Docker 之使用Dockerfile构建镜像

版本号	构建时间	备注
v1.0	2017.7.27	初稿完成

Dockerfile 是一个文本格式的配置文件，用户可以根据**Dockerfile**来快速创建自定义的镜像。

我们以创建使用JRE8一个Tomcat8镜像为例，介绍如何使用Dockerfile构建自定义的镜像。

1. 创建目录，作为所有操作的根目录

```
➜  mkdir tomcat8.0-jre8
➜  cd tomcat8.0-jre8
```

2. 下载`Server JRE 8`

关于`Server JRE`和`JRE`的区别参见这里

> **Server JRE:** It is used to deploy long-running java applications on server. It provides the fastest possible operating speed. It has been specifically fine tuned to maximize peak operating speed. It has highly aggressive algorithms to optimize the runtime performance of java application. It also includes variety of monitoring tools.
>
> **Client JRE:** It is used to run java applications on the end-users systems. It contains everything to run the java applications. It can start up faster and requires a smaller memory footprint.

下载地址在这里，需要先同意Oracle的License.获取到下载地址后使用wget下载，也可以下载后FTP到该目录下。

```
➜ tomcat8.0-jre8 wget http://download.oracle.com/otn-pub/java/jdk/8u144-b01/090f390dda5b47b9b721c7dfaa008135/jre-8u144-linux-x64.tar.gz\?AuthParam\=1501125394_e0a237af749f858df7b5b447d44c86f1
--2017-07-27 11:14:28--  http://download.oracle.com/otn-pub/java/jdk/8u144-b01/090f390dda5b47b9b721c7dfaa008135/jre-8u144-linux-x64.tar.gz?AuthParam=1501125394_e0a237af749f858df7b5b447d44c86f1
正在解析主机 download.oracle.com (download.oracle.com)... 65.222.200.131, 65.222.200.146
正在连接 download.oracle.com (download.oracle.com)|65.222.200.131|:80... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度： 75922776 (72M) [application/x-gzip]
正在保存至：“jre-8u144-linux-x64.tar.gz?AuthParam=1501125394_e0a237af749f858df7b5b447d44c86f1”

100%[=============================================================================================>] 75,922,776  2.70MB/s 用时 29s

2017-07-27 11:14:57 (2.52 MB/s) - 已保存 “jre-8u144-linux-x64.tar.gz?AuthParam=1501125394_e0a237af749f858df7b5b447d44c86f1” [75922776/75922776])

➜ tomcat8.0-jre8 ll
总用量 73M
-rw-r--r--. 1 root root 73M 7月  25 05:30 jre-8u144-linux-x64.tar.gz?AuthParam=1501125394_e0a237af749f858df7b5b447d44c86f1
➜ tomcat8.0-jre8 mv jre-8u144-linux-x64.tar.gz\?AuthParam=1501125394_e0a237af749f858df7b5b447d44c86f1 jre-8u144-linux-x64.tar.gz
➜ tomcat8.0-jre8 ll
总用量 73M
-rw-r--r--. 1 root root 73M 7月  25 05:30 jre-8u144-linux-x64.tar.gz
```

3.下载`Tomcat 8`

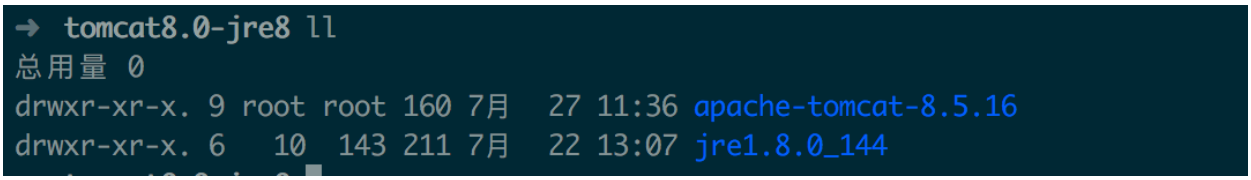
下载地址在这里，老规矩，wget 走起

```
➜  tomcat8.0-jre8 wget http://mirror.bit.edu.cn/apache/tomcat/tomcat-8/v8.5.16/bin/apache-tomcat-8.5.16.tar.gz
```

4. 解压Tomcat和jre，并删除相应的压缩包

```
➜  tomcat8.0-jre8 tar vxf apache-tomcat-8.5.16.tar.gz
➜  tomcat8.0-jre8 tar vxf jre-8u144-linux-x64.tar.gz
➜  tomcat8.0-jre8 rm -rf apache-tomcat-8.5.16.tar.gz
➜  tomcat8.0-jre8 rm -rf jre-8u144-linux-x64.tar.gz
```

处理后的文件夹的内容如下：



5. 编写Dockerfile

建议在文本编辑器中编辑好FTP上去，Vim大神请忽略...

