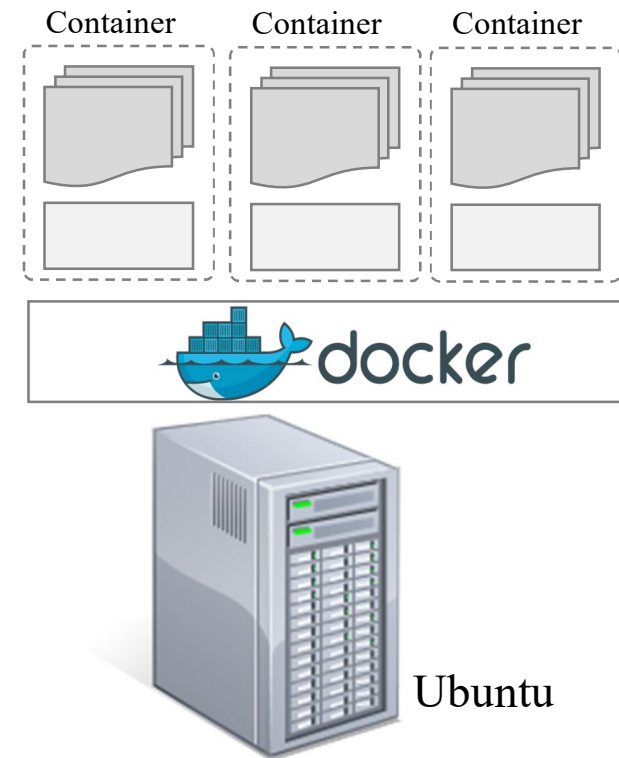
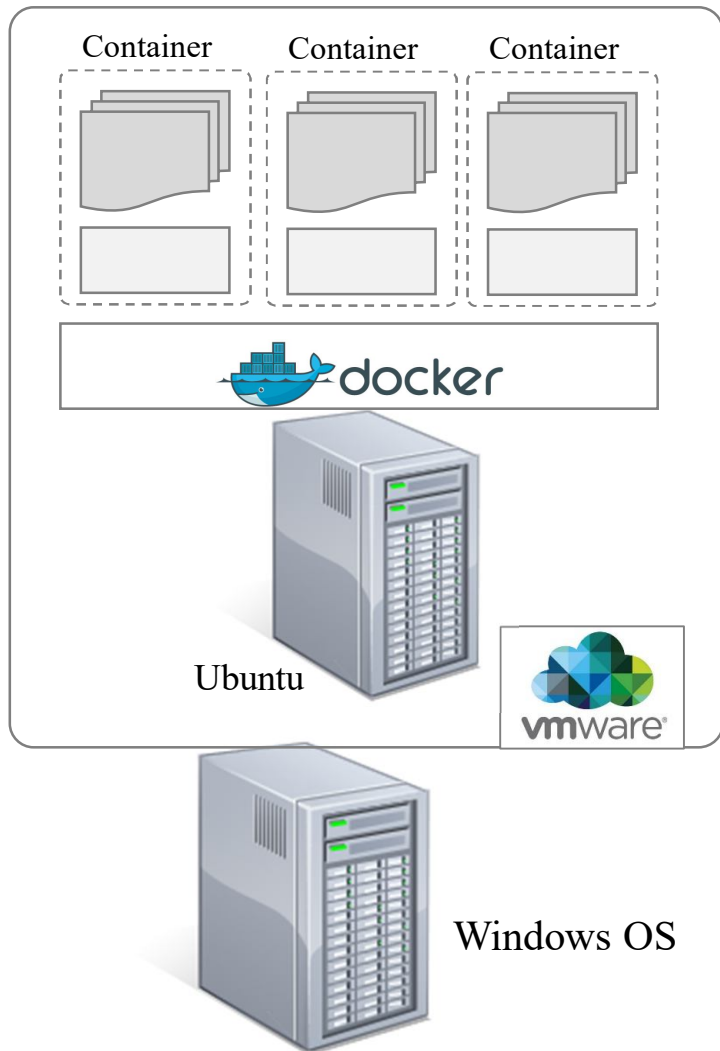
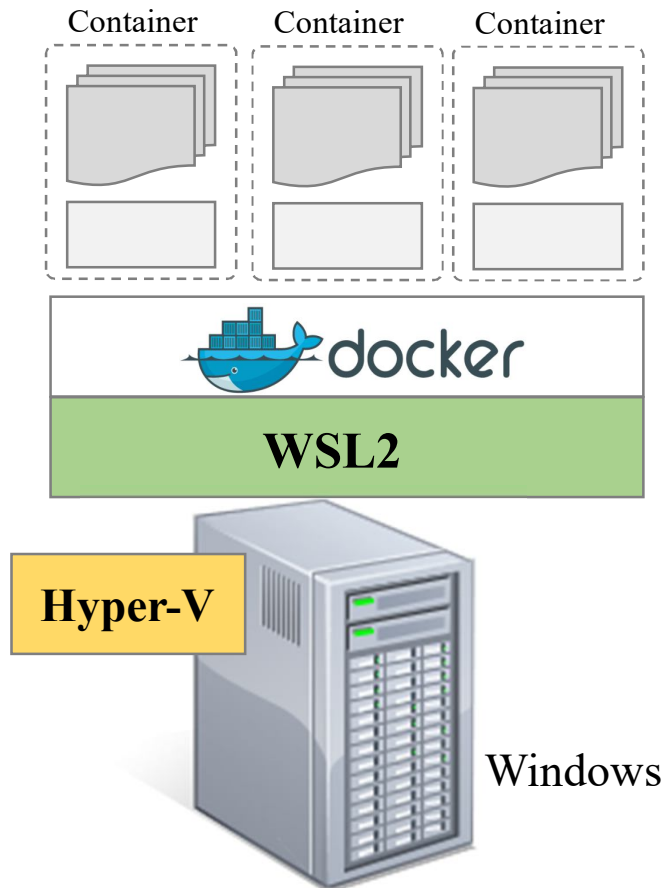


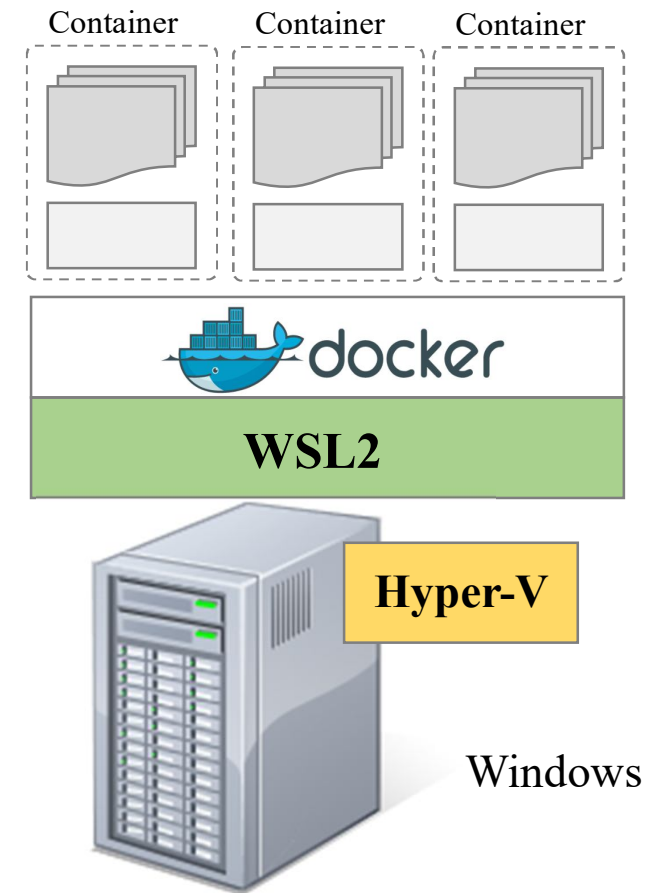
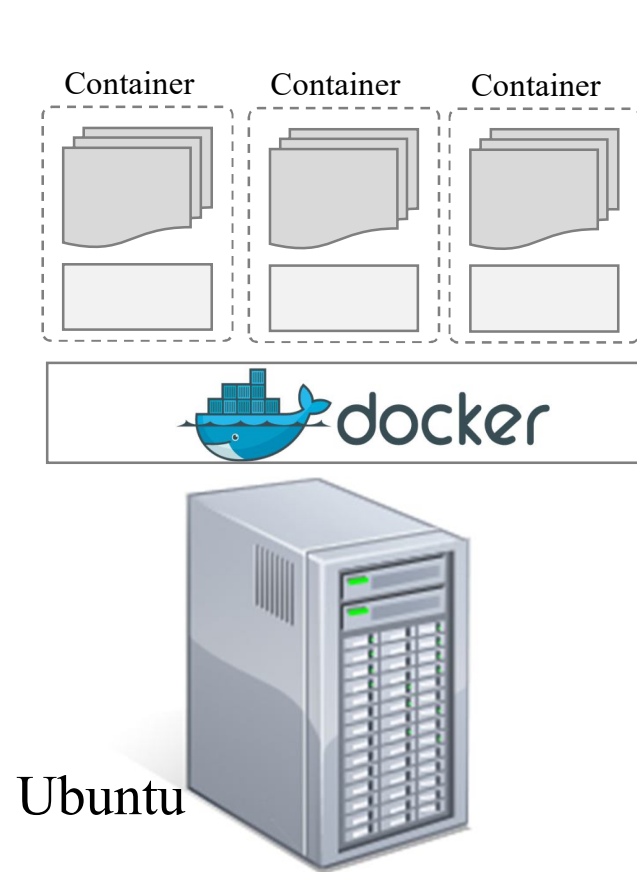
3. Docker 실습환경 구성

