### Các phím tắt:

Crtl +A : Di chuyển con trỏ đến dầu dòng

Crtl +E : Di chuyển con trỏ đến cuối dòng

Crtl +U : Xóa từ vị trí con trỏ đến dầu dòng

Crtl +K : Xóa từ vị trí con trỏ đến cuối dòng

Alt + B/F: Di chuyển con trỏ sang trái/phải một từ

Mở tab mới: Ctrl+Shift+T

Đóng tab hiện tại: Ctrl+Shift+W

Ls –R: hiển thị tập con bên trong

# Thư mục:

#mkdir: tạo mới một thư mục

Flag:

–p: parent tạo thư mục con cả khi chưa có thư mục

**#rmdir:** xóa thư mục rỗng

Flag:

–p: Xóa cả thư mục cha

–f : Xóa hết tất cả kể cả thư mục không rỗng

#**rm**: xóa tập tin hoăc thư mục

Flag:

–i: Đưa ra thông báo trước khi xóa

-r hoặc --recursive: Xóa tất cả các tệp và thư mục con bên trong thư mục được chỉ định.

-f hoặc --force: Bỏ qua các thông báo xác nhận và xóa các tệp được bảo vệ khỏi ghi mà không cần hỏi.

**Cây thư mục: tree**

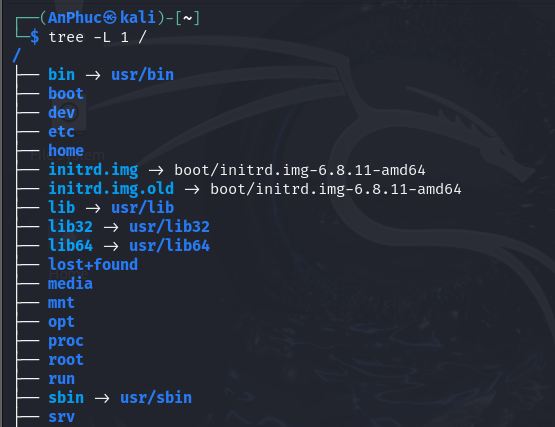
#tree: hiển thị cây thư mục

Option tree [directory][]

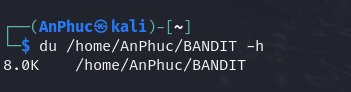
Flag:

–L: lấy từ mấy thư mục con

Ex: tree –L 1 / => lấy môt thư mục con từ root



du: (disk usage) Hiển thị thông tin về dung lượng sử dụng trên đĩa



# TẬP TIN

#Touch: tạo tập tin rỗng

#echo: tạo tập tin với nội dung sử dụng echo hoặc thêm dòng cho nội dung có sẵn

Option: echo “asdsad” >> text.txt

#Cat: sử dụng để xem nội dung bên trong tập tin

#cp: Sao chép tập tin

Flag:

–R, -r: Sao chép toàn bộ thư mục

#mv: Lệnh di chuyển tập tin hoặc đổi tên của tập tin

Flag:

–i: Nhắc trước khi di chuyển với tập tin/ thư mục đích đã tồn tại

–f: Ghi đè khi di chuyển tới tập tin/ thư mục đã tồn tại

Các lệnh quản lý file và thư mục

Kí tự:

?: sử dụng để đại diện cho một ký tự bất kỳ trong một mẫu tìm kiếm

\*: sử dụng để đại diện cho bất kỳ chuỗi kí tự nào (gồm cả chuỗi rỗng)

More and Less

#More: hiển thị tệp tin theo từng trang

#Less: hiển thị tệp tin một cách tương tác

Head and tail

#Head: xem 10 dòng đầu tiên của 1 tệp

#Tail: xem 10 dòng cuối cùng của 1 tệp

Flag:

–n: chọn số dòng muốn hiển thị

–f: theo dõi và hiển thị những công việc tiếp theo sẽ thực thi

#wc: in ra số dòng, số từ và số chữ trong một tệp

Flag:

–w: word

–l: line

–c: character

Định hướng nhập, xuất, cơ chế đường ống

**> < >> |**

**Kí hiệu:**

**>: xuất dữ liệu ra file, xóa dữ liệu file nếu file cũ đã tồn tại**

****

**>>: xuất dữ liệu và ghi thêm dữ liệu vào cuối file**

**<: Nhập dữ liệu từ file**

**| (dấu pipe) : được sử dụng để kết nối đầu ra của một lệnh với dầu vào của một lệnh khác, dữ liệu trước của đầu ra được truyền sang dữ liệu sau và thực hiện câu lệnh đó**

