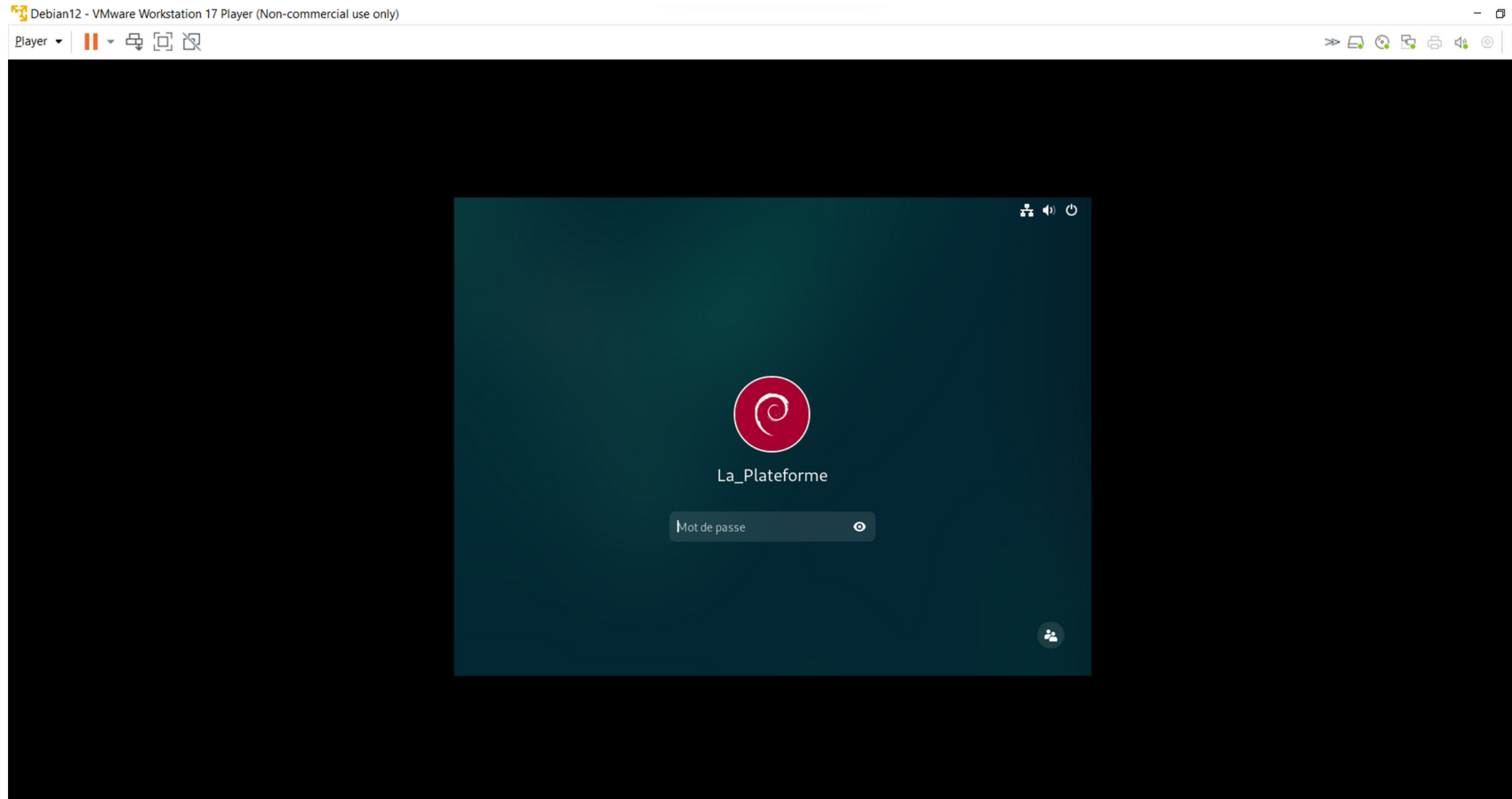


création d'une VM avec nom de session



commandes de recherche avancée

```
laplateforme@LaPlateforme:~$ echo "Que la force soit avec toi." > Documents/mon_texte.txt
laplateforme@LaPlateforme:~$ echo "Que la force soit avec toi." > Téléchargements/mon_texte.txt
laplateforme@LaPlateforme:~$ echo "Que la force soit avec toi." > Vidéos/mon_texte.txt
laplateforme@LaPlateforme:~$ echo "Que la force soit avec toi." > Images/mon_texte.txt
laplateforme@LaPlateforme:~$ grep -r "force" .
grep: ../.cache/gnome-software/appstream/components.xmlb : fichiers binaires correspondent
../.cache/gnome-software/odrs/ratings.json:      "io.github.hmlendea.geforcenow-electron": {
../.cache/gnome-software/odrs/ratings.json:      "io.github.theforceengine.tfe": {
../.cache/gnome-software/odrs/ratings.json:      "io.snapcraft.geforce-now-desktop-DxfnMqxqb58VizlP73LYFARQAzgsHtNn": {
../.cache/gnome-software/odrs/ratings.json:      "io.snapcraft.geforcenow-D0GhSSTbJyx4LV9sjd12L0qRaG4nbWDN": {
../.cache/gnome-software/odrs/ratings.json:      "io.snapcraft.geforcenow-electron-KwB1ULb6XF0c997eR1kJTG4d7efIdRJ2": {
grep: ../.cache/tracker3/files/http%3A%2F%2Ftracker.api.gnome.org%2Fontology%2Fv3%2Ftracker%23Documents.db-wal : fichiers binaires correspondent
grep: ../.cache/tracker3/files/meta.db : fichiers binaires correspondent
../.bashrc:#force_color_prompt=yes
../.bashrc:if [ -n "$force_color_prompt" ]; then
../.bashrc:unset color_prompt force_color_prompt
../Documents/mon_texte.txt:Que la force soit avec toi.
../Téléchargements/mon_texte.txt:Que la force soit avec toi.
../Images/mon_texte.txt:Que la force soit avec toi.
../Vidéos/mon_texte.txt:Que la force soit avec toi.
../.local/share/Trash/files/mon_texte.3.txt:Que la force soit avec toi. /root
../.local/share/Trash/files/mon_texte.2.txt:Que la force soit avec toi. /root
../.local/share/Trash/files/mon_texte.txt:Que la force soit avec toi. /root
../.local/share/Trash/files/mon_texte.4.txt:Que la force soit avec toi. /root
../.local/share/Trash/files/mon_texte.5.txt:Que la force soit avec toi. /root
../.local/share/Trash/files/mon_texte2.txt:Que la force soit avec toi.
../Bureau/mon_texte.txt:Que la force soit avec toi.
laplateforme@LaPlateforme:~$ █
```