Linux 환경에서 Docker 설치



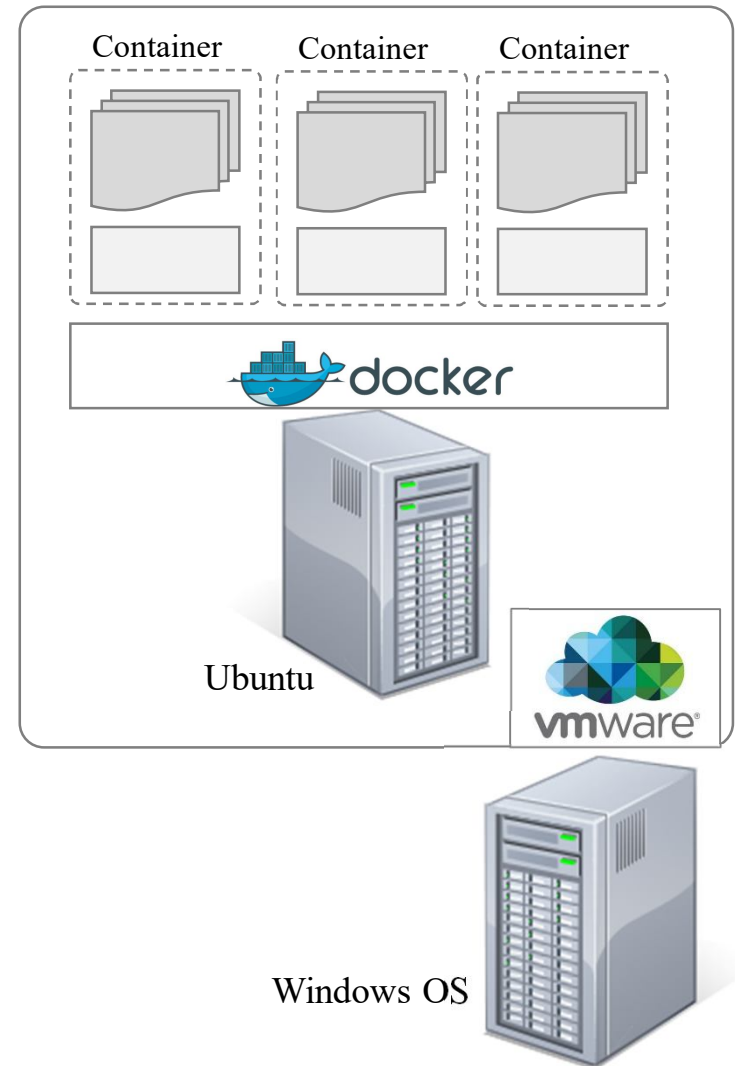
Windows 환경에서 Docker 설치





Linux 환경에서 Docker 설치

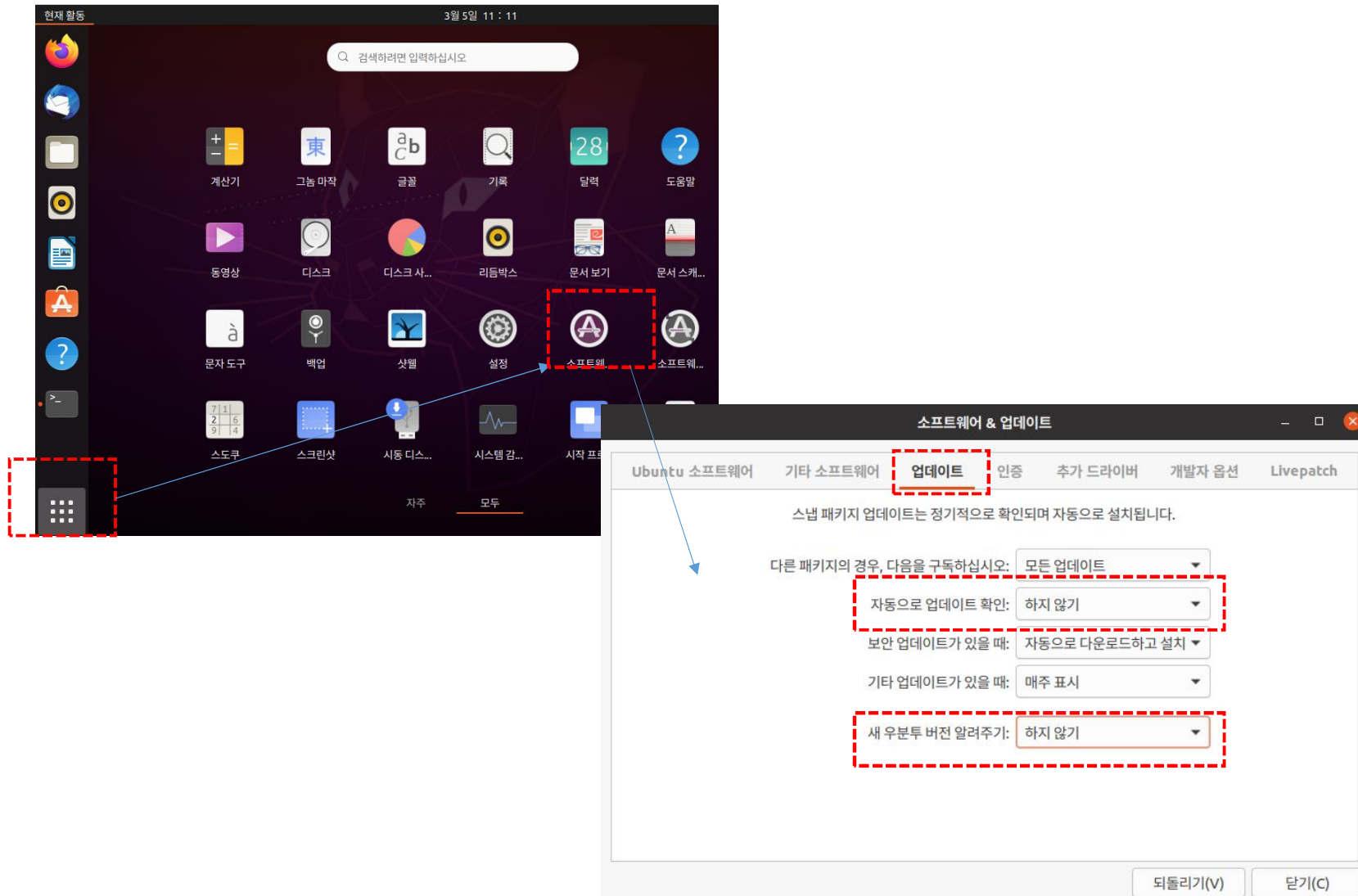
- 1) VMWare Install
- 2) Ubuntu Install
- 3) Docker Install



