



LAB 3

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

Họ tên và MSSV: Trần Đăng Khoa B2014926

Nhóm học phần: M02

- 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 Stream vào máy tính cá nhân (hoặc máy ảo) của bạn **nếu cần** (KHÔNG cần chụp hình minh họa).

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
```

```
hostname -I
```

```
whoami
```

```
df -H
```

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

(KHÔNG CÓ KHOẢNG TRẮNG SAU DẤU PHẨY)

```

trandangkhoa@localhost:~
[trandangkhoa@localhost ~]$ hostname
localhost.localdomain
[trandangkhoa@localhost ~]$ hostname -I
10.0.2.15
[trandangkhoa@localhost ~]$ whoami
trandangkhoa
[trandangkhoa@localhost ~]$ df -H

```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	2.3G	0	2.3G	0%	/dev
tmpfs	2.4G	0	2.4G	0%	/dev/shm
tmpfs	924M	9.6M	915M	2%	/run
/dev/mapper/cs-root	19G	5.3G	13G	29%	/
/dev/sda1	1.1G	278M	787M	27%	/boot
tmpfs	462M	107k	462M	1%	/run/user/1000

```

[trandangkhoa@localhost ~]$

```

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

```

[trandangkhoa@localhost ~]$ nano ./info.sh
[trandangkhoa@localhost ~]$ su
Password:
[root@localhost trandangkhoa]# chmod a+rx ./info.sh
[root@localhost trandangkhoa]# ls -l ./info.sh
-rwxr-xr-x. 1 trandangkhoa trandangkhoa 408 Mar 27 08:37 ./info.sh
[root@localhost trandangkhoa]# ./info.sh
.....
localhost.localdomain
.....
10.0.2.15
.....
./info.sh: line 6: whoaim: command not found
.....

```

Filesystem	Size	Used	Avail	Use%	Mounted on
devtmpfs	2.3G	0	2.3G	0%	/dev
tmpfs	2.4G	0	2.4G	0%	/dev/shm
tmpfs	924M	13M	912M	2%	/run
/dev/mapper/cs-root	19G	5.3G	13G	29%	/
/dev/sda1	1.1G	278M	787M	27%	/boot
tmpfs	462M	99k	462M	1%	/run/user/1000

```

.....

```

- 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 `/tmp/<YYYY-MM-DD>.tar` (YYYY-MM-DD là ngày hiện tại, ví dụ: 2020-04-22.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 hoạ).

```
[root@localhost trandangkhoa]# date
Mon Mar 27 08:49:21 AM +07 2023
[root@localhost trandangkhoa]# date +%F
2023-03-27
[root@localhost trandangkhoa]# nano ./backup.sh
[root@localhost trandangkhoa]# chmod a+rx ./backup.sh
[root@localhost trandangkhoa]# ./backup.sh
bat dau sao luu du lieu
.....
ngay gio he thong
Mon Mar 27 08:49:59 AM +07 2023
.....
tar: Removing leading '/' from member names
.....
thong tin du lieu sao luu
-rw-r--r--. 1 root root 5765120 Mar 27 08:49 /tmp/2023-03-27.tar
sao luu thanh cong!!!!
[root@localhost trandangkhoa]#
```

- 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).

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
    echo safe_rm_cycle
else
    echo "Warning :The recycling directory already exists."
fi
cp $1 safe_rmrecycle
rm $1
```

```
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
    #Thêm 1 dòng lệnh ở đây
```

