2025 Operating System

Lab 0. Lab overview & Linux on VM

2025.03.17

T.A. 김보승

bskim1102@dankook.ac.kr

Contact

- T.A. 김보승
- 연락처: bskim1102@dankook.ac.kr
- 연구실: 소프트웨어 ICT관 515호

Overview

- OS의 3가지 요소
  - Virtualization
  - Concurrency
  - Persistence

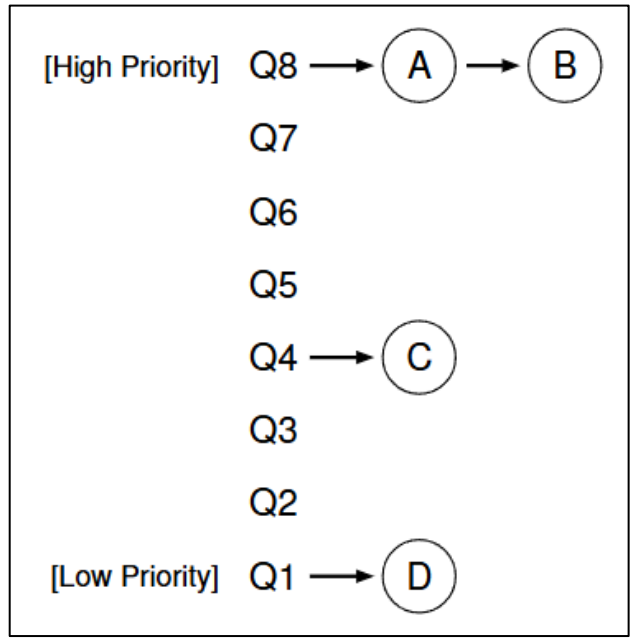
Operating Systems: Three Easy Pieces

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

- 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

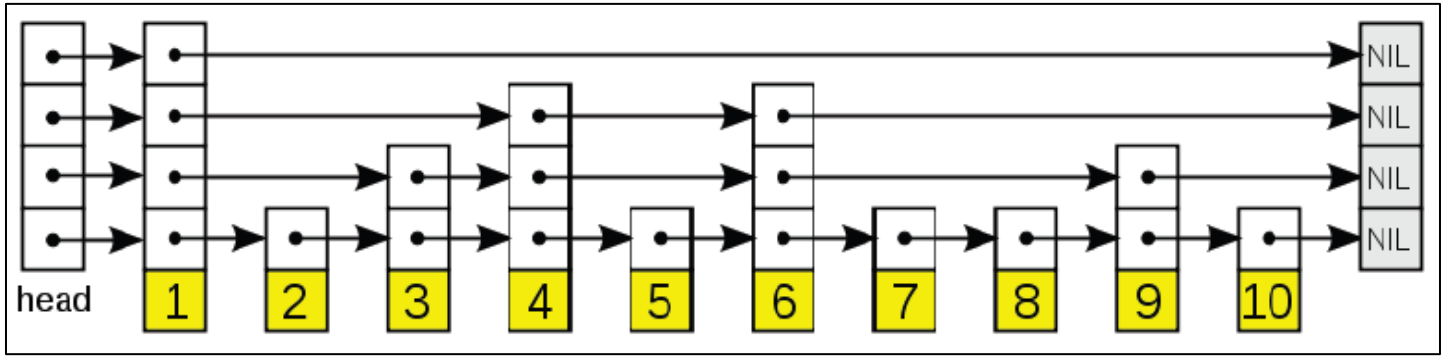


MLFQ Scheduling 예시

  - Context Switch Time, 워크로드에 따른 scheduling 결과 분석을 목표

Lab 2: Concurrency

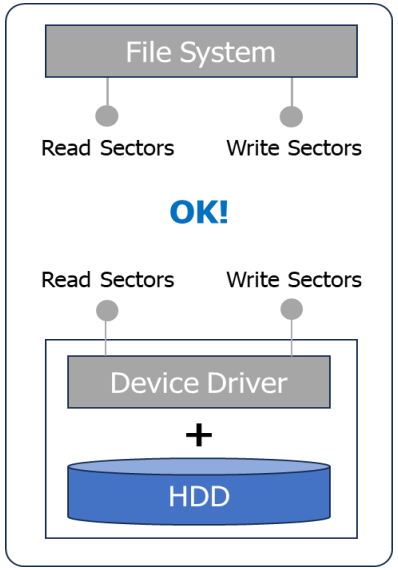
- Concurrent Data Structure
  - Skip-list를 아래 3가지 버전으로 구현
    - Without lock
    - Coarse-grained lock
    - Fine-grained lock
  - Critical Section에 대한 이해 및 Lock 사용에 따른 성능 변화 분석을 목표



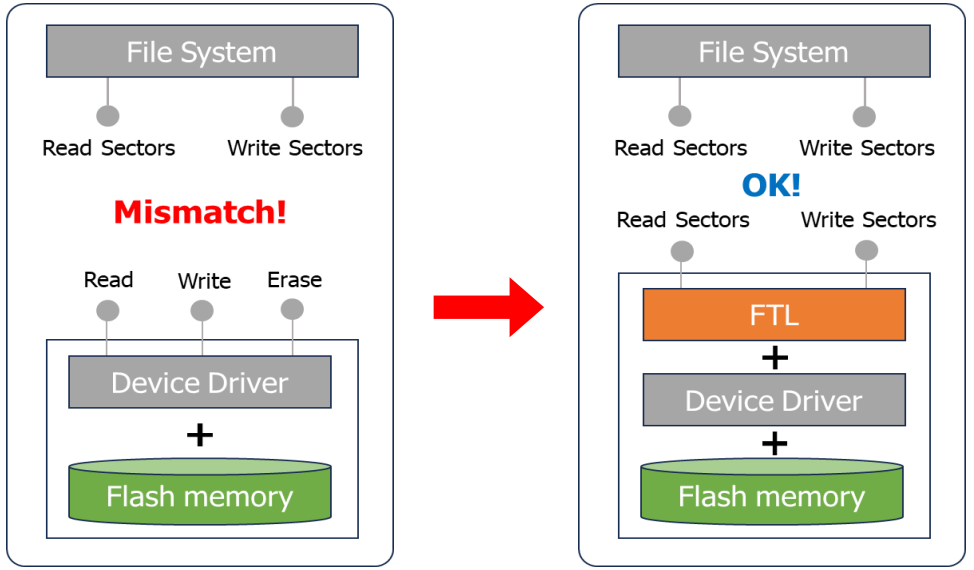
Skip-list 자료구조

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

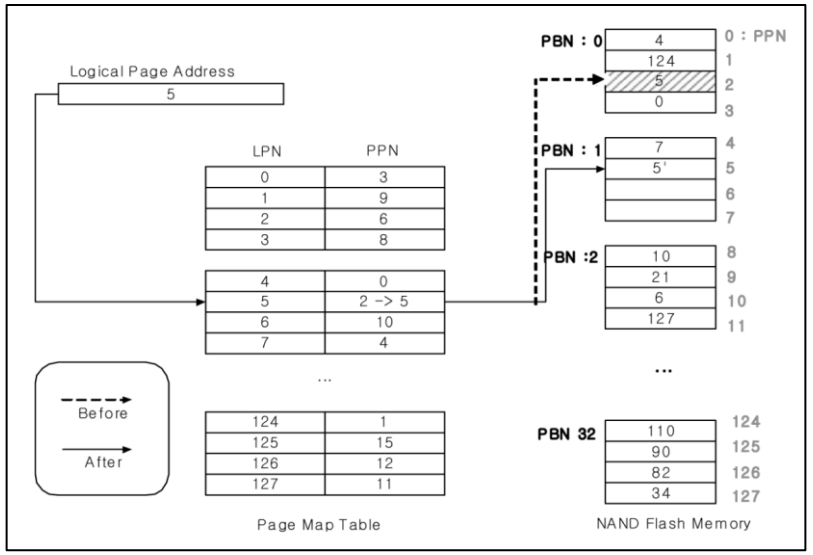


파일 시스템과 SSD간 데이터 전송 방식

https://codecapsule.com/2014/02/12/coding-for-ssds-part-3-pages-blocks-and-the-flash-translation-layer/

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



Page-mapping scheme

LPN: Logical Page Number

PPN: Physical Page Number

PBN: Physical Block Number

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

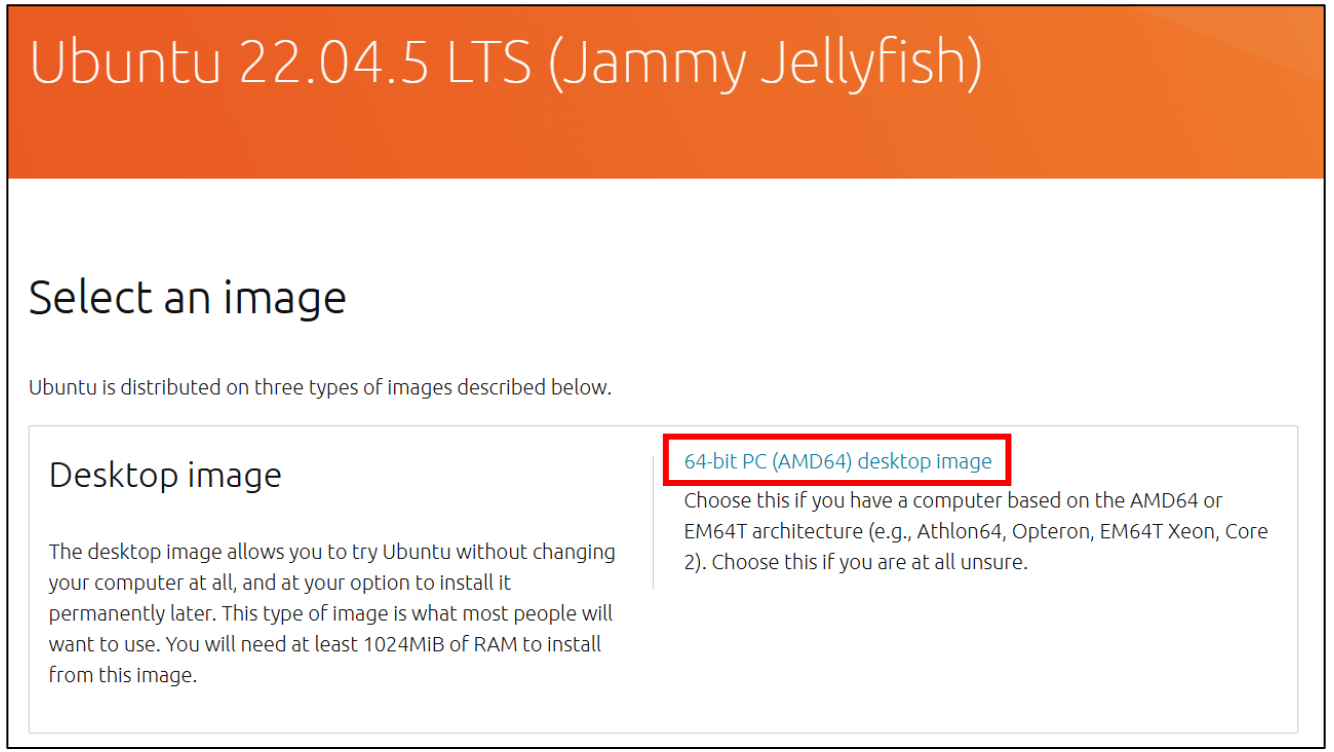
VirtualBox 다운로드 페이지

설치 후 VirtualBox 초기화면

1. Windows

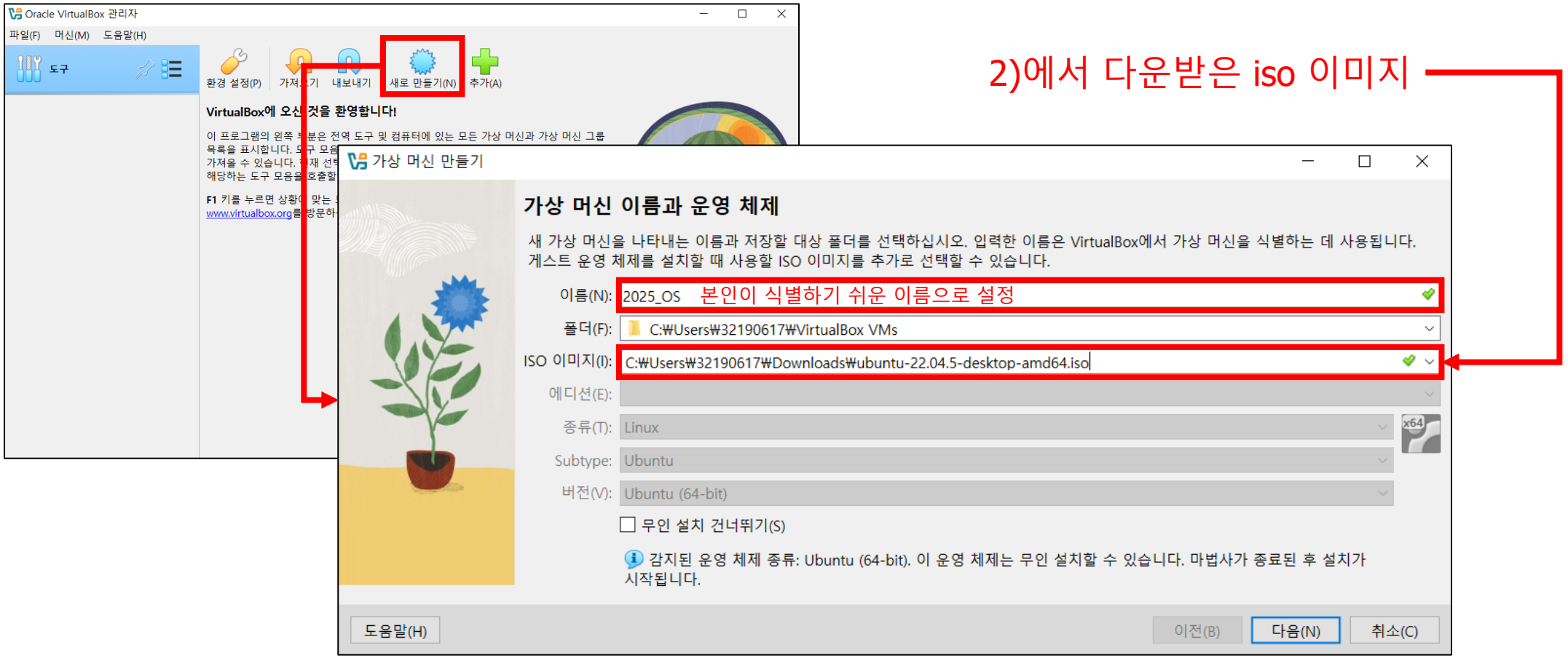
2) Ubuntu Desktop 22.04 이미지 다운로드 (AMD64)

