Linux Shell

- 快取指令
- Linux安裝SOP
- mdadm指令
- monit設定
- vim簡單語法
- SMART使用指令

快取指令

版本檢視

Centos

```
# cat /etc/redhat-release
CentOS Linux release 7.9.2009 (Core)
```

然後確認一下目前系統的核心版本，指令如下:

```
# uname -a
Linux oxoolcommunity.ossii.com.tw 3.10.0-1160.66.1.el7.x86_64 #1 SMP Wed May 18 16:02:34 UTC 2022
x86_64 x86_64 x86_64 GNU/Linux
```

備註:上方指令所有**linux**都可以使用

目前 CentOS **3.10.0-1160.66.1.el7.x86_64**，可以再使用 yum update 指令更新。

Debian

可以使用以下指令

```
cat /etc/debian_version
```

Ubuntu

```
lsb_release -a
```

```
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.5 LTS
Release:        22.04
Codename:       jammy
```

```
cat /etc/issue
```

```
Ubuntu 22.04.5 LTS \n \l
```

基本語法

!/bin/sh (或者是 #!/bin/bash) #宣佈shell指令

$? : 表示上一個指令的離開狀況，一般指令正常離開會傳回 0。不正常離開則會傳回 1、2 等數值。

$1 : 表示輸入的第一個參數，$2 則為第二個參數，依此類推。

$0 : shell script的檔名。

$@ : 即代表 $1, $2,....直到所有參數結束。也就是說 $@ 代表了 “$1” “$2” “$3”....。

$* : 所有參數無間隔的連在一起，成為單一個參數。也就是說 $* 代表了 “$1 $2 $3...”

使字串中包含', 可用反斜線 \ 跳脫。

雙引號""中若含有變數$var，會先將變數轉換成其實際的值，單引號''則會將$var當成是一個值，而不會作轉換動作。

echo "你們輸入的年齡是 $Y_AGE + $F_AGE 歲"
echo "你們輸入的年齡是 $((Y_AGE + F_AGE)) 歲"

```
#!/bin/bash
read -p "Please enter your age:" Y_AGE
read -p "Please enter your friend's age:" F_AGEsum1=$Y_AGE + $F_AGE
sum2=`expr $Y_AGE + $F_AGE`
sum3=$((Y_AGE + F_AGE))echo "你們輸入的年齡是 $Y_AGE + $F_AGE 歲"
echo "你們輸入的年齡是 $((Y_AGE + F_AGE)) 歲"
#Bash Shell 中內建原生不支援運算式，所以無法顯示
echo "你們輸入的年齡是 $sum1 歲"
echo "你們輸入的年齡是 $sum2 歲"
echo "你們輸入的年齡是 $sum3 歲"
```

新增使用者

```
#再ubuntu使用adduser會進行完整設定
sudo adduser {使用者名稱}
```

```
ron@DESKTOP-MB641C8:~$ sudo adduser ron1
Adding user `ron1' ...
Adding new group `ron1' (1001) ...
Adding new user `ron1' (1001) with group `ron1' ...
The home directory `/home/ron1' already exists.  Not copying from `/etc/skel'.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for ron1
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n]
```

[illegible]EX:sudo adduser {username} sudo

[illegible]useradd[illegible]([illegible])

```
-d [illegible] home[illegible]
-p [illegible]
-e [illegible]YYYY-MM-DD
-g [illegible]
-G [illegible]
-M [illegible]
-m [illegible]/etc/skel/[illegible]
```

[illegible]

```
useradd {username}
```

[illegible]
-d [illegible]
-g [illegible]
-G [illegible]
-e [illegible]
-f [illegible]
-s [illegible]shell

[illegible]ssh[illegible]sftp

Match[illegible]UsePAM[illegible]

```
Match user user名稱
ChrootDirectory /srv/ftp #更改根目錄
AllowTcpForwarding yes #可以使用ssh tcp forwarding
ForceCommand “強制指令”
```

使用特殊ip

Match User ansible,backup,support Address *,!176.x.x.x #加! 代表不要
#搭配hostname
Match user 使用者 Host “hostname”
#使用ip
Match address IP位置
#使用port
Match Localport 埠位

ex:
Match User user2, LocalPort 2201DenyUsers user2
備註: 可以使用正則表示法

使用服務端可以使用的port位

ListenAddress hostname|address

ListenAddress hostname:port

ListenAddress IPv4_address:port

ListenAddress [hostname|address]:port

使用AllowUsers 跟AllowGroups或denyUsers denyGroups來限制使用者可不可以使用ssh模式（可以限制帳號、ip或網域名稱等等

```
AllowUsers <user1> <user2> <user3>
AllowGroups <group>
DenyUsers *
DenyGroups no-sshDeny
Users username@192.168.1.*
```

/etc/hosts.allow 及 /etc/hosts.deny 限制 IP 連線

hosts.allow跟hosts.allow是藉由tcp_wrapper運作，可以安裝tcpd

使用指令檢查是否可以與sshd連動運作

```
ldd `which sshd` | grep libwarp
```

出現以下回應代表可以

```
libwrap.so.0 => /lib/x86_64-linux-gnu/libwrap.so.0 (0x00007fedac9e6000)
```

格式:<應用程式>:<ip>

EX:

```
sshd : 192.168.1.100
vsftpd : 192.168.1.*
vsftpd : ALL

sshd : 192.168.1.1,192.168.8.*
```

備註:

1. hosts.allow 優先順序大於hosts.deny。
2. any可以指任何ip, local指同網域, ip或網域前面增加EXCEPT則為除外。

shell 運算

```
n=1
m=2
echo $n+$m
結果:
1+2    #因為shell預設不做運算
```

```
n=1
m=2
echo $((n+m))
結果
3
```

環境變數

現在狀況, 重開機或重登入後部一定會一樣

```
echo $PATH    #顯示當下指令路徑
env           #顯示當前使用者變數
set           #顯示當前shell變數
export        #當前使用者shell變數
a=asd         #設定當前shell變數
```

export a=asd #設定當前使用者shell變數
unset a #取消變數
readonly a #設定只讀環境變數

永久環境變數更改

新增至/etc/profile #全部使用者
新增至$HOME/.profile #個別使用者
新增至/etc/bach.bachrc #更改全部使用bash shell
新增至$HOME/.bashrc #更改個別使用bash shell

更改完畢後需使用**source** 或 **.** 使起生效, 如**source /etc/profile** 或 **./etc/profile**

指令結果

2>/dev/null (忽略錯誤結果)

掃毒

安裝

```
apt install clamav clamdscan clamav-daemon clamav-freshclam
clamscan -r --quiet -l (掃描log檔案路徑) --exclude-dir (不掃描資料夾路徑) (要掃描的地方)
clamdscan - < (要掃描的檔案路徑名單)
cat 要掃描的檔案路徑名單)| clamdscan -
```

clamdtop 類似**htop**模式
補充:
回傳值**0** (沒有病毒) **1** (有病毒) **2** (有錯誤)
clamscan 與 **clamdscan**差異
clamscan較慢 (單核運作)
clamdscan較快且常駐 (多核運作、可用**clamdtop**看運作狀況)

設定防毒常駐自動掃描更動或新檔案

1. 新增參數到/etc/clamav/clamd.conf

```
OnAccessMaxFileSize 5M #即時掃描檔案大小
OnAccessExcludeRootUID yes #排除 Root (UID 0) 存取檔案時觸發掃描, 防止掃描到系統核心文件並避免潛在的循環
OnAccessExcludeUname clamav #止服務掃描自己產生的檔案導致無限循環
OnAccessIncludePath /home #要監控的資料夾
OnAccessIncludePath /tmp
```

備註: 建議監控/tmp /home等常存取的資料夾。

使用**rkhunter**對**rootkit**進行掃描

安裝

```
sudo apt install rkhunter
```

掃描

```
sudo rkhunter --check
```

確認沒問題而進行此次結果紀錄

```
sudo rkhunter --propupd
```

對暫存檔或大量記憶體使用與隱藏檔進行白名單設定

對共享記憶體使用進行忽略

```
ALLOWDEVFILE={檔案路徑}
```

對隱藏檔進行忽略

```
ALLOWHIDDENDIR={路徑}
```

檔案變更允許

```
ALLOW_FILE_ATTRIBUTES_CHANGE={檔案路徑}
```

背景執行

指令完後加"&"

