Linux 系統管理 (RHCSA 10)

Department of Computer Science and Information Engineering
Chaoyang University of Technology
Taichung, Taiwan, Republic of China

Instructor: De-Yu Wang (王德譽)
E-mail: dywang@mail.cyut.edu.tw
Phone: (04)23323000 ext 4538
Office: E738



Wednesday 11th November, 2026

- Instructor: De-Yu Wang
    1. Email: dywang@csie.cyut.edu.tw
    2. Homepage: http://dywang.csie.cyut.edu.tw
    3. 本文件 pdf 檔下載
    4. Phone: (04)23323000 ext 4538
    5. Office: E738
- 參考資料
    1. RedHat Documentation
    2. RedHat Customer Protal
    3. Getting started with Tuned

Contents

I 練習環境與基本操作 **1**

1 實機練習系統 **2**

1.1 動機 2
1.2 系統設計 2
1.3 練習系統製作-VirtualBox 3
1.4 練習系統製作-Raw Image 6
1.5 NAT 設計 7
1.6 虛擬機管理界面 8
1.7 虛擬機管理界面 V2 9
1.8 系統管理與評分 10
1.9 系統註冊–上課前先確定三件事 12
1.10 虛擬機設定 13
1.11 系統伺服器管理 14

2 命令列基本操作 **17**

2.1 Linux 指令 17
2.2 線上求助 18
2.3 man page 21
2.4 基礎指令操作 24
2.5 重要的幾個熱鍵與錯誤訊息的察看 37

3 vi 編輯器 **38**

3.1 vi 與 vim 38
3.2 vi 的使用 38
3.3 vim 的額外功能 41
3.4 vi 實機練習題 43

II 系統管理 KVM8 **46**

4 IPv4 網路設定 **47**
4.1 前言 47
4.2 nmcli 命令 47
4.3 修改網卡設定 48
4.4 * 刪除重新設定 49
4.5 SSH 設定與連線 50
4.6 主機名稱設定 51
4.7 實機練習 52

5 DNF 套件管理 **54**
5.1 前言 54
5.2 dnf 套件庫設定 55
5.3 DNF 套件管理指令 56
5.4 實機練習 57

6 * 正規表示法 **59**
6.1 前言 59
6.2 基礎正規表示法 59
6.3 延伸正規表示法 66
6.4 實機練習題 67

7 資料導向與管線處理 **69**
7.1 爲何要使用命令輸出重導向？ 69
7.2 資料流重導向 69
7.3 管線命令 (pipe) 72
7.4 實機練習 72

8 SELinux **73**
8.1 前言 73
8.2 SELinux Contexts 74
8.3 啓動、關閉與觀察 75
8.4 SELinux 基本用法 77
8.5 Httpd Port 80
8.6 Httpd SELinux Context 82
8.7 實機練習 84

9 群組管理 **85**
9.1 簡介 85
9.2 增加群組 86
9.3 刪除群組 86
9.4 群組更名 86
9.5 實機練習 87

10 帳號與身份管理 **89**
10.1 簡介 89
10.2 新增帳號 90
10.3 檢查帳號 92
10.4 * 刪除帳號 92
10.5 * 變更帳號 95
10.6 設定使用者密碼 96
10.7 * 帳號家目錄問題 98
10.8 * 帳號整體參數變更 99
10.9 實機練習 101

11 su 與 sudo **102**
11.1 su 切換使用者 102
11.2 sudo 以其他身份執行命令 104
11.3 sudo 帳號測試 104
11.4 sudo 群組測試 105
11.5 實機練習 106

12 帳號密碼策略 **108**
12.1 /etc/shadow 108
12.2 預設密碼過期 109
12.3 變更預設密碼過期策略 110
12.4 設定已存在帳號密碼過期狀態 111
12.5 實機練習 112

13 Linux 檔案權限與屬性 **114**
13.1 檔案權限 114
13.2 改變檔案權限 115
13.3 檔案的預設權限 117
13.4 檔案的特殊屬性 119
13.5 設定 SGID 120

13.6 SGID 測試 121
13.7 實機練習 124

14 Access Control Lists **125**
14.1 簡介 125
14.2 ACL 練習 127
14.3 *ACL 變更 128
14.4 實機練習 129

15 例行性命令 **130**
15.1 前言 130
15.2 Crontab 命令 131
15.3 循環排程語法 131
15.4 設定排程 132
15.5 實機練習 133

16 Chronyd vs. ntpd 校時 **134**
16.1 前言 134
16.2 chronyd 使用 134
16.3 立即校時 136
16.4 *makestep 校時 137
16.5 實機練習 138

17 Network File System, NFS **139**
17.1 NFS 簡介 139
17.2 NFS Client 端工具 139
17.3 * 手動掛載 140
17.4 autofs 自動掛載 141
17.5 autofs 自動掛載測試一 142
17.6 autofs 自動掛載測試二 143
17.7 實機練習 144

18 檔案搜尋 **145**
18.1 find 命令 145
18.2 找屬於某用户的檔案 147
18.3 找特定大小的檔案 148
18.4 找特定屬性的檔案 148
18.5 實機練習 151

19 Shell Script **152**
19.1 前言 152
19.2 執行 shell script 153
19.3 可執行檔 154
19.4 * 可執行檔別名 156
19.5 實機練習 157

20 檔案打包壓縮 **159**
20.1 檔案的壓縮 159
20.2 檔案的打包 161
20.3 檔案的打包壓縮 163
20.4 查看打包壓縮檔內容 164
20.5 解開打包壓縮檔 164
20.6 實機練習 165

21 systemd 管理服務 **167**
21.1 前言 167
21.2 systemctl 命令 168
21.3 服務啓動 171
21.4 開機自動啓動服務 173
21.5 建立自訂 Systemd 服務 174
21.6 Timer 單元手冊 177
21.7 Timer 實測 179
21.8 實機練習 180

III 儲存空間管理 KVM10 **182**

22 開機 **183**
22.1 前言 183
22.2 開機 targets 183
22.3 忘記密碼 185
22.4 實機練習 187

23 DNF 套件管理 **188**
23.1 前言 188
23.2 dnf 套件庫設定 189
23.3 DNF 套件管理指令 190
23.4 實機練習 191

24 Flatpak 套件管理 **193**
24.1 前言 193
24.2 Flatpak 命令說明 193
24.3 安裝 flatpak 套件庫 196
24.4 flatpak 搜尋與安裝 197
24.5 安裝後執行 198
24.6 實機練習 199

25 邏輯卷冊放大縮小 **200**
25.1 簡介 200
25.2 lvreduce 縮小 LV 201
25.3 lvextend 放大 LV 202
25.4 lvresize 放大縮小 LV 203
25.5 實機練習 206

26 邏輯卷冊管理 **207**
26.1 前言 207
26.2 新增分割區 207
26.3 產生 PV 211
26.4 產生 VG 212
26.5 產生 LV 213
26.6 格式化 214
26.7 開機掛載表 /etc/fstab 214
26.8 開機掛載 FS 215
26.9 開機啓動 Swap 217
26.10 實機練習 218

27 Tuned 性能調整 **219**
27.1 前言 219
27.2 tuned-adm 命令 220
27.3 調整工作配置 221
27.4 實機練習 222

Part I

練習環境與基本操作

Chapter 1

實機練習系統

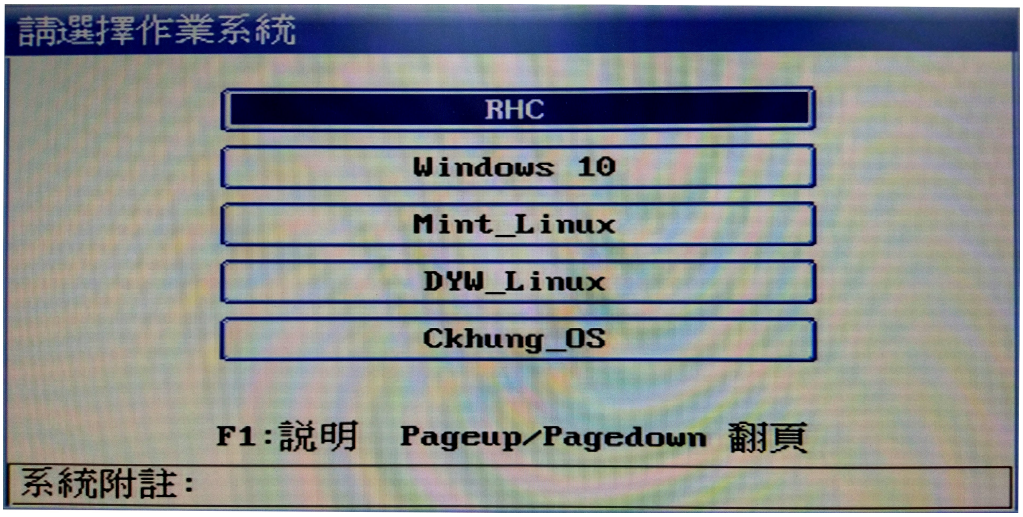
1.1 動機

1. 教學若有實際環境，讓學生實機操作練習，可以大大提昇學習成效。
2. Linux 實機練習，很容易因學生操作錯誤，而造成系統損壞。例如：硬碟分割、格式化及管理的練習。
3. 實機練習結束或操作不當造成當機，系統必須可以迅速還原，才能重複練習。
4. 系統必須可以批次快速修改或更新。
5. 系統必須可以依據學生實際操作狀況，批次進行檢查或評分。
6. 虛擬機練習所需伺服器資源，例如：LDAP 網路帳號、YUM repository 套件庫伺服器、NFS 分享、NTP 校時伺服器、ISCSI 硬碟分享、內部網域 DNS 伺服器、FTP 資源下載... 等環境必須架設好，才能提供虛擬機做各種狀況的練習。

1.2 系統設計

1. Linux 教學系統設計
   (a) 客製化 Linux 系統。
   (b) Linux 伺服器架設。
   (c) VNC 遠端桌面與 KVM 虛擬機。
2. Linux 教學系統安裝

(a) E517 電腦教室：方便學生上課使用。



(b) 個人隨身碟：插上隨身碟，以安裝光碟開機，不需要任何設定，自動安裝於隨身碟，不會影響電腦中的硬碟。

(c) LiveCD：以 LiveCD 光碟開機，開機後即可使用，只是所有操作與設定都是暫存，重新開機後即回到原始狀態。

1.3 練習系統製作-VirtualBox

1. 下載 DYW Linux 練習系統 vdi 檔，下載網址公告於本平台課程內，請登入查看。

```
[root@dywmsi ~]# unxz crt9.vdi.xz
```

2. 依自己使用的 OS (Mac Windows Linux) 下載安裝 VirtualBox。

```
https://www.virtualbox.org/wiki/Downloads
```

3. DYW Linux 練習系統，Host 主機以 virsh 管理 VM，最低硬體因所需 VM 個數不同：

(a) CPU 4 核心以上，若修的課不用內部虛擬機，設定 2 核心也可以。

(b) RAM 8G 以上，若修的課不用內部虛擬機，設定 4G RAM 也可以。

(c) RHCSA, RHCE 請設定 4 核心，8G RAM 以上

(d) CPU 支援虛擬化技術 (virtualization technology)，BIOS 啓用 VT。

4. VirtualBox 新增 VM

```
New(Ctrl+N)
    Name: 自訂，例如 crt9
    Type: Linux
    Version: 只要是 64-bit 就可以，例如 Fedora (64-bit)
    Next >
Memory size
    8192 MB
    Next >
Hard disk
    Use an existing virtual hard disk file
    新增解壓縮的 crt9.vdi，並選用這個硬碟檔。
    Create
```

5. 若要使用內部虛擬機，必須保證巢式虛擬化有開啓，假設匯入的 VM 名稱爲 crt9

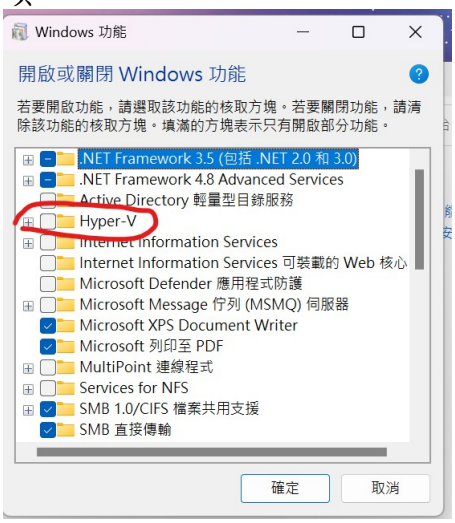
   (a) Host 是 Linux，在終端執行以下命令：

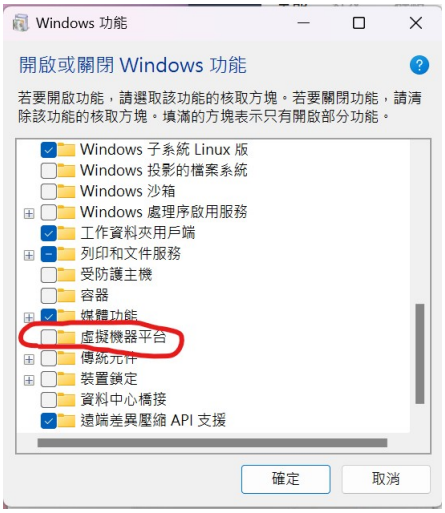
```
[dywang@dywmsi ~]$ VBoxManage modifyvm crt9 --nested-hw-virt on
```

   (b) Host 是 window，在命令提示 cmd 執行以下命令：

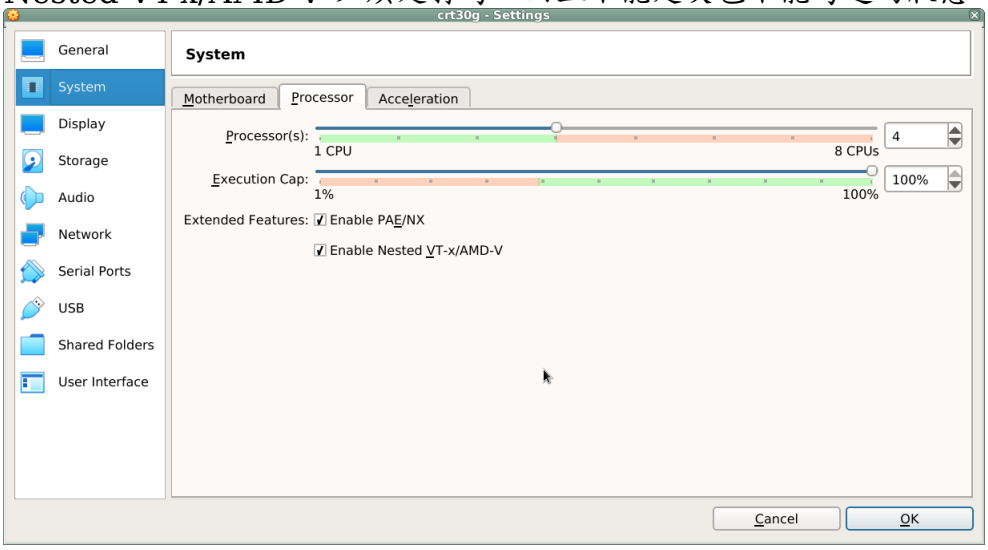
```
cd "c:\Program Files\Oracle\VirtualBox"
VBoxManage.exe modifyvm crt9 --nested-hw-virt on
```

   (c) Host 是 window，還必須關閉「Hyper-V」及「虛擬機器平台」，兩個選項。

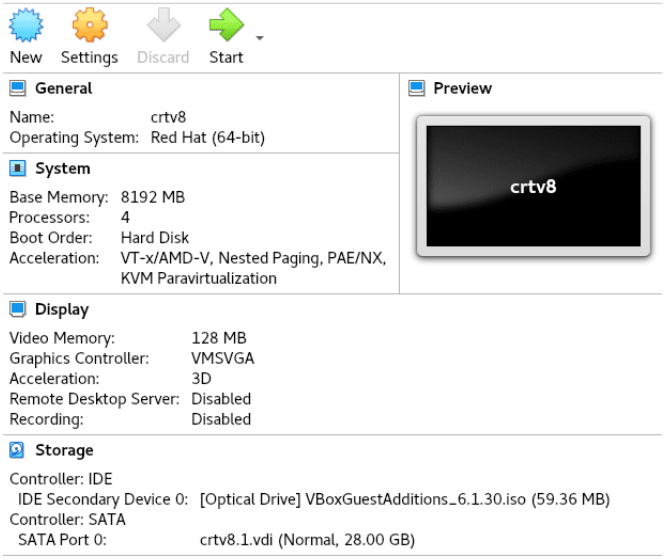




6. 進 VirtualBox 查看 VM crt9 的設定，system -> Processor -> Enable Nested VT-x/AMD-V 必須是打勾，而且不能是灰色不能勾選的狀態。



7. 使用 VirtualBox 或其他虛擬化工具，都要提供 VM crt9 足夠的 CPU 及 RAM，且要開啓巢式虛擬化，只要這三個要求達到，應該不會有□題，其他設定不懂可以先用預設值。若進不了圖形界面，可以打開 Display 的 3D 加速看看，VirtualBox VM 設定 (Settings) 參考值如下：



8. * 爲方便 VirtualBox 使用，建議下載附加套件 Oracle_VM_VirtualBox_Extension_Pack-6.1.30.vbox-extpack，放到你的 VirtualBox 目錄中，並於 VirtualBox 選單中加入附加 CD image。

```
Devices --> Insert Guest Additions CD image...
```

9. 開啓 DYW Linux 後，若虛擬機沒安裝，請先 rebuild 安裝虛擬機，練習後只需要 revert 還原，不用再要重新 rebuild，rebuild 要從學校 server 下載 image，要花較多時間。

1.4 練習系統製作-Raw Image

1. 下載 DYW Linux 練習系統 image 檔，或由 VDI 檔轉換格式，下載網址公告於本平台課程內，請登入查看。

2. 下載的 image 檔是 xz 壓縮檔，先用 unxz 解壓縮，windows 可用 winzip 解壓縮。

```
1 [root@dywmsi ~]# unxz crt9.img.xz
```

3. DYW Linux 練習系統，Host 主機以 virsh 管理 VM，最低硬體因所需 VM 個數不同：

   (a) CPU 4 核心以上，若修的課不用內部虛擬機，設定 2 核心也可以。

   (b) RAM 8G 以上，若修的課不用內部虛擬機，設定 4G RAM 也可以。

(c) RHCSA, RHCE 請設定 4 核心，8G RAM 以上

(d) CPU 支援虛擬化技術 (virtualization technology)，BIOS 啓用 VT。

4. 將 image 檔 dd 寫到 32G 以上硬碟或隨身碟，其中 /dev/sdX 是您要寫入的硬碟或隨身碟。

```
1 [root@dywmsi ~]# dd if=crt9.img of=/dev/sdX bs=100M
```

5. /dev/sdX 整顆硬碟的資料會被覆蓋，請確認要寫入的硬碟無誤再按 Enter 執行。

6. dd 完成後以 /dev/sdX 開機。

7. 開啓 DYW Linux 後，若虛擬機沒安裝，請先 rebuild 安裝虛擬機，練習後只需要 revert 還原，不用再要重新 rebuild，rebuild 要從學校 server 下載 image，要花較多時間。

1.5 NAT 設計

1. NAT 架設：電腦教室網路設置、管控學生上課上網。

2. 在 Linux，一行命令即可完成設定；在 Windows，Honda 老師亦幫忙撰寫批次檔。控制方式：

3. 任何設定皆不管控老師機，老師機可自由上網。

4. 學生機上網方式：

(a) 自由上網

(b) 不能上網

(c) 限制只能上學內網

(d) 限制只能上某一網站

(e) 限制只能上「依老師設定」的部分網站

(f) 只有某一學生機可自由上網

5. 不管老師做任何網路設定，下課後系統自動恢復成所有電腦皆可自由上網，以避免因老師忘了復原，而影響下節上課的老師。

1.6 虛擬機管理界面

1. 設計虛擬機管理 GUI 程式，方便學生管理虛擬機。程式包含虛擬機重安裝、還原、重新啓動及終端界面等四個功能。

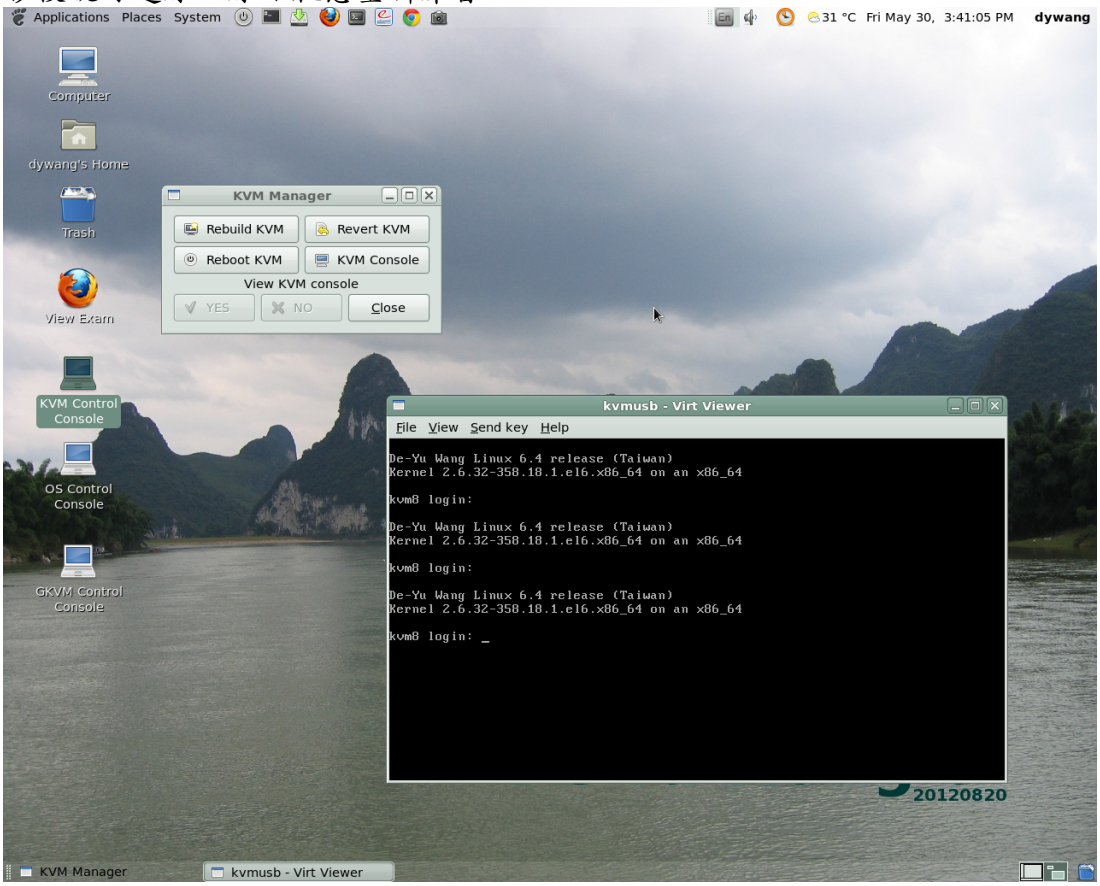
   (a) Rebuild KVM：重新安裝 KVM 虛擬機，約 10 幾分鐘。

   (b) Revert KVM：還原 KVM 虛擬機，約幾秒鐘到幾分鐘，請耐心等侯，不要在還原過程按管理界面的其他功能，以免造成虛擬機損壞。

   (c) Reboot KVM：重新啓動 KVM 虛擬機。

   (d) KVM Console：打開 KVM 虛擬機終端畫面。

2. 系統桌面中黑色背景視窗爲 KVM 虛擬機操作界面。原始 KVM 虛擬機主機名稱爲 111111，因已練習過，主機也已被設定成 kvm8，點擊 Revert KVM，幾秒後就可還原回原始狀態重新練習。



3. 在終端機以指令 virt-view 直接開啓虛擬機，出現錯誤訊息。

```
[dywang@dywH ~]$ sudo virt-viewer kvmusb
No protocol specified
Cannot open display:
Run 'virt-viewer --help' to see a full list of available command line
    options
```

4. 先檢查 Xwindow 認證變數 $XAUTHORITY 存不存在？

```
[dywang@dywH ~]$ echo $XAUTHORITY
```

5. 變數 XAUTHORITY 不存在，必須先設定 XAUTHORITY=~/.Xauthority 後再使用 sudo 啓動虛擬機，才能正常啓動。

```
[dywang@dywH ~]$ XAUTHORITY=~/.Xauthority sudo virt-viewer kvmusb
```

1.7 虛擬機管理界面 V2

1. 設計虛擬機管理 GUI 程式，方便學生管理虛擬機。程式包含虛擬機重安裝、還原、重新啓動、終端界面、啓用實體 USB 隨身碟、解除實體 USB 隨身碟的使用等六個功能。



(a) Rebuild KVM：重新安裝 KVM 虛擬機，直接下載 qcow2 映像檔，產生虛擬機，校內網路約 1 分多鐘。

(b) Revert KVM：還原 KVM 虛擬機，約幾秒鐘到幾分鐘，請耐心等侯，務必等黑色終端出現登入畫面，表示還原完成。練習前先關閉黑色終端畫面讓還原程式拍照，記下還原狀態，以免造成虛擬機損壞。

(c) Reboot KVM：重新啓動 KVM 虛擬機。

(d) KVM Console：打開 KVM 虛擬機終端畫面。

(e) Attach USB：插上 USB 隨身碟，按下此按鈕，KVM 虛擬機啓動實體 USB 隨身碟。

(f) Detach USB：KVM 虛擬機解除實體 USB 隨身碟的使用。

1.8 系統管理與評分

1. 系統管理腳本程式自動掃瞄目前開機的電腦，選擇要進行的工作，輸入密碼後即可自動完成。

```
[root@dywH ~]# e517sshpass.sh
myip=104
hosts=103,105
q(quit) iscsi halt reboot c(clear) l(lvm) rhcsa rhce
>  t(mounttest) k(kvmreboot) v(kvmrevert) s(servicerestart)
>  vi1 c1 lpvar c999 sh999 hp(hostpatch)? c999
keyin remote host root password:
```

2. 預設工作項目：

   (a) q(quit)：退出，無動作。

   (b) iscsi：學生機 ISCSI 硬碟分享架設。

   (c) halt：學生機關機。

   (d) reboot：學生機重新啓動。

   (e) c(clear)：登入學生機後清除部分資料，此部分保留修改彈性，目前入作爲限制學生機互相登入之限制，以防代作答情況。

   (f) l(lvm)：硬碟分割、格式化、自動掛載評分。

   (g) rhcsa：RHCSA 考照練習總評分。

   (h) rhce：RHCE 考照練習總評分。

   (i) t(mounttest)：測試學生機是否有掛載隨身碟或光碟。

   (j) k(kvmreboot)：學生機 KVM 虛擬機重新啓動。

   (k) v(kvmrevert)：學生機 KVM 虛擬機還原。

   (l) s(servicerestart)：啓動學生機主機服務。

   (m) vi1：vi 編輯器實機練習評分。

   (n) c1：C 語言第一支程式練習評分。

   (o) lpvar：shell 變數變化練習評分。

   (p) c999：C 語言程式實機練習評分。

   (q) sh999：Shell script 程式練習評分。

   (r) hp(hostpatch)：學生機網路修正後重新啓動網路。

3. 遠距教學，在家練習，自己評分上傳：

(a) 登入本平台，下載評分腳本及參數檔，以「資料庫」第一次評分爲例。

(b) 切換身分爲 root，root 密碼在平台部落格，請登入。

```
[dywang@deyu ~]$ su -
Password:
[root@deyu ~]#
```

(c) 切換工作目錄到評分程式的下載目錄，用瀏覽器下載預設是放在家目錄的 Downloads。

```
[root@deyu ~]# cd /home/dywang/Downloads
```

(d) 下載的評分程式沒有可執行的權限，先增加評分程式 mariadbeval1.sh.x 可執行屬性，使評分腳本 mariadbeval1.sh.x 成爲可執行檔。

```
[root@deyu Downloads]# chmod +x mariadbeval1.sh.x
```

(e) 查看目前有兩個檔案，評分程式爲 mariadbeval1.sh.x，參數檔爲 mariadbpara1.sh。

```
[root@deyu Downloads]# ll
-rwxrwxr-x. 1 root root 58656 May 21 13:29 mariadbeval1.sh.x
```

(f) 執行評分腳本 mariadbeval1.sh.x

```
[root@deyu Downloads]# ./mariadbeval1.sh.x
20210521-13:46:36
./mariadbeval1.sh.x
163.17.10.123
kvm=kvm19
163.17.10.123 10927000 王大呆
mariadbpara1.sh
mariadbeval1.sh.x_kvm19_1.mysqlversion:0
mariadbeval1.sh.x_kvm19_2.mariadb_enabled:0
mariadbeval1.sh.x_kvm19_3.mariadb_active:0
mariadbeval1.sh.x_kvm19_4.rootpw:0
```

(g) 如果評分參數檔不在同目錄，評分腳本 mariadbeval1.sh.x 無法執行。

```
[root@deyu Downloads]# mv mariadbpara1.sh /tmp/
```

```
[root@deyu Downloads]# ./mariadbeval1.sh
20210521-13:48:51
./mariadbeval1.sh
163.17.10.123
kvm=kvm19
163.17.10.123 10927000 王大呆
ls: cannot access mariadbpara1: No such file or directory
NO mariadbpara1 file
```

1.9 系統註冊-上課前先確定三件事

1. E517 電腦教室

   (a) E517 教室位置，靠窗爲第一排，確認自己電腦位置爲?-?。例如：1-1。

   (b) 在伺服器 (也就是開機進入圖形界面中有虛擬機管理界面的這台主機)，不是虛擬機，確認自己的電腦 IP。以下例子爲 192.168.1.140，IP 尾數爲 140。

```
[dywang@deyu ~]$ ifconfig | grep 192
          inet addr:192.168.1.140  Bcast:192.168.1.255  Mask
              :255.255.255.0
          inet addr:192.168.122.1  Bcast:192.168.122.255  Mask
              :255.255.255.0
```

   (c) 在伺服器 deyu.wang 主機一般帳號 dywang 家目錄，產生學號姓名檔 sid，檔案資訊格式爲「電腦位置 IP 尾數學號姓名」，每一欄位中間以一個空格隔開。以下例子電腦位置 1-11、IP 尾數 199、學號 123456，姓名王大呆。

```
[dywang@deyu ~]$ echo '1-11 199 123456 王大呆' > ~/sid
[dywang@deyu ~]$ cat ~/sid
1-11 199 123456 王大呆
```

2. 在家練習

   (a) 查詢練習系統對外 public ip 有三種方式：

```
[dywang@deyu ~]$ curl -4 ipinfo.io/ip
163.17.10.122[dywang@deyu ~]$
[dywang@deyu ~]$ curl -4 icanhazip.com
163.17.10.122
```

```
[dywang@deyu ~]$ curl -4  ifconfig.me
163.17.10.122[dywang@deyu ~]$
```

(b) 將 ip 學號姓名寫入 dywang 家目錄的 sid 檔，如下範例：

```
[dywang@deyu ~]$ echo '163.17.10.123 10827000 王大呆' > sid
```

(c) 查看 sid，ip 學號姓名務必正確，以免評分錯誤或無法匯入成績。

```
[dywang@deyu ~]$ cat sid
163.17.10.123 10827000 王大呆
```

1.10 虛擬機設定

1. 伺服器(也就是開機進入圖形界面中有虛擬機管理界面的這台主機)，不確定的話，先看命令列提示符號是不是 [dywang@deyu ~]$，或是 ifconfig 查目前所在主機的 IP：

   (a) Server 主機有兩個名稱 deyu.wang 及 server.deyu.wang

   (b) Server 與虛擬機網域 deyu.wang，使用網段 192.168.122.0/24。

   (c) Server IP 爲 192.168.122.1，虛擬機 IP 爲 192.168.122.X ，其中 X 依題目要求設定。

2. 虛擬機環境說明：

   (a) 虛擬機主機名稱 kvmX.deyu.wang IP 192.168.122.X，X 爲虛擬機編號，例如 5 或 7。

   (b) 虛擬機 root 密碼無法取得，統一重新設定爲 123qwe。

   (c) RHCE 練習環境增加
   網域 my111.wang，使用網段 192.168.111.0/24，
   server 主機增加此網域 IP 192.168.111.1。

   (d) 虛擬機網路設定：

```
IP Address:   192.168.122.X
Netmask:      255.255.255.0
Gateway:      192.168.122.1
Name server:  192.168.122.1
```

(e) RHCSA/RHCE 6,7 repository 套件庫網址：
http://dywang.csie.cyut.edu.tw/centos7.

(f) RHCSA 8 repository 套件庫網址：

i. BaseOS repo server:
http://dywang.csie.cyut.edu.tw/centos8/BaseOS.

ii. AppStream repo server:
http://dywang.csie.cyut.edu.tw/centos8/AppStream.

(g) kernel packages: 網路下載核心 RPM 檔
http://dywang.csie.cyut.edu.tw/kernel/centos7.

(h) LDAP server 設定：

i. 系統 deyu.wang

ii. base DN 爲 dc=deyu,dc=wang

iii. 證書 ftp://deyu.wang/pub/cacert.pem.

1.11 系統伺服器管理

1. 使用虛擬機練習時，伺服器 (也就是開機進入圖形界面中有虛擬機管理界面的這台主機) 必須正常運作，否則有些驗證無法進行。

2. 練習用到的 server 都已寫成腳本，如果運作有問題，可以執行腳本重啓。

1.11.1 DNS server

(a) 如果確定你的練習虛擬機 DNS 設定沒問題，但 ssh 卻無法使用主機名稱連線到虛擬機，可以在 server 機使用 host 命令看是否可以查到虛擬機的 IP？

(b) 沒有 DNS server 可查。

```
[dywang@deyu ~]$ host kvm7.deyu.wang
;; connection timed out; trying next origin
;; connection timed out; no servers could be reached
```

(c) 在 DNS server 查不到 kvm7.deyu.wang 的 IP。

```
[dywang@deyu ~]$ host kvm7.deyu.wang
Host kvm7.deyu.wang not found: 3(NXDOMAIN)
```

(d) 在伺服器 deyu.wang 主機切換成 root 身份。

```
[dywang@deyu ~]$ su -
Password: 123123
[root@deyu ~]#
```

(e) 執行 setdnsmasq.sh 腳本重啓 dnsmasq DNS 服務。

```
[root@deyu ~]# setdnsmasq.sh restart
```

(f) 再使用 host 查詢 kvm7.deyu.wang，已可查到其 IP 爲 192.168.122.7。

```
[root@deyu ~]# host kvm7.deyu.wang
kvm7.deyu.wang has address 192.168.122.7
kvm7.deyu.wang mail is handled by 1 20.
```

(g) 如果執行 setdnsmasq.sh 腳本重啓 dnsmasq DNS 服務未能解決問題，還可以使用 setdns.sh 腳本啓動 named DNS 服務。

```
[root@deyu ~]# setdns.sh restart
```

1.11.2 LDAP server

(a) 如果確認你的虛擬機 LDAP 設定沒問題，但還是無法以 ldapuser1 登入。在伺服器 deyu.wang 主機 (不是你的練習虛擬機)，先切換成 root 身份。

```
[dywang@deyu ~]$ su -
Password: 123123
[root@deyu ~]#
```

(b) 在伺服器執行 setldap1.sh restart 重新產生 LDAP 憑證。因爲憑證是新產生的，所以虛擬機必須再重新執行認證設定更新。

```
[root@deyu ~]# setldap1.sh restart
```

1.11.3 NFS server

(a) 虛擬機設定自動掛載必須查得到 server 有分享目錄，server 的 NFS 不像 LDAP 有憑證過期問題，所以一般都不需要重啓，但如果要重啓 NFS server，先在伺服器 deyu.wang 主機切換成 root 身份。

```
[dywang@deyu ~]$ su -
Password: 123123
[root@deyu ~]#
```

(b) 執行 setnfs.sh 腳本重新啓動。

```
[root@deyu ~]# setnfs.sh restart
```

Chapter 2

命令列基本操作

2.1 Linux 指令

1. 簡單指令

    - `man` - page 21：查詢指令使用手冊。
    - `date` - page 24：顯示日期及時間。
    - `cal` - page 27：顯示日月曆。
    - `bc` - page 31：簡單好用的計算機。
    - `sync` - page **??**：手動將緩衝於記憶體內的磁碟資料寫入磁碟。
    - `who` - page **??**：查看誰在線上。
    - `shutdown` - page **??**：通知線上使用者關機的時刻後關機。
    - `halt` - page **??**：關機。
    - `reboot` - page **??**：重新開機。
    - `chgrp` - page **??**：改變檔案所屬群組。
    - `chown` - page **??**：改變檔案擁有者。
    - `chmod` - page **??**：改變權限。
    - `cd` - page **??**：變換工作目錄。
    - `pwd` - page **??**：顯示目前所在的目錄。
    - `mkdir` - page **??**：建立新目錄。
    - `rmdir` - page **??**：刪除【空】的目錄。
    - `echo` - page **??**：顯示變數內容或文字。
    - `ls` - page **??**：顯示檔案名稱、屬性等。

- cp - page **??**：拷貝檔案或目錄。
- rm - page **??**：移除檔案或目錄。
- mv - page **??**：檔案移動或改名。
- cat - page **??**：由第一行開始連續 (concatenate) 顯示檔案內容。
- touch - page **??**：修改檔案時間，若檔案不存在，則產生新檔案。
- find - page **??**：功能強大的檔案搜尋指令。
- tar - page **??**：多用途打包壓縮指令。
- df - page **??**：查看檔案系統空間的使用狀況。
- fdisk - page **??**：對硬碟進行分割。
- mount - page **??**：掛載檔案系統。
- useradd - page 90：增加使用者。
- passwd - page **??**：設定或變更密碼。
- userdel - page **??**：刪除使用者。
- id - page **??**：查詢使用者的 UID、GID 及所擁有的群組。
- groupadd - page **??**：增加群組。
- su - page 102：轉換用户身份。
- crontab - page **??**：循環執行的工作排程。

2.2 線上求助

1. 大部分指令都有 --help 的選項，但不是所有指令皆有此選項，例如：

```
[root@mdk-dyw root]# ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuSUX nor --sort.

Mandatory arguments to long options are mandatory for short options too.
  -a, --all                  do not hide entries starting with .
  -A, --almost-all           do not list implied . and ..
      --author               print the author of each file
  -b, --escape               print octal escapes for nongraphic
      characters
      --block-size=SIZE      use SIZE-byte blocks
  -B, --ignore-backups       do not list implied entries ending with ~
  -c                         with -lt: sort by, and show, ctime (time of
       last
                               modification of file status information)
```

```
                               with -l: show ctime and sort by name
                               otherwise: sort by ctime
-C                             list entries by columns
    --color[=WHEN]             control whether color is used to
        distinguish file
                               types.  WHEN may be `never', `always', or
                                    `auto'
-d, --directory                list directory entries instead of contents,
                               and do not dereference symbolic links
-D, --dired                    generate output designed for Emacs' dired
    mode
-f                             do not sort, enable -aU, disable -lst
-F, --classify                 append indicator (one of */=@|) to entries
    --format=WORD              across -x, commas -m, horizontal -x, long -
        l,
                               single-column -1, verbose -l, vertical -C
    --full-time                like -l --time-style=full-iso
-g                             like -l, but do not list owner
-G, --no-group                 inhibit display of group information
-h, --human-readable  print sizes in human readable format (e.g., 1K
    234M 2G)
    --si                       likewise, but use powers of 1000 not 1024
-H, --dereference-command-line
                               follow symbolic links listed on the command
                                   line
    --dereference-command-line-symlink-to-dir
                               follow each command line symbolic link
                               that points to a directory
    --indicator-style=WORD append indicator with style WORD to entry
        names:
                               none (default), classify (-F), file-type
                                   (-p)
-i, --inode                    print index number of each file
-I, --ignore=PATTERN           do not list implied entries matching shell
    PATTERN
-k                             like --block-size=1K
-l                             use a long listing format
-L, --dereference              when showing file information for a
    symbolic
                               link, show information for the file the
                                   link
                               references rather than for the link
                                   itself
-m                             fill width with a comma separated list of
    entries
-n, --numeric-uid-gid          like -l, but list numeric UIDs and GIDs
-N, --literal                  print raw entry names (don't treat e.g.
    control
                               characters specially)
-o                             like -l, but do not list group information
-p, --file-type                append indicator (one of /=@|) to entries
-q, --hide-control-chars       print ? instead of non graphic characters
    --show-control-chars       show non graphic characters as-is (default
                               unless program is `ls' and output is a
```

```
                                terminal)
  -Q, --quote-name            enclose entry names in double quotes
      --quoting-style=WORD    use quoting style WORD for entry names:
                                literal, locale, shell, shell-always, c,
                                   escape
  -r, --reverse               reverse order while sorting
  -R, --recursive             list subdirectories recursively
  -s, --size                  print size of each file, in blocks
  -S                          sort by file size
      --sort=WORD             extension -X, none -U, size -S, time -t,
                                version -v
                              status -c, time -t, atime -u, access -u,
                                  use -u
      --time=WORD             show time as WORD instead of modification
                                 time:
                                atime, access, use, ctime or status; use
                                specified time as sort key if --sort=time
      --time-style=STYLE      show times using style STYLE:
                                full-iso, long-iso, iso, locale, +FORMAT
                              FORMAT is interpreted like `date'; if
                                  FORMAT is
                              FORMAT1<newline>FORMAT2, FORMAT1 applies to
                              non-recent files and FORMAT2 to recent
                                  files;
                              if STYLE is prefixed with `posix-', STYLE
                              takes effect only outside the POSIX locale
  -t                          sort by modification time
  -T, --tabsize=COLS          assume tab stops at each COLS instead of 8
  -u                          with -lt: sort by, and show, access time
                                with -l: show access time and sort by
                                   name
                                otherwise: sort by access time
  -U                          do not sort; list entries in directory
                                 order
  -v                          sort by version
  -w, --width=COLS            assume screen width instead of current
                                 value
  -x                          list entries by lines instead of by columns
  -X                          sort alphabetically by entry extension
  -1                          list one file per line
      --help     display this help and exit
      --version  output version information and exit

SIZE may be (or may be an integer optionally followed by) one of
                                 following:
kB 1000, K 1024, MB 1000*1000, M 1024*1024, and so on for G, T, P, E, Z,
                                  Y.

By default, color is not used to distinguish types of files.  That is
equivalent to using --color=none.  Using the --color option without the
optional WHEN argument is equivalent to using --color=always.  With
--color=auto, color codes are output only if standard output is
                                 connected
to a terminal (tty).
```

```
97
Report bugs to <bug-coreutils@gnu.org>.
```

