

Linux System Administration

리눅스 서버 구축

리눅스 서버 배포판

: Ubuntu

▶ Ubuntu

- ▶ Devian 계열 배포판(캐노니컬, 우분투 재단이 개발, 배포, 유지보수)
- ▶ 타 배포판에 비해 편리하고 사용자 친화적인 인터페이스로 많은 사용층
- ▶ 데스크탑에 중점을 둔 개발
 - ▶ 일반인의 리눅스에 대한 쉬운 접근을 가능하게 함
 - ▶ 개인용 데스크탑과 노트북에서 가장 인기 있는 리눅스 배포판
- ▶ 서버 플랫폼을 지원하는 별도의 서버 배포판이 있음
- ▶ <https://www.ubuntu.com/download/server> 에서 다운로드 가능 (현재 16.04 LTS가 최신)



리눅스 서버 배포판

: CentOS

▶ CentOS

- ▶ RedHat 계열의 리눅스 배포판
- ▶ RedHat Enterprise Linux(RHEL, 상용)와 완벽하게 호환되는 무료 기업용 컴퓨팅 플랫폼을 제공할 목적으로 만들어진 배포판
- ▶ 타 배포판에 비해 안정성 부문에서 가장 신뢰도가 높으며 관련 문서와 서버 패키지도 풍부
- ▶ <https://www.centos.org/download/> 에서 다운로드 할 수 있으며 최신 버전은 CentOS 7



디스크 파티션

- ▶ 일반적으로 리눅스를 설치할 디스크에 파티션을 나눠 사용
- ▶ 특별한 이유가 없는 한 다음과 같은 5개 정도의 파티션으로 나누어 설치

파티션	포맷 및 마운트 디렉토리
부트 파티션	ext3, ext4, xfs 파일 시스템으로 포맷하고 /boot 디렉토리로 사용
루트 파티션	ext3, ext4, xfs 파일 시스템으로 포맷하고 / 디렉토리로 사용
swap 파티션	swap 영역으로 포맷
kdump	ext3, ext4, xfs 파일 시스템으로 포맷하고 마운트 하지 않음
데이터용 파티션	애플리케이션 데이터영역으로서 사용

디스크 파티션

: 종류와 용량 배정

- ▶ 부트 파티션 : 커널과 초기 **RAM** 디스크를 포함. **GRUB**에 관련된 파일을 저장
 - ▶ 128MB 정도면 충분
- ▶ 루트 파티션(/) : 인스톨 예정 **RPM** 패키지와 애플리케이션 소프트웨어를 인스톨할 수 있는 정도의 사이즈가 필요
- ▶ Swap 영역은 **RHEL**에서 추천하는 계산 방식을 따름
 - ▶ 메모리 **2GB** 이하 : 물리 메모리의 2배
 - ▶ 메모리 **2GB** 이상 : 물리 메모리 + **2GB**
- ▶ **Kdump** : 물리 메모리 사이즈의 **2~3배** 정도
- ▶ 데이터 파티션 : 애플리케이션 데이터 영역으로 운용 중에 데이터가 계속 증가할 영역으로 따로 파티션을 설정해 두는 것이 좋음

디스크 파티션

: 디스크 관리의 기본 원칙

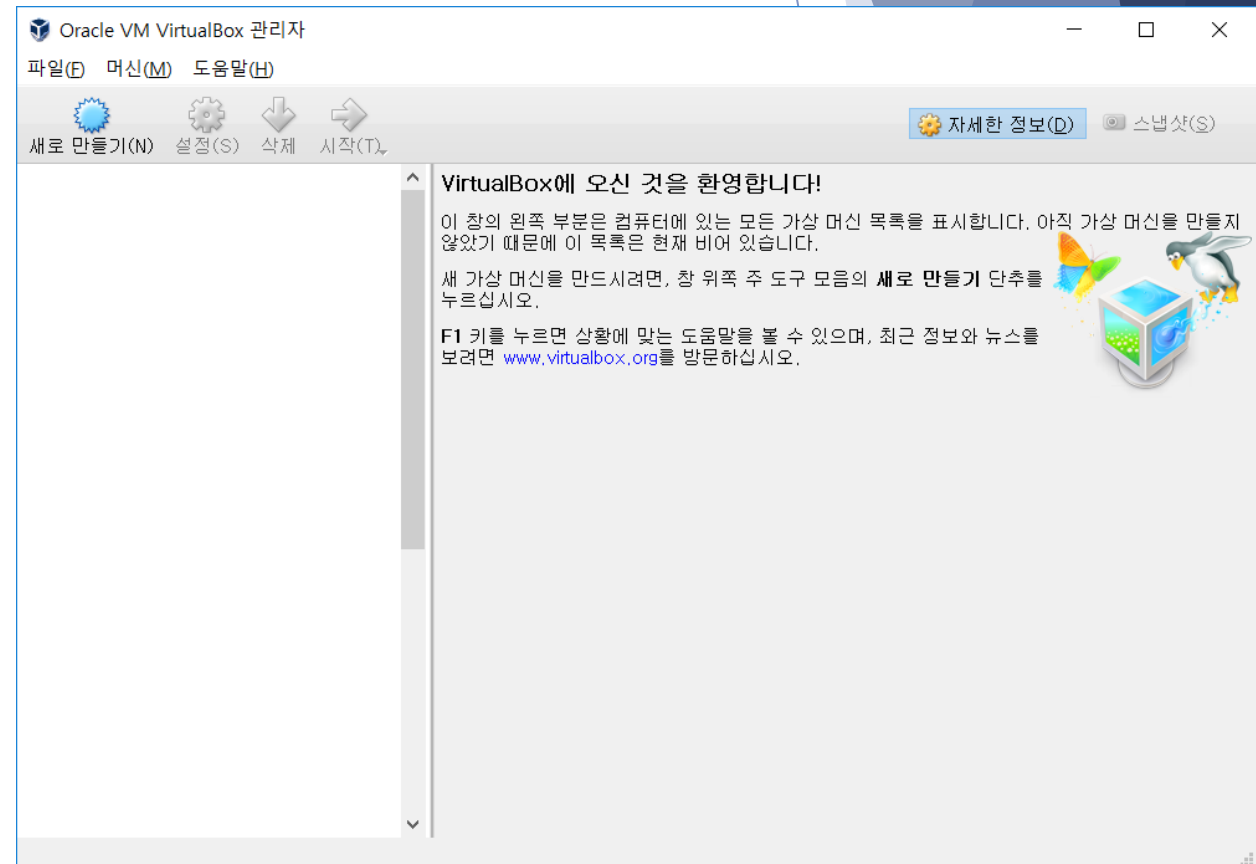
- ▶ 디스크 파티션 설정 화면에서는 “Default Layout”이 아닌 “Custom Layout”을 선택, 명시적으로 파티션을 구성할 것을 권장
- ▶ LVM(Logical Volume Management)는 사용하지 않는 것이 원칙(설치시)
- ▶ 데이터 양이 일정한 영역과 증가하는 영역으로 나뉘서 관리하는 것이 디스크 관리의 기본 원칙

Linux Install

With VirtualBox

VirtualBox

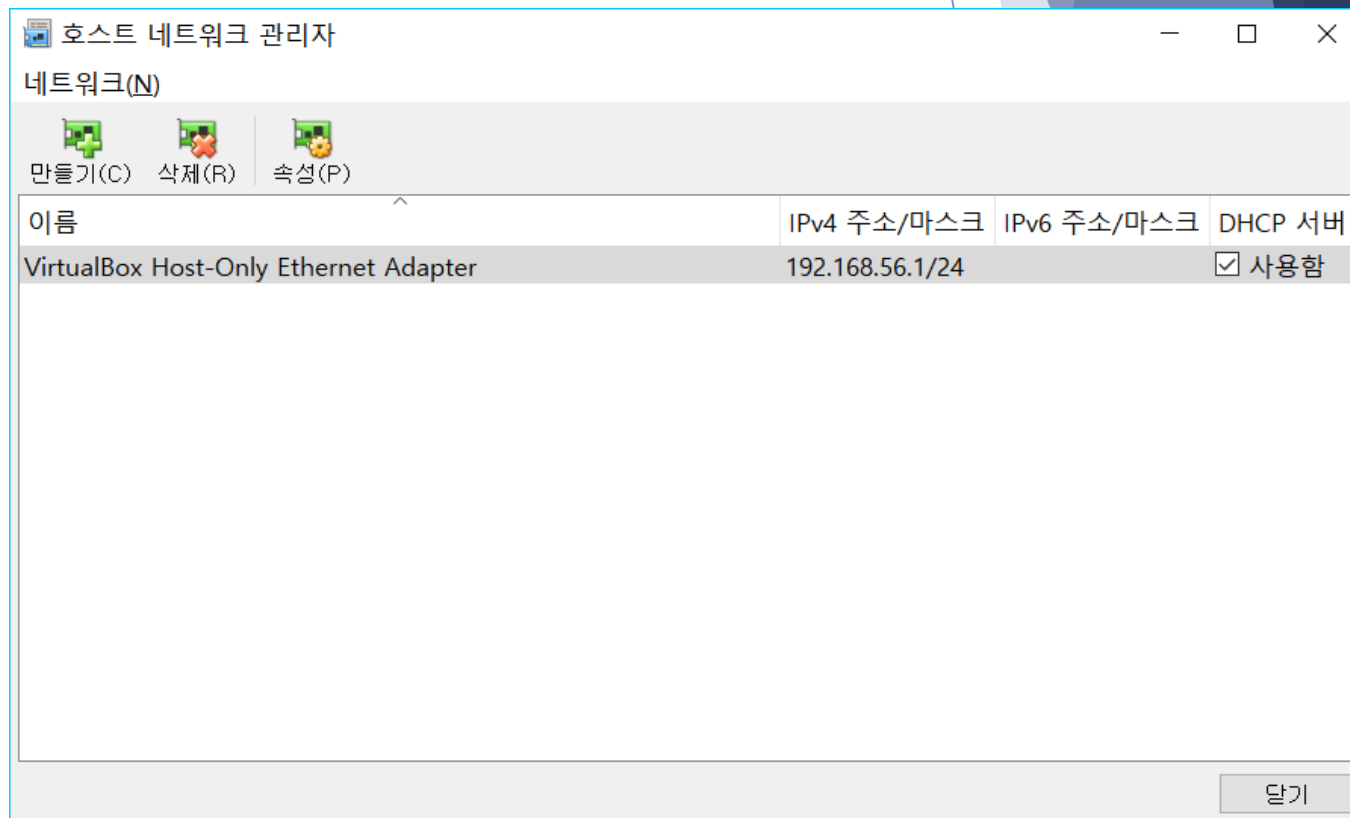
- ▶ Oracle에서 제공하는 가상환경 컨테이너
 - ▶ 실제 서버 장비의 운용 전에 테스트 용도로 활용
- ▶ <https://www.virtualbox.org/> 에서 다운로드 가능(현재 5.2가 최신)
- ▶ 최근에는 가상 서버를 이용한 상용 호스팅 서비스도 안정화 되어가고 있는 추세



