



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN - ĐHQG-HCM  
**KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**

# LINUX ADMINISTRATION

**QUẢN TRỊ MẠNG VÀ HỆ THỐNG**  
Networks and Systems Administration

---

MSc. Trần Thị Dung



- Linux Overview
- Linux OS installation
- User/group management
- Network management commands/utilities
- Network services

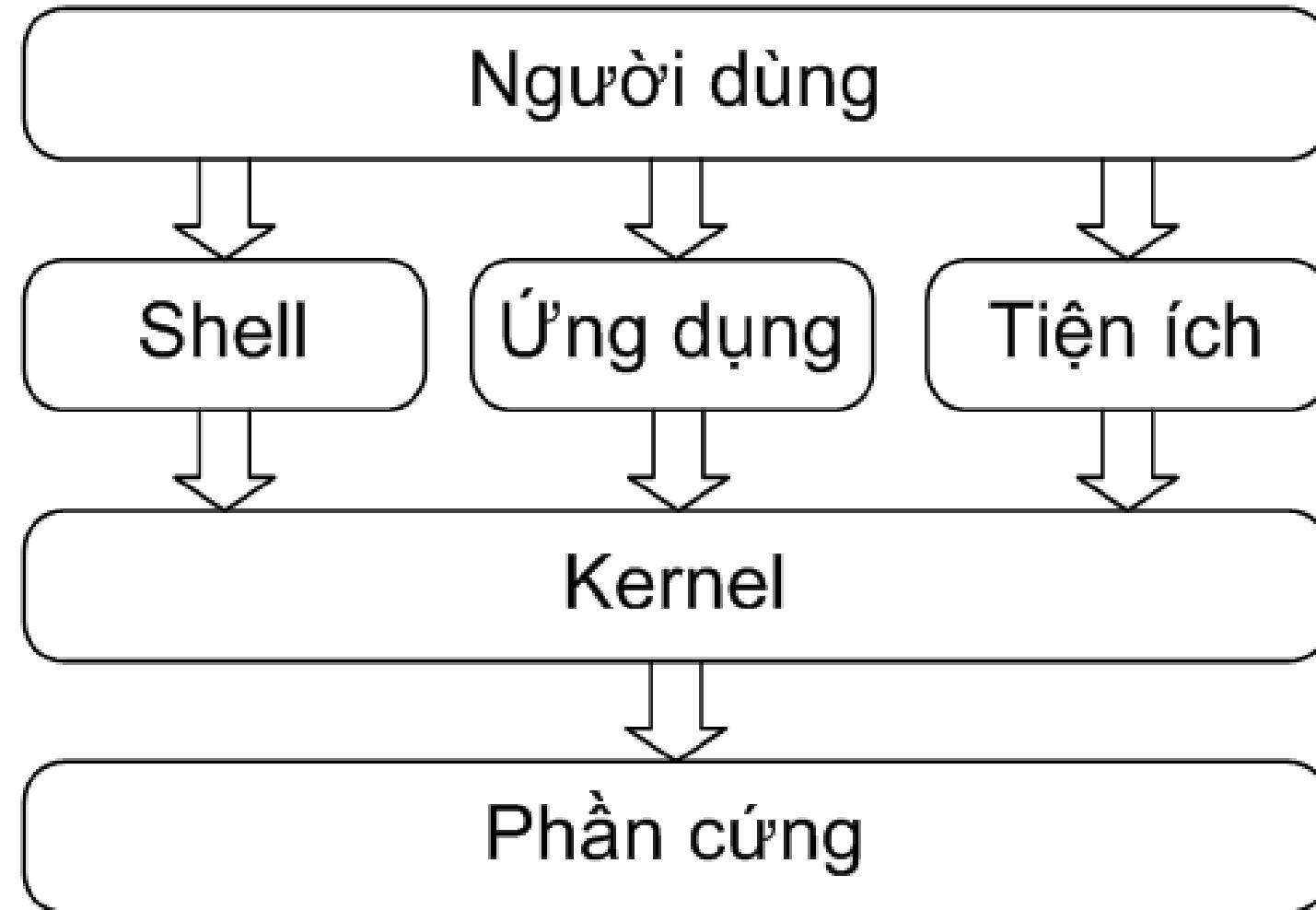
- Linux Overview
- Linux OS installation
- User/group management
- Network management commands/utilities
- Network services

