



Systeme, Scripts et Sécurité

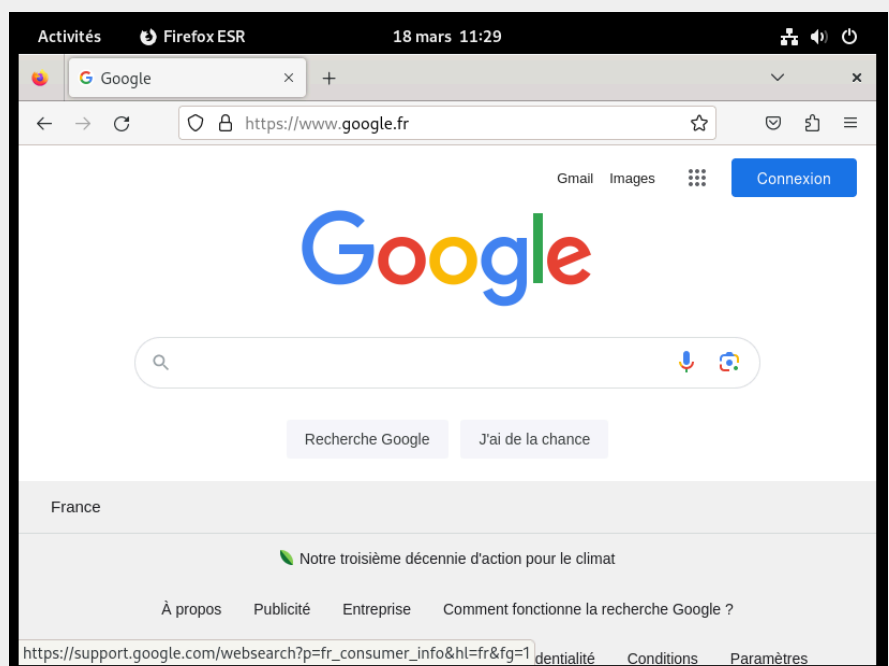
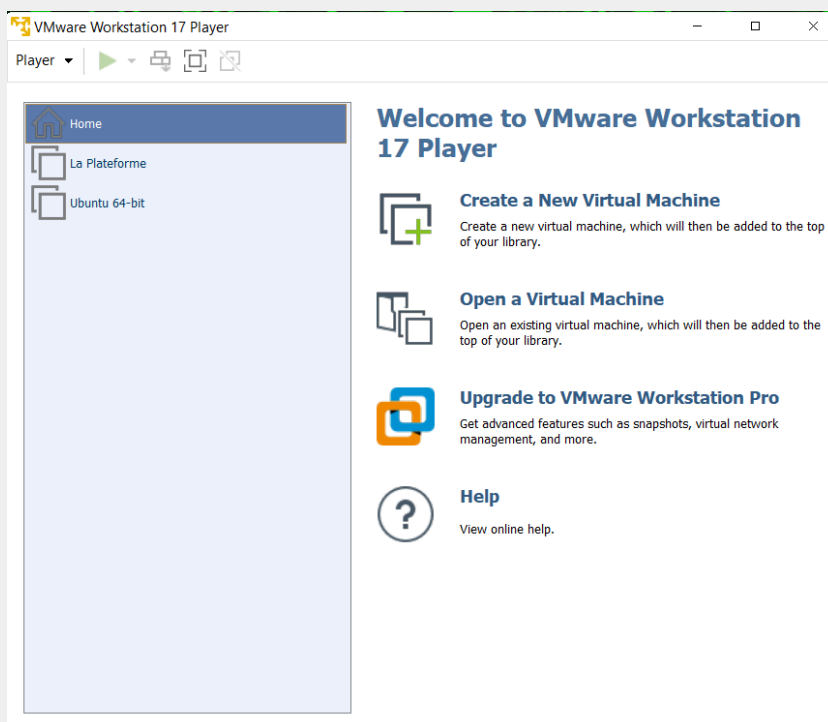
“ Un voyage à travers les commandes, Scripts et interactions avancées.(Avec les illuminati) ”

1/ Création d'une VM Debian

Création d'une machine virtuelle Debian avec interface graphique (AMD64)

La machine virtuelle a comme nom de session : La_Plateforme et mot de passe : LAPlateforme_.

Connexion de la machine à internet.





2/ Commandes de recherche avancée

Création de cinq fichiers textes nommés “mon_texte.txt” qui contient le texte suivant :

« Que la force soit avec toi. » .

Répartition des fichiers dans les répertoires suivants :

“Bureau”, “Documents”, “Téléchargement”, “Vidéos” et “Images”.

Utilisation du mot “force” pour localiser les cinq fichiers “mon_texte.txt” dans les différents répertoires.

