



链滴

Solo 博客 Arm64 Aarch64 ArmV8 版本搭建方法

作者: [wyisolo](#)

原文链接: <https://ld246.com/article/1664354746748>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Solo博客虽然D大没有提供Arm64版本的镜像文件，但是熟悉Docker的人应该都会利用docker build机制手工构建Armv8架构的Docker镜像。

#1.查看gitee上的源码Dockerfile文件#

```
FROM maven:3.8.4-openjdk-11 as MVN_BUILD

WORKDIR /opt/solo/
ADD ./tmp
RUN cd /tmp && mvn package -DskipTests -Pci -q && mv target/solo/* /opt/solo/ \
&& cp -f /tmp/src/main/resources/docker/* /opt/solo/

FROM openjdk:18-alpine
LABEL maintainer="Liang Ding<845765@qq.com>"

WORKDIR /opt/solo/
COPY --from=MVN_BUILD /opt/solo/ /opt/solo/
RUN apk add --no-cache ca-certificates tzdata

ENV TZ=Asia/Shanghai
ARG git_commit=0
ENV git_commit=$git_commit

EXPOSE 8080

ENTRYPOINT [ "java", "-cp", "lib/*:", "org.b3log.solo.Server" ]
```

从上面可知From关键字拉取了两个基础镜像包。好在Docker生态越来越好，基础镜像包都有对应的arm64版本的，直接替换就好。

#替换后的Dockerfile的文件内容如下所示#

其实也就是将语句【FROM openjdk:18-alpine】改成了【FROM jcxldn/openjdk-alpine:16-jdk】即可

```
FROM maven:3.8.4-openjdk-11 as MVN_BUILD

WORKDIR /opt/solo/
ADD ./tmp
RUN cd /tmp && mvn package -DskipTests -Pci -q && mv target/solo/* /opt/solo/ \
&& cp -f /tmp/src/main/resources/docker/* /opt/solo/

FROM jcxldn/openjdk-alpine:16-jdk
LABEL maintainer="Liang Ding<845765@qq.com>"

WORKDIR /opt/solo/
COPY --from=MVN_BUILD /opt/solo/ /opt/solo/
RUN sed -i 's!http://dl-cdn.alpinelinux.org/!https://mirrors.ustc.edu.cn/!g' /etc/apk/repositorie

RUN apk add --no-cache ca-certificates tzdata

ENV TZ=Asia/Shanghai
ARG git_commit=0
ENV git_commit=$git_commit
```

EXPOSE 8080

ENTRYPOINT ["java", "-cp", "lib/*:", "org.b3log.solo.Server"]

#3.通过Git 命令的Clone指令 下载源码文件到本地的arm64服务器家目录 执行本地编译即可#

```
git clone https://gitee.com/dl88250/solo.git
```

然后进入含有Dockerfile的文件夹内, 按照步骤1, 2 修改 Dockerfile中的基础镜像包文件名

```
指令 [RUN sed -i 's!http://dl-cdn.alpinelinux.org!https://mirrors.ustc.edu.cn!' /etc/apk/repositories]
```

作用用来进行软件包国内加速使用的镜像站

最后执行 `docker build -t 88250/solo:1.0 .` 命令进行本地化编译即可

#4.最后 如果懒的编译也可以下载我提供的编译好的arm64 solo包镜像 导入即可#

```
--envRUNTIME_DB="MYSQL"  
--envJDBC_USERNAME="root"  
--envJDBC_PASSWORD="123456"  
--envJDBC_DRIVER="com.mysql.cj.jdbc.Driver"  
--envJDBC_URL="jdbc:mysql://127.0.0.1:3306/solo?useUnicode=yes&characterEncoding=UTF  
8&useSSL=false&serverTimezone=UTC&allowPublicKeyRetrieval=true"  
88250/solo:1.0--listen_port=8080 --server_scheme=http --server_host=localhost --server_por  
t=localhost
```

具体的安装流程参考 D大gitee项目 [88250/solo](#)

最后希望D大后期可以直接在DockerHub中直接放出ARM64的兼容镜像, 这样也能方便大家在Arm
构上使用上D大的杰作!