**Linux Fundamentals 1**

Linux là 1 hệ điều hành được phát triển dựa trên Unix.

Giống như Windows có nhiều phiên bản 7,8,10 thì Linux cũng có nhiều bản phân phối như Ubuntu và Debian.

Vì Ubuntu rất nhẹ nên sẽ không có giao diện (GUI). Chúng ta phải dùng Terminal để tương tác với nó.

Một số câu lệnh quan trọng:

|  |  |
| --- | --- |
| Command | Description |
| echo | Output any text that we provide |
| whoami | Find out what user we're currently logged in as! |
| ls | listing |
| cd | change directory |
| cat | output content of file |
| pwd | find out full path to current directory |
|  |  |

**Searching for files**

**Find command:**

Giả sử bạn biết tên file nhưng không nhớ nó nằm ở đâu.

find -name password.txt

./folder1/password.txt

Nếu bạn muốn tìm những file có đuôi .txt

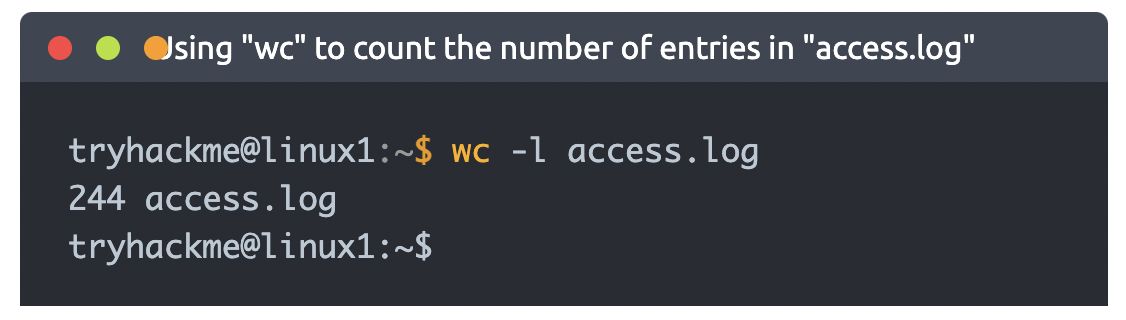
find -name \*.txt

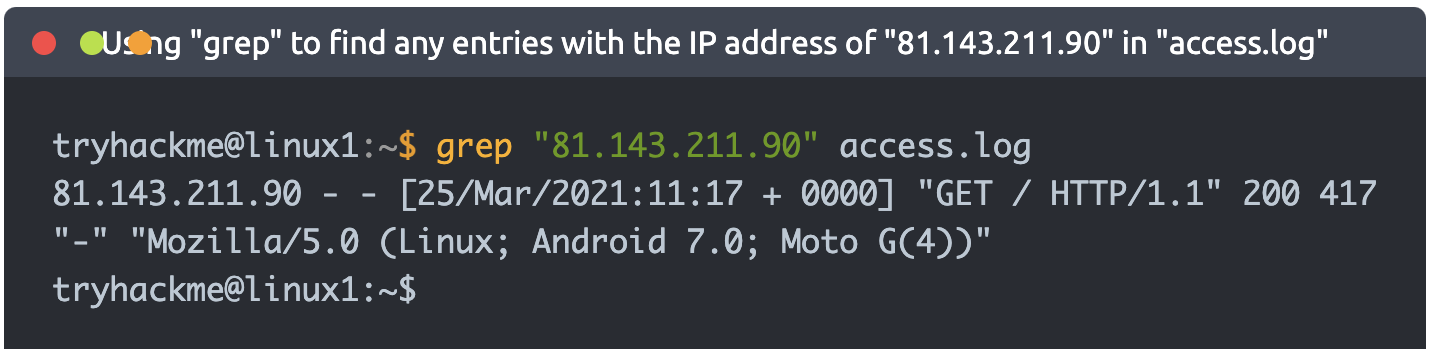
**Grep command:**

Câu lệnh grep cho phép bạn tìm kiếm nội dung xác định có trong file.

Ví dụ có 244 entries trong file access log của máy chủ web. Câu lệnh cat sẽ không hiệu quả. Nếu chúng ta muốn kiểm tra file access.log có địa chỉ IP xác định, xem tất cả entries sẽ không đem lại hiệu quả cao.

Chúng ta có thể dùng grep để tìm kiếm nội dung liên quan đến địa chỉ Ip xác định.





**Shell Operators**

**Toán tử &**

Cho phép chúng ta thực thi lệnh ở phía sau. Ví dụ, bạn muốn sao chép 1 tập tin lớn, nó sẽ mất nhiều thời gian và chúng ta không thể thực thi lệnh cho đến khi sao chép thành công.

Toán tử & sẽ thực thi lệnh và chạy ở background để chúng ta có thể thực thi 1 lệnh khác.

**Toán tử &&**

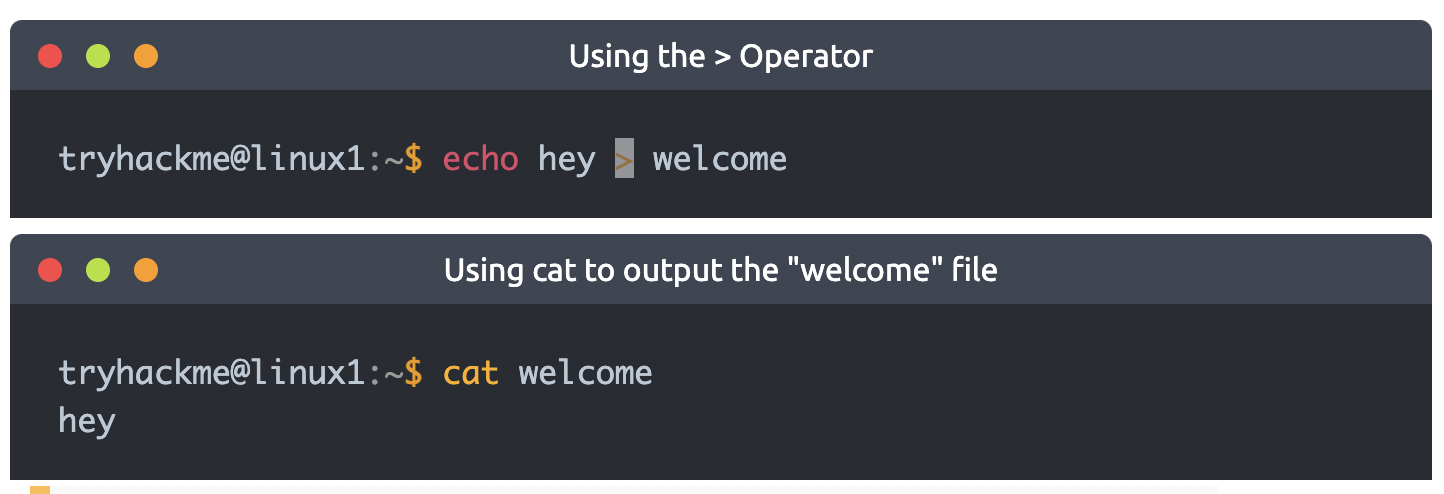
Giúp chạy nhiều câu lệnh cùng 1 lúc

**Toán tử >**

Ghi nội dung vào 1 file

Nếu file

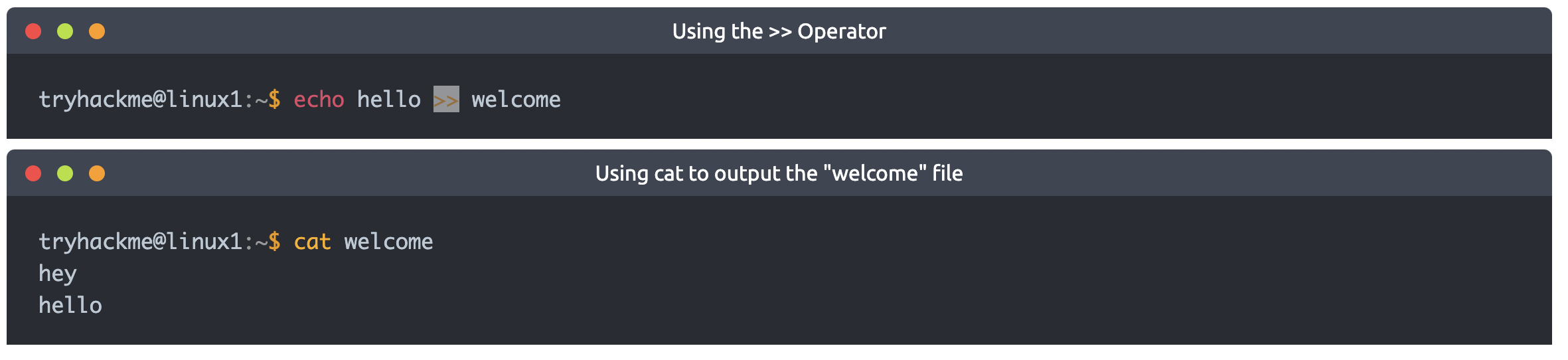
* Chưa có: Nó sẽ tạo 1 file mới
* Đã tồn tại: nội dung sẽ bị ghi đè



**Toán tử >>**

Ghi nội dung vào file.

