Linux Basics

2.1. Introduction to Linux System

Linux is an open source operating system. Its kernel was first released by Linus Benedict Torvalds on October 5, 1991. It inherits the network-centric design concept of Unix and is a stable multi-user network operating system.

In March 1994, Linux 1.0 was released with 170,000 lines of code. It was released under a completely free and free agreement at the time, and then officially adopted the GPL agreement.

In January 1995, Bob Young founded RedHat, which took GNU and Linux as the core and integrated more than 400 open source program modules to create a branded Linux, namely RedHat Linux, called Linux distribution, which was sold on the market.

In June 1996, the Linux 2.0 kernel was released, which can support multiple processors.

- Main features of Linux

Free and open source; fully compatible with POSIX 1.0 standard; multi-user, multi-tasking; has a good interface; supports multiple platforms.

- Linux major versions

There are about 300 Linux distributions, almost all of which can be run as server systems. Linux distributions rarely copy each other. The popular Linux server distributions are mainly the following:

Red Hat Enterprise Linux: This is the first Linux distribution for the commercial market. It has a server version and supports many processor architectures.

Debian: Debian is extremely stable, which makes it very suitable for servers.

CentOS: CentOS is an enterprise-level Linux distribution that is rebuilt using free source code from Red Hat Enterprise Linux. This rebuilt version completely removes registered trademarks and a very subtle change in Binary packages.

Ubuntu: Ubuntu is a derivative of Debian that focuses on its application in this market. It is common on servers, cloud computing, and even some mobile devices running Ubuntu Linux.

2.2, Ubuntu Overview

Ubuntu is a Linux operating system that focuses on desktop applications. Ubuntu is based on the Debian distribution and the Gnome desktop environment. Since version 11.04, the Ubuntu distribution has abandoned the Gnome desktop environment and changed to Unity. Since Ubuntu 18.04 LTS, Ubuntu distribution has started to use GNOME3 desktop environment again. In almost every Linux-related list, Ubuntu based on Debian has a place. Canonical's Ubuntu outperforms all other Linux server distributions - from simple installation, excellent hardware discovery, to world-class commercial support, Ubuntu has established unattainable and strict standards

2.3, Ubuntu file system

Unlike Windows, Ubuntu does not have the concept of drive letters. There is only one root directory [/], and all files are under it

```
├── bin # bin is the abbreviation of Binary. It stores the most commonly used executable files (binary) in the system.
├── boot # Here are the Linux kernel and system startup files, including Grub and lilo launcher programs.
├── dev # dev is the abbreviation of Device. This directory stores Linux external devices, such as hard disks, partitions, keyboards, mice, USB, etc.
├── etc # This directory is used to store all configuration files and subdirectories required for system management, such as passwd, hostname, etc.
├── home # The user's home directory. In Linux, each user has his own directory, which is generally named after the user's account.
| └── yahboom # User
| ├── Desktop # Desktop
| ├── Documents # Documents
| ├── Downloads # Downloads
| ├── Music # Music
| ├── Pictures # Pictures
| ├── Public # Shared
| ├── Templates # Templates
| ├── Videos # Videos
| ...
| ...
├── lost+found # This directory is usually empty. When the system is shut down illegally, some scattered files are stored here.
├── lib # Stores shared library files, including many library files used by programs in /bin and /sbin.
├── media # CD-ROM and USB devices automatically mounted by Ubuntu system, storing temporarily read files.
├── mnt # As the mount point of the mounted file system.
├── opt # As the storage directory for optional files and programs, it is mainly used by third-party developers to easily install and uninstall their software.
```

```
├── proc # This directory is a virtual directory, which is a mapping of system memory. It stores all processes marked as files. Compared with cpuinfo, it stores data on the current working status of the CPU.
├── root # This directory is the user home directory of the system administrator, also known as the super-authority user.
├── sbin # s means Super User. It stores system management programs used by system administrators, such as system management, directory query and other key command files.
├── srv # Stores service data provided by the system.
├── sys # System device and file hierarchy, and provides detailed kernel data information to user programs.
├── usr # Stores files and directories related to system users.
| ├── bin # Standard commands for users and administrators
| ├── games # Stores small games that come with XteamLinux
| ├── include # Used to store header files, c or c++, required for developing and compiling applications under Linux
| ├── lib # Link libraries for applications and packages
| ├── local # Application directory installed by the system administrator
| ├── sbin # Stores management programs used by the root super user
| └── src # Linux open source code
| └── share # Stores shared data
| ...
├── var # Variable-length files, especially some record data, such as log files and printer files
| ├── backups
| ├── cache # Application cache directory
| ├── crash # System error message
| ├── log # Log file
| ├── mail # Email
| └── tmp # Temporary file directory
| ...
├── tmp # This directory is used to store some temporary files. All users have read and write permissions to this directory.
...
```