Ubuntu 환경 설정

- 소프트웨어 업데이트 끄기
- 패키지 업데이트
- 네트워크 툴 설치
- SSH 프로그램 설치

1 소프트웨어 업데이트 기능 끄기



② 패키지 업데이트

#apt update

```
root@ubuntu:/etc/apt# apt update
받기:1 http://mirror.kakao.com/ubuntu focal InRelease [265 kB]
받기:2 http://mirror.kakao.com/ubuntu focal/main i386 Packages [718 kB]
받기:3 http://mirror.kakao.com/ubuntu focal/main amd64 Packages [970 kB]
받기:4 http://mirror.kakao.com/ubuntu focal/main Translation-ko [204 kB]
받기:5 http://mirror.kakao.com/ubuntu focal/main Translation-en [506 kB]
받기:6 http://mirror.kakao.com/ubuntu focal/main amd64 DEP-11 Metadata [494 kB]
받기:7 http://mirror.kakao.com/ubuntu focal/main amd64 c-n-f Metadata [29.5 kB]
받기:8 http://mirror.kakao.com/ubuntu focal/universe amd64 Packages [8,628 kB]
받기:9 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
받기:10 http://mirror.kakao.com/ubuntu focal/universe i386 Packages [4,642 kB]
받기:11 http://mirror.kakao.com/ubuntu focal/universe Translation-ko [703 kB]
받기:12 http://mirror.kakao.com/ubuntu focal/universe Translation-en [5,124 kB]
받기:13 http://mirror.kakao.com/ubuntu focal/universe amd64 DEP-11 Metadata [3,603 kB]
받기:14 http://mirror.kakao.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
```


Repository를 이용한 Application 설치

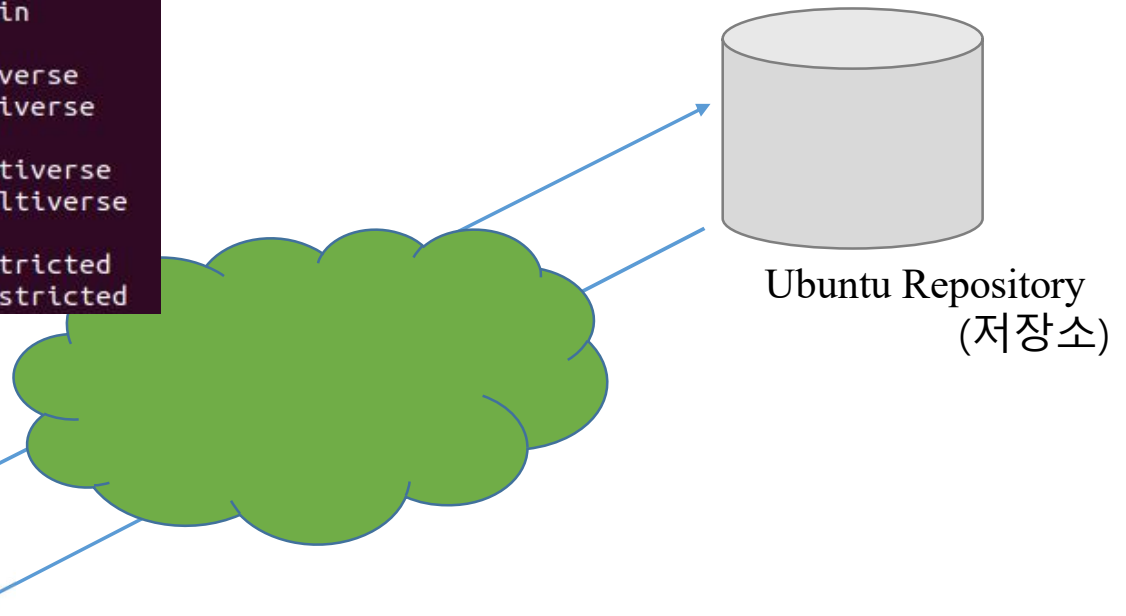
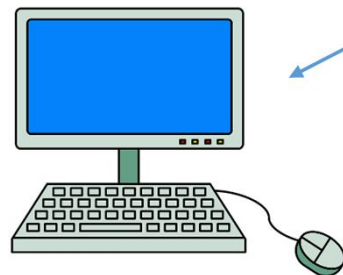
```
root@ubuntu:/etc/apt# ls
apt.conf.d  preferences.d  sources.list.bak  trusted.gpg
auth.conf.d  sources.list    sources.list.d
root@ubuntu:/etc/apt# cat sources.list
deb http://ftp.daumkakao.com/ubuntu/ focal main
deb http://archive.ubuntu.com/ubuntu/ focal main

deb http://ftp.daumkakao.com/ubuntu/ focal universe
deb http://archive.ubuntu.com/ubuntu/ focal universe

deb http://ftp.daumkakao.com/ubuntu/ focal multiverse
deb http://archive.ubuntu.com/ubuntu/ focal multiverse

deb http://ftp.daumkakao.com/ubuntu/ focal restricted
deb http://archive.ubuntu.com/ubuntu/ focal restricted
```

/etc/apt/sources.list



3 네트워크 툴 설치

#apt -y install net-tools

```
root@ubuntu:/etc/apt# apt -y install net-tools
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다
상태 정보를 읽는 중입니다... 완료
다음 새 패키지를 설치할 것입니다:
  net-tools
0개 업그레이드, 1개 새로 설치, 0개 제거 및 0개 업그레이드 안 함.
196 k바이트 아카이브를 받아야 합니다.
이 작업 후 864 k바이트의 디스크 공간을 더 사용하게 됩니다.
받기:1 http://mirror.kakao.com/ubuntu focal/main amd64 net-tools amd64 1.60+git20
내려받기 196 k바이트, 소요시간 0초 (2,017 k바이트/초)
Selecting previously unselected package net-tools.
(데이터베이스 읽는중 ...현재 183958개의 파일과 디렉터리가 설치되어 있습니다.)
Preparing to unpack .../net-tools_1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
net-tools (1.60+git20180626.aebd88e-1ubuntu1) 설정하는 중입니다 ...
Processing triggers for man-db (2.9.1-1) ...
root@ubuntu:/etc/apt#
```

④ SSH 프로그램 설치

```
#apt install -y openssh-server
```

```
root@docker:/etc/apt# apt install -y openssh-server
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다... 완료
상태 정보를 읽는 중입니다... 완료
The following additional packages will be installed:
  ncurses-term openssh-client openssh-sftp-server ssh-import-id
제안하는 패키지:
  keychain libpam-ssh monkeysphere ssh-askpass molly-guard
```

```
GNU nano 6.2 /etc/ssh/sshd_config *

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
#AddressFamily any
```

```
root@docker:/etc/apt# systemctl start sshd
root@docker:/etc/apt# systemctl status sshd
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-03-28 16:51:09 KST; 23s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 6024 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
```

