

**LAB 3**

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

| Họ tên và MSSV: Lê Trương Ngọc Duyên B2105569  Nhóm học phần: CT179-06 |
| --- |

*- 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).

1. **Shell scripting**
   1. Thực hiện các lệnh bên dưới và cho biết ý nghĩa của chúng (chụp hình minh hoạ):

hostname : hiển thị hostname hiện tại của hệ thống

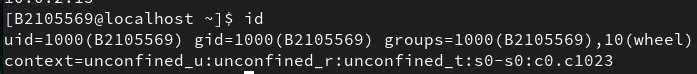


hostname -I : hiển thị địa chỉ IP của máy chủ



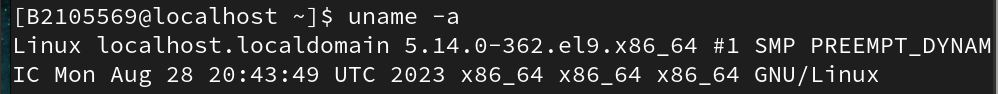
id : hiển thị thông tin người dùng hoàn chỉnh

* Mã số người dùng, tên đăng nhập của người dùng, nhóm mặc nhiên của người dùng, các nhóm mà người dùng thuộc vào



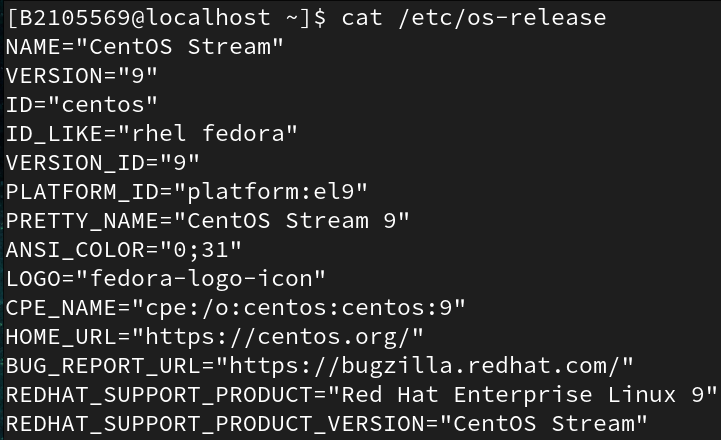
uname -a : hiển thị thông tin về phiên bản Linux mà ta sử dụng trong hệ điều hành CentOS9.

Như VD thì là phiên bản Linux 5.14

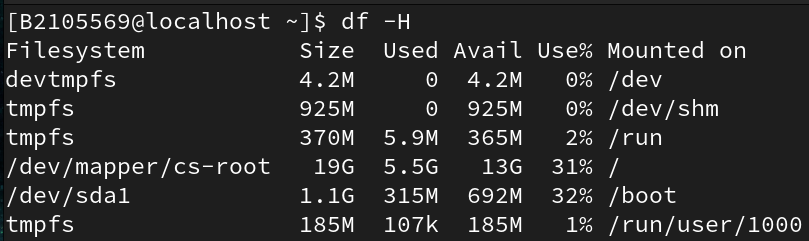


cat /etc/os-release : hiển thị phiên bản hệ điều hành Linux

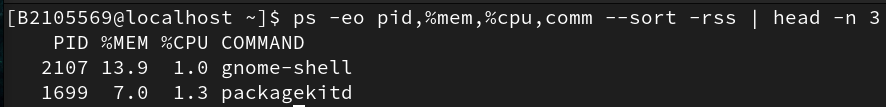
* Tên hệ điều hành, phiên bản hệ điều hành, ID phiên bản …