VirtualBox

: Network Setup

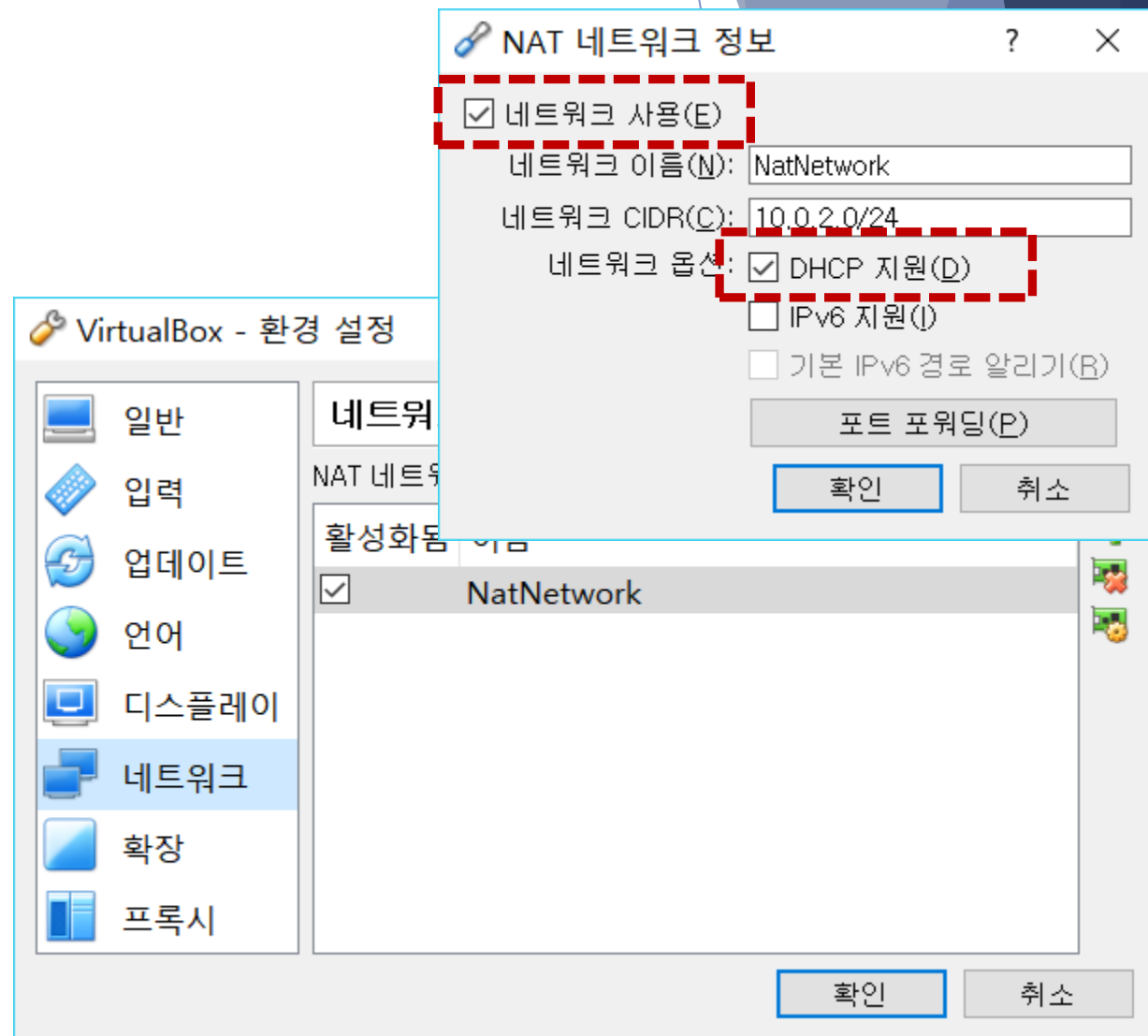
- ▶ 파일 > 호스트 네트워크 관리자에서 가상 네트워크 대역 설정



VirtualBox

: Network Setup

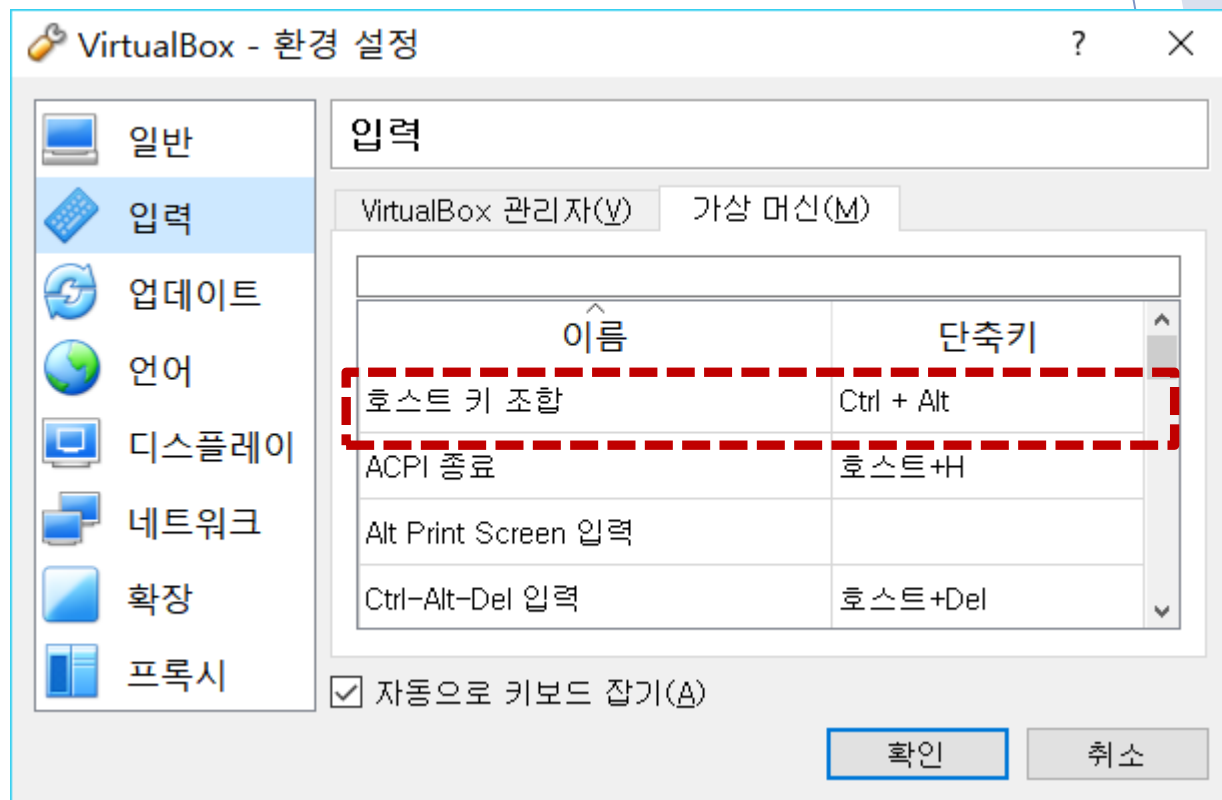
- ▶ 파일 > 환경 설정 > 네트워크
 - ▶ NAT 네트워크 설정



VirtualBox

: Host Key Setup

- ▶ 파일 > 환경 설정 > 입력 > 가상 머신 탭에서 호스트 키로 사용할 키 조합을 선택
 - ▶ 기본값 : 오른쪽 Ctrl 키

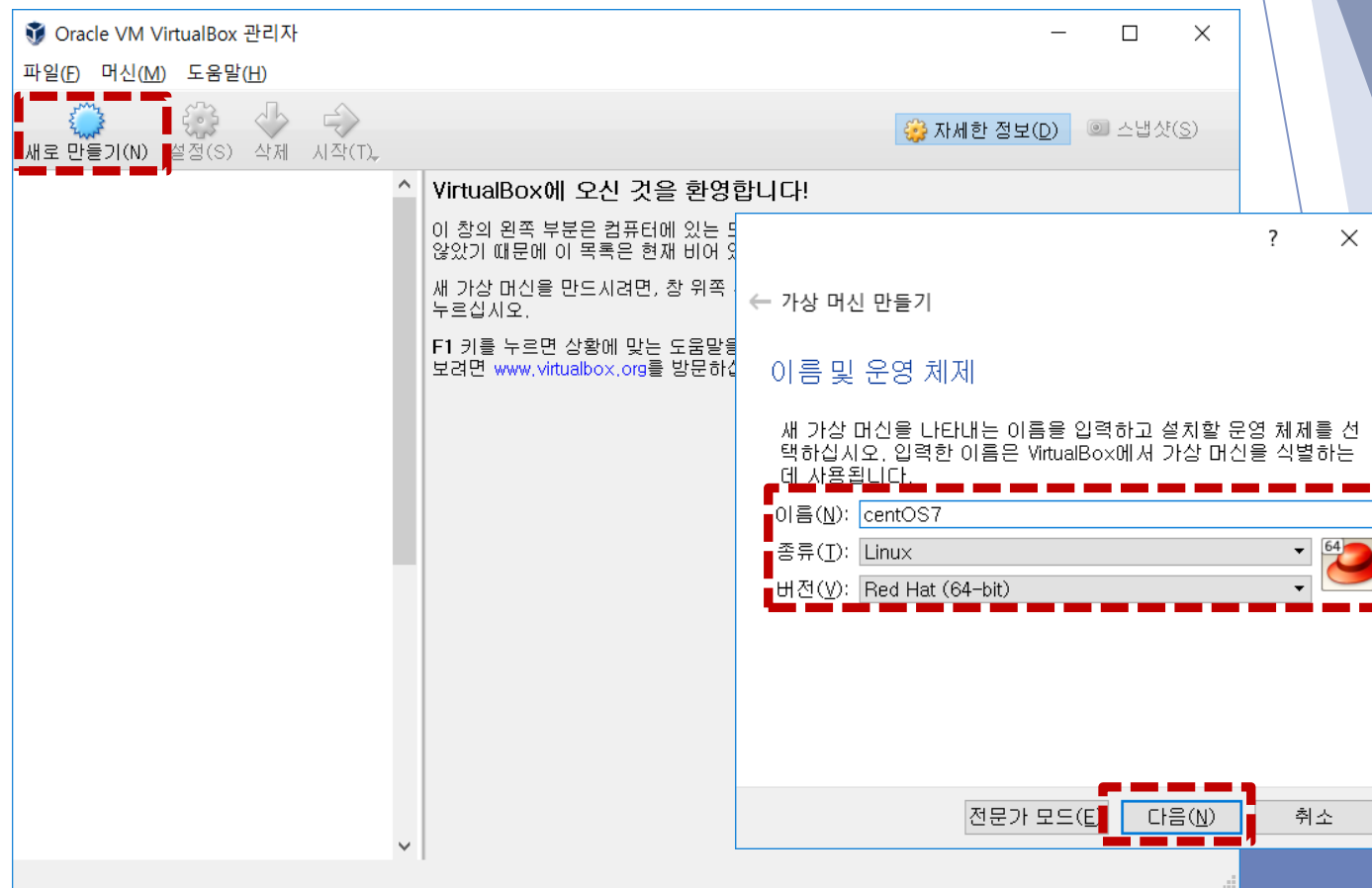


Linux Install

: on VirtualBox

▶ 새 가상머신 만들기

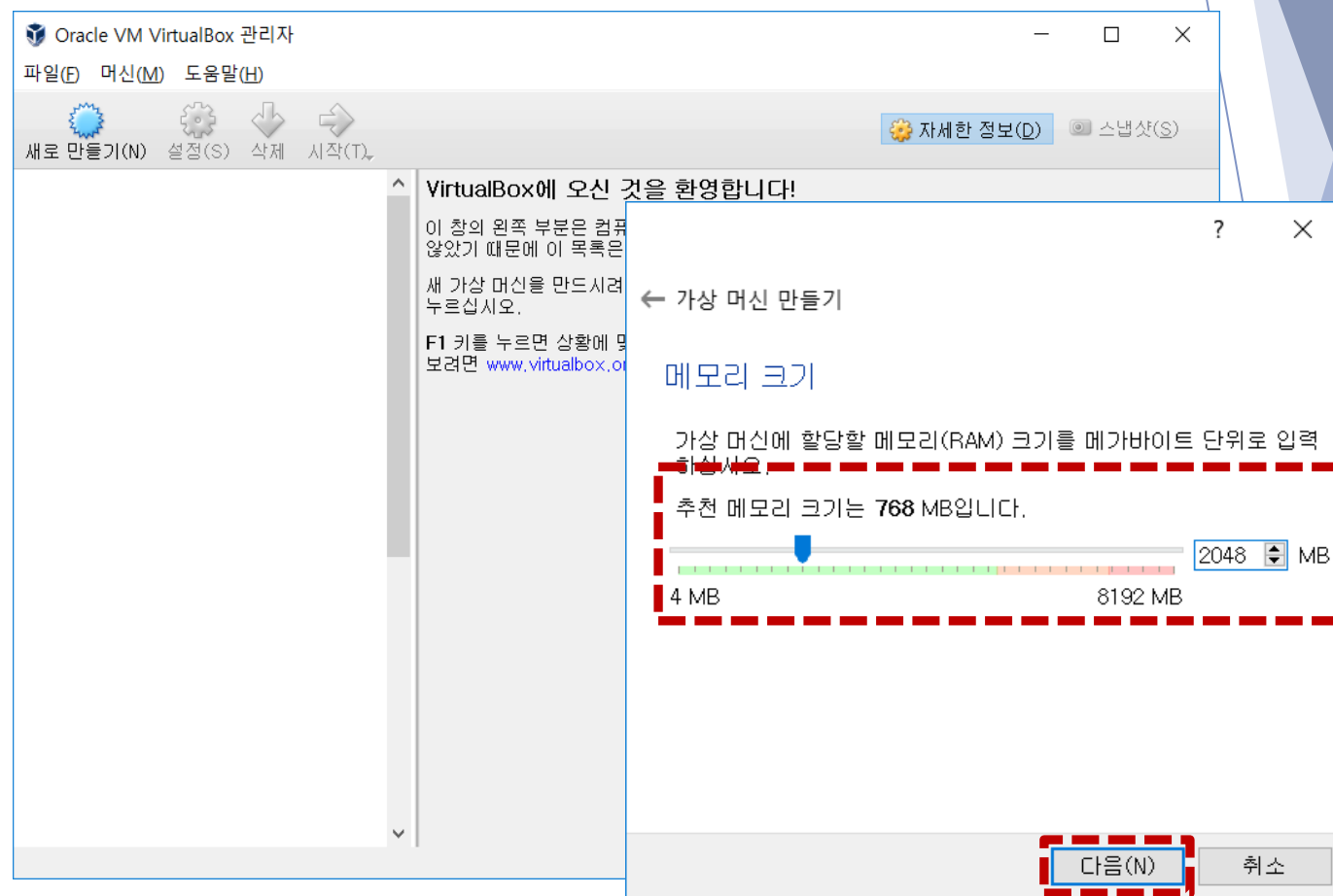
- ▶ 이름
- ▶ 종류
- ▶ 버전



Linux Install

: on VirtualBox

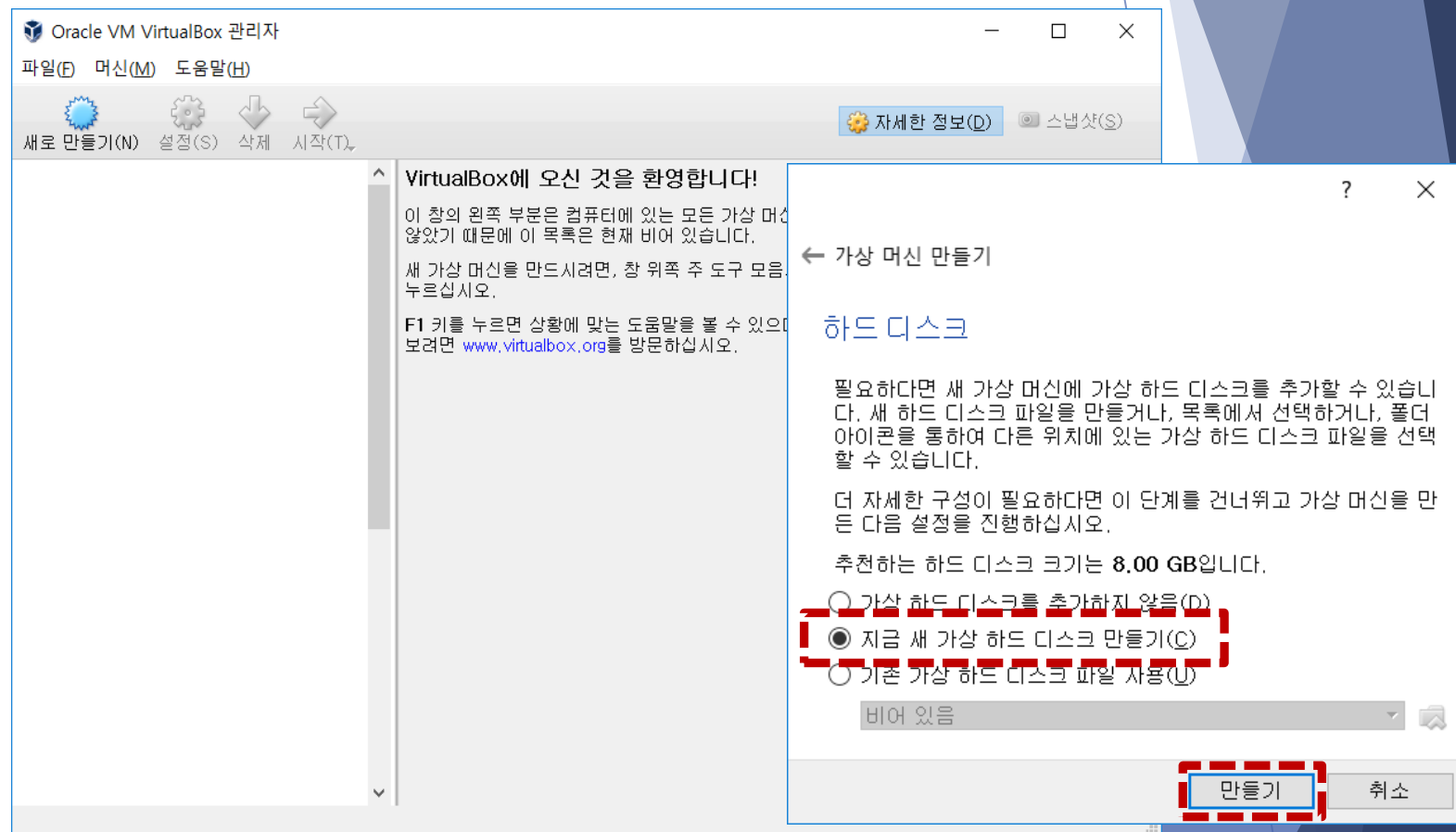
- ▶ 새 가상머신 만들기
 - ▶ 메모리 크기 설정



Linux Install

: on VirtualBox

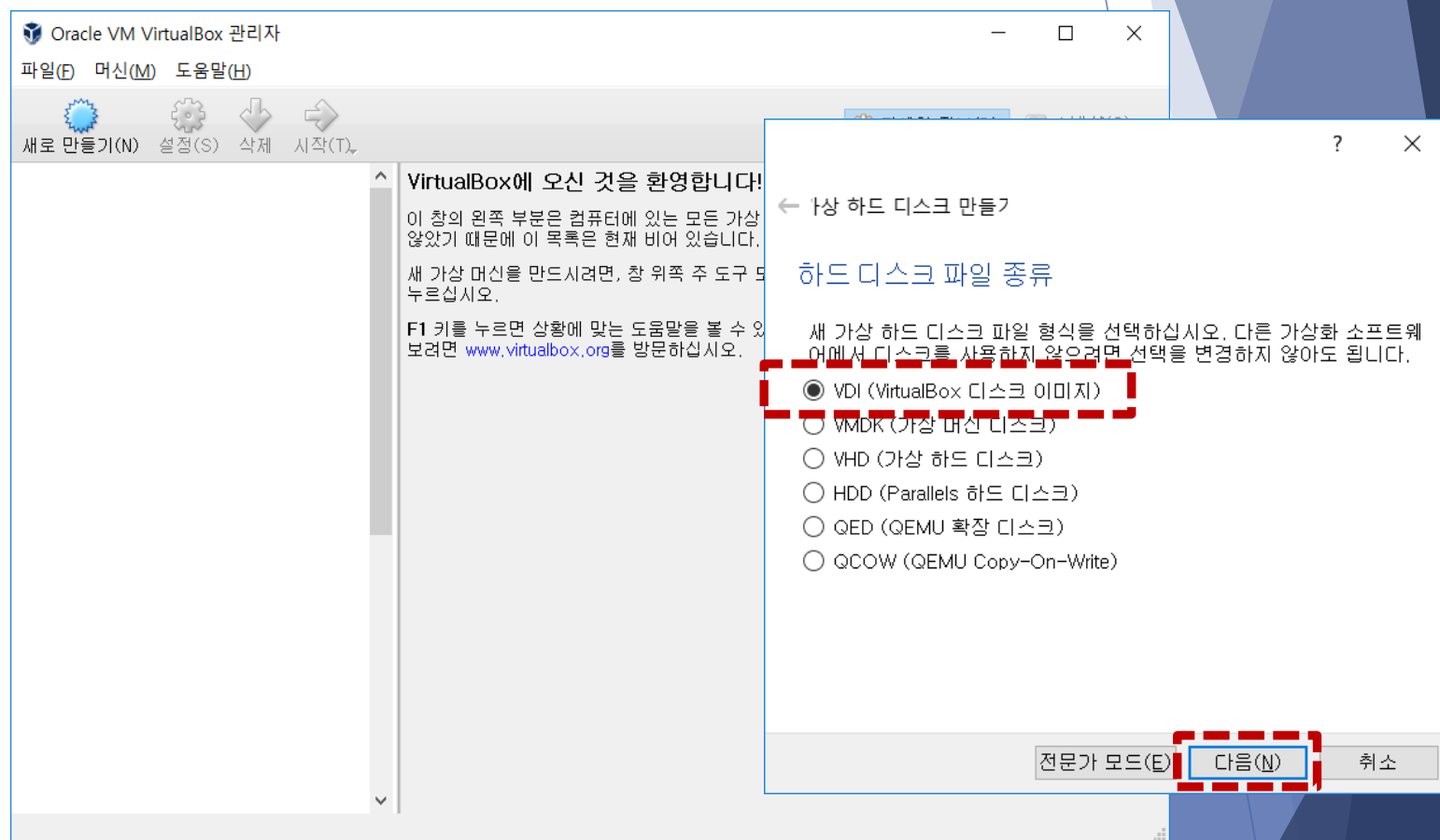