多指令

1|2 (將第一值令的結果輸入到第二值令)
1||2 （1成功就不執行2）
1&&2 （1成功就執行2）
1;2 （不管有無成功持續執行）

掛載遠端資料

```
sudo mount -t cifs -o username=<win_share_user>,password=<win_share_password>
//WIN_SHARE_IP/<share_name> /mnt/win_share
```

使用fstab掛載

```
〈來源路徑〉 〈掛載地方〉 cifs defaults,uid=〈本地使用者權限〉,gid=〈本地群組權限〉,username=〈帳號〉,password=〈密碼〉,iocharset=utf8 0 0
```

EX:1.fstab無法掛載遠端資料夾時可以修改smb版本看看，加入vers={1.0:2.0:3.0}
2.如果mount路徑有空格需要更換成\040

掛載印像檔.img出現錯誤可以新增 -o loop參數

```
sudo mount -0 loop {img來源檔} {掛載路徑}
```

檢查載點是否掛載 (**mountpoint**)

```
mountpoint <路徑>
```

參數:
-q:安靜模式

寄信mail

```
mail -s "This is the subject" somebody@example.com <<< 'This is the message' 〈直接輸入訊息〉
mail -s "Hello World" user@yourmaildomain.com < /home/user/mailcontent.txt 〈導入文件訊息〉
echo "This is the message body" | mail -s "This is the subject" mail@example.com
mail -s "Hello World" user1@example.com,user2@example.com 〈多收件者〉
```

-B 〈BCC收件者〉
-C 〈CC收件者〉
-aFrom:Harry\<harry@gmail.com\> 〈寄件者顯示資訊〉

界面安裝程式tasksel

安裝(輸入法)

```
apt install fcitx5 fcitx5-chewing
```

(中文注音輸入法)

完成後使用im-config做輸入法預設更改

備註: 日文輸入法為fcitx5-mozc。

使用tasksel安裝xfce4□□□出□ ubuntu xfce fail to start session時處理方式

可以安裝gdm3

```
sudo apt install gdm3
```

設定開機啟用圖形化介面

```
sudo systemctl set-default graphical
```

or

```
sudo systemctl set-default graphical.target
```

更改時區

```
dpkg-reconfigure tzdata
```

iptables

```
sudo ufw allow proto tcp from 192.168.56.0/24 to 192.168.56.203 port 22
```

關閉ICMP回應

開啟以下檔案並將所有input的ICMP更改成DROP即可。

```
vi /etc/ufw/before.rules
```

```
# ok icmp codes for INPUT
-A ufw-before-input -p icmp --icmp-type destination-unreachable -j DROP
-A ufw-before-input -p icmp --icmp-type time-exceeded -j DROP
-A ufw-before-input -p icmp --icmp-type parameter-problem -j DROP
-A ufw-before-input -p icmp --icmp-type echo-request -j DROP
```

port位運作檢查

```
lsof -i -P -n | grep LISTEN
netstat -tulpn | grep LISTEN
sudo nmap -sT -O localhost
```

檢查資料夾大小

```
du -h --max-depth=1
```

備註: **--max-depth=** 為下降階層

df -h看資料夾資訊時卡住，可以使用trace df查詢在哪裡卡住

```
trace df
```

umount -f強制關閉該資料夾

```
umount -f {資料夾}
killall -9 df
```

觀看電腦硬體配備

```
lshw
dmidecode -t 17 (等於dmidecode --type memory、dmidecode --type 17)
```

記憶體使用狀況

```
free
free -m
cat /proc/meminfo
```

桌面版也可以直接安裝hardinfo直接檢視

```
sudo apt install hardinfo
```

netplan手動配置網路

```
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:  # 網卡名稱
```

```
    addresses: [10.0.2.15/24]  # 靜態IP、遮罩
    gateway4: 10.0.2.1  # IPV4 Getway ip
    nameservers:
      addresses: [8.8.8.8,8.8.4.4] # DNS server ip, 若有多個就以逗號分隔
    dhcp4: no # 關閉 dhcp 自動取得 IP
  # 如果有 Host only 的話通常都是 enp0s8
  enp0s8:
    addresses: [192.168.56.101/24]  # 靜態IP、遮罩
    routes:
    - to: 192.168.56.1/24
      via: 192.168.56.1
      metric: 100
    #gateway4: 192.168.56.1  # IPV4 Getway ip
    #nameservers:
    #addresses: [8.8.8.8,8.8.4.4] # DNS server ip, 若有多個就以逗號分隔
    dhcp4: no # 關閉 dhcp 自動取得 IP
    dhcp6: no # 關閉 dhcp 自動取得 IP
version: 2
```

備註：sudo netplan try 進行連線 sudo netplan apply 遮蔽設定虛擬用

使用network manager的設定方式

```
network:
  version: 2
  ethernets:
    NM-f52160ba-1cb2-4d49-955e-84a6f51adb8d: #和檔案名稱一樣
      renderer: NetworkManager
      match:
        name: "enp0s8"
        macaddress: "FF:FF:FF:E3:AD:CA" #MAC
      addresses:
      - "192.168.100.51/24"
      ipv6-address-generation: "stable-privacy"
      wakeonlan: true
      networkmanager:
        uuid: "f52160ba-1cb2-4d49-955e-84a6f51adb8d" #只設定一個可以不用改
        name: "Ethernet connection 2" #連線名稱
        passthrough:
          ipv6.method: "ignore"
```

```
        proxy._: ""
```

有線網路呈現device not managed

方法1

```
sudo vi /etc/NetworkMnager/NetworkManager.conf
```

```
[main]
plugins=ifupdown,keyfile,ofono
dns=dnsmasq

[ifupdown]
managed=false    <==這一行改成true
```

```
sudo systemctl restart NetworkManager
```

```
nmcli d ---檢視運行狀況
sudo nmcli dev set (device name) managed yes ----設定device可以運作
```

上法若無效

```
sudo mv /usr/lib/NetworkManager/conf.d/10-globally-managed-devices.conf  /usr/lib/NetworkManager/conf.d/10-globally-managed-devices.conf_orig
sudo touch /usr/lib/NetworkManager/conf.d/10-globally-managed-devices.conf
sudo systemctl restart NetworkManager
nmcli d ---檢視運行狀況
```

開啟wifi radio

```
nmcli r wifi on
```

找出現在可以使用的wifi清單

```
nmcli d wifi list
```

連線到wifi

```
nmcli d wifi connect {SSID} password {password}
```

連線到影藏的wifi

```
nmcli c add type wifi con-name <name> ifname wlan0 ssid <ssid>
nmcli c modify <name> wifi-sec.key-mgmt wpa-psk wifi-sec.psk <password>

#establish wifi ssid
nmcli c up <name>
```

檢視網頁狀態

```
curl -I http://yourdomain/test.jpg
```

檔案下載到指定位置(wget)

```
wget -O 目的地 來源
wget -c 續傳
wget -i 所有路徑檔案（下載檔案裡面所有路徑的資料）
wget -b log紀錄地方 要下載的地方（背景下載）
wget --limit-rate=限制速度（下載限速）
wget --http-user=my_user --http-password=my_password http://www.example.com/my_file (http帳號密碼)
wget --ftp-user=my_user --ftp-password=my_password ftp://ftp.example.com/my_file(ftp帳號密碼)
wget --tries=50 (重試50次)
wget --user-agent="偽裝瀏覽器名稱" 下載路徑
wget --spider 查詢路徑是否正常
wget --mirror -p --convert-links -P 檔案 下載網站（下載整個網站）
wget -r http://www.example.com/ (遞迴下載符合類型之檔案)
wget --secure-protocol=TLSv1_2 {url site} --no-check-certificate 限制使用TLS1.2開啟
```

備註:可加用 -nd(檔案不分路徑)-np (不搜上層路徑) -A(只下載符合關鍵字)

EX:`wget -r -nd -np -A pdf,png http://www.example.com/path1/path2/`

Curl指令的使用

資料下載

```
curl -o {檔案名稱} {網頁路徑} #單一檔案下載
curl -O -C {網頁路徑}          #斷點續傳
curl -O --limit-rate {限制速度} {網頁路徑} #可以使用--max-filesize限制單一檔案大小
```