Commandes :

*Utilisation des commandes **echo** et **cp** pour créer le fichier puis le copier.*

*Utilisation de la commande **find**. -type f -name “mon_texte.txt” -exec grep -l “force” {} \;* pour localiser les fichiers.

```
hugo@LaPlateforme: ~  
hugo@LaPlateforme:~$ ls  
Bureau Documents Images Modèles Musique Public Téléchargements Vidéos  
hugo@LaPlateforme:~$ echo "Que la force soit avec toi" > ~/Bureau/mon_texte.txt  
hugo@LaPlateforme:~$ cp ~/Bureau/mon_texte.txt ~/Documents/  
hugo@LaPlateforme:~$ cp ~/Bureau/mon_texte.txt ~/Téléchargements/  
hugo@LaPlateforme:~$ cp ~/Bureau/mon_texte.txt ~/Vidéos/  
hugo@LaPlateforme:~$ cp ~/Bureau/mon_texte.txt ~/Images/  
hugo@LaPlateforme:~$
```

```
kheira@LaPlateforme: ~  
kheira@LaPlateforme:~$ find . -type f -name "mon_texte.txt" -exec grep -l "force"  
" {} \;  
./Images/mon_texte.txt  
./Bureau/mon_texte.txt  
./Téléchargements/mon_texte.txt  
./Documents/Plateforme/mon_texte.txt  
./Vidéos/mon_texte.txt  
./mon_texte.txt  
kheira@LaPlateforme:~$
```



3/ Compression et décompression de fichiers

Création d'un répertoire nommé "Plateforme" dans le dossier "Documents" de votre session et ajout du fichier "mon_texte.txt" précédemment créé.

Duplication du fichier quatre fois dans le même répertoire, formant ainsi un total de cinq fichiers dans le répertoire "Plateforme".

Archivage du répertoire "Plateforme" en utilisant les commandes "tar" et "gzip".

Décompression des archives créées.

Commandes :

*Création du répertoire Plateforme avec la commande **mkdir** et ajout des fichiers à l'intérieur avec la commande **mv** puis duplication des fichiers avec la commande **cp***

*Décompression des archives avec les commandes **tar.gz***

```
tom@Laplateforme: ~/Documents/Plateforme
tom@Laplateforme:~$ cp ~/Bureau/mon_texte.txt ~/Vidéos/
tom@Laplateforme:~$ cp ~/Bureau/mon_texte.txt ~/Documents/
tom@Laplateforme:~$ cp ~/Bureau/mon_texte.txt ~/Téléchargements/
tom@Laplateforme:~$ cp Images
cp: opérande de fichier cible manquant après 'Images'
Saisissez « cp --help » pour plus d'informations.
tom@Laplateforme:~$ cd Images
tom@Laplateforme:~/Images$ ls
mon_texte.txt
tom@Laplateforme:~/Images$ cd
tom@Laplateforme:~$ cd Documents/
tom@Laplateforme:~/Documents$ mkdir Plateforme
tom@Laplateforme:~/Documents$ ls
mon_texte.txt Plateforme
tom@Laplateforme:~/Documents$ mc mon_texte.txt Plateforme
bash: mc : commande introuvable
tom@Laplateforme:~/Documents$ mv mon_texte.txt Plateforme
tom@Laplateforme:~/Documents$ ls
Plateforme
tom@Laplateforme:~/Documents$ cd Plateforme
tom@Laplateforme:~/Documents/Plateforme$ ls
mon_texte.txt
tom@Laplateforme:~/Documents/Plateforme$
```

```
tom@Laplateforme:~/Documents/Plateforme
tom@Laplateforme:~$ ls
Bureau Images Musique SystemesScriptSecurite Vidéos
Documents Modèles Public Téléchargements
tom@Laplateforme:~$ cd Documents/
tom@Laplateforme:~/Documents$ ls
Plateforme
tom@Laplateforme:~/Documents$ cd Plateforme/
tom@Laplateforme:~/Documents/Plateforme$ ls
mon_texte.txt
tom@Laplateforme:~/Documents/Plateforme$ cp mon_texte.txt
cp: opérande de fichier cible manquant après 'mon_texte.txt'
Saisissez « cp --help » pour plus d'informations.
tom@Laplateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte1.txt
tom@Laplateforme:~/Documents/Plateforme$ ls
mon_texte1.txt mon_texte.txt
tom@Laplateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte2.txt
tom@Laplateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte3.txt
tom@Laplateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte4.txt
tom@Laplateforme:~/Documents/Plateforme$ ls
mon_texte1.txt mon_texte2.txt mon_texte3.txt mon_texte4.txt mon_texte.txt
tom@Laplateforme:~/Documents/Plateforme$
```

```
tom@Laplateforme:~/Documents
Plateforme/mon_texte4.txt
tom@Laplateforme:~/Documents$ ls
Plateforme Plateforme.tar.gz
tom@Laplateforme:~/Documents$ tar -cjvf Plateforme.tar.bz2 Plateforme
Plateforme/
Plateforme/mon_texte2.txt
Plateforme/mon_texte.txt
Plateforme/mon_texte3.txt
Plateforme/mon_texte1.txt
Plateforme/Plateforme.tar.gz
Plateforme/mon_texte4.txt
tom@Laplateforme:~/Documents$ ls
Plateforme Plateforme.tar.bz2 Plateforme.tar.gz
tom@Laplateforme:~/Documents$ tar -cjvf Plateforme.tar.xz Plateforme
Plateforme/
Plateforme/mon_texte2.txt
Plateforme/mon_texte.txt
Plateforme/mon_texte3.txt
Plateforme/mon_texte1.txt
Plateforme/Plateforme.tar.gz
Plateforme/mon_texte4.txt
tom@Laplateforme:~/Documents$ ls
Plateforme Plateforme.tar.bz2 Plateforme.tar.gz Plateforme.tar.xz
tom@Laplateforme:~/Documents$
```



4/ Manipulation de texte

Utilisation de la commande "awk" afin d'extraire des informations spécifiques d'un fichier CSV.