- ▶ 새 가상머신 만들기
 - ▶ 하드디스크 설정



Linux Install

: on VirtualBox

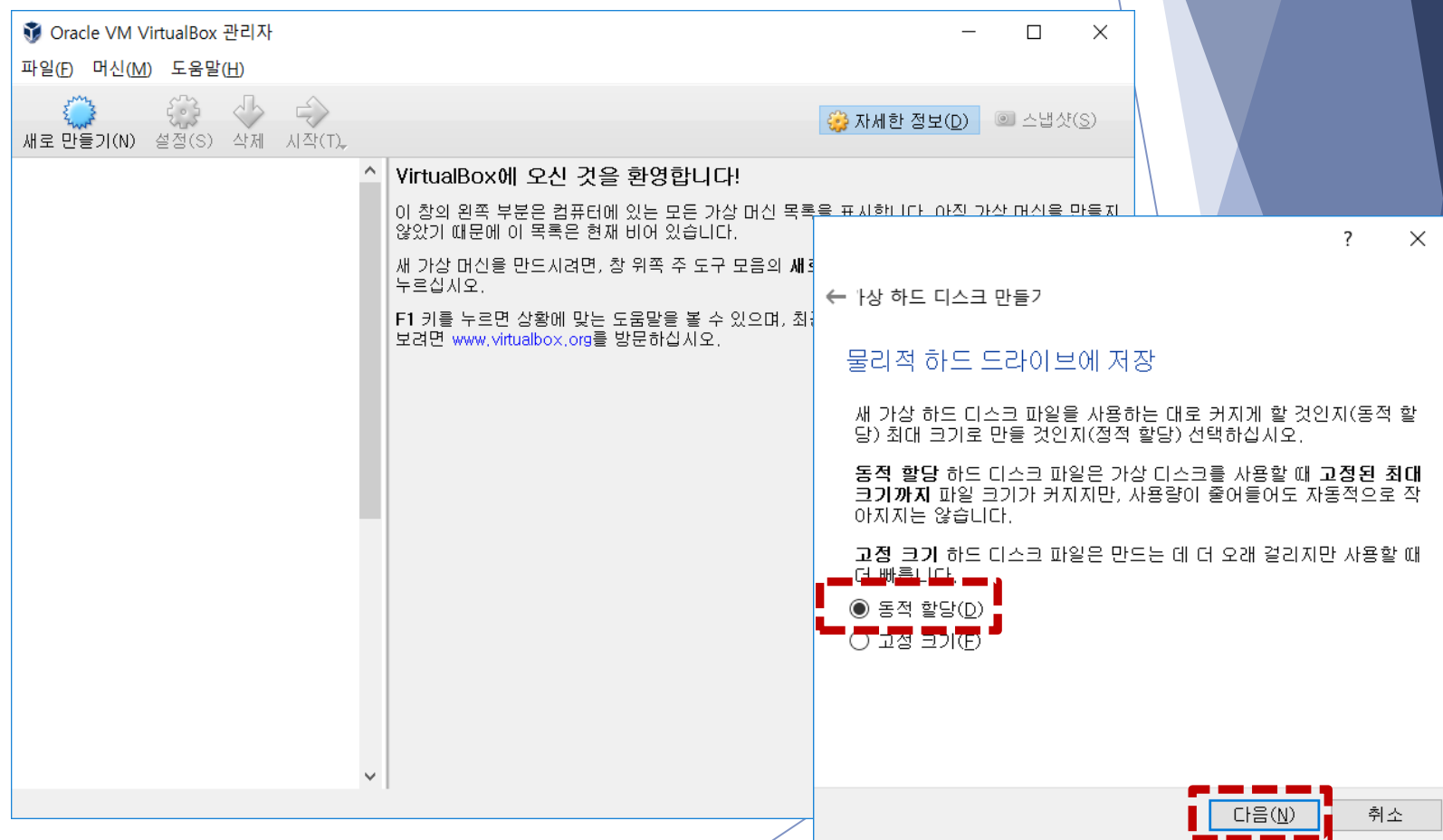
- ▶ 새 가상머신 만들기
 - ▶ 하드디스크 설정



Linux Install

: on VirtualBox

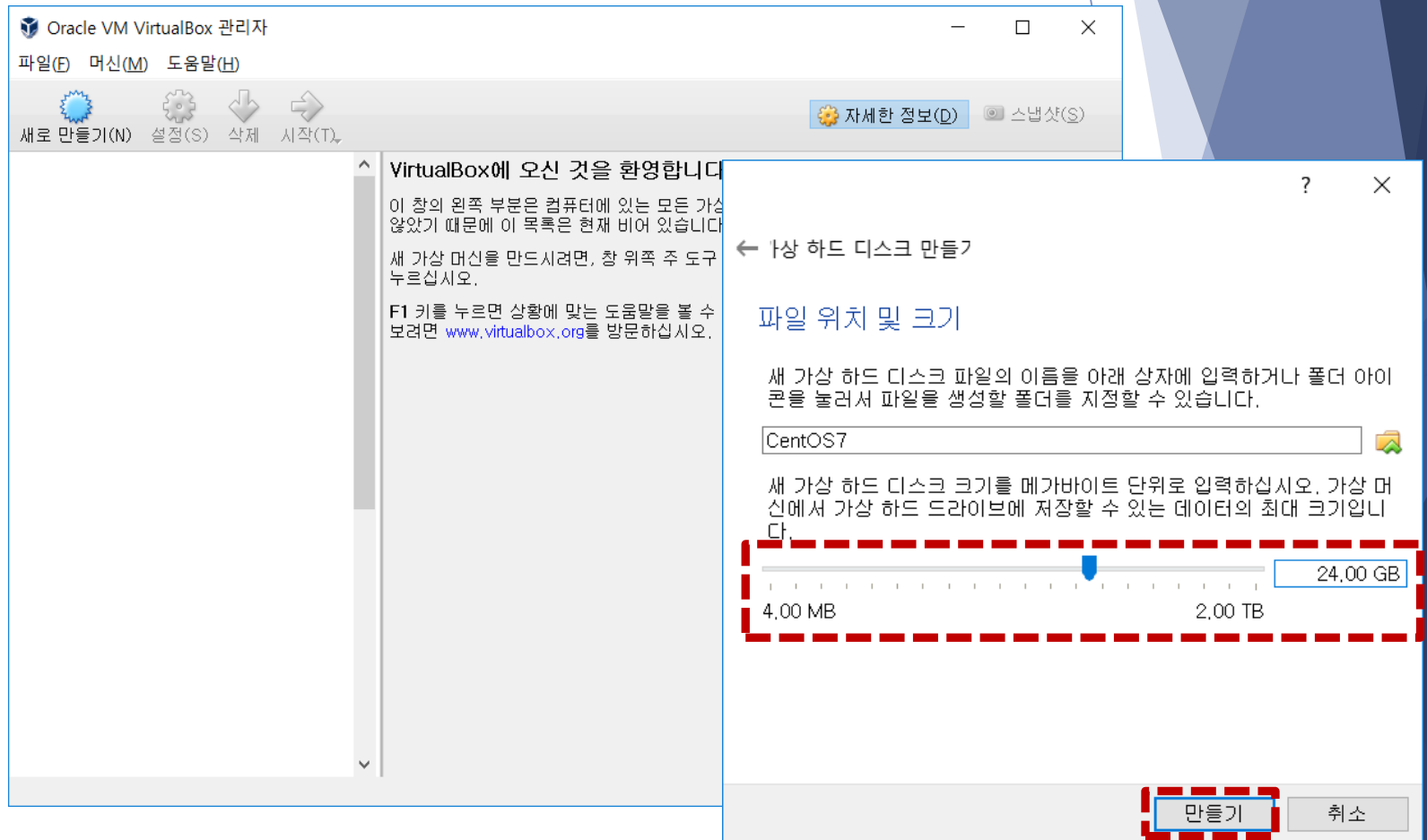
- ▶ 새 가상머신 만들기
 - ▶ 하드디스크 설정



Linux Install

: on VirtualBox

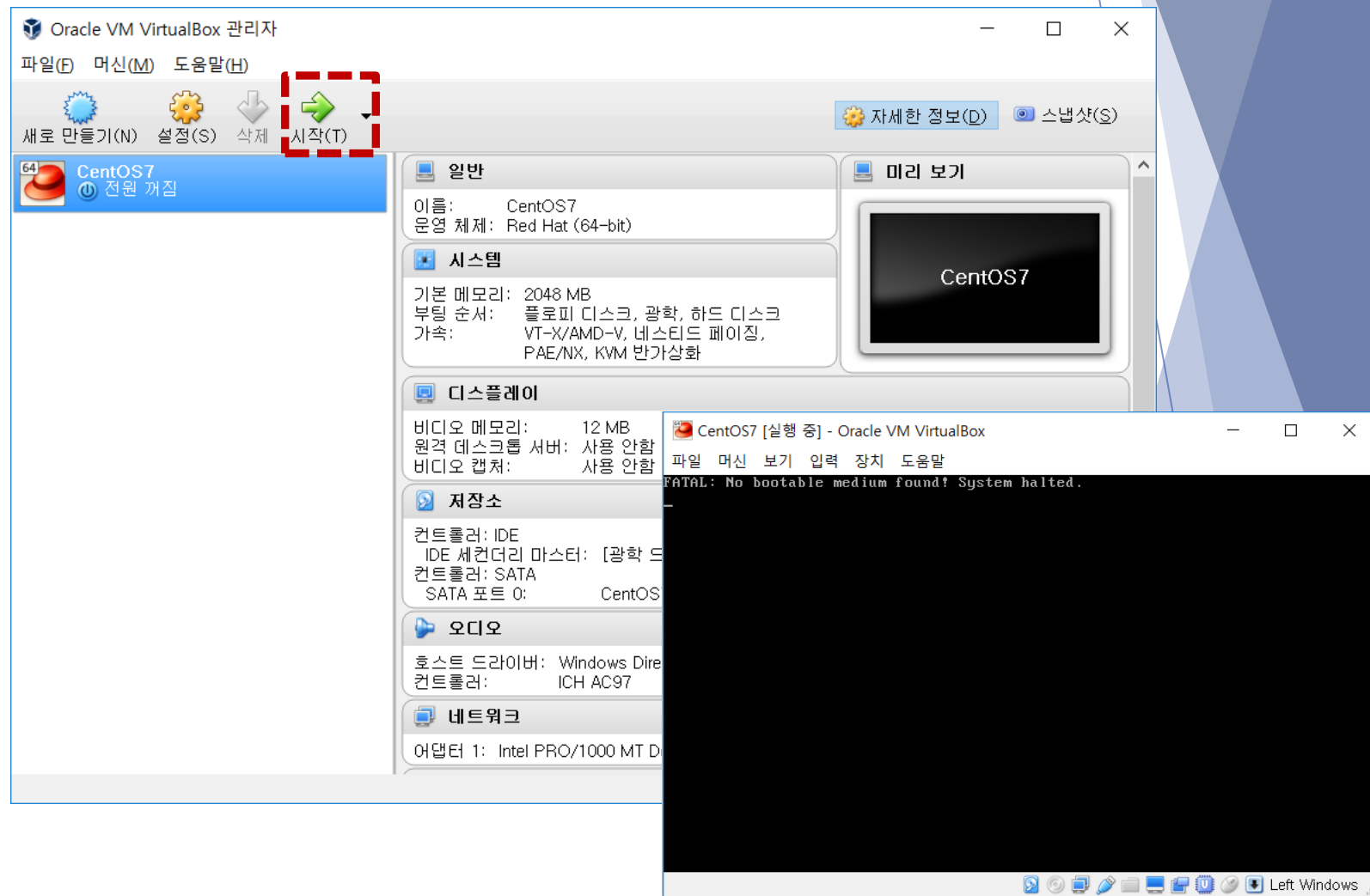
- ▶ 새 가상머신 만들기
 - ▶ 하드디스크 설정



Linux Install

: on VirtualBox

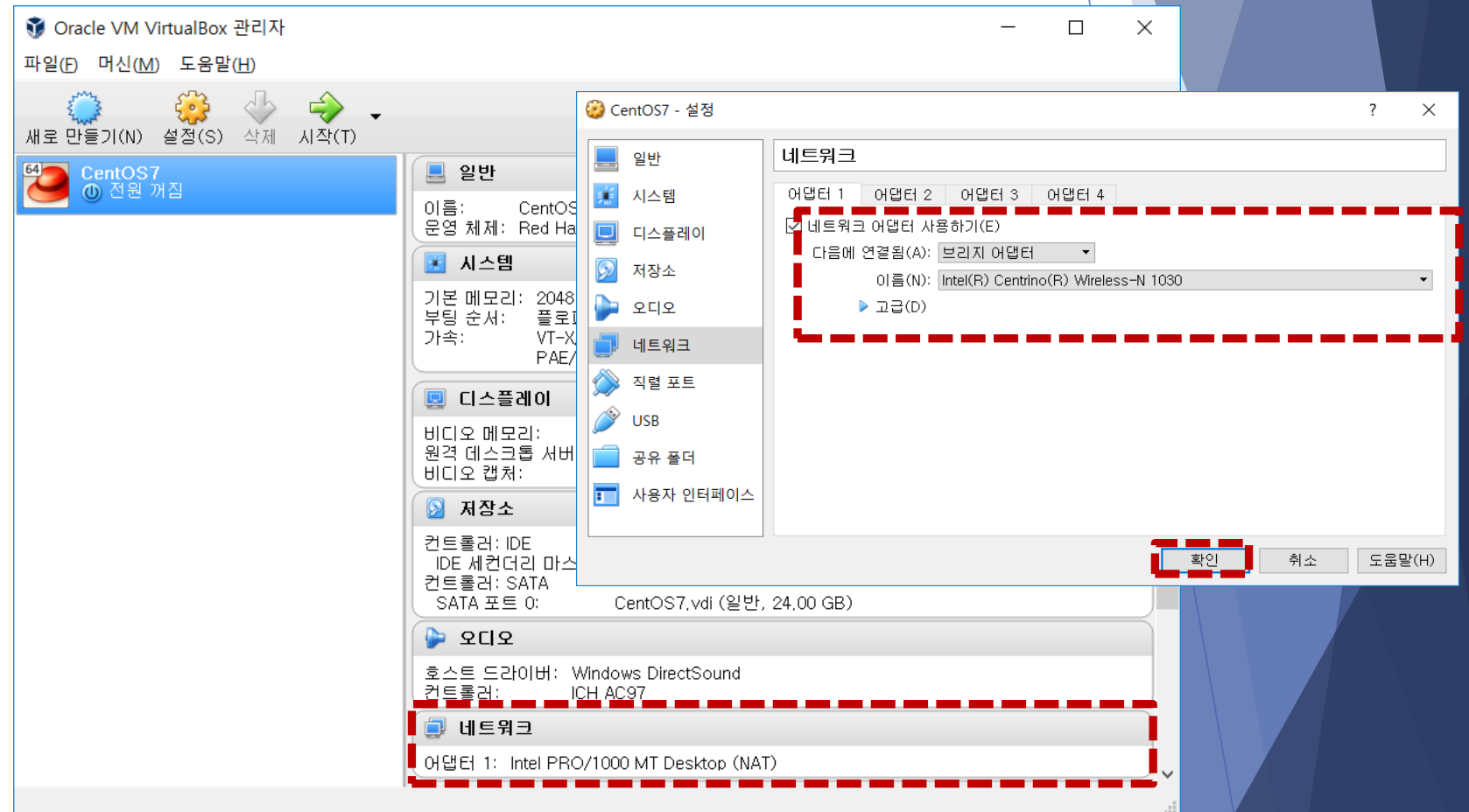
- ▶ 새 가상머신 만들기
 - ▶ 가상 머신 전원 켜기



Linux Install

: on VirtualBox

- ▶ 가상머신 설정
 - ▶ 네트워크 설정
- ▶ VirtualBox 지원 네트워크
 - ▶ NAT (기본)
 - ▶ NAT network
 - ▶ Bridge Adapter
 - ▶ Internal Network
 - ▶ Host-Only Network
 - ▶ Generic Driver

