

LINUX

2주차. 리눅스의 기본 개념

Run & Shutdown

X 윈도우(X window)

- 리눅스의 GUI 제공
- Ubuntu: GNOME / Kbuntu: KDE

종료

- 바탕화면의 ▼ 아이콘으로 종료
- 터미널/콘솔에서 종료

```
root@server: ~# poweroff
root@server: ~# shutdown -P now      # -P 옵션은 시스템 종료를 의미
root@server: ~# halt -P              # -P 옵션은 시스템 종료를 의미
root@server: ~# init 0               # 잠시 후 설명
```

📌 terminal 에서 root 사용자는 #, 일반 사용자는 \$로 구분함. 따라서 이후부터는 #과 \$로 root와 일반 사용자 표시를 대체함

shutdown 옵션

```
~# shutdown -P now           # -P (Poweroff) / now(지금 종료)
~# shutdown -P +10           # 10분 후 종료
~# shutdown -r 22:00          # -r(reboot) / 오후 10시에 재부팅
~# shutdown -c                # c(cancel): 예약된 shutdown 취소
~# shutdown -k +15            # 현재 접속한 사용자에게 15분 후 종료된다는 메시지를 보내지만 실제로는 종료되지 않음
```

📌 명령어의 옵션 목록을 보고 싶으면 '명령어 --help' 또는 '명령어 --h' 를 입력


🙋 Pop Quiz !?

사용자에게 3분 후 종료된다는 메시지를 보낸 후, 4분 후에 시스템을 reboot 하려면?

Quiz 정답 !!

사용자에게 3분 후 종료된다는 메시지를 보낸 후, 4분 후에 시스템을 reboot 하려면?

```
~# shutdown -k +3  
~# shutdown -r +4
```

 shutdown 명령어에 아무런 옵션을 주지 않으면 1분 후 종료

reboot 명령어

```
~# reboot
~# shutdown -r now
~# init 6
```

로그아웃

- 바탕화면의 ▼ 아이콘으로 로그아웃
- 터미널/콘솔에서 로그아웃

```
~$ logout          # ServerB 와 같은 text mode에서만 작동
~$ exit            # Bash Shell (desktop version terminal) 에서 작동
```

Virtual Console

가상 콘솔

- 가상의 모니터 개념
- 우분투는 총 6개 (2번~7번)의 가상 콘솔을 제공
- 7대의 모니터 연결 효과
- Server가 부팅 되자마자 자동으로 실행되는 X 윈도 (root 계정) 은 두번째 가상 콘솔
- 첫번째 콘솔은 처음 install 할 때 설정한 사용자 (user: ubuntu)
- 콘솔 간 전환: **Ctrl + Alt + F1~F7** / F1은 첫번째 사용자 로그인 화면

가상 콘솔 실습 - 1

1. Ctrl+Alt+F3 을 1초간 입력
2. 텍스트 모드의 3번째 가상 콘솔 → root/password 로 접속

```
Ubuntu 20.04.1 LTS server tty3
```

```
server login: root
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.8.0-41-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

0♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ .
0♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ .

Your Hardware Enablement Stack (HWE) is supported until April 2025.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

```
root@server:~#
```

가상 콘솔 실습 - 1

3. Ctrl+Alt+F4 1초간 입력

4. 텍스트 모드의 4번째 가상 콘솔 → ubuntu/ubuntu 로 접속

```
Ubuntu 20.04.1 LTS server tty4
```

```
server login: ubuntu  
Password:  
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.8.0-41-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:       https://ubuntu.com/advantage  
  
0♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ .  
0♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ .  
  
Your Hardware Enablement Stack (HWE) is supported until April 2025.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.
```

```
ubuntu@server:~$ _
```

가상 콘솔 실습 - 1

5. Ctrl+Alt+F3 입력 (root로 전환)

6. `shutdown -h +5` 입력

```
root@server:~# shutdown -h +5  
Shutdown scheduled for Mon 2021-02-08 22:24:27 KST, use 'shutdown -c' to cancel.
```

7. Ctrl+Alt+F4 입력 (ubuntu로 전환)

```
ubuntu@server:~$  
Broadcast message from root@server on tty3 (Mon 2021-02-08 22:19:27 KST):  
  
The system is going down for poweroff at Mon 2021-02-08 22:24:27 KST!  
  
Broadcast message from root@server on tty3 (Mon 2021-02-08 22:20:27 KST):  
  
The system is going down for poweroff at Mon 2021-02-08 22:24:27 KST!  
  