2. 其他基本指令之 -help 功能：

   (a) date - page 24：顯示日期及時間。

   (b) cal - page 27：顯示日月曆。

   (c) bc - page 31：簡單好用的計算機。

3. 一般指令之 -help 選項可能提供：

   (a) 指令使用方法；

   (b) 選項之用途；

   (c) 若選項需要加入參數，則說明參數之格式；

   (d) 簡易說明。

2.3 man page

1. man：查詢指令使用手冊。

```
[dywang@mdk-dyw ~]$ man name
2 參數說明：
name: 要查詢的指令。
4
#範例：查詢 cal(calendar) 指令用法。
6 [dywang@mdk-dyw ~]$ man cal
CAL(1)                    BSD General Commands Manual
    CAL(1)
8
NAME
10      cal - displays a calendar

12 SYNOPSIS
      cal [-smjy13] [[month] year]
14
DESCRIPTION
16      Cal displays a simple calendar.  If arguments are not specified,
          the cur-
      rent month is displayed.  The options are as follows:
18
      -1      Display single month output.  (This is the default.)
20
      -3      Display prev/current/next month output.
22
```

```
     -s      Display Sunday as the first day of the week.  (This is the
             default.)

     -m      Display Monday as the first day of the week.

     -j      Display Julian dates (days one-based, numbered from January
         1).

q    -y      Display a calendar for the current year.

     A single parameter specifies the year (1 - 9999) to be displayed;
        note
     the year must be fully specified: ``cal 89'' will not display a
        calendar
     for 1989.  Two parameters denote the month (1 - 12) and year.  If
        no
     parameters are specified, the current month's calendar is displayed
        .

     A year starts on Jan 1.

     The Gregorian Reformation is assumed to have occurred in 1752 on
        the 3rd
     of September.  By this time, most countries had recognized the
        reforma-
     tion (although a few did not recognize it until the early 1900's.)
         Ten
     days following that date were eliminated by the reformation, so the
         cal-
     endar for that month is a bit unusual.

HISTORY
     A cal command appeared in Version 6 AT&T UNIX.

OTHER VERSIONS
     Several much more elaborate versions of this program exist, with
        support
     for colors, holidays, birthdays, reminders and appointments, etc.
        For
     example, try the cal from http://home.sprynet.com/~cbagwell/
        projects.html
     or GNU gcal.

BSD                                 June 6, 1993
                                    BSD
```

2. man 的相關檔案

   (a) 設定檔 /etc/man.config

   (b) 資料目錄 /usr/share/man

   (c) 文件目錄 /usr/share/doc

3. 數字代號

代號	代表內容
1	使用者可以操作的指令或可執行檔
2	系統核心可呼叫的函式與工具
3	常用函式 (function) 與函式庫 (library)
4	裝置檔案的說明
5	設定檔
6	遊戲
7	慣例與協定
8	系統管理員可用的管理指令
9	跟核心有關的文件

4. 內容代號

代號	代表內容
NAME	簡短指令、資料名稱說明
SYNOPSIS	簡短指令下達語法簡介
DESCRIPTION	較完整的說明
OPTIONS	針對 SYNOPSIS 中列舉的所有參數說明
COMMANDS	當此程式執行中可下達的指令
FILES	此程式所使用、參考或連結的檔案
SEE ALSO	與此指令相關的說明
EXAMPLE	參考範例
BUGS	是否有相關的臭蟲
HISTORY	指令歷史
OTHER VERSIONS	是否有其他版本

5. man 相關按鍵有：

按鍵	進行工作
空白鍵	往下翻一頁。
page up	往上翻一頁。
page down	往下翻一頁。
home	回到最前頁。
end	到最終頁。
/word	向下搜尋 word 這個字。
?word	向上搜尋 word 這個字。
n,N	n 繼續下一個搜尋，N 繼續反向搜尋。
q	離開。

6. man 使用功能

```
[root@mdk-dyw root]# man --help
man, version 1.5m

usage: man [-adfhktwW] [section] [-M path] [-P pager] [-S list]
        [-m system] [-p string] name ...

  a : find all matching entries
  c : do not use cat file
  d : print gobs of debugging information
  D : as for -d, but also display the pages
  f : same as whatis(1)
  h : print this help message
  k : same as apropos(1)
  K : search for a string in all pages
  t : use troff to format pages for printing
  w : print location of man page(s) that would be displayed
      (if no name given: print directories that would be searched)
  W : as for -w, but display filenames only

  C file   : use `file' as configuration file
  M path   : set search path for manual pages to `path'
  P pager  : use program `pager' to display pages
  S list   : colon separated section list
  m system : search for alternate system's man pages
  p string : string tells which preprocessors to run
              e - [n]eqn(1)   p - pic(1)    t - tbl(1)
              g - grap(1)     r - refer(1)  v - vgrind(1)
```

2.4 基礎指令操作

1. date：顯示日期及時間。

```
[dywang@dywHome2 ~]$ date --help
Usage: date [OPTION]... [+FORMAT]
  or:  date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]
Display the current time in the given FORMAT, or set the system date.

  -d, --date=STRING         display time described by STRING, not `now'
                          ## 以 STRING 格式顯示時間，不是'現在'時間。

  -f, --file=DATEFILE       like --date once for each line of DATEFILE
  -r, --reference=FILE      display the last modification time of FILE
  -R, --rfc-2822            output date and time in RFC 2822 format
      --rfc-3339=TIMESPEC   output date and time in RFC 3339 format.
                            TIMESPEC=`date', `seconds', or `ns' for
                            date and time to the indicated precision.
  -s, --set=STRING          set time described by STRING
                          ## 以 STRING 格式設定時間。
  -u, --utc, --universal    print or set
```

```
                    ## 顯示目前的格林威治時間。
      --help     display this help and exit
      --version  output version information and exit

FORMAT controls the output.  The only valid option for the second form
specifies Coordinated Universal Time.  Interpreted sequences are:

  %%   a literal %
  %a   locale's abbreviated weekday name (e.g., Sun)
  %A   locale's full weekday name (e.g., Sunday)
  %b   locale's abbreviated month name (e.g., Jan)
  %B   locale's full month name (e.g., January)
  %c   locale's date and time (e.g., Thu Mar  3 23:05:25 2005)
  %C   century; like %Y, except omit last two digits (e.g., 21)
  %d   day of month (e.g, 01)
  %D   date; same as %m/%d/%y
  %e   day of month, space padded; same as %_d
  %F   full date; same as %Y-%m-%d
  %g   last two digits of year of ISO week number (see %G)
  %G   year of ISO week number (see %V); normally useful only with %V
  %h   same as %b
  %H   hour (00..23)
  %I   hour (01..12)
  %j   day of year (001..366)
  %k   hour ( 0..23)
  %l   hour ( 1..12)
  %m   month (01..12)
  %M   minute (00..59)
  %n   a newline
  %N   nanoseconds (000000000..999999999)
  %p   locale's equivalent of either AM or PM; blank if not known
  %P   like %p, but lower case
  %r   locale's 12-hour clock time (e.g., 11:11:04 PM)
  %R   24-hour hour and minute; same as %H:%M
  %s   seconds since 1970-01-01 00:00:00 UTC
  %S   second (00..60)
  %t   a tab
  %T   time; same as %H:%M:%S
  %u   day of week (1..7); 1 is Monday
  %U   week number of year, with Sunday as first day of week (00..53)
  %V   ISO week number, with Monday as first day of week (01..53)
  %w   day of week (0..6); 0 is Sunday
  %W   week number of year, with Monday as first day of week (00..53)
  %x   locale's date representation (e.g., 12/31/99)
  %X   locale's time representation (e.g., 23:13:48)
  %y   last two digits of year (00..99)
  %Y   year
  %z   +hhmm numeric timezone (e.g., -0400)
  %:z  +hh:mm numeric timezone (e.g., -04:00)
  %::z  +hh:mm:ss numeric time zone (e.g., -04:00:00)
  %:::z  numeric time zone with : to necessary precision (e.g., -04,
      +05:30)
  %Z   alphabetic time zone abbreviation (e.g., EDT)

```

```
時間方面 :

 %% : 印出 %
 %n : 下一行
 %t : 跳格
 %H : 小時(00..23)
 %I : 小時(01..12)
 %k : 小時(0..23)
 %l : 小時(1..12)
 %M : 分鐘(00..59)
 %p : 顯示本地 AM 或 PM
 %r : 直接顯示時間 (12 小時制，格式爲 hh:mm:ss [AP]M)
 %s : 以 1970 年 1 月 1 日 00:00:00 UTC 到目前爲止的秒數
 %S : 秒(00..61)
 %T : 直接顯示時間 (24 小時制)
 %X : 相當於 %H:%M:%S
 %Z : 顯示時區

 日期方面 :
 %a : 星期幾 (Sun..Sat)
 %A : 星期幾 (Sunday..Saturday)
 %b : 月份 (Jan..Dec)
 %B : 月份 (January..December)
 %c : 直接顯示日期與時間
 %d : 日 (01..31)
 %D : 直接顯示日期 (mm/dd/yy)
 %h : 同 %b
 %j : 一年中的第幾天 (001..366)
 %m : 月份 (01..12)
 %U : 一年中的第幾周 (00..53) (以 Sunday 爲一周的第一天的情形)
 %w : 一周中的第幾天 (0..6)
 %W : 一年中的第幾周 (00..53) (以 Monday 爲一周的第一天的情形)
 %x : 直接顯示日期 (mm/dd/yy)
 %y : 年份的最後兩位數字 (00.99)
 %Y : 完整年份 (0000..9999)


By default, date pads numeric fields with zeroes.
The following optional flags may follow `%':

  - (hyphen) do not pad the field
  _ (underscore) pad with spaces
  0 (zero) pad with zeros
  ^ use upper case if possible
  # use opposite case if possible

After any flags comes an optional field width, as a decimal number;
then an optional modifier, which is either
E to use the locale's alternate representations if available, or
O to use the locale's alternate numeric symbols if available.

Report bugs to <bug-coreutils@gnu.org>.

[dywang@mdk-dyw ~]$ date +%Y/%m/%d
```

```
2006/10/05

[dywang@dywOffice ~]$ date +%H:%M
09:37

[dywang@dywHome2 ~]$  # 顯示時間後跳行，再顯示目前日期:
[dywang@dywHome2 ~]$ date '+%T%n%D'
18:16:19
08/01/09

[dywang@dywHome2 ~]$  # 顯示月份與日
[dywang@dywHome2 ~]$ date '+%B %d'
August 01

[dywang@dywHome2 ~]$  # 明天
[dywang@dywHome2 ~]$ date --date="1 day" +%F
2009-08-02

[dywang@dywHome2 ~]$  # 後天
[dywang@dywHome2 ~]$  date --date="2 day" +%F
2009-08-03

[dywang@dywHome2 ~]$  # 昨天
[dywang@dywHome2 ~]$  date --date="1 day ago" +%F
2009-07-31

[dywang@dywHome2 ~]$  # 前天
[dywang@dywHome2 ~]$  date --date="2 day ago" +%F
2009-07-30

[dywang@dywHome2 ~]$  # 2009-1-1 的前一天
[dywang@dywHome2 ~]$  date --date="2009-1-1 1 day ago" +%F
2008-12-31

[dywang@dywHome2 ~]$  # 2008-2-28 的後一天
[dywang@dywHome2 ~]$  date --date="2008-2-28 1 day" +%F
2008-02-29

[dywang@dywHome2 ~]$  # 2008-2-28的兩個月前
[dywang@dywHome2 ~]$  date --date="2008-2-28 2 month ago" +%F
2007-12-28

[root@dywHome2 ~]#  # 設定時間爲 12:34:56，必須以 root 身份
[root@dywHome2 ~]# date -s '12:34:56'
Sat Aug  1 12:34:56 CST 2009
```

2. cal：顯示日月曆。

```
[dywang@dywHome2 ~]$ cal --help
cal: invalid option -- -
usage: cal [-13smjyV] [[month] year]

```

```
[dywang@mdk-dyw ~]$ man cal
CAL(1)                    BSD General Commands Manual
    CAL(1)

NAME
     cal - displays a calendar

SYNOPSIS
     cal [-smjy13] [[month] year]

DESCRIPTION
     Cal displays a simple calendar.  If arguments are not specified,
        the cur-
     rent month is displayed.  The options are as follows:

     -1      Display single month output.  (This is the default.)

     -3      Display prev/current/next month output.

     -s      Display Sunday as the first day of the week.  (This is the
             default.)

     -m      Display Monday as the first day of the week.

     -j      Display Julian dates (days one-based, numbered from January
          1).

     -y      Display a calendar for the current year.

     A single parameter specifies the year (1 - 9999) to be displayed;
        note
     the year must be fully specified: ``cal 89'' will not display a
        calendar
     for 1989.  Two parameters denote the month (1 - 12) and year.  If
        no
     parameters are specified, the current month's calendar is displayed
        .

     A year starts on Jan 1.

     The Gregorian Reformation is assumed to have occurred in 1752 on
        the 3rd
     of September.  By this time, most countries had recognized the
        reforma-
     tion (although a few did not recognize it until the early 1900's.)
          Ten
     days following that date were eliminated by the reformation, so the
          cal-
     endar for that month is a bit unusual.

HISTORY
     A cal command appeared in Version 6 AT&T UNIX.

OTHER VERSIONS
```

```
     Several much more elaborate versions of this program exist, with
         support
     for colors, holidays, birthdays, reminders and appointments, etc.
         For
     example, try the cal from http://home.sprynet.com/~cbagwell/
         projects.html
     or GNU gcal.

BSD                                June 6, 1993
                                   BSD

[dywang@dywHome2 ~]$ cal
    August 2009
Su Mo Tu We Th Fr Sa
                    1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
[dywang@dywHome2 ~]$ cal -1
    August 2009
Su Mo Tu We Th Fr Sa
                    1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
[dywang@dywHome2 ~]$ cal -3
     July 2009            August 2009          September 2009
Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa  Su Mo Tu We Th Fr Sa
          1  2  3  4                     1         1  2  3  4  5
 5  6  7  8  9 10 11   2  3  4  5  6  7  8   6  7  8  9 10 11 12
12 13 14 15 16 17 18   9 10 11 12 13 14 15  13 14 15 16 17 18 19
19 20 21 22 23 24 25  16 17 18 19 20 21 22  20 21 22 23 24 25 26
26 27 28 29 30 31     23 24 25 26 27 28 29  27 28 29 30
                      30 31
[dywang@dywHome2 ~]$ cal -s
    August 2009
Su Mo Tu We Th Fr Sa
                    1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
[dywang@dywHome2 ~]$ cal -m
    August 2009
Mo Tu We Th Fr Sa Su
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
```

```
24 25 26 27 28 29 30
31
[dywang@dywHome2 ~]$ cal -j
        August 2009
Sun Mon Tue Wed Thu Fri Sat
                         213
214 215 216 217 218 219 220
221 222 223 224 225 226 227
228 229 230 231 232 233 234
235 236 237 238 239 240 241
242 243
[dywang@dywHome2 ~]$ cal -y
                               2009

       January               February                March
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
             1  2  3    1  2  3  4  5  6  7    1  2  3  4  5  6  7
 4  5  6  7  8  9 10    8  9 10 11 12 13 14    8  9 10 11 12 13 14
11 12 13 14 15 16 17   15 16 17 18 19 20 21   15 16 17 18 19 20 21
18 19 20 21 22 23 24   22 23 24 25 26 27 28   22 23 24 25 26 27 28
25 26 27 28 29 30 31                          29 30 31

        April                 May                   June
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
          1  2  3  4                   1  2       1  2  3  4  5  6
 5  6  7  8  9 10 11    3  4  5  6  7  8  9    7  8  9 10 11 12 13
12 13 14 15 16 17 18   10 11 12 13 14 15 16   14 15 16 17 18 19 20
19 20 21 22 23 24 25   17 18 19 20 21 22 23   21 22 23 24 25 26 27
26 27 28 29 30         24 25 26 27 28 29 30   28 29 30
                       31
        July                  August                September
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
          1  2  3  4                      1          1  2  3  4  5
 5  6  7  8  9 10 11    2  3  4  5  6  7  8    6  7  8  9 10 11 12
12 13 14 15 16 17 18    9 10 11 12 13 14 15   13 14 15 16 17 18 19
19 20 21 22 23 24 25   16 17 18 19 20 21 22   20 21 22 23 24 25 26
26 27 28 29 30 31      23 24 25 26 27 28 29   27 28 29 30
                       30 31
       October               November              December
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
             1  2  3    1  2  3  4  5  6  7          1  2  3  4  5
 4  5  6  7  8  9 10    8  9 10 11 12 13 14    6  7  8  9 10 11 12
11 12 13 14 15 16 17   15 16 17 18 19 20 21   13 14 15 16 17 18 19
18 19 20 21 22 23 24   22 23 24 25 26 27 28   20 21 22 23 24 25 26
25 26 27 28 29 30 31   29 30                  27 28 29 30 31


[dywang@dywHome2 ~]$ cal -m3 9 2012
    August 2012          September 2012         October 2012
Mo Tu We Th Fr Sa Su  Mo Tu We Th Fr Sa Su  Mo Tu We Th Fr Sa Su
       1  2  3  4  5                   1  2   1  2  3  4  5  6  7
 6  7  8  9 10 11 12   3  4  5  6  7  8  9   8  9 10 11 12 13 14
13 14 15 16 17 18 19  10 11 12 13 14 15 16  15 16 17 18 19 20 21
20 21 22 23 24 25 26  17 18 19 20 21 22 23  22 23 24 25 26 27 28
```

```
152 27 28 29 30 31          24 25 26 27 28 29 30  29 30 31
```

3. bc：簡單好用的計算機。

```
[dywang@dywHome2 ~]$ bc --help
usage: bc [options] [file ...]
  -h  --help         print this usage and exit
  -i  --interactive  force interactive mode
  -l  --mathlib      use the predefine math routnes
  -q  --quiet        don't print initial banner
  -s  --standard     non-standard bc constructs are errors
  -w  --warn         warn about non-standard bc constructs
  -v  --version      print version information and exit

[dywang@mdk-dyw ~]$ man bc
## 以下只整理使用重點，完整使用手冊請自行參考 man bc。

   NUMBERS
       ...省略... The length is the total number of significant  decimal
            digits
       in a number and the scale is the total number of decimal  digits
            after
       the decimal point.  For example:
               .000001 has a length of 6 and scale of 6.
               1935.000 has a length of 7 and a scale of 3.

   VARIABLES
       ...省略... All  letters  must  be  lower case.
       ...省略... The default value of scale is 0.

   COMMENTS
       Comments in bc start with the characters /* and end with the
           characters
       */. ...省略... A single line comment starts at a # character
       and continues to the next end of the line. ...省略...

   EXPRESSIONS
       ...省略...

       - expr The result is the negation of the expression.

       ++ var The variable is incremented by one and  the  new  value
           is  the
              result of the expression.

       -- var The  variable  is  decremented  by  one and the new value
           is the
              result of the expression.

       var ++  The result of the expression is the value of the
           variable  and
              then the variable is incremented by one.
```

```
44      var -- The result of the expression is the value of the
            variable and
            then the variable is decremented by one.
46
        expr + expr
48          The result of the expression is the sum of the two
                expressions.
50      expr - expr
            The result of the expression is the difference of
                the two
52          expressions.
54      expr * expr
            The result of the expression is the product of the two
                expres-
56          sions.
58      expr / expr
            The result of the expression is the quotient of the two
                expres-
60          sions. The scale of the result is the value of the
                variable
            scale.
62
        expr % expr
64          The result of the expression is the "remainder". ...省略
                ...
66      expr ^ expr
            The result of the expression is the value of the first
                raised to
68          the second. The second expression must be an integer.
                ...省略...
70      ( expr )
            This alters the standard precedence to force the
                evaluation of
72          the expression.
74      var = expr
            The variable is assigned the value of the expression.
76
        var <op>= expr
78          This is equivalent to "var = var <op> expr" ...省略...
80      ...省略... relational operators are:
82      expr1 < expr2
            The result is 1 if expr1 is strictly less than expr2.
84
        expr1 <== expr2
86          The result is 1 if expr1 is less than or equal to expr2.
```

```
88      expr1 > expr2
            The result is 1 if expr1 is strictly greater than expr2.
90
        expr1 >= expr2
92          The result is 1 if expr1 is greater than or equal to expr2
                .
94      expr1 == expr2
            The result is 1 if expr1 is equal to expr2.
96
        expr1 != expr2
98          The result is 1 if expr1 is not equal to expr2.
100     ...省略...  The  boolean  operators are:
102     !expr  The result is 1 if expr is 0.
104     expr && expr
            The result is 1 if both expressions are non-zero.
106
        expr || expr
108         The result is 1 if either expression is non-zero.
110     The expression precedence is as follows: (lowest to highest)
            || operator, left associative
112         && operator, left associative
            ! operator, nonassociative
114         Relational operators, left associative
            Assignment operator, right associative
116         + and - operators, left associative
            *, / and % operators, left associative
118         ^ operator, right associative
            unary - operator, nonassociative
120         ++ and -- operators, nonassociative
122     ...省略...
124     length ( expression )
            The  value  of  the length function is the number of
                significant
126         digits in the expression.
128     read ( )
            The read function (an extension) will read  a  number
                from  the
130         standard   input,  ...省略...
132     scale ( expression )
            The value of the scale function is the number  of  digits
                 after
134         the decimal point in the expression.
136     sqrt ( expression )
```

```
            The value of the sqrt function is the square root of the
                expres-
138         sion.  If the expression is negative, a run time error is
                gener-
            ated.
140
    STATEMENTS
142     ...省略...

144     { statement_list }
            This is the compound statement.  It allows  multiple
                statements
146         to be grouped together for execution.

148     if ( expression ) statement1 [else statement2]
            ...省略...
150         If  the  expression  is  non-zero,  statement1  is
                executed.  If
            statement2 is present and the value of the expression is
                0, then
152         statement2 is executed.

154     while ( expression ) statement
            The while statement will execute the statement while the
                expres-
156         sion is non-zero.  ...省略...

158     for ( [expression1] ; [expression2] ; [expression3] ) statement
            The for statement controls repeated execution of the
                statement.
160         ...省略...

162
        break  This statement causes a forced exit of the most recent
            enclosing
164         while statement or for statement.

166     continue
            The continue statement (an extension)  causes  the  most
                recent
168         enclosing for statement to start the next iteration.

170     halt   ...省略...

172     return Return the value 0 from a function.  (See the section  on
             func-
            tions.)
174
        return ( expression )
176         Return  the  value  of the expression from a function. ...
                省略...

178 PSEUDO STATEMENTS
    ...省略...
```

```
180
    FUNCTIONS
182     ...省略...   A function is defined as follows:
                define name ( parameters ) { newline
184                 auto_list   statement_list }
        A function call is just an expression of the form "name(
            parameters)".
186
        The  auto_list  is  an  optional list of variables that are for "
            local"
188     use.  The syntax of the auto list (if present) is "auto name,
            ...  ;".
        ...省略...  The variables are then initialized to  zero ...省略
            ...
190
        For example, the following definitions are legal.
192             define d (n) { return (2*n); }
                define d (n)
194               { return (2*n); }

196 MATH LIBRARY
        ...省略...
198
        s (x)  The sine of x, x is in radians.
200
        c (x)  The cosine of x, x is in radians.
202
        a (x)  The arctangent of x, arctangent returns radians.
204
        l (x)  The natural logarithm of x.
206
        e (x)  The exponential function of raising e to the value x.
208
        j (n,x)
210             The bessel function of integer order n of x.

212 EXAMPLES
        ...省略...
214
        範例：定義階層 x! 函數
216             define f (x) {
                  if (x <== 1) return (1);
218               return (f(x-1) * x);
                }
220
    READLINE AND LIBEDIT OPTIONS
222     ...省略...

224 DIFFERENCES
        ...省略...
226
    LIMITS
228     ...省略...
```

```
ENVIRONMENT VARIABLES
        ...省略...

DIAGNOSTICS
        ...省略...

BUGS
        ...省略...

AUTHOR
       Philip A. Nelson
       philnelson@acm.org

ACKNOWLEDGEMENTS
       The author would like to thank  Steve  Sommars  (Steve.
           Sommars@att.com)
       for  his extensive help in testing the implementation.  Many
           great sug-
       gestions were given.  This is a much better product due to his
           involve-
       ment.



                                        .

                                            bc(1)

[dywang@mdk-dyw ~]$ bc
bc 1.06
Copyright 1991-1994, 1997, 1998, 2000 Free Software Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
(3+4)^2
49
4/3
1
scale=3
4/3
1.333

sacle=0
x=read()
100
for(i=0;i<x;i++) {
total+=(i+1)
}
total
5050

var1=10
var2=var1++
var1;var2
11
```

```
10
var2=++var1
var1;var2
12
12

define f (x) {
if (x <== 1) return (1);
return (f(x-1) * x);
}

f(4)
24
```

2.5 重要的幾個熱鍵與錯誤訊息的察看

1. [Tab] 按鍵：命令補全及檔案補齊

```
[dywang@mdk-dyw ~]$ ls[Tab]
ls          lsd         lsnetdrake  lspcidrake  lss16toppm
lsattr      lshal       lspci       lspgpot
[dywang@mdk-dyw ~]$ ls -al ~/.bash[Tab]
.bash_history  .bash_logout   .bash_profile  .bashrc        .bashrc~
```

2. [Ctrl]+c 按鍵：中斷目前程式

```
[dywang@mdk-dyw ~]$ find / -name *mdk*
```

3. [Ctrl]+d 按鍵：鍵盤輸入結束或取代 exit 直接離開文字界面。

4. 藉由螢幕上顯示的錯誤訊息解決問題。

```
[dywang@mdk-dyw ~]$ Date
bash: Date: command not found
[dywang@mdk-dyw ~]$ cal 13 2006
cal: illegal month value: use 1-12
```

Chapter 3

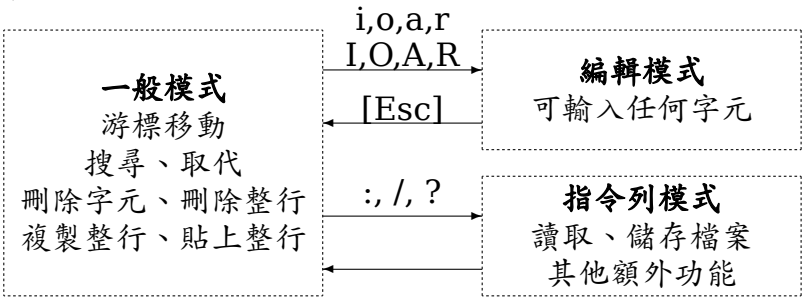
vi 編輯器

3.1 vi 與 vim

1. Linux 的系統中使用文字編輯器來編輯您的 Linux 參數設定檔。
2. Linux 與 Unix 系統中的參數檔幾乎都是 ASCII 碼的『純文字』檔。
3. vi 爲文書編輯器，vim 是 vi 的進階軟體。
4. vim 加入了支援正規表示法的搜尋架構、多檔編輯、區塊複製等功能。
5. vim 官方網站亦稱 vim 爲程式開發工具。
6. 大家來學 VIM（一個歷久彌新的編輯器）

3.2 vi 的使用

1. vi 共分爲三種模式，分別是『一般模式』、『編輯模式』與『指令列命令模式』三種。



2. 簡易範例

   (a) 使用 vi 進入一般模式

```
[dywang@deyu zzz]$ vim test.txt
```

(b) 按下 i 進入編輯模式，開始編輯文字。

(c) 按下 [Esc] 按鈕回到一般模式。

(d) 在一般模式中按下:wq 儲存後離開 vi。

3. 常用指令列表

一般模式	移動游標
h 或向左方向鍵	游標向左移動一個字元
j 或向下方向鍵	游標向下移動一個字元
k 或向上方向鍵	游標向上移動一個字元
l 或向右方向鍵	游標向右移動一個字元
[Ctrl]+f	螢幕『向前』移動一頁
[Ctrl]+b	螢幕『向後』移動一頁
[Ctrl]+d	螢幕『向前』移動半頁
[Ctrl]+u	螢幕『向後』移動半頁
+	游標移動到非空白字元的下一行
-	游標移動到非空白字元的上一行
n<space>	按下數字後再按空白鍵，游標會向右移動這一行的 n 個字元。例如 20<space> 則游標會向後面移動 20 個字元距離。
0	這是數字『0 』: 移動到這一行的最前面字元處。
$	移動到這一行的最後面字元處
H	游標移動到這個螢幕的最上方那一行
M	游標移動到這個螢幕的中央那一行
L	游標移動到這個螢幕的最下方那一行
G	游標移動到這個檔案的最後一行
nG	游標移動到這個檔案的第 n 行。例如 20G 則會移動到檔案的第 20 行 (可配合:set nu)
n<Enter>	游標向下移動 n 行

一般模式	搜尋與取代
/word	向游標之後尋找一個字串名稱爲 word 的字串。
?word	向游標之前尋找一個字串名稱爲 word 的字串。
n	重複前一個搜尋。
N	反向進行前一個搜尋動作。

:n1,n2s/word1/word2/g	在第 n1 與 n2 行之間尋找 word1 這個字串，並將該字串取代爲 word2 。加 g 參數取代該行所有匹配字串，否則只取代第一個字串。
:1,$s/word1/word2/g	從第一行到最後一行尋找 word1 字串，並將該字串取代爲 word2 。
:1,$s/word1/word2/gc	從第一行到最後一行尋找 word1 字串，並將該字串取代爲 word2 且在取代前顯示提示字元給使用者確認是否需要取代。

一般模式	**刪除、複製與貼上**
x, X	x 爲向後刪除一個字元，X 爲向前刪除一個字元
nx	向後刪除 n 個字元
dd	刪除游標所在的那一整行
ndd	刪除游標所在的向下 n 行，例如 20dd 則是刪除 20 行
d1G	刪除游標所在到第一行的所有資料
dG	刪除游標所在到最後一行的所有資料
dw	刪除一個字 (delete word)。不適用於中文。
yy	複製游標所在的那一行（常用）
nyy	複製游標所在的向下 n 行，例如 20yy 則是複製 20 行
y1G	複製游標所在行到第一行的所有資料
yG	複製游標所在行到最後一行的所有資料
p, P	p 爲複製的資料在游標下一行貼上，P 則爲貼在游標上一行。
J	將游標所在行與下一行的資料結合成同一行
u	復原前一個動作。
[Ctrl]+r	反復原前一個動作。
.	重做上一個動作。

進入編輯模式	
i, I	插入：i(I) 在目前的游標所在處 (所在行第一個非空白字元處) 插入輸入文字；
a, A	增加：a(A) 由目前游標所在的下一個 (最後一個) 字元處開始輸入：
o, O	插入新的一行：o(O) 從游標所在的下 (上) 一行插入新的一行；
r, R	取代：r 會取代游標所在的那一個字元；R 會一直取代游標所在的文字，直到按下 ESC 爲止；
Esc	退出編輯模式，回到一般模式中。

指令列命令模式	
:w	將編輯的資料寫入硬碟檔案中
:w!	若檔案屬性爲『唯讀』時，強制寫入該檔案

:q	離開 vi
:q!	若曾修改過檔案，又不想儲存，使用! 爲強制離開不儲存檔案。
:wq	儲存後離開，若爲:wq! 則爲強制儲存後離開
:x	與 wq 相同
:e!	將檔案還原到最原始的狀態。
ZZ	若檔案沒有更動，則不儲存離開，若檔案已經經過更動，則儲存後離開。
:w [filename]	將編輯的資料儲存成另一個檔案（類似另存新檔）
:r [filename]	在編輯的資料中，讀入另一個檔案的資料。亦即將『filename』這個檔案內容加到游標所在行後面。
:set nu	顯示行號，設定之後，會在每一行的字首顯示該行的行號
:set nonu	與 set nu 相反，爲取消行號。
:n1,n2 w [filename]	將 n1 到 n2 的內容儲存成 filename 這個檔案。
:! command	暫時離開 vi 到指令列模式下執行 command 的顯示結果。例如 [:! ls /home]

4. vi 編輯中回復與暫存檔

   (a) 如果編輯檔案爲 /home/csie/vitest.txt，則同時會有一個暫存檔 /home/csie/.vitest.txt.swp 產生，其爲隱藏檔。

   (b) 結束 vi 編輯時暫存檔也會刪除。

   (c) 若不知因素導致 vi 程式中斷，則在下次編輯該檔案時會出現是否回復的訊息。

3.3 vim 的額外功能

1. 如果使用 vi，在畫面右下角有目前游標所在行列號碼，則 vi 已被 vim 取代了。

```
[dywang@deyu zzz]$ ll /bin/vi*
lrwxrwxrwx  1 root root 20 Oct 17 13:11 /bin/vi -> /etc/alternatives/vi*
lrwxrwxrwx  1 root root 21 Oct 17 13:11 /bin/vim -> /etc/alternatives/
    vim*
[dywang@deyu zzz]$ ll /etc/alternatives/vi*
lrwxrwxrwx  1 root root 21 Oct 17 13:11 /etc/alternatives/vi ->
/usr/bin/vim-enhanced*
lrwxrwxrwx  1 root root 21 Oct 17 13:11 /etc/alternatives/vim ->
/usr/bin/vim-enhanced*
```

2. vim 具有顏色顯示的功能，並且還支援許多的程式語法。

3. vim /etc/man.config 出現訊息說明：

```
"/etc/man.config" [readonly] 150L, 4900C                    1,1
   Top
```

(a) 編輯檔案名稱爲 /etc/man.config；

(b) 檔案爲唯讀檔；

(c) 檔案共有 150 行，4900 字元；

(d) 目前游標所在位置爲第一行, 第一列；

(e) 目前頁面在最前頁。

4. 區塊選擇

	區塊選擇的按鍵意義
v	字元選擇，會將游標經過的地方反白選擇。
V	行選擇，會將游標經過的反白選擇。
[Ctrl]+v	區塊選擇，可以用長方形的方式選擇資料。
y	將反白的地方複製起來。
d	將反白的地方刪除掉。

5. 多檔案編輯

(a) vi 內使用:r filename 可將檔案 filename 的內容在游標處插入。

(b) 可於 vim 後接多個檔案來同時開啓多個檔案，例如：vim filename1 filename2 filename3。其相關按鍵有：

	多檔案編輯的按鍵
:n	編輯下一個檔案。
:N	編輯上一個檔案。
:files	列出目前開啓的所有檔案。

6. 多視窗功能

(a) 在指令列模式輸入:sp filename。

(b) 如果省略 filename 則兩視窗爲同一檔案。

	多視窗下的按鍵功能
:sp	開啓同一檔案於新視窗。
:sp filename	開啓檔案 filename 於新視窗。
[Ctrl]+wj	游標移動到下方的視窗。按法爲：先按下 [Ctrl] 不放，再下 w 後放開所有的按鍵，然後再按下 j。
[Ctrl]+wk	游標移動到上方的視窗。按法爲同上。
[Ctrl]+wq	結束下方視窗，與 [Ctrl]+w 移動到下方視窗後，再按下:q 離開相同。

7. vim 環境設定

   (a) 個人動作記錄檔案：~/.viminfo。例如：編輯同一檔案時，游標會在上次退出時的位置。

   (b) 整體 vim 的設定值放在 /etc/vimrc。

   (c) 若要更改 vim 設定，建議自行建立 ~/.vimrc。

	vim 的環境設定參數
:set nu	設定行號。
:set nonu	取消定行號。
:set hlsearch	將搜尋的字串反白。
:set autoindent	自動縮排。
:set noautoindent	不自動縮排。
:set backup	自動儲存備份。備份檔名爲 filename~。
:set rule	顯示右下角的狀態說明。
:set showmode	顯示 –insert– 等字眼在左下角的狀態列。
:set backspace=(012)	2 利用 backspace 例退鍵除任意字元；0 或 1 僅可刪除剛剛輸入的字元。
:set all	顯示目前所有的環境參數設定值。
:syntax (off|on)	是否依據程式相關語法顯示不同顏色。

3.4 vi 實機練習題

3.4.1 練習一

1. 在自己的家目錄建立一個新的目錄 zzz。

2. 進入目錄 zzz。

3. 下載檔案 viex1.txt
4. 將 viex1.txt 重新命名爲 vi1.txt
5. 使用 vi 或 vim 編輯 vi1.txt，執行以下動作：
   (a) 先到第 n 行，n 爲學號除 37 取餘數 +10。
   (b) 複製 3 行。
   (c) 再到第 20 行，往下貼上。
   (d) 再到最後一行，往下產生一行空白行方式進入”插入”模式進行編輯。
   (e) 加入文字’ABCDEabcde’。
   (f) 換行再加入’1234567890’。
   (g) 搜尋字串”MANPATH”，並將其取代爲”manpath”。
   (h) 存檔後退出。

3.4.2 練習二

1. 在自己的家目錄建立一個新的目錄 zzz。
2. 進入目錄 zzz。
3. 下載檔案 viex2.txt
4. 將 viex2.txt 重新命名爲 vi2.txt
5. 使用 vi 或 vim 編輯 vi2.txt，執行以下動作：
   (a) 先到第 n 行，n 爲學號除 27 取餘數 +8。
   (b) 刪除 3 行。
   (c) 再到第 30 行，往上貼上。
   (d) 再到第一行，往上產生一行空白行方式進入”插入”模式進行編輯。
   (e) 加入文字’ABCDEabcde’。
   (f) 換行再加入’1234567890’。
   (g) 從第 30 行至最後一行搜尋字串「MANPATH」，並將其取代爲「manpath」。
   (h) 存檔後退出。

3.4.3 練習三

1. 在自己的家目錄建立一個新的目錄 zzz。
2. 進入目錄 zzz。
3. 下載檔案 viex3.txt
4. 將 viex3.txt 重新命名爲 vi3.txt
5. 使用 vi 或 vim 編輯 vi3.txt，執行以下動作：
   (a) 先到第 n 行，n 爲學號除 44 取餘數 +11。
   (b) 將下一行合併至這一行
   (c) 再到第 20 行，刪除這一行到第一行的資料。
   (d) 再到最後一行，往下貼上。
   (e) 再到第 13 行，在這行的最後加入文字'ABCDEabcde'。
   (f) 換行再加入'1234567890'。
   (g) 從第 30 行至第 55 行搜尋字串"man"，並將其取代爲"MAN"。
   (h) 存檔後退出。

Part II

系統管理 KVM8

Chapter 4

IPv4 網路設定

4.1 前言

1. 網路 (網路介面) 卡由系統 daemon NetworkManager 管理，包含：
   (a) device: 網路卡。
   (b) connection: 設定網卡。
   (c) 每一個 connection 都有一個名字及 UUID 做爲識別。
   (d) 目錄 /etc/sysconfig/network-scripts 下 ifcfg-name 爲網卡 name 的設定檔，可以使用命令 nmcli 產生或編輯網卡連線 (connection)。
   (e) /etc/sysconfig/network-scripts/ifcfg-name 網卡設定檔，還可以跟 v6 版本一樣，直接以 vim 手動編輯。

4.2 nmcli 命令

1. 查看 device 狀態，除了 lo 外只有一個網路介面 enp1s0。

```
[root@a10a10 ~]# nmcli device
DEVICE  TYPE      STATE      CONNECTION
enp1s0  ethernet  connected  enp1s0
lo      loopback  unmanaged  --
```

2. 查詢網路連線狀態，網路連線 enp1s0 使用網路界面 enp1s0。

```
[root@a10a10 ~]# nmcli connection
NAME    UUID                                  TYPE      DEVICE
enp1s0  2a4b4018-d130-405b-8647-6f3a1744b525  ethernet  enp1s0
```

4.3 修改網卡設定

1. 不必刪除設定，直接使用 modify 選項變更設定。設定 ipv4 位址 192.168.122.8、遮罩 255.255.255.0、GATEWAY 192.168.122.1 及 DNS 192.168.122.1，最後一定要設定 ipv4.method 爲 manual，將網路連線設定爲手動，也就是自行設定 IP，不是由 DHCP 自動取得，否則原 DHCP 取得的 IP 還會存在。

```
[root@a10a10 ~]# nmcli connection modify enp1s0 \
ipv4.addresses 192.168.122.8/24 \
ipv4.gateway 192.168.122.1 \
ipv4.dns 192.168.122.1 \
ipv4.method manual
```

2. 將網路連線 enp1s0 啓動。

```
[root@a10a10 ~]# nmcli connection up enp1s0
Connection successfully activated (D-Bus active path: /org/freedesktop/
    NetworkManager/ActiveConnection/5)
```

3. 查看網卡設定，看手動設定是否正確？

```
[root@a10a10 ~]# nmcli connection show enp1s0 | grep -iE ipv?4
ipv4.method:                            manual
ipv4.dns:                               192.168.122.1
ipv4.dns-search:                        --
ipv4.dns-options:                       --
ipv4.dns-priority:                      0
ipv4.addresses:                         192.168.122.8/24
ipv4.gateway:                           192.168.122.1
ipv4.routes:                            --
ipv4.route-metric:                      -1
ipv4.route-table:                       0 (unspec)
ipv4.routing-rules:                     --
ipv4.ignore-auto-routes:                no
ipv4.ignore-auto-dns:                   no
ipv4.dhcp-client-id:                    --
ipv4.dhcp-iaid:                         --
ipv4.dhcp-timeout:                      0 (default)
ipv4.dhcp-send-hostname:                yes
ipv4.dhcp-hostname:                     --
ipv4.dhcp-fqdn:                         --
ipv4.dhcp-hostname-flags:               0x0 (none)
ipv4.never-default:                     no
ipv4.may-fail:                          yes
ipv4.required-timeout:                  -1 (default)
```

```
ipv4.dad-timeout:                       -1 (default)
ipv4.dhcp-vendor-class-identifier:      --
ipv4.dhcp-reject-servers:               --
IP4.ADDRESS[1]:                         192.168.122.8/24
IP4.GATEWAY:                            192.168.122.1
IP4.ROUTE[1]:                           dst = 192.168.122.0/24, nh =
    0.0.0.0, mt = 100
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh =
    192.168.122.1, mt = 100
IP4.DNS[1]:                             192.168.122.1
```

4.4 * 刪除重新設定

1. 刪除網路連線 System enp1s0。

```
[root@a10a10 ~]# nmcli connection delete enp1s0
Connection 'enp1s0' (f03ca672-c3f6-433a-b336-832566e18e76) successfully
    deleted.
```

2. 如果是用 ssh 連線，這時會斷線，必須回到虛擬機終端機重新「新增」網路連線。

```
[root@a10a10 ~]# nmcli connection add \
con-name enp1s0 ifname enp1s0 type ethernet \
ipv4.addresses 192.168.122.8/24 \
ipv4.gateway 192.168.122.1 \
ipv4.dns 192.168.122.1 \
ipv4.method manual
Connection 'enp1s0' (a7b3ec85-0fa4-4bf3-b775-da87d43deacc) successfully
    added.
```

3. 查看網路連線，有 enp1s0 及 virbr0。

```
[root@a10a10 ~]# nmcli connection
NAME    UUID                                  TYPE      DEVICE
enp1s0  2a4b4018-d130-405b-8647-6f3a1744b525  ethernet  enp1s0
```

4. 將網路連線 enp1s0 啓動。

```
[root@a10a10 ~]# nmcli connection up enp1s0
```

```
Connection successfully activated (D-Bus active path: /org/freedesktop/
    NetworkManager/ActiveConnection/5)
```