Nếu file đã tồn tại, nó sẽ ghi tiếp vào (không ghi đè)

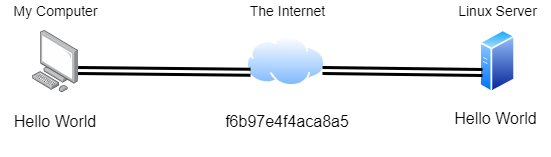


**Linux Fundamentals 2**

**SSH là gì và cách nó hoạt động?**

Secure Shell hay SSH chỉ đơn giản là 1 giao thức giữa các thiết bị trong dạng mã hóa.

Bằng mật mã, bất cứ đầu vào chúng ta gửi qua mạng đều được mã hóa sau đó giải mã khi chúng đến được máy đích.



Qua đó

* SSH cho phép chúng ta kết nối thiết bị từ xa
* Bất cứ dữ liệu giữa các thiết bị được mã hóa.

Cú pháp SSH:

ssh username@machine\_ip

**Flags and Switches**

Đa số các câu lệnh cho phép cung cấp đối số. Đối số được xác định bằng dấu gạch nối và 1 từ khóa xác định là flag hay switch.

Nếu không được chỉ định rõ, câu lệnh sẽ thực hiện những hành vi mặc định. Ví dụ ls liệt kê nội dung của thư mục đang hoạt động. Nhưng tập tin ẩn sẽ không hiện. Chúng ta cần dùng flag để mở rộng hành vi của nó.

Ví dụ, ls chỉ liệt kê nội dung của 1 thư mục đang hoạt động.

* ls -a (viết tắt là --all) sẽ liệt kê cả thư mục ẩn
* ls --help liệt kê các lựa chọn, cách dùng và mô tả

Câu lệnh man xuất ra 1 man page (viết tắt là manual) chứa tài liệu về câu lệnh linux và các ứng dụng.

**Filesystem Interaction**

|  |  |  |
| --- | --- | --- |
| **Command** | **Full Name** | **Purpose** |
| touch | touch | Create file |
| mkdir | make directory | Create a folder |
| cp | copy | Copy a file or folder |
| mv | move | Move a file or folder |
| rm | remove | Remove a file or folder |
| file | file | Determine the type of a file |

tạo file mới: touch note

tạo thư mục mới: mkdir mydirectory

xóa file: rm note

xóa thư mục: rm -R mydirectory

lệnh copy nhận 2 đối số:

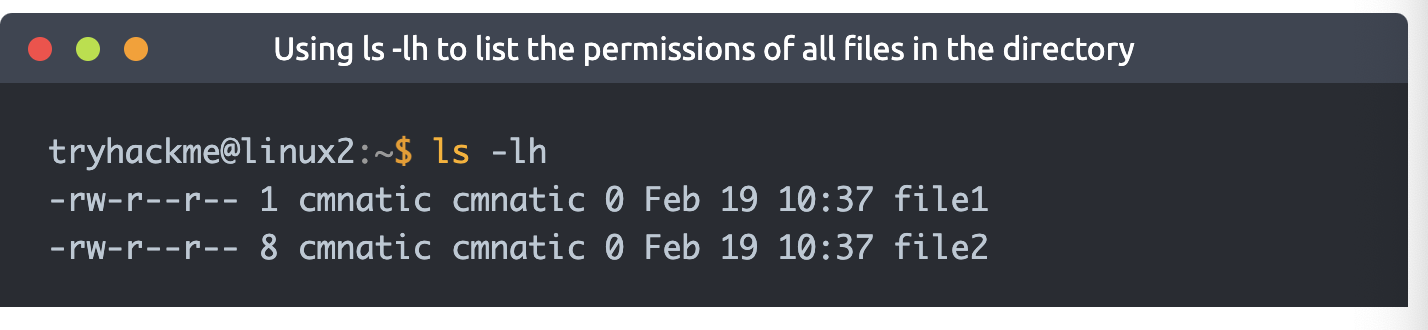
* tên của tập tin đã tồn tại
* tên của tệp mới khi sao chép

lệnh di chuyển cũng nhận 2 đối số như cp. Tuy nhiên, thay vì sao chép hay tạo 1 file mới, mv sẽ hợp nhất hay điều chỉnh tệp thứ hai. Chúng ta không chỉ dùng mv để di chuyển 1 tập tin đến 1 thư mục mới mà còn đổi được tên tập tin hay thư mục.

xác định loại tệp: file note

**Permissions 101**

Để xác định tập tin chúng ta được phép truy cập, ls -l



Mặc dù hơi đáng sợ nhưng 3 cột đầu tiên rất quan trọng để xác định các đặc điểm nhất định của tập tin hay thư mục.

Hành động nào được phép và người dùng hay nhóm có thể thực hiện hành động sau :

* Đọc
* Ghi
* Thực thi

file1 có dấu gạch nối theo sau là rw. Điều này có nghĩa là chỉ có chủ tập tin có thể đọc và ghi tệp này nhưng không thể thực thi nó.

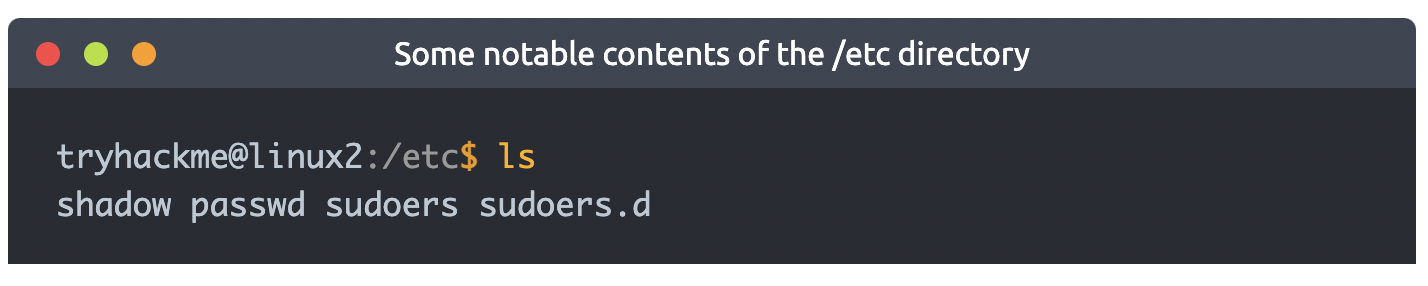
**Một số thư mục phổ biến**

**/etc**

Là 1 nơi phổ biến để lưu trữ các tệp hệ thống được dùng bởi hệ điều hành.

