

# TD Administration Systèmes et Virtualisation

Module R2.02 - BUT RT Semestre 2

Durée : 10h CM/TD + 18h TP

Objectif : Acquérir les compétences d'administration système Linux

---

## Programme des séances

Séance	Durée	Contenu
TD1	2h	Shell et Gestion des utilisateurs
TD2	2h	Système de fichiers et Permissions
TD3	2h	Processus et Services
TD4	2h	Stockage et Sauvegardes
TD5	2h	Virtualisation et Synthèse

## TD 1 : Shell et Gestion des utilisateurs (2h)

### Partie A : Commandes Shell essentielles (45min)

#### Navigation et fichiers de base

```
# Navigation
pwd                # Afficher répertoire courant
cd /chemin         # Changer de répertoire
cd ~               # Retour home
cd ..              # Répertoire parent
ls -la             # Lister avec détails

# Gestion fichiers/répertoires
mkdir nom_repertoire # Créer répertoire
mkdir -p path/to/dir # Créer arborescence
touch fichier        # Créer fichier vide
cp source dest       # Copier
mv source dest       # Déplacer/renommer
rm fichier           # Supprimer fichier
rm -rf repertoire    # Supprimer répertoire
```

#### Affichage et recherche

```
# Affichage contenu
cat fichier        # Afficher tout
less fichier       # Affichage paginé
head -n 10 fichier # 10 premières lignes
tail -n 10 fichier # 10 dernières lignes

# Recherche
grep "motif" fichier # Chercher dans fichier
find /path -name "*pattern*" # Chercher fichiers
```

#### Redirections et tubes

```
# Redirections
commande > fichier    # Rediriger sortie
commande >> fichier   # Ajouter à la fin
commande 2> errors    # Rediriger erreurs
commande &> tout       # Tout rediriger

# Tubes (pipes)
cmd1 | cmd2           # Sortie cmd1 → entrée cmd2
ps aux | grep apache  # Exemple pratique
```

#### Exercice 1 : Créer l'arborescence suivante

```
~/projet/
├── docs/
├── scripts/
└── backup/
```

Puis créer un fichier readme.txt dans docs/ et le copier dans backup/.

### Partie B : Gestion des utilisateurs (45min)

## Fichiers système importants

```
/etc/passwd    # Informations utilisateurs  
/etc/group     # Informations groupes  
/etc/shadow    # Mots de passe chiffrés
```

## Format /etc/passwd

```
login:x:UID:GID:commentaire:home:shell
```

## Commandes utilisateurs

```
# Création utilisateur  
useradd -u UID -g GROUP -m -s /bin/bash nom_user  
useradd -d /home/custom nom_user # Home personnalisé  
  
# Modification utilisateur  
usermod -a -G groupe_supplementaire user # Ajouter à un groupe  
usermod -c "Nouveau commentaire" user  # Modifier commentaire  
usermod -l nouveau_nom ancien_nom       # Renommer  
  
# Mot de passe  
passwd nom_user      # Définir mot de passe  
passwd -l nom_user   # Verrouiller compte  
passwd -u nom_user   # Déverrouiller compte  
  
# Informations  
id nom_user          # Afficher UID/GID et groupes  
who                  # Utilisateurs connectés  
w                    # Détails des connexions
```

## Gestion des groupes

```
# Création groupe  
groupadd -g GID nom_groupe  
  
# Modification groupe  
groupmod -n nouveau_nom ancien_nom  
  
# Suppression  
userdel nom_user      # Supprimer utilisateur  
groupdel nom_groupe   # Supprimer groupe
```

## Exercice 2 : Créer l'utilisateur alice avec :

- UID : 2001
- Groupe principal : users
- Home : /home/alice
- Shell : /bin/bash
- Mot de passe : test123

## Partie C : Aide et documentation (15min)

## Pages de manuel

```
man commande      # Manuel de la commande
man 5 passwd       # Section 5 (fichiers)
apropos mot_clé    # Rechercher dans les manuels
whatis commande    # Description courte
```

### Navigation dans man

- Espace : page suivante
- b : page précédente
- /motif : rechercher
- q : quitter

Exercice 3 : Trouver l'emplacement de l'UID dans le fichier passwd en utilisant le manuel.

