TP Chapitre 2

Etape 1

1 - Créez un script Bash nommé /usr/local/bin/moniteur.sh qui :
o Fonctionne en boucle infinie.
o Affiche la date et le contenu du répertoire /tmp toutes les 10 secondes.
o Enregistre ces informations dans le fichier /var/log/moniteur.log.

```shell
#!/bin/bash
while true
do
echo "[$(date)] Contenu de /tmp :" >> /var/log/moniteur.log
ls /tmp >> /var/log/moniteur.log
echo "------------------------------" >> /var/log/moniteur.log
sleep 10
done


REPONSE
sudo vim /usr/local/bin/moniteur.sh
```

2 - Rendez ce script exécutable

```shell
sudo chmod u+x /usr/local/bin/moniteur.sh
```

3 - Lancez-le en arrière-plan à l'aide de &.

```shell
sudo su/usr/local/bin/moniteur.sh &
```

4 - Vérifiez que le fichier /var/log/moniteur.log est bien mis à jour.

```shell
ls -l /var/log/moniteur.log
```

Question 1 : Quel commande permet de lancer un script en arrière-plan

```shell
&
```

question 2 : Quelle commande permet de suivre en temps réel un fichier de log ?

```shell
tail -f /var/log/moniteur.log
```

Etape 2

1 - Identifier le PID du script en cours d'éxécution

```shell
ps aux | grep "/usr/local/bin/moniteur.sh"
```

2 - Changez la priorité de ce processus pour qu'il ait une priorité plus faible que la normale

```shell
sudo -v
# entrer le mot de passe
sudo nice -n 10 /usr/local/bin/moniteur.sh &
```

3 - Arrêtez proprement le processus

```shell
sudo kill -15 760
```

4 - Redémarrez le script en utilisant une méthode qui le laisse fonctionner même après fermeture de la session.

```shell
sudo nohup /usr/local/bin/moniteur.sh>> /usr/local/bin/moniteur.sh
2>&1 &
```

Question 3 : Quelle commande permet de voir le PID d'un processus spécifique ?

```shell
pgrep -f "/usr/local/bin/moniteur.sh"
```

Question 4 : Quelle est la différence entre nice et renice ?

Commande	Utilisation	Exemple
`nice`	Définit la priorité au lancement d'un processus	`nice -n 10 ./script.sh` (lance avec une faible priorité)
`renice`	Modifie la priorité d'un processus déjà en cours	`renice -n 10 -p 12345` (change la priorité du PID 12345)

Question 5: Quelle commande permet d'arrêter un processus proprement ?

```shell
sudo kill -15 760
```

Question 6 : Citez deux méthodes pour exécuter un script en fond de manière persistante

```shell
sudo nohup /usr/local/bin/moniteur.sh
sudo tmux new-session -d -s /usr/local/bin/moniteur.sh
```

Etape 3 - Création d'un service systemd

1 - Créez un fichier /etc/systemd/system/moniteur.service avec les informations suivantes :
o Description du service.
o Chemin d'exécution du script.
o Redémarrage automatique si plantage.
o Lancement au niveau multi-utilisateur.

```ini
# sudo vim /etc/systemd/system/moniteur.service

[Unit]
Description=Service de surveillance personnalisé (Moniteur)
After=multi-user.target

[Service]
Type=simple
ExecStart=/chemin/vers/votre/script.sh
Restart=always
RestartSec=5s
User=root
Group=root
Environment=PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:
/sbin:/bin

[Install]
WantedBy=multi-user.target
```

2- Rechargez les unités systemd.

```shell
sudo systemctl daemon-reload
```

3- Activez et démarrez le service.

```shell
 sudo systemctl enable moniteur.service
 sudo systemctl start moniteur.service
```

4- Vérifiez son statut

```shell
sudo systemctl status moniteur.service
```

Question 7 : Quelle commande permet de recharger les fichiers de service systemd ?

```shell
sudo systemctl daemon-reload
```

