

Cours :

IFT-3830 : A-2023

Chargé de cours :

Daniel Ouimet

Travail :

T.P.-1

Date remise :

Dimanche le 15 octobre 2023

Équipe:

Gueorgui Poklitar

Ion Hincu

1. a.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# pgrep cupsd
926
[root@localhost ~]# _
```

pgrep : Recherche du processus qui renvoie de l'information sur le processus recherchée. Processus « cupsd » nous renvoie 926, qui est également son PID.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# man -s 1 -k process | grep ID
pidof (1)          - find the process ID of a running program.
[root@localhost ~]# pidof cupsd
926
[root@localhost ~]# _
```

2^e option à l'aide de la commande **pidof**, trouvée dans le man section 1, pidof renvoie spécifiquement le PID et donc une commande plus précise.

b.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# ps -eo user,pid,ppid,ni,command
```

Command ps avec les paramètres sur l'utilisateur, le id du processus, le id du parent de ce processus, niceness (facteur de priorité) et commande complète.

Résultat :

```
ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox
avahi        658      1  0 avahi-daemon: running [linux.local]
polkitd      659      1  0 /usr/lib/polkit-1/polkitd --no-debug
root        662      1  0 /usr/lib/systemd/systemd-machined
root        665      1  0 /usr/sbin/SMARTD -n -q never
root        667      1  0 /usr/sbin/mcelog --ignoreudev --daemon --foreground
dbus        671      1  0 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --
libstor+    673      1  0 /usr/bin/lsmd -d
root        674      1  0 /usr/sbin/alsactl -s -n 19 -c -E ALSA_CONFIG_PATH=/etc/alsa/alsactl.com
root        675      1  0 /usr/lib/systemd/systemd-logind
chronyd     678      1  0 /usr/sbin/chronyd
avahi       693      658  0 avahi-daemon: chroot helper
root        698      1  0 /bin/bash /usr/sbin/ksmtuned
root        706      1  0 /usr/sbin/ModemManager
root        707      1  0 /usr/libexec/platform-python -s /usr/sbin/firewalld --nofork --nopid
root        889      1  0 /usr/bin/UBoxDRMClient
root        892      1  0 /usr/sbin/UBoxService --pidfile /var/run/uboxadd-service.sh
root        907      1  0 /usr/sbin/NetworkManager --no-daemon
root        919      1  0 /usr/libexec/platform-python -Es /usr/sbin/tuned -l -P
root        921      1  0 /usr/sbin/sshd -D -oCiphers=aes256-gcm@openssh.com,chacha20-poly1305@op
root        926      1  0 /usr/sbin/cupsd -l
root        931      1  0 /usr/sbin/gssproxy -D
root        1191     1  0 /usr/sbin/rsyslogd -n
root        1202     1  0 /usr/sbin/crond -n
root        1206     1  0 /usr/sbin/atd -f
root        1222     1  0 login -- root
dnsmasq     1350      1  0 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --l
root        1352    1350  0 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --l
root        1373      1  0 /usr/lib/systemd/systemd --user
root        1377    1373  0 (sd-pam)
root        1391    1222  0 -bash
root        1773      2  0 [kworker/0:0-events]
root        1817      2  0 [kworker/0:2-ata_sff]
root        1885      2  0 [kworker/u2:1]
root        1929      2  0 [kworker/0:1-events]
root        1938     698  0 sleep 60
root        1939    1391  0 ps -eo user,pid,ppid,ni,command
[root@localhost ~]#
```

C.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# ps -eo user,pid,ppid,ni,command --sort ni
```

Même commande que dans « 1.b », mais avec « --sort ni » qui trie en ordre le niceness du plus petit au plus grand, plus le niceness est petit plus il est prioritaire.