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



1. Windows

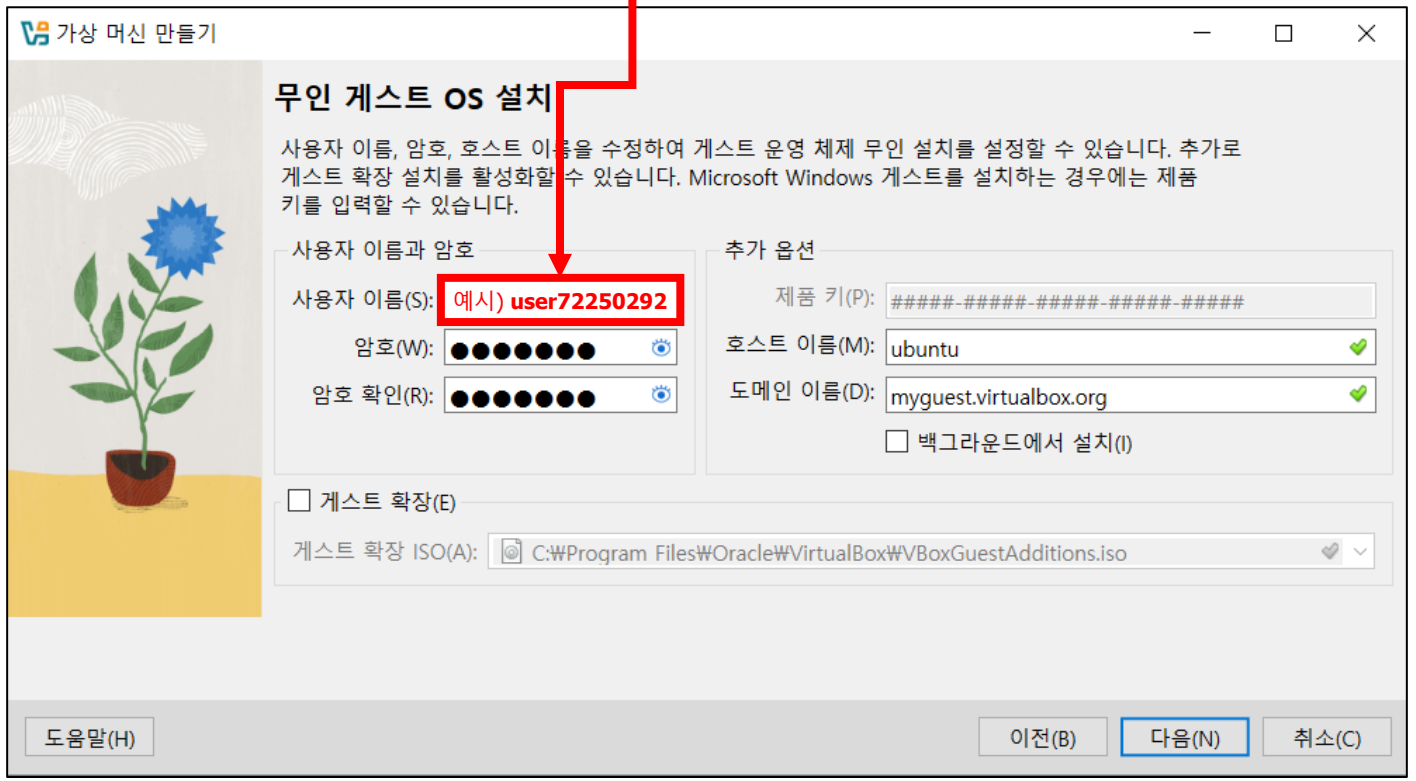
3) VM 생성 – VM 이름 및 ISO 이미지 선택



1. Windows

3) VM 생성 –계정 생성

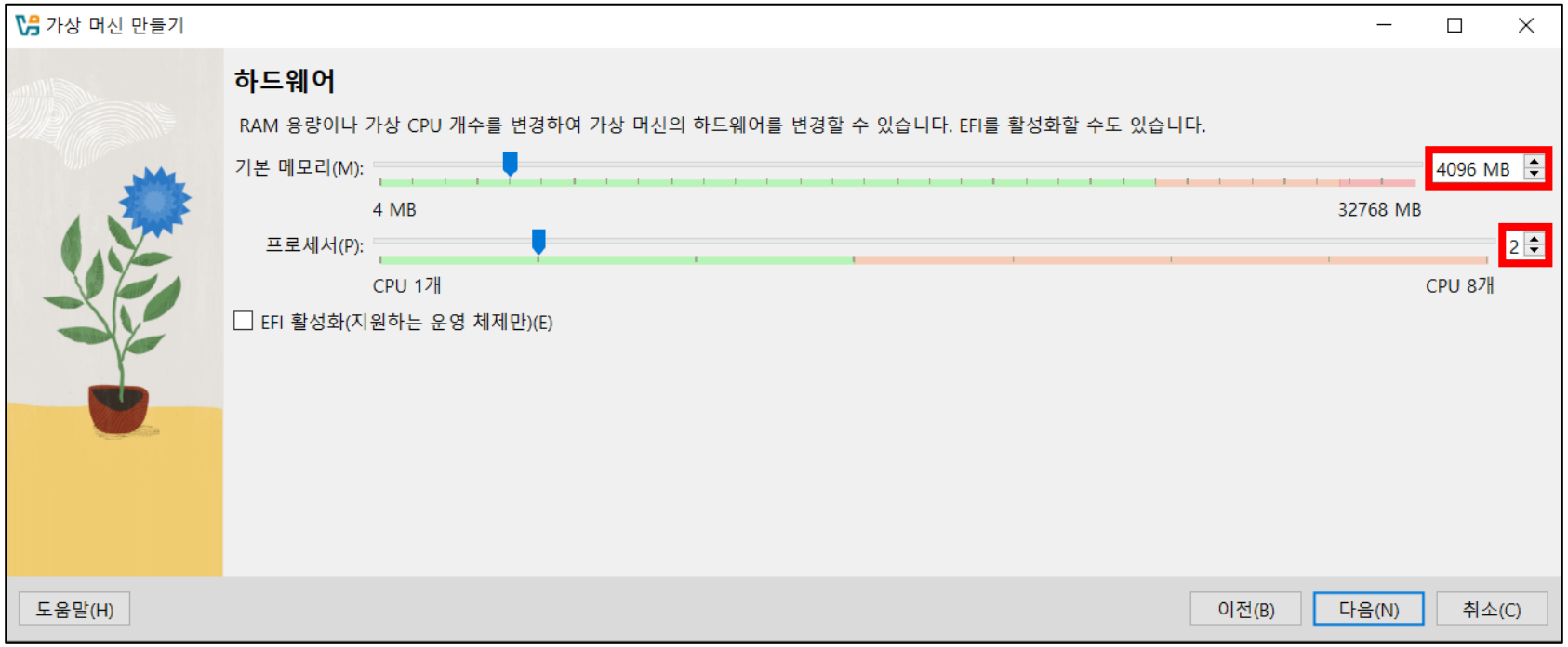
- 사용자 이름: **user + 학번으로 설정**



1. Windows

3) VM 생성 – 하드웨어 사양 선택

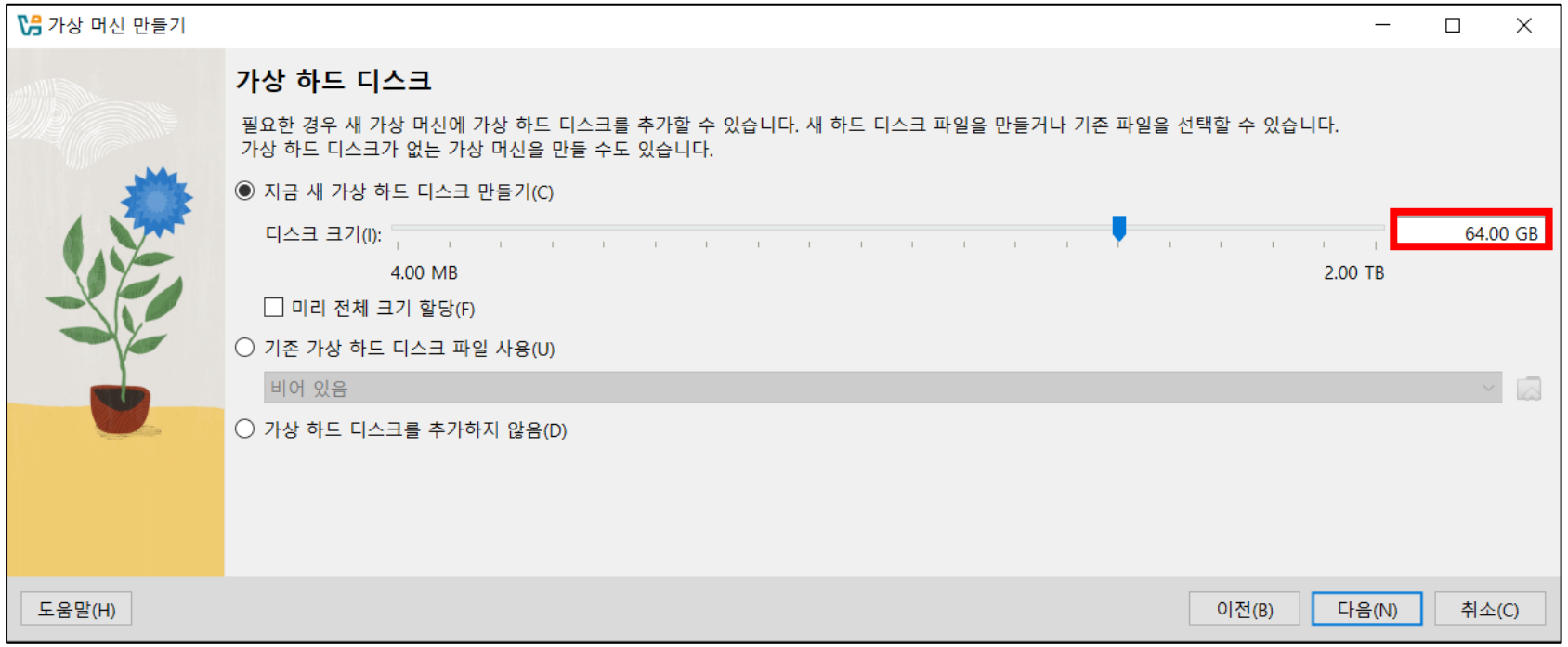
- 메모리: 4096MB, 프로세서: 2개
- 본인 컴퓨터 사양에 맞춰 설정 가능



