우분투(Ubuntu) 18.04 버전에서 도커(Docker) 설치 방법을 핵심만 정리했으니 바로 따라 해 보세요.

우분투 터미널(Terminal)에 아래의 명령어를 한줄 씩 입력해 실행해주세요.

sudo apt update

sudo apt install apt-transport-https ca-certificates curl software-properties-common

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable"

sudo apt update

apt-cache policy docker-ce

마지막 줄의 명령어를 입력하고 실행하게 되면 다음과 같은 메시지가 표시됩니다.

1. docker-ce:
2. Installed: (none)
3. Candidate: 18.06.1~ce~3-0~ubuntu
4. Version table:
5. 18.06.1~ce~3-0~ubuntu 500
6. 500 https*://download.docker.com/linux/ubuntu bionic/stable amd64 Packages*
7. 18.06.0~ce~3-0~ubuntu 500
8. 500 https*://download.docker.com/linux/ubuntu bionic/stable amd64 Packages*
9. 18.03.1~ce~3-0~ubuntu 500
10. 500 https*://download.docker.com/linux/ubuntu bionic/stable amd64 Packages*

Installed: (none) 인 것으로 보아 아직 도커가 설치되어 있지 않다는 것을 알 수 있습니다.

마지막으로 다음 명령어를 실행해 도커를 설치할 수 있습니다.

1. sudo apt install docker-ce

아래 명령어로 도커가 정상적으로 실행 중인지 확인할 수 있습니다.

1. sudo systemctl status docker

명령어를 실행하면 다음과 비슷한 메시지를 확인할 수 있습니다.

1. ● docker.service - Docker Application Container Engine
2. Loaded: **loaded** (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
3. Active: **active** (running) since Fri 2018-08-24 07:20:49 UTC; 51s ago
4. Docs: https*://docs.docker.com*
5. Main PID: **4527** (dockerd)
6. Tasks: 18
7. CGroup: /system.slice/docker.service
8. ├─4527 /usr/bin/dockerd -H fd*://*
9. └─4549 docker-containerd --config /var/run/docker/containerd/containerd.toml

**Docker 이미지 생성**

# docker pull ubuntu:18.04

**생성된 이미지 확인**

# docker images

REPOSITORY TAG IMAGE ID CREATED SIZE

ubuntu 18.04 4c108a37151f 5 days ago 64.2MB

**도커 컨테이너 네트워크 외부 접속 가능하게 설정하는 방법**

<https://cyberx.tistory.com/157>

**#도커에서 사용할 라우팅용 인터페이스 생성**

ip link add link enp6s0 mactest type macvlan mode bridge **#실제 인터페이스, 생성할 인터페이스**

ip address add 192.168.10.51 dev mactest

ip link set dev mactest up /이더넷 UP

**#라우팅 정보 비우기**

ip route flush dev enp6s0

ip route flush dev mactest

**#r라우팅 정보 추가**

ip route add 192.168.10.0/23 dev mactest metric 0

**#라우팅 테이블에 게이트웨이 정보 추가**

ip route add default via 192.168.10.254 metric 101

**#도커 네트워크 정보 추가**

docker network create -d macvlan -o macvlan\_mode=bridge --subnet=192.168.30.0/23 --gateway 192.168.30.254 -o parent=docker1 dockerbridge1

**해당 인터페이스를 사용하는 도커 실행 - ([사용할 인터페이스]**

docker run -d -it --net **[test]** --ip 192.168.10.60 ubuntu:18.04

\*도커에서 추가로 필요한 공유라이브러리의 경우에는 별도로 설치해주어야 한다. (stdlib32c++6 같은 32비트 라이브러리 같은 경우)

로컬 설치를 위해서 도커 설치파일과 이미지 파일 백업, 재설치하는 과정 기술해야할 듯

Dockerfile

#사용할 리눅스 버전

FROM ubuntu:18.04

#사전 설치할 라이브러리 설치

RUN apt update

RUN apt install -y lib32stdc++6

#사용할 어플리케이션을 컨테이너에 복사

COPY ./app/\* /root/

#작업디렉토리 설정

WORKDIR /root

#컨테이너 기동시 동작할 명령 입력

CMD ["bash", "/root/start\_server.sh"]

도커파일 빌드

Docker build –t .

도커 실행 명령어 IP주소와 ENV등을 조작하여 컨테이너의 매개변수를 변경시킨다

# docker run -d --rm --net dockerbridge1 --ip 192.168.30.62 –cap-add NET\_ADMIN --name vied03 iec850test:0.0

run : 컨테이너 기동명령

-d : 데몬으로 동작(백그라운드 동작)

-e PARAM=asdf : 환경변수 전달(여러 개 가능)

--cap-add NET\_ADMIN : 컨테이너 네트워크 관리자 권한 허용

--net dockerbridge1 : dockerbridge1 네트워크 어댑터 사용

--ip 192.168.30.62 : 컨테이너 IP주소 할당

--name veid03 : 컨테이너 이름 명명

Iec850test:0.0 : 해당 이미지로 컨테이너 생성