`, affichez et comptez le nombre de lignes dans `tp7_script.sh`.
10. Cherchez et affichez toutes les lignes contenant le mot "echo" dans `tp7_script.sh`.
11. En utilisant le `pipe`, cherchez et comptez combien de fois le caractère "#" apparaît dans `tp7_script.sh` (nombre de lignes de commentaires).
12. Utilisez `grep` avec `redirection` pour créer un deuxième fichier nommé `unclear_tp7_script.sh` contenant uniquement les lignes de `tp7_script.sh` qui ne sont pas des commentaires.
13. Affichez et remplacez toutes les occurrences du mot "echo" par "printf" dans `tp7_script.sh`.
14. Affichez et supprimez toutes les lignes vides dans `tp7_script.sh`.
15. Lisez attentivement le contenu de `tp7_script.sh`.
16. Que font les commandes : `useradd`, `usermod`, `groupadd`, `chown` et `chmod` ?
17. **Appelez le professeur pour montrer le travail accompli.**

## 2 Exécution du script et analyse des utilisateurs

Durée : 20 min, Note : 5 points

1. Déplacez-vous dans le répertoire TP7.
2. Essayez d'exécuter le script `tp7_script.sh` avec la commande `./tp7_script.sh`.
3. Analysez l'erreur affichée. Pourquoi le script ne s'exécute-t-il pas ?
4. Corrigez le problème en modifiant les permissions du script pour le rendre exécutable par le propriétaire.
5. Essayez à nouveau d'exécuter le script `./tp7_script.sh`.
6. Analysez la nouvelle erreur affichée. Pourquoi le script échoue-t-il maintenant ?
7. Exécutez le script avec les priviléges appropriés.
8. Affichez la première et les 2 dernières lignes du fichier `/etc/passwd`.
9. Exécutez la commande suivante et expliquez ce qu'elle fait :

```
cat /etc/passwd | grep -E "\$USER|amal|amine|root"
```

Dans votre explication, détaillez :

- \* Le rôle de `cat /etc/passwd`
  - \* Le rôle du pipe `|`
  - \* Le rôle de `grep -E`
  - \* Le rôle de l'option `-E`
  - \* La signification de `"$USER|amal|amine|root"`
  - \* Le symbole `|` dans l'expression régulière
  - \* Que contient `$USER` ?
10. Expliquez les champs dans la ligne suivante extraite de `/etc/passwd` :  
`amine:x:1002:1002::/home/amine:/bin/bash`
  11. Affichez les 3 dernières lignes du fichier `/etc/group`.
  12. Quelle est la différence entre les deux fichiers `/etc/passwd` et `/etc/group` ?
  13. **Appelez le professeur pour lui expliquer votre compréhension**

### 3 Analyse approfondie des permissions

Durée : 20 min, Note : 5 points

1. Listez tout le contenu du répertoire TP7 de manière récursive avec les permissions détaillées.
2. En remplissant le **tableau ci-dessous** (sur un papier), analysez chaque élément en indiquant :
  - \* Le type (F pour Fichier, D pour Répertoire)
  - \* Le propriétaire et le groupe
  - \* Les permissions en mode symbolique (exemple : `rwxr-xr-x`)
  - \* Une explication détaillée textuelle

Nom	Type	Propriétaire / Groupe	Symbolique	Explication
documents/				
tp7_script.sh				
unclear_tp7_script.sh				
site_web/				
site_web/index.html				

TABLE 1 – Analyse des permissions dans TP7

3. Pour chaque élément du tableau, expliquez la conversion des permissions symboliques en notation octale (**Voir le cours et l'annexe**) :
4. Analysez les permissions par défaut avec le `umask` :
  - \* Affichez votre `umask` actuel avec la commande `umask`
  - \* Expliquez comment le `umask` soustrait des permissions par défaut
5. Pour un `umask` de 0002, calculez les permissions par défaut pour :
  - \* Un nouveau fichier créé avec `touch` (base maximale : 666)
  - \* Un nouveau répertoire créé avec `mkdir` (base maximale : 777)
  - \* Vérifiez vos calculs pour le répertoire TP7 et le fichier `unclear_tp7_script.sh`
  - \* Pourquoi les autres fichiers et répertoires ne correspondent-ils pas à vos calculs ? Justifiez votre réponse en vous appuyant sur le fichier de script.
6. Analysez les répertoires `home` et comparez les permissions :
  - \* Listez les répertoires `home` avec permissions : `ls -l /home/`
  - \* Observez les trois répertoires `home` :
    - `amal` : permissions 755
    - `amine` : permissions 750
    - `$USER` (votre utilisateur) : permissions 750
  - \* Comparez les permissions en mode symbolique pour chaque cas