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

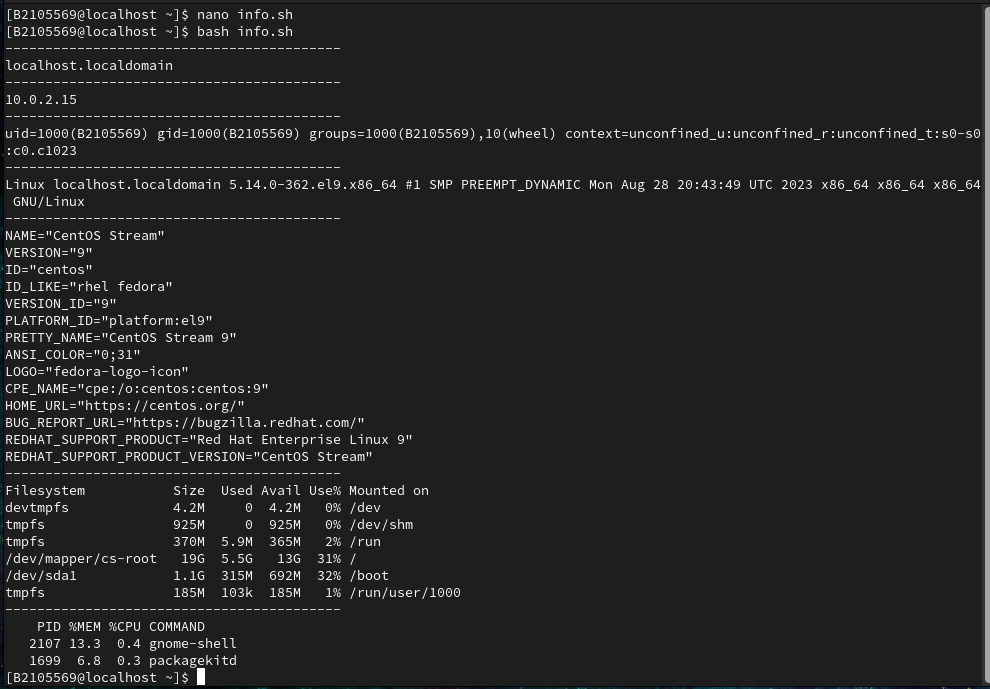


ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3 : chỉ in ra 3 hàng đầu thông tin các tiến trình (mã số tiến trình, phần trăm bộ nhớ, phần trăm CPU, lệnh thực thi tiến trình) có trên hệ điều hành, sắp xếp theo phần trăm bộ nhớ theo thứ tự giảm dần.



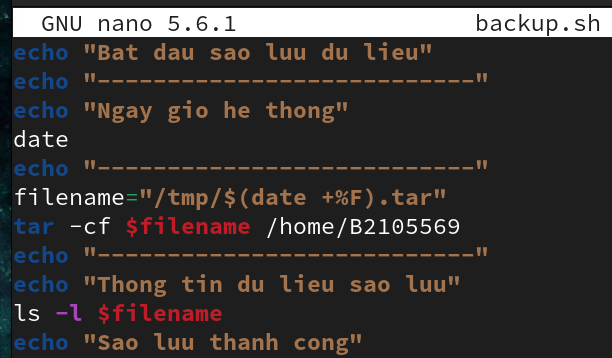
(KHÔNG CÓ KHOẢNG TRẮNG SAU DẤU PHẨY)

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

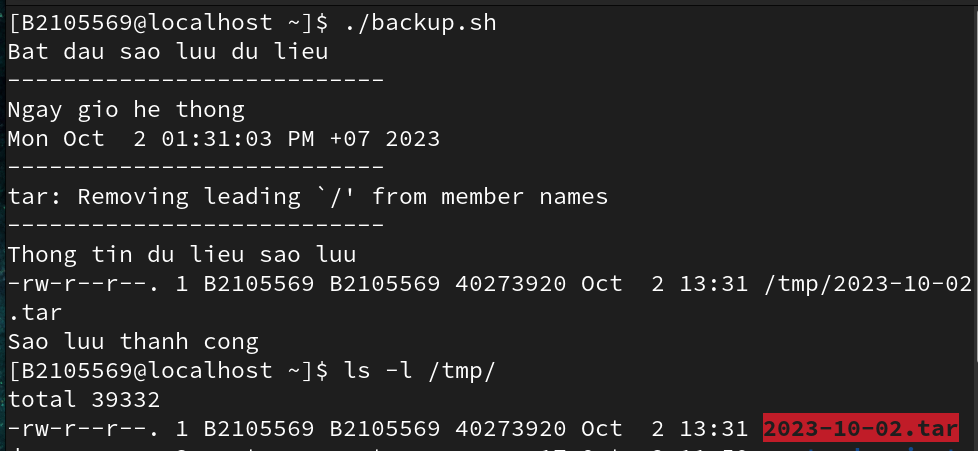


* 1. Viết shell script có tên backup.sh thực hiện:
* In ra ngày giờ hiện tại
* Nén toàn bộ thư mục cá nhân của người trong /homethành tập tin /tmp/<YYYY-MM-DD>.tar(YYYY-MM-DD là ngày hiện tại, ví dụ: 2023-09-25.tar)
* In thông tin đầy đủ của tập tin /tmp/<YYYY-MM-DD>.tar
* In thông ra thông báo “Sao lưu thành công!!!!”