5. 查看新增的網路連線設定 enp1s0 是否正確？是否生效？

```
[root@a10a10 ~]# nmcli connection show enp1s0 | grep -iE ipv?4
ipv4.method:                            manual
ipv4.dns:                               192.168.122.1
ipv4.dns-search:                        --
ipv4.dns-options:                       --
ipv4.dns-priority:                      0
ipv4.addresses:                         192.168.122.8/24
ipv4.gateway:                           192.168.122.1
ipv4.routes:                            --
ipv4.route-metric:                      -1
ipv4.route-table:                       0 (unspec)
ipv4.routing-rules:                     --
ipv4.ignore-auto-routes:                no
ipv4.ignore-auto-dns:                   no
ipv4.dhcp-client-id:                    --
ipv4.dhcp-iaid:                         --
ipv4.dhcp-timeout:                      0 (default)
ipv4.dhcp-send-hostname:                yes
ipv4.dhcp-hostname:                     --
ipv4.dhcp-fqdn:                         --
ipv4.dhcp-hostname-flags:               0x0 (none)
ipv4.never-default:                     no
ipv4.may-fail:                          yes
ipv4.required-timeout:                  -1 (default)
ipv4.dad-timeout:                       -1 (default)
ipv4.dhcp-vendor-class-identifier:      --
ipv4.dhcp-reject-servers:               --
IP4.ADDRESS[1]:                         192.168.122.8/24
IP4.GATEWAY:                            192.168.122.1
IP4.ROUTE[1]:                           dst = 192.168.122.0/24, nh =
    0.0.0.0, mt = 100
IP4.ROUTE[2]:                           dst = 0.0.0.0/0, nh =
    192.168.122.1, mt = 100
IP4.DNS[1]:                             192.168.122.1
```

4.5 SSH 設定與連線

1. sshd 允許 root 登入的參數 PermitRootLogin 設定：

   (a) yes：允許

   (b) no：不允許

(c) prohibit-password (without-password)：不允許以輸入密碼的方式登入

(d) forced-commands-only：只允許在登入命令用公鑰授權登入。

2. PermitRootLogin 預設是 prohibit-password，修改成 yes。

```
[root@a10a10 ~]# vim /etc/ssh/sshd_config
[root@a10a10 ~]# grep PermitRoot /etc/ssh/sshd_config
#PermitRootLogin prohibit-password
PermitRootLogin yes
# the setting of "PermitRootLogin without-password".
```

3. 重新啓動 sshd 服務。

```
[root@a10a10 ~]# systemctl restart sshd.service
```

4. 從 host 主機以 ssh 登入 kvm8。

```
[dywang@deyu ~]$ ssh root@kvm8.deyu.wang
The authenticity of host 'kvm8.deyu.wang (192.168.122.8)' can't be
    established.
ECDSA key fingerprint is SHA256:
    WaRiVLOS0JYkmNagaoEUu3MoCDEzqxougcjB0uJ99QQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'kvm8.deyu.wang' (ECDSA) to the list of known
     hosts.
root@kvm8.deyu.wang's password:
Last login: Mon Aug 29 15:35:12 2022 from 192.168.122.1
```

4.6 主機名稱設定

1. 遠端連線 192.168.122.8。

```
[dywang@deyu ~]$ ssh root@192.168.122.8
root@k192.168.122.8's password:
Last login: Mon Sep 19 10:51:26 2016 from deyu.wang
[root@a10a10 ~]#
```

2. 查看主機名稱爲 a10a10。

```
[root@a10a10 ~]# hostname
a10a10
```

3. 設定主機名稱爲 kvm8.deyu.wang。

```
[root@a10a10 ~]# hostnamectl set-hostname kvm8.deyu.wang
```

4. 查看主機名稱狀態。

```
[root@a10a10 ~]# hostname
kvm8.deyu.wang
```

5. hostnamectl 查看狀態。

```
[root@kvm8 ~]# hostnamectl status
     Static hostname: kvm8.deyu.wang
           Icon name: computer-vm
             Chassis: vm
          Machine ID: b920560a972a4f0d923db624ec350d8a
             Boot ID: 25dde91b7d28412cace4f6810c0e7555
        Product UUID: 125fc636-00b9-4beb-8f98-7abea52b9634
      Virtualization: kvm
    Operating System: AlmaLinux 10.0 (Purple Lion)
         CPE OS Name: cpe:/o:almalinux:almalinux:10::baseos
      OS Support End: Fri 2035-06-01
OS Support Remaining: 9y 7month 3w 1d
              Kernel: Linux 6.12.0-55.9.1.el10_0.x86_64
        Architecture: x86-64
     Hardware Vendor: Red Hat
      Hardware Model: KVM
    Firmware Version: 1.16.3-7.el10.alma.1
       Firmware Date: Tue 2014-04-01
        Firmware Age: 11y 6month 1w
```

4.7 實機練習

1. 查詢目前乙太網路

   (a) 查詢網卡名稱，將網卡名稱寫到 /root/netif。

   (b) 查詢網卡連線名稱，將連線名稱寫到 /root/netcon。

2. 設定目前乙太網路
   (a) ipv4 位址 192.168.122.8
   (b) 遮罩 255.255.255.0
   (c) GATEWAY 192.168.122.1
   (d) DNS 192.168.122.1
3. 設定主機名稱爲 kvm8.deyu.wang

Chapter 5

DNF 套件管理

5.1 前言

1. 何謂 RPM？
   (a) RPM 全名是『RedHat Package Manager』簡稱爲 RPM。
   (b) 由 Red Hat 公司發展出來，由於 RPM 使用方便，所以成了目前最熱門的套件管理程式。
   (c) RPM 是以資料庫記錄的方式來將所需要的套件安裝到 Linux 主機的一套管理程式。
   (d) RPM 將要安裝的套件先編譯過，並且打包好，安裝時 RPM 會先依照套件裡的紀錄資料查詢相依屬性套件是否滿足，若滿足則予以安裝，若不滿足則不予安裝。
2. RPM 套件的屬性相依
   (a) RPM 打包套件檔案時，會同時加入套件的訊息。例如：
      i. 版本、
      ii. 打包套件者、
      iii. 相依屬性的套件、
      iv. 套件的功能說明、
      v. 套件的所有檔案與目錄紀錄。
   (b) 當要安裝某個以 RPM 型態提供的套件時，如果資料庫顯示其相依套件不存在，則會顯示錯誤訊息。
   (c) 屬性相依的克服方式
      i. 手動下載並安裝好所有相依套件。

ii. 利用 dnf 安裝 rpm 套件。

3. DNF

(a) RHEL/AlmaLinux 都提供套件庫 (repository) 供 dnf 下載安裝、升級、移除。

(b) 在安裝好 RHEL/AlmaLinux 系統後，都會將 DNF 的套件庫設定到同版本的下載連結點。

(c) 利用 dnf 安裝 rpm 套件時，dnf 會在設定的套件庫尋找未安裝的相依套件加以安裝。

(d) 也可自行建立套件庫，設定爲 dnf 的套件庫，dnf 就可以升級，安裝，移除套件庫中的套件。

5.2 dnf 套件庫設定

1. 設定 dnf 套件庫 (repository) 設定檔，必須在 /etc/yum.repo.d/ 目錄下，建立.repo 檔案，檔案設定方式可查詢 dnf.conf 手冊。

```
[root@kvm8 ~]# man dnf.conf
```

2. 撰寫 /etc/yum.repos.d/dywang.repo，建立 AlmaLinux 10 BaseOS 及 AppStream 兩類套件。

```
[root@kvm8 ~]# vi /etc/yum.repos.d/redhat.repo
[root@kvm8 ~]# cat /etc/yum.repos.d/redhat.repo
[AppStream]
name=App Stream
baseurl=http://dywang.csie.cyut.edu.tw/alma10/AppStream
gpgcheck=0

[BaseOS]
name=Base OS
baseurl=http://dywang.csie.cyut.edu.tw/alma10/BaseOS
gpgcheck=0
```

3. .repo 檔參數說明：

(a) [base]：套件庫的名稱，中刮號一定要存在，名稱可以自訂但不可重複。

(b) name：說明這個套件庫的意義。

(c) mirrorlist=：列出這個套件庫可以使用的映射站台，如果不想使用，可以註解到這行；

(d) baseurl=：指定固定套件庫的實際網址。

(e) enable=1：預設啓動套件庫，可以不設定。如果不想啓動可以使用 enable=0。

(f) gpgcheck=1：指定是否需要查閱 RPM 檔案內的數位簽章。

(g) gpgkey=：數位簽章的公鑰檔所在位置。

5.3 DNF 套件管理指令

1. dnf [Tab][Tab] 提示 dnf 工具有 112 可命令可用。

```
[root@kvm8 ~]# dnf [Tab][Tab]
Display all 112 possibilities? (y or n)
```

2. 列出 dnf 使用的所有套件庫

```
[root@kvm8 ~]# dnf repolist
repo id                                     repo name
AppStream                                   App Stream
BaseOs                                      Base OS
```

3. 搜尋 ftp 相關的套件

```
[root@kvm8 ~]# dnf search ftp
baseos                                          8.9 MB/s | 1.5 MB
    00:00
appstream                                        12 MB/s | 1.5 MB
    00:00
Last metadata expiration check: 0:00:01 ago on Wed 08 Oct 2025 02:54:42
    PM CST.
========================== Name & Summary Matched: ftp
    ==========================
ftp.x86_64 : The standard UNIX FTP (File Transfer Protocol) client
python3-requests-ftp.noarch : FTP transport adapter for python3-requests
syslinux-tftpboot.noarch : SYSLINUX modules in /tftpboot, available for
    network
                           : booting
tftp.x86_64 : The client for the Trivial File Transfer Protocol (TFTP)
tftp-server.x86_64 : The server for the Trivial File Transfer Protocol (
    TFTP)
vsftpd.x86_64 : Very Secure Ftp Daemon
```

```
============================== Name Matched: ftp
    ==============================
lftp.x86_64 : A sophisticated file transfer program
============================== Summary Matched: ftp
    ==============================
curl.x86_64 : A utility for getting files from remote servers (FTP, HTTP
    , and
            : others)
nbdkit-curl-plugin.x86_64 : HTTP/FTP (cURL) plugin for nbdkit
wget.x86_64 : A utility for retrieving files using the HTTP or FTP
    protocols
```

4. 試安裝 vsftpd？如果出現 Is this ok [y//N]: 表示設定無誤可以取得套件進行安裝。y 安裝、N 不安裝，輸入 n 不安裝退出。

```
[root@kvm8 ~]# dnf install vsftpd
Last metadata expiration check: 3:23:43 ago on Tue 30 Sep 2025 03:11:06
    PM CST.
Dependencies resolved.
================================================================================
 Package         Architecture    Version               Repository
              Size
================================================================================

Installing:
 vsftpd          x86_64          3.0.5-9.el10          AppStream
              171 k

Transaction Summary
================================================================================

Install  1 Package

Total download size: 171 k
Installed size: 344 k
Is this ok [y/N]: n
Operation aborted.
```

5.4 實機練習

1. 建立 DNF 安裝套件倉庫來源

   (a) http://dywang.csie.cyut.edu.tw/alma10/BaseOS

   (b) http://dywang.csie.cyut.edu.tw/alma10/AppStream

2. 取消 RPM 檔案數位簽章檢查

3. 試著安裝套件 vsftpd

Chapter 6

* 正規表示法

6.1 前言

- 什麼是正規表示法
  1. 處理字串的方法，以行爲單位進行字串的處理行爲。
  2. 透過一些特殊符號的輔助，讓使用者輕易的達到搜尋、取代，限定某特定字串的處理程序。
  3. 延伸的正規表示法：可以作群組的字串『或 (or)』及『且 (and)』的處理。
  4. 正規表示法與萬用字元不一樣，不可混淆。
- 正規表示法用途
  1. 系統管理員的可以透過『正規表示法』的功能，將登錄的資訊進行處理，僅取出『有問題』的資訊來進行分析。
  2. 一般支援正規表示法的軟體可利用其進行字串的處理，例如：『郵件伺服器』以『字串』的比對來過濾郵件。
  3. 軟體中輸入欄位的格式限制，例如：只能爲數字。
- vi, grep, awk ,sed 等工具皆支援正規表示法。

6.2 基礎正規表示法

- 重要特殊字元 (characters)

```
RE字符          意義與範例
^word    待搜尋的字串(word)在行首。
         #範例：grep -n '^#' re.txt
```

```
          搜尋行首爲 # 開始的那一行。
word$     待搜尋的字串(word)在行尾。
          #範例：grep -n '!$' re.txt
          將行尾爲 ! 的那一行列印出來。
  .       代表『任意一個』字符，一定是一個任意字符。
          #範例：grep -n 'e.e' re.txt
          搜尋的字串可以是 (eve) (eae) (eee) (e e)， 但不能僅有 (ee)。
          亦即 e 與 e 中間『一定』僅有一個字元，而空白字元也是字元。
  \       跳脫字符，將特殊符號的特殊意義去除。
          #範例：grep -n \' re.txt
          搜尋含有單引號 ' 的那一行。
  *       重複零個或多個的前一個 RE 字符
          #範例：grep -n 'ess*' re.txt
          找出含有 (es) (ess) (esss) 等等的字串。
\{n,m\}   連續 n 到 m 個的『前一個 RE 字符』
          若爲 \{n\} 則是連續 n 個的前一個 RE 字符，
          若是 \{n,\} 則是連續 n 個以上的前一個 RE 字符。
          #範例：grep -n 'go\{2,3\}g' re.txt
          在 g 與 g 之間有 2 個到 3 個的 o 存在的字串，亦即 (goog)(gooog)
  [ ]     在 [ ] 當中『謹代表一個待搜尋的字元』
          #範例：grep -n 'g[ld]' re.txt
          搜尋含有 (gl) 或 (gd) 的那一行
          #範例：grep -n '[0-9]' re.txt
          搜尋含有任意數字的那一行。
          在字元集合 [ ] 中的減號 - 是代表兩個字元之間的所有連續字元。
          [^]：^ 在 [ ] 內時， 代表的意義是『反向選擇』
          #範例：grep -n 'oo[^t]' re.txt
          搜尋的字串可以是 (oog) (ood) 但不能是 (oot)。
```

- 以 grep 擷取字串

1. 編輯 re.txt 純文字檔

```
[root@test root]# vi re.txt
"Open Source" is a good mechanism to develop programs.
apple is my favorite food.
Football game is not use feet only.
this dress doesn't fit me.
However, this dress is about $ 3183 dollars.
GNU is free air not free beer.
Her hair is very beauty.
I can't finish the test.
Oh! The soup taste good.
motorcycle is cheap than car.
This window is clear.
the symbol '*' is represented as start.
Oh! My god!
The gd software is a library for drafting programs.
You are the best is mean you are the no. 1.
The world  is the same with "glad".
I like dog.
```

```
19 google is the best tools for search keyword.
   goooooogle yes!
21 go! go! Let's go.
   # I am csie
```

2. 搜尋特定字串：從檔案 re.txt 中取得特定字串 the

```
[root@test root]# grep -n 'the' re.txt
2 8:I can't finish the test.
  12:the symbol '*' is represented as start.
4 15:You are the best is mean you are the no. 1.
  16:The world  is the same with "glad".
6 18:google is the best tools for search keyword.
```

3. 該行沒有字串 the 時，才顯示在螢幕上：

```
[root@test root]# grep -vn 'the' re.txt
2 # 螢幕上出現的行列爲除了 8,12,15,16,18 五行之外的其他行列。
```

4. 取得不論大小寫的字串 the：

```
[root@test root]# grep -in 'the' re.txt
2 8:I can't finish the test.
  9:Oh! The soup taste good.
4 12:the symbol '*' is represented as start.
  14:The gd software is a library for drafting programs.
6 15:You are the best is mean you are the no. 1.
  16:The world  is the same with "glad".
8 18:google is the best tools for search keyword.
```

5. 利用 [ ] 來搜尋集合字元

```
[root@test root]# grep -n 't[ae]st' re.txt
2 8:I can't finish the test.
  9:Oh! The soup taste good.
4 # [ ] 裡面不論有幾個字元，都謹代表某『一個』字元。
```

6. 搜尋有 oo 的字元：

```
[root@test root]# grep -n 'oo' re.txt
2 1:"Open Source" is a good mechanism to develop programs.
  2:apple is my favorite food.
4 3:Football game is not use feet only.
```

```
9:Oh! The soup taste good.
18:google is the best tools for search keyword.
19:gooooooogle yes!
```

7. 不想要 oo 前面有 g ，可以利用在集合字元的反向選擇 [^]

```
[root@test root]# grep -n '[^g]oo' re.txt
2:apple is my favorite food.
3:Football game is not use feet only.
18:google is the best tools for search keyword.
19:gooooooogle yes!
```

8. oo 前面不想要有小寫字元：

```
[root@test root]# grep -n '[^a-z]oo' re.txt
3:Football game is not use feet only.
```

9. 要求字串是數字與英文：[a-zA-Z0-9]

```
[root@test root]# grep -n '[0-9]' re.txt
5:However, this dress is about $ 3183 dollars.
15:You are the best is mean you are the no. 1.
```

10. 行首與行尾字元 ^ $：

```
[root@test root]# grep -n '^the' re.txt
12:the symbol '*' is represented as start.

[root@test root]# grep -n '^[a-z]' re.txt
2:apple is my favorite food.
4:this dress doesn't fit me.
10:motorcycle is cheap than car.
12:the symbol '*' is represented as start.
18:google is the best tools for search keyword.
19:gooooooogle yes!
```

11. 不要開頭是英文字母：

```
[root@test root]# grep -n '^[^a-zA-Z]' re.txt
1:"Open Source" is a good mechanism to develop programs.
20:# I am csie
```

12. 行尾結束爲小數點 (.) 的那一行：

```
[root@test root]# grep -n '\.$' re.txt
1:"Open Source" is a good mechanism to develop programs.
2:apple is my favorite food.
3:Football game is not use feet only.
4:this dress doesn't fit me.
10:motorcycle is cheap than car.
11:This window is clear.
12:the symbol '*' is represented as start.
15:You are the best is mean you are the no. 1.
16:The world  is the same with "glad".
17:I like dog.
18:google is the best tools for search keyword.
# 小數點具有其他意義，必須要使用跳脫字元(\)來加以解除其特殊意義。
# 第 5~9 行最後面也是 .？
```

13. Windows 的 notepad 會主動加上 ^M 作爲斷行的判斷。

```
[root@test root]# cat -A re.txt
However, this dress is about $ 3183 dollars.^M$
```

14. 找出哪一行是『空白行』?

```
[root@test root]# grep -n '^$' re.txt
21:

[root@test root]# cat /etc/syslog.conf
[root@test root]# grep -v '^$' /etc/syslog.conf | grep -v '^#'
```

15. 任意一個字元. 與重複字元 *

```
#『.』代表『絕對有一個任意字元』
[root@test root]# grep -n 'g..d' re.txt
1:"Open Source" is a good mechanism to develop programs.
9:Oh! The soup taste good.
16:The world  is the same with "glad".
#  『*』代表的是『重複 0 個或多個前面的 RE 字符』
[root@test root]# grep -n 'ooo*' re.txt
1:"Open Source" is a good mechanism to develop programs.
2:apple is my favorite food.
3:Football game is not use feet only.
9:Oh! The soup taste good.
18:google is the best tools for search keyword.
19:gooooooogle yes!
```

16. 字串開頭與結尾都是 g，但是兩個 g 之間僅能存在至少一個 o

```
# 亦即 gog, goog, gooog.... 等。
[root@test root]# grep -n 'goo*g' re.txt
18:google is the best tools for search keyword.
19:goooooogle yes!
```

17. 找出 g 開頭與 g 結尾的字串，當中的字元可有可無。(錯誤找法)

```
[root@test root]# grep -n 'g*g' re.txt
1:"Open Source" is a good mechanism to develop programs.
3:Football game is not use feet only.
9:Oh! The soup taste good.
13:Oh!  My god!
14:The gd software is a library for drafting programs.
16:The world  is the same with "glad".
17:I like dog.
18:google is the best tools for search keyword.
19:goooooogle yes!
# g*g 的內容是 g, gg, ggg, gggg。
```

18. 找出 g 開頭與 g 結尾的字串，當中的字元可有可無。

```
[root@test root]# grep -n 'g.*g' re.txt
1:"Open Source" is a good mechanism to develop programs.
14:The gd software is a library for drafting programs.
18:google is the best tools for search keyword.
19:goooooogle yes!
```

19. 要找出『任意數字』的行列：

```
[root@test root]# grep -n '[0-9][0-9]*' re.txt
5:However, this dress is about $ 3183 dollars.
15:You are the best is mean you are the no. 1.
# 使用 grep -n '[0-9]' re.txt 也可以得到相同的結果。
```

20. 限定連續 RE 字符範圍 {}：找到兩個 o 的字串。

```
[root@test root]# grep -n 'o\{2\}' re.txt
1:"Open Source" is a good mechanism to develop programs.
2:apple is my favorite food.
3:Football game is not use feet only.
9:Oh! The soup taste good.
18:google is the best tools for search keyword.
19:goooooogle yes!
```

21. 找出 g 後面接 2 到 5 個 o ，然後再接一個 g 的字串：

```
[root@test root]# grep -n 'go\{2,5\}g' re.txt
18:google is the best tools for search keyword.
```

22. 找出 2 個 o 以上的 goooo....g？除了可以是 gooo*g ，也可以是：

```
[root@test root]# grep -n 'go\{2,\}g' re.txt
18:google is the best tools for search keyword.
19:gooooooogle yes!
```

23. 找出 2 個 o 以上的 goooo....g，當找到一筆即停止。

```
[root@test root]# grep -m 1 'go\{2,\}g' re.txt
18:google is the best tools for search keyword.
19:gooooooogle yes!
```

- 『正規表示法的特殊字元』與『萬用字元』之差異：

字元或代表意義	萬用字元	正規表示法的特殊字元
*	0 至無限多個字元	重複 0 到多個的前一個 RE 字符
反向選擇	[!range]	[^range]

```
例題：不支援正規表示法的 ls 工具中
『ls -l * 』：代表列出任意檔名的檔案；
『ls -l a* 』：代表列出以 a 爲開頭的任何檔名的檔案。
在正規表示法中要找到含有以 a 爲開頭的檔案，必須搭配支援正規表示法的工具
   grep：
ls | grep -n '^a.*'
```

- 例題：以正規表示法顯示行號方式找出目錄 /etc 以下符合下列條件之檔案。
  1. 包含 boot 或 root 字串的檔案。
  2. 包含 b 開頭 t 結尾之字串的檔案。
  3. b 開頭 t 結尾且中間 1 個 o 以上之字串的檔案。
  4. 包含 b 開頭 t 結尾且中間 2 至 4 個 o 之字串的檔案。
  5. 包含行首爲 root 之字串的檔案。

6. 包含行尾爲 root 之字串的檔案。

- 例題：想要查出來檔案中含有! 與 > 的字行 (! 在正規表示法中並不是特殊字元)：

```
grep -n '[!>]' re.txt
```

6.3 延伸正規表示法

- grep 支援的是基礎型的正規表示法，延伸正規表示法 egrep 是 grep -E 的命令別名。

- 延伸型正規表示法之特殊符號

```
RE 字符          意義與範例
  +       重複『一個或一個以上』的前一個 RE 字符
          #範例：egrep -n 'go+d' re.txt
          搜尋 (god) (good) (goood)... 等等的字串。
  ?       『零個或一個』的前一個 RE 字符
          #範例：egrep -n 'go?d' re.txt
          搜尋 (gd) (god) 這兩個字串。
  |       用或( or )的方式找出數個字串
          #範例：egrep -n 'gd|good|dog' re.txt
          搜尋 gd 、 good 或 dog 這三個字串。
 ( )      找出『群組』字串或作爲『多個重複群組』的判別
          #範例：egrep -n 'g(la|oo)d'
          搜尋 (glad) 或 (good) 這兩個字串。
          #範例：egrep -n 'A(xyz)+C' re.txt
          找開頭是 A 結尾是 C ，中間有一個以上的 "xyz" 字串。
```

- 例題：去除空白行與行首爲 # 的行列

```
#基礎型的正規表示法需要使用到管線命令來搜尋兩次
grep -v '^$' re.txt | grep -v '^#'
#支援延伸型正規表示法可以簡化爲：
egrep -v '^$|^#' re.txt
```

- 例題：以延伸型正規表示法顯示行號方式，找出目錄 /etc 以下符合下列條件之檔案。

1. 包含 boot 或 root 字串的檔案。

2. 包含 b 開頭 t 結尾且中間 0 個或 1 個 o 之字串的檔案。
3. b 開頭 t 結尾且中間 1 個 o 以上之字串的檔案。

6.4 實機練習題

6.4.1 練習一

1. .txt 檔案存檔都存到 /tmp/$mysid.$evalname 目錄，目錄若不存在，必須自行建立，其中變數 $mysid 爲自己的學號，$evalname 爲評分程式名稱不加附檔名。

2. 下載檔案 re1.txt
3. 使用 grep 對 re1.txt 執行以下搜尋動作，不需要列印行號，結果分別導向到檔案 myre1.1.txt 至 myre1.3.txt，不要做任何的更動。
   (a) 導向到 myre1.1.txt
      i. you 其中 y 大小寫都可以。
      ii. tast 或 test。
      iii. oo 前不是 g，也不是 t。
   (b) 導向到 myre1.2.txt
      i. 剛好三個阿拉伯數字。
      ii. 兩個以上阿拉伯數字。
   (c) 導向到 myre1.3.txt
      i. 行首是大寫英文字母。
      ii. 行尾不是'.' 句點。

6.4.2 練習二

1. .txt 檔案存檔都存到 /tmp/$mysid.$evalname 目錄，目錄若不存在，必須自行建立，其中變數 $mysid 爲自己的學號，$evalname 爲評分程式名稱不加附檔名。

2. 下載檔案 re2.txt
3. 使用 grep 對 re2.txt 執行以下搜尋動作，不需要列印行號，所有結果依序累加到檔案 myre2.txt，不要做任何的更動。

(a) 導向到 myre2.1.txt。

i. 開頭 b 結尾是 t，中間 1 個以上 a 的字串。

ii. 開頭 b 結尾是 t，中間 0 個或 1 個 a 的字串。

iii. dog, feet, good 三個字串。

(b) 導向到 myre2.2.txt。

i. boot 或 babt。

ii. 開頭 b 結尾是 t，中間 pqr 重複 1 次以上。

6.4.3 練習三

1. 在 kvm8 執行 nmcli connection show 目前的乙太網路連線設定，利用管線命令，以 grep 過濾 ipv?4' 大小寫不分，結果導向到 myipv4.txt。

Chapter 7

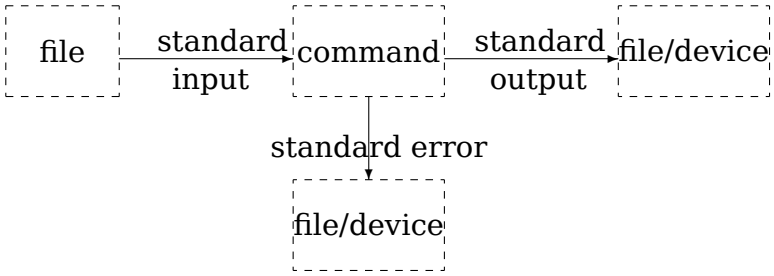
資料導向與管線處理

7.1 爲何要使用命令輸出重導向？

1. 儲存螢幕輸出的資訊；
2. 背景執行中的程式，不希望他干擾螢幕正常的輸出結果時；
3. 儲存系統例行命令（例如寫在 /etc/crontab 中的檔案）的執行結果；
4. 將已知的可能錯誤訊息丟掉『2> /dev/null 』；
5. 錯誤訊息與正確訊息需要分別輸出時。

7.2 資料流重導向

1. 指令執行過程之資料傳輸



2. 執行 cat /etc/crontab /etc/dywang，standard input 與 standard output 同時出現。

```
[dywang@dywHome ~]$ cat /etc/crontab /etc/dywang
# /etc/crontab 存在，螢幕出現以下 standard output
SHELL=/bin/bash

```

```
MAILTO=root
HOME=/

# run-parts
01 * * * * root nice -n 19 run-parts --report /etc/cron.hourly
02 4 * * * root nice -n 19 run-parts --report /etc/cron.daily
22 4 * * 0 root nice -n 19 run-parts --report /etc/cron.weekly
42 4 1 * * root nice -n 19 run-parts --report /etc/cron.monthly
# /etc/dywang 不存在，螢幕出現以下 standard error
cat: /etc/dywang: No such file or directory
```

3. 傳送指令

(a) 標準輸入 (stdin) ：代碼爲 0 ，使用 < 或 << ；

(b) 標準輸出 (stdout)：代碼爲 1 ，使用 > 或 >> ；

(c) 標準錯誤輸出 (stderr)：代碼爲 2 ，使用 2> 或 2>> ；

command	>, 1>	裝置或檔案	標準輸出至檔案，該檔案被覆蓋或建立。
	>>, 1>>		標準輸出至檔案，該檔案被建立或累加。
	2>		錯誤輸出至檔案，該檔案被覆蓋或建立。
	2>>		錯誤輸出至檔案，該檔案被建立或累加。
	<		輸入
	<<		結束的輸入字元

4. 檔案的建立方式

```
[root@linux ~]# ls -l /  >  ~/rootfile
# 將 ls -l / 要列出到螢幕上的資料『重新導向』到 ~/rootfile 檔案。
```

(a) 該檔案 (本例中是 ~/rootfile) 若不存在，系統會自動建立；

(b) 檔案存在時，系統會先將檔案內容清空再將資料寫入；

(c) 以 > 輸出到一個既存檔案中，該檔案會被覆蓋。

5. 下達 > 與 >>

```
# 將目前目錄下的檔案資訊儲存到 list.txt
[root@linux ~]# ls -al > list.txt

# 將根目錄下的資料累加到 list.txt
[root@linux ~]# ls -al / >> list.txt
```

6. 下達 2>

```
[csie@linux ~]$ find /home -name testing
find: /home/test1: Permission denied   <== Starndard error
find: /home/root: Permission denied    <== Starndard error
find: /home/masda: Permission denied   <== Starndard error
/home/csie/testing                   <== Starndard output

[csie@linux ~]$ find /home -name testing > list_right 2> list_error
[csie@linux ~]$ find /home -name testing > list_right 2> /dev/null
```

7. 下達 2>&1：標準輸出與標準錯誤輸出同時寫入同一個檔案

```
[csie@linux ~]$ find /home -name testing > list 2> list  <== 錯誤寫法
## 習題：請問上面命令的執行結果爲何？
[csie@linux ~]$ find /home -name testing > list 2>&1     <== 正確寫法
```

8. 使用 grep 命令過濾檔案 /usr/share/doc/autofs-5.0.7/README 中包含 configure 字串的所有行，共有 3 行。

```
[root@kvm7 ~]# grep configure /usr/share/doc/autofs-5.0.7/README
    ./configure
to configure the system.  See README.options for options that you can
give configure.
```

9. grep 命令過濾檔案 /usr/share/doc/autofs-5.0.7/README 中包含 configure 字串的 3 行原先顯示在螢幕上，現導向到檔案 /root/autofs.txt。

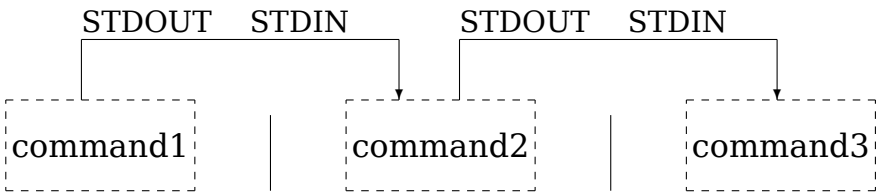
```
[root@kvm7 ~]# grep configure /usr/share/doc/autofs-5.0.7/README > /root
   /autofs.txt
```

10. 查看檔案 /root/autofs.txt 內容，與 grep 命令過濾檔案 /usr/share/doc/autofs-5.0.7/README 中包含 configure 字串的 3 行完全一樣。

```
[root@kvm7 ~]# cat /root/autofs.txt
    ./configure
to configure the system.  See README.options for options that you can
give configure.
```

7.3 管線命令 (pipe)

1. 管線命令的處理



2. 例題：使用 ls 指令輸出後的內容，被 less 讀取，並且利用 less 的功能前後翻動相關的資訊。

```
[root@linux ~]# ls -al /etc | less
```

7.4 實機練習

1. 資料導向

   (a) ls /etc/audit 導向到 /root/redirect1

   (b) ls /etcdy 錯誤導向到 /root/redirect2

   (c) ls /usr/local 導向到 /root/redirect3

   (d) ls /usr/etcdy 錯誤累加導向到 /root/redirect3

   (e) ls /bin/a* /lib64/autofs1 的 stdout 及 stderr 同時導向到 /root/redirect4

2. 管線命令過濾字串

   (a) 從檔案 /usr/share/doc/xz/README 中找出所有包含字串 info 的行列，導向到檔案 /root/info。

   (b) 從檔案 txtfile (例如 /usr/share/doc/systemd/README 中找出所有包含字串 str 的行列，導向到檔案 wlist。

   (c) 使用 nmcli 命令 show 目前的網路連線設定，經由管線命令一次找出所有包含 ipv4 IPV4 ip4 IP4 等字串的行，導向到檔案 myipv4。

Chapter 8

SELinux

8.1 前言

1. 自主式存取控制 (Discretionary Access Control, DAC)
   (a) 傳統的檔案權限與帳號關係：依據程序的擁有者與檔案資源的 rwx 權限來決定有無存取的能力。
   (b) 若該程序屬 root 權限，則其可在系統上進任何資源存取。
   (c) 若某個目錄權限爲 777，則任何帳號的程序都可任意存取及寫入。
2. 系統出現問題
   (a) 內部員工的資源誤用遠高於外部攻擊。
   (b) 員工資源誤用：系統管理員爲了自己方便，將防火牆完全關閉或某個檔案目錄的權限設定爲 777。
   (c) 爲控管員工資源誤用問題，美國國家安全局 (NSA) 開發安全增強式 Linux（SELinux, Security-Enhanced Linux）模組，並整合到 Linux 核心。
   (d) SELinux 是一種強制存取控制（mandatory access control, MAC）的實現。
3. 強制存取控制
   (a) 針對特定的程序與特定的檔案資源進行權限的控管。
   (b) WWW server 啓動的程序爲 httpd，selinux 預設僅能在/var/www 目錄下存取檔案。
4. 開啓 SELinux 出現的現象：
   (a) httpd 配置正常，防火牆也放行，但用户就是無法訪問？

(b) vsftpd 配置正常，防火牆也放行，但客户端訪問提示無法轉換家目錄？

(c) 關閉 SELinux 後一切正常，因此很多人對 SELinux 有了極大的偏見，認爲 SELinux 只有帶來麻煩，於是選擇在安裝系統後或出現問題時關閉 SELinux。

(d) 實際上，SELinux 在很多時候提供關鍵的保障，所以還是要開啓 SELinux，出現無法訪問等問題時，以修改 SELinux 安全上下文、布林值... 等方式來解決。

8.2 SELinux Contexts

1. Linux 執行程序或檔案會以 SELinux 上下文 (Contexts) 做標記，內容包含：SELinux user, role, type, level，以這些資訊來控制存取決定。

2. ls -Z 查詢檔案的 SELinux contexts，列出四個欄位 `user:role:type:level`，分別以冒號隔開。

```
[root@kvm8 ~]# ls -Z anaconda-ks.cfg
-rw-------. root root system_u:object_r:admin_home_t:s0 anaconda-ks.cfg
```

(a) user：使用者身份，以確認 SELinux 策略，Linux user 帳號會被對應到 SELinux user，執行以下命令可以列出兩者的對應，其中 `__default__` 表示登入的用户，一般是使用 useradd 增加的帳號；`system_u` 表示系統程序，由執行使用者可以確認身份。

```
[root@kvm8 ~]# semanage login -l
Login Name           SELinux User         MLS/MCS Range
    Service
__default__          unconfined_u         s0-s0:c0.c1023       *
root                 unconfined_u         s0-s0:c0.c1023       *
system_u             system_u             s0-s0:c0.c1023       *
```

(b) role：SELinux 以 Role-Based Access Control (RBAC) 控制存取，角色 (role) 決定是否可存取 domain，而每個 user 可以有多個角色，但同一時間只能有一個角色。

(c) type：指定資料型別，規則中定義何種程序型別訪問何種檔案。

(d) level：Multi-Level Security (MLS) and Multi-Category Security (MCS)，僅在 MLS 策略下才有用。

3. SELinux Contexts 四個欄位說明：

項目	用户 (user)	角色 (role)	類型 (type)	級別 (level)
程序	system_u 爲系統服務程序，受到管制。unconfined_u 爲不管制程序，通常都是用户自己開啓的，如 bash。	system_r 爲系統服務程序，受到管制。unconfined_r 爲不管制程序，通常都是用户自己開啓的，如 bash。	在 targeted 策略下，只有兩者的類型對應上（不代表相同），程序才能存取檔案。	MLS 安全級別，僅在 MLS 策略下才有意義。
檔案	system_u 爲系統服務創建的檔案。unconfined_u 爲用户自己創建的檔案。	均爲 object_r，表示一個檔案。	在 targeted 模式下，只有兩者的類型對應上了（不代表相同），程序才能存取檔案。	MLS 安全級別，僅在 MLS 模式下才有意義。

8.3 啓動、關閉與觀察

1. SELinux 開機啓動模式

```
[root@kvm8 ~]# vi /etc/selinux/config
[root@kvm8 ~]# cat /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
# See also:
# https://docs.fedoraproject.org/en-US/quick-docs/
#getting-started-with-selinux/#getting-started-with-selinux-selinux-
    states-and-modes
#
# NOTE: In earlier Fedora kernel builds, SELINUX=disabled would also
# fully disable SELinux during boot. If you need a system with SELinux
# fully disabled instead of SELinux running with no policy loaded, you
# need to pass selinux=0 to the kernel command line. You can use grubby
# to persistently set the bootloader to boot with selinux=0:
#
#    grubby --update-kernel ALL --args selinux=0
#
# To revert back to SELinux enabled:
#
#    grubby --update-kernel ALL --remove-args selinux
#
```

```
SELINUX=enforcing
# SELINUXTYPE= can take one of these three values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes
    are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

2. 觀察目前 SELinux 狀態，設定檔及目前狀態皆爲 enforcing。

```
[root@kvm8 ~]# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33
```

3. getenforce 取得目前 SELinux 狀態 Enforcing

```
[root@kvm8 ~]# getenforce
Enforcing
```

4. setenforce 關閉 SELinux

```
[root@kvm8 ~]# setenforce 0
```

5. getenforce 再查得目前 SELinux 狀態 Permissive

```
[root@kvm8 ~]# getenforce
Permissive
```

6. setenforce 再開啓 SELinux

```
[root@kvm8 ~]# setenforce 1
```

8.4 SELinux 基本用法

1. coreutils 的工具如 ps、ls 可以增加 Z 選項的方式取得 SELinux 的資訊。

```
[root@kvm8 ~]# ls -Z anaconda-ks.cfg
system_u:object_r:admin_home_t:s0 anaconda-ks.cfg
[root@kvm8 ~]# ps auxZ | grep apache
system_u:system_r:httpd_t:s0  apache  15694  0.0  0.2 215320  3040 ?  S
    Oct14   0:00 /usr/sbin/httpd -DFOREGROUND
```

2. chcon 修改檔案或目錄的 SELinux contexts。

```
[root@kvm8 ~]# chcon --help
Usage: chcon [OPTION]... CONTEXT FILE...
  or:  chcon [OPTION]... [-u USER] [-r ROLE] [-l RANGE] [-t TYPE] FILE
      ...
  -u, --user=USER        set user USER in the target security context
  -r, --role=ROLE        set role ROLE in the target security context
  -t, --type=TYPE        set type TYPE in the target security context
  -l, --range=RANGE      set range RANGE in the target security context
  -R, --recursive        operate on files and directories recursively
  -v, --verbose          output a diagnostic for every file processed
```

3. chcon 範例

```
[root@kvm8 ~]# ls -Z /etc/resolv.conf
-rw-r--r--. root root system_u:object_r:net_conf_t:s0  /etc/resolv.conf
[root@kvm8 ~]# cp /etc/resolv.conf .
[root@kvm8 ~]# ls -Z resolv.conf
-rw-r--r--. root root unconfined_u:object_r:admin_home_t:s0 resolv.conf
[root@kvm8 ~]# chcon -t net_conf_t resolv.conf
[root@kvm8 ~]# ls -Z resolv.conf
-rw-r--r--. root root unconfined_u:object_r:net_conf_t:s0 resolv.conf
```

4. restorecon 恢復檔案或目錄的 SELinux contexts 到預設值。

```
[root@kvm8 ~]# restorecon
usage:  restorecon [-iFnprRv0] [-e excludedir] pathname...
usage:  restorecon [-iFnprRv0] [-e excludedir] -f filename
```

5. restorecon 範例

```
[root@kvm8 ~]# restorecon resolv.conf
[root@kvm8 ~]# ls -Z resolv.conf
-rw-r--r--. root root unconfined_u:object_r:admin_home_t:s0 resolv.conf
```

6. getsebool 查詢 SELinux 布林值型規則的狀態。

```
[root@kvm8 ~]# getsebool
usage:  getsebool -a or getsebool boolean...
```

7. getsebool 範例

```
[root@kvm8 ~]# getsebool -a | grep ssh
fenced_can_ssh --> off
selinuxuser_use_ssh_chroot --> off
sftpd_write_ssh_home --> off
ssh_chroot_rw_homedirs --> off
ssh_keysign --> off
ssh_sysadm_login --> off
```

8. setsebool 開關 SELinux 布林值型規則。

```
[root@kvm8 ~]# setsebool
Usage:  setsebool [ -NPV ] boolean value | bool1=val1 bool2=val2...
```

9. setsebool 範例，沒加 -P 選項只是現在生效，重開機後會恢復到原設定值。

```
[root@kvm8 ~]# setsebool sftpd_write_ssh_home on
[root@kvm8 ~]# getsebool -a | grep ssh
fenced_can_ssh --> off
selinuxuser_use_ssh_chroot --> off
sftpd_write_ssh_home --> on
ssh_chroot_rw_homedirs --> off
ssh_keysign --> off
ssh_sysadm_login --> off
```

10. semanage 設定 SELinux 策略的某些元素，例如：檔案 context、程序埠號等。

```
[root@kvm8 ~]# semanage
usage: semanage [-h]
```

```
                {...,port,..,fcontext,...}
                ...
semanage is used to configure certain elements of SELinux policy with-
    out
requiring modification to or recompilation from policy source.

    port              Manage network port type definitions
    fcontext          Manage file context mapping definitions
    boolean           Manage booleans to selectively enable
        functionality
```

11. semanage 範例一：添加目錄的 SELinux context。