1. Windows

3) VM 생성 – 하드웨어 사양 선택

- 디스크 크기: 64GB
- 여유공간이 부족하다면 작게 설정

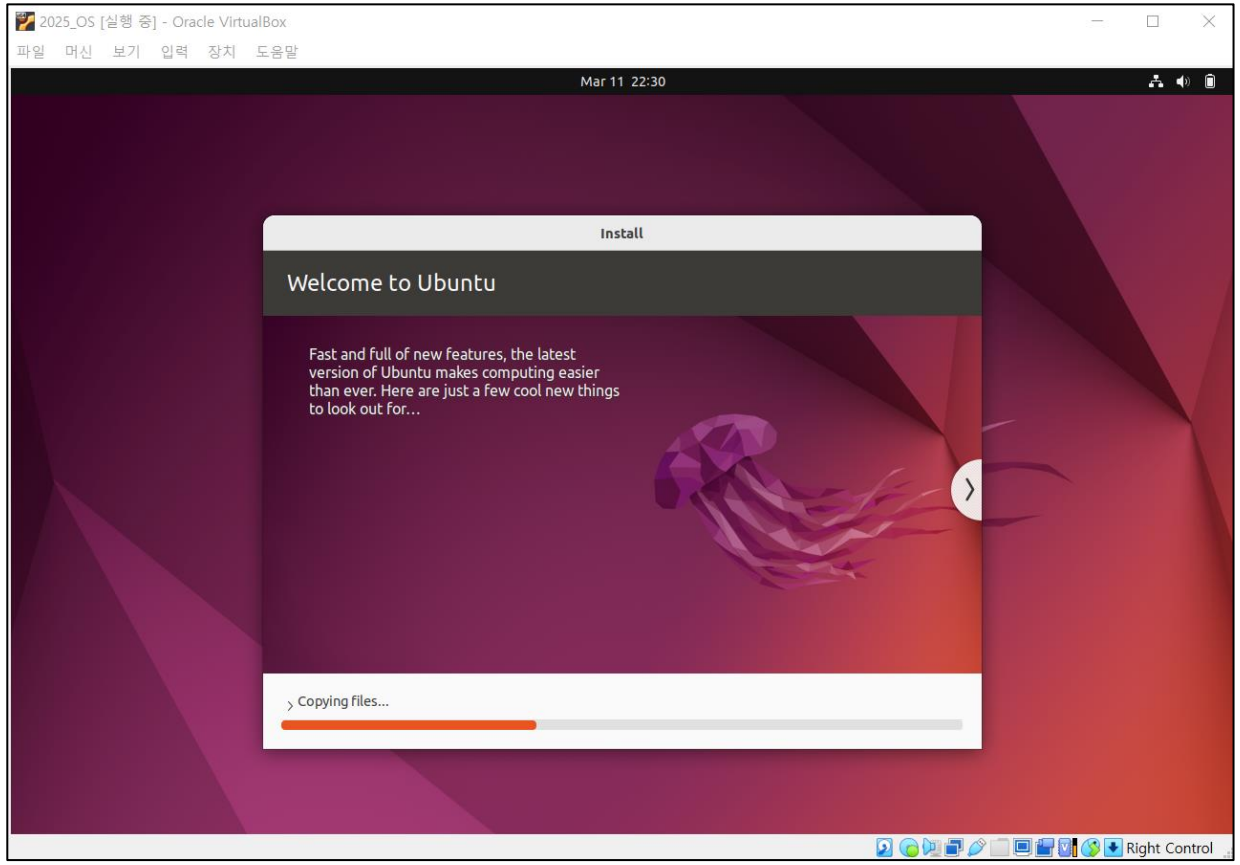


1. Windows

3) VM 생성 – 완료

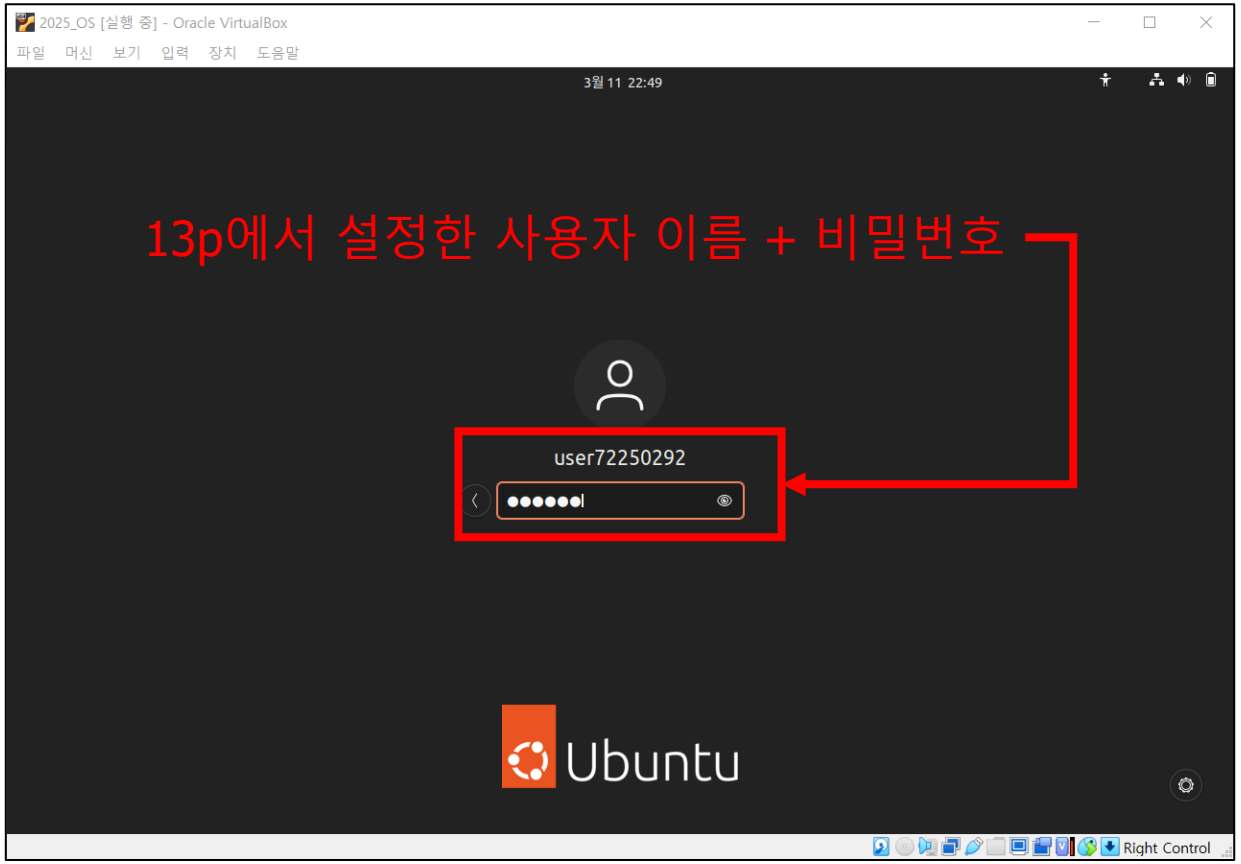
1. Windows

4) 초기 설정 – ubuntu 설치



1. Windows

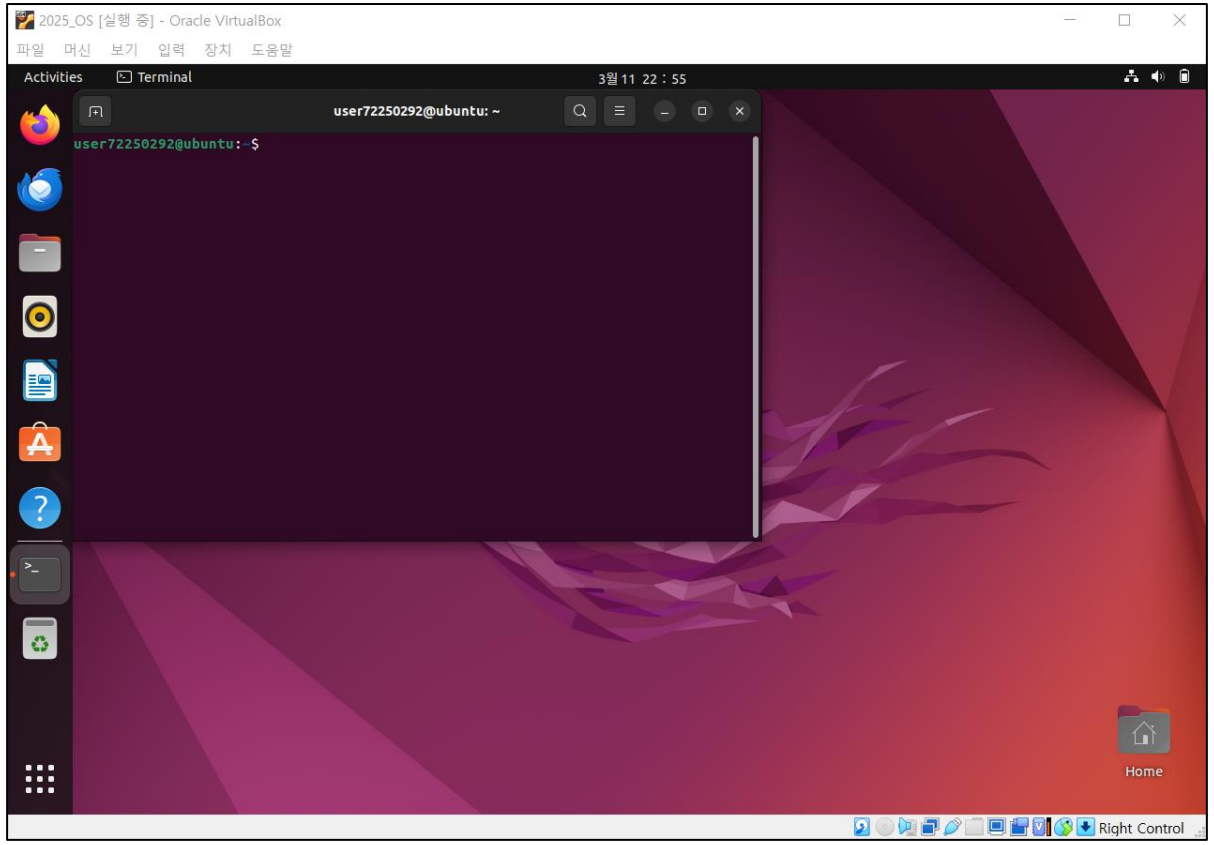
4) 초기 설정 – 계정 확인 및 로그인



1. Windows

4) 초기 설정 – 터미널 열기

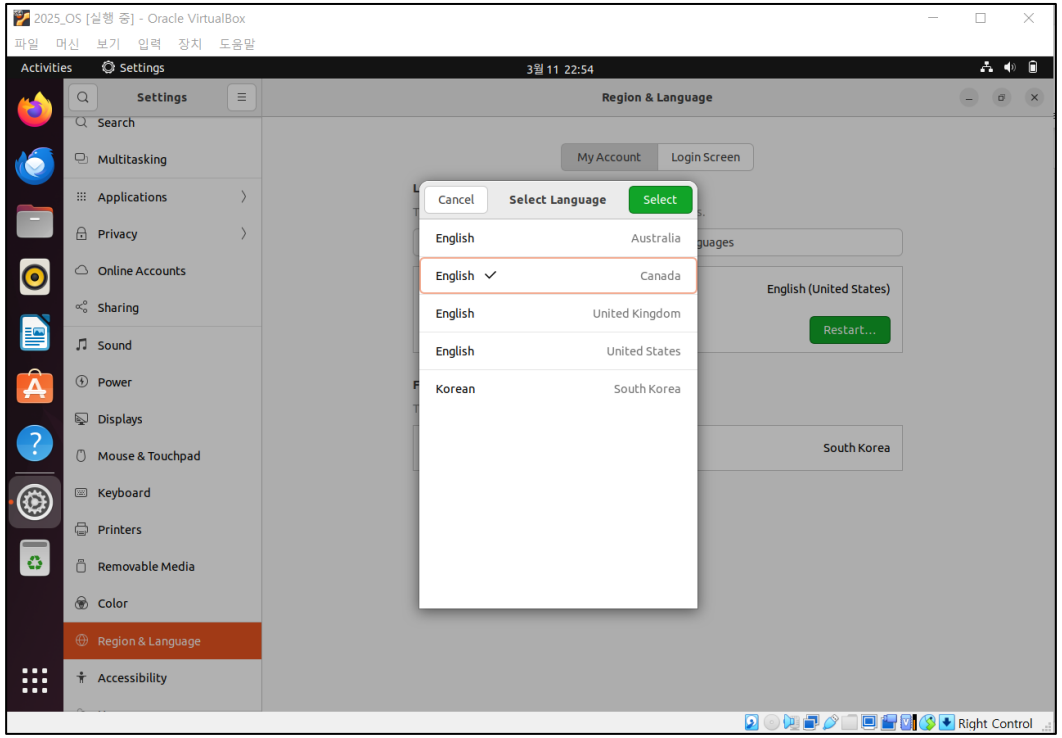
- 터미널 단축키: Ctrl + Alt + t



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 13p의 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

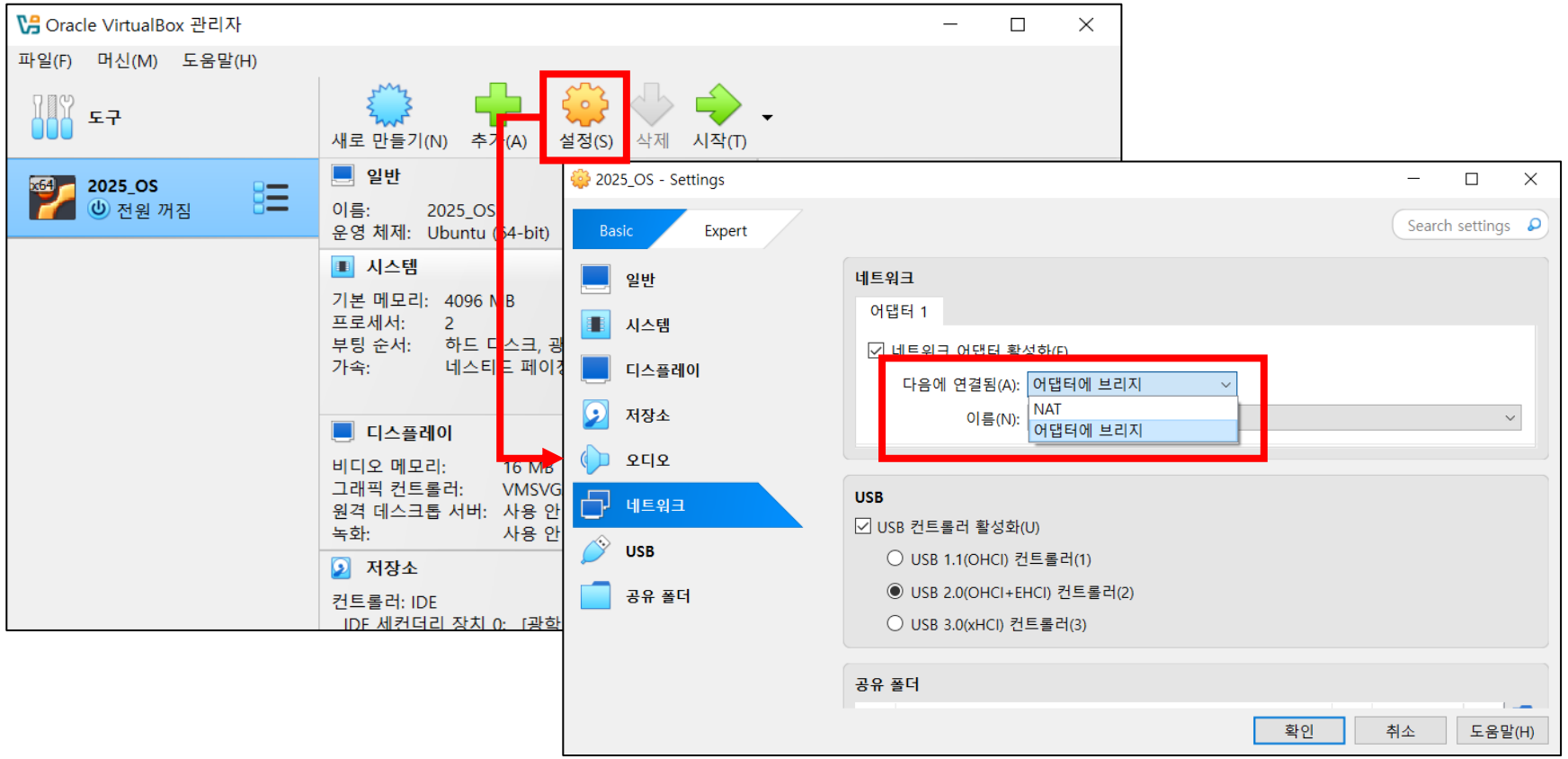
5) Optional – SSH 설정

- 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

```
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)
```

- $ ifconfig
  - IP 주소 확인

```
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: 24p에서 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:
```

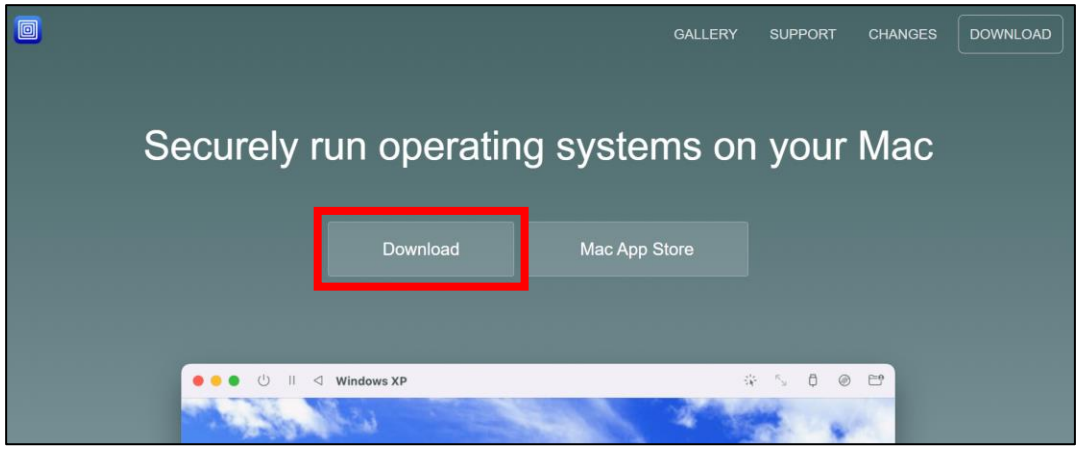
UTM 설치

- 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

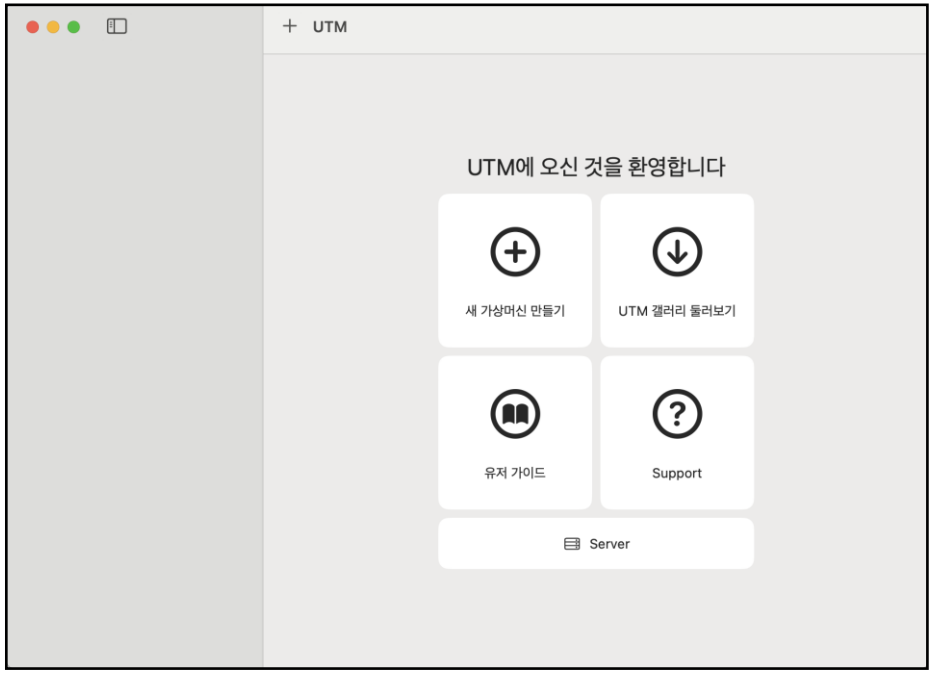
2. Mac (ARM)

