



### LAB 3

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

Họ tên và MSSV: Lê Hải Đăng - B2203716

Nhóm học phần: CT179/03

*- 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.*

### 1. Cài đặt CentOS

Thực hiện cài đặt CentOS 9 vào máy tính cá nhân (hoặc máy ảo) của bạn.

### 2. Quản trị với shell scripting

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

Hostname: xác định tên địa chỉ host.

```
[b2203716@localhost ~]$ hostname  
haidang  
[b2203716@localhost ~]$
```

hostname -I: hiển thị địa chỉ ip của host.

```
[b2203716@localhost ~]$ hostname -I  
10.0.2.15  
[b2203716@localhost ~]$
```

Whoami: xuất ra tên tài khoản người dùng đang đăng nhập.

```
[b2203716@localhost ~]$ whoami  
b2203716  
[b2203716@localhost ~]$
```

df -H: liệt kê các phân vùng của ổ cứng, thư mục mà nó đang mount tới và dung lượng của ổ cứng đang được sử dụng là bao nhiêu.

```
[b2203716@localhost ~]$ df -H
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.2M   0   4.2M   0% /dev
tmpfs           2.5G   0   2.5G   0% /dev/shm
tmpfs           972M  9.6M  962M   1% /run
/dev/mapper/cs-root 19G  5.8G   13G  32% /
/dev/sda1       1.1G  477M  531M  48% /boot
/dev/sdb1       11G   25k   10G   1% /data
tmpfs           486M  115k  486M   1% /run/user/1000
[b2203716@localhost ~]$
```

```
ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3:
```

- pid: mã số tiến trình
- %mem: phần trăm bộ nhớ
- %cpu: phần trăm cpu
- comm: lệnh thực thi tiến trình
- --sort -rss: sắp xếp theo phần trăm bộ nhớ giảm dần
- head -n 3: hiển thị 3 dòng đầu tiên của kết quả

```
[b2203716@localhost ~]$ ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3
PID %MEM %CPU COMMAND
2446  7.7  0.9 gnome-software
2192  7.1  8.1 gnome-shell
[b2203716@localhost ~]$
```

(KHÔNG CÓ KHOẢNG TRẮNG SAU DẤU PHẨY)

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

```
b2203716@localhost:~ — nano ./info.sh
GNU nano 5.6.1                               ./info.sh
echo "-----"
hostname
echo "-----"
hostname -I
echo "-----"
whoami
echo "-----"
df -H
echo "-----"
ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3
echo "---Thuc thi 2.1 hoang thanh---"
```

```
[b2203716@localhost ~]$ ./info.sh
-----
haidang
-----
10.0.2.15
-----
b2203716
-----
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.2M   0   4.2M   0% /dev
tmpfs           2.5G   0   2.5G   0% /dev/shm
tmpfs           972M  9.6M  962M   1% /run
/dev/mapper/cs-root 19G  5.8G   13G  32% /
/dev/sda1       1.1G  477M  531M  48% /boot
/dev/sdb1       11G   25k   10G   1% /data
tmpfs           486M  111k  486M   1% /run/user/1000
-----
  PID %MEM %CPU COMMAND
    2446  7.7  0.5 gnome-software
    2192  7.1  7.5 gnome-shell
---Thực thi 2.1 hoang thanh---
[b2203716@localhost ~]$
```

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

- In ra ngày giờ hiện tại
- Nén toàn bộ thư mục **/home** thành tập tin **/backup/<YYYY-MM-DD>.tar** (YYYY-MM-DD là ngày hiện tại, ví dụ: 2024-10-07.tar)
- In thông tin đầy đủ của tập tin **/backup/<YYYY-MM-DD>.tar**
- In 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 "-----***-----"
echo "- Ngày giờ hiện tại: "
date
echo "-----"
filename="/backup/$(date +%F).tar"
sudo tar -cf $filename /home
echo "-----"
ls -l $filename
echo "Sao lưu thành công!!!!"
```