Thực thi backup.sh để kiểm tra (chụp hình minh hoạ).



* date : in ra ngày giờ hiện tại
* filename=“/tmp/$(date +%F).tar” : Tạo ra biến filename là /tmp/<YYYY-MM-DD>.tar
* tar -cf $filename /home/B2105569 : Nén toàn bộ thư mục cá nhân của người trong /homethành tập tin có tên sử dụng biến filename



* 1. Bổ sung dòng các lệnh cần thiết vào shell script safe\_rm.sh bên dưới để thực hiện công việc sau:

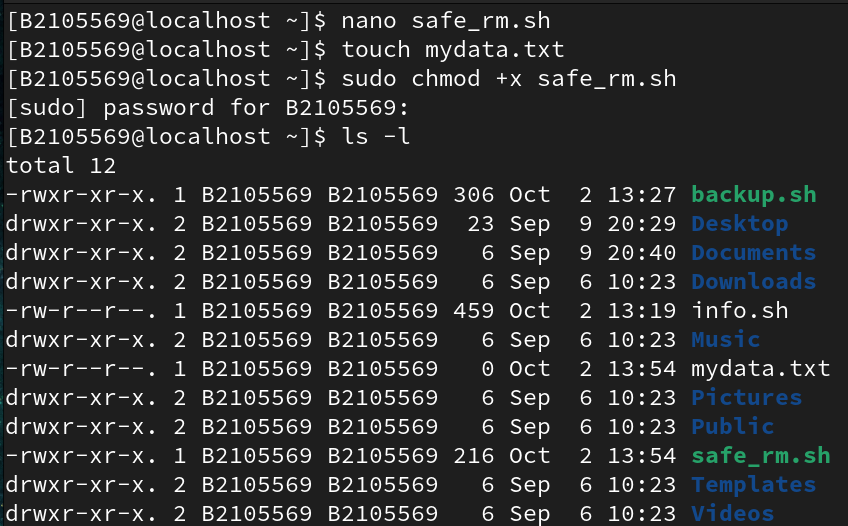
- Nhận 01 tham số từ dòng lệnh. Thông báo lỗi và thoát khỏi chương trình nếu không có tham số hoặc nhiều hơn 01 tham số.

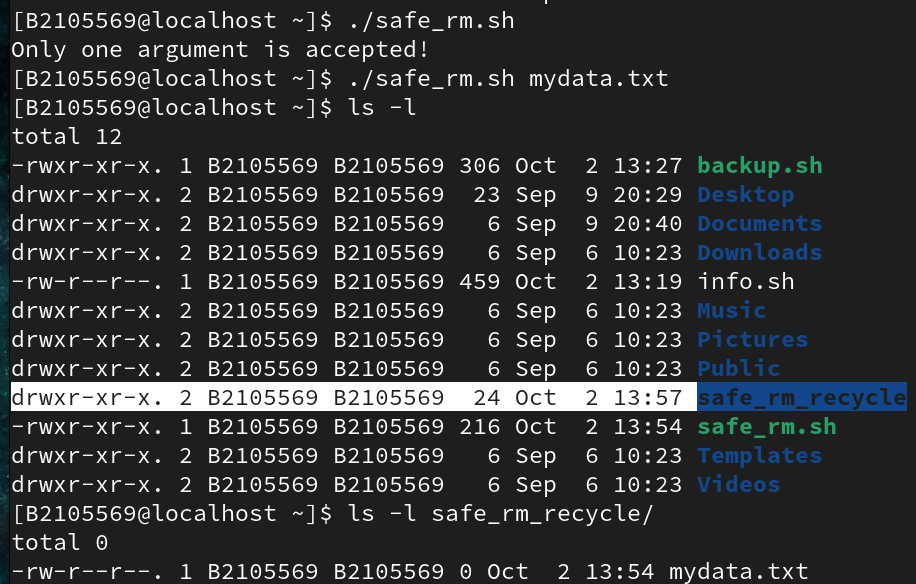
- Tạo thư mục với tên “safe\_rm\_recycle” nếu chưa có. Sao chép tập tin với tên là tham số 01 vào thư mục “safe\_rm\_recycle” vừa tạo. Sau đó xóa tập tin.

- Tạo tập tin có tên mydata.txt*,* thực thi safe\_rm.sh để xóa tập tin mydata.txt (chụp hình minh hoạ).