```
[root@kvm8 ~]# mkdir /usr/share/www
[root@kvm8 ~]# ls -Z /usr/share/www
[root@kvm8 ~]# ls -Zd /usr/share/www
drwxr-xr-x. root root unconfined_u:object_r:usr_t:s0   /usr/share/www
[root@kvm8 ~]# semanage fcontext -a -t httpd_sys_content_t "/usr/share/
    www(/.*)?"
[root@kvm8 ~]# ls -Zd /usr/share/www
drwxr-xr-x. root root unconfined_u:object_r:usr_t:s0   /usr/share/www
[root@kvm8 ~]# restorecon -Rv /usr/share/www
restorecon reset /usr/share/www context unconfined_u:object_r:usr_t:s0
->unconfined_u:object_r:httpd_sys_content_t:s0
[root@kvm8 ~]# ls -Zd /usr/share/www
drwxr-xr-x. root root unconfined_u:object_r:httpd_sys_content_t:s0 /usr/
    share/www
```

12. semanage 範例二：添加 httpd 程序允許訪問的 port。

```
[root@kvm8 ~]# semanage port -l | grep ^http_port
http_port_t      tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
[root@kvm8 ~]# semanage port -a -t http_port_t -p tcp 8989
[root@kvm8 ~]# semanage port -l | grep ^http_port
http_port_t      tcp      8989, 80, 81, 443, 488, 8008, 8009, 8443, 9000
```

13. semanage 範例三：刪除 httpd 程序允許訪問的 port。

```
[root@kvm8 html]# semanage port -d -t http_port_t -p tcp 8989
[root@kvm8 html]# semanage port -l | grep ^http_port
http_port_t       tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
```

8.5 Httpd Port

1. 查看 httpd 服務的狀態，enabled (開機啓動)，Active:failed (啓動失敗)。從狀態說明，查到原因是無法啓動在 82 port。

```
[root@kvm8 ~]# systemctl status httpd.service
× httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled;
         preset: disabled)
     Active: failed (Result: exit-code) since Thu 2025-10-09 13:36:57
         CST; 21min ago
 Invocation: 9032bd5315bf4dfc99ab9ee27cfd05d2
       Docs: man:httpd.service(8)
    Process: 800 ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND (code=
        exited, status=1/FAILURE)
   Main PID: 800 (code=exited, status=1/FAILURE)
     Status: "Reading configuration..."
   Mem peak: 7.6M
        CPU: 123ms

Oct 09 13:36:56 kvm8.deyu.wang systemd[1]: Starting httpd.service -
The Apache HTTP Server...
Oct 09 13:36:56 kvm8.deyu.wang (httpd)[800]: httpd.service: Referenced
but unset environment variable evaluates to an empty string: OPTIONS
Oct 09 13:36:57 kvm8.deyu.wang httpd[800]: AH00558: httpd: Could not
reliably determine the server's fully qualified domain name,
using kvm8.deyu.wang. Set the 'ServerNam>
Oct 09 13:36:57 kvm8.deyu.wang httpd[800]: (13)Permission denied:
    AH00072:
make_sock: could not bind to address [::]:82
Oct 09 13:36:57 kvm8.deyu.wang httpd[800]: (13)Permission denied:
    AH00072:
make_sock: could not bind to address 0.0.0.0:82
Oct 09 13:36:57 kvm8.deyu.wang httpd[800]: no listening sockets
    available,
shutting down
Oct 09 13:36:57 kvm8.deyu.wang httpd[800]: AH00015: Unable to open logs
Oct 09 13:36:57 kvm8.deyu.wang systemd[1]: httpd.service: Main process
    exited,
code=exited, status=1/FAILURE
Oct 09 13:36:57 kvm8.deyu.wang systemd[1]: httpd.service: Failed with
    result
'exit-code'.
Oct 09 13:36:57 kvm8.deyu.wang systemd[1]: Failed to start httpd.service
    -
The Apache HTTP Server.
```

2. 查詢 http_port_t，82 port 不是 httpd 程序允許訪問的 port。

```
[root@kvm8 ~]# semanage port --list | grep http
```

```
http_cache_port_t              tcp      8080, 8118, 8123, 10001-10010
http_cache_port_t              udp      3130
http_port_t                    tcp      80, 81, 443, 488, 8008, 8009,
    8443, 9000
pegasus_http_port_t            tcp      5988
pegasus_https_port_t           tcp      5989
```

3. 新增 82 爲 httpd 服務允許訪問的埠號。

```
[root@kvm8 ~]# semanage port --add --type http_port_t --proto tcp 82
```

4. 再查詢 http_port_t，82 port 已經是 httpd 程序允許訪問的 port。

```
[root@kvm8 ~]# semanage port --list | grep http
http_cache_port_t              tcp      8080, 8118, 8123, 10001-10010
http_cache_port_t              udp      3130
http_port_t                    tcp      82, 80, 81, 443, 488, 8008,
    8009, 8443, 9000
pegasus_http_port_t            tcp      5988
pegasus_https_port_t           tcp      5989
```

5. 再重新啓動 httpd 服務，成功啓動服務。

```
[root@kvm8 ~]# systemctl restart httpd.service
```

6. 再查看 httpd 服務的狀態，enabled (開機啓動)，Active: active (running)。

```
[root@kvm8 ~]# systemctl status httpd.service
 httpd.service - The Apache HTTP Server
    Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled;
        preset: disabled)
    Active: active (running) since Thu 2025-10-09 14:04:50 CST; 2min 12
        s ago
 Invocation: 2b1a669432084054ba07537add6dcdc5
      Docs: man:httpd.service(8)
  Main PID: 1617 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec:
        0; Bytes served/sec:   0 B/sec"
     Tasks: 177 (limit: 10692)
    Memory: 13.1M (peak: 13.3M)
       CPU: 607ms
    CGroup: /system.slice/httpd.service
            1617 /usr/sbin/httpd -DFOREGROUND
            1618 /usr/sbin/httpd -DFOREGROUND
```

```
15              1619 /usr/sbin/httpd -DFOREGROUND
                1621 /usr/sbin/httpd -DFOREGROUND
17              1632 /usr/sbin/httpd -DFOREGROUND

19 Oct 09 14:04:50 kvm8.deyu.wang systemd[1]: Starting httpd.service -
   The Apache HTTP Server...
21 Oct 09 14:04:50 kvm8.deyu.wang (httpd)[1617]: httpd.service: Referenced
   but unset environment variable evaluates to an empty string: OPTIONS
23 Oct 09 14:04:50 kvm8.deyu.wang httpd[1617]: Server configured,
   listening on: port 82
25 Oct 09 14:04:50 kvm8.deyu.wang systemd[1]: Started httpd.service -
   The Apache HTTP Server.
```

8.6 Httpd SELinux Context

1. 查看網頁伺服器根目錄下有 file1, file2, file3 三個檔案，權限是任何人都可讀取。

```
[root@kvm8 ~]# ll /var/www/html/
2 total 12
-rw-r--r--. 1 root root 10 Sep 27 17:10 file1
4 -rw-r--r--. 1 root root 10 Sep 27 17:10 file2
-rw-r--r--. 1 root root 10 Sep 27 17:10 file3
```

2. 使用 curl 訪問網頁 file1，回應沒有權限存取，而 file2 及 file3 成功存取。

```
1 [root@kvm8 ~]# curl http://localhost:82/file1
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
3 <html><head>
<title>403 Forbidden</title>
5 </head><body>
<h1>Forbidden</h1>
7 <p>You don't have permission to access this resource.</p>
</body></html>
9 [root@kvm8 ~]# curl http://localhost:82/file2
web test2
11 [root@kvm8 ~]# curl http://localhost:82/file3
web test3
```

3. 使用 ls 加 -Z 選項查看 file1，file2 及 file3 的 selinux fcontext type，file1 是 admin_home_t，而 file2 及 file3 是 httpd_sys_content_t，所以 SELinux 策略不允許 httpd 存取 file1。

```
[root@kvm8 ~]# ll -Z /var/www/html/
total 12
-rw-r--r--. 1 root root system_u:object_r:admin_home_t:s0        10 Sep
    27 17:10 file1
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
    27 17:10 file2
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
    27 17:10 file3
```

4. 使用 restorecon 還原 file1 的 selinux fcontext type。

```
[root@kvm8 ~]# restorecon -v /var/www/html/file1
```

5. 查看 file1 的 selinux fcontext type 沒有變化。

```
[root@kvm8 ~]# ll -Z /var/www/html/
total 12
-rw-r--r--. 1 root root system_u:object_r:admin_home_t:s0        10 Sep
    27 17:10 file1
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
    27 17:10 file2
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
    27 17:10 file3
```

6. 使用 semanage 列出 selinux 對檔案 file1 的 fcontext 設定爲 admin_home_t，所以還原時不會變動。

```
[root@kvm8 ~]# semanage fcontext --list | grep file1
/var/www/html/file1        all files       system_u:object_r:
    admin_home_t:s0
```

7. 使用 semanage 刪除 selinux 對檔案 file1 的 fcontext 設定。

```
[root@kvm8 ~]# semanage fcontext --delete --type admin_home_t /var/www/
    html/file1
```

8. 再使用 semanage 列出 selinux 對檔案 file1 的 fcontext 設定，已不存在。

```
[root@kvm8 ~]# semanage fcontext --list | grep file1
```

9. 將 /var/www/html 目錄下所有檔案，恢復預設的 selinux contexts。

```
[root@kvm8 ~]# restorecon -v /var/www/html/file1
Relabeled /var/www/html/file1 from system_u:object_r:admin_home_t:s0 to
system_u:object_r:httpd_sys_content_t:s0
[root@kvm8 ~]# curl http://localhost:82/file1
web test1
```

10. 查看 SELinux Contexts，file1，file2 及 file3 的 selinux fcontext type 都是 httpd_sys_content_t。

```
[root@kvm8 ~]# ll -Z /var/www/html/
total 12
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
   27 17:10 file1
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
   27 17:10 file2
-rw-r--r--. 1 root root system_u:object_r:httpd_sys_content_t:s0 10 Sep
   27 17:10 file3
```

11. 使用 curl 訪問網頁 file1，成功回應檔案內容。

```
[root@kvm8 ~]# curl http://localhost:82/file1
web test1
```

8.7 實機練習

1. 解決 kvm8 httpd 網頁服務啓動問題。
   (a) 查看 kvm8 網頁服務是否啓動？若無法啓動查看爲何無法啓動？
   (b) 依錯誤訊息解決
   (c) 重啓網頁 httpd 服務必須成功啓動。
2. 解決 kvm8 httpd 網頁根目錄下 file1 file2 檔案存取問題。
   (a) 試著連線 kvm8 網頁根目錄下的所有檔案，看看是否都可連線存取？
   (b) 若無法連線存取，請解決問題。

Chapter 9

群組管理

9.1 簡介

1. 群組的 ID(Group ID, GID)，/etc/group 存所有群組，每一行以『:』分隔，共分爲四項：

```
[root@kvm8 ~]# head -4 /etc/group
root:x:0:
bin:x:1:
daemon:x:2:
sys:x:3:
```

   (a) 群組名稱。

   (b) 群組密碼：密碼記錄在 /etc/gshadow，很少使用群組登入，故通常不需要設定。

   (c) GID。

   (d) 支援的帳號名稱：即群組裡的所有帳號。

2. 主要群組 (primary group)：

   (a) 使用者一登入系統就擁有的群組權限，其記錄在 /etc/passwd 的第四欄位。

   (b) 不需要在 /etc/group 的第四個欄位寫入該帳號。

3. 附屬群組 (supplementary group)：

   (a) 該帳號在 /etc/group 中所屬的群組（第四個欄位中包含該帳號）。

   (b) 一個帳號可同時支援主要群組及附屬群組，只要這兩群組擁有的功能該帳號皆可使用。

(c) 新建的檔案或目錄依附屬群組設定權限。

9.2 增加群組

1. 查看 group 開頭的命令。

```
[root@kvm8 ~]# group[TAB][TAB]
groupadd   groupdel   groupmems  groupmod   groups
```

2. 增加 sharegrp 群組。

```
[root@kvm8 ~]# groupadd sharegrp
```

3. 查詢 sharegrp 群組是否存在？

```
[root@kvm8 ~]# getent group sharegrp
sharegrp:x:1000:
```

9.3 刪除群組

1. 刪除 sharegrp 群組。

```
[root@kvm8 ~]# groupdel sharegrp
```

2. 再查詢 sharegrp 群組，已不存在。

```
[root@kvm8 ~]# getent group sharegrp
```

9.4 群組更名

1. 增加 deyugrp 群組。

```
[root@kvm8 ~]# groupadd deyugrp
```

2. 查詢 deyugrp 群組。

```
[root@kvm8 ~]# getent group deyugrp
deyugrp:x:1000:
```

3. 使用 groupmod 變更群組名稱，先查看輔助說明，選項 -n 可以變更群組名稱。

```
[root@kvm8 ~]# groupmod --help
Usage: groupmod [options] GROUP

Options:
  -a, --append                  append the users mentioned by -U option
      to the group
                                without removing existing user members
  -g, --gid GID                 change the group ID to GID
  -h, --help                    display this help message and exit
  -n, --new-name NEW_GROUP      change the name to NEW_GROUP
  -o, --non-unique              allow to use a duplicate (non-unique)
      GID
  -p, --password PASSWORD       change the password to this (encrypted)
                                PASSWORD
  -R, --root CHROOT_DIR         directory to chroot into
  -P, --prefix PREFIX_DIR       prefix directory where are located the /
      etc/* files
  -U, --users USERS             list of user members of this group
```

4. 使用 groupmod 將群組 deyugrp 變更爲 sharegrp。

```
[root@kvm8 ~]# groupmod -n sharegrp deyugrp
```

5. 查詢 deyugrp 群組。

```
[root@kvm8 ~]# getent group sharegrp
sharegrp:x:1000:
```

9.5 實機練習

1. 群組管理

   (a) 新增群組 shareg

(b) 變更群組名稱：shareg 改成 sharegrp

(c) 新增群組 sharey，再刪除，看看是否成功？

Chapter 10

帳號與身份管理

10.1 簡介

1. 帳號與 ID
   (a) 帳號 ID：使用者的 ID(User ID, UID) 與群組的 ID(Group ID, GID)
   (b) Linux 主機只認識 UID 及 GID，不認識我們輸入的帳號名稱。
   (c) /etc/passwd 及 /etc/group 分別記錄 UID 及 GID 的對應。
2. 如何登入 Linux 主機？(系統必須讀取 /etc/passwd 與 /etc/shadow)
   (a) 先找尋 /etc/passwd 裡面是否有這個帳號？
      i. 如果沒有則跳出，
      ii. 如果有則將該帳號對應的 UID 與 GID 讀出，該帳號的家目錄與 shell 設定也一併讀出；
   (b) 核對密碼表：核對輸入的密碼與 /etc/shadow 對應的帳號與 UID 之密碼是否相符？
   (c) 如果一切正確，就進入 Shell 控管的階段。
3. /etc/passwd 每一行一個帳號，以『:』分隔成七個欄位。

```
[root@kvm8 ~]# head /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/usr/sbin/nologin
```

```
operator:x:11:0:operator:/root:/usr/sbin/nologin
```

4. /etc/passwd 七個欄位，以 root 爲例。

   (a) 帳號名稱：root

   (b) 密碼：x ，密碼雜湊存在 /etc/shadow

   (c) UID：0

ID 範圍	該 ID 使用者特性
0	系統管理員，只要將 UID 改爲 0 即可成爲系統管
0 ~ 99	保留給系統使用
100 ~ 999	保留給應用程式使用。
1000 ~ 65535	一般使用者的 UID 從 1000 編起。

   (d) 用户所屬主要群組的 GID：0，群組名稱存在 /etc/group。

   (e) 備註說明：Super User

   (f) 用户家目錄：/root

   (g) 登入使用的 Shell：/bin/bash

10.2 新增帳號

1. useradd：增加使用者。

```
[root@kvm8 ~]# useradd --help
Usage: useradd [options] LOGIN
       useradd -D
       useradd -D [options]

Options:
      --badname                 do not check for bad names
  -b, --base-dir BASE_DIR       base directory for the home directory of
      the
                                new account
      --btrfs-subvolume-home    use BTRFS subvolume for home directory
  -c, --comment COMMENT         GECOS field of the new account
  -d, --home-dir HOME_DIR       home directory of the new account
  -D, --defaults                print or change default useradd
      configuration
  -e, --expiredate EXPIRE_DATE  expiration date of the new account
  -f, --inactive INACTIVE       password inactivity period of the new
      account
  -F, --add-subids-for-system   add entries to sub[ud]id even when
      adding a system user
  -g, --gid GROUP               name or ID of the primary group of the
      new
```

```
                                account
  -G, --groups GROUPS           list of supplementary groups of the new
                                account
  -h, --help                    display this help message and exit
  -k, --skel SKEL_DIR           use this alternative skeleton directory
  -K, --key KEY=VALUE           override /etc/login.defs defaults
  -l, --no-log-init             do not add the user to the lastlog and
                                faillog databases
  -m, --create-home             create the user's home directory
  -M, --no-create-home          do not create the user's home directory
  -N, --no-user-group           do not create a group with the same name
       as
                                the user
  -o, --non-unique              allow to create users with duplicate
                                (non-unique) UID
  -p, --password PASSWORD       encrypted password of the new account
  -r, --system                  create a system account
  -R, --root CHROOT_DIR         directory to chroot into
  -P, --prefix PREFIX_DIR       prefix directory where are located the /
      etc/* files
  -s, --shell SHELL             login shell of the new account
  -u, --uid UID                 user ID of the new account
  -U, --user-group              create a group with the same name as the
       user
  -Z, --selinux-user SEUSER     use a specific SEUSER for the SELinux
      user mapping
      --selinux-range SERANGE   use a specific MLS range for the SELinux
          user mapping

參數：
-u:  指定帳號的 UID。
-n:  群組爲 users。
-g:  後接之群組爲「主要群組(primary group)」，變動 /etc/passwd。
-G:  後接之群組爲「附屬群組(supplementary group)」，變動 /etc/group。
-M:  不要建立使用者家目錄。
-m:  建立使用者家目錄。
-d:  指定某個目錄爲家目錄，而不使用預設目錄。
-s:  指定使用的 shell。
```

2. 新增帳號 deyu1，其附屬群組爲 sharegrp。

```
[root@kvm8 ~]# useradd --groups sharegrp deyu1
```

3. 新增帳號 deyu2，其附屬群組爲 sharegrp。

```
[root@kvm8 ~]# useradd --groups sharegrp deyu2
```

4. 新增帳號 deyu3，此帳號不能使用互動式 shell 登入。

```
[root@kvm8 ~]# useradd --shell /sbin/nologin deyu3
```

10.3 檢查帳號

1. 檢查帳號 deyu1 及 deyu2，附屬群組爲 sharegrp，帳號 deyu3 沒有附屬群組。

```
[root@kvm8 ~]# id deyu1
uid=1000(deyu1) gid=1001(deyu1) groups=1001(deyu1),1000(sharegrp)
[root@kvm8 ~]# id deyu2
uid=1001(deyu2) gid=1002(deyu2) groups=1002(deyu2),1000(sharegrp)
[root@kvm8 ~]# id deyu3
uid=1002(deyu3) gid=1003(deyu3) groups=1003(deyu3)
```

2. 檢查帳號 deyu3，使用的 shell 爲不能登入的 /sbin/nologin。

```
[root@kvm8 ~]# getent passwd deyu3
deyu3:x:1002:1003::/home/deyu3:/sbin/nologin
```

3. 新增帳號 deyu4，並指定其 uid 爲 3584。

```
[root@kvm8 ~]# useradd --uid 3584 deyu4
```

4. 檢查帳號 deyu4，uid 爲 3584。

```
[root@kvm8 ~]# id deyu4
uid=3584(deyu4) gid=3584(deyu4) groups=3584(deyu4)
```

10.4 * 刪除帳號

1. userdel 刪除帳號，先查輔助說明，選項 -r 可以一併刪除帳號的家目錄及信箱。

```
[root@kvm8 ~]# userdel --help
Usage: userdel [options] LOGIN

Options:
  -f, --force                   force some actions that would fail
      otherwise
                                e.g. removal of user still logged in
                                or files, even if not owned by the user
  -h, --help                    display this help message and exit
  -r, --remove                  remove home directory and mail spool
  -R, --root CHROOT_DIR         directory to chroot into
  -Z, --selinux-user            remove any SELinux user mapping for the
      user
```

2. userdel 刪除帳號 deyu1。

```
[root@kvm8 ~]# userdel deyu1
```

3. userdel 刪除帳號 deyu1 時，沒有使用選項 -r 一併刪除帳號的家目錄及信箱，所有 /home/deyu1 還存在，所屬帳號及群組都以 ID 顯示，因該 ID 的帳號及群組已不存在。

```
[root@kvm8 ~]# ll /home/
total 21
drwx------. 2  1000  1001  1024 Nov 23 11:26 deyu1
drwx------. 2 deyu2 deyu2  1024 Nov 23 11:26 deyu2
drwx------. 2 deyu3 deyu3  1024 Nov 23 11:26 deyu3
drwx------. 2 deyu4 deyu4  1024 Nov 23 11:27 deyu4
drwx------. 2 root  root  12288 Jul 10 11:53 lost+found
```

4. useradd 再新增帳號 deyu1 時，警告家目錄 /home/deyu1 存在，所以沒有複製任何檔案到家目錄。

```
[root@kvm8 ~]# useradd --groups sharegrp deyu1
useradd: warning: the home directory already exists.
Not copying any file from skel directory into it.
Creating mailbox file: File exists
```

5. 但查看 /home 目錄中的 deyu1 目錄，並不屬於帳號 deyu1，原因爲新增的帳號，其 UID 並不是 1001。

```
[root@kvm8 ~]# ll /home/
total 21
drwx------. 2  1000  1001  1024 Nov 23 11:26 deyu1
drwx------. 2 deyu2 deyu2  1024 Nov 23 11:26 deyu2
drwx------. 2 deyu3 deyu3  1024 Nov 23 11:26 deyu3
drwx------. 2 deyu4 deyu4  1024 Nov 23 11:27 deyu4
drwx------. 2 root  root  12288 Jul 10 11:53 lost+found
```

6. 若再使用 userdel 加選項 -r 刪除帳號及其家目錄，出現信箱 /var/spool/mail/deyu1 及 /home/deyu1 不屬於帳號 deyu1，無法刪除。

```
[root@kvm8 ~]# userdel -r deyu1
userdel: /var/spool/mail/deyu1 not owned by deyu1, not removing
userdel: /home/deyu1 not owned by deyu1, not removing
[root@kvm8 ~]# ll /home/
total 21
drwx------. 2  1000  1001  1024 Nov 23 11:26 deyu1
drwx------. 2 deyu2 deyu2  1024 Nov 23 11:26 deyu2
drwx------. 2 deyu3 deyu3  1024 Nov 23 11:26 deyu3
drwx------. 2 deyu4 deyu4  1024 Nov 23 11:27 deyu4
drwx------. 2 root  root  12288 Jul 10 11:53 lost+found
```

7. 使用 rm 加選項 -rf 強制刪除 /home/deyu1 目錄及其次目錄，刪除信箱 /var/spool/mail/deyu1。

```
[root@kvm8 ~]# rm -rf /home/deyu1
[root@kvm8 ~]# rm -f /var/spool/mail/deyu1
```

8. 再重新新增帳號 deyu1，正常產生家目錄 /home/deyu1 及信箱 /var/spool/mail/deyu1。

```
[root@kvm8 ~]# useradd --groups sharegrp deyu1
[root@kvm8 ~]# ll /home/
total 21
drwx------. 2 deyu1 deyu1  1024 Nov 23 11:34 deyu1
drwx------. 2 deyu2 deyu2  1024 Nov 23 11:26 deyu2
drwx------. 2 deyu3 deyu3  1024 Nov 23 11:26 deyu3
drwx------. 2 deyu4 deyu4  1024 Nov 23 11:27 deyu4
drwx------. 2 root  root  12288 Jul 10 11:53 lost+found
[root@kvm8 ~]# ll /var/spool/mail/
total 0
-rw-rw----. 1 deyu1 mail 0 Nov 23 11:34 deyu1
-rw-rw----. 1 deyu2 mail 0 Nov 23 11:26 deyu2
-rw-rw----. 1 deyu3 mail 0 Nov 23 11:26 deyu3
```

```
-rw-rw----. 1 deyu4 mail 0 Nov 23 11:27 deyu4
-rw-rw----. 1 rpc   mail 0 Jul 10 11:53 rpc
```

10.5 * 變更帳號

1. usermod 可以變更帳號，先查輔助說明。

   (a) -G：變更附屬群組。

   (b) -s：變更帳號登入時使用的 shell。

   (c) -u：變更帳號的 uid。

   (d) -l：變更帳號登入的名稱。

```
[root@kvm8 ~]# usermod --help
Usage: usermod [options] LOGIN

Options:
  -c, --comment COMMENT         new value of the GECOS field
  -d, --home HOME_DIR           new home directory for the user account
  -e, --expiredate EXPIRE_DATE  set account expiration date to
      EXPIRE_DATE
  -f, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -g, --gid GROUP               force use GROUP as new primary group
  -G, --groups GROUPS           new list of supplementary GROUPS
  -a, --append                  append the user to the supplemental
      GROUPS
                                mentioned by the -G option without
                                    removing
                                him/her from other groups
  -h, --help                    display this help message and exit
  -l, --login NEW_LOGIN         new value of the login name
  -L, --lock                    lock the user account
  -m, --move-home               move contents of the home directory to
      the
                                new location (use only with -d)
  -o, --non-unique              allow using duplicate (non-unique) UID
  -p, --password PASSWORD       use encrypted password for the new
      password
  -R, --root CHROOT_DIR         directory to chroot into
  -s, --shell SHELL             new login shell for the user account
  -u, --uid UID                 new UID for the user account
  -U, --unlock                  unlock the user account
  -Z, --selinux-user SEUSER     new SELinux user mapping for the user
      account
```

2. 變更帳號 deyu1 爲 linda。

```
[root@kvm8 ~]# usermod -l linda deyu1
[root@kvm8 ~]# id linda
uid=3585(linda) gid=3585(deyu1) groups=3585(deyu1),1000(sharegrp)
```

3. 變更帳號 linda 的 uid 爲 4000。

```
[root@kvm8 ~]# usermod -u 4000 linda
[root@kvm8 ~]# id linda
uid=4000(linda) gid=3585(deyu1) groups=3585(deyu1),1000(sharegrp)
```

4. 變更帳號 linda 的登入 shell 爲 /sbin/nologin。

```
[root@kvm8 ~]# usermod -s /sbin/nologin linda
[root@kvm8 ~]# getent passwd linda
linda:x:4000:3585::/home/deyu1:/sbin/nologin
```

5. 變更帳號 linda 的附屬群組爲 apache。

```
[root@kvm8 ~]# usermod -G apache linda
[root@kvm8 ~]# id linda
uid=4000(linda) gid=3585(deyu1) groups=3585(deyu1),48(apache)
```

10.6 設定使用者密碼

1. passwd 命令設定或變更密碼

```
[root@kvm8 ~]# passwd --help
Usage: passwd [options] [LOGIN]

Options:
  -a, --all                   report password status on all accounts
  -d, --delete                delete the password for the named
      account
  -e, --expire                force expire the password for the named
      account
  -h, --help                  display this help message and exit
  -k, --keep-tokens           change password only if expired
  -i, --inactive INACTIVE     set password inactive after expiration
                              to INACTIVE
```

```
  -l, --lock                  lock the password of the named account
  -n, --mindays MIN_DAYS      set minimum number of days before
      password
                              change to MIN_DAYS
  -q, --quiet                 quiet mode
  -r, --repository REPOSITORY change password in REPOSITORY repository
  -R, --root CHROOT_DIR       directory to chroot into
  -P, --prefix PREFIX_DIR     directory prefix
  -S, --status                report password status on the named
      account
  -u, --unlock                unlock the password of the named account
  -w, --warndays WARN_DAYS    set expiration warning days to WARN_DAYS
  -x, --maxdays MAX_DAYS      set maximum number of days before
      password
                              change to MAX_DAYS
  -s, --stdin                 read new token from stdin

參數：
--stdin  ：經由標準輸入取得密碼，配合管線處理，可大量更改帳號密碼。
```

2. 使用 --stdin 選項設定 deyu1, deyu2, deyu3, deyu4 的密碼。

```
[root@kvm8 ~]# echo 123qwe | passwd --stdin deyu1
BAD PASSWORD: The password is shorter than 8 characters
[root@kvm8 ~]# echo 123qwe | passwd --stdin deyu2
BAD PASSWORD: The password is shorter than 8 characters
[root@kvm8 ~]# echo 123qwe | passwd --stdin deyu3
BAD PASSWORD: The password is shorter than 8 characters
[root@kvm8 ~]# echo 123qwe | passwd --stdin deyu4
BAD PASSWORD: The password is shorter than 8 characters
```

3. ssh deyu1 測試密碼，登入成功。

```
[root@kvm8 ~]# ssh deyu1@kvm8
The authenticity of host 'kvm8 (192.168.122.8)' can't be established.
ED25519 key fingerprint is SHA256:UW9HYWrgRRWBjnpqAMPE+
    Nu4eM1K1FyU15owqZ5YVsw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'kvm8' (ED25519) to the list of known hosts.
deyu1@kvm8's password:
```

4. 登出 deyu1。

```
[deyu1@kvm8 ~]$
logout
```

```
3 Connection to kvm8 closed.
```

10.7 * 帳號家目錄問題

1. 刪除帳號 deyu1，但沒有下 -r 選項，原帳號 deyu1 的家目錄 /home/deyu1 還是存在，但擁有者不存在，所以只顯示原帳號 deyu1 的 UID 1003。

```
1 [root@kvm8 ~]# userdel deyu1
[root@kvm8 ~]# ll /home/
3 total 20
drwx------. 2  1003 deyu1    4096 Nov 14 05:48 deyu1
5 drwx------. 2 deyu2 deyu2    4096 Nov 14 05:50 deyu2
drwx------. 2 deyu3 deyu3    4096 Nov 13 14:33 deyu3
7 drwx------. 2 deyu4 deyu4    4096 Nov 14 05:50 deyu4
```

2. 刪除群組 deyu1，目錄 /home/deyu1 所屬的群組不存在，所以也只顯示原群組 deyu1 的 GID 1004。

```
1 [root@kvm8 ~]# groupdel deyu1
[root@kvm8 ~]# ll /home/
3 total 20
drwx------. 2  1003     1004 4096 Nov 14 05:48 deyu1
5 drwx------. 2 deyu2 deyu2    4096 Nov 14 05:50 deyu2
drwx------. 2 deyu3 deyu3    4096 Nov 13 14:33 deyu3
7 drwx------. 2 deyu4 deyu4    4096 Nov 14 05:50 deyu4
```

3. 重新產生帳號 deyu1，出現家目錄已存在，不把 skel 目錄的設定複製過去的警告訊息。

```
1 [root@kvm8 ~]# useradd --groups sharegrp deyu1
useradd: warning: the home directory already exists.
3 Not copying any file from skel directory into it.
```

4. 查看 /home/deyu1 的擁有者及所屬群組都不對。

```
1 [root@kvm8 ~]# ll /home/
total 20
3 drwx------. 2  1003     1004 4096 Nov 14 05:48 deyu1
drwx------. 2 deyu2 deyu2    4096 Nov 14 05:50 deyu2
5 drwx------. 2 deyu3 deyu3    4096 Nov 13 14:33 deyu3
```

```
drwx------. 2 deyu4 deyu4    4096 Nov 14 05:50 deyu4
```

5. chwon 變更 /home/deyu1 的擁有者及所屬群組爲 deyu1。

```
[root@kvm8 ~]# chown deyu1.deyu1 /home/deyu1/
```

6. 再查看 /home/deyu1 的擁有者及所屬群組都是 deyu1。

```
[root@kvm8 ~]# ll /home/
total 20
drwx------. 2 deyu1 deyu1    4096 Nov 14 05:48 deyu1
drwx------. 2 deyu2 deyu2    4096 Nov 14 05:50 deyu2
drwx------. 2 deyu3 deyu3    4096 Nov 13 14:33 deyu3
drwx------. 2 deyu4 deyu4    4096 Nov 14 05:50 deyu4
```

7. su 切換到帳號 deyu1，正常存取家目錄的環境變數。

```
[root@kvm8 ~]# su - deyu1
[deyu1@kvm8 ~]$ exit
logout
```

10.8 * 帳號整體參數變更

1. 變更用户名、群組名及其家目錄，查看家目錄，用户 deyu1 主要群組是 deyu1，家目錄是 /home/deyu1。

```
[root@kvm8 ~]# ll /home/
total 23
drwx------. 2 deyu1 deyu1     1024 Nov 22 20:23 deyu1
drwx------. 2 deyu2 deyu2     1024 Nov 23 07:43 deyu2
drwx------. 2 deyu3 deyu3     1024 Nov 22 19:21 deyu3
drwx------. 2 deyu4 deyu4     1024 Nov 23 07:45 deyu4
drwx------. 2 root  root     12288 Jul 10 13:46 lost+found
```

2. 變更群組名 deyu1 爲 linda，/home/deyu1 的群組由 deyu1 變更成 linda 了。

```
[root@kvm8 ~]# groupmod -n linda deyu1
[root@kvm8 ~]# ll /home/
total 23
drwx------. 2 deyu1 linda     1024 Nov 22 20:23 deyu1
drwx------. 2 deyu2 deyu2     1024 Nov 23 07:43 deyu2
drwx------. 2 deyu3 deyu3     1024 Nov 22 19:21 deyu3
drwx------. 2 deyu4 deyu4     1024 Nov 23 07:45 deyu4
drwx------. 2 root  root     12288 Jul 10 13:46 lost+found
```

3. 變更用户名 deyu1 爲 linda，/home/deyu1 的用户由 deyu1 變更成 linda 了。

```
[root@kvm8 ~]# usermod -l linda deyu1
[root@kvm8 ~]# ll /home/
total 23
drwx------. 2 linda linda     1024 Nov 22 20:23 deyu1
drwx------. 2 deyu2 deyu2     1024 Nov 23 07:43 deyu2
drwx------. 2 deyu3 deyu3     1024 Nov 22 19:21 deyu3
drwx------. 2 deyu4 deyu4     1024 Nov 23 07:45 deyu4
drwx------. 2 root  root     12288 Jul 10 13:46 lost+found
```

4. 變更用户名 linda 的家目錄名，由 deyu1 變爲 linda，/home/deyu1 變成 /home/linda。

```
[root@kvm8 ~]# mv /home/deyu1 /home/linda
[root@kvm8 ~]# ll /home/
total 23
drwx------. 2 deyu2 deyu2     1024 Nov 23 07:43 deyu2
drwx------. 2 deyu3 deyu3     1024 Nov 22 19:21 deyu3
drwx------. 2 deyu4 deyu4     1024 Nov 23 07:45 deyu4
drwx------. 2 linda linda     1024 Nov 22 20:23 linda
drwx------. 2 root  root     12288 Jul 10 13:46 lost+found
```

5. 變更 /home/deyu1 目錄名爲 /home/linda 後，切換身份爲 linda 卻找不到家目錄，原因爲 linda 的，/home/deyu1 變成 /home/linda。

```
[root@kvm8 ~]# su - linda
su: warning: cannot change directory to /home/deyu1: No such file or
    directory
-bash-4.2$ exit
logout
```

6. linda 找不到家目錄原因爲 linda 的家目錄，還是設定爲 /home/deyu1，必須變更爲 /home/linda。

```
[root@kvm8 ~]# usermod -d /home/linda linda
```

7. 再切換身份爲 linda，正常進入家目錄 /home/linda。

```
[root@kvm8 ~]# su - linda
Last login: Fri Nov 30 08:31:56 CST 2018 on pts/0
[linda@kvm8 ~]$ exit
logout
```

8. 其他用户 deyu2 deyu3 deyu4 也用同樣的方式改名。

10.9 實機練習

1. 建立用户帳號
   (a) deyu1 附屬群組爲 sharegrp
   (b) deyu2 附屬群組爲 sharegrp
   (c) deyu3 無法使用 shell 登入
   (d) deyu4 指定 uid 爲 3854
   (e) 四個帳號的密碼皆設定爲 123qwe
2. 帳號名稱、UID、附屬群組設定錯誤皆可使用 usermod 修改。
3. 若刪除帳號重新新增，刪除時必須連「家目錄」一併刪除，否則新增的帳號可能無法存取家目錄。

Chapter 11

su 與 sudo

11.1 su 切換使用者

1. su：轉換帳號身份。

```
[root@kvm8 ~]# su --help

Usage:
 su [options] [-] [<user> [<argument>...]]

Change the effective user ID and group ID to that of <user>.
A mere - implies -l.  If <user> is not given, root is assumed.

Options:
 -m, -p, --preserve-environment      do not reset environment variables
 -w, --whitelist-environment <list>  don't reset specified variables

 -g, --group <group>             specify the primary group
 -G, --supp-group <group>        specify a supplemental group

 -, -l, --login                  make the shell a login shell
 -c, --command <command>         pass a single command to the shell with
     -c
 --session-command <command>     pass a single command to the shell with
     -c
                                   and do not create a new session
 -f, --fast                      pass -f to the shell (for csh or tcsh)
 -s, --shell <shell>             run <shell> if /etc/shells allows it
 -P, --pty                       create a new pseudo-terminal
 -T, --no-pty                    do not create a new pseudo-terminal (
    bad security!)

 -h, --help                      display this help
 -V, --version                   display version

For more details see su(1).

參數：
```

```
-, -l, --login username ：變換身份爲 username，沒指定 username 則爲 root
    ，
32      取得一個新的 shell，且使用 username 的所有相關環境設定檔。
-m, -p, --preserve-environment ：使用目前的環境設定，而不重新讀取新使用
    者的設定檔。
34 -c, --command <command>  ：僅進行一次指令 command。
```

2. su - 登入 deyu1，使用 deyu1 的環境變數，所在目錄爲 deyu1 的家目錄。

```
[root@kvm8 ~]# su --login deyu1
2 Last login: Wed Oct 22 14:14:00 CST 2025 from 192.168.122.8 on pts/1
[deyu1@kvm8 ~]$ pwd
4 /home/deyu1
[deyu1@kvm8 ~]$
6 logout
```

3. su 切換爲帳號 deyu1，使用原帳號 root 的環境變數，所在目錄爲 root 的家目錄。

```
[root@kvm8 ~]# su deyu1
2 [deyu1@kvm8 root]$ pwd
/root
4 [deyu1@kvm8 root]$
exit
```

4. 由帳號 deyu1 使用 su 變換身份爲 root，使用原帳號 deyu1 的環境變數，所以所在目錄爲 deyu1 的家目錄。

```
1 [deyu1@kvm8 ~]$ su
Password:
3 [root@kvm8 deyu1]# pwd
/home/deyu1
5 [root@kvm8 deyu1]# exit
exit
```

5. 由帳號 deyu1 使用 su - 變換身份爲 root，使用帳號 root 的環境變數，所以所在目錄爲 root 的家目錄。

```
[deyu1@kvm8 ~]$ su -
2 Password:
Last login: Wed Oct 22 13:35:12 CST 2025 from 192.168.122.1 on pts/0
4 [root@kvm8 ~]# pwd
```

```
/root
[deyu1@kvm8 ~]$ exit
logout
```

11.2 sudo 以其他身份執行命令

1. sudo 命令可以其他帳號的權限執行命令，不指定帳號就是以 root 權限，執行時是輸入自己的密碼，不是 root 的密碼。

2. 可以使用 sudo 用 root 權限執行命令的帳號設定在 /etc/sudoers 檔案中，一般不會直接編輯 /etc/sudoers，而是執行 visudo 設定，格式如下：

```
帳號名稱     來源主機=(可切換帳號) 可執行的指令
%群組名稱     來源主機=(可切換帳號) 可執行的指令
```

3. 執行 sudo 時不用輸入密碼，可在「可執行的指令」前加上 NOPASSWD:。

```
帳號名稱     來源主機=(可切換帳號) NOPASSWD:可執行的指令
%群組名稱     來源主機=(可切換帳號) NOPASSWD:可執行的指令
```

11.3 sudo 帳號測試

1. 範例：visudo 編輯 /etc/sudoers，增加帳號 deyu9 可免密碼以 root 權限從任何主機執行任何命令。

```
[root@kvm8 ~]# visudo
[root@kvm8 ~]# grep deyu /etc/sudoers
deyu9    ALL=(ALL)    NOPASSWD:ALL
```

2. 新增帳號 deyu9

```
[root@kvm8 ~]# useradd deyu9
```

3. 登入爲 deyu9

```
[root@kvm8 ~]# su - deyu9
```

4. 一般帳號 deyu9 無法使用 hwclock 命令查詢硬體時間

```
[deyu9@kvm8 ~]$ hwclock
hwclock: Cannot access the Hardware Clock via any known method.
hwclock: Use the --verbose option to see the details of our search for
    an access method.
```

5. 一般帳號 deyu9 可以使用 sudo hwclock 查詢硬體時間

```
[deyu9@kvm8 ~]$ sudo hwclock
2025-10-22 14:41:12.572572+08:00
```

6. 登出 deyu9

```
[deyu9@kvm8 ~]$
logout
```

11.4 sudo 群組測試

1. 範例：visudo 編輯 /etc/sudoers，增加群組 dygrp 可以 root 權限從任何主機執行命令 /sbin/hwclock。

```
[root@kvm8 ~]# visudo
[root@kvm8 ~]# grep dygrp /etc/sudoers
%dygrp  ALL=(ALL)   /sbin/hwclock
```

2. 新增群組 dygrp，並新增帳號 deyu21，其附屬群組爲 dygrp。

```
[root@kvm8 ~]# groupadd dygrp
[root@kvm8 ~]# useradd --groups dygrp deyu21
```

3. 查詢帳號 deyu21，其附屬群組爲 dygrp。

```
[root@kvm8 ~]# id deyu21
uid=3586(deyu21) gid=3587(deyu21) groups=3587(deyu21),3586(dygrp)
```

4. 設定 deyu21 的密碼

```
[root@kvm8 ~]# echo qweqwe | passwd --stdin deyu21
BAD PASSWORD: The password is shorter than 8 characters
```

5. 登入 deyu21

```
[root@kvm8 ~]# su --login deyu21
```

6. 一般帳號 deyu21 無法使用 hwclock 命令查詢硬體時間

```
[deyu21@kvm8 ~]$ hwclock
hwclock: Cannot access the Hardware Clock via any known method.
hwclock: Use the --verbose option to see the details of our search for
    an access method.
```

7. 一般帳號 deyu21 附屬群組 dygrp，可以使用 sudo hwclock 查詢硬體時間。

```
[deyu21@kvm8 ~]$ sudo hwclock

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

For security reasons, the password you type will not be visible.