# Linux Overview

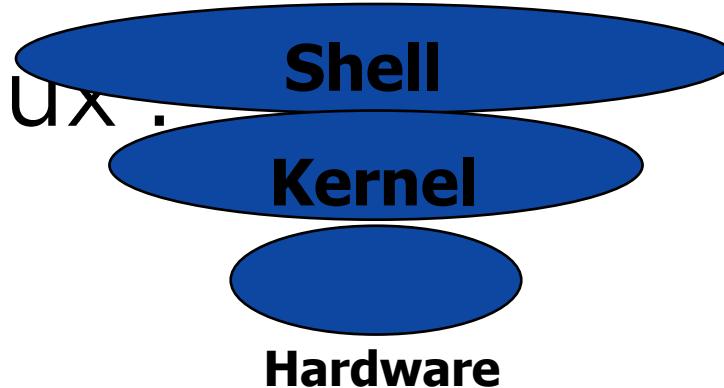
- Linux is a combination of software called **GNU/Linux**, which defines the *operating system*.
  - GNU is the *free software* that provides open source equivalents of many common UNIX commands.
  - The Linux part of this combination is the *Linux kernel*, which is the core of the operating system.
- The story of Linux begins with UNIX, an operating system developed at AT&T Bell Labs in the 1970s.

## Linux is Open Source

- Historically, most software has been issued under a closed-source license.
- This means that you get the right to use the executable program or machine code, but cannot see the source code.
- The development of Linux closely parallels the rise of *open source software*.
- The open source philosophy is that you have a right to obtain the software source code and to modify it for your own use.



- Provide the command line for user to working with the kernel.
- Types of shell in Linux .
  - C Shell (%)
  - Bourne Shell (\$)
  - Korn Shell (\$)
  - ...



○ Kernel is the center of Linux, contain the source code to control the system.

- Is the bridge of application and hardware
- Using swap space to store the processed data of the program.



**Kernel**

**Hardware**

# Linux distribution



- Linux Overview
- Linux OS installation
- User/group management
- Network management commands/utilities
- Network services

CentOS 7

Install CentOS 7

Test this media & install CentOS 7

Troubleshooting



Press Tab for full configuration options on menu items.

Automatic boot in 58 seconds...

- Step 1 : Start the OS.
- Step 2 : BIOS finds a Boot Device
- Step 3 : Control passes to the MBR of the first bootable device detected
- Step 4 : MBR points to boot record of selected operating system.
- Step 5 : Display the Operating Systems Kernel.
- Step 6 : Identify the operation level.
- Step 7 : Execute the script for the operation level
- Step 8 : Display the login requirement



- Structure

**Command [options] [arguments]**

- Command : command name
- Options : are used to modify the core behavior of a command
- Arguments: are used to provide additional information

**Note:**

- Control statements allow you to use multiple commands at once or run additional commands. (“;”, “&&”, “||”)
- Can combine the different options in one command.

- Ví dụ : [root@server01 home]# ls -a -l /etc

# Terminal key

- **^C** cancel the command
- **^D** end-of-file
- **^\** quit the executed command
- **^S** stop display on the screen
- **^H** delete 1 character
- **^W** delete 1 word
- **Arrow** moving inside a command

# Basic command line

Command	
<b>Date</b>	Display date
<b>Who</b>	Display the current user
<b>pwd</b>	Display the current folder
<b>Cal</b>	Display the calendar
<b>ps</b>	Display the process
<b>ls</b>	Display the file/folder
<b>Head</b>	Display the content from beginning
<b>Tail</b>	Display the content from the end
<b>Hostname</b>	Display or change computer name
<b>Passwd</b>	Change user password
<b>Su</b>	Switch to the other user

- **man** manual of the command
- **info** manual in Info format
- Using **man**
  - **\$ man command**
  - **\$ man -k keyword**
- Key in man page:
  - **spacebar** next page
  - **q** quit
  - **/keyword** find in man page

## ○ Login requirement

- Login: <username>
- Password: <password>
- After login:  
`[username@computername folder] #`
- Example: `[root@server01 home] #`

## ○ User logout: exit or logout.

# Shutdown and Reboot

- Shutdown :

- Init 0
- Shutdown -**hy** **t** (shutdown after t minutes)
- Halt
- Poweroff

- Reboot

- Init 6
- Reboot
- Shutdown -**ry** **t** (reboot after t minutes)

- Linux Overview
- Linux OS installation
- **User/group management**
- Network management commands/utilities
- Network services

- User account :
  - Each user has a unique name and id (UID).
  - Each user belong to at least 1 (primary group).
- Group:
  - Each group has a unique name and id (GID).
  - Each group contain 1 or many members.
- Note:
  - Username and group name are unique.
  - User ID (UID) and Group ID (GID) can be the same