Voici le résultat :

```
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# ps -eo user,pid,ppid,ni,command --sort ni  
USER      PID     PPID  NI  COMMAND  
root        3         2 -20  [rcu_gp]  
root        4         2 -20  [rcu_par_gp]  
root        5         2 -20  [slub_flushwq]  
root        7         2 -20  [kworker/0:0H-events_highpri]  
root       10         2 -20  [mm_percpu_wq]  
root       20         2 -20  [netns]  
root       24         2 -20  [writeback]  
root       28         2 -20  [crypto]  
root       29         2 -20  [kintegrityd]  
root       30         2 -20  [kblockd]  
root       31         2 -20  [blkcg_punt_bio]  
root       32         2 -20  [tpm_dev_wq]  
root       33         2 -20  [md]  
root       34         2 -20  [edac-poller]  
root       36         2 -20  [kworker/0:1H-kblockd]  
root      116         2 -20  [kthrotld]  
root      117         2 -20  [acpi_thermal_pm]  
root      118         2 -20  [kmpath_rdacd]  
root      119         2 -20  [kaluad]  
root      120         2 -20  [ipv6_addrconf]  
root      121         2 -20  [kstrp]  
root      160         2 -20  [zswap-shrink]  
root     335         2 -20  [iprt-VBoxWQueue]  
root     340         2 -20  [ata_sff]  
root     345         2 -20  [scsi_tmf_0]  
root     347         2 -20  [scsi_tmf_1]  
root     355         2 -20  [scsi_tmf_2]  
root     396         2 -20  [xfsalloc]  
root     397         2 -20  [xfs_mru_cache]  
root     398         2 -20  [xfs-buf/sda1]  
root     399         2 -20  [xfs-conv/sda1]  
root     401         2 -20  [xfs-cil/sda1]  
root     402         2 -20  [xfs-reclaim/sda]  
root     403         2 -20  [xfs-gc/sda1]  
root     405         2 -20  [xfs-log/sda1]  
root     612         2 -20  [ext4-rsv-conver]  
root     650         2 -20  [rpciod]  
root     651         2 -20  [xpriod]  
root    2101         1 -11  /usr/bin/pulseaudio --start  
root     636         1  -4  /sbin/auditd  
root     638         1  -4  /usr/sbin/sedispatch  
root        0         0   0  /usr/lib/systemd/systemd --switched-root --system --deserialize 17  
root        2         0   0  [kthreadd]  
root        6         2   0  [kworker/0:0-events_power_efficient]  
root        8         2   0  [kworker/0:1-events]  
root        9         2   0  [kworker/u2:0-events_unbound]  
root       11         2   0  [rcu_tasks_rude_]  
root       12         2   0  [rcu_tasks_trace]  
root       13         2   0  [ksoftirqd/0]  
root       14         2   0  [rcu_sched]  
root       15         2   -  [migration/0]  
root       16         2   -  [watchdog/0]  
root       17         2   0  [cpuhp/0]  
root       19         2   0  [kdevtmpfs]  
root       21         2   0  [kauditd]  
root       22         2   0  [khungtaskd]  
root       23         2   0  [oom_reaper]
```

La plus grande priorité, au sommet de la liste nous avons PID 3, de la commande [rcu_gp], il a un niceness de -20 et occupe la première position en termes de priorité.

d.

La commande **ps tree** montre toute l'arborescence des processus.

```
ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox
[root@localhost ~]# ps tree
systemd├── ModemManager─2*[{ModemManager}]
      ├── NetworkManager─2*[{NetworkManager}]
      ├── UBoxDRMClient─3*[{UBoxDRMClient}]
      ├── UBoxService─8*[{UBoxService}]
      ├── alsaactl
      ├── atd
      ├── auditd├── sedispatch
                └── 2*[{auditd}]
      ├── avahi-daemon─avahi-daemon
      ├── chronyd
      ├── crond
      ├── cupsd
      ├── dbus-daemon─{dbus-daemon}
      ├── dnf─{dnf}
      ├── dnsmasq─dnsmasq
      ├── firewalld─{firewalld}
      ├── gssproxy─5*[{gssproxy}]
      ├── ksmtd─sleep
      ├── login─bash─ps tree
      ├── lsmd
      ├── mcelog
      ├── polkitd─5*[{polkitd}]
      ├── rpcbind
      ├── rsyslogd─2*[{rsyslogd}]
      ├── smartd
      ├── sshd
      ├── systemd─{sd-pam}
      ├── systemd-journal
      ├── systemd-logind
      ├── systemd-machine
      ├── systemd-udevd
      └── tuned─4*[{tuned}]
[root@localhost ~]#
```

e.