網頁要求

```
curl -L {網址}   #跳轉到新網頁
curl -v {網址}   #顯示通訊過程
curl -X {GET|POST|DELETE|PUT等等} {網頁路徑}
curl -H 'kev:value' {網頁路徑}  #添加請求header, 可多重使用
EX:curl -H 'Accept-Language: en-US' -H 'Secret-Message: xyzzy' http://www.example.com/test

curl -X POST -d '參數' {網頁路徑}  |請求參數
EX:curl -d '{"user":"zhangsan", "password":"123456"}' -H 'Content-Type:application/json'
http://www.example.com/login

curl -F 'name1=文件1;name2=文件2' {網頁路徑} #檔案上傳
```

????? `-F` ?????? HTTP ????? `Content-Type: multipart/form-data` ?????MIME??? `application/octet-stream`

> **?????? MIME ??**???????MIME???"image/png"
> `curl -F 'file=@photo.png;type=image/png' https://google.com/profile`
> **???????**????????????"photo.png"????????????? "me.png"
> `curl -F 'file=@photo.png;filename=me.png' https://google.com/profile`

```
curl -A 'user-agent' {網頁網址}     #設定Agent
curl -b '參數' {網頁網址}    #設定cookie
curl -u 'user[:password]' {網頁網址}   #設定使用者和帳號
```

限制使用加密方式連線

```
curl {url} --verbose  --tlsv1.2 --tls-max 1.2
```

代理伺服器 (Proxy)

若要讓 wget 透過代理伺服器抓取資料，可以在 ~/.wgetrc 這個設定檔中加入代理伺服器的設定:

```
use_proxy=yes
http_proxy=http://proxy.yoyodyne.com:18023/
```

```
wget -e use_proxy=yes -e http_proxy=http://proxy.yoyodyne.com:18023/ http://www.example.com/
```

```
use_proxy=on
http_proxy=http://username:password@proxy.server.address:port/
https_proxy=http://username:password@proxy.server.address:port/
ftp_proxy=http://username:password@proxy.server.address:port/
```

Flush BOTH DNS caches

You can do it alltoghether, just copy and paste on a terminal:

```
sudo systemd-resolve --flush-caches
sudo nscd -i hosts
```

使用letsencrypt

letsencrypt憑證申請

```
certbot certonly --webroot --webroot-path=/var/www/html -d www.domain.com
```

rdp遠端linux

client安裝xrdp

```
sudo apt install xrdp
```

firewall開啟3389

如出現黑畫面，在/etc/xrdp/startwm.sh的測試和執行Xsession的行之前增加以下

```
unset DBUS_SESSION_BUS_ADDRESS
unset XDG_RUNTIME_DIR
```

當遇到以下狀況時，代表root沒有得到相關權限

02:52:37.658 Main Warn could not connect to display :11.0
02:52:37.658 Main Info Could not load the Qt platform plugin "xcb" in "" even though it was

found.

解法:

```
sudo cp ~/.Xauthority /root
```

使用rdp無法輸入中文

解法: 在Remmina->Preferences->RDP->Use client keyboard mapping把前面的勾去掉即可

xrdp後無法開啟firefox

解法:

不一定會發生，目前知道的原因為新版的firefox使用snap安裝可能有bug所致。

1. 移除現有的firefox連結設定新的apt 函式庫(
2. 到firefox官網下載檔案直接執行。
3. 改使用其他瀏覽器。

disk檢測GUI檢測程式

```
sudo apt install gnome-disk-utility
```

硬碟查指令

```
badblocks -wsv (硬碟位置)
```

補充: **-w**破壞性寫入測試**-n**非破壞性測試**-s**進度**-v**測試結果**-p**測試次數**-o**輸出至檔案

```
mkfs.fat -l (測試結果檔案) (硬碟位置) --避開壞軌建立新磁區
```

補充: 可用**-c**在建立新磁區前測試有無壞軌並避過

fsck.fat (硬碟位置) 修復檔案系統

fsck (硬碟位置) 可以直接看硬碟有無問題

- 0: 沒有錯誤。
- 1: 檔案系統錯誤已修正。
- 2: 系統需要重新開機。
- 4: 檔案系統錯誤未修正。
- 8: 操作錯誤。
- 16: 使用方式錯誤。
- 32: 使用者取消執行。
- 128: 共享函式庫 (shared library) 錯誤。

補充: **-A**檢查所有硬碟**-R**避過以掛載硬碟**-t**檢查的格式**-f**強制檢查**-a**自動修復

增加虛擬機硬碟空間（**resize**）

1. 在虛擬主機設定增加空間
2. 虛擬機設定
    1. **ext4**

```
growpart /dev/sda 2
#2為partition 2
resize2fs /dev/sda2
```

備註: 使用growpart 需要先安裝cloud-guest-utils , rezize 需要先安裝e2fack

檔案資料夾找尋

```
find (搜尋資料夾) -name 檔案名稱 (使用-iname忽略大小寫)
```

```
find ./ -name \*.c -exec grep -wnHA5 main {} \;
```

補充: -type d:目錄 f:檔案 l:連結
-perm 指定檔案權限 ex:find . -type f -perm 0666 （可以在-perm 前增加!成為排除條件）
-exec 可以指定搜尋出的結果接續做指令 ex:find . -type f -name "檔案名稱" -exec rm -f {} \; find / -type d -perm 777 -prin
{} \;
-empty 找尋空白檔案或資料夾
-user（-group）找出檔案擁有者(群組)
-nouser（-group）沒有擁有者(群組)的檔案，後面若加使用者名稱則是找非該使用者（群組）的檔案
-path 找出特殊目錄路徑的檔案
-maxdepth 最多找幾層
-mindepth 從第幾層開始
-llnks 檔案有幾個連結
-newer file 找尋修改時間比file還要新的檔案
-used 文件最後一次存取是在最後一次修改幾天之後
-ok 詢問找出來的檔案要不要執行接下來的動作
-printf format 找到的檔案使用要顯示的方式
\a 警告鈴聲
\b 回退
\c 立即停止以當前格式輸出，刷新輸出設備。
\f 表格結束
\n 新行
\r 換行
\t 水平tab
\v 豎直tab
\\ 輸出自身`\'
\NNN ASCII編碼是NNN(八進制)的字符

□□□`\'□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

%%　□□□□`%'

%a　□□□□□□□□□□□□□□C□□`ctime'□□□□□□

%Ak　□□□□□□□□□□□□□□k□□□□□□`@'□□□C□□`strftime'□□□□□□□□□□k□□□□□□□□□□□□□□□□□□□□□□□□□□`strftime'□□

@ □Jan. 1, 1970, 00:00 GMT□□□□

□□□□□

H □□(00..23)

I □□(01..12)

k □□(0..23)

l □□(1..12)

M □□(00..59)

p □□□AM□□PM

r 12□□□□□□□(hh:mm:ss [AP]M)

S □(00..61)

T 24□□□□□□□(hh:mm:ss)

X □□□□□□□□□(H:M:S)

Z □□(□□□EDT)□□□□□□□□□□□□

□□□□□

a □□□□□□□□□□□□□□(Sun..Sat)

A □□□□□□□□□□□□□□□□(Sunday..Saturday)

b □□□□□□□□□□(Jan..Dec)

B □□□□□□□□□□□□(January..December)

c □□□□□□□□□□(Sat Nov 04 12:02:33 EST 1989)

d □□□□□□□□(01..31)

D □□(mm/dd/yy)

h □b□□

j □□□□□□□(001..366)

m □□(01..12)

U □□□□□□□□□□□□□□□□□□(00..53)

w □□□□□□□□(0..6)

W □□□□□□□□□□□□□□□□□□(00..53)

x □□□□□□□(mm/dd/yy)

y □□□□□□□(00..99)

Y □□(1970...)