2.4, Common commands

2.4.1, Add

Create a new file

```
touch test.txt

Create a new folder

```python
mkdir test # Create a file
mkdir -p test/src # Create a test folder and create a src folder in the test folder
```

Copy

```
sudo cp test.txt test_copy.txt # Copy a file
```
```

2.4.2, Delete

-i	Execute interactively
-f	Force deletion, ignore non-existent files, no prompt
-r	Recursively delete the contents of the directory

```
sudo rm test.txt # Delete files | Empty folders
sudo rm -r test # Delete folders and their contents
```

2.4.3, Change

- mv Move, rename

```
sudo mv test test_new # Change the test folder to test_new
sudo mv test.txt test_new.txt # Change the test.txt file to test_new.txt
```

- chmod Modify file permissions

Permission settings

Symbol	Meaning
+	Add permissions
-	Revoke permissions
=	Set permissions

rwx

Letter permissions	Meaning
r	read means read permissions. For a directory, if there is no r permission, it means that the contents of this directory cannot be viewed through ls.
w	write means write permissions. For a directory, if there is no w permission, it means that new files cannot be created in the directory.
x	excute indicates executable permission. For a directory, if there is no x permission, it means that you cannot enter the directory through cd.

```
sudo chmod +rwx test.txt
```

Shortcut to add all permissions

```
sudo chmod 777 test.txt
```

- Change password

Set root password

```
sudo passwd root
```

Set username and password

```
sudo passwd username
```

2.4.4, check

- Check system version

```
lsb_release -a # Release version number
uname -a # Kernel version and system bit number
cat /proc/version # Kernel version and gcc version
```

- Check hardware information

```
curl cip.cc or ifconfig # Check IP address
cat /proc/cpuinfo or lscpu # cpu information
sudo dmidecode -t memory # Memory information
df -h # Check the space of all mounted file systems
which python3 # View command location
v4l2-ctl --list-formats-ext # View camera device parameters
nproc # View the number of cores
```

- View file information

```
la # Display all subdirectories and files under the specified directory,
including hidden files
ll # Display detailed information of the file in a list format
ls -h # Display file size in a humanized way
cat test.txt # View file content
tree # View file directory (need to install tree)
```

tree installation command

```
sudo apt install tree
```

- Finding Files

```
find ./ -name test.sh # Find all files or directories named test.sh in the
current directory
find ./ -name '*.sh' # Find all files or directories with the suffix .sh in the
current directory
find ./ -name "[A-Z]*" # Find all files or directories starting with uppercase
letters in the current directory
```

2.4.5, Others

- tar command

Tar usage format: tar [parameter] package file name file

```
-c # Generate archive file, create package file
-v # List the detailed process of archiving and dearchiving, and display the
progress
-f # Specify the archive file name, f must be followed by .tar file, so the
option must be placed at the end
-t # List the files contained in the archive
-x # Unzip the archive file
```

