



### LAB 3

## SỬ DỤNG SHELL SCRIPTING, QUẢN LÝ TIẾN TRÌNH, TẬP TIN NHẬT KÝ HỆ THỐNG

Họ tên và MSSV: Trần Minh Khôi

Nhóm học phần: 02

- Các sinh viên bị phát hiện sao chép bài của nhau sẽ nhận 0đ cho tất cả bài thực hành của môn này.
- Bài nộp phải ở dạng PDF, hình minh họa phải rõ ràng chi tiết. Hình minh họa chỉ cần chụp ở nội dung thực hiện, không chụp toàn màn hình.
- Video hướng dẫn ở cuối bài.

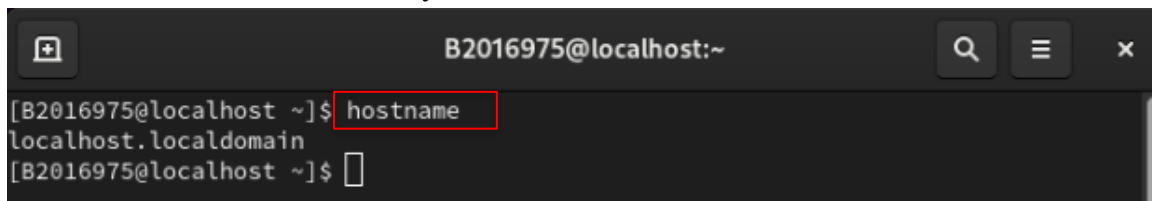
### 1. Cài đặt CentOS

Thực hiện cài đặt CentOS 9 Stream vào máy tính cá nhân (hoặc máy ảo) của bạn **nếu cần** (KHÔNG cần chụp hình minh họa).

### 2. Shell scripting

2.1. Thực hiện các lệnh bên dưới và cho biết ý nghĩa của chúng (chụp hình minh họa):

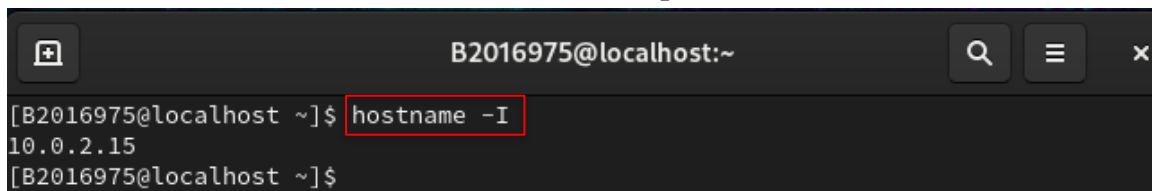
hostname : in ra tên của máy tính



```
B2016975@localhost:~  
[B2016975@localhost ~]$ hostname  
localhost.localdomain  
[B2016975@localhost ~]$
```

Hình 1: Lệnh hostname in ra tên của máy tính

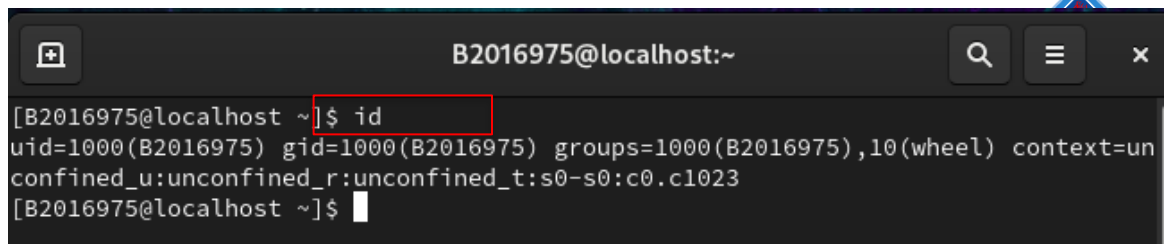
hostname -I : in ra địa chỉ IP của máy



```
B2016975@localhost:~  
[B2016975@localhost ~]$ hostname -I  
10.0.2.15  
[B2016975@localhost ~]$
```

Hình 2: Lệnh hostname -I in ra địa chỉ IP của máy

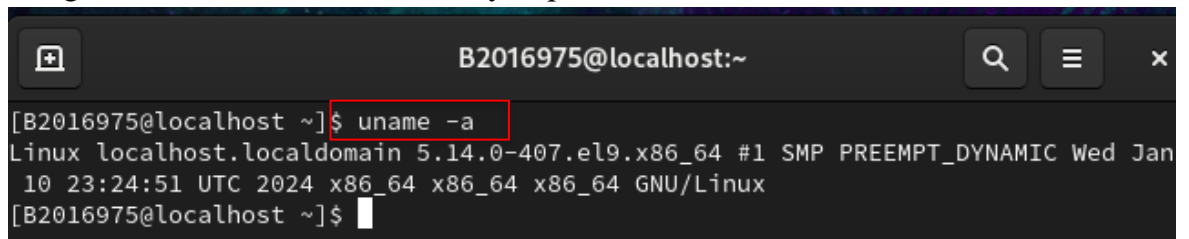
id : in ra các thành phần như : uid(mã số người dùng,username(tên đăng nhập người dùng), gid(nhóm mặc định của người dùng), groups( các nhóm mà người dùng thuộc vào)



```
B2016975@localhost:~  
[B2016975@localhost ~]$ id  
uid=1000(B2016975) gid=1000(B2016975) groups=1000(B2016975),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[B2016975@localhost ~]$
```

Hình 3: Lệnh id in ra uid, gid, groups.

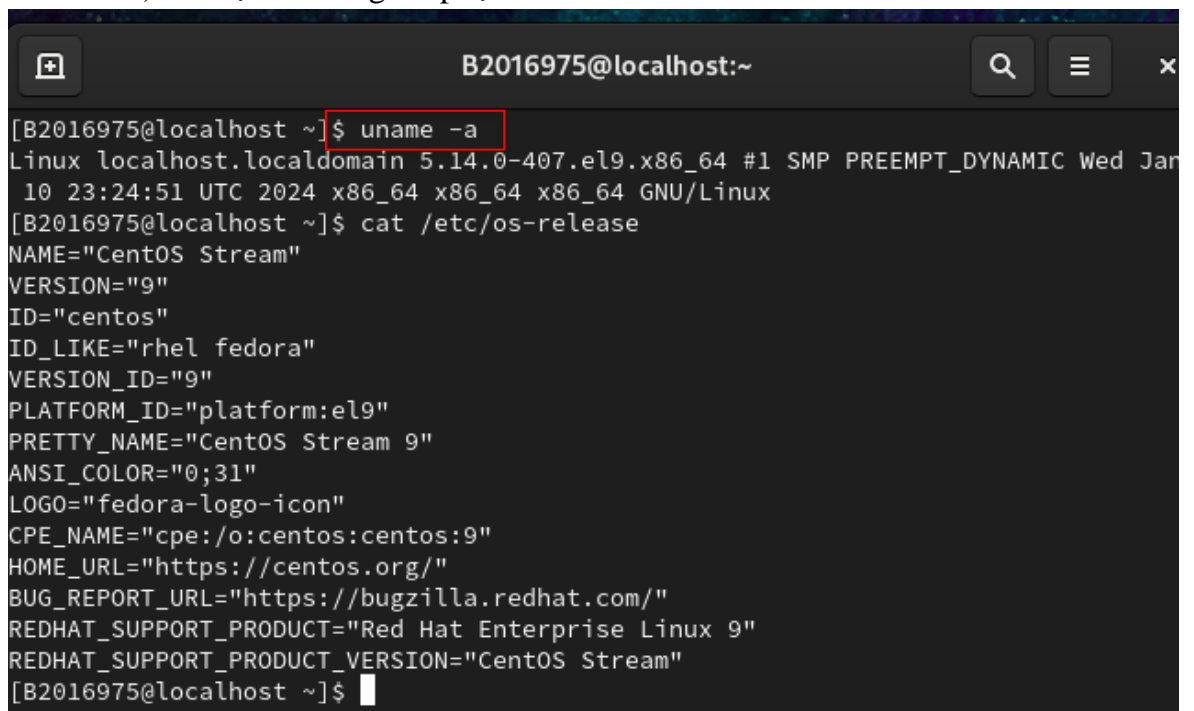
uname -a: Hiển thị thông tin về phiên bản LINUX mà chúng ta sử dụng trong hệ điều hành CentOS 9, ở đây là phiên bản 5.14



```
B2016975@localhost:~  
[B2016975@localhost ~]$ uname -a  
Linux localhost.localdomain 5.14.0-407.el9.x86_64 #1 SMP PREEMPT_DYNAMIC Wed Jan 10 23:24:51 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux  
[B2016975@localhost ~]$
```

