

**LAB 3**

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

|  |
| --- |
| Họ tên và MSSV: Nguyễn Hoàng Huy - B2011967  Nhóm học phần: QTHT CT17901 |

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

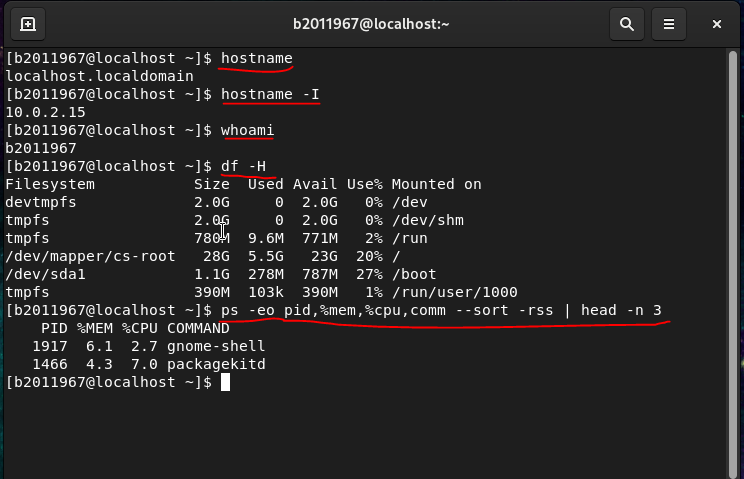
Hostname: xác định tên địa chỉ host trong mạng máy tính.

hostname -I: xác định địa chỉ ip của mạng máy tính.

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

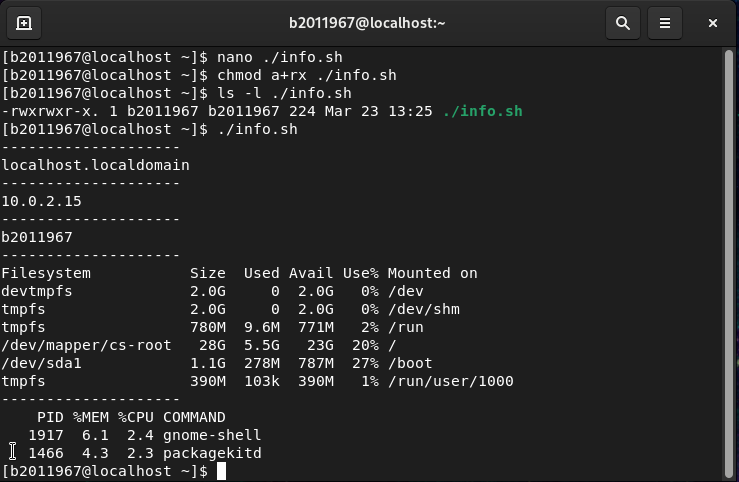
df -H: xem không gian đĩa trống trên mỗi phân vùng.

ps -eo pid,%mem,%cpu,comm --sort -rss | head -n 3: hiển thị một số tiến trình.



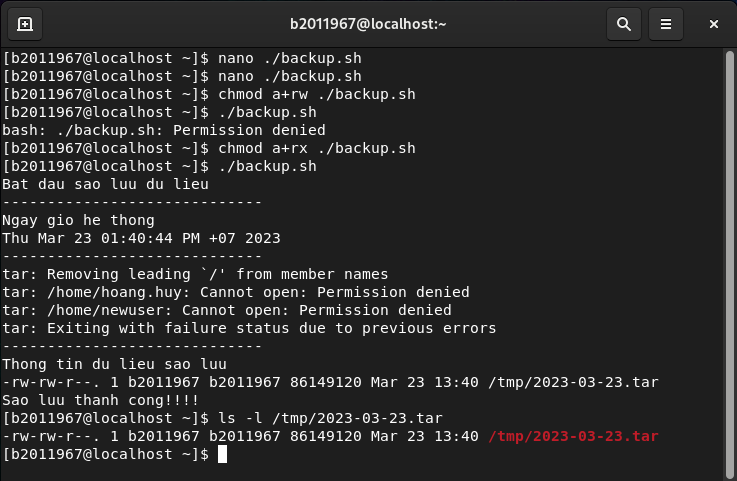
(KHÔNG CÓ KHOẢNG TRẮNG SAU DẤU PHẨY)

* 1. Viết shell script có tên info.sh 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 /homethành 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 /tmpn/<YYYY-MM-DD>.tar
* In thông ra thông báo “Sao lưu thành công!!!!”