Ví dụ, file sudoers chứa danh sách các người dùng và nhóm có quyền chạy sudo như root

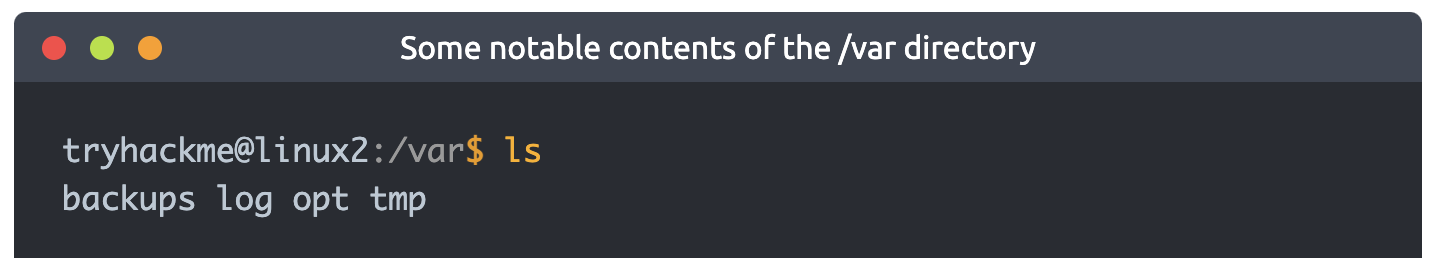
Những file như passwd và shadow đặc biệt dành cho Linux vì chúng cho biết cách hệ thống của bạn lưu trữ mật khẩu của mỗi người dùng bằng mật mã hóa như sha512.



**/var**

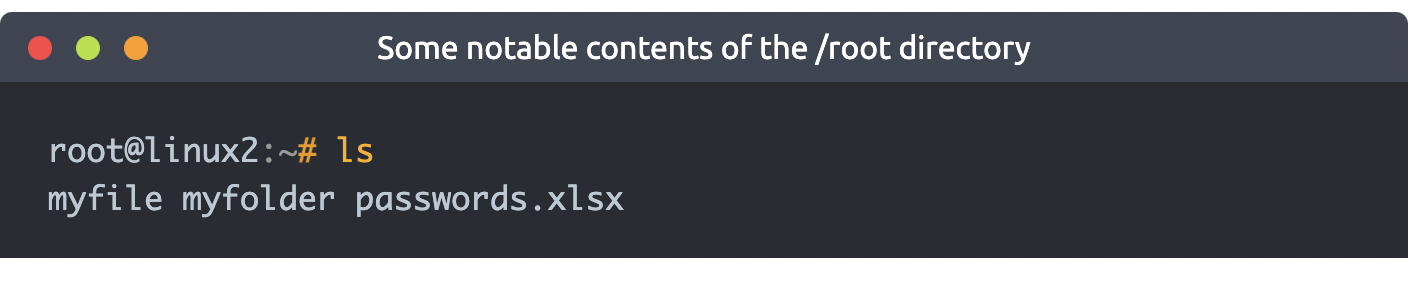
Var viết tắt là biến. Thư mục này lưu trữ dữ liệu mà thường xuyên được truy cập hay viết bởi các dịch vụ hay ứng dụng chạy trên hệ thống.

Ví dụ, những file log từ các dịch vụ và ứng dụng đang chạy được viết ở đây (/var/log), hay dữ liệu khác không nhất thiết phải được liên kết với người dùng cụ thể (cơ sở dữ liệu).

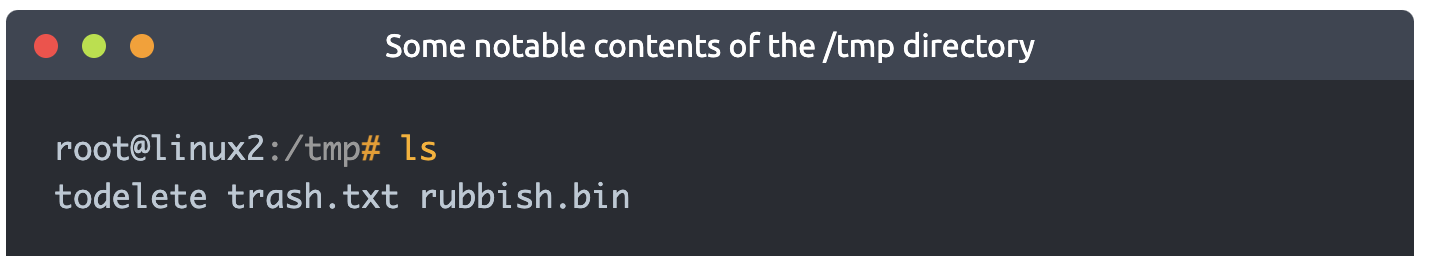


**/root**

Không như thư mục /home, thư mục root là home cho người dùng hệ thống “root”. Điều đáng nói là thư mục này sẽ có dữ liệu về root.



**/tmp**

Đây là thư mục root duy nhất trên Linux. Nó chỉ là tạm thời và lưu trữ dữ liệu chỉ sử dụng 1 đến 2 lần. Tương tự như bộ nhớ trên máy tính, khi máy tính được khởi động lại, nội dung của nó sẽ bị mất hết.

**Linux Fundamentals 3**

Text Editors: nano, vim

**Những tính năng hữu dụng:**

**Download Files:**

Một tính năng rất cơ bản của máy tính là có thể vận chuyển file. Ví dụ, bạn có thể tải 1 chương trình, 1 đoạn mã hay 1 bức tranh. Có rất nhiều cách để chúng ta truy suất tập tin:

wget command:cho phép chúng ta tải những tập tin từ web qua http giống như bạn đang truy cập file trên trình duyệt web. Chúng ta chỉ cần cung cấp địa chỉ của mã nguồn mong muốn tải xuống.

wget https://assets.tryhackme.com/additional/linuxfundamentals/part3/myfile.txt

**Vận chuyển tập tin từ máy chủ của bạn -SCP (SSH)**

Secure copy hay SCP là 1 phương tiện sao chép file một cách bảo mật. Không như lệnh cp, lệnh này cho phép bạn vận chuyển tập tin giữa hai máy tính bằng SSH để cung cấp xác thực và mã hóa.

Giả sử chúng ta biết tên người dùng và mật khẩu của 1 user trên hệ thống hiện tại và 1 user trên hệ thống từ xa. Ví dụ, sao chép 1 tập tin từ máy của bạn đến máy đích có thông tin sau:

|  |  |
| --- | --- |
| Variable | Value |
| The IP address of the remote system | 192.168.1.30 |
| User on the remote system | ubuntu |
| Name of the file on the local system | important.txt |
| Name that we wish to store the file as on the remote system | transferred.txt |

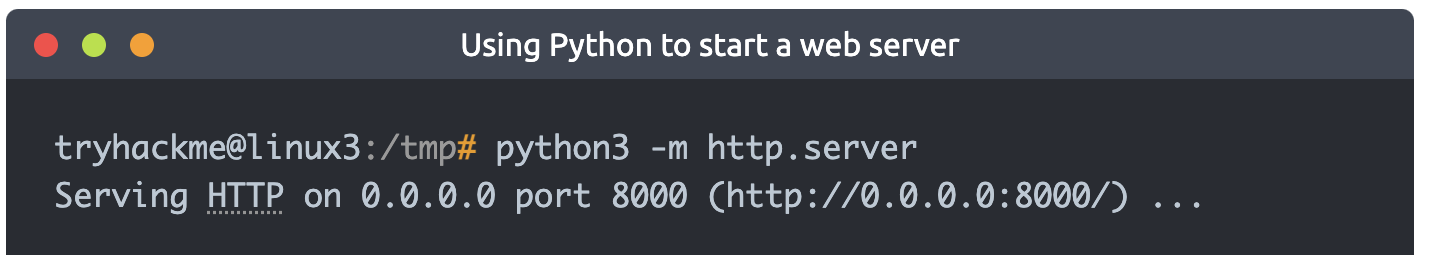
scp important.txt ubuntu@192.168.1.30:/home/ubuntu/transferred.txt