- Quel est le processus parent de tous les processus? **systemd**
- Quel est le processus parent de la commande que vous avez trouvé en (d)? **bash**

2.

a.

```
ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox
[root@localhost ~]# cd /
[root@localhost /]# mount /dev/sda5 /mnt
[root@localhost /]# ls mnt
lost+found
[root@localhost /]#
```

Monte la partition sda5 sur /mnt. Le montage est trouvé.

```
ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox
[root@localhost /]# cd /mnt
[root@localhost mnt]# vi toto.txt
```

Création du fichier texte toto, dans le /mnt.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# mount /dev/sda5 /mnt
[root@localhost ~]# ls mnt
lost+found toto toto.txt
[root@localhost ~]# umount /dev/sda5 /mnt
umount: /mnt: not mounted.
[root@localhost ~]# ls mnt
[root@localhost ~]#
```

Test pour vérifier si le fichier toto.txt est bien monté sur /mnt. Toto est trouvé. On procède donc avec le un-mount (umount)... /mnt n'est plus monté.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# find -name toto.txt
./opt/toto.txt
[root@localhost ~]#
```

Le fichier peut être remonté de la même façon sur opt, voici un test pour voir s'il est bien monté sur /opt. Normalement `find -name toto.txt` va montrer tous les répertoires où toto.txt se trouve. On note bien qu'il n'est plus monté sur /mnt, mais montre seulement /opt. Nous avons donc réussi à démonter le fichier et le remonter ailleurs.

b.

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# fdisk -l
Disk /dev/sda: 15.3 GiB, 16444817408 bytes, 32118784 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xba8b6534

Device      Boot      Start         End      Sectors   Size Id Type
/dev/sda1                2048    23070719   23068672    11G 83 Linux
/dev/sda2            23070720    27922431   4851712    2.3G 82 Linux swap / Solaris
/dev/sda3      *    27922432    30019583   2097152    1G 83 Linux
/dev/sda4            30019584    32118783   2099200    1G  5 Extended
/dev/sda5            30023680    32118783   2095104   1023M 83 Linux
[root@localhost ~]#
```

La commande `fdisk -l`, montre toutes les informations nécessaires. Nous avons l'information sur le secteur du disque, qui décrivent les partitions sda1 à sda5, incluant sda3. La grandeur en Gbytes et le type de partition. La partition sda2 contient la mémoire virtuelle (swap).

ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox

```
[root@localhost ~]# tune2fs -l /dev/sda5 | grep "Inode count"
Inode count:                65536
[root@localhost ~]#
```

La partition sda5 peut contenir jusqu'à 65,536 fichiers, maximum. 1 inode = 1 espace fichier.

c.

```
ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox
[root@localhost ~]# e2label /dev/sda5 export
[root@localhost ~]# e2label /dev/sda5
export
[root@localhost ~]# _
```

La première commande donne le nom d'export et puis la deuxième commande renvoi le nom donné précédemment, est donc une façon de vérifier...

3.

a.

```
ift3830-AlmaLinux8-A2023 Clone [Running] - Oracle VM VirtualBox
[root@localhost ~]# systemctl set-default graphical.target
Removed /etc/systemd/system/default.target.
Created symlink /etc/systemd/system/default.target → /usr/lib/systemd/system/graphical.target.
[root@localhost ~]# _
```

Nous changeons le runlevel de défaut pour le runlevel 5, qui est le mode graphique. Puis suivi par la commande reboot, pour activer la modification. Comme démontré dans les prochaines questions nous sommes bien dans le mode graphique pour la suite du TP.

b.