Création d'un script python pour créer un fichier CSV et l'ajout des données suivantes :

- Jean, 25 ans, Paris
- Marie, 30 ans, Lyon
- Pierre, 22 ans, Marseille
- Sophie, 35 ans, Toulouse

Extraction des informations relatives aux villes de chaque personne avec la commande "awk".

Commandes :

*La commande **awk '{print \$3}' FS=',' data.csv**, permet d'extraire les informations relatives aux villes.*

```
GNU nano 7.2
#!/bin/bash

cat > script.py << EOF
import csv
data = [{"Jean", "25 ans", "Paris"},
        {"Marie", "30 ans", "Lyon"},
        {"Pierre", "22 ans", "Marseille"},
        {"Sophie", "35 ans", "Toulouse"}]

with open("data.csv", "w") as f:
    writer = csv.writer(f)
    writer.writerows(data)
EOF

chmod +x script.py

python3 script.py
```



5/ Gestion des processus

Afin d'explorer les processus, on recense tous ceux qui sont actifs sur le système.

Utilisation d'une commande permettant de fermer un processus, puis exécution de celle-ci pour terminer un processus spécifique.

Il peut être nécessaire de terminer un processus de façon forcée à l'aide d'une commande. Comparaison et observation des différences entre le processus de terminaison normale et celui effectué de manière forcée.

Commandes :