# TÌM KIẾM

**#grep: tìm kiếm văn bản dòng lệnh, chuỗi kí tự trong file hoặc thư mục**

**Flag:**

**–c: đếm số lần xuất hiện của string (count)**

**–I: bỏ qua phân biệt hoa thường**

**–v: Loc kết quả không khớp**

**–n: hiển thị số dòng của từ tìm kiếm trong file**

**–E: sử dụng biểu thức chính quy để tìm kiếm**

**–o: chỉ trùng với cấu trúc**

**#find: tìm kiếm tập tin hoặc thư mục**

**Có thể tìm theo tên, thời gian, nhóm , userID, …**

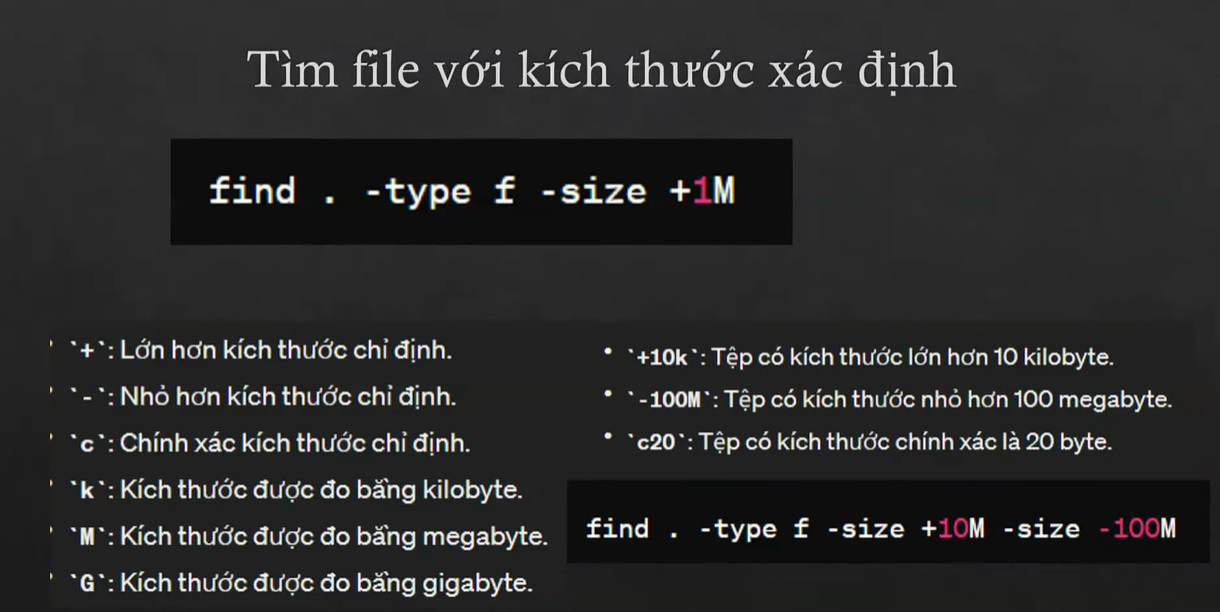
**Flag:**

**–name: tên**

**–atime, -mtime,-ctime: dựa trên thời gian truy cập, sửa đổi tg, hoặc tạo**

**–perm: quyền truy cập**

**–size: kích thước**

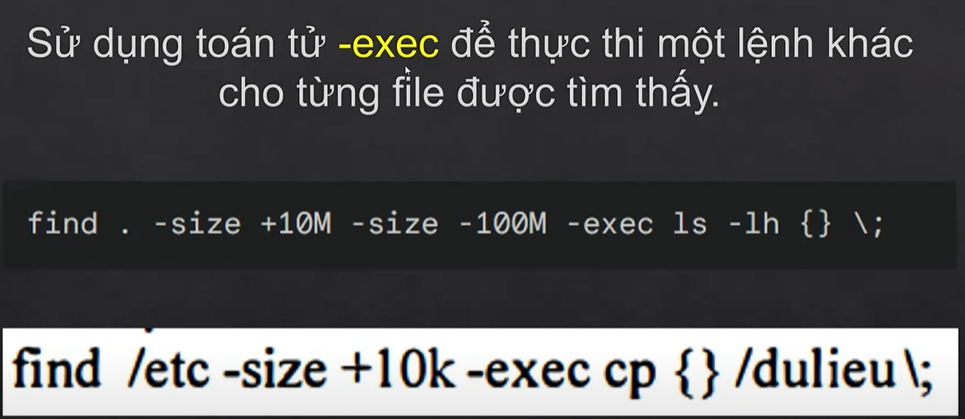
****

**–type f: tìm file**

**–gid, -uid: nhóm, người dùng**

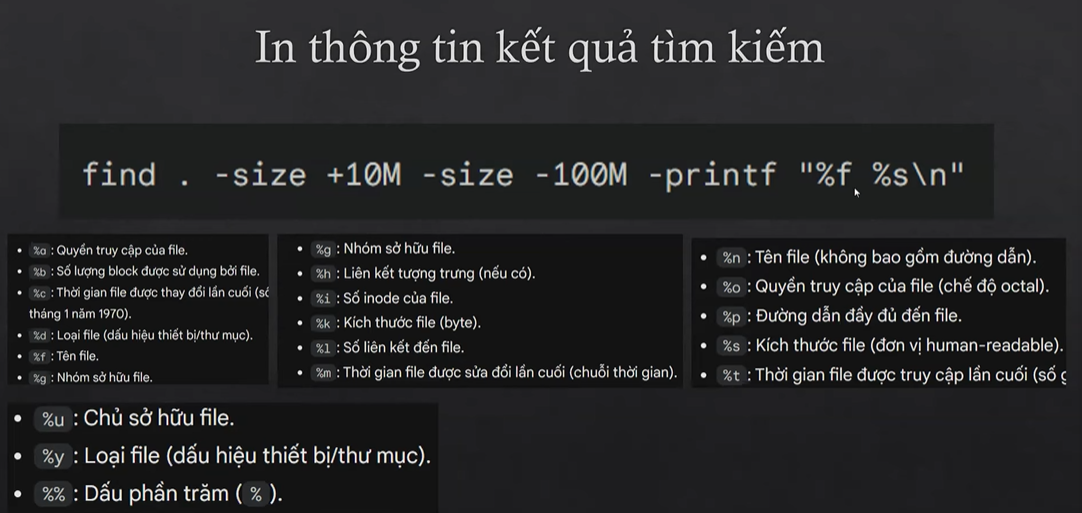
**–maxdepth: tìm kiếm theo chiều sâu của thư mục con**

