UNIX Basics + shell commands

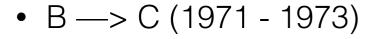
Michael Tsai 2019/02/25

Where UNIX started



Ken Thompson & Dennis Ritchie

- Multics OS project (1960s) @ Bell Labs
- UNIX on scavenged PDP-7 (1969)
- Space Travel game
- Good environment to do programming + a "fellowship" could form.





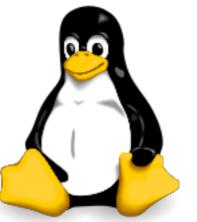
DEC PDP-7

Reading: http://www.faqs.org/docs/artu/ch02s01.html

Homework

- Read
 - "A Brief History of System Administration" from the textbook, available on NTU COOL
 - Optional: Origins and History of Unix, 1969-1995 (more details)
- Zuvio question: what would you do in the "era of Unix"?

Where LINUX



started

- Linus Torvalds: announced the Linux project (1991)
 - high cost of Sun's Unix
- Parallel: 386BSD
- Linux has Internet capability & X (1993)
- GNU toolkit



Richard Stallman & FSF

- "20 years of open-source software in different UNIX platforms"
- A "cheap UNIX system for everyone"



Linus Benedict Torvalds

Human-Machine-Interface

remote (SSH, HTTP, ...)

Hardware

Supercomputer **Computer Cluster** Mainframe computer

Distributed computing

Keyboard & Mouse

also Braille, Touch-Display, Speech recognition Graphics tablet, 3D-Mouse, Wii nunchak, etc.

Touch-Display

Attitude sensor, Motion sensor, Speech recognition

Speech recognition Attitude sensor Motion sensor

Display, Sound Vibration

Desktop Computer

Workstation **Home Computer** Desktop replacement laptop Thin client

Mobile computer

Note-/ Net-/ Smartbook Tablet Smartphone PDA / Handheld game console

Wearable Computer

Wristwatch Virtual Retina Display Head-mounted display

remote (SSH, HTTP, Serial, I2C, ...)



Embedded Computer

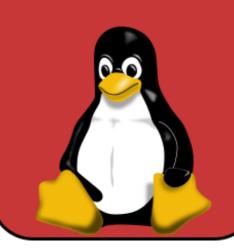
Customer-premises equipment Measurement Equipment Laboratory Equipment Layer3-Switches other embedded systems

Linux kernel

High-performance computing (HPC)

> Real-time computing (RTC)

Linux Process Scheduler **Linux Security Modules** Linux Network scheduler Network stack Netfilter Linux device drivers Linux file system drivers



Web server solution stacks (LAMP) Distributed Computing Routing daemons

an

open-source

Software Development Package management systems

CAD, CAM & CAE Software

Office

Image Processing Desktop Publishing (DTP)

Windowing Systems Desktop UI

Touch UI

Wearable UI

Video processing software 3D computer graphics Computer animation Motion graphics

Digital Audio Workstation DJ Mixing Software Video games Home cinema solutions

Debian software archives: 37,000 software packages

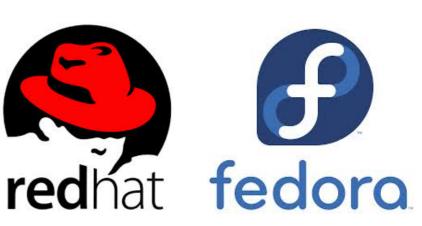
Shells)

Linux flavors















Linux族譜:

http://upload.wikimedia.org/wikipedia/commons/ 1/1b/Linux_Distribution_Timeline.svg

SSH to remote system

- https://wslab.csie.ntu.edu.tw/SSH_tutorial.html
- Make sure your terminal handles 中文 well keyword: encoding, UTF-8, or Big5
- Choose a good font. Make the font size larger.

Exercise 1: SSH login without password

- Make sure 中文 is displayed correctly.
- Answer the question on zuvio: terminal font and size.
- Disclaimer:
 do NOT copy your private key over the network!
 do this ONLY on your own computer!
- keyword: ssh-keygen

MAN: your online manual

- Your first man: man man
- Sections 1-9 of the man pages
- [] 可有可無
- | 選一個
- ... 重複
- man -k 要找的東西 —> 列出所有相關的

Choose your editor

Standard editor: Vim

For beginner: nano or joe

• (learn vim online): https://www.openvim.com

Pipes and redirection

- 0 (STDIN), 1 (STDOUT), 2 (STDERR)
- >: STDOUT 到檔案 (覆蓋)
 - >>: STDOUT 到檔案 (加到尾巴)
 - >&: STDOUT + STDERR 到檔案
 - 2>: STDERR 到檔案
 - <: 檔案餵給STDIN
 - |: 左邊程式STDOUT接到右邊程式的STDIN
- Example: echo "test message" > /tmp/blahblah
- Example: find / -name core 2> /dev/null