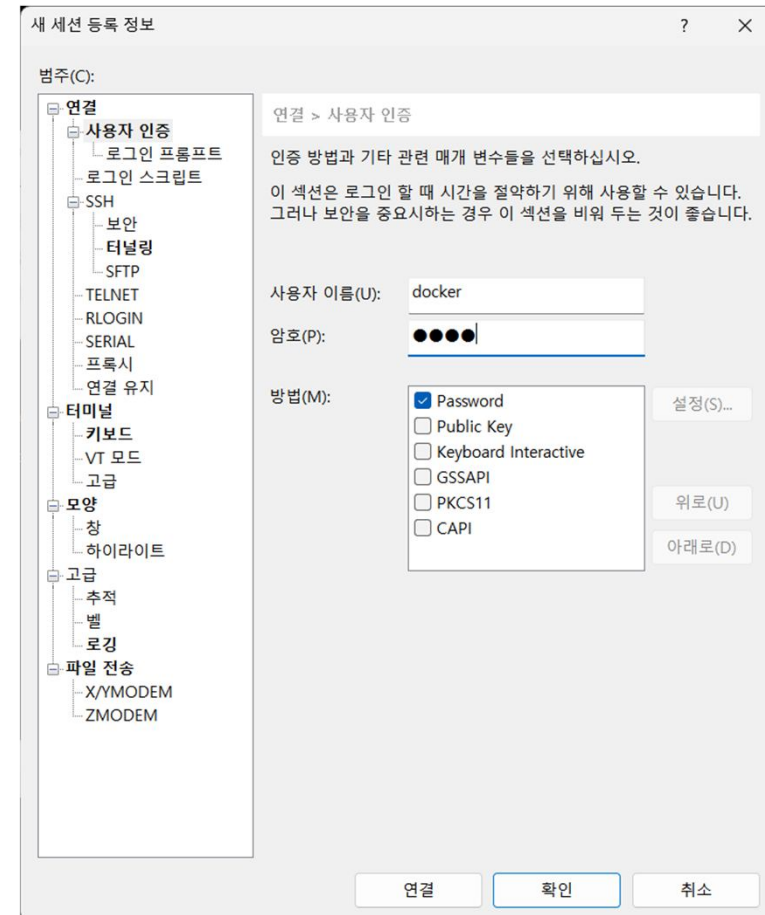
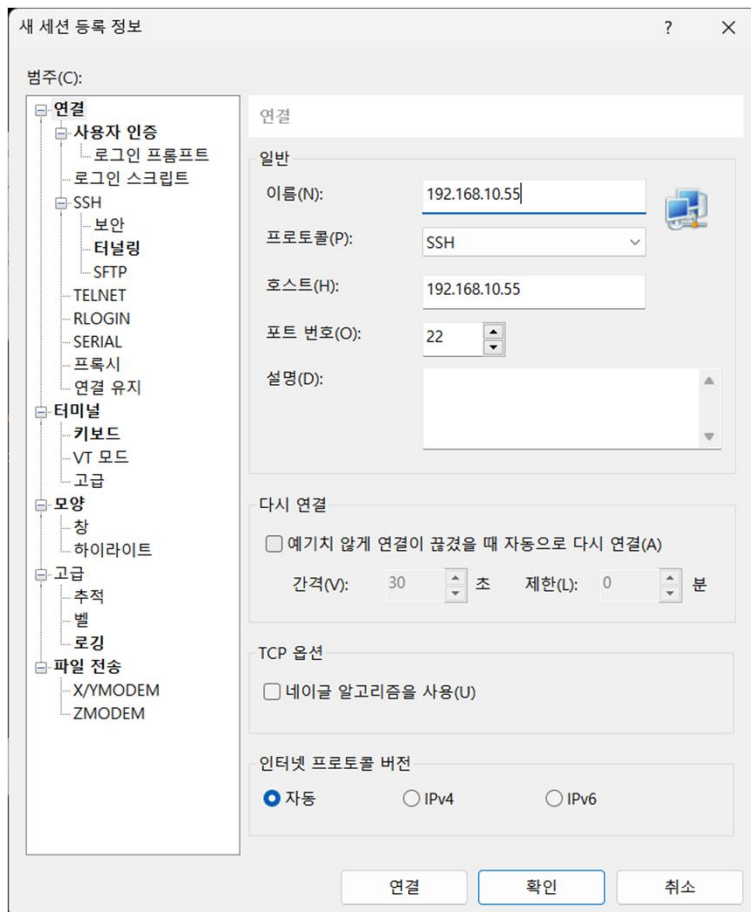
#nano /etc/ssh/sshd_config

