

# 2025 Operating System

## Lab 0. Lab overview & Linux on VM

2025.03.17

T.A. 김보승

[bskim1102@dankook.ac.kr](mailto:bskim1102@dankook.ac.kr)

# Contact

- T.A. 김보승
- 연락처: [bskim1102@dankook.ac.kr](mailto:bskim1102@dankook.ac.kr)
- 연구실: 소프트웨어 ICT관 515호

# Overview

## ■ OS의 3가지 요소

- Virtualization
- Concurrency
- Persistence

## Operating Systems: Three Easy Pieces

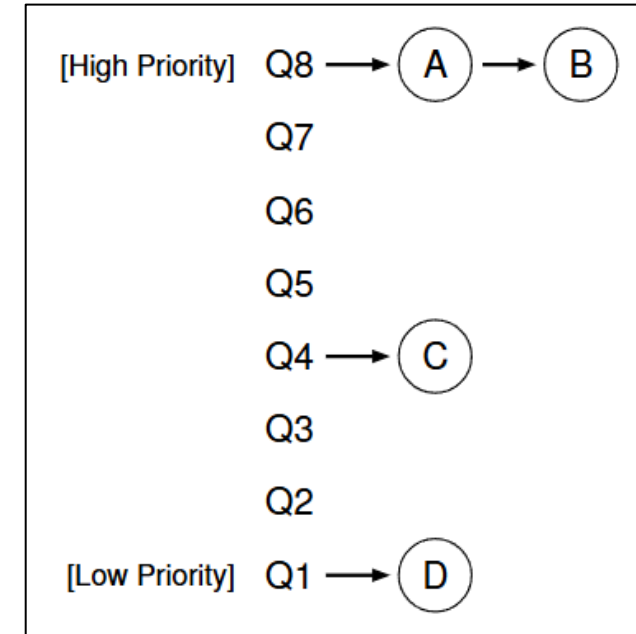
Intro	Virtualization		Concurrency	Persistence	Security
<a href="#">Preface</a>	3 <a href="#">Dialogue</a>	12 <a href="#">Dialogue</a>	25 <a href="#">Dialogue</a>	35 <a href="#">Dialogue</a>	52 <a href="#">Dialogue</a>
<a href="#">TOC</a>	4 <a href="#">Processes</a>	13 <a href="#">Address Spaces</a> <small>code</small>	26 <a href="#">Concurrency and Threads</a> <small>code</small>	36 <a href="#">I/O Devices</a>	53 <a href="#">Intro Security</a>
1 <a href="#">Dialogue</a>	5 <a href="#">Process API</a> <small>code</small>	14 <a href="#">Memory API</a>	27 <a href="#">Thread API</a> <small>code</small>	37 <a href="#">Hard Disk Drives</a>	54 <a href="#">Authentication</a>
2 <a href="#">Introduction</a> <small>code</small>	6 <a href="#">Direct Execution</a>	15 <a href="#">Address Translation</a>	28 <a href="#">Locks</a> <small>code</small>	38 <a href="#">Redundant Disk Arrays (RAID)</a>	55 <a href="#">Access Control</a>
	7 <a href="#">CPU Scheduling</a>	16 <a href="#">Segmentation</a>	29 <a href="#">Locked Data Structures</a>	39 <a href="#">Files and Directories</a>	56 <a href="#">Cryptography</a>
	8 <a href="#">Multi-level Feedback</a>	17 <a href="#">Free Space Management</a>	30 <a href="#">Condition Variables</a> <small>code</small>	40 <a href="#">File System Implementation</a>	57 <a href="#">Distributed</a>
	9 <a href="#">Lottery Scheduling</a> <small>code</small>	18 <a href="#">Introduction to Paging</a>	31 <a href="#">Semaphores</a> <small>code</small>	41 <a href="#">Fast File System (FFS)</a>	
	10 <a href="#">Multi-CPU Scheduling</a>	19 <a href="#">Translation Lookaside Buffers</a>	32 <a href="#">Concurrency Bugs</a>	42 <a href="#">FSCK and Journaling</a>	<b>Appendices</b>
	11 <a href="#">Summary</a>	20 <a href="#">Advanced Page Tables</a>	33 <a href="#">Event-based Concurrency</a>	43 <a href="#">Log-structured File System (LFS)</a>	<a href="#">Dialogue</a>
		21 <a href="#">Swapping: Mechanisms</a>	34 <a href="#">Summary</a>	44 <a href="#">Flash-based SSDs</a>	<a href="#">Virtual Machines</a>
		22 <a href="#">Swapping: Policies</a>		45 <a href="#">Data Integrity and Protection</a>	<a href="#">Dialogue</a>
		23 <a href="#">Complete VM Systems</a>		46 <a href="#">Summary</a>	<a href="#">Monitors</a>
		24 <a href="#">Summary</a>		47 <a href="#">Dialogue</a>	<a href="#">Dialogue</a>
				48 <a href="#">Distributed Systems</a>	<a href="#">Lab Tutorial</a>
				49 <a href="#">Network File System (NFS)</a>	<a href="#">Systems Labs</a>
				50 <a href="#">Andrew File System (AFS)</a>	<a href="#">xv6 Labs</a>
				51 <a href="#">Summary</a>	

## ■ 각 요소에 대한 실습 과제가 나갈 예정

# Lab 1: Virtualization

## ■ CPU Scheduling

- 아래 4가지 CPU Scheduler Simulator 구현
  - Round-Robin(RR)
  - Multi Level Feedback Queue(MLFQ)
  - Lottery
  - Stride



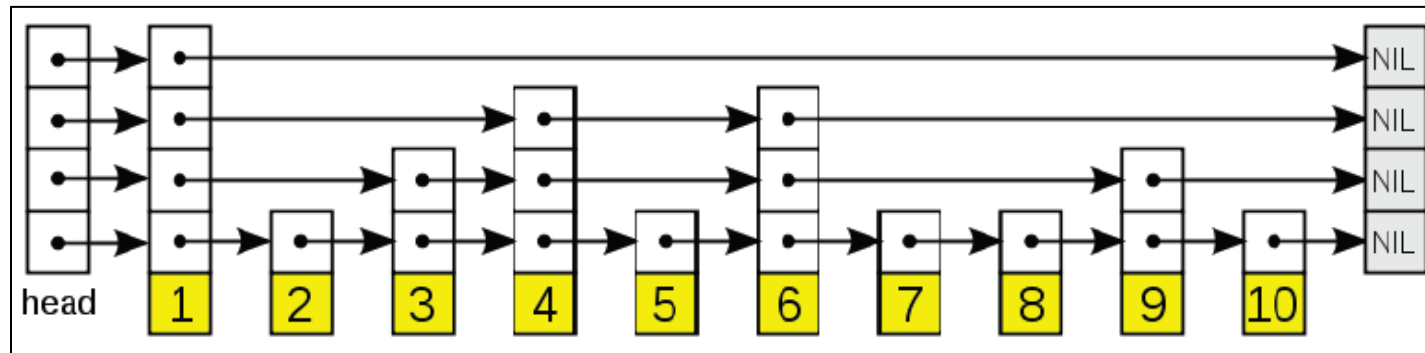
MLFQ Scheduling 예시

- Context Switch Time, 워크로드에 따른 scheduling 결과 분석을 목표

# Lab 2: Concurrency

## ■ Concurrent Data Structure

- Skip-list를 아래 3가지 버전으로 구현
  - Without lock
  - Coarse-grained lock
  - Fine-grained lock
- Critical Section에 대한 이해 및 Lock 사용에 따른 성능 변화 분석을 목표

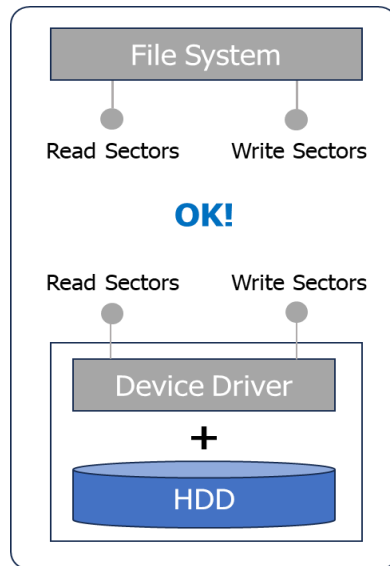


Skip-list 자료구조

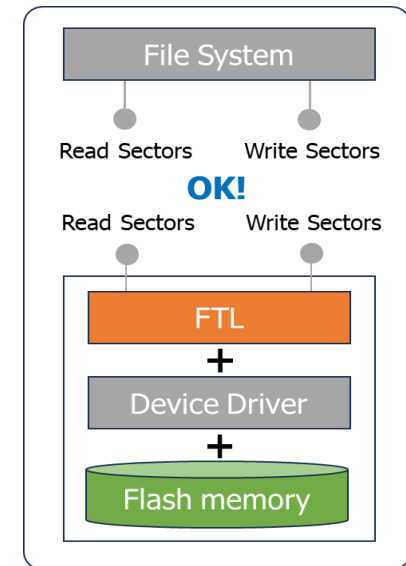
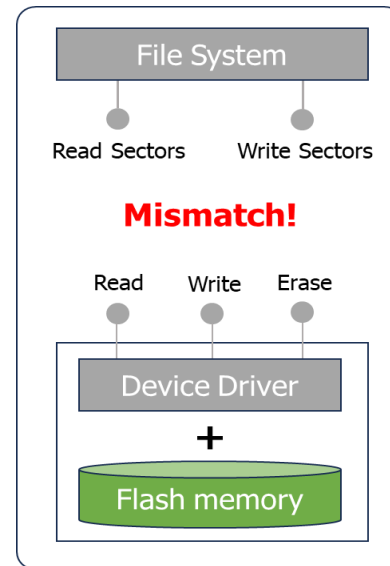
# Lab 3: Persistence

## ■ Flash Translation Layer (FTL)

- 기존 파일 시스템은 HDD 기반으로 설계되어 SSD 인터페이스와 호환되지 않음
- 이를 해결하기 위해 FTL 계층을 추가하여 기존 인터페이스와의 호환성을 확보



파일 시스템과 HDD 간 데이터 전송 방식

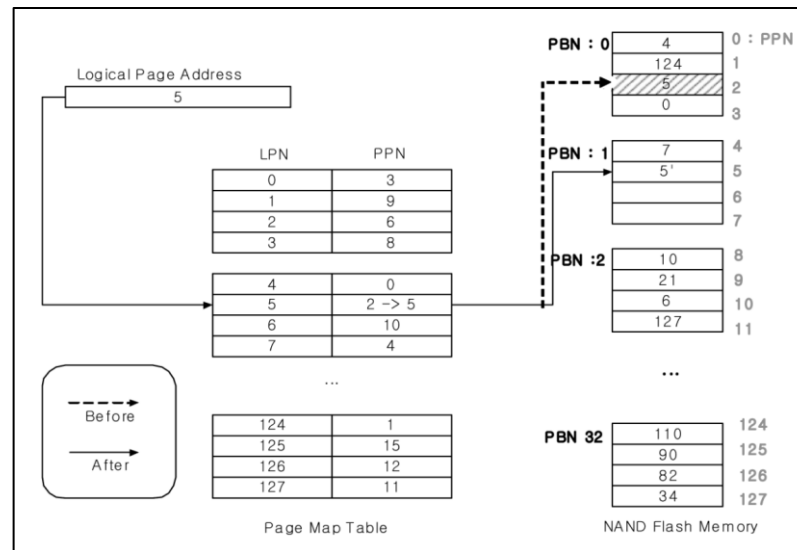


파일 시스템과 SSD간 데이터 전송 방식

# Lab 3: Persistence

## ■ Flash Translation Layer (FTL)

- FTL을 page-mapping 방식으로 구현
  - Mapping table 및 Flash memory status
- SSD 동작 방식 및 GC(Garbage Collection)에 대한 이해를 목표



LPN: Logical Page Number  
PPN: Physical Page Number  
PBN: Physical Block Number

Page-mapping scheme

# Submission Guide

## ■ 공통 과제 설명

### - 보고서 양식

- 정해주는 제목 양식에 맞춰 **pdf 형식**으로 제출, ex) os\_lab0\_학번\_이름.pdf
- 코드 및 터미널 화면 첨부 시 **흰색 바탕화면** 권장

### - 코드 양식

- 코드 상단에 **작성자 정보** 기입, ex) 이름, 학번, 날짜, 내용
- 코드 설명하는 **주석** 달기
- **make** 명령어로 컴파일이 되고 정상적으로 실행이 되어야 함
- C++로 작성

### - 제출 방식

- **구글 폼**으로 제출

## ■ 추후에 과제가 나갈 때마다 채점기준 및 보고서 구성에 대한 자세한 설명 예정



# VirtualBox 설치

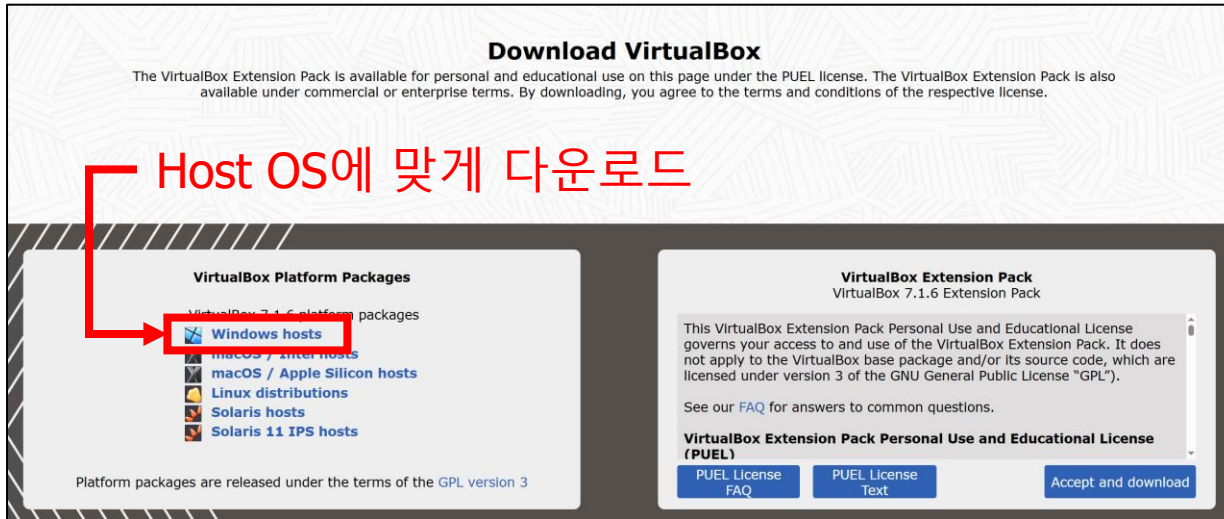
## ■ VirtualBox란?

