**机构端检核平台安装部署**

目录

[**一、准备内容** 5](#_Toc50419521)

[**1、环境** 5](#_Toc50419522)

[**1）、前置要求及准备** 5](#_Toc50419523)

[**2）、信息确认** 5](#_Toc50419524)

[**二、基础环境部署** 6](#_Toc50419525)

[**1、部署工作节点调整** 6](#_Toc50419526)

[**2、安装java环境** 7](#_Toc50419527)

[**3、安装docker环境** 7](#_Toc50419528)

[**1)、yum安装** 7](#_Toc50419529)

[**2)、存储调整** 8](#_Toc50419530)

[**3)、重启docker** 8](#_Toc50419531)

[**4)、验证** 8](#_Toc50419532)

[**1** 新的存储目录生成相关目录文件 8](#_Toc50419533)

[**4、配置docker-compose** 9](#_Toc50419534)

[**5、导入镜像** 9](#_Toc50419535)

[**6、mysql容器配置** 9](#_Toc50419536)

[**三、机构端应用平台部署** 11](#_Toc50419537)

[**1、hive客户端安装配置** 11](#_Toc50419538)

[**2、HIVE库创建及初始化** 14](#_Toc50419539)

[**3、check\_script安装配置** 15](#_Toc50419540)

[**4、数据管控平台后端服务安装配置** 15](#_Toc50419541)

[**1)、前置需求** 15](#_Toc50419542)

[**2)、配置调整** 16](#_Toc50419543)

[**3)、启动服务** 16](#_Toc50419544)

[**5、数据管控平台前端服务安装配置** 17](#_Toc50419545)

[**1)、前端文件准备** 17](#_Toc50419546)

[**2)、前端配置调整** 17](#_Toc50419547)

[**6、自动检核服务安装配置** 19](#_Toc50419548)

[**1)、配置调整** 19](#_Toc50419549)

[**2)、启动服务** 20](#_Toc50419550)

[**四、验证** 20](#_Toc50419551)

[**1、机构数据准备** 20](#_Toc50419552)

[**2、文件上报状态查看** 21](#_Toc50419574)

[**机构端增强版：** 21](#_Toc50419575)

[**五、附录** 23](#_Toc50419591)

[**附录一 安装包获取** 23](#_Toc50419592)

**一、准备内容**

**1、环境**

**1）、前置要求及准备**

|  |  |
| --- | --- |
| **列表** | **要求** |
| 操作系统 | centos7.8 64bit |
| 硬件配置 | 8核32G,系统盘>=300G,数据盘>=500G,系统盘和数据盘必须分开 |
| 账号密码 | 安装部署使用账户为root用户 |
| 目录 | 创建目录：mkdir -p /source,方便存放安装包 |
| java环境 | 无需安装java，统一部署java环境 |
| 安装包获取 | 1.CentOS 7.8安装镜像，FTP+现场复制（5G）  2.CDH安装包，FTP+现场复制（80G）  3.应用包(含应用程序，Docker CE，MySQL镜像，Nginx镜像，JDK),FTP+现场复制 （1.5G）  4.BlockChain（含BlockChain镜像，REST（可信存证服务）镜像），FTP+现场复制（3G）  备注：ftp获取方式详见附录一 |
| 端口开通 |  |

**2）、信息确认**

上传部署压缩文件（eastworks-org.zip）存储目录：

/source

使用root登录部署环境后，先按照如下信息清单，确认节点具体系统环境。

|  |  |  |
| --- | --- | --- |
| **信息清单** | **执行命令** | **备注** |
| 内存资源 | free -mh | 详见《监管数据管控平台基础环境资源要求（增强版）》 |
| cpu资源 | cat /proc/cpuinfo |grep processor |wc -l | 详见《监管数据管控平台基础环境资源要求（增强版）》 |
| 磁盘确认 | mount |grep boot |grep sda |wc -l | 集群为实体机： mount |grep boot |grep sda |wc -l 结果>=1 集群为虚拟机： mount |grep boot |grep vda |wc -l 结果>=1 |
| mount |grep boot |grep vda |wc -l |
| fdisk -l |grep vda |grep Disk |awk '{print $3$4}' | >= 500G |
| fdisk -l |grep sda |grep Disk |awk '{print $3$4}' | >= 500G |
| 地址信息 | ip addr |  |
| 目录检查 | ll /data/ |grep dfs01 |wc -l | =1 |
| java环境 | java -version | 显示 -bash: java: command not found 为未安装 |
| 防火墙 | systemctl stop firewalld | 关闭 |
| systemctl is-enabled firewalld | disabled |
| selinux | getenforce | Disabled |
| cat /etc/selinux/config |grep SELINUX= |grep -v ^# | 需disabled |
| ssh相关极限配置检查 | cat /etc/ssh/sshd\_config |grep PermitRootLogin | 需开启root密码ssh登录配置，执行命令 配置参数为 yes |

**二、基础环境部署**

本文部署以host01为例，进行配置部署。

|  |  |  |  |
| --- | --- | --- | --- |
| 主机名 | IP地址 | 别名 | 备注 |
| host01.iwellmass.com | 10.0.0.1 | east-01 | Nginx |
| auto\_check\_org |
| east\_org\_front |
| east\_org |
| host02.iwellmass.com | 10.0.0.10 | east-02 | mysql |