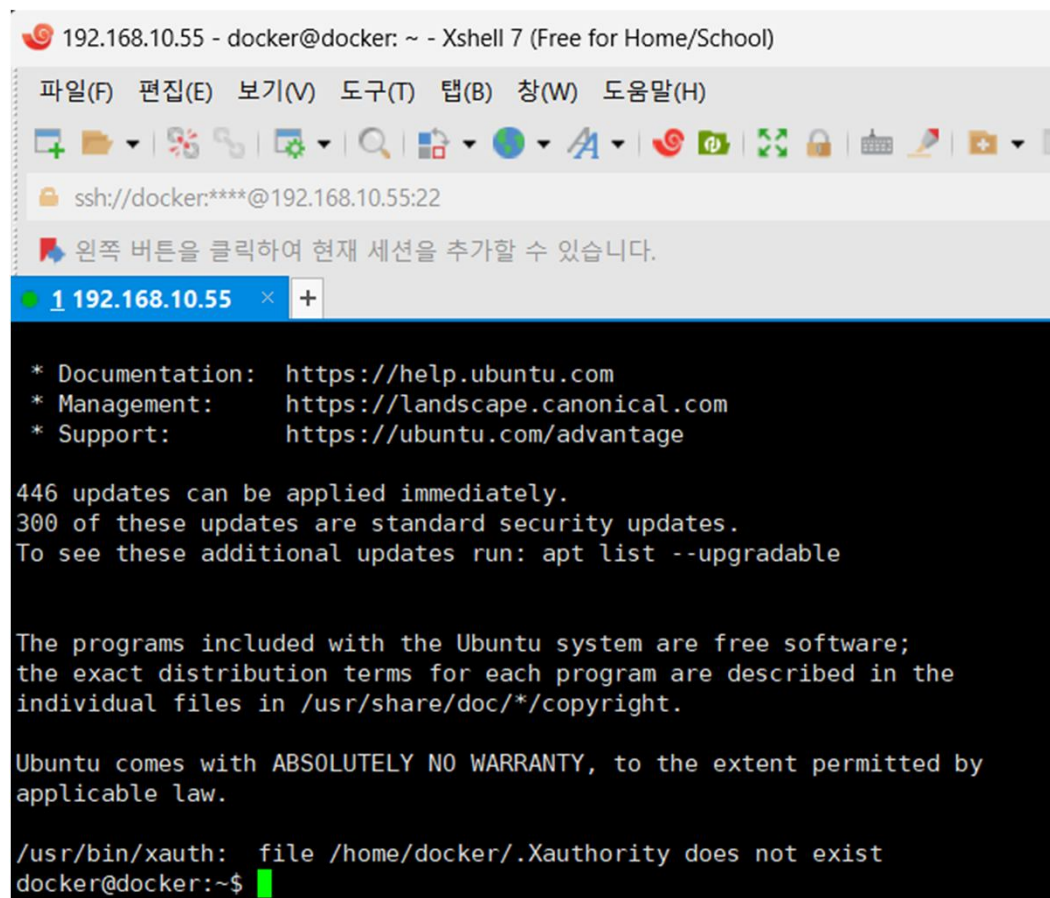
Port 22

#systemctl start sshd

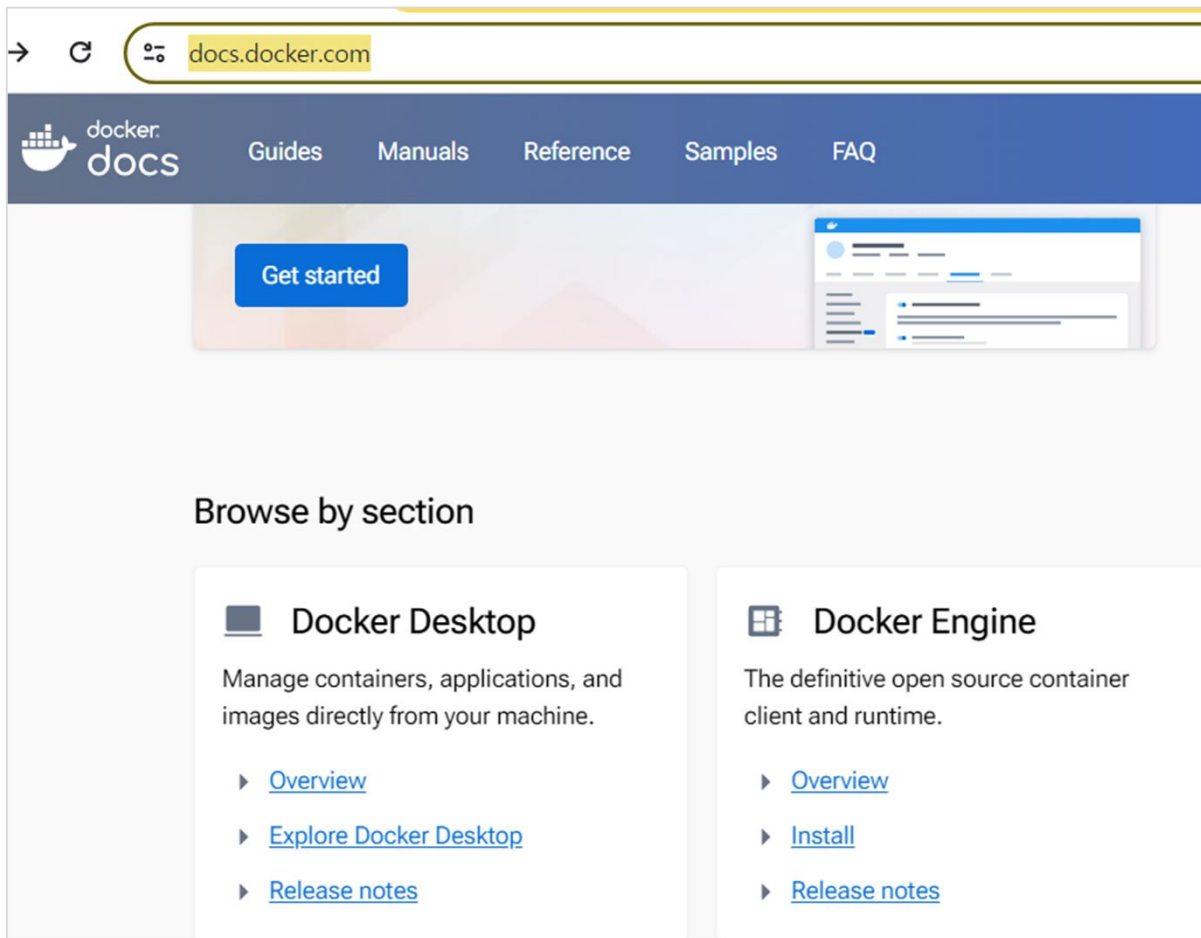
#systemctl status sshd

5 SSH 클라이언트 프로그램 설치 (xshell)





Ubuntu에 Docker Install



<https://docs.docker.com>

<https://docs.docker.com/desktop/install/ubuntu/>

The screenshot shows a web browser window with the URL `docs.docker.com/desktop/install/ubuntu/`. The page has a blue header with the Docker logo and navigation links: Guides, Manuals, Reference, Samples, and FAQ. A left sidebar contains a table of contents with links to Overview, Get Docker, Docker Desktop, Overview, Install Docker Desktop, Install on Mac, Understand permission requirements for Mac, Install on Windows, Understand permission requirements for Windows, Install on Linux, Installation per Linux distro, Install on Ubuntu (highlighted), Install on Debian, Install on Fedora, Install on Arch, Sign in, Explore Docker Desktop, and Upgrade Docker Desktop. The main content area is titled "Install Docker Desktop on Ubuntu". It features a light blue box with a "1 Docker Desktop terms" section, stating that commercial use in larger enterprises requires a paid subscription. Below this, a paragraph says "This page contains information on how to install, launch and upgrade Docker Desktop on an Ubuntu distribution." and a blue button labeled "DEB package". A note mentions "For checksums, see Release notes". The "Prerequisites" section follows, stating "To install Docker Desktop successfully, you must:" and listing three bullet points: meeting system requirements, having a 64-bit version of Ubuntu (Jammy Jellyfish 22.04 or Mantic Minotaur 23.10), and installing `gnome-terminal` for non-Gnome environments. The text "Docker Desktop is supported on x86_64 (or amd64) architecture." is also present.

docs.docker.com/desktop/install/ubuntu/

docker docs Guides Manuals Reference Samples FAQ

Overview
Get Docker
Docker Desktop
Overview
Install Docker Desktop
Install on Mac
Understand permission requirements for Mac
Install on Windows
Understand permission requirements for Windows
Install on Linux
Installation per Linux distro
Install on Ubuntu
Install on Debian
Install on Fedora
Install on Arch
Sign in
Explore Docker Desktop
Upgrade Docker Desktop

Install Docker Desktop on Ubuntu

1 Docker Desktop terms

Commercial use of Docker Desktop in larger enterprises (more than 250 employees OR more than \$10 million USD in annual revenue) requires a [paid subscription](#).

This page contains information on how to install, launch and upgrade Docker Desktop on an Ubuntu distribution.

[DEB package](#)

For checksums, see [Release notes](#)

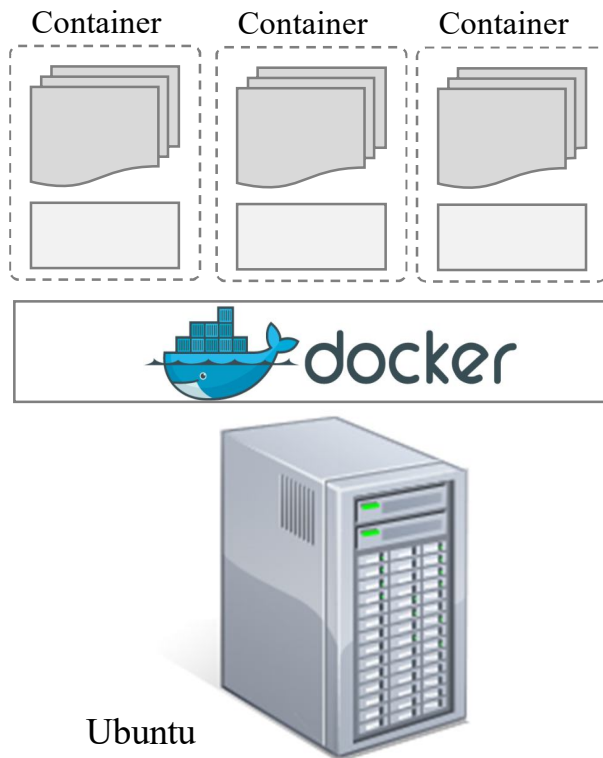
Prerequisites

To install Docker Desktop successfully, you must:

- Meet the [system requirements](#)
- Have a 64-bit version of either the latest LTS version (Ubuntu Jammy Jellyfish 22.04) or the current non-LTS version (Ubuntu Mantic Minotaur 23.10). Docker Desktop is supported on `x86_64` (or `amd64`) architecture.
- For non-Gnome Desktop environments, `gnome-terminal` must be installed:

리눅스 기반의 **Docker 설치 방법**

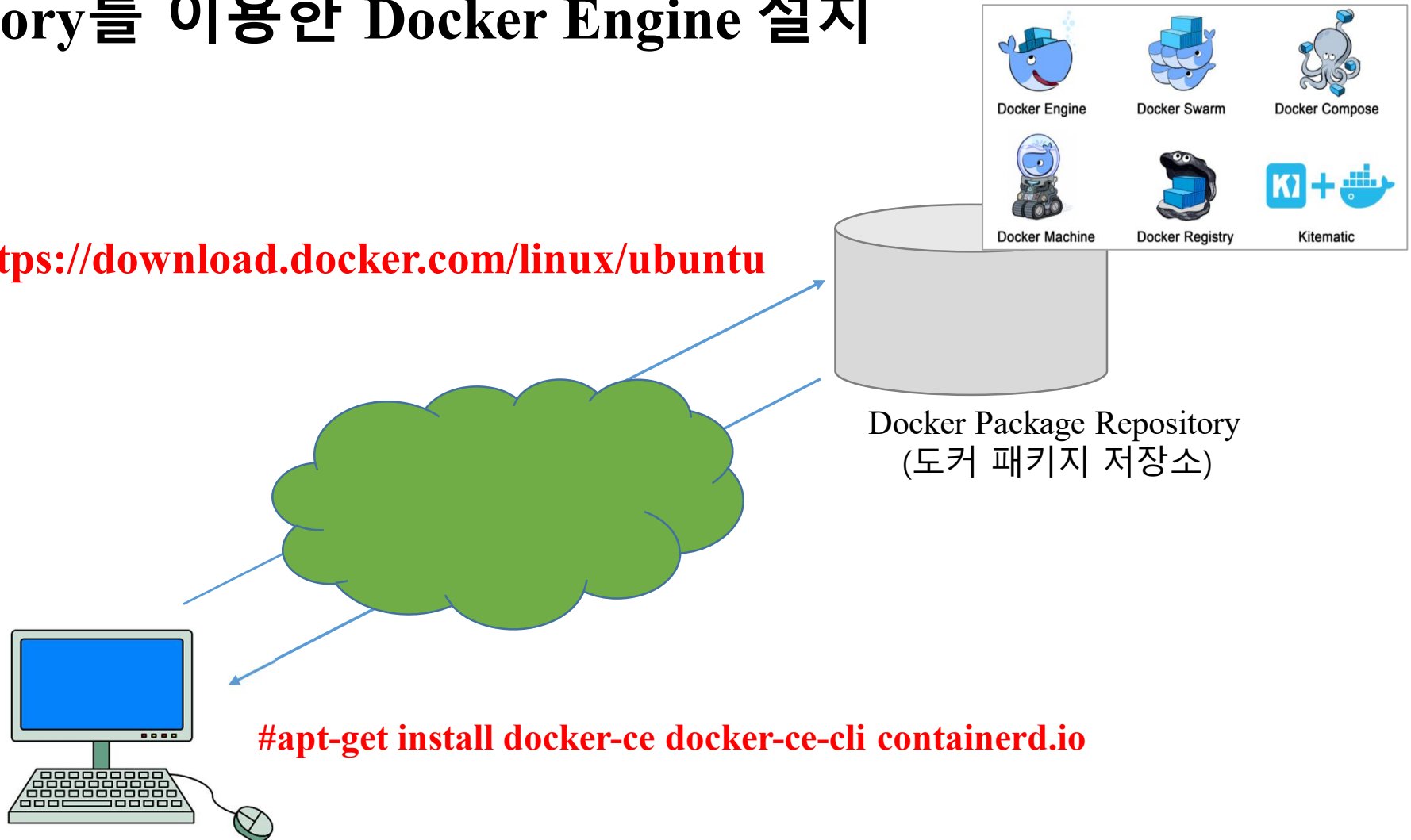
* <https://docs.docker.com/desktop/install/ubuntu/>