**–exec: (thực thi) sử dụng để thực thi một lệnh khác**

****

**{} => đại diện cho tập hợp toàn bộ lệnh ở phía trc**

**\; => dấu kết thúc cho lệnh**

****

**sudo find . -type f -printf "%f %g %u %m %M %p %y\n" | awk '{printf "%-20s %-10s %-10s %-5s %-5s %-5s %-5s\n", $1, $2, $3, $4, $5, $6, $7}'**

**#whereis: tìm kiếm và hiển thị ra các tệp thực thi, tệp nguồn , và tệp hướng dẫn của lệnh đó**

**Flag:**

**–b: tìm file thực thi**

**–m:tìm file mã nguồn**

**–s: tìm trang man**

**–u: tìm trong thư mục do người dùng cài đặt**

#Which: tìm kiếm và hiển thị ra tệp thực thi trong đường dẫn hệ thống => chỉ hiển thị tệp đầu tiên được tìm thấy

- Which bash, which ls

#Whatis : trả về một mô tả ngắn giới hiệu về cách sử dụng của lệnh

-Whatis ls, whatis cat, …

# NÉN VÀ GIẢI NÉN TẬP TIN

## Các câu câu lệnh giải nén

### Gzip và Gunzip

Cú pháp câu lệnh: gzip [option][file] và gunzip

* Flag
  + –c, --stdout: In ra kết quả giải nén trên màng hình
  + –d : giải nén tập tin được nén hoặc sử dụng gunzip.
  + –f, --force: Bắt buộc nén tệp tin mà không hỏi lại người dùng
  + –r: nén tất cả các tệp tin trong thư mục và các thư mục con.

### Tar

Cú pháp câu lệnh: tar option [file..]

* Flag:
  + –c, --create: tạo một tệp tin nén mới.
  + –x, --extract: giải nén một tệp tin nén đã tồn tại.
  + –f, --file: đặt tên cho tệp tin nén
  + –v: hiển thị thông tin chi tiết về quá trình làm việc
  + –z, --gzip: sử dụng nén gzip khi tạo hoặc giải nén
  + –j, --bzip2: sử dụng nén bzip2 khi tọa hoặc nén
  + –r: Thêm các tệp tin vào tệp tin nén đã tồn tại
  + –t, --list: liệt kê nội dung của tệp tin nén mà không giải nén

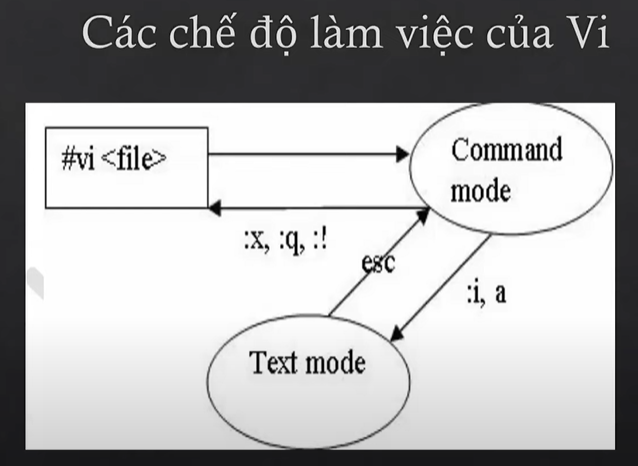
# TRÌNH SOẠN THẢO

## VI

* Không có giao diện, hướng màng hình
* Cho phép người dùng sử dụng bàn phím để chỉnh sửa, thay đổi

#vi [file name]

* Chế độ làm việc ủa vi



* Giải thích: Khi mở file vi thì mới đầu vào sẽ là chế độ sử dụng dòng lệnh, để chuyển sang chế độ văn bản ta sử dụng :i, a và ngược lại chuyển từ text mode sang command mode ta nhấn esc
* Các chế độ làm việc của vi
  + Command mode: sử dụng các lệnh để làm việc
  + Text mode: chỉnh sửa văn bản
  + Để thoát khỏi vi ta phải đang ở chế độ command mode và thực hiện các lệnh thoát
    - Thoát và không lưu: “:q!”
    - Thoát và lưu: “:x”, “:wq”
* Tất tần tật các lệnh trong vi