Broadcast message from root@server on tty3 (Mon 2021-02-08 22:21:27 KST):  
  
The system is going down for poweroff at Mon 2021-02-08 22:24:27 KST!
```

가상 콘솔 실습 - 1

8. Ctrl+Alt+F3 입력 (root로 전환)

9. **shutdown -c** 입력

```
root@server:~# shutdown -h +5  
Shutdown scheduled for Mon 2021-02-08 22:24:27 KST, use 'shutdown -c' to cancel.  
root@server:~# shutdown -c
```

10. Ctrl+Alt+F4 입력 (ubuntu로 전환)

```
Broadcast message from root@server on tty3 (Mon 2021-02-08 22:23:06 KST):  
The system shutdown has been cancelled
```

가상 콘솔 실습 - 2

1. Ctrl+Alt+F3 입력 (root로 전환)
2. **shutdown -k +10** 입력 (🧑 Pop Quiz !? 무슨 뜻?)

```
root@server:~# shutdown -k +10  
Shutdown scheduled for Mon 2021-02-08 22:48:04 KST, use 'shutdown -c' to cancel.
```

3. Ctrl+Alt+F4 입력 (ubuntu로 전환)

```
Broadcast message from root@server on tty3 (Mon 2021-02-08 22:38:04 KST):  
  
The system is going down for poweroff at Mon 2021-02-08 22:48:04 KST!
```

4. Ctrl+Alt+F3 입력 (root로 전환)
5. **shutdown -c** 입력
6. 3번, 4번 가상 콘솔 모두 **logout**
7. Ctrl+Alt+F2 입력 (X 윈도 전환)

Run Level

런 레벨

런레벨	영문 모드	설명	비고
0	Power Off	종료 모드	
1	Rescue	시스템 복구 모드	단일 사용자 모드
2	Multi-User		사용하지 않음
3	Multi-User	텍스트 모드의 다중 사용자 모드	
4	Multi-User		사용하지 않음
5	Graphical	그래픽 모드의 다중 사용자 모드	
6	Reboot		

[표 4-1] 리눅스의 런레벨

- 런 레벨 확인

```
~# cd /lib/systemd/system
```

/lib/systemd/system로 진입

```
~# ls -l runlevel?.target
```

-l: 한 줄에 파일 하나

runlevel 뒤에 알 수 없는 문자가 있는 이름의 확장자 target인 파일을 보여주는 명령어

런 레벨

```
root@server:/lib/systemd/system# ls -l runlevel?.target
lrwxrwxrwx 1 root root 15 2월 5 02:23 runlevel0.target -> poweroff.target
lrwxrwxrwx 1 root root 13 2월 5 02:23 runlevel1.target -> rescue.target
lrwxrwxrwx 1 root root 17 2월 5 02:23 runlevel2.target -> multi-user.target
lrwxrwxrwx 1 root root 17 2월 5 02:23 runlevel3.target -> multi-user.target
lrwxrwxrwx 1 root root 17 2월 5 02:23 runlevel4.target -> multi-user.target
lrwxrwxrwx 1 root root 16 2월 5 02:23 runlevel5.target -> graphical.target
lrwxrwxrwx 1 root root 13 2월 5 02:23 runlevel6.target -> reboot.target
```

- 실제 파일에 연결된 링크 파일(Linked File) → 뒤에서 배움
- Server 를 부팅하면 X윈도 화면이 나옴 == 런 레벨 **init 5** 로 지정
- ServerB 를 부팅하면 텍스트 모드 화면이 나옴 == 런 레벨 **init 3** 로 지정

런 레벨 실습

1. `ls -l /lib/systemd/system/default.target` 입력

```
root@server:~# ls -l /lib/systemd/system/default.target
lrwxrwxrwx 1 root root 16  2월  5 02:23 /lib/systemd/system/default.target -> graphical.target
```

- 2.

```
~# ln -sf /lib/systemd/system/multi-user.target /lib/systemd/system/default.target
```

ln은 링크 파일을 만드는 명령어, 뒤에서 배움

3. `ls -l /lib/systemd/system/default.target` 입력

```
root@server:~# ln -sf /lib/systemd/system/multi-user.target /lib/systemd/system/default.target
root@server:~# ls -l /lib/systemd/system/default.target
lrwxrwxrwx 1 root root 37  2월  9 00:06 /lib/systemd/system/default.target -> /lib/systemd/system/multi-user.target
```

4. reboot

런 레벨 실습

5. 텍스트 모드에서 **startx** 입력 → X 윈도 화면
6. 런 레벨 3 위에서 런 레벨 5를 가동시키는 것
- 7.