## TD 2 : Système de fichiers et Permissions (2h)

### Partie A : Arborescence Linux (30min)

#### Structure des répertoires principaux

```
/          # Racine du système
├── bin/    # Commandes essentielles
├── etc/    # Fichiers de configuration
├── home/   # Répertoires utilisateurs
├── usr/    # Programmes utilisateur
├── var/    # Données variables (logs, caches)
├── tmp/    # Fichiers temporaires
├── dev/    # Fichiers de périphériques
└── mnt/    # Points de montage
```

#### Types de liens

```
# Lien physique (hard link)
ln fichier_source lien_physique

# Lien symbolique (soft link)
ln -s chemin_cible lien_symbolique
ln -s ../dossier/fichier lien_relatif
```

Exercice 4 : Créer un fichier data.txt, puis créer :

- Un lien physique data\_hard
- Un lien symbolique absolu data\_sym\_abs
- Un lien symbolique relatif data\_sym\_rel

Observer les différences avec `ls -li`.

### Partie B : Système de permissions (75min)

#### Format des permissions

```
rwxrwxrwx
|||└─ autres (other)
||└─ groupe (group)
|└─ propriétaire (user)
└─ type de fichier
```

#### Signification des permissions

Fichier		Répertoire
r	Lire le contenu	Lister le contenu
w	Modifier le contenu	Créer/supprimer fichiers
x	Exécuter	Traverser (cd)

## Notation octale

Binaire	Octal	Permissions
000	0	---
001	1	--x
010	2	-w-
011	3	-wx
100	4	r--
101	5	r-x
110	6	rw-
111	7	rwX

## Commandes de gestion

```
# Modifier permissions
chmod 755 fichier          # rwxr-xr-x
chmod u+x fichier          # Ajouter exécution au propriétaire
chmod g-w fichier          # Retirer écriture au groupe
chmod o=r fichier          # Autres = lecture seule

# Modifier propriétaire
chown user fichier         # Changer propriétaire
chown user:group fichier   # Changer propriétaire et groupe
chgrp group fichier        # Changer groupe seulement

# Récursif
chmod -R 755 repertoire    # Appliquer récursivement
chown -R user:group rep    # Changer propriétaire récursivement
```

Exercice 5 : Créer un répertoire test\_perms/ contenant :

- public.txt : lecture pour tous (644)
- private.txt : lecture/écriture propriétaire seul (600)
- executable.sh : exécutable pour propriétaire et groupe (750)

## Droits spéciaux

```
# SUID (Set User ID) - bit 4
chmod 4755 fichier          # rwsr-xr-x
# S'exécute avec les droits du propriétaire

# SGID (Set Group ID) - bit 2
chmod 2755 repertoire       # rwxr-sr-x
# Fichiers créés héritent du groupe du répertoire

# Sticky bit - bit 1
chmod 1755 repertoire       # rwxr-xr-t
# Seul le propriétaire peut supprimer ses fichiers
```

Exercice 6 : Examiner les permissions de /usr/bin/passwd et expliquer pourquoi il a le bit SUID.

## TD 3 : Processus et Services (2h)

### Partie A : Gestion des processus (60min)

#### Affichage des processus

```
ps                # Processus de la session courante
ps aux            # Tous les processus (format BSD)
ps -ef            # Tous les processus (format System V)
top               # Affichage dynamique
htop              # Version améliorée de top
```

#### Colonnes importantes de ps aux

- USER : propriétaire du processus
- PID : Process ID (identifiant unique)
- PPID : Parent Process ID
- %CPU : pourcentage CPU utilisé
- %MEM : pourcentage mémoire utilisée
- COMMAND : commande exécutée

#### Contrôle des processus

```
# Signaux importants
kill PID          # SIGTERM (15) - arrêt propre
kill -9 PID       # SIGKILL (9) - arrêt forcé
kill -STOP PID    # SIGSTOP - suspension
kill -CONT PID    # SIGCONT - reprendre

# Par nom de processus
killall nom_processus # Tuer tous les processus du nom
pkill pattern         # Tuer selon un motif
```

#### Gestion des jobs

```
commande &        # Lancer en arrière-plan
jobs              # Lister les jobs
fg %1             # Ramener job 1 au premier plan
bg %1             # Remettre job 1 en arrière-plan
Ctrl+Z           # Suspendre processus courant
Ctrl+C           # Arrêter processus courant

# Processus détaché
nohup commande & # Continue après déconnexion
```

#### Exercice 7 :

1. Lancer `sleep 300 &` en arrière-plan
2. Vérifier avec `jobs`
3. Le ramener au premier plan
4. Le suspendre avec `Ctrl+Z`
5. Le remettre en arrière-plan

## 6. Le tuer

### Partie B : Services système (60min)

#### Gestion avec systemd

```
# État d'un service
systemctl status nom_service
systemctl is-active nom_service
systemctl is-enabled nom_service

# Contrôle des services
systemctl start nom_service      # Démarrer
systemctl stop nom_service       # Arrêter
systemctl restart nom_service    # Redémarrer
systemctl reload nom_service     # Recharger config

# Démarrage automatique
systemctl enable nom_service     # Activer au boot
systemctl disable nom_service    # Désactiver au boot

# Lister services
systemctl list-units --type=service
systemctl list-units --state=running
```