- 오라클에서 개발한 오픈 소스 가상화 소프트웨어
- 하나의 운영 체제(OS)에서 다른 운영 체제를 가상 환경으로 실행할 수 있음
  - 강의에서는 Guest OS로 Linux Ubuntu를 사용
- Windows, macOS, Linux 등 다양한 운영 체제에서 사용 가능

# 1. Windows

## 1) VirtualBox 설치

- <https://www.virtualbox.org>



VirtualBox 다운로드 페이지

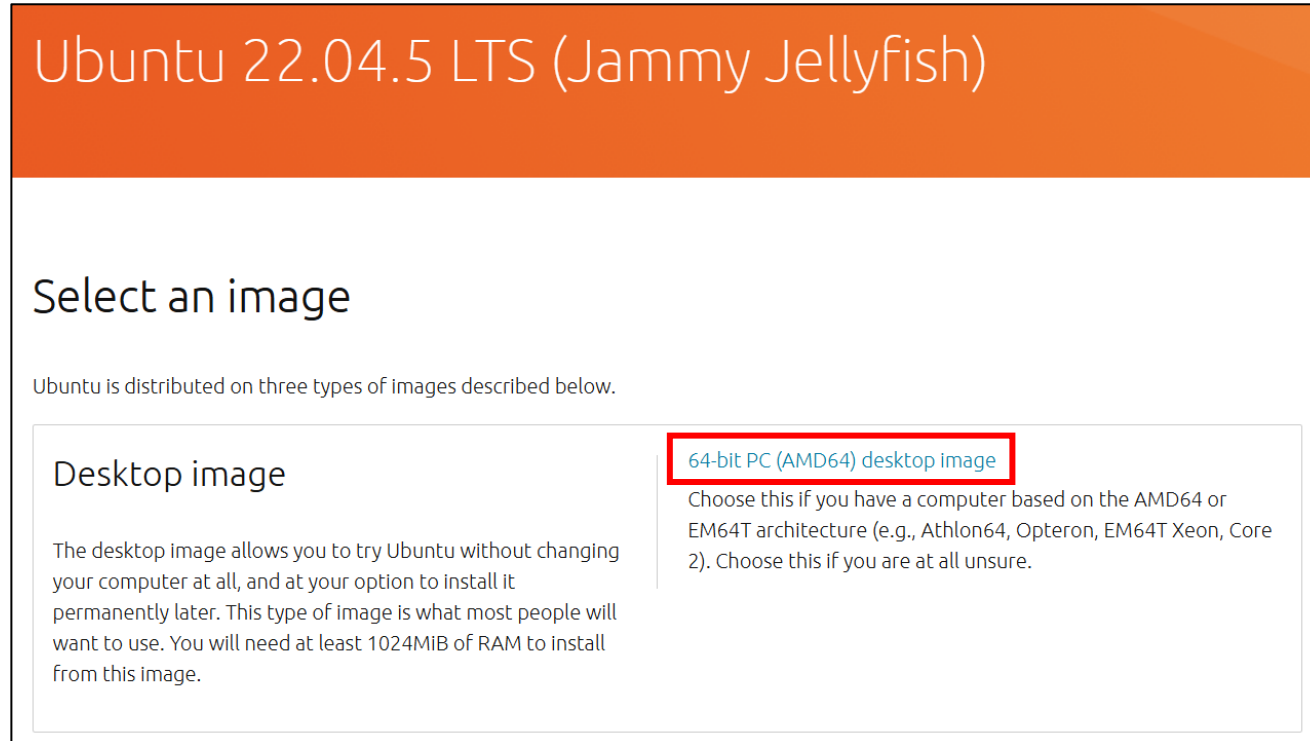


설치 후 VirtualBox 초기화면

# 1. Windows

## 2) Ubuntu Desktop 22.04 이미지 다운로드 (AMD64)

- <https://releases.ubuntu.com/jammy/>



# 1. Windows

## 3) VM 생성 – VM 이름 및 ISO 이미지 선택



# 1. Windows

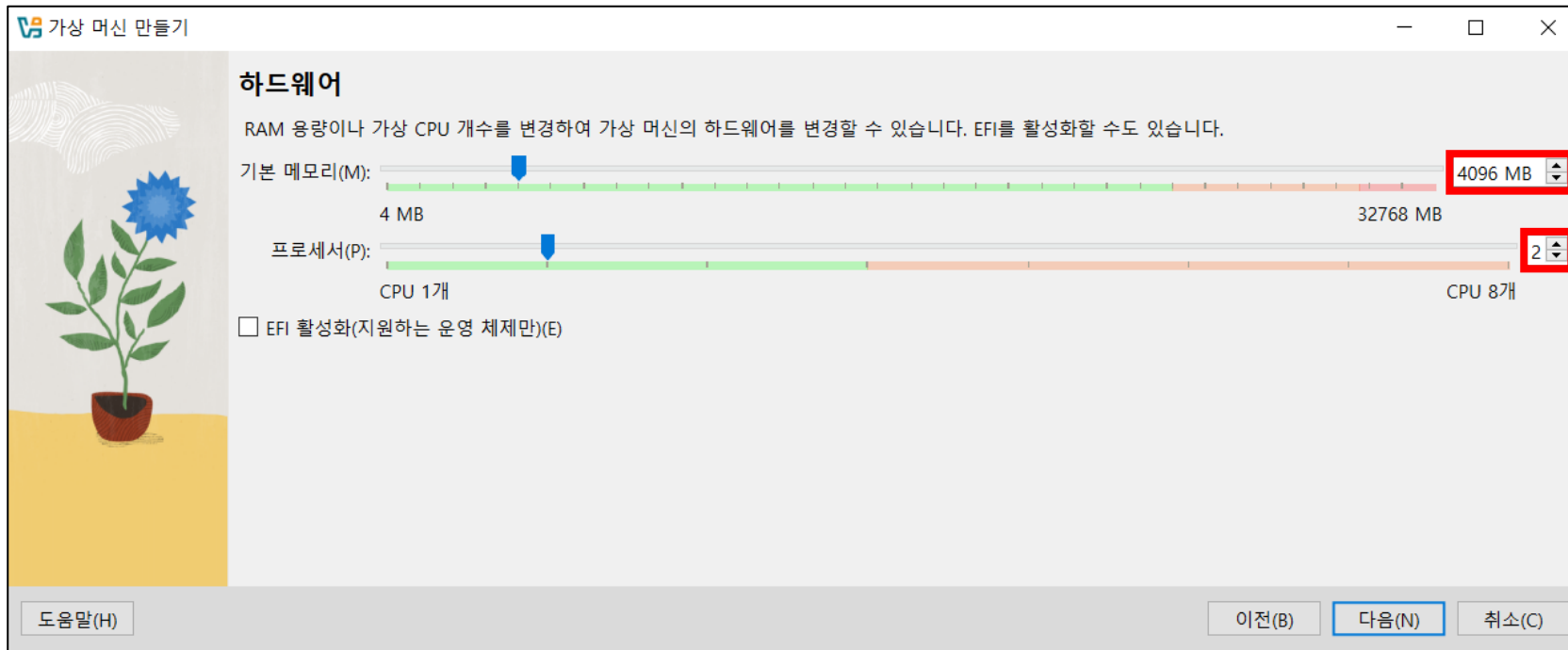
## 3) VM 생성 -계정 생성

- 사용자 이름: **user + 학번으로 설정**

# 1. Windows

## 3) VM 생성 – 하드웨어 사양 선택

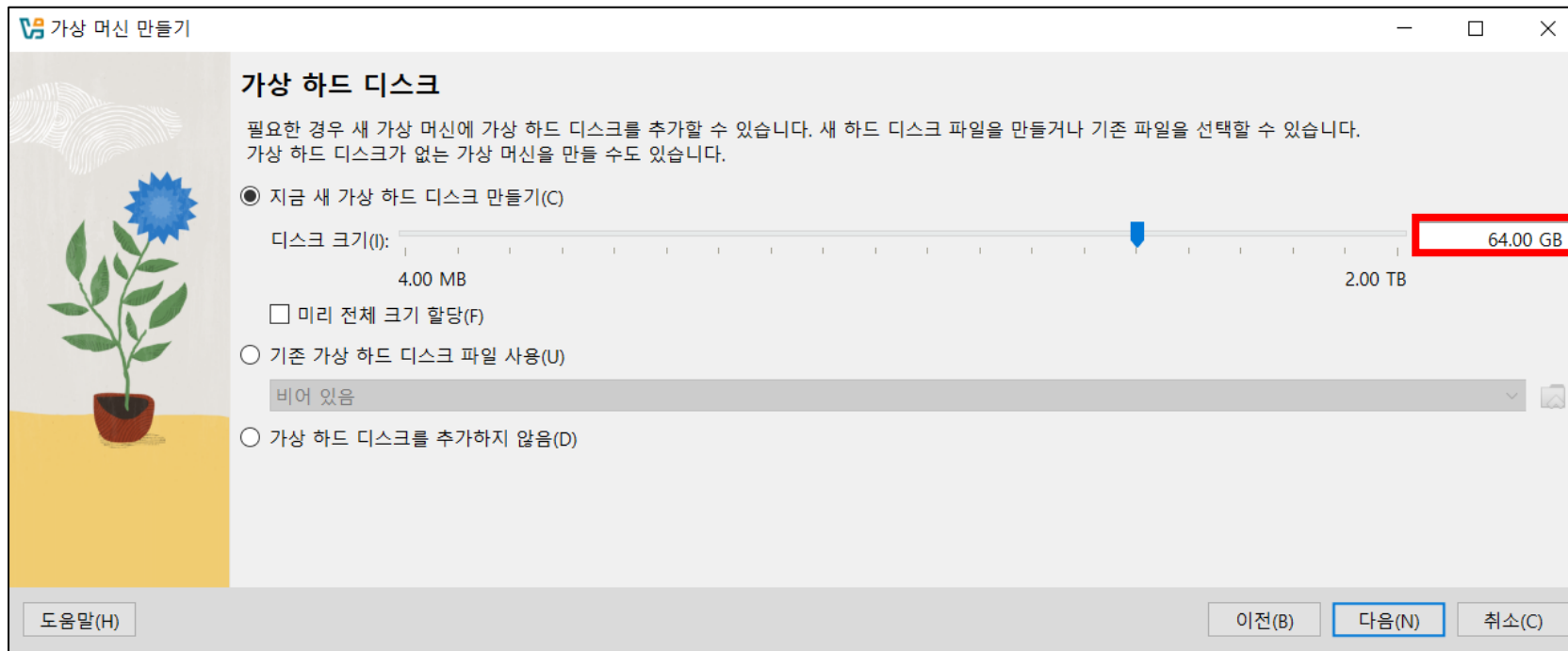
- 메모리: 4096MB, 프로세서: 2개
- 본인 컴퓨터 사양에 맞춰 설정 가능