**下文部署使用的用户为 系统root用户！**

**1、部署工作节点调整**

确保以上验证及准备工作完成后，以系统用户root登录hosts01节点，并使用root用户对下文命令操作

创建应用运行目录

mkdir -p /data/dfs01/east

切换目录并解压任务文件

cd /source/

unzip eastworks-org.zip

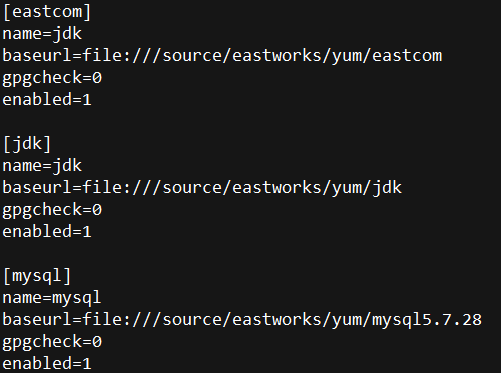
备份本地yum配置文件

mkdir -p /tmp/repobak

mv /etc/yum.repos.d/\* /tmp/repobak/

拷贝本地源文件到hosts01相关yum配置目录下

cp /source/eastworks-org/east-local.repo /etc/yum.repos.d/



验证源

yum clean all

yum repolist

结果显示类似以下内容为正常

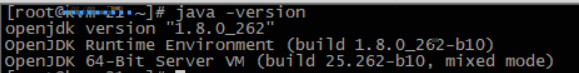


**2、安装java环境**

yum install java

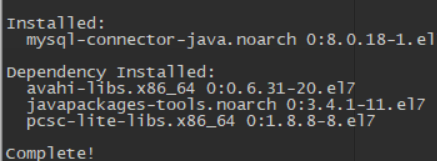
执行如下命令，验证java服务

java -version



安装插件

yum install mysql-connector-java-8\*



**3、安装docker环境**

**1)、yum安装**

yum install docker-ce

**2)、存储调整**

创建容器存储目录

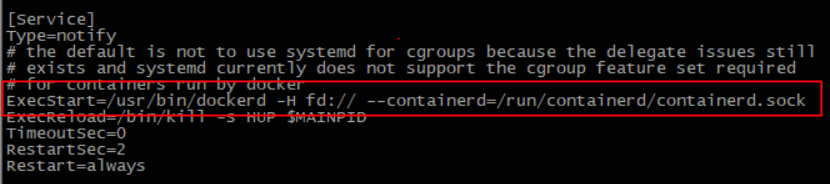
mkdir -p /data/dfs01/east/docker

修改docker.service文件，使用-g参数指定存储位置

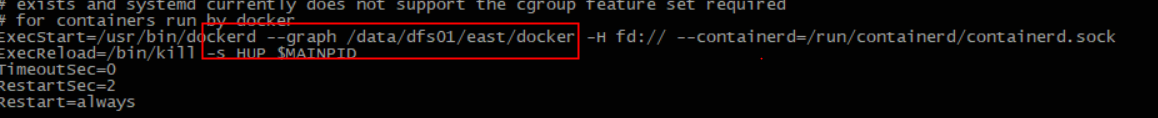
vi /usr/lib/systemd/system/docker.service