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

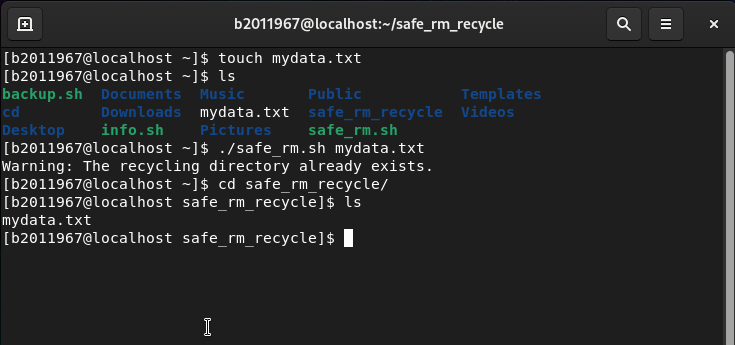


* 1. Bổ sung dòng các lệnh cần thiết vào shell script safe\_rm.sh bên dưới để thực hiện công việc sau:

- 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  cp “$1” ./safe\_rm\_recycle  #Thêm 1 dòng lệnh ở đây  rm “$1” |

1. **Lên lịch công việc định kỳ với cron**

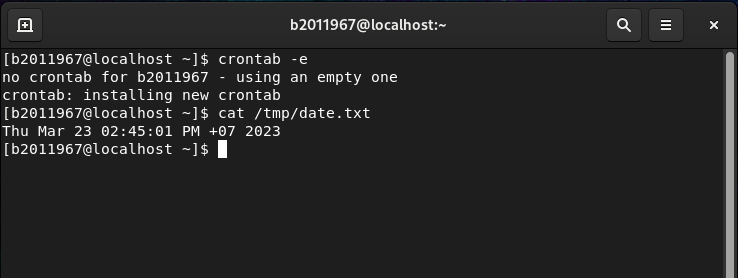
Cron là một tiện ích trong Linux cho phép máy tính thiết lập thời gian biểu để thực hiện công việc một cách định kỳ. Một crontab file chứa danh sách các lệnh sẽ được thực thi, kèm theo thông tin về thời điểm lặp lại việc thực thi. Để hiệu chỉnh file crontab với trình soạn thảo nano, ta dùng các lệnh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