1) UTM 설치

- https://mac.getutm.app/



UTM 다운로드 페이지

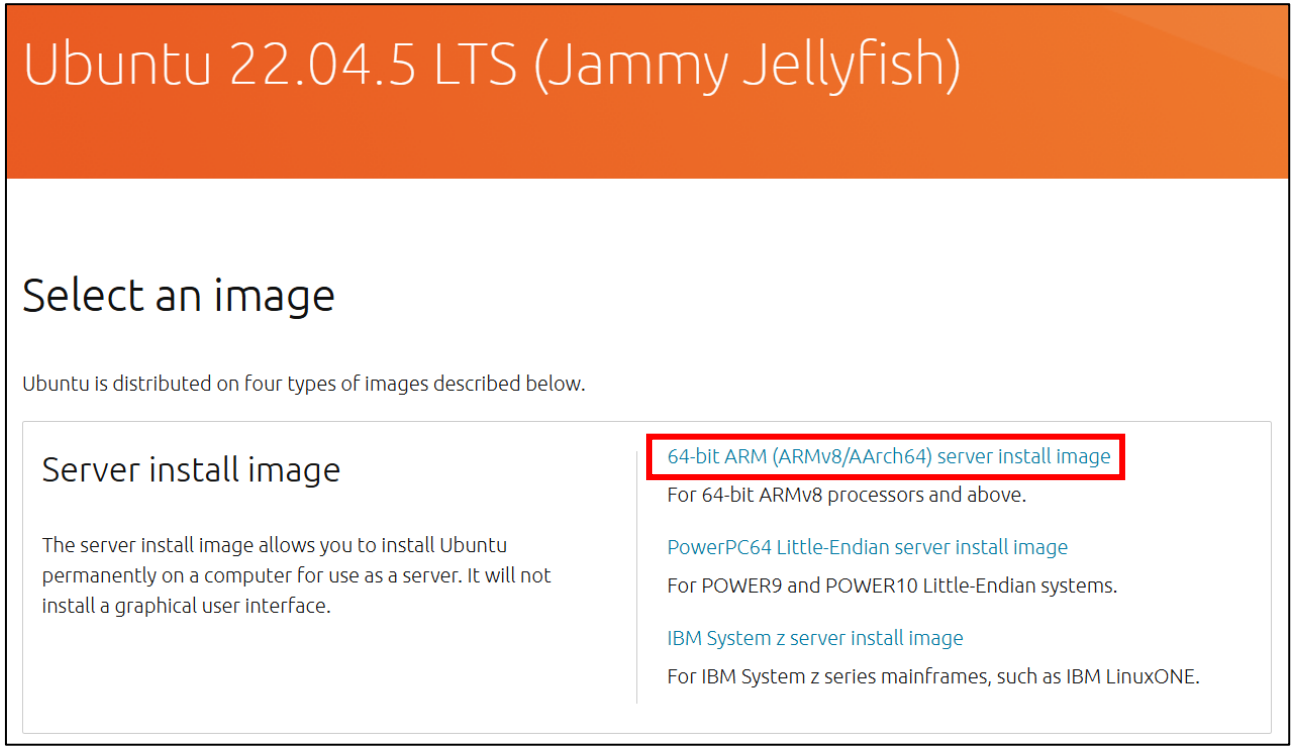


설치 후 UTM 초기화면

2. Mac (ARM)

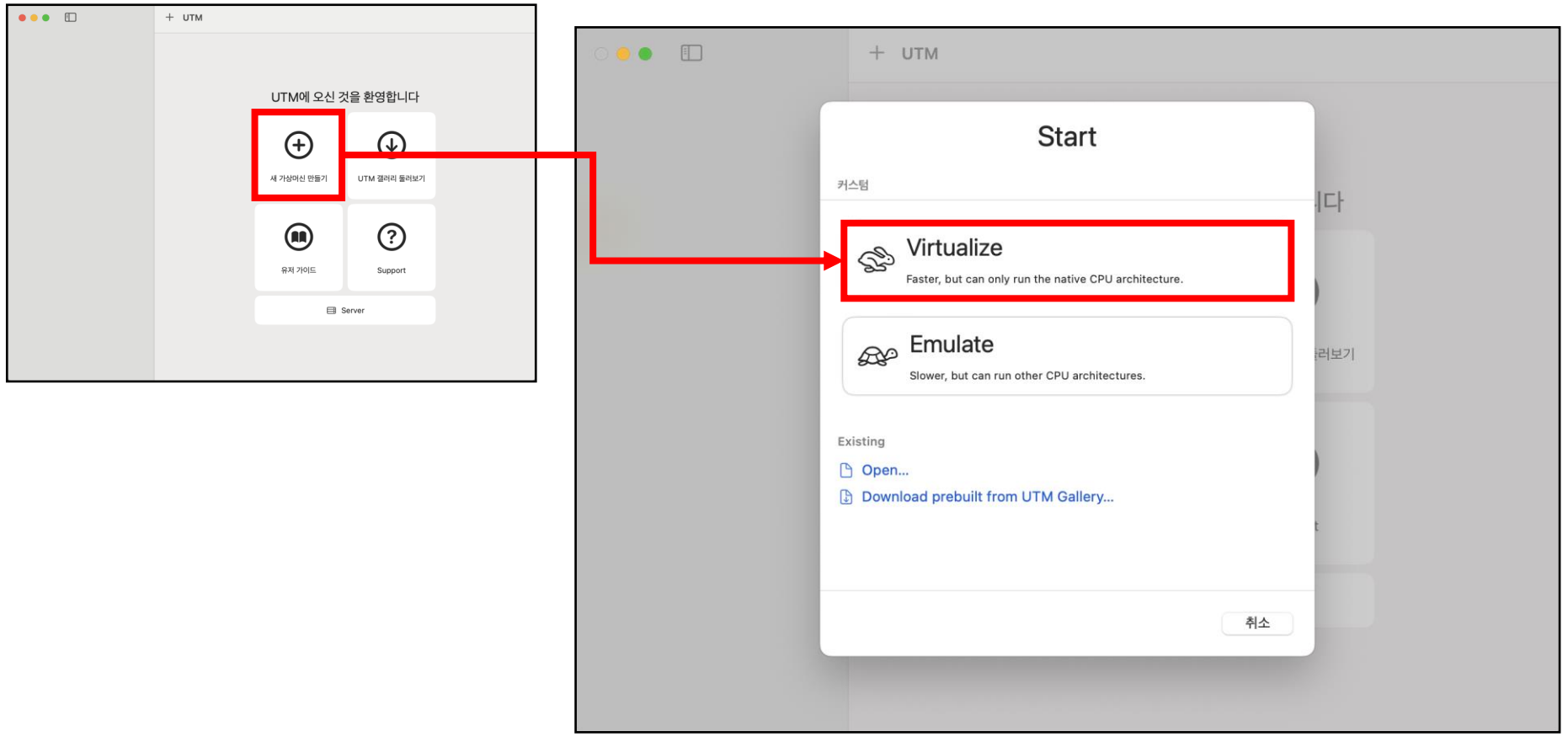
2) Ubuntu Desktop 22.04 이미지 다운로드 (ARM)

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



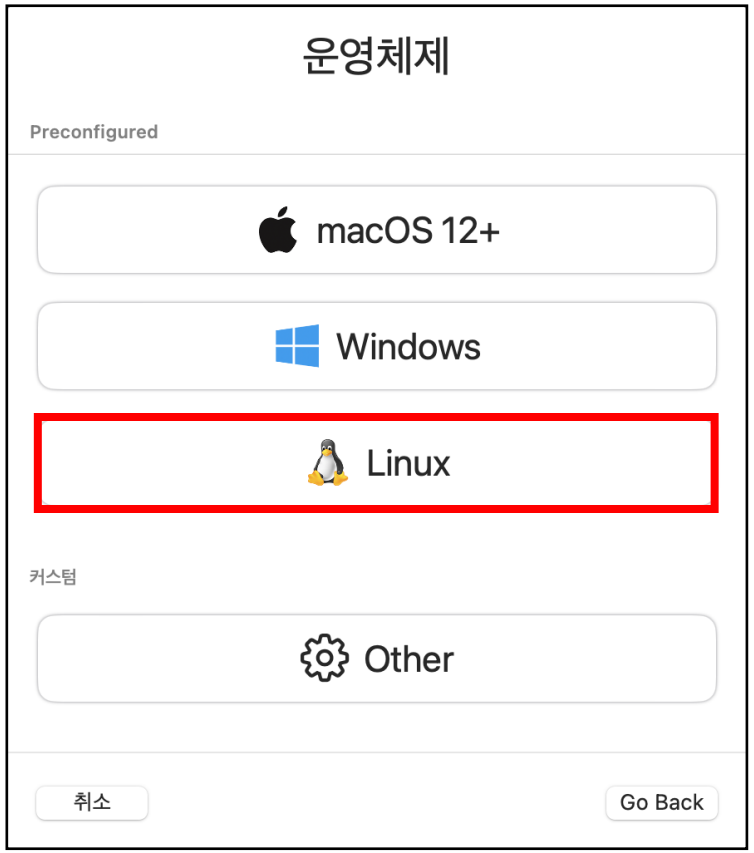
2. Mac (ARM)

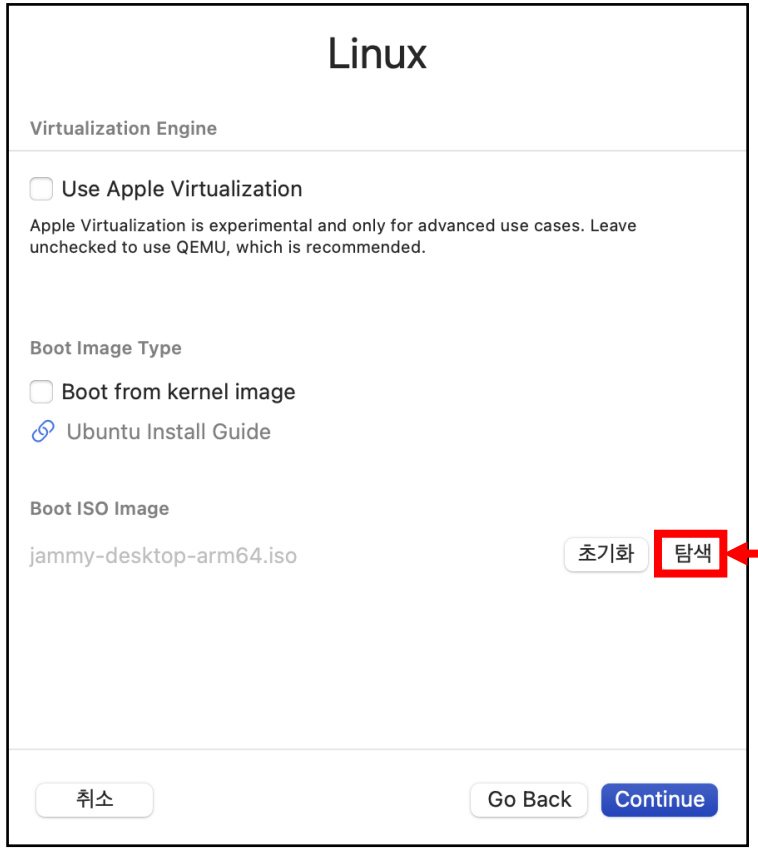
3) VM 생성



2. Mac (ARM)

3) VM 생성





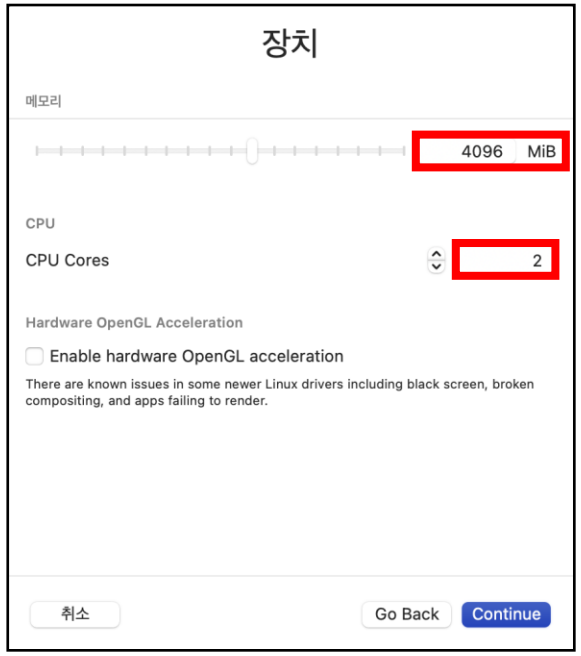
2)에서 다운받은 iso 이미지

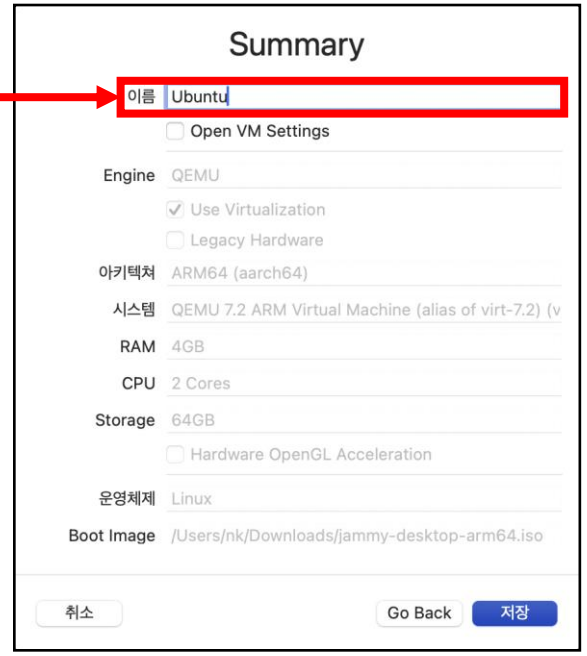
2. Mac (ARM)

