Lê Trương Ngọc Duyên – B2105569

ÔN TẬP LÝ THUYẾT QUẢN TRỊ HỆ THỐNG

WELCOME

Người quản trị hệ thống làm những gì?

- Quản lý tài khoản user
- Quản lý phần cứng
- Sao lưu *(quan trọng nhất),* khôi phục hệ thống tập tin
- Cài đặt và cấu hình phần mềm và dịch vụ mới
- Giữ cho hệ thống và dịch vụ hoạt động ổn định
    - Theo dõi hệ thống và mạng
    - Khắc phục sự cố
- Lập tài liệu hệ thống (mô tả phần cứng, quy định mật khẩu)
- Kiểm tra bảo mật an ninh
- Hỗ trợ user, điều chỉnh hiệu suất và hơn thế nữa

DESKTOPS

QUẢN LÝ NHIỀU MÁY CÁ NHÂN DESKTOPS

3 công việc chính:

- Cài đặt ban đầu
- Nâng cấp, cập nhật
- Cấu hình mạng

Yêu cầu:

- Cài đặt ban đầu phải đồng nhất giữa các máy
- Cập nhật phải nhanh
- Cấu hình mạng tốt nhất phải quản lý tập trung

➔**Giải pháp** là càng tự động hóa càng tốt

-CÀI ĐẶT BAN ĐẦU OS VÀ ỨNG DỤNG

Tự động hóa giải quyết đc nhiều vấn đề:

- Tiết kiệm thời gian/tiền; giảm lỗi; đồng bộ hóa

Tự động toàn phần sẽ tốt hơn 1 phần. Tự động 1 phần sẽ tốt hơn là ko tự động.

-CẬP NHẬT HỆ THỐNG VÀ ỨNG DỤNG

Theo thời gian, thì sẽ có bugs, lỗ hổng bảo mật và phiên bản mới ➔ Cần cập nhật tự động hóa

Cập nhật thực hiện trên máy đang hoạt động

Thực hiện cập nhật cực kỳ cẩn thận. Triển khai từ từ

-CẤU HÌNH MẠNG

Thiết lập các thông số mạng cho các thiết bị, máy tính để chúng có thể kết nối được mạng

Có 2 cách:

- Cấu hình tĩnh (static/manual)
- Cấu hình động (dynamic/automatic)

DHCP (Dynamic Host Configuration Prototype): giao thức cấu hình mạng động

➔ Giải pháp cấu hình mạng cho 1 *số lượng lớn máy tính*: ***DHCP*** (Dynamic Host Configuration Prototype): giao thức cấu hình mạng động

QUẢN LÝ MÁY CHỦ SERVER

- Máy chủ phục vụ hàng trăm, ngàn user.
- Yêu cầu độ tin cậy và hoạt động liên tục
- Yêu cầu bảo mật an ninh chặt chẽ

Máy chủ thường:

- Có cấu hình OS khác với các máy desktops
- Đc triển khai trong trung tâm dữ liệu ***(data center)***
- Có các hợp đồng duy trì
- Hệ thống đĩa sao lưu
- Phải đc truy cập từ xa (SSH)

-PHẦN CỨNG MÁY CHỦ

Mua phần cứng máy chủ:

- Không gian bên trong lớn hơn
- Hiệu năng CPU tốt hơn
- Đĩa cứng, mạng và I/O có hiệu năng cao hơn
- Nhiều lựa chọn để nâng cấp

Lựa chọn nhà cung cấp có độ tin cậy cao

-SAO LƯU DỮ LIỆU

Lê Trương Ngọc Duyên – B2105569

Máy chủ thường chứa dữ liệu quan trọng → phải thường xuyên sao lưu để phục hồi dữ liệu khi có sự cố

-MÁY CHỦ ĐC ĐẶT TRONG TRUNG TÂM DỮ LIỆU (DATA CENTER)

Trung tâm dữ liệu là nơi đặt các máy chủ server và 1 số thiết bị CNTT khác

Trung tâm dữ liệu cung cấp:

- Nguồn điện
- Mạng tốc độ cao
- Hệ thống làm mát (điều hòa, quạt, chất lỏng)
- Hệ thống phòng cháy chữa cháy
- An ninh vật lý (bảo vệ, cửa, khóa...)

-QUẢN TRỊ TỪ XA

Thường máy chủ đc đặt trong data center, ở đó thường lạnh, ồn và có thể xa văn phòng quản trị → Quản trị từ xa.

-Ổ CỨNG

Máy chủ thường có nhiều ổ cứng.

➔Sử dụng công nghệ ***RAID***: là công nghệ kết hợp nhiều ổ cứng vật lý thành 1 hệ thống ổ cứng →để đảm bảo an toàn dữ liệu, tăng tốc độ đọc ghi

- Cho phép dữ liệu ghi xuống nhiều ổ cứng cùng lúc, khi 1 ổ cứng bị hư thì dữ liệu vẫn an toàn
- RAID 0: chia đều dữ liệu ra cho từng ổ cứng → 1 ổ hư sẽ mất dữ liệu. Mục đích: tận dụng tốc độ truy xuất nhanh
- RAID 1: (mirror) dữ liệu đc nhân lên → dữ liệu trong các ổ đều giống nhau. Nếu 1 ổ cứng hư thì đảm bảo dữ liệu vì còn những ổ cứng khác.
- RAID 5: cần dùng >=3 ổ cứng và cho phép tối đa 1 ổ cứng bị sự cố
- RAID 10 (1+0): kết hợp giữa RAID 0 và RAID 1

-BỘ NGUỒN

Trên máy chủ thường có nhiều bộ nguồn.

-ROUTER

Router thường có 2 bộ nguồn.

-HOT-SWAP

Các phần cứng có dự phòng như ổ cứng, bộ nguồn... có chức năng hot-swap

→Hot-swap: Trong quá trình thay phần cứng thì server vẫn hoạt động bình thường.

XÂY DỰNG MÁY CHỦ

Khi cần máy chủ thường có 3 giải pháp:

- Mua máy chủ và xây data center mini
- Mua máy chủ và thuê chỗ đặt máy chủ
- Thuê máy chủ

QUẢN LÝ DỊCH VỤ

Quản trị dịch vụ là lý do chúng ta mua máy chủ

Có nhiều dịch vụ:

- DNS, email, chứng thực, mạng
- Điều khiển từ xa, truy cập Internet, dịch vụ tập tin

Cung cấp 1 dịch vụ nghĩa là:

- Ko chỉ kết hợp giữa phần cứng và phần mềm
- Làm cho dịch vụ tin cậy
- Mở rộng dịch vụ
- Theo dõi, duy trì và hỗ trợ dịch vụ

-THIẾT KẾ 1 DỊCH VỤ BỀN VỮNG

Lấy yêu cầu người dùng

- Lý do cài dịch vụ
- Chỉ ra mức độ chất lượng dịch vụ (dịch vụ càng nhanh càng tốt → chi phí càng lớn)

Những yêu cầu cho dịch vụ hoạt động

- Dịch vụ đó hoạt động thì cần dịch vụ nào khác?
- Làm sao để quản trị dịch vụ
- Mở rộng dịch vụ ntn?
- Làm sao để nâng cấp?
- Xem xét có cần phần cứng phần mềm gì đặc biệt?
- Xem xét tác động đến hiệu năng mạng?

Xem ngân sách có đáp ứng đủ?

Xem xét các kiến trúc/công nghệ mở

Tuân thủ theo nguyên tắc KISS (Keep It Simple, Stupid)

Triển khai trong môi trường có hỗ trợ (data center)

Tin cậy

- Xây dựng trên phần cứng, server tin cậy
- Có dự phòng cần thiết (ổ cứng, nguồn điện, mạng... dự phòng để khi có sự cố vẫn hđ bình thường)

Hạn chế truy cập

Theo dõi hiệu năng dịch vụ

Theo dõi dịch vụ

ACCOUNTS

Tập tin **etc** là thư mục dùng để lưu trữ các tập tin cấu hình của hệ thống

THE /ETC/PASSWD FILE

/etc/passwd là nơi lưu trữ thông tin người dùng (dạng text)

Thông tin của mỗi user đc lưu trên 1 dòng, mỗi thành phần cách nhau bằng ***":"***

- Tên đăng nhập (login name)
- Mật khẩu đc băm (hiện giờ đã đc lưu trong /etc/shadow nên thay thế bằng ***"x"***)
- Mã số người dùng (UID number)
- Mã số người dùng mặc nhiên (default GID number)
- Full name, office, home phone (ko bắt buộc)
- Thư mục cá nhân của người dùng (home directory). VD: /home/<login name>
- Login shell (môi trường shell). VD: /bin/bash

-Login name/username

- Username phải là duy nhất trên hệ thống, ko đc trùng nhau
- Tối đa là 32 ký tự
- Bất kỳ ký tự nào, ngoại trừ xuống hàng và dấu hai chấm

-Encrypted passwords

- Ko đc lưu trong /etc/passwd nữa mà lưu trong ***/etc/shadow***

-UID number

- User có quyền trong hệ thống thì UID càng nhỏ
- Root có UID là 0
- Còn user thực tế thì có UID>=1000

