数字货币交易所教程

一、准备工作

1、前端服务器：centos7.6 4核8G 40G硬盘

2、后端服务器：centos7.6 8核16G 100G固态

前端IP: 8.210.3.171 内网: 172.31.41.01

后端IP: 47.242.154.123 内网: 172.31.41.02

3、钱包服务器: centos7.6 4核8G 500G硬盘

4、域名: test.cn解析 @、www、api、admin指向前端服务器IP

5、阿里云OSS, 短信平台, 邮箱验证

二、安装宝塔并配置

1、安装宝塔

curl -sSO http://download.bt.cn/install/install\_panel.sh && bash install\_panel.sh

2、登录宝塔，前端服务器安装Nginx、MongoDB、Redis、Kafka、Mysql5.7(或使用RDS)

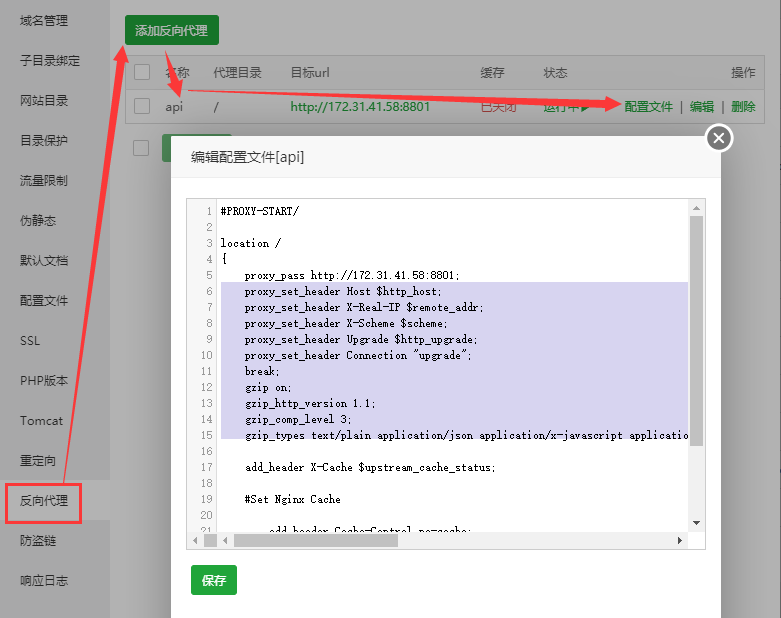
1. 、前端服务器Nginx配置新增如下附件(双击打开下面附件即可看到)

**(里面的172.19.76.150改为你自己的后端服务器ip)**

（2）、前端服务器添加agent.test.cn、api.test.cn、admin.test.cn、www.test.cn站点

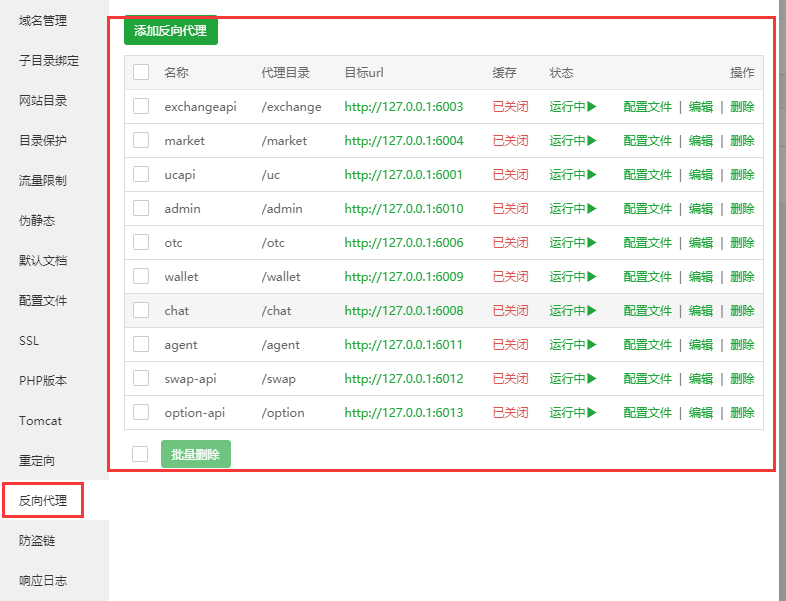
（3）、前端服务器配置api.test.cn反向代理设置

4、后端服务器添加一个内网IP: 172.31.41.02:8801 的站点

(1)、配置后端Jar包反向代理到此站点，如下图:

exchange、market、uc、admin、otc、wallet、chat、agent、swap、option



(2)、后端XX.XX.XX.XXX:8801站点反向代理——打开配置文件（如下图）

(双击打开附件即可看到)把里面的内容复制到上图红框内，覆盖原有内容（后端所有反向代理模块都需要替换一下）

(3)、后端安装mysql, 新建coin数据库，上传coinku.sql并导入coinku数据库

(4)、后端安装JDK，命令：yum install java-1.8.0-openjdk

(5)、后端服务器安装kafka并配置启动

nohup ./bin/zookeeper-server-start.sh ./config/zookeeper.properties >/dev/null 2>&1 &

nohup ./bin/kafka-server-start.sh ./config/server.properties >/dev/null 2>&1 &

(6)、安装MongoDB、Redis 配置密码

(7)、打包前端Vue——前台、后台、代理商

(8)、后端打包,修改默认 config文件里配置

\*服务器安全组和宝塔面板开放端口

22 80 443 8888 28901-28903 38901 48901 28985 38985 48958

后端启动命令（前三个必须按照顺序启动）

nohup java -Xms512m -Xmx512m -jar cloud.jar >/dev/null 2>&1 &

nohup java -jar exchange.jar >/dev/null 2>&1 &

nohup java -jar market.jar >/dev/null 2>&1 &

ps -ef|grep java 查看java运行包

前台

nohup java -jar ucenter-api.jar >/dev/null 2>&1 &

后台

nohup java -jar admin-api.jar >/dev/null 2>&1 &

聊天

nohup java -Xms512m -Xmx512m -jar chat.jar >/dev/null 2>&1 &

币币交易

nohup java -Xms512m -Xmx512m -jar exchange-api.jar >/dev/null 2>&1 &

钱包接口

nohup java -Xms512m -Xmx512m -jar wallet.jar >/dev/null 2>&1 &

场外交易

nohup java -Xms512m -Xmx512m -jar otc-api.jar >/dev/null 2>&1 &

合约

nohup java -Xms512m -Xmx512m -jar contract-swap-api.jar >/dev/null 2>&1 &

nohup java -Xms512m -Xmx512m -jar contract-option-api.jar >/dev/null 2>&1 &

代理

nohup java -Xms512m -Xmx512m -jar agent-api.jar >/dev/null 2>&1 &

机器人

nohup java -Xms512m -Xmx512m -jar er\_market.jar >/dev/null 2>&1 &

nohup java -Xms512m -Xmx512m -jar er\_robot\_normal.jar >/dev/null 2>&1 &

钱包RPC

nohup java -Xms512m -Xmx512m -jar usdt-1.2.jar >/dev/null 2>&1 &

nohup java -Xms512m -Xmx512m -jar bitcoin-1.2.jar >/dev/null 2>&1 &

nohup java -Xms512m -Xmx512m -jar eth-1.2.jar >/dev/null 2>&1 &

nohup java -Xms512m -Xmx512m -jar eos-1.2.jar >/dev/null 2>&1 &

nohup java -Xms512m -Xmx512m -jar erc-eusdt-1.2.jar >/dev/null 2>&1 &

**05\_Web\_Front（前端前台）**

**修改/05\_Web\_Front/src/main.js第32行**

**把 http://127.0.0.1 修改为 http://api.test.cn（可以是你前端服务器api.xx.com域名）**

**04\_Web\_Admin（前端后台）**

**搜索127.0.0.1:6010有两个文件存在**

**/04\_Web\_Admin/src/service/http.js第五行**

**/04\_Web\_Admin/src/config/api.js第三行**

1. **替换为后端服务器ip:6010直接访问后端admin-api.jar模块端口，但是可能会暴露后端服务器外网IP（注意防火墙和安全组是否开启6010端口）**
2. **替换成api.test.cn（使用域名时候不带端口号）**

**04和05项目命令**

**npm install**

**npm run dev (用于本地测试，如上传服务器可以直接忽略此命令)**

**npm run build**

**\*前端网页与后端Jar模块通讯是通过api.test.cn站点反向代理与后端站点的反向代理实现的**

**\*前端www.test.cn和admin.test.cn对应的是05\_Web\_Front和04\_Web\_Admin下的dist目录编译好的文件**

打包后端要求:

1. 项目用了Lombok插件，无论用什么IDE工具，请务必先安装Lombok插件  
   2、项目用了QueryDsl，如果遇见以Q开头的类找不到，请先编译一下对应的core模块，例如core、exchange-core、xxx-core这种模块  
   3、jdk版本1.8以上，找不到的jar包在项目jar文件夹下（有可能在项目目录下的lib目录）  
   4、初始化sql在sql文件夹中配置文件

5、所有配置基本都在\00\_framework\config\application-dev.properties (有的可能需要在java文件修改，这部分自行研究)  
 6、配置文件application.properties打开这个设置会自动建表  
 #jpa  
 spring.jpa.hibernate.ddl-auto=update

**钱包节点安装教程**

1. **BTC和USDT**

**1.1 下载节点并解压**

wget <https://github.com/OmniLayer/omnicore/releases/download/v0.8.2/omnicore-0.8.2-x86_64-linux-gnu.tar.gz>

**1.2 创建data目录和配置文件**

mkdir data

touch bitcoin.conf

**1.3 配置bitcoin.conf**

# 数据存储目录（此路径为上面建立的数据储存路径的完整路径）

datadir=/mnt/btc/data

# 使用测试网络（0：正式网，1：测试网）

testnet=0

# 告知 Bitcoin-Qt 和 bitcoind 接受JSON-RPC命令（是否启用命令和接受RPC服务）

server=1

# 设置 gen=1 以尝试比特币挖矿

gen=0

# 启用交易索引

txindex=1

# 后台执行（是否后台执行）

daemon=0

# 监听 RPC 链接,正式默认端口8333 测试默认18333（最好设置好，免得不清楚）

rpcport=8333

#RPC服务账号和密码，不设置的话是有默认密码的，本文没去深究默认，直接用自己设置的

rpcuser=bizzan

rpcpassword=123456789

#允许那些IP访问RPC接口，以下写法为默认所有ip都可访问，请自己修改成你自己的IP地址

rpcallowip=0.0.0.0/0

rpcconnect=127.0.0.1

**1.4 运行节点**

进入/mnt/btc/omnicore-0.8.2/bin目录执行

nohup ./omnicored -conf=/mnt/btc/bitcoin.conf -reindex &

**1.5 运行相应jar包**

1. **ETH**
   1. **下载节点并解压**

wget <https://gethstore.blob.core.windows.net/builds/geth-linux-amd64-1.9.15-0f77f34b.tar.gz>



* 1. **添加环境变量**

echo "export PATH=$PATH:/mnt/eth/geth-linux-amd64-1.9.15-0f77f34b" >> /etc/profile

**2.3 进入目录geth-linux-amd64-1.9.15-0f77f34b里,执行命令**

nohup geth --datadir /mnt/eth/data --cache 1024 --rpc --rpcapi web3,eth,net,db,personal,admin --rpccorsdomain "\*" --syncmode light --allow-insecure-unlock &

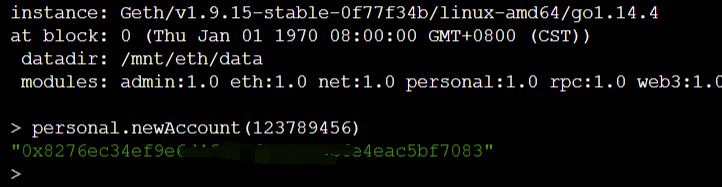
* 1. **进入geth控制台**

创建账户

geth attach http://127.0.0.1:8545

设置密码

personal.newAccount(123789456)



得到一串钱包地址填到rpc配置里

coin.ignore-from-address=0x8276ec34\*\*\*\*\*\*dc6e4eac5bf7083

配置coin.withdraw-wallet=下图文件名



配置coin.withdraw-wallet-password=123789456

**优化配置**

exchange\_order表,member\_transaction表加索引

启动命令优化参考

java -server -Djava.security.egd=file:/dev/./urandom -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Xms1g -Xmx1g -Xss256k -XX:+UseG1GC -XX:MaxGCPauseMillis=1000 -XX:InitiatingHeapOccupancyPercent=80 -XX:ParallelGCThreads=8 -XX:ConcGCThreads=8 -XX:+DisableExplicitGC -jar admin-api.jar