- \* Pour chaque répertoire, déterminez : qui peut entrer ? Qui peut lire les fichiers à l'intérieur ?
7. Identifiez et expliquez le problème de sécurité :
- \* Quel répertoire home a des permissions différentes des autres ? Lequel est problématique ?
  - \* Testez : en tant qu'utilisateur autre que amal, pouvez-vous accéder au fichier `/home/amal/notes.txt` ?
  - \* Testez : en tant qu'utilisateur autre que amal, pouvez-vous accéder au fichier `/home/amine/` ?
  - \* Quel est le risque exact de cette différence de permissions ?
  - \* Cherchez dans le fichier `tp7_script.sh` la ligne qui cause ce problème
  - \* Comparez cette ligne avec les autres permissions configurées par le script
8. Proposez une solution cohérente :
- \* Pourquoi le script configure-t-il différemment les répertoires home de amal et amine ?
  - \* Expliquez pourquoi les permissions 750 (ou 700) sont plus appropriées que 755
  - \* Proposez la commande `chmod` pour corriger les permissions de `/home/amal`
  - \* Vérifiez votre correction en testant à nouveau l'accès aux fichiers
9. Appelez le professeur pour lui expliquer votre compréhension.

## A Annexe

### Commandes de gestion des utilisateurs

- `useradd -m -s /bin/bash utilisateur` : Crée un nouvel utilisateur avec un répertoire home et le shell bash
- `usermod -a -G groupe utilisateur` : Ajoute un utilisateur à un groupe secondaire
- `groupadd groupe` : Crée un nouveau groupe
- `chown propriétaire:groupe fichier` : Change le propriétaire et le groupe d'un fichier
- `chmod permissions fichier` : Change les permissions d'un fichier ou répertoire

### Commandes grep pour les filtres

- `grep motif fichier` : Affiche les lignes contenant le motif
- `grep -E "motif1|motif2" fichier` : Affiche les lignes contenant motif1 OU motif2 (expressions régulières étendues)
- `grep "^motif" fichier` : Affiche les lignes commençant par le motif
- `grep -v motif fichier` : Affiche les lignes NE contenant PAS le motif (inversion)
- `cat fichier | grep motif` : Combine cat avec grep via un pipe

### Commandes sed pour les transformations

- `sed 's/ancien/nouveau/g' fichier` : Remplace toutes les occurrences de "ancien" par "nouveau"
- `sed '/^$/d' fichier` : Supprime les lignes vides
- `sed -n '3,7p' fichier` : Affiche uniquement les lignes 3 à 7
- `sed 's/motif/remplacement/' fichier` : Remplace (sans le 'g', seule la première occurrence par ligne)

## Tableau de conversion octal/symbolique

Octal	Symbolique	Signification
0	—	Aucun droit
1	-x	Exécution seule
2	-w-	Écriture seule
3	-wx	Écriture et exécution
4	r-	Lecture seule
5	r-x	Lecture et exécution
6	rw-	Lecture et écriture
7	rwx	Lecture, écriture et exécution

TABLE 2 – Correspondance octal/symbolique

## Exemples de permissions complètes :

Octal	Symbolique	Propriétaire	Groupe	Autres
755	rwxr-xr-x	rwx	r-x	r-x
750	rwxr-x—	rwx	r-x	—
700	rwx—	rwx	—	—
644	rw-r-r-	rw-	r-	r-
640	rw-r—	rw-	r-	—
600	rw—	rw-	—	—

TABLE 3 – Exemples de permissions complètes

**Note :** Pour les autres concepts (*umask*, fichiers */etc/passwd*, */etc/group*), consultez votre cours magistral.