THE /ETC/SHADOW FILE

/etc/shadow là nơi chứa thông tin mật khẩu của user

Chỉ những người có quyền quản trị mới đc mở tập tin này

Mã hóa và giải mã

- Abc123 ⇔ ?!*#@... (mã hóa mật khẩu, 2 chiều)
- Sử dụng giải thuật DES
- Nhược điểm: Khi đăng nhập, máy tính phải giải mã mk được lưu để xem mk nhập vào có giống mới mật khẩu đã mã hóa trên máy không ⇒ máy tính biết được mật khẩu

Hash – băm

- Abc123 ➔ ?!*#@... (băm mật khẩu, 1 chiều)
- Mật khẩu có kích thước cố định mặc dù đầu vào có kích thước bao nhiêu
- 2 dữ liệu đầu vào khác nhau thì mã hóa khác nhau
- Nhược điểm:
  - Mật khẩu giống nhau thì mã hóa giống nhau
  - Từng ký tự có 1 mã hóa riêng (chỉ cần biết đc thì sẽ phá đc passwd)
- Giải pháp : thêm 1 giá trị R1 ngẫu nhiên rồi mới mã hóa

Các thuật toán HASH tiêu biểu:

- MD5: `$1$`
- SHA-512: `$6$`
- Yescrypt: `$y$`

THE /ETC/GROUP FILE

/etc/group chứa các nhóm user

Thông tin của mỗi nhóm đc lưu trên 1 dòng, mỗi thành phần cách nhau bằng ***":"***

- Tên nhóm
- Mật khẩu nhóm (nếu có)
- Mã số nhóm (GID)
- Danh sách các user thuộc nhóm

➔Muốn phân quyền cho user thì thường phân quyền trên nhóm

- Mỗi khi tạo ra 1 tài khoản thì hệ thống sẽ tạo ra 1 nhóm trùng tên với tên TK. Và TK đó sẽ thuộc nhóm đó

ADDING USERS (THÊM NG DÙNG)

- Tạo tài khoản người dùng với lệnh `adduser (hoặc useradd)`
- Đặt mật khẩu cho người dùng với lệnh `passwd`
- Thay đổi thông tin người dùng với lệnh `usermod`

REMOVING USERS (XÓA NG DÙNG)

- Xóa người dùng với lệnh `userdel`
- Xóa luôn dữ liệu cá nhân người dùng với lệnh `userdel -r`

DISABLING LOGINS (VÔ HIỆU HÓA ĐĂNG NHẬP)

Thỉnh thoảng có thể khóa 1 TK nào đó

Có thể khóa TK bằng cách khóa mật khẩu

- Khóa MK với lệnh `$sudo usermod –L <tên tk>` →Thông tin MK trong tập tin shadow sẽ có dấu "!"
- Mở khóa MK với lệnh `$sudo usermod –U <tên tk>`

ACCOUNT MANAGEMENT UTILITIES (CÔNG CỤ QUẢN LÝ TK)

`adduser` (`useradd`) để tạo tài khoản người dùng

`usermod` để thay đổi thông tin người dùng

`userdel` để xóa người dùng

`groupadd`, `groupmod`, `groupdel` lần lượt để thêm nhóm, thay đổi thông tin nhóm và xóa nhóm

- `$sudo usermod –aG <tên nhóm> <tên user>` để thêm user vào 1 nhóm
- `$sudo gpasswd –d <tên user> <tên nhóm >` để đưa user ra khỏi nhóm

THE SUPERUSER

Tài khoản ***root*** có UID là ***0***

Tài khoản superuser này có thể thực hiện bất cứ chuyện gì trên bất cứ tập tin hoặc tiến trình.

Tất cả tài khoản còn lại là bình thường, có quyền giống nhau và ko có quyền QTHT

Thực hiện thao tác hạn chế trên hệ thống ➔ Phải đăng nhập vào TK root mơi thực hiện đc những thao tác quản trị này:

Tạo user, thay đổi nhóm user, thay đổi tên máy tính, cấu hình mạng...

Chọn mật khẩu cho TK root

- >=8 ký tự
- Dễ nhớ, khó đoán, duy nhất, ko tiết lộ
- Thường xuyên đổi mật khẩu root

BECOMING ROOT

Có thể đăng nhập trực tiếp vào HT bằng TK root ➔ Ko khuyến khích

Being root

- Khi nắm TK root thì phải có trách nhiệm
- Ko đưa MK root cho ng khác
- Ko tạo TK mới với UID là 0
- Chỉ sử dụng TK root cho công việc quản trị
- Thay đổi MK root thường xuyên
- Cận thận

su

Lệnh **`su`**: switch users - chuyển đổi người dùng

- `$su <tên tk>`

sudo

Lệnh **`sudo`**: thực hiện lệnh gì đó bằng quyền quản trị

- `$sudo <lệnh>`

Để cho 1 TK có quyền sudo thì thêm TK đó vào nhóm **`wheel`** (nhóm wheel có toàn quyền sudo)

➔Lợi ích của `sudo`:

- Kiểm toán: đc ghi log lại hết
- Thực hiện đc mọi quyền mà ko cần cấp TK root
- MK root ít người biết ➔ an toàn hơn
- **`sudo`** nhanh hơn **`su`**

➔Nhược điểm của `sudo`:

- Người dùng có quyền sudo phải bảo vệ TK đó như bảo vệ TK root
- Khi mở tập tin `/etc/sudoers` phải cẩn thận

INSTALLATION

DAEMONS

Là các tiến trình background chạy nền, user ko giao tiếp trực tiếp với TT background

Thường viết tắt là `d`

Tương ứng với service trên Windows

Các loại daemons (service) phổ biến: có tên kết thúc bằng chữ `d`

- `httpd`: dịch vụ web
- `vsftpd`: dịch vụ ftp → cho phép download/upload file

Lệnh `$sudo systemctl <start / stop / restart / status> <tên dịch vụ>`: để điều khiển các dịch vụ

FTP SERVERS

FTP – File Transfer Protocol: giao thức thực hiện tập tin để upload hoặc download file/dir

SOFTWARE INSTALLATION (CÀI ĐẶT PHẦN MỀM)

1 người QTHT phải biết:

- Cài đặt hệ điều hành
- Tự động hóa việc cài đặt cho số lượng lớn máy tính
- Customize (tinh chỉnh) cho hệ thống
- Giữ cho hệ thống đc cập nhật

Package management (Quản lý gói phần mềm)

Quản lý, cài đặt và nâng cấp thì sử dụng công cụ quản lý phần mềm

- .rpm cho Red Hat
- .deb cho Debian, Ubuntu

Công cụ quản lý gói phần mềm sẽ tự động tải và cài đặt phần mềm

- Red Hat có công cụ yum, dnf
- Debian, Ubuntu có công cụ apt, snap
- Mac có công cụ brew

dnf là bản cải tiến của **yum** (CentOS)

- Cài đặt: `$sudo dnf install <tên phần mềm> -y`
- Gỡ bỏ: `$sudo dnf remove <tên phần mềm> -y`

BOOTING

BOOTING PCs (KHỞI ĐỘNG MÁY TÍNH)

Có 2 cách khởi động chính:

- BIOS booting
  - **BIOS** đc thực thi → ***MBR*** (512B first) đc nạp lên (MBR chứa thông tin boot loader)→ Boot loader (***GRUB***) (lựa chọn OS để khởi động)→ Nạp OS kernel → Nhận diện phần cứng → Single user → Chạy startup script (thiết lập các trạng thái ban đầu) → Multi-user
- UEFI booting: hiện nay đang đc sử dụng, có những tính năng khởi động an toàn, nâng cấp hơn.
  - UEFI đc thực thi → Tạo phân vùng riêng trên ổ cứng để chứa thông tin boot loader → *Tiếp tục giống BIOS*

BIOS booting:

- Khi nhấn nút Power, máy tính bắt đầu thực hiện đoạn mã khởi động trong bộ nhớ ROM (BIOS – Basic Input Output System)
- Muốn tạm dừng chỉnh sửa BIOS để thiết lập các cấu hình thì nhấn phím đặc biệt (Del, F8, F11) tùy máy
- Đọc 512B đầu tiên của đĩa khởi động, đc gọi là MBR-Master Boot Record
- ***MBR*** chứa chtr mà chỉ định phân vùng nào sẽ tải boot loader.