*Affichage des processus actifs avec la ligne de commande : **ps aux***

*Fermeture d'un processus : **kill** suivie du **PID***

```
tom@Laplateforme: ~/Documents
Plateforme/mon_texte.txt
Plateforme/mon_texte3.txt
Plateforme/mon_texte1.txt
Plateforme/Plateforme.tar.gz
Plateforme/mon_texte4.txt
tom@Laplateforme:~/Documents$ ls
Plateforme Plateforme.tar.bz2 Plateforme.tar.gz Plateforme.tar.xz
tom@Laplateforme:~/Documents$ ps aux
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.3 168072 12564 ?        Ss   10:45   0:02 /sbin/init
root           2  0.0  0.0      0     0 ?        S    10:45   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   10:45   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   10:45   0:00 [rcu_par_gp]
root           5  0.0  0.0      0     0 ?        I<   10:45   0:00 [slub_flushwq]
root           6  0.0  0.0      0     0 ?        I<   10:45   0:00 [netns]
root           8  0.0  0.0      0     0 ?        I<   10:45   0:00 [kworker/0:0H-events]
root          10  0.0  0.0      0     0 ?        I<   10:45   0:00 [mm_percpu_wq]
root          11  0.0  0.0      0     0 ?        I    10:45   0:00 [rcu_tasks_kthread]
root          12  0.0  0.0      0     0 ?        I    10:45   0:00 [rcu_tasks_rude_kthr]
root          13  0.0  0.0      0     0 ?        I    10:45   0:00 [rcu_tasks_trace_kth]
root          14  0.0  0.0      0     0 ?        S    10:45   0:00 [ksoftirqd/0]
root          15  0.0  0.0      0     0 ?        I    10:45   0:00 [rcu_preempt]
root          16  0.0  0.0      0     0 ?        S    10:45   0:00 [migration/0]
root          17  0.3  0.0      0     0 ?        I    10:45   0:09 [kworker/0:1-events]
root          18  0.0  0.0      0     0 ?        S    10:45   0:00 [cpuhp/0]
root          20  0.0  0.0      0     0 ?        S    10:45   0:00 [kdevtmpfs]
root          21  0.0  0.0      0     0 ?        I<   10:45   0:00 [inet_frag_wq]
root          22  0.0  0.0      0     0 ?        S    10:45   0:00 [kauditd]
root          24  0.0  0.0      0     0 ?        S    10:45   0:00 [khungtaskd]
root          26  0.0  0.0      0     0 ?        S    10:45   0:00 [oom_reaper]
root          27  0.0  0.0      0     0 ?        I<   10:45   0:00 [writeback]
root          29  0.0  0.0      0     0 ?        S    10:45   0:00 [kcompactd0]
root          30  0.0  0.0      0     0 ?        SN   10:45   0:00 [ksmd]
root          31  0.0  0.0      0     0 ?        SN   10:45   0:01 [khugepaged]
root          32  0.0  0.0      0     0 ?        I<   10:45   0:00 [kintegrityd]
root          33  0.0  0.0      0     0 ?        I<   10:45   0:00 [kblockd]
root          34  0.0  0.0      0     0 ?        I<   10:45   0:00 [blkcg_punt_bio]
root          35  0.0  0.0      0     0 ?        I<   10:45   0:00 [tpm_dev_wq]
root          36  0.0  0.0      0     0 ?        I<   10:45   0:00 [edac-poller]
root          37  0.0  0.0      0     0 ?        I<   10:45   0:00 [devfreq_wq]
```

```
tom@Laplateforme: ~/SystemeScriptSecure
tom      1601  0.0  0.2 456256 9612 ?        Ssl  12:25   0:00 /usr/libexec/gsd-usb-protection
tom      1602  0.0  0.7 340124 27716 ?        Ssl  12:25   0:00 /usr/libexec/gsd-wacom
tom      1603  0.0  2.2 754928 87876 ?        Sl   12:25   0:03 /usr/bin/gnome-software --gapplication
tom      1689  0.0  0.7 2727512 29600 ?        Sl   12:25   0:00 /usr/bin/gjs /usr/share/gnome-shell/or
tom      1726  0.0  0.2 234504 9268 ?        Sl   12:25   0:00 /usr/libexec/ibus-dconf
tom      1727  0.0  0.8 344828 32988 ?        Sl   12:25   0:05 /usr/libexec/ibus-extension-gtk3
tom      1729  0.0  0.1 234444 7332 ?        Sl   12:25   0:00 /usr/libexec/ibus-portal
tom      1732  0.0  0.4 342136 17720 ?        Sl   12:25   0:00 /usr/libexec/gsd-printer
tom      1777  0.0  0.5 621916 21100 ?        Ssl  12:25   0:00 /usr/libexec/xdg-desktop-portal
tom      1792  0.0  0.2 460644 9640 ?        Ssl  12:25   0:00 /usr/libexec/xdg-document-portal
root     1802  0.0  0.0 2480 972 ?        Ss   12:25   0:00 fusermount3 -o rw,nosuid,nodev,fsname=
tom      1807  0.0  0.7 377532 29764 ?        Ssl  12:25   0:00 /usr/libexec/xdg-desktop-portal-gnome
tom      1820  0.0  0.2 160804 9244 ?        Sl   12:25   0:02 /usr/libexec/ibus-engine-simple
tom      1828  0.0  1.5 703420 60824 ?        Ssl  12:25   0:01 /usr/libexec/tracker-miner-fs-3
tom      1834  0.0  0.1 156440 5536 ?        Ssl  12:25   0:00 /usr/libexec/dconf-service
tom      1846  0.0  0.6 340540 26380 ?        Ssl  12:26   0:00 /usr/libexec/xdg-desktop-portal-gtk
tom      1889  0.0  0.2 160124 8552 ?        Ssl  12:26   0:00 /usr/libexec/gvfsd-metadata
tom      1904  0.0  1.5 748116 60980 ?        Sl   12:26   0:00 /usr/bin/gnome-calendar --gapplication
tom      1961  0.0  0.3 311848 12204 ?        Sl   12:26   0:00 /usr/libexec/gvfsd-trash --spawner :1.
tom      2206  0.1  1.3 560184 53108 ?        Ssl  12:26   0:14 /usr/libexec/gnome-terminal-server
tom      2242  0.0  0.1 8392 5352 pts/0    Ss   12:26   0:00 bash
tom      2338  0.0  1.7 230148 67532 ?        S    12:39   0:08 /usr/bin/Xwayland :0 -rootless -nores
tom      2342  0.0  2.1 549088 81928 ?        Ssl  12:39   0:00 /usr/libexec/gsd-xsettings
tom      2365  0.0  0.5 190768 22064 ?        Sl   12:39   0:00 /usr/libexec/ibus-x11
root     3445  0.0  0.0      0     0 ?        I    14:25   0:00 [kworker/u256:0-ext4-rsv-conversion]
root     3541  0.2  0.0      0     0 ?        I    14:47   0:03 [kworker/0:3-ata_sff]
root     3542  0.0  0.0      0     0 ?        I    14:47   0:00 [kworker/u256:2-ext4-rsv-conversion]
root     3543  0.2  0.0      0     0 ?        I    14:52   0:02 [kworker/0:1-events_freezable_power_]
root     3546  0.0  0.0      0     0 ?        I    14:57   0:00 [kworker/0:0-ata_sff]
root     3547  0.0  0.0      0     0 ?        I    15:02   0:00 [kworker/u256:1-events_unbound]
root     3548  0.1  0.0      0     0 ?        I    15:02   0:00 [kworker/0:2-events]
tom      3577  318  0.1 11344 4892 pts/0    R+   15:07   0:00 ps aux
tom@Laplateforme:~/SystemeScriptSecure$ kill 666
```



6/ Surveillance des ressource

Mise en place d'une surveillance en temps réel de l'utilisation du CPU, de la mémoire et d'autres ressources système.

Enregistrement des données dans un fichier CSV.

