# Iotsecurecloud 云端部署文档

环境：全新centos7 64系统

## 部署环境准备

### 安装jdk

Jdk版本：1.8

将jdk 文件夹拷贝的系统中，进入到jdk文件夹，执行脚本jdk\_install.sh 安装jdk，执行完脚本后，执行如下命令：source /etc/profile 刷新环境变量。

安装是否成功测试命令：java -version,出现下图说明安装成功。



### 安装redis

将redis文件夹拷贝到系统中，

进入到redis文件夹执行如下命令：

1. tar -zxvf redis-5.0.5.tar.gz
2. cd redis-5.0.5 && make 安装redis
3. vi redis.conf 将此处修改为 yes

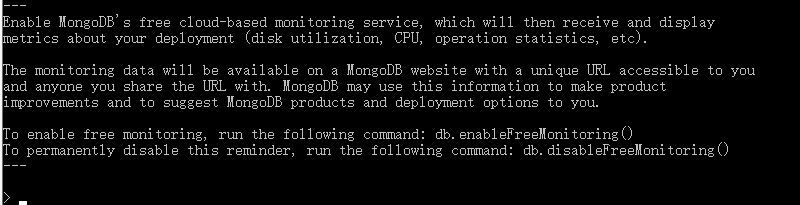


1. ./src/redis-server redis.conf 启动redis 使用默认端口6379，此处redis.conf和启动脚本redis-server不在一个目录下。

### 安装mongo

将mongo文件夹拷贝到系统目录下，进入目录执行如下命令：

1. mkdir -p /data/db 创建mongo数据存储目录
2. mkdir /usr/local/log 在log文件下创建mongo.log 文件
3. /mongo\_install.sh 执行脚本，执行完脚本默认前台启动，此时可强制退出。
4. source /etc/profile 刷新环境变量。
5. mongod --dbpath /data/db --logpath /usr/local/log/mongo.log --fork 后台启动mongodb。看到successfully 说明启动成功。
6. 执行mongo命令 出现如下画面



就可以在此处输入命令操作mongodb了。

use iotdb

> db.createCollection("device")

{ "ok" : 1 }

> db.createCollection("event")

{ "ok" : 1 }

> db.createCollection("module")

{ "ok" : 1 }

> db.createCollection("device\_event")

{ "ok" : 1 }

> show collections

device

device\_event

event

Module

Ctrl + c

1. mongo导入数据和导出数据命令。

cd /root/mongo/iotdb/

mongoimport -d iotdb -c device --file device.csv

2019-09-20T10:58:59.614+0800 connected to: localhost

2019-09-20T10:58:59.622+0800 imported 100 documents

[root@winter iotdb]# mongoimport -d iotdb -c event --file event.csv

2019-09-20T10:59:43.962+0800 connected to: localhost

2019-09-20T10:59:43.964+0800 imported 12 documents

[root@winter iotdb]# mongoimport -d iotdb -c module --file module.csv

2019-09-20T11:00:00.634+0800 connected to: localhost

2019-09-20T11:00:00.636+0800 imported 5 documents

[root@winter iotdb]# mongoimport -d iotdb -c device\_event --file device\_event.csv

2019-09-20T11:00:23.628+0800 connected to: localhost

2019-09-20T11:00:23.639+0800 imported 240 documents

use iotdb

switched to db iotdb

> db.device.find().pretty()

执行如下命令：检查数据导入是否成功。

1. NGINX安装

将nginx.tar.gz压缩包放入到/root 目录下

cd /usr/local/src/

mv /root/nginx/nginx-1.6.2.tar.gz .

[root@winter src]# mv /root/nginx/pcre-8.35.tar.gz .

tar -zxvf pcre-8.35.tar.gz

tar -zxvf pcre-8.35.tar.gz

cd pcre-8.35/

./configure

make && make install

pcre-config --version

cd ..

tar -zxvf nginx-1.6.2.tar.gz

cd nginx-1.6.2/

./configure --prefix=/usr/local/webserver/nginx --with-http\_stub\_status\_module --with-http\_ssl\_module --with-pcre=/usr/local/src/pcre-8.35

make

make install

/usr/local/webserver/nginx/sbin/nginx -v

nginx version: nginx/1.6.2

到此运行环境准备完毕。

## 后端部署

1. clone 代码：git clone <http://192.168.0.167/root/iotsecurecloud.git>
2. 进入到后端工程代码的E:\yanshi\iotsecurecloud\IoTBackEnd\eladmin-master pom.xml目录下执行 mvn package 打包 后端代码。

PS E:\yanshi\iotsecurecloud\IoTBackEnd\eladmin-master\eladmin-system\src\main\resources\config> cp \*.yml C:\Users\winter\Desktop\yanshi\iotBackEnd\

PS E:\yanshi\iotsecurecloud\IoTBackEnd\eladmin-master\eladmin-system\target> cp .\eladmin-system-2.1.jar C:\Users\winter\Desktop\yanshi\iotBackEnd

把iotBackEnd打成压缩包复制到服务器

tar -zxvf iotBackEnd.tar.gz

cd iotBackEnd/

vi application.yml 将redis密码设置为空

vi application-dev.yml 修改mongodb的地址127.0.0.1

./start.sh

tail -f all\_log.log

firewall-cmd --add-port=8001/tcp --permanent

firewall-cmd --add-port=80/tcp --permanent

firewall-cmd --reload

firewall-cmd --list-all

后端部署完成

## 前端部署

修改E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master\config\prod.env.js

BASE\_API: '"http://192.168.75.133:8001"'

后端IP地址和端口

mkdir E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master\static

进入到前端工程代码的E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master目下：执行

npm run build 命令打包前端代码

拷贝E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master\dist 到 C:\Users\winter\Desktop\yanshi\iotFront

打包 C:\Users\winter\Desktop\yanshi\iotFront 上传到服务器

vi /usr/local/webserver/nginx/conf/nginx.conf

修改location如下

/ {

root /root/iotFront/dist; dist上传路径

index index.html index.htm;

try\_files $uri $uri/ /index.html;

}

修改 user root

执行

/usr/local/webserver/nginx/sbin/nginx 命令启动NGINX即可。

1. 测试数据配置

E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master\src\views\device\list\index.vue

this.ws = new WebSocket("ws://192.168.75.133:8001/socketServer/aaa");

E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master\src\views\device\topography\index.vue

修改：

this.ws = new WebSocket("ws://192.168.75.133:8001/mapSocketServer/bbb");

修改：

E:\yanshi\iotsecurecloud\IoTFrontEnd\iotadmin-qd-master\src\views\security\eventanalysis\index.vue

this.ws = new WebSocket("ws://192.168.75.133:8001/analysisMapSocketServer/ccc");