%b　□□□□□□512□□□□□□□(□□□□)□

%c　□□□□□□□□□□□□□□□□□C□□`ctime'□□□□□□

%Ck　□□□□□□□□□□□□□□□□□k□□□□□□%A□

%d　□□□□□□□□□□□□0□□□□□□□□□□□□□□

%f　□□□□□□□□□□□□(□□□□□□□□)□

%F　□□□□□□□□□□□□□□□□□-fstype□□□

%g　□□□□□□□□□□□□□□□□□□□□□ID□

%G　□□□□□□□□□□ID□

%h　□□□□□□□□□□□□(□□□□□□□□)□

%H　□□□□□□□□□□□□□□□

%i　□□□i□□□(16□□)□

%k　□□□□□□1kB□□□□□(□□□□)□

%l 符號鏈接的目標(如果文件不是一個符號鏈接，那麼結果是空字符串)。
%m 文件的權限位(8進制)。
%n 文件的硬連接數。
%p 文件名。
%P 文件名，去掉了據以找到了文件的命令行參數的名稱部分。
%s 文件大小，以字節為單位。
%t 文件最後一次修改的時間。格式是C函數`ctime'返回值的格式。
%Tk 文件最後一次修改的時間。格式以k指定，類似於%A。
%u 文件的用戶名，如果用戶沒有名稱就是數字形式的用戶ID。
%U 文件的數字形式的用戶ID。
在一個`%'字符後面使用任何其他字符，`%'將被忽略(但是其他字符會被打印出來)。

-mtime(-mmin) 最後修改天數（分鐘） ex:find /home -type f -mtime 7 (另有最後存取時間 atime、amin、檔案資訊建立時間ctime、cmin mtime +7 -mtime -14 (搜尋7天以上14天以下修改的檔案)

-size 找尋檔案大小 ex:find /home -type f -size 50M(可以用+當以上-當以下) ex:find /home -type f -size +50M -200M

find "(路徑)" -name "*.php" -exec grep -H "(要找的字)" {} \; (可以找尋檔案中的文字)

find . -type f -mtime -3 | grep -v "(路徑)" | grep -v "(路徑)" (找尋3天內修改檔案)

! expr 如果結果false會返回ture, 同-not

-a(-and) 及

expr1 -o(-or) expr2 1為真就不執行2

找尋內容批配狀況

```
grep (想要的內容) (檔案名稱)
```

ls (路徑)|grep 找尋的文字 顯示名子有該文字的的檔案

補充: -i 忽視大小寫

-n 顯示行數

-V 顯示沒有搜尋文字的行數

-r 遞迴搜尋

-e PATTERN, –regexp=PATTERN

-E, –extended-regex

--incloude 找尋檔案類型

-A(B;C) 多顯示前（後;前後）行數 ex:grep -A 2 tt test.txt (顯示結果及前2行)

可以使用正規表示法

```
df -h | grep -vE '^Filesystem|tmpfs' | awk '{ print $1 " " $5 }'  (排除以Filesystem或tmpfs起頭的行，並顯示第1,5列)
```

```
grep -n $query $file | awk -F':' '{print $1}'  (找尋文字所在的行數)
```

‘PATTERN’ or "PATTERN"

grep -e 'root' /etc/passwd

root:x:0:0:root:/root:/bin/bash
(PATTERN1 | PATTERN 2)
grep -e '(root|www)' /etc/passwd
[Char] , [^Char]
grep -e b[^e]n /etc/group

[:alnum:] # 文字或是數位字元,表示為 [A-Z,a-z,0-9]
[:alpha:] # 文字字元,表示為 [A-Z,a-z]
[:lower:] # 小寫字元,表示為 [A-Z]
[:upper:] # 大寫字元,表示為 [a-z]
[:digit:] # 數位字元,表示為 [0-9]
[:xdigit:] # 十六進位數字 0-9,A(10),B(11),C(12),D(13),E(14),F(15),表示為 [0-9, a-f, A-F]
[:space:] # 所有空白字元 ,含 空白鍵, Tab, CR (新行)
[:graph:] # 除了空字元 (空白鍵與 Tab 按鍵) 外的其他鍵.
[:print:] # 非空字元 ,包括空格.
[:cntrl:] # 控制字元.
[:punct:] # 標點符號.
^ , $
grep -e "^ben" /etc/passwd root@ubuntu:~# grep -e "bash$" /etc/passwd

\{\}

x\{m\} 在字串裡面字元 x 重覆 m次 .’0\{5\}’ 表示整個字串裡面 0 這個字元重覆了 5 次.
x\{m,\} 在字串裡面字元 x 重覆 m次 .’0\{5\}’ 表示整個字串裡面 0 這個字元重覆了至少 5 次(以上的也算).
x\{m,n\} 在字串裡面字元 x 重覆 m次 .’0\{1,5\}’ 表示整個字串裡面 0 這個字元重覆了 1(最少)到 5(最多)次.
[a-z]\{m,n\} 在字串裡面字元 a-z 重覆 m次 .’a-z\{1,5\}’ 表示整個字串裡面 0 這個字元重覆了 1(最少)到 5(最多)次.

下面範例用來搜尋出 指令 #ifconfig 裡面提供的 IP , Bcast 及 Mask 相關資訊.

```
ifconfig ens33 | grep -e "[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}"
inet addr:172.16.15.130  Bcast:172.16.15.255  Mask:255.255.255.0
```

更新遇到the package lists or status file could not be parsed or opened的錯誤訊息

```
sudo rm /var/lib/apt/lists/* -vf
sudo apt-get clean
sudo apt-get update
```

桌面錄影軟體kazam安裝

```
sudo apt install kazam
sudo apt install openshot (影片製作軟體)
```

cron 出現(CRON) info (No MTA installed, discarding output)的訊息

主要是因為cron執行指令時出現錯誤而cron是使用郵件做通知，故沒有安裝MTA（EX:postfix）就會出現此提示

可以使用（執行指令）> /dev/null 2>&1 忽略錯誤訊息或（執行指令）> （結果放置檔案）2>&1來做紀錄

cron設定方法

格式: 分 時 日 月 禮拜幾 執行角色 指令 EX:0 * * * * root cmd.sh （在整點的時候使用root權限執行cmd.sh）

備註: "*"代表任何時刻都要執行

","分開2個或多個不一樣的執行時間, ex: 0,30 or 0,15,30,45

"-"代表範圍內時間都要執行ex: 0-15 or 22-30

"*/n" 每隔n個時間執行一次

在星期幾的資料中, 0代表禮拜日,6為禮拜六

@reboot 重開機執行一次

@yearly 一年執行一次, 同@annually

@monthly 一個月執行一次

@weekly 一個禮拜執行一次

@daily 一天執行一次, 同@midnight

@hourly 每小時執行一次

使用last 來顯示登入紀錄

last ---顯示所有登入紀錄

last (name) ---顯示特定人員的登入紀錄

補充:

-n （次數） ---顯示最近的幾次

-F 顯示登出與登入時間

-ad ---顯示主機名稱

-s 何時開始使間 格式: XXXX-XX-XX XX:XX:XX or -Xdays

-t 何時結束時間

mutt 郵件設定

```
apt install mutt
```

vim ~/.muttrc 輸入以下資訊

```
set from = "XXXX@yourdomain.com"
set realname = "XXX"
set smtp_url = "smtps://XXXX@yourdomain.com@mail.yourdomain.com:465/"
set smtp_pass = "PASSWORD"
set header_cache = "~/.mutt/cache/headers"
set message_cachedir = "~/.mutt/cache/bodies"
```

```
set certificate_file = "~/.mutt/certificates"
```

ssmtp 郵件測試

```
#
# Config file for sSMTP sendmail
#
# The person who gets all mail for userids < 1000
# Make this empty to disable rewriting.
root=SENDMAIL

# The place where the mail goes. The actual machine name is required no
# MX records are consulted. Commonly mailhosts are named mail.domain.com
mailhub=mail Domain

# Where will the mail seem to come from?
#rewriteDomain=

# The full hostname
hostname=HOSTNAME

# Are users allowed to set their own From: address?
# YES - Allow the user to specify their own From: address
# NO - Use the system generated From: address
FromLineOverride=YES

AuthUser=USER ACCOUNT
AuthPass=PASSWORD

#UseSTARTTLS=YES
UseTLS=YES
```

指令

```
echo "bodys"| mutt -s "title" (收件人郵件) -b (收件人郵件) -c (收件人郵件) -a （附件路徑）
mutt -s "title" (收件人郵件) -b (收件人郵件) -c (收件人郵件) < 內容路徑.txt -a （附件路徑）
```

ssh進入系統通知

需要先安裝mutt、mail或ssmtps等軟體

全部使用者都要通知，請將下面訊息增加至/etc/bash.bashrc, 若是個別使用者請增加至~/.bashrc

```
echo 'ALERT - Root Shell Access (PVE) on:' `date` `who` | mail -s "Alert: Root Access from `who | cut -d'(' -f2 | cut -d')' -f1`" (收件者)
```