#### Gestion des logs

```
# Logs système avec journalctl
journalctl                # Tous les logs
journalctl -u nom_service # Logs d'un service
journalctl -f             # Suivi en temps réel
journalctl --since "1 hour ago" # Depuis 1 heure
journalctl -p err         # Erreurs seulement

# Logs traditionnels
tail -f /var/log/syslog      # Suivi du syslog
tail -f /var/log/apache2/error.log # Logs Apache
```

#### Exercice 8 : Installer et configurer le service SSH :

1. Installer openssh-server
2. Vérifier son statut
3. L'activer au démarrage
4. Consulter ses logs



## TD 4 : Stockage et Sauvegardes (2h)

### Partie A : Gestion des disques (60min)

#### Commandes de visualisation

```
# Vue d'ensemble
lsblk                # Arborescence des périphériques
df -h                # Espace utilisé par filesystem
du -sh /chemin       # Espace utilisé par répertoire
fdisk -l             # Partitions des disques

# Informations détaillées
blkid                # UUID et types de filesystems
mount                # Points de montage actifs
findmnt              # Arborescence des montages
```

#### Partitionnement avec fdisk

```
sudo fdisk /dev/sdX   # Ouvrir l'outil de partitionnement

# Commandes dans fdisk :
# n - nouvelle partition
# p - partition primaire
# e - partition étendue
# d - supprimer partition
# w - écrire les modifications
# q - quitter sans sauvegarder
```

#### Formatage et montage

```
# Formatage
sudo mkfs.ext4 /dev/sdX1    # Formater en ext4
sudo mkfs.ntfs /dev/sdX1    # Formater en NTFS
sudo mkfs.vfat /dev/sdX1    # Formater en FAT32

# Montage manuel
sudo mkdir /mnt/mon_disque   # Créer point de montage
sudo mount /dev/sdX1 /mnt/mon_disque # Monter
sudo umount /mnt/mon_disque  # Démonter

# Montage automatique (/etc/fstab)
/dev/sdX1 /mnt/data ext4 defaults 0 2
```

Exercice 9 : Créer un fichier "disque" virtuel de 100MB, le formater et le monter.

### Partie B : Sauvegardes avec tar (60min)

#### Création d'archives

```
# Archive simple
tar cvf archive.tar dossier/    # Créer archive
tar cvf archive.tar file1 file2 dir1/ # Plusieurs éléments

# Archives compressées
tar cvzf archive.tar.gz dossier/ # Compression gzip
```

```
tar cvjf archive.tar.bz2 dossier/ # Compression bzip2
tar cvJf archive.tar.xz dossier/  # Compression xz
```

### Options importantes de tar

- c : créer une archive
- x : extraire une archive
- t : lister le contenu
- v : mode verbeux (affichage détaillé)
- f : spécifier le nom de fichier
- z : compression/décompression gzip
- j : compression/décompression bzip2

### Extraction et consultation

```
# Extraction
tar xvf archive.tar           # Extraire
tar xvzf archive.tar.gz      # Extraire archive gzip
tar xvf archive.tar -C /dest/ # Extraire vers destination

# Consultation
tar tvf archive.tar          # Lister le contenu
tar tvzf archive.tar.gz      # Lister contenu archive gzip
```

### Modes absolu et relatif

```
# Mode absolu (conserve chemins complets)
tar cvfP archive.tar /etc/passwd # Inclut le /

# Mode relatif (sans / initial)
cd /
tar cvf archive.tar etc/passwd   # Relatif depuis /
```

### Exercice 10 :

1. Créer un dossier backup\_test/ avec quelques fichiers
2. Créer une archive tar simple
3. Créer une archive compressée gzip
4. Comparer les tailles
5. Extraire l'archive dans un autre répertoire

## TD 5 : Virtualisation et Synthèse (2h)

### Partie A : Concepts de virtualisation (45min)

#### Types de virtualisation

##### Machines virtuelles (VMs) :

- Hyperviseur de Type 1 (bare metal) : VMware ESXi, Xen
- Hyperviseur de Type 2 (hosted) : VirtualBox, VMware Workstation
- Isolation complète, OS complet pour chaque VM

##### Conteneurs :

- Partage du noyau de l'OS hôte
- Plus légers et rapides que les VMs
- Docker, LXC, Podman