Question 8 : Quelle commande permet d'activer un service au démarrage ?

```shell
sudo systemctl enable moniteur.service
```

Question 9 : Quelle commande permet de consulter l'état d'un service ?

```shell
sudo systemctl status moniteur.service
```

Question 10 : Que signifie Restart=always dans un fichier de service ?
Restart=always signifie que le service sera toujour redémarré automatiquement

Etape 4 - Analyse et diagnostic

1-Modifiez le script pour qu'il provoque une erreur (ex. ls /chemin/inexistant).

```ini
# sudo vim /etc/systemd/system/moniteur.service

[Unit]
Description=Service de surveillance personnalisé (Moniteur)
After=multi-user.target

[Service]
Type=simple
ExecStart=/chemin/inexistant
Restart=always
RestartSec=5s
User=root
Group=root
Environment=PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:
/sbin:/bin

[Install]
WantedBy=multi-user.target
```

2-Redémarrez le service.

```shell
sudo systemctl restart moniteur.service
```

3-Observez les messages d’erreurs via journalctl.

SHELL

```
sudo journalctl -u moniteur.service
avril 08 13:32:30 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 08 13:51:26 kali systemd[1]: Stopping moniteur.service -
Service de surveillance personnalisé (Moniteur)...
avril 08 13:51:26 kali systemd[1]: moniteur.service: Deactivated
successfully.
avril 08 13:51:26 kali systemd[1]: Stopped moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 08 13:51:26 kali systemd[1]: moniteur.service: Consumed
1.528s CPU time, 1.6M memory peak.
-- Boot c05c7d63d38a466c9c1e99fd80325a25 --
avril 09 11:31:03 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:36:21 kali systemd[1]: Stopping moniteur.service -
Service de surveillance personnalisé (Moniteur)...
avril 09 11:36:21 kali systemd[1]: moniteur.service: Deactivated
successfully.
avril 09 11:36:21 kali systemd[1]: Stopped moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:36:21 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:36:59 kali systemd[1]: Stopping moniteur.service -
Service de surveillance personnalisé (Moniteur)...
avril 09 11:36:59 kali systemd[1]: moniteur.service: Deactivated
successfully.
avril 09 11:36:59 kali systemd[1]: Stopped moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:36:59 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:36:59 kali (xistanti)[4226]: moniteur.service: Unable
to locate executable '/chemin/inexistanti': No such file or
directory
avril 09 11:36:59 kali (xistanti)[4226]: moniteur.service: Failed
at step EXEC spawning /chemin/inexistanti: No such file or
directory
avril 09 11:36:59 kali systemd[1]: moniteur.service: Main process
exited, code=exited, status=203/EXEC
avril 09 11:36:59 kali systemd[1]: moniteur.service: Failed with
result 'exit-code'.
avril 09 11:37:04 kali systemd[1]: moniteur.service: Scheduled
restart job, restart counter is at 1.
```

```
avril 09 11:37:04 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:37:04 kali (xistanti)[4269]: moniteur.service: Unable
to locate executable '/chemin/inexistanti': No such file or
directory
avril 09 11:37:04 kali (xistanti)[4269]: moniteur.service: Failed
at step EXEC spawning /chemin/inexistanti: No such file or
directory
avril 09 11:37:04 kali systemd[1]: moniteur.service: Main process
exited, code=exited, status=203/EXEC
avril 09 11:37:04 kali systemd[1]: moniteur.service: Failed with
result 'exit-code'.
avril 09 11:37:10 kali systemd[1]: moniteur.service: Scheduled
restart job, restart counter is at 2.
avril 09 11:37:10 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
avril 09 11:37:10 kali (xistanti)[4320]: moniteur.service: Unable
to locate executable '/chemin/inexistanti': No such file or
directory
avril 09 11:37:10 kali (xistanti)[4320]: moniteur.service: Failed
at step EXEC spawning /chemin/inexistanti: No such file or
directory
avril 09 11:37:10 kali systemd[1]: moniteur.service: Main process
exited, code=exited, status=203/EXEC
avril 09 11:37:10 kali systemd[1]: moniteur.service: Failed with
result 'exit-code'.
avril 09 11:37:15 kali systemd[1]: moniteur.service: Scheduled
restart job, restart counter is at 3.
avril 09 11:37:15 kali systemd[1]: Started moniteur.service -
Service de surveillance personnalisé (Moniteur).
```

Question 11 : Quelle commande permet de consulter les logs d'un service ?