- Home directory:
  - Each user has a home directory that have the same name with user and inside the folder `/home/`
  - This directory contains the private data of user.
- Environment information of user
  - `/etc/skel/`
  - Thư mục `/etc/skel/` chứa các tập tin và thư

- Some commands require administrative or root privileges.
- Using root has risks, it is recommended to use **sudo** or **su** command to execute commands as root.
- Risks with logging in as root:
  - Everything will run as root (background processes, executables)
  - May forget you are logged in as root
  - May accidentally run non-admin tasks as root

# Tạo tài khoản người dùng - useradd

- Cú pháp :

**useradd [options] ... username**

- Một số tùy chọn :

- **-c** Mô tả thông tin tài khoản người dùng.
- **-m** Tạo thư mục chủ nếu nó chưa tồn tại.
- **-u uid** User ID.
- **-G group[...]** Danh sách nhóm
- **-d home\_dir** Tạo thư mục chủ home\_dir.
- **-g initial\_group** Tên nhóm hoặc GID.

- Ví dụ :

```
# useradd -g studs -c "Student 01"  
stud01
```

# Creating a User

- Once you know what values to use, you can create a user by using the useradd command:

```
root@localhost:~# useradd -u 1000 -g users -G wheel,research -c 'Jane Doe' jane
```

- Information about new user jane is automatically added to /etc/passwd and /etc/shadow while information about supplemental groups is added to /etc/group and /etc/gshadow.
- A /var/spool/mail/jane file would also be created as well as the /home/jane directory.

# Setting a User Password

- There are a few ways a password can be set up:
  - User can execute passwd command
  - Admin can execute passwd command with username as an argument
  - Graphical tools
- Users must follow guidelines for entering passwords while root can bypass warnings:

```
root@localhost:~# passwd jane  
Enter new UNIX password:  
BAD PASSWORD: it is WAY to short  
BAD PASSWORD: is too simple  
Retype new UNIX password:
```

# Modifying a User

- Use `who`, `w`, and `last` commands to verify if user is currently logged in.
- The `usermod` command offers multiple options for modifying users.
- Examples of `usermod` options include:

Short Option	Long Option	Description
<code>-c</code>	<code>--comment COMMENT</code>	Sets the value of the GECOS or comment field to COMMENT.
<code>-d HOME_DIR</code>	<code>--home HOME_DIR</code>	Sets HOME_DIR as a new home directory for the user.
<code>-e EXPIRE_DATE</code>	<code>--expiredate EXPIRE_DATE</code>	Set account expiration date to EXPIRE_DATE.

## Deleting a User

- When you delete a user account, you also need to decide whether to delete the user's home directory.
- To delete a user without deleting the user's home directory execute:

```
root@localhost:~# userdel jane
```

- To delete a user and their home directory execute:

```
root@localhost:~# userdel -r jane
```

- Groups provide a way for users to share files.
- When creating a group, the grep command can be used to verify configuration and changes.
- The getent command can be used to show local and network-based groups.

```
root@localhost:~# grep root /etc/group
root:x:0:

root@localhost:~# getent group root
root:x:0:
```

# Creating a Group

- Execute the **groupadd** command by root user to create a new group.
- To specify a group ID (GID) for the group, use the -g option.

- If -g option is not used, **groupadd** will automatically assign a GID one value higher than the last added in /etc/group file.

```
root@localhost:~# groupadd -g 506 research
root@localhost:~# grep research /etc/group
research:x:506:
root@localhost:~# groupadd development
root@localhost:~# grep development /etc/group
development:x:507:
```

- Remember: In some distributions, when a user ID is created a UPG is also created.
- Avoid creating GIDs in the same range as UIDs created in the future.
- GIDs under 1000 are reserved for system use. To bypass this and assign a <1000 GID, use the -r option.

- A portable group name functions correctly with other systems.
- Guidelines for creating a portable group name:
  - For first character, use an underscore \_ character or lowercase alphanumeric a-z character
  - After first character, possible characters can be alphanumeric, dash, or underscore
  - Using more than 16 characters can be problematic
  - Last character should not be a hyphen -

# Modifying a Group

- The **groupmod** command can be used to either change the name of the group (with the **-n** option) or change the GID (with the **-g** option).

```
root@localhost:~# groupmod -n clerks sales
```

```
root@localhost:~# groupmod -g 10003 clerks
```

- Changing the group name won't cause any problems with accessing files
- Changing the GID will cause files to no longer be associated with that group.
- To search for orphaned files, use the find command with the **-nogroup** option.