```
~# ln -sf /lib/systemd/system/graphical.target /lib/systemd/system/default.target
```

8. reboot

Auto Complete & History



자동 완성

- 파일 이름의 일부만 입력하고 Tab을 누르면 자동 완성

도스 키

- 이전에 입력한 명령어를  /  를 이용하여 다시 나타나게 하는 것

도스 키 실습

1.  /  를 이용하여 과거의 명령어들을 탐색
- 2.

```
~# history
```

```
~# history -c          # -c: clear
```

자동 완성 실습 - 1

1. 비슷한 이름이 없을 때

```
~# cd ~                # ~: home directory
~# ls
~# cd 다 Tab           # '다' 입력 후 Tab 누름
```

2. 비슷한 이름이 있을 때

```
~# cd /etc
/etc# cd sys Tab       # 'sys' 입력 후 Tab 누름
```

```
/etc# cd sys Tab Tab  # 'sys' 입력 후 Tab 2번 누름
```

```
root@server:/etc# cd sys
sysctl.conf  sysctl.d/  systemd/
```

자동 완성 실습 - 1

3.

```
/etc# cd syst Tab          # 'syst' 입력 후 Tab 누름  
/etc# cd systemd/ne Tab    # 'ne' 입력 후 Tab 누름
```

```
root@server:~# cd /etc/  
root@server:/etc# cd systemd/network
```

Editor

에디터

- 메모장

에디터 실습 - gedit

1.

```
~#gedit
```

2. 아무 내용을 입력 후 저장 버튼 클릭 후, root의 home directory 에 test.txt 로 저장

3.

```
~#gedit /root/test.txt
```

에디터 실습 - nano

1.

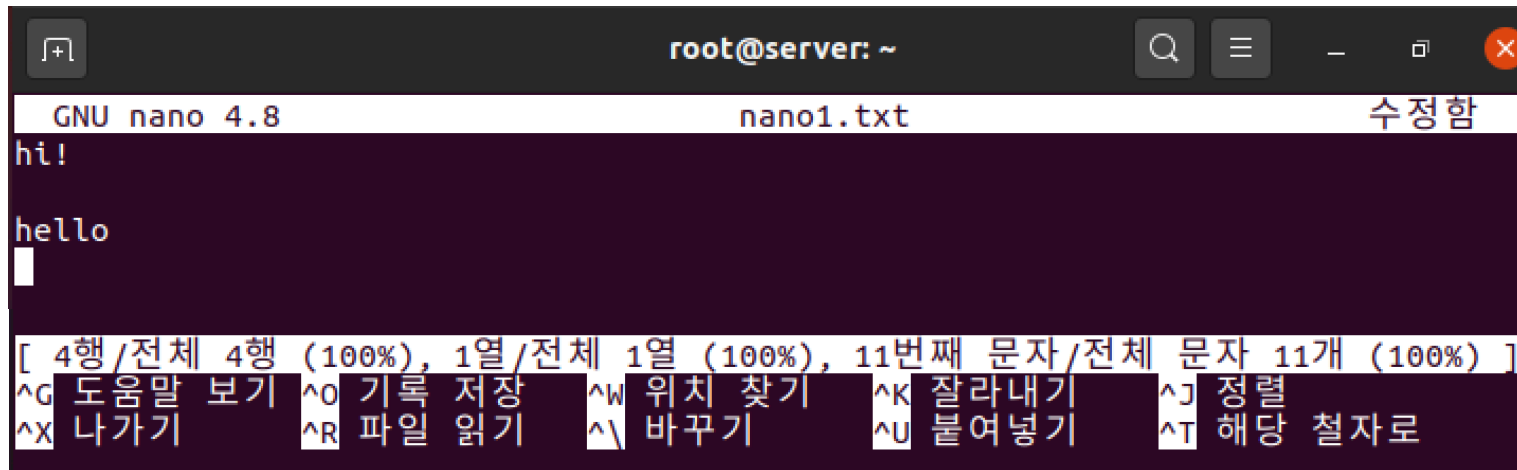
```
~#nano
```

2. 아무 내용을 입력 후 **Ctrl+X** → **Y** 누른 후, **nano1.txt** 로 저장

3.

```
~#nano nano1.txt
```

4. 내용 추가 후 **Ctrl+C** & **Ctrl+X** → **Y** → **Enter**



The screenshot shows the GNU nano 4.8 text editor interface. The title bar indicates the user is root@server: ~. The editor is editing a file named nano1.txt, which is marked as '수정함' (modified). The content of the file is 'hi!' followed by a blank line and then 'hello'. The status bar at the bottom shows the current position: [4행 / 전체 4행 (100%), 1열 / 전체 1열 (100%), 11번째 문자 / 전체 문자 11개 (100%)]. Below the status bar, there are two rows of keyboard shortcuts: ^G for help, ^O for write out, ^W for where to go, ^K for kill line, ^J for join lines, ^X for exit, ^R for read in, ^\ for replace, ^U for undo, and ^T for toggle case.