Package

```
tar -cvf xxx.tar * # All files in the current directory
tar -cvf xxx.tar *.txt # Files ending with .txt
tar -cvf xxx.tar my-file my-dir # Package the specified directory or file
```

Unpack

```
tar -xvf xxx.tar # Unpack to the current directory
tar -xvf xxx.tar -C my-dir # Unzip to the specified directory (need to create
the my-dir directory first)
```

- zip, unzip commands

Compressed file: zip [-r] target file (no extension) source file

```
zip bak * # All files in the current directory, you can also specify a file
zip -r bak * # All files in the current directory & directory recursively
```

Unzip file: unzip -d directory file after decompression compressed file

```
unzip -d ./target_dir bak.zip # Unzip to the specified directory
unzip bak.zip # Unzip to the current directory
```

- ln command

Soft link: Soft link does not occupy disk space, and the soft link becomes invalid if the source file is deleted. Commonly used, can be created for files or folders

```
ln -s source file link file
```

Hard link: Hard link can only link ordinary files, not directories. Even if the source file is deleted, the link file still exists

```
ln source file link file
```

- scp remote copy

```
scp jetson@192.168.16.66:/home/jetson/xxx.tar.gz /home/yahboom/ # Copy files from remote to local
scp /home/yahboom/xxx.png jetson@192.168.16.66:/home/jetson/ # Copy files from local to remote
scp -r jetson@192.168.16.66:/home/jetson/test /home/yahboom/ # Copy directory from remote to local -r
scp -r /home/yahboom/test jetson@192.168.16.66:/home/jetson/ # Copy directory from local to remote -r
```

- wget file download

Search for a random image address on Baidu as an example.

```
wget "https://img0.baidu.com/it/u=3911542037,2006161295&fm=224&fmt=auto&gp=0.jpg" # Download ordinary files (double quotes are required for Baidu links)
wget -O yahboom.jpg "https://img0.baidu.com/it/u=3911542037,2006161295&fm=224&fmt=auto&gp=0.jpg" # Save the file with the specified file name
```

- Others

```
nautilus . # Open the current file
cd ~ # Switch to the current user's home directory (/home/user directory)
cd . # Switch to the current directory
cd - # You can enter the last directory
cd / # Switch to the system root directory/
pwd # Display the current path
echo "HelloWorld" # Output HelloWorld information to the console
which # View command location
```

2.5, Editor

2.5.1, vim

vim is an upgraded version of vi. The most common difference is that it can display some special information of system files in multiple colors.

- Installation command

```
sudo apt install vim
```

- Three main modes

Command mode (edit mode): default mode, move cursor, cut/paste text (interface performance: file name is displayed in the lower left corner or is empty)
Insert mode (input mode): modify text (interface performance: —INSERT– is displayed in the lower left corner) In insert mode, press the ESC key to return to command mode
Last line mode (extended mode): save, exit, etc. (interface performance: —VISUAL– is displayed in the lower left corner) In last line mode, press the ESC key twice in a row to return to last line mode

- Mode switching

Command mode switches to edit mode

```
i # Insert mode to enter edit mode
a # Append mode to enter edit mode
o # Start editing at the beginning of the next line of the current line
O # Start editing at the beginning of the previous line of the current line
```