[sudo] password for deyu21:
2025-10-22 14:44:45.757316+08:00
```

11.5 實機練習

1. visudo 設定：deyu9, deyu21, dygrp, vgs 請依變數取代。

(a) 增加帳號 deyu9 可免密碼，從任何主機，切換任何帳號，執行任何命令。

(b) 增加群組 dygrp 可以從任何主機，切換任何帳號，需要密碼可執行 /sbin/vgs。

2. 建立測試帳號及群組

(a) 群組 dygrp

(b) 帳號 deyu9，不屬於 dygrp 群組。

(c) 帳號 deyu21，附屬群組 dygrp 群組。

3. 測試

(a) 切換帳號 deyu9

i. 執行 vgs，標準錯誤 (stderr) 導向到 `/tmp/vgs.9.err`。

ii. 執行 sudo vgs，標準輸出 (stdout) 導向到 `/tmp/vgs.9.out`。

(b) 切換帳號 deyu21。

i. 執行 vgs，標準錯誤 (stderr) 導向到 `/tmp/vgs.21.err`。

ii. 執行 sudo vgs，標準輸出 (stdout) 導向到 `/tmp/vgs.21.out`。

Chapter 12

帳號密碼策略

12.1 /etc/shadow

1. Linux 帳號表在 /etc/passwd，帳號密碼雜湊後存在 /etc/shadow，每行密碼有九個欄位，以冒號「:」分隔如下：

```
[root@kvm8 ~]# head /etc/shadow
root:$y$j9T$ivvQGLYWsGJFi/7Y.TBtNKE9$5Ha7kycsD::0:99999:7:::
bin:*:20244:0:99999:7:::
daemon:*:20244:0:99999:7:::
adm:*:20244:0:99999:7:::
lp:*:20244:0:99999:7:::
sync:*:20244:0:99999:7:::
shutdown:*:20244:0:99999:7:::
halt:*:20244:0:99999:7:::
mail:*:20244:0:99999:7:::
operator:*:20244:0:99999:7:::
```

2. 密碼檔九個欄位：

   (a) 帳號名稱，要與 /etc/passwd 相同才可。

   (b) 密碼：眞正的密碼但已加密，密碼欄第一個字爲「*」或者是「!」，表示這個帳號不會被用來登入。

   (c) 上次更動密碼的日期，計算 Linux 日期的時間是以 1970 年 1 月 1 日爲 0，而 1971 年 1 月 1 日則爲 365。數字查看方法：

      i. passwd 指令：-S 選項輸出帳號 bin 狀態，可以看到更動日期。

```
[root@kvm8 ~]# passwd -s bin
bin L 2025-06-05 0 99999 7 -1
```

ii. date 指令：以 ISO 8601 格式輸出 1970-01-01 加 20244 天的日期。

```
[root@kvm8 ~]# date -I -d '1970-01-01 + 20244 days'
2025-06-05
```

(d) 上次更動密碼的日期，計算 Linux 日期的時間是以 1970 年 1 月 1 日爲 0，而 1971 年 1 月 1 日則爲 365。

(e) 密碼需要重新變更的天數，99999 表示密碼不需要重新設定。

(f) 密碼需要變更前的警告：上例 7 表示 7 天內系統會警告帳號。

(g) 密碼過期的恕限時間：當密碼失效後，n 天內還可以登入。

(h) 帳號失效日期：計算方式同欄位 3。

(i) 保留：看以後有無新功能加入。

12.2 預設密碼過期

1. Linux 系統預設的密碼過期策略在 /etc/login.defs。

```
[root@kvm8 ~]# grep PASS_  /etc/login.defs
#   PASS_MAX_DAYS   Maximum number of days a password may be used.
# 密碼需要重新變更的天數，99999 表示密碼不需要重新設定。
#   PASS_MIN_DAYS   Minimum number of days allowed between password
    changes.
# 密碼不可被更動的天數，0 表示密碼隨時可以更動。
#   PASS_MIN_LEN    Minimum acceptable password length.
# 密碼最短長度，5 表示密碼不能少於5個字元。
#   PASS_WARN_AGE   Number of days warning given before a password
    expires.
# 密碼需要變更前的警告，7 表示7天之內系統會警告帳號。
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_MIN_LEN    8
PASS_WARN_AGE   7
PASS_CHANGE_TRIES   5
PASS_ALWAYS_WARN    yes
#PASS_MAX_LEN       8
```

2. 新增帳號 deyu11

```
[root@kvm8 ~]# useradd deyu11
```

3. chage 命令查詢變醱帳號密碼過期狀態。

```
[root@kvm8 ~]# chage --help
Usage: chage [options] LOGIN

Options:
  -d, --lastday LAST_DAY        set date of last password change to
      LAST_DAY
  -E, --expiredate EXPIRE_DATE  set account expiration date to
      EXPIRE_DATE
  -h, --help                    display this help message and exit
  -i, --iso8601                 use YYYY-MM-DD when printing dates
  -I, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -l, --list                    show account aging information
  -m, --mindays MIN_DAYS        set minimum number of days before
      password
                                change to MIN_DAYS
  -M, --maxdays MAX_DAYS        set maximum number of days before
      password
                                change to MAX_DAYS
  -R, --root CHROOT_DIR         directory to chroot into
  -P, --prefix PREFIX_DIR       directory prefix
  -W, --warndays WARN_DAYS      set expiration warning days to WARN_DAYS
```

4. 查詢帳號 deyu11 的密碼過期狀態，如預設策略。

```
[root@kvm8 ~]# chage --list deyu11
Last password change                    : Oct 21, 2025
Password expires                        : never
Password inactive                       : never
Account expires                         : never
Minimum number of days between password change      : 0
Maximum number of days between password change      : 99999
Number of days of warning before password expires   : 7
```

5. 查詢帳號 deyu11 在 /etc/shadow 的各欄位值，與 chage 查詢結果一樣。

```
[root@kvm8 ~]# getent shadow deyu11
deyu11:!:20382:0:99999:7:::
```

12.3 變更預設密碼過期策略

1. 修改預設密碼過期策略，密碼最長使用天數 30 天。

```
[root@kvm8 ~]# vi /etc/login.defs
[root@kvm8 ~]# grep ^PASS_ /etc/login.defs
PASS_MAX_DAYS   30
PASS_MIN_DAYS   0
PASS_MIN_LEN    8
PASS_WARN_AGE   7
PASS_CHANGE_TRIES   5
PASS_ALWAYS_WARN    yes
```

2. 新增帳號 deyu12

```
[root@kvm8 ~]# useradd deyu12
```

3. 查詢帳號 deyu12 的密碼過期狀態，如預設策略，最長使用天數 30 天。

```
[root@kvm8 ~]# chage --list deyu12
Last password change                    : Oct 21, 2025
Password expires                        : Nov 20, 2025
Password inactive                       : never
Account expires                         : never
Minimum number of days between password change      : 0
Maximum number of days between password change      : 30
Number of days of warning before password expires   : 7
```

4. 查詢帳號 deyu12 在 /etc/shadow 的各欄位值，與 chage 查詢結果一樣。

```
[root@kvm8 ~]# getent shadow deyu12
deyu12:!:20382:0:30:7:::
```

12.4 設定已存在帳號密碼過期狀態

1. 設定帳號 deyu11 密碼 20 天後必須變更。

```
[root@kvm8 ~]# chage --maxdays 20 deyu11
[root@kvm8 ~]# chage --list deyu11
Last password change                    : Oct 21, 2025
Password expires                        : Nov 10, 2025
Password inactive                       : never
Account expires                         : never
Minimum number of days between password change      : 0
Maximum number of days between password change      : 20
```

```
Number of days of warning before password expires   : 7
```

2. 設定帳號 deyu11 密碼 5 天內不能變更。

```
[root@kvm8 ~]# chage --mindays 5 deyu11
[root@kvm8 ~]# chage --list deyu11
Last password change                                : Oct 21, 2025
Password expires                                    : Nov 10, 2025
Password inactive                                   : never
Account expires                                     : never
Minimum number of days between password change      : 5
Maximum number of days between password change      : 20
Number of days of warning before password expires   : 7
```

3. 設定帳號 deyu11 密碼過期前 6 天警告。

```
[root@kvm8 ~]# chage --warndays 6 deyu11
[root@kvm8 ~]# chage --list deyu11
Last password change                                : Oct 21, 2025
Password expires                                    : Nov 10, 2025
Password inactive                                   : never
Account expires                                     : never
Minimum number of days between password change      : 5
Maximum number of days between password change      : 20
Number of days of warning before password expires   : 6
```

12.5 實機練習

1. 設定預設密碼過期略策，也就是新增的帳號，其密碼會有以下策略。
   (a) 最長不用變更天數 30
   (b) 可以變更的最短天數 4
   (c) 密碼過期前 6 天警告
   (d) 密碼長度至少 8
2. 新增帳號 deyu11，chage 查看是否如預設，以管線命令配合 tee 命令導向到 /root/chage1。
3. 新增帳號 deyu12，chage 設定
   (a) 最長不用變更天數 20

(b) 可以變更的最短天數 7

(c) 密碼過期前 4 天警告

4. chage 查看 deyu12 帳號，是否如變更，並導向 /root/chage2。

Chapter 13

Linux 檔案權限與屬性

13.1 檔案權限

1. 輸入指令 `ls -al` 查看根目錄下的目錄及檔案。

```
[root@kvm8 ~]# ls -al /
total 24
dr-xr-xr-x.  18 root root  235 Sep 27 17:09 .
dr-xr-xr-x.  18 root root  235 Sep 27 17:09 ..
dr-xr-xr-x.   2 root root    6 Oct 29  2024 afs
lrwxrwxrwx.   1 root root    7 Oct 29  2024 bin -> usr/bin
dr-xr-xr-x.   5 root root 4096 Sep 27 17:12 boot
drwxr-xr-x.  20 root root 3300 Oct 14 14:01 dev
drwxr-xr-x.  98 root root 8192 Oct 21 17:55 etc
drwxr-xr-x.   7 root root   73 Oct 21 17:53 home
lrwxrwxrwx.   1 root root    7 Oct 29  2024 lib -> usr/lib
lrwxrwxrwx.   1 root root    9 Oct 29  2024 lib64 -> usr/lib64
drwxr-xr-x.   2 root root    6 Oct 29  2024 media
drwxr-xr-x.   2 root root    6 Oct 29  2024 mnt
drwxr-xr-x.   2 root root    6 Oct 29  2024 opt
dr-xr-xr-x. 186 root root    0 Oct 14 14:00 proc
dr-xr-x---.   3 root root  186 Oct 21 17:52 root
drwxr-xr-x.  31 root root  880 Oct 14 14:02 run
lrwxrwxrwx.   1 root root    8 Oct 29  2024 sbin -> usr/sbin
drwxr-xr-x.   2 root root    6 Oct 29  2024 srv
dr-xr-xr-x.  13 root root    0 Oct 14 14:01 sys
drwxrwxrwt.  11 root root 4096 Oct 22 15:44 tmp
drwxr-xr-x.  12 root root  144 Sep 27 17:00 usr
drwxr-xr-x.  20 root root 4096 Sep 27 17:11 var
```

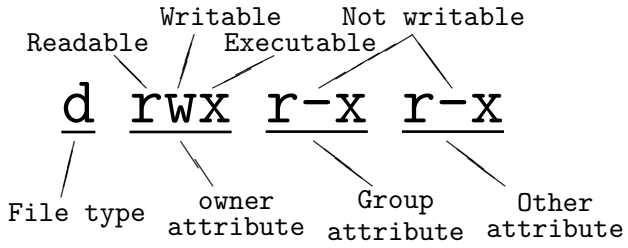
說明：

(a) total 24：檔案共 24 Blocks。

(b) 各欄位說明：

欄位一	欄位二	欄位三	欄位四	欄位五	欄位六	欄位七
drwxr-xr-x	20	root	root	4096	Sep 27 17:11	var
檔案屬性	硬連結數目	擁有者	擁有者群組	大小	建檔日期	檔名

i. 第一欄位 drwxr-xr-x，共有檔案的 10 個屬性：



13.2 改變檔案權限

1. chmod 輔助說明

```
[root@kvm8 ~]# chmod --help
Usage: chmod [OPTION]... MODE[,MODE]... FILE...
  or:  chmod [OPTION]... OCTAL-MODE FILE...
  or:  chmod [OPTION]... --reference=RFILE FILE...
Change the mode of each FILE to MODE.
With --reference, change the mode of each FILE to that of RFILE.

  -c, --changes          like verbose but report only when a change is
      made
  -f, --silent, --quiet  suppress most error messages
  -v, --verbose          output a diagnostic for every file processed
      --dereference      affect the referent of each symbolic link,
                           rather than the symbolic link itself
  -h, --no-dereference   affect each symbolic link, rather than the
      referent
      --no-preserve-root  do not treat '/' specially (the default)
      --preserve-root    fail to operate recursively on '/'
      --reference=RFILE  use RFILE's mode instead of specifying MODE
          values.
                         RFILE is always dereferenced if a symbolic link
                             .
  -R, --recursive        change files and directories recursively

The following options modify how a hierarchy is traversed when the -R
option is also specified.  If more than one is specified, only the final
one takes effect. '-H' is the default.

  -H                     if a command line argument is a symbolic link
                         to a directory, traverse it
  -L                     traverse every symbolic link to a directory
                         encountered
  -P                     do not traverse any symbolic links
```

```
      --help        display this help and exit
      --version     output version information and exit

Each MODE is of the form '[ugoa]*([-+=]([rwxXst]*|[ugo]))+|[-+=][0-7]+'.

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Full documentation <https://www.gnu.org/software/coreutils/chmod>
or available locally via: info '(coreutils) chmod invocation'
```

2. chmod 命令設定

```
chmod [ -R ] xyz 檔案或目錄
chmod [ -R ] u = r 檔案或目錄
             g + w 檔案或目錄
             o - x 檔案或目錄
             a     檔案或目錄
chmod u=rwx,go=rx 檔案或目錄
chmod a-x 檔案或目錄
```

(a) xyz 爲三組 rwx 屬性數值之加總。

r : 4

w : 2

x : 1

(b) -rwxr-xr-- 之屬性數值爲 754。

(c) 符號類型

u : user

g : group

o : other

a : all

+ : 加入

- : 除去

= : 設定

3. 例題：若從光碟複製某一目錄 abc（包含次目錄之所有檔案）至硬碟，則目前使用者雖爲檔案擁有者，但卻無法修改該檔案，請問如何以簡單的方式，讓使用者可以修改所有檔案？

```
chmod u+w -R abc
```

13.3 檔案的預設權限

1. umask：新增檔案或目錄預設權限，不指定 mode 列出目前預設權限，指定 mode 則設定預設權限。

```
[root@kvm8 ~]# umask --help
umask: umask [-p] [-S] [mode]
    Display or set file mode mask.

    Sets the user file-creation mask to MODE.  If MODE is omitted,
        prints
    the current value of the mask.

    If MODE begins with a digit, it is interpreted as an octal number;
    otherwise it is a symbolic mode string like that accepted by chmod
        (1).

    Options:
      -p    if MODE is omitted, output in a form that may be reused as
          input
      -S    makes the output symbolic; otherwise an octal number is
          output

    Exit Status:
    Returns success unless MODE is invalid or an invalid option is given
        .

參數說明：
-S(Symbolic): 以符號類型顯示權限
```

2. 切換身份爲一般帳號 deyu6。

```
[root@kvm8 ~]# su - deyu6
```

3. 執行 umask 不加 mode 參數，輸出 0022。

```
[deyu6@kvm8 ~]$ umask
0022
```

4. 新增檔案 file1 及目錄 dir1。

```
[deyu6@kvm8 ~]$ touch file1
[deyu6@kvm8 ~]$ mkdir dir1
```

5. umask 0022 表示擁有者不扣除任何權限，同群組及其他使用者皆須扣除『寫』的權限，所以建立新檔案或目錄的權限如下：

```
[deyu6@kvm8 ~]$ ll
total 4
drwxr-xr-x. 2 deyu6 deyu6 4096 Apr 29 16:29 dir1
-rw-r--r--. 1 deyu6 deyu6    0 Apr 29 16:29 file1
```

6. 執行 umask 指定 mode 0022。

```
[deyu6@kvm8 ~]$ umask 0002
```

7. 執行 umask 以符號方式查詢。

```
[deyu6@kvm8 ~]$ umask -S
u=rwx,g=rwx,o=rx
```

8. 新增檔案 file2 及目錄 dir2。

```
[deyu6@kvm8 ~]$ touch file2
[deyu6@kvm8 ~]$ mkdir dir2
```

9. umask 0002 表示擁有者及同群組不扣除任何權限，其他使用者皆須扣除『寫』的權限，所以建立新檔案或目錄的權限如下：

```
[deyu6@kvm8 ~]$ ll
total 8
drwxr-xr-x. 2 deyu6 deyu6 4096 Apr 29 16:29 dir1
drwxrwxr-x. 2 deyu6 deyu6 4096 Apr 29 16:42 dir2
-rw-r--r--. 1 deyu6 deyu6    0 Apr 29 16:29 file1
-rw-rw-r--. 1 deyu6 deyu6    0 Apr 29 16:42 file2
```

10. 登出 deyu6，再登入後 umask 查詢又回到 0222，表示 umask 設定只是目前的 shell 生效。

```
[deyu6@kvm8 ~]$ exit
logout
[root@kvm8 ~]# su - deyu6
Last login: Thu Oct 27 12:17:24 CST 2022 on pts/0
```

```
[deyu6@kvm8 ~]$ umask
0022
```

11. 要讓 deyu6 的 umask 維持 0002，必須將其寫入.bashrc。

```
[deyu6@kvm8 ~]$ vim .bashrc
[deyu6@kvm8 ~]$ tail -1 .bashrc
umask 0002
```

12. 登出 deyu6，再登入 deyu6，取得 shell 時會執行.bashrc，因此 umask 0002 會生效。

```
[deyu6@kvm8 ~]$ exit
logout
[root@kvm8 ~]# su - deyu6
Last login: Thu Oct 27 12:17:51 CST 2022 on pts/0
[deyu6@kvm8 ~]$ umask
0002
```

13.4 檔案的特殊屬性

1. 檔案的特殊屬性 SUID/SGID/SBIT

   (a) Set UID (SUID)；設置使文件在執行階段具有文件擁有者的權限，僅可用在「二進位制執行檔」上。

   (b) Set GID (SGID)：只對目錄有效，任何帳號在此目錄下建立的文件，所屬群組皆與該目錄所屬的群組相同。

   (c) Sticky Bit (SBIT)：防刪除屬性，只對目錄有效，使用者在該目錄下建立檔案或目錄時，只有檔案擁有者與 root 才有權力刪除。

2. SUID/SGID/SBIT 權限的顯示：使用 `ls -l` 列出檔案屬性，x 屬性，設定特殊屬性時顯示爲小寫字母 (s, s, t)

```
rwsrw-r--     s 表示設置 SUID 屬性
rwxrwsrw-     s 表示設置 SGID 屬性
rwxrw-rwt     t 表示設置 SBIT 屬性
```

3. SUID/SGID/SBIT 權限的設定：使用 chmod 如

```
chmod 4755      第一個數字 4 表示設置 SUID 屬性
chmod 2770      第一個數字 2 表示設置 SGID 屬性
chmod 1777      第一個數字 1 表示設置 SBIT 屬性
```

4. SUID 例子：任何帳號可以用 passwd 變更密碼，修改 /etc/shadow 中的密碼雜湊。，/usr/bin/passwd 的屬性爲 -rwsr-xr-x，其中 s 就是 SUID。

```
[root@kvm8 ~]# whereis passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man1/passwd.1.gz
[root@kvm8 ~]# ll /usr/bin/passwd
-rwsr-xr-x. 1 root root 91416 Nov  4  2024 /usr/bin/passwd
[root@kvm8 ~]# ll /etc/shadow
----------. 1 root root 1672 Oct 23 13:55 /etc/shadow
```

5. SGID 例子：/usr/log/joural 目錄群組爲 systemd-joural，任何帳號在此目錄中的檔案所屬目錄皆爲 systemd-joural。/run/log/joural 的屬性爲 drwxr-sr-x，其中 s 就是 SGID。

```
[root@kvm8 ~]# ll -d /run/log/journal
drwxr-sr-x+ 3 root systemd-journal 60 Oct 16 14:08 /run/log/journal
```

6. SBIT 例子：/tmp 目錄的屬性爲 drwxrwxrwt，其中 t 就是 SBIT。任何帳號都可在 /tmp 目錄產生檔案，但只有自己及 root 可刪除。

```
[root@kvm8 ~]# ll -d /tmp
drwxrwxrwt. 7 root root 4096 Oct 23 15:03 /tmp
```

13.5 設定 SGID

1. 先建立練習用目錄 /home/shared

```
[root@kvm8 ~]# mkdir /home/shared
```

2. 查看目錄 /home/shared，目前擁有者及群組皆爲 root，擁有者可讀、可寫、可訪問此目錄，同群組及其他人只能訪問及讀取。

```
[root@kvm8 ~]# ll -d /home/shared
drwxr-xr-x. 2 root root 6 Oct 23 14:40 /home/shared
```

3. 改變目錄 /home/shared 群組爲 sharegrp，如果 sharegrp 群組不存在，必須先新增。

```
[root@kvm8 ~]# chgrp sharegrp /home/shared
```

4. 查看目錄 /home/shared 群組爲 sharegrp。

```
[root@kvm8 ~]# ll -d /home/shared
drwxr-xr-x. 2 root sharegrp 6 Oct 23 14:40 /home/shared
```

5. 改變目錄 /home/shared，群組可讀、可寫、可訪問此目錄，其他人沒有任何權限，且設定 SGID，讓帳號在此目錄內產生的檔案，所屬群組皆爲 sharegrp。

```
[root@kvm8 ~]# chmod 2770 /home/shared
```

6. 再查看目錄 /home/shared，其屬性出現 drwxrws---，符合要求。

```
[root@kvm8 ~]# ll -d /home/shared
drwxrws---. 2 root sharegrp 6 Oct 23 14:40 /home/shared
```

13.6 SGID 測試

1. 查詢 deyu1 及 deyu2 的附屬群組是 sharegrp。

```
[root@kvm8 ~]# getent group sharegrp
sharegrp:x:2205:deyu1,deyu2
```

2. 登入 deyu1。

```
[root@kvm8 ~]# su - deyu1
Last login: Thu Oct 16 15:32:33 CST 2025 on pts/0
```

3. deyu1 的附屬群組是 sharegrp，可以進入 SGID 的目錄 /home/shared。

```
[deyu1@kvm8 ~]$ cd /home/shared/
[deyu1@kvm8 shared]$
```

4. 查看目錄 /home/shared 中沒有任何檔案。

```
[deyu1@kvm8 shared]$ ll
total 0
```

5. 產生空檔案 deyu1.txt。

```
[deyu1@kvm8 shared]$ echo 'deyu1 test' > deyu1.txt
[deyu1@kvm8 shared]$ cat deyu1.txt
deyu1 test
```

6. 查看新增的檔案 deyu1.txt，擁有者是 deyu1，因爲目錄 /home/shared SGID，所以在此目錄中新增的檔案所屬群組是 sharegrp。

```
[deyu1@kvm8 shared]$ ll
total 4
-rw-r--r--. 1 deyu1 sharegrp 11 Oct 23 16:08 deyu1.txt
```

7. 退出帳號 deyu1，再切換成 deyu2。

```
[deyu1@kvm8 shared]$ exit
logout
[root@kvm8 ~]# su - deyu2
```

8. deyu2 的附屬群組也是 sharegrp，所以也可以進到 SGID 的目錄 /home/shared。

```
[deyu2@kvm8 ~]$ cd /home/shared/
[deyu2@kvm8 shared]$
```

9. 目錄 /home/shared 中同群組的帳號，都可以讀取，所以 deyu2 可以查看 deyu1 產生的檔案 deyu1.txt。

```
[deyu2@kvm8 ~]$ cd /home/shared/
[deyu2@kvm8 shared]$ ll
total 4
-rw-r--r--. 1 deyu1 sharegrp 11 Oct 23 16:08 deyu1.txt
[deyu2@kvm8 shared]$ cat deyu1.txt
deyu1 test
```

10. 因目錄 /home/shared 屬性 2770 ，同群組帳號 deyu2 刪除 deyu1.txt，雖警告移除寫入保護的檔案，還是可按 y 刪除。

```
[deyu2@kvm8 shared]$ rm deyu1.txt
rm: remove write-protected regular file 'deyu1.txt'? y
[deyu2@kvm8 shared]$ ll
total 0
```

11. 帳號 deyu2 在目錄 /home/shared 中新增 deyu2.txt，其所屬群組一樣爲 sharegrp。

```
[deyu2@kvm8 shared]$ echo 'deyu2 test' > deyu2.txt
[deyu2@kvm8 shared]$ ll
total 4
-rw-r--r--. 1 deyu2 sharegrp 11 Oct 23 16:12 deyu2.txt
```

12. 退出帳號 deyu2，再切換成 deyu4。

```
[deyu2@kvm8 shared]$ exit
logout
[root@kvm8 ~]# su - deyu4
```

13. deyu4 不是 sharegrp 群組成員，所以無法進入目錄 /home/shared。

```
[deyu4@kvm8 ~]$ cd /home/shared/
-bash: cd: /home/shared/: Permission denied
```

14. 退出 deyu4 帳號。

```
[deyu4@kvm8 ~]$ exit
logout
```

13.7 實機練習

1. 建立帳號的預設權限。

   (a) 設定帳號 deyu6 預設權限爲 0002，且登入後必須立即生效。

   (b) deyu6 登入後，分別建立 maskf 檔案及 maskd 目錄。

   (c) 觀察檔案 maskf 的屬性是否爲 664？目錄 maskd 的屬性是否爲 775？

2. 建立 shareg 團隊工作目錄，shareg 群組，homeshare 目錄及 deyu1.f, deyu2.f 中的 deyu1, deyu2 請依參數檔變更。

   (a) 建立 homeshare 目錄，並設定其爲群組 shareg 團隊工作目錄，擁有者及群組有 rwx 權限，其他沒有任何權限，且 SETGID。

   (b) deyu1 與 deyu2 都屬於 shareg 群組，分別用 deyu1 及 deyu2 帳號在 homeshared 目錄中建立檔案 deyu1.f 及 deyu2.f。

   (c) 觀察 deyu1.f 及 deyu2.f 所屬的群組，看看 deyu1 能不能更改 deyu2.f 內容？deyu2 是否可以變更 deyu1.f 內容？

   (d) deyu4 不屬於 shareg 群組，測試 deyu4 是否可以進入 homeshare 目錄？將標準錯誤 (stderr) 導向到 /home/deyu4/cd.shared。

Chapter 14

Access Control Lists

14.1 簡介

1. 什麼是 ACL (Access Control List)？

   (a) 提供傳統的 owner,group,others 的 read,write,execute 權限之外的細部權限設定。

   (b) ACL 可以針對單一使用者，單一檔案或目錄來進行 r, w, x 的權限規範。

   (c) ACL 主要可以針對以下項目來控制權限：

      i. 使用者 (user)：可以針對使用者來設定權限；

      ii. 群組 (group)：針對群組爲對象來設定其權限；

      iii. 預設屬性 (mask)：針對在該目錄下在建立新檔案/目錄時，規範新資料的預設權限。

2. 安裝 acl

```
[root@kvm8 ~]# dnf install acl
```

3. setfacl 輔助說明

```
setfacl 2.3.2 -- set file access control lists
Usage: setfacl [-bkndRLP] { -m|-M|-x|-X ... } file ...
  -m, --modify=acl        modify the current ACL(s) of file(s)
  -M, --modify-file=file  read ACL entries to modify from file
  -x, --remove=acl        remove entries from the ACL(s) of file(s)
  -X, --remove-file=file  read ACL entries to remove from file
  -b, --remove-all        remove all extended ACL entries
  -k, --remove-default    remove the default ACL
      --set=acl           set the ACL of file(s), replacing the current
          ACL
```

```
      --set-file=file     read ACL entries to set from file
      --mask              do recalculate the effective rights mask
  -n, --no-mask           don't recalculate the effective rights mask
  -d, --default           operations apply to the default ACL
  -R, --recursive         recurse into subdirectories
  -L, --logical           logical walk, follow symbolic links
  -P, --physical          physical walk, do not follow symbolic links
      --restore=file      restore ACLs (inverse of `getfacl -R')
      --test              test mode (ACLs are not modified)
  -v, --version           print version and exit
  -h, --help              this help text

選項說明：
-m ：設定後續的 acl 參數給檔案使用，不可與 -x 合用；
-x ：刪除後續的 acl 參數，不可與 -m 合用；
-b ：移除所有的 ACL 設定參數；
-k ：移除預設的 ACL 參數；
-R ：遞迴設定 acl，即包括次目錄都會被設定；
-d ：設定『預設 acl 參數』，只對目錄有效，在該目錄新建的資料會引用此預設值
```

4. 設定檔案 acl

```
setfacl -m u[ser]:someuser:perms  file
setfacl -m g[roup]:somegroup:perms file
```

5. getfacl 輔助說明

```
[root@kvm8 ~]# getfacl --help
getfacl 2.3.2 -- get file access control lists
Usage: getfacl [-aceEsRLPtpndvh] file ...
  -a, --access            display the file access control list only
  -d, --default           display the default access control list only
  -c, --omit-header       do not display the comment header
  -e, --all-effective     print all effective rights
  -E, --no-effective      print no effective rights
  -s, --skip-base         skip files that only have the base entries
  -R, --recursive         recurse into subdirectories
  -L, --logical           logical walk, follow symbolic links
  -P, --physical          physical walk, do not follow symbolic links
  -t, --tabular           use tabular output format
  -n, --numeric           print numeric user/group identifiers
      --one-file-system   skip files on different filesystems
  -p, --absolute-names    don't strip leading '/' in pathnames
  -v, --version           print version and exit
  -h, --help              this help text
```

14.2 ACL 練習

1. 先複製測試檔案

```
[root@kvm8 ~]# cp /etc/fstab /var/tmp/
```

2. 查看複製的檔案擁有者，群組皆爲 root，目前權限爲 644，root 有讀取及寫入的權限。

```
[root@kvm8 ~]# ll /var/tmp/fstab
-rw-r--r--. 1 root root 615 Oct 23 15:05 /var/tmp/fstab
```

3. 使用 getfacl 查看 /var/tmp/fstab 目前的 ACL，沒有其他用户有特殊的權限。

```
[root@kvm8 ~]# getfacl /var/tmp/fstab
getfacl: Removing leading '/' from absolute path names
# file: var/tmp/fstab
# owner: root
# group: root
user::rw-
group::r--
other::r--
```

4. 改變 /var/tmp/fstab 的 ACL，用户 deyu1 有讀取及寫入的權限。

```
[root@kvm8 ~]# setfacl -m u:deyu1:rw /var/tmp/fstab
```

5. 改變 /var/tmp/fstab 的 ACL，用户 deyu2 沒有任何權限。

```
[root@kvm8 ~]# setfacl -m u:deyu2:- /var/tmp/fstab
```

6. 再使用 getfacl 查看 /var/tmp/fstab 目前的 ACL，其他用户 deyu1 及 deyu2 有特殊的權限。

```
[root@kvm8 ~]# getfacl /var/tmp/fstab
getfacl: Removing leading '/' from absolute path names
# file: var/tmp/fstab
# owner: root
# group: root
```

```
user::rw-
user:deyu1:rw-
user:deyu2:---
group::r--
mask::rw-
other::r--
```

7. 列出 /var/tmp/fstab 可以看到權限部分多了個 +，且權限不是原本的“644”？

```
[root@kvm8 ~]# ll /var/tmp/fstab
-rw-rw-r--+ 1 root root 694 Oct 15 06:29 /var/tmp/fstab
```

14.3 *ACL 變更

1. 測試檔案 /var/tmp/fstab 目前的 ACL。

```
[root@kvm8 ~]# getfacl /var/tmp/fstab
getfacl: Removing leading '/' from absolute path names
# file: var/tmp/fstab
# owner: root
# group: root
user::rw-
user:deyu2:---
user:deyu1:rw-
group::r--
mask::rw-
other::r--
```

2. /var/tmp/fstab 新增 deyu4 可讀寫。

```
[root@kvm8 ~]# setfacl -m u:deyu4:rw /var/tmp/fstab
```

3. 查詢 /var/tmp/fstab 的 ACL 有 deyu4 可讀寫。

```
[root@kvm8 ~]# getfacl /var/tmp/fstab
getfacl: Removing leading '/' from absolute path names
# file: var/tmp/fstab
# owner: root
# group: root
user::rw-
user:deyu2:---
```

```
user:deyu4:rw-
user:deyu1:rw-
group::r--
mask::rw-
other::r--
```

4. 刪除 /var/tmp/fstab ACL 中的 deyu4 可讀寫。

```
[root@kvm8 ~]# setfacl -x u:deyu4 /var/tmp/fstab
```

5. 查詢 /var/tmp/fstab 的 ACL 沒有 deyu4 設定了。

```
[root@kvm8 ~]# getfacl /var/tmp/fstab
getfacl: Removing leading '/' from absolute path names
# file: var/tmp/fstab
# owner: root
# group: root
user::rw-
user:deyu2:---
user:deyu1:rw-
group::r--
mask::rw-
other::r--
```

14.4 實機練習

1. 建立包含 ACL 的檔案
   (a) 複製 /etc/fstab 到 /var/tmp 目錄下
   (b) 查看 /var/tmp 目錄下的 fstab 檔案，其 acl 有無特殊的用户權限設定？
   (c) 設定 deyu1 可讀寫 fstab
   (d) 設定 deyu2 不可讀寫 fstab
   (e) 不要變動其他設定
2. 變更 ACL 的檔案
   (a) 查看 /var/tmp 目錄下的 fstab 檔案，其 acl 有無特殊的用户權限設定？
   (b) 設定 deyu4 可讀寫 fstab
   (c) 刪除 deyu4 可讀寫 fstab
   (d) 不要變動其他設定

Chapter 15

例行性命令

15.1 前言

1. 例行性：每隔一定的週期要執行的工作；
2. 常見的例行性工作：
   (a) 進行登錄檔的資料輪替：適時適量將登錄檔備份，並以新開的檔案進行記錄。
   (b) rpm 資料庫的建立。
   (c) 建立 locate 的資料庫。
   (d) 進行程序的分析：刪除僵屍程序，以保持記憶體的工作能力。
   (e) 登錄檔視察：解析有問題的紀錄檔，以維護主機的安全性。
3. cron 的使用限制：
   (a) 先在 /etc/cron.allow 檔案中尋找可以使用 crontab 指令的使用者。
   (b) 若無 /etc/cron.allow 檔案，則尋找 /etc/cron.deny 檔案不能使用 crontab 指令的使用者。
   (c) 若上述兩檔案皆不存在，則只有 root 可以使用 crontab 指令。
   (d) 預設只有空白的 /etc/cron.deny 檔，即所有人皆可使用 crontab 指令。

```
[root@kvm8 ~]# ll /etc/cron [TAB][TAB]
cron.d/       cron.deny     cron.monthly/ cron.weekly/
cron.daily/   cron.hourly/  crontab
[root@kvm8 ~]# cat /etc/cron.deny
```

15.2 Crontab 命令

(a) crontab 命令說明

```
[root@kvm8 ~]# crontab
crontab: usage error: file name or - (for stdin) must be specified
Usage:
 crontab [options] file
 crontab [options]
 crontab -n [hostname]

Options:
 -u <user>  define user
 -e         edit user's crontab
 -l         list user's crontab
 -r         delete user's crontab
 -i         prompt before deleting
 -n <host>  set host in cluster to run users' crontabs
 -c         get host in cluster to run users' crontabs
 -T <file>  test a crontab file syntax
 -s         selinux context
 -V         print version and exit
 -x <mask>  enable debugging

Default operation is replace, per 1003.2

參數說明：
-u user ：只有 root 能下達的參數，視察或編譯其他使用者的 crontab 內
    容
-l      ：列出 crontab 的內容
-e      ：編輯 crontab 的內容
-r      ：刪除 crontab 的內容
```

15.3 循環排程語法

(a) 每行排程有「分時日月週執行工作」六個參數，前五個時間數字參數分別代表：

數字代表的意義	分鐘	小時	日期	月份	週
範圍	0-59	0-23	1-31	1-12	0-6 (0 或 7 爲星期天)

(b) 時間設定輔助字符：

特殊字符	代表意義
*	代表任何時刻都接受。
,	代表分隔時段。 例如：在 0300 及 0600 時工作，則 `0 3,6 * * * command`
-	代表一段時間範圍內。 例如：在 0800 到 1200 之間每小時的 20 分都進行工作，則 `20 8-12 * * * command`
/n	每隔 n 單位間隔。 例如每 5 分鐘進行一次，則 `*/5 * * * * command`

15.4 設定排程

1. crontab -e 編輯排程。

```
[root@kvm8 ~]# crontab -e
crontab: installing new crontab
```

2. 查看排程：每星期一 14:00 touch 檔案 /tmp/crontab_test1。

```
[root@kvm8 ~]# crontab -l
0 14 * * 1 /usr/bin/touch /tmp/crontab_test1
```

3. 在星期一 14:00 後查看 /tmp/crontab_test1，存在且修改時間 14:00。

```
[root@kvm8 ~]# ll /tmp/crontab_test1
-rw-r--r--. 1 root root 0 Nov 11 14:00 /tmp/crontab_test1
```

4. 刪除排除

```
[root@kvm8 ~]# crontab -r
[root@kvm8 ~]# crontab -l
no crontab for root
```

5. 例題：帳號 deyu2 建立一個例行性工作，在每天 17:00 執行 /bin/echo "closing time"。

(a) 方法一：先切換到 deyu2，再執行 crontab -e 進行編輯，完成編輯後退出回到 root 身份。

```
[root@kvm8 ~]# su - deyu2
[deyu2@kvm8 ~]$ crontab -e
no crontab for deyu2 - using an empty one
crontab: no changes made to crontab
[deyu2@kvm8 ~]$ exit
logout
```

(b) 方法二：直接以 root 身份執行 crontab -e 加上 -u deyu2 參數指定編輯用户 deyu2 的工作排程，就可以不用切換身份。

```
[root@kvm8 ~]# crontab -e -u deyu2
no crontab for deyu2 - using an empty one
crontab: installing new crontab
```

(c) 編輯完用户 deyu2 的工作排程後，以 -l 選項查看排程內容是否正確。

```
[root@kvm8 ~]# crontab -l -u deyu2
0 17 * * * /bin/echo "closing time"
```

15.5 實機練習

1. 用户 cronuser 依下列時間執行例行性命令"/bin/echo hiya"。

   (a) 每 2 小時

   (b) 0-6 點的 12 分

   (c) 每月 1 及 15 日的 10 點 15 分

   (d) 每週日的 13 點每 3 分鐘

   (e) 每天 18 點 25 分

Chapter 16

Chronyd vs. ntpd 校時

16.1 前言

1. RHEL/CentOS 7 校時新套件 chrony 取代 ntp。
   (a) 更快速以及更精確的同步。
   (b) 頻率校正的範圍較大。
   (c) 擁有對於時鐘頻率快速變更的反應速度。
   (d) 在初始同步後不會發生 clock stepping。Stepping 意思是時間突然從不正確的時間一步調到正確時間的合理誤差時間，如果誤差小於 128ms 則以 slewing 擺動修正。
   (e) 適用於間歇性的網路連線。
2. chrony 不完全支援 ntp 中的所有可用功能，因此基於相容性上的原因，RHEL/CentOS 依然提供 ntp 套件，使用時必須明確地移除 chrony 並安裝 ntp 來代替。
3. Chrony 的時間設定演算法擁有幾項 ntp 實作所沒有的優點。

16.2 chronyd 使用

1. chrony 是 RHEL/CentOS 8 預設的校時 dameon，設定檔在 /etc/chrony.conf，新增校時伺服器 server.deyu.wang，其中 iburst 選項是用來加速初始同步。

```
[root@kvm8 ~]# vim /etc/chrony.conf
[root@kvm8 ~]# grep wang /etc/chrony.conf
pool server.deyu.wang iburst
```

2. 重新啓動 chronyd。

```
[root@kvm8 ~]# systemctl restart chronyd.service
```

3. chronyd 預設開機啓動，不放心可再設定一次。

```
[root@kvm8 ~]# systemctl enable chronyd.service
```

4. chronyd 使用 tracking 命令追踪校時來源伺服器，tracking 命令可以用 TAB 提示補齊，等一段時間，系統校正好時間後，Leap status : Normal。

```
[root@kvm8 ~]# chronyc tracking
Reference ID    : C0A87A01 (deyu.wang)
Stratum         : 3
Ref time (UTC)  : Thu Oct 30 06:05:02 2025
System time     : 0.000722108 seconds fast of NTP time
Last offset     : +0.000779472 seconds
RMS offset      : 0.000287981 seconds
Frequency       : 11.713 ppm slow
Residual freq   : +11.393 ppm
Skew            : 0.021 ppm
Root delay      : 0.005864071 seconds
Root dispersion : 0.001327947 seconds
Update interval : 64.0 seconds
Leap status     : Normal
[root@kvm8 ~]# chronyc source
sourcename   sources      sourcestats
```

5. chronyc 使用 sources 命令檢查 chronyd 校時來源伺服器，sources 可用 TAB 提示補齊，deyu.wang (server.deyu.wang 與 deyu.wang 是同一台主機) 已在表列中。

```
[root@kvm8 ~]# chronyc sources
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================

^* deyu.wang                     2   6    37    43    +90us[ +869us] +/-
    3636us
```

16.3 立即校時

1. 查詢目前時間

```
[root@kvm8 ~]# date
Thu Oct 30 02:08:30 PM CST 2025
```

2. 將時間設定成不是今天的日期 (2025-1-1)。

```
[root@kvm8 ~]# date -s 2025-1-1
Wed Jan  1 12:00:00 AM CST 2025
```

3. 不關閉 chronyd 服務，使用 -Q 選項，可以查看時間誤差，但不會眞的校時。

```
[root@kvm8 ~]# chronyd -Q "pool deyu.wang iburst"
2024-12-31T16:00:05Z chronyd version 4.6.1 starting (+CMDMON +NTP +
    REFCLOCK +RTC
+PRIVDROP +SCFILTER +SIGND +ASYNCDNS +NTS +SECHASH +IPV6 +DEBUG)
2024-12-31T16:00:05Z Disabled control of system clock
2024-12-31T16:00:10Z System clock wrong by 26143711.518471 seconds (
    ignored)
2024-12-31T16:00:10Z chronyd exiting
```

4. 不關閉 chronyd 服務，使用 -q 選項，無法校時。

```
[root@kvm8 ~]# chronyd -q "pool deyu.wang iburst"
2024-12-31T16:00:33Z chronyd version 4.6.1 starting (+CMDMON +NTP +
    REFCLOCK +RTC
+PRIVDROP +SCFILTER +SIGND +ASYNCDNS +NTS +SECHASH +IPV6 +DEBUG)
2024-12-31T16:00:33Z Fatal error : Another chronyd may already be
    running
(pid=21047), check /run/chrony/chronyd.pid
```

5. 要用 chronyd 立即校時，必須先關閉 chronyd 服務。

```
[root@kvm8 ~]# systemctl stop chronyd.service
```

6. chronyd 立即校時，校正 26143711.518381 秒。