*safe\_rm.sh*

| if [ "$#" -ne 1 ]  then  echo "Only one argument is accepted!"  #Thêm 1 dòng lệnh ở đây  exit  fi  if [ ! -d "safe\_rm\_recycle" ]  then  #Thêm 1 dòng lệnh ở đây  mkdir safe\_rm\_recycle    else  echo "Warning: The recycling directory already exists."  fi  #Thêm 1 dòng lệnh ở đây  mv $1 safe\_rm\_recycle |
| --- |





1. **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

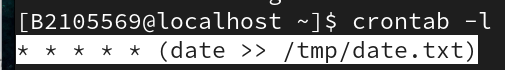
$ crontab -e

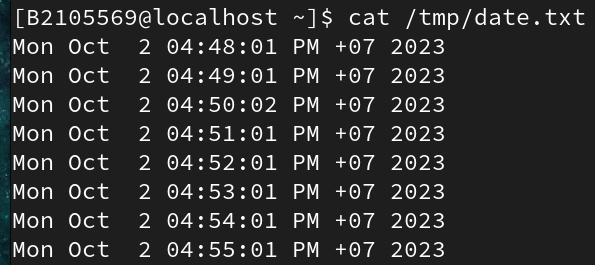


Cho biết cú pháp để thực hiện các yêu cầu sau từ crontab file:

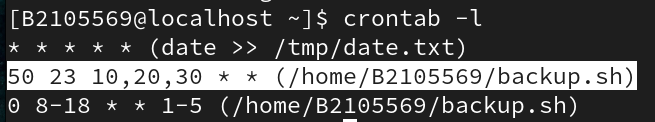
* 1. Chạy lệnh date mỗi phút một lần, sau đó ghi kết quả vào cuối tập tin /tmp/date.txt(chụp hình minh hoạ)

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

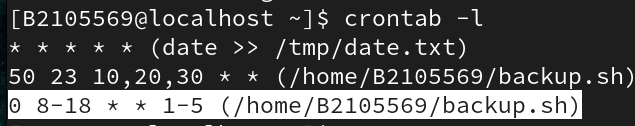




* 1. Thực thi backup.sh ở Câu **2.3** vào 23:50 giờ ngày 10,20 và 30 hàng tháng (chụp hình minh hoạ).



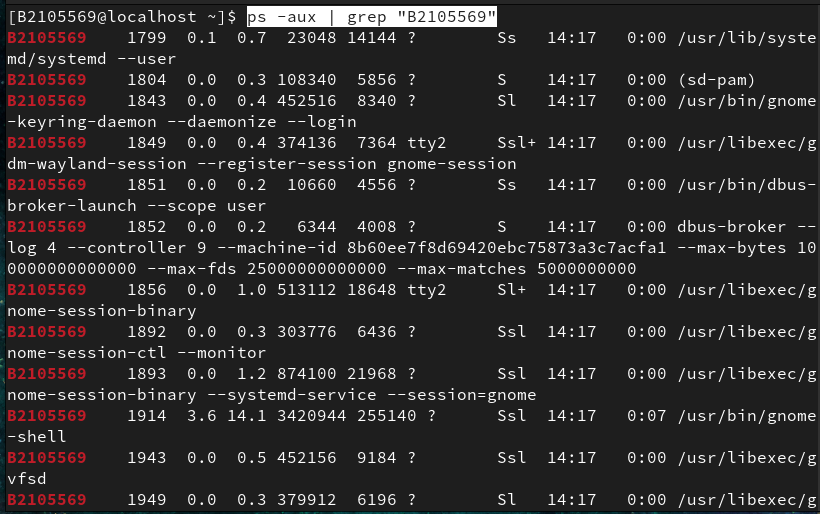
* 1. Thực thi backup.sh ở Câu **2.3** vào mỗi giờ 1 lần, từ 8:00 đến 18:00, trong các ngày làm việc (thứ hai đến thứ sáu) trong tuần (chụp hình minh hoạ).