Hình 4 Lệnh uname -a in ra thông tin về phiên bản LINUX

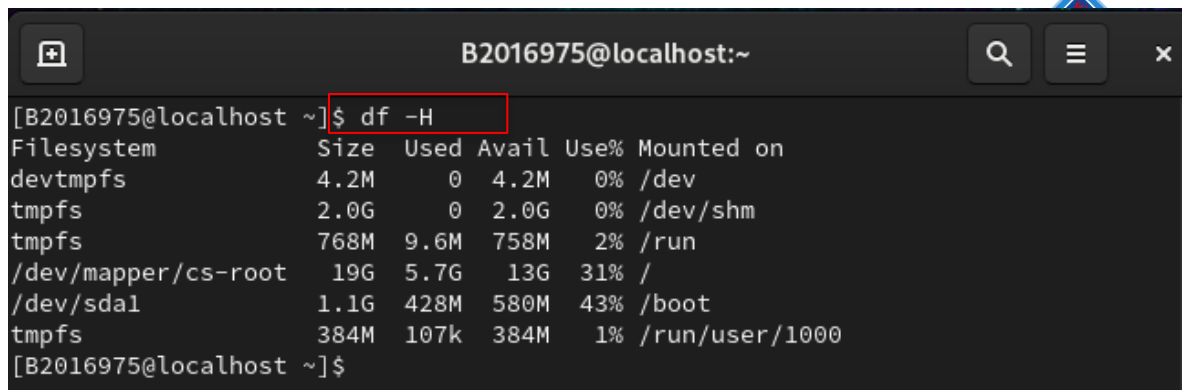
cat /etc/os-release: lệnh cat này in ra nội dung của tập in /etc/os-release( chứa thông tin NAME(tên hệ điều hành), VERSION(phiên bản hệ điều hành) và một số thông tin phụ khác.



```
B2016975@localhost:~  
[B2016975@localhost ~]$ uname -a  
Linux localhost.localdomain 5.14.0-407.el9.x86_64 #1 SMP PREEMPT_DYNAMIC Wed Jan 10 23:24:51 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux  
[B2016975@localhost ~]$ cat /etc/os-release  
NAME="CentOS Stream"  
VERSION="9"  
ID="centos"  
ID_LIKE="rhel fedora"  
VERSION_ID="9"  
PLATFORM_ID="platform:el9"  
PRETTY_NAME="CentOS Stream 9"  
ANSI_COLOR="0;31"  
LOGO="fedora-logo-icon"  
CPE_NAME="cpe:/o:centos:centos:9"  
HOME_URL="https://centos.org/"  
BUG_REPORT_URL="https://bugzilla.redhat.com/"  
REDHAT_SUPPORT_PRODUCT="Red Hat Enterprise Linux 9"  
REDHAT_SUPPORT_PRODUCT_VERSION="CentOS Stream"  
[B2016975@localhost ~]$
```

Hình 5 Lệnh cat /etc/os-release in ra thông tin của tập tin os-release trong thư mục /etc

df -H:liệt kê các phân vùng của ổ cứng( đang mount tới các tập tin nào, dung lượng ổ cứng đang được sử dụng là bao nhiêu).

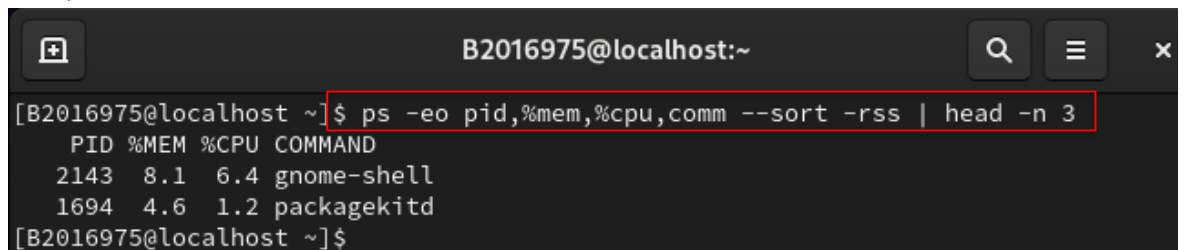


```

B2016975@localhost:~
[B2016975@localhost ~]$ df -H
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.2M   0    4.2M   0% /dev
tmpfs           2.0G   0    2.0G   0% /dev/shm
tmpfs           768M   9.6M  758M   2% /run
/dev/mapper/cs-root 19G   5.7G  13G   31% /
/dev/sda1       1.1G  428M  580M  43% /boot
tmpfs           384M  107k  384M   1% /run/user/1000
[B2016975@localhost ~]$
  
```

Hình 6 lệnh `df -H` liệt kê các phân vùng của ổ cứng, dung lượng lưu trữ...v.v

`ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3`: in ra thông tin pid(mã số tiến trình), %mem(phần trăm bộ nhớ), %cpu, comm(lệnh thực thi tiến trình) --sort(sắp xếp theo phần trăm bộ nhớ), -rss(theo phần trăm giảm dần) |head -n 3(in ra 2 tiến trình sử dụng nhiều phần trăm bộ nhớ nhất)



```

B2016975@localhost:~
[B2016975@localhost ~]$ ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3
  PID %MEM %CPU COMMAND
   2143   8.1   6.4 gnome-shell
   1694   4.6   1.2 packagekitd
[B2016975@localhost ~]$
  
```

Hình 7Lệnh `ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3` in ra các thông tin như pid, %memory, %cpu theo phần trăm bộ nhớ theo phần trăm giảm dần của 2 tiến trình sử dụng nhiều nhất.