資料對校

md5sum 、sha256sum
md5sum (檔案名稱)
md5sum -c (之前產生的對校報告)

如何顯示特殊符號

有些符號Bash上有特殊意義，比如$'"`等等，若要顯示這些符號可以使用增加反斜線\

date的用法

```
date --date="now"          (現在時間，等於直接輸入date)
date --date="yesterday"    (昨天，同date --date="1 days ago")
date --date="3 days ago"   (三天前)
date --date="tomorrow"     (明天，同date --date="1 days")
date --date="3 days"       (三天後)
date +"%Y %_m-%_d, %H:%_M" --date="3 days"  (為要顯示的"輸出格式"+"時間")
```

補充:
如有搭配指令輸出應使用$(date +"%Y_%m_%d" --date="now")的格式
"_" ： 0用空白填塞
"-" ：0不填塞
"^"：將輸出的小寫改為大寫

```
%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)
```

%I 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)

date -r 檔案名稱 #可以縣市檔案修改時間

ex: date -r test.txt +%F date -r test.txt +'%Y-%m-%d %H:%M:%S'

read 用法

read -s -n1 -p "按任意鍵繼續 ... "

read –p "提示" 變數資料 #使用p來做輸入提示

read –t N 變數名稱 #-t是限制輸入時間

複製的使用方式

cp (來源檔) (目的路徑或檔案名稱)

補充:

-r 遞迴複製

-p 保留模式、所有權跟使間

-a 保留權限

rsync (來源檔) (目的路徑或檔案名稱)

補充:

-a 保留權限

-v 詳細輸出

-h 人類讀取模式

--bwlimit 限速1024代表1MB

echo 加 tee 輸入多文件

```
echo "Hello World!" | sudo tee -a abc.txt backup.txt
```

備註: 同時也會顯示結果

printf "The path of shell interpreter is %s\n" $SHELL >> abc.txt

sed 用法

型態: sed [-nefri] ‘command’ 輸入文件

-n 安靜模式

-e 直接進入編輯模式

-f 直接將sed動作寫到文件

-r 支援正規型延伸表示法

-i 直接修改讀取的資料但不顯示

command意思

a 新增到下一行sed '5a 新增的文字' 文件（新增到第6行後）

c 取代這一行 sed '2c 取代的文字' 文件（取代第2行）

d 刪除這一行 sed '3d' 文件（刪除第三行）

i 插入到上一行

p 將某行資料顯示出來

s 取代動作 sed 's/搜尋的字/新的字/g' 文件

補充:

$可以用以代表最後一行

連續多行 3,5d 刪除第3-5行

sed -i ‘s/abc/\ /g’ filename.txt 用反斜線做空白或特殊符號替代

sed 's/^.//' 文件（刪除第一個字）

指令逾期刪除

```
timeout 10 tail -f /var/log/httpd/access.log （指令10秒後刪除）
```

rsync 資料備份

型態: rsync (參數) （來源路徑/檔案） （目的路徑/檔案）

參數:

-v 輸出比較詳細的資訊

-r 遞迴備份子資料夾

-a 備份保留原資料的所有資訊，同參數-rlptgoD

-z 啟動壓縮

-h 以較人性化的方式做顯示

-i 可以知道個別檔案的變動形式

-l 複製符號連結

-p 複製檔案權限

-t 複製時間屬性

-g 複製群組屬性

-o 複製擁有者屬性

-D 複製裝置屬性

-q -quiet忽略非錯誤信息

-c -checksum基於checksum校驗，而非mod-time和size

-R -relative使用相對路徑

-u -update跳過接受端較新的文件

-d -dirs僅傳輸文件而非遞歸

-l -links將軟鏈接當作軟鏈接來拷貝

-L -copy-links拷貝鏈接對應的文件或者目錄而非鏈接本身

-copy-unsafe-links 僅不安全的鏈接才被拷貝

-safe-links 忽略那些鏈接到目錄樹外的鏈接

-k -copy-dirlinks將鏈接翻譯成其鏈接的目錄

-K -keep-dirlinks將目錄鏈接在接收端轉換成目錄

-H -hard-links保留硬鏈接

-A -acls保留ACL（暗示-perms）

-t -times保留修改時間

--bwlimit 頻寬限制

--delete 同步刪除資料

--exclude 排除資料

--include 要備份的資料

--min-size 要備份檔案的最小容量

--max-size 要備份檔案的最大容量

--remove-source-files 刪除來源資料

--existing 更新既有檔案

--whole-file 強制同步整個資料夾

--progress 顯示傳輸速度跟剩餘時間

-del -delete-during的別名

-delete-before 接收端在發送前刪除，而不是發送過程中

-delete-during 接收端在發送過程中刪除

-delete-delay 在發送過程中尋找文件，在發送完成後刪除

-delete-after 接收端在發送完成後刪除

-delete-excluded 在接收端刪除排除的文件

-ignore-errors 即使有I / O錯誤也刪除

-force 即使目錄不是空的也刪除

-max-delete = NUM最多刪除文件的數目

-partial 保持部分傳輸的文件

-partial-dir = DIR制定部分傳輸文件的存放目錄

-delay-updates 先傳輸，最後再更新，保持原子性

-m -prune-empty-dirs接收端刪除空目錄
-numeric-ids 接收端不要將uid / gid映射為用戶名和組名
-timeout = SECONDS設置I / O超時時間，s為單位
-contimeout = SECONDS設置鏈接服務端的時間
-I -ignore-times即使mtime和size都相同也不跳過
-size-only 只要大小相同就跳過
-modify-window = NUM比對時間時制定精確範圍，範圍內都認為時間相同
-T -temp-dir = DIR指定創建臨時文件的目錄
-y -fuzzy文件在接收端不存在的情況下，在當前目錄下尋找一個基礎文件，以加快傳輸
-compare-dest = DIR接收端除了和發送端對比還和這裡指定的目錄對比，適合備份上次備份改變的文件
-copy-dest = DIR和-compare-dest類似，只是接收端會用本地拷貝來複製那些未改變的文件
-link-dest = DIR和-compare-dest類似，只是接收端會建立那些未改變文件的硬鏈接
-compress-level = NUM指定壓縮等級
-skip-compress = LIST不壓縮指定後綴的文件
-C -cvs-exclude以CSV的方式自動忽略文件
-f -filter = RULE新增一個文件過濾規則
-F與-filter ='dir-merge /.rsync-filter'重複相同: -filter =' - .rsync-filter'
-exclude = PATTERN排除規則PATTERN
-exclude-from = FILE從文件中讀取排除規則
-include = PATTERN不要排除指定規則的文件
-include-from = FILE從文件中讀取包含的規則
-files-from = FILE從文件中讀取文件列表
-0, -from0 all
-from / filter文件由0分隔
-s -protect-args參數不許要空格分割; 只有通配符特殊字符
-address = ADDRESS綁定監聽的地址
-port = PORT制定端口號
-sockopts = OPTIONS制定TCP選項
-blocking-io在遠程shell中使用阻塞I / O.
-stats給出文件統計信息
-8, -8-bit-output輸出時不對高位字符轉義
-h, -human-readable以易於閱讀的方式打印數字
-progress顯示傳輸進度
-P與-partial-progress相同
-i -itemize-changes打印更新的總結信息
-out-format = FORMAT以特定的格式打印更新信息
-log-file = FILE日誌文件
-log-file-format = FMT日誌文件格式
-password-file = FILE密碼文件
-list-only僅列出文件
-write-batch = FILE將批量更新寫入文件
-only-write-batch = FILE和-write-batch類似但沒有更新目的地
-read-batch = FILE從文件中讀取批量更新任務
-protocol = NUM使用舊版本的協議
-iconv = CONVERT_SPEC要求文件名字符轉義
-4, -ipv4更喜歡IPv4