Boot loaders

- Là chtr quản lý OS trên máy tính, cho phép lựa chọn OS để khởi động
- ***GRUB*** là boot loader hiện đại của Linux

Cấu hình phần cứng

- Sau khi OS khởi động lên thì sẽ nhận diện phần cứng → OS nhận diện có những phần cứng nào và giao tiếp phần cứng

- Driver là phần mềm nhỏ nằm giữa OS và phần cứng. Để các phần cứng có thể giao tiếp đc với OS thì cần driver

Tạm dừng qtr khởi động

- Tạm dừng qtr khởi động (nếu muốn) → khởi động thủ công để thực hiện 1 số công việc nào đó (xử lý lỗi...) .
- Nếu bỏ qua thì OS sẽ tiếp tục khởi động

Booting into single user mode (vào chế độ đơn ng dùng)

Startup scripts

Là đoạn mã boot cho hệ thống để OS sẵn sàng phục vụ

- Cấu hình tên máy
- Setting múi giờ
- Quét đĩa
- Xóa tập tin ko cần thiết
- Cấu hình mạng
- Chạy những dịch vụ cần thiết

Multisuer operation (Chế độ đa ng dùng)

- Lựa chọn ng dùng nào để đăng nhập vào hệ thống

MỨC ĐỘ HOẠT ĐỘNG (INIT AND RUN LEVELS)

Có 7 mức độ thực thi, đc đánh dấu từ 0→6

- 0: đang shut down
- 1: đang ở chế độ single user
- **2 → 5**: đang ở chế độ multi-user
  - 2: ko có mạng
  - 3: có mạng (ko có GUI)
  - 4: có mạng nhưng chưa dùng (dự phòng)
  - 5: có mạng + GUI (giao diện đồ họa)
- 6: đang khởi động lại

→Để chuyển đổi các mức độ thực thi dùng lệnh `$sudo telinit <mức độ: 0→6>`

REBOOTING AND SHUTTING DOWN (Khởi động lại và tắt máy)

Ko cần thường xuyên khởi động lại và tắt máy như PC. Chỉ khi cần thiết:

- Thêm hoặc gỡ bỏ phần cứng
- Thay đổi cấu hình booting

SHUTDOWN

- `$shutdown`: sau 60s mới tắt
- `$shutdown` **`now`**: tắt liền
- `$shutdown` **`-r +15`**: sau 15p tắt và khởi động lại (với tham số-r)
- `$shutdown` **`-h`** `20:00`: đặt giờ tắt (với tham số -h)

DISKS

DISK INTERFACES (GIAO DIỆN ĐĨA)

Là những chuẩn, công nghệ cho phép ta giao tiếp với ổ cứng

2 dòng ổ cứng phổ biến:

- HDD: lưu trữ bằng từ tính. Dùng cho lưu trữ lượng lớn dữ liệu
- SSD: lưu trữ bằng bộ nhớ Ưu: tốc độ đọc ghi nhanh, gọn nhẹ, ít tốn năng lượng nhưng hơi mắc

Một số chuẩn, công nghệ giao tiếp ổ cứng:

- SCSI: khá phổ biến trên server đời cũ
- IDE: có 2 công nghệ là chuẩn ATA và SATA. SATA cho phép giao tiếp với ổ cứng
- Fibre Channel
- USB: Dùng cho ổ cứng rời
- PCIe: phổ biến gần đây

→ Công nghệ phổ biến hiện nay: SATA hoặc PCIe

➔Những nơi lưu trữ dl lớn dùng thiết bị chuyên dụng

Black box

- 40+2 SATA ổ cứng
- Dùng công nghệ RAID (nhiều ổ cứng kết hợp với nhau)
- Lên đến 80TB

Sun X4500 (của Oracle)

- 48 SATA ổ cứng
- Lên đến 48TB

THÊM Ổ CỨNG VÀO LINUX

Gắn ổ cứng vào

Lê Trương Ngọc Duyên – B2105569

Khởi động để kiểm tra OS có nhận diện đc ổ cứng chưa

Phân vùng ổ cứng

- Trên linux có lệnh **`fdisk`** để phân vùng ổ cứng

Format định dạng ổ cứng: chỉ ra cho OS biết trên ổ cứng dùng chuẩn tập tin gì

- ***Chuẩn tập tin*** là chỉ ra cách dữ liệu đc lưu trên ở cứng
- Chuẩn tập tin trên OS Windows là FAT, NTFS
- Chuẩn tập tin trên OS Linux là ext4
- Trên Linux có lệnh **`mkfs`** để định dạng phân vùng ổ cứng

Add entries to **`/etc/fstab`**

Mount ổ cứng: là gắn kết thư mục tới 1 phân vùng, khi giao tiếp với phân vùng thì qua thư mục đó. VD: khi ghi dl vào thư mục là ta ghi dl vào phân vùng đó

- Dùng lệnh **`mount`** `<đg dẫn ổ cứng> <đg dẫn thư mục>` để mount ổ cứng vào thư mục
- Lệnh **`df`** `<đg dẫn thư mục>` để kiểm tra lại thông tin

Phân vùng ổ cứng

Có 2 công nghệ: Disk partitions và Logical Volume

PARTITIONS

- Là công nghệ cũ, cho phép chia ổ cứng lớn thành nhiều phân vùng nhỏ.
- Mỗi phân vùng nhỏ thì OS xem nó như ổ cứng độc lập → Format các chuẩn tập tin khác nhau trên các phân vùng đó đều đc

VOLUME

Là công nghệ mới, có nhiều tính năng mới hơn, tốt hơn Disk partitions

- Sử dụng và phân vùng ổ cứng hiệu quả hơn
- Tạo nhiều phân vùng và những phân vùng có thể nằm trên nhũng ổ cứng khác nhau. VD: có 2 ổ cứng, phân vùng C nằm phân nửa ổ cứng 1 và phân vùng D nằm nửa ổ cứng 1 và nguyên ổ cứng 2
- Tăng hoặc giảm kích thước của phân vùng khi hệ thống đang hoạt động (ngoại trừ phân vùng chứa OS)

→ Nhiều OS đều hỗ trợ tính năng Volume

Filesystems (Hệ thống tập tin)

Chuẩn tập tin là chỉ ra cách dữ liệu đc lưu trên ở cứng

- Chuẩn tập tin trên OS Linux là ext4 (Fourth Extended File System)
- Chuẩn tập tin trên OS Windows là FAT, NTFS

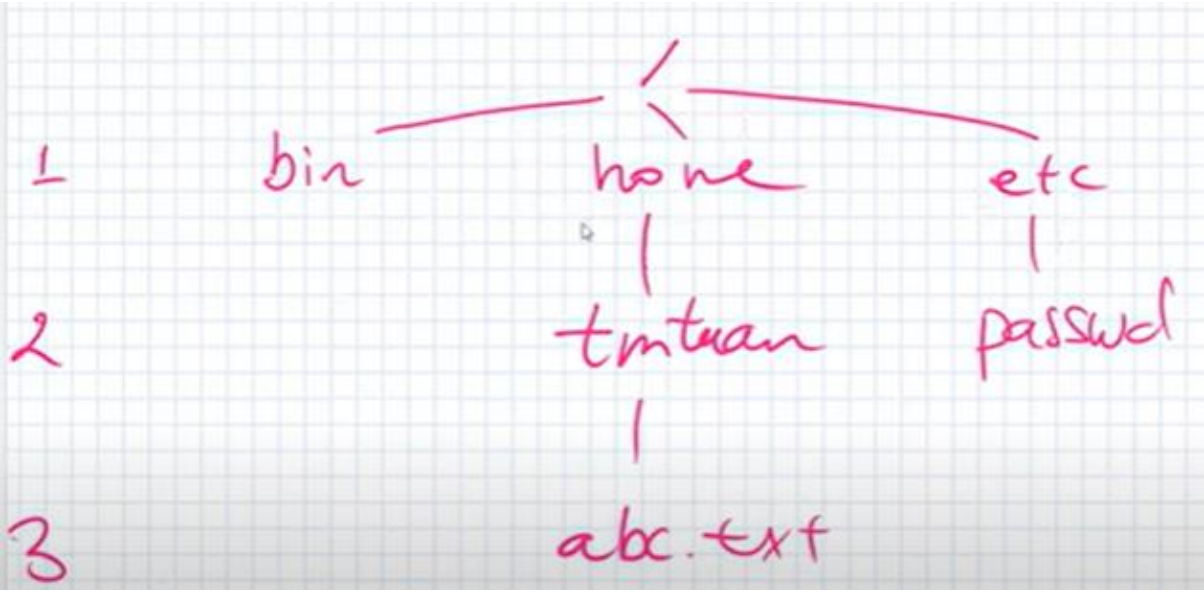
/etc/fstab

Hầu như mọi filesystem sẽ đc gắp kết (mount) tự động vào tập tin root khi khởi động máy sẽ chứa trong **`/etc/fstab`**

Lệnh `fsck`: check and repair filesystems

- Để quét, kiểm tra và sửa lỗi hệ thống tập tin

FILE TREE ORGANIZATION



Tổ chức hệ thống tập tin theo dạng cây

Thư mục cấp 1 là do hệ thống tự động tạo ra

- `/bin`: chứa những lệnh cơ bản của hệ thống
- `/dev`: (device) chứa những file thiết bị như phân vùng ổ cứng, thiết bị ngoại vi. VD: USB...
- `/etc`: chứa những file cấu hình hệ thống
- `/home`: chứa những file cá nhân của từng người dùng
- `/var`: lưu lại tập tin ghi các số liệu biến đổi (/var/log; /var/lib...)
- `/usr`: chứa các ứng dụng, thư viện, tài liệu và mã nguồn các chương trình thứ cấp
- ....

