**6-3｜Docker镜像仓库的讲解**

**Docker镜像仓库的讲解**

如果你希望自己本地打好的镜像可以挪到远程服务器上时候，那么这个时候就需要借助一个镜像仓库的帮助了。

Docker镜像仓库你可以理解为是一种临时存放Docker镜像文件的平台，我们在本地上打好了镜像后，将它push到这个平台，接着在远程服务器上可以通过 docker pull [images-name] 去拉取镜像。

要想实现上述的这种效果，这里需要你去购买一个私服的仓库，或者自己购置一台云服务器，然后部署类似于Docker hub这类仓库平台。

如果我们的已有现成的镜像仓库，那么要想将本地镜像推送成功，那么就按照以下步骤去执行即可。

首先，我们需要在本地会话里面去向远程仓库建立一个会话，为后续的push操作做准备，命令如下：

|  |
| --- |
| SQL `docker login --username=你的用户名 registry.baidubce.com |

然后，将我们的镜像push上去：

|  |
| --- |
| Java docker push registry.baidubce.com/qiyu-live-test/qiyu-live-user-provider-docker:1.0.1 |

这里要注意，不是所有的镜像都可以随意push成功的，这里对镜像的名字有特别的要求，格式如下：

|  |
| --- |
| Java docker push 镜像仓库地址/仓库的命名空间/本地镜像的名字:版本号 |

如果你当前的镜像名字不是这种格式的话，建议可以使用docker tag去进行调整。

docker tag 用于给镜像打标签，语法如下：

|  |
| --- |
| SQL docker tag SOURCE\_IMAGE[:TAG] TARGET\_IMAGE[:TAG] |

① 比如我现在有一个 spirngboot-web-01 镜像：

|  |
| --- |
| SQL [root@localhost ~]$ docker images REPOSITORY TAG IMAGE ID CREATED SIZE spirngboot-web-01 latest 1e1148e4cc2c 2 weeks ago 202MB |

我对 centos 进行开发，开发了第一个版本，我就可以对这个版本打标签，打完标签后会生成新的镜像：

|  |
| --- |
| SQL [root@localhost ~]$ docker tag spirngboot-web-01 spirngboot-web-01:v1 [root@localhost ~]$ docker images REPOSITORY TAG IMAGE ID CREATED SIZE spirngboot-web-01 latest 1e1148e4cc2c 2 weeks ago 202MB spirngboot-web-01 v1 1e1148e4cc2c 2 weeks ago 202MB |

我继续对 centos 进行开发，开发了第二个版本，继续打标签：

|  |
| --- |
| SQL [root@localhost ~]$ docker tag spirngboot-web-01 spirngboot-web-01:v2 [root@localhost ~]$ docker images REPOSITORY TAG IMAGE ID CREATED SIZE spirngboot-web-01 latest 1e1148e4cc2c 2 weeks ago 202MB spirngboot-web-01 v1 1e1148e4cc2c 2 weeks ago 202MB spirngboot-web-01 v2 1e1148e4cc2c 2 weeks ago 202MB |

以此类推，每开发一个版本打一个标签，如果以后我想回滚版本，就可以使用指定标签的镜像来创建容器：

|  |
| --- |
| SQL [root@localhost ~]$ docker run -itd spirngboot-web-01:v1 |

如果推送成功的话，命令返回结果大概如下：

|  |
| --- |
| SQL  docker@default:~$ docker push registry.baidubce.com/qiyu-live-test/qiyu-live-user-provider-docker:1.0.1 The push refers to a repository [docker.io/registry.baidubce.com/qiyu-live-test/qiyu-live-user-provider-docker]  464a44ea0195: Layer already exists  29b57e33a4da: Pushed  d649a240e453: Layer already exists  d0757a6730d0: Layer already exists  768dcfe5d05f: Layer already exists  f5cfc06b640d: Layer already exists  9669d6b73383: Layer already exists  latest: digest: sha256:1e7562a15ef1728f213922d9633be67f3025447d0a641e333a8ec5107749c386 size: 11802  docker@default:~$ |

实现了镜像的推送之后，我们就可以在远程服务器上拉取镜像，然后进行执行：

|  |
| --- |
| Java docker run -p 9090:9090 --name qiyu-live-user-provider-01  --add-host 'qiyu.nacos.com:相关ip'  --add-host 'cloud.db:相关ip'  --add-host 'qiyu.rmq.com:相关ip'  registry.baidubce.com/qiyu-live-test/qiyu-live-user-provider-docker:1.0.1 |