```
root@localhost:~
File Edit View Search Terminal Help
[root@localhost ~]# systemctl list-units --type=target --state=active | wc -l
32
[root@localhost ~]# systemctl list-units --type=service --state=active | wc -l
71
[root@localhost ~]# █
```

Résultat des commandes avec paramètres, il y a donc 32 targets avec statue « actif » et 71 services avec statue « actif ».

c.

```
root@localhost:~
File Edit View Search Terminal Help
[root@localhost ~]# systemctl status cups
● cups.service - CUPS Scheduler
   Loaded: loaded (/usr/lib/systemd/system/cups.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-10-04 20:58:37 EDT; 1 day 22h ago
     Docs: man:cupsd(8)
    Main PID: 973 (cupsd)
      Status: "Scheduler is running..."
        Tasks: 1 (limit: 50548)
      Memory: 3.1M
    CGroup: /system.slice/cups.service
            └─973 /usr/sbin/cupsd -l

Oct 04 20:58:36 localhost.localdomain systemd[1]: Starting CUPS Scheduler...
Oct 04 20:58:37 localhost.localdomain systemd[1]: Started CUPS Scheduler.
Oct 04 21:00:24 localhost.localdomain cupsd[973]: REQUEST localhost - - "POST / HTTP/1.1" 200 360 Create-Printer-Subscriptions successful-ok
Oct 04 21:00:25 localhost.localdomain cupsd[973]: REQUEST localhost - - "POST / HTTP/1.1" 200 151 Cancel-Subscription client-error-not-found
Oct 04 23:05:05 localhost.localdomain cupsd[973]: REQUEST localhost - - "POST / HTTP/1.1" 200 183 Renew-Subscription client-error-not-found
[root@localhost ~]# █
```

Avec cette commande, nous obtenons un résumé qui contient toutes les informations qu'on recherche. Cups est effectivement actif, depuis 1jour et 22heurs et occuper 3.1M de mémoire. Lors de cet essai, le PID est 973.

d.

```
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# journalctl -k -p crit | grep "kernel" | wc -l  
1  
[root@localhost ~]# journalctl -k -p crit  
-- Logs begin at Wed 2023-10-04 20:58:09 EDT, end at Fri 2023-10-06 19:18:45 EDT. --  
Oct 04 20:58:10 localhost.localdomain kernel: Warning: Unmaintained hardware is detected: e1000:100E:8056 @ 0000:00:03.0  
[root@localhost ~]#
```

La première commande nous montre qu'il y a une erreur critique provenant du kernel.
La deuxième nous montre le log de cette erreur spécifique.

e.

```
[root@localhost ~]# cd /etc/init.d  
[root@localhost init.d]# vi monservice_
```

Création du script monservice.

```
File Edit View Search Terminal Help  
#!/bin/bash  
  
# chkconfig: 235 10  
  
# description: message lors du démarrage  
  
echo "Démarrage de mon ordi IFT3830"  
  
exit 0  
~  
~  
~  
~  
~
```

Le script est donné.

```
localhost login: root  
Password:  
Last login: Thu Oct 12 19:49:29 on tty1  
[root@localhost ~]# systemctl status monservice.service  
■ monservice.service - SYSU: Message lors du démarrage.  
   Loaded: loaded (/etc/rc.d/init.d/monservice; generated)  
   Active: inactive (dead)  
     Docs: man:systemd-sysv-generator(8)  
[root@localhost ~]# systemctl enable monservice.service  
monservice.service is not a native service, redirecting to systemd-sysv-install.  
Executing: /usr/lib/systemd/systemd-sysv-install enable monservice  
[root@localhost ~]#
```

Par la suite on peut passer la commande pour activer le fichier. En utilisant « enable » sous systemctl.