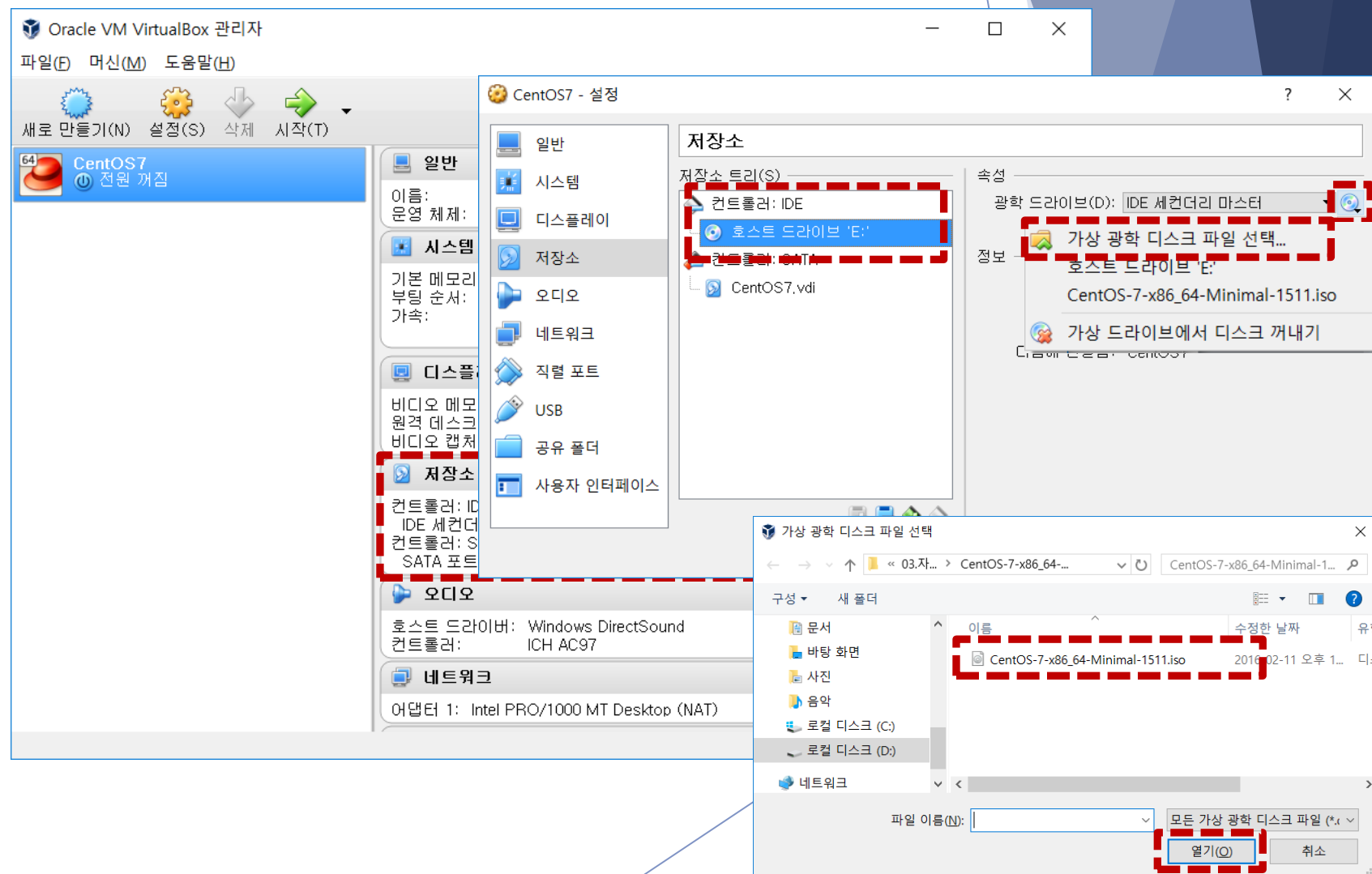


Linux Install

: on VirtualBox

▶ 가상머신 설정

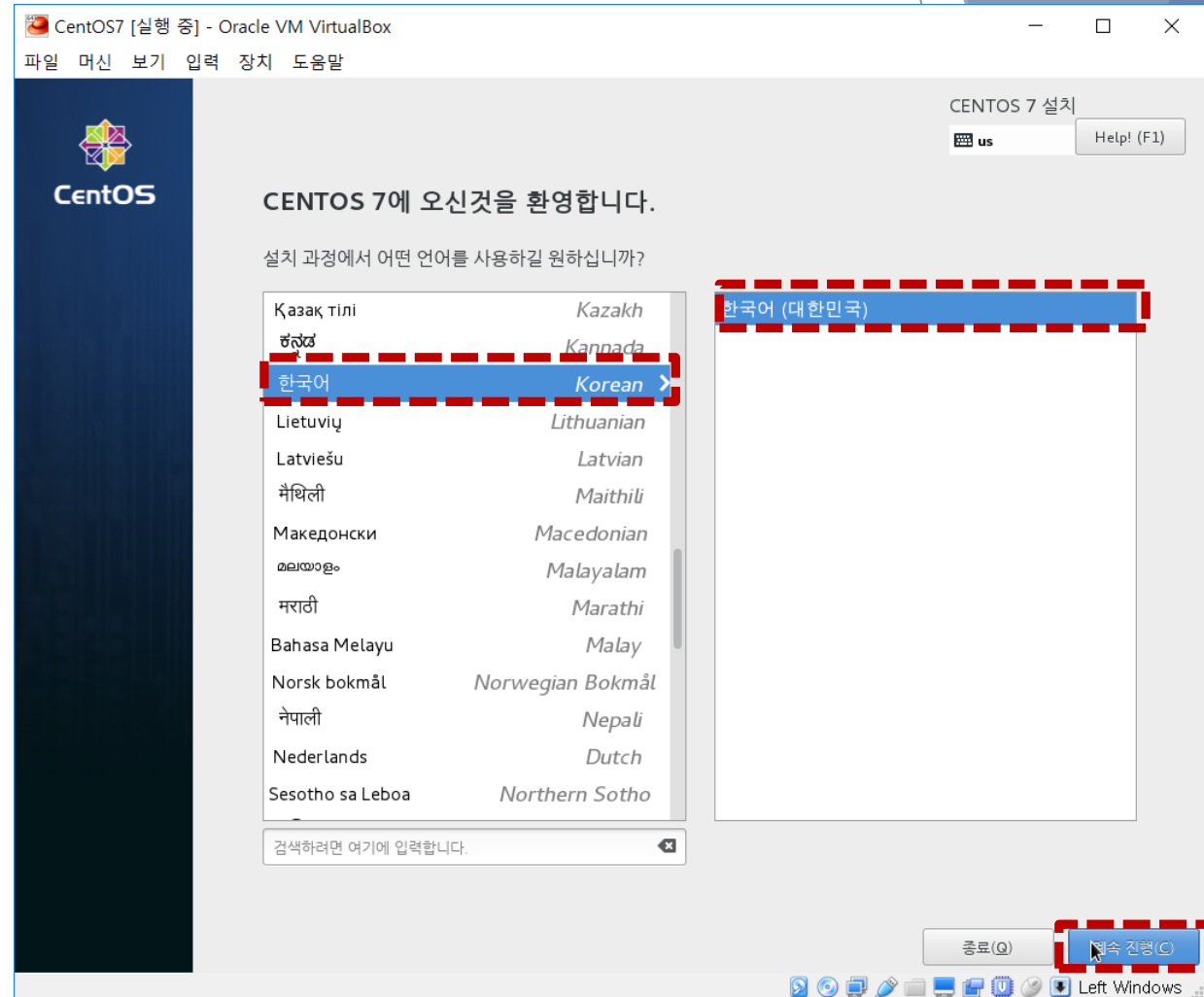
- ▶ 설치 디스크 이미지 삽입
- ▶ 삽입 후 가상머신 실행



Linux Install

: on VirtualBox

- ▶ 설치 과정 진행
 - ▶ 언어 선택



Linux Install

: on VirtualBox

- ▶ 설치 과정 진행
 - ▶ 설치 요약 항목 확인



Linux Install

: on VirtualBox

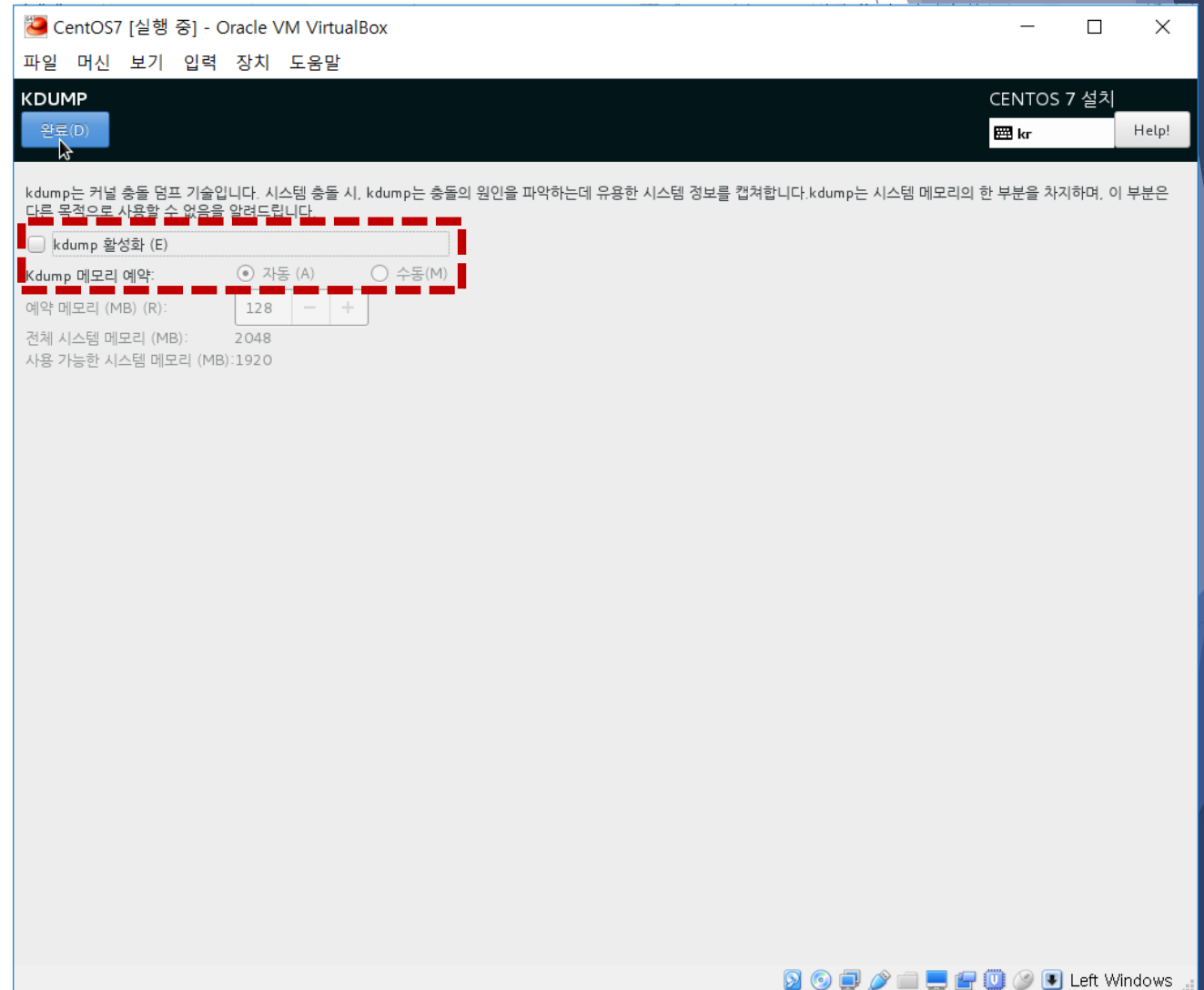
- ▶ 설치 과정 진행
 - ▶ KDUMP 비활성화



Linux Install

: on VirtualBox

- ▶ 설치 과정 진행
 - ▶ KDUMP 비활성화



Linux Install

: on VirtualBox

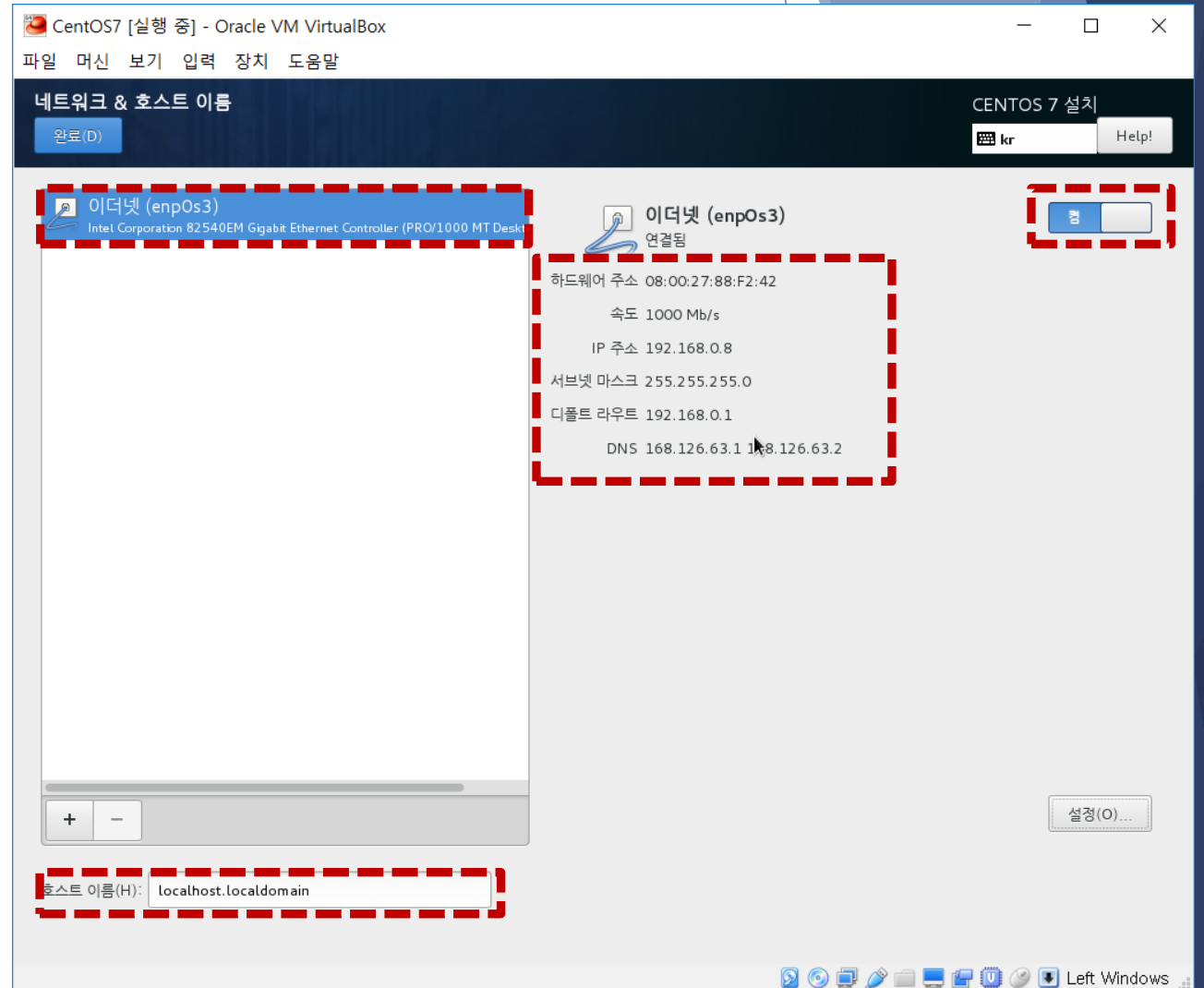
- ▶ 설치 과정 진행
 - ▶ 네트워크 및 호스트명 설정



Linux Install

: on VirtualBox

- ▶ 설치 과정 진행
 - ▶ 네트워크 및 호스트명 설정



Linux Install

: on VirtualBox

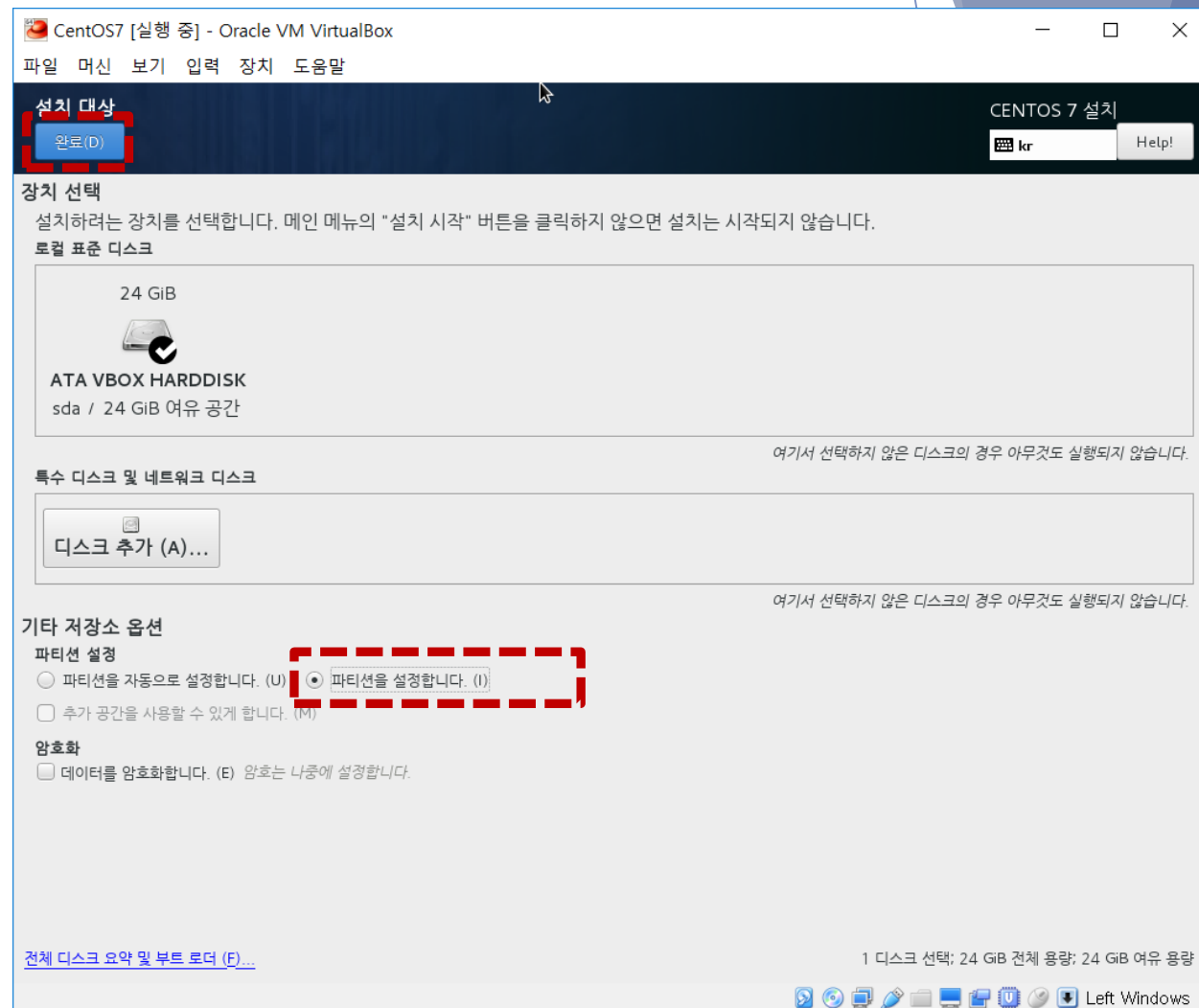
- ▶ 설치 과정 진행
 - ▶ 파티션 설정



Linux Install

: on VirtualBox

- ▶ 설치 과정 진행
 - ▶ 파티션 설정



Linux Install

: on VirtualBox

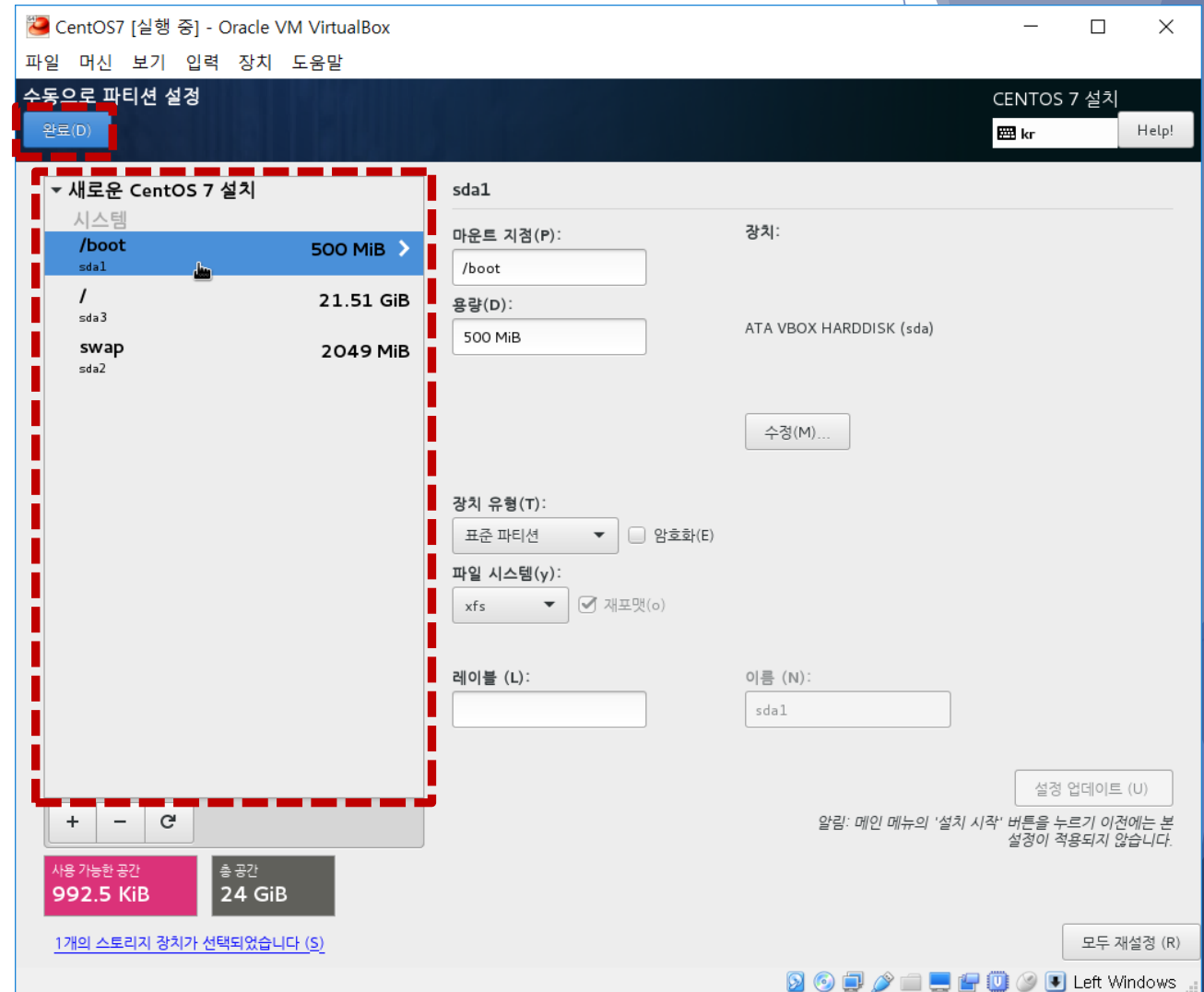
- ▶ 설치 과정 진행
 - ▶ 파티션 설정



Linux Install

: on VirtualBox

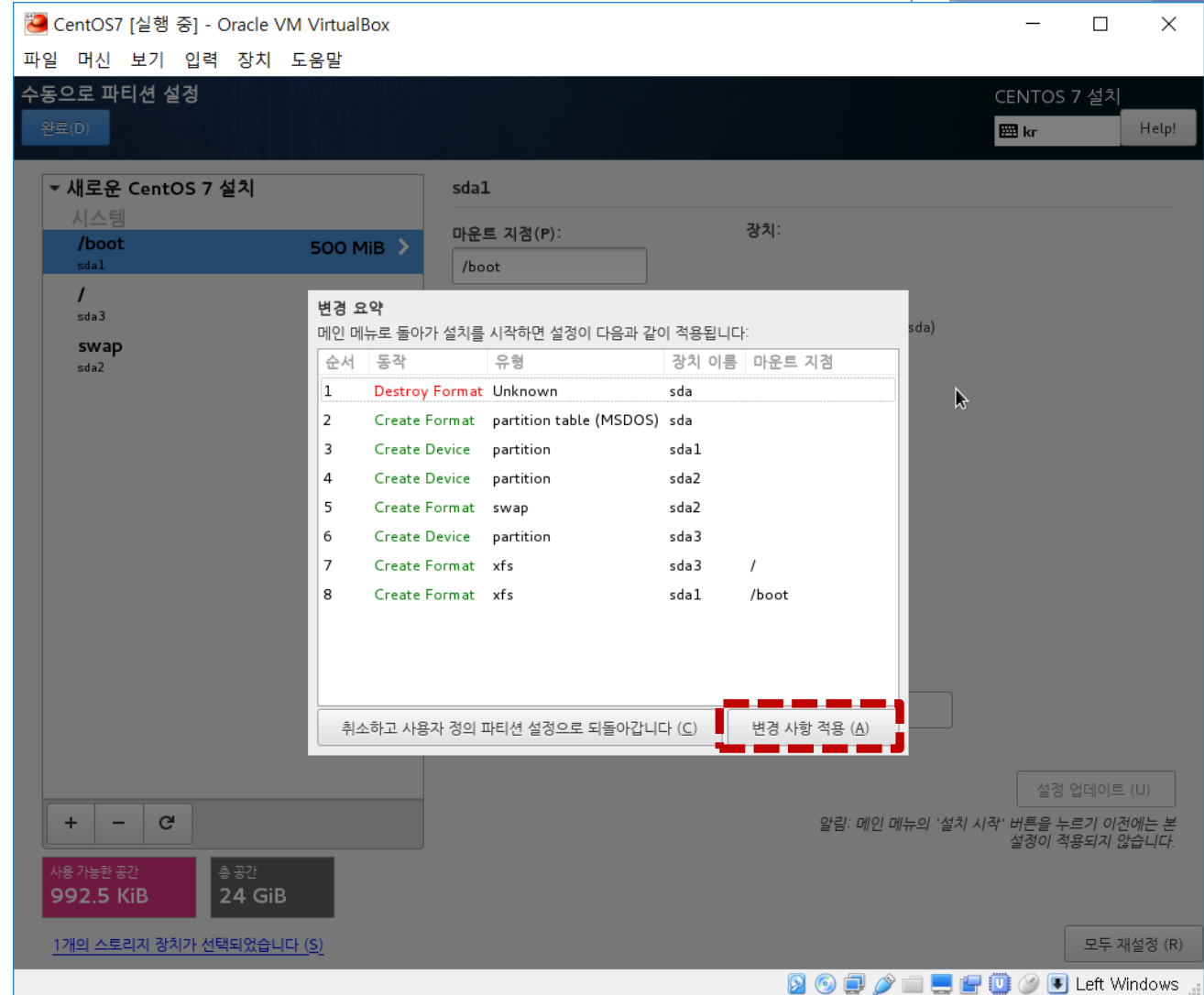
- ▶ 설치 과정 진행
 - ▶ 파티션 설정



Linux Install

: on VirtualBox

- ▶ 설치 과정 진행
 - ▶ 파티션 설정



Linux Install

: on VirtualBox

▶ 설치 시작

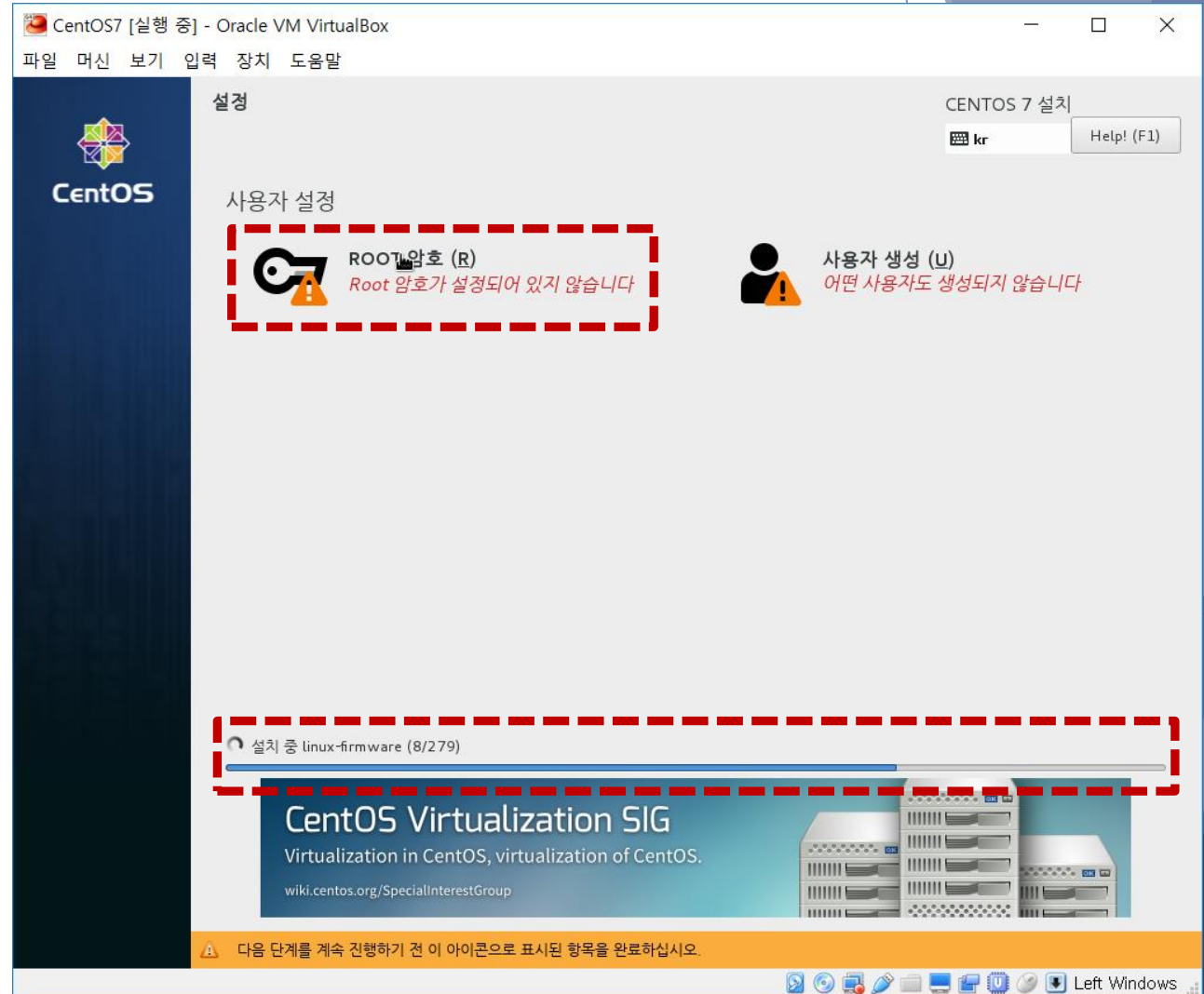


Linux Install

: on VirtualBox

▶ 설치중

▶ root 암호 설정

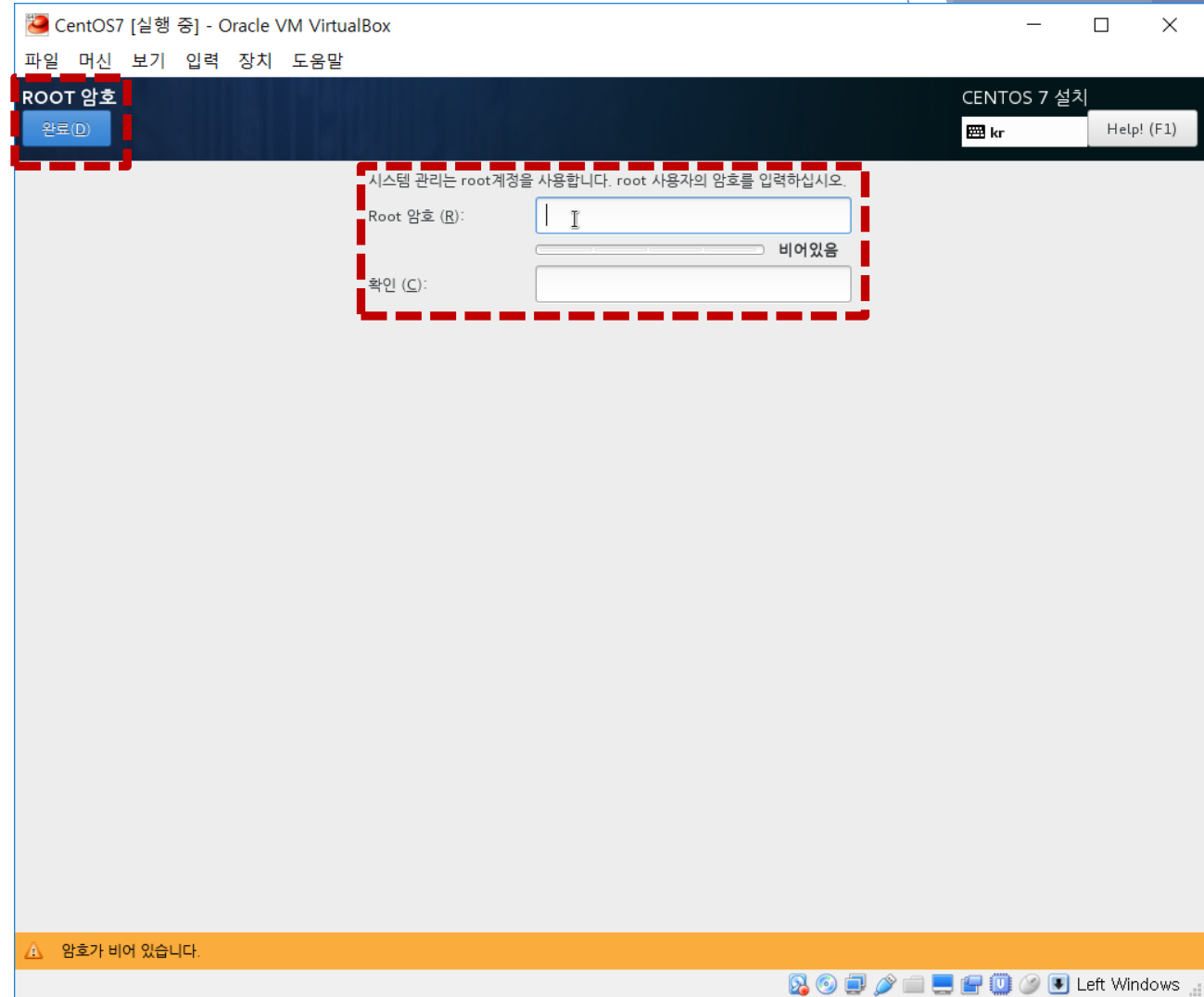


Linux Install

: on VirtualBox

▶ 설치중

▶ root 암호 설정

