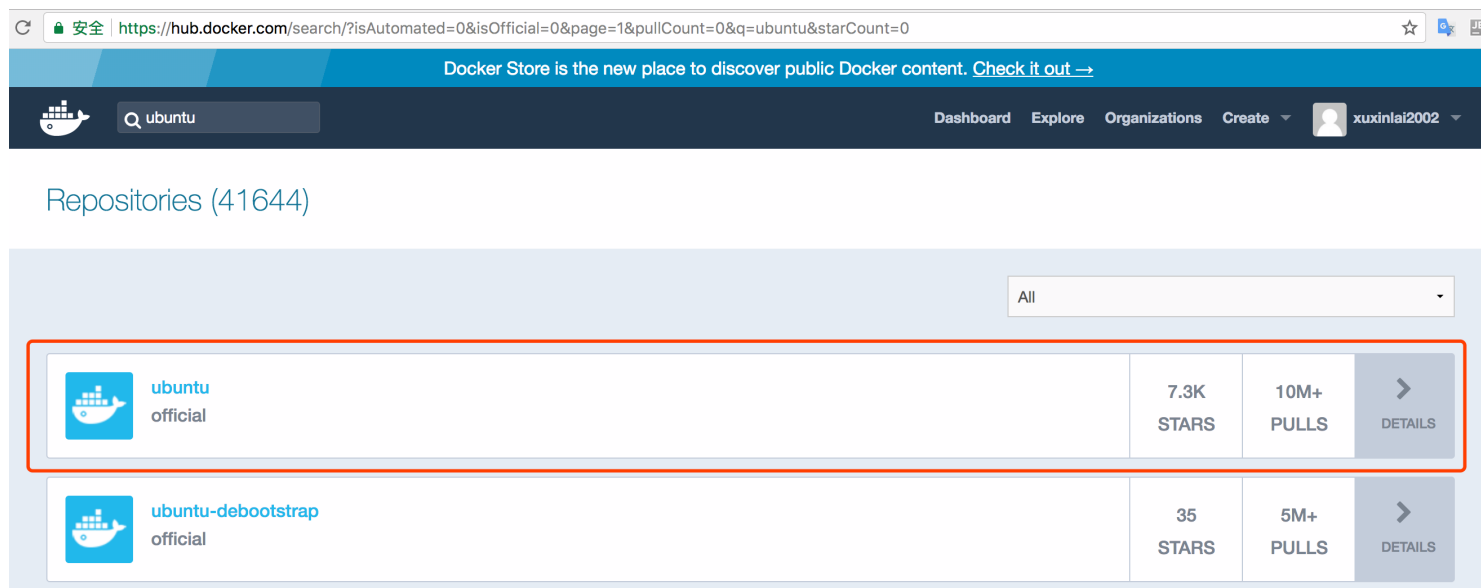


Mac OSX-docker Ubuntu truffle 智能合约创建和部署

一、Ubuntu环境搭建准备

1、从docker hub选择下载相关容器 <https://hub.docker.com/search/>



坑：1）在此网站下载需要注册，否则搜索不到结果；

2）在此网站注册难以成功，建议使用可以翻墙的手机进行注册。

2、docker加载操作系统

1）在MacBook Pro的lauchpad中的“其他”有个“终端”打开，同时打开docker软件

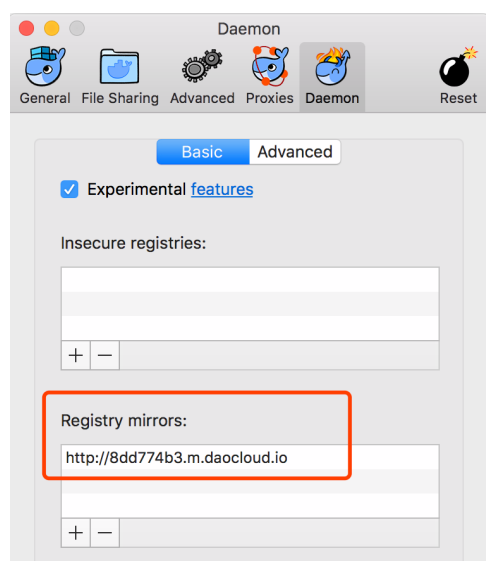
2）在终端中输入：docker pull ubuntu

```
xxl:blockchain xuxinlai$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
22dc81ace0ea: Pull complete
1a8b3c87dba3: Pull complete
91390a1c435a: Pull complete
07844b14977e: Pull complete
b78396653dae: Pull complete
Digest: sha256:ec6f15ffd0a8c9d6f52e2782dd02d91b3d3d2ce4b9faf6b58f40050a5be0406
Status: Downloaded newer image for ubuntu:latest
```