**Cung cấp tệp từ máy chủ của bạn -Web**

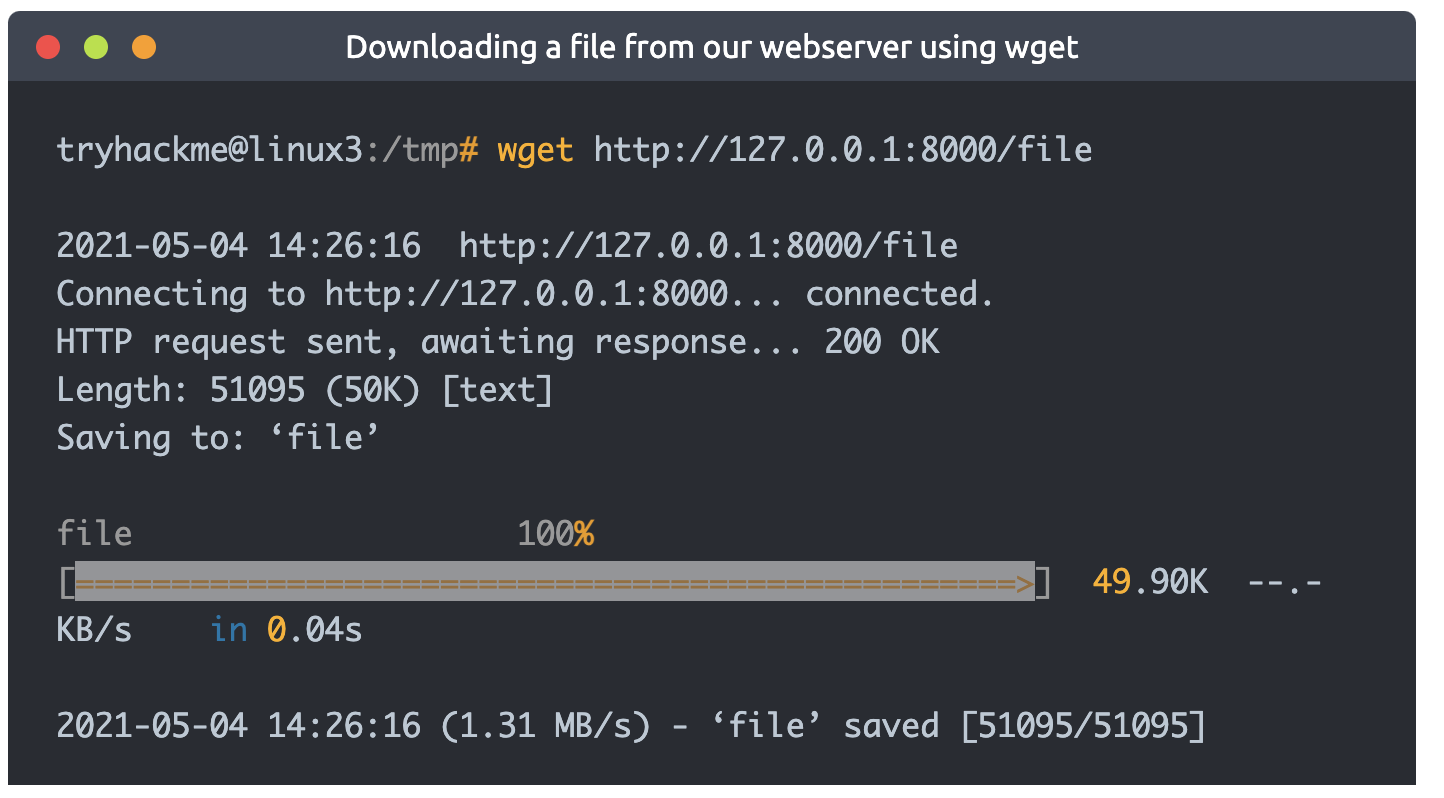
Các máy Ubuntu được đóng gói sẵn với Python3. Python cung cấp 1 module nhẹ và dễ sử dụng gọi là “HTTPServer”. Module này biến máy tính của bạn thành 1 máy chủ web nhanh và dễ mà bạn có thể dùng để cung cấp tệp của mình nơi chúng được tải xuống bởi một máy tính khác bằng lệnh wget hay curl.

HTTPServer của Python3 sẽ cung cấp tập tin trong thư mục mà bạn chạy lệnh. Tất cả chúng ta cần làm là chạy python3 -m  http.server

Chúng ta đang cung cấp 1 thư mục gọi là máy chủ web có 1 file.



Bây giờ dùng wget để tải tập tin bằng địa chỉ IP và tên của file.

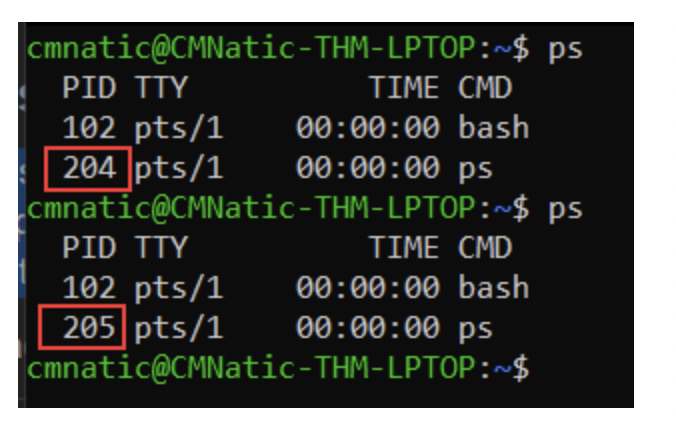


**Processes 101**

Processes là các chương trình đang chạy trên máy của bạn. Chúng được quản lý bởi nhân, nơi mà mỗi process sẽ có 1 ID riêng cũng như là PID. PID tăng dần theo thứ tự mà process bắt đầu. Ví dụ, process thứ 60 sẽ có PID là 60.

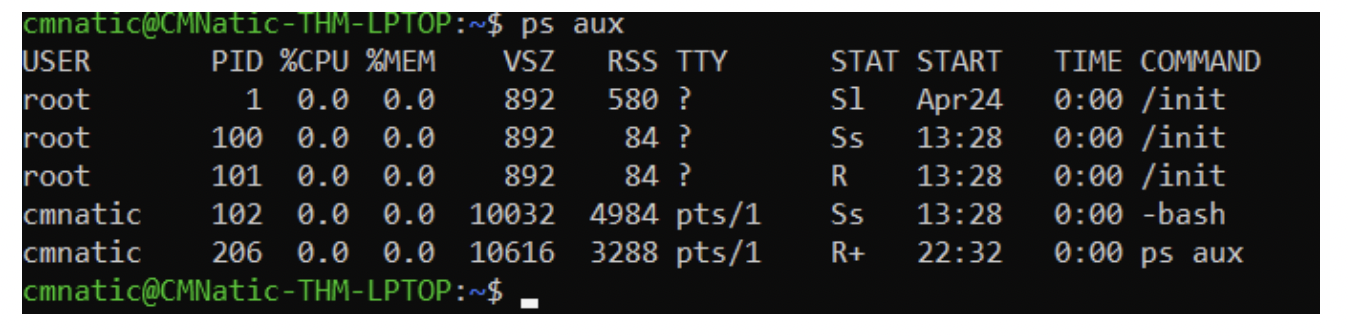
**Viewing Processes**

Chúng ta có thể dùng lệnh ps để cung cấp 1 danh sách các processes đang chạy dưới dạng phiên của người dùng và thêm một vài thông tin như status code, phiên đang chạy nó, thời gian sử dụng CPU và tên của chương trình hay lệnh đang thực thi.



Trong hình trên, process thứ hai là ps có pid là 204 và sau đó trong lệnh ở dưới, nó tăng lên 205.

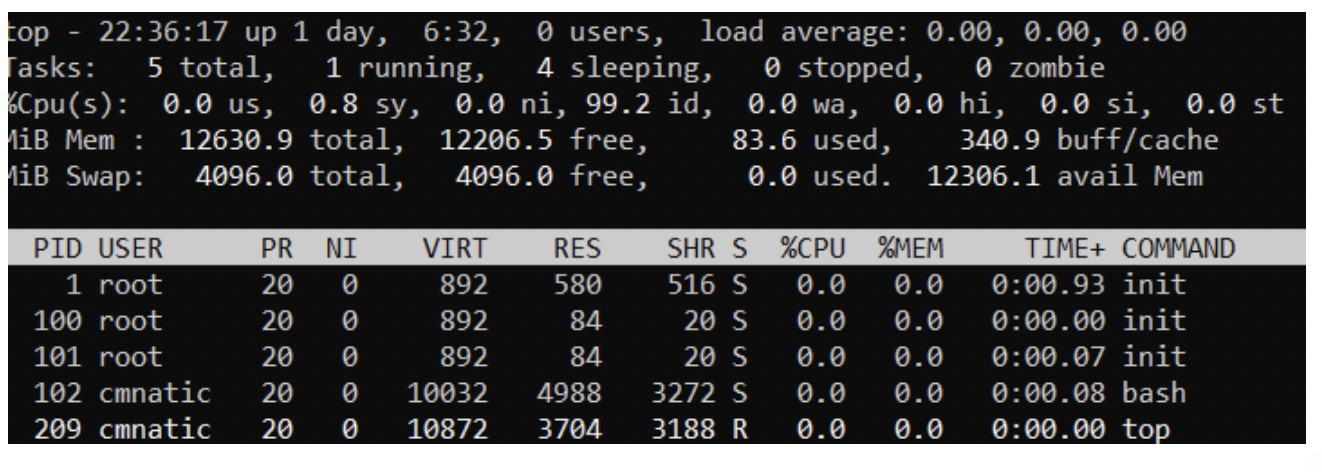
Để xem các processes chạy bởi người dùng khác và không chạy từ 1 phiên như các processes hệ thống, chúng ta cần cung cấp thêm aux như sau:



Lưu ý là chúng ta chỉ thấy được 5 processes.