Créer le répertoire "Plateforme" dans le dossier Document.
Dupliquer le fichier.txt

```
laplateforme@LaPlateforme:~$ mkdir /home/laplateforme/Documents/Plateforme
laplateforme@LaPlateforme:~$ cp ~/Documents/mon_texte.txt ~/Documents/Plateforme/
laplateforme@LaPlateforme:~$ cd ~/Documents/Plateforme
laplateforme@LaPlateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte2.txt
laplateforme@LaPlateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte3.txt
laplateforme@LaPlateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte4.txt
laplateforme@LaPlateforme:~/Documents/Plateforme$ cp mon_texte.txt mon_texte5.txt
laplateforme@LaPlateforme:~/Documents/Plateforme$ █
```

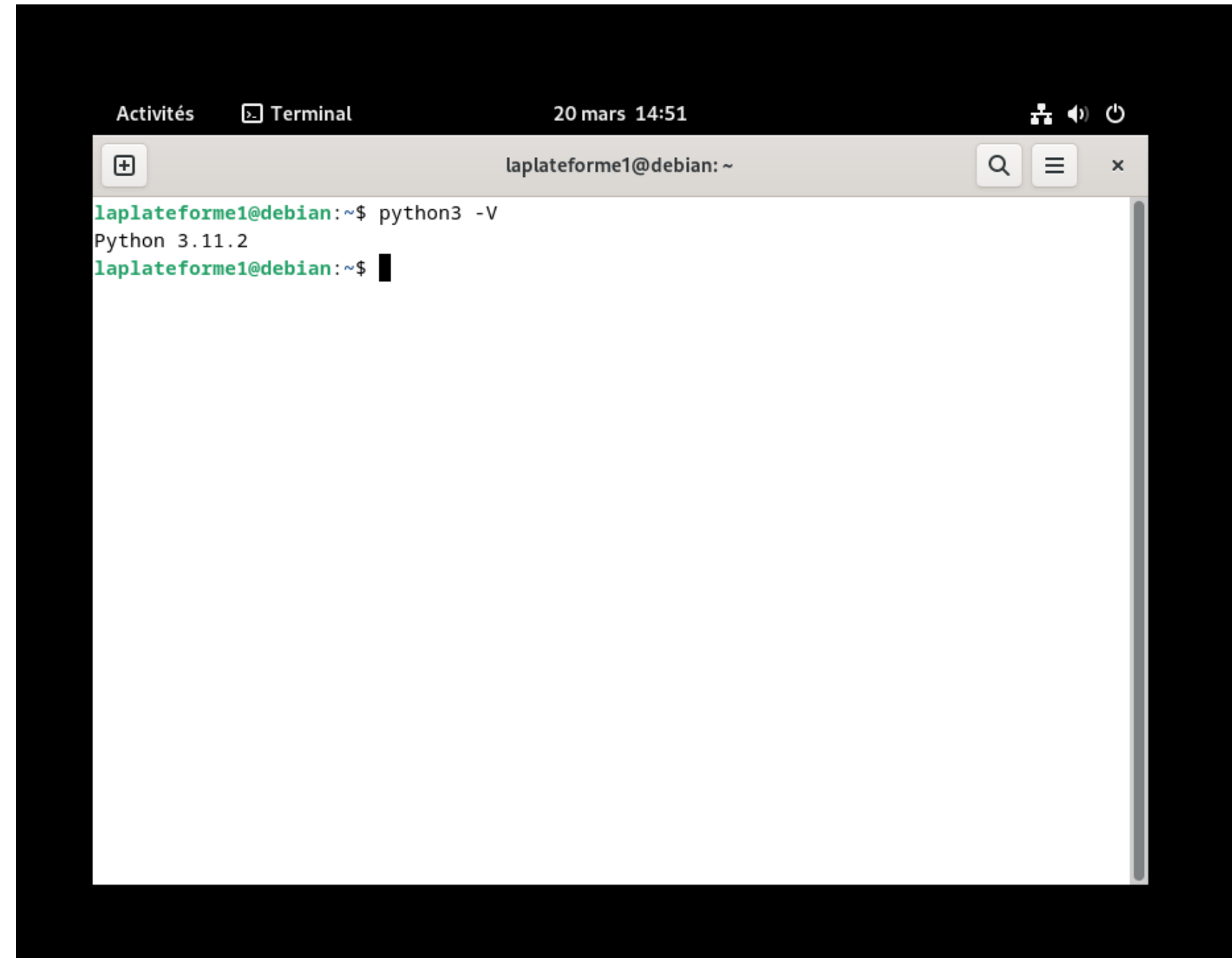
Compression

```
laplateforme@LaPlateforme:~/Documents/Plateforme$ tar -czvf Plateforme.tar.gz ~/Documents/Plateforme
tar: Suppression de « / » au début des noms des membres
/home/laplateforme/Documents/Plateforme/
/home/laplateforme/Documents/Plateforme/mon_texte.txt
/home/laplateforme/Documents/Plateforme/mon_texte4.txt
/home/laplateforme/Documents/Plateforme/mon_texte2.txt
/home/laplateforme/Documents/Plateforme/mon_texte3.txt
/home/laplateforme/Documents/Plateforme/mon_texte5.txt
```