3、配置容器加速

1）打开网址：<https://www.daocloud.io/> ,然后完成注册流程；

2) 打开网址: <https://www.daocloud.io/mirror#accelerator-doc> , 选择macOS , 根据网页的提示进行操作;



坑: 1) 使用此网站的加速器同样需要注册, 普通方法注册不成功, 同样选择上面的方法, 使用翻墙的苹果手机进行注册;

2) 添加镜像地址的时候, 去桌面右上角查找。

二、Ubuntu环境搭建与设置

1、创建并进入容器实例 (说明: “实例”笔者是这样理解的并称为之。)

1) 在终端输入: `docker run -p 8080:8080 --name=test -it ubuntu /bin/bash`, 创建容器实例, 同时进入容器实例;

```
sh-3.2# docker run -p 8080:8080 --name=test -it ubuntu
root@c16c95a5fd7c:/#
```

2) 在终端输入: `cat /etc/issue`, 查看容器实例的版本;

```
[root@c16c95a5fd7c:/# cat /etc/issue
Ubuntu 18.04 LTS \n \l
```

2、ubuntu环境设置

1) 安装环境组件:

`apt-get update` //同步 /etc/apt/sources.list 和 /etc/apt/sources.list.d 中列出的源的索引,这样才能获取到最新的软件包

`apt-get install vim`

`apt-get install sudo`

2) 添加用户:

`adduser wyk` //添加用户

`su wyk` //切换用户

3) 用户权限操作:

`exit`

`chmod 777 /etc/sudoers`

`vim /etc/sudoers`

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
wyk     ALL=(ALL:ALL) NOPASSWD:ALL[]
# Members of the admin group may gain root
%admin  ALL=(ALL) ALL
```

`chmod 440 /etc/sudoers`

`su wyk`

三、truffle环境搭建

1、环境搭建准备

1) 安装nodejs , npm, testrpc

```
sudo apt-get install curl
curl -sL https://deb.nodesource.com/setup_8.x | sudo -E bash -

sudo apt-get install nodejs
node -v

sudo apt-get install npm
npm -v

sudo npm install -g ethereumjs-testrpc
```

2) 安装git, solc和solc-cli

```
sudo apt-get install git
sudo apt-get install software-properties-common
sudo add-apt-repository ppa:ethereum/ethereum
```

sudo apt-get update //更新 /etc/apt/sources.list 和 /etc/apt/sources.list.d 中列出的源的地址,这样才能获取到最新的软件包

```
sudo apt-get install solc
```

3) 安装truffle和geth

```
sudo npm install -g cnpm --registry=https://registry.npm.taobao.org
sudo cnpm install -g truffle@3.2.1

sudo add-apt-repository -y ppa:ethereum/ethereum
sudo apt-get update

sudo apt-get install ethereum
```

三、指令补充:

docker ps -a //查看docker中所有存活和挂掉的容器实例;

```
192:~ zimays$ docker ps -a
CONTAINER ID        IMAGE               PORTS              COMMAND              NAMES              CREATED
STATUS
8cec9f80ca38        ubuntu             privatechain       "/bin/bash"         privatechain       22 hours ago
Exited (127) 2 hours ago
6f4696d520a3        ubuntu             wyktest           "/bin/bash"         wyktest            2 days ago
Exited (100) 36 hours ago
7493e277acf0        ubuntu             stan               "/bin/bash"         stan               3 days ago
Up 2 hours
```

`docker start -ai casename/ID` //启动名称为casename的容器，此指令有利于再次启动之前操作过的容器；

`docker run --name=test -it ubuntu /bin/bash` //创建指定名字的容器实例；

`docker rm casename/caseid` //删除指定的容器实例

`sudo passwd root` //修改docker密码，例如123456

`docker run -p 127.0.0.1:8080:8080 --name=test -it ubuntu /bin/bash`//指定端口映射创建实例

`docker rename casename_old casename_new` //修改创建容器的名字，由casename_old变为casename_new；

四、智能合约部署

- 1、在根目录创建智能合约目录：