SHELL

```
sudo journalctl -u moniteur.service
```

Question 12 : Que se passe-t-il si le script échoue avec une erreur ? Le service continue-t-il à tourner ?
si on ne consulte pas les logs, rien ne se passe. Mais après la consultation de log, on voit qu'il y a des erreurs
Oui le service continue à tourner

```shell
moniteur.service - Service de surveillance personnalisé (Moniteur)
     Loaded: loaded (/etc/systemd/system/moniteur.service; enabled;
preset: disabled)
     Active: activating (auto-restart) (Result: exit-code) since
Wed 2025-04-09 11:40:55 EDT; 2s ago
 Invocation: 6a4f55c2d00e46a0b97de94554a9d871
    Process: 6326 ExecStart=/chemin/inexistanti (code=exited,
status=203/EXEC)
   Main PID: 6326 (code=exited, status=203/EXEC)
   Mem peak: 1.4M
        CPU: 10ms
```

Question de synthèse

Question 13 : Quelle différence y a-t-il entre un processus lancé manuellement et un service systemd ?

Processus manuel	Service systemd
Démarré avec un terminal (`./script.sh`)	Démarré par `systemd` via `systemctl start`
Dépend de la session du terminal	Tourne en **arrière-plan indépendamment** de la session
Ne redémarre pas en cas d'erreur	Peut être configuré pour **redémarrer automatiquement** (`Restart=always`)
Interactif (lié à l'utilisateur)	**Automatisé**, démarré au **boot** si activé
Difficile à superviser ou journaliser	**Géré proprement** avec `journalctl`, `status`, `start`, `stop`, etc.

Question 14 : Pourquoi est-il conseillé de ne pas exécuter un service en tant que root ?
Raison principale : la sécurité.

- Si le service est compromis, il pourrait faire **tout ce qu'un super-utilisateur peut faire** (supprimer des fichiers système, ouvrir des ports critiques, etc.).
- C'est un **principe de moindre privilège** : un service ne doit avoir **que les droits nécessaires à son fonctionnement**.
- Ça limite les dégâts potentiels en cas de faille.

Question 15 : Comment sécuriser un service personnalisé (systemd) ?

- **Créer un utilisateur dédié** au service :

```shell
sudo useradd --system --no-create-home monservice
```

```
- Dans le fichier `.service` :
    ```ini
    User=monservice Group=monservice
```

- **Restreindre les permissions** sur les fichiers utilisés :
  - Fichiers logs dans `/var/log/monservice/` avec `chown monservice`.
- Ajouter des directives de sécurité dans `[Service]` :

```ini
NoNewPrivileges=true
PrivateTmp=true
ProtectSystem=full
ProtectHome=true
ReadOnlyPaths=/etc
```

```
- Utiliser `CapabilityBoundingSet=` pour limiter les permissions
système.

Question 16 : Si un service ne démarre pas, quelles sont les étapes
de diagnostic à suivre?

- **Vérifier l'état du service** :
```Bash
sudo systemctl status nom_service
```

- **Lire les logs détaillés** :

```shell
journalctl -u nom_service
```

- **Regarder le fichier `.service` pour erreurs de syntaxe**.
- **Tester manuellement la commande `ExecStart`** dans un terminal.
- **Vérifier les permissions** du script et des fichiers utilisés.
- **Recharger systemd si le fichier a changé** :

```shell
sudo systemctl daemon-reload
```