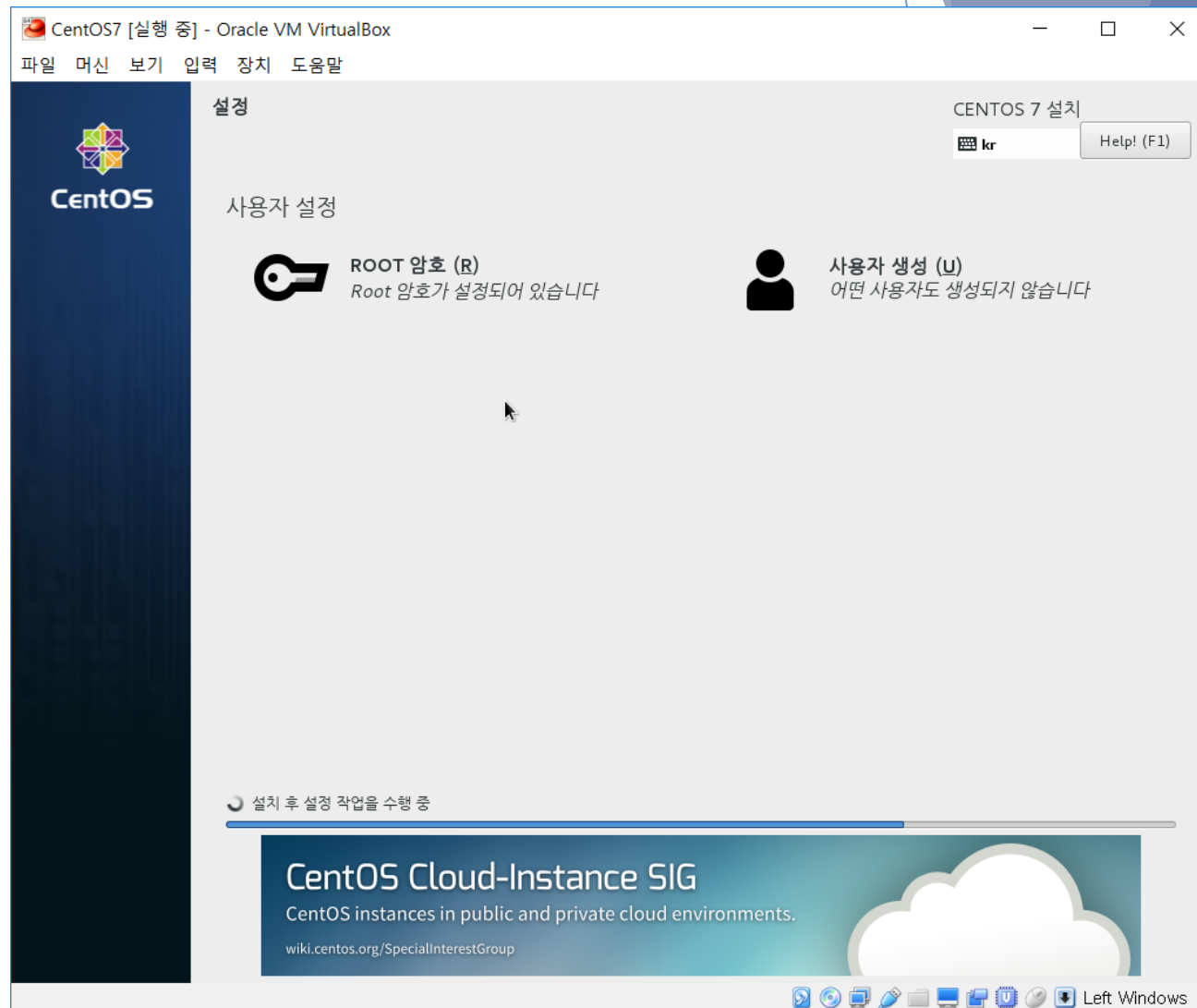


Linux Install

: on VirtualBox

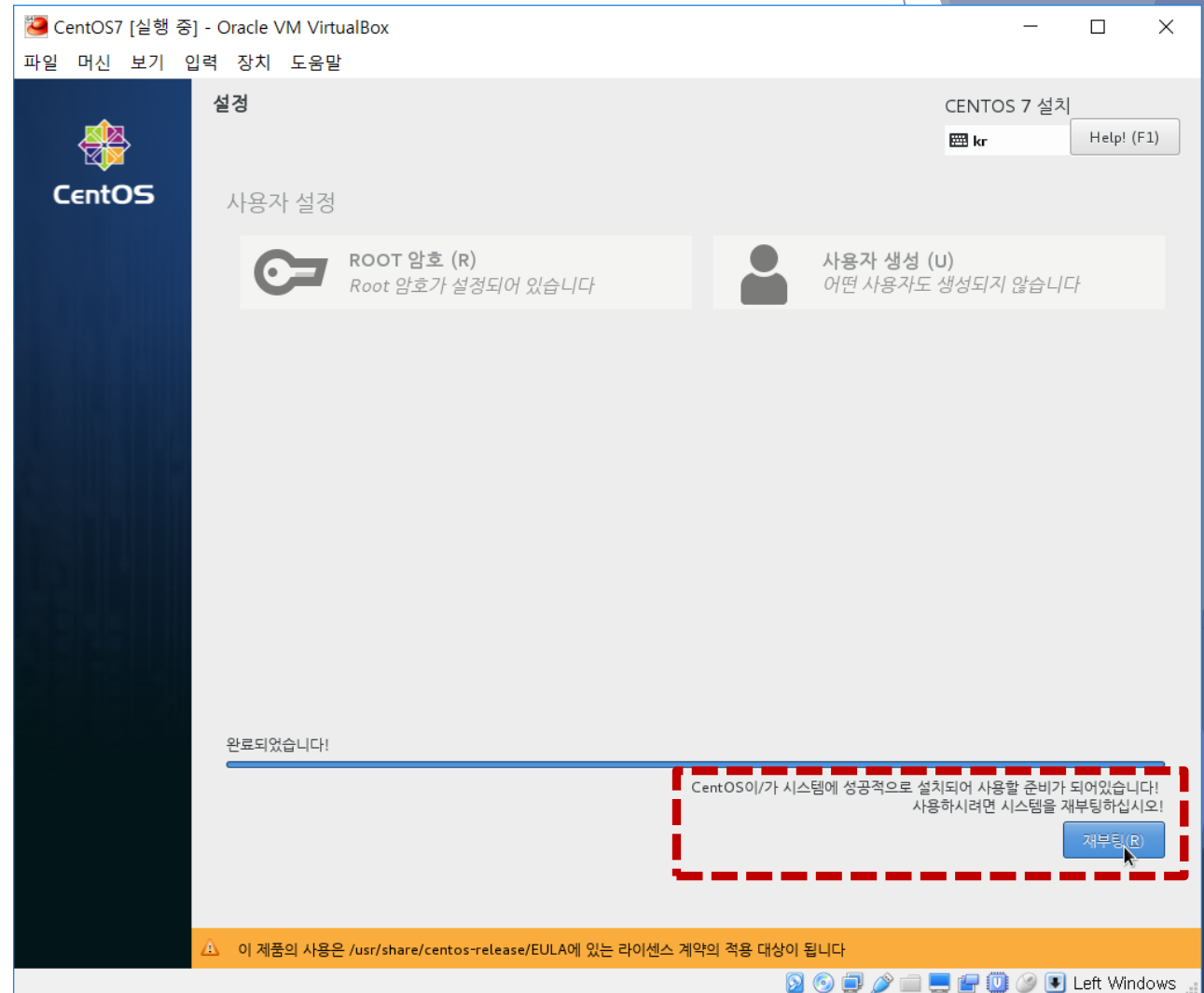
▶ 설치중

▶ root 암호 설정



Linux Install : on VirtualBox

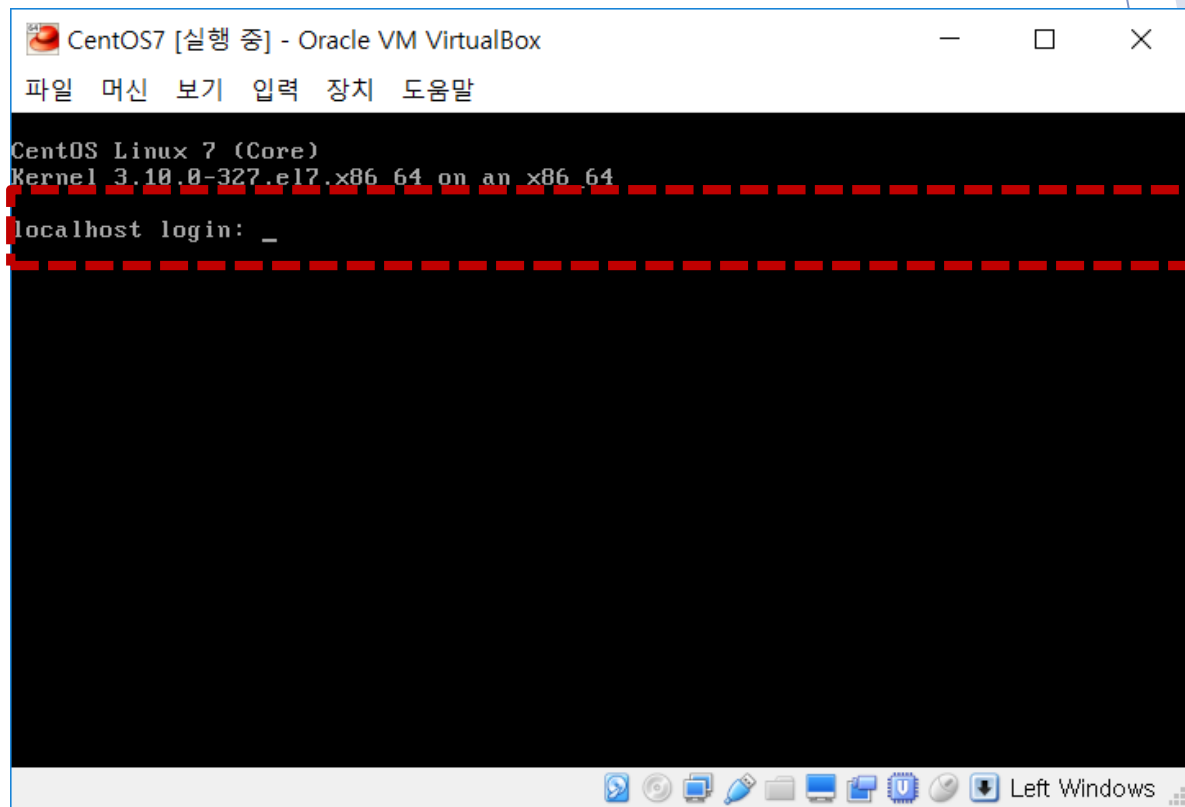
▶ 설치 완료



Linux Install

: on VirtualBox

▶ 재부팅



Linux Basic Operation

: 로그인/로그아웃

▶ 로그인

- ▶ 로그인 프롬프트에서 계정과 비밀번호를 입력

```
CenOS Linux 7 (Core)  
Kernel 3.10.0-327.el7.x86_64 on an x86_64
```

```
localhost login: _
```

▶ 로그아웃

```
[root@localhost ~]# exit  
또는  
[root@localhost ~]# logout
```

▶ 리눅스 버전 확인

```
[root@localhost ~]# uname -a  
Linux localhost.localdomain 3.10.0-327.el7.x86_64 #1 SMP Thu Nov 19 22:10:57 UTC 2015 x86_64  
x86_64 GNU/Linux  
[root@localhost ~]# uname -r  
3.10.0-327.el7.x86_64
```

Linux Basic Operation

: 시스템 종료

- ▶ shutdown : 다양한 옵션으로 시스템을 종료할 수 있는 가장 일반적인 명령

```
[root@localhost ~]# shutdown -h now          ( 지금 halt )  
[root@localhost ~]# shutdown -h +10          ( 10분 후에 halt )  
[root@localhost ~]# shutdown -r 22:10        ( 22:10 에 reboot )  
[root@localhost ~]# shutdown -c              ( 진행 중인 shutdown cancel )  
[root@localhost ~]# shutdown -h +10 "hurry up" ( 지금 접속 중인 사용자에게 메세지 전송 하고 10분 후에 halt )
```

- ▶ halt : shutdown -h now와 동일

```
[root@localhost ~]# halt
```

- ▶ init : 런레벨(시스템 실행 모드)를 설정하는 명령. 0을 주면 종료

```
[root@localhost ~]# init 0