➔Vì tổ chức theo dạng cây nên muốn truy cập đến file hay thư mục phải dùng **đường dẫn (PATHNAMES)**. Có 2 loại đường dẫn:

- **Tuyệt đối:** là đg dẫn đầy đủ tính từ thư mục gốc xuống. VD: /home/tmtuan/abc.txt
- **Tương đối**: là đg dẫn tính từ vị trí hiện hành đang làm việc. VD: /tmtuan/abc.txt

Lê Trương Ngọc Duyên – B2105569

FILE TYPES (LOẠI TẬP TIN)

Linux định nghĩa **7** loại tập tin:

- [-]: tập tin thông thường
- [d]: thư mục
- [c]: các tập tin thiết bị dạng ký tự (character)
- [b]: các tập tin thiết bị dạng nhị phân (block)
- [s]: tập tin để giao tiếp dl qua mạng, thường sử dụng trong lập trình mạng
- [p]: tập tin để giao tiếp giữa 2 chương trình, thường sử dụng trong LT mạng
- [l]: tập tin liên kết đại diện (như shortcut)

FILE ATTRIBUTES (THUỘC TÍNH TẬP TIN)

-Các file trên Linux dùng 12 mode bits và chia thành 4 nhóm, mỗi nhóm 3 bits

3 bits đầu:

- 4000 – setuid
- 2000 – setgid
- 1000 – sticky bit
    - Chỉ có chủ sở hữu của file or thư mục có thể xóa hoặc đổi tên files
    - Giữ cho /tmp private và an toàn hơn

9 bits sau: Quyền cho tập tin → chỉ ra các user có quyền trên tập tin nào đó

- 3 bits đầu là quyền của ***người dùng*** (chủ sở hữu tập tin). Tạo ra tập tin → chủ sở hữu
- 3 bits tiếp là quyền của ***nhóm***
- 3 bít cuối là quyền của ***những ng khác***

-Có 3 quyền cơ bản:

- **r**-read: cho phép mở file lên để đọc – xem bên trong thư mục có chứa gì
- **w**-write: cho phép sửa đổi nội dung file – thay đổi nội dung thư mục
- **x**-execute: cho phép chạy file lên – đi xuyên ngang thư mục

-Quyền mặc nhiên trên các file hay thư mục mới đc tạo:

- Trên file là 666
- Trên thư mục là 777

-Để phân quyền mặc nhiên trên các file hay thư mục mới đc tạo dùng lệnh **umask**

- **umask** = quyền mặc nhiên của file/dir – quyền mặc nhiên mong muốn

-Để thay đổi quyền thì có 2 lệnh thường dùng:

chmod: thay đổi quyền cho tập tin

- `$sudo chmod u+x <file or dir>`: thêm quyền execute cho user
- `$sudo chmod g-w <file or dir>`: xóa quyền write của group
- `$sudo chmod o-r <file or dir>`: xóa quyền read của other (những ng khác)
- `$sudo chmod a=rwx <file or dir>`: thêm cho cả 3 quyền rwx
- `$sudo chmod 750 <file or dir>`: user toàn quyền; group đc read và execute; còn others ko có quyền gì cả.
    - `r`=4=2^2; `w`=2=2^1; `x`=1=2^0

`chown`: thay đổi chủ sở hữu và nhóm.

- Cú pháp: `$sudo chown <user>:<group> <file or dir>`
- `$sudo chown cantho:sinhvien file.txt`: đổi chủ sở hữu là cantho và nhóm là sinhvien
- `$sudo chown cantho file.txt`: đổi chủ sở hữu là cantho
- `$sudo chown :sinhvien file.txt`: đổi nhóm là sinhvien

ACL (access control list) dùng để phân quyền cho nhiều nhóm người dùng khác nhau

$getfacl <file or dir>: xem các quyền truy cập đầy đủ

Lệnh setfacl dùng để thay đổi quyền truy cập.

- `-m` (modify): thay đổi quyền
- `-x` (remove): gỡ bỏ quyền
- `$sudo setfacl <-m ; -x> <u | g | o>:<tên u|g>:<quyền> <file or dir>`
- `$sudo setfacl -m u:sinhvien:rwx file.txt`
- `$sudo setfacl -m g:cantho:--x /dir`
- `$sudo setfacl -r o:rw- /dir`

PROCESSES

CONTROLLING PROCESSES (QUẢN LÝ TIẾN TRÌNH)

COMPONENTS OF A PROCESS (CÁC THÀNH PHẦN CỦA 1 TT)

Đứng trên quan điểm của OS, 1 tiến trình là

- 1 ko gian địa chỉ (gồm các trang bộ nhớ chứa code, thư viện và dữ liệu)
- Cấu trúc dl gồm:
    - Ko gian địa chỉ của TT

  - Trạng thái thực thi của TT: running, sleeping, stop
  - Độ ưu tiên thực thi
  - Các tài nguyên mà TT sử dụng: %CPU, RAM...
  - Chủ sở hữu TT là ai
  - Các chỉ thi đang đc thực thi bởi TT

PROCESS ATTRIBUTES (CÁC THUỘC TÍNH CỦA TT)

- Process ID – PID: mã số TT
- Parent PID – PPID: tiến trình cha của TT
- 2 loại mã số user – UID và EUID (effective UID)
  - UID: mã số user đang sử dụng TT
  - EUID (effective UID); mã số user mà nó hưởng quyền
- 2 loại mã group – GID và EGID
- Niceness: độ dễ thương và niceness ngược với độ ưu tiên. Càng dễ thương thì độ ưu tiên càng thấp. VD: dễ thương nhất thì có độ ưu tiên 1.
- Control terminal: là kênh nhập xuất để giao tiếp với TT. VD: stdin là keyboard, stdout là sreen

SIGNALS (TÍN HIỆU)

Signals dùng để điều khiển các TT

Khi muốn điều khiển TT nào đó thì gửi tới nó 1 signal. VD: Gửi signal STOP để tạm dừng TT

- Ctrl+C: gửi tín hiệu hủy đến TT
- Ctrl+Z: gửi tín hiệu tạm dừng đến TT

IMPORTANT SIGNALS

- **`KILL`**: hủy, đóng TT
- **`STOP`**: tạm dừng thực thi TT
- **`CONT`**: tiếp tục thực thi TT

Sending signals

`$`**`kill`** `-s <SIGNAL> <Mã số TT>:` cho phép gửi signal đến 1 TT

- `$kill -s STOP 1234`: tạm dừng TT có mã số là 1234
- `$kill -s CONT 1234`: tiếp tục TT có mã số 1234
- `$kill -s KILL 1234`: hủy (đóng) TT có mã số 1234

`$`**`killall`** `-s <SIGNAL> <Tên TT>:` gửi signal đến **tất cả** các TT có tên đó

- `$killall -s STOP httpd:` tất cả các TT có tên là httpd sẽ nhận tín hiệu tạm dừng

➔Để tìm mã số của TT nào đó thì dùng lệnh `$pgrep <tên TT>`

PROCESS STATES (TRẠNG THÁI TT)

1 TT tồn tại 1 trong 4 trạng thái:

- **Runnable**: đang thực thi
- **Sleeping**: đang ngủ. Khi có tài nguyên thì TT sẽ tự thức dậy
- **Zombie**: chuẩn bị chết
- **Stopped**: tạm dừng. Muốn thực thi tiếp tục thì phải được đánh thức bằng signal `CONT`

ĐỘ ƯU TIÊN

Độ ưu tiên tỷ lệ nghịch với niceness (độ dễ thương)

Độ niceness của các TT là từ -20 (độ ưu tiên cao, ko nice) đên +19 (độ ưu tiên thấp, very nice); và 0 là mặc nhiên

User có thể tăng độ niceness, nhưng ko đc giảm độ niceness (trừ root)

- $nice +5 ~/bin/longtask : đặt độ ưu tiên tăng 5 cho TT
- $sudo renice -5 8829: đặt lại độ ưu tiên – giảm 5 cho TT (sudo)

THEO DÕI TIẾN TRÌNH

Lệnh `$ps -aux`: dùng để xem thông tin của tất cả các TT có trên OS như:

- PID, UID, độ ưu tiên, control terminal
- %MEM, %CPU, trạng thái

Lệnh `$top`: dùng để xem thông tin của tất cả các TT có trên OS

- Sắp theo TT nào mà dùng nhiều tài nguyên thì đưa lên trên
- Tự động cập nhật lại sau 5 giây

CREATING PERIODIC PROCESSES (TẠO RA CÁC TT TUẦN HOÀN)

Thay vì phải thực hiện thủ công công việc lặp đi lặp lại mỗi ngày, tuần, tháng, thì nên tự động hóa

Dịch vụ cron

Cho phép thực hiện công việc theo thời gian đc lập trước.

Dịch vụ **cron** tự động thức dậy mỗi phút 1 lần và kiểm tra có cần phải thực hiện gì ko, nếu ko thì nó sleep.

Quản lý crontabs

`$crontab -e` : lập lịch công việc

`$crontab -l` : xem các công việc đã thiết lập lịch

Crontab file


```
<minute> <hour> <dayOfMonth> <month> <dayOfWeek(0-7)>
<user thực thi> <lệnh>
```

- dayOfWeek: 0 và 7 là chủ nhật ; 1 là thứ 2 ; 6 là thứ 7
- user thực thi: nếu ko nhập vào thì user mặc nhiên sẽ thực thi
- Nếu ở phần nào có dấu * thì là lúc nào cũng đc
- 0 0 1 1 * ltnd (echo “HPNY”) → Vào lúc 0 giờ 0 phút ngày 1 tháng 1 thì user ltnd in ra “HPNY”
- 30 2 * * 1 (echo “Hello”) → Vào lúc 2 giờ 30 phút mỗi thứ 2 thì in ra “Hello”
- * * * * * <lệnh> → mỗi phút thực hiện lệnh

➔Nên tận dụng dịch vụ **cron** để tự động hóa công việc thay vì làm thủ công

MONITORING

SERVICE MONITORING (THEO DÕI HỆ THỐNG)

Motivation for monitoring (động lực theo dõi HT → theo dõi HT để làm gì?)

- Nhanh chóng phát hiện và xử lý sự cố
- Xác định đc nguồn gốc, nguyên nhân sự cố
- Dự đoán và tránh đc các sự cố trong tương lai
- Thu thập dữ liệu để phục vụ cho công việc QTHT

Historical data (Thu thập dl lịch sử)

- Thu thập dl lịch sử để cải thiện trong future

Real-time monitoring (Theo dõi thời gian thực)

Tại thời điểm này, hệ thống đang hoạt động ntn?