Décompression

```
laplateforme@LaPlateforme:~/Documents/Plateforme$ tar -xzf Plateforme.tar.gz
home/laplateforme/Documents/Plateforme/
home/laplateforme/Documents/Plateforme/mon_texte.txt
home/laplateforme/Documents/Plateforme/mon_texte4.txt
home/laplateforme/Documents/Plateforme/mon_texte2.txt
home/laplateforme/Documents/Plateforme/mon_texte3.txt
home/laplateforme/Documents/Plateforme/mon_texte5.txt
laplateforme@LaPlateforme:~/Documents/Plateforme$ █
```


Installation de python avec vérification de la version



A screenshot of a Linux terminal window. The window title bar shows 'Activités', 'Terminal', and the date '20 mars 14:51'. The terminal prompt is 'laplateforme1@debian: ~'. The user has entered the command 'python3 -V', and the output is 'Python 3.11.2'. The prompt is now ready for the next command.

```
laplateforme1@debian:~$ python3 -V
Python 3.11.2
laplateforme1@debian:~$
```

Manipulation de texte



```
Activités Terminal 20 mars 15:33
```

```
laplateforme1@debian: ~
```

```
GNU nano 7.2 Nouvel espace *
```

```
Jean, 25 ans, Paris
```

```
Marie, 30 ans, Lyon
```

```
Pierre, 22 ans, Marseille
```

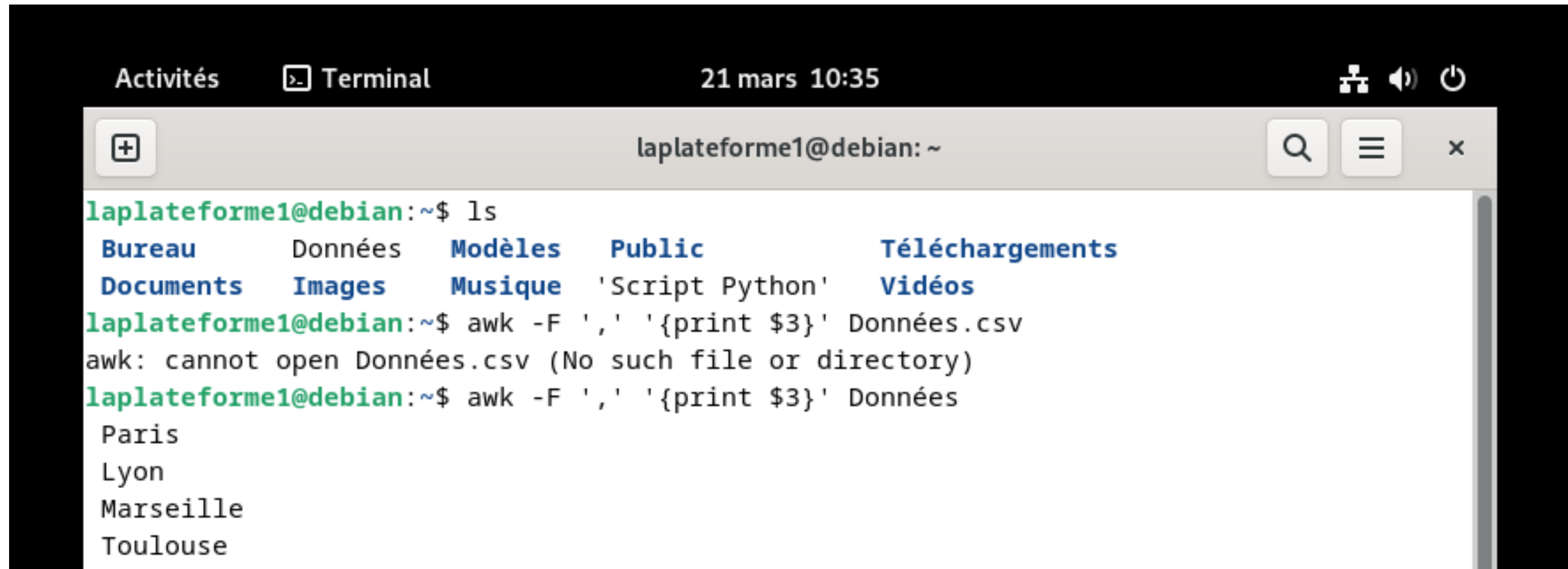
```
Sophie, 35 ans, Toulouse
```

```
Sauver l'espace modifié ?
```

```
O Oui
```

```
N Non ^C Annuler
```

Manipulation de texte en utilisant la commande “awk”



A terminal window titled "laplateforme1@debian: ~" showing the execution of the 'ls' and 'awk' commands. The 'ls' command lists the contents of the home directory, including 'Données'. The 'awk' command is used to print the third column of 'Données.csv', resulting in a list of cities: Paris, Lyon, Marseille, and Toulouse.

```
laplateforme1@debian:~$ ls
Bureau      Données    Modèles    Public      Téléchargements
Documents   Images     Musique    'Script Python'  Vidéos

laplateforme1@debian:~$ awk -F ',' '{print $3}' Données.csv
awk: cannot open Données.csv (No such file or directory)

laplateforme1@debian:~$ awk -F ',' '{print $3}' Données
Paris
Lyon
Marseille
Toulouse
```