-6, -ipv6更喜歡IPv6
-version打印幫助信息
(-h) -help打印這個幫組信息 (-h僅在單獨使用時與-help同意)
補充:
如果是要做遠端資料的**rsync**, 可下 **rsync -avh /home/ron user@ipaddr:**路徑
如果使用的不是預設的埠位, 則必須要加入指定的埠位, 如**rsync -avzh -e "ssh -p** 埠位**"** (來源路徑/檔案) (目的路徑/檔案)
若是使用**rsync server**做資料傳遞, 冒號**":"**將會再多一個, **EX:rsync -avh /home/ron user@ipaddr::**路徑

SCP相關設定

```
scp -r 本機路徑 username@hostname:遠端路徑
```

ssh 相關設定

ssh-keygen 產生憑證
補充:
-t 形式 EX:rsa
-b 加密位元 預設3072

取得公鑰方式
方法1:

```
ssh-copy-id -i key_path username@ipaddr
```

方法2:
將公鑰複製到server的~/.ssh/authorized_keys 文件裡

完成後需要調整/etc/ssh/sshd_config
PasswordAuthentication no

sshd_config設定
port 22 #設定ssh使用埠位
listenaddress 0.0.0.0 #設定監聽ip
hostkey /etc/ssh/ssh_host_key #設定私鑰路徑
authorizedkeysfile #公鑰路徑
serverkeybits 1024 #定義伺服氣密要的位數
logingracetime 600 #登入等待時間
keyregenerationinterval 3600 #多久時間後在重新產生密鑰
compression no #封包壓縮, 延遲到身份認證之後輸入delay
MaxAuthTrues 3 #登入測試幾次
PermitRootLogin no #禁止root使用ssh登入
Printlastlogin yes #顯示最後一次登入訊息
clientaliveInterval 900 #設定靜置多久就關閉連線

clinetalivecountmax 0 #需跟上述設定一起使用
allowusers username #允許登入的使用者
denyusers username #禁止登入的使用者
allowgroups username #允許登入的群組
denygroups username #禁止登入的群組
permitemptypassword no #不允許無密碼登入
ignorerhosts yes #忽略Rhost文件
ignoreuserknownhosts yes #忽略已知連線電腦(known_hosts)
hostbasedauthentication no #禁止使用主機認證
x11forwarding no #禁止gui使用ssh通道
strictmodes yes #檢查用戶配件文件, 不安全無法登入
pubkeyauthentication yes #使用憑證登入
GSSAPIauthentication no #關閉GSSAPI認證
kerberosauthentication no #禁止kerberos登入
Ciphers aes128-ctr,aes192-ctr,aes256-ctr #避免使用弱密碼
MACs hmac-sha2-256,hmac-sha2-512 #避免使用弱密碼
Banner 路徑 #登入警示語氣
passwordauthentication yes #禁止密碼登入
補充: **authorized_keys權限設定600 ~/.ssh 設定700**

ssh 預前輸入密碼使用

安裝sshpass
使用方式
sshpass -p 'password' ssh username@ipaddr 'cmd'
使用環境參數設定
export SSHPASS='password' #也可以直接新增到/etc/profile 或 $HOME/.profile可以永久使用
ssh -e ssh username@ipaddr 'cmd'

批次更改檔案名稱

```
rename 's/(要更改的部份)/(更改成的方式)/' (符合的檔案)
```

EX:rename 's/DSC_/4inlibra-Ron-雲家檸檬大王-/' *.JPG

nfs使用說明

nfs server的使用方式
安裝:

```
sudo apt install nfs-kernel-server
```

檢視是否運作:

```
sudo netstat -tulnp | grep rpcbind   #會看到111udp跟tcp監聽
udo netstat -tuln | grep :2049       #看nfs是否有使用2049的tcp或udp運作
```

```
vi /etc/exports                      #修改可以連線的client跟權限
```

模式: 要分享的目錄路徑 允許存取的NFS客戶端(一些選項)

ex:/home/magiclen/shared-files 192.168.56.104(rw,sync,no_subtree_check) 192.168.100.0/24(ro,sync,no_subtree_check) *.4inlibra.com(ro,sync,no_subtree_check)

權限設定:

- rw: 允許讀寫。
- ro: 唯讀。這個是預設值。
- root_squash: NFS的客戶端如果是root使用者(UID為0)或是root群組(GID=0)，那麼NFS伺服器端會將其UID或是GID視為65534，代表「匿名者」或
- no_root_squash: NFS的客戶端如果是root使用者或是root群組，就會直接在NFS伺服器上使用這個身份。
- all_squash: FS的客戶端如果不是root使用者或是root群組，NFS伺服器端會將其UID或是GID視為65534，代表「匿名者」或是「匿名群組」。
- no_all_squash: NFS的客戶端如果不是root使用者或是root群組，就會直接在NFS伺服器上使用這個身份。這是預設值。
- no_subtree_check: 不進行子樹檢查(subtree_check)。當分享的目錄路徑不是整個檔案系統，而是其中的一個目錄的話，啟用「subtree_check」就可以在存
- nohide:「exported tree」中，如果某分享的目錄(例如「/exports/home」)是在另一個分享的目錄(例如/exports)之下，則這兩個目錄都需要被掛載，該分享的目錄(/exports/home)才可以被存
- crossmnt: 這個選項可以讓該目錄在被掛載時，使其在「exported tree」中的下層目錄也自動加上nohide選項。換句話說，只要掛載上層目錄，就可以存取到下層目錄，下層目錄不
- fsid: 這個選項可以設定每個NFS的ID，大部分的情況下不需要自己設定ID。0或是root表示分享目錄的根目錄的ID，也就是說可以藉由設定fsid=0或是fsid=root，來讓某個目錄

載入設定:

```
sudo exportfs -r
```

說明: -a 或 -r 都可以重新載入，-ua停止分享

查看分享:

```
sudo exportfs
```

查看現況:

```
showmount
```

參數:

- --exports: 查看目錄的分享狀態。
- --directories: 顯示哪些目錄正在被掛載。
- --all: 顯示哪些目錄正在被哪些客戶端名稱掛載。

nfs client的使用方式

安裝:

```
sudo apt install nfs-common
```

掛載:

```
sudo mount -t nfs 192.168.56.103:/home/ron/shared-files /mnt/nfs
```

```
192.168.72.136:/mnt/sharedfolder /mnt/client_sharedfolder nfs defaults 0 0 #設定到fstab
```

- rw: 允許讀寫。
- ro: 唯讀。
- rsize: 設定每次從NFS客戶端傳給NFS伺服器的資料量，最常用的值為4096。愈可靠的網路可以將這個值設高一點，增加傳輸效率。
- wsize: 設定每次從NFS伺服器傳給NFS客戶端的資料量，最常用的值為4096。愈可靠的網路可以將這個值設高一點，增加傳輸效率。
- udp: 對於支援UDP和TCP的NFS第3版，可以指定要使用UDP來傳送檔案。
- tcp: 對於支援UDP和TCP的NFS第3版，可以指定要使用TCP來傳送檔案。
- nfsvers: 設定NFS伺服器的版本。
- retry: 設定要允許多長的時間來重新嘗試掛載，單位是分鐘。預設值是10000分鐘。
- timeo: 設定每次嘗試掛載時，經過多久就算是逾時，單位是0.1秒。預設值是0.7秒。
- hard: 不管RPC逾時，一直嘗試掛載。為預設值。
- soft: 嘗試掛載，直到RPC逾時。
- fg: 在前景嘗試掛載。為預設值。
- bg: 在背景嘗試掛載。
- intr: 嘗試掛載時可以被使用者中斷。
- nointr: 嘗試掛載時不可以被使用者中斷。為預設值。

NFS伺服器的主機名稱:分享的目錄路徑 掛載點 nfs _netdev 0 0

如果要讓linux可以編輯ntfs檔案格式

```
apt install ntfs-3g
```

設定使用swap

swap -s #顯示現有設定，或使用 free -h

使用分割區做swap

```
sudo mkswap /dev/sda #新增實會增加一個UUID，也可以使用-U 來自行新增
sudo swapon /dev/sda #啟用新增的swap，若SSD有支援TRIM請使用-d --discard，可讓空間交換更有效率
```

備註：更改**fstab**的**swap**路徑，也可以增加**discard**在**defaults**旁邊

使用檔案做swap

```
sudo fallocate -l (建立的空間) /檔案名稱
```

也可使用 sudo dd if=/dev/zero of=/(檔案名稱) bs=1M count=512 #大小=bs*count

```
sudo chmod 600 /(檔案名稱) #建立權限
sudo mkswap /(檔案名稱) #初始化swap
sudo swapon /(檔案名稱) #啟用swap
```

