

Title: Spring Boot and Docker-compose setting arguments

Post Body:

I'm probably new on the docker area and try to combine my Spring-Boot project with docker and docker-compose, but somehow docker-compose does not work.

I'm using the com.spotify maven plugin as described in the spring boot documentation. I can build and run docker images with the spotify plugin and also from command line using the docker command, but not when I try to use docker-compose up.

Here I always get the error 'Error: Invalid or corrupt jarfile /app.jar'.

I'm sure that it has to do with the ARG JAR\_FILE which is passed from the spotify plugin to the docker command. I don't know how to setup this ARG\_FILE in the docker-compose.yml so that i'm able to start with docker-compose up and also run only the docker image using the spotify plugin.

Here is my setup:

pom.xml

```
<?xml version='1.0' encoding='UTF-8'?>
```

```
http://maven.apache.org/xsd/maven-4.0.0.xsd"> 4.0.0
```

```
<groupId>eu.devroyal</groupId> <artifactId>someProject</artifactId> <version>0.0.1-SNAPSHOT</version> <packaging>jar</packaging>
```

Dockerfile:

```
FROM openjdk:8-jdk-alpine VOLUME /tmp ARG JAR_FILE COPY ${JAR_FILE} app.jar ENTRYPOINT ['java', '-Djava.security.egd=file:/dev/urandom']
```

docker-compose.yml:

```
version: '3' services: web: environment: - JAR_FILE=target/someProject-app.jar build: ./
```

Accepted Answer:

At the time you build the dockerfile, you must add this parameter

```
docker build --build-arg JAR_FILE=jar_file_path .
```

alternative way to do it in the docker-compose.yml

```
version: '3' services: web: build: context: ./ args: JAR_FILE: jar_file_path ports:
```

Highest Rated Answer:

I would recommend this way

```
FROM openjdk:8-jdk-alpine VOLUME /tmp COPY target/someproject-app.jar /app.jar ENTRYPOINT ['java', '-Djava.security.egd=file:/dev/urandom']
```

When Dockerfile gets executed the respective jar in the target will be copied.