(KHÔNG CÓ KHOẢNG TRẮNG SAU DẤU PHẨY)



- 2.2.** Viết shell script có tên `info.sh` lần lượt thực hiện tất cả các lệnh ở 2.1 (chụp hình minh họa).

```

GNU nano 5.6.1 info.sh
echo "-----"
hostname
echo "-----"
hostname -I
echo "-----"
id
echo "-----"
cat /etc/os-release
echo "-----"
ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3
    
```

Hình 8 Thông tin file shell script có tên `info.sh` lần lượt thực hiện tất cả các lệnh ở 2.1.

```

B2016975@localhost:~$ bash info.sh
-----
localhost.localdomain
-----
10.0.2.15
-----
uid=1000(B2016975) gid=1000(B2016975) groups=1000(B2016975),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
-----
NAME="CentOS Stream"
VERSION="9"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="9"
PLATFORM_ID="platform:el9"
PRETTY_NAME="CentOS Stream 9"
ANSI_COLOR="0;31"
LOGO="fedora-logo-icon"
CPE_NAME="cpe:/o:centos:centos:9"
HOME_URL="https://centos.org/"
BUG_REPORT_URL="https://bugzilla.redhat.com/"
REDHAT_SUPPORT_PRODUCT="Red Hat Enterprise Linux 9"
REDHAT_SUPPORT_PRODUCT_VERSION="CentOS Stream"
-----
  PID %MEM %CPU COMMAND
    2143  8.3  5.1 gnome-shell
    1694  4.6  0.8 packagekitd
[B2016975@localhost ~]$
    
```

Hình 9 Thông tin sau khi chạy file `info.sh`

- 2.3.** Viết shell script có tên `backup.sh` thực hiện:

- In ra ngày giờ hiện tại



- Nén toàn bộ thư mục cá nhân của người trong /home thành tập tin /tmp/<YYYY-MM-DD>.tar (YYYY-MM-DD là ngày hiện tại, ví dụ: 2023-09-25.tar)
  - In thông tin đầy đủ của tập tin /tmp/<YYYY-MM-DD>.tar
  - In thông ra thông báo "Sao lưu thành công!!!!"
- Thực thi backup.sh để kiểm tra (chụp hình minh họa).

```

GNU nano 5.6.1 backup.sh
echo "Bat dau sao luu du lieu"
echo "-----"
echo "Ngày gio he thong"
date
echo "-----"
filename="/tmp/${date +%F}.tar"
tar -cf $filename /home/b2016975
echo "Thông tin du lieu sao luu"
ls -l $filename
echo "Sao luu thanh cong"
    
```

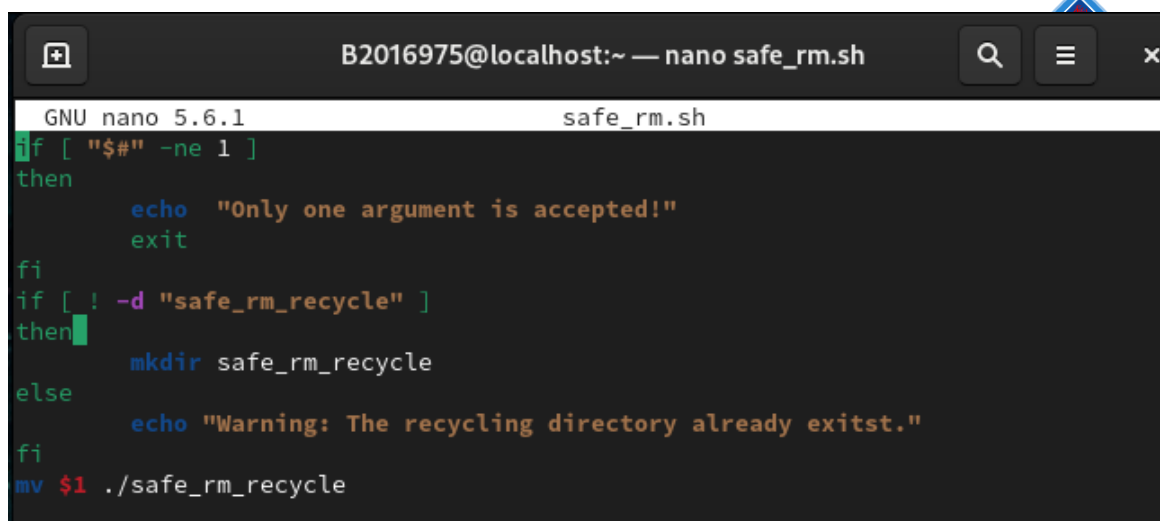
Hình 10 Thông tin file backup.sh

```

[B2016975@localhost ~]$ bash backup.sh
Bat dau sao luu du lieu
-----
Ngày gio he thong
Wed Mar  6 11:19:12 AM +07 2024
-----
tar: Removing leading `/' from member names
Thông tin du lieu sao luu
-rw-r--r--. 1 B2016975 B2016975 157102080 Mar  6 11:19 /tmp/2024-03-06.tar
Sao luu thanh cong
[B2016975@localhost ~]$
    
```

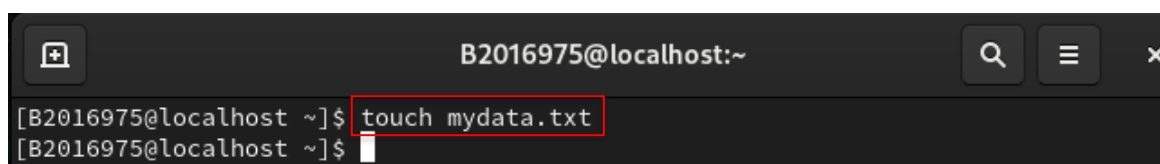
Hình 11 Thông tin sau khi chạy file backup.sh

- 2.4.** Bổ sung dòng các lệnh cần thiết vào shell script `safe_rm.sh` bên dưới để thực hiện công việc sau:
- Nhận 01 tham số từ dòng lệnh. Thông báo lỗi và thoát khỏi chương trình nếu không có tham số hoặc nhiều hơn 01 tham số.
  - Tạo thư mục với tên "safe\_rm\_recycle" nếu chưa có. Sao chép tập tin với tên là tham số 01 vào thư mục "safe\_rm\_recycle" vừa tạo. Sau đó xóa tập tin.
  - Tạo tập tin có tên `mydata.txt`, thực thi `safe_rm.sh` để xóa tập tin `mydata.txt` (chụp hình minh họa).



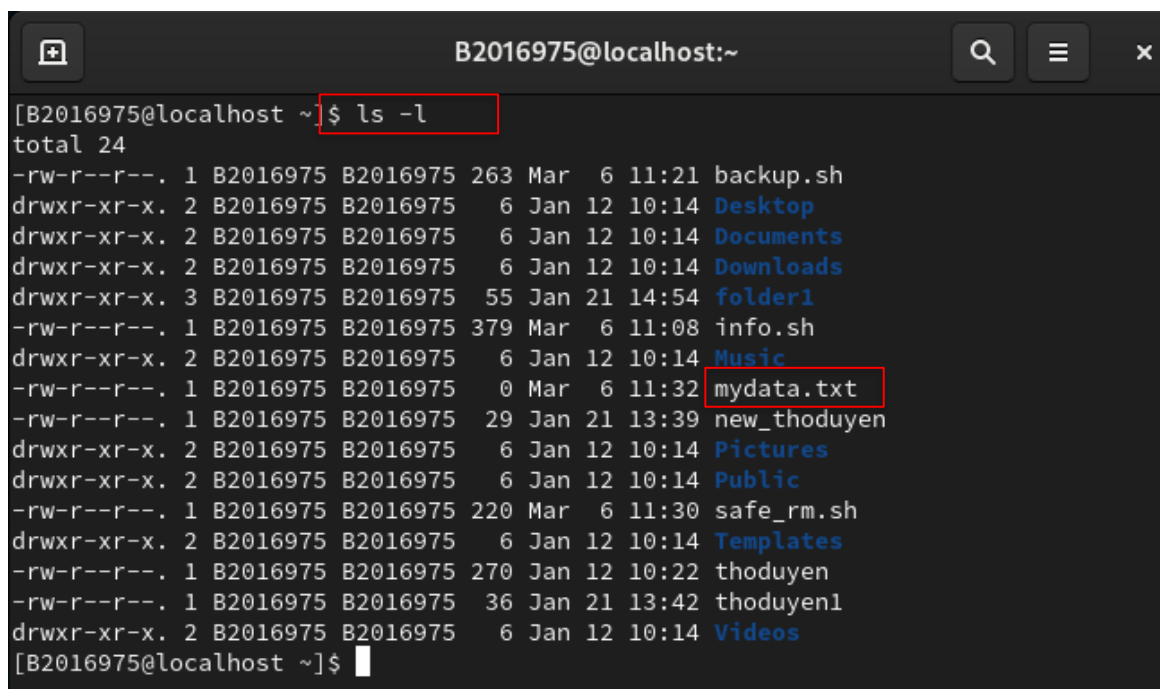
```
B2016975@localhost:~ — nano safe_rm.sh
GNU nano 5.6.1 safe_rm.sh
if [ "$#" -ne 1 ]
then
    echo "Only one argument is accepted!"
    exit
fi
if [ ! -d "safe_rm_recycle" ]
then
    mkdir safe_rm_recycle
else
    echo "Warning: The recycling directory already existst."
fi
mv $1 ./safe_rm_recycle
```