Một lệnh hữu dụng khác là top; top cho bạn biết trạng thái trong thời gian thực của các processes đang chạy trên hệ thống thay vì chỉ xem 1 lần. Các trạng thái sẽ được làm lại mỗi 10 giây nhưng cũng tự làm lại khi bạn dùng arrow keys để duyệt nhiều dòng khác nhau.

Một lệnh rất tốt để nhìn vào trong hệ thống của bạn.



**Quản lý Processes**

Bạn có thể gửi các tín hiệu để hủy các processes. Để kill 1 lệnh, ta có thể dùng lệnh kill với pid tương ứng của nó. Ví dụ để kill pid 1337, chúng ta dùng kill 1337.

Dưới đây là 1 vài tín hiệu chúng ta có thể gửi đến 1 process khi nó bị kill:

* SIGTERM: kill process nhưng cho phép nó làm một vài việc dọn dẹp trước.
* SIGKILL: kill process không làm bất cứ dọn dẹp gì
* SIGSTOP: dừng/ đình chỉ 1 process

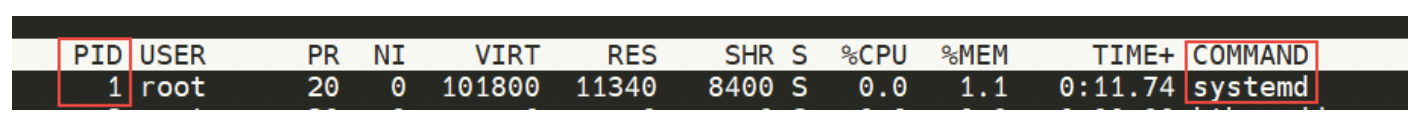
**Các processes bắt đầu như thế nào?**

Trước tiên hãy nói về namespaces. Hệ điều hành dùng namespaces để phân chia các mã nguồn hợp lí trên máy tính (như CPU, RAM và độ ưu tiên) cho các quá trình. Các processes sẽ được phép truy cập 1 lượng tài nguyên nhất định của máy tính.

Namespaces rất tốt để bảo mật vì đây là cách cô lập các processes với nhau - chỉ những cái trong cùng 1 namespace mới có thể nhìn thấy nhau.

Trước đó, chúng ta đã nói về cách pid hoạt động và đây sẽ là tác dụng của nó. Process có id là 0 là 1 process bắt đầu khi khởi động hệ thống. Quá trình này là sự khởi động hệ thống trên Ubuntu như systemd, được dùng để quản lý các processes của 1 người dùng và ngồi giữa hệ điều hành và người dùng.

Ví dụ, khi 1 hệ thống khởi động, systemd là 1 trong những process đầu tiên được bắt đầu. Bất cứ chương trình hay phần mềm chúng ta muốn bắt đầu sẽ bắt đầu như 1 quá trình con của systemd. Điều đó đồng nghĩa với việc nó được kiểm soát bởi systemd nhưng sẽ chạy 1 quá trình riêng của nó (mặc dù chia sẻ các mã nguồn từ systemd) để giúp chúng ta dễ xác định và thích hơn.



**Khởi động các dịch vụ, process**

Một vài ứng dụng có thể được khởi động trên hệ thống. Ví dụ, các máy chủ web, cơ sở dữ liệu hay vận chuyển tệp. Phần mềm này thường rất quan trọng và thường được yêu cầu bắt đầu trong quá trình khởi động hệ thống bởi các quản trị viên.

Trong ví dụ này, chúng ta sẽ nói về máy chủ web apache để bắt đầu apache một cách thủ công và sau đó yêu cầu hệ thống khởi động apache2.

Công dụng của câu lệnh systemctl: cho phép chúng ta tương tác với quá trình systemd. Định dạng systemctl: systemctl [option] [service]

Ví dụ, để yêu cầu apache khởi động, chúng ta dùng systemctl start apache2

. Nếu chúng ta muốn dừng apache, thay thế option là stop.

Có 4 lựa chọn với systemctl:

* Start
* Stop
* Enable
* Disable

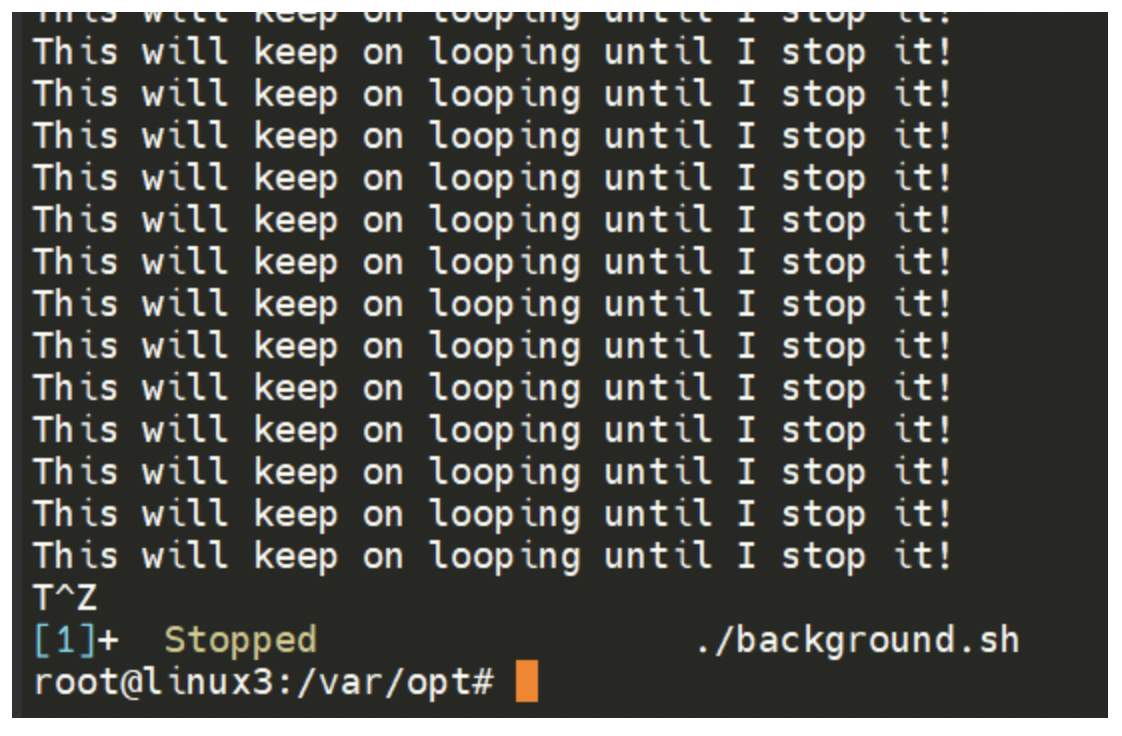
**Giới thiệu về Backgrounding và Foregrounding trong Linux**

Các quá trình có thể chạy ở hai trạng thái: background và foreground.

Những lệnh như echo sẽ chạy ở foreground, sao chép tệp ở background.

Khi 1 lệnh chạy background, chúng ta có thể chạy thêm nhiều lệnh khác.

