

==========================================================================

修改以ExecStart开头的行：(因为我的系统是centos 7的，所以修改为下面得)

ExecStart=/usr/bin/dockerd -H tcp://0.0.0.0:2375 -H unix://var/run/docker.sock

1

**如果是centos7以下的话，就把ExecStart修改为：**

ExecStart=/usr/bin/dockerd -H fd:// -H tcp://0.0.0.0:2375

1

修改后保存文件，然后通知docker服务做出的修改

systemctl daemon-reload

1

重启docker服务

service docker restart

1

接下来测试一下看是否能连接到docker api。上面的2375就是对应端口

curl http://localhost:2375/version

1

============================================================================

<build>

<finalName>changjiang</finalName>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

<plugin>

<groupId>com.spotify</groupId>

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

<version>0.4.14</version>

<configuration>

<!-- 注意imageName一定要是符合正则[a-z0-9-\_.]的，否则构建不会成功 -->

<!-- 详见：https://github.com/spotify/docker-maven-plugin Invalid repository name ... only [a-z0-9-\_.] are allowed-->

<imageName>39.99.41.155:5000/suibian/${project.artifactId}:${project.version}</imageName> <!-- 镜像名称-->

<baseImage>ascdc/jdk8</baseImage>

<entryPoint>["java", "-jar", "/${project.build.finalName}.jar"]</entryPoint>

<resources>

<resource>

<targetPath>/</targetPath>

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

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

</resource>

</resources>

<dockerHost>http://39.99.41.155:2375</dockerHost> <!--远程服务器docker的地址 -->

</configuration>

</plugin>

</plugins>

</build>

运行maven命令

mvn clean package docker:build -DpushImage

===========

把镜像搞为容器运行

docker run -d \

-p 8181:5000 \

--restart=always \

--name registry \

registry