```

[root@localhost init.d]# ls -l
total 28
-rw-r--r--. 1 root root 18434 Aug 10 2022 functions
-rw-r--r--. 1 root root 133 Oct 12 20:23 monservice
-rw-r--r--. 1 root root 1161 Jun 29 16:28 README
[root@localhost init.d]# chmod u+x monservice
[root@localhost init.d]# ls -l
total 28
-rw-r--r--. 1 root root 18434 Aug 10 2022 functions
-rwxr--r--. 1 root root 133 Oct 12 20:23 monservice
-rw-r--r--. 1 root root 1161 Jun 29 16:28 README
[root@localhost init.d]# ./monservice
Démarrage de mon ordi IFT3830
[root@localhost init.d]# _

```

On vérifie l'accès du fichier monservice, puis on donne l'accès avec la commande chmod. On test à nouveau, et on voit qu'il est accessible. Effectivement quand on exécute le script, il renvoie le echo demandée.

```

[root@localhost ~]# systemctl status monservice.service
■ monservice.service - SYSU: Message lors du démarrage.
   Loaded: loaded (/etc/rc.d/init.d/monservice; generated)
   Active: active (exited) since Thu 2023-10-12 20:31:14 EDT; 22s ago
     Docs: man:systend-sysu-generator(8)
   Process: 662 ExecStart=/etc/rc.d/init.d/monservice start (code=exited, status=0/SUCCESS)
    Tasks: 0 (limit: 50548)
   Memory: 0B
    CGroup: /system.slice/monservice.service

Oct 12 20:31:14 localhost.localdomain systemd[1]: Starting SYSU: Message lors du démarrage....
Oct 12 20:31:14 localhost.localdomain monservice[662]: Démarrage de mon ordi IFT3830
Oct 12 20:31:14 localhost.localdomain systemd[1]: Started SYSU: Message lors du démarrage..
[root@localhost ~]# _

```

Suite à un reboot, on voit que le service est toujours bien actif, et semble même avoir été brièvement exécuté lors du démarrage (voir le log).

4.

```

File Edit View Search Terminal Help
[root@localhost ~]# vi whichare
[root@localhost ~]# chmod u+x whichare
[root@localhost ~]# ./whichare 150
Répertoire /usr/local/bin :

Répertoire /usr/local/sbin :

Répertoire /usr/bin :
/usr/bin/false
/usr/bin/ls
/usr/bin/pulseaudio
/usr/bin/lsm
/usr/bin/vmtoolsd
/usr/bin/start-pulseaudio-x11

Répertoire /usr/sbin :
/usr/sbin/alsactl

Répertoire /root/bin :

[root@localhost ~]#

```

Donc creation du fichier whichare, on donne son acces par la suite, et un test du fichier avec un X de 150, qui équivaut à 150min. Nous obtenons un resultat similaire à l'exemple dans le tp1 du numero 4.


```
File Edit View Search Terminal Help
#!/bin/bash

if [ "$#" -ne 1 ]
then
    echo "Le nombre de paramètre n'est pas correct."
    exit 1
fi

IFS=";"

for dir in $PATH
do
    echo "Répertoire $dir :"
    find "$dir" -maxdepth 1 -type f -name "*ls*" -amin -$1
    echo -e
done

exit 0
~
~
~
~
```

Voici le script construit avec le vi. Prendre note que la commande **find** ne prends pas de valeur dénommé « \$# », il faut donc lui donnée une valeur « \$1 » pour le -amin avec un - devant le paramètre pour mentionner tant minutes «, et moins », \$1 qui est le premier paramètre donné dans la commande. Dans notre cas, le X, dans « whichare X ».

5.

```
File Edit View Search Terminal Help
[root@localhost ~]# vi nettoyer_repertoire
[root@localhost ~]# chmod u+x nettoyer_repertoire
[root@localhost ~]#
```

Donc création du script avec vi, puis avec commande pour donner accès.