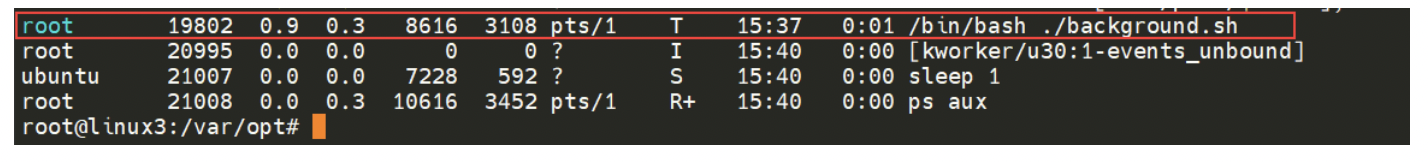
Chúng ta có thể làm điều tương tự khi thực thi tập lệnh. Dùng Ctrl+Z để background 1 process. Nó cũng rất hiệu quả khi muốn dừng thực thi 1 tập lệnh nào đó.



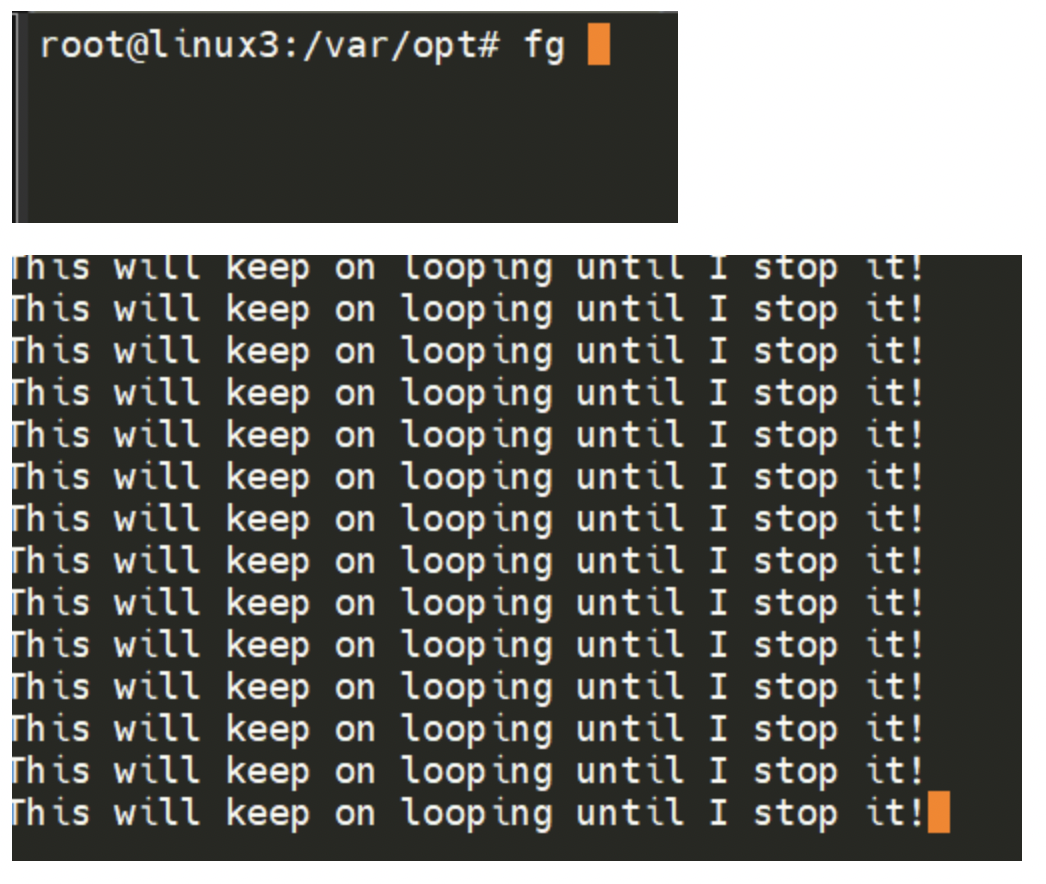
Bằng việc Ctrl+Z, terminal không còn lắp đầy các tin nữa cho đến khi chúng ta foreground nó.

**Foreground 1 process**

Bây h chúng ta có 1 quá trình chạy ở background, ví dụ background.sh có thể được kiểm tra bằng lệnh ps aux, chúng ta có thể mang process này ra foreground để tương tác.



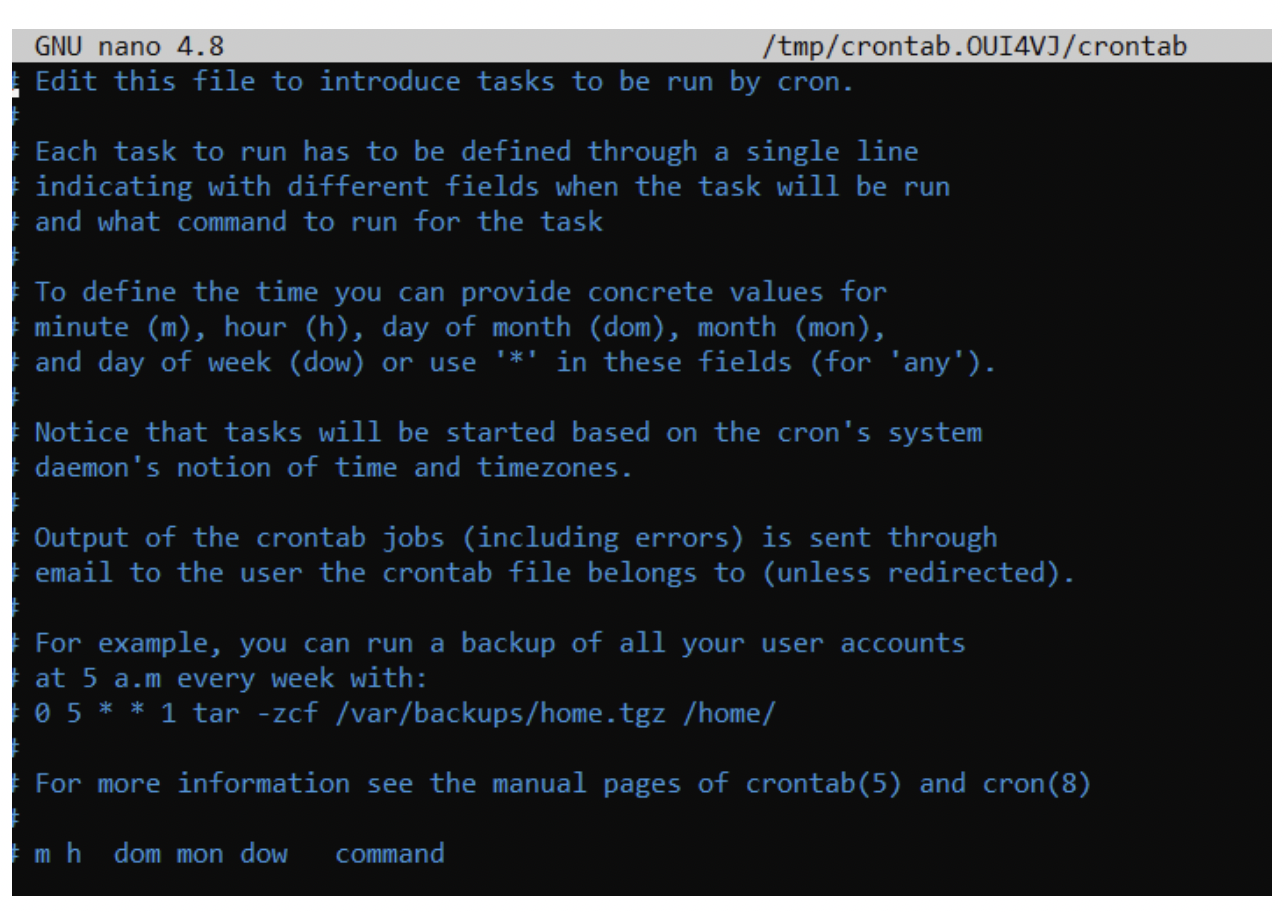
Ta có thể background 1 quá trình bằng Ctrl+Z hay toán tử &, chúng ta có thể dùng fg để mang nó lại foreground.