Command mode switches to last line mode

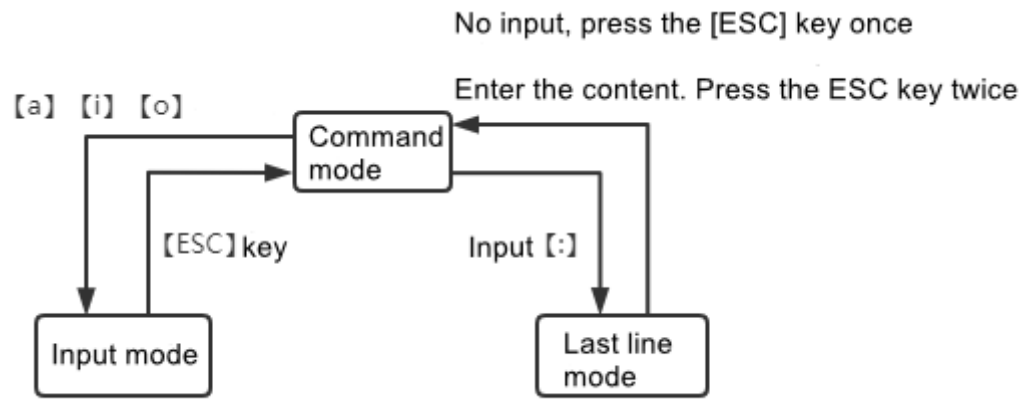
```
: # Enter a colon [:]
```

Switch from last line mode to command mode: press [esc]

Switch from edit mode to command mode: press [esc]

Esc key: exit to current mode

Esc key: always return to command mode



- Last line mode

```
w # Save
q # Exit
q! # Force exit
x # Save and exit
set nu # Display line number
set # nonu Hide line number
0,$d # vim deletes all file contents:
/string # Search for string string from the cursor backwards; press n to locate the next one, and shfit+n to locate the previous one.
g/string # Search for string. Make the cursor stop at the beginning of the first line of string found.
```

- Command mode

```
yy # copy
p # paste
3yy # copy 3 lines
2p # paste twice
dd # cut
3dd # cut 3 lines
u # undo
Ctrl + r # reverse undo
dd # delete current line
```

```
dG # delete current line to the end of the file
dH # delete current line to the beginning of the file
gg # jump to the first line of the current document
G # jump to the last line of the current document
^ # jump to the beginning of the current line
$ # jump to the end of the current line
h # move left one character
j # move down one line
k # move up one line
l # move right one character
PageDown (or Ctrl+F) # scroll down one screen
PageUp (or Ctrl+B) # scroll up one screen
```

2.5.2, nano

nano is a text editor for Unix and Unix-like systems, a copy of Pico.

- Installation

```
sudo apt install nano
```

Create/Open File

```
nano path + file name
For example: nano test_nano.txt
```

Control Commands

```
Ctrl+v # Next page
Ctrl+y # Previous page
Ctrl+w # Search for a word or phrase
Ctrl+x # Close current text, exit nano, return to shell
Ctrl+\ # Search and replace
Ctrl+k # Cut text line
Ctrl+u # Paste text line
Ctrl+c # Display cursor position in text
```

2.5.3, gedit

The usage of gedit is not much different from Notepad in Windows.

In the editor, we can click the "Open" button to browse the list of recently opened files and open the file; click the "Save" button to save the file currently being edited; click the menu bar on the right to perform more operations, etc.
The shortcut keys are the same as those in Windows:

```
Ctrl + s to save the file
Ctrl + Shift + s to save as
Ctrl + f to search for text content
```

The gedit editor must be started when the interface can be displayed. It cannot be started remotely without an interface such as ssh, jupyter, putty, etc.

2.6, Ubuntu software operation commands

```
sudo apt-get update # Update source
sudo apt-get install package # Install package
sudo apt-get remove package # Delete package
sudo apt-cache search package # Search for software package
sudo apt-cache show package # Get package information, such as description,
size, version, etc.
sudo apt-get install package --reinstall # Reinstall package
sudo apt-get -f install # Repair installation
sudo apt-get remove package --purge # Delete package, including configuration
files, etc.
sudo apt-get build-dep package # Install related compilation environment
sudo apt-get upgrade # Update installed packages
sudo apt-get dist-upgrade # Upgrade system
sudo apt-cache depends package # Understand which packages the package depends
on
sudo apt-cache rdepends package # View which packages the package depends on
sudo apt-get source package # Download the source code of the package
sudo apt-get clean && sudo apt-get autoclean # Clean up useless packages
sudo apt-get check # Check for damaged dependencies
```