内容如下：

```
FROM ubuntu:16.04

# 创建者的基本信息
MAINTAINER sunaowei@qq.com

# 添加jre和tomcat的镜像到容器的/usr/local目录下
ADD jre1.8.0_144 /usr/local/jre
ADD apache-tomcat-8.5.16 /usr/local/tomcat

# 设置环境变量
ENV CATALINA_HOME /usr/local/tomcat
ENV JAVA_HOME /usr/local/jre
ENV PATH $CATALINA_HOME/bin:$JAVA_HOME/bin:$PATH

# 工作目录
WORKDIR $CATALINA_HOME

# 更改系统时区设置
RUN ln -sf /usr/share/zoneinfo/Asia/Shanghai  /etc/localtime

# 添加可执行权限
RUN chmod +x $CATALINA_HOME/bin/*.sh

# 暴露端口
EXPOSE 8080

# 制定启动容器时执行的默认命令
CMD ["catalina.sh", "run"]
```

6. 编译镜像

```
➜  tomcat8.0-jre8 docker build -t hctomcat:8.0-jre8 .
```

`-t`:`name:tag`

`.`:当前文件夹,Dockerfile的所在路径

编译过程如下图:

```
➜ tomcat8.0-jre8 docker build -t hctomcat:8.0-jre8 .
Sending build context to Docker daemon    220MB
Step 1/12 : FROM ubuntu:16.04
 ---> 14f60031763d
Step 2/12 : MAINTAINER sunaowei@qq.com
 ---> Running in 6fa5c71af746
 ---> b8dff57f97fd
Removing intermediate container 6fa5c71af746
Step 3/12 : ADD jre1.8.0_144 /usr/local/jre
 ---> 01713989f207
Removing intermediate container 45e380acb819
Step 4/12 : ADD apache-tomcat-8.5.16 /usr/local/tomcat
 ---> 46eae2a34dfa
Removing intermediate container 2e471b924549
Step 5/12 : ENV CATALINA_HOME /usr/local/tomcat
 ---> Running in d26b37db3ded
 ---> dbaaafae583f
Removing intermediate container d26b37db3ded
Step 6/12 : ENV JAVA_HOME /usr/local/jre
 ---> Running in afcecbad557c
 ---> 221b6b31b06e
Removing intermediate container afcecbad557c
Step 7/12 : ENV PATH $CATALINA_HOME/bin:$JAVA_HOME/bin:$PATH
 ---> Running in 677c7edd960f
 ---> 966a01622b59
Removing intermediate container 677c7edd960f
Step 8/12 : WORKDIR /usr/local/tomcat/bin
 ---> a8cb065a235e
Removing intermediate container 624c0314fb00
Step 9/12 : RUN ln -sf /usr/share/zoneinfo/Asia/Shanghai  /etc/localtime
 ---> Running in 01993f86d213
 ---> 61ea92495656
Removing intermediate container 01993f86d213
Step 10/12 : RUN chmod +x $CATALINA_HOME/bin/*.sh
 ---> Running in 1c1aaefb800a
 ---> 1550780e5cbe
Removing intermediate container 1c1aaefb800a
Step 11/12 : EXPOSE 8080
 ---> Running in 94a65af0141c
 ---> 809ab306f9f7
Removing intermediate container 94a65af0141c
Step 12/12 : CMD catalina.sh run
 ---> Running in 2a296a47be4e
 ---> 771c22f5209e
Removing intermediate container 2a296a47be4e
Successfully built 771c22f5209e
Successfully tagged hctomcat:8.0-jre8
```