Hình 12 Thông tin file safe\_rm.sh



```
B2016975@localhost:~
[B2016975@localhost ~]$ touch mydata.txt
[B2016975@localhost ~]$
```

Hình 13 Tạo ra một file có tên mydata.txt



```
B2016975@localhost:~
[B2016975@localhost ~]$ ls -l
total 24
-rw-r--r--. 1 B2016975 B2016975 263 Mar  6 11:21 backup.sh
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Desktop
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Documents
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Downloads
drwxr-xr-x. 3 B2016975 B2016975 55 Jan 21 14:54 folder1
-rw-r--r--. 1 B2016975 B2016975 379 Mar  6 11:08 info.sh
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Music
-rw-r--r--. 1 B2016975 B2016975  0 Mar  6 11:32 mydata.txt
-rw-r--r--. 1 B2016975 B2016975 29 Jan 21 13:39 new_thoduyen
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Pictures
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Public
-rw-r--r--. 1 B2016975 B2016975 220 Mar  6 11:30 safe_rm.sh
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Templates
-rw-r--r--. 1 B2016975 B2016975 270 Jan 12 10:22 thoduyen
-rw-r--r--. 1 B2016975 B2016975 36 Jan 21 13:42 thoduyen1
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Videos
[B2016975@localhost ~]$
```

Hình 14 Dùng lệnh ls -l để xác nhận có file mydata.txt được tạo ra

```

B2016975@localhost:~
[B2016975@localhost ~]$ bash safe_rm.sh mydata.txt
[B2016975@localhost ~]$ ls -l
total 24
-rw-r--r--. 1 B2016975 B2016975 263 Mar  6 11:21 backup.sh
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Desktop
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Documents
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Downloads
drwxr-xr-x. 3 B2016975 B2016975 55 Jan 21 14:54 folder1
-rw-r--r--. 1 B2016975 B2016975 379 Mar  6 11:08 info.sh
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Music
-rw-r--r--. 1 B2016975 B2016975 29 Jan 21 13:39 new_thoduyen
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Pictures
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Public
drwxr-xr-x. 2 B2016975 B2016975 24 Mar  6 11:35 safe_rm_recycle
-rw-r--r--. 1 B2016975 B2016975 220 Mar  6 11:30 safe_rm.sh
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Templates
-rw-r--r--. 1 B2016975 B2016975 270 Jan 12 10:22 thoduyen
-rw-r--r--. 1 B2016975 B2016975 36 Jan 21 13:42 thoduyen1
drwxr-xr-x. 2 B2016975 B2016975  6 Jan 12 10:14 Videos
[B2016975@localhost ~]$

```

Hình 15 Thực thi file `safe_rm.sh` với tham số truyền vào là file `mydata.txt` và lệnh `ls -l` để xác nhận file `mydata.txt` đã được xóa, một thư mục có tên `safe_rm_recycle` được tạo ra chứa thông tin của file đã được xóa.

```

B2016975@localhost:~
[B2016975@localhost ~]$ ls -l safe_rm_recycle/
total 0
-rw-r--r--. 1 B2016975 B2016975 0 Mar  6 11:32 mydata.txt
[B2016975@localhost ~]$

```

Hình 16 Liệt kê nội dung của thư mục `safe_rm_recycle.sh`

`safe_rm.sh`

```

if [ "$#" -ne 1 ]
then
    echo "Only one argument is accepted!"
    #Thêm 1 dòng lệnh ở đây
fi
if [ ! -d "safe_rm_recycle" ]
then
    #Thêm 1 dòng lệnh ở đây
else
    echo "Warning: The recycling directory already exists."
fi

```



#Thêm 1 dòng lệnh ở đây

### 3. Lên lịch công việc định kỳ với cron

Cron là một tiện ích trong Linux cho phép thiết lập thời gian thực hiện công việc một cách định kỳ. Một crontab file chứa danh sách các lệnh sẽ được thực thi, kèm theo thông tin về thời điểm thực thi. Để hiệu chỉnh file crontab với trình soạn thảo nano, ta dùng các lệnh sau:

```
$ export EDITOR=nano
$ crontab -e
```

```
B2016975@localhost:~$ export EDITOR=nano
B2016975@localhost:~$ crontab -e
crontab: no changes made to crontab
B2016975@localhost:~$
```

Hình 17 Khai báo biến môi trường `EDITOR=nano` bằng lệnh `export` và thực hiện lệnh `crontab -e` để tiến hành lên thời khóa biểu cho các công việc bằng trình soạn thảo nano

Cho biết cú pháp để thực hiện các yêu cầu sau từ crontab file:

Việc lên thời khóa biểu bao gồm: nhập vào 5 con số chính là thời gian mà công việc cần thực thi (giá trị nào mà không cần quan tâm tới thì để dấu '\*' (dấu sao)) và công việc cần thực hiện

- + 1<sup>st</sup> number: phút
- + 2<sup>nd</sup> number: giờ
- + 3<sup>rd</sup> number: ngày
- + 4<sup>th</sup>: tháng
- + 5<sup>th</sup>: thứ ( số 1 đại diện cho thứ 2)

Vd : 30 2 \* \* 1 (cd /home/joe4/project; make): vào lúc 2 giờ 30 phút thứ 2 hàng tuần thì 2 lệnh cd /home/joe4/project; make sẽ tự động thực hiện.

**3.1.** Chạy lệnh `date` mỗi phút một lần, sau đó ghi kết quả vào cuối tập tin /tmp/date.txt (chụp hình minh họa)

```
B2016975@localhost:~ — crontab -e
GNU nano 5.6.1 /tmp/crontab.nhSJDn
* * * * * (date>> /tmp/date.txt)
```

Hình 18 Chạy lệnh `date` mỗi phút một lần vào trong file date.txt



```

B2016975@localhost:~
[B2016975@localhost ~]$ crontab -e
no crontab for B2016975 - using an empty one
crontab: installing new crontab
[B2016975@localhost ~]$ crontab -l
* * * * * (date>> /tmp/date.txt)
[B2016975@localhost ~]$ cat /tmp/date.txt
Wed Mar  6 12:19:04 PM +07 2024
Wed Mar  6 12:19:36 PM +07 2024
[B2016975@localhost ~]$ cat /tmp/date.txt
Wed Mar  6 12:19:04 PM +07 2024
Wed Mar  6 12:19:36 PM +07 2024
Wed Mar  6 12:21:01 PM +07 2024
[B2016975@localhost ~]$ cat /tmp/date.txt
Wed Mar  6 12:19:04 PM +07 2024
Wed Mar  6 12:19:36 PM +07 2024
Wed Mar  6 12:21:01 PM +07 2024
[B2016975@localhost ~]$ cat /tmp/date.txt
Wed Mar  6 12:19:04 PM +07 2024
Wed Mar  6 12:19:36 PM +07 2024
Wed Mar  6 12:21:01 PM +07 2024
Wed Mar  6 12:22:01 PM +07 2024
[B2016975@localhost ~]$

```

Hình 19 Dùng lệnh `crontab -l` để kiểm tra và kiểm tra bằng lệnh `cat /tmp/date.txt` để kiểm tra xem sau 1 phút có xuất hiện thêm dòng mới hay không?