**Bảo trì hệ thống của bạn: tự động**

Người dùng muốn đặt trước các thao tác sau khi hệ thống được khởi động. Ví dụ, chạy lệnh, sao lưu file hay thực thi các chương trình yêu thích như Spotify hay Chrome.

Chúng ta sẽ nói về quá trình cron, hay chi tiết hơn, chúng ta sẽ tương tác với nó qua crontabs. Crontab là 1 trong những processes được bắt đầu trong lúc khởi động, chịu trách nhiệm quản lí các cron jobs.



Crontab là 1 file đặc biệt với định dạng được xác định bởi quá trình cron để thực thi lệnh từng bước một. Crontabs yêu cầu 6 giá trị sau:

|  |  |
| --- | --- |
| Value | Description |
| MIN | What minute to execute at |
| HOUR | What hour to execute at |
| DOM | What day of the month to execute at |
| MON | What month of the year to execute at |
| DOW | What day of the week to execute at |
| CMD | The actual command that will be executed. |

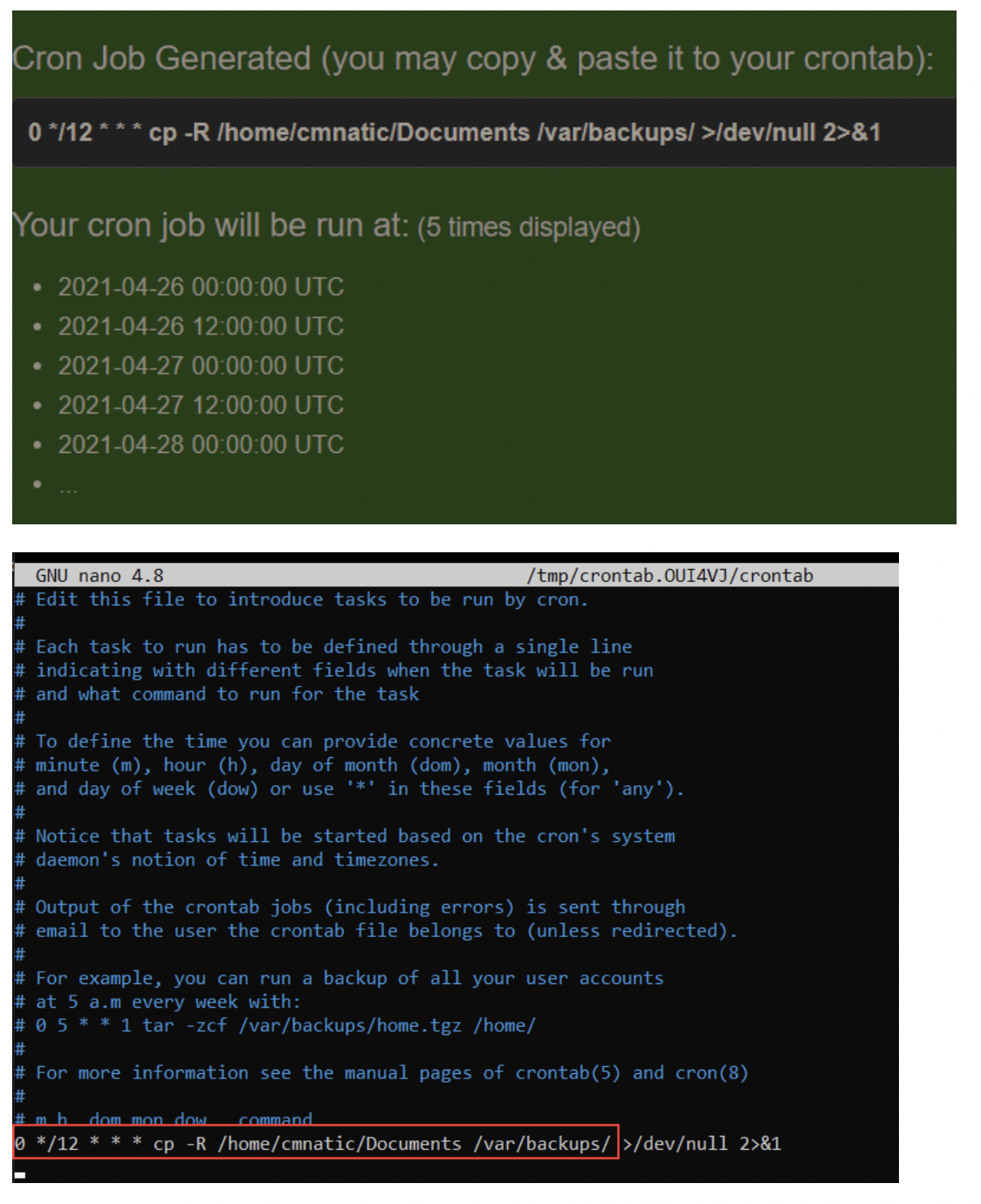
Cùng xem ví dụ của các tệp sao lưu. Bạn có thể mong muốn sao lưu các tài liệu của người dùng cmnatic mỗi 12 giờ. Chúng ta có thể dùng định dạng sau:

0 \*12 \* \* \* cp -R /home/cmnatic/Documents /var/backups/

Một đặc điểm thú vị của crontabs là hỗ trợ wildcard hay asterisk (\*). Nếu chúng ta không muốn cấp giá trị cho 1 miền cụ thể, ví dụ, chúng ta không quan tâm đến tháng, ngày hay năm nó được thực thi - chỉ thực thi sau mỗi 12 h, chúng ta chỉ đơn giản đặt \*.

Mới bắt đầu sẽ cảm thấy bối rối, đó là tại sao các mã nguồn tuyệt vời như Crontab Generator sinh ra, cho phép bạn dùng 1 ứng dụng thân thiện để tạo ra dịnh dạng của riêng bạn. Ví dụ, trang “Cron Guru”.

Các Crontabs có thể được chỉnh sửa bằng lệnh crontab -e , bạn có thể chọn 1 editor như nano để chỉnh sửa cron tab.

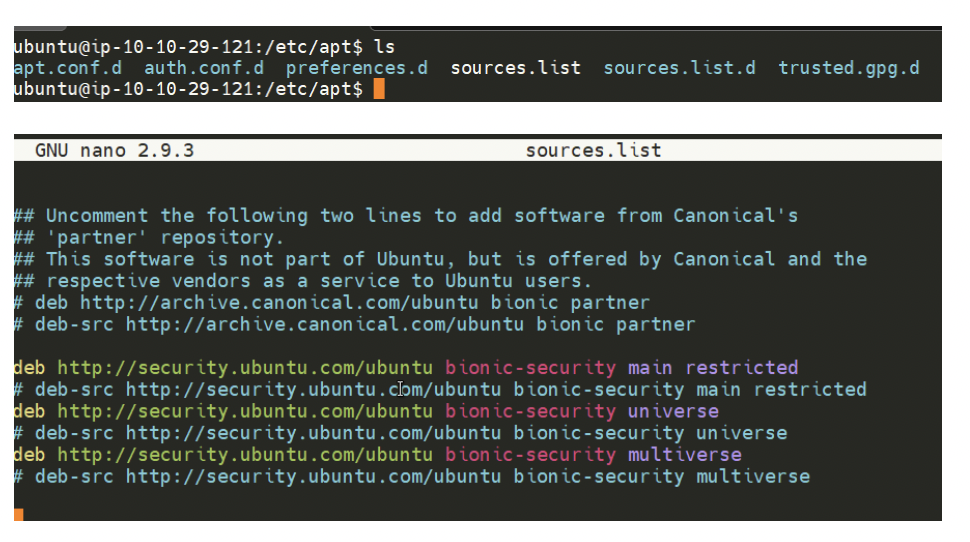


**Bảo trì hệ thống của bạn: Quản lí gói**

**Giới thiệu về kho chứa các gói và phần mềm**

Khi nhà phát triển muốn gửi phần mềm cho cộng đồng, họ sẽ gửi nó đến kho chứa apt. Nếu được chấp thuận, các chương trình và công cụ của họ sẽ được tung ra thị trường. Hai trong số tính năng ấn tượng nhất của Linux: User accessbility và các công cụ mã nguồn mở.

Khi dùng lệnh ls trên Ubuntu 20.04 Linux, các lệnh này đóng vai trò là gateway/registry.