3) VM 생성

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

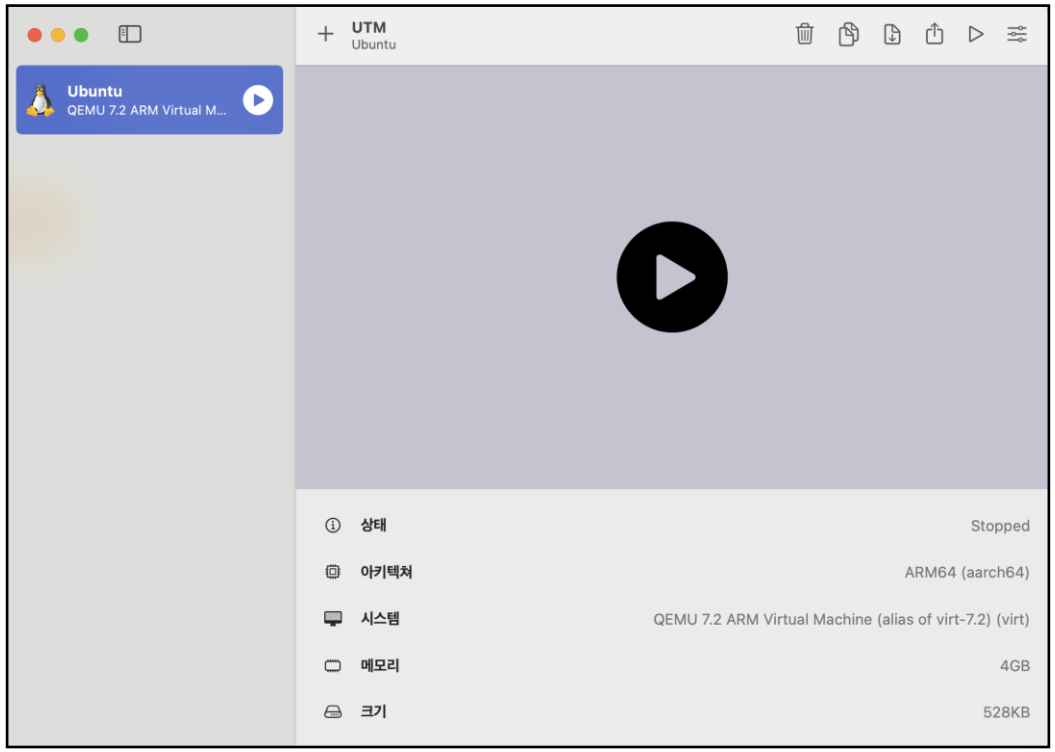
본인이 식별하기 쉬운 이름으로 설정





2. Mac (ARM)

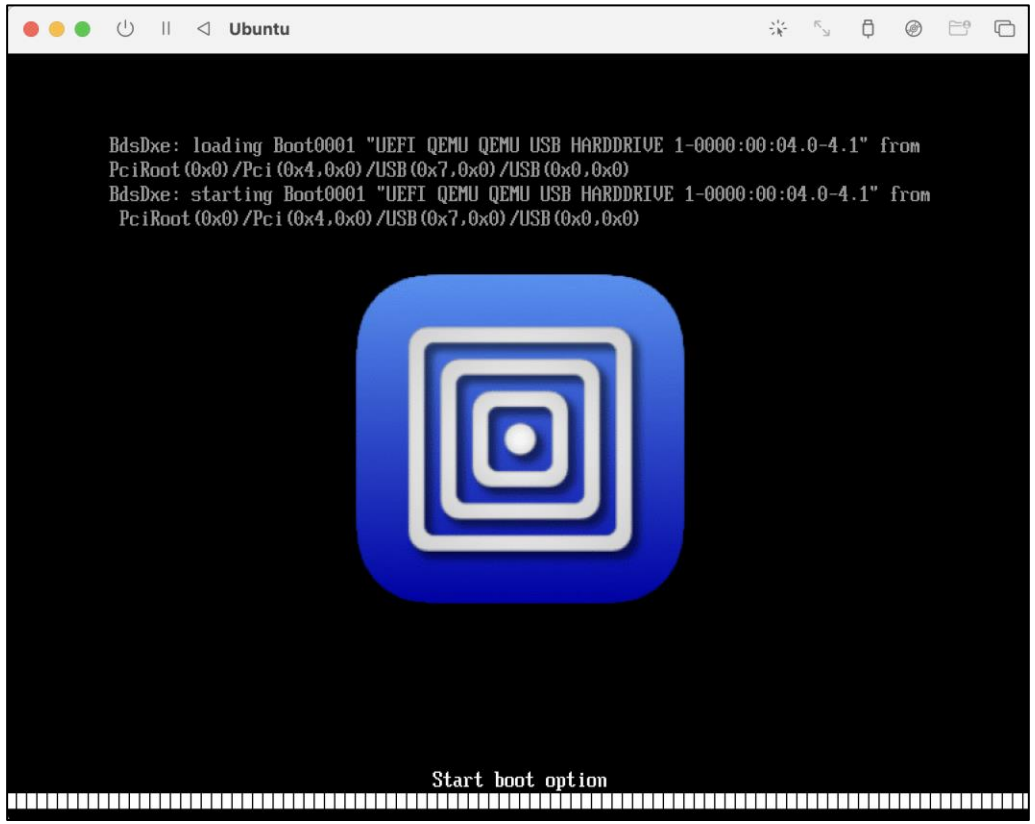
3) VM 생성 - 완료



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

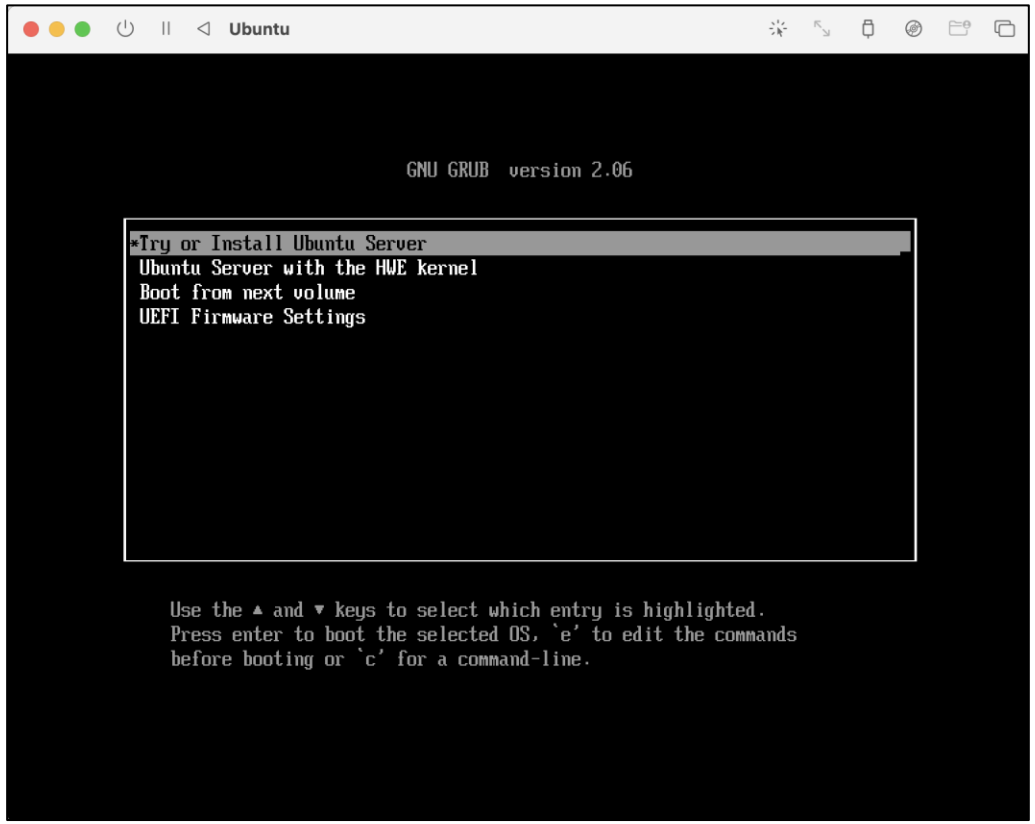
- 초기 VM 실행 화면



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

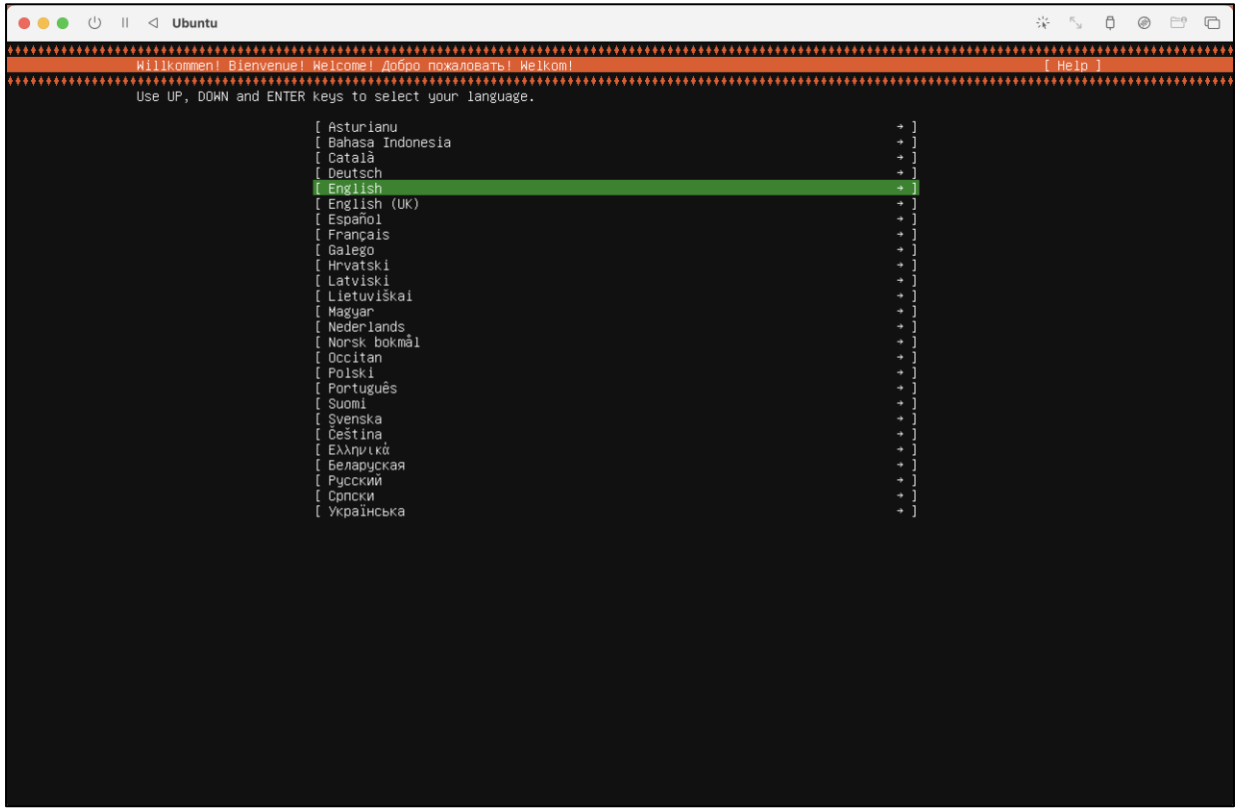
- 단계에 따라 설치 진행



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

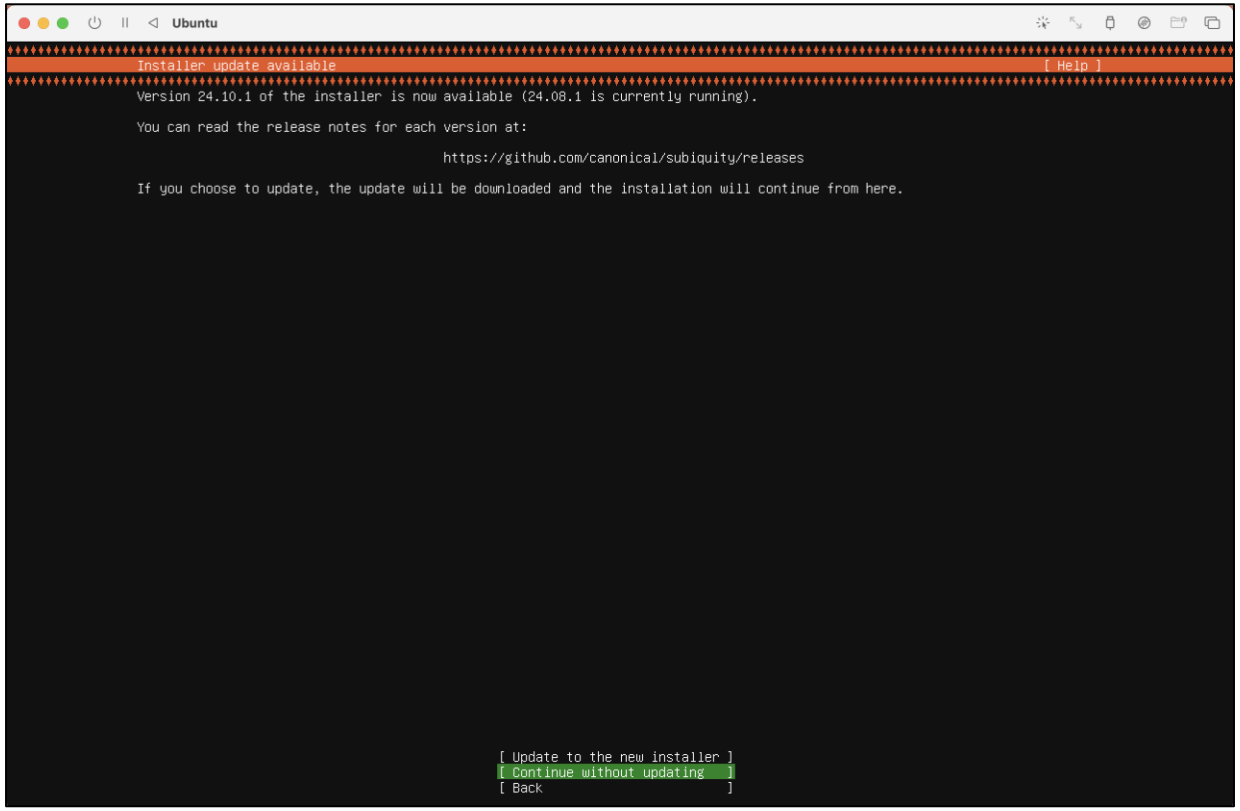