- 3.2.** Thực thi `backup.sh` ở Câu 2.3 vào 23:50 giờ ngày 10, 20 và 30 hàng tháng (chụp hình minh họa).

```

B2016975@localhost:~ — crontab -e
GNU nano 5.6.1 /tmp/crontab.sQY6Lm
* * * * * (date>> /tmp/date.txt)
50 23 10,20,30 * * (/home/B2016975/backup.sh)
0 8-18 * * 1-5 (/home/B2016975/backup.sh)

```

- 3.3.** Thực thi `backup.sh` ở Câu 2.3 vào mỗi giờ 1 lần, từ 8:00 đến 18:00, trong các ngày làm việc (thứ hai đến thứ sáu) trong tuần (chụp hình minh họa).

```

B2016975@localhost:~ — crontab -e
GNU nano 5.6.1 /tmp/crontab.sQY6Lm
* * * * * (date>> /tmp/date.txt)
50 23 10,20,30 * * (/home/B2016975/backup.sh)
0 8-18 * * 1-5 (/home/B2016975/backup.sh)

```



#### 4. Quản lý tiến trình

Tìm hiểu và thực hiện các yêu cầu sau:

4.1. Tìm tất cả các tiến trình được thực thi bởi người dùng <Mã số sinh viên>.

```

B2016975@localhost:~
[B2016975@localhost ~]$ ps -aux | grep "B2016975"
B2016975  5040  0.0  0.3 23576 14428 ?        Ss   22:22   0:00 /usr/lib/systemd/systemd --user
B2016975  5043  0.0  0.2 174540 7984 ?          S    22:22   0:00 (sd-pam)
B2016975  5071  0.0  0.1 452820 7196 ?          Sl   22:22   0:00 /usr/bin/gnome-keyring-daemon --daemonize --login
B2016975  5078  0.0  0.2 374156 7904 tty2        Ssl+ 22:22   0:00 /usr/libexec/gdm-wayland-session --register-session gnome-session
B2016975  5081  0.0  0.1 10680 4480 ?          Ss   22:22   0:00 /usr/bin/dbus-broker-launch --scope user
B2016975  5083  0.0  0.1  6448 3968 ?          S    22:22   0:00 dbus-broker --log 4 --controller 9 --machine-id 39fcb5c8d3be46cfbacbcd0d6c5d6a3 --max-bytes 1000000000000000 --max-fds 2500000000000000 --max-matches 5000000000
B2016975  5087  0.0  0.5 513136 19488 tty2        Sl+  22:22   0:00 /usr/libexec/gnome-session-binary
B2016975  5131  0.0  0.2 303796 7620 ?          Ssl  22:22   0:00 /usr/libexec/gnome-session-ctl --monitor
B2016975  5133  0.0  0.5 808588 20772 ?          Ssl  22:22   0:00 /usr/libexec/gnome-session-binary --systemd-service --session=gnome
B2016975  5149  3.4  8.2 4044364 310660 ?        Rsl  22:22   3:07 /usr/bin/gnom
    
```

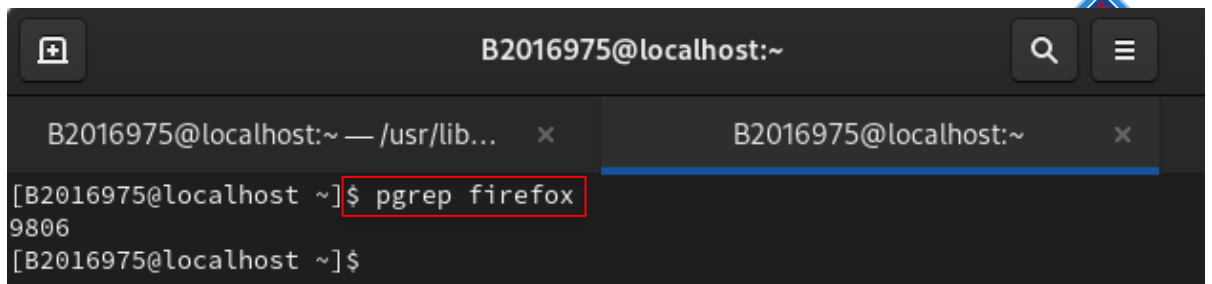
Hình 20 Liệt kê tất cả tiến trình được thực thi bởi người dùng B2016975

4.2. Mở trình duyệt Firefox. Sau đó dùng lệnh `pgrep` tìm PID của firefox. Giảm độ ưu tiên của tiến trình firefox thành 10.

```

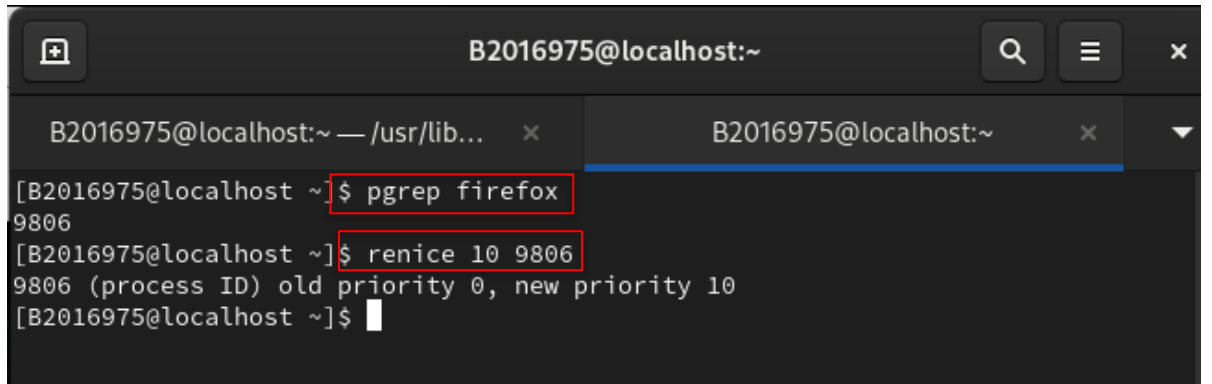
B2016975@localhost:~ — /usr/lib64/firefox/firefox
B2016975@localhost:~ — /usr/lib... x B2016975@localhost:~ x
[B2016975@localhost ~]$ firefox
    
```

Hình 21 Dùng lệnh `firefox` để mở trình duyệt firefox



```
B2016975@localhost:~  
[B2016975@localhost ~]$ pgrep firefox  
9806  
[B2016975@localhost ~]$
```

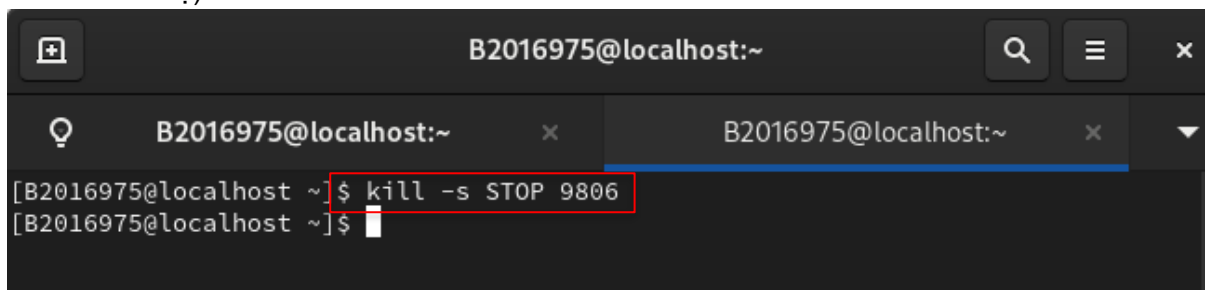
Hình 22 Dùng lệnh `pgrep` để xem PID của `firefox`



```
B2016975@localhost:~  
[B2016975@localhost ~]$ pgrep firefox  
9806  
[B2016975@localhost ~]$ renice 10 9806  
9806 (process ID) old priority 0, new priority 10  
[B2016975@localhost ~]$
```