```
File Edit View Search Terminal Help
#!/bin/bash

endsWith() {
    case $1 in
        **$2) return 0;;
        *) return 1;;
    esac
}

for directory in "$@"; do
    if [ -d "$directory" ]; then
        cd "$directory" || exit 1

        for file in $(ls -A 2> /dev/null); do
            if endsWith "$file" ".pdf"; then
                if [ -f "../$file" ]; then
                    rm "$file"
                else
                    mv "$file" ..
                fi
            else
                cat "$file" >> "../$file"
                rm "$file"
            fi
        done
        cd ..
        rm -r "$directory"
    else
        echo "Le répertoire '$directory' n'existe pas."
    fi
done

exit 0
~
~
~
```

Voici le scripte dans le vi.

Nous procédons à la vérification/test du script...

```
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# mkdir dir1 dir2  
[root@localhost ~]# touch dir1/fichier.txt dir1/fichier.pdf dir2/fichier.txt dir2/fichier.pdf  
[root@localhost ~]# ls  
.: anaconda-ks.cfg bin Desktop dir2 Downloads initial-setup-ks.cfg Music nettoyoy_repertoire Pictures RPM.tar Videos  
[root@localhost ~]#
```

Pour faire le test, nous créons 2 répertoires dir1 et dir2. Chaque répertoire à un fichier text et pdf, dans les deux répertoires ils ont le même nom exactement (fichier.txt et fichier.pdf).

```
File Edit View Search Terminal Help  
[root@localhost ~]# cd dir1/  
[root@localhost dir1]# vi fichier.txt  
[root@localhost dir1]# vi fichier.pdf  
[root@localhost dir1]# cd  
[root@localhost ~]# cd dir2/  
[root@localhost dir2]# vi fichier.txt  
[root@localhost dir2]# vi fichier.pdf  
[root@localhost dir2]# cd  
[root@localhost ~]#
```

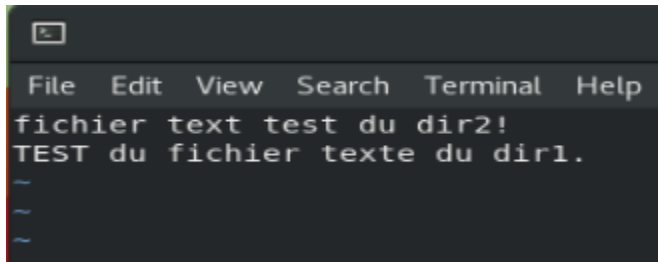
Ici on modifie les fichiers pour donner un contenu pour voir la modification à la suite du scripte. Le contenu est simple, s'il s'agit du fichier texte dans le dir 1, le contenu sera « fichier format texte test du dir1 », « fichier format pdf test du dir1 » par exemple, et ainsi de suite...

```
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# ls  
.: anaconda-ks.cfg bin Desktop dir2 Downloads initial-setup-ks.cfg Music nettoyoy_repertoire Pictures RPM.tar Videos  
[root@localhost ~]# ./nettoyoy_repertoire dir2 dir1  
[root@localhost ~]# ls  
.: anaconda-ks.cfg bin Desktop Downloads fichier.txt initial-setup-ks.cfg Music nettoyoy_repertoire Pictures RPM.tar Videos  
[root@localhost ~]#
```

Donc nous en somme à l'exécution du test nous voyons bien le « avant et après », nous avons dir 1 et dir2, qui disparaissent pour n'en laisser qu'un version du texte et une version du pdf, puisqu'ils ont le même nom dans les deux répertoires.

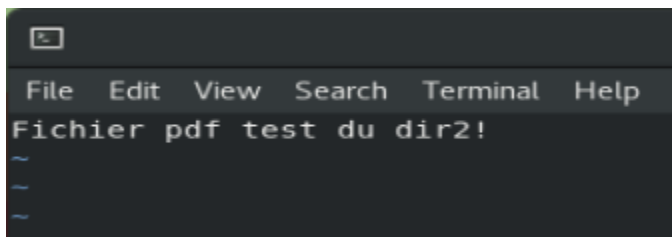
Cependant, on se rappelle que le fichier texte doit concaténer les contenus alors que le pdf doit écraser l'autre. Dans notre test nous avons choisi de déclarer dir2 avant dir1, donc logiquement le fichier texte aura concaténer le fichier dir2 avec le dir1 et le pdf du dir2 écrasera celui du dir1.

Voici les résultats :