```
cd /
```

```
sudo mkdir truffletest
```

- 2、初始化：`sudo truffle init webpack`

- 3、修改配置文件：`sudo vim truffle.js/truffle-config.js`

内容修改如下：

```
module.exports = { networks: {
    development: {
      host: "localhost",
      port: 8545,
      network_id: "*" // mach any network id
    }
  }
};
```

- 4、编译：`sudo truffle compile`

- 5、启动ethereumjs-testrpc：`sudo testrpc &`

- 4、部署：`sudo truffle migrate`

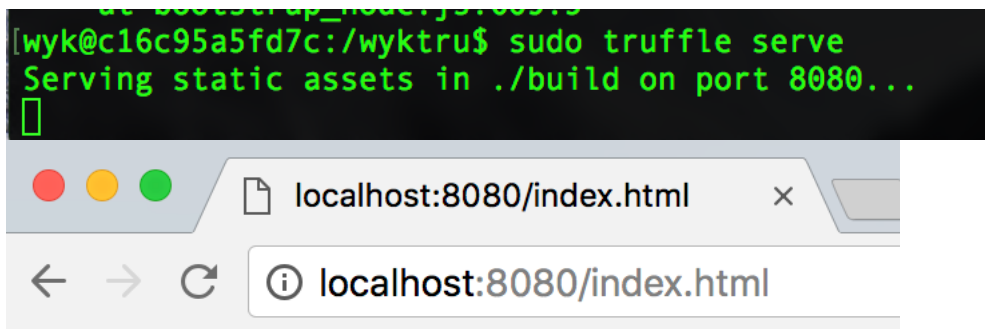
- 5、启动服务：`npm run dev/truffle serve`

6、浏览器查看：在浏览器输入：localhost:8080，查看部署结果

四、问题

1、首要问题浏览器无法查看结果：

1)truffle server 查看结果截图



Cannot GET /index.html

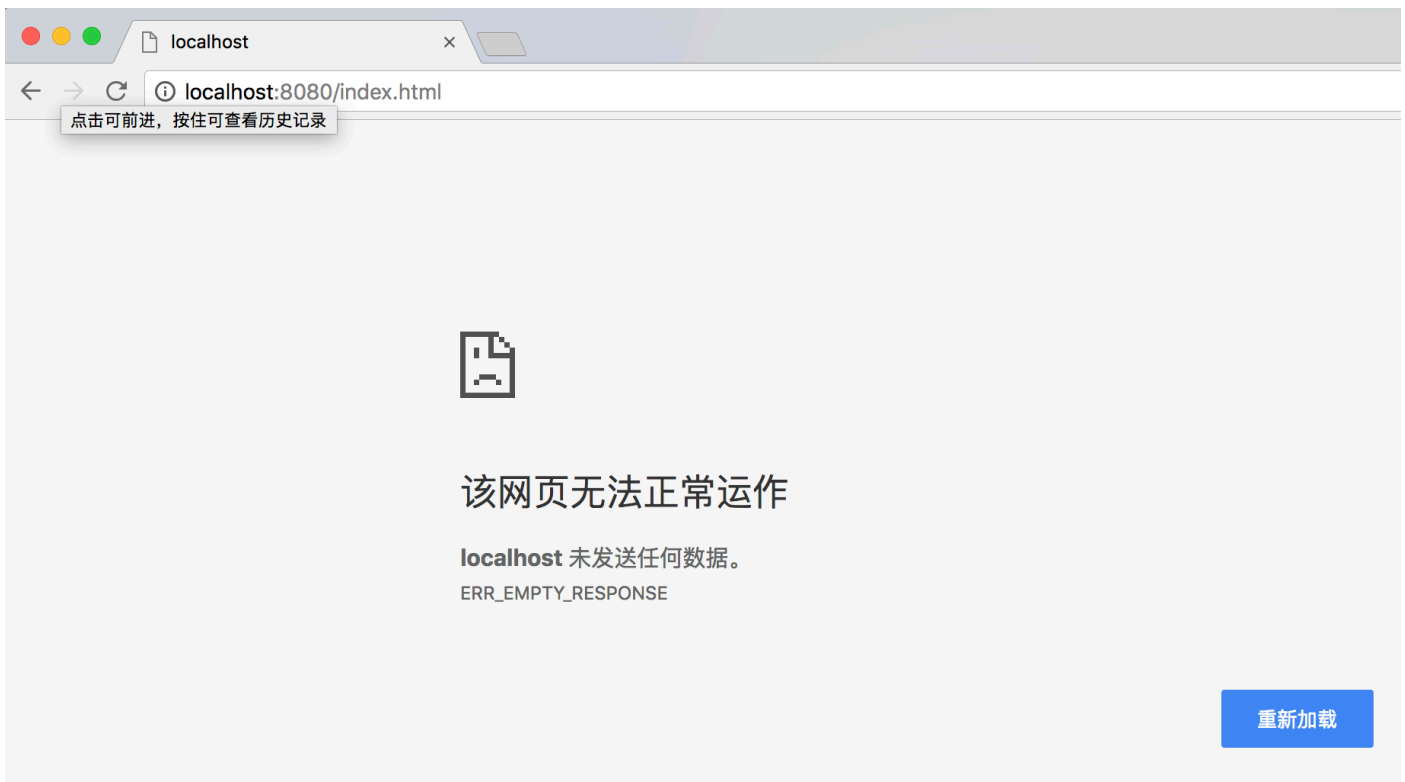
2) npm run dev 查看结果截图