ExecStart=/usr/bin/dockerd --graph /data/dfs01/east/docker

修改前



修改后



重新加载配置文件

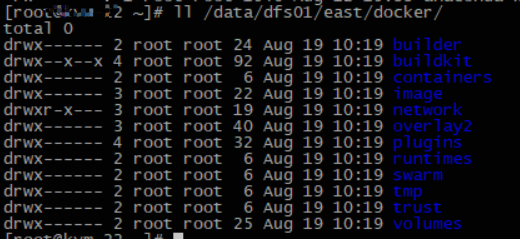
systemctl daemon-reload

**3)、重启docker**

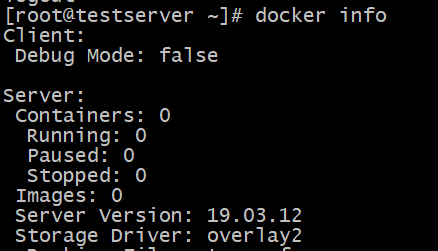
systemctl restart docker.service

**4)、验证**

**1** 新的存储目录生成相关目录文件



**2** 在命令行输入docker info ，输出如下内容，表示docker服务启动正常



**4、配置docker-compose**

cp /source/eastworks-org/docker-compose /usr/local/bin/

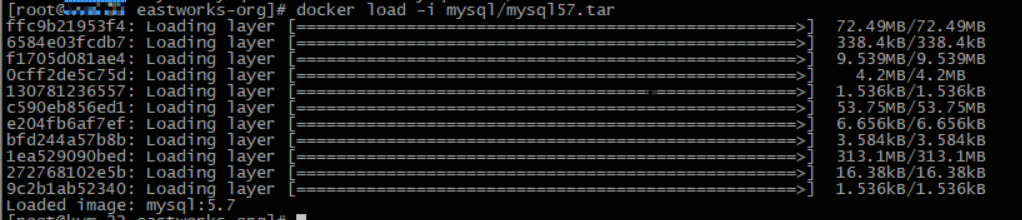
chmod 755 /usr/local/bin/docker-compose

验证

whereis docker-compose

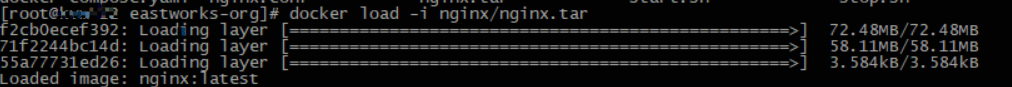


**5、导入镜像**



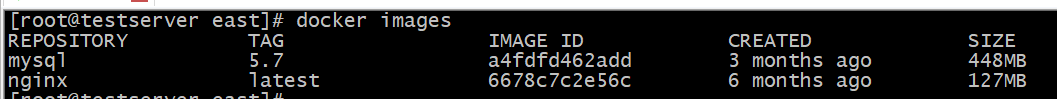
导入nginx基础镜像

docker load -i nginx/nginx.tar



验证

执行docker images命令，显示如下，表示镜像导入成功



**6、mysql容器的安装配置（需要在检核应用的mysql数据库服务器上操作）**

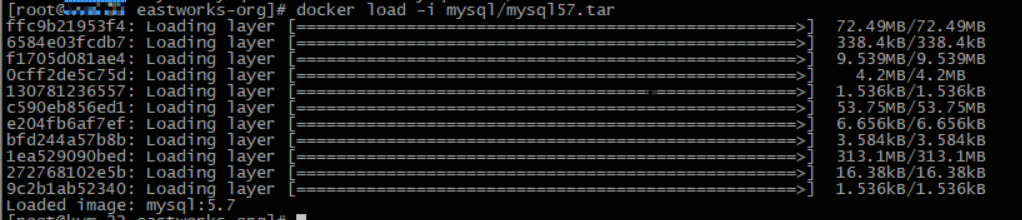
Mysql 服务需要单独部署在host02服务器上，做到应用和数据相互隔离。

上面的1，3，4步骤需要在host02上执行

cd /source/eastworks-org

导入mysql基础镜像

docker load -i mysql/mysql57.tar

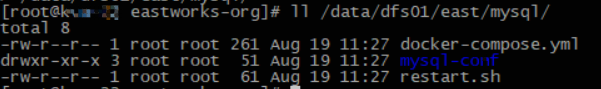


创建应用目录

mkdir -p /data/dfs01/east/mysql

cp -r /source/eastworks-org/mysql/{docker-compose.yml,mysql-conf,restart.sh} \

/data/dfs01/east/mysql/



启动mysql容器

cd /data/dfs01/east/mysql

docker-compose up -d



docker ps



应用库初始化

查询mysql容器ID，登录容器使用。本文mysql容器ID为23e13658b7c5



将初始化SQL拷贝入容器

docker cp /source/eastworks-org/mysql/east\_org\_v2.2.sql 23e13658b7c5:/

docker cp /source/eastworks-org/mysql/v2.2\_1t\_sys\_person\_in\_charge.sql 23e13658b7c5:/

docker cp /source/eastworks-org/mysql/v2.2\_2.sql 23e13658b7c5:/

注：23e13658b7c5 为本文容器ID，机构端操作此步骤，请确认本地ID

登录容器

docker exec -ti 【mysql容器ID】 /bin/bash

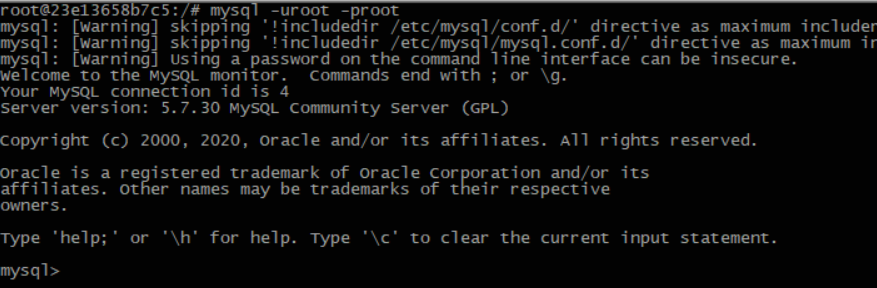
本文：

docker exec -ti 23e13658b7c5 /bin/bash

登录容器后，执行命令 ls -l 查看验证sql文件

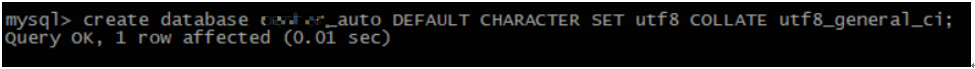
保持登录mysql容器，创建数据库并初始化

mysql -uroot -proot



执行创建库语句，创建应用库

create database org\_auto DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci



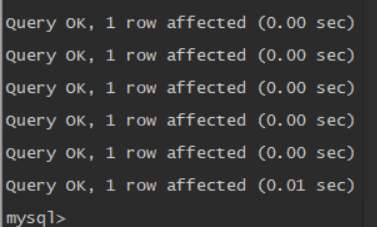
执行SQL语句，初始化数据库

use org\_auto;

