[**第二十四篇： springboot整合docker**](http://blog.csdn.net/forezp/article/details/71024219)

参考：<http://blog.csdn.net/forezp/article/details/71024219>

转载请标明出处：   
<http://blog.csdn.net/forezp/article/details/71024219>   
本文出自[方志朋的博客](http://blog.csdn.net/forezp)

这篇文篇介绍，怎么为 springboot程序构建一个docker镜像。docker 是一个开源的应用容器引擎，基于 Go 语言 并遵从Apache2.0协议开源。Docker 可以让开发者打包他们的应用以及依赖包到一个轻量级、可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。容器是完全使用沙箱机制，相互之间不会有任何接口（类似 iPhone 的 app）,更重要的是容器性能开销极低。

准备工作

环境：

* linux环境或mac,不要用windows
* jdk 8
* maven 3.0
* docker

对docker一无所知的看[docker教程](http://www.runoob.com/docker/docker-tutorial.html)。

创建一个springboot工程

引入web的起步依赖，创建一个 Controler:

@SpringBootApplication

@RestController

public class SpringbootWithDockerApplication {

@RequestMapping("/")

public String home() {

return "Hello Docker World";

}

public static void main(String[] args) {

SpringApplication.run(SpringbootWithDockerApplication.class, args);

}

}

将springboot工程容器化

Docker有一个简单的[dockerfile](https://docs.docker.com/engine/reference/builder/)文件作为指定镜像的图层。让我们先创建一个 dockerFile文件：

src/main/docker/Dockerfile:

FROM frolvlad/alpine-oraclejdk8:slim

VOLUME /tmp

ADD springboot-with-docker-0.0.1-SNAPSHOT.jar app.jar

RUN sh -c 'touch /app.jar'

ENV JAVA\_OPTS=""

ENTRYPOINT [ "sh", "-c", "java $JAVA\_OPTS -Djava.security.egd=file:/dev/./urandom -jar /app.jar" ]

我们通过maven 构建docker镜像。

在maven的pom目录，加上docker镜像构建的插件

<properties>

<docker.image.prefix>springio</docker.image.prefix>

</properties>

<build>

<plugins>

<plugin>

<groupId>com.spotify</groupId>

<artifactId>docker-maven-plugin</artifactId>

<version>0.4.11</version>

<configuration>

<imageName>${docker.image.prefix}/${project.artifactId}</imageName>

<dockerDirectory>src/main/docker</dockerDirectory>

<resources>

<resource>

<targetPath>/</targetPath>

<directory>${project.build.directory}</directory>

<include>${project.build.finalName}.jar</include>

</resource>

</resources>

</configuration>

</plugin>

</plugins>

</build>

注意：${docker.image.prefix} 为你在 docker官方仓库的用户名，如果你不需要上传镜像，随便填。

通过maven 命令：

第一步：mvn clean

第二步： mvn package docker:bulid ,如下：

Step 2/6 : VOLUME /tmp   
—> Running in a98be3878053   
—> 8286e98b54c5   
Removing intermediate container a98be3878053   
Step 3/6 : ADD springboot-with-docker-0.0.1-SNAPSHOT.jar app.jar   
—> c6ce13e50bbd   
Removing intermediate container a303a3058869   
Step 4/6 : RUN sh -c ‘touch /app.jar’   
—> Running in cf231afe700e   
—> 9a0ec8936c00   
Removing intermediate container cf231afe700e   
Step 5/6 : ENV JAVA\_OPTS “”   
—> Running in e192597fc881   
—> 2cb0d73bbdb0   
Removing intermediate container e192597fc881   
Step 6/6 : ENTRYPOINT sh -c java $JAVA\_OPTS -Djava.security.egd=file:/dev/./urandom -jar /app.jar   
—> Running in ab85f53fcdd8   
—> 60fdb5c61692   
Removing intermediate container ab85f53fcdd8   
Successfully built 60fdb5c61692   
[INFO] Built forezp/springboot-with-docker   
[INFO] ————————————————————————   
[INFO] BUILD SUCCESS   
[INFO] ————————————————————————   
[INFO] Total time: 01:45 min   
[INFO] Finished at: 2017-04-19T05:37:44-07:00   
[INFO] Final Memory: 19M/48M   
[INFO] ————————————————————————

镜像构建成功。查看镜像：

docker images

显示：

forezp/springboot-with-docker latest 60fdb5c61692 About a minute ago 195 MB

启动镜像：

$ docker run -p 8080:8080 -t forezp/springboot-with-docker

打开浏览器访问 localhost:8080;浏览器显示：Hello Docker World。   
说明docker 的springboot工程已部署。

停止镜像：

docker stop 60fdb5c61692

删除镜像：

docker rm 60fdb5c61692

参考资料

[https://docs.docker.com/engine/reference/builder/)](https://docs.docker.com/engine/reference/builder/))

<http://www.runoob.com/docker/docker-tutorial.html>

源码下载

<https://github.com/forezp/SpringBootLearning>

优秀文章推荐：

* 更多springboot 教程：[springBoot非官方教程 | 文章汇总](http://blog.csdn.net/forezp/article/details/70341818)
* 更多springcoud 教程：[史上最简单的 SpringCloud 教程 | 文章汇总](http://blog.csdn.net/forezp/article/details/70148833)