- 언어 설정: English



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

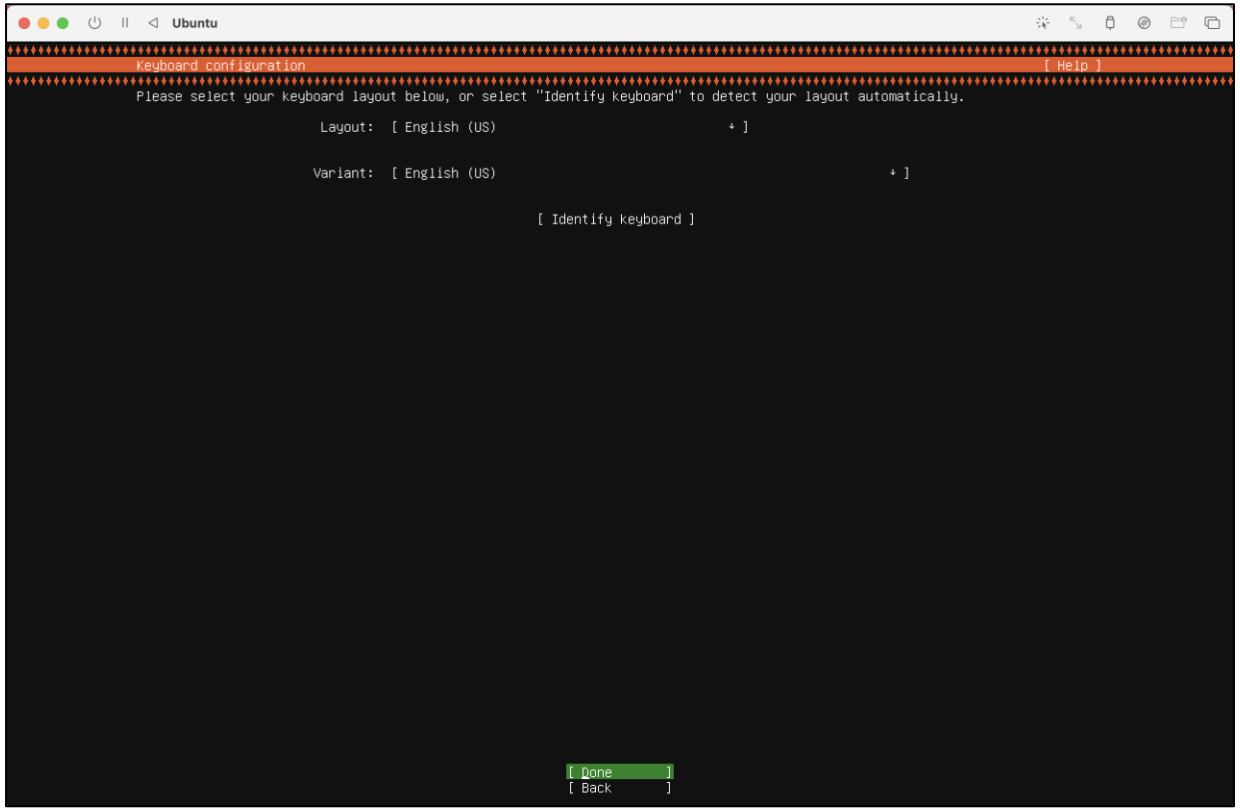
- 24버전으로 업데이트: 안함



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

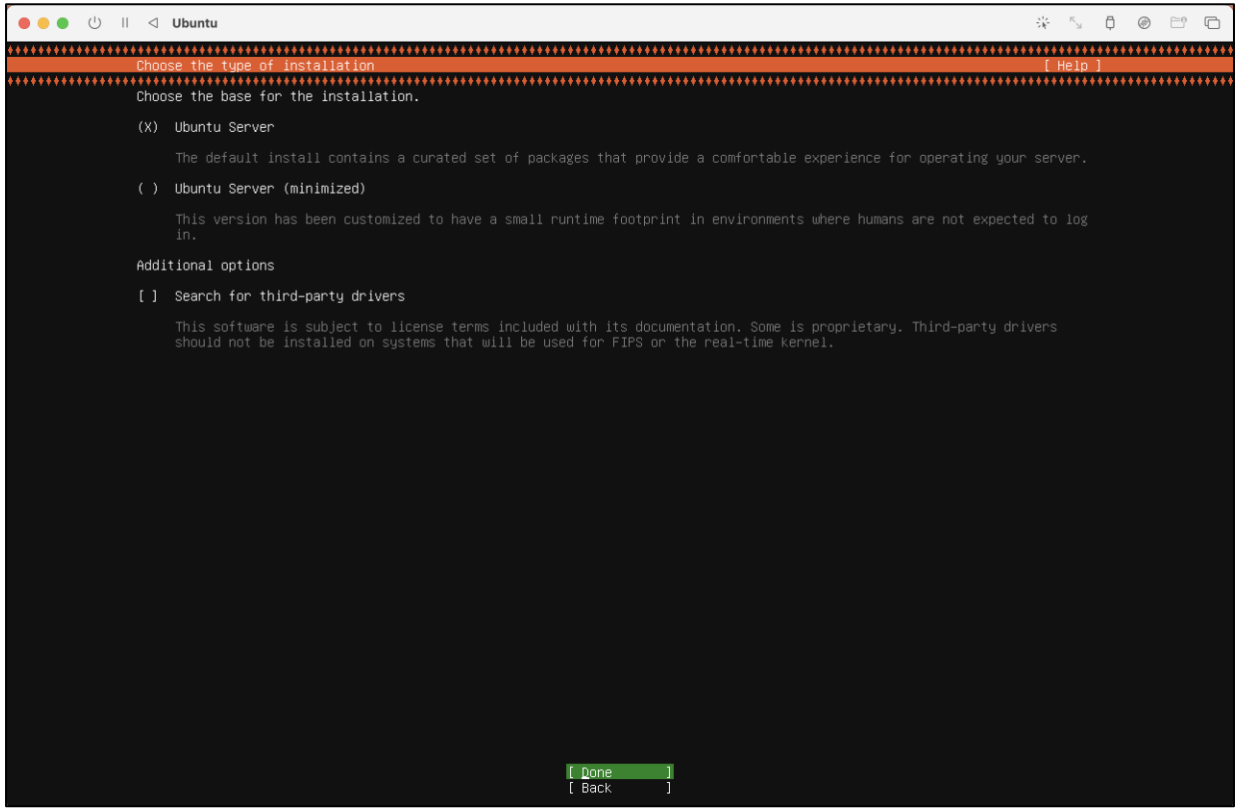
- 키보드 레이아웃: English



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

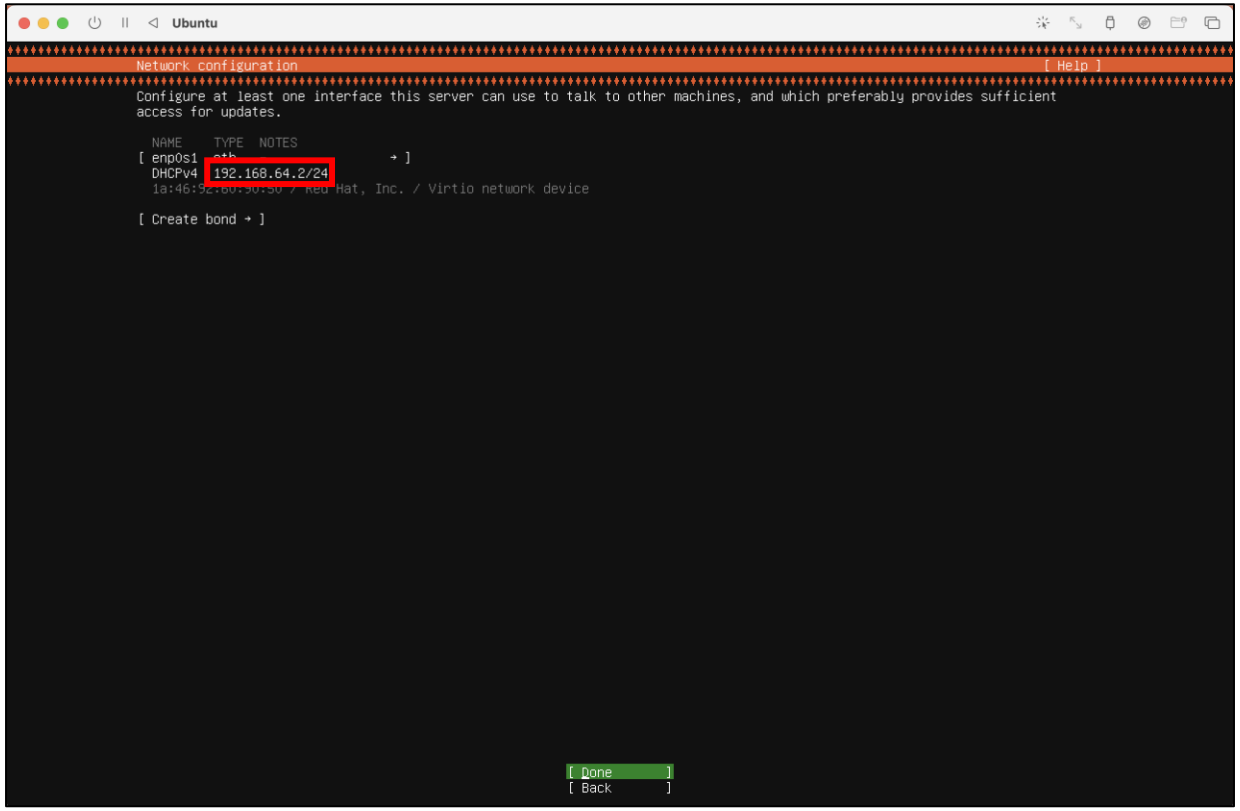
- Ubuntu 설치: Default



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

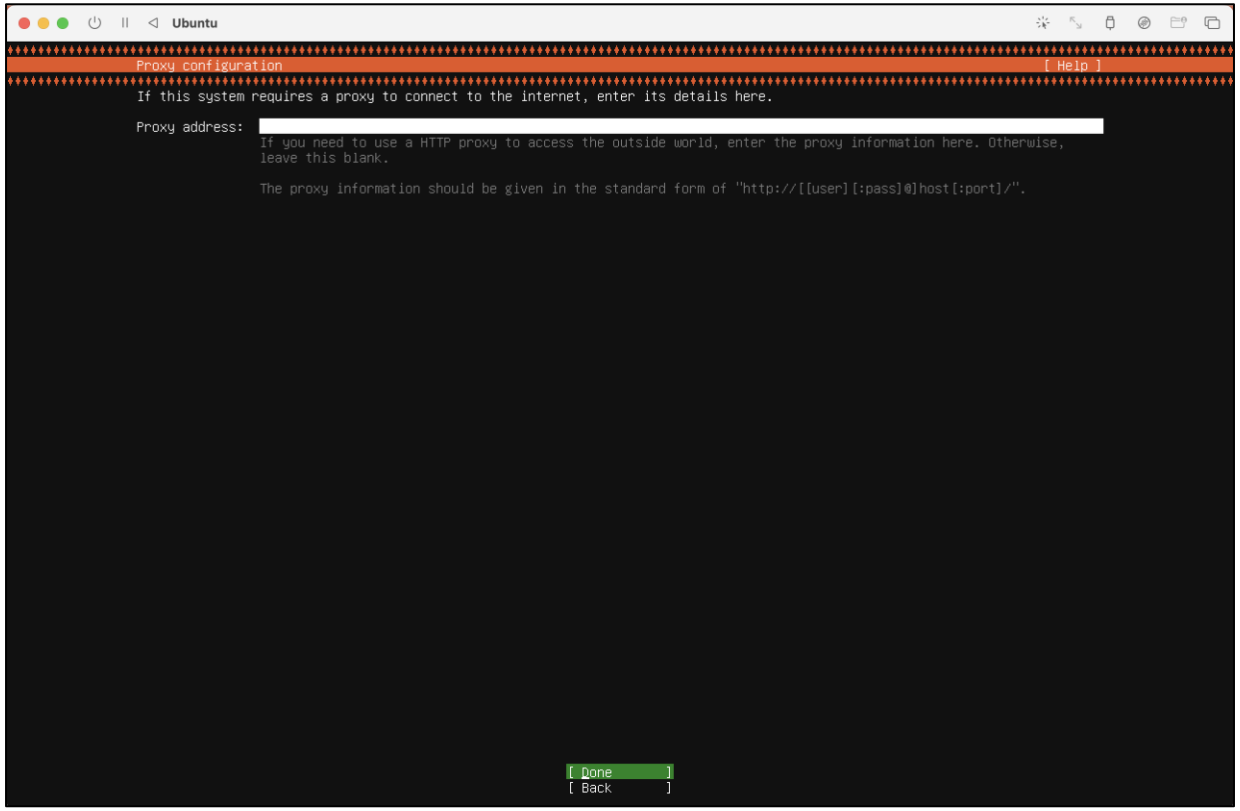
- 네트워크 설정: 그대로 진행, SSH를 사용할 것이라면 IP 주소 기억할 것