Trong khi các nhà cung cấp hệ điều hành sẽ duy trì các kho chứa của họ, bạn cũng có thể thêm các kho chứa cộng đồng trong danh sách của bạn. Điều này cho phép bạn mở rộng các tính năng của hệ điều hành. Các kho chứa bổ sung có thể được thêm vào bằng lệnh add-apt-repository hay liệt kê nhà cung cấp khác. Ví dụ, một vài nhà cung cấp sẽ có kho chứa gần vị trí địa lý của họ hơn.

**Quản lý kho chứa của bạn (thêm hoặc xóa)**

Bình thường chúng ta dùng lệnh apt để cài đặt phần mềm trong hệ thống Ubuntu. Lệnh apt là 1 phần của phần mềm quản lý gói cũng được gọi là apt. Apt chứa cả 1 bộ công cụ cho phép chúng ta quản lý các gói và nguồn của các phần mềm, cài đặt hay xóa chúng.

Một phương pháp để thêm các kho chứa là dùng lệnh add-apt-repository nhưng chúng ta sẽ thêm và xóa 1 kho chứa một cách thủ công. Trong khi bạn có thể cài đặt phần mềm qua trình cài đặt gói như dpkg, các lợi ích của apt là khi chúng ta cập nhật hệ thống - các kho chứa phần mềm cũng sẽ kiểm tra để cập nhật.

Trong ví dụ này, chúng ta sẽ thêm text editor như Sublime Text đến máy Ubuntu như 1 kho chứa vì nó không phải là 1 phần của kho chứa mặc định của Ubuntu. Khi thêm phần mềm, tính toàn vẹn của gói được đảm bảo bằng GPG keys. Những keys này về cơ bản là 1 kiểm tra an toàn. Nếu keys không phù hợp với cái mà hệ thống tin tưởng và cái mà nhà phát triển dùng, sau đó phần mềm sẽ không được tải.

Vì vậy, để bắt đầu chúng ta cần thêm GPG key cho các nhà phát triển của Sublime Text 3.

1. Tải GPG key và dùng apt-key để tin nó:

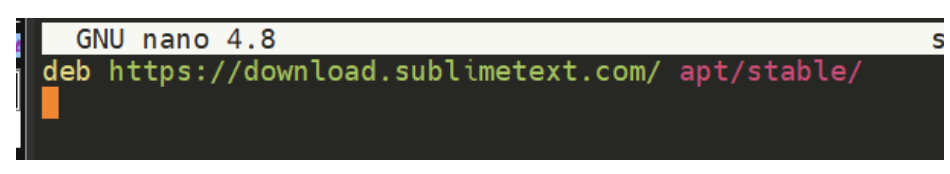
wget -qO - https://download.sublimetext.com/sublimehq-pub.gpg | sudo apt-key add -

2. Bây giờ chúng ta đã thêm key này vào danh sách tin cậy, chúng ta có thể thêm kho chứa của Sublime Text 3 đến danh sách các mã nguồn apt. Cách tốt nhất là có 1 tệp riêng cho mỗi kho chứa của bên thứ 3.

2.1 Tạo 1 tệp tên sublime-text.list trong /etc/apt/sources.list.d và nhập thông tin kho chứa như sau:



2.2 Và dùng Nano để thêm và lưu kho chứa Sublime Text 3 trong tệp mới tạo



2.3 Sau khi chúng ta đã thêm điểm vào này, chúng ta cần update apt để nhận ra điểm vào mới này bằng lệnh apt update

2.4 Khi update thành công, chúng ta có thể tiếp tục cài đặt phần mềm mà chúng ta tin tưởng và thêm vào apt bằng lệnh apt install sublime-text

Loại bỏ các gói dễ dàng như đảo ngược. Quá trình này được làm như sau:

add-apt-repository --remove ppa:PPA\_Name/ppa hay xóa tệp mà chúng ta đã thêm trước đó

Tiếp theo, xóa phần mềm:

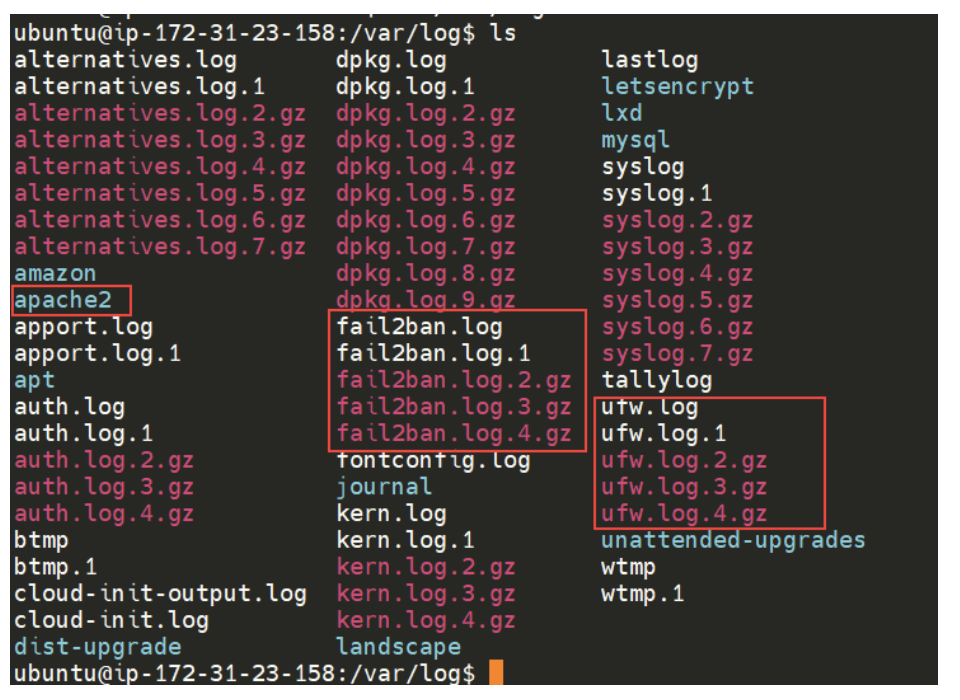
apt remove [software-name-here] i.e. apt remove sublime-text

**Bảo trì hệ thống của bạn: Logs**

Đặt ở thư mục /var/log, những tệp và thư mục chứa thông tin log cho các ứng dụng và dịch vụ chạy trên hệ thống của bạn. Hệ điều hành rất tốt khi tự động quản lí các logs này trong 1 process được biết đến như “rotating”.

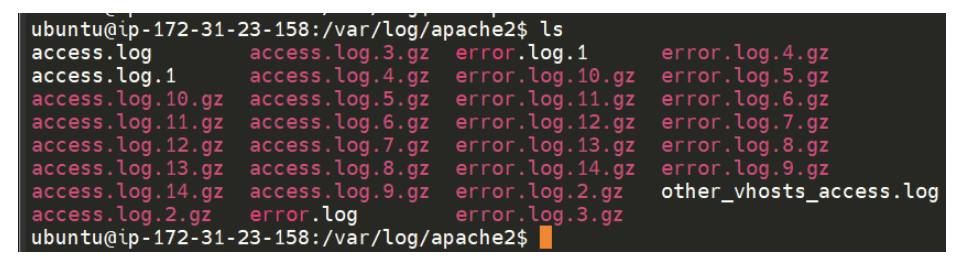
Tôi đã tô đậm một vài logs từ 3 dịch vụ đang chạy trên hệ thống Ubuntu:

* Máy chủ web apache
* Logs cho dịch vụ fail2ban, được dùng để theo dõi những nỗ lực brute force.
* Dịch vụ UFW được dùng như tường lửa



Những dịch vụ và logs là 1 cách rất tốt để theo dõi tình trạng của hệ thống và bảo vệ nó. Không chỉ có vậy, các logs cho máy chủ web chứa thông tin về mỗi yêu cầu - cho phép các nhà phát triển hay quản trị viên phân tích hiệu suất và điều tra các hoạt động của kẻ xâm nhập. Ví dụ, hai loại tệp logs dưới đây rất thú vị:

* access log
* error log



Và tất nhiên, logs lưu trữ thông tin về cách hệ điều hành chạy và các hoạt động được thực hiện bởi người dùng như các nỗ lực xác thực.