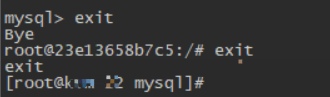
source /east\_org\_v2.2.sql;

source /v2.2\_1t\_sys\_person\_in\_charge.sql;

source /v2.2\_2.sql;



exit退出数据库，exit退出容器。



如果某些机构需要自建mysql服务，建议使用mysql5.7版本，

需要注意如下配置参数：

[mysqld]

collation-server = utf8\_unicode\_ci

init-connect='SET NAMES utf8'

character-set-server = utf8

lower\_case\_table\_names = 1

skip-name-resolve

sql\_mode=STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,NO\_AUTO\_CREATE\_USER,NO\_ENGINE\_SUBSTITUTION

max\_connections=1000

default-time-zone='+8:00'

备注：如果有条件的机构，可以在能够访问检核应用mysql数据库的windows机器中安装navicat 软件, 这样操作mysql更加方便。

**三、机构端应用平台部署**

**1、hive客户端安装配置**

**此步骤，需要远程拷贝本机构CDH集群上的hive相关应用文件到本地检核应用服务器（本文为host01）。根据设备节点间联通情况，选择下文方式进行拷贝。端口信息列表中端口策略都已确认允许。**

本地应用节点IP地址：10.0.0.1

创建hive客户端目录

mkdir /opt/cloudera

cd /opt/cloudera

1)、网络联通情况

scp -r 10.0.0.2:/opt/cloudera/parcels/CDH-6.0.1-1.cdh6.0.1.p0.590678/lib/hadoop\* \

opt/cloudera/

**注：上文是一条完成的命令, 10.0.0.2为本机构的CDH集群服务器的IP**

继续执行：

scp -r 10.0.0.2:/opt/cloudera/parcels/CDH-6.0.1-1.cdh6.0.1.p0.590678/lib/hive \

opt/cloudera/

**注：上文是一条完成的命令**

2)、网络不连通情况：

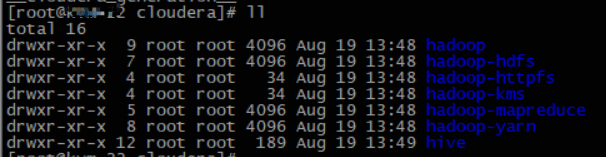
通过U盘或其他介质，登录CDH集群后，将如下内容进行拷贝：

/opt/cloudera/parcels/CDH-6.0.1-1.cdh6.0.1.p0.590678/lib/hadoop\*

/opt/cloudera/parcels/CDH-6.0.1-1.cdh6.0.1.p0.590678/lib/hive

注：cp命令 使用-LR参数，将软连接转换为真实文件拷贝。

拷贝后复制到本应用节点设备的 opt/cloudera/ 目录下。



**环境变量调整**

注： 由于执行应用程序的用户环境问题，可将环境变量添加到 /etc/profile。

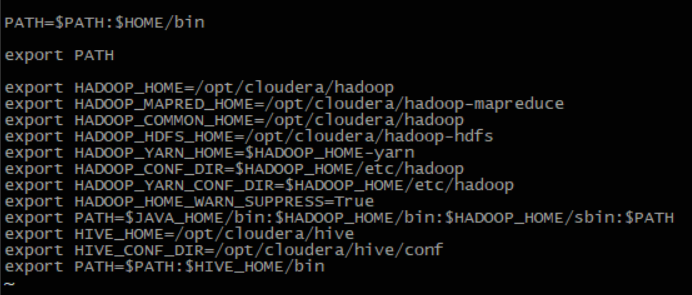
普通用户添加环境变量到普通用户家目录的 /home/用户/.bash\_profile中