```
wyk@c16c95a5fd7c:/wyktru$ sudo npm run dev
> truffle-init-webpack@0.0.2 dev /wyktru
> webpack-dev-server

Project is running at http://localhost:8080/
webpack output is served from /
Hash: e58f5537204c28dd77d9
Version: webpack 2.7.0
Time: 2225ms

  Asset      Size  Chunks             Chunk Names
  app.js    1.63 MB          0  [emitted]  [big]  main
  index.html 925 bytes          0  [emitted]

chunk    {0} app.js (main) 1.61 MB [entry] [rendered]
[
  [71] ./app/javascripts/app.js 3.64 kB {0} [built]
  [72] (webpack)-dev-server/client?http://localhost:8080 7.93 kB {0} [built]
  [73] ./build/contracts/MetaCoin.json 3.05 kB {0} [built]
  [111] ./~/loglevel/lib/loglevel.js 7.86 kB {0} [built]
  [117] ./~/querystring-es3/index.js 127 bytes {0} [built]
  [119] ./~/strip-ansi/index.js 161 bytes {0} [built]
  [122] ./app/stylesheets/app.css 905 bytes {0} [built]
  [163] ./~/truffle-contract/index.js 2.64 kB {0} [built]
  [197] ./~/url/url.js 23.3 kB {0} [built]
  [199] ./~/web3/index.js 193 bytes {0} [built]
  [233] (webpack)-dev-server/client/overlay.js 3.67 kB {0} [built]
  [234] (webpack)-dev-server/client/socket.js 1.08 kB {0} [built]
  [235] (webpack)/hot nonrecursive ^\.\/log$ 160 bytes {0} [built]
  [236] (webpack)/hot/emitter.js 77 bytes {0} [built]
  [237] multi (webpack)-dev-server/client?http://localhost:8080 ./app/javascripts/app.js 40 bytes {0} [built]
  + 223 hidden modules
webpack: Compiled successfully.
```



2、怀疑问题方向；

- 1) docker容器和宿主本机端口映射问题；
- 2) webpack环境问题；
- 3) 端口映射需要指明宿主本机的IP；
- 4) 宿主机的docker容器iptables不可用，可能有关联；
- 5) 流程参考资料：<https://www.aliyun.com/jiaocheng/133353.html>

2018年5月5日 星期六