Gestions des processus

```
la_plateforme@debian: ~  
la_plateforme@debian:~$ ps aux  
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
root         1  0.0  0.6 102484 12484 ?        Ss   15:22   0:00 /sbin/init  
root         2  0.0  0.0      0     0 ?        S    15:22   0:00 [kthreadd]  
root         3  0.0  0.0      0     0 ?        I<   15:22   0:00 [rcu_gp]  
root         4  0.0  0.0      0     0 ?        I<   15:22   0:00 [rcu_par_gp]  
root         5  0.0  0.0      0     0 ?        I<   15:22   0:00 [slub_flushwq]  
root         6  0.0  0.0      0     0 ?        I<   15:22   0:00 [netns]  
root         8  0.0  0.0      0     0 ?        I<   15:22   0:00 [kworker/0:0H-events  
root        10  0.0  0.0      0     0 ?        I<   15:22   0:00 [mm_percpu_wq]  
root        11  0.0  0.0      0     0 ?        I    15:22   0:00 [rcu_tasks_kthread]  
root        12  0.0  0.0      0     0 ?        I    15:22   0:00 [rcu_tasks_rude_kthr  
root        13  0.0  0.0      0     0 ?        I    15:22   0:00 [rcu_tasks_trace_kth  
root        14  0.0  0.0      0     0 ?        S    15:22   0:00 [ksoftirqd/0]  
root        15  0.0  0.0      0     0 ?        I    15:22   0:00 [rcu_preempt]  
root        16  0.0  0.0      0     0 ?        S    15:22   0:00 [migration/0]  
root        18  0.0  0.0      0     0 ?        S    15:22   0:00 [cpuhp/0]  
root        20  0.0  0.0      0     0 ?        S    15:22   0:00 [kdevtmpfs]  
root        21  0.0  0.0      0     0 ?        I<   15:22   0:00 [inet_frag_wq]  
root        22  0.0  0.0      0     0 ?        S    15:22   0:00 [kauditd]  
root        23  0.0  0.0      0     0 ?        S    15:22   0:00 [khungtaskd]  
root        24  0.0  0.0      0     0 ?        S    15:22   0:00 [oom_reaper]  
root        27  0.0  0.0      0     0 ?        I<   15:22   0:00 [writeback]  
root        28  0.0  0.0      0     0 ?        S    15:22   0:00 [kcompactd0]  
root        29  0.0  0.0      0     0 ?        SN   15:22   0:00 [ksmd]
```

```
la_plateforme@debian: ~  
top - 16:06:02 up 43 min,  1 user,  load average: 0.00, 0.00, 0.00  
Tasks: 155 total,  1 running, 154 sleeping,  0 stopped,  0 zombie  
%Cpu(s):  0.7 us,  0.7 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st  
MiB Mem :  1967.3 total,  553.9 free,  882.4 used,  695.6 buff/cache  
MiB Swap:  975.0 total,  975.0 free,  0.0 used. 1084.9 avail Mem  
  
   PID USER      PR  NI  VIRT  RES  SHR S  %CPU  %MEM    TIME+  COMMAND  
  1196 la_plat+  20   0 3226904 268032 127960 S   2.0  13.3   0:11.27 gnome-shell  
  2075 la_plat+  20   0  11600   5104   3204 R   0.3   0.3   0:00.01 top  
     1 root      20   0  102484  12484   9176 S   0.0   0.6   0:00.67 systemd  
     2 root      20   0      0      0      0 S   0.0   0.0   0:00.00 kthreadd  
     3 root       0 -20      0      0      0 I   0.0   0.0   0:00.00 rcu_gp  
     4 root       0 -20      0      0      0 I   0.0   0.0   0:00.00 rcu_par_gp  
     5 root       0 -20      0      0      0 I   0.0   0.0   0:00.00 slub_flushwq  
     6 root       0 -20      0      0      0 I   0.0   0.0   0:00.00 netns  
     8 root       0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H-e+  
    10 root       0 -20      0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq  
    11 root      20   0      0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_kthr+  
    12 root      20   0      0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_rude+  
    13 root      20   0      0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_trac+  
    14 root      20   0      0      0      0 S   0.0   0.0   0:00.11 ksoftirqd/0  
    15 root      20   0      0      0      0 I   0.0   0.0   0:00.35 rcu_preempt  
    16 root      rt    0      0      0      0 S   0.0   0.0   0:00.02 migration/0  
    18 root      20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
```