#### Avantages de la virtualisation

- Isolation : sécurité entre environnements
- Économies : meilleure utilisation des ressources
- Flexibilité : déploiement rapide, snapshots
- Portabilité : migration entre serveurs

#### Concepts Incus/LXC

```
# Gestion des conteneurs (sera approfondi en TP)
incus list                # Lister conteneurs
incus launch ubuntu:20.04 test # Créer conteneur
incus exec test -- /bin/bash  # Se connecter au conteneur
incus stop test           # Arrêter conteneur
incus delete test         # Supprimer conteneur
```

### Partie B : Synthèse et cas pratique (75min)

#### Révision des concepts clés

1. Navigation : pwd, cd, ls, mkdir, cp, mv, rm
2. Utilisateurs : useradd, usermod, passwd, id
3. Permissions : chmod, chown, lecture des droits
4. Processus : ps, kill, jobs, systemctl
5. Stockage : df, mount, tar

#### Exercice de synthèse : Configuration d'un mini-serveur

Scénario : Vous devez préparer un serveur pour héberger une application web.

## Étapes à réaliser :

1. Créer un utilisateur de service :
2. # Créer utilisateur 'webapp' sans shell de connexion
3. `sudo useradd -r -s /usr/sbin/nologin webapp`
4. Préparer l'arborescence :
5. # Créer structure pour l'application
6. `sudo mkdir -p /opt/webapp/{www,logs,config,backup}`
- 7.
8. # Définir les propriétaires et permissions
9. `sudo chown -R webapp:webapp /opt/webapp`
10. `sudo chmod 755 /opt/webapp`
11. `sudo chmod 755 /opt/webapp/www`
12. `sudo chmod 750 /opt/webapp/logs`
13. `sudo chmod 640 /opt/webapp/config`
14. Créer un script de sauvegarde :
15. # Script de sauvegarde automatique
16. `cat > /usr/local/bin/backup_webapp.sh << 'EOF'`
17. `#!/bin/bash`
18. `DATE=$(date +%Y%m%d_%H%M)`
19. `BACKUP_DIR="/opt/webapp/backup"`
- 20.
21. # Créer archive de l'application
22. `tar czf $BACKUP_DIR/webapp_$DATE.tar.gz /opt/webapp/www`
- 23.
24. # Nettoyer anciennes sauvegardes (> 7 jours)
25. `find $BACKUP_DIR -name "webapp_*.tar.gz" -mtime +7 -delete`
- 26.
27. `echo "Sauvegarde webapp_$DATE.tar.gz créée"`
28. `EOF`
- 29.
30. # Rendre exécutable
31. `sudo chmod +x /usr/local/bin/backup_webapp.sh`
32. Installer et configurer un service web :
33. # Installer Apache
34. `sudo apt update`
35. `sudo apt install apache2`
- 36.
37. # Configurer démarrage automatique
38. `sudo systemctl enable apache2`
- 39.
40. # Vérifier le statut
41. `sudo systemctl status apache2`
42. Surveillance système :
43. # Vérifier l'espace disque
44. `df -h`
- 45.
46. # Monitorer les processus Apache
47. `ps aux | grep apache2`
- 48.
49. # Consulter les logs
50. `sudo tail -f /var/log/apache2/access.log`

## Questions de validation :

1. Comment vérifier que l'utilisateur webapp a été créé correctement ?
2. Quelles permissions sont nécessaires pour qu'Apache puisse lire les fichiers web ?

3. Comment programmer l'exécution automatique du script de sauvegarde ?
4. Que faire si le service Apache ne démarre pas ?

## Ressources et références

### Commandes de base à retenir

```
# Navigation
pwd, cd, ls, mkdir, cp, mv, rm, find

# Utilisateurs
useradd, usermod, passwd, id, who, su, sudo

# Permissions
chmod, chown, chgrp, umask

# Processus
ps, top, kill, jobs, fg, bg, nohup

# Services
systemctl, journalctl

# Stockage
df, du, mount, umount, fdisk, tar

# Aide
man, apropos, --help
```


### Fichiers système importants

/etc/passwd	# Utilisateurs
/etc/group	# Groupes
/etc/shadow	# Mots de passe
/etc/fstab	# Points de montage
/var/log/	# Logs système

### Raccourcis clavier utiles

- Tab : autocomplétion
- Ctrl+C : arrêter processus
- Ctrl+Z : suspendre processus
- Ctrl+L : effacer terminal
- Ctrl+R : recherche dans l'historique
- ↑/↓ : naviguer dans l'historique

---

 Note : Ces TD seront approfondis dans les TP correspondants où vous mettrez en pratique ces concepts sur des cas concrets d'administration système.