# 1. Windows

## 3) VM 생성 – 하드웨어 사양 선택

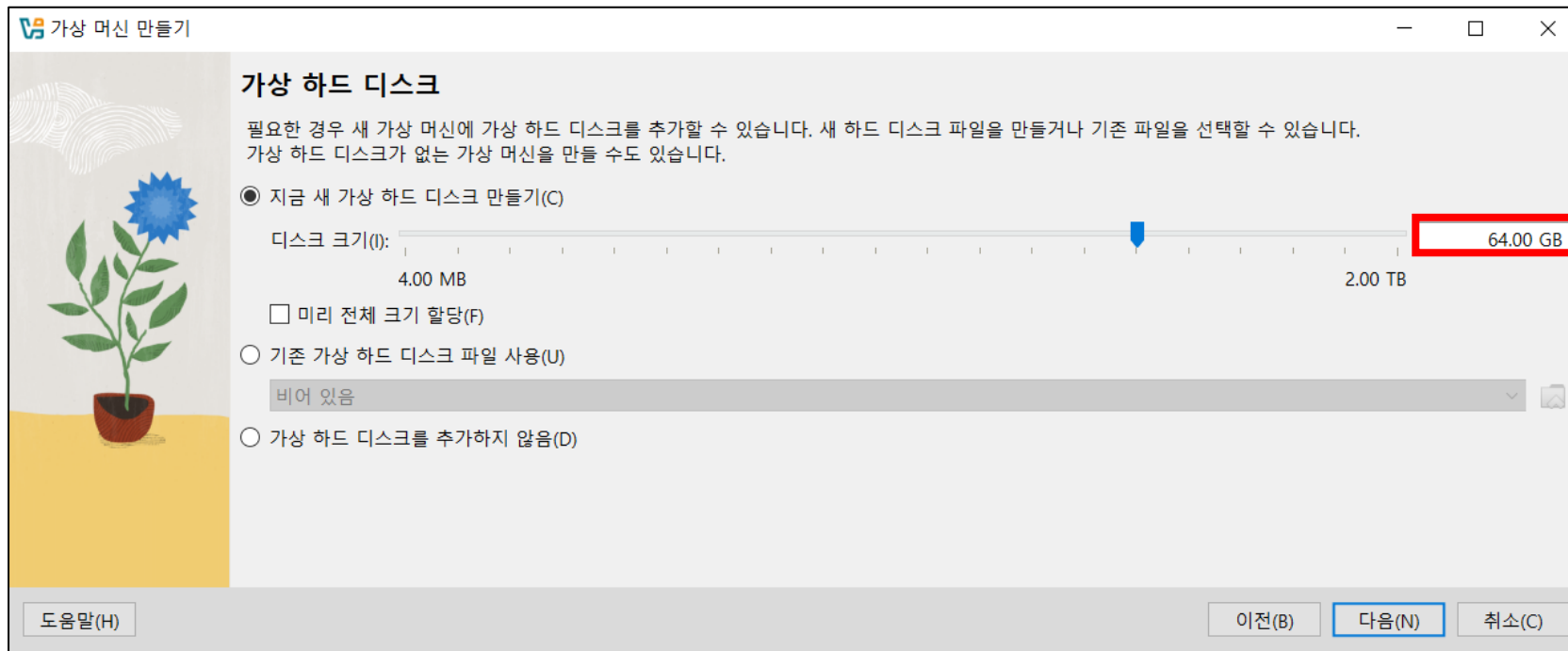
- 디스크 크기: 64GB
- 여유공간이 부족하다면 작게 설정



# 1. Windows

## 3) VM 생성 – 하드웨어 사양 선택

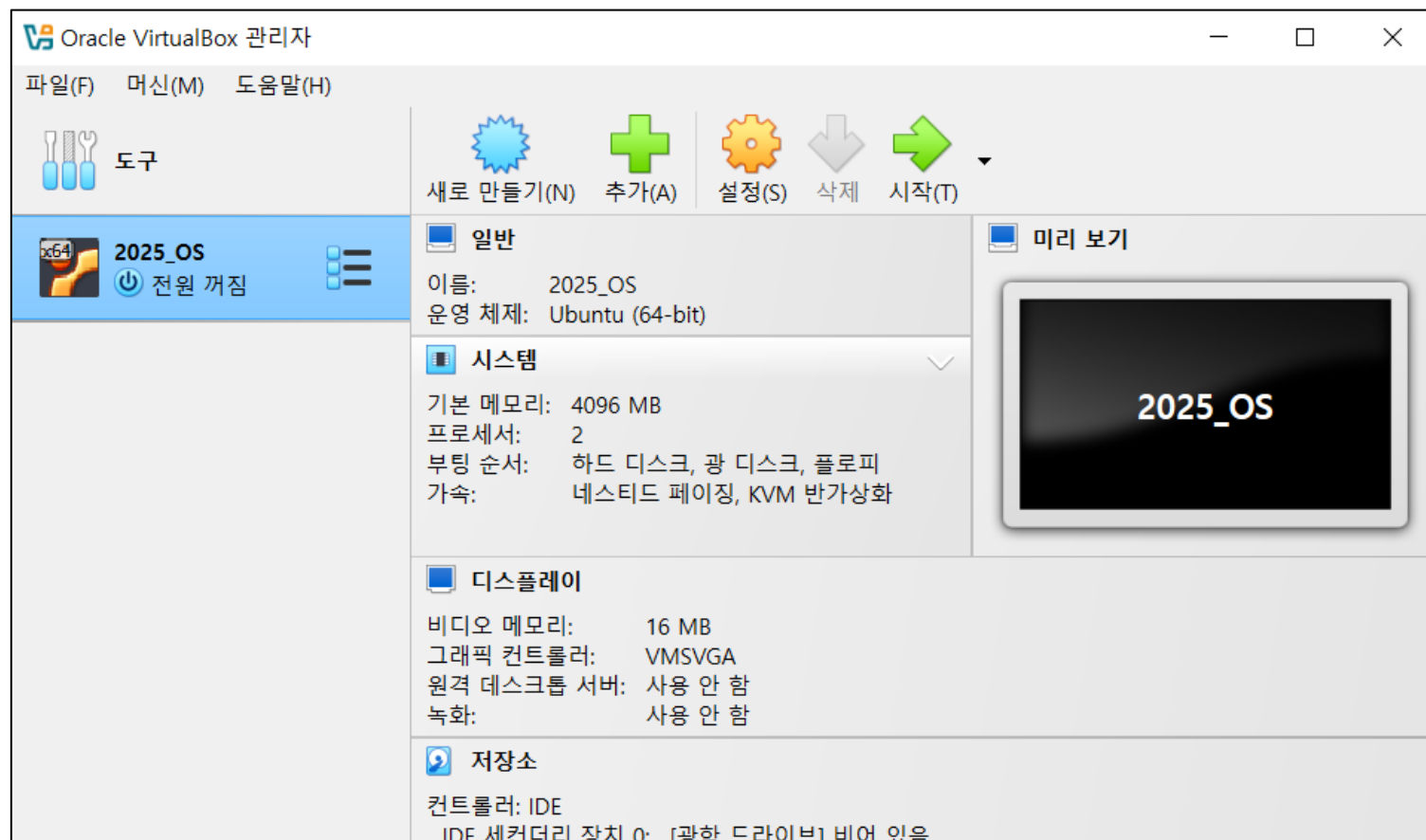
- 디스크 크기: 64GB
- 여유공간이 부족하다면 작게 설정





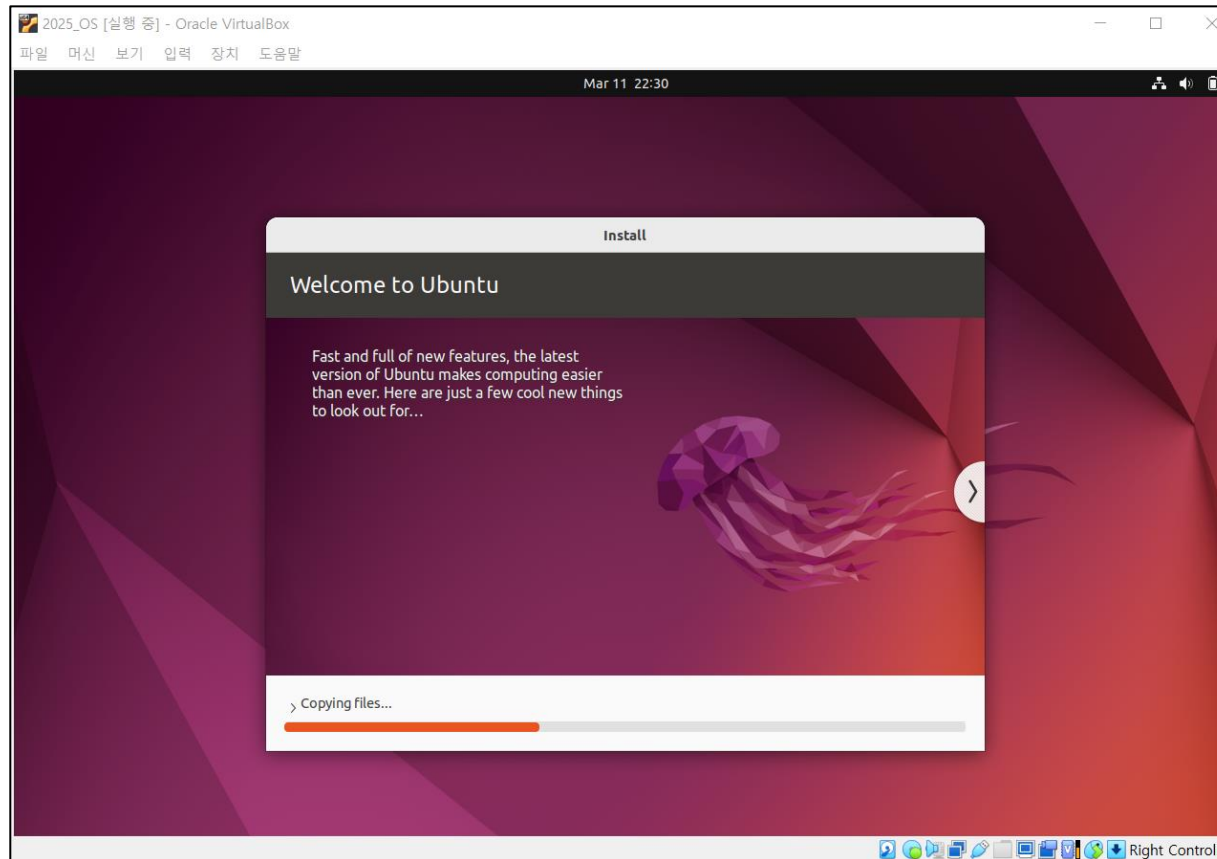
# 1. Windows

## 3) VM 생성 - 완료



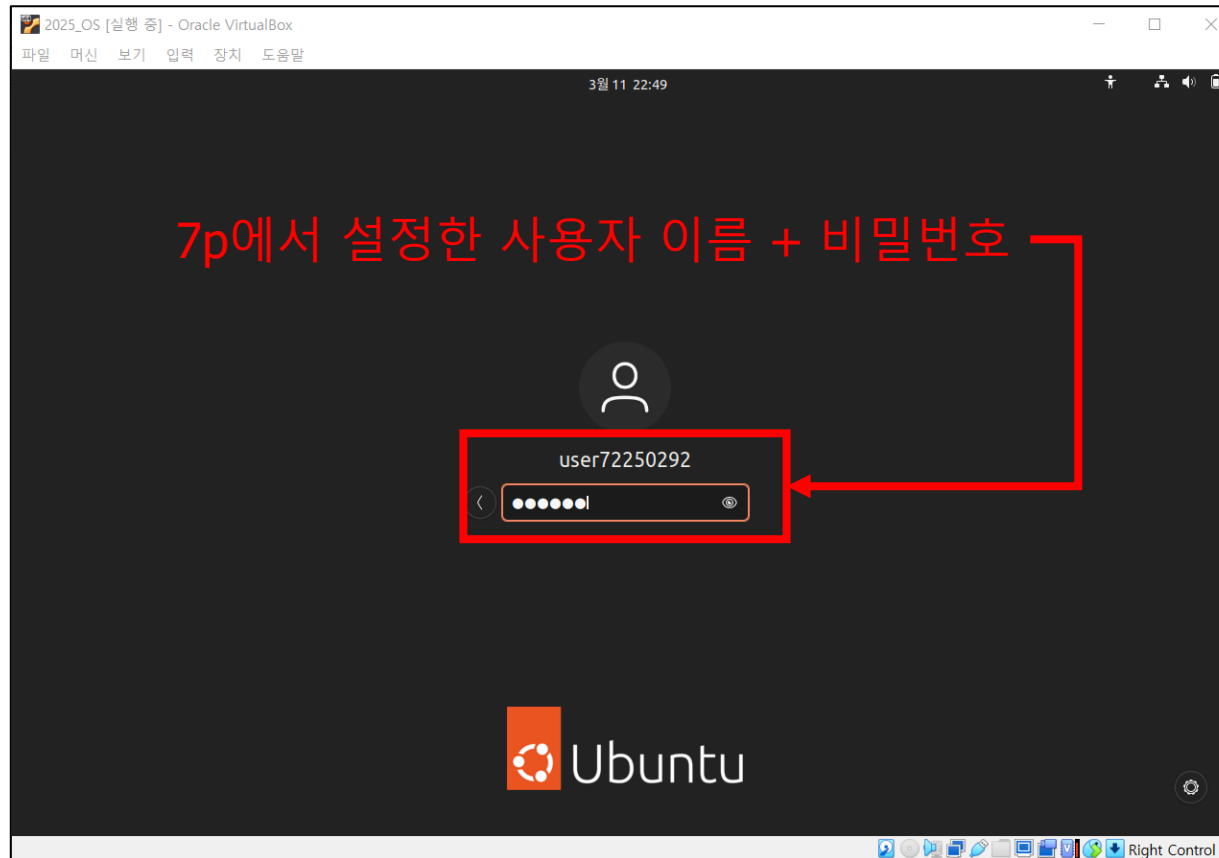
# 1. Windows

## 4) 초기 설정 – ubuntu 설치



# 1. Windows

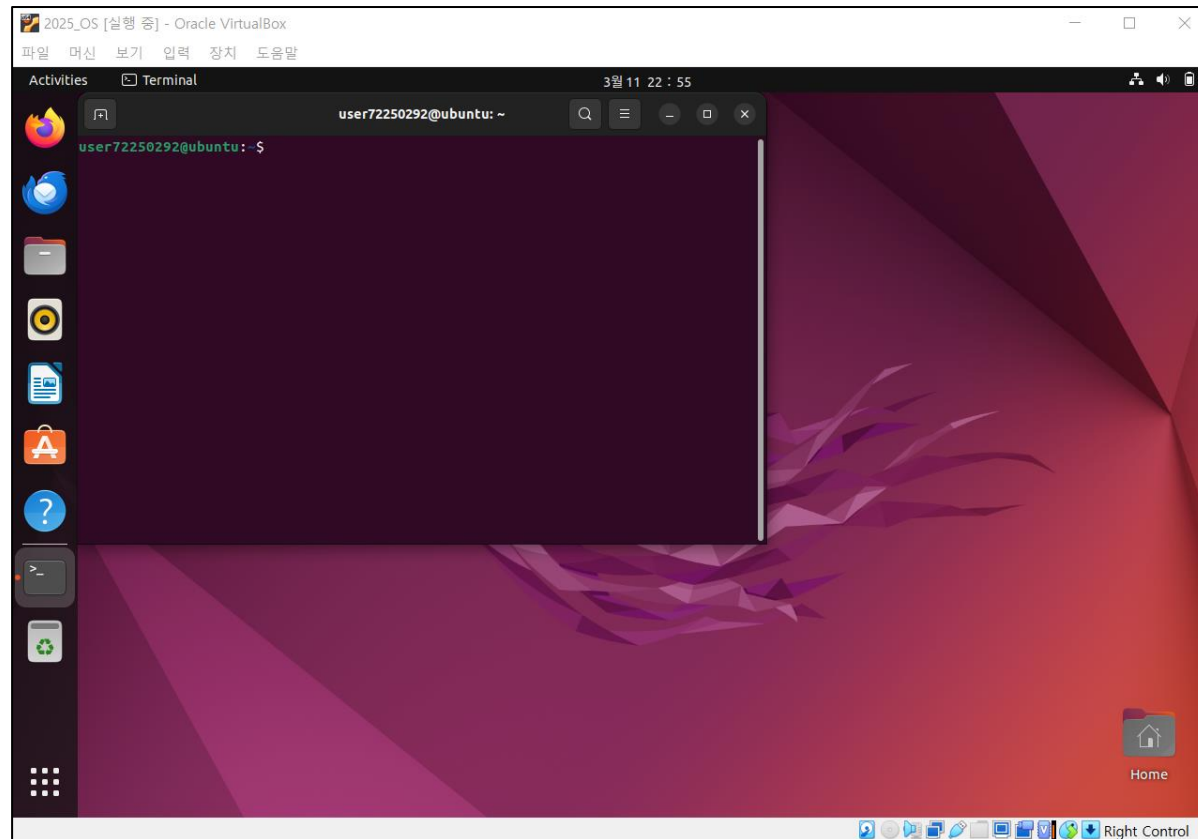
## 4) 초기 설정 - 계정 확인 및 로그인



# 1. Windows

## 4) 초기 설정 - 터미널 열기

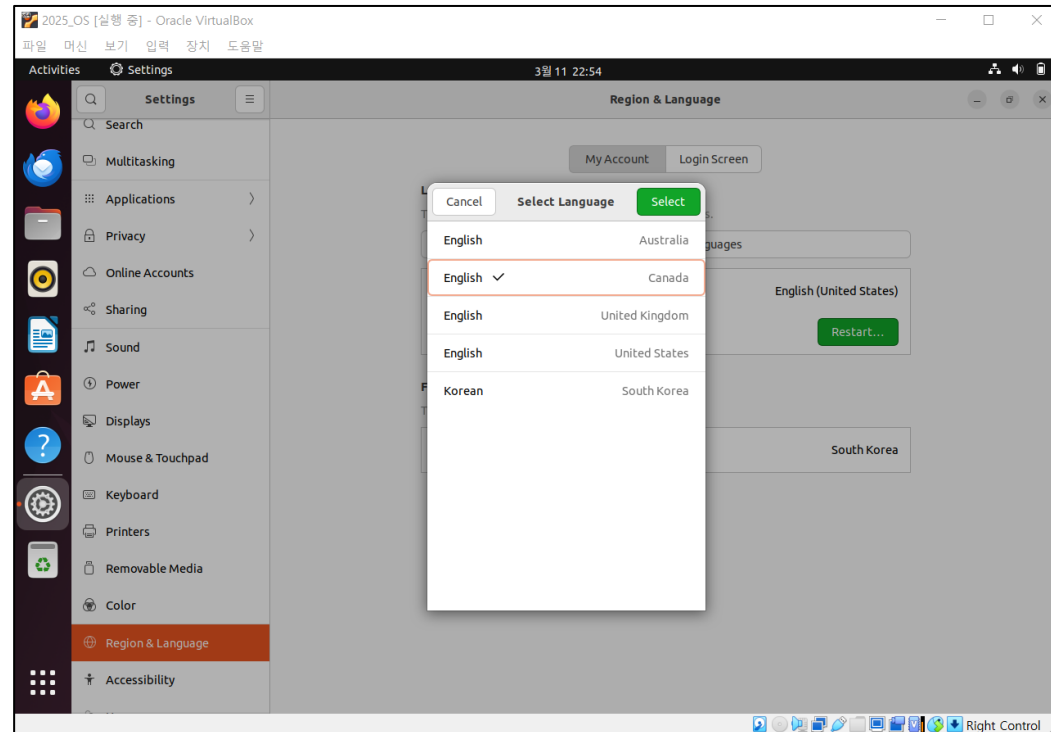
- 터미널 단축키: Ctrl + Alt + t



# 1. Windows

## 4) 초기 설정 – 터미널 열기

- 만약 터미널이 안 열린다면 Settings → Region & Language → Language
- English(United State)를 English(Canada)로 변경 후 restart



# 1. Windows

## 4) 초기 설정 – sudo 권한 설정

- sudo 명령어 실행 시 오류가 난다면 아래 과정 수행

```
user72250292@ubuntu:~$ sudo apt update
[sudo] password for user72250292:
user72250292 is not in the sudoers file. This incident will be reported.
```

- root로 전환 후 7p 비밀번호 입력, {username}을 sudo 그룹에 추가하여 권한 부여
  - \$ su
  - \$ usermod -aG sudo {username}
  - \$ reboot
- 재부팅후 sudo 명령어 작동 확인

```
user72250292@ubuntu:~$ sudo apt update
[sudo] password for user72250292:
Ign:1 http://kr.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://kr.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://kr.archive.ubuntu.com/ubuntu jammy-backports InRelease
```

# 1. Windows

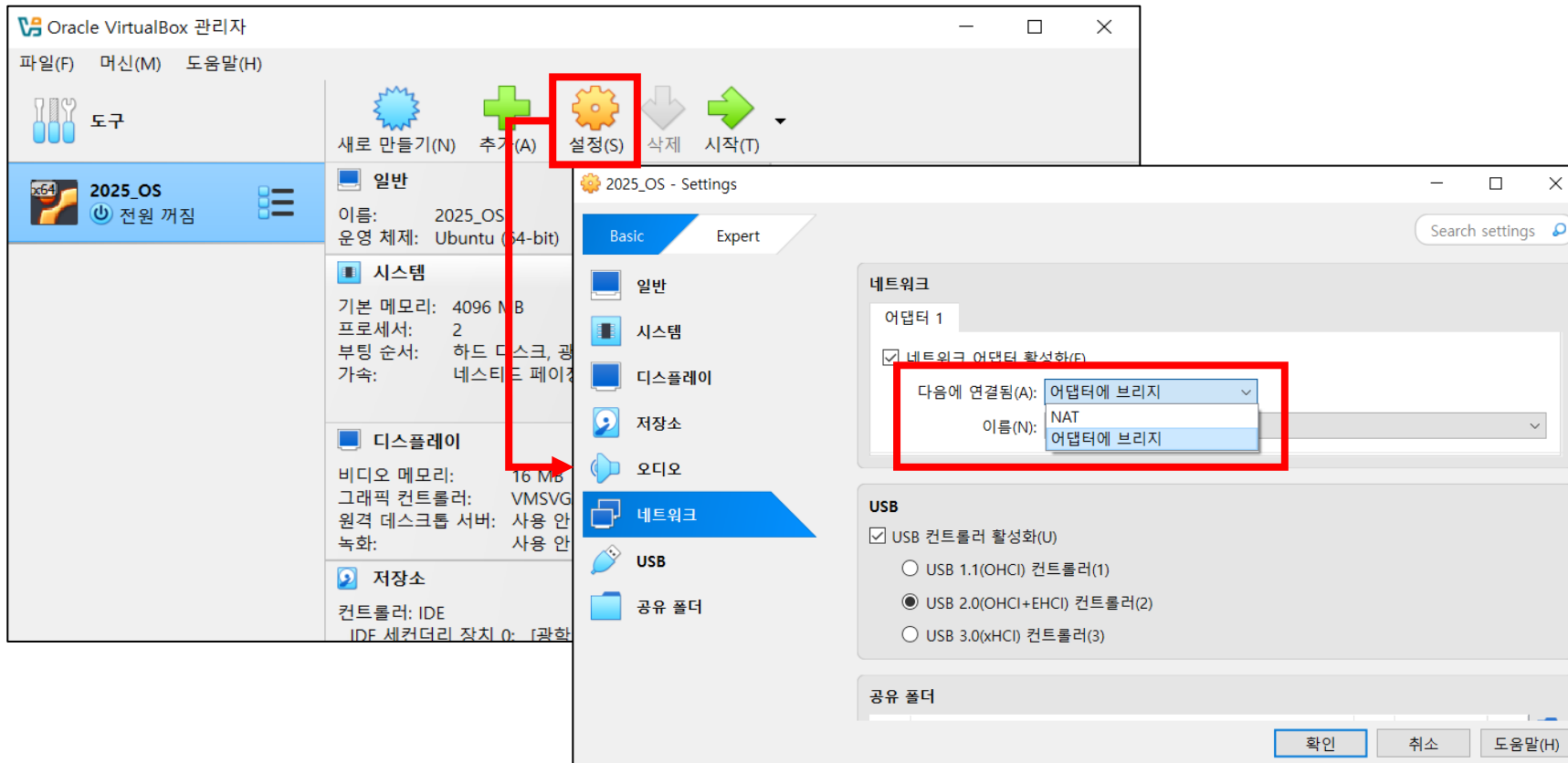
## 5) Optional – SSH 설정

- SSH: 다른 컴퓨터에 원격으로 접속하여 명령을 실행하고 정보를 주고받을 수 있는 프로토콜
- 호스트(윈도우)에서 VM(우분투)에 원격으로 접속해 터미널 사용가능