Surveillance des ressources système

Activities Terminal Mar 22 12:11

la_plateforme@debian: ~

la_plateforme@debian:~\$ sh surveillance-ressources-système.sh

la_plateforme@debian:~\$

Activities Terminal Mar 22 14:23

la_plateforme@debian: ~

CPU[| 2.0%] Tasks: 108, 246 thr, 57 kthr; 1 running

Mem[| 1.19G/1.92G] Load average: 0.06 0.23 0.18

Swp[| 10.6M/975M] Uptime: 00:12:38

Main I/O

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1203	la_platefo	20	0	3180M	278M	132M	S	0.7	14.2	0:53.66	/usr/bin/gnome-shell
3217	la_platefo	20	0	8160	4248	3360	R	0.7	0.2	0:00.12	htop
1	root	20	0	164M	12448	9204	S	0.0	0.6	0:00.59	/sbin/init
221	root	20	0	49904	20552	18916	S	0.0	1.0	0:00.20	/lib/systemd/systemd
250	root	20	0	27812	6120	3356	S	0.0	0.3	0:00.07	/lib/systemd/systemd
264	systemd-ti	20	0	90224	6956	6004	S	0.0	0.3	0:00.03	/lib/systemd/systemd
378	systemd-ti	20	0	90224	6956	6004	S	0.0	0.3	0:00.00	/lib/systemd/systemd
441	root	20	0	231M	7492	6568	S	0.0	0.4	0:00.04	/usr/libexec/account
445	avahi	20	0	8288	3492	3124	S	0.0	0.2	0:00.04	avahi-daemon: runnin
448	root	20	0	6608	2656	2404	S	0.0	0.1	0:00.00	/usr/sbin/cron -f
449	messagebus	20	0	11752	6432	3964	S	0.0	0.3	0:00.35	/usr/bin/dbus-daemon
451	root	-2	0	224M	5460	4964	S	0.0	0.3	0:00.01	/usr/libexec/low-mem
453	polkitd	20	0	303M	12172	7096	S	0.0	0.6	0:00.22	/usr/lib/polkit-1/po
454	root	20	0	231M	6632	6016	S	0.0	0.3	0:00.02	/usr/libexec/power-p
455	root	20	0	227M	6276	5760	S	0.0	0.3	0:00.01	/usr/libexec/switche
457	root	20	0	25308	7836	6796	S	0.0	0.4	0:00.08	/lib/systemd/systemd

F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice -F8Nice +F9Kill F10Quit

la_plateforme@debian: ~

top - 16:06:02 up 43 min, 1 user, load average: 0.00, 0.00, 0.00


Tasks: 155 total, 1 running, 154 sleeping, 0 stopped, 0 zombie




%Cpu(s): 0.7 us, 0.7 sy, 0.0 ni, 98.3 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st

MiB Mem : 1967.3 total, 553.9 free, 882.4 used, 695.6 buff/cache

MiB Swap: 975.0 total, 975.0 free, 0.0 used. 1084.9 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1196	la_plat+	20	0	3226904	268032	127960	S	2.0	13.3	0:11.27	gnome-shell
2075	la_plat+	20	0	11600	5104	3204	R	0.3	0.3	0:00.01	top
1	root	20	0	102484	12484	9176	S	0.0	0.6	0:00.67	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_flushwq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-e+
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
11	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_kthr+
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_rude+
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_trac+
14	root	20	0	0	0	0	S	0.0	0.0	0:00.11	ksoftirqd/0
15	root	20	0	0	0	0	I	0.0	0.0	0:00.35	rcu_preempt
16	root	rt	0	0	0	0	S	0.0	0.0	0:00.02	migration/0
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0

Activities Text Editor Mar 22 11:55 

Open ▾  • surveillance-ressources-système.sh  

fichier.txt • surveillance-ressources-système.sh

```
#!/bin/bash

echo "Horodatage,CPU(%),Memoire(utilisee/total),Swap(utilise/total)" > system_resources.csv

for i in $(seq 0 59); do
    horodatage=$(date +"%Y-%m-%d %H:%M:%S")
    cpu=$(top -bn1 | grep "Cpu(s)" | sed "s/.*, *\([0-9.]*\)%* id.*\/\1/" | awk '{print 100 - $1}')
    memoire=$(free | grep Mem | awk '{print $3/"$2}')}
    swap=$(free | grep Swap | awk '{print $3/"$2}')}
    echo "$horodatage,$cpu,$memoire,$swap" >> system_resources.csv
    sleep 1
done
```

fichier .csv.png

Text Import - [system_resources.csv]

Import

Character set:

Unicode (UTF-8)

Locale:

Default - English (USA)

From row:

1

−

+

Separator Options

Fixed width

Separated by

✓

Tab

✓

Comma

✓

Semicolon

Space

Other

Merge delimiters

Trim spaces

String delimiter:

"