* Những câu lệnh tối thiểu cần phải nhớ
  + :set nu : hiển thị số dòng của file
  + :set nonu: không hiển thị số dòng
  + dd: xóa dòng ngay vị trí con trỏ
  + dG: Xóa từ vị trí con trỏ đến cuối file (G: tổ hợp shift + g)
  + :n : di chuyển con trỏ đến dòng số n
  + u: undo
  + /text” : tìm kiếm từ text trong file, dùng phím n để xem kết quả kế tiếp
* Nhóm lệnh khi chèn đoạn văn bản
  + i: trước dấu con trỏ
  + l: sau kí tự đầu tiên trên dòng
  + a: sau dấu con trỏ
  + A: sau kí tự cuối cùng trên dòng
  + o: dưới dòng hiện tại
  + O: trên dòng hiện tại
  + r: thay thế 1 kí tự hiện hành
  + R: thay thế cho đến khi nhấn
* Nhóm lệnh di chuyển với con trỏ
  + v: bôi đen từ vị trí con trỏ để sử dụng các câu lệnh
  + V: bôi đen từ vị trí đầu đến cuối câu ở vị trí có con trỏ
  + b : sang trái 1 space với kí tự đầu
  + e: sang trái 1 space với kí tự cuối
  + w: sang trái 1 space với kí tự đầu
  + h/j/k/l: Move cursor left/down/up/right
  + H/M/L: Di chuyển con trỏ lên đầu/ giữa/ cuối trang
  + ( / ) : đầu câu và cuối câu
  + { / }: đầu đoạn, cuối đoạn văn
  + 0, Home: di chuyển đến đầu hàng
  + $, End: di chuyển đến cuối hàng
  + G: di chuyển đến cuối dòng của file
* **Nhóm lệnh xóa**
  + **ndw (n là số từ): xóa n từ**
  + **dd: xóa một dòng**
  + **x: xóa 1 kí tự**
* **Nhóm lệnh tìm kiếm** 
  + **?, n: tìm trở lên**
  + **/ , N: tìm trở xuống**
  + **\*/An: tìm từ kế tiếp của An**
  + **\*?An: tìm từ kết thúc là An**
* **Tìm kiếm và thay thế** 
  + **:s/text1/text2/g: thay thế text1 = text2 ngay vị trí con trỏ chuột = > thêm % trước : để thay thế toàn bộ text**
* **Copy, past, undo** 
  + **y: copy**
  + **p: paste**
  + **y$: copy từ vị trí hiện tại đến vị trí cuối cùng**
  + **nyy: copy n dòng liên tiếp**
  + **u: Undo lại thao tác trước đó**

# lệnh cut cắt và trích xuất

CUT

* Cắt từ trong một tập tin
* Flag:
  + –b: cắt theo byte
  + –c: cắt theo số phần tử từ n-m
  + –f: filed cắt theo trường
  + –d: xác định kí tự ngăn cách giữa cái trường dữ liệu

reconnaissane web tools

# BIỂU THỨC CHÍNH QUY

**Giúp tìm kiếm, thay thế theo một cấu trúc mẫu văn bản nào đó.**

sudo : super user do

top: -> sử dụng để xem các tiến trình đang hoạt động, CPU, vv

PID: Tiến trình -> nhấn s để thay đổi thời gian làm mới -> sử dụng l để lọc ra các tiến trình đang chạy, các ct không hoạt động sẽ k hiển thị

-> k: kill sử dụng để tắt một tiến trình sử dụng PID

pidof nameprocess: -> tìm ra được PID của một tiến trình

kill -> tắt một tiến trình -KILL -9: những flag được sử dụng để đóng tiền trình một cách ép buộc và mạnh hơn so với lệnh kill thông thường

ps -ux: liệt kê tất cả các tiến trình đang chạy -a tất cả người dùng

echo

# phân quyền hệ thống file và thư mục

## Các kí hiệu về phân quyền

Ex: -rw-rw-r-- 1 type of file - onwer - group - orther user

dòng đầu tiên nều là

- : là một tệp bình thường

d: thì là thư mục

c : là kí tự đặc biệt

b : là một tệp nhị phân

r: quyền đọc

w: quyền viết

x: quyền thực thi

## Câu lệnh chmod

sử dụng chmod để thay đổi quyền

a: all

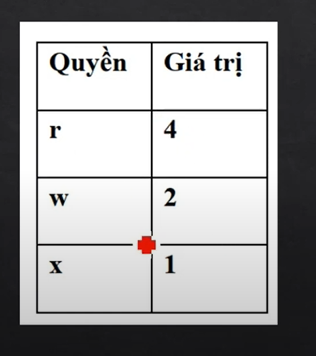
g: group

u: owner

o: orthers

+: thêm quyền

-: bớt quyền



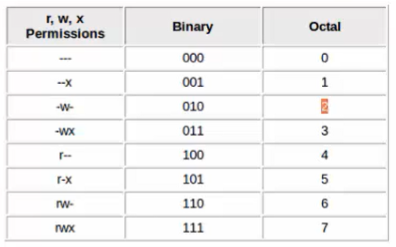
EX: chmod ug+wr pass\_mysql.txt

* R: Áp dụng đối với thư mục làm cho lệnh cps tác dụng lên tất cả các thư mục con

directory permission: thay đổi quyền của một thư mục

Octal and Numerical permissions (chmod)

r w x = 1 1 1 = 7 binary



Ex: chmod 764 = -rwxrw-r--

#chown: đổi người sở hữu tập tin

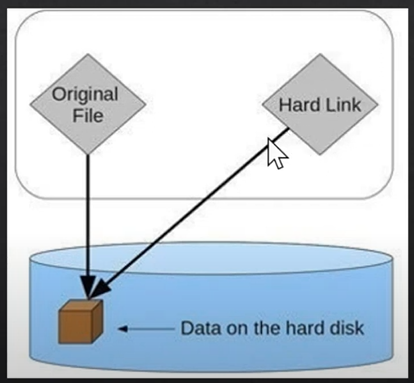
#chgrp: đổi nhóm sở hữu mới

# Hard link và symbolic link

## LINKS

* Cách để tạo ra một tham chiếu tới các tệp hoặc thư mục khác
* Có 2 loại tham chiếu: hard link và symbolic link