```
GNU nano 4.8 nano1.txt 수정함
hi!
hello
[ 4행 / 전체 4행 (100%), 1열 / 전체 1열 (100%), 11번째 문자 / 전체 문자 11개 (100%) ]
^G 도움말 보기 ^O 기록 저장 ^W 위치 찾기 ^K 잘라내기 ^J 정렬
^X 나가기 ^R 파일 읽기 ^\ 바꾸기 ^U 붙여넣기 ^T 해당 철자로
```

에디터 실습 - vi

1.

```
~#vi
```

```
# vi Editor 에서 ECS 누르고
```

```
:q Enter
```

2.

```
~#vi new.txt
```

3. I(Insert) 또는 A(Append) 누르고 아무 내용 입력 후, ESC → :wq(write & quit) → Enter

 Pop Quiz !?

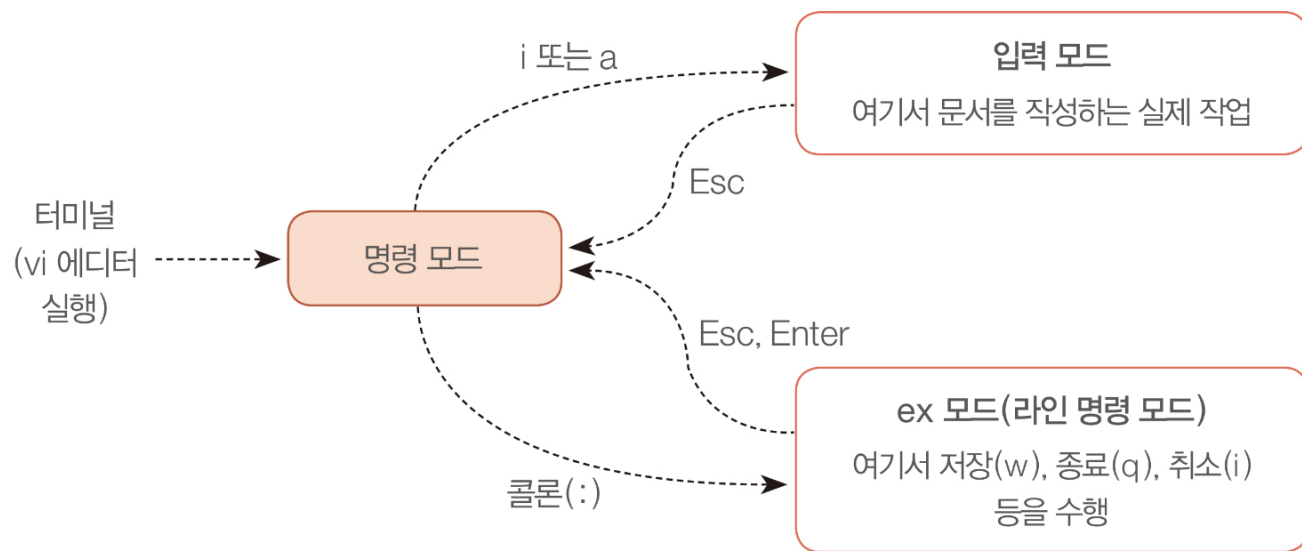
vi 에디터에서 중간 저장하는 방법은?

👤 Quiz 정답 !!

vi 에디터에서 중간 저장하는 방법은?

```
# vi Editor 에서 ECS 누르고
```

```
:w Enter
```



[그림 4-34] vi 에디터 사용법 개요도

에디터 실습 - vi

1.

```
~#vi
```

2. I(Insert) 또는 A(Append) 누르고 아무 내용 입력 후, ESC → :wq test3.txt → Enter

3.

```
~#vi test3.txt
```

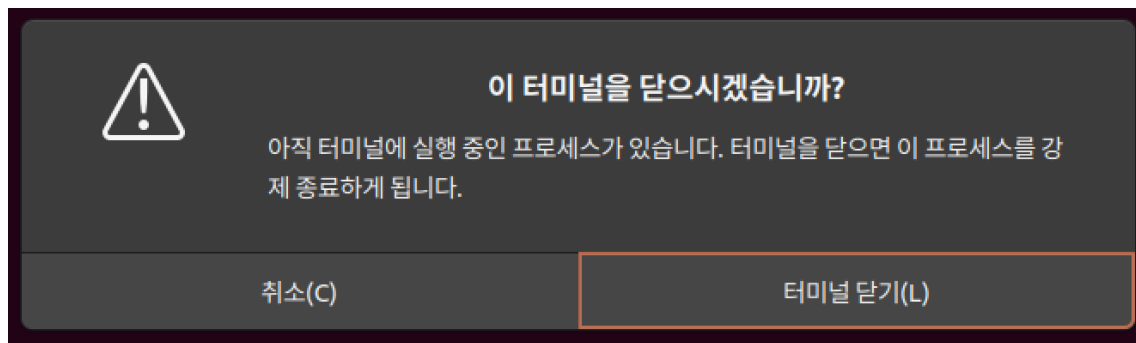
4. I(Insert) 또는 A(Append) 누르고 아무 내용 입력 후, ESC → :q!(!는 강제로 실행) → Enter

에디터 실습 - vi swap

1.