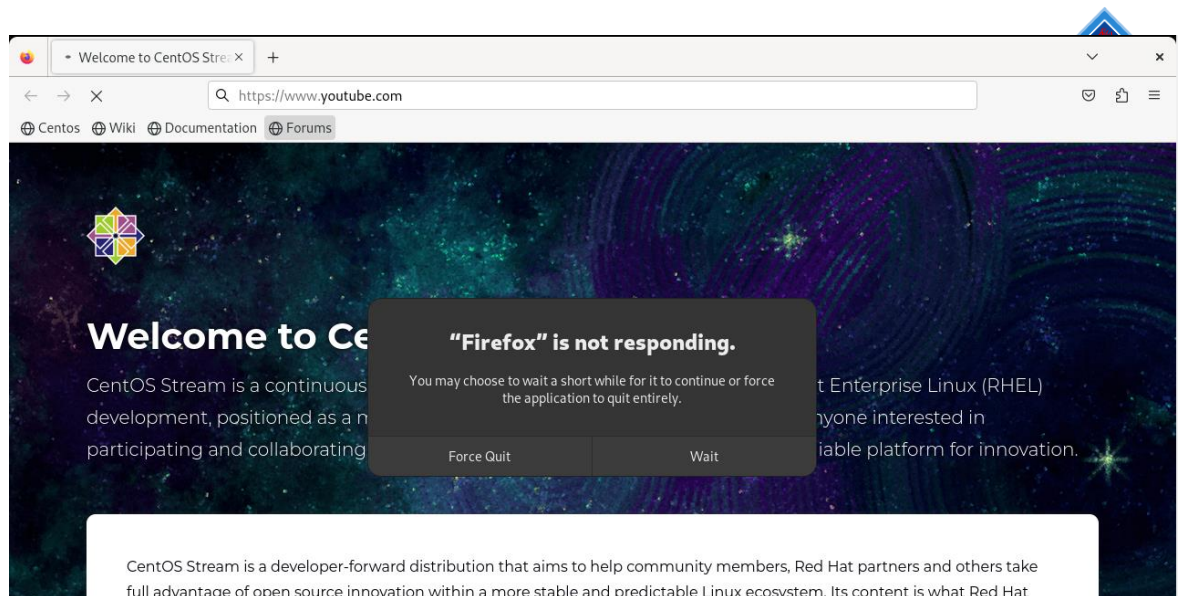
Hình 23 Độ ưu tiên của tiến trình `firefox` đã được giảm thành 10

- 4.3. Dùng lệnh `kill` để tạm dừng tiến trình `firefox` (chụp hình minh họa). Điều gì xảy ra khi bạn dịch chuyển cửa sổ `firefox` hoặc nhấn chọn menu của nó ngay lúc này (chụp hình minh họa)?



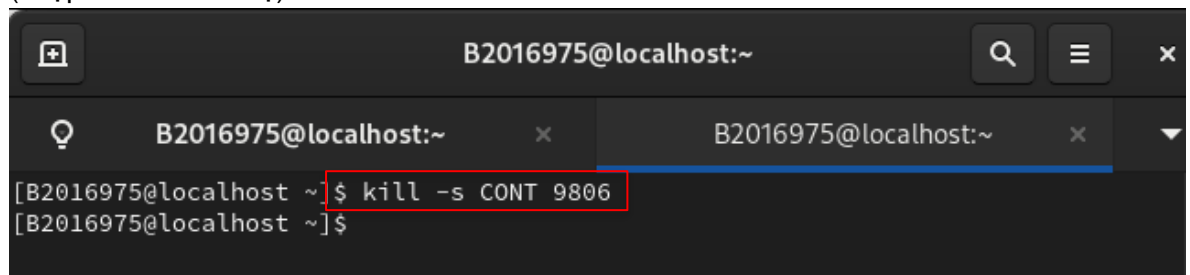
```
B2016975@localhost:~  
[B2016975@localhost ~]$ kill -s STOP 9806  
[B2016975@localhost ~]$
```

Hình 24 Dùng lệnh `kill -s STOP 9806` để tiến hành tạm dừng tiến trình `firefox` trong đó 9806 là mã định danh của tiến trình `firefox`

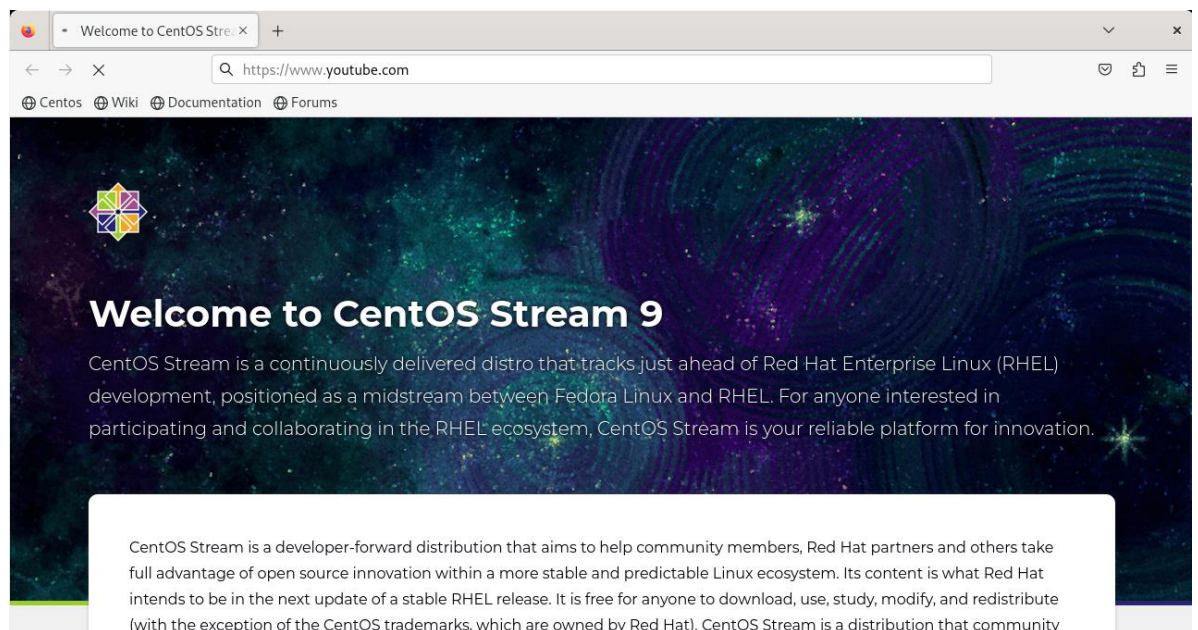


Hình 25 Firefox gửi tín hiệu "is not responding" sau khi dừng tiến trình bằng lệnh `kill -s STOP`

- 4.4. Dùng lệnh `kill` để phục hồi trạng thái trước đó của firefox và quan sát kết quả (chụp hình minh họa).