备份配置文件

cp /root/.bash\_profile /root/.bash\_profile.`date +%Y%m%d`

使用root用户执行以下命令，进行环境变量调整：

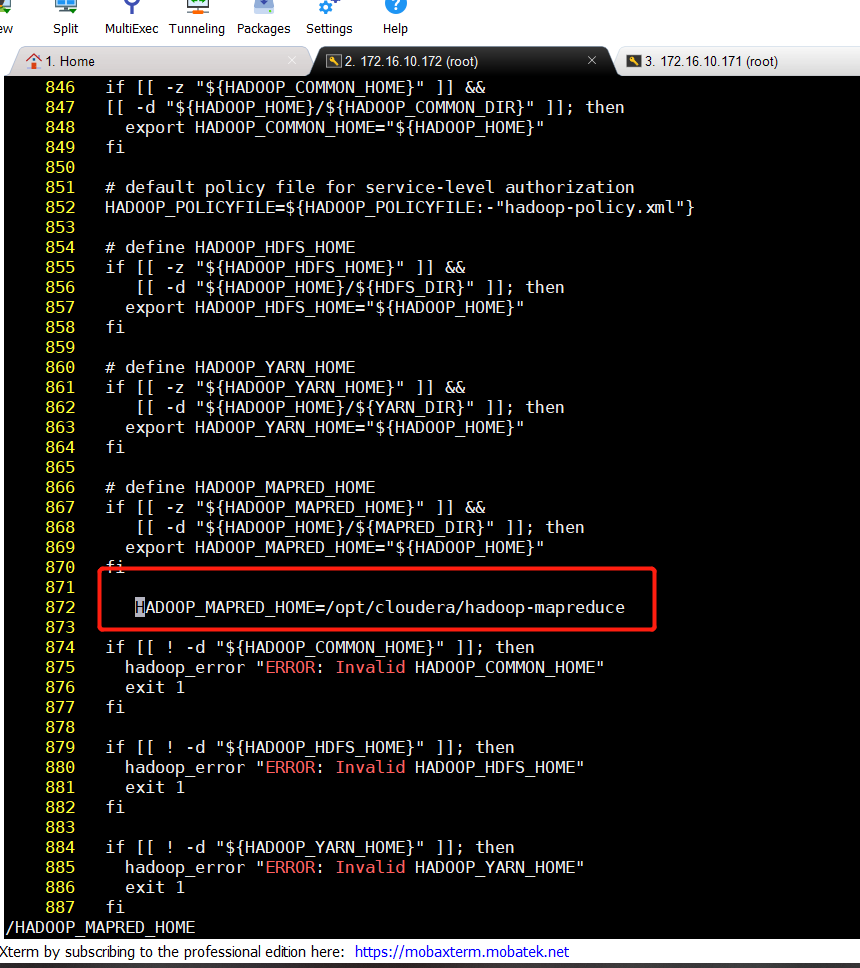
|  |
| --- |
| echo "export HADOOP\_HOME=/opt/cloudera/hadoop" >> /root/.bash\_profile |
| echo "export HADOOP\_MAPRED\_HOME=/opt/cloudera/hadoop-mapreduce" >> \ /root/.bash\_profile |
| echo "export HADOOP\_COMMON\_HOME=/opt/cloudera/hadoop" >> /root/.bash\_profile |
| echo "export HADOOP\_HDFS\_HOME=/opt/cloudera/hadoop-hdfs" >> /root/.bash\_profile |
| echo "export HADOOP\_YARN\_HOME=\$HADOOP\_HOME-yarn" >> /root/.bash\_profile |
| echo "export HADOOP\_CONF\_DIR=\$HADOOP\_HOME/etc/hadoop" >> /root/.bash\_profile |
| echo "export HADOOP\_YARN\_CONF\_DIR=\$HADOOP\_HOME/etc/hadoop" >> /root/.bash\_profile |
| echo "export HADOOP\_HOME\_WARN\_SUPPRESS=True" >> /root/.bash\_profile |
| echo "export PATH=\$JAVA\_HOME/bin:\$HADOOP\_HOME/bin:\$HADOOP\_HOME/sbin:\$PATH" >> \ /root/.bash\_profile |
| echo "export HIVE\_HOME=/opt/cloudera/hive" >> /root/.bash\_profile |
| echo "export HIVE\_CONF\_DIR=/opt/cloudera/hive/conf" >> /root/.bash\_profile |
| echo "export PATH=\$PATH:\$HIVE\_HOME/bin" >> /root/.bash\_profile |



执行如下命令修改 hadoop-functions.sh 防止环境检查失败

vi /opt/cloudera/hadoop/libexec/hadoop-functions.sh

HADOOP\_MAPRED\_HOME=/opt/cloudera/hadoop-mapreduce



主机名调整

使用root用户修改/etc/hosts文件，将CDH集群中的所有主机的主机名及IP解析加入到检核应用服务器（hive客户端所在设备）的hosts文件中。具体域名及IP根据实际情况调整。

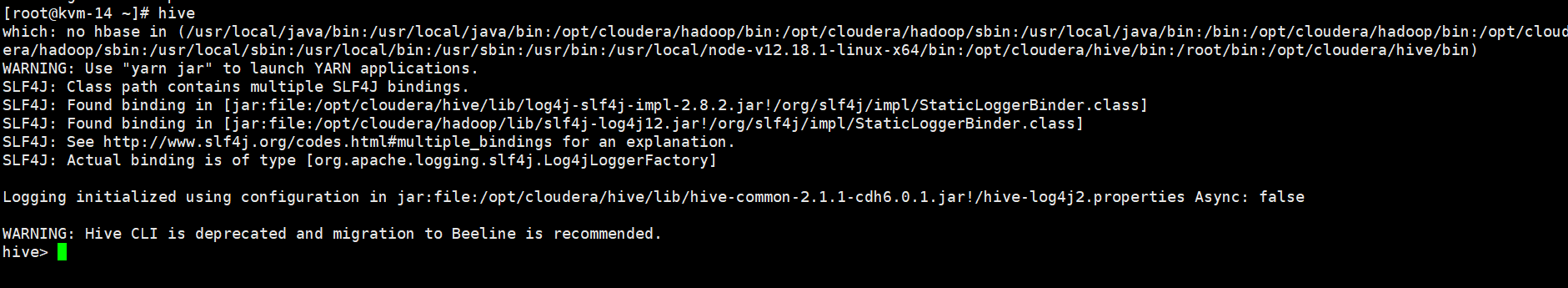
示例：

vi /etc/hosts

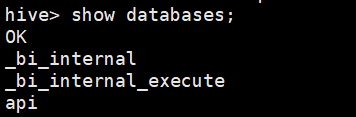
10.0.0.1 cdh-01.iwellmass.com

验证

命令行输入hive,显示如下内容，



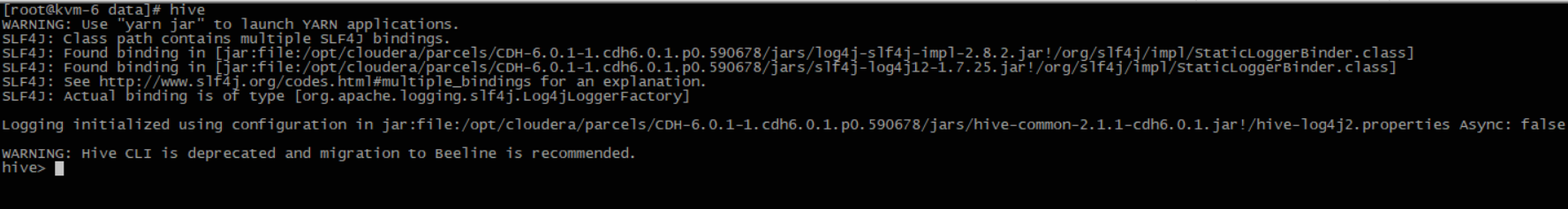
再输入show databases; 能很快返回结果，则表示hive客户端安装成功