```
File Edit View Search Terminal Help
fichier text test du dir2!
TEST du fichier texte du dir1.
~
~
~
```

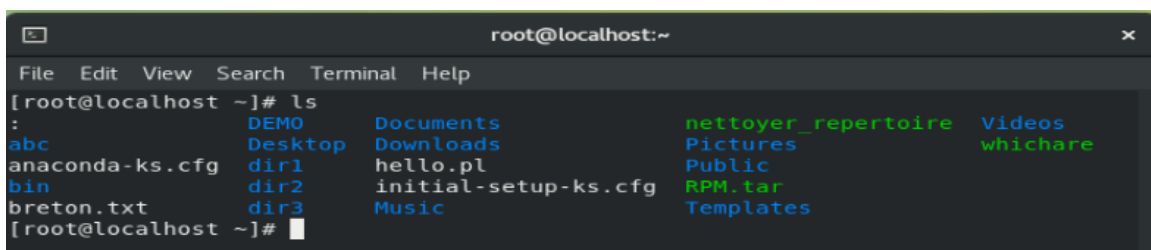
Fichier texte : On remarque qu'il y a bien une concaténation, qui prends le fichier du dir2 et combine le contenu du dir1 après le contenu du dir2. Le test et l'exécution est donc bien réussi.



```
File Edit View Search Terminal Help
Fichier pdf test du dir2!
~
~
~
```

Fichier PDF : Ici il n'y a pas de concaténation. Seulement le contenu du fichier du dir2 est présent et a donc effectivement écrasé le contenu du fichier.pdf du dir1. Le test du script à fonctionner comme on le voulait.

2^e etape, nous pouvons le tester avec le fichier test offert dans le cours pour le TP1 exercice 5 :



```
root@localhost:~
File Edit View Search Terminal Help
[root@localhost ~]# ls
:          DEMO      Documents  nettoyer_repertoire  Videos
abc        Desktop    Downloads  Pictures             whichare
anaconda-ks.cfg  dir1      hello.pl   Public
bin          dir2      initial-setup-ks.cfg RPM.tar
breton.txt    dir3      Music      Templates
[root@localhost ~]#
```

On voit bien que les fichiers test ont été ajouter, et on peut commencer le 2e test. Nous allons cette fois tester avec dir1, dir2 et dir3 dans ce même ordre.

```
root@localhost:~  
File Edit View Search Terminal Help  
[root@localhost ~]# ./nettoyer_repertoire dir1 dir2 dir3  
[root@localhost ~]# ls  
:  
Documents initial-setup-ks.cfg Public  
abc dos.zip jeep.jpg RPM.tar  
anaconda-ks.cfg Downloads mots-d-enfants Templates  
bin firewall.pdf Music Videos  
breton.txt halt nettoyer_repertoire whichare  
DEMO hello.pl pensee Word.doc  
Desktop IMPORTANT.pdf Pictures  
[root@localhost ~]#
```

Voici le résultat, inspectons les fichiers... dir1 dir2 et dir3 sont effectivement décomposé comme anticiper.

Fichiers texte :