export EDITOR=nano

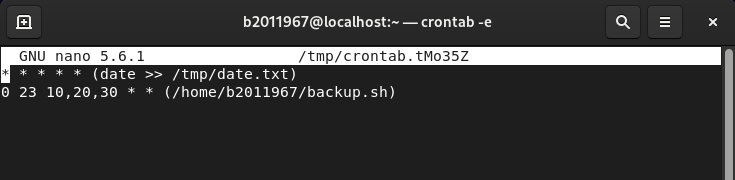
crontab -e

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

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



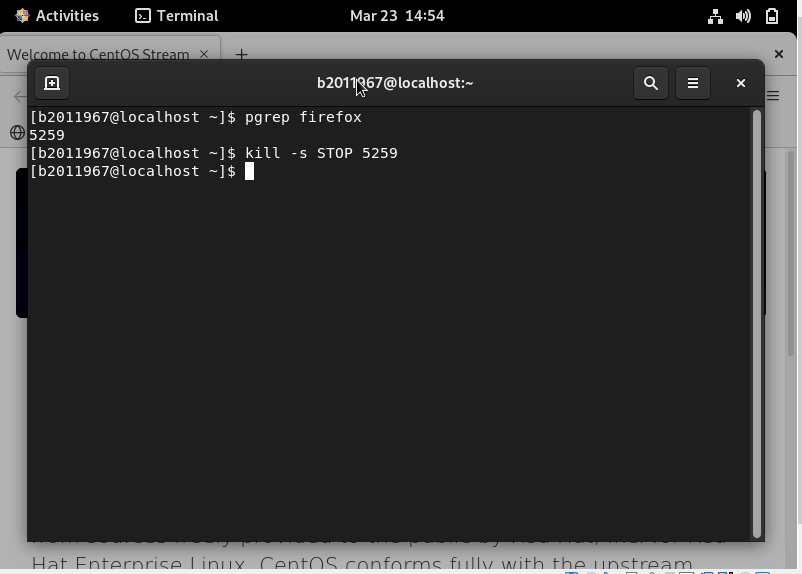
* 1. Thực thi backup.sh ở Câu **2.3** vào 23:50 giờ ngày 10,20 và 30 hàng tháng (chụp hình minh hoạ).



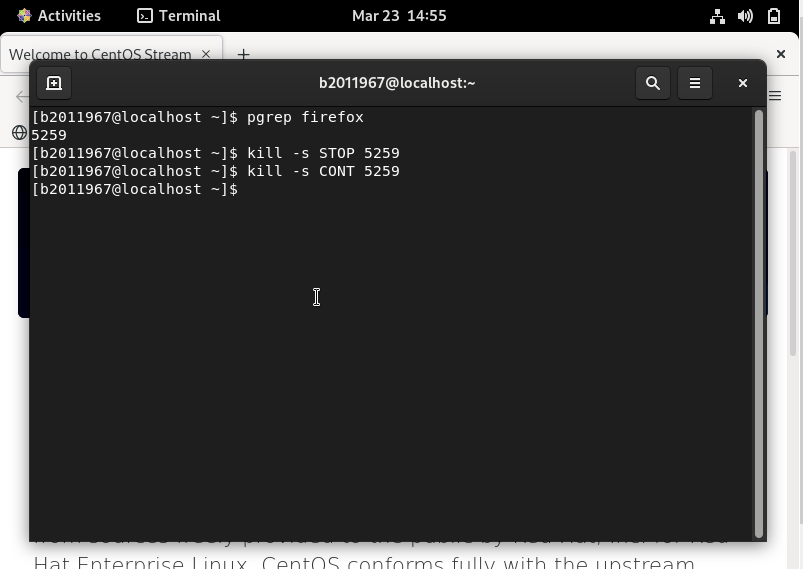
1. **Thao tác với tiến trình**

Tìm hiểu và thực hiện các yêu cầu sau:

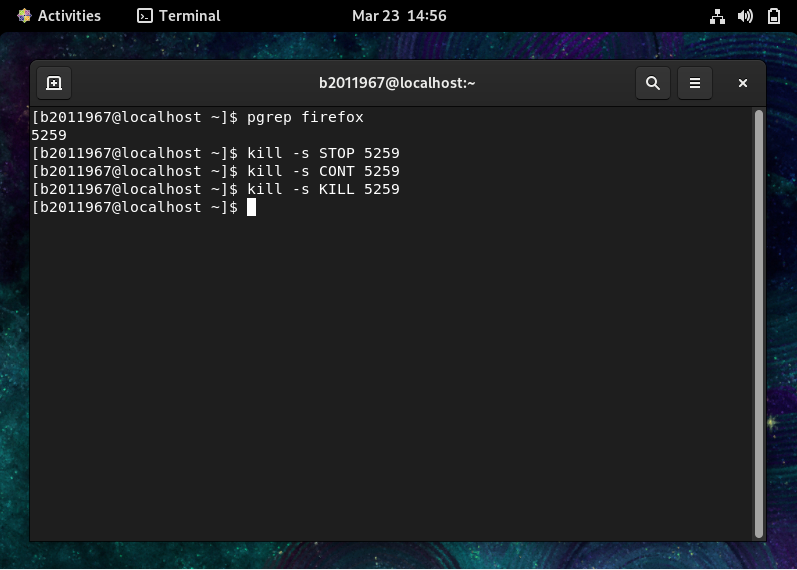
* 1. Mở trình duyệt Firefox. Sau đó dùng lệnh ps -auxhoặc pgrep tìm PID của firefox. Tiếp theo, dùng lệnh kill để tạm dừng tiến trình firefox (chụp hình minh hoạ). Điều gì xảy ra khi bạn dịch chuyển cửa sổ firefox hoặc nhấn chọn menu của nó ngay lúc này (chụp hình minh hoạ)?