- Repository 를 이용한 설치
- 파일 다운로드 후 설치
 - 네트워크가 연결이 안 될 경우
- Script를 이용한 설치

Repository를 이용한 Docker Engine 설치

<https://download.docker.com/linux/ubuntu>



1단계. Docker 저장소 지정

① 유틸리티 파일 설치

```
apt-get update
```

```
apt-get install ca-certificates curl
```

② Docker 공식 GPG key 추가 (도커 인증서 저장)

```
install -m 0755 -d /etc/apt/keyrings
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o  
etc/apt/keyrings/docker.asc
```

```
chmod a+r /etc/apt/keyrings/docker.asc
```

③ Docker Repository URL 등록

```
echo "deb [arch=$(dpkg --print-architecture) signed-  
by=/etc/apt/keyrings/docker.asc]  
https://download.docker.com/linux/ubuntu
```

```
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

```
apt-get update
```

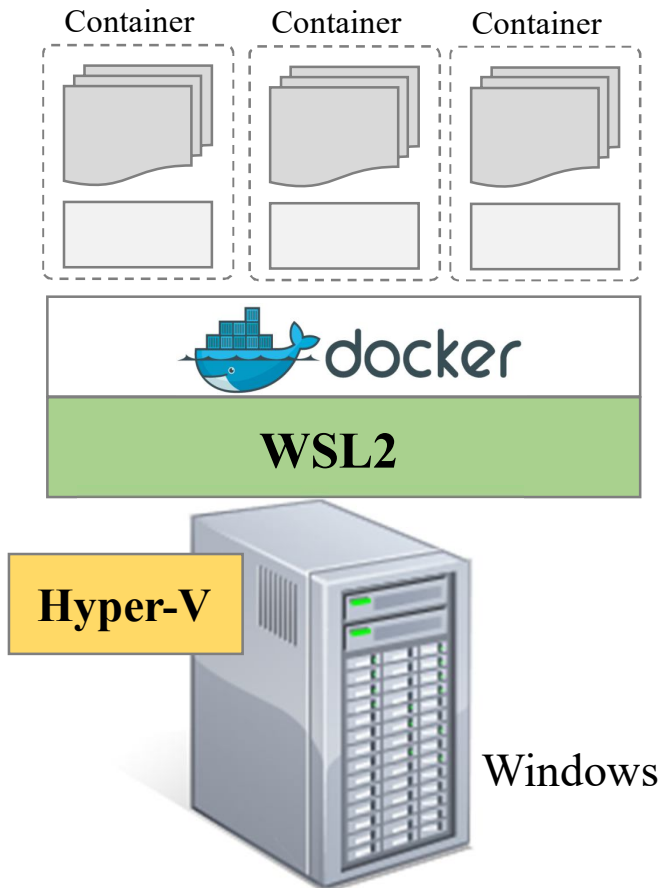
2단계. Docker 패키지 설치

```
apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

3단계. 설치 확인

docker version

Windows 환경에서 Docker 설치



1) Hyper-V 가상화 기능 활성화

2) Hub.docker.com 계정 등록

3) Docker Desktop 설치

- WSL2(Windows Subsystem for Linux v.2)를 통해

리눅스 커널 설치

- Docker 설치

4) Docker 동작 상태 확인

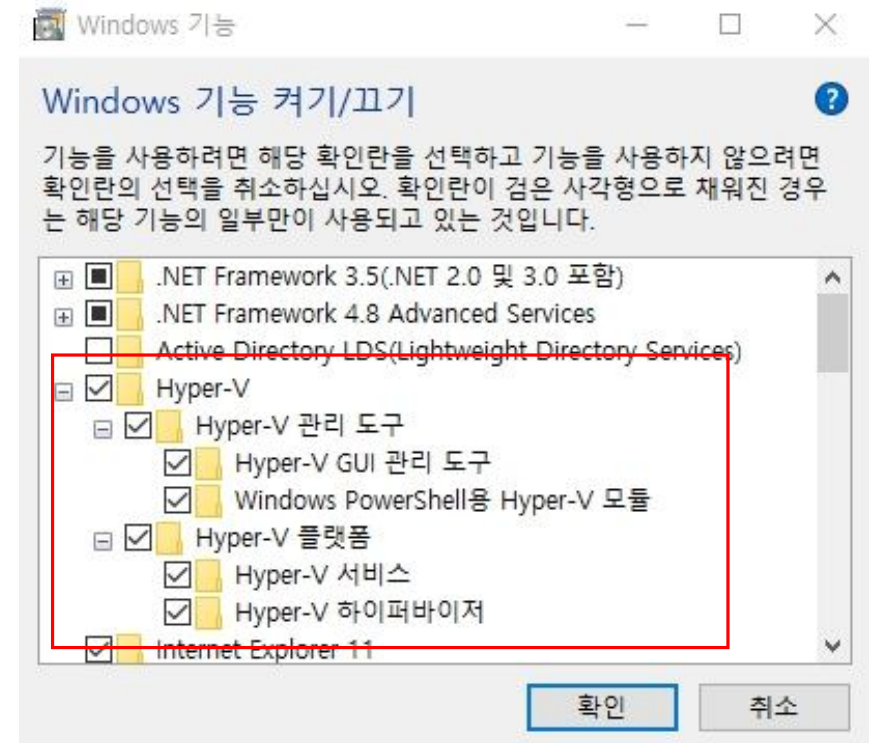
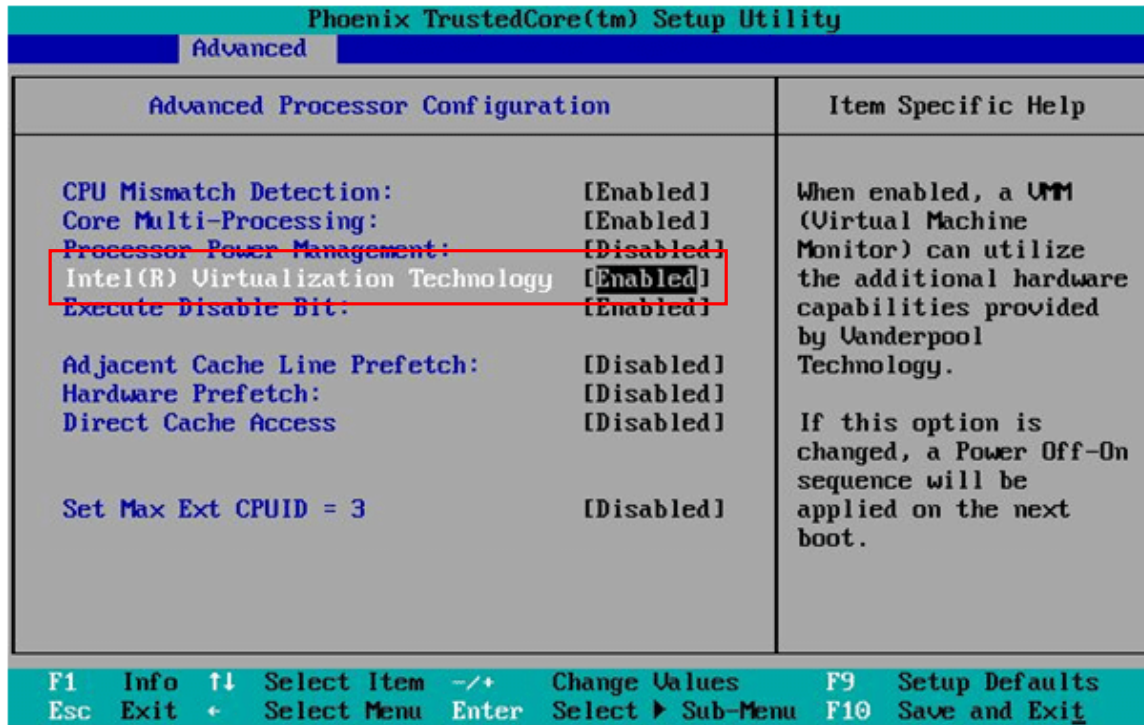
Docker DeskTop