- Cảnh báo cho ng QTHT ngay lập tức về sự cố

Có 2 thành phần:

- Theo dõi hệ thống: kiểm tra trạng thái, đọc thông báo lỗi, kiểm tra hệ thống con
- Cảnh báo: nhận diện sự cố và thông báo cho ng QTHT

Monitoring

Theo dõi ở 2 mức độ:

- Theo dõi hệ thống còn hoạt động ko: server/mạng/lỗi ứng dụng
- Theo dõi khả năng: CPU, băng thông dùng bao nhiêu %

Alerting (Cảnh báo)

Việc theo dõi sẽ vô ích nếu ko có hệ thống cảnh báo

Active monitoring (theo dõi chủ động)

- Đừng chỉ theo dõi và cảnh bảo, hãy làm gì đó để xử lý sự cố nhanh
- Giải pháp tạm thời nhưng vẫn cần sửa lâu dài

Application response time monitoring (thời gian phản hồi các ứng dụng)

➔TÓM LẠI:

2 dạng theo dõi hệ thống:

Thu thập dl lịch sử

- Lên kế hoạch để hệ thống hđ tốt hơn
- Cải tiến

Theo dõi và cảnh báo thời gian thực

- Phát hiện sự cố nhanh nhất có thể

SYSLOG AND LOG FILES (TẬP TIN NHẬT KÝ)

Logging policies (chính sách ghi nhật ký system)

Log file để ghi lại những sự kiện xảy ra trên system

Nếu log files quá nhiều thì nên làm gì?

- Rotate (cuộn) log files, giữ data trong thời gian cố định
- Nén log file và lưu trữ ra bên ngoài
- Xóa ngay lập tức (ko nên)
- Xóa định kỳ (ko nên)

Lê Trương Ngọc Duyên – B2105569

Linux log files

Hầu hết các log files nằm trong thư mục `/var/log`

Các log files:

`/var/log/wtmp`

- Ghi lại những lần user đăng nhập và đăng xuất
- Là dạng nhị phân → Dùng lệnh **`last`** để đọc

`/var/log/btmp`

- Ghi lại những lần user đăng nhập ko đc, nhập sai passwd

`/var/log/secure`

- Ghi lại những sự kiện liên quan đến an ninh hệ thống (tạo user mới, cấu hình...) → cực kỳ quan trọng

BACKUPS

BACKUP AND RESTORE (SAO LƯU VÀ KHÔI PHỤC)

Tại sao cần phải phục hồi (restore) dl từ các sao lưu (backup) trước?

- Data bị mất
- Thiết bị hư (ổ cứng hư)
- Con người xóa data do vô tình or cố tình
- Data bị hư hại do lỗi của user, do tia gamma...

→Cần hệ thống sao lưu đáng tin cậy

3 lý do cần khôi phục

- Tập tin bị xóa do sự cố (phổ biến nhất)
- Ổ cứng bị hư: Nếu xài công nghệ RAID thì có thể giảm thiểu thiệt hại
- Lưu trữ ra ngoài, nào cần sẽ khôi phục lại

Có 2 dạng sao lưu phổ biến:

- Full backup (level 0): sao lưu tất cả file trên 1 phân vùng ổ cứng
- Incremental backup (level 1): chỉ sao lưu những file thay đổi kể từ lần cuối full backup

Backup policies (quy định sao lưu)

- Giải thích why cần sao lưu
- Sao lưu những gì
- Khi nào sao lưu

➔Việc sao lưu nên tự động hóa/tập trung hóa

Phương tiện sao lưu

- Sử dụng ổ đĩa rời
- Sử dụng các băng từ (tapes): rẻ nhất

Các chương trình lưu trữ

Lệnh `$tar` dùng để đóng gói nhiều tập tin (giống zip trên Windows)

Lệnh **`$dd`** (data dump) dùng để sao lưu, sao chép toàn bộ phân vùng ổ cứng

NETWORK ARCHITECTURE

QUẢN TRỊ VÀ CẤU HÌNH MẠNG

NETWORKING HARDWARE

2 chuẩn mạng:

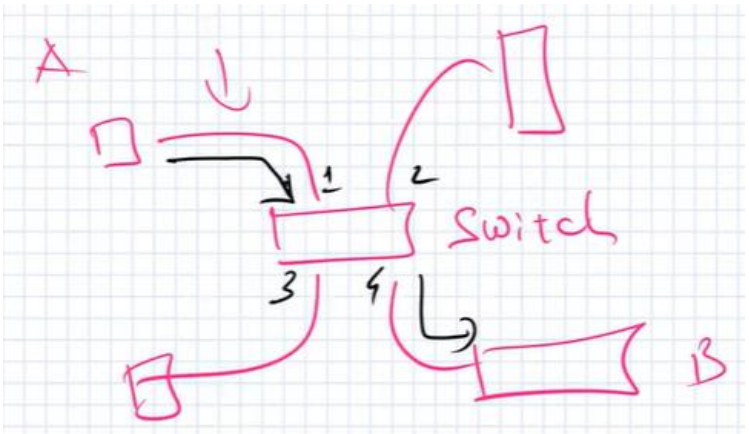
- **Ethernet**: dùng cáp (cáp xoắn đôi-UTP)
  - Tên mã của Ethernet là 862.3 đc phát triển bởi IEEE
  - Có nhiều chuẩn con:
    - Cáp đồng: số đầu là tốc độ, chữ T là cáp xoắn đôi; khoảng cách giữa các tbị là 100m→kết nối các tbị giữa các phòng, 1 tòa nhà VD:100BASE-T, 10GBASE-T...
    - Cáp quang: khoảng cách từ vài trăm mét tới vài chục cây số→ kết nối các tbị giữa. 2 tòa nhà xa nhau VD: 1000BASE-LX, 10GBASE-ER, 10GBASE-SR...
- **Wi-Fi**: mạng ko dây
  - Tên mã của Wi-Fi là 802.11 đc phát triển bởi IEEE
  - Có nhiều chuẩn con: a, b, c, n... Mỗi chuẩn có thông số khác nhau
  - Hoạt động trên 2 tầng số chính: 2.4GHz và 5GHz

Connecting ethernets (Những phần cứng kết nối ethernet)

Lê Trương Ngọc Duyên – B2105569

Switch:

- Là bộ chia mạng
- Có các cổng (port/interface) để gắn vào card mạng của máy tính bằng dây cáp (cáp đồng or cáp quang)
- Có cơ chế học đc vị trí: Mỗi máy tính ở vị trí/cổng nào. VD: A cổng 1 gửi data cho cổng 4 là B thì gửi đúng là A đến B



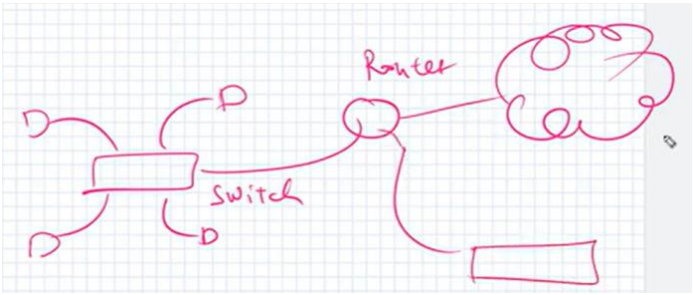
Hub

- Giao diện bên ngoài của hub giống như switch
- Ko đc thông minh như switch, ko có cơ chế học vị trí. VD: A gửi data cho B → A gửi data cho tất cả các cổng (ngoại trừ nó). C và D nhận đc data nhưng ko phải gửi cho nó thì bỏ qua; còn B là của nó thì nó nhận
  ➔Tạo ra những dữ liệu mạng ko cần thiết → Giảm hiệu năng mạng của hệ thống; Vấn đề bảo mật

➔Ko dùng Hub nữa mà dùng Switch

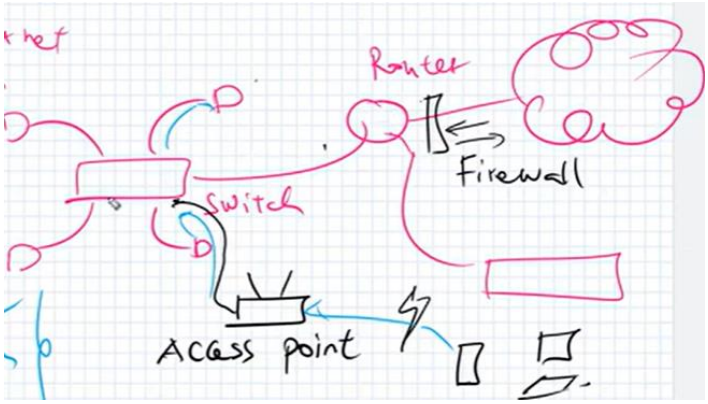
Router

- Nối kết giữa nhánh mạng này với nhánh mạng khác
- Cho phép nối kết ra Internet
- Nối kết với những nhánh mạng khác. VD: Tòa nhà này và tòa nhà khác



Access Point

Để kết nối Wi-Fi có thiết bị gọi là Access Point (có 2 râu ăng-ten). Access Point có 1 nối kết từ cái Switch→phát sóng vô tuyến



?Thiết bị kết nối Internet ở nhà chúng ta gọi là gì?