```

- ▶ systemctl

```
[root@localhost ~]# systemctl poweroff (Halt & Power-Off)
```

```
[root@localhost ~]# systemctl halt (Halt)
```

Linux Basic Operation

: 시스템 재시작

- ▶ reboot 명령 또는 init 명령을 이용하여 runlevel 6을 주면 재시작

```
# reboot
```

(또는)

```
# init 6
```

(또는)

```
# shutdown -r now
```

- ▶ systemctl을 사용한 재시작도 가능

```
# systemctl reboot
```


Linux Basic Operation

: Run Level

- ▶ 리눅스는 시스템이 기동될 때 런 레벨(Run Level: 실행 레벨) 값을 참조
- ▶ 런 레벨의 종류
 - ▶ 0: 시스템 정지 (halt)
 - ▶ 1: 싱글 유저 모드로 기동(시스템 복구시 사용)
 - ▶ 2: 멀티 유저 (네트워크를 사용하지 않는 텍스트 멀티 유저 모드)
 - ▶ 3: 멀티 유저 (일반적인 쉘 기반의 텍스트 멀티 유저 모드)
 - ▶ 4: 사용하지 않음
 - ▶ 5: GUI 멀티 유저 모드 (X-Window)
 - ▶ 6: 시스템 재기동 (Reboot)
- ▶ 일반적으로 서버를 운용할 때는 3 또는 5를 지정
- ▶ 보통 서버에서는 GUI를 사용하지 않기 때문에 3을 지정하는 것이 일반적

Run Level	Name	Description
0	<i>Halt</i>	Shuts down all services when the system will not be rebooted.
1	<i>Single User</i>	Used for system maintenance. No Networking capabilities.
2	<i>MultiUser</i> <i>No Network Support</i>	Used for maintenance and system testing.
3	<i>MultiUser</i> <i>Network Support</i>	Non-Graphical Text Mode operations for server systems.
4	-	Custom Mode, used by SysAdmin
5	<i>Graphical</i> <i>X11</i>	Graphical login with same usability of Run Level 3.
6	<i>Reboot</i>	Shuts down all services when the system is being rebooted.