Commandes :

Affichage les informations d'utilisations des ressources systèmes : top

Enregistrement des données :

```
top -b -n 1 | tail -n +8 | awk '{print $0","}' > output.csv
```

```
kheira@LaPlateforme: ~  
top - 14:07:02 up 1:26, 1 user, load average: 0,37, 0,28, 0,19  
Tâches: 208 total, 1 en cours, 207 en veille, 0 arrêté, 0 zombie  
%Cpu(s): 0,3 ut, 0,7 sy, 0,0 ni, 99,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st  
MiB Mem : 3784,6 total, 2027,0 libr, 1282,7 util, 718,0 tmp/cache  
MiB Éch : 975,0 total, 975,0 libr, 0,0 util. 2501,9 dispo Mem  
  
PID UTIL. PR NI VIRT RES SHR S %CPU %MEM TEMPS+ COM.  
1384 kheira 20 0 0,0p 0,0p 0,0p S 0,7 8,0 0:53.93 gnome-s+  
2565 root 20 0 0,0p 0,0p 0,0p I 0,7 0,0 0:00.43 kworker+  
706 root 20 0 0,0p 0,0p 0,0p S 0,3 0,1 0:00.80 apache2  
2205 kheira 20 0 0,0p 0,0p 0,0p S 0,3 1,4 0:08.28 gnome-t+  
2446 kheira 20 0 0,0p 0,0p 0,0p R 0,3 0,1 0:07.94 top  
1 root 20 0 0,0p 0,0p 0,0p S 0,0 0,3 0:01.31 systemd  
2 root 20 0 0,0p 0,0p 0,0p S 0,0 0,0 0:00.01 kthreadd  
3 root 0 -20 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 rcu_gp  
4 root 0 -20 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 rcu_par+  
5 root 0 -20 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 slub_fl+  
6 root 0 -20 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 netns  
7 root 20 0 0,0p 0,0p 0,0p I 0,0 0,0 0:23.24 kworker+  
8 root 0 -20 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 kworker+  
10 root 0 -20 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 mm_perc+  
11 root 20 0 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 rcu_tas+  
12 root 20 0 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 rcu_tas+  
13 root 20 0 0,0p 0,0p 0,0p I 0,0 0,0 0:00.00 rcu_tas+
```

Import de texte - [output.csv]

X

Importer

Jeu de caractères : Unicode (UTF-8)

Locale : Par défaut - Français (France)

À partir de la ligne : 1

Options de séparateur

☐ Largeur fixe
☒ Séparé par

☒ Tabulation
☒ Virgule
☒ Point-virgule
☐ Espace
☐ Autre

☐ Fusionner les séparateurs
☐ Espaces superflus

Séparateur de chaîne de caractères : "

Autres options

☐ Formater les champs entre guillemets comme texte
☐ Détecter les nombres spéciaux
☐ Évaluer les formules

Champs

Type de colonne :

	Standard						Standard	Standard
1	1384 kheira	20	0	3352352	304308	133112 S	6 2 7 9	0:36.92 gnome-s+
2	1571 kheira	20	0	386272	14352	7116 S	6 2 0 4	0:00.51 ibus-da+
3	2565 root	20	0	0	0	0 I	6 2 0 0	0:00.01 kworker+
4	2573 kheira	20	0	11708	5336	3400 R	6 2 0 1	0:00.02 top
5	1 root	20	0	168032	12508	9184 S	0 0 0 3	0:01.30 systemd
6	2 root	20	0	0	0	0 S	0 0 0 0	0:00.01 kthreadd
7	3 root	0	-20	0	0	0 I	0 0 0 0	0:00.00 rcu_gp
8	4 root	0	-20	0	0	0 I	0 0 0 0	0:00.00 rcu_part
9	5 root	0	-20	0	0	0 I	0 0 0 0	0:00.00 slub_flt
10	6 root	0	-20	0	0	0 I	0 0 0 0	0:00.00 netns

Aide

Annuler

OK



7/ Scripting avancés système

Développement d'un script Shell visant à automatiser la sauvegarde périodique du répertoire « Plateforme » créé précédemment.

Intégration d'une fonctionnalité de gestion de l'historique des sauvegardes, permettant ainsi de conserver un suivi chronologique des opérations effectuées.

Commandes :

```
date=`date +%Y-%m-%d`  
SOURCE="/home/hugo/Documents/Plateforme"  
BACKUP="/home/hugo/Documents/backup"  
HISTORIC="/home/hugo/Documents/backup/$date"  
  
if [ ! -d "$SOURCE" ]; then  
    echo "Erreur le fichier Plateforme n'existe pas"  
    exit 1  
fi  
  
if [ ! -d "$BACKUP" ]; then  
    mkdir "$BACKUP"  
fi  
  
if [ ! -d "$HISTORIC" ]; then  
    mkdir "$HISTORIC"  
fi  
  
cp -R "$SOURCE"/* "$HISTORIC"  
  
if [ $? -eq 0 ]; then  
    echo "Sauvegarde effectuée avec succès."  
else  
    echo "Erreur lors de la sauvegarde."  
fi
```

```
GNU nano 7.2  
# Edit this file to introduce tasks to be run by cron.  
#  
# Each task to run has to be defined through a single line  
# indicating with different fields when the task will be run  
# and what command to run for the task  
#  
# To define the time you can provide concrete values for  
# minute (m), hour (h), day of month (dom), month (mon),  
# and day of week (dow) or use '*' in these fields (for 'any').  
#  
# Notice that tasks will be started based on the cron's system  
# daemon's notion of time and timezones.  
#  
# Output of the crontab jobs (including errors) is sent through  
# email to the user the crontab file belongs to (unless redirected).  
#  
# For example, you can run a backup of all your user accounts  
# at 5 a.m every week with:  
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/  
#  
# For more information see the manual pages of crontab(5) and cron(8)  
#  
# m h dom mon dow  command  
* * * * * /home/hugo/SystemeScriptSecurite/scripting_avancé.sh
```