```
[root@kvm8 ~]# chronyd -q "pool deyu.wang iburst"
2024-12-31T16:00:58Z chronyd version 4.6.1 starting (+CMDMON +NTP +
    REFCLOCK +RTC
+PRIVDROP +SCFILTER +SIGND +ASYNCDNS +NTS +SECHASH +IPV6 +DEBUG)
2024-12-31T16:00:58Z Initial frequency -11.714 ppm
2024-12-31T16:01:02Z System clock wrong by 26143711.518381 seconds (step
    )
2025-10-30T06:09:34Z chronyd exiting
```

7. 再查時間，校時成功。

```
[root@kvm8 ~]# date
Thu Oct 30 02:09:39 PM CST 2025
```

8. 恢復啓動 chronyd 服務

```
[root@kvm8 ~]# systemctl start chronyd.service
```

16.4 *makestep 校時

1. chronyd 服務的設定檔 /etc/chrony.conf 中有一行 makestep threshold limit，以下例子表示當前 3 次時間相差 1.0s，就跳躍式校時，以加速校時。

```
[root@kvm8 ~]# grep makestep /etc/chrony.conf
makestep 1.0 3
```

2. 將時間設定成不是今天的日期 (2019-11-30)。

```
[root@kvm8 ~]# date -s 2019-11-30
Sat Nov 30 00:00:00 CST 2019
```

3. makestep 立即校時。

```
[root@kvm8 ~]# chronyc makestep
200 OK
```

4. 再查時間，沒有更動 (問題待查)。

```
[root@kvm8 ~]# date
Sat Nov 30 00:00:07 CST 2019
```

5. 如果有安裝 ntpdate，使用 ntpdate 立即校時。

```
[root@kvm7 ~]# ntpdate server.deyu.wang
 9 Dec 13:47:38 ntpdate[1866]: step time server 192.168.122.1 offset
    827238.381751 sec
```

6. 校時成功。

```
[root@kvm8 ~]# date
Mon Dec  9 13:47:41 CST 2019
```

16.5 實機練習

1. 設定自動效時

   (a) 自動校時伺服器 server.deyu.wang

   (b) 確認 chronyd 服務是否開機啓動 (enable)？導向到 /root/chronyd-enabled。

   (c) 確認 chronyd 服務是否啓動 (active)？導向到 /root/chronyd-active

   (d) chronyc 查詢設定的伺服器是否出現？管線處理過濾 deyu，導向到 /root/chronyc-sources。

Chapter 17

Network File System, NFS

17.1 NFS 簡介

1. 讓遠端主機可以掛載使用本機的檔案系統。
2. 系統管理員可藉 NFS 整合網路上的服務器。例如：多台主機使用相同的 LDAP 提供的帳號，同一帳號當然使用相同的家目錄，若各主機上都各自建立每一用户的家目錄，則用户的資料將無法同步。此時使用 NFS 就可提供家目錄的整合。
3. RHEL 6 (CentOS 6) 支援 NFSv2, NFSv3, NFSv4。
4. NFS 需要 nfslock 及 rpcbind 服務。但 NFSv4 不再使用 rpcbind。

17.2 NFS Client 端工具

1. 如果沒有 nfs-utils，必須先安裝。

```
[root@kvm8 ~]# dnf install nfs-utils
```

2. showmount 指令用法

```
[root@kvm8 ~]# showmount --help
Usage: showmount [-adehv]
       [--all] [--directories] [--exports]
       [--no-headers] [--help] [--version] [host]
```

3. showmount 查看 NFS 伺服器 deyu.wang 分享的目錄。

```
[root@kvm8 ~]# showmount -e deyu.wang
Export list for deyu.wang:
/rhome       192.168.122.0/24
/home/guests 192.168.122.0/24
```

17.3 * 手動掛載

1. 建立掛載 NFS 的掛載點 /home/guests，目錄已存在且目錄內沒有任何檔案或次目錄。

```
[root@kvm8 ~]# mkdir /home/guests
mkdir: cannot create directory ‘/home/guests’: File exists
[root@kvm8 ~]# ll /home/guests/
total 0
```

2. 手動掛載 NFS 於掛載點 /home/guests。

```
[root@kvm8 ~]# mount deyu.wang:/home/guests /home/guests
```

3. 查詢掛載狀況。

```
[root@kvm8 ~]# df -h /home/guests/
Filesystem              Size  Used Avail Use% Mounted on
deyu.wang:/home/guests   99G   61G   33G  65% /home/guests
```

4. /home/guests 目錄內出現 ldapuser1 及 ldapuser2 兩個目錄。

```
[root@kvm8 ~]# ll /home/guests
drwxr-xr-x. 2 ldapuser1 ldapuser1 62 Aug 18  2023 ldapuser1
drwxr-xr-x. 2      2202      2202 62 Aug 18  2023 ldapuser2
```

5. 將掛載的 NFS 目錄卸載

```
[root@kvm8 ~]# umount /home/guests
```

17.4 autofs 自動掛載

1. 爲何需要自動掛載 autofs？
   (a) 讓用户端在有使用到 NFS 檔案系統時才掛載；
   (b) 手動掛載要先建立掛載點，自動掛載則不需事先建立掛載點。
   (c) 當 NFS 檔案系統使用完畢後，讓 NFS 自動卸載，以避免可能的 RPC 錯誤。

2. 如果沒安裝 autofs 套件，必須先安裝。

```
1 [root@kvm8 ~]# dnf install autofs
```

3. 建立主要設定檔 /etc/auto.master
   (a) 格式：< 預設目錄 > < 資料對應檔 >
   (b) 預設目錄：用户端要使用 /home/guests/ldapuser1，會到資料對應檔中找次目錄 ldapuser1 的對應。
   (c) 資料對應檔的檔名是可以自行設定的，此例中使用 /etc/auto.dyw。

```
1 [root@kvm8 ~]# vim /etc/auto.master
/home/guests    /etc/auto.dyw
```

4. 資料對應檔格式：

```
[本地端目錄]  [-掛載參數]  [伺服器所提供的目錄]
```

本地端目錄 ：指的是在 /etc/auto.master 內指定的目錄之次目錄，本地端目錄 "*" 表示 /home/guests 下的任何次目錄。

-掛載參數 ：就是 rw,bg,soft 等參數，可有可無；

伺服器所提供的目錄 ：例如 deyu.wang:/home/guests/& 中 "&" 表示對應到 "*"，當使用到目錄 /home/guests/ldapuser1，則 "*" 及 "&" 即爲 ldapuser1。

5. 建立資料對應檔內的掛載資訊，若 NFS server 有限制版本爲 v3，則必須加入參數 -vers=3。

```
[root@kvm8 ~]# vim /etc/auto.dyw
[root@kvm8 ~]# cat /etc/auto.dyw
*  -vers=3  deyu.wang:/home/guests/&
```

6. 查看開機是否啓動 autofs？disabled 未啓動。

```
[root@kvm8 ~]# systemctl is-enabled autofs.service
disabled
```

7. 設定開機啓動 autofs，且立即啓動。

```
[root@kvm8 ~]# systemctl enable --now autofs.service
Created symlink /etc/systemd/system/multi-user.target.wants/autofs.
    service →
/usr/lib/systemd/system/autofs.service.
```

17.5 autofs 自動掛載測試一

1. 確認目前並沒有掛載任何來自 deyu.wang NFS 伺服器的目錄

```
[root@kvm8 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
devtmpfs                  901M     0  901M   0% /dev
tmpfs                     915M     0  915M   0% /dev/shm
tmpfs                     915M  8.5M  907M   1% /run
tmpfs                     915M     0  915M   0% /sys/fs/cgroup
/dev/mapper/vg_kvm8-root  6.9G  1.5G  5.1G  22% /
/dev/vda1                 190M  126M   50M  72% /boot
tmpfs                     183M     0  183M   0% /run/user/0
```

2. 切換工作目錄到 /home/guests/ldapuser1，此時檔案系統自動掛載。

```
[root@kvm8 ~]# cd /home/guests/ldapuser1
[root@kvm8 ldapuser1]# pwd
/home/guests/ldapuser1
```

3. 查看 /home/guests/ldapuser1 是掛載在 912.168.122.1 的 /home/guests/ldapuser1。

```
[root@kvm8 ldapuser1]# df -h /home/guests/ldapuser1
Filesystem                              Size  Used Avail Use% Mounted on
192.168.122.1:/home/guests/ldapuser1    99G   61G   33G  65% /home/guests
    /ldapuser1
```

4. 查看 /home/guests/ldapuser1 掛載版本 3。

```
[root@kvm8 ~]# mount | grep ldap
deyu.wang:/home/guests/ldapuser1 on /home/guests/ldapuser1 type nfs
(rw,relatime,vers=3,rsize=1048576,wsize=1048576,namlen=255,hard,proto=
    tcp,
timeo=600,retrans=2,sec=sys,mountaddr=192.168.122.1,mountvers=3,
mountport=20048,mountproto=udp,local_lock=none,addr=192.168.122.1)
```

5. 退回家目錄。

```
[root@kvm8 ldapuser1]# cd
```

6. 卸載 /home/guests/ldapuser1。

```
[root@kvm8 ~]# umount /home/guests/ldapuser1
```

17.6 autofs 自動掛載測試二

1. 系統有一用户 ldapuser1，其家目錄爲 /home/guests/ldapuser1。

```
[root@kvm8 ~]# getent passwd ldapuser1
ldapuser1:x:1002:1002::/home/guests/ldapuser1:/bin/bash
```

2. 登入 ldapuser1，使用到家目錄 /home/guests/ldapuser1，此時檔案系統自動掛載。

```
[root@kvm8 ~]# su - ldapuser1
[ldapuser1@kvm8 ~]$ pwd
/home/guests/ldapuser1
```

3. 查看 /home/guests/ldapuser1 是掛載在 912.168.122.1 的 /home/guests/ldapuser1。

```
[ldapuser1@kvm8 ~]$ df -h /home/guests/ldapuser1/
Filesystem                              Size  Used Avail Use% Mounted on
192.168.122.1:/home/guests/ldapuser1    99G   61G   33G  65% /home/guests
    /ldapuser1
```

4. 登出 ldapuser1。

```
[ldapuser1@kvm8 ~]$ exit
logout
```

17.7 實機練習

1. 自動掛載設定

   (a) 查看 deyu.wang 所有分享的 NFS 目錄，看是否有 /home/guests？查詢結果導向到 /root/smount。

   (b) getent 查看系統中是否有 ldapuser1 帳號，導向到 /root/ldapuser，觀察其家目錄在哪？

   (c) 設定自掛載伺服器 deyu.wang:/home/guests 到本機的 /home/guests。

   (d) 掛載 NFS 版本爲 3。

   (e) 切換用户爲 ldapuser1，看是否有家目錄？

   (f) 確認 autofs 服務是否開機啓動 (enable)？導向到 /root/autofs-enabled。

   (g) 確認 autofs 服務是否啓動 (active)？導向到 /root/autofs-active。

Chapter 18

檔案搜尋

18.1 find 命令

1. 功能強大的檔案搜尋指令 find

```
[root@kvm8 ~]# find --help
Usage: find [-H] [-L] [-P] [-Olevel] [-D debugopts] [path...] [
    expression]

Default path is the current directory; default expression is -print.
Expression may consist of: operators, options, tests, and actions.

Operators (decreasing precedence; -and is implicit where no others are
    given):
      ( EXPR )   ! EXPR   -not EXPR   EXPR1 -a EXPR2   EXPR1 -and EXPR2
      EXPR1 -o EXPR2   EXPR1 -or EXPR2   EXPR1 , EXPR2

Positional options (always true):
      -daystart -follow -nowarn -regextype -warn

Normal options (always true, specified before other expressions):
      -depth -files0-from FILE -maxdepth LEVELS -mindepth LEVELS
      -mount -noleaf -xautofs -xdev -ignore_readdir_race -
          noignore_readdir_race

Tests (N can be +N or -N or N):
      -amin N -anewer FILE -atime N -cmin N -cnewer FILE -context
          CONTEXT
      -ctime N -empty -false -fstype TYPE -gid N -group NAME -ilname
          PATTERN
      -iname PATTERN -inum N -iwholename PATTERN -iregex PATTERN
      -links N -lname PATTERN -mmin N -mtime N -name PATTERN -newer FILE
      -nouser -nogroup -path PATTERN -perm [-/]MODE -regex PATTERN
      -readable -writable -executable
      -wholename PATTERN -size N[bcwkMG] -true -type [bcdpflsD] -uid N
      -used N -user NAME -xtype [bcdpfls]

Actions:
```

```
      -delete -print0 -printf FORMAT -fprintf FILE FORMAT -print
      -fprint0 FILE -fprint FILE -ls -fls FILE -prune -quit
      -exec COMMAND ; -exec COMMAND {} + -ok COMMAND ;
      -execdir COMMAND ; -execdir COMMAND {} + -okdir COMMAND ;

Other common options:
      --help                   display this help and exit
      --version                output version information and exit

Valid arguments for -D:
exec, opt, rates, search, stat, time, tree, all, help
Use '-D help' for a description of the options, or see find(1)

Please see also the documentation at https://www.gnu.org/software/
    findutils/.
You can report (and track progress on fixing) bugs in the "find"
program via the GNU findutils bug-reporting page at
https://savannah.gnu.org/bugs/?group=findutils or, if
you have no web access, by sending email to <bug-findutils@gnu.org>.

參數：
1. 與時間有關的參數：
   -atime n ：在 n 天之前的『一天之內』被 access 過的檔案；
   -ctime n ：在 n 天之前的『一天之內』被 change 過狀態的檔案；
   -mtime n ：在 n 天之前的『一天之內』被 modification 過的檔案；
   -newer file ：比檔案 file 還新的檔案。
2. 與使用者或群組名稱有關的參數：
   -uid n ：使用者的帳號 ID，UID=n 的檔案
   -gid n ：群組名稱的 ID，GID=n 的檔案
   -user name ：使用者帳號名稱爲 name 的檔案
   -group name：群組名稱爲 name 的檔案
   -nouser    ：檔案的擁有者不存在的檔案
   -nogroup   ：檔案的擁有群組不存在的檔案
3. 與檔案權限及名稱有關的參數：
   -name filename：檔案名稱爲 filename 的檔案；
   -size [+-]SIZE：比 SIZE 還要大(+)或小(-)的檔案。SIZE 的規格有：
                   c:byte， k:1024bytes M:1048576bytes。例如：-size +50k
                     。
   -type TYPE    ：檔案的類型爲 TYPE 的檔案，類型主要有：一般正規檔案 (f
      ),
                   裝置檔案 (b, c), 目錄 (d), 連結檔 (l), socket (s),
                   及 FIFO (p) 等屬性。
   -perm mode  ：檔案屬性『剛好等於』 mode 的檔案
   -perm -mode ：檔案屬性『必須包含所有 mode 的屬性』的檔案，權限數值大
      於等於 mode。
   -perm +mode ：已不支援。
4. 額外可進行的動作：
   -exec command {} \; ：-exec 後再接指令 command 來處理搜尋的結果，
                         {} 爲找到的內容，
                         分號 ; 爲命令的結束符號，但要用反斜線 \ 跳脫。
   -print        ：將結果列印到螢幕上（預設動作）
```

18.2 找屬於某用户的檔案

1. 先找出系統中屬於用户 deyu8 的檔案，輸出除了標準輸出 (stdout) 及目錄外，還有標準錯誤 (stderr)。

```
[root@kvm8 ~]# find / -user deyu8
/var/spool/mail/deyu8
/usr/local/bin/garmin
/usr/lib64/libbryton.so.1
find: ??/proc/7636/task/7636/fd/7??: No such file or directory
find: ??/proc/7636/task/7636/fdinfo/7??: No such file or directory
find: ??/proc/7636/fd/8??: No such file or directory
find: ??/proc/7636/fdinfo/8??: No such file or directory
```

2. 建立要將檔案複製過去的目標目錄 /root/findresults。

```
[root@kvm8 ~]# mkdir /root/findresults
```

3. 查詢目標目錄 /root/findresults 內是不是有之前找到的那些檔案？

```
[root@kvm8 ~]# ll /root/findresults/
total 388
-rw-r-----. 1 root root      0 Dec  9 10:16 deyu8
-rw-r--r--. 1 root root     11 Dec  9 10:16 garmin
-rwxr-xr-x. 1 root root 391104 Dec  9 10:16 libbryton.so.1
```

4. 使用 find 內建的 -exec 選項執行 cp 複製命令，將找到的內容複製製到 findre 目錄。

```
[root@kvm8 ~]# mkdir findre
[root@kvm8 ~]# find / -user deyu8 -exec cp {} findre \;
find: ‘/proc/105595/task/105595/fd/7’ : No such file or directory
find: ‘/proc/105595/task/105595/fdinfo/7’ : No such file or directory
find: ‘/proc/105595/fd/8’ : No such file or directory
find: ‘/proc/105595/fdinfo/8’ : No such file or directory
[root@kvm8 ~]# ll findre
total 336
-rw-r-----. 1 root root      0 Dec  8 19:35 deyu8
-rw-r--r--. 1 root root     11 Dec  8 19:35 garmin
-rwxr-xr-x. 1 root root 339328 Dec  8 19:35 libbryton.so.1
```

18.3 找特定大小的檔案

1. 找目錄 /usr 中大於 10M 的檔案。

```
[root@kvm8 ~]# find /usr -size +10M
/usr/lib64/libicudata.so.60.3
/usr/lib64/ceph/libceph-common.so.0
/usr/lib64/libmozjs-60.so.0.0.0
/usr/bin/podman
/usr/libexec/cni/dhcp
/usr/share/GeoIP/GeoLite2-City.mmdb
```

2. 找目錄 /usr/local 中小於 2k 的檔案。

```
[root@kvm8 ~]# find /usr/local/ -size -2k
/usr/local/bin/garmin
```

3. 找目錄 /usr 中介於 290k 到 300k 的檔案。

```
[root@kvm8 ~]# find /usr/ -size -300k -size +290k
/usr/lib64/libgfrpc.so.0.0.1
/usr/lib64/libnettle.so.6.5
/usr/lib64/ld-2.28.so
/usr/lib64/girepository-1.0/NM-1.0.typelib
/usr/lib64/libncursesw.so.6.1
/usr/lib64/libselinux.so.1
/usr/bin/qemu-ga
/usr/libexec/libvirt_leaseshelper
/usr/share/locale/ru/LC_MESSAGES/gnupg2.mo
/usr/share/locale/sv/LC_MESSAGES/NetworkManager.mo
/usr/share/doc/libxslt/ChangeLog
/usr/lib/locale/zh_SG.gbk/LC_CTYPE
/usr/lib/locale/zh_CN.gbk/LC_CTYPE
/usr/lib/firmware/ath10k/QCA9377/hw1.0/board-2.bin
/usr/lib/firmware/bnx2x/bnx2x-e2-7.8.2.0.fw
```

18.4 找特定屬性的檔案

1. 找目錄 /usr/local 中屬性是 644 的檔案。

```
[root@kvm8 ~]# find /usr/local/ -perm 644
/usr/local/bin/garmin
[root@kvm8 ~]# ll /usr/local/bin/garmin
```

```
-rw-r--r--. 1 deyu8 deyu8 11 Dec 25  2020 /usr/local/bin/garmin
```

2. 找目錄 /usr/local 中屬性包含 644 的檔案。

```
[root@kvm8 ~]# find /usr/local/ -perm -644
/usr/local/
/usr/local/lib64
/usr/local/bin
/usr/local/bin/garmin
/usr/local/src
/usr/local/etc
/usr/local/games
/usr/local/libexec
/usr/local/sbin
/usr/local/include
/usr/local/share
/usr/local/share/info
/usr/local/share/applications
/usr/local/share/man
/usr/local/share/man/man5
/usr/local/share/man/man3
/usr/local/share/man/man9
/usr/local/share/man/man6x
/usr/local/share/man/man3x
/usr/local/share/man/man1x
/usr/local/share/man/mann
/usr/local/share/man/man2x
/usr/local/share/man/man4x
/usr/local/share/man/man9x
/usr/local/share/man/man5x
/usr/local/share/man/man6
/usr/local/share/man/man8x
/usr/local/share/man/man8
/usr/local/share/man/man7x
/usr/local/share/man/man2
/usr/local/share/man/man4
/usr/local/share/man/man7
/usr/local/share/man/man1
/usr/local/lib
[root@kvm8 ~]# ll -d /usr/local/lib
drwxr-xr-x. 2 root root 4096 May 11  2019 /usr/local/lib
```

3. 找目錄 /usr 中屬性包含 SETUID 的檔案。

```
[root@kvm8 ~]# find /usr -perm -4000
/usr/bin/passwd
/usr/bin/sudo
/usr/bin/pkexec
/usr/bin/su
```

```
/usr/bin/umount
/usr/bin/newgrp
/usr/bin/chage
/usr/bin/mount
/usr/bin/crontab
/usr/bin/gpasswd
/usr/libexec/dbus-1/dbus-daemon-launch-helper
/usr/sbin/unix_chkpwd
/usr/sbin/mount.nfs
/usr/sbin/userhelper
/usr/sbin/grub2-set-bootflag
/usr/sbin/pam_timestamp_check
/usr/lib/polkit-1/polkit-agent-helper-1
[root@kvm8 ~]# ll /usr/bin/passwd
-rwsr-xr-x. 1 root root 33600 Apr  7  2020 /usr/bin/passwd
```

4. 找目錄 /usr 中屬性包含 SETGID 的檔案。

```
[root@kvm8 ~]# find /usr -perm -2000
/usr/bin/write
/usr/libexec/openssh/ssh-keysign
/usr/libexec/utempter/utempter
[root@kvm8 ~]# ll /usr/bin/write
-rwxr-sr-x. 1 root tty 21344 Apr 24  2020 /usr/bin/write
```

5. 搜尋 Linux 系統中 SGID 目錄：

```
[root@kvm8 ~]# find / -perm -2000 2>/dev/null
/run/log/journal
/run/log/journal/23dc31d09bc35ea4f7b3e127f22919fc
/usr/bin/write
/usr/libexec/openssh/ssh-keysign
/usr/libexec/utempter/utempter
/home/shared
```

6. 搜尋 Linux 系統中 SBIT 目錄：

```
[root@kvm8 ~]# find / -perm -1000 2>/dev/null
/sys/fs/bpf
/dev/mqueue
/dev/shm
/tmp
/tmp/systemd-private-922af849fd5e4d4285a98af342213321-httpd.service-
    CiR6sP/tmp
/tmp/systemd-private-922af849fd5e4d4285a98af342213321-chronyd.service-
    EpvurN/tmp
```

```
/var/tmp
/var/tmp/systemd-private-922af849fd5e4d4285a98af342213321-chronyd.
    service-QwbDlW/tmp
/var/tmp/systemd-private-922af849fd5e4d4285a98af342213321-httpd.service-
    Fvgsvc/tmp
```

18.5 實機練習

1. 找用户檔案
   (a) 找系統中所有屬於 deyu8 的檔案
   (b) 建立目錄 /root/findresults
   (c) 將找到的檔案複製到 /root/findresults 目錄中
2. 找特定大小檔案
   (a) 找目錄 /usr/share 中大於 1M 的檔案，導向到 /root/find1
   (b) 找目錄 /usr/share 中小於 5M 的檔案，導向到 /root/find2
   (c) 找目錄 /usr/share 中小於 300k 且大於 250k 的檔案，導向到 /root/find3
3. 找特定屬性檔案
   (a) 找目錄 /etc/systemd 屬性包含 664 的檔案，導向到 /root/find4
   (b) 找系統中屬性包含 SUID 的檔案，導向到 /root/find5
   (c) 找系統中屬性包含 SGID 的檔案，導向到 /root/find6
   (d) 找系統中屬性包含 SBIT 的檔案，導向到 /root/find7

Chapter 19

Shell Script

19.1 前言

1. 何謂 Shell Script

   (a) 文字介面下讓使用者與系統溝通的一個工具介面。

   (b) 利用 shell 的功能所寫的一支『程式 (program)』。

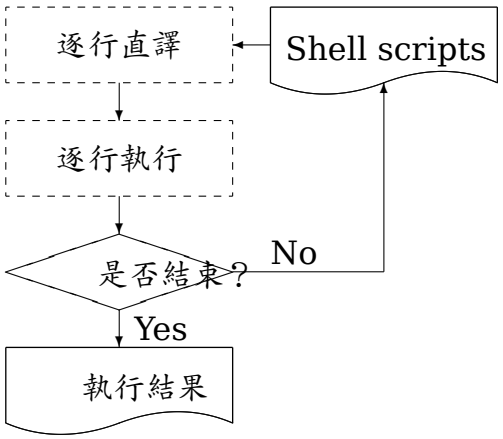
   (c) 將一些 shell 的語法與指令寫成純文字檔；

   (d) 可搭配正規表示法、管線命令與資料流重導向等功能；

   (e) shell script 更提供陣列、迴圈、條件與邏輯判斷等重要功能；

   (f) 使用者也可以直接以 shell 來撰寫程式。

2. Shell scripts 執行流程：



3. 爲什麼要學習 shell scripts?

   (a) 自動化管理的重要依據：自動處理分析主機狀態，若有問題才通知。

(b) 追蹤與管理系統的重要工作：Linux 系統的服務 (services) 啓動的介面，在目錄 /etc/init.d/ 下，所有的檔案都是 scripts。

(c) 簡單入侵偵測功能：主動分析系統登錄檔。

(d) 連續指令單一化：彙整在 command line 下達的連續指令。例如：/etc/rc.d/rc.local 裡的資料。

(e) 簡易的資料處理：處理數據資料的比對，文字資料的處理等。

(f) 跨平台支援與學習歷程較短：幾乎所有的 Unix Like 上面都可以跑 shell script 。

(g) shell script 是一項很好的系統管理工具，但數值運算的速度較慢，且使用的 CPU 資源較多。

4. Scripts 撰寫注意事項：

(a) 指令與參數間的多個空白會被忽略掉；

(b) 空白行及 [tab] 不會被理會；

(c) 讀到 [Enter] 符號，就開始執行該行命令；

(d) 一行的內容太多，可以使用 \[Enter] 來延伸至下一行；

(e) 加在 # 後面的字，全部被視爲註解。

19.2 執行 shell script

1. 寫好 shell script，例如：newfind.sh，內容只有一行 find 指令，找目錄 /usr 內大於 30M 的檔案。

```
[root@kvm8 ~]# vi newfind.sh
[root@kvm8 ~]# cat newfind.sh
find /usr/ -size +30M
```

2. 直接使用 sh 或 bash 執行腳本 newfind.sh。

```
[root@kvm8 ~]# find /usr/ -size +30M
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib64/libLLVM.so.19.1
/usr/lib64/libgallium-24.2.8.so
/usr/share/fonts/google-noto-sans-mono-cjk-vf-fonts/NotoSansMonoCJK-VF.
    ttc
/usr/share/fonts/google-noto-sans-cjk-vf-fonts/NotoSansCJK-VF.ttc
/usr/share/fonts/google-noto-serif-cjk-vf-fonts/NotoSerifCJK-VF.ttc
```

3. 要獨立執行 newfind.sh 腳本，必須先增加其可執行屬性再執行。

```
[root@kvm8 ~]# ./newfind.sh
-bash: ./newfind.sh: Permission denied
[root@kvm8 ~]# chmod +x ./newfind.sh
[root@kvm8 ~]# ./newfind.sh
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib64/libLLVM.so.19.1
/usr/lib64/libgallium-24.2.8.so
/usr/share/fonts/google-noto-sans-mono-cjk-vf-fonts/NotoSansMonoCJK-VF.
    ttc
/usr/share/fonts/google-noto-sans-cjk-vf-fonts/NotoSansCJK-VF.ttc
/usr/share/fonts/google-noto-serif-cjk-vf-fonts/NotoSerifCJK-VF.ttc
```

4. newfind.sh 增加可執行的屬性後，執行結果沒問題，是因在 linux 的文字界面預設是使用 /bin/bash 執行腳本，如果是其他直譯程式，例如：python, curl 等腳本，也直接執行一定會出錯。

5. 腳本的第一行「井號」及「驚嘆號」"#!" 組成的字元組爲 shebang (hashbang)，其後面接的內容是直譯器指令。可以宣告此腳本使用的直譯器指令，就可以正確的直譯器指令執行腳本，

```
[root@kvm8 ~]# vi newfind.sh
[root@kvm8 ~]# cat newfind.sh
#!/bin/bash
find /usr/ -size +30M
```

19.3 可執行檔

1. 可執行檔放置目錄，使用文字界面執行檔案時，直接輸入指令名稱就可執行，完全不必輸入絕對路徑或相對路徑指明執行檔所在位置，是因爲環境變數 PATH 設定好執行檔可能的所在位置，系統會自動在這些目錄中搜尋。

```
[root@kvm8 ~]# echo $PATH
/root/.local/bin:/root/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr
    /bin
```

2. 目前的目錄爲 /root，目錄內有 newfind.sh 可執行檔，但執行 newfind.sh 卻回應沒有此一命令，原因爲執行命令時，linux 系統只會在 PATH 變數包含的目錄中尋找指令。

```
[root@kvm8 ~]# pwd
/root
[root@kvm8 ~]# ll newfind.sh
-rwxr-xr-x. 1 root root 34 Nov 19 14:15 newfind.sh
[root@kvm8 ~]# newfind.sh
-bash: newfind.sh: command not found
```

3. 若要執行不是存在 PATH 變數指定的目錄的執行檔，必須明確指定路徑才可執行，以絕對路徑 /root/newfind.sh 或相對路徑./newfind.sh 都可以，其中. 表示目錄的位置。

```
[root@kvm8 ~]# /root/newfind.sh
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib64/libLLVM.so.19.1
/usr/lib64/libgallium-24.2.8.so
/usr/share/fonts/google-noto-sans-mono-cjk-vf-fonts/NotoSansMonoCJK-VF.
    ttc
/usr/share/fonts/google-noto-sans-cjk-vf-fonts/NotoSansCJK-VF.ttc
/usr/share/fonts/google-noto-serif-cjk-vf-fonts/NotoSerifCJK-VF.ttc
[root@kvm8 ~]# ./newfind.sh
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib64/libLLVM.so.19.1
/usr/lib64/libgallium-24.2.8.so
/usr/share/fonts/google-noto-sans-mono-cjk-vf-fonts/NotoSansMonoCJK-VF.
    ttc
/usr/share/fonts/google-noto-sans-cjk-vf-fonts/NotoSansCJK-VF.ttc
/usr/share/fonts/google-noto-serif-cjk-vf-fonts/NotoSerifCJK-VF.ttc
```

4. 若要讓所有用户都可執行 newfind.sh，必須將其移至環境變數 PATH 指定的目錄中，此例，將 newfind.sh 移至 /usr/loca/bin 目錄，就可以不指定執行檔位置直接執行。

```
[root@kvm8 ~]# mv newfind.sh /usr/local/bin/
[root@kvm8 ~]# newfind.sh
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib64/libLLVM.so.19.1
/usr/lib64/libgallium-24.2.8.so
/usr/share/fonts/google-noto-sans-mono-cjk-vf-fonts/NotoSansMonoCJK-VF.
    ttc
/usr/share/fonts/google-noto-sans-cjk-vf-fonts/NotoSansCJK-VF.ttc
/usr/share/fonts/google-noto-serif-cjk-vf-fonts/NotoSerifCJK-VF.ttc
```

19.4 * 可執行檔別名

1. 如果執行檔已存在，只是執行時要加太長的參數很不方便，也可在用户登入時必須讀取的設定檔 /etc/bashrc 中加入別名。alias 設定在 /etc/bashrc 或 /etc/profile 兩個檔案 ssh 登入都會生效，但圖形界面開啓 gnome-terminal 不會讀取 /etc/profile，所以不建議放在 /etc/profile。

```
[root@kvm8 ~]# vi /etc/bashrc
[root@kvm8 ~]# tail -1 /etc/bashrc
alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'
```

2. 重新讀取 /etc/bashrc，別名 qstat 已生效。

```
[root@kvm8 ~]# source /etc/bashrc
[root@kvm8 ~]# alias
alias cp='cp -i'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l.='ls -d .* --color=auto'
alias ll='ls -l --color=auto'
alias ls='ls --color=auto'
alias mv='mv -i'
alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'
alias rm='rm -i'
alias which='alias | /usr/bin/which --tty-only --read-alias --show-dot
    --show-tilde'
```

3. 查看時過濾 qstat 更清楚。

```
[root@kvm8 ~]# alias | grep qstat
alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'
```

4. 直接執行看看。

```
[root@kvm8 ~]# qstat
  PID TT       USER     COMMAND    RSZ
    1 ?        root     systemd   7452
    2 ?        root     kthreadd     0
    3 ?        root     ksoftirq     0
    5 ?        root     kworker/     0
    6 ?        root     kworker/     0
    7 ?        root     migratio     0
    8 ?        root     rcu_bh       0
```

```
10     9 ?        root     rcuob/0      0
.....................................
12 12944 pts/0    root     ps        1276
```

5. 切換一般用户 deyu1，一樣可以執行 qstat，出現的結果不會一樣。

```
[root@kvm8 ~]# su - deyu1
2 Last login: Sun Aug  9 07:42:07 CST 2015 on pts/0
[deyu1@kvm8 ~]$ qstat
4   PID TT       USER     COMMAND    RSZ
      1 ?        root     systemd   7452
6     2 ?        root     kthreadd     0
      3 ?        root     ksoftirq     0
8     5 ?        root     kworker/     0
      6 ?        root     kworker/     0
10    7 ?        root     migratio     0
      8 ?        root     rcu_bh       0
12    9 ?        root     rcuob/0      0
.....................................
14 12973 pts/0    deyu1    ps        1280
```

6. 退出 deyu1。

```
[deyu1@kvm8 ~]$ exit
2 logout
```

19.5 實機練習

1. 寫腳本 newfind
   (a) 第一行 shebang 宣告使用的直譯器指令
   (b) newfind.sh 所有人可執行
   (c) 找目錄 /usr 中小於 300k 且大於 250k 的檔案，導向到 /root/newfind.txt
   (d) 把 newfind.sh 放到 PATH 路徑可以找到的 /usr/local/bin 目錄，讓所有人可以不用指定路徑就可執行。
2. 寫腳本 newfind1 找特定屬性檔案
   (a) 第一行 shebang 宣告使用的直譯器指令
   (b) 腳本包含以下三行命令

i. 找系統中屬性包含 SUID 的檔案

ii. 找系統中屬性包含 SGID 的檔案

iii. 找系統中屬性包含 SBIT 的檔案

(c) 把 newfind1.sh 放到 PATH 路徑可以找到的 /usr/bin 目錄，讓所有人可以不用指定路徑就可執行。

Chapter 20

檔案打包壓縮

20.1 檔案的壓縮

1. 壓縮檔案的副檔名

.bz2	bzip2 程式壓縮的檔案；
.gz	gzip 程式壓縮的檔案；
.xz	xz 程式壓縮的檔案；
.tar	tar 程式只打包的資料，沒有壓縮；
.tar.bz2	tar 程式打包的檔案，且經過 bzip2 壓縮。
.tar.gz (.tgz)	tar 程式打包的檔案，且經過 gzip 壓縮。
.tar.xz	tar 程式打包的檔案，且經過 xz 壓縮。

2. 若沒安裝 bzip2，要先安裝。

```
[root@kvm8 ~]# dnf install bzip2
```

3. bzip2 輔助說明

```
[root@kvm8 ~]# bzip2 --help
bzip2, a block-sorting file compressor.  Version 1.0.8, 13-Jul-2019.

   usage: bzip2 [flags and input files in any order]

   -h --help           print this message
   -d --decompress     force decompression
   -z --compress       force compression
   -k --keep           keep (don't delete) input files
   -f --force          overwrite existing output files
   -t --test           test compressed file integrity
   -c --stdout         output to standard out
   -q --quiet          suppress noncritical error messages
```

```
   -v --verbose        be verbose (a 2nd -v gives more)
   -L --license        display software version & license
   -V --version        display software version & license
   -s --small          use less memory (at most 2500k)
   -1 .. -9            set block size to 100k .. 900k
   --fast              alias for -1
   --best              alias for -9

   If invoked as `bzip2', default action is to compress.
              as `bunzip2',  default action is to decompress.
              as `bzcat', default action is to decompress to stdout.

   If no file names are given, bzip2 compresses or decompresses
   from standard input to standard output.  You can combine
   short flags, so `-v -4' means the same as -v4 or -4v, &c.

參數：
-d    :解壓縮
-z    :壓縮
```

4. bzip2 範例

```
[root@kvm8 ~]# echo test > test.txt
[root@kvm8 ~]# ll test.*
-rw-r--r--. 1 root root 5 Oct 30 15:59 test.txt
[root@kvm8 ~]# bzip2 -z test.txt
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 45 Oct 30 15:59 test.txt.bz2
[root@kvm8 ~]# bzip2 -d test.txt.bz2
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 5 Oct 30 15:59 test.txt
```

5. gzip 輔助說明

```
[root@kvm8 ~]# gzip --help
Usage: gzip [OPTION]... [FILE]...
Compress or uncompress FILEs (by default, compress FILES in-place).

Mandatory arguments to long options are mandatory for short options too.

  -c, --stdout      write on standard output, keep original files
      unchanged
  -d, --decompress  decompress
  -f, --force       force overwrite of output file and compress links
  -h, --help        give this help
  -k, --keep        keep (don't delete) input files
  -l, --list        list compressed file contents
  -L, --license     display software license
```

```
  -n, --no-name     do not save or restore the original name and
     timestamp
  -N, --name        save or restore the original name and timestamp
  -q, --quiet       suppress all warnings
  -r, --recursive   operate recursively on directories
      --rsyncable   make rsync-friendly archive
  -S, --suffix=SUF  use suffix SUF on compressed files
      --synchronous synchronous output (safer if system crashes, but
         slower)
  -t, --test        test compressed file integrity
  -v, --verbose     verbose mode
  -V, --version     display version number
  -1, --fast        compress faster
  -9, --best        compress better

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

參數：
-d    :解壓縮
-# --fast --best:# 爲數字，--fast 是 1 壓縮速度最快；--best 是 9 最佳壓
   縮比，預設是 6。
```

6. gzip 範例

```
[root@kvm8 ~]# gzip test.txt
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 34 Oct 30 15:59 test.txt.gz
[root@kvm8 ~]# gzip -d test.txt
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 5 Oct 30 15:59 test.txt
```

20.2 檔案的打包

1. tar 命令

```
[root@kvm8 ~]# tar --help
Usage: tar [OPTION...] [FILE]...
GNU 'tar' saves many files together into a single tape or disk archive,
   and can
restore individual files from the archive.

Examples:
  tar -cf archive.tar foo bar  # Create archive.tar from files foo and
     bar.
```

```
  tar -tvf archive.tar         # List all files in archive.tar verbosely
      .
  tar -xf archive.tar          # Extract all files from archive.tar.

 Main operation mode:
  -A, --catenate, --concatenate   append tar files to an archive
  -c, --create               create a new archive
      --delete               delete from the archive (not on mag tapes!)
  -d, --diff, --compare      find differences between archive and file
      system
  -r, --append               append files to the end of an archive
      --test-label           test the archive volume label and exit
  -t, --list                 list the contents of an archive
  -u, --update               only append files newer than copy in
      archive
  -x, --extract, --get       extract files from an archive
......以下省略.....................

