

# [OS] Docker Desktop과 WSL

태그

작성중

## Windows에서 Docker를 사용하는 방법

### ▼ 기술적면에서 분류(뇌피셜)

Docker Desktop을 사용할 때 자체 가상화 기술이 두 개로 발전한 듯 보인다.

### 1. Hyper-V

- Docker는 리눅스 기반 프로그램으로 Hyper-V를 사용하는 VM에서 실행된다.

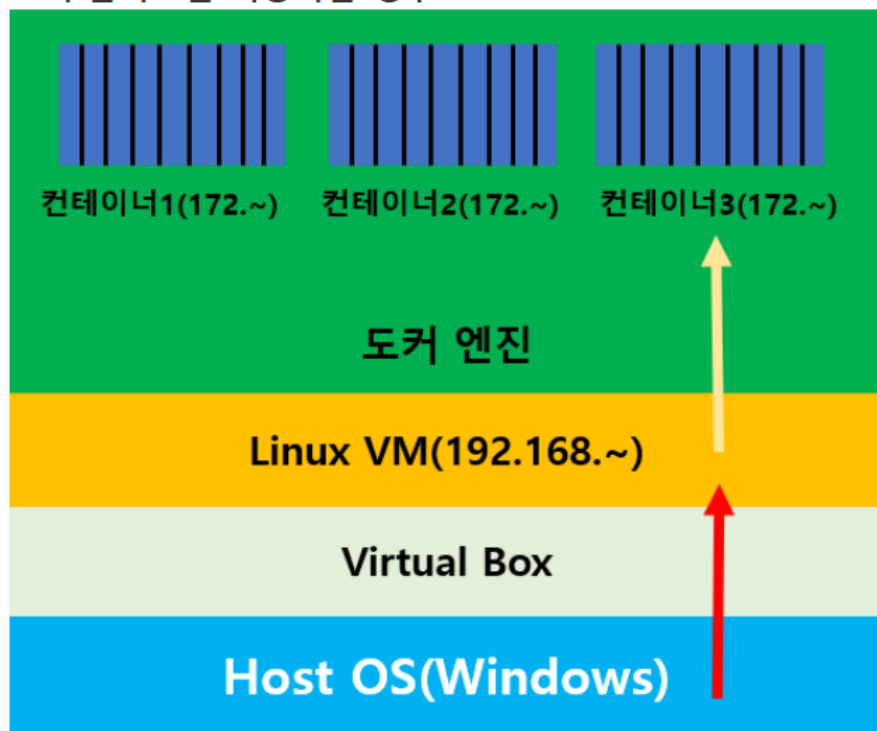
### 2. WSL 사용

- 가상환경과 다르다는 듯

둘 다 Docker Desktop을 사용할 수 있고, 필요에 따라 WSL에 일부 기능(ex Docker Engine)만을 연결해서 사용한다는 듯.

### 1. Docker tool box

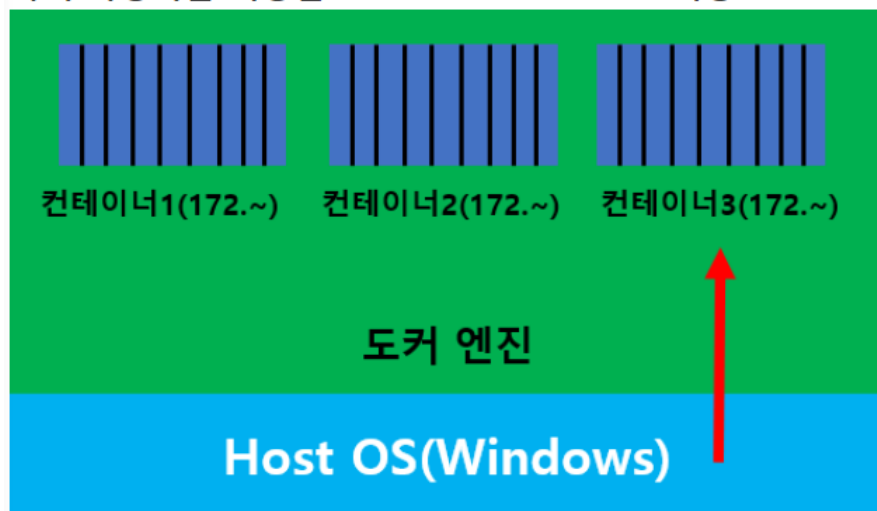
도커 툴박스를 이용하는 경우



- Host 운영체제에 리눅스 가상머신(VM - Oracle의 VirtualBox의 가상화 기술)을 생성하고 그 안에 Docker를 설치.  
→ 컨테이너의 네트워크까지 가상 네트워크가 두 개 생성됨
- 2번의 포트포워딩
  - HOST OS → 가상 머신의 내부 IP
  - Linux 가상 머신 → 도커 컨테이너 NAT IP
- Window10처럼 Hyper-V 가상화 지원 환경에서 사용  
→ 이전 버전의 경우 별도의 가상화 공간을 생성하여 설치 및 사용가능

## 2. Docker for Windows

자체 가상화를 이용한 Docker for Windows 이용

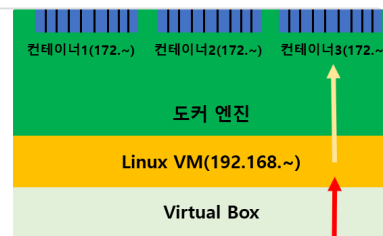


- 자체 가상화 기술로 리눅스 환경 만들.