Other Options

Format quoted field as text

Detect special numbers

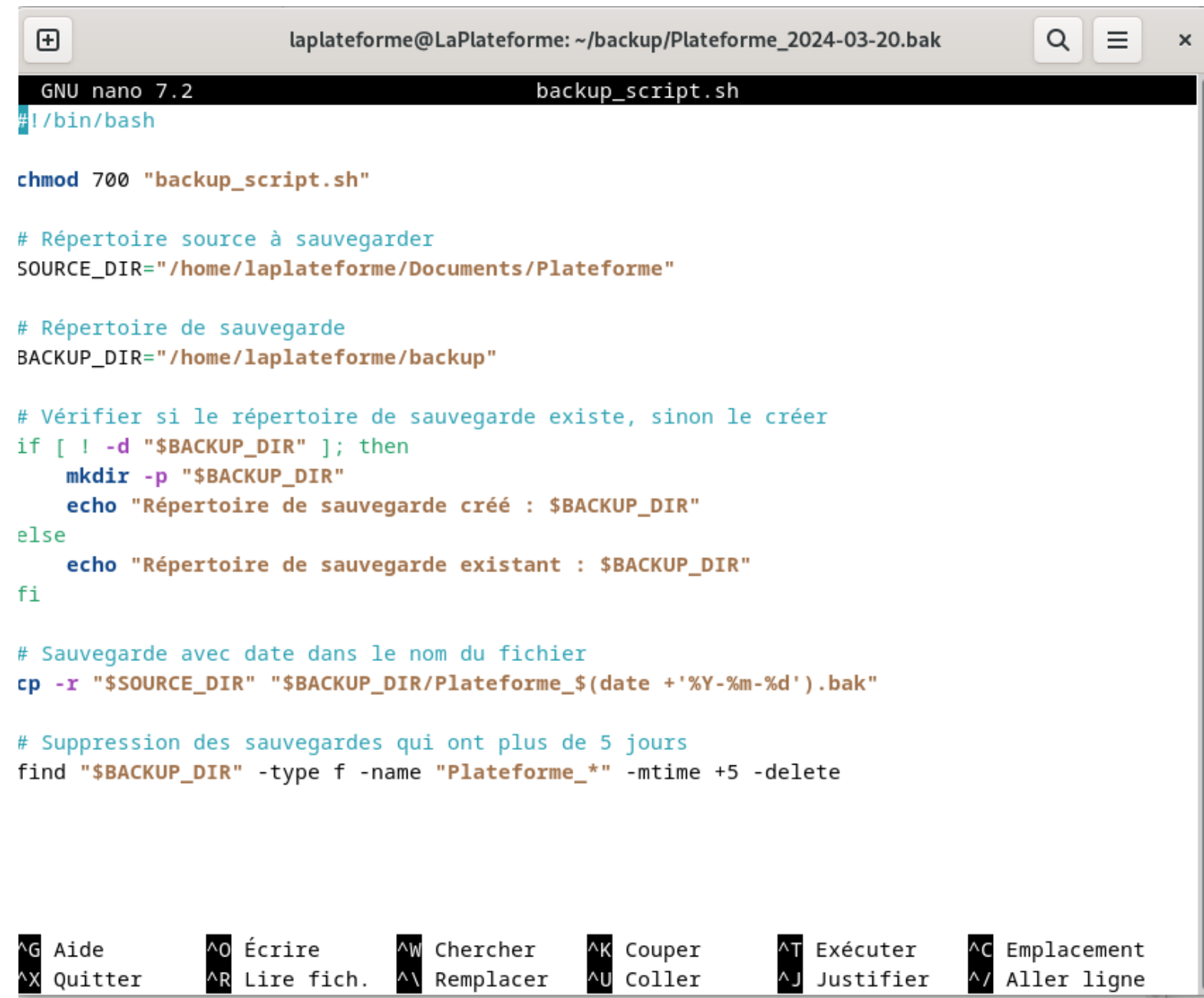
Evaluate formulas

Fields

Column type:

	Standard	Standard	Standard	Standard	
1	Horodatage	CPU(%)	Memoire(utilisee/total)	Swap(utilise/total)	
2	2024-03-22 11:51:03	0	1610800/2014512	646060/998396	
3	2024-03-22 11:51:04	0	1610732/2014512	646060/998396	
	2024-03-22 11:51:05	0	1610732/2014512	646060/998396	

scripting avancé



```
laplateforme@LaPlateforme: ~/backup/Plateforme_2024-03-20.bak
GNU nano 7.2 backup_script.sh
#!/bin/bash

chmod 700 "backup_script.sh"

# Répertoire source à sauvegarder
SOURCE_DIR="/home/laplateforme/Documents/Plateforme"

# Répertoire de sauvegarde
BACKUP_DIR="/home/laplateforme/backup"

# Vérifier si le répertoire de sauvegarde existe, sinon le créer
if [ ! -d "$BACKUP_DIR" ]; then
    mkdir -p "$BACKUP_DIR"
    echo "Répertoire de sauvegarde créé : $BACKUP_DIR"
else
    echo "Répertoire de sauvegarde existant : $BACKUP_DIR"
fi

# Sauvegarde avec date dans le nom du fichier
cp -r "$SOURCE_DIR" "$BACKUP_DIR/Plateforme_$(date +%Y-%m-%d').bak"

# Suppression des sauvegardes qui ont plus de 5 jours
find "$BACKUP_DIR" -type f -name "Plateforme_*" -mtime +5 -delete
```

^G Aide ^O Écrire ^W Chercher ^K Couper ^T Exécuter ^C Emplacement
^X Quitter ^R Lire fich. ^\ Remplacer ^U Coller ^J Justifier ^_ Aller ligne

Automatisation des mises à jour logicielles



The screenshot shows a terminal window with a title bar indicating the user is 'laplateforme' on a machine named 'LaPlateforme'. The terminal is running the 'nano' text editor, editing a file named 'mise_a_jour_logiciels.sh'. The script content is as follows:

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

chmod 700 "mise_a_jour_logiciels.sh"

# Informer l'utilisateur que la recherche de mises à jour va commencer
echo "Recherche de mises à jour disponibles..."

# Mettre à jour les informations des dépôts
sudo apt update > /dev/null

# Vérifier si des mises à jour sont disponibles
if sudo apt list --upgradable | grep -qE '[:,alnum:]'; then
    # Afficher les mises à jour disponibles
    echo "Des mises à jour sont disponibles :"
    sudo apt list --upgradable