➔ Thiết bị đa chức năng: Router, Switch, Access Point

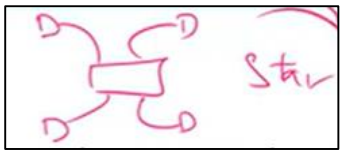
Firewall (Tường lửa)

Tường lửa chuyên dụng phần cứng để bảo vệ cho toàn bộ nhánh mạng

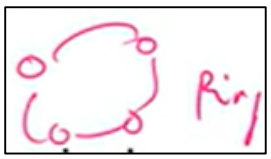
Lê Trương Ngọc Duyên – B2105569

NETWORK TOPOLOGY (HÌNH TRẠNG MẠNG)

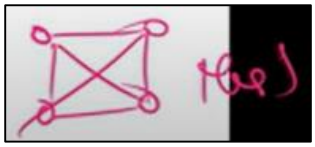
Dạng STAR: Phổ biến do dễ bảo trì, mở rộng



Dạng RING



Dạng MESH: Tất cả tbị sẽ nối kết các tbị còn lại



Dạng BUS: Dạng hình thẳng

NETWORKING

TCP/IP

Bộ giao thức mạng (networking protocol) phổ biến

Nền tảng của Internet

PROTOCOLS (GIAO THỨC)

Giao thức là cách thức giao tiếp

TCP – Transmission Control Protocol (Giao thức điều khiển truyền tin)

Thuộc tầng vận chuyển (Transport Layer)

IP – Internet Protocol

Đóng gói tập tin:

- Chia nhỏ dữ liệu thành nhiều gói tin

- Định dạng và nhận dạng các gói tin

UDP

- Livestream

TCP/IP NETWORK STACK

Chia hệ thống mạng thành 5 tầng or 4 tầng

5 tầng (cũ):

- Application Layer: sử dụng tên miền
- Transport Layer: chia nhỏ và tái hợp dữ liệu lại
- Network Layer: giao tiếp liên mạng giữa 2 máy tính khác nhau
- Datalink Layer: cho phép 2 tbị kề nhau giao tiếp với nhau
- Physical Layer: đường truyền vật lý

4 tầng (mới, gần đây)

- Application Layer
- Transport Layer
- Network Layer
- Link Layer (gom 2 tầng Datalink và Physical lại)

ADDRESSING (ĐỊNH ĐỊA CHỈ)

Có 4 loại địa chỉ:

+Địa chỉ vật lý (**MAC**): độ dài 48 bits→6 bytes

- Nằm ở tầng Physical và Datalink
- Cố định trên một thiết bị
- Các card mạng có con chip chứa địa chỉ MAC
- Đ/c MAC duy nhất, ko có trùng nhau
- VD: Physical address (MAC): 90-0F-0C-46-82-59

+Địa chỉ luận lý (**IP**)

- Nằm ở tầng Network
- Địa chỉ IP giúp định địa chỉ cho các tbị trên mạng, trên Internet; giúp phân biệt tbị này với tbị khác
- Cấu hình được, thay đổi đc phụ thuộc vào nhanh mạng mà ta nối kết vào
- IPv4: 32 bits, dạng thập phân và IPv6: 128 bits, dạng hệ thập lục phân

+Cổng (**port**)

- Nằm ở tầng Transport
- Cổng cho cho phép định địa chỉ các dịch vụ ở trên 1 tbị

- Cổng có độ dài 16 bits → Có độ dài từ 0 tới 6535.
- Từ 0→1024 cho các dịch vụ phổ biến. VD:
  - ***Web (HTTP)*** cổng 80,
  - ***FTP*** cổng 20-21,
  - ***SSH*** cổng 22,
  - ***DNS*** cổng 53
  - ***DHCP*** cổng 67
  - Mail cổng 25

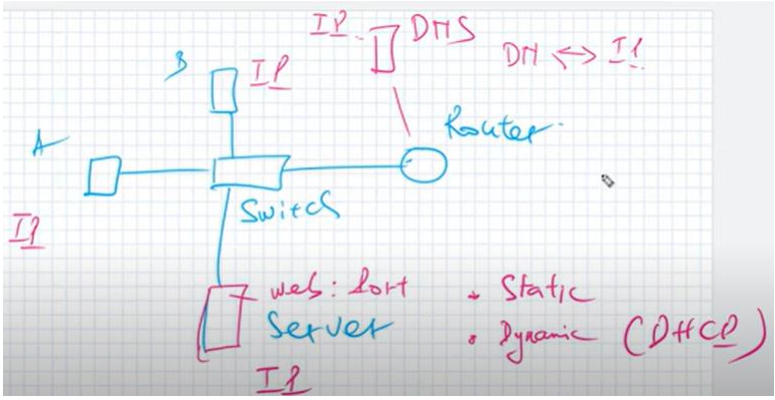
+Tên miền **(Domain name)**

- Nằm ở tầng Application
- Tên miền là giải pháp thay thế đ/c IP dạng số. Do tên miền dễ nhớ, thân thiện với user hơn
- Cần trung gian là các server chạy dịch vụ tên miền → **DNS (Domain Name Server):** chứa tên miền và IP tương ứng tên miền đó

-Địa chỉ IP

Cấu hình tĩnh: cấu hình thủ công

Cấu hình động (***DHCP***): tự động cấu hình



-Cổng (port)

Khi cài đặt dịch vụ trên server thì gán số hiệu cổng hoặc để mặc định

-DNS

Giúp chuyển đổi tên miền thành đ/c IP và ngược lại

***Địa chỉ IP và đ/c MAC** chia làm 3 loại:

- Unicast: là đ/c đại diện cho 1 tbị
- Broadcast: là đ/c đại diện cho tất cả các tbị trong 1 nhánh mạng
- Multicast: là đ/c đại diện cho 1 nhóm các tbị

ĐỊA CHỈ IP

IPv4 có độ dài 4 bytes → 32 bits. Tổng số đ/c là 2^{32}

IPv6 ra đời để giải quyết việc thiếu địa chỉ của IPv4

VD IPv4: 192.168.1.250

1 phần thập phân có độ dài từ 0 → 255

Lê Trương Ngọc Duyên – B2105569

Chia thành 5 lớp dựa vào giá trị của phần thập phân đầu tiên:

Lớp A: 1→126

- Phần thập phân đầu tiên (8) là mạng và 3 phần thập phân còn lại (24) là host ➔ Mặt nạ mạng 8 bits
- VD: 10***.0.0.0***/8
  - Giữ nguyên giá trị 8 bits đầu tiên và tùy ý thay đổi 24 bits còn lại để đặt cho các nhánh mạng
  - 10.0.0.0, 10.0.0.1, ..., 10.255.255.254, 10.255.255.255
  - ➔1 nhánh mạng ở lớp A có 2^{32} địa chỉ. Tuy nhiên có 2 địa chỉ đầu tiên (đ/c mạng: 10.0.0.0) và cuối cùng (đ/c broadcast: 10.255.255.255) ko dùng. Nên chỉ dùng đc $2^{24} - 2$ đ/c (10.0.0.1→10.255.255.254)

Lớp B: 128→191

- 2 phần thập phân đầu tiên (16) là mạng và 2 phần thập phân còn lại (16) là host ➔ Mặt nạ mạng 16 bits
- VD: 172.30***.0.0***/16
  - Giữ nguyên giá trị 16 bits đầu tiên và tùy ý thay đổi 16 bits còn lại để đặt cho các nhánh mạng
  - 172.30.0.0, 172.30.0.1, ..., 172.30.255.254, 172.30.255.255
  - ➔1 nhánh mạng ở lớp B có 2^{16} địa chỉ. Tuy nhiên có 2 địa chỉ đầu tiên (đ/c mạng: 172.30.0.0) và cuối cùng (đ/c broadcast: 172.30.255.255) ko dùng. Nên chỉ dùng đc $2^{16} - 2$ đ/c (172.30.0.1→172.30.255.254)

Lớp C: 192→223

- 3 phần thập phân đầu tiên (24) là mạng và phần thập phân cuối (8) là host ➔ Mặt nạ mạng 24 bits
- VD: 192.168.10***.0***/24
  - Giữ nguyên giá trị 24 bits đầu tiên và tùy ý thay đổi 8 bits còn lại để đặt cho các nhánh mạng
  - 192.168.10.0, 192.168.10.1, ..., 192.168.10.254, 192.168.10.255
  - ➔1 nhánh mạng ở lớp C có 2^8=256 địa chỉ. Tuy nhiên có 2 địa chỉ đầu tiên (đ/c mạng - đại diện nhánh mạng: 192.168.10.0) và cuối cùng (đ/c broadcast – đại diện cho all tbị: 192.168.10.255) ko dùng. Nên chỉ dùng đc $2^8 - 2 = 254$ đ/c (192.168.10.1→192.168.10.254)

Lớp D: 224→239. Dùng cho đ/c multicast

Lớp E: 240→255. Ít dùng, để dự phòng hoặc thí nghiệm

VD: 192.168.7.100 là lớp C; 40.100.80.250 là lớp A; 172.256.80.123 là địa chỉ sai vì đ/c từ 0→255 nên 256 là sai

Lê Trương Ngọc Duyên – B2105569

SUBNETTING (CHIA MẠNG CON)