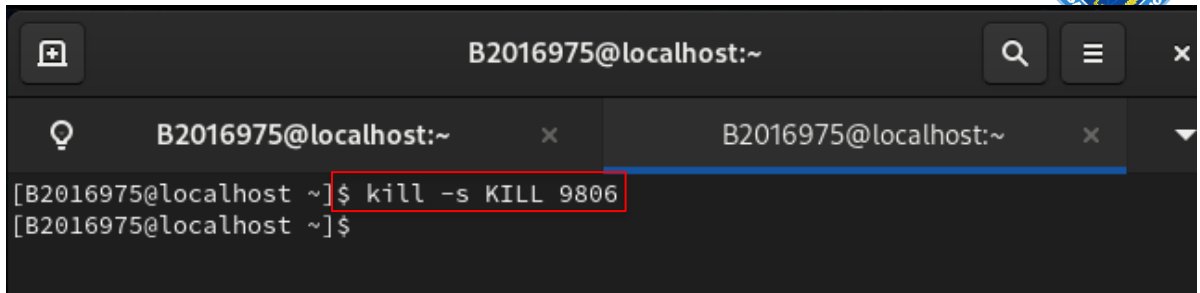


Hình 26 Dùng lệnh `kill -s CONT` để phục hồi trạng thái trước đó của firefox



Hình 27 Firefox đã phục hồi trạng thái đang chạy sau khi dùng lệnh `CONT`

#### 4.5. Dùng lệnh `kill` để hủy tiến trình firefox (chụp hình minh họa).



```

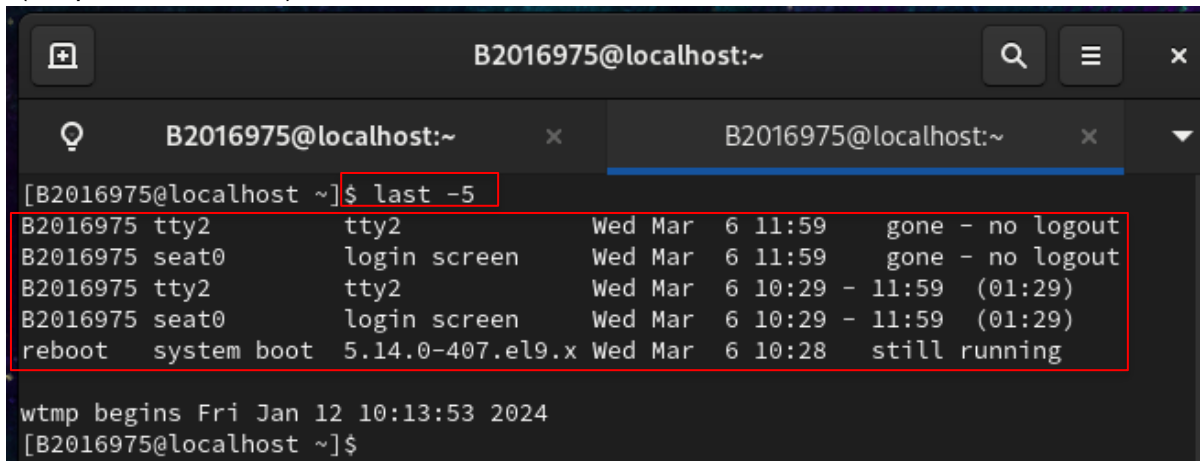
B2016975@localhost:~
B2016975@localhost:~
[B2016975@localhost ~]$ kill -s KILL 9806
[B2016975@localhost ~]$
    
```

Hình 28 Firefox đã chấm dứt bằng lệnh `kill -s KILL`

### 5. Tập tin log

Tìm hiểu và thực hiện các yêu cầu sau:

#### 5.1. Tìm thông tin về người dùng, thời gian của 5 lần đăng nhập sau cùng vào hệ thống (chụp hình minh họa).



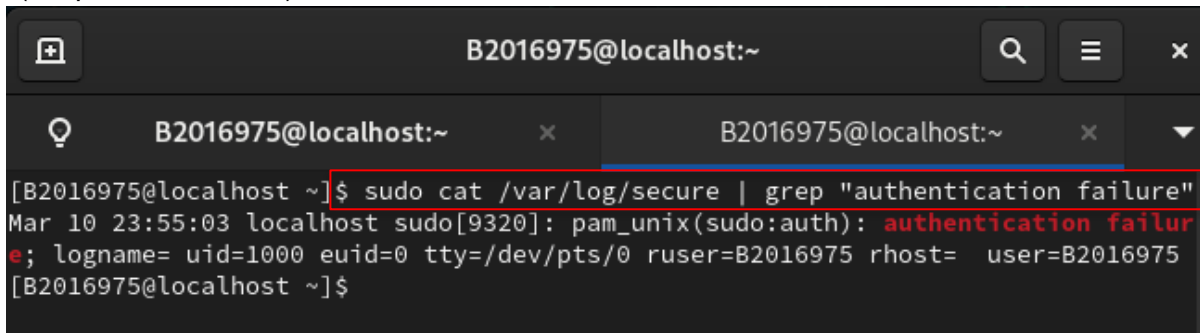
```

B2016975@localhost:~
B2016975@localhost:~
[B2016975@localhost ~]$ last -5
B2016975 tty2      tty2      Wed Mar  6 11:59   gone - no logout
B2016975 seat0    login screen Wed Mar  6 11:59   gone - no logout
B2016975 tty2      tty2      Wed Mar  6 10:29 - 11:59   (01:29)
B2016975 seat0    login screen Wed Mar  6 10:29 - 11:59   (01:29)
reboot  system boot  5.14.0-407.el9.x Wed Mar  6 10:28   still running

wtmp begins Fri Jan 12 10:13:53 2024
[B2016975@localhost ~]$
    
```

Hình 29 Dùng lệnh `last -5` để tìm thông tin về người dùng, thời gian của 5 lần đăng nhập vào hệ thống

#### 5.2. Hiển thị thông tin các lần đăng nhập KHÔNG thành công vào hệ thống gần đây nhất (chụp hình minh họa).



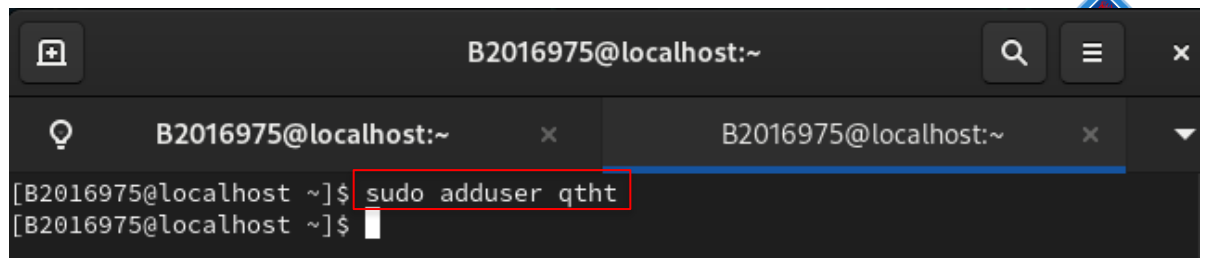
```

B2016975@localhost:~
B2016975@localhost:~
[B2016975@localhost ~]$ sudo cat /var/log/secure | grep "authentication failure"
Mar 10 23:55:03 localhost sudo[9320]: pam_unix(sudo:auth): authentication failure; logname= uid=1000 euid=0 tty=/dev/pts/0 ruser=B2016975 rhost= user=B2016975
[B2016975@localhost ~]$
    
```

Hình 30 dùng lệnh `cat /var/log/secure | grep "authentication failure"` để xem thông tin các tiến trình đăng nhập không thành công vào hệ thống gần nhất