1. **Quản lý tiến trình**

Tìm hiểu và thực hiện các yêu cầu sau:

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



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

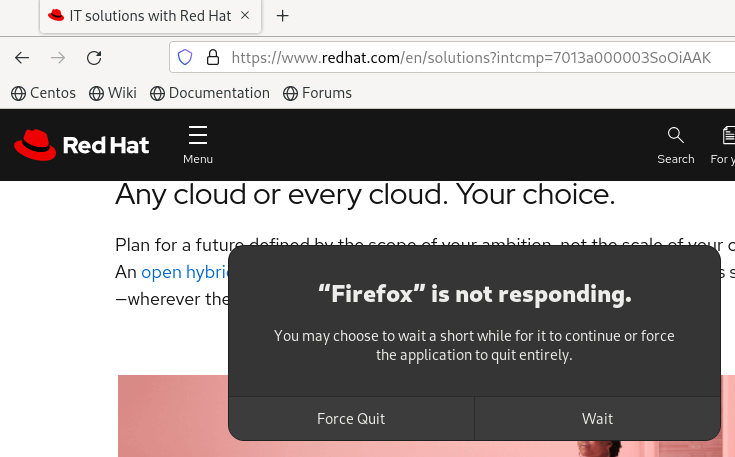




* 1. Dùng lệnh kill để tạm dừng tiến trình firefox (chụp hình minh hoạ). Điều gì xảy ra khi bạn dịch chuyển cửa sổ firefox hoặc nhấn chọn menu của nó ngay lúc này (chụp hình minh hoạ)?



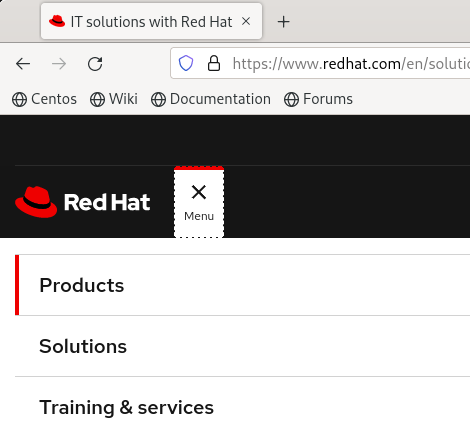
Khi dịch chuyển cửa sổ firefox hoặc nhấn chọn menu của nó ngay lúc này thì sẽ hiện lên bảng thông báo như phía dưới đây:



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



Kết quả:



* 1. Dùng lệnh kill để hủy tiến trình firefox (chụp hình minh hoạ).



1. **Tập tin log**

Tìm hiểu và thực hiện các yêu cầu sau:

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

Không thể dùng lệnh **cat /var/log/wtmp** để mở trực tiếp tập tin **wtmp** do nó có dạng nhị phân

⇒Dùng lệnh **last** để đọc dữ liệu từ bản ghi **wtmp** và hiển thị nó trong cửa sổ terminal

Để tìm thông tin về người dùng, thời gian của 5 lần đăng nhập sau cùng vào hệ thống có 2 cách:

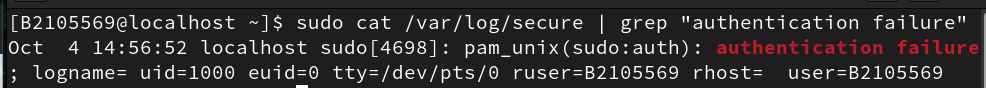
* **last -5**
* **last | head -n 5**



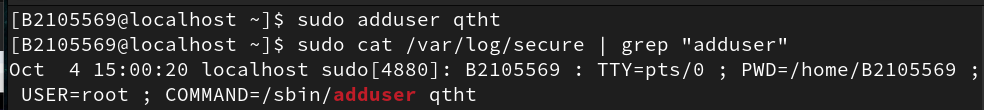
* 1. Hiển thị thông tin các lần đăng nhập KHÔNG thành công vào hệ thống gần đây nhất (chụp hình minh hoạ).

Không thể dùng lệnh **cat /var/log/btmp** để mở trực tiếp tập tin **btmp** do nó có dạng nhị phân

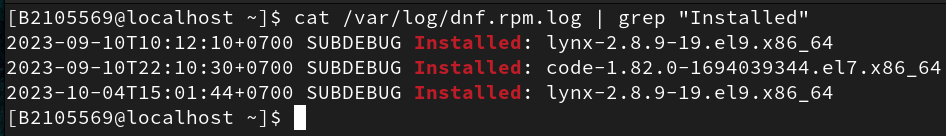
⇒Dùng lệnh **last** với tham số **-f** để truyền tên tập tin vào



* 1. Tạo một người dùng mới qtht. Tìm thời gian người dùng được tạo ra (chụp hình minh hoạ).



* 1. Tìm thông tin tên và thời gian của phần mềm được cài vào hệ thống gần đây (chụp hình minh hoạ).



--- Hết ---