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



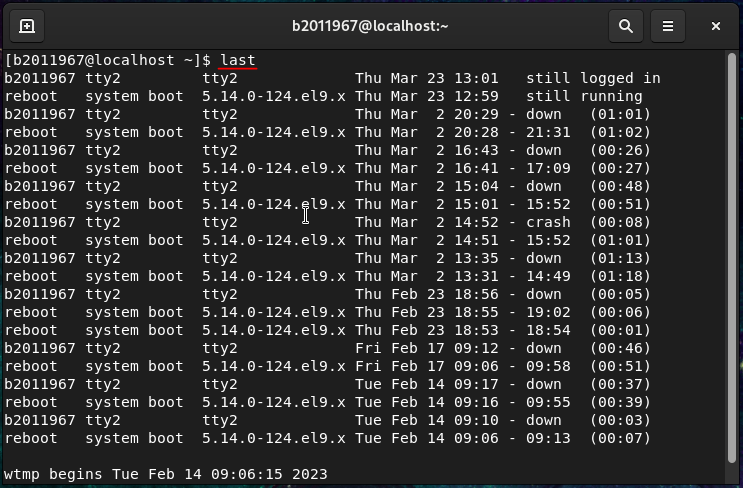
* 1. Dùng lệnh kill để hủy tiến trình firefox (chụp hình minh hoạ).



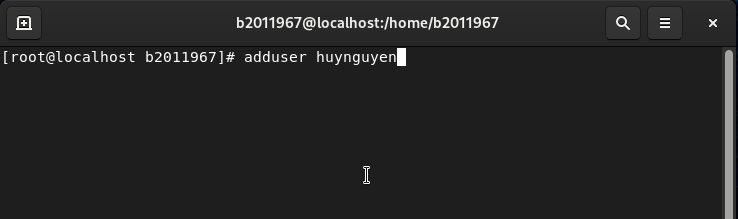
1. **Tập tin log**

Tìm hiểu và thực hiện các yêu cầu sau:

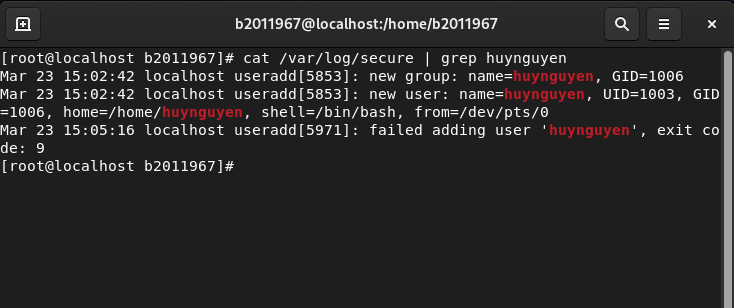
* 1. Tìm thông tin về người dùng, thời gian của lần đăng nhập sau cùng vào hệ thống (chụp hình minh hoạ).



* 1. Tạo một người dùng mới.



* 1. Tìm thời gian người dùng ở 5.2 được tạo ra (chụp hình minh hoạ).



--- Hết ---