# 1. Windows

## 5) Optional – SSH 설정

- VirtualBox 네트워크 → 어댑터에 브릿지 설정





# 1. Windows

## 5) Optional – SSH 설정

- VM 실행 후 아래 명령어 실행
- `$ sudo apt install -y openssh-server net-tools`
  - openssh-server: SSH 접속을 가능하게 해주는 패키지
  - net-tools: ip 정보 및 네트워크 관련 정보를 확인할 수 있는 패키지
- `$ systemctl status sshd`
  - SSH 서버가 정상적으로 동작 중임을 확인
- `$ ifconfig`
  - IP 주소 확인

```
user72250292@ubuntu:~$ systemctl status sshd
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled;
   Active: active (running) since Tue 2025-03-11 23:32:17 KS
   Docs: man:sshd(8)
         man:sshd_config(5)
```

```
user72250292@ubuntu:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
   inet 192.168.0.22 netmask 255.255.255.0 broadcast 192.168.0.255
   inet6 fe80::82b3:3427:b6c3:6741 prefixlen 64 scopeid 0x20<link>
   ether 08:00:27:de:b3:a2 txqueuelen 1000 (Ethernet)
```

# 1. Windows

## 5) Optional – SSH 설정

- Host(윈도우)에서 터미널 실행 (VM은 실행 중인 상태)
- > ssh {username}@{IP address}
  - IP address: 19p에서 확인한 IP (VM 재시작시 바뀔 수 있으니 확인 필요)

```
PS C:\Users\Boseung> ssh user72250292@192.168.0.22
The authenticity of host '192.168.0.22 (192.168.0.22)' can't be established.
ED25519 key fingerprint is SHA256:Rr9M0oNx9L8ZtHU53pnmLsRasyxfG3ur5VgypctWvLc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.22' (ED25519) to the list of known hosts.
user72250292@192.168.0.22's password: 
```

# UTM 설치

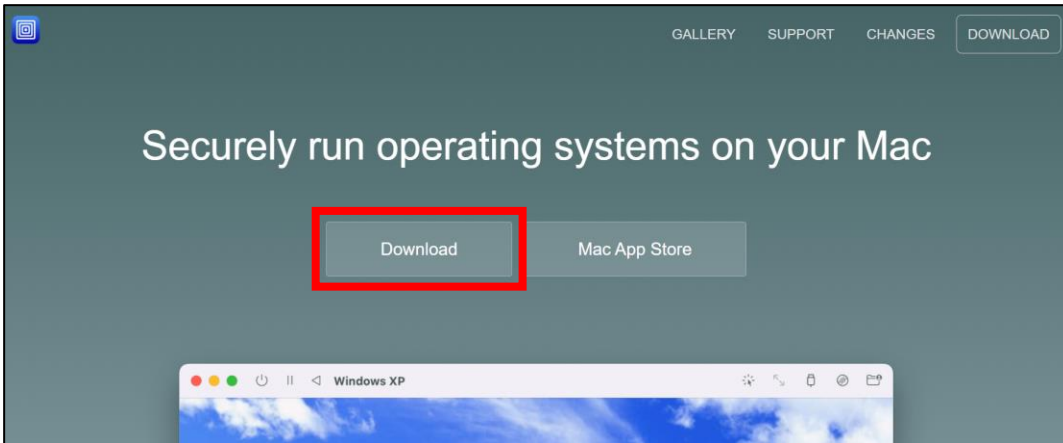
## ■ UTM이란?

- macOS에서 가상화를 지원하는 오픈 소스 가상화 소프트웨어
- Apple Silicon에서 다양한 운영 체제를 실행할 수 있음
- Windows, Linux 등 다양한 운영 체제를 가상 환경에서 실행 가능
  - 강의에서는 Guest OS로 Linux를 사용

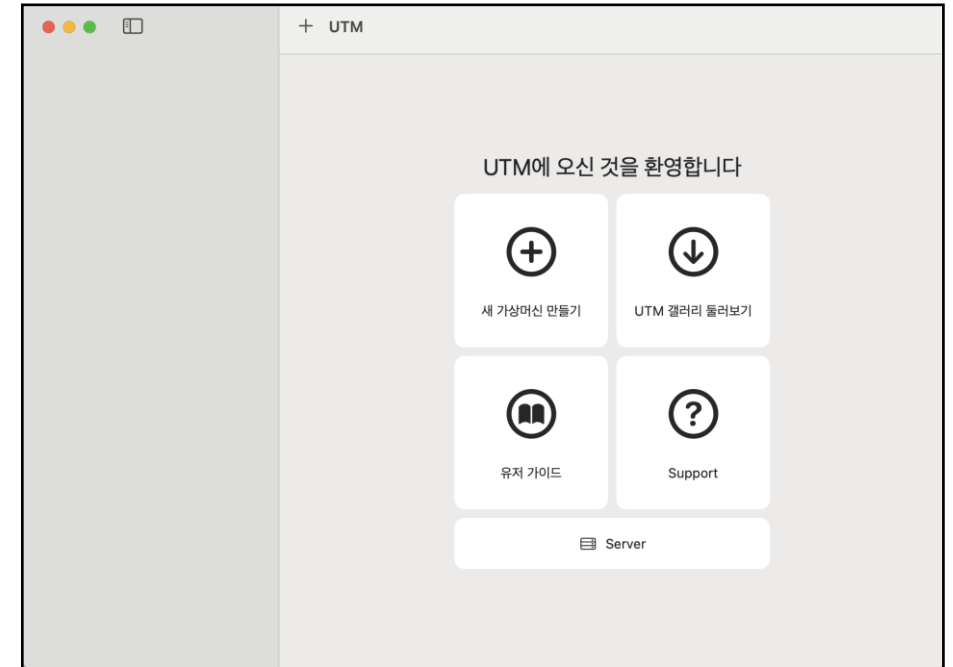
## 2. Mac (ARM)

### 1) UTM 설치

- <https://mac.getutm.app/>



UTM 다운로드 페이지

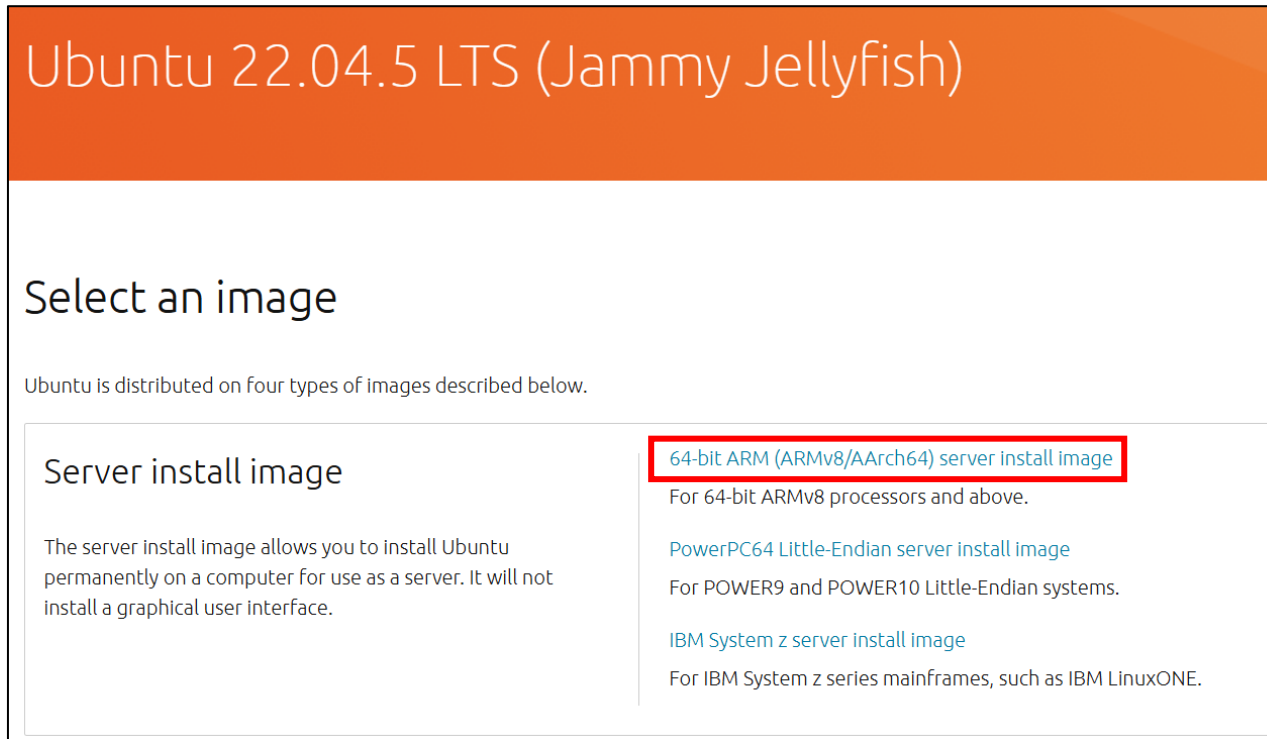


설치 후 UTM 초기화면

## 2. Mac (ARM)

### 2) Ubuntu Desktop 22.04 이미지 다운로드 (ARM)

- <https://cdimage.ubuntu.com/releases/jammy/release/>



Ubuntu 22.04.5 LTS (Jammy Jellyfish)

Select an image

Ubuntu is distributed on four types of Images described below.

**Server install image**

The server install image allows you to install Ubuntu permanently on a computer for use as a server. It will not install a graphical user interface.

**64-bit ARM (ARMv8/AArch64) server install image**

For 64-bit ARMv8 processors and above.

[PowerPC64 Little-Endian server install image](#)

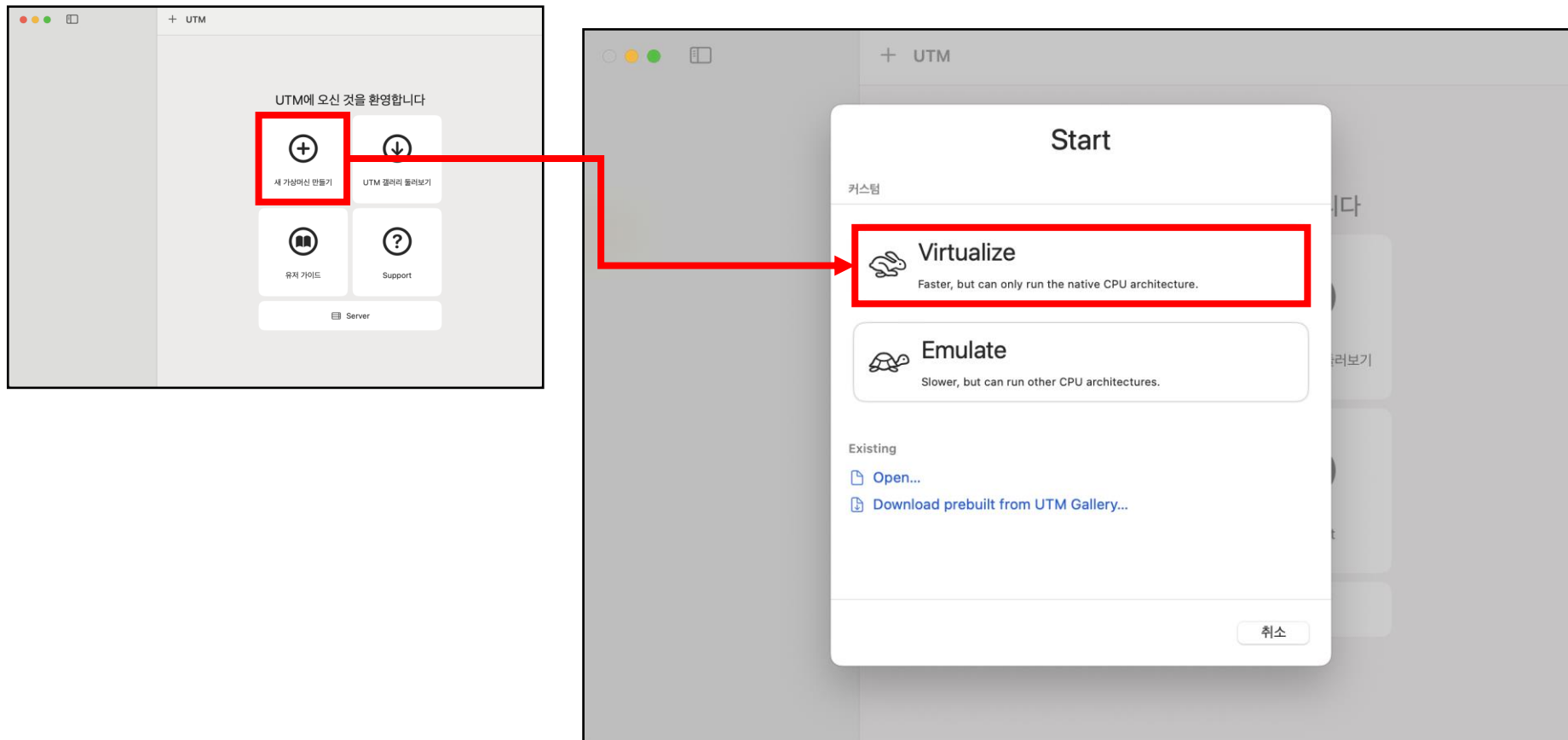
For POWER9 and POWER10 Little-Endian systems.

[IBM System z server install image](#)

For IBM System z series mainframes, such as IBM LinuxONE.

## 2. Mac (ARM)

### 3) VM 생성





## 2. Mac (ARM)


### 3) VM 생성

운영체제


Preconfigured

 macOS 12+

 Windows

 Linux

커스텀

 Other

취소 Go Back

Linux

Virtualization Engine

☐ Use Apple Virtualization  
Apple Virtualization is experimental and only for advanced use cases. Leave unchecked to use QEMU, which is recommended.

Boot Image Type

☐ Boot from kernel image  
[Ubuntu Install Guide](#)

Boot ISO Image

jammy-desktop-arm64.iso

초기화 탐색

취소 Go Back Continue

2)에서 다운받은 iso 이미지

## 2. Mac (ARM)

### 3) VM 생성

- 메모리: 4096MB, Core: 2개, 디스크: 64GB
- 본인 컴퓨터 사양에 맞춰 설정 가능

본인이 식별하기 쉬운 이름으로 설정

장치

메모리

4096 MiB

CPU

CPU Cores

2

Hardware OpenGL Acceleration

☐ Enable hardware OpenGL acceleration

There are known issues in some newer Linux drivers including black screen, broken compositing, and apps failing to render.

취소 Go Back Continue

Storage

크기

Specify the size of the drive where data will be stored into.

