

Administration Système

Administrez et maintenez des systèmes d'exploitation : services, sauvegardes, supervision, virtualisation et automatisation.
Niveau : Avancé • Durée : ~15h • 7 chapitres • Certificat inclus

1. Introduction à l'administration système

L'administrateur système installe, configure, sécurise et maintient les serveurs et les postes de travail d'une organisation. C'est un rôle transversal qui combine Linux (ou Windows Server), réseau et sécurité.

Mission	Exemple concret
Installation	Déployer un serveur web sur une nouvelle instance EC2
Configuration	Paramétrer Apache, PHP, MySQL
Supervision	Surveiller l'utilisation CPU/RAM/disque
Sécurité	Appliquer les correctifs, gérer les accès
Sauvegarde	Planifier et vérifier des sauvegardes régulières

2. Installation et configuration de serveurs

```
sudo apt update && sudo apt upgrade -y
sudo apt install apache2 -y
sudo systemctl enable apache2
sudo systemctl start apache2

# Vérifier la configuration avant de redémarrer

sudo apache2ctl configtest
sudo systemctl restart apache2
```

Les fichiers de configuration Apache se trouvent dans `/etc/apache2/` (sur Debian/Ubuntu), notamment `apache2.conf` et les fichiers de sites dans `sites-available/`.

■ Toujours tester avant de redémarrer : `apache2ctl configtest` évite de casser le service en production avec une erreur de syntaxe.

3. Gestion des services avec systemd

```
sudo systemctl status mysql
# voir l'état d'un service

sudo systemctl stop mysql
# arrêter

sudo systemctl start mysql
# démarrer

sudo systemctl restart mysql
# redémarrer

sudo systemctl enable mysql
# démarrage automatique au boot

journalctl -u apache2 -f
# suivre les logs du service en direct
```

■■ systemd : c'est le système d'initialisation standard des distributions Linux modernes (Ubuntu, Debian, CentOS...), qui gère le démarrage et la supervision de tous les services.

4. Sauvegardes et restauration

```
# Sauvegarder une base MySQL

mysqldump -u admin -p cours_info > sauvegarde_$(date +%F).sql


# Restaurer une base à partir d'une sauvegarde

mysql -u admin -p cours_info < sauvegarde_2026-07-01.sql


# Sauvegarder les fichiers du site

tar -czf site_backup_$(date +%F).tar.gz /var/www/html
```

■■ Tester ses sauvegardes : une sauvegarde jamais restaurée n'est pas fiable — testez régulièrement la procédure de restauration complète, idéalement sur un serveur de test.

5. Supervision et journaux (logs)

```
top / htop
# charge CPU/RAM en temps réel

df -h
# espace disque disponible

free -h
# mémoire disponible

tail -f /var/log/apache2/error.log
# suivre les erreurs Apache en direct

tail -f /var/log/apache2/access.log
# suivre les requêtes reçues
```

Une erreur « 404 introuvable » ou « 500 » côté client se retrouve toujours dans error.log côté serveur — c'est le premier réflexe de diagnostic pour un administrateur.

■ Diagnostiquer un 404 Apache : vérifiez que le fichier demandé existe réellement au bon chemin (ls /var/www/html/...) et que les permissions permettent à Apache (utilisateur www-data) de le lire.

6. Virtualisation et conteneurs (Docker)

Machine virtuelle (VM)	Conteneur (Docker)
Virtualise le matériel + un OS complet	Partage le noyau de l'hôte, isole seulement l'application
Plus lourd (Go de RAM/disque)	Léger et rapide à démarrer (secondes)
Isolation forte	Isolation plus légère mais suffisante pour la plupart des usages

```
docker --version
docker pull php:8.2-apache
docker run -d -p 80:80 -v $(pwd):/var/www/html php:8.2-apache
docker ps
# conteneurs actifs
```

■ ■ EC2 + Docker : il est possible de combiner les deux : une instance EC2 (la VM) qui exécute des conteneurs Docker à l'intérieur pour isoler plusieurs applications.

7. Automatisation et projet final

```
#!/bin/bash

USAGE=$(df -h / | awk 'NR==2 {print $5}' | tr -d '%')

if
[ "$USAGE" -gt 80 ];
then

echo "ALERTE : disque à ${USAGE}% le $(date)" >> /var/log/alerte_disque.log

fi

crontab -e
*/30 * * * * /home/ubuntu/verifier_disque.sh
# toutes les 30 minutes
```

■ **Projet final** : Mettez en place, sur votre instance EC2, une routine automatisée qui : sauvegarde la base cours_info chaque nuit avec mysqldump, vérifie que le service Apache est actif, et enregistre le résultat dans un fichier de log daté. Planifiez le tout avec cron.

Exercices & Quiz de vérification

Testez vos connaissances avant de passer au cours suivant. Chaque question a une correction détaillée à dérouler.

1. Quelle commande sauvegarde une base MySQL en fichier .sql ?

- A. mysql --backup
- B. mysqldump
- C. mysqlsave

Correction : Réponse : B. mysqldump -u utilisateur -p base > fichier.sql exporte l'intégralité d'une base.

2. Où trouve-t-on les erreurs Apache pour diagnostiquer un problème ?

- A. /var/log/apache2/error.log
- B. /home/apache/errors
- C. Dans le navigateur uniquement

Correction : Réponse : A. Ce fichier journal enregistre toutes les erreurs serveur (404, 500, erreurs PHP...).

3. Quelle est la principale différence entre une VM et un conteneur Docker ?

- A. Aucune différence
- B. Le conteneur partage le noyau de l'hôte, la VM virtualise un OS complet
- C. Docker ne fonctionne que sur Windows

Correction : Réponse : B. C'est pourquoi les conteneurs démarrent en quelques secondes alors qu'une VM peut prendre plusieurs minutes.

4. Quel outil planifie une tâche récurrente comme une sauvegarde automatique ?

- A. systemctl
- B. cron
- C. mysqldump

Correction : Réponse : B. cron exécute un script à intervalles réguliers définis dans la crontab.

Ressources supplémentaires

- [Documentation officielle Ubuntu Server](#)
- [Documentation officielle Apache HTTP Server 2.4](#)
- [Documentation officielle Docker — Prise en main](#)