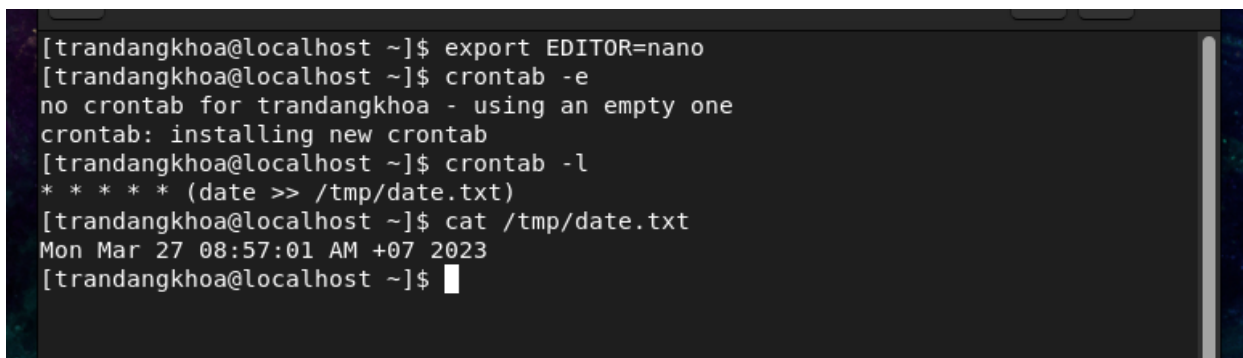
3. Lên lịch công việc định kỳ với cron

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)



```
[trandangkhoa@localhost ~]$ export EDITOR=nano
[trandangkhoa@localhost ~]$ crontab -e
no crontab for trandangkhoa - using an empty one
crontab: installing new crontab
[trandangkhoa@localhost ~]$ crontab -l
* * * * * (date >> /tmp/date.txt)
[trandangkhoa@localhost ~]$ cat /tmp/date.txt
Mon Mar 27 08:57:01 AM +07 2023
[trandangkhoa@localhost ~]$
```

- 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).

```
[trandangkhoa@localhost ~]$  
[trandangkhoa@localhost ~]$ crontab -e  
crontab: installing new crontab  
[trandangkhoa@localhost ~]$ crontab -l  
* * * * * (date >> /tmp/date.txt)  
0 23 10,20,30 * * (/home/tdkhoa/backup.sh)  
[trandangkhoa@localhost ~]$
```

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 (chụp hình minh họa)?
- 4.2. Dùng lệnh `kill` để phục hồi trạng thái trước đó của firefox và quan sát kết quả (chụp hình minh họa).
- 4.3. Dùng lệnh `kill` để hủy tiến trình firefox (chụp hình minh họa).

```
[trandangkhoa@localhost ~]$ pgrep firefox  
3459  
[trandangkhoa@localhost ~]$ kill -s STOP 3459  
[trandangkhoa@localhost ~]$ kill -s CONT 3459  
[trandangkhoa@localhost ~]$ kill -s KILL 3459  
[trandangkhoa@localhost ~]$
```

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).

```
[trandangkhoa@localhost ~]$ last  
trandang tty2          tty2          Mon Mar 27 09:14    still logged in  
reboot  system boot      5.14.0-124.el9.x Mon Mar 27 09:14    still running  
trandang tty2          tty2          Mon Mar 27 08:35    - down (00:38)  
reboot  system boot      5.14.0-124.el9.x Mon Mar 27 08:34    - 09:13 (00:38)  
trandang tty2          tty2          Mon Mar 27 07:56    - down (00:37)  
trandang tty2          tty2          Mon Mar 27 07:38    - 07:55 (00:17)  
reboot  system boot      5.14.0-124.el9.x Mon Mar 27 07:37    - 08:34 (00:56)  
  
wtmp begins Mon Mar 27 07:37:39 2023  
[trandangkhoa@localhost ~]$ last | head -n 1  
trandang tty2          tty2          Mon Mar 27 09:14    still logged in
```

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

```
[trandangkhoa@localhost ~]$ sudo adduser corona
adduser: user 'corona' already exists
[trandangkhoa@localhost ~]$ sudo less /var/log/secure
[trandangkhoa@localhost ~]$ sudo cat /var/log/secure | grep corona
Mar 27 09:33:04 localhost sudo[3374]: trandangkhoa : TTY=pts/0 ; PWD=/home/trand
angkhoa ; USER=root ; COMMAND=/sbin/adduser corona
Mar 27 09:33:05 localhost useradd[3376]: new group: name=corona, GID=1002
Mar 27 09:33:05 localhost useradd[3376]: new user: name=corona, UID=1002, GID=10
02, home=/home/corona, shell=/bin/bash, from=/dev/pts/0
Mar 27 09:34:25 localhost sudo[3436]: trandangkhoa : TTY=pts/0 ; PWD=/home/trand
angkhoa ; USER=root ; COMMAND=/sbin/adduser corona
Mar 27 09:34:25 localhost useradd[3438]: failed adding user 'corona', exit code:
9
[trandangkhoa@localhost ~]$
```

--- Hết ---