[Docker] Docker Toolbox(도커 툴박스) 와 Docker for Windows의 차이점

Docker Toolbox(도커 툴박스) 와 Docker for Windows의 차이점 Windows 환경에 도커를 설치하는 경우에는 도커 툴박스를 이용하거나 Docker for Windows를 설치해 사용합니다. 이 둘의 차이는 운영체제가 자체 가상화를 지원하여 가상화를 이용하는

 <https://dololac.tistory.com/355>



# Docker Desktop

- 선치화 환경설정 편의, 개발 및 운영을 위해 필요한 많은 요소를 가지고 있음
  - 도커 엔진(Docker Engine)
  - 도커 CLI 클라이언트(Docker CLI Client)
  - 도커 빌드x(Docker Buildx)
  - 도커 컴포즈(Docker Compose)
  - 도커 콘텐츠 트러스트(Docker Content Trust)
  - 쿠버네티스(Kubernetes)
  - 도커 자격 증명 도우미(Credential Helper) 등이 함께 들어있음

## WSL (Windows Subsystem for Linux) **[자세히]**

- Windows 컴퓨터에서 동시에 Windows와 Linux의 전원에 액세스할 수 있다.
- WSL를 통해 Linux 배포판을 설치하고 기존 가상 머신 또는 이중 부팅 설정의 오버헤드 없이 Windows에서 직접 Linux 애플리케이션, 유틸리티 및 Bash 명령줄 도구를 사용할 수 있다.
- WSL2에서 Docker 정식 지원
- VM Ware나 Virtual Box같은 가상화 프로그램은 Hyper-V 가상화 아키텍처를 사용하기 때문에 WSL2와 동시에 사용할 수 없다.

## WSL 설치 명령**[자세히]**

```
# 관리자 권한으로 PowerShell 혹은 Windows 명령 프롬프트 열기
wsl --install
# 기본 버전(1 or 2) 설정
wsl --set-default-version <Version#>
```

- 설치 후 Linux의 사용자 계정 및 암호를 만들어야함.

## WSL과 Docker

- Docker Desktop을 설치하지 않고 도커를 쓸 수 있음
- WSL2에 Docker Engine을 설치하고 윈도우에는 Docker CLI Client를 설치해 둘을 연결하는 것이 핵심.

## WSL2에 도커 엔진 설치


```
# 도커 설치 스크립트 실행
curl -sSl get.docker.com | sh

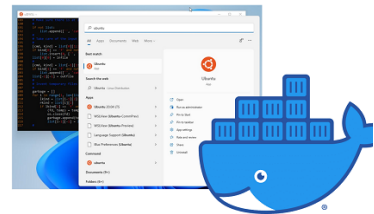
# 도커 버전 확인
docker --version

# 편의를 위한 sudo 권한 설정
# 현재 사용자 권한 확인
grep -E 'sudo|wheel' /etc/group
# sudoer 그룹에 사용자 추가
usermod -aG sudo $USER
# sudo 명령어 쓸 수 있는 조건에 그룹이 들어가 있는지 확인
sudo grep -E '%sudo|wheel' /etc/sudoers
# sudo 권한 설정 확인 후 사용자를 docker 그룹에 추가
sudo usermod -aG docker $USER
```

## ! 중단

이후는 다음 참조


도커 데스크톱 없이 구축하는 WSL2와 도커 개발 환경 - 넷마블 기술 블로그  
안녕하세요, 넷마블 기술분석팀 박정욱입니다. 도커 데스크톱(Docker Desktop) 유료  
화 정책에 대해 대부분 알고 계시지만, 가끔 아직 모르시는 분들이 있으신 것 같습니  
다. 도커 데스크톱 유료화 정책 발표가 2021년 8월 31일에 있었으니, 어느덧 1년이 넘  
 <https://netmarble.engineering/docker-on-wsl2-without-docker-desktop/>



현재 Docker Desktop을 설치하고 실습하려했는데, 리눅스 환경을 구축하기 위한 설명이 주되어서  
약간 내용이 일관되지 못한것 같다. Docker Desktop 먼저 실습하고 돌아올까한다.

## 참고


### WSL 설치

개발자는 Windows 컴퓨터에서 동시에 Windows와 Linux의 전원에 액세스  
할 수 있습니다. WSL(Linux용 Windows 하위 시스템)을 통해 개발자는  
Linux 배포판(예: Ubuntu, OpenSUSE, Kali, Debian, Arch Linux 등)을 설  
 <https://learn.microsoft.com/ko-kr/windows/wsl/install>



## WSL을 사용하는 이유

Microsoft는 2020년 5월 Windows10의 업데이트에서 WSL( Windows Subsystem for Linux)의 2번째 버전을 발표하였다. WSL의 첫 번째 버전은 고급 개발자들용으로만 공개되었던 점에 비해 이번에는 범용 개발에 쓰일 수

 <https://vlog.io/@mooh2jj/WSL%EC%9D%84-%EC%82%AC%EC%9A%A9%ED%95%98%EB%8A%94-%EC%9D%B4%EC%9C%A0>