參數：
-c   ：建立一個打包檔案(create)；
-x   ：解開一個打包檔案；
-t   ：查看 tarfile 內的檔案；(c/x/t 不可同時存在！)
-z   ：同時用 gzip 壓縮；
-j   ：同時用 bzip2 壓縮；
-J   ：同時用 xz 壓縮；
-v   ：壓縮的過程中顯示檔案；
-f   ：使用檔名，在 f 之後要立即接檔名！
-p   ：保留檔案的原來屬性（屬性不會依據使用者而變）
-P   ：保留絕對路徑；
-N   ：比後面接的日期(yyyy/mm/dd)還要新的才會被打包；
--exclude FILE：排除 FILE 。
```

2. 將檔案 test.txt 打包爲 test.tar

```
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 5 Oct 30 15:59 test.txt
[root@kvm8 ~]# tar cvf test.tar test.txt
test.txt
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 10240 Oct 30 17:07 test.tar
-rw-r--r--. 1 root root     5 Oct 30 15:59 test.txt
```

3. 將/usr/local/bin 整個目錄打包爲 testlocal.tar

```
[root@kvm8 ~]# tar cvf testbin.tar /usr/local/bin
tar: Removing leading `/' from member names
/usr/local/bin/
/usr/local/bin/ex200
/usr/local/bin/garmin
```

```
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 20480 Oct 30 17:07 testbin.tar
-rw-r--r--. 1 root root 10240 Oct 30 17:07 test.tar
-rw-r--r--. 1 root root     5 Oct 30 15:59 test.txt
```

20.3 檔案的打包壓縮

1. 以 c 選項打包，j 選項用 bzip2 壓縮，將 /usr/local/bin 整個目錄打包並壓縮爲 testbin.tar.bz2。

```
[root@kvm8 ~]# tar jcvf testbin.tar.bz2 /usr/local/bin
tar: Removing leading `/' from member names
/usr/local/bin/
/usr/local/bin/ex200
/usr/local/bin/garmin
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 20480 Oct 30 17:07 testbin.tar
-rw-r--r--. 1 root root  4332 Oct 30 17:11 testbin.tar.bz2
-rw-r--r--. 1 root root 10240 Oct 30 17:07 test.tar
-rw-r--r--. 1 root root     5 Oct 30 15:59 test.txt
```

2. 以 c 選項打包，z 選項用 gzip 壓縮，將 /usr/local/bin 整個目錄打包並壓縮爲 testbin.tgz。

```
[root@kvm8 ~]# tar zcvf testbin.tgz /usr/local/bin
tar: Removing leading `/' from member names
/usr/local/bin/
/usr/local/bin/ex200
/usr/local/bin/garmin
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 20480 Oct 30 17:07 testbin.tar
-rw-r--r--. 1 root root  4332 Oct 30 17:11 testbin.tar.bz2
-rw-r--r--. 1 root root  4209 Oct 30 17:12 testbin.tgz
-rw-r--r--. 1 root root 10240 Oct 30 17:07 test.tar
-rw-r--r--. 1 root root     5 Oct 30 15:59 test.txt
```

3. 以 c 選項打包，J 選項用 xz 壓縮，將 /usr/local/bin 整個目錄打包並壓縮爲 testbin.tar.xz。

```
[root@kvm8 ~]# tar Jcvf testbin.tar.xz /usr/local/bin
tar: Removing leading `/' from member names
/usr/local/bin/
/usr/local/bin/ex200
```

```
/usr/local/bin/garmin
[root@kvm8 ~]# ll test*
-rw-r--r--. 1 root root 20480 Oct 30 17:07 testbin.tar
-rw-r--r--. 1 root root  4332 Oct 30 17:11 testbin.tar.bz2
-rw-r--r--. 1 root root  3560 Oct 30 17:13 testbin.tar.xz
-rw-r--r--. 1 root root  4209 Oct 30 17:12 testbin.tgz
-rw-r--r--. 1 root root 10240 Oct 30 17:07 test.tar
-rw-r--r--. 1 root root     5 Oct 30 15:59 test.txt
```

20.4 查看打包壓縮檔內容

1. 以 t 選項查看打包檔，j 選項用 bzip2 解壓縮，不解打包壓縮下查看 testbin.tar.bz2 的內容。

```
[root@kvm8 ~]# tar jtvf testbin.tar.bz2
drwxr-xr-x root/root         0 2025-09-27 17:10 usr/local/bin/
-rwxr-xr-x root/root     16720 2025-09-27 17:10 usr/local/bin/ex200
-rw-r--r-- deyu8/deyu8      11 2025-09-27 17:10 usr/local/bin/garmin
```

2. 以 t 選項查看打包檔，z 選項用 gzip 解壓縮，不解打包壓縮下查看 testbin.tgz 的內容。

```
[root@kvm8 ~]# tar ztvf testbin.tgz
drwxr-xr-x root/root         0 2025-09-27 17:10 usr/local/bin/
-rwxr-xr-x root/root     16720 2025-09-27 17:10 usr/local/bin/ex200
-rw-r--r-- deyu8/deyu8      11 2025-09-27 17:10 usr/local/bin/garmin
```

3. 以 t 選項查看打包檔，J 選項用 xz 解壓縮，不解打包壓縮下查看 testbin.tar.xz 的內容。

```
[root@kvm8 ~]# tar Jtvf testbin.tar.xz
drwxr-xr-x root/root         0 2025-09-27 17:10 usr/local/bin/
-rwxr-xr-x root/root     16720 2025-09-27 17:10 usr/local/bin/ex200
-rw-r--r-- deyu8/deyu8      11 2025-09-27 17:10 usr/local/bin/garmin
```

20.5 解開打包壓縮檔

1. 以 x 選項查看打包檔，j 選項用 bzip2 解壓縮，解開打包壓縮解開 testbin.tar.bz2，解開後檔案在目前目錄下。

```
[root@kvm8 ~]# tar jxvf testbin.tar.bz2
usr/local/bin/
usr/local/bin/ex200
usr/local/bin/garmin
```

2. 以 x 選項查看打包檔，z 選項用 gzip 解壓縮，解開打包壓縮解開 testbin.tgz，解開後檔案在目前目錄下。

```
[root@kvm8 ~]# tar zxvf testbin.tgz
usr/local/bin/
usr/local/bin/ex200
usr/local/bin/garmin
```

3. 以 x 選項查看打包檔，J 選項用 xz 解壓縮，解開打包壓縮解開 testbin.tar.xz，解開後檔案在目前目錄下。

```
[root@kvm8 ~]# tar Jxvf testbin.tar.xz
usr/local/bin/
usr/local/bin/ex200
usr/local/bin/garmin
```

20.6 實機練習

1. 將整個 /usr/local (由變數 tzdir 決定) 目錄整個打包壓縮成以下檔案：
   (a) 壓縮方式 gzip，存成檔案 /root/tzfile.tgz。
   (b) 壓縮方式 bzip2，存成檔案 /root/tzfile.tar.bz2。
   (c) 壓縮方式 xz，存成檔案 /root/tzfile.tar.xz。
2. 不解開打包壓縮檔的情況下，查看以下檔案的內容，看是否與原始資料相同？
   (a) /root/tzfile.tgz 查看結果導向到 /root/tzfile.tgz.t。
   (b) /root/tzfile.tar.bz2 查看結果導向到 /root/tzfile.tar.bz2.t。
   (c) /root/tzfile.tar.xz 查看結果導向到 /root/tzfile.tar.xz.t。
3. 解壓縮打包，查看解開的檔案是否與原始資料相同？

(a) 建立目錄 /root/mytar.gz

(b) 切換工作目錄到 /root/mytar.gz

(c) 解打包壓縮 /root/tzfile.tgz

(d) 建立目錄 /root/mytar.bz

(e) 切換工作目錄到 /root/mytar.bz

(f) 解打包壓縮 /root/tzfile.tar.bz2

(g) 建立目錄 /root/mytar.xz

(h) 切換工作目錄到 /root/mytar.xz

(i) 解打包壓縮 /root/tzfile.tar.xz

Chapter 21

systemd 管理服務

21.1 前言

1. Systemd 是 Linux 的系統和服務管理工具，在 RHEL 7 Systemd 爲預設的 Init System。
   (a) v6 PID=1 是 init。
   (b) v7 PID=1 是 systemd
2. Systemd 除向下相容 SysV 的 init 腳本外，更提供許多功能，例如：
   (a) 平行處理能力，增加開機速度。
   (b) 自動服務相依管理，避免過久的 timeout，例如：網路不通時，就不會像 v6 還試圖連上網路。
3. 單元 (unit) 是 systemd 管理系統及服務的基本單位，使用單元檔案 (unit file) 描述單元並定義其行爲的配置指令，讓 system 知道如何操作及管理單元。
4. Systemd 依功能分成各種單元類型，例如：
   (a) Service units (.service): 這是最常用的單元，定義如何啓動、關閉服務。
   (b) Target units (.target): 透過相依關係將一些單元組合在一起，例如 graphical.target 單元組合進入圖形界面服務，將其設爲預設的 target，則開機就會進入圖形界面。
   (c) Timer units (.timer): 安排單元活動的時間，類似 cron 例行性排程工作。

21.2 systemctl 命令

1. systemctl 命令用來管理各種不同型態的 systemd 單元。

```
[root@kvm8 ~]# systemctl -t help
Available unit types:
service
socket
target
device
mount
automount
snapshot
timer
swap
path
slice
scope
```

2. systemctl 輔助說明

```
[root@kvm8 ~]# systemctl --help
systemctl [OPTIONS...] {COMMAND} ...

Query or send control commands to the systemd manager.

  -h --help           Show this help
     --version        Show package version
  -t --type=TYPE      List only units of a particular type
     --state=STATE    List only units with particular LOAD or SUB or
         ACTIVE stat
  -p --property=NAME  Show only properties by this name
  -a --all            Show all loaded units/properties, including dead/
      empty
                      ones. To list all units installed on the system,
                          use
                      the 'list-unit-files' command instead.
     --reverse        Show reverse dependencies with 'list-dependencies'
  -l --full           Don't ellipsize unit names on output
     --fail           When queueing a new job, fail if conflicting jobs
         are
                      pending
     --irreversible   When queueing a new job, make sure it cannot be
         implicitly
                      cancelled
     --ignore-dependencies
                      When queueing a new job, ignore all its
                          dependencies
```

```
     --show-types     When showing sockets, explicitly show their type
  -i --ignore-inhibitors
                      When shutting down or sleeping, ignore inhibitors
     --kill-who=WHO   Who to send signal to
  -s --signal=SIGNAL  Which signal to send
  -H --host=[USER@]HOST
                      Show information for remote host
  -P --privileged     Acquire privileges before execution
  -q --quiet          Suppress output
     --no-block       Do not wait until operation finished
     --no-wall        Don't send wall message before halt/power-off/
         reboot
     --no-reload      When enabling/disabling unit files, don't reload
         daemon
                      configuration
     --no-legend      Do not print a legend (column headers and hints)
     --no-pager       Do not pipe output into a pager
     --no-ask-password
                      Do not ask for system passwords
     --system         Connect to system manager
     --user           Connect to user service manager
     --global         Enable/disable unit files globally
     --runtime        Enable unit files only temporarily until next
         reboot
  -f --force          When enabling unit files, override existing
      symlinks
                      When shutting down, execute action immediately
     --root=PATH      Enable unit files in the specified root directory
  -n --lines=INTEGER  Number of journal entries to show
  -o --output=STRING  Change journal output mode (short, short-monotonic
      ,
                      verbose, export, json, json-pretty, json-sse, cat)
     --plain          Print unit dependencies as a list instead of a
         tree

Unit Commands:
  list-units                      List loaded units
  list-sockets                    List loaded sockets ordered by address
  start [NAME...]                 Start (activate) one or more units
  stop [NAME...]                  Stop (deactivate) one or more units
  reload [NAME...]                Reload one or more units
  restart [NAME...]               Start or restart one or more units
  try-restart [NAME...]           Restart one or more units if active
  reload-or-restart [NAME...]     Reload one or more units if possible,
                                  otherwise start or restart
  reload-or-try-restart [NAME...] Reload one or more units if possible,
                                  otherwise restart if active
  isolate [NAME]                  Start one unit and stop all others
  kill [NAME...]                  Send signal to processes of a unit
  is-active [NAME...]             Check whether units are active
  is-failed [NAME...]             Check whether units are failed
  status [NAME...|PID...]         Show runtime status of one or more
      units
  show [NAME...|JOB...]           Show properties of one or more
```

```
                                    units/jobs or the manager
70    set-property [NAME] [ASSIGNMENT...]
                                    Sets one or more properties of a unit
72    help [NAME...|PID...]         Show manual for one or more units
      reset-failed [NAME...]        Reset failed state for all, one, or
          more
74                                  units
      list-dependencies [NAME]      Recursively show units which are
          required
76                                  or wanted by this unit or by which
                                        this
                                    unit is required or wanted
78
    Unit File Commands:
80    list-unit-files               List installed unit files
      enable [NAME...]              Enable one or more unit files
82    disable [NAME...]             Disable one or more unit files
      reenable [NAME...]            Reenable one or more unit files
84    preset [NAME...]              Enable/disable one or more unit files
                                    based on preset configuration
86    is-enabled [NAME...]          Check whether unit files are enabled

88    mask [NAME...]                Mask one or more units
      unmask [NAME...]              Unmask one or more units
90    link [PATH...]                Link one or more units files into
                                    the search path
92    get-default                   Get the name of the default target
      set-default NAME              Set the default target
94
    Job Commands:
96    list-jobs                     List jobs
      cancel [JOB...]               Cancel all, one, or more jobs
98
    Snapshot Commands:
100   snapshot [NAME]               Create a snapshot
      delete [NAME...]              Remove one or more snapshots
102
    Environment Commands:
104   show-environment              Dump environment
      set-environment [NAME=VALUE...] Set one or more environment variables
106   unset-environment [NAME...]   Unset one or more environment
          variables

108   Manager Lifecycle Commands:
      daemon-reload                 Reload systemd manager configuration
110   daemon-reload                 Reload systemd manager configuration
      daemon-reexec                 Reexecute systemd manager
112
    System Commands:
114   default                       Enter system default mode
      rescue                        Enter system rescue mode
116   emergency                     Enter system emergency mode
      halt                          Shut down and halt the system
118   poweroff                      Shut down and power-off the system
```

```
  reboot                          Shut down and reboot the system
120 kexec                         Shut down and reboot the system with
      kexec
  exit                            Request user instance exit
122 switch-root [ROOT] [INIT]     Change to a different root file system
  suspend                         Suspend the system
124 hibernate                     Hibernate the system
  hybrid-sleep                    Hibernate and suspend the system
```

21.3 服務啓動

1. 查看服務 firewalld 狀態，目前爲 active，正在執行中。

```
1  [root@kvm8 ~]# systemctl status firewalld.service
     firewalld.service - firewalld - dynamic firewall daemon
3       Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled;
            vendor>
        Active: active (running) since Fri 2022-09-23 09:45:32 CST; 1 week
            6 days >
5         Docs: man:firewalld(1)
     Main PID: 640 (firewalld)
7       Tasks: 2 (limit: 11111)
        Memory: 42.5M
9          CPU: 4.410s
        CGroup: /system.slice/firewalld.service
11                640 /usr/bin/python3 -s /usr/sbin/firewalld --nofork --
                    nopid

13 Sep 23 09:45:16 999999 systemd[1]: Starting firewalld - dynamic firewall
       daemon>
   Sep 23 09:45:32 999999 systemd[1]: Started firewalld - dynamic firewall
      daemon.
15 [root@kvm8 ~]# systemctl status firewalld.service
     firewalld.service - firewalld - dynamic firewall daemon
17      Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled;
            vendor>
        Active: inactive (dead) since Thu 2022-10-06 10:13:55 CST; 2s ago
19        Docs: man:firewalld(1)
       Process: 640 ExecStart=/usr/sbin/firewalld --nofork --nopid
           $FIREWALLD_ARGS>
21    Main PID: 640 (code=exited, status=0/SUCCESS)
            CPU: 5.179s
23
   Sep 23 09:45:16 999999 systemd[1]: Starting firewalld - dynamic firewall
       daemon>
25 Sep 23 09:45:32 999999 systemd[1]: Started firewalld - dynamic firewall
      daemon.
```

```
Oct 06 10:13:54 kvm8.deyu.wang systemd[1]: Stopping firewalld - dynamic
    firewal>
Oct 06 10:13:55 kvm8.deyu.wang systemd[1]: firewalld.service:
    Deactivated succe>
Oct 06 10:13:55 kvm8.deyu.wang systemd[1]: Stopped firewalld - dynamic
    firewall>
Oct 06 10:13:55 kvm8.deyu.wang systemd[1]: firewalld.service: Consumed
    5.179s C>
```

2. 關閉服務 firewalld。

```
[root@kvm8 ~]# systemctl stop firewalld.service
```

3. 查看服務 firewalld 狀態，目前爲 inactive，關閉中。

```
[root@kvm8 ~]# systemctl status firewalld.service
  firewalld.service - firewalld - dynamic firewall daemon
     Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled;
          vendor>
     Active: inactive (dead) since Thu 2022-10-06 10:13:55 CST; 2s ago
       Docs: man:firewalld(1)
    Process: 640 ExecStart=/usr/sbin/firewalld --nofork --nopid
        $FIREWALLD_ARGS>
   Main PID: 640 (code=exited, status=0/SUCCESS)
        CPU: 5.179s

Sep 23 09:45:16 999999 systemd[1]: Starting firewalld - dynamic firewall
     daemon>
Sep 23 09:45:32 999999 systemd[1]: Started firewalld - dynamic firewall
    daemon.
Oct 06 10:13:54 kvm8.deyu.wang systemd[1]: Stopping firewalld - dynamic
    firewal>
Oct 06 10:13:55 kvm8.deyu.wang systemd[1]: firewalld.service:
    Deactivated succe>
Oct 06 10:13:55 kvm8.deyu.wang systemd[1]: Stopped firewalld - dynamic
    firewall>
Oct 06 10:13:55 kvm8.deyu.wang systemd[1]: firewalld.service: Consumed
    5.179s C>
```

4. 查看服務 firewalld 目前是否啓動？結果爲 inactive。

```
[root@kvm8 ~]# systemctl is-active firewalld.service
```

```
inactive
```

5. 啓動服務 firewalld。

```
[root@kvm8 ~]# systemctl start firewalld.service
```

6. 再查看服務 firewalld 目前是否啓動？結果爲 active。

```
[root@kvm8 ~]# systemctl is-active firewalld.service
active
```

7. 也可以用 restart 選項，重新啓動服務 firewalld。

```
[root@kvm8 ~]# systemctl restart firewalld.service
```

8. 再查看服務 firewalld 目前是否啓動？結果爲 active。

```
[root@kvm8 ~]# systemctl is-active firewalld.service
active
```

21.4 開機自動啓動服務

1. 查看服務 firewalld 開機是否自動啓動？結果爲 enabled。

```
[root@kvm8 ~]# systemctl is-enabled firewalld.service
enabled
```

2. 關閉服務 firewalld 開機自動啓動。

```
[root@kvm8 ~]# systemctl disable firewalld.service
Removed /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
```

3. 再查看服務 firewalld 開機是否自動啓動？結果爲 disabled。

```
[root@kvm8 ~]# systemctl is-enabled firewalld.service
disabled
```

4. 再啓動服務 firewalld 開機自動啓動。

```
[root@kvm8 ~]# systemctl enable firewalld.service
Created symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.
    service →
/usr/lib/systemd/system/firewalld.service.
Created symlink /etc/systemd/system/multi-user.target.wants/firewalld.
    service →
/usr/lib/systemd/system/firewalld.service.
```

5. 再查看服務 firewalld 開機是否自動啓動？結果爲 enabled。

```
[root@kvm8 ~]# systemctl is-enabled firewalld.service
enabled
```

21.5 建立自訂 Systemd 服務

1. 建立服務要執行的腳本 /usr/local/bin/usrfind.sh，將找到的結果導向到家目錄的 usrfind.txt。

```
[root@kvm8 ~]# vi /usr/local/bin/usrfind.sh
[root@kvm8 ~]# cat /usr/local/bin/usrfind.sh
#!/bin/bash
find /usr/ -size +10M > ~/usrfind.txt
```

2. 將 /usr/local/bin/usrfind.sh 加上可執行屬性。

```
[root@kvm8 ~]# chmod +x /usr/local/bin/usrfind.sh
```

3. 查看 systemd 服務的單元檔案的說明文件，其中範例一如下：

```
[root@kvm8 ~]# man 5 systemd.unit
       Example 1. Allowing units to be enabled

       The following snippet (highlighted) allows a unit (e.g.  foo.
           service)
       to be enabled via systemctl enable:

           [Unit]
           Description=Foo

           [Service]
           ExecStart=/usr/sbin/foo-daemon
```

4. 建立 systemd 服務單位設定檔 /usr/lib/systemd/system/usrfind.service

```
[root@kvm8 ~]# vi /usr/lib/systemd/system/usrfind.service
[root@kvm8 ~]# cat /usr/lib/systemd/system/usrfind.service
[Unit]
Description=usrfind.sh

[Service]
ExecStart=/usr/local/bin/usrfind.sh
```

5. 重新載入 Systemd 設定檔

```
[root@kvm8 ~]# systemctl daemon-reload
```

6. 啓動 usrfind 服務

```
[root@kvm8 ~]# systemctl start usrfind.service
```

7. 查看 usrfind 服務狀態，因爲只有一行指令，不是背景執行的常駐程式，所以啓動後執行完馬上退出。

```
[root@kvm8 ~]# systemctl status usrfind.service
  usrfind.service - usrfind.sh
     Loaded: loaded (/usr/lib/systemd/system/usrfind.service; static)
     Active: inactive (dead)

Sep 29 20:54:54 kvm8.deyu.wang systemd[1]: Starting usrfind.service -
    usrfind.sh...
Sep 29 20:54:54 kvm8.deyu.wang systemd[1]: usrfind.service: Deactivated
    successfully.
Sep 29 20:54:54 kvm8.deyu.wang systemd[1]: Finished usrfind.service -
    usrfind.sh.
```

8. 查看家目錄於 20:54 建立 usrfind.txt 與 usrfind 服務狀態顯示的時間一攻。

```
[root@kvm8 ~]# ll usrfind.txt
-rw-r--r--. 1 root root 500 Sep 29 20:54 usrfind.txt
```

9. 家目錄建立的 usrfind.txt，內容爲查詢的結果。

```
[root@kvm8 ~]# cat usrfind.txt
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib/firmware/nvidia/ga102/gsp/gsp-535.113.01.bin.xz
/usr/lib/firmware/nvidia/tu102/gsp/gsp-535.113.01.bin.xz
/usr/lib/modules/6.12.0-55.9.1.el10_0.x86_64/vmlinuz
/usr/lib64/libicudata.so.74.2
/usr/lib64/libLLVM.so.19.1
/usr/lib64/libgallium-24.2.8.so
/usr/share/fonts/google-noto-sans-mono-cjk-vf-fonts/NotoSansMonoCJK-VF.
    ttc
/usr/share/fonts/google-noto-sans-cjk-vf-fonts/NotoSansCJK-VF.ttc
/usr/share/fonts/google-noto-serif-cjk-vf-fonts/NotoSerifCJK-VF.ttc
```

21.6 Timer 單元手冊

1. 先查 systemd.timer 說明文件，其中 OPTIONS 提到要包含 [Unit] 及 [Timer] 兩個 sections，尤其 [Timer] 包含計時器的設定一定要存在。

```
[root@kvm8 ~]# man 5 systemd.timer | grep ^OPTIONS -A6
OPTIONS
       Timer unit files may include [Unit] and [Install] sections, which
            are
       described in systemd.unit(5).

       Timer unit files must include a [Timer] section, which carries
       information about the timer it defines. The options specific to
           the
       [Timer] section of timer units are the following:
```

2. 查詢 systemd.timer(5) 手冊的 OnCalender 參數說明，計時器的定義可以查詢 systemd.time(7) 手冊。

```
[root@kvm8 ~]# man 5 systemd.timer | grep OnCalendar=$ -A4
       OnCalendar=
           Defines realtime (i.e. wallclock) timers with calendar event
           expressions. See systemd.time(7) for more information on the
               syntax
           of calendar event expressions. Otherwise, the semantics are
               similar
           to OnActiveSec= and related settings.
```

3. 查詢 systemd.time(7) 手冊。

```
[root@kvm8 ~]# man 7 systemd.time | grep ^CALENDAR -A50
CALENDAR EVENTS
       Calendar events may be used to refer to one or more points in
           time in a
       single expression. They form a superset of the absolute
           timestamps
       explained above:

           Thu,Fri 2012-*-1,5 11:12:13
              週      年-月-日 時:分:秒

       The above refers to 11:12:13 of the first or fifth day of any
           month of
```

```
        the year 2012, but only if that day is a Thursday or Friday.
12
        The weekday specification is optional. If specified, it should
            consist
14      of one or more English language weekday names, either in the
        abbreviated (Wed) or non-abbreviated (Wednesday) form (case does
            not
16      matter), separated by commas. Specifying two weekdays separated
            by ".."
        refers to a range of continuous weekdays.  "," and ".."  may be
18      combined freely.

20                              以下以時段"分"爲例：
        冒號 ".." 時段範圍   3..5 3至5 分
22      逗號 ","  分隔時段   3,5  3或5 分
        星號 "*"  任意時段    *   不指定分，每分都符合
24      斜線 "/"  每時段重複 0/5  每5分鐘重複一次

26      In the date and time specifications, any component may be
            specified as
        "*" in which case any value will match. Alternatively, each
            component
28      can be specified as a list of values separated by commas. Values
            may be
        suffixed with "/" and a repetition value, which indicates that
            the
30      value itself and the value plus all multiples of the repetition
            value
        are matched. Two values separated by ".."  may be used to
            indicate a
32      range of values; ranges may also be followed with "/" and a
            repetition
        value, in which case the expression matches all times starting
            with the
34      start value, and continuing with all multiples of the repetition
            value
        relative to the start value, ending at the end value the latest.
36
        A date specification may use "~" to indicate the last day in a
            month.
38      For example, "*-02~03" means "the third last day in February,"
            and "Mon
        *-05~07/1" means "the last Monday in May."
40
        日期中的連接號 "~" 代表最後幾天
42      例如：
                "*-02~03"          2月的最後3天
44              "Mon *-05~07/1"    5月的最後星期一

46      The seconds component may contain decimal fractions both in the
            value
        and the repetition. All fractions are rounded to 6 decimal places
            .
48
```

```
        Either time or date specification may be omitted, in which case
            *-*-*
50      and 00:00:00 is implied, respectively. If the seconds component
            is not
        specified, ":00" is assumed.
52
        不指定時間或日期可以省略，例如
54      *:0/5      每5分鐘
        *:*:0/5    每5秒
56
        Timezone can be specified as the literal string "UTC", or the
            local
58      timezone, similar to the supported syntax of timestamps (see
            above), or
        the timezone in the IANA timezone database format (also see above
            ).
60
        The following special expressions may be used as shorthands for
            longer
62      normalized forms:

64              minutely → *-*-* *:*:00
                  hourly → *-*-* *:00:00
66                 daily → *-*-* 00:00:00
                  monthly → *-*-01 00:00:00
68                 weekly → Mon *-*-* 00:00:00
```

21.7 Timer 實測

1. 修改 usrfind.sh 導向的檔名含導向時間的”分”及”秒”，以利測試。

```
[root@kvm8 ~]# vi /usr/local/bin/usrfind.sh
2 [root@kvm8 ~]# cat /usr/local/bin/usrfind.sh
#!/bin/bash
4 find /usr/ -size +10M > ~/usrfind.$(date +%M-%S).txt
```

2. 編輯 usrfind.timer 設定每 2 分鐘執行一次，如果沒有 [Install] 章節，則爲 static 單元，無法使用 systemctl enable 設定開機啓動，以下 [Install] 章節使用 WantedBy=timers.target 在開機時 enable 這個單元。

```
[root@kvm8 ~]# vi /usr/lib/systemd/system/usrfind.timer
2 [root@kvm8 ~]# cat /usr/lib/systemd/system/usrfind.timer
[Unit]
```

```
4 Description=Run the script every 2 minute

6 [Timer]
OnCalendar=*:0/2
8
[Install]
10 WantedBy=timers.target
```

3. 設定開機啓動 usrfind.timer 計時器，且現在啓動。

```
[root@kvm11 ~]# systemctl enable --now usrfind.timer
```

4. 查看 usrfind.timer 計時器狀態 active。

```
1  usrfind.timer - Run the script every 2 minute
     Loaded: loaded (/usr/lib/systemd/system/usrfind.timer; enabled;
         preset: disabled)
3    Active: active (waiting) since Wed 2025-12-03 15:06:36 CST; 6s ago
 Invocation: b0dba8e66b184072bdd7c2f4ef8becbc
5    Trigger: Wed 2025-12-03 15:08:00 CST; 1min 16s left
   Triggers:   usrfind.service
7
Dec 03 15:06:36 kvm11.deyu.wang systemd[1]: Stopped usrfind.timer - Run
    the script every 2 minute.
9 Dec 03 15:06:36 kvm11.deyu.wang systemd[1]: Stopping usrfind.timer - Run
     the script every 2 minute...
Dec 03 15:06:36 kvm11.deyu.wang systemd[1]: Started usrfind.timer - Run
    the script every 2 minute.
```

5. 幾分鐘後查看 usrfind.sh 導向結果，2:58 及 13:00 各有一檔案，表示每 2 分鐘導向一次，符合計時器設定。

```
[root@kvm8 ~]# ll ~/usrfind*
2 -rw-r--r--. 1 root root 500 Dec  3 15:06 /root/usrfind06-34.txt
-rw-r--r--. 1 root root 500 Dec  3 15:08 /root/usrfind08-09.txt
```

21.8 實機練習

1. 使用 systemctl 控制服務

(a) 查看 chronyd 服務是否啓動 (active)？結果導向到 /root/systemd-active。

(b) 查看 chronyd 服務是否開機啓動 (enable)？結果導向到 /root/systemd-enabled。

(c) 把關閉 chronyd 服務的完整指令寫到 /root/systemd-stop。

(d) 把啓動 chronyd 服務的完整指令寫到 /root/systemd-start。

(e) 把重新啓動 chronyd 服務的完整指令寫到 /root/systemd-restart。

(f) 把查詢 chronyd 服務狀態的完整指令寫到 /root/systemd-status。

2. 完成講義中 usrfind.service 服務及 usrfind.timer 計時器，其中 usrfind 依照參數變更名稱。

3. 建立自訂的 myfind 服務及計時器

(a) 建立 /usr/local/bin/myfind.sh 可執行腳本。
myfind.sh 找系統中所有屬於 deyu8 的檔案，導向到 /root/d8file.MM.SS，其中 MM 爲導向時間”分”，SS 爲時間”秒”。

(b) 建立 myfind 計時器

i. 每 3 分執行一次。

ii. 開機啓動且現在啓動計時器。

iii. 查看導向的檔案 /root/d8file.MM.SS 是否如預期，每 3 分產生一個檔案。

4. 建立自訂的 mytouch 服務及計時器

(a) 建立 /usr/local/bin/mytouch.sh 可執行腳本。
mytouch.sh touch 產生空檔 /root/touch.MM.SS，其中 MM 爲導向時間”分”，SS 爲時間”秒”。

(b) 建立 mytouch 計時器

i. 每 3 秒執行一次。

ii. 開機啓動且現在啓動計時器。

iii. 查看產生的空檔案 /root/touch.MM.SS 是否如預期，每 3 秒產生一個檔案。

Part III

儲存空間管理 KVM10

Chapter 22

開機

22.1 前言

1. Linux 出現開機問題或忘記 root 密碼時，必須進入維護模式。
2. RHEL 6 開關機層級分爲 0,1,2,3,4,5,6 等層級，1 爲單人模式，也就維修模式。
3. RHEL 6 開機時在 grub 的 kernel 參數中指定開機進入單人模式，也就是層級 1，就可免 root 密碼進入系統互動式命令模式。
4. RHEL 7 後的版本開機已不分層級，而是以目標 (targets) 來分。
5. 開機分成 4 個目標 targets，其開機結果如下表：

Target	Purpose
graphical.target	多人模式，支援圖形及文字兩種方式登入，等效 v6 的層級 3 及 5。
multi-user.target	多人模式，只支援文字登入方式，等效 v6 的層級 3。
rescue.target	sulogin (Single-user login)，單人模式，等效 v6 的層級 1。
emergency.target	sulogin ，單人模式，完成 initramfs 載入且系統根目錄 / 掛載成唯讀，等效 v6 開機掛載不成功時的維修模式。

22.2 開機 targets

1. 指定 target 開機：
   (a) 出現 grub boot loader 開機選單時，立即碰觸鍵盤任意鍵，boot loader 會暫停。

(b) 按下'e'，編輯選單

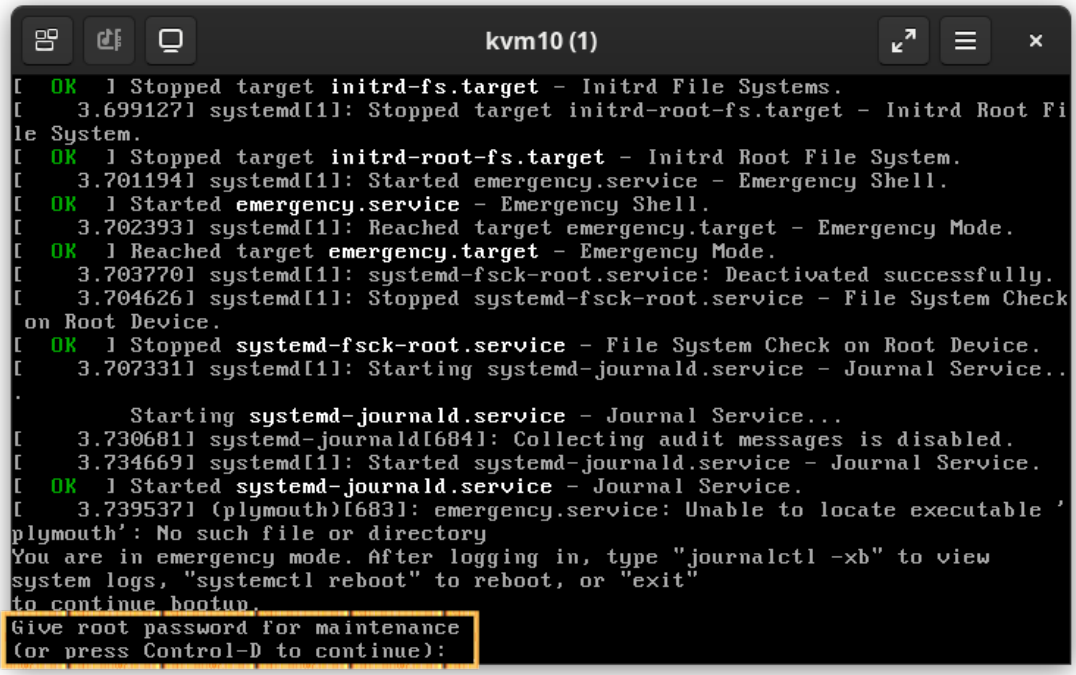
(c) 移動上下鍵至 linux 核心參數行

(d) 在核心參數行最後加入 systemd.unit=desired.target，desired 四種 target 之一，以 emergency.target 爲例。



(e) 按下 Ctrl+x 或 F10 重新以這個設定開機。

(f) 不管以哪一種 target 開機，都還是需要 root 密碼才能進行維修。

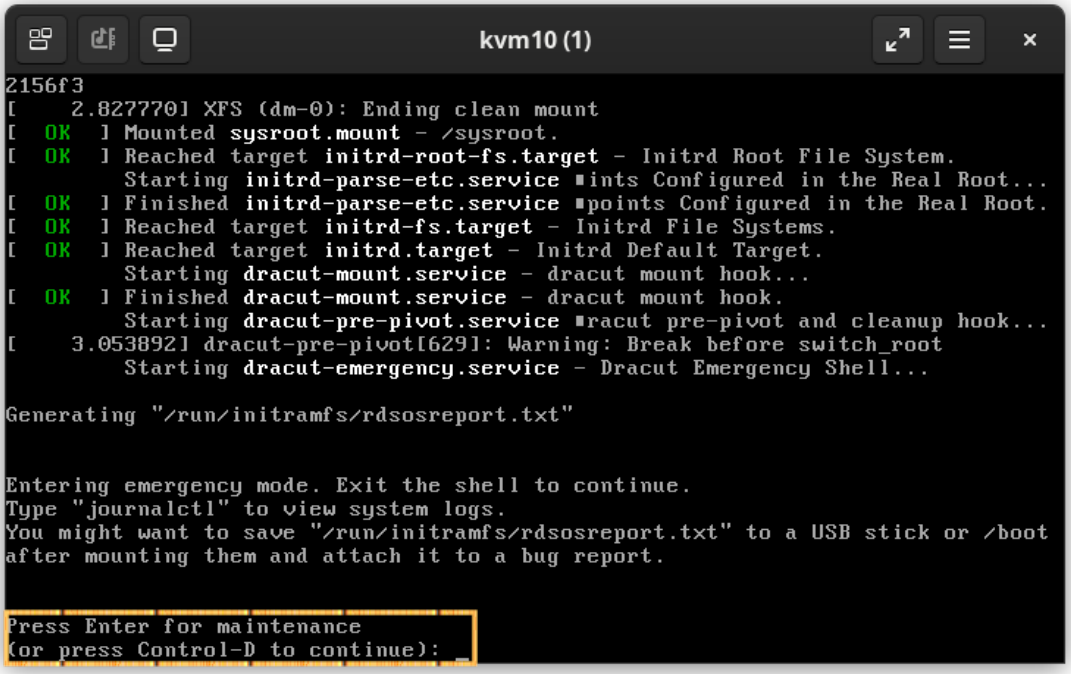


22.3 忘記密碼

1. 出現 grub boot loader 開機選單時，立即碰觸鍵盤任意鍵，boot loader 會暫停。
2. 選擇 rescue 開機選項，按下'e'，編輯選單。
3. 移動上下鍵至 linux 核心參數行行尾加入 rd.break。
4. 按下 Ctrl+x 或 F10 重新以這個設定開機。



5. 不用密碼，按 Enter 進入維修，或 Contol-D 繼續，按 Enter 進入維修。



6. 開機後的互動式命令環境是在 initramfs 這個開機系統上，不是最後開機的系統，最後開機系統目前掛載在 /sysroot，且掛載成唯讀，必須重新掛載成可寫入，才能變更密碼，步驟如下：

   (a) 重新掛載 /sysroot 成可讀可寫。

   ```
   sh-5.2# mount -oremount,rw /sysroot
   ```

   (b) chroot 工作目錄到 /sysroot，這個版本 initramfs 沒有 chroot 命令，所以借用 /sysroot 下的 /usr/sbin/chroot 命令。

   ```
   sh-5.2# /sysroot/usr/sbin/chroot /sysroot
   ```

   (c) 設定新的 root 密碼。

   ```
   sh-5.2# passwd
   ```

   (d) 因爲在此情況下，SELinux 並沒有啓動，對所有檔案的變更，可能會造成檔案的 context 不正確，爲確保開機時重新設定 SELinux context，必須在根目錄下產生隱藏檔.autorelabel。

   ```
   sh-5.2# touch /.autorelabel
   ```

(e) 退出 chroot

```
sh-5.2# exit
exit
sh-5.2#
```

(f) 退出系統，即重新開機。

```
sh-5.2# exit
```

22.4 實機練習

1. kvm10 root 密碼忘記了，重新設定 root 密碼爲 123qwe。

Chapter 23

DNF 套件管理

23.1 前言

1. 何謂 RPM？
   (a) RPM 全名是『RedHat Package Manager』簡稱爲 RPM。
   (b) 由 Red Hat 公司發展出來，由於 RPM 使用方便，所以成了目前最熱門的套件管理程式。
   (c) RPM 是以資料庫記錄的方式來將所需要的套件安裝到 Linux 主機的一套管理程式。
   (d) RPM 將要安裝的套件先編譯過，並且打包好，安裝時 RPM 會先依照套件裡的紀錄資料查詢相依屬性套件是否滿足，若滿足則予以安裝，若不滿足則不予安裝。
2. RPM 套件的屬性相依
   (a) RPM 打包套件檔案時，會同時加入套件的訊息。例如：
      i. 版本、
      ii. 打包套件者、
      iii. 相依屬性的套件、
      iv. 套件的功能說明、
      v. 套件的所有檔案與目錄紀錄。
   (b) 當要安裝某個以 RPM 型態提供的套件時，如果資料庫顯示其相依套件不存在，則會顯示錯誤訊息。
   (c) 屬性相依的克服方式
      i. 手動下載並安裝好所有相依套件。

ii. 利用 dnf 安裝 rpm 套件。

3. DNF

(a) RHEL/AlmaLinux 都提供套件庫 (repository) 供 dnf 下載安裝、升級、移除。

(b) 在安裝好 RHEL/AlmaLinux 系統後，都會將 DNF 的套件庫設定到同版本的下載連結點。

(c) 利用 dnf 安裝 rpm 套件時，dnf 會在設定的套件庫尋找未安裝的相依套件加以安裝。

(d) 也可自行建立套件庫，設定爲 dnf 的套件庫，dnf 就可以升級，安裝，移除套件庫中的套件。

23.2 dnf 套件庫設定

1. 設定 dnf 套件庫 (repository) 設定檔，必須在 /etc/yum.repo.d/ 目錄下，建立.repo 檔案，檔案設定方式可查詢 dnf.conf 手冊。

```
[root@kvm8 ~]# man dnf.conf
```

2. 撰寫 /etc/yum.repos.d/dywang.repo，建立 AlmaLinux 10 BaseOS 及 AppStream 兩類套件。

```
[root@kvm8 ~]# vi /etc/yum.repos.d/redhat.repo
[root@kvm8 ~]# cat /etc/yum.repos.d/redhat.repo
[AppStream]
name=App Stream
baseurl=http://dywang.csie.cyut.edu.tw/alma10/AppStream
gpgcheck=0

[BaseOS]
name=Base OS
baseurl=http://dywang.csie.cyut.edu.tw/alma10/BaseOS
gpgcheck=0
```

3. .repo 檔參數說明：

(a) [base]：套件庫的名稱，中刮號一定要存在，名稱可以自訂但不可重複。

(b) name：說明這個套件庫的意義。

(c) mirrorlist=：列出這個套件庫可以使用的映射站台，如果不想使用，可以註解到這行；

(d) baseurl=：指定固定套件庫的實際網址。

(e) enable=1：預設啓動套件庫，可以不設定。如果不想啓動可以使用 enable=0。

(f) gpgcheck=1：指定是否需要查閱 RPM 檔案內的數位簽章。

(g) gpgkey=：數位簽章的公鑰檔所在位置。

23.3 DNF 套件管理指令

1. dnf [Tab][Tab] 提示 dnf 工具有 112 可命令可用。

```
[root@kvm8 ~]# dnf [Tab][Tab]
Display all 112 possibilities? (y or n)
```

2. 列出 dnf 使用的所有套件庫

```
[root@kvm8 ~]# dnf repolist
repo id                                     repo name
AppStream                                   App Stream
BaseOs                                      Base OS
```

3. 搜尋 ftp 相關的套件

```
[root@kvm8 ~]# dnf search ftp
baseos                                          8.9 MB/s | 1.5 MB
    00:00
appstream                                        12 MB/s | 1.5 MB
    00:00
Last metadata expiration check: 0:00:01 ago on Wed 08 Oct 2025 02:54:42
    PM CST.
========================== Name & Summary Matched: ftp
    ==========================
ftp.x86_64 : The standard UNIX FTP (File Transfer Protocol) client
python3-requests-ftp.noarch : FTP transport adapter for python3-requests
syslinux-tftpboot.noarch : SYSLINUX modules in /tftpboot, available for
    network
                           : booting
tftp.x86_64 : The client for the Trivial File Transfer Protocol (TFTP)
tftp-server.x86_64 : The server for the Trivial File Transfer Protocol (
    TFTP)
vsftpd.x86_64 : Very Secure Ftp Daemon
```

```
================================ Name Matched: ftp
    ================================
lftp.x86_64 : A sophisticated file transfer program
============================== Summary Matched: ftp
    ==============================
curl.x86_64 : A utility for getting files from remote servers (FTP, HTTP
    , and
             : others)
nbdkit-curl-plugin.x86_64 : HTTP/FTP (cURL) plugin for nbdkit
wget.x86_64 : A utility for retrieving files using the HTTP or FTP
    protocols
```

4. 試安裝 vsftpd？如果出現 Is this ok [y//N]: 表示設定無誤可以取得套件進行安裝。y 安裝、N 不安裝，輸入 n 不安裝退出。

```
[root@kvm8 ~]# dnf install vsftpd
Last metadata expiration check: 3:23:43 ago on Tue 30 Sep 2025 03:11:06
    PM CST.
Dependencies resolved.
================================================================================
 Package         Architecture    Version               Repository
             Size
================================================================================

Installing:
 vsftpd          x86_64          3.0.5-9.el10          AppStream
             171 k

Transaction Summary
================================================================================

Install  1 Package

Total download size: 171 k
Installed size: 344 k
Is this ok [y/N]: n
Operation aborted.
```

23.4 實機練習

1. 建立 DNF 安裝套件倉庫來源

   (a) http://dywang.csie.cyut.edu.tw/alma10/BaseOS

   (b) http://dywang.csie.cyut.edu.tw/alma10/AppStream

2. 取消 RPM 檔案數位簽章檢查

3. 試著安裝套件 vsftpd

Chapter 24

Flatpak 套件管理

24.1 前言

1. Linux 不同的發行版有各種套件管理員，安裝應用程式指令都不一樣，還會有套件相依性 (dependencies) 的問題。

2. Flatpak 是 Linux 的軟體套件打包格式，同時支援各種發行版，解決應用程式相依需求。

24.2 Flatpak 命令說明

1. 安裝 flatpak 套件。

```
[root@kvm10 ~]# dnf install flatpak
```

2. flatpak 輔助說明。

```
[root@kvm10 ~]# flatpak --help
Usage:
  flatpak [OPTION…] COMMAND

Builtin Commands:
 Manage installed applications and runtimes
  install               Install an application or runtime
  update                Update an installed application or runtime
  uninstall             Uninstall an installed application or runtime
  mask                  Mask out updates and automatic installation
  pin                   Pin a runtime to prevent automatic removal
  list                  List installed apps and/or runtimes
  info                  Show info for installed app or runtime
  history               Show history
```

```
  config                  Configure flatpak
  repair                  Repair flatpak installation
  create-usb              Put applications or runtimes onto removable
      media
  preinstall              Install flatpaks that are part of the operating
       system