## Deleting a Group

- The `groupdel` command can be used to delete a group.

```
root@localhost:~# groupdel clerks
```

- Files in the deleted group will become orphaned.
- Only supplementary groups can be deleted.

# User/Group file

- /etc/passwd
- /etc/shadow
- /etc/group
- /etc/login.defs
- /etc/default/useradd

- Linux Overview
- Linux OS installation
- User/group management
- Network management commands/utilities
- Network services

- Two important questions to consider when configuring network devices:
  - Wired or wireless?
    - Wireless includes additional security features
  - DHCP or static?
    - DHCP will provide an IP address and subnet mask (a number used to identify what subnetwork an IP address belongs to).
    - Static means to manually provide network information to the host.
    - Wireless uses DHCP

# Using Configuration Files to Configure the Network

- There will be times when no GUI-based tool will be available. In those cases, it is helpful to know the configuration files that are used to store and modify network data.
- Primary IPv4 Configuration File: /etc/sysconfig/network-scripts/ifcfg-eth0
  - To configure as DHCP, change BOOTPROTO value to dhcp.

```
root@localhost:~# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE="eth0"
BOOTPROTO=none
NM_CONTROLLED="yes"
Output Omitted...
```

# Using Configuration Files to Configure the Network

- Primary IPv6 Configuration File:  
**/etc/sysconfig/network-scripts/ifcfg-eth0**
  - Same file as IPv4 on CentOS
  - To configure IPv6 on your system the following would need to be added to the file:

```
IPV6INIT=yes  
IPV6ADDR=<IPv6 IP Address>  
IPV6_DEFAULTGW=<IPv6 IP Gateway Address>
```

# Domain Name Service (DNS)

- In order for the computer to associate an IP address with the URL or hostname request, the computer relies upon the DNS service of another computer.
- The address of the DNS server is stored in the /etc/resolv.conf file.
- For example this server is associated with the IP address 192.168.1.2 by the DNS server:

```
sysadmin@localhost:~$ host example.com
example.com has address 192.168.1.2
sysadmin@localhost:~$
```

# Domain Name Service (DNS)

- Name resolution on a Linux host is accomplished by 3 critical files:
  - /etc/hosts - Contains a table of hostnames to IP addresses
  - /etc/resolv.conf - Contains the IP addresses of the name servers the system uses to resolve names to IP addresses
  - /etc/nsswitch.conf - Used to modify where hostname lookups occur
- Commands or programs on the system, (i.e., browser) will request a connection with a remote computer by DNS name.
- The system will consult various files in a particular order to attempt to resolve that name into a usable IP address.

- There are several commands that you can use to view network information and troubleshoot network issues:
  - ifconfig
  - ip
  - route
  - ping
  - netstat
  - ss
  - dig
  - host
  - ssh

# The ifconfig Command

- The `ifconfig` command stands for "interface configuration".
- Used to display network configuration information:
  - In the output, the IP address of the primary network device (`eth0`) is `192.168.1.2` and the device is currently active (`UP`).

```
root@localhost:~# ifconfig

eth0      Link encap:Ethernet  HWaddr b6:84:ab:e9:8f:0a
          inet  addr:192.168.1.2  Bcast:0.0.0.0  Mask:255.255.255.0
                  inet6 addr: fe80::b484:abff:fee9:8f0a/64 Scope:Link
                      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
                      RX packets:95 errors:0 dropped:4 overruns:0 frame:0
                      TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1000
                      RX bytes:25306 (25.3 KB)  TX bytes:690 (690.0 B)
```

## The ip Command

- The ip command is replacing the ifconfig command.
- The ip command has increased functionality and set of options.
- The form `ip [OPTIONS] OBJECT COMMAND` is as follows:
- Both (ip and ifconfig) show the type of interface, protocols, hardware and IP addresses, network masks and other various information about each of the active interfaces on the system.

# The ifconfig v.s. ip Commands

```
root@localhost:~# ifconfig
eth0      Link encap:Ethernet HWaddr 00:0c:29:71:f0:bb
          inet addr:172.16.241.140 Bcast:172.16.241.255 Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe71:f0bb/64 Scope:Link
                    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          errors:0 dropped:0 overruns:0 frame:0
                    TX packets:1201 errors:0 dropped:0 overruns:0 carrier:0
                    collisions:0 txqueuelen:1000
                    RX bytes:8933700 (8.9 MB)  TX bytes:117237 (117.2 KB)
            37 (117.2 KB)
```

```
root@localhost:~# ip addr show
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:71:f0:bb brd ff:ff:ff:ff:ff:ff
    inet 172.16.241.140/24 brd 172.16.241.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe71:f0bb/64 scope link
        valid_lft forever preferred_lft forever
```