```
~#vi new.txt
```

2. I(Insert) 또는 A(Append) 누르고 아무 내용 입력 후, 터미널 오른쪽 상단의 X 버튼 클릭 → **터미널 닫기** 클릭



3.

```
~#vi new.txt
```

에디터 실습 - vi swap

4.

```
E325: ATTENTION
Found a swap file by the name ".new.txt.swp"
    owned by: root   dated: 화  2월 09 02:02:57 2021
    file name: ~root/new.txt
    modified: YES
    user name: root   host name: server
    process ID: 2017
While opening file "new.txt"
    dated: 화  2월 09 01:28:21 2021

(1) Another program may be editing the same file.  If this is the case,
    be careful not to end up with two different instances of the same
    file when making changes.  Quit, or continue with caution.
(2) An edit session for this file crashed.
    If this is the case, use ":recover" or "vim -r new.txt"
    to recover the changes (see ":help recovery").
    If you did this already, delete the swap file ".new.txt.swp"
    to avoid this message.
"new.txt" 1 line, 3 characters
Press ENTER or type command to continue
```

5. Enter → **:q!** → Enter

에디터 실습 - vi swap

6.

```
~#ls -a          # -a: all / 숨김 파일도 모두 표시
```

```
root@server:~# ls -a
.          .cache      .new.txt.swp  snap        문서
..         .config    .profile     test.txt    바탕화면
.Xauthority .gnupg     .xsession-errors test3.txt   비디오
.bash_history .local    nano1.txt    공개        사진
.bashrc      .mozilla  new.txt      다운로드   음악
템플릿
```

7.

```
~#rm -f .new.txt.swp  # -f: force / 물음 없이 강제로 지움
```

8. 정상적으로 열리는 지 확인

```
~#vi new.txt
```


에디터 실습 - vi 그 외

- 어떠한 개발자도 vi의 모든 명령어와 단축키를 외우고 있지 않음!!
- 꼭 필요한 **i**, **:wq!**, **:q!**, **yy**(행 복사), **dd**(행 삭제), **p**(행 붙여 넣기) 등을 제외한 다른 명령어는 필요에 의해 검색해서 사용
- 책 p.161 ~ 단축키 모음

Manual

도움말

```
~#man ls
```

```
# ls의 상세
```

- **--help** 는 자주 쓰는 옵션 위주로 설명 (--help도 어쨌거나 옵션!)
- **man** 은 명령어의 기능과 모든 옵션에 대해 설명

Mount

마운트

- 물리적인 장치를 특정 위치에 연결 시키는 과정
- 리눅스에서는 하드디스크의 파티셔닝, CD/DVD, USB 메모리 등 **외부 장치를 사용하려면 지정한 위치에 연결**해줘야 함

마운트 실습 - Server

1. 스냅샷을 활용하여 초기 상태로 복구
- 2.

```
~#mount
```

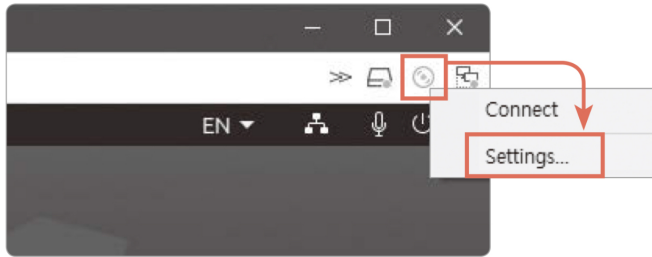
```
root@server:~# mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,noexec,relatime,size=940828k,nr_inodes=23
5207,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmo
de=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=193956k,mode=755
)
/dev/sda5 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,rela
time)
```

- 3.