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

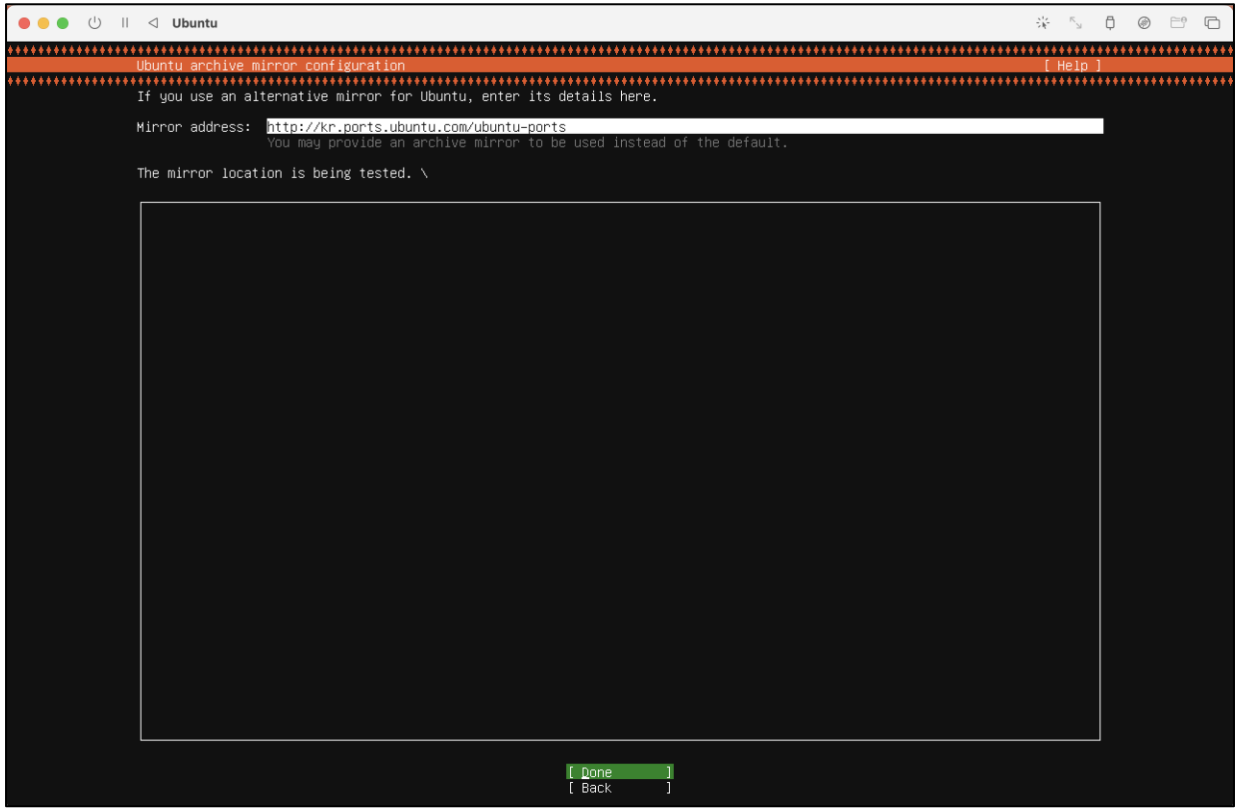
- Proxy 주소: 그대로 진행



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

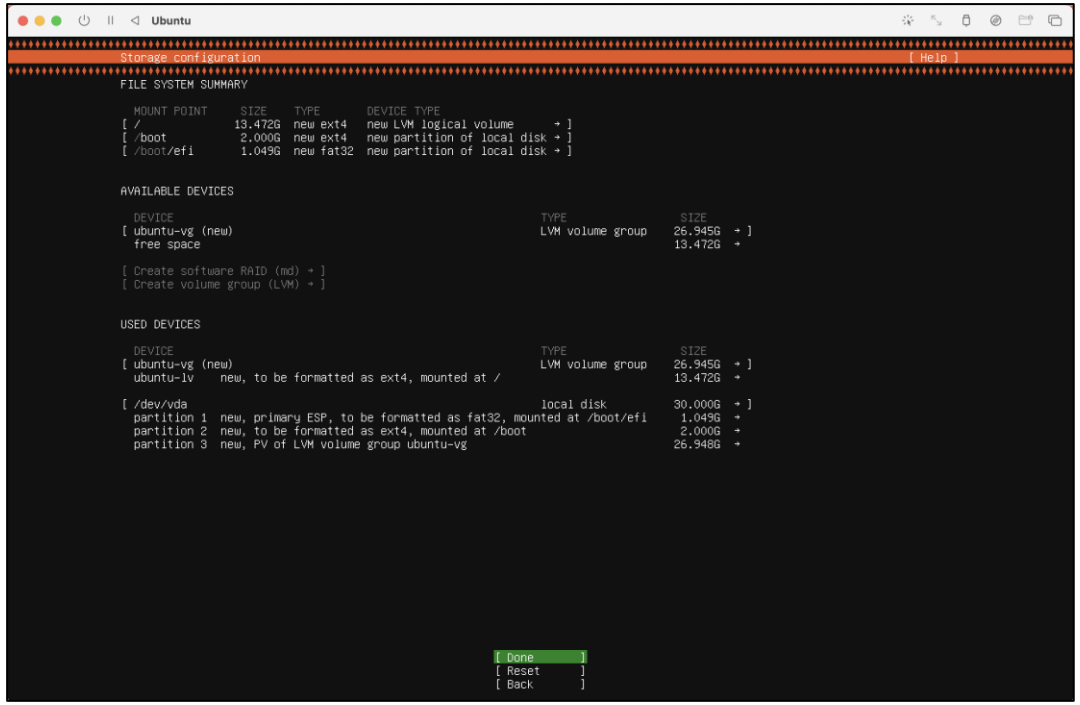
- Mirror 서버: 그대로 진행

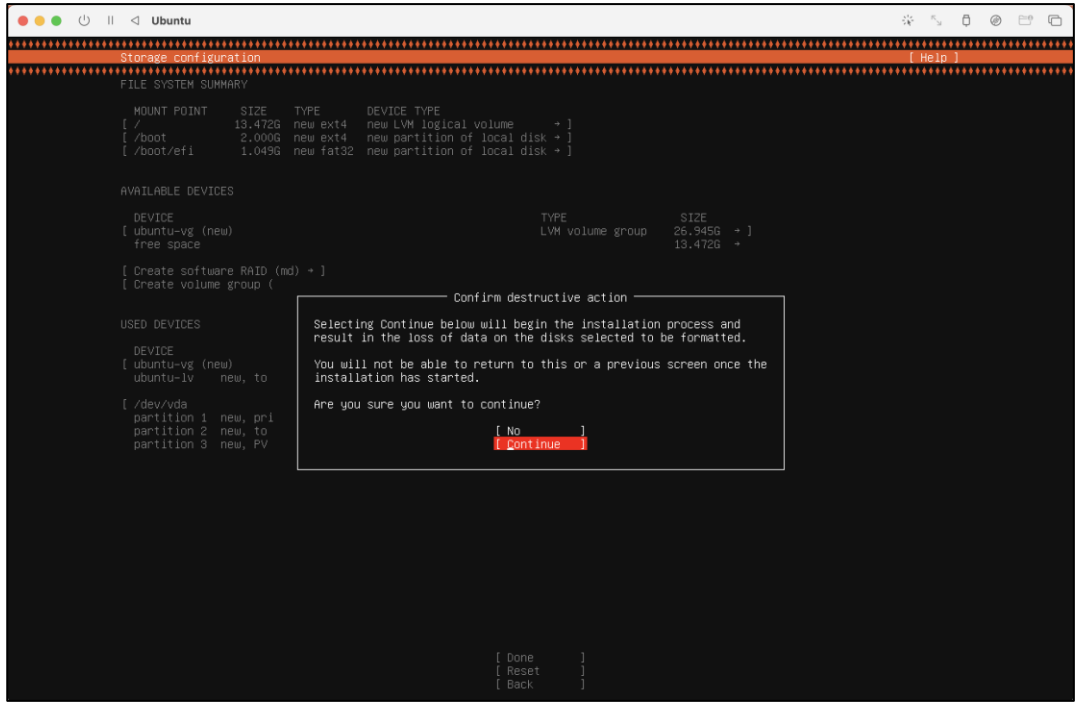


2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

- 파일 시스템 설정: 그대로 진행

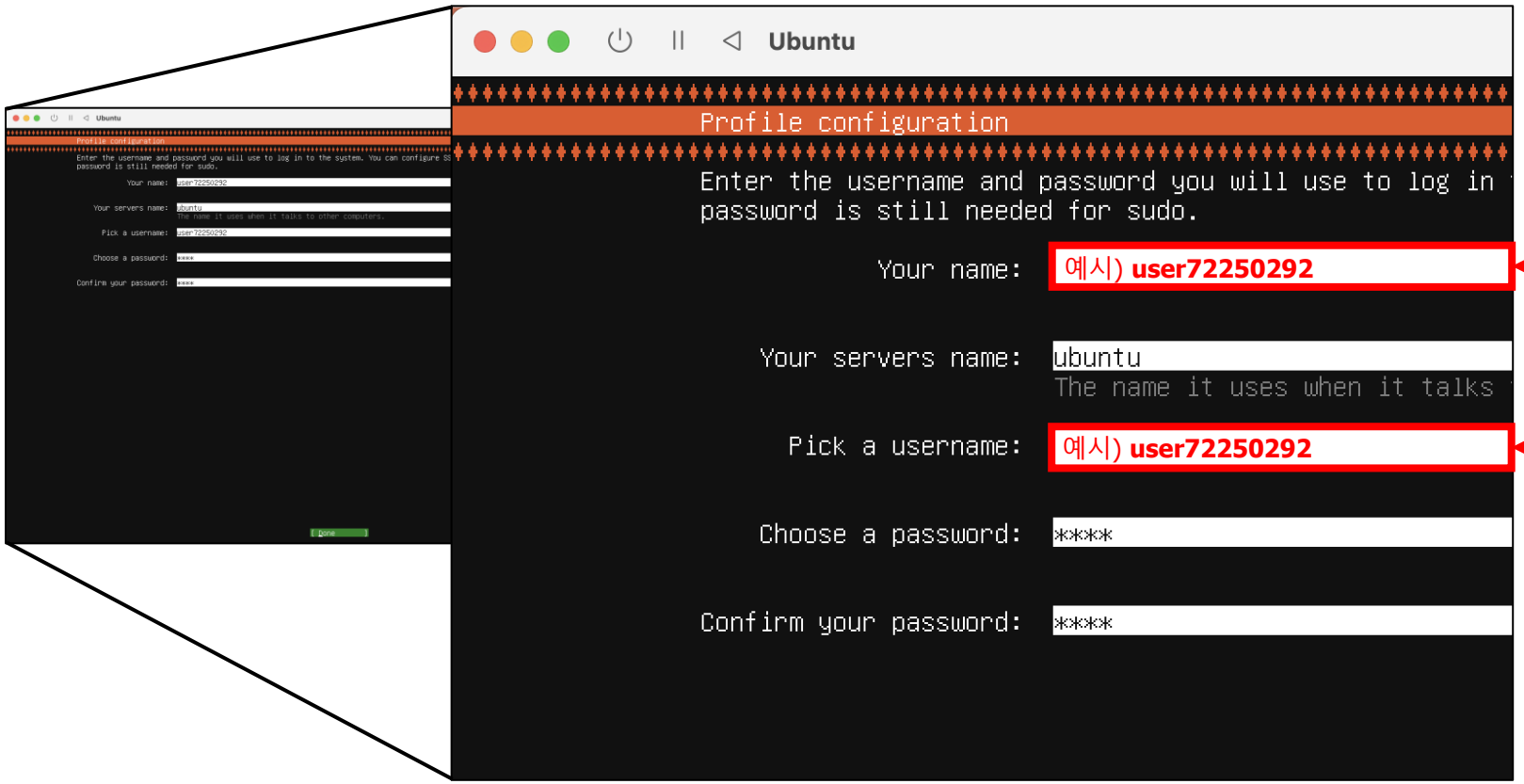




2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

- 계정 설정

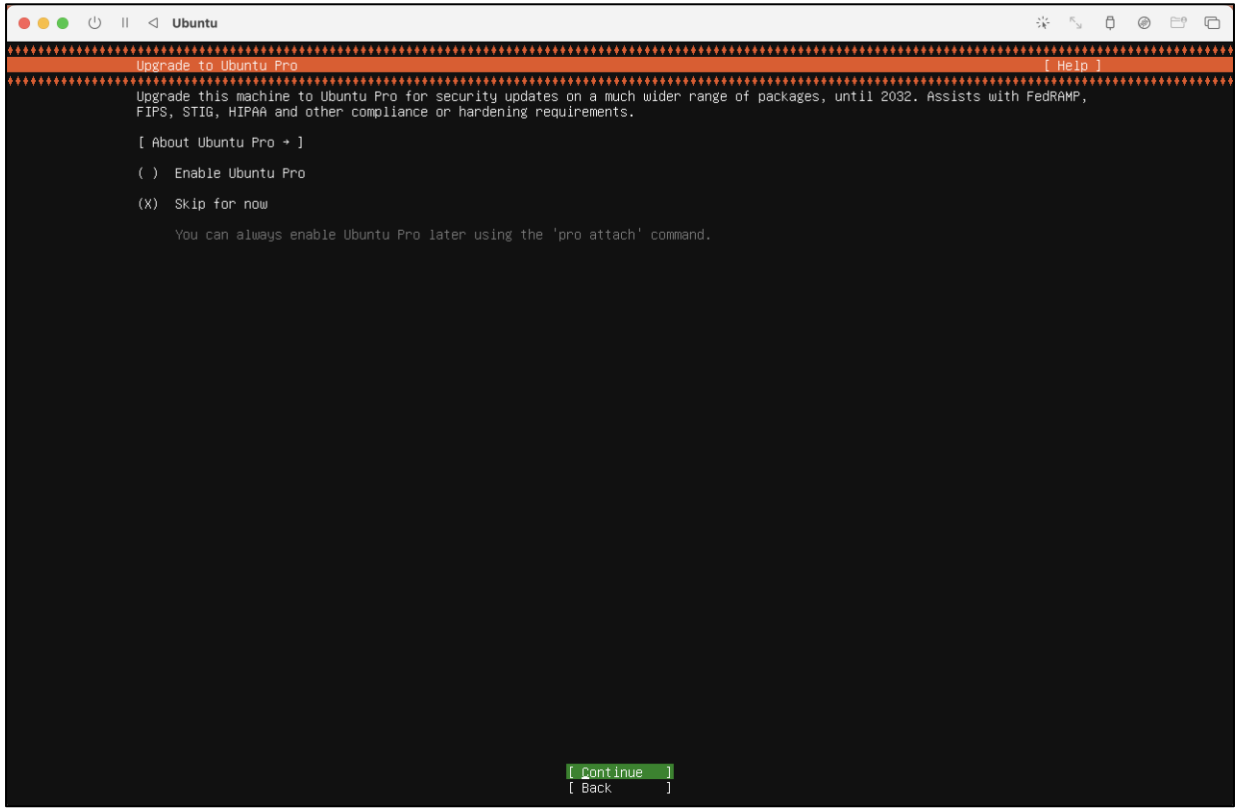


user + 학번으로 설정

2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

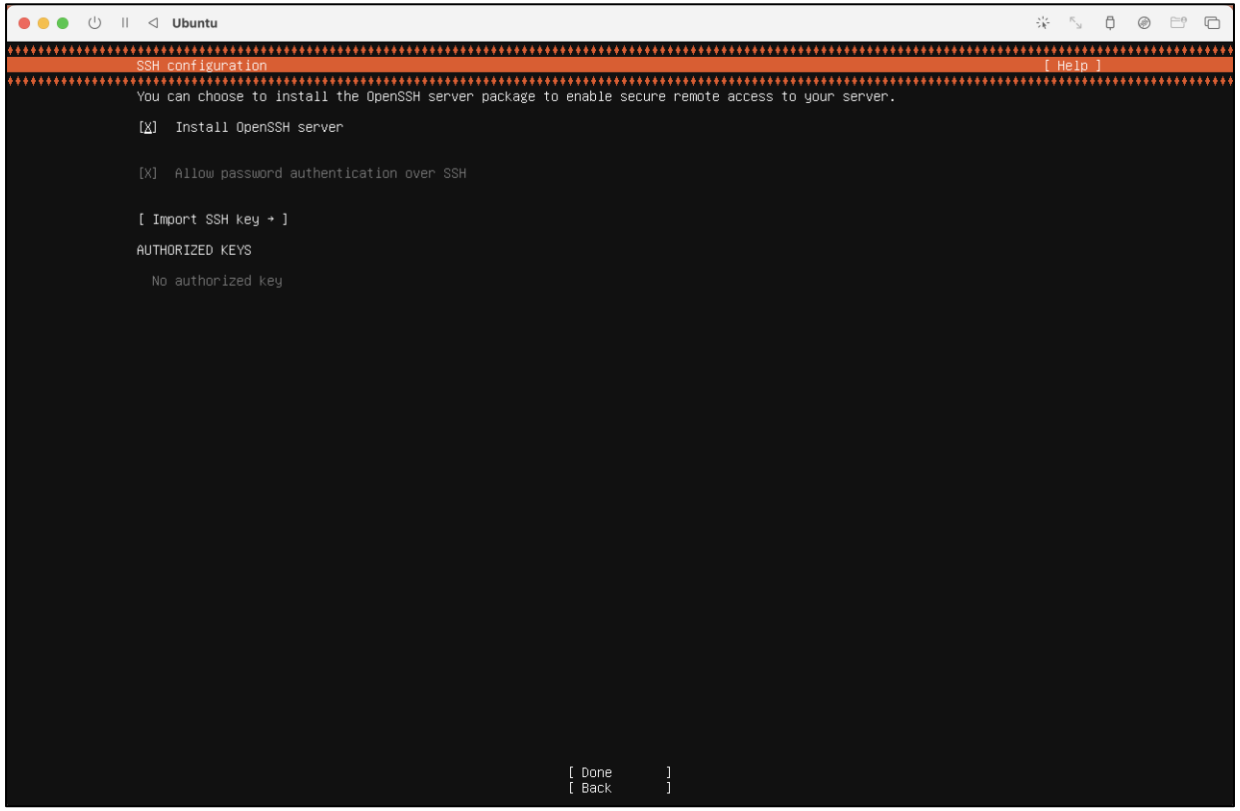
- Ubuntu pro로 업데이트: Skip



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

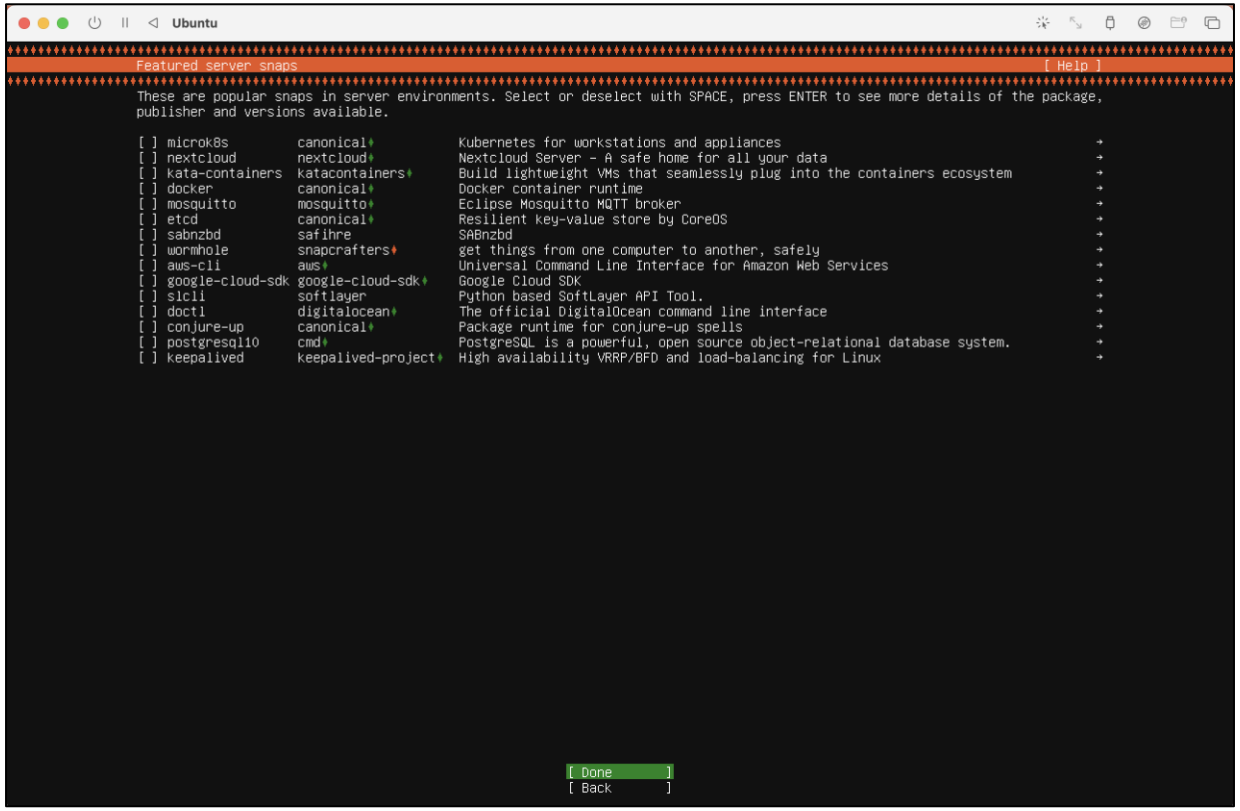
- SSH 서버 설정: SSH를 사용할 것이라면 체크 아니면 Skip



2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

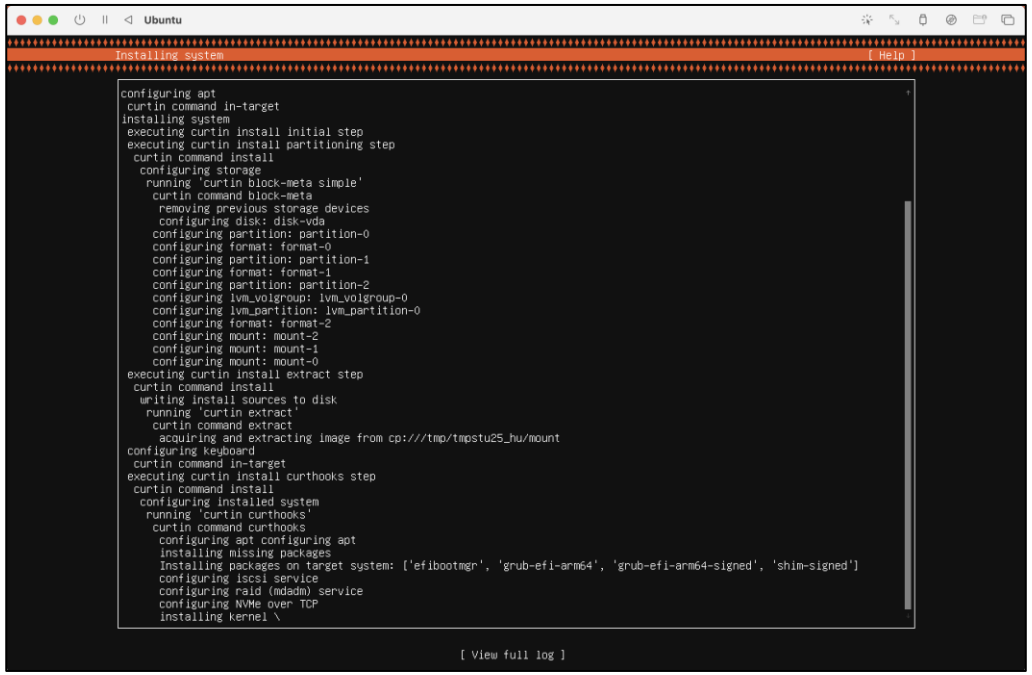
- 서버 snap 패키지 설치: Skip

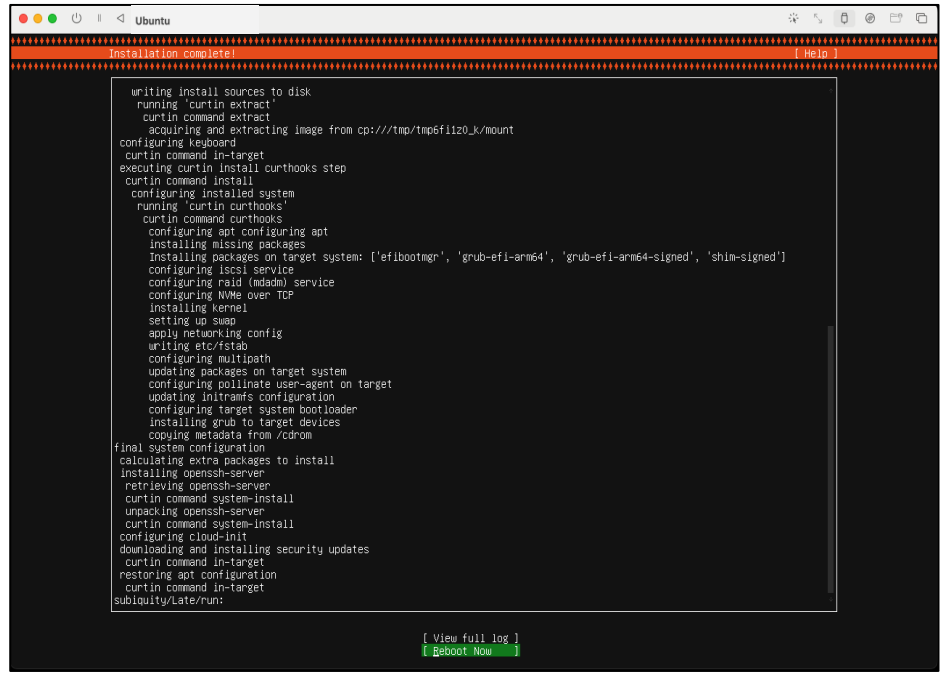


2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

- Ubuntu 설치 진행 화면, 완료되면 reboot

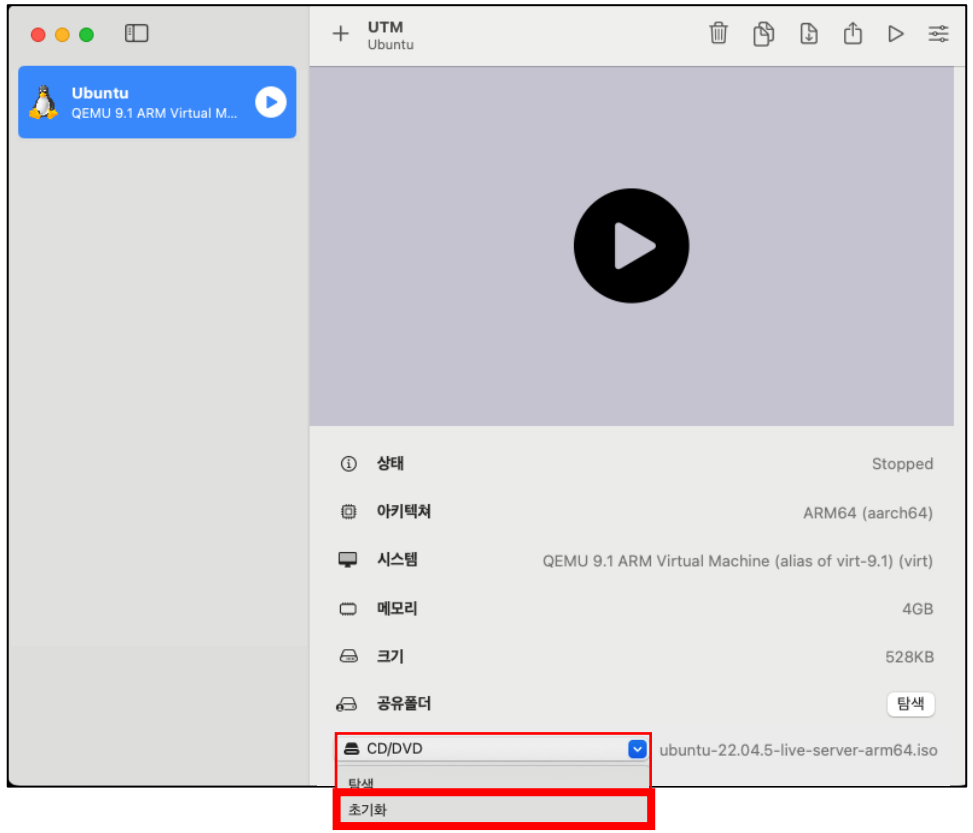