    # Proposer à l'utilisateur de mettre à jour les logiciels
    read -p "Voulez-vous mettre à jour les logiciels ? (o/n) : " response
    if [ "$response" = "o" ] || [ "$response" = "O" ]; then
        # Mettre à jour les logiciels
        echo "Mise à jour en cours..."
        sudo apt upgrade -y
        echo "Mise à jour terminée."
    else
        echo "Mise à jour annulée."
    fi
else
    echo "Aucune mise à jour disponible."
fi
```

At the bottom of the terminal, there is a status bar with various keyboard shortcuts for navigating and editing the file:

^G Aide	^O Écrire	^W Chercher	^K Couper	^T Exécuter	^C Emplacement	M-U Annuler
^X Quitter	^R Lire fich.	^I Remplacer	^U Coller	^J Justifier	^_ Aller ligne	M-E Refaire

Gestion des dépendances logicielles en nano avec sécurisation du script



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

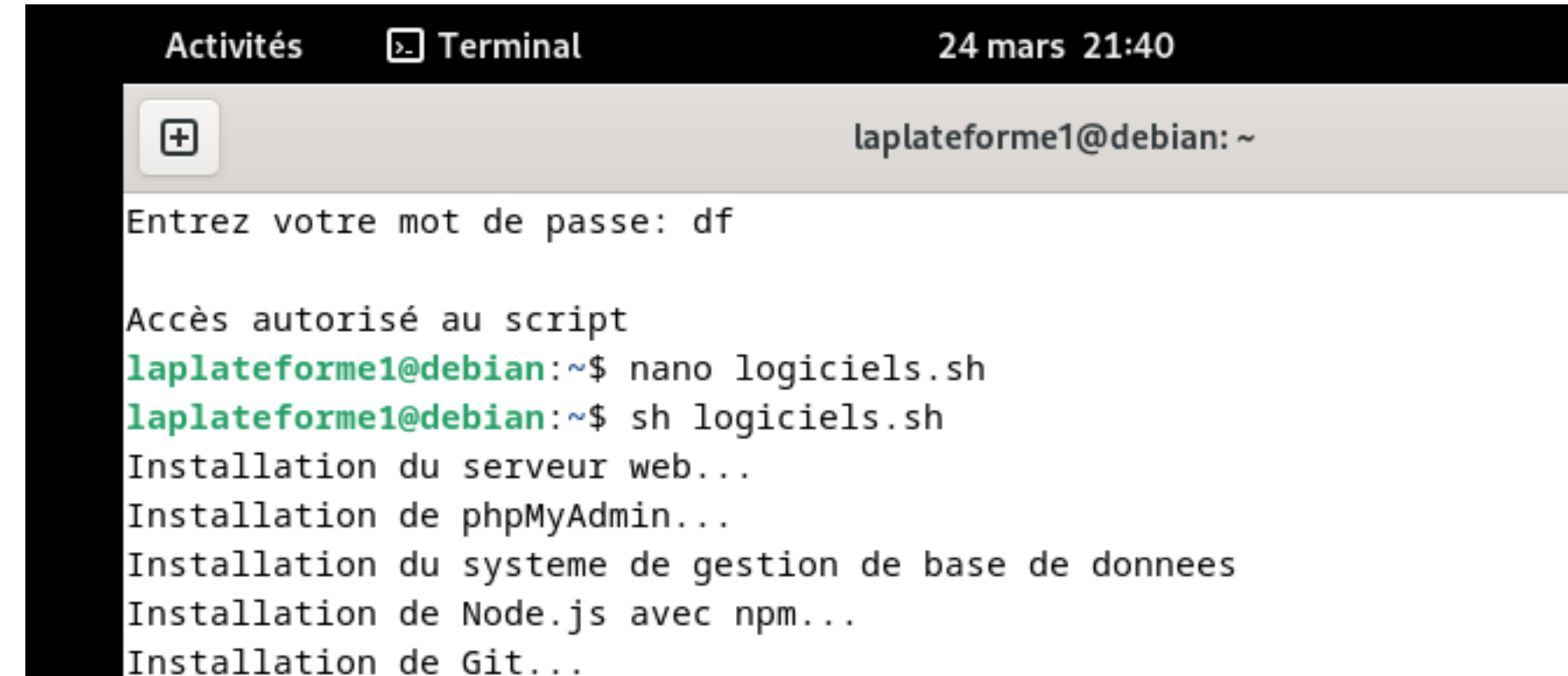
chmod 700 logiciels.sh

command_exists() {
command -v "$1" >/dev/null 2>&1
}

if ! command_exists apache2 && ! command_exists nginx;
then
    echo "Installation du serveur web..."
fi
if ! command_exists phpmyadmin;
then
    echo "Installation de phpMyAdmin..."
fi
if ! command_exists mysql && ! command_exists mariadb;
then
    echo "Installation du systeme de gestion de base de donnees"
fi

[ Lecture de 66 lignes ]
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^_ Aller ligne
```

sur le terminal



```
Activités Terminal 24 mars 21:40
laplateforme1@debian: ~
Entrez votre mot de passe: df

Accès autorisé au script
laplateforme1@debian:~$ nano logiciels.sh
laplateforme1@debian:~$ sh logiciels.sh
Installation du serveur web...
Installation de phpMyAdmin...
Installation du systeme de gestion de base de donnees
Installation de Node.js avec npm...
Installation de Git...
```

Identification les risques liés à la négligence et sécurisation de scripts

Sécuriser ses scripts

Identification des risques liés à la négligence de la sécurité

La négligence de la sécurité peut entraîner une multitude de risques et de conséquences néfastes, en fonction du contexte spécifique.

Voici quelques-uns des risques courants liés à la négligence de la sécurité :

Violations de la confidentialité des données :

La négligence de la sécurité peut conduire à des failles dans la protection des données sensibles, ce qui expose ces informations à des accès non autorisés et à des violations de la confidentialité.

Perte de données :

Une sécurité insuffisante peut entraîner la perte accidentelle ou intentionnelle de données critiques, ce qui peut avoir un impact grave sur les opérations commerciales et la réputation de l'organisation.

Attaques informatiques :

Les cybercriminels exploitent souvent les vulnérabilités de sécurité pour mener des attaques telles que les piratages, les logiciels malveillants et les attaques par déni de service (DDoS), ce qui peut entraîner des dommages financiers, opérationnels et de réputation.

