
Docker에 JENNIFER 설치하기

1.1 JAVA Application이 구동되는 Docker 환경



Docker의 기본적인 사용법은 알고 있다고 가정한 상태에서 작성되었습니다.

1.1.1 사전 준비

- JENNIFER Java Agent 패키지



이 튜토리얼에서는 별도로 구동되고 있으며, 유효한 라이선스가 있는 JENNIFER Server가 존재합니다. Java Agent 패키지와 JENNIFER Server가 없다면, 홈페이지에서 2주짜리 무료 트라이얼 라이선스를 신청하여 설치하면 됩니다.

1.1.2 요약

- JENNIFER Java agent 패키지 다운로드
- 해당 패키지를 Dockerfile이 있는 Root 작업 디렉토리로 옮기기
- 패키지를 컨테이너로 복사하고, 압축해제하기
- Agent 설정 파일을 컨테이너로 복사하기
- 모니터링 할 Container를 업데이트하기

1.1.3 상세



이번 튜토리얼은 톰캣 애플리케이션 서버를 예제로 합니다. 위에서 설명한 바와 같이, Dockerfile이 있는 Root 작업 디렉토리에 JENNIFER Java Agent 패키지를 준비해 주세요.

아래와 같은 구조로 Root 작업 디렉토리에 Dockerfile, 설치 패키지, Agent 설정 파일, setenv.sh 스크립트를 구성합니다.

```
├── java_agent_install
│   ├── Dockerfile
│   ├── jennifer-agent-java-x.x.x.x.zip
│   ├── jennifer.conf
│   └── setenv.sh
```

- jennifer.conf : 에이전트 설정파일로, 내용은 다음 단락에서 구성합니다.
- setenv.sh : JENNIFER agent가 구동될 JVM의 Argument들을 설정하는 스크립트입니다.

1.1.3.1 JENNIFER.conf 파일

에이전트 설정 파일을 수정합니다. jennifer.conf 파일을 편집해 아래와 같은 형태로 만들어 주세요.

server_address와 server_port 가 접속하고자 하는 JENNIFER Data server의 IP와 Port, 해당 에이전트가 속할 domain_id를 반드시 지정해 주어야 합니다.

```
server_address = 192.168.9.37
server_port = 5000

# default log_dir "{agent_home}/logs"
# log_dir = {agent_home}/logs
# log_file =
# log_level : 1(info), 2(debug), 3(trace)
log_level = 1
log_encoding = UTF-8
log_rotation = true
# log_max_age : Working when log_rotation is activated. Valid range is (1-365)
# log_max_age = 0

# domain_id : Valid range is (1-32767)
domain_id = 1000
```



jennifer-agent-java-x.x.x.x.zip 형태의 에이전트 설치 패키지에 압축을 해제해 보면, jennifer.conf 파일이 있는데, 해당 파일을 복사하고, 편집해 사용할 수도 있습니다. 다만, 원본 zip파일이 Root 작업 디렉토리에 존재해야 합니다.

1.1.3.2 setenv.sh 수정

setenv.sh 를 수정해 아래 CATALINA_OPTS 값을 추가합니다.

```
# JENNIFER Java Agent
AGENT_HOME=/opt/agent.java
export CATALINA_OPTS="$CATALINA_OPTS -javaagent:$AGENT_HOME/jennifer.jar -Djennifer.config=$AGENT_HOME/conf/jennifer.conf"
```

1.1.3.3 Dockerfile 구성

마지막으로, dockerfile을 구성합니다. 이 구성에 따라 Docker는 다음의 작업을 진행하게 될 것입니다.

1. 에이전트 설치 패키지를 컨테이너 내부에 복사하고, 압축을 해제해 /opt/agent.java 디렉토리에 위치시킨다.
2. jennifer.conf 파일을 conf 디렉토리로 복사한다
3. setenv.sh 파일을 CATALINA_HOME/bin 디렉토리로 복사한다

```
FROM alpine AS builder

LABEL maintainer="Khalid Saeed khalid@jennifersoft.com"
ENV AGENT_VERSION=5.3.2.7

RUN apk --no-cache add unzip

COPY jennifer-agent-java-${AGENT_VERSION}.zip /
RUN unzip -oq jennifer-agent-java-${AGENT_VERSION}.zip -d /tmp

FROM tomcat:8.5-jre8-alpine

RUN apk update && apk upgrade

COPY --from=builder /tmp/agent.java /opt/agent.java
COPY jennifer.conf /opt/agent.java/conf/jennifer.conf
COPY setenv.sh ${CATALINA_HOME}/bin
```



AGENT_VERSION이 다운로드 받은 설치 패키지의 버전과 일치해야만 합니다.

위 예제는 Multi-stage builder를 사용했지만, 어떤 방식으로든 dockerfile을 구성하면 됩니다. dockerfile이 구성되면 아래와 같이 build하고 컨테이너를 실행할 수 있습니다.

```
docker build -t jennifer-java .  
docker run -d --name test jennifer-java
```

정상적으로 실행되었다면, JENNIFER 대시보드 화면에서 새로운 인스턴스가 생성된 것을 확인할 수 있습니다.

1.1.4 코드 예제

<https://github.com/JENNIFER-University/javaagent-docker-example>