8/ Automatisation des mises à jour logicielles

Création d'un script automatisant la recherche de mise à jour des logiciels existants sur le système afin procéder à la mise à jour des logiciels.

Commandes :

*Installation de crontab avec la commande **apt-get install cron** puis utilisation de **crontab** (planificateur de tâches) pour planifier une heure pour les mises à jour.*

```
tom@Laplateforme: ~  
GNU nano 7.2 /tmp/crontab.gstfcd/crontab *  
# Edit this file to introduce tasks to be run by cron.  
#  
# Each task to run has to be defined through a single line  
# indicating with different fields when the task will be run  
# and what command to run for the task  
#  
# To define the time you can provide concrete values for  
# minute (m), hour (h), day of month (dom), month (mon),  
# and day of week (dow) or use '*' in these fields (for 'any').  
#  
# Notice that tasks will be started based on the cron's system  
# daemon's notion of time and timezones.  
#  
# Output of the crontab jobs (including errors) is sent through  
# email to the user the crontab file belongs to (unless redirected).  
#  
# For example, you can run a backup of all your user accounts  
# at 5 a.m every week with:  
0 5 * * * home/tom/SystemeScriptSecurite/maj.sh  
#  
# For more information see the manual pages of crontab(5) and cron(8)  
#  
# m h dom mon dow command
```

```
tom@Laplateforme: ~/SystemeScriptSecurite  
GNU nano 7.2 maj.sh *  
#!/bin/bash  
  
update_packages() {  
    echo " Mise à jour des paquets en cours !"  
    sudo apt update && sudo apt upgrade -y  
}  
  
echo " Recherche de mises à jour en cours ! "  
sudo apt update > /dev/null 2>&1  
  
if [ $? -eq 0 ]; then  
    echo " des mises à jours sont disponibles ! "  
    read -p " Souhaitez vous mettre à jour ? (O/N) ? " choice  
    case "$choice" in  
        [oO])  
            echo "Lancement des mises à jour"  
            update_packages  
            ;;  
        [nN])  
            echo "Mise à jour annulé !"  
            ;;  
        *)  
            echo "Choix invalide, mise à jour annulé ! "  
            ;;  
    esac  
else  
    echo "Pas de mises à jour disponible"  
fi
```

```
tom@Laplateforme: ~$ cd SystemeScriptSecurite/  
tom@Laplateforme: ~/SystemeScriptSecurite$ ./maj.sh  
Recherche de mises à jour en cours !  
[sudo] Mot de passe de tom :  
des mises à jours sont disponibles !  
Souhaitez vous mettre à jour ? (O/N) ? o  
Lancement des mises à jour  
Mise à jour des paquets en cours !  
Atteint :1 http://deb.debian.org/debian bookworm InRelease  
Atteint :2 http://deb.debian.org/debian-security bookworm-security InRelease  
Atteint :3 http://deb.debian.org/debian bookworm-updates InRelease  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances... Fait  
Lecture des informations d'état... Fait  
2 paquets peuvent être mis à jour. Exécutez « apt list --upgradable » pour les v  
oir.  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances... Fait  
Lecture des informations d'état... Fait  
Calcul de la mise à jour... Fait  
Les paquets suivants seront mis à jour :  
  firefox-esr firefox-esr-l10n-fr
```

```
tom@Laplateforme: ~/SystemeScriptSecurite$ ./maj.sh  
Recherche de mises à jour en cours !  
Pas de mises à jour disponible
```

```
tom@Laplateforme: ~/SystemeScriptSecurite$ ./maj.sh  
Recherche de mises à jour en cours !  
des mises à jours sont disponibles !  
Souhaitez vous mettre à jour ? (O/N) ? n  
Mise à jour annulé !
```




9/ Gestion des dépendances logicielles

Élaboration d'un script ayant pour objectif de simplifier l'installation et la gestion des dépendances logicielles pour un projet web, tout assurant la compatibilité entre les différentes versions.