### Hard link



* Là 1 liên kết trực tiếp đến một inode trong hệ thống
* Mỗi một lần muốn tạo ra file hardlink cũng giống như tạo ra một file mới nhưng những file này đều trỏ cùng 1 inode trong ổ cứng
* Đều trỏ đến một vị trí trên ổ cứng giống nhau
* Muốn xóa một hardlink phải xóa toàn bộ những thằng đang trỏ đến cùng vị trí đó trên ổ cứng

#### **Lưu ý**

* Không thể tạo hard link cho thư mục, chỉ hd với các tệp
* Khi xóa một hard link, dữ liệu vẫn còn tồn tại cho đến khi không còn lk nào trỏ tới nó nữa
* Khi thay đổi nd của tệp thông qua hard link thực chất là đang thay đổi dữ liệu trong inode mà tất cả hard link đang trỏ tới . Do đó, tất cả thay đổi nào khi thực hiện sẽ được thay đổi trên tất cả các hard link cùng trỏ tới inode đó

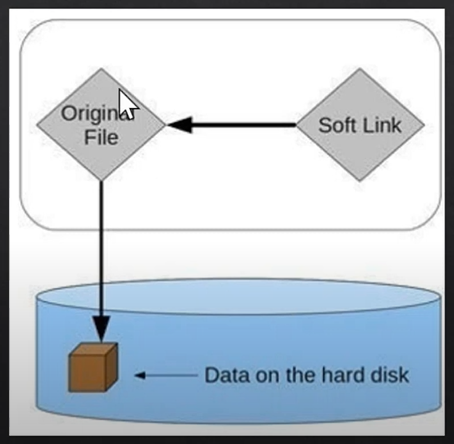
#### **Khi nào nên sử dụng hard link**

#### Tạo ra bản sao sử dụng ở nhiều nơi, nhưng không muốn phát sinh dung lượng, và khi muốn backup dữ liệu ở nhiều nơi khác nhau nhưng không làm tăng cái dung lượng của ổ cứng, truy cập nhanh chóng với nhiều tên khác nhau.

* Khi cần tiết kiệm không gian đĩa và không mốn sao chép dữ liệu thực sự.
* Khi muốn duy trì các phiên bản và sao lưu dữ liệu hiệu quả.
* Khi muốn truy cập nhanh chóng đến các tệp với nhiều tên khác nhau.

Cấu trúc tạo hard link: ln [tệp gốc] [tệp hl]

### Symbolic link (soft link)



* Không trỏ thẳng trực tiếp đến vùng nhớ của ổ cứng mà sẽ trỏ đến file gốc (giống như tạo sort cut trên win)
* Là một tập tin khác hoàn toàn
* Nếu xóa file gốc thì sl sẽ bị lỗi

### Lưu ý

* SL cho phép tạo lk tới các tệp và thư mục ở các vị trí khác nhau trên hệ thống tệp
* Có thể tạo, xóa hoặc di chuyển các sl mà không ảnh hưởng đến tệp gốc.
* Sl có thể trỏ tới các tệp và thư mục trên các ổ đĩa khác nhau.
* Truy cập có thể bị chậm hơn so với tệp gốc

Khi nào sử dụng sl

* Khi cần tạo liên kết các tệp và thư mục ở các vị trí khác nhau
* Khi tạo các biến thể mà không làm ảnh hưởng đến tệp gốc
* Khi cần liên kết giữa các ổ đĩa hoặc phân vùng khác

ln –s [file gốc] [slink]

# Bash script

#Which: trả về một đường dẫn chứa tệp hoặc lệnh được thực thi

Which bash, which ls

#Whatis : trả về một mô tả ngắn giới hiệu về cách sử dụng của lệnh

Whatis ls, whatis cat, …

**----------------------------------------------------------------------------------------------------------------**

**USER**

# useradd: dung để tạo người dùng mới

Flag:

- m => dùng để tạo thư mục chính mặc định cho người dùng này

–s => dùng để mặc định shell –m /usr/bin/bash

–g => chỉ định nhóm người dùng –g users or –g root

–c => thêm các mô tả về người dùng này : -c “This is a root user”

–aG => a thêm người dùng vào nhóm mà không loại bỏ họ khỏi các nhóm khác đang thuộc về

G Chỉ định các nhóm bổ xung mà người dùng sẽ tham gia

# userdel: dùng để xóa người dùng

Flag: Nếu không có flag thì sẽ xóa hết chừa lại thư mục home của người dùng đó => giúp quản lý được nội dung, dữ liệu của người đã làm

–r : xóa hết tất cả kể cả thư mục home của người dùng

#usermod: Thêm người dùng vào nhóm bất kỳ

Flag:

–a: Thêm người dùng vào nhóm mà không loại bỏ khỏi các nhóm đang ở hiện tại

–G: chỉ định các nhóm bổ xung mà sẽ tham gia

Ex: sudo usermod -aG root AnPhuc

Thiết lập chính sách (policy) cho user

Chage: [option] login name

* Flag:
  + –l: xem chính sách của user
  + –E: Thiết lập ngày hết hạn account
  + – I: thiết lập số ngày bị khóa sau khi mk hết hạn
  + –m: thiết lập số ngày tối thiểu cho phép thay đổi password
  + –M: số ngày tối đa
  + –W: số ngày cảnh báo trước khi hết hạn mk

**----------------------------------------------------------------------------------------------------------------**

**GROUP Managemet**

**Đường đẫn đến tệp chứa các nhóm đang có ở trên máy: /etc/group**

**#groups: xem người dùng hiện tại đang ở những nhóm nào**

Ex:

┌──(kali㉿kali)-[~]

└─$ **groups**

**kali adm dialout cdrom floppy sudo audio dip video plugdev users netdev bluetooth scanner wireshark kaboxer**

#groupadd: Thêm nhóm

* Flag:
  + –g: xác định GID của nhóm

#groupdel: Xóa nhóm

#groupmod: Sửa thông tin nhóm

* Flag:
  + –g: sửa GID
  + –n: sửa tên nhóm

#gpasswd: Thêm người dùng vào một nhóm

Flag:

–a: Thêm người dùng vào nhóm

–d: Xóa người dùng khỏi nhóm

┌──(kali㉿kali)-[/var/log/apt]

└─$ **sudo gpasswd -a AnPhuc adm**

**Adding user AnPhuc to group adm**

#bashrc file: được thực thi khi bắt đầu mở một session mới

#df: Hiển thị, báo cáo dung lượng ổ đia trống đang được hệ thống sử dụng

#du: hiển thị các tệp và dung lượng ổ đĩa ở tại một thư mục nào đó

Flag:

–h: hiển thị giá trị cho người có thể đọc

–s: tóm tắt và hiển thị tổng dung lượng có trong thư mục

#free: Hiển thị tổng dung lượng còn trống và đã được sử dụng

Flag:

–m:

–g

–t

#watch: lệnh thực thi lặp đi lặp lại 1 tiến trình trong một khoảng thời gian (mặc định là 2s)

Flag:

* n: số n tầng xuất dược thực thi

┌──(AnPhuc㉿kali)-[~]

└─$ **watch -n 1 df -h**

**----------------------------------------------------------------------------------------------------------------**

LỊCH CALENDER\_COMMAND

#cal: in ra lịch tháng hiện tại trên máy

#date: in ra ngày, giờ hiện tại của máy

Flag:

–s: set Thay đổi ngày giờ trên máy

+%h %d %y %D ….

Thực thi nhiều lệnh trên cùng một dòng

Sử dụng ; ngăn cách giữa nhiều lệnh. VD: ls ; pwd

Nếu một lệnh bị lỗi, nguyên câu lệnh sẽ vẫn có thể thực thi

Sử dung && VD: ls && pwd && date

Nếu một lệnh bị lỗi, nguyên câu lệnh sẽ không thể thực thi

Sử dụng ||. Nếu một lệnh đúng thì lệnh phí sau sẽ không thực thi và ngược lại

# Cài đặt và cấu hình mạng

## Các file cấu hình network

### /etc/hosts

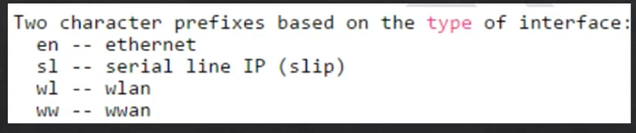
* Dùng để phân giải hostname không thể phân giải đc, sd để thay dns

### /etc/resolv.conf

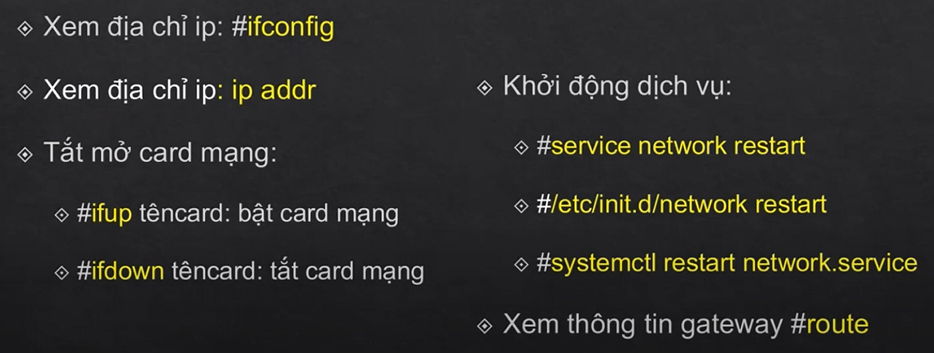
* Chi định DNS server để phân giải tên miền