備註: 更改fstab的swap路徑

```
swapoff /(路徑) 關閉swap
```

更改swap轉換優先權

```
更新鎖定 (不要更新)
```

```
vi /etc/sysctl.conf
```

```
vm.swappiness = 10 # 0 (完全禁用) ~100 (盡量轉換), 60為預設值
```

更新鎖定 (不要更新)

```
dpkg --get-selections | grep mongodb  #搜尋套件
sudo apt-mark hold mongodb mongodb-dev #鎖住套件
sudo apt-mark auto mongodb-dev  #自動模更新模式
sudo apt-mark unhold mongodb-dev #解鎖模式
sudo apt-mark manual mongodb-dev #更改成手動模式
sudo apt-mark showhold     #檢視停用自動更新清單
sudo apt-mark showmanual   #檢視手動移除清單
sudo apt-mark showauto | more #檢視自動更新清單
```

安裝deb出現相依錯誤時, 可以使用以下方式安裝相依套件

```
sudo apt install --fix-missing
```

完成後就可以安裝

Ubuntu Kernel更換使用

http://kernel.ubuntu.com/~kernel-ppa/mainline/ #kernel放置區域
#其中 -lowlatency 是低延遲版本的 kernel, 除了 lowlatency 之外還有 -generic、-rt 、-preempt 等, 沒特殊要求選擇 -generic 就行。

```
grep "menuentry 'Ubuntu" /boot/grub/grub.cfg  #檢視現行可用kernel選項
sudo nano /etc/default/grup  #進入修改GRUB_DEFAULT=0 至要的選項
sudo update-grub #更新設定
```

umount遇到device is busy時該怎麼處理

```
fuser -cuk (目標路徑) #刪除在該資料夾運行的行程
```

備註:

-c
Same as -m option, used for POSIX compatibility.

-u, --user
Append the user name of the process owner to each PID.

-k, --kill
Kill processes accessing the file. Unless changed with -SIGNAL, SIGKILL is sent. An fuser process never kills itself, but may kill other fuser processes. The effective user ID of the process executing fuser is set to its real user ID before attempting to kill.kill.

diff比較用法

```
diff -bur 來源1 來源2
```

備註: b為忽略空白的差異
u以 unified 格式輸出檔案內容的差異
r為遞迴式搜尋
q只列出差異性的檔案

case ... esac

```
#!/bin/bash
m='MacDonald'case $m in
    MacDonald*) echo "Ronald McDonald"
            ;;
    KFC*) echo "Harland David Sanders"
            ;;
    TKK*)     echo "阿勇"
            ;;
    *)      echo "You are really health!"
esac
```

for

```
#!/bin/bash
for loop in 1 2 3; do
    echo "number: $loop"
```

```
done
```

```
#!/bin/bash
echo -n "請問你要幾個檔案"
read F
for ((i=1 ; i<=F ; i++))dotouch $i.js
echo $idone
echo "已經給你 $F 個檔案囉了!"
```

while

```
#!/bin/bash
echo -n "請問你要幾個檔案"
read FINDEX=1
# 當條件成立，就會不斷執行 (le表示小於或等於)
while [ $INDEX -le $F ]
do
  # 輸出妹
    touch $INDEX.js
    echo -n "$INDEX"
  # INDEX 的值會加1
    (( INDEX++ ))
done
echo ""echo "已經給你 $F 個檔案囉了!"
```

```
#!/bin/bash
echo "按下 Ctrl + C 中斷..."LENGTH=0
while :
do
    echo -ne "\r["
    sleep 0.2
    while [ $LENGTH -le 10 ]
    do
        sleep 0.1
        echo -n ">"
        (( LENGTH++ ))
    done
    LENGTH=0
    echo -en "\r                    "
done
```

一次讀取一行

```
#!/bin/bash
echo -n "請輸入要讀取的文字檔名稱: "
read FNINDEX=1
while read line
do
        echo "イルカポリス: $line"
        (( INDEX++ ))
done <$FN
```

```
#!/bin/bash
echo -n "請問你要幾個檔案"
read F
counter=0
until [ $counter = $F ]; do
    ((counter++)) #放在後面會變無限迴圈
    echo $counter
    touch $counter.js

done
echo "已經給你 $F 個檔案囉了!
```

dd指令的使用

```
sudo dd if=/dev/sda of=/dev/sdb
```

備註: 可以使用conv設定參數, sync使用I/O sync方式, noerror會忽略錯誤。
可以使用*.img (*.iso) 將資料匯出或複製成印象檔。

e2fsck修復指令的使用

```
e2fsck -a -y /dev/sda1    #檢查或修復硬碟
```

參數:

-a: 檢查 partition, 如發現問題會自動修復。
-b: 設定 superblock 位置。
-B size: 指定 size 作為區塊大小。
-c: 檢查 partition 是否有壞軌。

-C file: 將檢查結果儲存到 file。
-d: 輸出 e2fsck debug 結果。
-f: e2fsck 預設只會對錯誤的檔案系統檢查，加上 -f 是強制檢查。
-F: 在檢查前將硬碟的 buffer cache 清空，避免發生錯誤。
-l list: 記錄了壞軌區塊加入 list 中。
-d：列印 e2fsck 的 debug 结果。
-f：强制检查。
-n: 以 (read-only) 開啟檔案系統
-p: 關閉互動模式，如有問題自動修復，等同 -a。
-v: 顯示詳細報告。
-y: 啟用使用者互動模式。

圖檔檢視軟體

```
sudo apt install gthumb
```

Linux安裝SOP

安裝ubuntu or debian service版

一、update

二、install ufw(啟用ufw) & clamav clamdscan clamav-freshclam clamav-daemon qemu-guest-agent(PVE使用)

補充說明：記得設定排程掃毒跟即時掃毒等功能，還有rkhunter

三、ssh使用安裝

四、關閉自動更新還有不需要的服務

```
sudo vi /etc/apt/apt.conf.d/20auto-upgrades
```

APT::Periodic::Update-Package-Lists "1"; #更改改0
APT::Periodic::Unattended-Upgrade "1"; #更改改0

五、關閉cloud-init

建立一個空白檔案在/etc/cloud/裡面

```
sudo touch /etc/cloud/cloud-init.disabled
sudo reboot
```

六、關閉ipv6

方式1

```
sudo vim /etc/sysctl.conf
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

方式2

```
sudo vi /etc/default/grub
GRUB_CMDLINE_LINUX_DEFAULT="ipv6.disable=1"
sudo update-grub
```

完成後重開機

安裝桌面版本 (tasksel)

安裝libreoffice,remmina

mdadm指令

請注意blkid確認格式ex:blkid /dev/sda

```
mdadm --detail --scan >> /etc/mdadm.conf
```

掃描新Raid並紀錄至mdadm.conf

```
mdadm /dev/md10 --fail
```

(要讓他失效的硬碟) ---使硬碟失效

```
mdadm --manage /dev/md1 --remove
```

(要移除的硬碟)---移除不好的硬碟(要先離線或使其失效)

```
mdadm --manage /dev/md1 --add
```

(要加入的硬碟) ---加入新的硬碟

開機後不見的Raid找回

```
mdadm --stop /dev/md0
```

---先停止Raid

```
mdadm --assemble --force /dev/md0
```

(原本在Raid裡面的硬碟)

備註: 不可以對原陣列硬碟做改變

```
update-initramfs -u
```

Raid重新配置後都需要做的動作

```
mdadm --create --verbose /dev/md0 --level=(raid類型) --raid-devices=(硬碟數) --spare-devices=(硬碟數) (加入的硬碟)
```

新增陣列

```
mdadm --grow --size=max /dev/md1
```

大容硬碟更換完畢後須做的指令

```
resize2fs /dev/md0 -p
```

用 resize2fs 來擴增檔案系統容量(xfs改用xfs_growfs)

```
cat /proc/mdstat
```

查看目前raid進度

```
mdadm --examine （要查詢的硬碟）
```

要查詢Raid各個硬碟的狀況

Raid變換類型步驟（操作前重要資料請備份）

1.視狀況增加硬碟

```
mdadm --manage /dev/md1 --add (要加入的硬碟)
```

2.更改Raid模式(此以Raid1->Raid5)