Le script doit installer les éléments suivants :

- Un serveur Web (Apache ou Nginx)
- phpMyAdmin
- Un système de gestion de base de données relationnelle (MySQL ou MariaDB)
- Un environnement JavaScript côté serveur (Node.js) avec npm
- Un système de contrôle de version (Git)

Commandes :

```
Activités Terminal 22 mars 14:50
kheira@LaPlateforme: ~/SystemeScriptSecurite
script_logiciels.sh *
GNU nano 7.2
#!/bin/bash

bash "./maj.sh"

choice=0
choice2=0

while [[ "$choice" != "1" && "$choice" != "2" ]]; do

    read -p "Souhaitez-vous installer Apache (1) ou Nginx (2) ?" choice

    if [[ "$choice" != "1" && "$choice" != "2" ]]; then
        echo "Option invalide, veuillez choisir 1 ou 2"
    fi
done

case "$choice" in
    [1])
        echo "Installation d'Apache2"
        sudo apt install -y apache2
        ;;
    [2])
        echo "Installation de Nginx"
        sudo apt install -y nginx
        ;;
esac

while [[ "$choice2" != "1" && "$choice2" != "2" ]]; do

    read -p "Souhaitez-vous installer MySQL (1) ou Mariadb (2) ?" choice2

    if [[ "$choice2" != "1" && "$choice2" != "2" ]]; then
        echo "Option invalide, veuillez choisir 1 ou 2"
    fi
done

case "$choice2" in
    1)
        echo "Installation de MySQL"
        sudo apt install -y mysql-server
        sudo apt install -y phpmyadmin
        ;;
    2)
        echo "Installation de Mariadb et de Phpmyadmin"
        sudo apt install -y mariadb-server
        sudo apt install -y phpmyadmin
        ;;
esac

sudo apt install -y nodejs npm

sudo apt install -y git
```



```
kheira@LaPlateforme: ~/SystemeScriptSecurite
Outil de configuration des paquets

Configuration de phpmyadmin

Le paquet phpmyadmin a besoin d'une base de données installée et configurée avant de pouvoir être utilisé. Ceci peut si
nécessaire être géré par dbconfig-common.

Si vous êtes un administrateur de bases de données expérimenté et savez que vous voulez procéder à cette configuration
vous-même, ou si votre base de données est déjà installée et configurée, vous pouvez refuser cette option. Des précisions sur
la procédure se trouvent dans /usr/share/doc/phpmyadmin.

Autrement, vous devriez choisir cette option.

Faut-il configurer la base de données de phpmyadmin avec dbconfig-common ?

<Oui> <Non>
```

```
kheira@LaPlateforme: ~/SystemeScriptSecurite
kheira@LaPlateforme:~/SystemeScriptSecurite$ ./script_logiciels.sh
Souhaitez-vous installer Apache (1) ou Nginx (2) ?3
Option invalide, veuillez choisir 1 ou 2
Souhaitez-vous installer Apache (1) ou Nginx (2) ?1
Installation d'Apache2
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
apache2 est déjà la version la plus récente (2.4.57-2).
apache2 passé en « installé manuellement ».
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
Souhaitez-vous installer MySQL (1) ou Mariadb (2) ?1
Installation de MySQL
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Aucune version du paquet mysql-server n'est disponible, mais il existe dans la base
de données. Cela signifie en général que le paquet est manquant, qu'il est devenu obsolète
ou qu'il n'est disponible que sur une autre source

E: Le paquet « mysql-server » n'a pas de version susceptible d'être installée
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
dbconfig-common dbconfig-mysql default-mysql-client icc-profiles-free libconfig-inifiles-perl libdbd-mariadb-perl libdbi-perl

Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
git est déjà la version la plus récente (1:2.39.2-1.1).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
kheira@LaPlateforme:~/SystemeScriptSecurite$
```



10/ Sécuriser ses scripts

Identification des risques liés à la négligence de la sécurité des scripts.
Sécurisation des différents scripts développés précédemment.

Commandes :

Utilisation de l'option **-n** pour limiter le nombre de caractères lu dans un script et dans les **choices**, pour éviter l'injection de code lors du lancement du script.

La négligence de la sécurité des scripts Shell peut présenter plusieurs risques ,
notamment :