## Định dạng card mạng

* Ls /sys/class/net

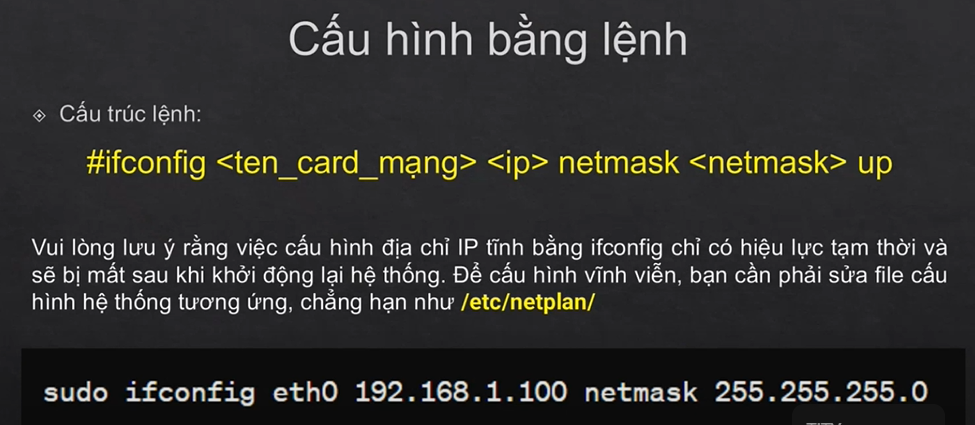


## Các lệnh network

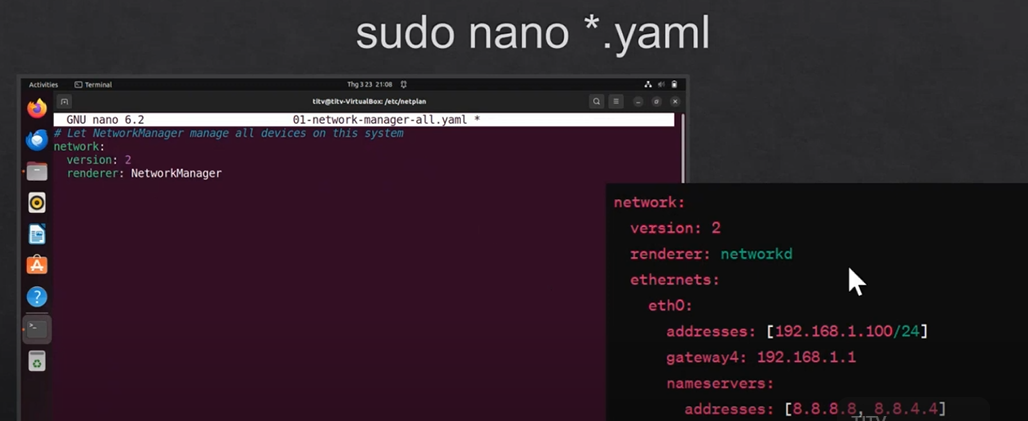


## Cấu hình địa chỉ IP

### Cấu hình bằng lệnh



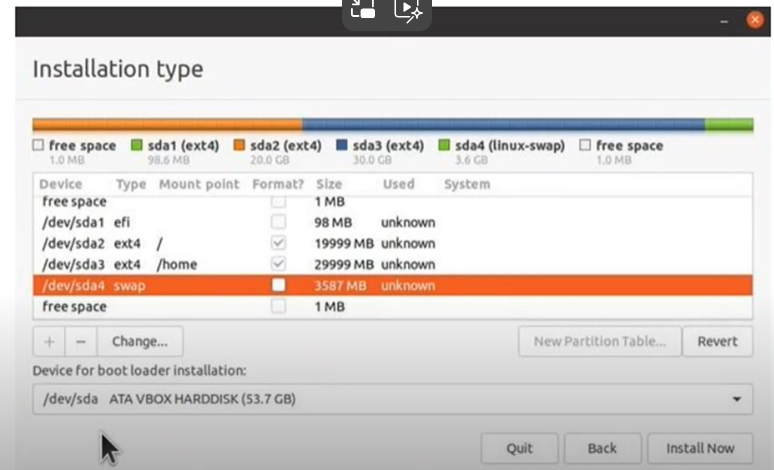
### Cấu hình trên file



Cấu hình trên giao diện

* Sử dụng nmtui

# Linux File Systems



## Hệ thống tệp (File System)

Là cơ chế tổ chức và quản lý các tệp và thư mục trên ổ đĩa. Các hệ thống tệp phổ biến trong Linux bao gồm:

* ext4: Hệ thống tệp mặc định và phổ biến nhất, hỗ trợ các tính năng như journaling, hỗ trợ kích thước tệp và hệ thống tệp lớn.
* XFS: Hệ thống tệp hiệu suất cao, thích hợp cho các ứng dụng yêu cầu quản lý tệp lớn.
* Btrfs: Hệ thống tệp hiện đại với nhiều tính năng như snapshot, RAID, và tự sửa lỗi.
* FAT32/NTFS: Thường được sử dụng cho các thiết bị lưu trữ di động để tương thích với cả Linux và Windows.

## Một số lệnh dùng để quản lý ổ cứng

Fdisk: để phân chia phân vùng (partition) của ô cứng

# LẬP trình SHELL

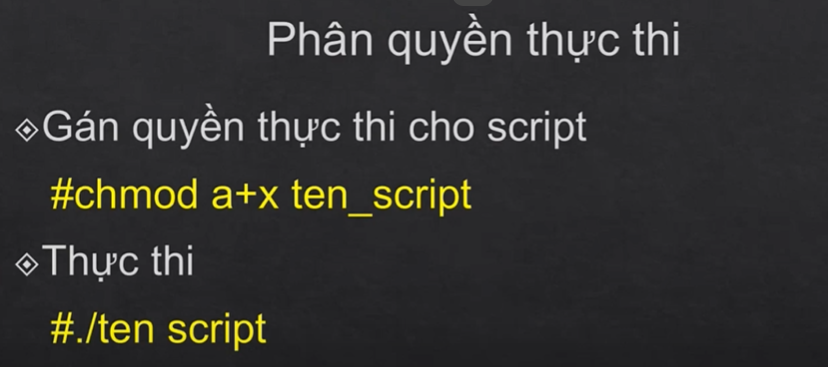
* Là việc viết ra các tập lệnh (script) để thực hiện các tác vụ trong môi trường dòng lệnh.
* Script shell thường được sử dụng để tự động hóa các công việc lặp đi lặp lại hoặc dể thực hiện các tác vụ phức tạp.