Linux Basic Operation

: Run Level

- ▶ RHEL7에서는 `init`이 `systemd`로 대체되어 숫자 기반의 런레벨이 아니라 각 런레벨에 대한 설정 세트를 통해서 런레벨을 변경함

```
#systemctl rescue (런레벨1)
```

```
#systemctl isolate multi-user.target (런레벨2)  
#systemctl isolate runlevel3.target (런레벨2)
```

```
#systemctl isolate graphical.target (런레벨5)  
#systemctl isolate runlevel5.target (런레벨5)
```

Targets in systemd

- `poweroff.target`
- `rescue.target`
- `multiuser.target`
- `multiuser.target`
- `multiuser.target`
- `graphical.target`
- `reboot.target`

Runlevels with init

- 0
- 1
- 2
- 3
- 4
- 5
- 6

Linux Basic Operation

: Run Level

- ▶ /etc/inittab에 설정된 다음 내용을 리눅스가 기동될 때 참고

```
id:3:initdefault:
```

- ▶ cat 명령으로 내용을 확인

```
[root@localhost ~]# cat /etc/inittab
# inittab is no longer used when using systemd.
#
# ADDING CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
#
# Ctrl-Alt-Delete is handled by /usr/lib/systemd/system/ctrl-alt-del.target
#
# systemd uses 'targets' instead of runlevels. By default, there are two main targets:
#
# multi-user.target: analogous to runlevel 3
# graphical.target: analogous to runlevel 5
#
# To view current default target, run:
# systemctl get-default
#
# To set a default target, run:
# systemctl set-default TARGET.target
#
```

Linux Basic Operation

: Run Level

- ▶ 마찬가지로 RHEL7에서 systemd로 대체되어 이 부분도 변경됨
- ▶ /lib/systemd/system 파일의 target 파일들의 설정 내용을 참고함

```
[root@localhost ~]# ls -l /lib/systemd/system | grep runlevel
```

```
lrwxrwxrwx. 1 root root 15 2월 11 17:28 runlevel0.target -> poweroff.target
lrwxrwxrwx. 1 root root 13 2월 11 17:28 runlevel1.target -> rescue.target
drwxr-xr-x. 2 root root 49 2월 11 17:28 runlevel1.target.wants
lrwxrwxrwx. 1 root root 17 2월 11 17:28 runlevel2.target -> multi-user.target
drwxr-xr-x. 2 root root 49 2월 11 17:28 runlevel2.target.wants
lrwxrwxrwx. 1 root root 17 2월 11 17:28 runlevel3.target -> multi-user.target
drwxr-xr-x. 2 root root 49 2월 11 17:28 runlevel3.target.wants
lrwxrwxrwx. 1 root root 17 2월 11 17:28 runlevel4.target -> multi-user.target
drwxr-xr-x. 2 root root 49 2월 11 17:28 runlevel4.target.wants
lrwxrwxrwx. 1 root root 16 2월 11 17:28 runlevel5.target -> graphical.target
drwxr-xr-x. 2 root root 49 2월 11 17:28 runlevel5.target.wants
lrwxrwxrwx. 1 root root 13 2월 11 17:28 runlevel6.target -> reboot.target
-rw-r--r--. 1 root root 761 11월 20 13:49 systemd-update-utmp-runlevel.service
```

```
[root@localhost ~]#
```

Linux Basic Operation

: Run Level

- ▶ **systemd** 상에서 부팅 런레벨은 다음 명령으로 설정하고 현재 설정 상태를 알아볼 수 있음

```
[root@localhost ~]# systemctl get-default  
multi-user.target
```

```
[root@localhost ~]# systemctl set-default graphical.target  
Removed symlink /etc/systemd/system/default.target.  
Created symlink from /etc/systemd/system/default.target to /usr/lib/systemd/system/graphical.target.
```

```
[root@localhost ~]# systemctl set-default multi-user.target  
Removed symlink /etc/systemd/system/default.target.  
Created symlink from /etc/systemd/system/default.target to /usr/lib/systemd/system/multi-user.target.
```

```
[root@localhost ~]# systemctl get-default  
multi-user.target
```

```
[root@localhost ~]#
```

Linux Basic Setup

: 패키지 설치 및 업데이트

- ▶ yum 커맨드를 사용, 패키지 업데이트 및 새로운 패키지 설치가 가능
- ▶ yum 레파지토리의 구성 내용을 확인해 보시다

```
[root@localhost ~]# yum repolist
```

Loaded plugins: fastestmirror

Loading mirror speeds from cached hostfile

* base: centos.mirror.cdnetworks.com

* extras: centos.mirror.cdnetworks.com

* updates: centos.mirror.cdnetworks.com

repo id	repo name	status
base/7/x86_64	CentOS-7 - Base	9,007
extras/7/x86_64	CentOS-7 - Extras	191
updates/7/x86_64	CentOS-7 - Updates	552
repolist: 9,750		

- ▶ 네트워크를 통한 인스톨 및 업데이트 가능
- ▶ 인스톨 CD(or DVD)의 패키지가 업데이트 되었을 수 있기 때문에 업데이트 권장

Linux Basic Setup

: 패키지 설치 및 업데이트

▶ 설치 패키지 업데이트

```
[root@localhost ~]# yum update
```

```
Loaded plugins: fastestmirror
```

```
Loading mirror speeds from cached hostfile
```

```
* base: centos.mirror.cdnetworks.com
```

```
* extras: centos.mirror.cdnetworks.com
```

```
* updates: centos.mirror.cdnetworks.com
```

```
Resolving Dependencies
```

```
--> Running transaction check
```

```
---> Package bind-libs-lite.x86_64 32:9.9.4-29.el7 will be updated
```

```
---> Package bind-libs-lite.x86_64 32:9.9.4-29.el7_2.2 will be an update
```

```
---> Package bind-license.noarch 32:9.9.4-29.el7 will be updated
```

```
...
```

```
...
```

```
Transaction Summary
```

```
=====
```

```
Install 1 Package
```

```
Upgrade 26 Packages
```

```
Total download size: 68 M
```

```
Is this ok [y/d/N]:
```

- ▶ 확인 메시지에 **y**를 입력하면 업데이트를 진행
- ▶ 커널 업데이트를 포함하는 경우 재부팅 필요

Linux Basic Setup

: 패키지 설치 및 업데이트

▶ 새 패키지 설치

```
[root@localhost ~]# yum install rdate  
[root@localhost ~]# yum install gcc  
[root@localhost ~]# yum install make  
[root@localhost ~]# yum install wget  
[root@localhost ~]# yum install gcc-c++  
[root@localhost ~]# yum install cmake  
[root@localhost ~]# yum install net-tools  
[root@localhost ~]# yum install bind-utils  
[root@localhost ~]# yum install psmisc
```

▶ 자주 사용되는 yum 커맨드

- ▶ yum clean all : 패키지 캐시 파일 삭제
- ▶ yum list : 모든 패키지 표시
- ▶ yum remove <패키지명> : 패키지 삭제
- ▶ yum update <패키지명> : 패키지 업데이트

Linux Basic Setup

: 사용자 계정 추가

- ▶ **root** 계정은 시스템에서 허용하는 모든 일을 할 수 있는 계정. 작업 중 사고가 발생할 가능성이 많음
- ▶ **root** 계정은 시스템 설정 변경, 패키지 설치 등 꼭 필요할 때만 사용하기를 권장하며 원격 로그인 등에 사용하기에는 보안상 위험
- ▶ 가급적 **root** 계정은 꼭 필요할 때만 사용하고 일반 계정으로 들어와 작업 진행
- ▶ 사용자 추가
 - ▶ **useradd** 혹은 **adduser** 커맨드를 이용
 - ▶ **-g** 뒤에 그룹 명 부여, **-d** 뒤에는 계정 홈 디렉터리 지정
 - ▶ 로그인을 허용하기 위해 **passwd** 커맨드를 이용, 비밀번호를 설정

```
[root@localhost ~]# useradd -g wheel -d /home/bituser bituser
[root@localhost ~]# passwd bituser
bituser 사용자의 비밀번호 변경 중
새 암호:
새 암호 재입력:
passwd: 모든 인증 토큰이 성공적으로 업데이트되었습니다.
[root@localhost ~]#
```

Linux Basic Setup

: 디스크 상태 확인과 파티션 확인

▶ 디스크 상태 확인

▶ df 커맨드

```
192.168.0.8:22 - root@localhost:~ - Xshell 5 (Free for Home/School)
1 192.168.0.8:22
[root@localhost ~]# clear
[root@localhost ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3        22G   1.2G   21G   6% /
devtmpfs         992M     0  992M   0% /dev
tmpfs            1001M     0  1001M   0% /dev/shm
tmpfs            1001M   8.4M   993M   1% /run
tmpfs            1001M     0  1001M   0% /sys/fs/cgroup
/dev/sda1        497M  125M   372M  26% /boot
tmpfs            201M     0   201M   0% /run/user/0
tmpfs            201M     0   201M   0% /run/user/1000
[root@localhost ~]#
```

▶ 파티션 확인

▶ fdisk 커맨드

```
192.168.0.8:22 - root@localhost:~ - Xshell 5 (Free for Home/School)
1 192.168.0.8:22
[root@localhost ~]# fdisk -l

Disk /dev/sda: 25.8 GB, 25769803776 bytes, 50331648 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
Disk label type: dos
Disk identifier: 0x0000ec73

   Device Boot      Start         End      Blocks    Id  System
/dev/sda1 *          2048       1026047       512000    83  Linux
/dev/sda2            1026048     5222399     2098176    82  Linux swap / Solaris
/dev/sda3            5222400     50331647     22554624    83  Linux
[root@localhost ~]#
```

Linux Basic Setup

: mount와 fstab

▶ mount

- ▶ 특정 디바이스(device)를 사용하기 위해 하드웨어와 디렉터리를 연결하는 작업
- ▶ 하드웨어 중 하드디스크의 장치명이 **/dev/sda1**이라고 한다면 이 장치명을 지정된 마운트 포인트(디렉터리 위치)와 연결하는 작업

▶ /etc/fstab

- ▶ mount 설정을 영구적으로 할 수 있도록 하는 설정파일

```
$ cat /etc/fstab
```

```
#  
# /etc/fstab  
# Created by anaconda on Thu Feb 11 17:26:56 2016  
#  
# Accessible filesystems, by reference, are maintained under '/dev/disk'  
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info  
#  
UUID=81980325-9871-4c01-b67f-ff3489670ac3 / xfs defaults 0 0  
UUID=86bba47e-dca1-466c-9512-433e61220534 /boot xfs defaults 0 0  
UUID=a0c235d4-c8f8-4dc1-842b-1fa813f8a5e0 swap swap defaults 0 0
```

Linux Basic Setup

: mount와 fstab

▶ blkid

- ▶ UUID를 파티션으로 바꿔주는 명령

```
[bituser@localhost ~]$ blkid
```

```
/dev/sda1: UUID="86bba47e-dca1-466c-9512-433e61220534" TYPE="xfs"  
/dev/sda2: UUID="a0c235d4-c8f8-4dc1-842b-1fa813f8a5e0" TYPE="swap"  
/dev/sda3: UUID="81980325-9871-4c01-b67f-ff3489670ac3" TYPE="xfs"
```

▶ /etc/fstab + blkid

- ▶ 파일 시스템 장치명 /dev/sda3 는 마운트 포인트(/ 디렉터리)에 연결되어 있다
- ▶ 파일 시스템 장치명 /dev/sda3 는 xfs 파일 시스템이다
- ▶ 파일 시스템 장치명 /dev/sda3는 default(rw, nouser, auto, exec, suid) 속성을 가지는 파일 시스템이다
- ▶ 덤프(백업)가 불가능하며 fsck가 무결성 체크를 하지 않는다

/dev/sda3	/	xfs	defaults	0 0
/dev/sda1	/boot	xfs	defaults	0 0
/dev/sda2	swap	swap	defaults	0 0

Linux Basic Setup

: mount와 fstab

▶ mount, unmount 커맨드

- ▶ 현재 시스템에 마운트되어 있는 정보를 출력 (이 정보는 /etc/mtab에 저장)
- ▶ 사용자 또는 시스템이 빈번히 마운트(mount)와 마운트 해제(unmount) 하는 정보와 영구적 마운트 정보를 함께 표시

▶ CD-ROM 마운트하기

```
[root@localhost cdrom]# mount -t iso9660 /dev/cdrom /mnt/cdrom
```

▶ CD-ROM 마운트 해제하기

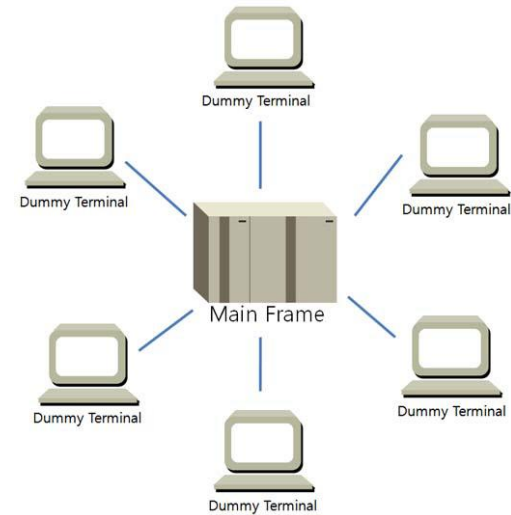
```
[root@localhost cdrom]# umount /dev/cdrom
```

Linux Basic Setup

: Terminal

▶ Dummy Terminal

- ▶ 1970년대 메인프레임 시대에 여러 사용자와 메인프레임 사이의 인터페이스 역할을 수행
- ▶ 키보드와 모니터만으로 구성되어 있었고 자체 처리 능력은 없었음
- ▶ 명령 프롬프트와 같은 텍스트 기반 터미널이라는 의미에서 콘솔(Console)이라 칭하기도 함



Linux Basic Setup

: Terminal

▶ Remote Terminal (원격 터미널)

