**Dockerfile이 있는 위치에서 아래 명령어 실행으로 이미지 생성**

Docker build –tag dockerzap:0.1



**이미지 생성이 완료되었으면 아래와 같이 명령어를 통해 도커 컨테이너 생성**

docker run -it -v /hostPATH:/ZAP\_2.9.0/plugin -p 8081:8080 -i dockerzap:0.1

**volume 옵션은 호스트 경로에서 custom plugin을 넣을 부분으로 수정**

ex)

docker run -it -v /home/user/zap/ZAP\_2.9.0/plugin:/ZAP\_2.9.0/plugin -p 8081:8080 -i dockerzap:0.1

**백그라운드 실행을 원하면**

docker run -d -v /home/user/zap/ZAP\_2.9.0/plugin:/ZAP\_2.9.0/plugin -p 8081:8080 -i dockerzap:0.1

**과 같이 -d 옵션을 넣어 컨테이너 생성**

Dockerfile

From ubuntu:16.04

ADD ZAP\_2.9.0 /ZAP\_2.9.0

ADD entrypoint.sh /entrypoint.sh

# Install OpenJDK-8

RUN apt-get update && \

    apt-get install -y openjdk-8-jdk && \

    apt-get install -y ant && \

    apt-get clean;

# Fix certificate issues

RUN apt-get update && \

    apt-get install ca-certificates-java && \

    apt-get clean && \

    update-ca-certificates -f;

# Setup JAVA\_HOME -- useful for docker commandline

ENV JAVA\_HOME /usr/lib/jvm/java-8-openjdk-amd64/

RUN export JAVA\_HOME

RUN chmod -R 777 ZAP\_2.9.0

RUN chmod 777 /entrypoint.sh

# set port

EXPOSE 8080 8081

CMD /ZAP\_2.9.0/zap.sh -daemon -host 0.0.0.0 -config api.key=3343 -config api.addrs.addr.name=.\* -config api.addrs.addr.regex=true

8080 포트와 8081포트를 expose 하였음. 다른 포트가 필요하다면 추가적으로 기입

Api key는 3343으로 적어놓았으나 run할 때 command를 덮어씌워 바꾸거나 image를 생성하기전 dockerfile에 api.key=3343 부분을 수정하여 api key 수정 가능

모든 address에 대해 permission을 허용했음. 특정 ip에 대해서만 허용하려면 api.addrs.addr.name과 api.addrs.addr.regex 부분을 수정하면 됨

Entrypoint.sh는 추가적으로 필요한 커맨드가 있을지 확실하지 않아 추가하였음. 없다면 포함하지 않아도 zap 실행에 문제는 없음