64 GiB

취소 Go Back Continue

Summary

이름 Ubuntu

☐ Open VM Settings

Engine QEMU

☒ Use Virtualization

☐ Legacy Hardware

아키텍처 ARM64 (aarch64)

시스템 QEMU 7.2 ARM Virtual Machine (alias of virt-7.2) (v

RAM 4GB

CPU 2 Cores

Storage 64GB

☐ Hardware OpenGL Acceleration

운영체제 Linux

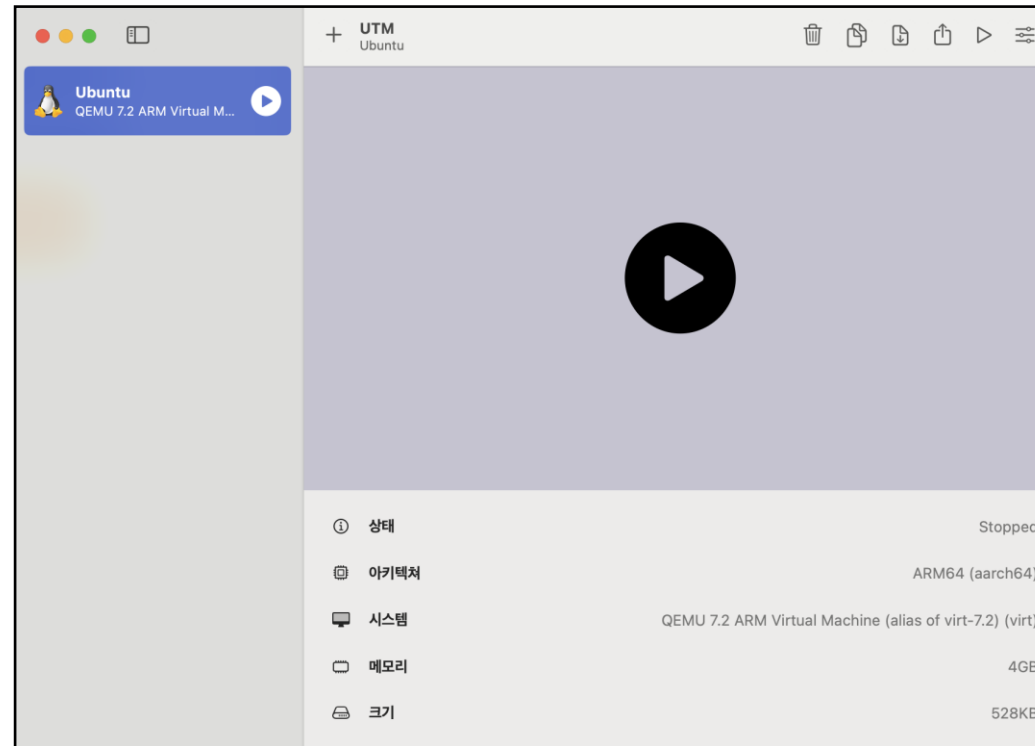
Boot Image /Users/nk/Downloads/jammy-desktop-arm64.iso

취소 Go Back 저장



## 2. Mac (ARM)

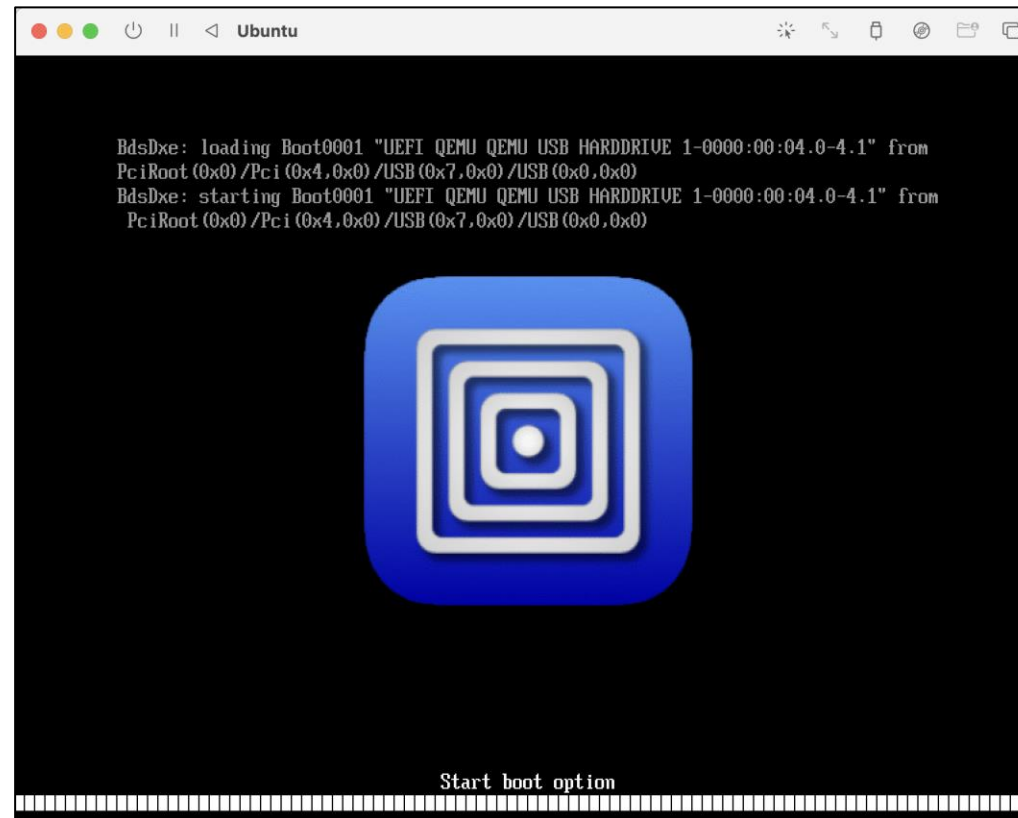
### 3) VM 생성 - 완료



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

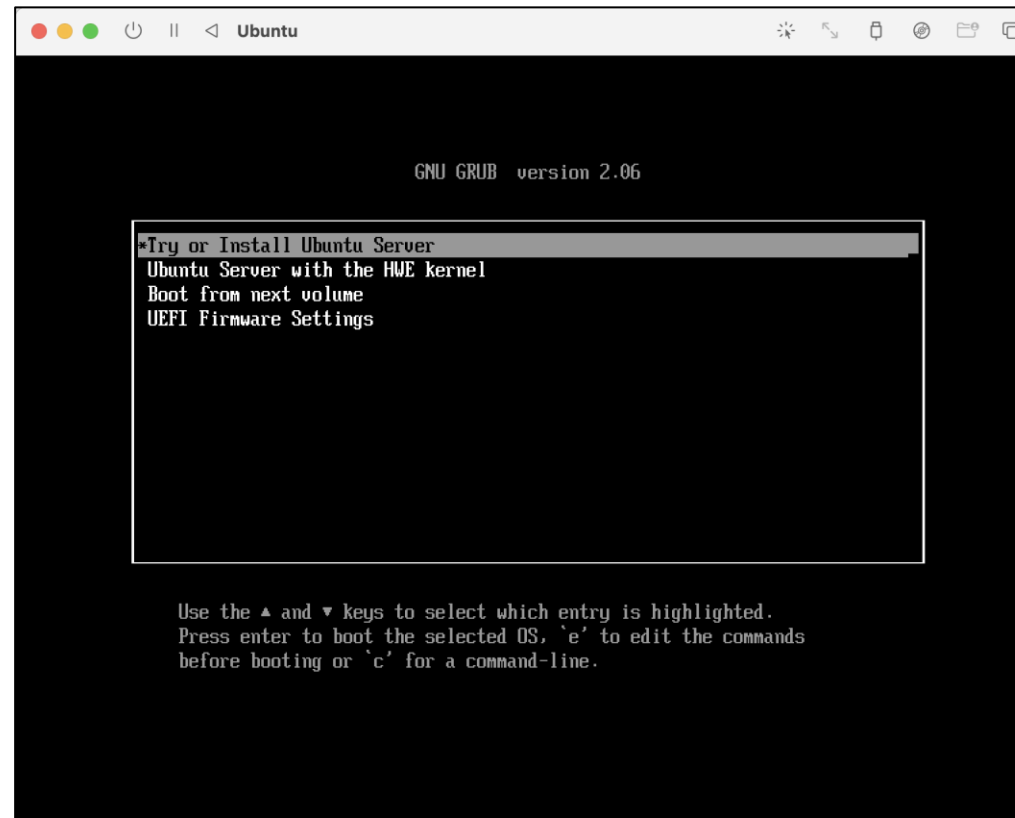
- 초기 VM 실행 화면



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

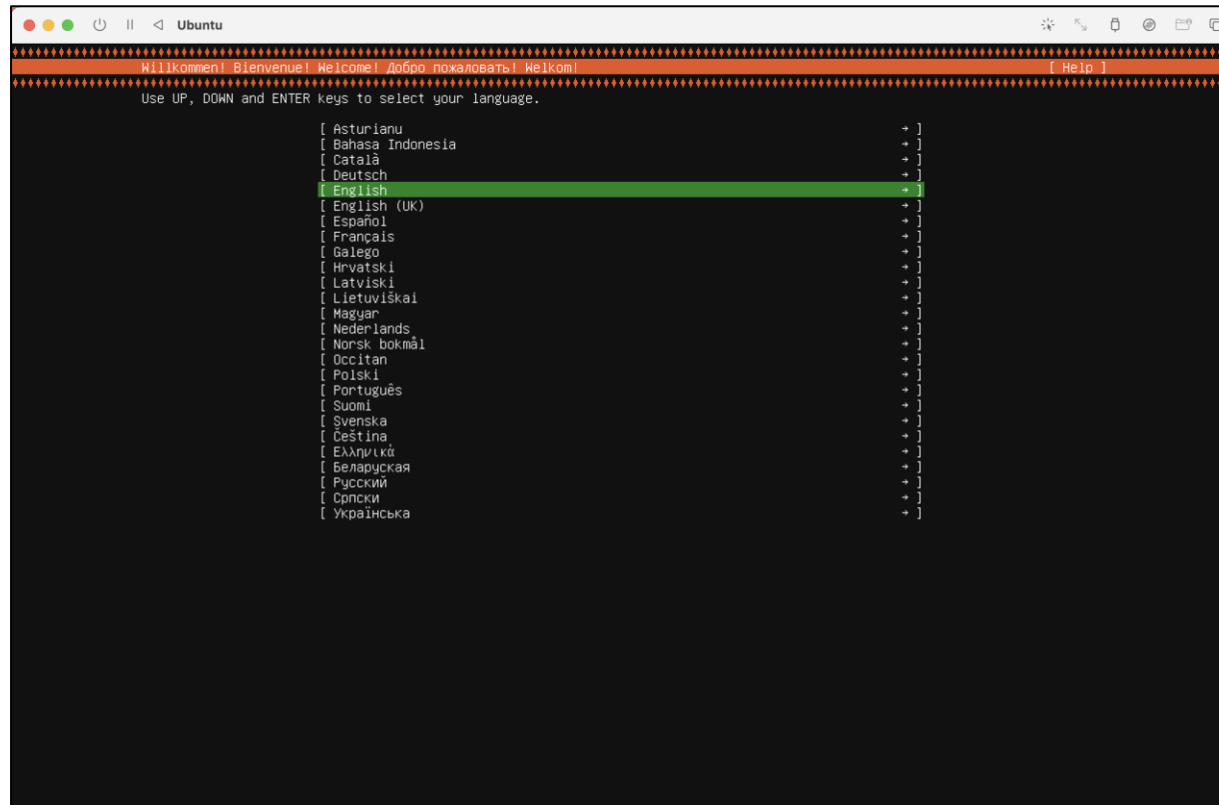
- 단계에 따라 설치 진행



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

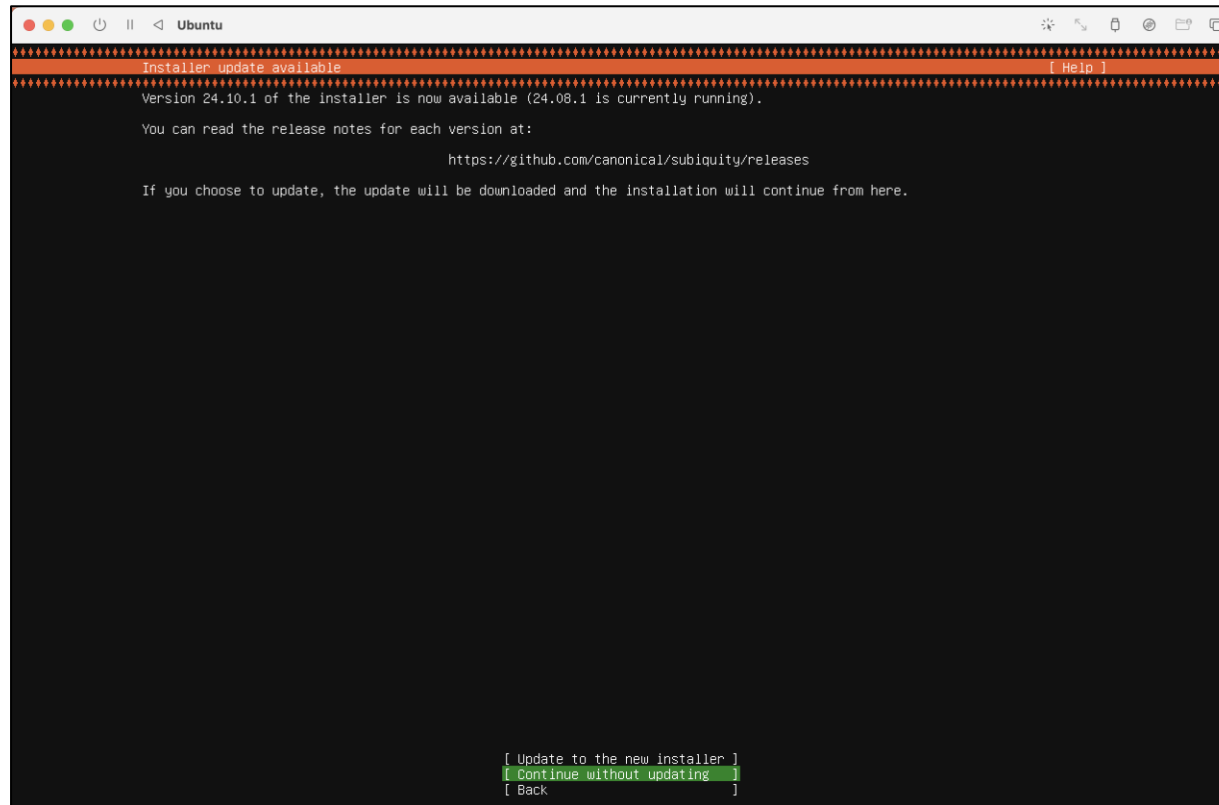
- 언어 설정: English



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

- 24버전으로 업데이트: 안함



The screenshot shows a terminal window titled "Ubuntu" with a dark background and orange borders. The text inside the window reads:

```
*****
Installer update available [ Help ]
*****
Version 24.10.1 of the installer is now available (24.08.1 is currently running).

You can read the release notes for each version at:

    https://github.com/canonical/subiquity/releases

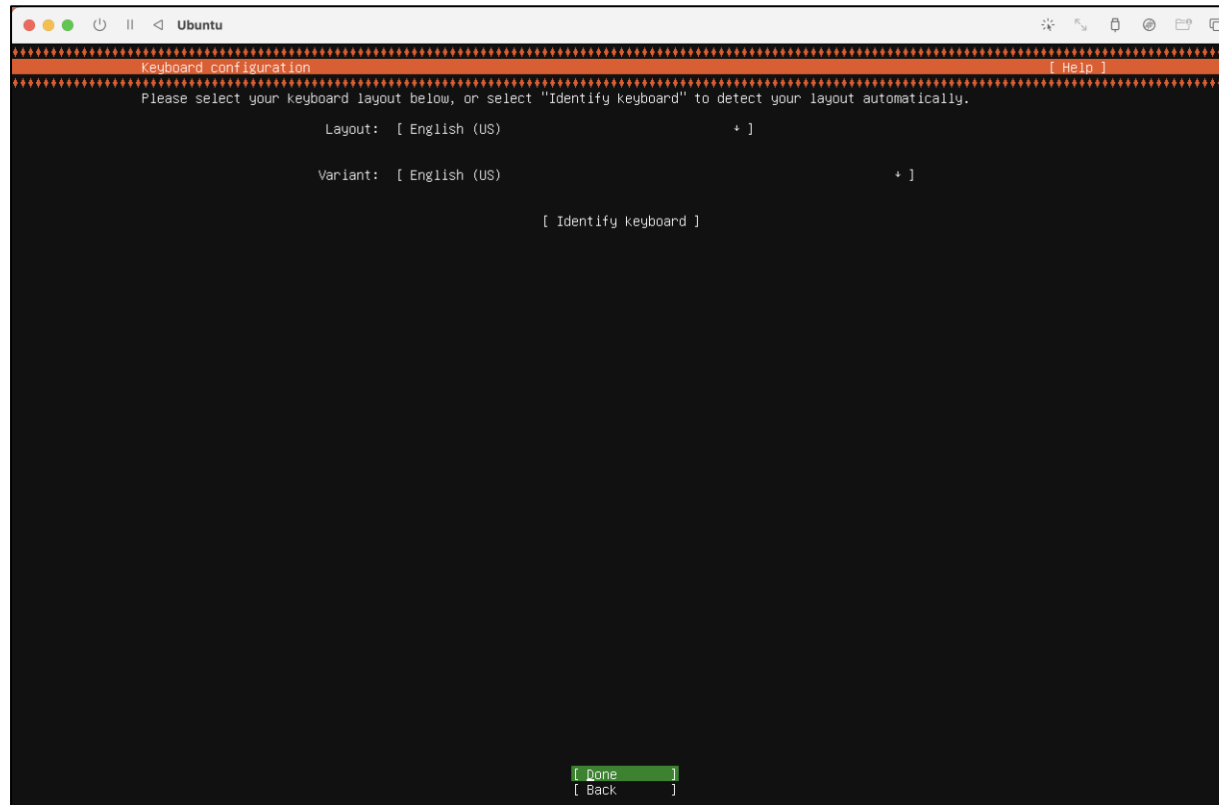
If you choose to update, the update will be downloaded and the installation will continue from here.

[ Update to the new installer ]
[ Continue without updating ]
[ Back ]
```

## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

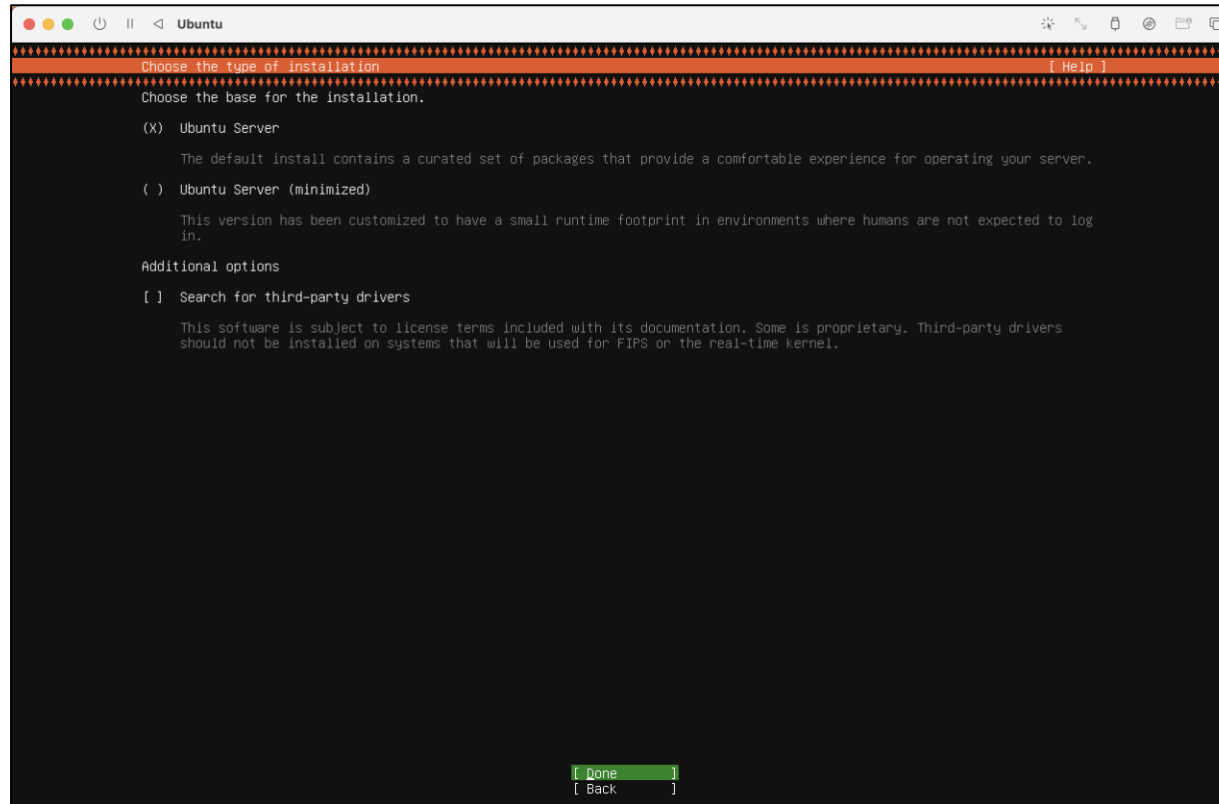
- 키보드 레이아웃: English



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

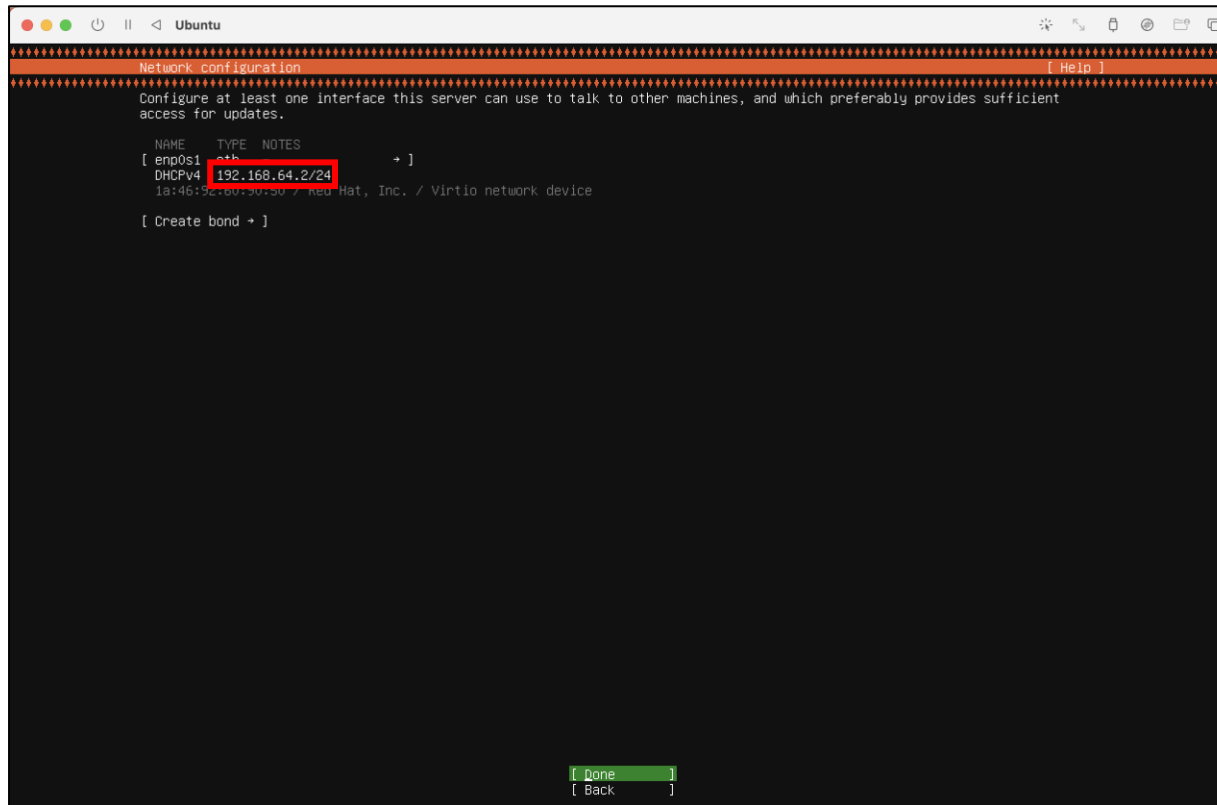
- Ubuntu 설치: Default



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

- 네트워크 설정: 그대로 진행, SSH를 사용할 것이라면 IP 주소 기억할 것

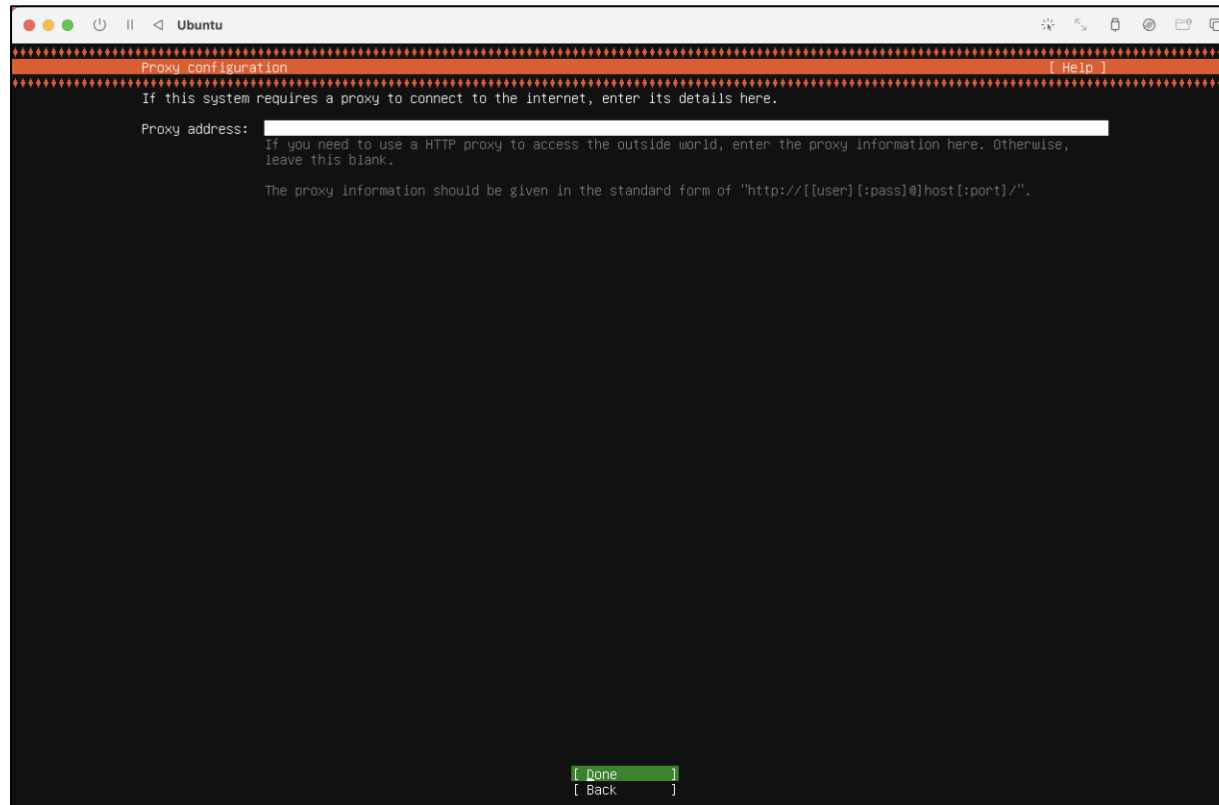




## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

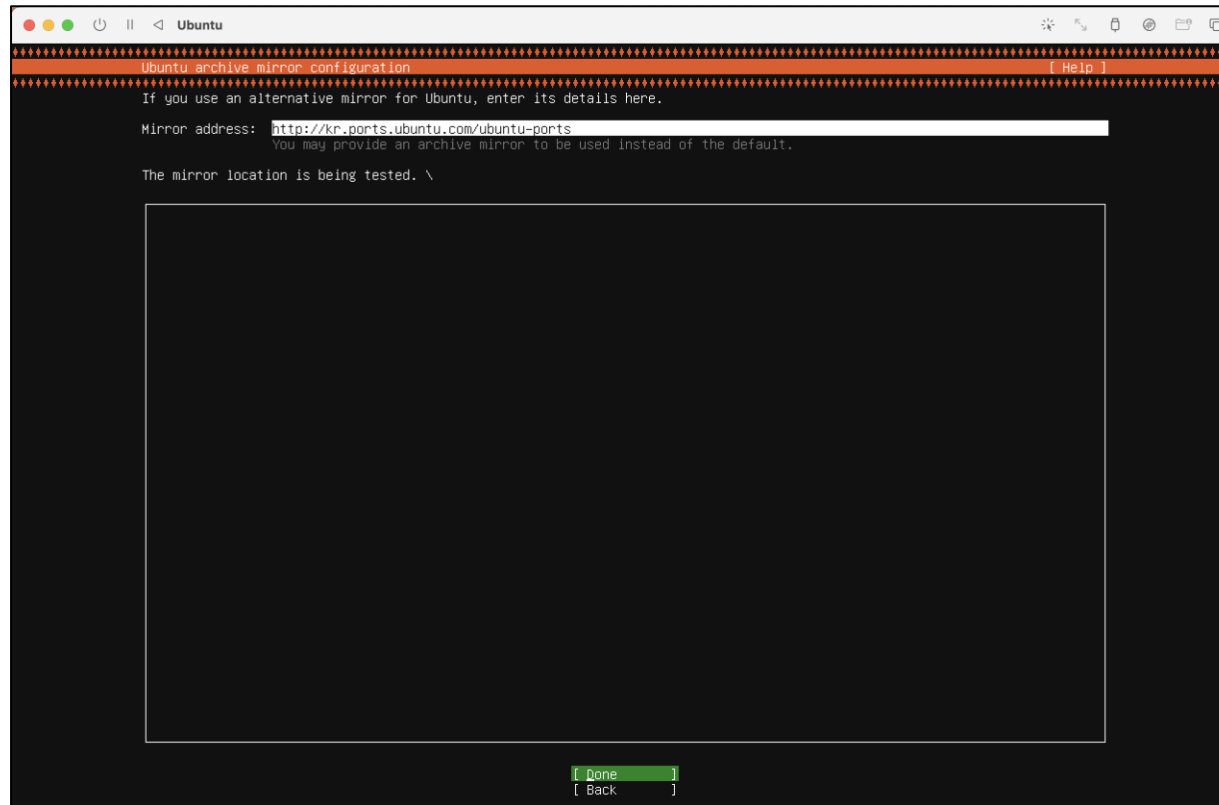
- Proxy 주소: 그대로 진행



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

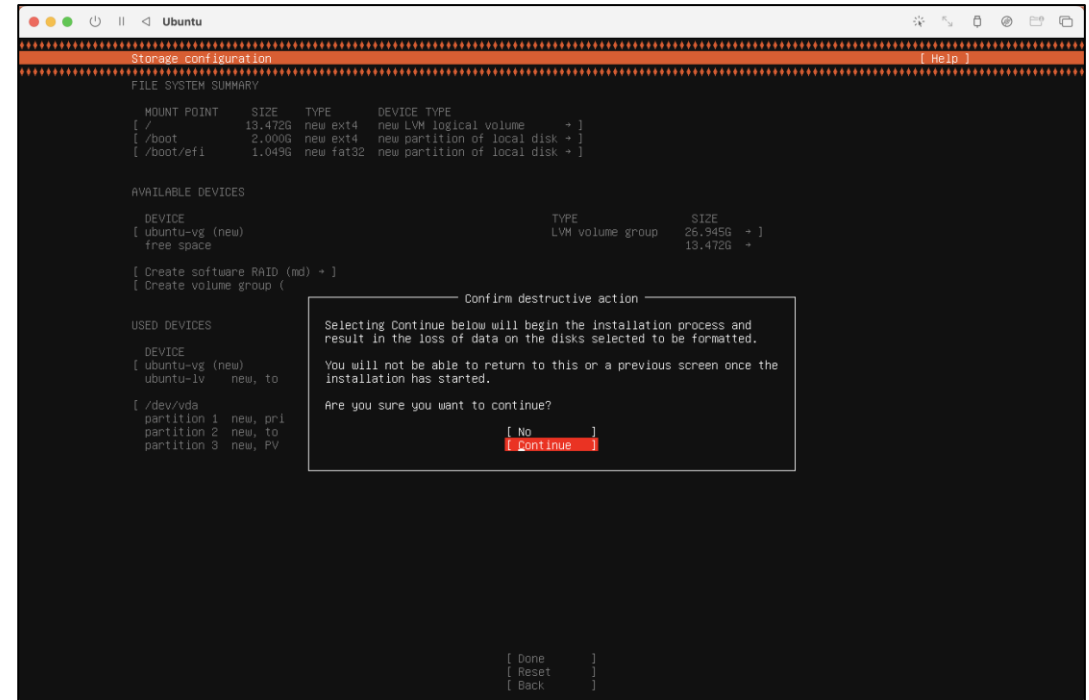
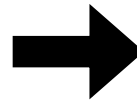
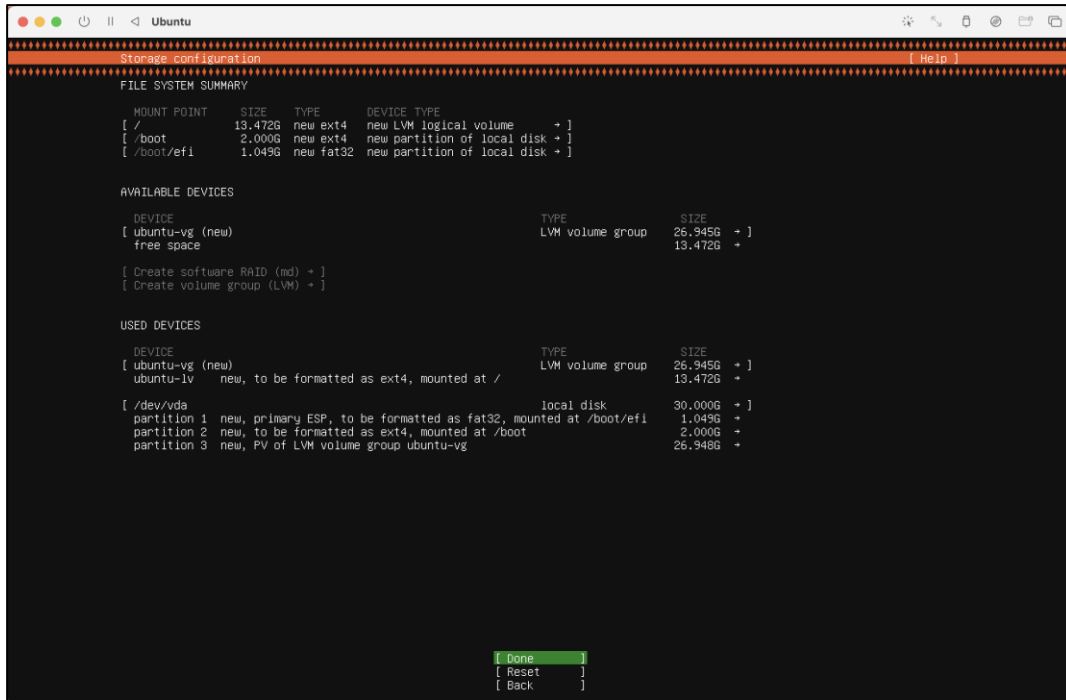
- Mirror 서버: 그대로 진행



## 2. Mac (ARM)

### 4) 초기 설정 - Ubuntu 설치

- 파일 시스템 설정: 그대로 진행



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

#### - 계정 설정

Profile configuration

Enter the username and password you will use to log in to the system. You can configure it later, but a password is still needed for sudo.