- 컨테이너화된 애플리케이션 및 마이크로서비스를 구축하고 공유할 수 있는 Mac, Linux, Windows 환경용 원클릭 설치 애플리케이션
- 설치된 머신에서 컨테이너, 애플리케이션, 이미지를 관리할 수 있는 간단한 GUI를 제공
- 관련 패키지를 포함
 - Docker Engine, Docker CLI client, Docker Compose, Docker Content Trust, **Kubernetes**, and Credential Helper

WSL2(Windows Subsystem for Linux 2)

- 윈도우에서 리눅스를 사용할 수 있게 해주는 기능
- 윈도우의 가상화 기능을 활용해서 윈도우 위에서 리눅스를 사용할 수 있게해줌

1단계. Hyper-V 활성화



systeminfo

```
관리자: 명령 프롬프트

로그온 서버:      \\DESKTOP-M5C3TKU
하픽스:           하픽스 14개 설치됨
                  [01]: KB5013624
                  [02]: KB5013887
                  [03]: KB4537759
                  [04]: KB4557968
                  [05]: KB5003791
                  [06]: KB5006120
                  [07]: KB5007115
                  [08]: KB5014699
                  [09]: KB5006753
                  [10]: KB5007273
                  [11]: KB5011651
                  [12]: KB5014032
                  [13]: KB5014035
                  [14]: KB5005699


네트워크 카드:    NIC 1개 설치됨
                  [01]: Realtek PCIe GbE Family Controller
                        연결 이름: 이더넷
                        DHCP 사용: 예
                        DHCP 서버: 192.168.35.1
                        IP 주소
                        [01]: 192.168.35.70
                        [02]: fe80::80a2:b12c:29b0:2a3d

Hyper-V 요구 사항:
                  VM 모니터 모드 확장: 예
                  펌웨어에 가상화 사용: 아니요
                  두 번째 수준 주소 변환: 예
                  데이터 실행 방지 사용 가능: 예


C:\Windows\system32>
```


2단계. Hub.docker.com 계정 등록

hub.docker.com/signup

 docker

Create your account


 Continue with Google

 Continue with GitHub

OR

Email

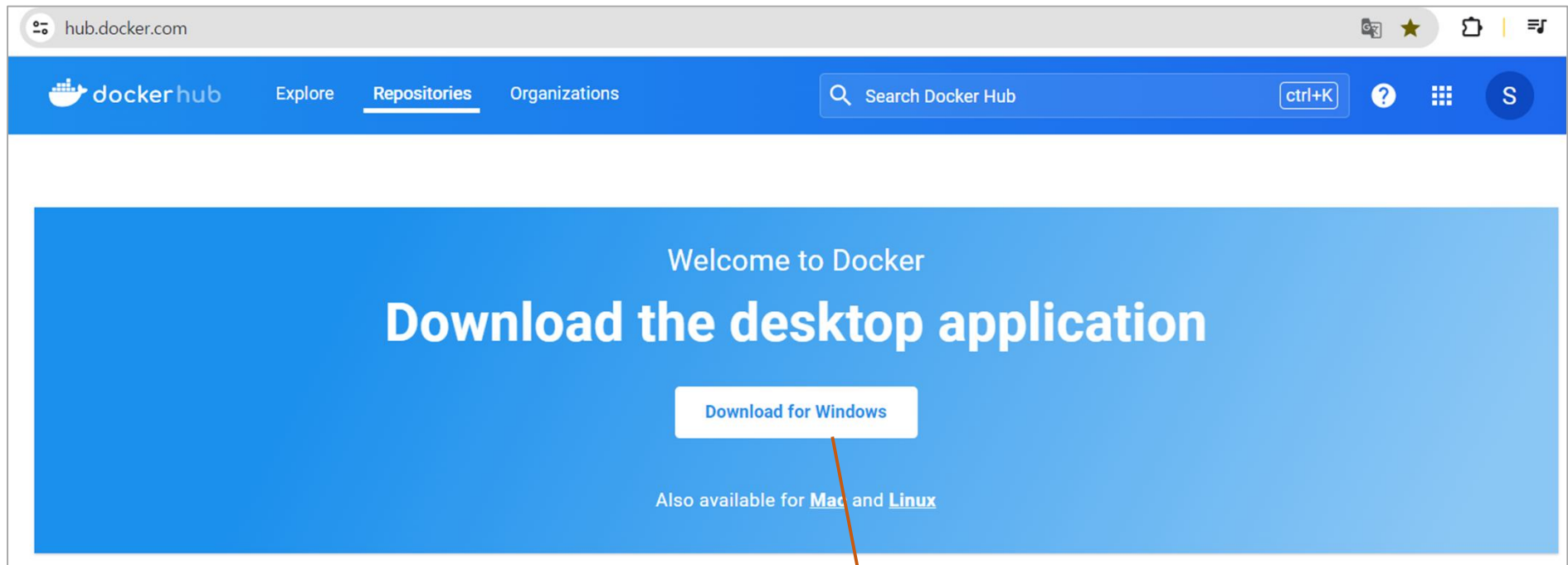
Username

Password 

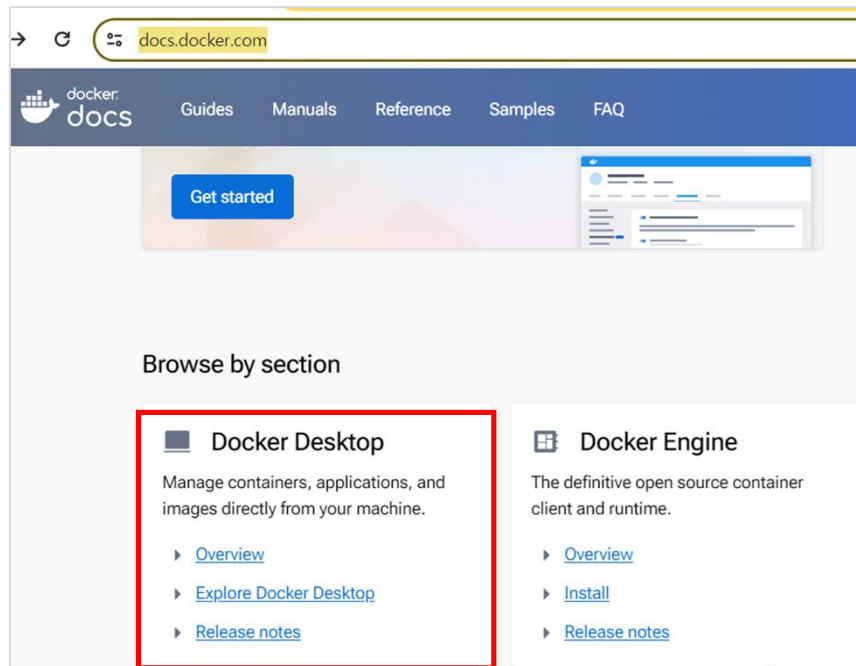
☐ Send me occasional product updates and announcements.