```
[b2203716@haidang ~]$ ./backup.sh
-----***-----
- Ngay gio hien tai:
Mon Oct 7 02:32:31 PM +07 2024
-----
tar: Removing leading `/' from member names
-----
-rw-r--r--. 1 root root 161484800 Oct 7 14:32 /backup/2024-10-07.tar
Sao lưu thành công!!!
[b2203716@haidang ~]$
```

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

Cron là một tiện ích trong Linux cho phép máy tính thiết lập thời gian biểu để 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 lặp lại việc thực thi. Để hiệu chỉnh file crontab với trình soạn thảo nano, ta dùng các lệnh sau:

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

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

- 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)

```
* * * * * (date >> /tmp/date.txt)
```

```
[b2203716@localhost ~]$ crontab -e
crontab: installing new crontab
[b2203716@localhost ~]$ cat /tmp/date.txt
echo "---Tap tin /tmp/data.txt---"
Sat Oct 5 04:26:01 PM +07 2024
Sat Oct 5 04:27:01 PM +07 2024
Sat Oct 5 04:28:01 PM +07 2024
Sat Oct 5 04:29:01 PM +07 2024
Sat Oct 5 04:30:01 PM +07 2024
[b2203716@localhost ~]$
```

- 3.2.** Thực thi **backup.sh** ở 2.3 vào 23:30 phút ngày 10, 20 và 28 hàng tháng (chụp hình minh họa).

```
GNU nano 5.6.1 /tmp/crontab.4RWI6T
* * * * * (date >> /tmp/date.txt)
30 23 10,20,28 * * (sudo /home/b2203716/backup.sh)
```

### 4. Thao tác với tiến trình

Tìm hiểu và thực hiện các yêu cầu sau:

- 4.1.** Mở trình duyệt Firefox. Sau đó dùng lệnh **ps -aux** hoặc **pgrep** tìm PID của firefox. Tiếp theo, 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?

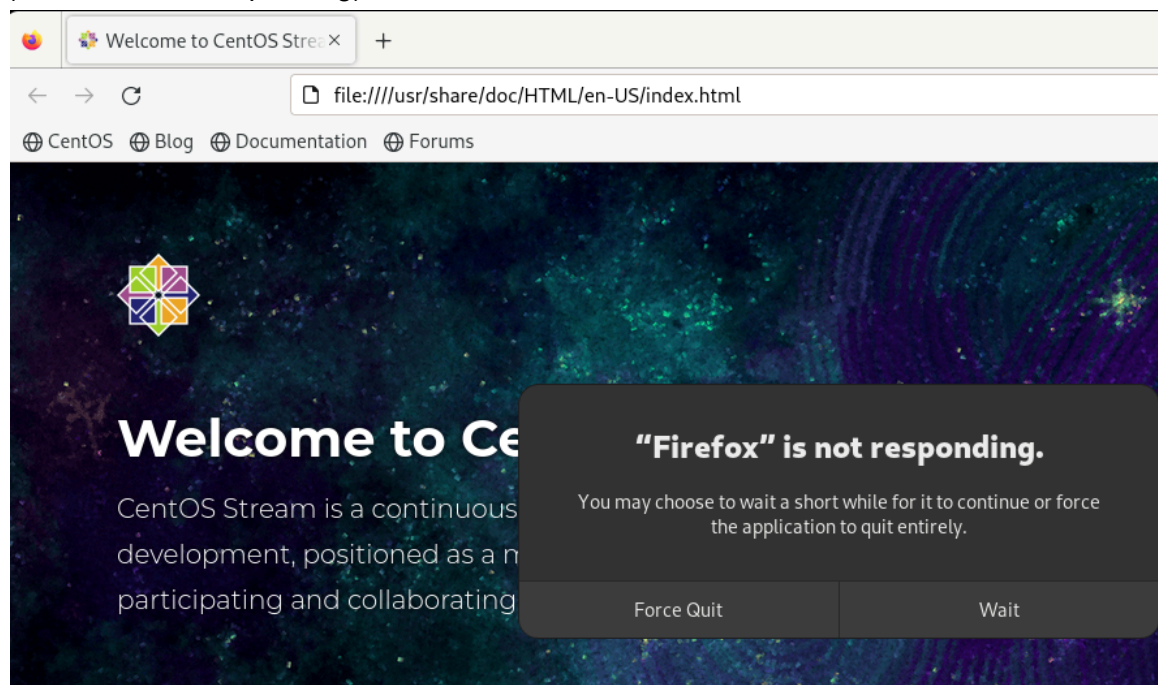
- Tìm mã tiến trình firefox