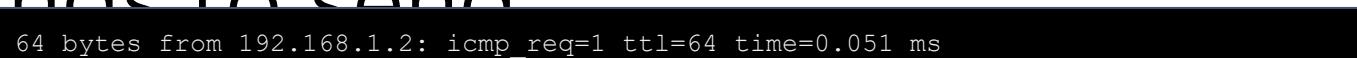
# The route Command

- The route command can be used to view a table that describes where network packages are sent.

```
root@localhost:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref Use Iface
192.168.1.0     *              255.255.255.0   U      0      0      0 eth0
default         192.168.1.1    0.0.0.0        UG     0      0      0 eth0
```

- Any network package sent to a machine in the 192.168.1 network is not sent to a gateway machine (the \* indicates "no gateway").
- All other network packets are sent to the host with the IP address of 192.168.1.1 (the router).

# The ping Command

- The ping command can be used to determine if another machine is "reachable".
- By default, the ping command will continue sending packages endlessly.
- Use the -c option followed by a number to limit how many pings to send.  


```
64 bytes from 192.168.1.2: icmp_req=1 ttl=64 time=0.051 ms
```
- A successful ping looks like:
- If the   

```
From 192.168.1.2 icmp_seq=1 Destination Host Unreachable
```

a message stating, Destination Host Unreachable will display:

# The netstat Command

- The netstat command is used to display information about network connections as well as display the routing table similar to the route command.

```
root@localhost:~# netstat -r

Kernel IP routing table

Destination     Gateway         Genmask        Flags   MSS Window irtt Iface
192.168.1.0     *           255.255.255.0   U        0 0          0 eth0
default         192.168.1.1   0.0.0.0       UG       0 0          0 eth0
```

- The netstat command is also commonly used to display open ports:

```
root@localhost:~# netstat -tln

Active Internet connections (only servers)

Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 192.168.1.2:53          0.0.0.0:*
                                         LISTEN
tcp      0      0 127.0.0.1:53           0.0.0.0:*
                                         LISTEN
```

## The dig Command

- The dig command will perform queries on the DNS server to determine if the information needed is available on the server.
- For example, the dig command is used to determine the IP address of the example.com host:

```
root@localhost:~# dig example.com  
;  
; <>> DiG 9.8.1-P1 <>> example.com  
Output omitted...  
  
example.com.      86400    IN      A      192.168.1.2  
Output omitted...
```

- The DNS server has the IP address (192.168.1.2) to hostname (example.com) translation information in its database.

# The host Command

- The host command works with DNS to associate a hostname with an IP address:

```
root@localhost:~# host example.com
example.com has address 192.168.1.2
```

- The host command can also be used in reverse if an IP address is known, but the domain name is not:

```
root@localhost:~# host 192.168.1.2
2.1.168.192.in-addr.arpa domain name pointer example.com.
2.1.168.192.in-addr.arpa domain name pointer cserver.example.com.
```

- Other options exist to query the various aspects of a DNS such as CNAME (canonical name) and SOA (Start of Authority).

# The ssh Command

- The ssh command will allow you to connect to another machine across the network, log in and then perform tasks on the remote machine:

```
root@localhost:~# ssh bob@test

The authenticity of host 'test (127.0.0.1)' can't be established.
RSA key fingerprint is c2:0d:ff:27:4c:f8:69:a9:c6:3e:13:da:2f:47:e4:c9.
Are you sure you want to continue connection (yes/no)? yes
Warning: Permanently added 'test' (RSA) to the list of known hosts.
bob@test's password:

bob@test:~$
```

- To return back to the local machine, use the exit command.

## ○ RSA key fingerprint

- If you answer yes at the prompt (asking to verify the machine's identity), the RSA key fingerprint of the remote machine will be stored on your local system:

```
RSA key fingerprint is c2:0d:ff:27:4c:f8:69:a9:c6:3e:13:da:2f:47:e4:c9.  
Are you sure you want to continue connection (yes/no)? yes  
Warning: Permanently added 'test' (RSA) to the list of known hosts.
```

- When you attempt to ssh to the same machine in the future, the RSA key fingerprint provided by the remote machine is compared to the copy stored on the local machine.
- If they don't match, you will see an error message.

- Linux Overview
- Linux OS installation
- User/group management
- Network management commands/utilities
- **Network services**

# Network services

- DHCP
- DNS
- Web Server
- File service