路徑

- 樹狀的結構
- /:根目錄,以及分隔. e.g., /home/hsinmu
- ~: 我的家目錄
- ~hsinmu: hsinmu這個帳號的家目錄

Some commands to learn

- 檔案管理: Is, cd, mkdir, rm, mv, cp, find, pwd
- 文字檔案處理: cat, less, tail
- 程序管理: ps, kill, top
- 寫shell script常用: grep, sort, wc, cut, echo, tee

File attributes

Example:

```
ls -ld dsa/
drwxr-xr-x 2 hsinmu users 4096 10月 14 2010 dsa1/
ld -l tmp
-rw-r--r-- 1 hsinmu users 12 3月 9 16:08 tmp
```

- Role: owner, group owner, and others
- 檔案: x: 執行, w: 寫入, r: 讀取
- 對目錄來說:
 - x: 可以進去, r:列出裡面的檔案,
 - w:新建、刪掉目錄中的檔案或改名

Change file permission/ ownership

- chmod: change file permission
- Examples:
 chmod u+w blah
 chmod 755 blah (7=rwx, 5=r-x)
 chmod ug=rw,o=r blah
 chmod a-x blah
 chmod g=u blah
- chown: change file ownership
- Example: chown nobody:nobody blah

More advanced permission control: access control list (ACL)

- OS & filesystem dependent
- Identify user/group and then apply the permission
- POSIX-style ACLs are supported by ext* + a few other filesystems on Linux

Possible ACL entries

Format	Example	Sets permissions for
user::perms	user::rw-	The file's owner
user:username:perms	user:htlin:rw-	A specific user
group::perms	group::r-x	The group that owns the file
group:groupname:perms	group:users:rw-	A specific group
other::perms	others::	All others
mask::perms	mask::rwx	All but owner and other

ACL examples

- getfacl: get file access control lists
 Example: getfacl tmp
- setfacl: set file access control lists
 Example:
 setfacl -m user::r,user:htlin:---,group:users:rw tmp
 setfacl -x user:htlin tmp

(-m: modify; -x remove)

Shell

- We will teach bash "the Bourne-again shell"
- Default login shell on most systems
- Check if it is your current shell: echo \$SHELL
- If not, you need to change it.
 Temp solution: bash -I
 (run the shell as if it is a login shell.
 避免環境參數問題)
- Commands are either shell built-in or a script/executable

&& 和 ||

- &&: 前面執行成功了, 後面才會執行
- ||: 前面執行失敗了, 後面才會執行
- Example 1: lpr /tmp/t2 && rm /tmp/t2
- Example 2:
 cp —preserve —recursive /etc/* /spare/backup \
 || echo "Did NOT make backup"

變數

• 給值時直接用

Example: hsinmu_dir='/nfs/home/hsinmu'

• 拿值的時候前面加\$

Example: echo \$hsinmu_dir

• {}可以指定變數名稱到哪邊

Example: echo \${hsinmu_dir}-dir

• ": 照著印出所有

"": 替換裡面該被執行的部分或變數

``: 執行裡面的指令,並將output放在這個位置

Example 1: echo 'my current work dir is `pwd`'

Example 2: echo "my current work dir is `pwd`"

Example: showusage

- Elements to learn:
 - #!/bin/bash
 - if else fi elif
 - \$# \$0 \$1 \$2: command-line argument
 - \$#: 有幾個argument,
 - \$0: 指令本身,
 - \$1,\$2,...: 第幾個參數
 - function裡面: \$# 有幾個參數, \$1, \$2, 第幾個參數

test

Example: if [\$message_leve -le \$LOG_LEVEL]; then

String	Numeric	True if
x=y	x -eq y	x is equal to y
x!=y	x -ne y	x is not equal to y
x <y< td=""><td>x -lt y</td><td>x is less than y</td></y<>	x -lt y	x is less than y
X<=y	x -le y	x is less or equal to y
X>Y	x -gt y	x is greater than y
x>=y	x -ge y	x is greater or equal to y
-n x		x is not null
-Z X		x is null

-d file	file exists and is a directory
-e file	file exists
-f file	file exists and is a regular file
-r file	you have read permission on
-s file	file exists and is not empty
-w file	you have write permission on
file1 -nt file2	file1 is newer than file2
file1 -ot file2	file1 is older than file 2

Example: str_and_number

\$((var)): 把var裡面的東西當作數學式子計算並替換

$$a=1$$

$$b=\$((2))$$

$$c = a + b$$

$$d=\$((\$a+\$b))$$

List of other things you can read

- Regular expression: very powerful tool (with grep)!
 (Hint: HW1)
 2.3
- In bash shell script:



- while and for loop
- array
- vimtutor
- File attributes: setuid, setgid, and sticky bit