Từ 1 nhánh mạng lớn → chia ra thành nhiều mạng con

Mục đích: tiết kiệm đ/c IP

IP PRIVATE

IP Class	From	To	CIDR Range
A	10.0.0.0	10.255.255.255	10.0.0.0/8
B	172.16.0.0	172.31.255.255	172.16.0.0/12
C	192.168.0.0	192.168.255.255	192.168.0.0/16

NAT (NETWORK ADDRESS TRANSLATION)

Dùng NAT để kết hợp giữa IP private và IP public để giải quyết việc thiếu địa chỉ IPv4

Các tbị trong mạng cục bộ thì đc đặt IP private

Còn nối kết trực tiếp ra ngoài Internet của router thì đặt IP public

Khi 1 tbị bên trong giao tiếp ra ngoài thì dịch vụ NAT trên router sẽ chuyển đổi địa chỉ: lấy đ/c IP public thay thế đ/c IP private và kết nối ra ngoài

→Giúp cho nhiều tbị bên trong mạng cục bộ có thể dùng chung 1 hay vài đ/c IP public

➔Kỹ thuật ***NAT*** giúp giải quyết việc thiếu đ/c IPv4

DNS

Nhiệm vụ: chuyển đổi tên miền thành đ/c IP và ngược lại

Phần mềm **BIND** để triển khai dịch vụ tên miền

DATA CENTER

Data center (trung tâm dữ liệu): là nơi đặt các máy chủ và các tbị CNTT khác

Nhiệm vụ: cung cấp những điều kiện để cho các máy chủ hoạt động ổn định.

- Hệ thống làm mát, điều hòa ko khí
- Hệ thống điện, nguồn điện dự phòng

- Phương tiện bảo vệ khỏi thảm họa tự nhiên như hỏa hoạn, lũ lụt
- Chế độ an ninh: người bảo vệ, cửa nẻo, camera...

VIRTIALIZATION (ẢO HÓA)

ẢO HÓA

Ảo hóa là kỹ thuật che dấu các đặc tính vật lý của các tài nguyên điện toán ra khỏi các hệ thống, ứng dụng, các user tương tác với chúng

- Cho phép tạo ra những tài nguyên điện toán ảo
- CPU ảo; RAM ảo; Ổ cứng ảo; Mạng ảo ➔ Gom lại thành máy ảo

Có 2 hướng **chức năng** ngược lại lẫn nhau:

- Kết hợp nhiều tài nguyên vật lý lại thành 1 tài nguyên ảo. VD: Có nhiều ổ cứng vật lý kết hợp thành 1 ổ cứng ảo lớn
- Từ 1 tài nguyên vật lý tạo ra nhiều tài nguyên ảo. VD: 1 CPU vật lý có thể tạo ra nhiều CPU ảo chạy trên CPU vật lý

MÁY ẢO

Máy ảo hệ thống

Từ 1 tài nguyên vật lý thành nhiều tài nguyên ảo có thể ko cùng kiến trúc, HĐH với tài nguyên vật lý

Giống như 1 máy tính thiệt, có CPU, RAM, OS... . Hoạt động độc lập

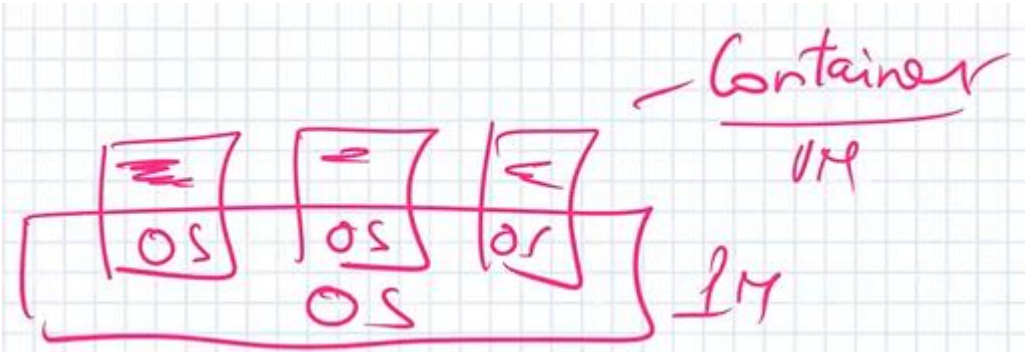
Để tạo ra những máy ảo hệ thống và quản lý chúng cần 1 phần mềm đặc biệt là Virtual Machine Monitor (VMM), hay còn đc gọi là ***Hypervision***

Virtual Box hay VM Wave là Hypervision

Container (OS virtualization)

Các máy ảo dùng chung OS với máy vật lý

Ưu điểm: hiệu năng tốt hơn



VIRTUAL APPLIANCES

Chỉ các máy ảo đc cài đặt sẵn các phần mềm cần thiết

Mục đích: Giúp loại bỏ việc cài đặt, cấu hình các phần mềm phức tạp

BENEFITS TO VIRTUALIZATION

Lợi ích về **chi phí**

- Tăng đc hiệu suất sử dụng phần cứng → Tiết kiệm chi phí

Lợi ích về **chi phí vận hành**

Lợi ích về **điều hành**

- Triển khai các phần mềm dễ dàng hơn
- Tạo môi trường cô lập giúp chạy những phần mềm ko tin cậy
- Trạng thái của máy ảo có thể lưu, dừng, restart...

OTHER TECHNOLOGIES

- Oracle: VirtualBox
- Windows: Hyper-V, VMWare
- Linux: KVM
- MAC: Parallels

NFS (NETWORK FILE SYSTEM)

NFS (Network File System): hệ thống tập tin mạng

Cho phép hệ thống chia sẻ hệ thống tập tin với các máy tính khác

Linux có dịch vụ ***SAMBA***

THỰC HÀNH QUẢN TRỊ HỆ THỐNG

LAB1: SỬ DỤNG LỆNH CƠ BẢN

Để tìm kiếm thông tin hướng dẫn về 1 lệnh or tiện ích nào đó trong Linux, dùng lệnh:

- **`$<lệnh> --help`** VD: `$ls --help`
- **`$man <lệnh>`** VD: `$man ls`

Lệnh **$pwd** (print working directory): hiển thị thư mục làm việc hiện hành

Lệnh **$cd** (change directory): chuyển đổi thư mục làm việc

- **$cd ~** : chuyển qua thư mục cá nhân của user
- **$cd ..** : chuyển ra thư mục phía trên 1 cấp của thư mục hiện hành, còn gọi là thư mục cha

Lệnh **$ls** : liệt kê nội dung thư mục hiện hành

- `$ls -l` : liệt kê nội dung thư mục với đầy đủ thông tin bao gồm quyền, size, ngày/giờ

Lệnh **$grep "<chuỗi kỳ tự>" <tập tin>** (global regular expression print): tìm chuỗi ký tự trong tập tin đc chỉ định

Lệnh **$sed 's/<từ cần thay thế>/<từ thay thế>' <tập tin>**: sửa đổi nội dung của 1 tập tin

Lệnh **$cat <tập tin>** (concatenate): hiển thị toàn bộ nội dung tập tin

Để hiển thị tập tin theo phân trang, dùng lệnh:

- **$more <tập tin> $less <tập tin>**

Lệnh **$head [-<số dòng>] <tập tin>:** hiển thị 1 số dòng (mặc định 10 dòng) ở đầu tập tin

- `$head abc.txt` : hiển thị 10 dòng đầu tiên của tập tin abc.txt
- `$head -3 abc.txt` : hiển thị 3 dòng đầu tiên của tập tin abc.txt

Lệnh **$tail [-<số dòng>] <tập tin>:** hiển thị 1 số dòng (mặc định 10 dòng) ở cuối tập tin

- `$tail abc.txt` : hiển thị 10 dòng cuối cùng của tập tin abc.txt
- `$tail -1 abc.txt` : hiển thị 1 dòng cuối cùng của tập tin abc.txt

Lệnh **$cp <tập tin or thư mục> <đích đến>** : sao chép tập tin or thư mục, cũng có thể đổi tên tập tin

- VD: `$cp abc ./Document/` : sao chép tập tin abc vào thư mục Document

Lệnh **$mv <tập tin or thư mục> <đích đến>** : di chuyển tập tin or thư mục, cũng có thể đổi tên tập tin

- VD: `$mv abc ./Desktop/abcdef`: di chuyển tập tin abc vào thư mục Desktop và đổi tên thành abcdef

Lệnh **$mkdir <thư mục>:** tạo 1 thư mục mới

Lệnh **$rm <tập tin or thư mục>** : xóa tập tin or thư mục

Các tập tin có phần mở rộng:

- **.rpm** cho Red Hat
- **.deb** cho Debian, Ubuntu

Để cài đặt or cập nhật ứng dụng, dùng lệnh:

- **$sudo yum install/update/remove <ứng dụng>**
- **$sudo dnf install/update/remove <ứng dụng> [-y]**

➔Muốn dùng lệnh này phải có quyền sudo hoặc là root

Để cập nhật tất cả các ứng dụng trong hệ thống, dùng lệnh:

- **$sudo yum update**
- **$sudo dnf update**

LAB2: QUẢN LÝ TÀI KHOẢN NGƯỜI DÙNG, Ổ CỨNG VÀ HỆ THỐNG TẬP TIN

QUẢN LÝ TÀI KHOẢN

Lệnh **$sudo adduser <tài khoản>** : tạo tài khoản ng dùng mới

Lệnh **$sudo passwd <TK>** : thay đổi mật khẩu cho tài khoản ng dùng or TK nhóm

Trong tập tin **/etc/passwd**:

```
duyen.le:x:1001:1001::/home/duyen.le:/bin/bash
```