- Exposition de clés d'API et d'informations sensibles : Si les scripts contiennent des clés d'API ou d'autres informations sensibles, ces données peuvent être exposées si les scripts sont compromis ou consultés par des personnes non autorisées.
- Injection de code : Les scripts Shell peuvent être vulnérables aux injections de code si les entrées utilisateur ne sont pas correctement validées ou échappées. Cela peut permettre à un attaquant d'exécuter des commandes malveillantes sur le système.
- Accès non autorisé : Les scripts peuvent accorder un accès non autorisé aux ressources système si les autorisations ne sont pas correctement configurées ou si les scripts sont modifiés.
- Attaques par déni de service (DoS) : Les scripts mal conçus peuvent être exploités pour lancer des attaques par déni de service, par exemple en exécutant des boucles infinies ou en consommant toutes les ressources système disponibles.

```
kheira@LaPlateforme: ~/SystemeScriptSecurite
GNU nano 7.2 script_logiciels.sh *
#!/bin/bash

bash "./maj.sh"

choice=0
choice2=0

while [[ "$choice" != "1" && "$choice" != "2" ]]; do

    read -n 1 -p "Souhaitez-vous installer Apache (1) ou Nginx (2) ?" choice

    if [[ "$choice" != "1" && "$choice" != "2" ]]; then
        echo "Option invalide, veuillez choisir 1 ou 2"
    fi
done

case "$choice" in
    [1])
        echo "Installation d'Apache2"
        sudo apt install -y apache2
        ;;
    [2])
        echo "Installation de Nginx"
        sudo apt install -y nginx
        ;;
esac

while [[ "$choice2" != "1" && "$choice2" != "2" ]]; do

    read -n 1 -p "Souhaitez-vous installer MySQL (1) ou Mariadb (2) ?" choice2

    if [[ "$choice2" != "1" && "$choice2" != "2" ]]; then
        echo "Option invalide, veuillez choisir 1 ou 2"
    fi
done
```



```
GNU nano 7.2                               kheira@LaPlateforme: ~/SystemeScriptSecurite
#!/bin/bash

update_packages() {
    echo " Mise à jour des paquets en cours !"
    sudo apt update && sudo apt upgrade -y
}

echo " Recherche de mises à jour en cours ! "
sudo apt update > /dev/null 2>&1

if [ $? -eq 0 ]; then
    echo " des mises à jours sont disponibles ! "
    read -n 1 -p " Souhaitez vous mettre à jour ? (O/N) ? " choice
    case "$choice" in
        [oO])
            echo "Lancement des mises à jour"
            update_packages
            ;;
        [nN])
            echo "Mise à jour annulé !"
            ;;
        *)
            echo "Choix invalide, mise à jour annulé ! "
            ;;
    esac
else
    echo "Pas de mises à jour disponible"
fi
```



11/ Utilisation d'API Web dans un script

Création d'un script Shell afin d'exploiter les données d'une API Web

Sécurisation de la communication avec l'API.

Inclusion d'une gestion d'erreurs afin d'anticiper tout comportement inattendu en cas de problème avec l'API.

Utilisation du logging et enregistrement des différentes requêtes vers l'API ainsi que les réponses envoyées par l'API.

Commandes :

Pour exploiter les données d'une API Web il nous faut :

- L'URL de l'API
- La clé d'API
- L'ID de recherche
- détail du script:

```
GNU nano 7.2 api_script.sh
#!/bin/bash

source "/home/hugo/secret.cfg"

url="https://api.football-data.org/v4/competitions/FL1/matches"

start_of_week=$(date -d "last Monday" "+%Y-%m-%d")
end_of_week=$(date -d "next Sunday" "+%Y-%m-%d")

response=$(curl -s "${url}?dateFrom=${start_of_week}&dateTo=${end_of_week}" \
-H "X-Auth-Token: ${API_KEY}")

if [[ ${response} =~ "<html>" ]]; then
    echo "Erreur de récupération des données, veuillez vérifier votre connection internet ou votre clé API."
    exit 1
fi

match_list=$(echo "${response}" | jq -r '.matches[] | .homeTeam.name, .awayTeam.name, .utcDate')

echo "Liste des matchs de la Ligue 1 de la semaine du ${start_of_week} au ${end_of_week}:"
echo "-----"
for entry in $(echo "${match_list}"); do
    echo "${entry}"
done
```

`source "/home/hugo/secret.cfg"`: Cette ligne source un fichier nommé "secret.cfg" situé dans le répertoire "/home/hugo/", ce qui signifie qu'il va exécuter le contenu de ce fichier dans le script en cours. Cela est couramment utilisé pour charger des configurations sensibles ou des variables d'environnement. Dans ce fichier se trouve la clé API au format : `API_KEY="XXXXXXXXXXXXXXXXXXXX"`

`url="https://api.football-data.org/v4/competitions/FL1/matches"`: Cette ligne définit une variable nommée "url" qui contient l'URL de l'API de football utilisée pour récupérer les données des matchs de la Ligue 1.



```
response=$(curl  
-s "${url}?dateFrom=${start_of_week}&dateTo=${end_of_week}" \
```

: Cette ligne utilise la commande `curl` pour envoyer une requête HTTP à l'URL de l'API de football, spécifiant les dates de début et de fin de la semaine en cours comme paramètres de requête. Le résultat de la requête est stocké dans la variable "response". L'option "-s" est utilisée pour supprimer la sortie de progression de `curl`

```
-H "X-Auth-Token: ${API_KEY}"): Cette partie de la ligne curl envoie un en-tête HTTP avec la clé d'authentification API stockée dans la variable "API_KEY".
```

```
if [[ ${response} =~ "<html>" ]]; then: Cette ligne vérifie si la réponse de l'API contient "<html>", ce qui pourrait indiquer une erreur de récupération des données plutôt que les données de match attendues.
```

```
match_list=$(echo "${response}" | jq -r '.matches[] |  
.homeTeam.name, .awayTeam.name, .utcDate'): Cette ligne utilise jq, un outil de ligne de commande pour traiter des données JSON, pour extraire et formater les noms des équipes à domicile et à l'extérieur ainsi que la date des matchs de la réponse JSON de l'API. Le résultat est stocké dans la variable "match_list".
```