```
[b2203716@haidang ~]$ pgrep firefox
4483
[b2203716@haidang ~]$
```

- Chạy lệnh tạm dừng trên tiến trình

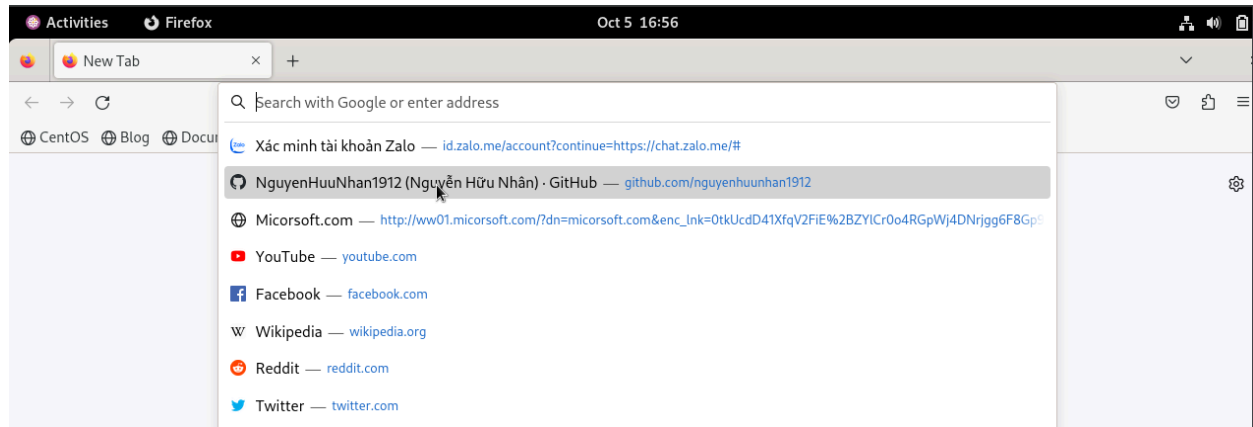
```
[b2203716@haidang ~]$ kill -19 4483
[b2203716@haidang ~]$
```

- Ta thấy tiến trình firefox khi dừng nó sẽ hiển thị không phản hồi ("FireFox" í not responding)



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

```
[b2203716@haidang ~]$ kill -18 4483
[b2203716@haidang ~]$
```



4.3. Dùng lệnh **kill** để hủy tiến trình firefox (chụp hình minh họa).

```
[b2203716@haidang ~]$ pgrep firefox
5229
[b2203716@haidang ~]$ kill -9 5229
[b2203716@haidang ~]$
```

## 5. Tập tin log

Tìm hiểu và thực hiện các yêu cầu sau:

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

```
[b2203716@haidang ~]$ last | head -n 1
b2203716 tty2          tty2                Sat Oct  5 15:15    still logged in
[b2203716@haidang ~]$
```

5.2. Tạo một người dùng mới.

```
[b2203716@haidang ~]$ sudo useradd haidangleee
[sudo] password for b2203716:
[b2203716@haidang ~]$
```

5.3. Tìm thời gian người dùng ở 5.2 được tạo ra (chụp hình minh họa)

```
[b2203716@haidang ~]$ sudo less /var/log/secure | grep haidangleee
Oct  5 17:00:43 localhost sudo[5639]: b2203716 : TTY=pts/1 ; PWD=/home/b2203716 ; USER=root ; COMMAND=/sbin/useradd haidangleee
Oct  5 17:00:43 localhost useradd[5649]: new group: name=haidangleee, GID=1007
Oct  5 17:00:43 localhost useradd[5649]: new user: name=haidangleee, UID=1003, GID=1007, home=/home/haidangleee, shell=/bin/bash, from=/dev/pts/1
[b2203716@haidang ~]$
```

--- Hết ---