2. Mac (ARM)

4) 초기 설정 – Ubuntu 설치

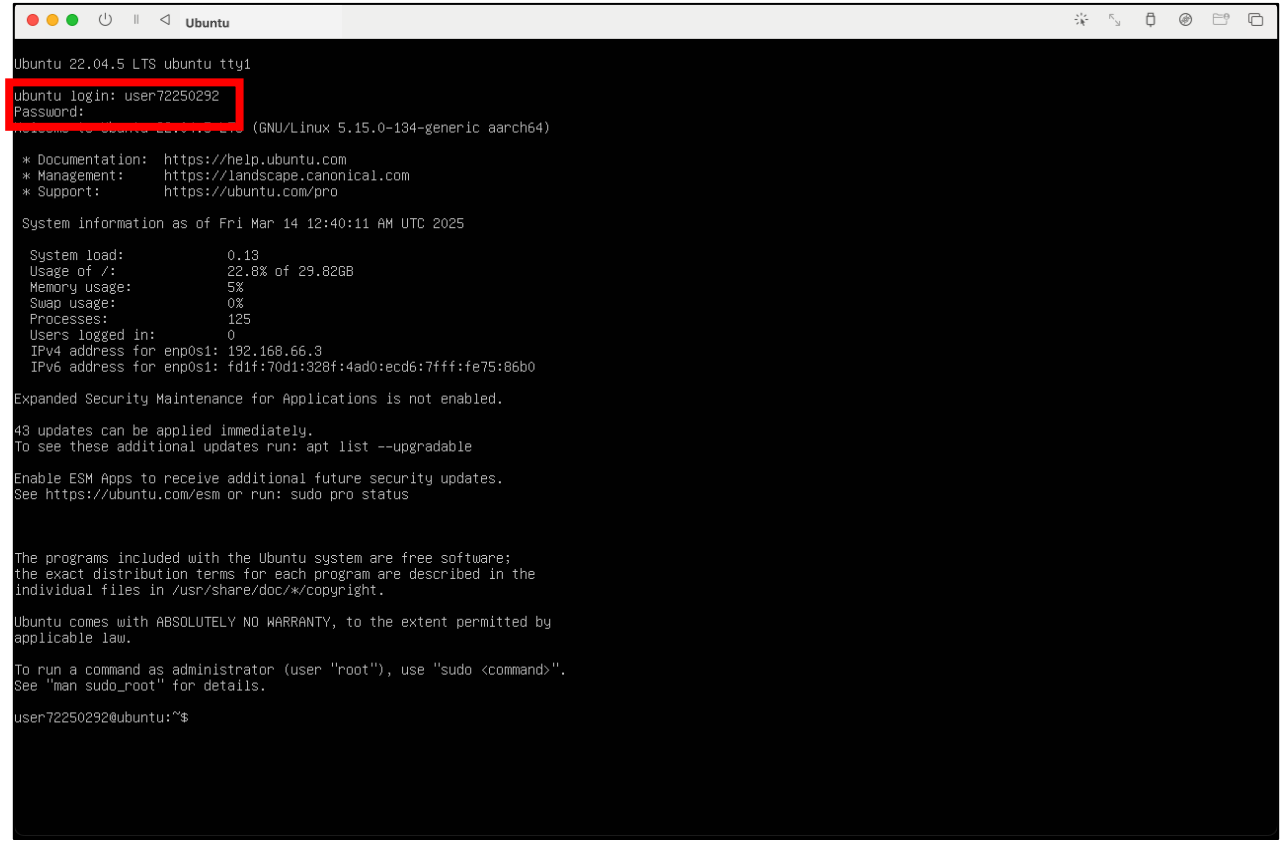
- 검은 화면이 뜬다면 VM 종료, CD/DVD 초기화 후 다시 시작



2. Mac (ARM)

4) 초기 설정 – 로그인

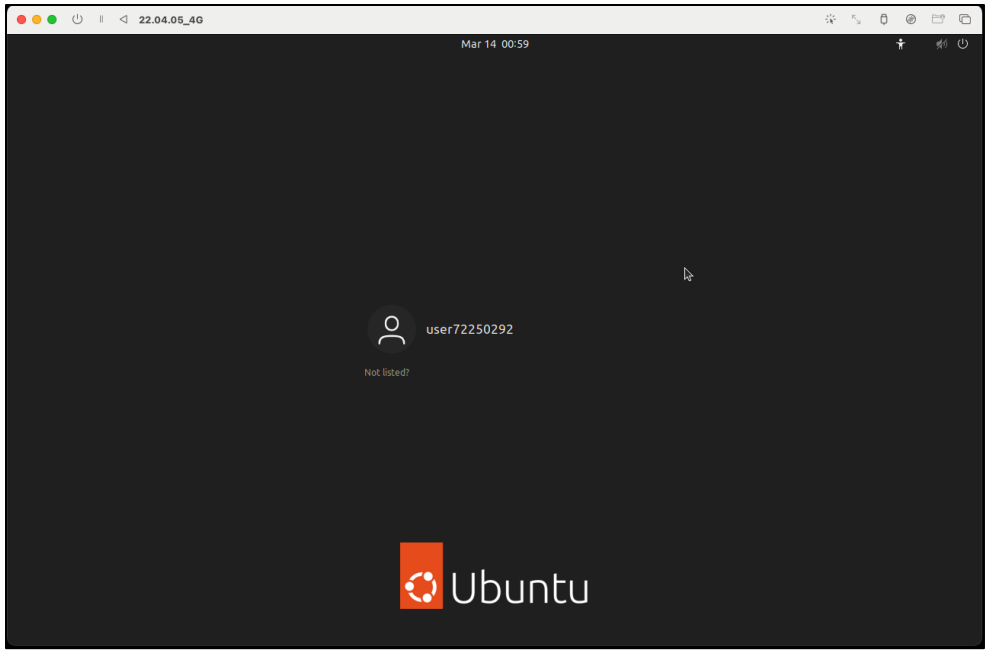
- {username} + 비밀번호 입력



2. Mac (ARM)

5) Optional – GUI 환경으로 변경

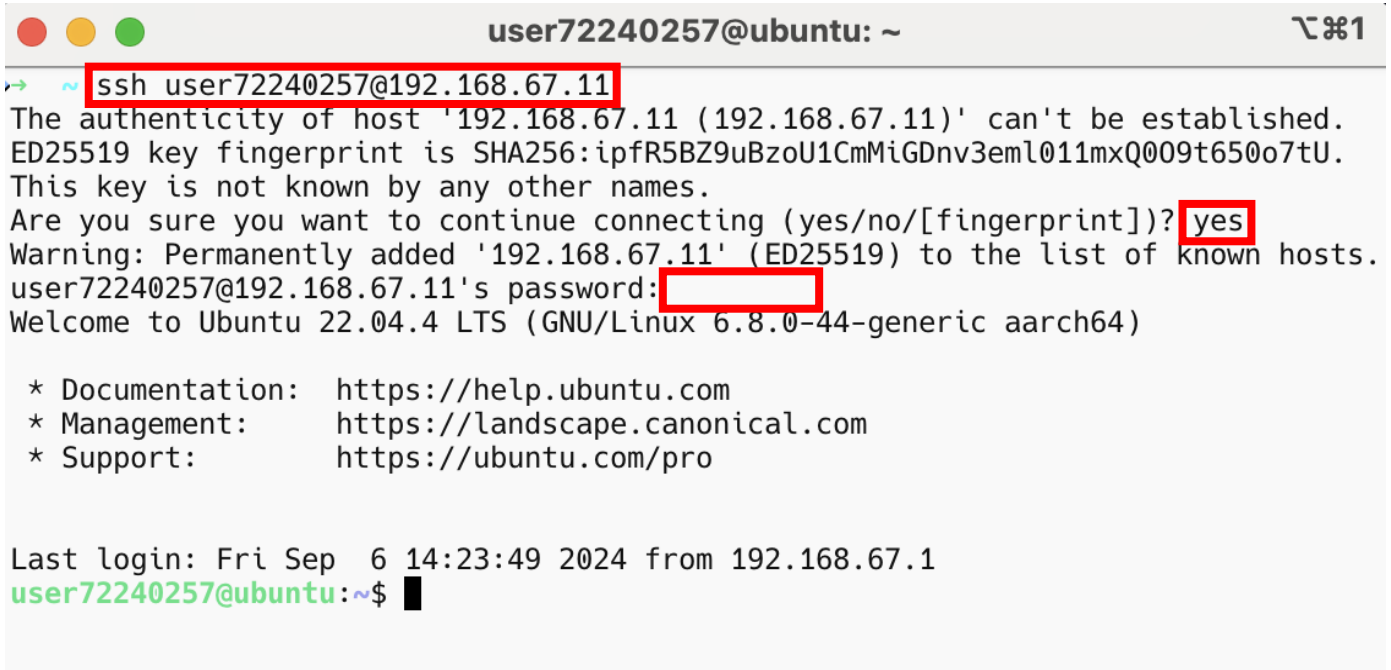
- $ sudo apt install ubuntu-desktop
- 설치 완료 후 $ sudo reboot
- reboot 후 다음 화면이 나온다면 성공



2. Mac (ARM)

5) Optional – SSH 설정

- Host(macOS)에서 터미널 실행 (VM은 실행 중인 상태)
- ~ ssh {username}@{IP address}
  - 39p에서 알아낸 IP 주소 또는 $ ip a로 알아낸 IP 주소



Lab 0. Reading Assignment

- 보고서 제출 과제
  - OSTEP **2장 요약**
    - 링크: https://pages.cs.wisc.edu/~remzi/OSTEP/intro.pdf
  - "The Unix Time-Sharing System" **논문 요약**
    - 링크: https://dsf.Berkeley.edu/cs262/unix.pdf
  - OSTEP 2장과 Unix 논문의 **내용 비교**
  - 이 강의를 수강하면서 달성하고 싶은 **개인적인 목표**
  - **추가 점수**: OSTEP 2장의 예제 코드를 리눅스 환경에서 실행하고 결과 제출

Lab 0. Reading Assignment

- 이러닝 과제 칸에 pdf 형식으로 제출
- 제목: **os_lab0_학번_이름.pdf**
- 기한: **2025.03.30 23:59**까지
- 분량은 자유
- 한국어로 작성 가능

Q&A?

2025.03.17

T.A. 김보승

bskim1102@dankook.ac.kr