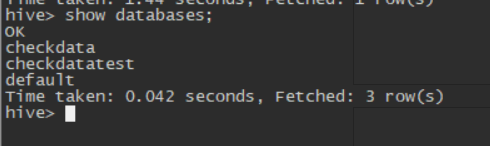


**2、HIVE库创建及初始化**

登陆hive数据库



查询hive数据库

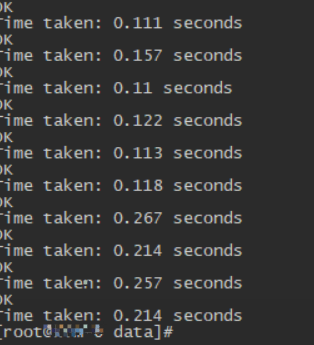


根据实际情况新建hive数据库（新建的数据库不能在库中存在）

hive> create database checkdata;

hive数据库初始化

hive -f /data/checkdata.sql

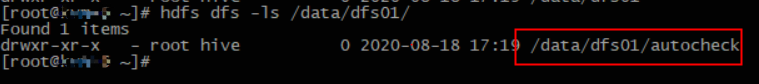


**3、check\_script安装配置**

**注：此步骤配置在CDH集群节点操作**。

以root，登录CDH集群节点后，上传 checkScript-2.2.jar

查看hdfs存储目录是否存在



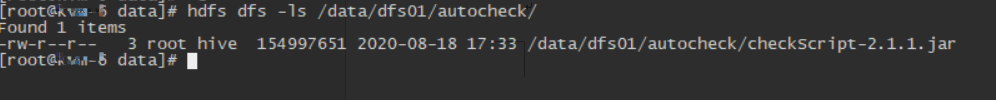
如不存在，进行创建

hdfs dfs -mkdir /data/dfs01/autocheck

上传脚本文件

hdfs dfs -put -f checkScript-2.1.1jar /data/dfs01/autocheck/

hdfs dfs -ls /data/dfs01/autocheck/



**4、数据管控平台后端服务安装配置**

**1)、前置需求**

进行如下信息的

|  |  |  |
| --- | --- | --- |
| **清单** | **命令** | **备注** |
| mysql数据库IP 默认端口（3306） | telnet 10.0.0.1 3306 | 连通 |
| livy服务器IP 默认端口（8998） | telnet 10.0.0.2 8998 | 连通 |
| rabbitmq服务器 端口 （5672） | telnet 19.116.250.81 5672 | 连通 |
| 区块链环境IP 端口 （9000） | telnet 10.0.0.10 9000 | 连通 |
| ftp服务端IP 端口(21，20000-23000) | telnet 19.116.250.81 21 | 连通 |
| hdfs-browsefiles服务端IP 端口(9866) | telnet 10.0.0.x 9866 | 连通 |
| zookeeper服务端IP 端口(2181) | telnet 10.0.0.x 2181 | 连通 |
| hive.Metastore服务端IP 端口(9083) | telnet 10.0.0.x 9083 | 连通 |
| hdfs.namenode服务端IP 端口(8020) | telnet 10.0.0.x 8020 | 连通 |
| hive.server服务端IP 端口(10000) | telnet 10.0.0.x 10000 | 连通 |

**注：本文mysql数据库节点地址为10.0.0.1 ； livy服务器节点IP：10.0.0.2 ；cdh集群的IP:10.0.0.x**

**根据操作环境具体IP进行连通性测试。**

**局端MQ 地址： 19.116.250.81，局端FTP 地址：19.116.250.81**

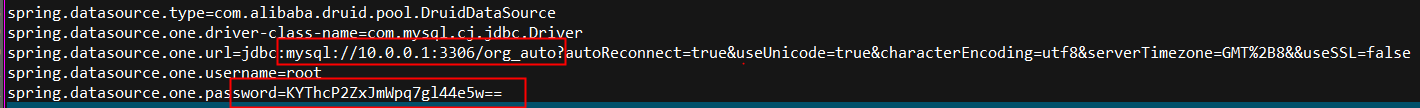
**注：注意设备中不同版本的服务端口冲突及配置文件、脚本文件中端口的定义准确。**