```
~#umount /dev/cdrom
```

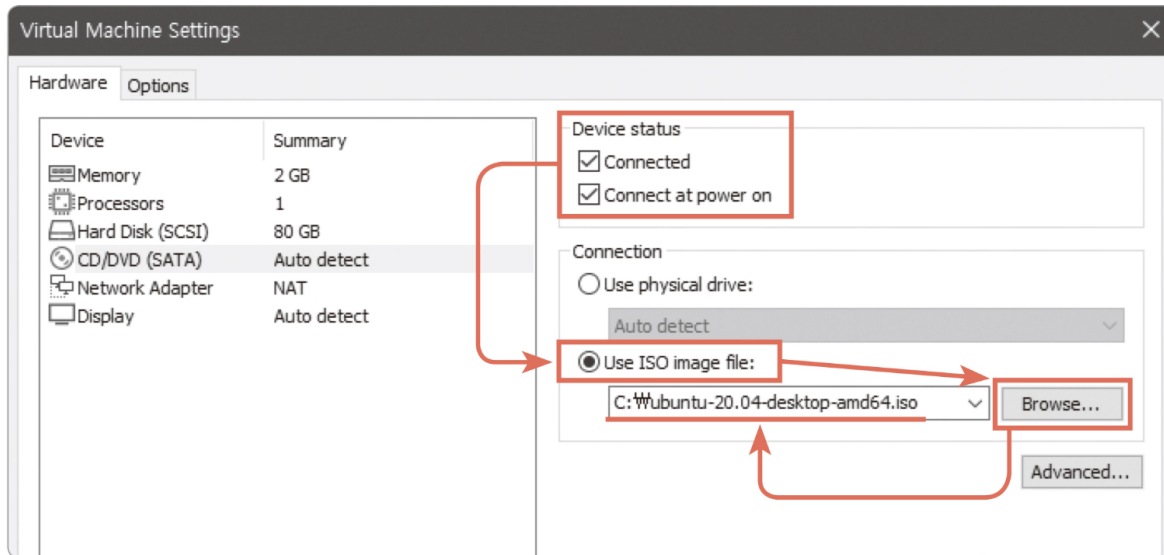
마운트 실습 - Server

4. 아이콘 클릭 / VMware → [Player] → [Removable Devices] → [CD/DVD (SATA)] → [Settings]



[그림 4-42] CD/DVD 마운트 1

5.



[그림 4-43] CD/DVD 마운트 2

마운트 실습 - Server

6.

```
~#mount
```

```
/dev/sr0 on /media/root/Ubuntu 20.04.1 LTS amd64 type iso9660 (ro,nosuid,nodev,
relative,nojoliet,check=s,map=n,blocksize=2048,uid=0,gid=0,dmode=500,fmode=400,
uhelper=udisks2)
root@server:~#
```

7. 자동으로 마운트 된 디렉토리는 /media/ 이고, 마운트 된 CD/DVD의 이름으로 디렉토리 자동 생성됨

8.

```
~#cd /media/root/Ubuntu\ 20.04.1\ LTS\ amd64/
...#pwd                # pwd: 현재 디렉토리의 절대 경로를 보여줌
...#ls
...#cd casper
.../casper#ls -l
```


마운트 실습 - Server

9. **filesystem.squashfs** 은 우분투 전체가 들어있는 파일 / 우분투를 설치하면 압축이 풀리면서 전체 시스템 구성

```
root@server:/media/root/Ubuntu 20.04.1 LTS amd64/casper# ls -l
합계 2154482
-r--r--r-- 1 root root      56693  8월  1  2020 filesystem.manifest
-r--r--r-- 1 root root       2890  8월  1  2020 filesystem.manifest-minimal-rem
ove
-r--r--r-- 1 root root       3839  8월  1  2020 filesystem.manifest-remove
-r--r--r-- 1 root root         11  8월  1  2020 filesystem.size
-r--r--r-- 1 root root 2103640064  8월  1  2020 filesystem.squashfs
-r--r--r-- 1 root root       833  8월  1  2020 filesystem.squashfs.gpg
-r--r--r-- 1 root root   90821026  8월  1  2020 initrd
-r--r--r-- 1 root root  11662080  8월  1  2020 vmlinuz
```

10.

```
.../casper#umount /dev/cdrom
```

```
root@server:/media/root/Ubuntu 20.04.1 LTS amd64/casper# umount /dev/cdrom
umount: /media/root/Ubuntu 20.04.1 LTS amd64: target is busy.
```

위치가 /media/root/의 하위이기 때문에 마운트 해제 실패

마운트 실습 - Server

11.