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

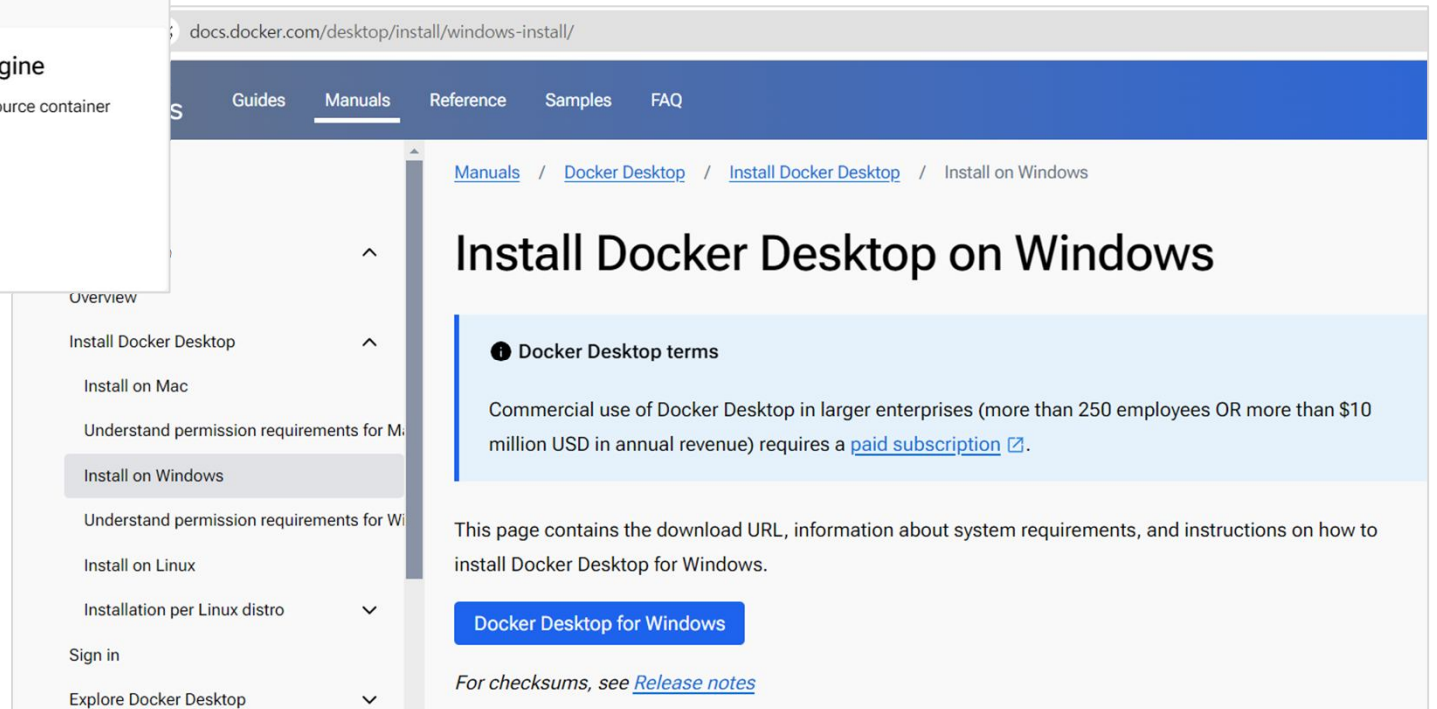
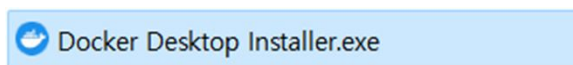
3단계. Docker Desktop 다운로드



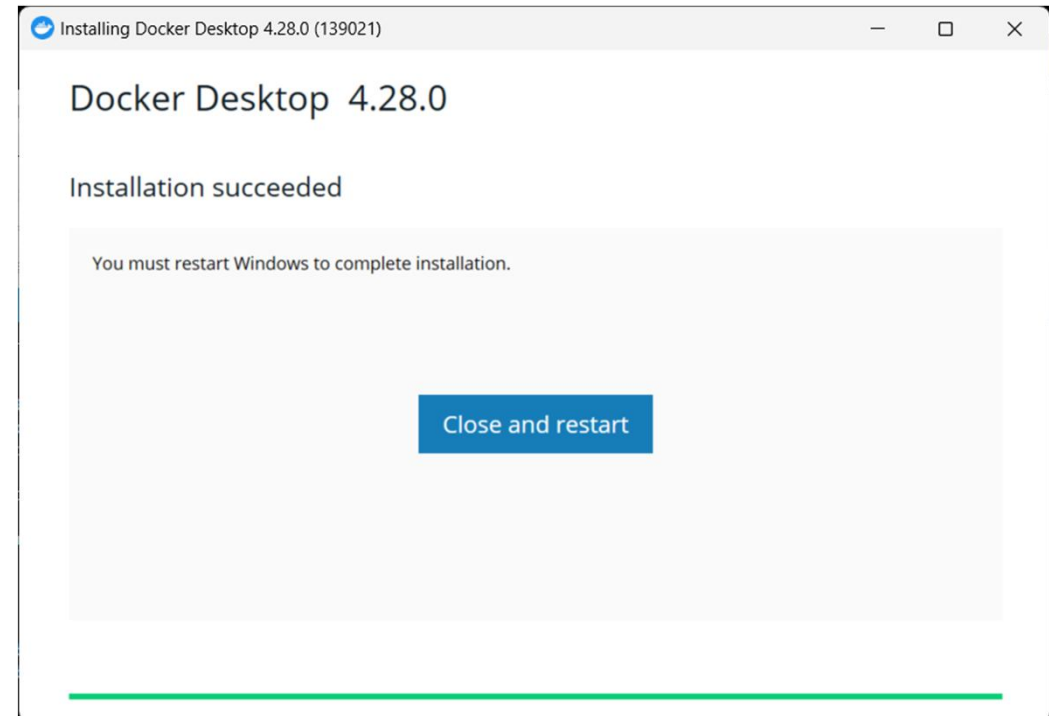
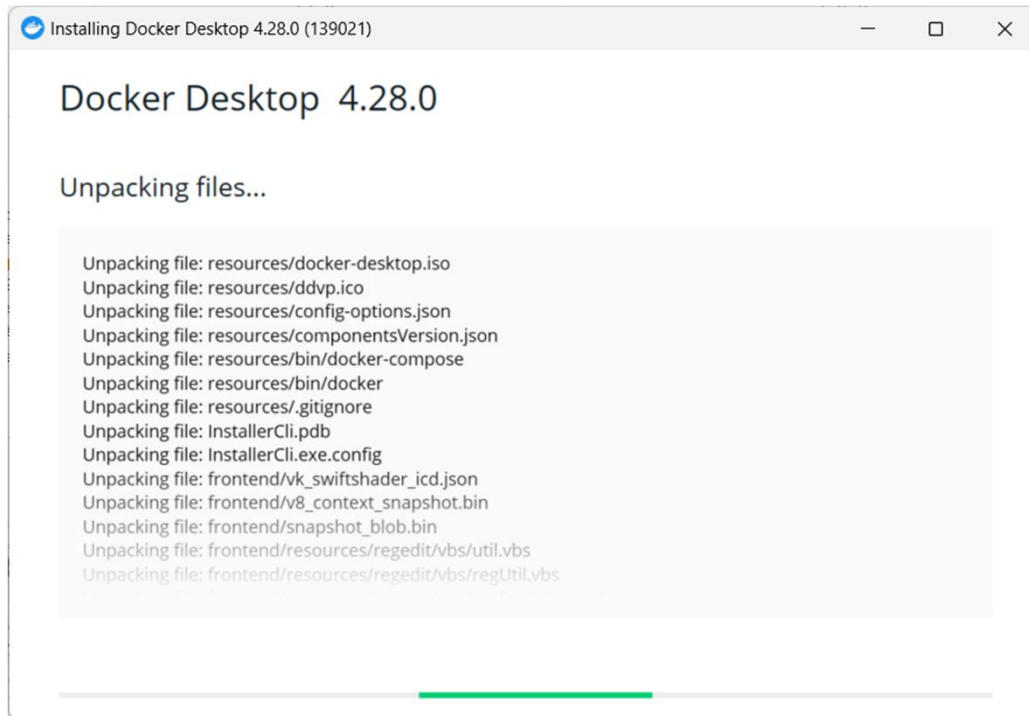
Docker Desktop Installer.exe




<https://docs.docker.com/desktop/install/windows-install/>



4단계. Docker Desktop 설치



5단계. Docker Hub 로그인 후 Docker Engine 활성화




Sign in





Sign in to Docker to continue to Docker Desktop.


Username or email address


[Continue](#)


 docker desktop


[Ctrl+K](#)


   


 Containers


 Images


 Volumes

 Builds


 Dev Environments BETA

 Docker Scout

Extensions 


 Add Extensions

Containers [Give feedback](#)



Your running containers show up here

A container is an isolated environment for your code

 ?

What is a container?

5 mins

```
1 FROM node
2 RUN mkdir -p
3 WORKDIR /app
4 COPY package
```

How do I run a container?

6 mins

[View more in the Learning center](#)

34

6단계. PowerShell을 이용한 Docker 확인

```
PS C:\> docker version
Client:
  Cloud integration: v1.0.35+desktop.11
  Version:          25.0.3
  API version:      1.44
  Go version:       go1.21.6
  Git commit:       4debf41
  Built:            Tue Feb  6 21:13:02 2024
  OS/Arch:          windows/amd64
  Context:          default

Server: Docker Desktop 4.28.0 (139021)
Engine:
  Version:          25.0.3
  API version:      1.44 (minimum version 1.24)
  Go version:       go1.21.6
  Git commit:       f417435
  Built:            Tue Feb  6 21:14:25 2024
  OS/Arch:          linux/amd64
```