**2)、配置调整**

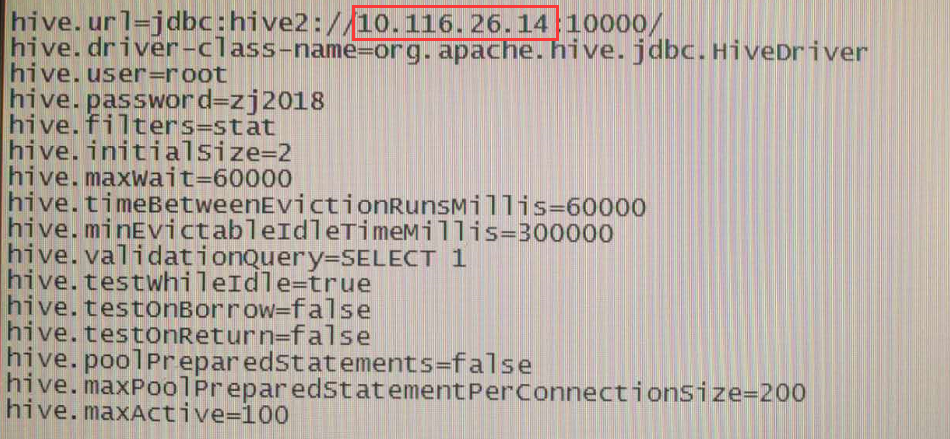
cp -r /source/eastworks-org/east\_org /data/dfs01/east/

cd /data/dfs01/east/east\_org/

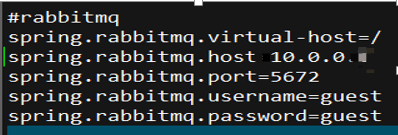
vi application-prod.properties



**注：密码为密文。红框部分修改为我们安装的mysql地址及相关账户密码。即host02（10.0.0.10）的地址**



**注：此地址为 CDH集群HiveServer2 地址**



**注：此地址为 局端 安装的MQ所在节点的地址19.116.250.81**

**3)、启动服务**

cd /data/dfs01/east/east\_org

chmod 755 start.sh stop.sh

./start.sh eastOrg-2.2.jar

验证：

ss -lntp |grep 8098



**5、数据管控平台前端服务安装配置**

**注：端口问题，如运行多个版本服务，请注意前端服务的端口冲突及转发后端服务的端口配置**

**1)、前端文件准备**

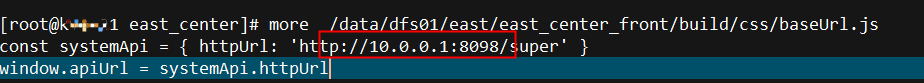
拷贝前端文件

cp -r /source/eastworks-org/east\_org\_front /data/dfs01/east/

**2)、前端配置调整**

修改配置IP信息，本文为本机IP地址

vi /data/dfs01/east/east\_org\_front/build/css/baseUrl.js



**3)、web访问验证**

浏览器输入：http://10.0.0.1 账户admin 密码admin



**4)、参数调整**

登录web界面后依次点击 系统管理-参数管理，进行参数的具体调整。





样本号忽略

业务数据库名称: hive数据库的名称，使用之前初始化的hive数据库

Livy服务器地址: 填写安装cdh集群时部署的livy服务器的地址

livy上传到hdfs的路径: 填写上面部署的check\_script服务的绝对路径

区块链服务地址：填写本机构端部署的区块链的服务器IP地址

driver 容器内存：spark.driver.memory

driver 容器核数：spark.driver.core

executor容器内存：spark.executor.memory

executor 容器核数：spark.executor.core

备注：driver和executor是spark执行任务的两种container容器，每个application执行时会启动1个driver容器和若干个executor容器。执行spark任务时，yarn资源会调度给每个container分配的核数和内存

container的内存，核数配置大一些，并发就会小些，container执行能力强一点，

container内存和核数不要配置得过小，可能会导致任务执行会报错。

根据cdh集群的实际资源情况配置，建议容器的内存：核数=4：1

自动扫描文件基目录：本文使用该地址/data/dfs01/east/autoscanpath

**注：/data/dfs01/east/autoscanpath 目录，是本文默认配置的标准上传目录，此目录在数据盘，各机构请核对本地环境大数据盘具体位置。部分银行数据盘为 /udv/data**

修改银行相关信息；



备注： 银行机构代码和金融许可证需特别注意，请务必填写正确，写错会影响检核应用。

其他内容也请根据机构实际情况填写

**6、自动检核服务安装配置**

**1)、配置调整**

拷贝服务应用

cp -r /source/eastworks-org/auto\_check\_org /data/dfs01/east/

mkdir -p /data/dfs01/east/autoscanpath

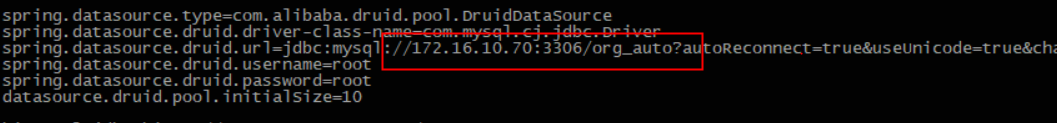
mkdir -p /data/dfs01/east/autoscanpath/Bxxxxxxxxxx (创建以机构金融许可证命名的目录)