```
.../casper#cd /media  
/media#umount /dev/cdrom
```

12.

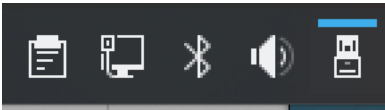
```
/media#mount
```

/dev/sr() 사라짐

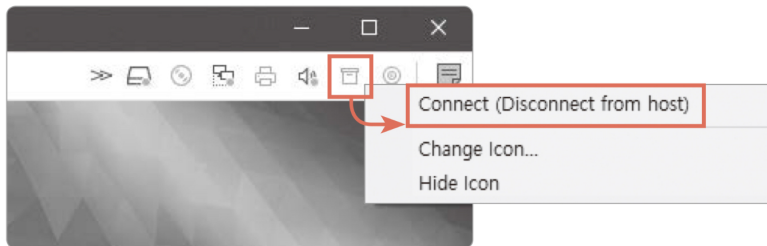
13. [Settings] → CD/DVD 해제

마운트 실습 – Client (Kubuntu)

1. [Settings/Edit] → [USB Controller] → [USB Compatibility] → **USB 3.0**
2. Client 부팅
3. USB 장착
- 4.



5. 아이콘 우클릭 후, **USB Connect**



[그림 4-54] USB 메모리 사용법 3

마운트 실습 – Client (Kubuntu)

6. 파일 관리자로 열기 클릭



7. [Konsole]

```
~$cd /media/ 현재사용자이름/ USB이름
...$pwd
...$ls
...$cp /boot/conf* .           # cp: copy, .: 현재 디렉토리/ boot 디렉토리 내에 있는 conf로 시작하는 모든 파일 현재 디렉토리로 복사
...$ls
```

```
config-5.8.0-25-generic
config-5.8.0-41-generic
```

마운트 실습 – Client (Kubuntu)

8.

```
...$mount
```

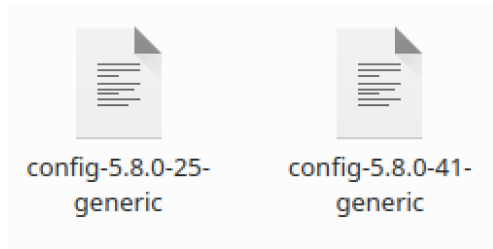
```
/dev/sdb1 on /media/ubuntu/YEH00N_S type vfat (rw,nosuid,nodev,relatime,uid=1000,gid=1000,mask=0022,dmask=0022,codepage=437,ioccharset=iso8859-1,shortname=mixed,showexec=utf8,flush,errors=remount-ro,uhelper=udisks2)
```

9.

```
...$cd /media
```

```
/media$umount /dev/sdb1
```

10. 아이콘 우클릭 후, **USB Disconnect** → 파일 관리자로 열기 클릭



11.

```
/media$halt -p
```

마운트 실습 – SeverB

📌 이번 실습에서 USB 내용이 지워질 수도 있으니 중요한 파일은 백업!

1. [Settings/Edit] → 〈Add〉 → [USB Controller] 선택 → 〈Finish〉
2. ServerB 부팅
3. **root/password** 로 로그인
- 4.

```
~#mount
```

5. Ubuntu.iso 파일은 넣고 USB 연결
- 6.

```
~#mount
```

아무 변화 없음 → **텍스트 모드는 자동 마운트 X**

마운트 실습 – SeverB

7.

```
~#ls /dev/sd*           # dev 디렉토리 내의 sd로 시작하는 파일 모두 출력
```

```
root@server-b:~# ls /dev/sd*  
/dev/sda /dev/sda1 /dev/sda2 /dev/sdb /dev/sdb1
```

8.

```
~#ls -l /media           # media 하위 디렉토리 및 파일 확인  
~#mkdir /media/usb /media/cdrom      # media 하위에 usb, cdrom 디렉토리 생성  
~#ls -l /media  
~#mount /dev/cdrom /media/cdrom      # CD/DVD 마운트  
~#mount /dev/sdb1 /media/usb         # USB 마운트
```

```
root@server-b:~# mount /dev/cdrom /media/cdrom  
mount: /media/cdrom: WARNING: device write-protected, mounted read-only.
```

마운트 실습 – SeverB

9.

```
~#mount
```

```
/dev/sr0 on /media/cdrom type iso9660  
/dev/sdb1 on /media/usb type vfat (rw
```

10.

```
~#ls /media/cdrom  
~#ls /media/usb
```

11.

```
~#cp /media/usb/VM*.확장자 .           # media/usb 에서 VM으로 시작하는 특정 확장자의 파일을 현재 디렉토리에 복사  
~#ls  
~#cp /boot/vm* /media/usb             # 적당한 파일을 /media/usb 에 복사  
~#ls /media/usb
```


마운트 실습 – SeverB

12.

```
~#umount /media/cdrom  
~#umount /media/usb
```

13. [Settings/Edit] → [USB Controller] → Disconnect

14.