- ▶ 더미 터미널이 데스크톱 혹은 어떤 형태의 컴퓨터로 변화
- ▶ 네트워크의 발달로 더미 터미널과 메인프레임(서버) 사이의 단순한 키보드/모니터 연결이 아닌 네트워크 연결로 클라이언트/서버(C/S) 연결이 가능
- ▶ 네트워크 연결을 통해 한쪽이 서버, 한쪽이 클라이언트가 된다
- ▶ 네트워크로 연결된 상태에서 클라이언트가 서버를 제어할 수 있는데 이를 원격 터미널이라 함
- ▶ 터미널 연결이 되면, 연결된 서버에 명령을 내리고 프로그램을 실행하기 위해 클라이언트는 서버에서 연결시 만들어준 셸을 사용하게 됨
- ▶ 원격 터미널 연결을 하기 위해서는 서버와 클라이언트에 네트워크 프로그램이 실행되어야 하고 두 프로그램간 통신을 위한 프로토콜이 필요(telnet, ssh)

Linux Basic Setup

: Terminal

- ▶ 터미널(CLI, TUI)의 장점
 - ▶ 간편하고 빠르다
 - ▶ **GUI**에 비해 가볍다
 - ▶ 강력하다
 - ▶ 언제나 사용할 수 있다 (복구 환경, **GUI**가 지원 안되는 환경)
- ▶ 터미널의 종류
 - ▶ 콘솔(Console)
 - ▶ 키보드와 키보드가 연결된 모니터
 - ▶ 터미널을 부르는 다른 이름으로 터미널과 혼용해서 부르기도 함
 - ▶ 네트워크를 통한 원격 액세스가 규범처럼 여겨지기 때문에 유용성은 높지 않음
 - ▶ 네트워크나 **GUI**를 사용하지 않는 환경에서는 필요
 - ▶ 컴퓨터의 로컬 접속은 보안의 이유로도 중요

Linux Basic Setup

: Terminal

▶ 터미널의 종류

▶ GUI 가상 터미널

- ▶ X Window에서 가상 터미널을 제공하는 프로그램
- ▶ 컴퓨터 바로 앞에 키보드와 모니터를 연결해 놓은 것과 같은 터미널 프로그램
- ▶ 다수의 터미널을 제공하는 것과 같다

▶ 원격 터미널(**telnet** 또는 **ssh**)

- ▶ 가상 터미널이라는 장점을 고스란히 제공하면서 네트워크상의 어떤 컴퓨터에서도 접속이 가능
- ▶ 보안상의 이유로 **telnet** 보다 **ssh**를 많이 사용
- ▶ **ssh**는 유닉스 계열 OS가 설치된 컴퓨터에 원격으로 접속하는 실질적 표준
- ▶ 원격 터미널을 요청하고 사용하는 클라이언트와 이를 허용하는 서버 프로그램이 있어야 함

Linux Basic Setup

: SSH 서버 설정

- ▶ SSH 서버 실행 여부 확인하기

```
$ ps -ef | grep sshd
```

- ▶ SSH 서버가 자동 실행 상태인지 확인해 봅니다

```
[root@localhost cdrom]# systemctl is-enabled sshd.service  
enabled  
[root@localhost cdrom]#
```

- ▶ enabled가 아니라면 자동 실행 상태로 바꿉니다

```
[root@localhost cdrom]# systemctl enable sshd.service
```

- ▶ 보안을 위해 root로 원격 접속을 막을 수 있습니다

- ▶ /etc/ssh/sshd_config 파일에 다음 내용을 수정합니다
PermitRootLogin no

- ▶ sshd 서버를 재시작합니다

```
[root@localhost cdrom]# systemctl restart sshd.service
```

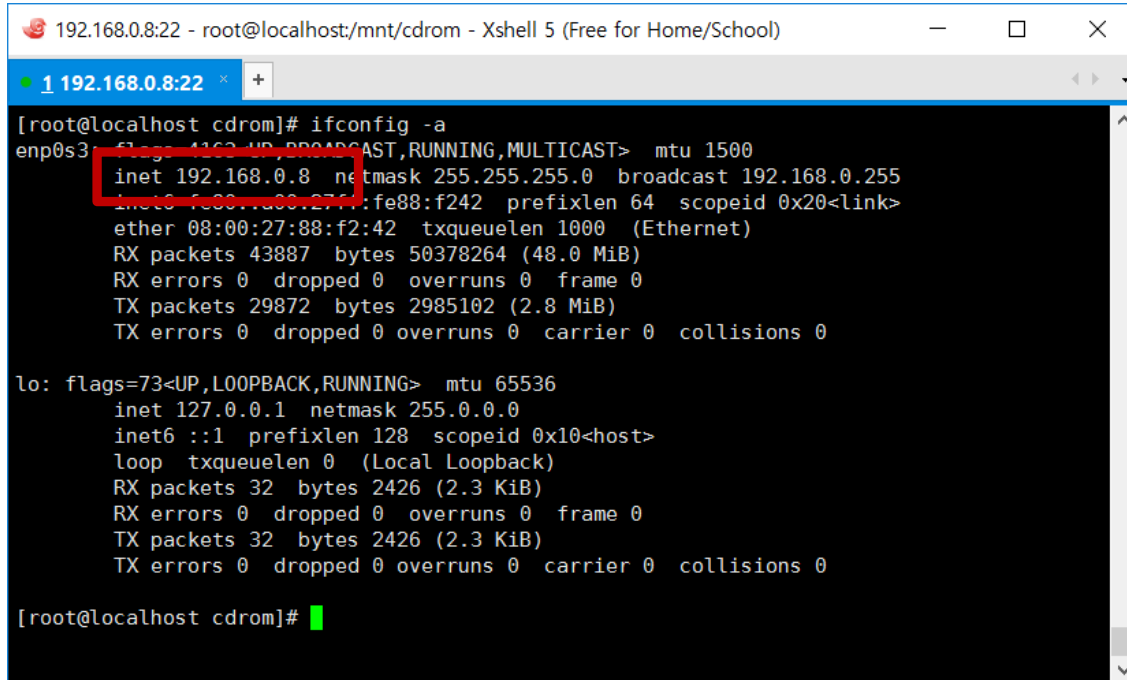
Linux Basic Setup

: SSH 서버 설정

- ▶ SSH 서버 IP를 확인합니다

```
$ ifconfig -a
```

- ▶ network interface의 상태를 확인합니다



```
192.168.0.8:22 - root@localhost:/mnt/cdrom - Xshell 5 (Free for Home/School)
1 192.168.0.8:22 * +
[root@localhost cdrom]# ifconfig -a
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.8 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::f277:fe88:f242 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:88:f2:42 txqueuelen 1000 (Ethernet)
    RX packets 43887 bytes 50378264 (48.0 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 29872 bytes 2985102 (2.8 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

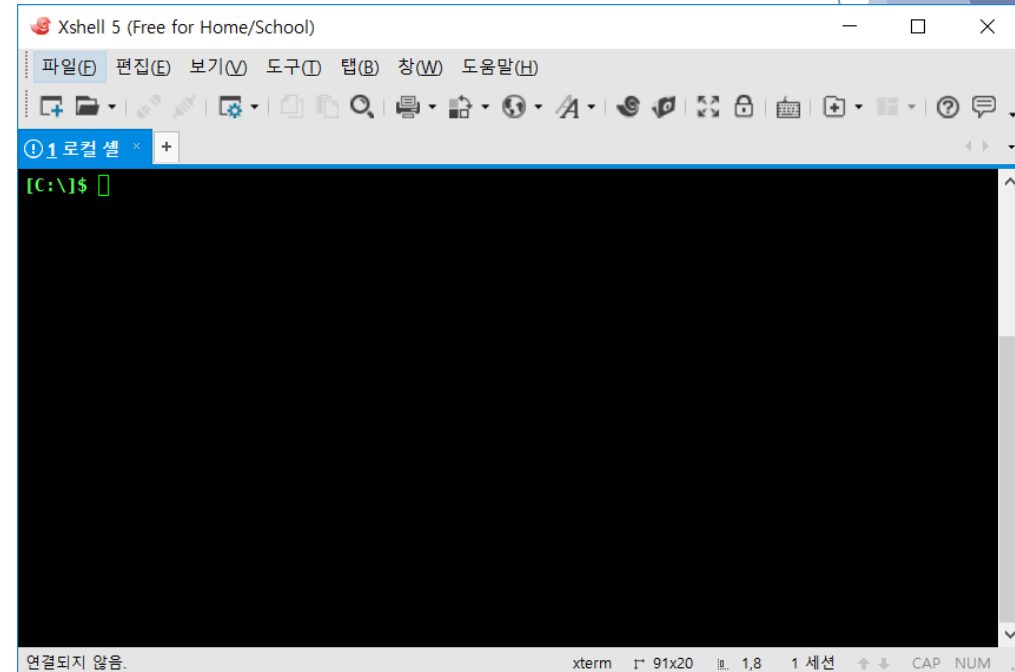
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 32 bytes 2426 (2.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 32 bytes 2426 (2.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost cdrom]#
```

Linux Basic Setup

: SSH 클라이언트 접속

- ▶ SSH 클라이언트 프로그램 설치
 - ▶ 클라이언트 운영체제에 맞는 SSH 클라이언트 프로그램을 구해서 설치
 - ▶ 거의 모든 운영체제 (Unix, Linux, OSX, Windows)를 지원하는 다수의 SSH 클라이언트가 있음
 - ▶ Windows의 경우 Putty 또는 Xshell을 많이 사용



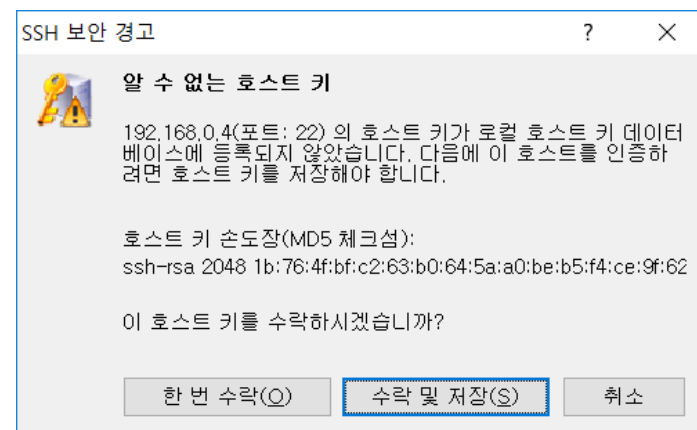
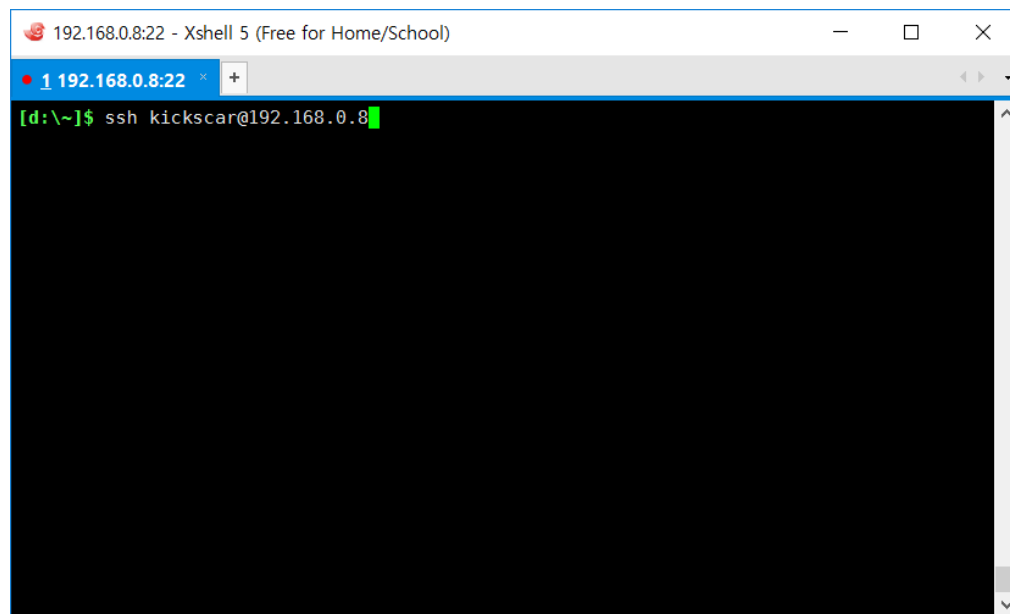
Linux Basic Setup

: SSH 클라이언트 접속

▶ SSH 서버에 접속하기

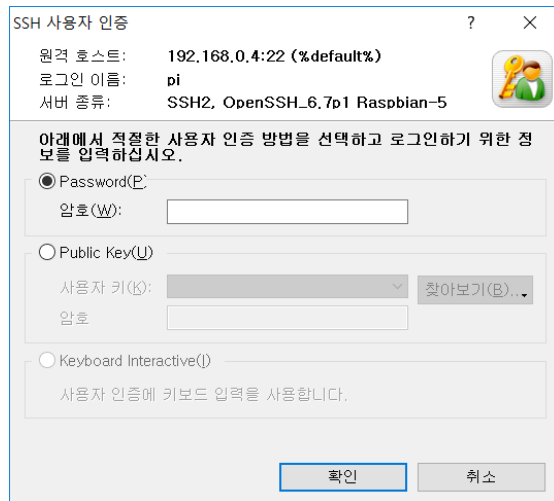
```
[C:\]$ ssh myaccount@192.168.0.4
```

▶ 접속 계정과 SSH 서버가 실행중인 시스템의 IP를 @뒤에 적어주고 엔터를 눌러 연결

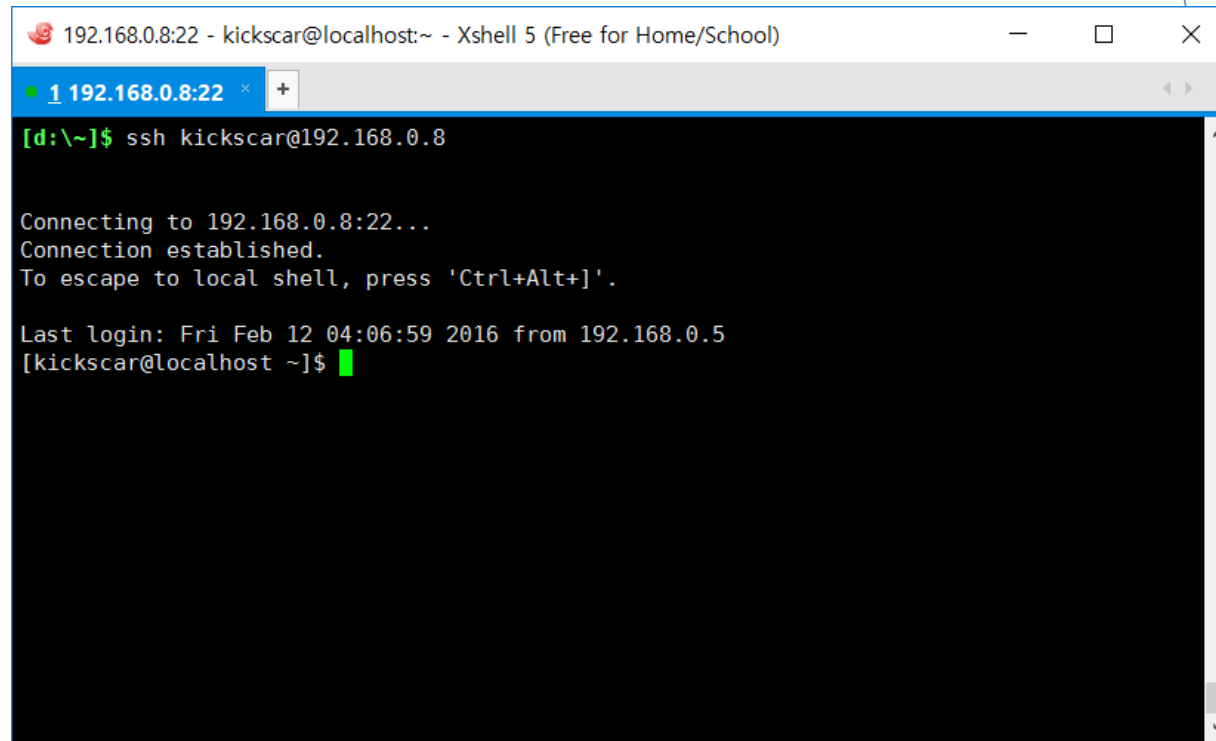


Linux Basic Setup

: SSH 클라이언트 접속



접속 계정의 암호 입력



접속 성공 화면