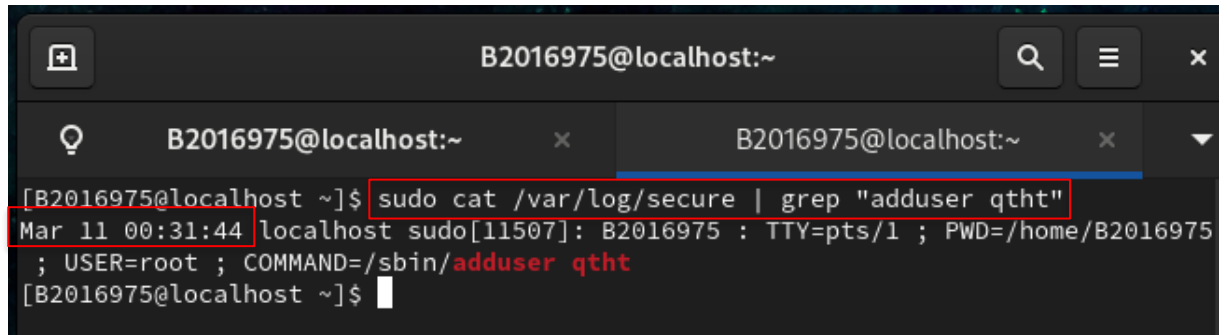
#### 5.3. Tạo một người dùng mới `qtht`. Tìm thời gian người dùng được tạo ra (chụp hình minh họa).





```
B2016975@localhost:~  
[B2016975@localhost ~]$ sudo adduser qtht  
[B2016975@localhost ~]$
```

Hình 31 Tạo ra người dùng qtht

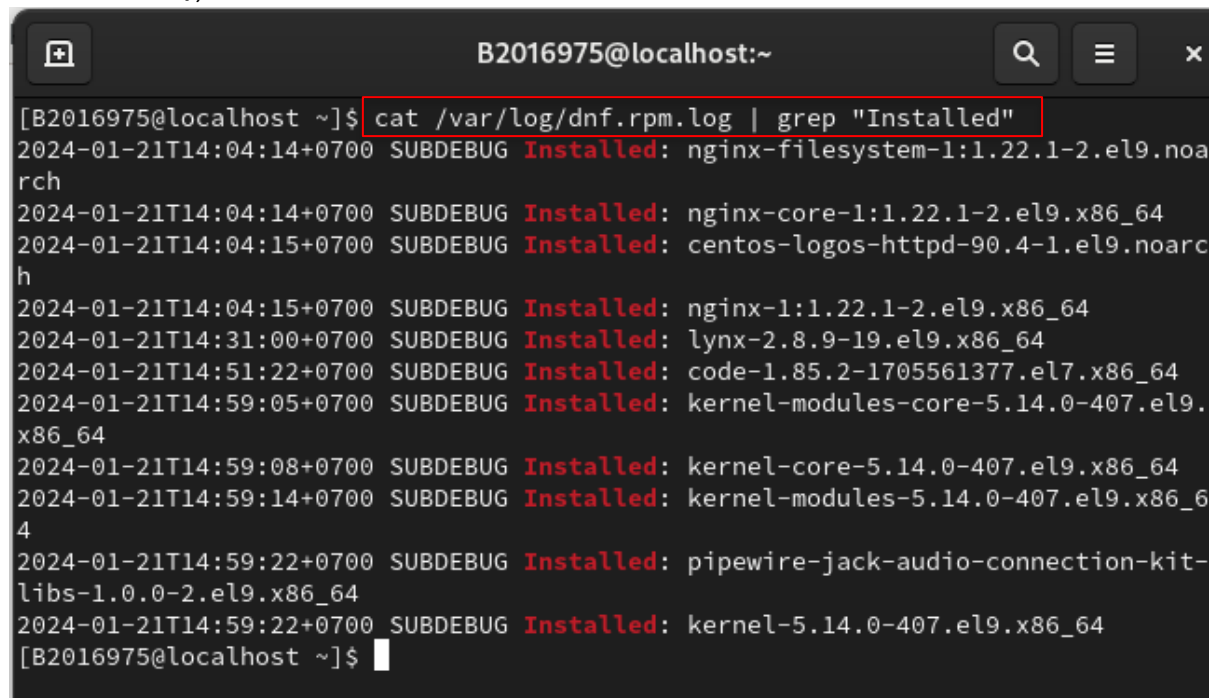


```
B2016975@localhost:~  
[B2016975@localhost ~]$ sudo cat /var/log/secure | grep "adduser qtht"  
Mar 11 00:31:44 localhost sudo[11507]: B2016975 : TTY=pts/1 ; PWD=/home/B2016975  
; USER=root ; COMMAND=/sbin/adduser qtht  
[B2016975@localhost ~]$
```

Hình 32 Dùng lệnh `cat /var/log/secure | grep "adduser qtht"` để xem thời gian người dùng quản trị hệ thống được thêm vào hệ thống

Thời gian người dùng quản trị hệ thống được thêm vào là : 00:31:44h ngày 11 tháng 03.

- 5.4. Tìm thông tin tên và thời gian của phần mềm được cài vào hệ thống gần đây (chụp hình minh họa).



```
B2016975@localhost:~  
[B2016975@localhost ~]$ cat /var/log/dnf.rpm.log | grep "Installed"  
2024-01-21T14:04:14+0700 SUBDEBUG Installed: nginx-filesystem-1:1.22.1-2.el9.noar  
rch  
2024-01-21T14:04:14+0700 SUBDEBUG Installed: nginx-core-1:1.22.1-2.el9.x86_64  
2024-01-21T14:04:15+0700 SUBDEBUG Installed: centos-logos-httpd-90.4-1.el9.noarc  
h  
2024-01-21T14:04:15+0700 SUBDEBUG Installed: nginx-1:1.22.1-2.el9.x86_64  
2024-01-21T14:31:00+0700 SUBDEBUG Installed: lynx-2.8.9-19.el9.x86_64  
2024-01-21T14:51:22+0700 SUBDEBUG Installed: code-1.85.2-1705561377.el7.x86_64  
2024-01-21T14:59:05+0700 SUBDEBUG Installed: kernel-modules-core-5.14.0-407.el9.  
x86_64  
2024-01-21T14:59:08+0700 SUBDEBUG Installed: kernel-core-5.14.0-407.el9.x86_64  
2024-01-21T14:59:14+0700 SUBDEBUG Installed: kernel-modules-5.14.0-407.el9.x86_6  
4  
2024-01-21T14:59:22+0700 SUBDEBUG Installed: pipewire-jack-audio-connection-kit-  
libs-1.0.0-2.el9.x86_64  
2024-01-21T14:59:22+0700 SUBDEBUG Installed: kernel-5.14.0-407.el9.x86_64  
[B2016975@localhost ~]$
```

Hình 33 Dùng lệnh `cat /var/log/dnf.rpm.log | grep "Installed"` để tìm thông tin và thời gian của phần mềm được cài vào hệ thống gần đây.



--- Hết ---

**Video hướng dẫn làm bài:**

- + Hướng dẫn làm bài: <https://youtu.be/MgrW8zeh02E>
- + Hướng dẫn câu 1:
  - Cài đặt VirtualBox: <https://youtu.be/JC-tXZmgXII>
  - Cài đặt CentOS 9 stream: <https://youtu.be/DG8-FA0vCY4>
- + Hướng dẫn câu 2: [https://youtu.be/YVLB1\\_m27ko](https://youtu.be/YVLB1_m27ko)
- + Hướng dẫn câu 3: <https://youtu.be/NK8tXW06oLI>
- + Hướng dẫn câu 4: <https://youtu.be/Ynb1dNGmiH4>
- + Hướng dẫn câu 5: <https://youtu.be/0QQVqpP6Jww>