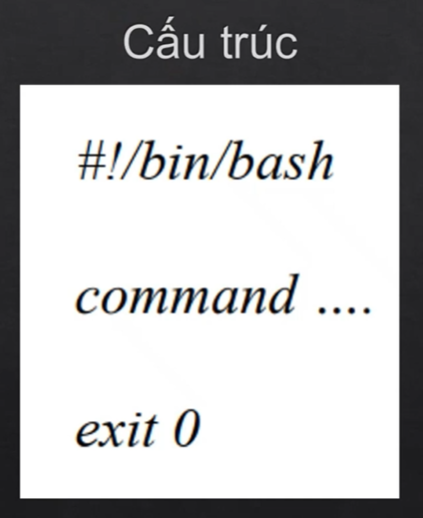
## Các loại shell thông dụng trên Unix/linux

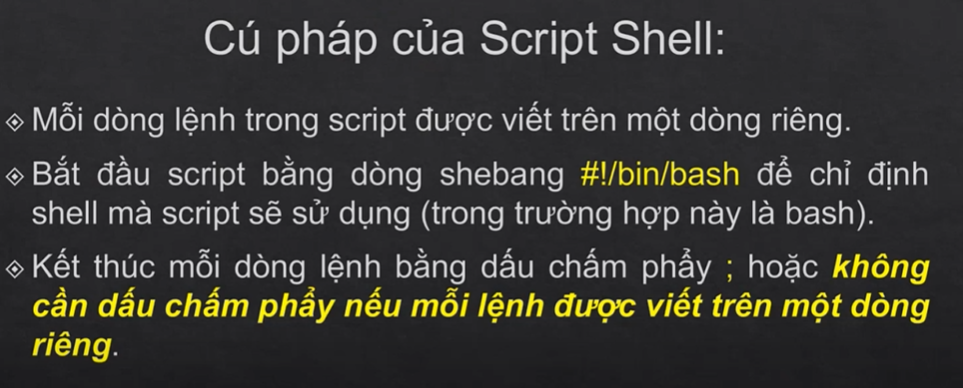
Sh (shell Bourne): Shell nguyển thủy có mặt trên hầu hết các hệ thống linux. Rất hữu dụng cho việc lập trình shell nhưng nó không sử lý tương tác người dùng như các shell khác…

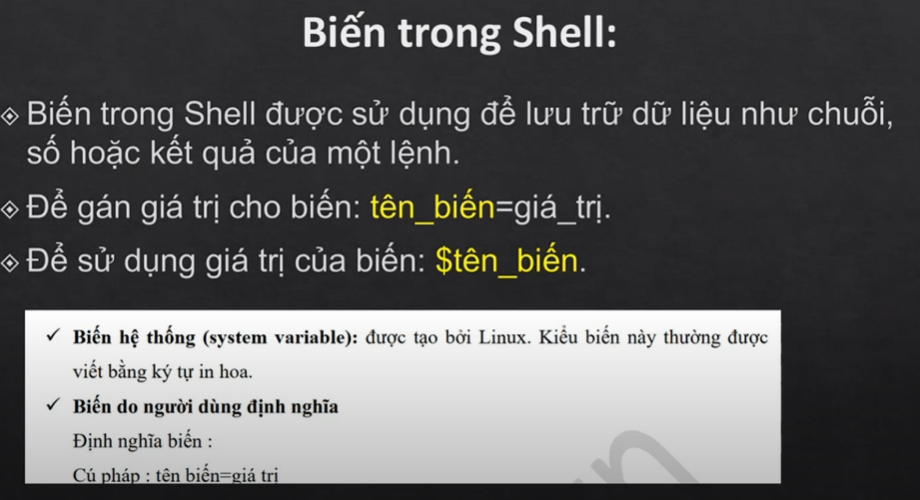
Bash: Phần mở rộng của sh, kế thừa những gì sh đã có và phát huy những gì sh chưa có. Đây là shell mặc định trên các hệ thống linux.

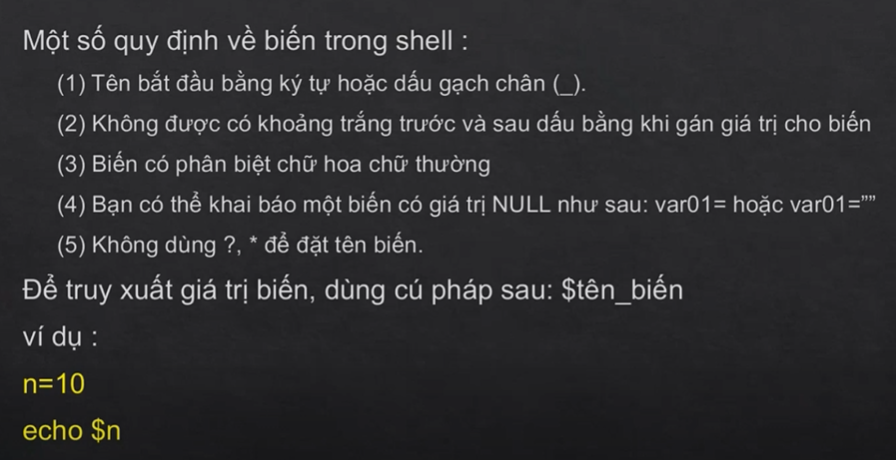
Csh, tcsh và zsh: Dòng shell sử dụng cấu trúc của C. Là shell thông dụng thứ 2 sau bash shell.











## LỆNH NHẬP / XUẤT

### LỆNH IN CHUỖI

Echo

Dấu backticks(‘) thay cho $()