Frais de récupération :

En cas de violation de sécurité, des coûts importants peuvent être engagés pour enquêter sur l'incident, remédier aux dommages, notifier les parties concernées et mettre en place des mesures correctives pour éviter de futures violations.

Non-conformité réglementaire :

Dans de nombreux secteurs, il existe des réglementations strictes en matière de sécurité des données et de protection de la vie privée. La négligence de la sécurité peut entraîner des violations de ces réglementations, ce qui expose l'organisation à des amendes et à d'autres sanctions juridiques.

Perte de confiance des clients et des partenaires :

Les violations de sécurité et les failles de confidentialité peuvent compromettre la confiance des clients et des partenaires commerciaux, ce

qui peut avoir un impact durable sur les relations commerciales et la réputation de l'entreprise.

Interruption des activités commerciales :

Les attaques informatiques et les violations de sécurité peuvent entraîner une interruption des opérations commerciales, ce qui peut entraîner des pertes financières importantes et des perturbations dans la prestation de services.

Vol d'identité :

Les données personnelles exposées lors de violations de sécurité peuvent être utilisées pour commettre des fraudes d'identité, ce qui peut avoir des conséquences financières et juridiques graves pour les individus concernés.

Perte de compétitivité :

Une réputation entachée par des failles de sécurité peut nuire à la capacité d'une organisation à attirer de nouveaux clients, à conclure des partenariats commerciaux et à rivaliser sur le marché.

Domination de l'infrastructure :

Dans certains cas, des attaques sophistiquées peuvent permettre aux attaquants de prendre le contrôle de l'infrastructure informatique d'une organisation, ce qui peut entraîner des dommages étendus et prolongés.

La sécurisation des scripts

La sécurisation des scripts informatiques est une étape cruciale pour prévenir les failles de sécurité et les attaques potentielles. Voici quelques bonnes pratiques pour sécuriser vos scripts :

Validation des entrées utilisateur :

Assurez-vous de valider et de filtrer toutes les données provenant de sources externes (utilisateur, fichiers, etc.) pour éviter les attaques par injection de code (comme les injections SQL, les injections XSS, etc.).

Évitez l'exécution de commandes système directes :

Si vous devez exécuter des commandes système à partir de vos scripts, utilisez des fonctions ou des bibliothèques qui permettent d'effectuer cette opération de manière sécurisée, en évitant les commandes directes qui peuvent être manipulées par des utilisateurs malveillants.

Évitez les privilèges excessifs :

Assurez-vous que vos scripts n'ont que les privilèges nécessaires pour accomplir leur tâche. Évitez d'utiliser des comptes d'administration ou des privilèges root lorsque ce n'est pas nécessaire.

Gestion sécurisée des mots de passe et des informations sensibles :

Si vos scripts manipulent des mots de passe ou d'autres informations sensibles, assurez-vous de les stocker de manière sécurisée, de préférence en utilisant des méthodes de hachage et de salage pour les mots de passe.

Mises à jour régulières :

Assurez-vous de maintenir vos scripts à jour en appliquant les correctifs de sécurité et en mettant à jour les dépendances logicielles utilisées par vos scripts.

Limitier l'accès aux fichiers et aux ressources :

Assurez-vous que vos scripts n'ont accès qu'aux fichiers et aux ressources nécessaires à leur fonctionnement. Limitez les permissions d'accès aux fichiers et utilisez des mécanismes de contrôle d'accès si nécessaire.

Journalisation et surveillance :

Mettez en place des mécanismes de journalisation pour enregistrer les activités des scripts et surveiller les tentatives d'accès non autorisées ou les comportements suspects.

Gestion des erreurs sécurisée :

Évitez de divulguer des informations sensibles dans les messages d'erreur. Assurez-vous que les messages d'erreur ne révèlent pas d'informations sur la structure interne de vos scripts ou sur les données manipulées.

Sécurisation des communications réseau :

Si vos scripts communiquent sur un réseau, assurez-vous que les communications sont sécurisées en utilisant des protocoles sécurisés comme HTTPS, SSH, etc., et en mettant en œuvre des mécanismes de chiffrement adéquats.

Test de sécurité :

Effectuez régulièrement des tests de sécurité sur vos scripts pour identifier et corriger les éventuelles vulnérabilités. Vous pouvez utiliser des outils d'analyse statique et dynamique ainsi que des tests d'intrusion pour évaluer la sécurité de vos scripts.

1

2

3

Utilisation d'API Web dans un script



```
Activités Terminal 24 mars 22:58
laplateforme@LaPlateforme: ~
GNU nano 7.2 API_Web.sh
#!/bin/bash

# Limiter les permissions du fichier
chmod 700 "API_Web.sh"

# Fonction pour effectuer une requête GET à l'API GitHub
function get_user_info() {
    local username="$1"
    local api_url="https://api.github.com/users/$username"

    # Effectuer la requête GET à l'API et afficher les informations de l'utilisateur
    curl -s "$api_url"
}

# Demander à l'utilisateur le nom d'utilisateur GitHub
read -p "Entrez le nom d'utilisateur GitHub : " username

# Appeler la fonction pour récupérer les informations sur l'utilisateur
response=$(get_user_info "$username")

# Vérifier si la requête a réussi
if [[ $response == *"login"* ]]; then
    # Si la requête a réussi, afficher les informations sur l'utilisateur
    echo "$response"
else
    # Si la requête a échoué, afficher un message d'erreur
    echo "Erreur : Impossible de récupérer les informations pour l'utilisateur $username."
fi

# Arrêt immédiat en cas d'échec
set -e

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler  M-A Marquer
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^_ Aller ligne M-E Refaire  M-6 Copier
```