```
~#halt -p
```

마운트 실습 – Server ISO Mount

1. 스냅샷을 활용하여 초기 상태로 복구
2. Server 부팅
- 3.

```
~#dpkg --get-selections genisoimage    # dpkg: Debian Package
```

```
root@server:~# dpkg --get-selections genisoimage  
genisoimage                                install
```

- 4.

```
~#genisoimage -r -J -o boot.iso /boot    # boot 디렉토리의 모든 파일을 boot.iso 로 만듦  
                                           # -r -J: 8글자 이상의 파일 이름 및 대소문자를 구분 인식 / -o: 파일 출력
```

```
Total translation table size: 0  
Total rockridge attributes bytes: 26419  
Total directory bytes: 43008  
Path table size(bytes): 52  
Max brk space used 44000  
69011 extents written (134 MB)
```

마운트 실습 – Server ISO Mount

5.

```
~#mkdir /media/iso # 마운트 할 디렉토리 생성
~#mount -o loop boot.iso /media/iso
~#ls -l /media/iso
~#ls -l /boot
```

```
-r--r--r-- 1 root root 4738430 7월 10 2020 System.map-5.4.0-42-generic
-r--r--r-- 1 root root 5515823 1월 19 02:39 System.map-5.8.0-41-generic
-r--r--r-- 1 root root 237773 7월 10 2020 config-5.4.0-42-generic
-r--r--r-- 1 root root 248291 1월 19 02:39 config-5.8.0-41-generic
dr-xr-xr-x 4 root root 2048 2월 5 02:30 grub
lr-xr-xr-x 1 root root 27 2월 5 02:29 initrd.img -> initrd.img-5.8.0-41-generic
-r--r--r-- 1 root root 49525112 2월 5 02:29 initrd.img-5.4.0-42-generic
-r--r--r-- 1 root root 51579133 2월 5 02:30 initrd.img-5.8.0-41-generic
lr-xr-xr-x 1 root root 27 2월 5 02:23 initrd.img.old -> initrd.img-5.4.0-42-generic
-r--r--r-- 1 root root 182704 2월 14 2020 memtest86+.bin
-r--r--r-- 1 root root 184380 2월 14 2020 memtest86+.elf
-r--r--r-- 1 root root 184884 2월 14 2020 memtest86+_multiboot.bin
lr-xr-xr-x 1 root root 24 2월 5 02:29 vmlinuz -> vmlinuz-5.8.0-41-generic
-r--r--r-- 1 root root 11662080 8월 1 2020 vmlinuz-5.4.0-42-generic
-r--r--r-- 1 root root 9716640 1월 19 02:41 vmlinuz-5.8.0-41-generic
lr-xr-xr-x 1 root root 24 2월 5 02:29 vmlinuz.old -> vmlinuz-5.4.0-42-generic
```

```
root@server:~# ls -l /boot
합계 130672
-rw----- 1 root root 4738430 7월 10 2020 System.map-5.4.0-42-generic
-rw----- 1 root root 5515823 1월 19 02:39 System.map-5.8.0-41-generic
-rw-r--r-- 1 root root 237773 7월 10 2020 config-5.4.0-42-generic
-rw-r--r-- 1 root root 248291 1월 19 02:39 config-5.8.0-41-generic
drwxr-xr-x 4 root root 4096 2월 5 02:30 grub
lrwxrwxrwx 1 root root 27 2월 5 02:29 initrd.img -> initrd.img-5.8.0-41-generic
-rw-r--r-- 1 root root 49525112 2월 5 02:29 initrd.img-5.4.0-42-generic
-rw-r--r-- 1 root root 51579133 2월 5 02:30 initrd.img-5.8.0-41-generic
lrwxrwxrwx 1 root root 27 2월 5 02:23 initrd.img.old -> initrd.img-5.4.0-42-generic
-rw-r--r-- 1 root root 182704 2월 14 2020 memtest86+.bin
-rw-r--r-- 1 root root 184380 2월 14 2020 memtest86+.elf
-rw-r--r-- 1 root root 184884 2월 14 2020 memtest86+_multiboot.bin
lrwxrwxrwx 1 root root 24 2월 5 02:29 vmlinuz -> vmlinuz-5.8.0-41-generic
-rw-r--r-- 1 root root 11662080 8월 1 2020 vmlinuz-5.4.0-42-generic
-rw----- 1 root root 9716640 1월 19 02:41 vmlinuz-5.8.0-41-generic
lrwxrwxrwx 1 root root 24 2월 5 02:29 vmlinuz.old -> vmlinuz-5.4.0-42-generic
```

6.

```
~#umount /media/iso
```