Your name: 예시) user72250292

Your servers name: ubuntu  
The name it uses when it talks to other computers.

Pick a username: 예시) user72250292

Choose a password: \*\*\*\*

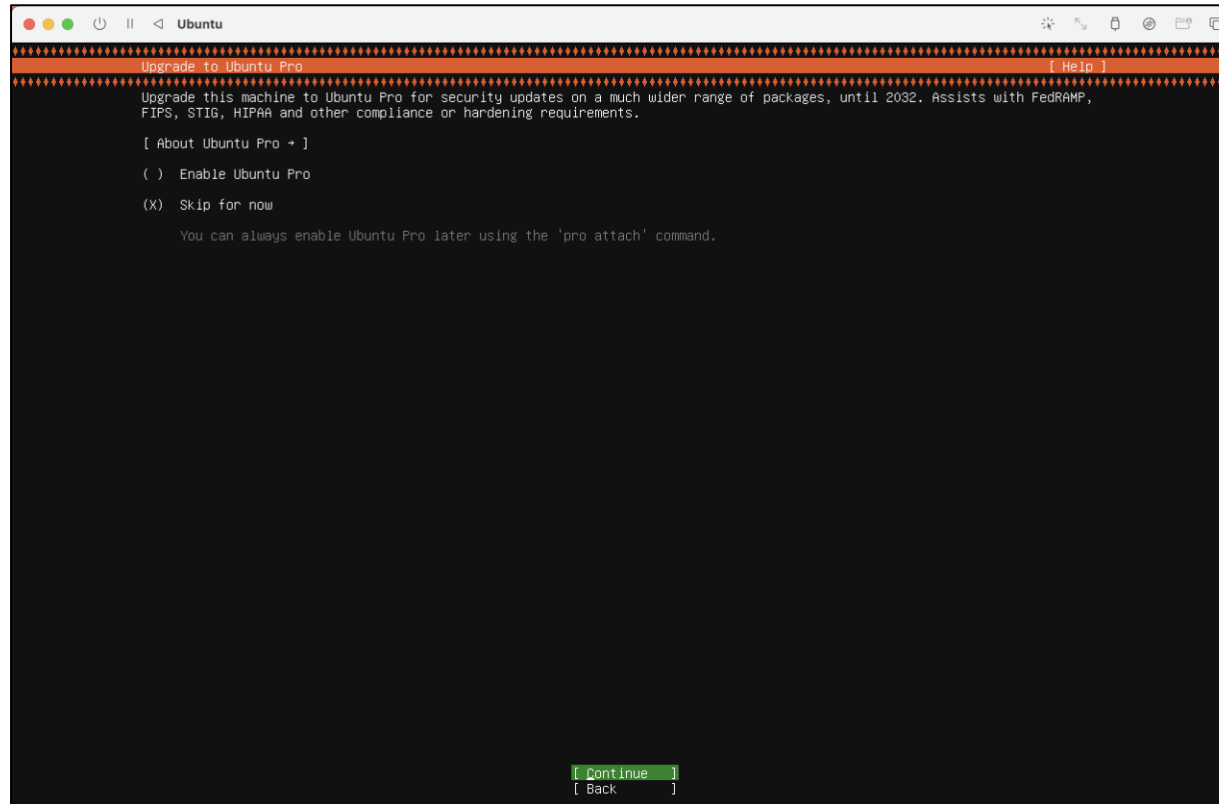
Confirm your password: \*\*\*\*

user + 학번으로 설정

## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

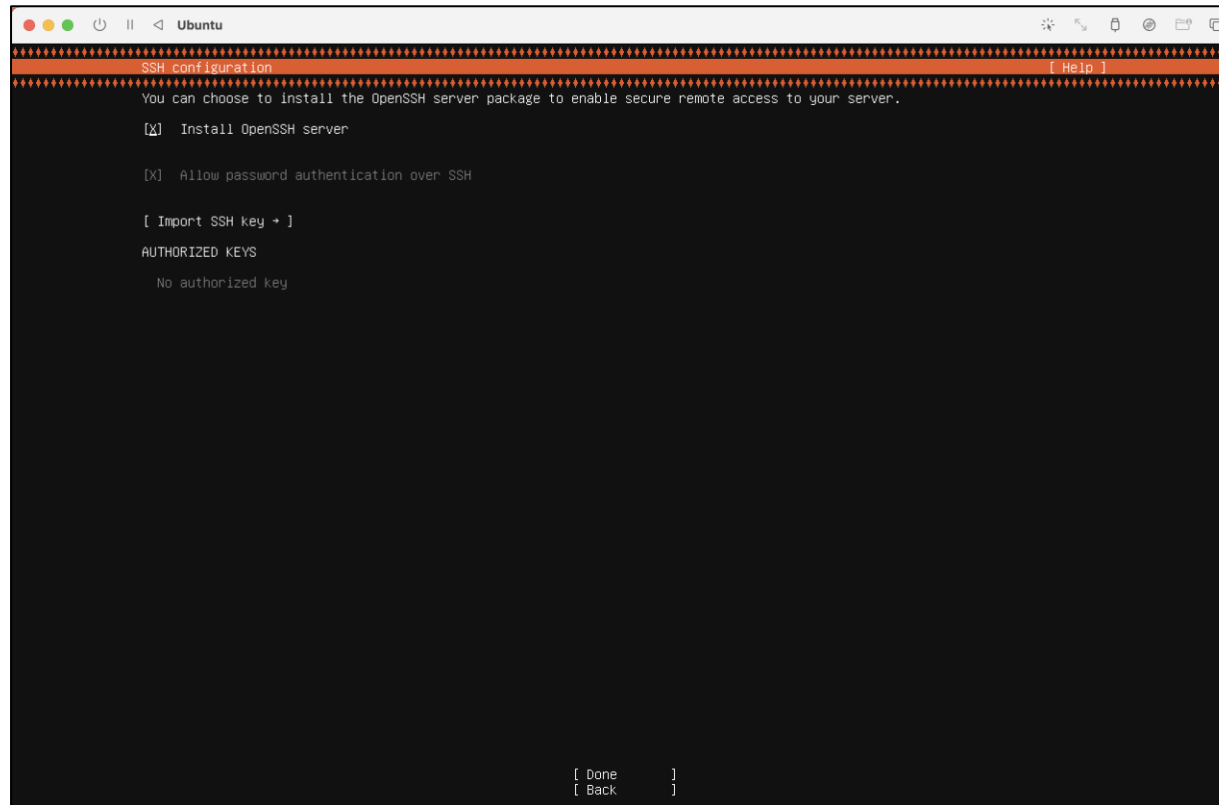
- Ubuntu pro로 업데이트: Skip



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

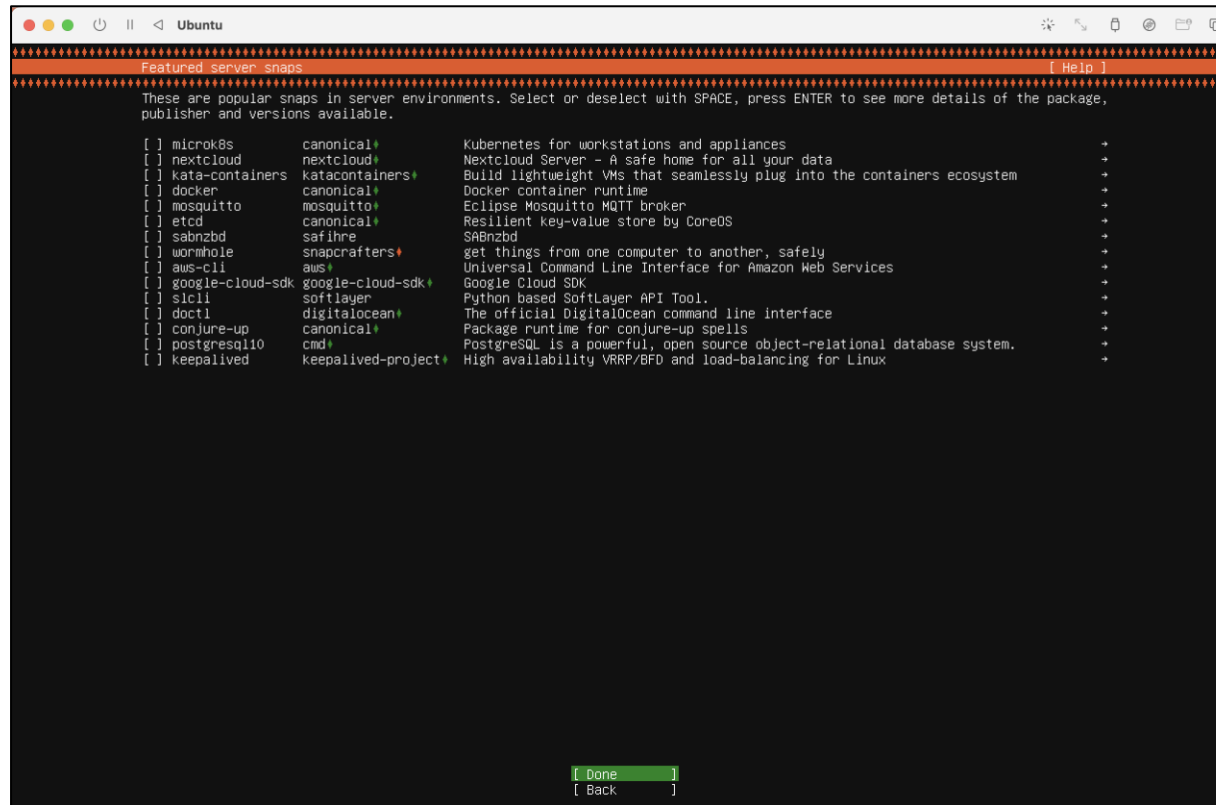
- SSH 서버 설정: SSH를 사용할 것이라면 체크 아니면 Skip



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

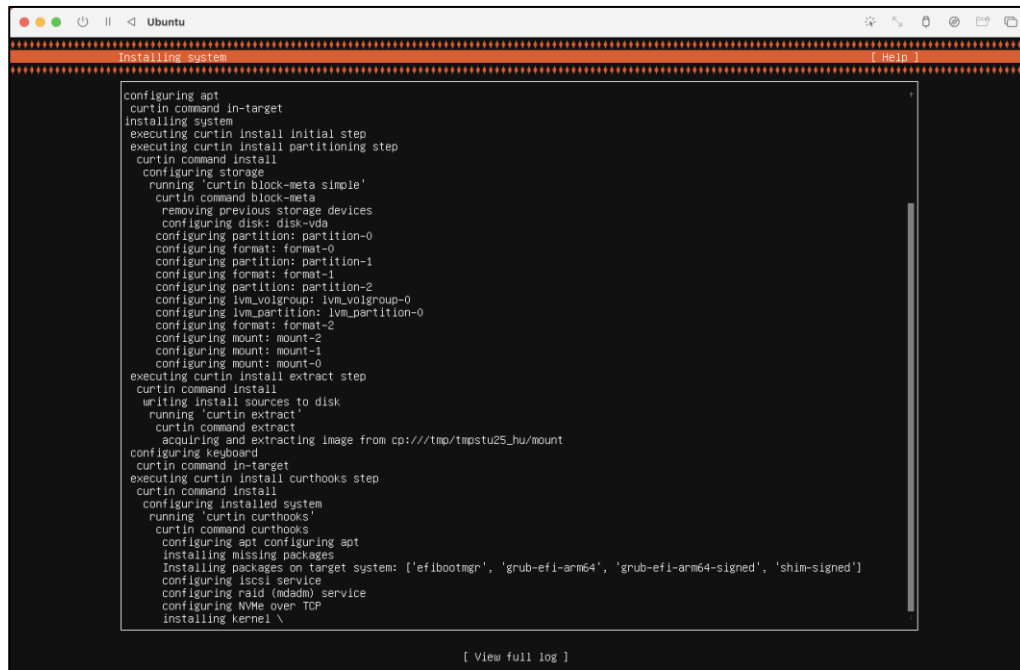
- 서버 snap 패키지 설치: Skip



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

- Ubuntu 설치 진행 화면, 완료되면 reboot

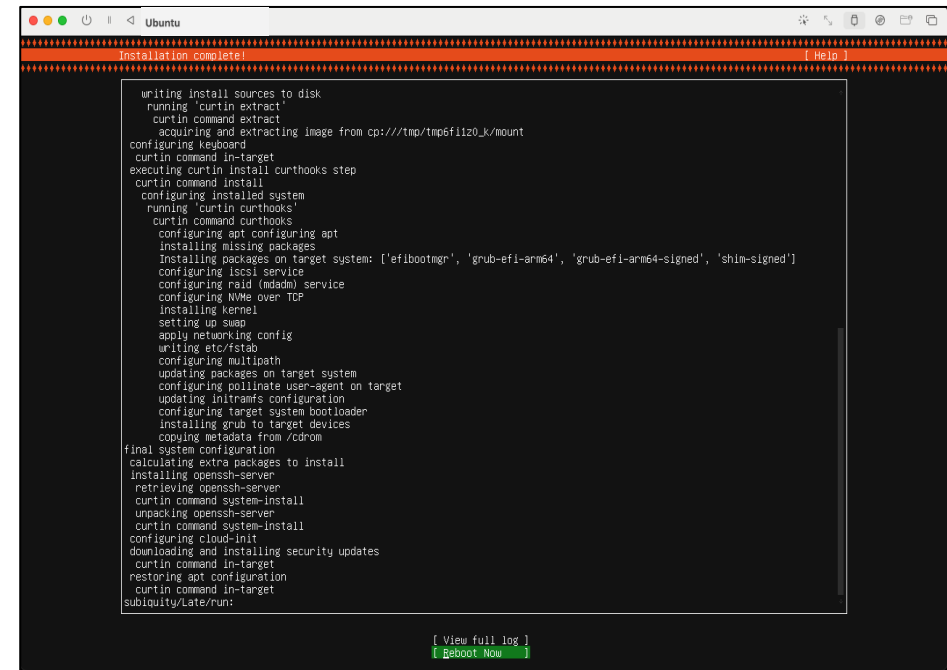
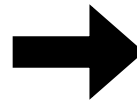


A screenshot of the Ubuntu installation progress window. The window has a title bar with standard Ubuntu window controls and a 'Help' button. The main content area shows a list of installation steps in a monospaced font. The steps include configuring apt, installing system, executing curtin install initial step, executing curtin install partitioning step, curtin command install, configuring storage, running 'curtin block-meta simple', curtin command block-meta, removing previous storage devices, configuring disk: disk-vda, configuring partition: partition-0, configuring format: format-0, configuring partition: partition-1, configuring format: format-1, configuring partition: partition-2, configuring lvm\_voigroup: lvm\_voigroup-0, configuring lvm\_partition: lvm\_partition-0, configuring format: format-2, configuring mount: mount-2, configuring mount: mount-1, configuring mount: mount-0, executing curtin install extract step, curtin command install, writing install sources to disk, running 'curtin extract', curtin command extract, acquiring and extracting image from cp:///tmp/tmpst25\_hu/mount, configuring keyboard, curtin command in-target, executing curtin install curthooks step, curtin command install, configuring installed system, running 'curtin curthooks', curtin command curthooks, configuring apt, installing missing packages, installing packages on target system: ['efibootmgr', 'grub-efi-arm64', 'grub-efi-arm64-signed', 'shim-signed'], configuring lscsi service, configuring raid (mdadm) service, configuring NVMe over TCP, and installing kernel. At the bottom of the window, there is a link that says '[ View full log ]'.

```
installing system
[ Help ]

configuring apt
curtin command in-target
installing system
executing curtin install initial step
executing curtin install partitioning step
curtin command install
  configuring storage
    running 'curtin block-meta simple'
    curtin command block-meta
    removing previous storage devices
    configuring disk: disk-vda
    configuring partition: partition-0
    configuring format: format-0
    configuring partition: partition-1
    configuring format: format-1
    configuring partition: partition-2
    configuring lvm_voigroup: lvm_voigroup-0
    configuring lvm_partition: lvm_partition-0
    configuring format: format-2
    configuring mount: mount-2
    configuring mount: mount-1
    configuring mount: mount-0
  executing curtin install extract step
  curtin command install
    writing install sources to disk
    running 'curtin extract'
    curtin command extract
    acquiring and extracting image from cp:///tmp/tmpst25_hu/mount
  configuring keyboard
  curtin command in-target
executing curtin install curthooks step
curtin command install
  configuring installed system
  running 'curtin curthooks'
  curtin command curthooks
  configuring apt
  installing missing packages
  installing packages on target system: ['efibootmgr', 'grub-efi-arm64', 'grub-efi-arm64-signed', 'shim-signed']
  configuring lscsi service
  configuring raid (mdadm) service
  configuring NVMe over TCP
  installing kernel \

[ View full log ]
```