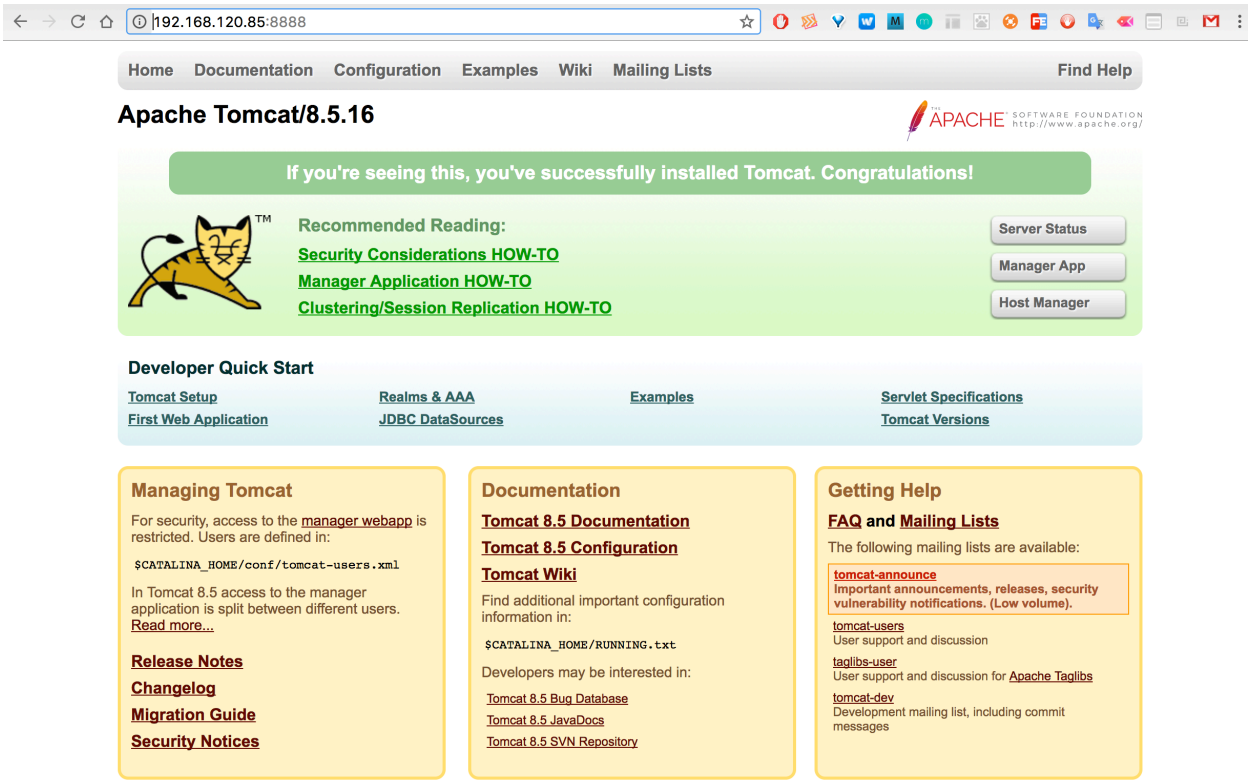
我们使用`docker images`可以查看到该镜像

```
➜ tomcat8.0-jre8 docker images
REPOSITORY                  TAG                 IMAGE ID            CREATED             SIZE
hctomcat                    8.0-jre8            771c22f5209e        About a minute ago  339MB
registry                    latest              751f286bc25e        6 days ago          33.2MB
ubuntu                      16.04               14f60031763d        6 days ago          120MB
rediscluster_sentinel       latest              d2df8bf4bd60        2 weeks ago         98.9MB
haproxy-redis               latest              52a0a9d1102d        2 weeks ago         5.51MB
redis_sentinel              latest              009e7a0ee0a6        2 weeks ago         99MB
alpine                      3.5                 074d602a59d7        4 weeks ago         3.99MB
tomcat                      8.0-jre8            34c69d29100d        4 weeks ago         334MB
tomcat                      8.0-jre7            26eadc84d43b        4 weeks ago         357MB
redis                       3                   4e482b286430        4 weeks ago         98.9MB
redis                       latest              4e482b286430        4 weeks ago         98.9MB
nginx                       latest              c246cd3dd41d        4 weeks ago         107MB
vmware/harbor-jobservice    v1.1.2              4ef0a7a33734        6 weeks ago         163MB
vmware/harbor-ui            v1.1.2              4ee8f190f366        6 weeks ago         183MB
vmware/harbor-adminserver   v1.1.2              cdcf1bed7eb4        6 weeks ago         142MB
vmware/harbor-db            v1.1.2              fcb8aa7a0640        6 weeks ago         329MB
vmware/registry             2.6.1-photon        0f6c96580032        2 months ago        150MB
zookeeper                   latest              5291027d4199        2 months ago        143MB
vmware/harbor-notary-db     mariadb-10.1.10     64ed814665c6        3 months ago        324MB
vmware/nginx                1.11.5-patched      8ddadb143133        3 months ago        199MB
vmware/notary-photon        signer-0.5.0        b1eda7d10640        4 months ago        156MB
vmware/notary-photon        server-0.5.0        6e2646682e3c        4 months ago        157MB
vmware/harbor-log           v1.1.2              9c46a7b5e517        5 months ago        192MB
```

7. 我们使用该镜像创建一个容器

```
➜  tomcat8.0-jre8 docker run -d --name tomcat -p 8888:8080 hctomcat:8.0-jre8
67b0f532f03a3a4ceab55032fb310f4ef4586ba04b87ee0a3b5d260f1a22d561
```

容器正常启动，我们访问一下看是否正常。



能正常访问，至此，我们的一个Tomcat镜像就构建完成了。

8. push到私有库中

首先我们登陆到我们的私有仓库，打上标签，并`push`到*library*中

```
➜  registry docker login 192.168.120.85
Username: admin
Password:
Login Succeeded
➜  registry docker tag hctomcat:8.0-jre8 192.168.120.85/library/hctomcat:8.0-jre8
➜  registry docker push 192.168.120.85/library/hctomcat:8.0-jre8
The push refers to a repository [192.168.120.85/library/hctomcat]
1af1a4ed7ea8: Pushed
049fa24a600c: Pushed
887b58b2ccb0: Pushed
c1ac78de2350: Pushed
26b126eb8632: Pushed
220d34b5f6c9: Pushed
8a5132998025: Pushed
aca233ed29c3: Pushed
e5d2f035d7a4: Pushed
8.0-jre8: digest:
sha256:dcfddd42443f2b0bc273425034a103d68ddcda6e7e81918bb83bb1381207d928 size: 2195
```

到此，我们制作的第一个Tomcat镜像就完成了。