- Tên tài khoản: duyen.le
- Mật khẩu: x → thông tin mk đc lưu trong tập tin shadow
- UID: 1001
- GID: 1001
- Thư mục cá nhân của ng dùng: /home/duyen.le
- Login shell: /bin/bash

Trong tập tin **/etc/group**

```
di21v7a2:x:1002:duyen.le
```

- Tên nhóm: di21v7a2
- Mật khẩu x → thông tin mk đc lưu trong tập tin shadow

- GID: 1002
- Danh sách các TK thuộc nhóm này: duyen.le

Trong tập tin **`/etc/shadow`**:

- Dựa vào ký tự giữa 2 dấu $ đầu tiên biết đc TK dùng giải thuật băm gì
  - `$y$`: yescrypt
  - `$6$`: SHA-512
  - `$1$`: MD5

Lệnh **`$sudo usermod -e <ngày hết hạn> <TK>`**: thiết lập ngày hết hạn cho TK theo định dạng mm/dd/yyyy

Lệnh **`$sudo chage -l <TK>`**: xem thông tin của TK

Lệnh **`$sudo groupadd <nhóm>`**: tạo nhóm mới

Lệnh **`$sudo usermod -aG <nhóm> <TK>`**: thêm TK ng dùng vào nhóm

Lệnh **`$groups <TK>`**: kiểm tra 1 TK thuộc những nhóm nào

Lệnh **`$sudo usermod -L <TK>`**: khóa TK

- Khi mở file `/etc/shadow` thì mật khẩu của TK sẽ có thêm dấu **!** ở phía đầu chuỗi

Lệnh **`$sudo usermod -U <TK>`**: mở khóa TK

QUYỀN ROOT VÀ SUDO

Cấp quyền sudo cho TK:

-B1: Dùng lệnh **`$sudo nano /etc/sudoers`** để mở tập tin sudoers. Tập tin này dùng để quản lý các quyền sudo trên hệ thống

- Cho phép all các user thuộc nhóm wheel thực thi bất kỳ lệnh nào trên hệ thống
  ➔ Thêm TK cần cấp quyền sudo vào nhóm wheel

-B2: Dùng lệnh **`$sudo usermod -aG wheel <TK cần cấp quyền sudo>`**

Thu hồi quyền sudo cho TK:

Để thu hồi quyền sudo của 1 TK, có thể đưa TK đó ra khỏi nhóm có toàn quyền

Lệnh **`$sudo gpasswd -d <TK cần xóa khỏi nhóm> <nhóm>`** : xóa TK ra khỏi nhóm

ĐĨA VÀ PHÂN VÙNG Ổ CỨNG

Lệnh **`$fdisk`** : là tiện ích quản lý phân vùng đĩa cứng

- **`$sudo fdisk -l`** : liệt kê các phân vùng ổ cứng trên hệ thống
- **`$sudo fdisk <ổ cứng cần phân vùng>`**: phân vùng ổ cứng

Lệnh **`$sudo mkfs.ext4 <tên phân vùng>`**: định dạng phân vùng theo chuẩn `ext4`

Lệnh **`$sudo mount <tên phân vùng> <đg dẫn thư mục>`**: gắn ổ cứng vào đường dẫn cụ thể

- Lệnh `$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 đc sử dụng bao nhiêu

PHÂN QUYỀN TRÊN HỆ THỐNG TẬP TIN

Lệnh **`$sudo chown [<chủ sở hữu>]:[<nhóm chủ sh>] <tập tin>`** : thay đổi chủ sở hữu và nhóm chủ sở hữu của tập tin

Lệnh **`chmod`** : thay đổi quyền truy cập của user tới tập tin or thư mục

`$sudo chmod <giá trị mode> <tập tin or thư mục>`

- r = $2^2 = 4$ w = $2^1 = 2$ x = $2^0 = 1$
- 777: rwxrwxrwx → Mọi người đều có toàn quyền
- 755: rwxr-xr-x → Chủ sở hữu toàn quyền, những người còn lại chỉ có quyền đọc và thực thi
- 750: rwxr-x— → Chủ sở hữu toàn quyền, nhóm người dùng có quyền đọc và thực thi, những người khác không có bất kỳ quyền gì cả
- 644: rw-r--r– → Chủ sở hữu có quyền đọc viết, những người còn lại chỉ có quyền đọc

Lệnh **`$sudo chmod <u/g/o><+/- ><r/w/x> <tập tin or thư mục>`**

- `$sudo chmod u+x abc.txt` : thêm quyền execute cho chủ sở hữu
- `$sudo chmod g+r abc.txt` : thêm quyền read cho nhóm chủ sở hữu
- `$sudo chmod o-w abc.txt` : xóa quyền write của những người khác

Lệnh `$touch <tập tin>`: tạo tập tin rỗng

Kỹ thuật ACL (Access Control List)

→Dùng để phân quyền cho nhiều nhóm ng dùng, nhiều ng dùng khác nhau trên 1 thư mục, tập tin

Lệnh **$getfacl <tập tin/thư mục>** : xem các quyền truy cập đầy đủ trên tập tin, thư mục

Lệnh `$setfacl` : thay đổi quyền truy cập trên tập tin, thư mục

- –m (modify): thay đổi quyền của tập tin, thư mục

Cú pháp: **$sudo setfacl –m <u/g/o>:<ID/Tên>:<quyền> <tập tin/thư mục>**

- `$sudo setfacl –m u:duyenle:rwx /abc`
- `$sudo setfacl –m g:sinhvien:r-x /abc`
- `$sudo setfacl –m o:--- /abc`

LAB3: USE SHELL SCRIPTING, QUẢN LÝ TIẾN TRÌNH, TẬP TIN NHẬT KÝ HỆ THỐNG

SHELL SCRIPTING

Lệnh $hostname : hiển thị host name hiện tại của hệ thống

Lệnh $hostname –I : hiển thị địa chỉ IP của máy chủ

Lệnh $id : hiển thị thông tin user (gồm UID, tên đăng nhập của user, nhóm mặc nhiên của user, các nhóm mà user thuộc vào)

Lệnh $uname –a : hiển thị thông tin về phiên bản Linux mà ta sử dụng trong HĐH CentOS9

Lệnh $ps –eo pid,%mem,%cpu,comm --sort –rss | head –n 3 : chỉ in ra 4 hàng đầu thông tin các tiến trình (PID, % bộ nhớ, % CPU, lệnh thực thi tiến trình) có trên HĐH, sắp xếp theo phần trăm bộ nhớ theo thứ tự giảm dần

Lệnh $nano <tên script>.sh : mở trình soạn thảo nano lên để viết shel script

Lệnh $bash <tên script>.sh : chạy đoạn script bằng môi trường bash shell

Muốn chạy đoạn script trực tiếp mà ko cần lệnh bash thì phải cấp quyền thực thi (execute) cho đoạn script đó

- $sudo chmod +x <tên script>.sh

➔$./<tên script>.sh

Lệnh **$tar –cf <file lưu dl nén>.tar <nội dung cần nén>** : nén toàn bộ tập tin or thư mục vào đích đến

- Nội dung cần nén có thể có nhiều nội dung, phân tách nhau bằng dấu cách
- VD: `$tar -cf /def.tar /abc /abc1` : nén thư mục abc và abc1 thành thư mục /def.tar

LÊN LỊCH CÔNG VIỆC ĐỊNH KỲ VỚI CRON

Lệnh **`$export EDITOR=nano`** : hiệu chỉnh file crontab với trình soạn thảo nano

Lệnh **`$crontab -e`** : Lên lịch công việc

- `<phút> <giờ> <ngày trong tháng> <tháng> <ngày trong tuần> [<user thực thi>] <lệnh/công việc>`

Lệnh **`$crontab -l`** : Xem lại các công việc đã lên lịch

QUẢN LÝ TIẾN TRÌNH

Lệnh **`$ps -aux | grep "<user>"`** : tìm tất cả các tiến trình đc thực thi bởi user

Lệnh **`$pgrep <tiến trình>`** : tìm PID của tiến trình

Lệnh **`$renice <độ ưu tiên giảm> <PID>`** : Giảm độ ưu tiên của tiến trình với PID

Lệnh **`$kill -s <STOP/CONT/KILL> <PID>`**: tạm dừng/tiếp tục (phục hồi trạng thái trc đó)/hủy tiến trình với PID

TẬP TIN LOG

Ko thể dùng lệnh $cat /var/log/wtmp để mở trực tiếp file wtmp do nó có dạng nhị phân

➔Lệnh **`$last | head -n 5`** : hiển thị thông tin về user, time của 5 lần đăng nhập sau cùng vào hệ thống

Ko thể dùng lệnh $cat /var/log/btmp để mở trực tiếp file wtmp do nó có dạng nhị phân

➔ Lệnh **`$sudo cat /var/log/secure | grep "authentication failure"`** : hiển thị thông tin các lần đăng nhập ko thành công vào hệ thống

Lệnh **`$sudo cat /var/log/secure`**

- | grep "adduser" : tìm thời gian user đc tạo ra

- | grep “Installed” : tìm thông tin tên và thời gian của phần mềm đc cài đặt vào hệ thống gần đây

LAB4: CẤU HÌNH MẠNG VÀ CÀI ĐẶT SSH, FTP, WEB SERVER