```
mdadm --grow /dev/md1 --level=5 --raid-devices=3
```

3.完成後容量更新

```
mdadm --grow --size=max /dev/md1
```

(可能不用作)

```
resize2fs /dev/md0 -p
```

用 resize2fs 來擴增檔案系統容量(xfs改用xfs_growfs)

```
update-initramfs -u
```

Raid重新配置後都需要做的動作

monit設定

修改monitrc

修改路徑在/etc/monit/monitrc

檢查時間間隔

```
set daemon  60
```

log放置位置

```
set log /data/monit/logs
```

web監控路徑

```
set httpd port 2812 and
    use address 202.107.204.55
# only accept connection from localhost (drop if you use M/Monit)
    allow 0.0.0.0/0.0.0.0
# allow localhost to connect to the server and
    allow admin:monit
# require user 'admin' with password 'monit'
```

設定如果沒有mount成功

```
# mount.conf
check filesystem data with path /dev/sdb
      if does not exist then exec "/bin/mount /dev/sdb /data"
```

[`python server.py` 的进程，那就启动。

```
# recommender.conf
check process recommender with MATCHING "python server.py"
    if does not exist then exec "/usr/bin/nohup python /data/Recommender/src_tornado/server/server.py &"
    if changed pid then alert
```

monit常用指令

```
monit -t # 配置文件检测
monit # 启动monit daemon
monit -c /var/monit/monitrc # 启动monit daemon时指定配置文件
monit reload # 当更新了配置文件需要重载
monit status # 查看所有服务状态
monit status nginx # 查看nginx服务状态
monit stop all # 停止所有服务
monit stop nginx # 停止nginx服务
monit start all # 启动所有服务
monit start nginx # 启动nginx服务
monit -V # 查看版本
```

郵件通知配置

```
set mailserver  smtp.monit.ro port 587
set mail-format { from: monit@monit.ro subject: $SERVICE $EVENT at $DATE on $HOST
 message: Monit $ACTION $SERVICE $EVENT at $DATE on $HOST : $DESCRIPTION.
       Yours sincerely,
          Monit
  }

set alert guletz@monit.ro
```

```
set mailserver smtp.gmail.com port 587
 username "username" password "password"
 using tls
```

```
set mail-format {
 from: Monit Alert <monit@$HOST>
 subject: 通知 專題伺服器 -- $EVENT $SERVICE
 message: $SERVICE => $EVENT
日期: $DATE
動作: $ACTION
主機: $HOST
說明: $DESCRIPTION
```

檢視系統

```
check system localhost
    if loadavg (1min) > 10 then alert
```

```
    if loadavg (5min) > 6 then alert
    if memory usage > 75% then alert
    if cpu usage (user) > 70% then alert
    if cpu usage (system) > 60% then alert
    if cpu usage (wait) > 75% then alert
```

```
check system $HOST
 if loadavg (1min) per core > 2 for 5 cycles then alert
 if loadavg (5min) per core > 1.5 for 10 cycles then alert
 if cpu usage > 75% for 5 cycles then alert
 if memory usage > 80% for 1 cycles then alert
```

單一檔案hash值

```
check file passwd with path /etc/passwd
 if changed checksum then alert
```

系統效能監控

```
check file bashrc with path /etc/bash.bashrc
 if changed checksum then alert
```

目錄修改時間監控

```
check filesystem root_directory PATH /
 if write rate > 50 MB/s for 10 cycles then alert
 if read rate > 50 MB/s for 10 cycles then alert
```

網路介面卡上下行速度監控

```
check network enp3s0 with interface enp3s0
 if download > 20 MB/s for 10 cycles then alert
 if upload > 20 MB/s for 10 cycles then alert
```

vim簡單語法

----檢視模式----

u 還原上一步

y 複製

d 剪下

p 貼上

:e {檔案}　#編輯別的檔案

gg　　#移動到第一行

:g　　#移動到最後一行

:new or :vnew #建立新編文件

:w {檔案}　　#另存新檔

:r {檔案}　　#將該檔案內容加進來

:n1,n2 w {檔案}　#將二行之間的文字另存新檔

:! {指令}　　#暫時執行指令並看結果

ctrl+w　　#各編輯視窗跳動

:set nu　　#顯示行數

/{字元}　　#向下搜尋 可使用N繼續找下一個

?(字元)　　#向上搜尋 可使用N繼續找上一個

:n1,n2s/word1/word2/g　#指定第幾行到第幾行進行更換

:%s/舊/新/g　#替換

:%s/舊/新/gc　#全文替換, 但會確認

SMART使用指令

S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology)相關介紹

smart指令的使用

安裝

```
apt install smartmontools -y
```

檢視基本資料

```
smartctl -i {硬碟}
```

開啟SMART功能

```
smartctl -s on (硬碟)
```

簡單檢測（只回覆pass/failed）

```
smartctl -H (硬碟)
```

檢查硬碟

```
smartctl -t (檢查方式) (硬碟)
```

備註:

檢查方式:

offline # 1. No entry is placed in the selftest log.
2. The effects of this test are visible only in that it updates the SMART offline Attribute values
short # 立即聽到 harddisk 在做野 (check the electrical and mechanical performance)
long # 相當於 Offline Extended self-test
conveyance(輸送) # intended to identify damage incurred during transporting of the device
select,M-N # to test a range of disk LBAs
pending,N
afterselect,on afterselect,off

加d sat可以檢查usb硬碟

最常使用short

檢驗進度查詢

```
smartctl -c (硬碟)
```

終止測試

```
smartctl -X
```

檢查測試結果

```
smartctl -l error /dev/sdd
```

備註:

-l type # Prints either the SMART Error Log (TYPE: error, selftest, selective, directory, ssd)
-l error => offline test 要用此看
prints the Summary SMART error log
SMART disks maintain a log of the most recent five non-trivial errors
the disk power-on lifetime at which the error occurred is recorded
-l selftest => "short", "long" test 要用此看
The time at which the test took place, measured in hours of disk lifetime
-l ssd => prints the Solid State Media percentage used endurance indicator
(0 indicates as new condition while 100 indicates the device is at the end of its lifetime)
If any errors were detected, the Logical Block Address (LBA) of the first error is printed in decimal notation.

顯示所有硬碟資料

```
smartctl -A (硬碟)
```

知識補充:

哪幾個S.M.A.R.T.數據項目是最關乎HDD/SDD健康度的?

S.M.A.R.T.有好多項目, 其中最能反映磁盤健康狀態的指標有下列幾個:

- **Reallocated Sector Count** 重定[illegible]的指標, 因為它顯示了已經被發現為有缺陷並且被重新分配的扇區數量。這個數值的高值或快速增加可能指示硬碟驅動器正在失效。
- **Current Pending Sector Count** 等候[illegible]等待重新映射的不穩定扇區的數量。增長的掛起扇區數量可以是硬碟退化的徵兆。
- **Seek Error Rate** 尋道[illegible]用於HDD, 表示找不到正確軌道的失敗率。高數值可能表明一個機械問題。
- **Read Error Rate** 讀取[illegible]從磁碟讀取資料時發生的硬體錯誤率。高數值可能表明磁碟有問題。
- **Uncorrectable Sector Count** 無法[illegible]寫入一個扇區時的不可糾正錯誤的數量。高數值是硬碟驅動器可能故障的嚴重警告。

- **Spin Retry Count:** [illegible] HDD [illegible] - [illegible] HDD [illegible]

[illegible] S.M.A.R.T. [illegible] Current [illegible] Normalized, Worst [illegible] Threshold [illegible]

S.M.A.R.T. [illegible] S.M.A.R.T. [illegible] “Current” [illegible] “Worst” [illegible] “

“Current” [illegible]

- [illegible]
- [illegible]

“Worst” [illegible]

- [illegible]
- [illegible]

“Normalized” [illegible]

- [illegible] “[illegible]” [illegible] 0-100 [illegible] 0-200 [illegible]
- [illegible]

“Threshold” [illegible]

- [illegible] “[illegible]” [illegible] “[illegible]” [illegible]
- [illegible] “[illegible]” [illegible] “[illegible]” [illegible]

[illegible] S.M.A.R.T. [illegible] S.M.A.R.T. [illegible]

[illegible]

S.M.A.R.T. [illegible]