```
root@localhost:~  
File Edit View Search Terminal Help  
#!/usr/bin/perl  
print "Hello world\n";  
#!/usr/bin/perl  
print "Hello world\n";  
#!/usr/bin/perl  
print "Hello world\n";  
~
```

À noter que hello.pl existait déjà hors des 3 répertoires, dans dir1 et dir3, donc il y a 3 versions concaténer de ce fichier.

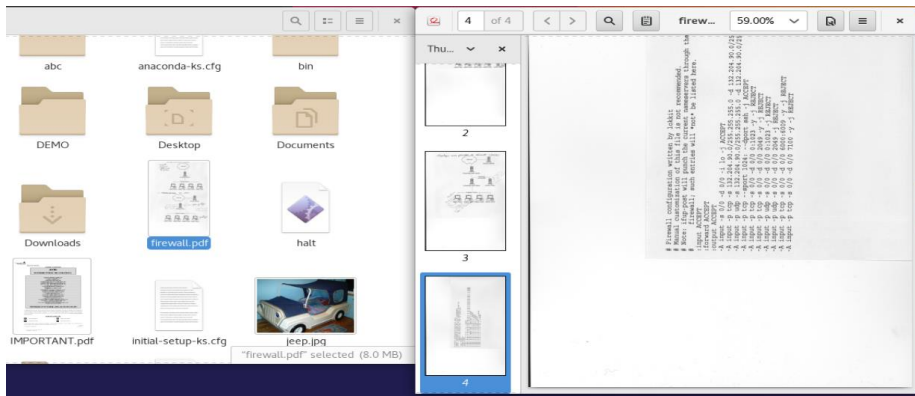
```
root@localhost:~  
File Edit View Search Terminal Help  
Il faut lire le texte AVANT de voir la piece jointe (breton.jpg)  
L'histoire est veridique, puisque l'annonce est tiree d'un  
journal breton  
~
```

Breton.txt, lui n'est pas affecter, aucun autre fichier portait le même nom parmi les répertoires décomposés.

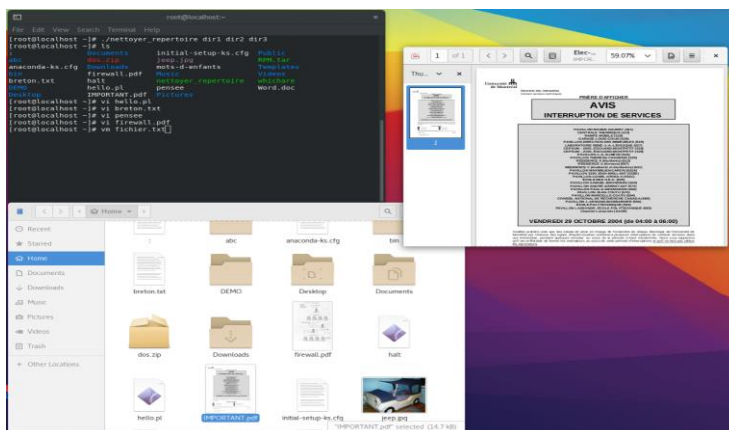
```
root@localhost:~  
File Edit View Search Terminal Help  
Les petits esprits discutent des gens.  
Qui perd de l'argent, perd beaucoup.  
Qui perd un ami, perd encore plus.  
Qui perd l'espérance, perd tout.  
Les belles et jeunes personnes sont des accidents de la nature.  
Mais les belles et vieilles personnes sont des oeuvres d'art.  
Apprend des erreurs des autres.  
Tu ne peux vivre assez longtemps pour toutes les faire toi-même.  
Hier est de l'histoire. Demain est un mystère. Aujourd'hui est un cadeau.  
To the world you might be one person, but to one person you might be the world.  
Sometimes the majority only means that all the fools are on the same side.  
Going to church does not make you a Christian anymore than going to McDonald's  
makes you a hamburger.  
I don't have to attend every argument I'm invited to.
```

Le fichier pensee qui n'est pas un pdf encore, lui est concaténer puisqu'il a une copie dans dir1 et dir2 avec le même nom.

PDF :



Firewall.pdf, lui a été écrasé à 2 reprises puisque le fichier existait sous dir1 dir2 et dir3. Exactement 1 fichier existe pour firewall, et il n'est pas concaténée, comme démontrer sur la photo.



De même est constater pour le fichier pdf « IMPORTANT ». Le teste est réussie et le scripte fonctionne comme anticiper.