

## 通过现有仓库创建mongo docker

小精灵的探索 <http://blog.sina.com.cn/elfexplore> 2017-05-17 01:42:04

1. 目前国内外有很多云提供各类的image 镜像， mongo也算是基本配置了

```
docker pull daocloud.io/library/mongo:3
```

其中 daocloud.io/library/ 也是docker的一个国内仓库，网速来说比国外下载要快些

3 是tag标签，当然可以写别的数值例如： mongo:3.7

2. 这个版本镜像的mongo程序在 /usr/local/bin目录下， 运行该镜像的一个 docker 容器（进入bash控制命令行）

```
docker run -ti daocloud.io/library/mongo:3 /bin/bash
```

3. 一般来说我们会手动指定mongo的数据路径与日志路径，并启动mongodb服务器

```
mkdir mongo
```

```
cd mongo
```

```
mkdir data
```

```
usr/local/bin/mongod
```

```
--dbpath=/usr/local/mongodb/data/
```

```
--logpath=/usr/local/mongodb/data/mongodb.log
```

```
--logappend&
```

数据、日志路径以及日志保存方式，启动mongod 服务守护进程

这样，在docker 内部就能够调用 mongo 访问 monggo了。

4. 但上述变更只要退出这个docker 容器就不会保存，那太可惜了。 所以最要持久化这个docker 容器为一个本地镜像，以后按照这个本地镜像，就可以启动已经配置好的mongo。 -----

4.1 exit 当前mongo 容器

4.2 通过commit 将容器携带当前状态保存成一个本地镜像

```
docker commit -m "mongo init" -a "dennis"
```

[docker\_container\_id] mongo\_init:3

-m commit的描述； -a 用户名称

一串数字和字幕，标识那个做过修改的docker container

5. 从新的本地镜像 初始化一个容器；

```
docker run -d -p 27017:27017 -d -ti mongo_init:3
```

-p 27017:27017 将docker的监听端口（后者）映射到前面的用户占用端口

-d 后台进程

6. 对于后台进程，可以通过docker ps 查看，然后登入到后台进程

```
docker attach [container id]
```

7. 登陆 docker 容器后，可以通过exit 退出，但如果exit突出了 docker也stop了。但可以通过ctl + P + Q 回到主命令行，但不退出。

8. 个人经验：

本来在第七步骤的情况下，用户就可以通过键入：外部主机:映射端口对应本docker的应用。例如可以通过 <http://localhost:27017> 访问在docker内的mongo。但是我就是不行。后来发现 主机网络设置中，虚拟网卡 hyper-V和virtualBox 同时各有一个虚拟网络，我将virtualBox屏蔽掉，就没问题了。

另外 调用 docker run的时候，有一个参数 --net对应 host, bridge等4种类型，具体使用和含义参考：

<http://www.infoq.com/cn/articles/docker-network-and-pipeline-open-source-explanation-practice/>

参考文献：

[http://blog.sina.com.cn/s/blog\\_628cc2b70102wcvd.html](http://blog.sina.com.cn/s/blog_628cc2b70102wcvd.html)

本文地址：[http://blog.sina.com.cn/s/blog\\_69cf1be90102xdi3.html](http://blog.sina.com.cn/s/blog_69cf1be90102xdi3.html)

所属分类：杂谈

相关评论：

阅读次数：