Find applications and runtimes
  search                  Search for remote apps/runtimes

Manage running applications
  run                     Run an application
  override                Override permissions for an application
  make-current            Specify default version to run
  enter                   Enter the namespace of a running application
  ps                      Enumerate running applications
  kill                    Stop a running application

Manage file access
  documents               List exported files
  document-export         Grant an application access to a specific file
  document-unexport       Revoke access to a specific file
  document-info           Show information about a specific file

Manage dynamic permissions
  permissions             List permissions
  permission-remove       Remove item from permission store
  permission-set          Set permissions
  permission-show         Show app permissions
  permission-reset        Reset app permissions

Manage remote repositories  管理遠端套件庫的指令
  remotes                 List all configured remotes
  remote-add              Add a new remote repository (by URL)
                          加一新遠端套件庫
  remote-modify           Modify properties of a configured remote
  remote-delete           Delete a configured remote
  remote-ls               List contents of a configured remote
  remote-info             Show information about a remote app or runtime

Build applications
  build-init              Initialize a directory for building
  build                   Run a build command inside the build dir
  build-finish            Finish a build dir for export
  build-export            Export a build dir to a repository
  build-bundle            Create a bundle file from a ref in a local
      repository
  build-import-bundle     Import a bundle file
  build-sign              Sign an application or runtime
  build-update-repo       Update the summary file in a repository
  build-commit-from       Create new commit based on existing ref
  repo                    Show information about a repo

Help Options:
```

```
  -h, --help              Show help options
67
Application Options:
69  --version               Print version information and exit
  --default-arch          Print default arch and exit
71  --supported-arches      Print supported arches and exit
  --gl-drivers            Print active gl drivers and exit
73  --installations         Print paths for system installations and exit
  --print-updated-env     Print the updated environment needed to run
      flatpaks
75  --print-system-only     Only include the system installation with --
      print-updated-env
  -v, --verbose           Show debug information, -vv for more detail
77  --ostree-verbose        Show OSTree debug information
```

3. flatpak 管理遠端套件庫的指令 remote-add 增加新的套件庫命令輔助說明：

```
1 [root@kvm10 ~]# flatpak remote-add --help

3 Note that the directories

5 '/var/lib/flatpak/exports/share'
  '/root/.local/share/flatpak/exports/share'
7
  are not in the search path set by the XDG_DATA_DIRS environment variable
      , so
9 applications installed by Flatpak may not appear on your desktop until
      the
  session is restarted.
11
  Usage:
13  flatpak remote-add [OPTION…] NAME LOCATION - Add a remote repository

15 Help Options:
    -h, --help                            Show help options
17
  Application Options:
19  --no-gpg-verify                       Disable GPG verification
    --no-enumerate                        Mark the remote as don't
        enumerate
21  --no-use-for-deps                     Mark the remote as don't use for
        deps
    --prio=PRIORITY                       Set priority (default 1, higher
        is more prioritized)
23  --subset=SUBSET                       The named subset to use for this
        remote
    --title=TITLE                         A nice name to use for this
        remote
25  --comment=COMMENT                     A one-line comment for this
        remote
    --description=DESCRIPTION             A full-paragraph description for
        this remote
```

```
  --homepage=URL                    URL for a website for this remote
  --icon=URL                        URL for an icon for this remote
  --default-branch=BRANCH           Default branch to use for this
      remote
  --collection-id=COLLECTION-ID     Collection ID
  --gpg-import=FILE                 Import GPG key from FILE (- for
      stdin)
  --filter=FILE                     Set path to local filter FILE
  --disable                         Disable the remote
  --authenticator-name=NAME         Name of authenticator
  --authenticator-option=KEY=VALUE  Authenticator option
  --authenticator-install           Autoinstall authenticator
  --no-authenticator-install        Don't autoinstall authenticator
  --no-follow_redirect              Don't follow the redirect set in
      the summary file
  -u, --user                        Work on the user installation
                                    指定只有某使用者可使用該套件庫
  --system                          Work on the system-wide
      installation (default)
  --installation=NAME               Work on a non-default system-wide
       installation
  --if-not-exists                   Do nothing if the provided remote
       exists
  --from                            LOCATION specifies a
      configuration file, not the repo location
  -v, --verbose                     Show debug information, -vv for
      more detail
  --ostree-verbose                  Show OSTree debug information
```

24.3 安裝 flatpak 套件庫

1. 切換成一般用户 deyu5。

```
[root@kvm10 ~]# su - deyu5
Last login: Tue Sep 30 16:19:38 CST 2025 on pts/0
```

2. flatpak -user 按 2 次 Tab 鍵，系統提示可用的所有命令。

```
[deyu5@kvm10 ~]$ flatpak --user [TAB][TAB]
build                 enter                 preinstall
build-bundle          history               ps
build-commit-from     info                  remote-add
build-export          install               remote-delete
build-finish          kill                  remote-info
build-import-bundle   list                  remote-ls
build-init            make-current          remote-modify
build-sign            mask                  remotes
```

```
10 build-update-repo     override              repair
   config                permission-remove     repo
12 create-usb            permission-reset      run
   document-export       permissions           search
14 document-info         permission-set        uninstall
   documents             permission-show       update
16 document-unexport     pin
```

3. 加入名爲 flathub 的遠端 flatpak 套件庫，--user 設定只有用户 deyu5 可使用。

```
[deyu5@kvm10 ~]$ flatpak remote-add --user --if-not-exists flathub \
2 http://dywang.csie.cyut.edu.tw/flathub.flatpakrepo
```

4. remotes 查看安裝的套件庫，有一名爲 flathub 的套件庫。

```
[deyu5@kvm10 ~]$ flatpak --user remotes
2 Name    Options
flathub
```

24.4 flatpak 搜尋與安裝

1. 切換成一般用户 deyu5。

```
```

2. search 搜尋 evince 套件。

```
[deyu5@kvm10 ~]$ flatpak --user search evince
2 Name       Description                    Application ID   Version Branch
     Remotes
Document ⋯ Document viewer for popular ⋯ org.gnome.Evince 48.1
    stable flathub
```

3. install 安裝 evince 套件。

```
1 [deyu5@kvm10 ~]$ flatpak --user install evince
Looking for matches⋯
```

```
Found ref 'app/org.gnome.Evince/x86_64/stable' in remote 'flathub' (
    user).
Use this ref? [Y/n]: y
Required runtime for org.gnome.Evince/x86_64/stable (runtime/org.gnome.
Platform/x86_64/48) found in remote flathub
Do you want to install it? [Y/n]: y

org.gnome.Evince permissions:
    ipc       fallback-x11          pulseaudio            wayland
        ID                    Branch      Op  Remote    Download
 1. [ ] org.freedesktop...    24.08       i   flathub   144.7 MB / 145.4
     MB
 2. [ ] org.freedesktop...    24.08extra  i   flathub    24.0 MB / 145.4
     MB
 3. [ ] org.freedesktop...    2.5.1       i   flathub   913.7 kB / 971.4
     kB
 4. [ ] org.gnome.Evince...   stable      i   flathub     9.8 kB / 1.5 MB
 5. [ ] org.gnome.Platform.   48          i   flathub    18.6 kB / 394.5
     MB
 6. [ ] org.gnome.Platform    48          i   flathub   346.2 MB / 404.9
     MB
 7. [ ] org.gnome.Evince      stable      i   flathub     6.9 MB / 12.2
     MB

Installation complete.
```

24.5 安裝後執行

1. list 列出安裝的套件，其中有安裝的 pdf 文件顯示器 Evince。

```
[deyu5@kvm10 ~]$ flatpak --user list
Name                          Application ID                  Version
    Branch
Mesa                          ···eedesktop.Platform.GL.default 25.2.3
    24.08
Mesa (Extra)                  ···eedesktop.Platform.GL.default 25.2.3
    24.08extra
openh264                      ···freedesktop.Platform.openh264 2.5.1
    2.5.1
Document Viewer               org.gnome.Evince                 48.1
    stable
GNOME Application Platform ··· org.gnome.Platform                    48
```

2. run 執行 evince，run 後必須接應用程式 ID，不必打字，可用 [Tab] 鍵提示及補齊。Evince 是 GUI 應用程式，執行時顯示無法開啓畫面。

```
[deyu5@kvm10 ~]$ flatpak --user run org.gnome.Evince
Cannot parse arguments: Cannot open display:
```

3. 切換身份回 root，測試 root 查不到 flatpak 套件庫，也沒有可用的應用程式。

```
[deyu5@kvm10 ~]$
logout
[root@kvm10 ~]# flatpak remotes

[root@kvm10 ~]# flatpak list
```

24.6 實機練習

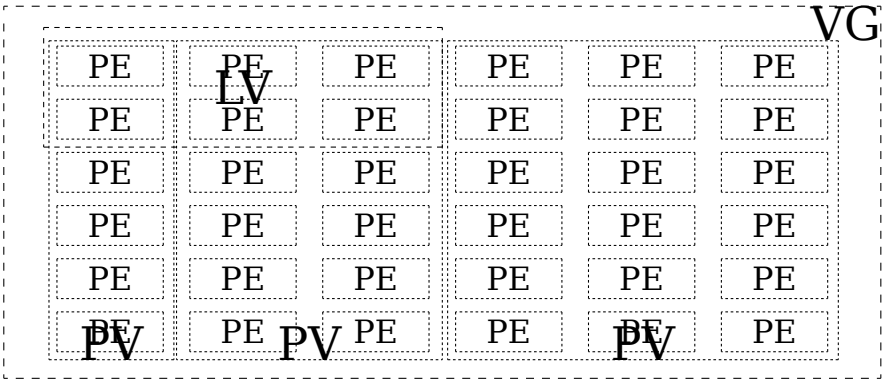
1. 用户 deyu5 建立專用 flatpak 套件庫，並安裝應用程式。
   (a) myflat eog nano 依參數作答
   (b) 使用 http://dywang.csie.cyut.edu.tw/flathub.flatpakrepo 設定名爲 flathub 的套件庫。
   (c) 查看安裝的套件庫，導向到 ~/flatpak.remotes。
   (d) 搜尋 eog 套件，導向到 ~/flatpak.search。
   (e) 安裝 nano
   (f) 列出安裝的套件，導向到 ~/flatpak.list。
   (g) info 查看 nano 的訊息，導向到 ~/flatpak.info。

Chapter 25

邏輯卷冊放大縮小

25.1 簡介

1. 何謂 LVM？
   - LVM (Logical Volume Manager)：邏輯卷冊管理。
   - LVM 目的爲彈性進行 filesystem 的擴充或縮小。
2. PV, PE, VG, LV 的意義
   - Physical Volume, PV, 實體捲軸：partition 時需要調整系統識別碼 (system ID) 成爲 44 (LVM 的識別碼)。
   - Volume Group, VG, 捲軸群組：可以由一或多個 PV 組合合成。
   - Physical Extend, PE, 實體延伸區塊：LVM 最小的儲存區塊。
   - Logical Volume, LV, 邏輯卷冊：從 VG 切割而成，爲最後可被格式化使用之格式，LV 的裝置檔名通常指定爲『/dev/vgname/lvname 』的樣式。



3. LV 放大縮小，要特別注意！lv 縮小要先卸載，無法卸載的分割區，不要使用此方法做。例如根目錄無法卸載，且 lvm 相關命令在 /sbin 目錄下，直接將掛載根目錄 / 的分割區，做放大縮小，風險非常高，可能造成資料的損壞。

25.2 lvreduce 縮小 LV

1. 查看 LV myvdb-vo 掛載在 /reports，大小爲 89M (100M)。

```
[root@kvm10 ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       853M     0  853M   0% /dev/shm
tmpfs                       342M  5.0M  337M   2% /run
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                   436M  211M  226M  49% /boot
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                       171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo         89M   14K   83M   1% /reports
```

2. lvreduce 縮小 LV myvdb-vo 檔案系統爲 50M。

```
[root@kvm10 ~]# lvreduce -L 50M /dev/myvdb/vo --resizefs
  Rounding size to boundary between physical extents: 52.00 MiB.
  File system ext4 found on myvdb/vo mounted at /reports.
  File system size (100.00 MiB) is larger than the requested size (52.00
      MiB).
  File system reduce is required using resize2fs.
  File system unmount is needed for reduce.  卸載
  File system fsck will be run before reduce.  檢查檔案系統
Continue with ext4 file system reduce steps: unmount, fsck, resize2fs? [
    y/n]:y
  Reducing file system ext4 to 52.00 MiB (54525952 bytes) on myvdb/vo...
unmount /reports
unmount done
e2fsck /dev/myvdb/vo
/dev/myvdb/vo: 11/25584 files (9.1% non-contiguous), 12081/102400 blocks
e2fsck done
resize2fs /dev/myvdb/vo 53248k
resize2fs 1.47.1 (20-May-2024)
Resizing the filesystem on /dev/myvdb/vo to 53248 (1k) blocks.
The filesystem on /dev/myvdb/vo is now 53248 (1k) blocks long.

resize2fs done
remount /dev/myvdb/vo /reports
remount done
  Reduced file system ext4 on myvdb/vo.
  Size of logical volume myvdb/vo changed from 100.00 MiB (25 extents)
      to 52.00 MiB (13 extents).
  Logical volume myvdb/vo successfully resized.
```

3. 再檢查 LV myvdb-vo 大小爲 44M (50M)。

```
[root@kvm10 ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       853M     0  853M   0% /dev/shm
tmpfs                       342M  5.0M  337M   2% /run
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                   436M  211M  226M  49% /boot
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                       171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo         44M   14K   41M   1% /reports
```

25.3 lvextend 放大 LV

1. 查看 LV myvdb-vo 掛載在 /reports，大小爲 24M (30M)。

```
[root@kvm10 ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       853M     0  853M   0% /dev/shm
tmpfs                       342M  5.0M  337M   2% /run
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                   436M  211M  226M  49% /boot
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                       171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo         44M   14K   41M   1% /reports
```

2. lvextend 放大 LV myvdb-vo 爲 100M。

```
[root@kvm10 ~]# lvextend -L 100M --resizefs /dev/myvdb/vo
  File system ext4 found on myvdb/vo mounted at /reports.
  Size of logical volume myvdb/vo changed from 52.00 MiB (13 extents) to
      100.00 MiB (25 extents).
  Extending file system ext4 to 100.00 MiB (104857600 bytes) on myvdb/vo
      ...
resize2fs /dev/myvdb/vo
resize2fs 1.47.1 (20-May-2024)
Filesystem at /dev/myvdb/vo is mounted on /reports; on-line resizing
    required
old_desc_blocks = 1, new_desc_blocks = 1
```

```
The filesystem on /dev/myvdb/vo is now 102400 (1k) blocks long.

resize2fs done
  Extended file system ext4 on myvdb/vo.
  Logical volume myvdb/vo successfully resized.
```

3. 再檢查 LV myvdb-vo 大小爲 89M (100M)。

```
[root@kvm10 ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       853M     0  853M   0% /dev/shm
tmpfs                       342M  5.0M  337M   2% /run
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                   436M  211M  226M  49% /boot
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                       171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo         89M   14K   85M   1% /reports
```

25.4 lvresize 放大縮小 LV

1. lvresize 指令不用理會是放大還是縮小，只要加上 -resizefs 選項，就可同時 resize ext 檔案系統的大小。

2. 先檢查 LV vo 在 VG myvdb 內，大小爲 89M (100M)。

```
[root@kvm10 ~]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                    4.0M     0  4.0M   0% /dev
tmpfs                       853M     0  853M   0% /dev/shm
tmpfs                       342M  4.9M  337M   2% /run
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                   436M  211M  226M  49% /boot
/dev/mapper/myvdb-vo         89M   14K   82M   1% /reports
tmpfs                       1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                       171M  8.0K  171M   1% /run/user/0
```

3. lv 檔案系統的檔案有兩種。

```
[root@kvm10 ~]# ll /dev/myvdb/vo
lrwxrwxrwx. 1 root root 7 Dec  4 14:09 /dev/myvdb/vo -> ../dm-2
[root@kvm10 ~]# ll /dev/mapper/myvdb-vo
lrwxrwxrwx. 1 root root 7 Dec  4 14:09 /dev/mapper/myvdb-vo -> ../dm-2
```

4. LV vo 大小改爲 50M，其中選項 -resize2fs 是同時 resize ext 檔案系統的大小。

```
[root@kvm10 ~]# lvresize -L 50M --resizefs /dev/mapper/myvdb-vo
  Rounding size to boundary between physical extents: 52.00 MiB.
  File system ext4 found on myvdb/vo mounted at /reports.
  File system size (100.00 MiB) is larger than the requested size (52.00
      MiB).
  File system reduce is required using resize2fs.
  File system unmount is needed for reduce.
  File system fsck will be run before reduce.
Continue with ext4 file system reduce steps: unmount, fsck, resize2fs? [
    y/n]:y
  Reducing file system ext4 to 52.00 MiB (54525952 bytes) on myvdb/vo...
unmount /reports
unmount done
e2fsck /dev/myvdb/vo
/dev/myvdb/vo: 11/25584 files (0.0% non-contiguous), 12081/102400 blocks
e2fsck done
resize2fs /dev/myvdb/vo 53248k
resize2fs 1.47.1 (20-May-2024)
Resizing the filesystem on /dev/myvdb/vo to 53248 (1k) blocks.
The filesystem on /dev/myvdb/vo is now 53248 (1k) blocks long.

resize2fs done
remount /dev/myvdb/vo /reports
remount done
  Reduced file system ext4 on myvdb/vo.
  Size of logical volume myvdb/vo changed from 100.00 MiB (25 extents)
      to 52.00 MiB (13 extents).
  Logical volume myvdb/vo successfully resized.
```

5. 再檢查 LV myvdb-vo 大小爲 44M (50M)。

```
[root@kvm10 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                  4.0M     0  4.0M   0% /dev
tmpfs                     853M     0  853M   0% /dev/shm
tmpfs                     342M  4.9M  337M   2% /run
tmpfs                     1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                 436M  211M  226M  49% /boot
```

```
tmpfs                     1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                     171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo       44M   14K   40M   1% /reports
```

6. 以單一指令 lvresize 將 LV myvdb-vo 大小改爲 250M，不用理會是放大還是縮小，其中 -resizefs 選項是一併執行 resize2fs 的命令。

```
[root@kvm10 ~]# lvresize -L 250M --resizefs /dev/mapper/myvdb-vo
  Rounding size to boundary between physical extents: 252.00 MiB.
  File system ext4 found on myvdb/vo mounted at /reports.
  Size of logical volume myvdb/vo changed from 52.00 MiB (13 extents) to
       252.00 MiB (63 extents).
  Extending file system ext4 to 252.00 MiB (264241152 bytes) on myvdb/vo
      ...
resize2fs /dev/myvdb/vo
resize2fs 1.47.1 (20-May-2024)
Filesystem at /dev/myvdb/vo is mounted on /reports; on-line resizing
    required
old_desc_blocks = 1, new_desc_blocks = 2
The filesystem on /dev/myvdb/vo is now 258048 (1k) blocks long.

resize2fs done
  Extended file system ext4 on myvdb/vo.
```

7. 再檢查 LV myvdb-vo 大小爲 231M (250M)。

```
[root@kvm10 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root  4.3G  3.1G  1.2G  73% /
devtmpfs                  4.0M     0  4.0M   0% /dev
tmpfs                     853M     0  853M   0% /dev/shm
tmpfs                     342M  4.9M  337M   2% /run
tmpfs                     1.0M     0  1.0M   0% /run/credentials/
    systemd-journald.service
/dev/vda2                 436M  211M  226M  49% /boot
tmpfs                     1.0M     0  1.0M   0% /run/credentials/
    getty@tty1.service
tmpfs                     171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo      231M   14K  220M   1% /reports
```

25.5 實機練習

1. df -h 查詢目前掛載的檔案系統，其中名爲 vo 的 LV 大小，將其寫到 /root/lvsize0，單位 M 不用寫。

2. 比較 vosize 與查詢到的 vo 大小，是放大還是縮小？將結果寫到 /root/lvmod，放大寫入”extend”，縮小寫入”reduce”。

3. 決定使用標準的放大縮小作法，或使用單一指令，完成 lv 的大小調整。將 vo 的大小調整爲 vosize。

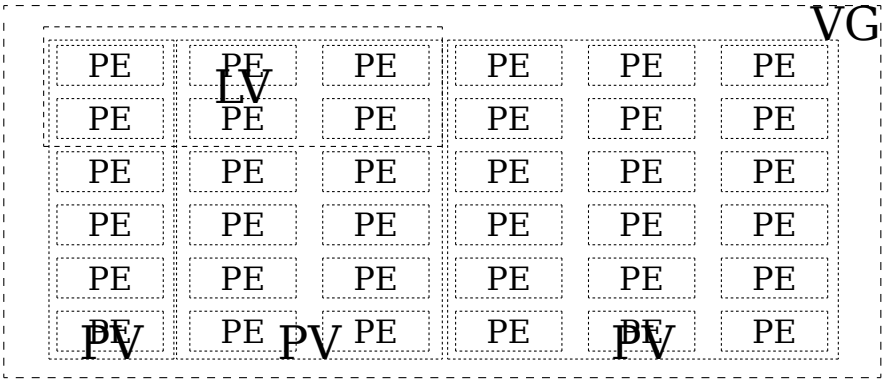
4. df -h 再查詢掛載的檔案系統，觀察名爲 vo 的 LV 大小是否已經調整爲 vosize？將其大小寫到 /root/lvsize1，單位 M 不用寫，查到的大小與 vosize 差十幾 M 都是正確。

Chapter 26

邏輯卷冊管理

26.1 前言

1. LVM (Logical Volume Manager)：邏輯卷冊管理，目的爲彈性進行 filesystem 的擴充或縮小。
2. PV, PE, VG, LV 的意義
    - Physical Volume, PV, 實體捲軸：partition 時需要調整系統識別碼 (system ID) 成爲 44 (LVM 的識別碼)。
    - Volume Group, VG, 捲軸群組：可以由一或多個 PV 組合合成。
    - Physical Extend, PE, 實體延伸區塊：LVM 最小的儲存區塊。
    - Logical Volume, LV, 邏輯卷冊：從 VG 切割而成，爲最後可被格式化使用之格式，LV 的裝置檔名通常指定爲『/dev/vgname/lvname 』的樣式。



26.2 新增分割區

1. fdisk 進行 vdb 分割區管理。

```
[root@kvm10 ~]# fdisk /dev/vdb

Welcome to fdisk (util-linux 2.40.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

This disk is currently in use - repartitioning is probably a bad idea.
It's recommended to umount all file systems, and swapoff all swap
partitions on this disk.
```

2. 命令 m：可以查看輔助說明

```
Command (m for help): m

Help:

  GPT
   M   enter protective/hybrid MBR

  Generic
   d   delete a partition
   F   list free unpartitioned space
   l   list known partition types
   n   add a new partition
   p   print the partition table
   t   change a partition type
   v   verify the partition table
   i   print information about a partition
   e   resize a partition

  Misc
   m   print this menu
   x   extra functionality (experts only)

  Script
   I   load disk layout from sfdisk script file
   O   dump disk layout to sfdisk script file

  Save & Exit
   w   write table to disk and exit
   q   quit without saving changes

  Create a new label
   g   create a new empty GPT partition table
   G   create a new empty SGI (IRIX) partition table
   o   create a new empty MBR (DOS) partition table
   s   create a new empty Sun partition table
```

3. 命令 p：列出分割表，目前有 vdb1 一個分割區。

```
Command (m for help): p

Disk /dev/vdb: 3 GiB, 3221225472 bytes, 6291456 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 289C3DC7-CC6A-489A-9CB5-552D550FDAD3

Device     Start    End Sectors  Size Type
/dev/vdb1   2048 821247  819200  400M Linux LVM
```

4. 命令 n：新增分割區，都直接按 Enter 使用預設建議：新增編號 2 的分割區，分割區起始及結尾 sector 都用建議值，使用 vdb 剩餘所有空間。

```
Command (m for help): n
Partition number (2-128, default 2):
First sector (821248-6291422, default 821248):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (821248-6291422, default
    6289407):

Created a new partition 2 of type 'Linux filesystem' and of size 2.6 GiB
    .
```

5. 命令 p：再列出分割區，出現 vdb2 分割區。

```
Command (m for help): p
Disk /dev/vdb: 3 GiB, 3221225472 bytes, 6291456 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 289C3DC7-CC6A-489A-9CB5-552D550FDAD3

Device      Start     End Sectors  Size Type
/dev/vdb1    2048  821247  819200  400M Linux LVM
/dev/vdb2  821248 6289407 5468160  2.6G Linux filesystem
```

6. 命令 t：變更 vdb2 分割區的 Type。

```
Command (m for help): t
Partition number (1,2, default 2):
```

7. 命令 L：查詢 Type 編號。

```
Partition type or alias (type L to list all): L
  1 EFI System                     C12A7328-F81F-11D2-BA4B-00A0C93EC93B
  2 MBR partition scheme           024DEE41-33E7-11D3-9D69-0008C781F39F
  3 Intel Fast Flash               D3BFE2DE-3DAF-11DF-BA40-E3A556D89593
  4 BIOS boot                      21686148-6449-6E6F-744E-656564454649
.....................................................
 44 Linux LVM                      E6D6D379-F507-44C2-A23C-238F2A3DF928
.....................................................
```

8. 命令 44：使用 Linux LVM Type。

```
Partition type or alias (type L to list all): 44

Changed type of partition 'Linux filesystem' to 'Linux LVM'.
```

9. 命令 p：再列出分割區 vdb2 的 Type 已改成 Linux LVM。

```
Command (m for help): p
Disk /dev/vdb: 3 GiB, 3221225472 bytes, 6291456 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 289C3DC7-CC6A-489A-9CB5-552D550FDAD3

Device      Start     End Sectors  Size Type
/dev/vdb1    2048  821247  819200  400M Linux LVM
/dev/vdb2  821248 6289407 5468160  2.6G Linux LVM
```

10. 命令 w：寫入分割區表並退出。

```
Command (m for help): w
The partition table has been altered.
Syncing disks.
```

11. lsblk 查詢硬碟，已出現 vdb2。

```
[root@kvm10 ~]# lsblk
NAME              MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
vda               252:0    0    6G  0 disk
  vda1            252:1    0    1M  0 part
```

```
  vda2            252:2    0  500M  0 part /boot
  vda3            252:3    0  4.4G  0 part
    vg_kvm10-root 253:0    0  4.3G  0 lvm  /
    vg_kvm10-swap 253:1    0  124M  0 lvm  [SWAP]
vdb               252:16   0    3G  0 disk
  vdb1            252:17   0  400M  0 part
    myvdb-vo      253:2    0  252M  0 lvm  /reports
  vdb2            252:18   0  2.6G  0 part
```

26.3 產生 PV

1. 查看 pv 可用指令。

```
[root@kvm10 ~]# pv [TAB}{TAB]
pvchange   pvcreate   pvmove     pvresize   pvscan
pvck       pvdisplay  pvremove   pvs
```

2. pvcreate 輔助說明。

```
[root@kvm10 ~]# pvcreate --help
  pvcreate - Initialize physical volume(s) for use by LVM

  pvcreate PV ...
    [ -f|--force ]
.............................................................
  Use --longhelp to show all options and advanced commands.
```

3. pvs 查看目前只有兩個 PV。

```
[root@kvm10 ~]# pvs
  PV         VG       Fmt  Attr PSize   PFree
  /dev/vda3  vg_kvm10 lvm2 a--    4.39g      0
  /dev/vdb1  myvdb    lvm2 a--  396.00m 296.00m
```

4. 將 /dev/vdb2 建立爲 PV。

```
[root@kvm10 ~]# pvcreate /dev/vdb2
  Physical volume "/dev/vdb2" successfully created.
```

5. pvs 再查看已新增 /dev/vdb2 PV，但還沒有 VG。

```
[root@kvm10 ~]# pvs
  PV         VG       Fmt  Attr PSize   PFree
  /dev/vda3  vg_kvm10 lvm2 a--    4.39g      0
  /dev/vdb1  myvdb    lvm2 a--  396.00m 144.00m
  /dev/vdb2           lvm2 ---   <2.61g  <2.61g
```

26.4 產生 VG

1. 查看 vg 可用指令。

```
[root@kvm10 ~]# vg
vgcfgbackup      vgcreate         vgimportclone    vgremove
vgcfgrestore     vgdisplay        vgimportdevices  vgrename
vgchange         vgexport         vgmerge          vgs
vgck             vgextend         vgmknodes        vgscan
vgconvert        vgimport         vgreduce         vgsplit
```

2. VG 輔助說明。

```
[root@kvm10 ~]# vgcreate --help
  vgcreate - Create a volume group

  vgcreate VG_new PV ...
  ..................................
    [ -s|--physicalextentsize Size[m|UNIT] ]
```

3. vgs 查詢 VG，目前有兩個 VG。

```
[root@kvm10 ~]# vgs
  VG       #PV #LV #SN Attr   VSize   VFree
  myvdb      1   1   0 wz--n- 396.00m 144.00m
  vg_kvm10   1   2   0 wz--n-   4.39g      0
```

4. 在 PV /dev/vdb2 上建立名爲 vg_mnt PE size 2M 的 VG。

```
[root@kvm10 ~]# vgcreate --physicalextentsize 2M vg_mnt /dev/vdb2
  Volume group "vg_mnt" successfully created
```

5. 再查看 VG，出現 vg_mnt。

```
[root@kvm10 ~]# vgs
  VG       #PV #LV #SN Attr   VSize   VFree
  myvdb      1   1   0 wz--n- 396.00m 144.00m
  vg_kvm10   1   2   0 wz--n-   4.39g      0
  vg_mnt     1   0   0 wz--n-  <2.61g  <2.61g
```

26.5 產生 LV

1. 查看 lv 可用指令。

```
[root@kvm10 ~]# lv [TAB][TAB]
lvchange        lvm             lvm_import_vdo  lvremove
lvconvert       lvmconfig       lvmpolld        lvrename
lvcreate        lvmdevices      lvmsadc         lvresize
lvdisplay       lvmdiskscan     lvmsar          lvs
lvextend        lvmdump         lvreduce        lvscan
[root@kvm10 ~]# lvcreate --help
  lvcreate - Create a logical volume

  Create a linear LV.
  lvcreate -L|--size Size[m|UNIT] VG
........................................
    [ -l|--extents Number[PERCENT] ]
........................................
    [ -n|--name String ]
```

2. 在 vg_mnt 上建立名爲 pub，大小爲 12 個 PE 的 LV。

```
[root@kvm10 ~]# lvcreate --extents 12 --name pub vg_mnt
  Logical volume "pub" created.
```

3. 查看 LV 出現 pub 在 vg_mnt VG 內。

```
[root@kvm10 ~]# lvs
  LV   VG       Attr       LSize   Pool Origin Data%  Meta%  Move Log
       Cpy%Sync Convert
  vo   myvdb    -wi-ao---- 252.00m
  root vg_kvm10 -wi-ao----  <4.27g
  swap vg_kvm10 -wi-ao---- 124.00m
  pub  vg_mnt   -wi-a-----  24.00m
```

4. 在 vg_mnt 再新建立名爲 swap，大小爲 120M 的 LV，做爲 swap。

```
[root@kvm10 ~]# lvcreate -L 120M --name swap vg_mnt
  Logical volume "swap" created.
```

5. 再查看 LV 出現 swap 在 vg_mnt VG 內。

```
[root@kvm10 ~]# lvs
  LV   VG       Attr       LSize   Pool Origin Data%  Meta%  Move Log
      Cpy%Sync Convert
  vo   myvdb    -wi-ao---- 252.00m
  root vg_kvm10 -wi-ao----  <4.27g
  swap vg_kvm10 -wi-ao---- 124.00m
  pub  vg_mnt   -wi-a-----  24.00m
  swap vg_mnt   -wi-a----- 120.00m
```

26.6 格式化

1. 依照要求 (此例爲 vfat) 格式化 vg_mnt-pub。

```
[root@kvm10 ~]# mkfs.vfat /dev/vg_mnt/pub
mkfs.fat 4.2 (2021-01-31)
```

2. 格式化 vg_mnt-swap爲 swap 。

```
[root@kvm10 ~]# mkswap /dev/vg_mnt/swap
Setting up swapspace version 1, size = 120 MiB (125825024 bytes)
no label, UUID=464fbe18-97a2-43c6-a3e9-4278097d0998
```

26.7 開機掛載表 /etc/fstab

1. 系統掛載限制：

   (a) 根目錄 / 一定要掛載且最先掛載。

   (b) 其它掛載點 (mount points) 必須爲自已建立的目錄。

   (c) 所有 mount points 在同一時間之內，只能掛載一次。

(d) 所有 partitions 在同一時間之內，只能掛載一次。

(e) 若要卸載，必須先將工作目錄移到 mount point (及其子目錄) 之外。

2. /etc/fstab 共有六個欄位，細節請參考手冊。

```
[root@kvm10 ~]# man 5 /etc/fstab
```

(a) 磁碟裝置代號或該裝置的 Label

(b) 掛載點 (mount point)

(c) 檔案系統格式

(d) 檔案系統掛載參數

(e) 能否被 dump 備份指令作用：0 表示不做 dump 備份，1, 2 表示要進行 dump 的動作，1 優先於 2。

(f) 是否以 fsck 檢驗磁區：0 是不要檢驗，1, 2 是要檢驗，1 優先於 2。

26.8 開機掛載 FS

1. 建立 vg_mnt-pub 掛載目錄 (此例爲 /mnt/pub)。

```
[root@kvm10 ~]# mkdir /mnt/pub
```

2. 編輯 /etc/fstab 設定開機自動掛載 vg_mnt-pub 於 /mnt/pub，vg_mnt-swap 爲 swap，建議選擇 /etc/fstab 類似的行，yy 複製，p 貼上，再來修改。

```
[root@kvm10 ~]# vi /etc/fstab
[root@kvm10 ~]# tail -2 /etc/fstab
/dev/vg_mnt/pub                         /mnt/pub    vfat    defaults
    0 0
/dev/vg_mnt/swap                        none        swap    defaults
    0 0
```

3. 重新依 /etc/fstab 掛載表掛載檔案系統，因爲修改了 /etc/fstab，必須執行 systemctl daemon-reload。

```
[root@kvm10 ~]# mount -a
mount: (hint) your fstab has been modified, but systemd still uses
       the old version; use 'systemctl daemon-reload' to reload.
```

```
[root@kvm10 ~]# systemctl daemon-reload
```

4. 再重新依 /etc/fstab 掛載表掛載檔案系統，如果沒有錯誤訊息表示正常。

```
[root@kvm10 ~]# mount -a
```

5. 查看 vg_mnt-pub 已成功掛載於 /mnt/pub，且 type 爲 vfat。

```
[root@kvm10 ~]# df -Th
Filesystem                Type      Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm10-root xfs       4.3G  3.1G  1.2G  73% /
devtmpfs                  devtmpfs  4.0M     0  4.0M   0% /dev
tmpfs                     tmpfs     853M     0  853M   0% /dev/shm
tmpfs                     tmpfs     342M  5.0M  337M   2% /run
tmpfs                     tmpfs     1.0M     0  1.0M   0% /run/
    credentials/systemd-journald.service
/dev/vda2                 xfs       436M  211M  226M  49% /boot
tmpfs                     tmpfs     1.0M     0  1.0M   0% /run/
    credentials/getty@tty1.service
tmpfs                     tmpfs     171M  8.0K  171M   1% /run/user/0
/dev/mapper/myvdb-vo      ext4      231M   14K  220M   1% /reports
/dev/mapper/vg_mnt-pub    vfat       24M     0   24M   0% /mnt/pub
```

6. 檢查 VG vg_mnt 的 PE size 是否如要求 (此例爲 2M)。

```
[root@kvm10 ~]# vgdisplay vg_mnt
  --- Volume group ---
  VG Name               vg_mnt
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  3
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                2
  Open LV               2
  Max PV                0
  Cur PV                1
  Act PV                1
  VG Size               <2.61 GiB
  PE Size               2.00 MiB
  Total PE              1334
  Alloc PE / Size       72 / 144.00 MiB
  Free  PE / Size       1262 / 2.46 GiB
  VG UUID               nRn8UJ-TPLK-AOCY-VV5F-tg6N-yx4S-dgigZ1
```

7. 檢查 LV vg_mnt-pub 的 LE 是否如要求 (此例爲 12 個 PE)。

```
[root@kvm10 ~]# lvdisplay /dev/vg_mnt/pub
  --- Logical volume ---
  LV Path                /dev/vg_mnt/pub
  LV Name                pub
  VG Name                vg_mnt
  LV UUID                w6B3rZ-y4E6-UjS9-hlOB-i9h6-Fv8Q-f2DD6F
  LV Write Access        read/write
  LV Creation host, time kvm10.deyu.wang, 2025-12-04 14:36:16 +0800
  LV Status              available
  # open                 1
  LV Size                24.00 MiB
  Current LE             12
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     8192
  Block device           253:3
```

26.9 開機啓動 Swap

1. 查詢啓動的 swap，只有 /dev/dm-1。

```
[root@kvm10 ~]# swapon --show
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition 124M   0B   -2
```

2. 啓動掛載表 /etc/fstab 中設定的所有 swap。

```
[root@kvm10 ~]# swapon --all
```

3. 再查詢啓動的 swap，其中 /dev/dm-4 就是 LV vg_mnt-swap，大小是 120M。

```
[root@kvm10 ~]# swapon --show
NAME      TYPE      SIZE USED PRIO
/dev/dm-1 partition 124M   0B   -2
/dev/dm-4 partition 120M   0B   -3
```

26.10 實機練習

1. 在 kvm10 的硬碟 vdb 新增一分割區，大小包含硬碟未使用的全部空間，type 是 Linux LVM。
2. pvcreate 初始化新分割的 LVM 空間。
3. 在新初始化的 LVM 實體空間，建立名爲vg_mnt，PE 大小爲 2M 的 VG。
4. 在新建立的vg_mnt，新增名爲 pub，大小爲 12 個 PE 的 LV。
5. 在新建立的vg_mnt，新增名爲 swap，大小爲 48M 的 LV。
6. 將名爲 pub LV 格式化爲 vfat。
7. 將名爲 swap LV 格式化爲 swap。
8. 建立目錄 /mnt/pub。
9. 設定開機自動掛載名爲 pub 的 LV 於目錄 /mnt/pub。
10. 設定開機自動啓動名爲 swap 的 LV 爲 swap。

Chapter 27

Tuned 性能調整

27.1 前言

1. Tuned 是一個強大的服務程序，可根據系統使用情況信息，自動動態調整 Linux 的設定，以提昇性能。
2. 自動動態調整設定取決於系統活動及使用的調整配置文件 (tuning profile)。
3. 預設情況下，tuned 不會動態調整系統設置，但可經由修改 tuned 服務程序的運行方式，允許它根據系統使用情況動態更改設置。
4. 可使用 tuned-adm 命令工具管理服務程序的運行。
5. 查看 tuned 服務狀態，開機預設是啓動，且目前是正常運作中。

```
[root@kvm10 ~]# systemctl status tuned.service
  tuned.service - Dynamic System Tuning Daemon
     Loaded: loaded (/usr/lib/systemd/system/tuned.service; enabled;
         preset: en>
     Active: active (running) since Thu 2025-12-04 14:09:17 CST; 49min
         ago
 Invocation: 27dab42f05d443fea76db6505afe14cb
       Docs: man:tuned(8)
             man:tuned.conf(5)
             man:tuned-adm(8)
   Main PID: 854 (tuned)
      Tasks: 4 (limit: 10698)
     Memory: 17.2M (peak: 19.1M)
        CPU: 2.116s
     CGroup: /system.slice/tuned.service
               854 /usr/bin/python3 -Es /usr/sbin/tuned -l -P

Dec 04 14:09:16 kvm10.deyu.wang systemd[1]: Starting tuned.service -
    Dynamic Sy>
Dec 04 14:09:17 kvm10.deyu.wang systemd[1]: Started tuned.service -
    Dynamic Sys>
```

27.2 tuned-adm 命令

1. tuned-adm 輔助說明。

```
[root@kvm10 ~]# tuned-adm --help
usage: tuned-adm [-h] [--version] [--debug] [--async] [--timeout TIMEOUT
    ]
                 [--loglevel LOGLEVEL]
                 {list,active,off,profile,profile_info,recommend,verify,
                 auto_profile,profile_mode,instance_acquire_devices,
                 get_instances,instance_get_devices}
                 ...

Manage tuned daemon.

positional arguments:
  {list,active,off,profile,profile_info,recommend,verify,auto_profile,
  profile_mode,instance_acquire_devices,get_instances,
      instance_get_devices}
    列出可用的配置
    list                  list available profiles or plugins (by default
                          profiles)
    列出目前的配置
    active                show active profile
    off                   switch off all tunings
    切換可用的配置
    profile               switch to a given profile, or list available
        profiles
                          if no profile is given
    profile_info          show information/description of given profile or
                          current profile if no profile is specified
    列出建議的配置
    recommend             recommend profile
    verify                verify profile
    auto_profile          enable automatic profile selection mode, switch
        to the
                          recommended profile
    profile_mode          show current profile selection mode
    instance_acquire_devices
                          acquire devices from other instances and assign
                              them
                          to the given instance
    get_instances         list active instances of a given plugin or all
        active
                          instances if no plugin is specified
    instance_get_devices
                          list devices assigned to a given instance
```

```
39 options:
   -h, --help            show this help message and exit
41 --version, -v         show program's version number and exit
   --debug, -d           show debug messages
43 --async, -a           with dbus do not wait on commands completion and
                         return immediately
45 --timeout TIMEOUT, -t TIMEOUT
                         with sync operation use specific timeout instead
                              of
47                       the default 600 second(s)
   --loglevel LOGLEVEL, -l LOGLEVEL
49                       level of log messages to capture (one of debug,
                            info,
                         warn, error, console, none). Default: console
```

27.3 調整工作配置

1. 查詢目前使用的 profile。

```
[root@kvm10 ~]# tuned-adm active
2 Current active profile: throughput-performance
```

2. 查看建議使用的 profile 爲 virtual-guest。

```
[root@kvm10 ~]# tuned-adm recommend
2 virtual-guest
```

3. 將工作 profile 切換到系統建議的 virtual-guest。

```
[root@kvm10 ~]# tuned-adm profile virtual-guest
```

4. 再查看目前使用的 profile 已改成 virtual-guest。

```
1 [root@kvm10 ~]# tuned-adm active
Current active profile: virtual-guest
```

27.4 實機練習

1. ll 查詢 /usr/lib/tuned 目錄下的配置文件，導向到 /root/tunedprofiles。
2. 查詢 tuned 服務是否設定開機啓動，導向到 /root/tuned-enabled。
3. 查詢 tuned 服務現在是否啓動，導向到 /root/tuned-active。
4. 查詢 tuned 服務目前啓用的 profile，並將查詢命令寫成腳本 /root/tuned.active.sh。
5. 查詢 tuned 服務建議的 profile，並將命令寫成腳本 /root/tuned.recommend.sh。
6. 將預設的 profile 設定成到系統建議的 profile。