A screenshot of the Ubuntu installation complete window. The window has a title bar with standard Ubuntu window controls and a 'Help' button. The main content area shows a list of post-installation steps in a monospaced font. The steps include writing install sources to disk, running 'curtin extract', curtin command extract, acquiring and extracting image from cp:///tmp/tmpf1iz20\_k/mount, configuring keyboard, curtin command in-target, executing curtin install curthooks step, curtin command install, configuring installed system, running 'curtin curthooks', curtin command curthooks, configuring apt, installing missing packages, installing packages on target system: ['efibootmgr', 'grub-efi-arm64', 'grub-efi-arm64-signed', 'shim-signed'], configuring lscsi service, configuring raid (mdadm) service, configuring NVMe over TCP, installing kernel, setting up swap, apply networking config, writing etc/fstab, configuring multipath, updating packages on target system, configuring pollinate user-agent on target, updating initramfs configuration, configuring target system bootloader, installing grub to target devices, copying metadata from /cdrom, final system configuration, calculating extra packages to install, installing openssh-server, retrieving openssh-server, curtin command system-install, unpacking openssh-server, curtin command system-install, configuring cloud-init, downloading and installing security updates, curtin command in-target, restoring apt configuration, curtin command in-target, and subiquity/late/run. At the bottom of the window, there are two links: '[ View full log ]' and '[ Reboot Now ]'.

```
Installation complete!
[ Help ]

writing install sources to disk
  running 'curtin extract'
  curtin command extract
  acquiring and extracting image from cp:///tmp/tmpf1iz20_k/mount
configuring keyboard
curtin command in-target
executing curtin install curthooks step
curtin command install
  configuring installed system
  running 'curtin curthooks'
  curtin command curthooks
  configuring apt
  installing missing packages
  installing packages on target system: ['efibootmgr', 'grub-efi-arm64', 'grub-efi-arm64-signed', 'shim-signed']
  configuring lscsi service
  configuring raid (mdadm) service
  configuring NVMe over TCP
  installing kernel
  setting up swap
  apply networking config
  writing etc/fstab
  configuring multipath
  updating packages on target system
  configuring pollinate user-agent on target
  updating initramfs configuration
  configuring target system bootloader
  installing grub to target devices
  copying metadata from /cdrom
final system configuration
  calculating extra packages to install
  installing openssh-server
  retrieving openssh-server
  curtin command system-install
  unpacking openssh-server
  curtin command system-install
  configuring cloud-init
  downloading and installing security updates
  curtin command in-target
  restoring apt configuration
  curtin command in-target
  subiquity/late/run

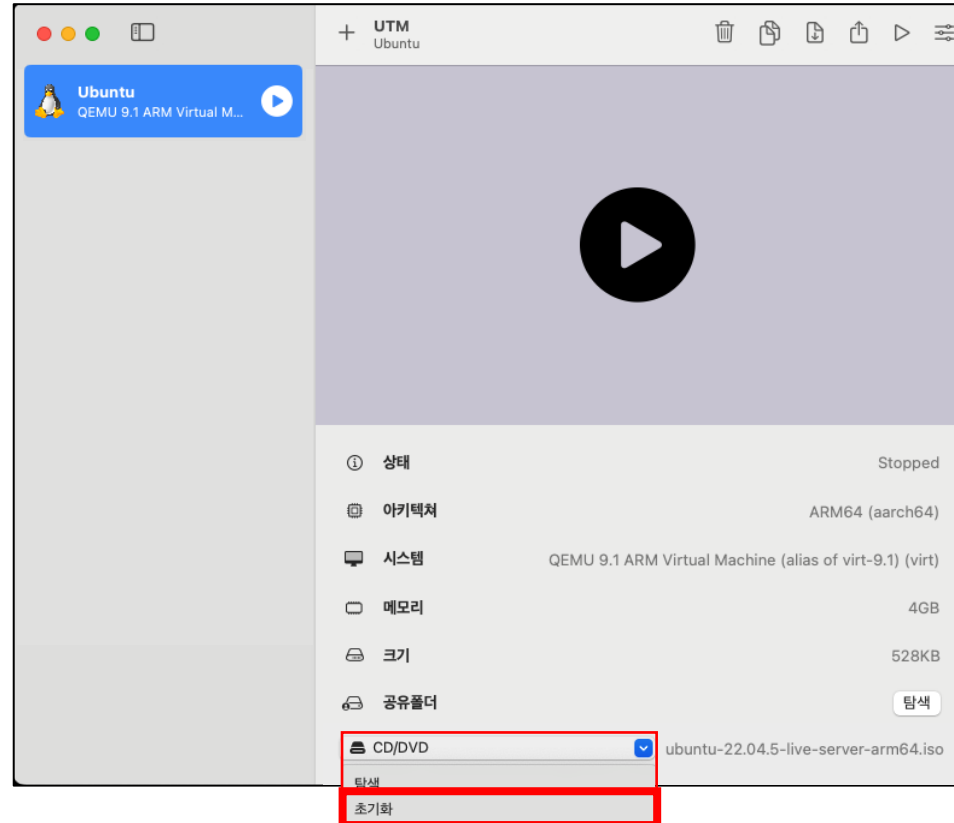
[ View full log ]
[ Reboot Now ]
```



## 2. Mac (ARM)

### 4) 초기 설정 – Ubuntu 설치

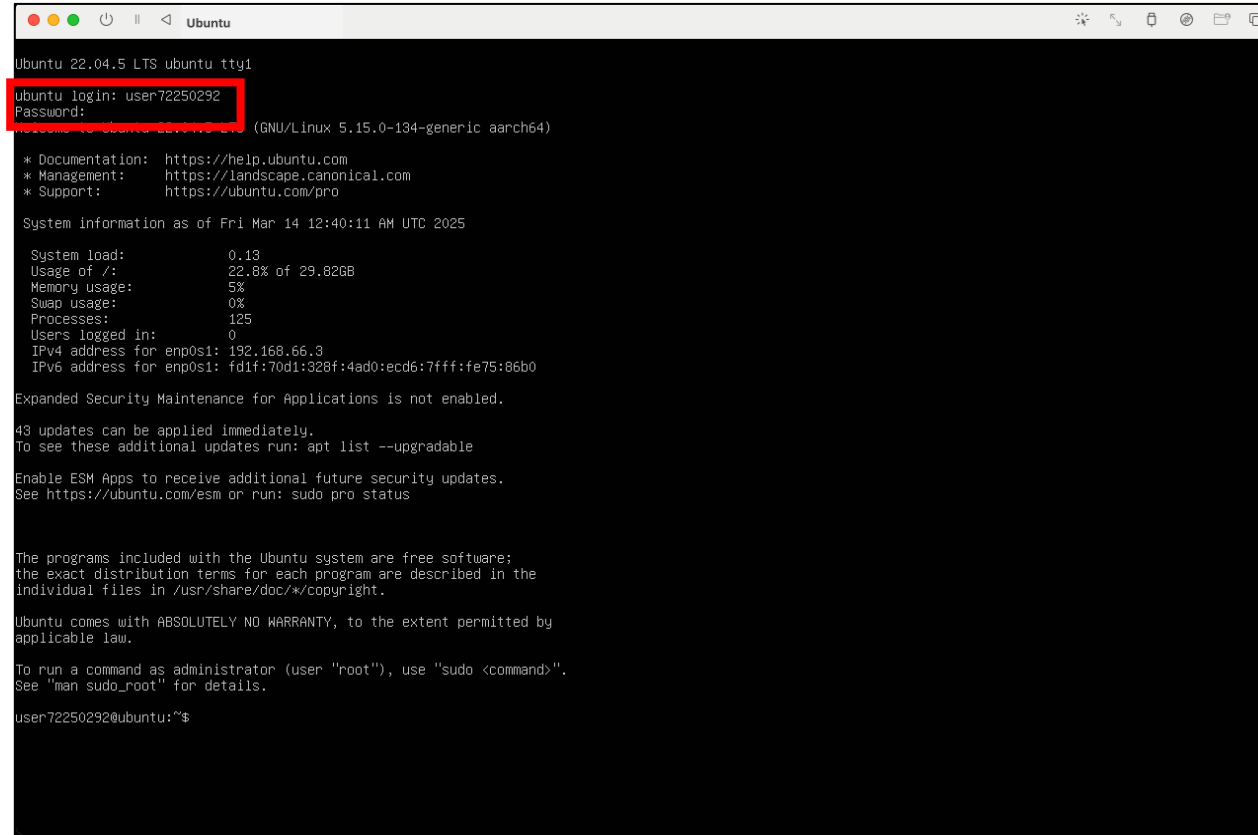
- 검은 화면이 뜬다면 VM 종료, CD/DVD 초기화 후 다시 시작



## 2. Mac (ARM)

### 4) 초기 설정 - 로그인

- {username} + 비밀번호 입력



```
Ubuntu 22.04.5 LTS ubuntu tty1
Ubuntu login: user72250292
Password:
(GNU/Linux 5.15.0-134-generic aarch64)

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

System information as of Fri Mar 14 12:40:11 AM UTC 2025

System load:          0.13
Usage of /:            22.8% of 29.82GB
Memory usage:         5%
Swap usage:           0%
Processes:            125
Users logged in:      0
IPv4 address for enp0s1: 192.168.66.3
IPv6 address for enp0s1: fd1f:70d1:328f:4ad0:ecd6:7fff:fe75:86b0

Expanded Security Maintenance for Applications is not enabled.

43 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

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.

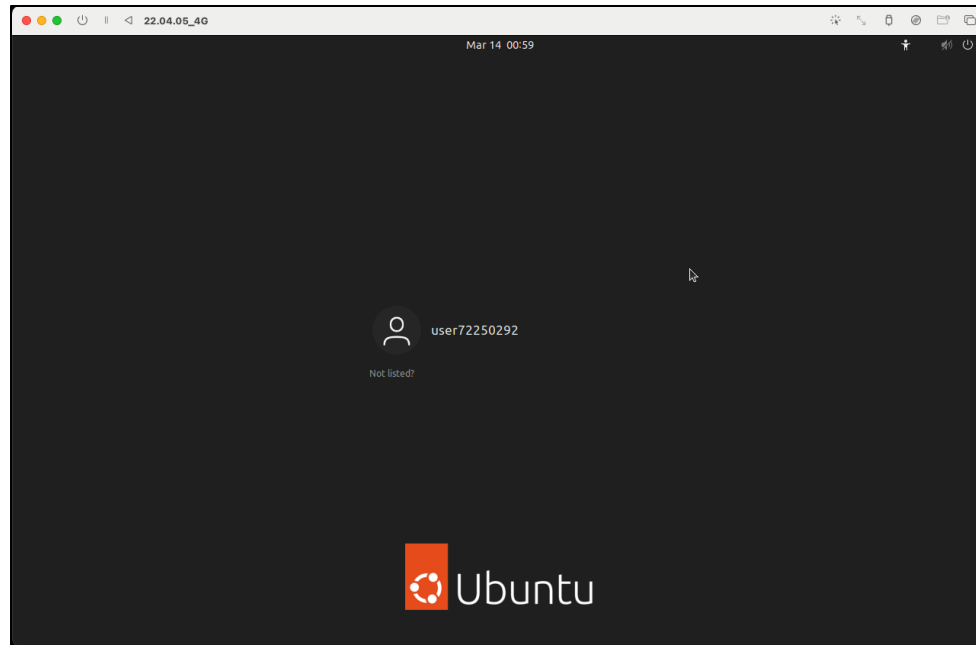
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

user72250292@ubuntu:~$
```

## 2. Mac (ARM)

### 5) Optional – GUI 환경으로 변경

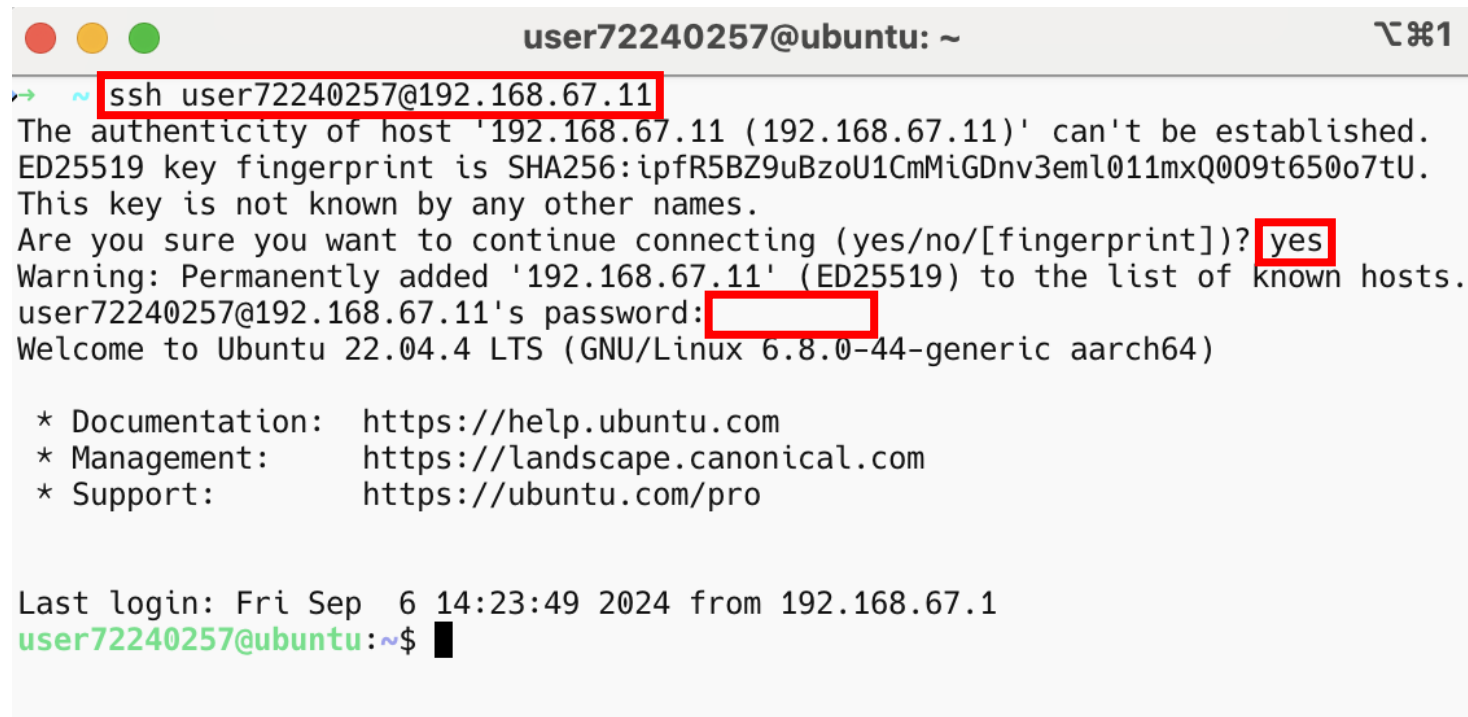
- \$ sudo apt install ubuntu-desktop
- 설치 완료 후 \$ sudo reboot
- reboot 후 다음 화면이 나온다면 성공



## 2. Mac (ARM)

### 5) Optional – SSH 설정

- 호스트에서 SSH로 VM에 접속, 39p에서 알아낸 IP 주소 또는 \$ ip a로 알아낸 IP 주소
- ~ ssh {username}@{IP address}

A terminal window titled 'user72240257@ubuntu: ~' showing an SSH connection attempt. The command 'ssh user72240257@192.168.67.11' is entered. The terminal displays a warning about the host's authenticity and asks for confirmation to continue. The user responds 'yes'. The terminal then shows the warning that the host has been added to the known hosts list, followed by a password prompt where the user's input is masked. After successful authentication, the terminal displays the Ubuntu 22.04.4 LTS login banner and system information, including documentation, management, and support links. The last login time is shown as 'Fri Sep 6 14:23:49 2024 from 192.168.67.1'. The prompt changes to 'user72240257@ubuntu:~\$'.

```
user72240257@ubuntu: ~
~ ssh user72240257@192.168.67.11
The authenticity of host '192.168.67.11 (192.168.67.11)' can't be established.
ED25519 key fingerprint is SHA256:ipfR5BZ9uBzoU1CmMiGDnv3eml011mxQ009t650o7tU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.67.11' (ED25519) to the list of known hosts.
user72240257@192.168.67.11's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.8.0-44-generic aarch64)

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

Last login: Fri Sep  6 14:23:49 2024 from 192.168.67.1
user72240257@ubuntu:~$
```

# Q&A?

2025.03.17

T.A. 김보승

[bskim1102@dankook.ac.kr](mailto:bskim1102@dankook.ac.kr)