备注：Bxxxxxxxxxx 为本机构的金融许可证

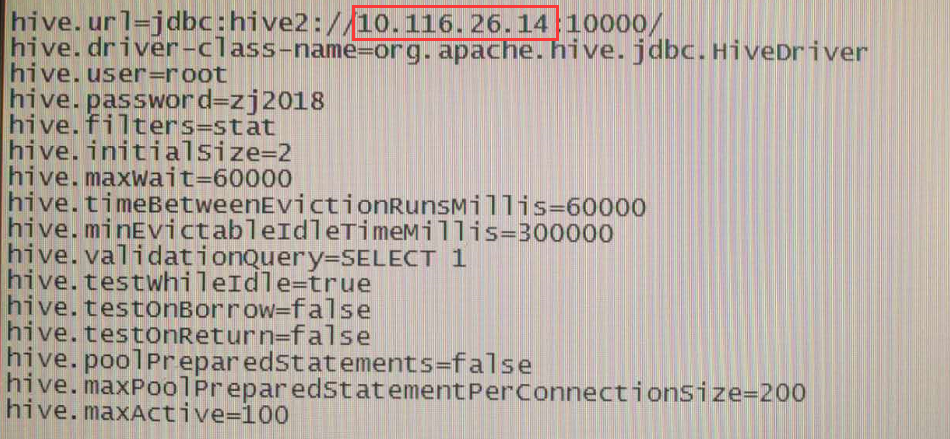
服务配置调整

cd /data/dfs01/east/auto\_check\_org/

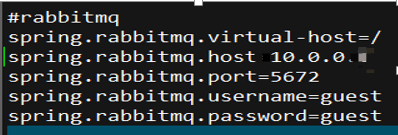
vi application-prod.properties



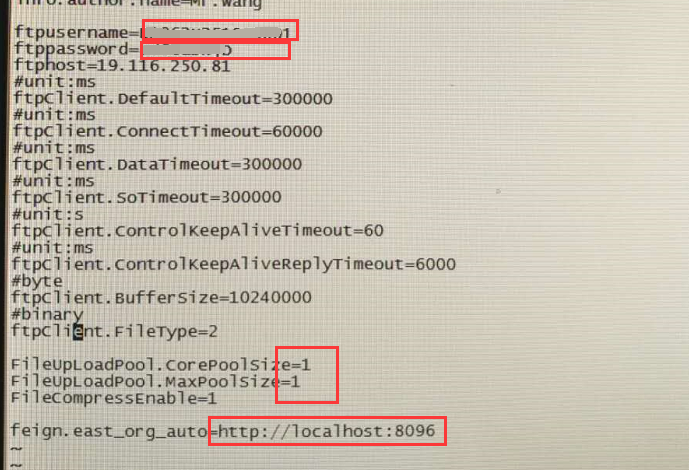
**注：红框部分修改为我们安装的mysql地址及相关账户密码。即本机10.0.0.1的地址**



**注：此地址为 CDH集群HiveServer2 地址**



**注：此地址请修改为局端MQ服务所在节点的地址19.116.250.81**



**注：**

**ftpusername:ftp账号**

**ftppassword:ftp密码**

**ftphost: 局端ftpserver服务器地址 19.116.250.81**

**注：ftp的用户密码，为新设置的用户密码，账号为各银行机构金融许可证号，密码请找胡洋老师查询。**

**注： 在配置文件最下方，有后端数据管控平台后端服务端口配置。**

**FileUpLoadPool.CorePoolSize=1**

**FileUploadPool.MaxCorePoolSize=1**

**Feign.east\_org\_auto 根据实际的后端检核应用（east\_org）的监听端口填写**

**2)、启动服务**

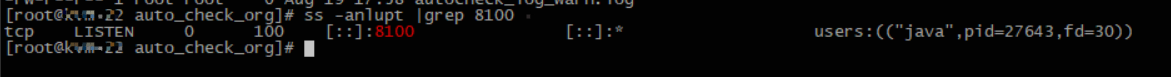
cd /data/dfs01/east/auto\_check\_org

chmod 755 start.sh stop.sh

./start.sh autoCheckOrg-2.2-enhance.jar

验证：

ss -anlupt |grep 8100



**注：**

机构端自动检核服务包说明

autoCheckOrg-2.2-enhance.jar CDH集群适配版本

**四、验证**

**1、机构数据准备**

要求：

T+1

状态表全量，在接口文件名主体部分增加-ALL（BXXXXXXXXXXXXX-JGXXB-CJRQ-ALL.TXT）

状态表增变量（保持接口文件表名）

明细表，保持接口文件表名

总计 204个文件（包含\*.txt,\*.log,\*-ALL.txt,\*-ALL.log）

T+月

状态表全量，在接口文件名主体部分增加-ALL（BXXXXXXXXXXXXX-JGXXB-CJRQ-ALL.TXT）

明细表，保持接口文件表名

总计 118个文件（包含\*.txt,\*.log,\*-ALL.txt,\*-ALL.log）

准备好文件后，请在之前创建的/data/dfs01/east/autoscanpath/Bxxxxxxx 目录下面新建日期时间目录，

例如（linux方式）：

**cd** /data/dfs01/east/autoscanpath/Bxxxxxxx

mkdir 20200908 && cd 20200908

mv /xxxxx/\*.txt . #注意一定要先传.txt文件，后传.log后缀的文件

mv /xxxxx/\*.log .

通过windows 服务器传文件

可以在windows上安装winscp客户端，通过winscp连到检核应用服务器，然后新建20200908目录，最后传文件到/data/dfs01/east/autoscanpath/Bxxxxxxx/20200908目录下面

备注：

a) winscp连接检核服务器的账号必须对/data/dfs01/east/autoscanpath/Bxxxxxx/目录有写权限

b)先传.txt文件，后传.log 文件，顺序不能搞反

**2、文件上报状态查看**

**机构端增强版：**

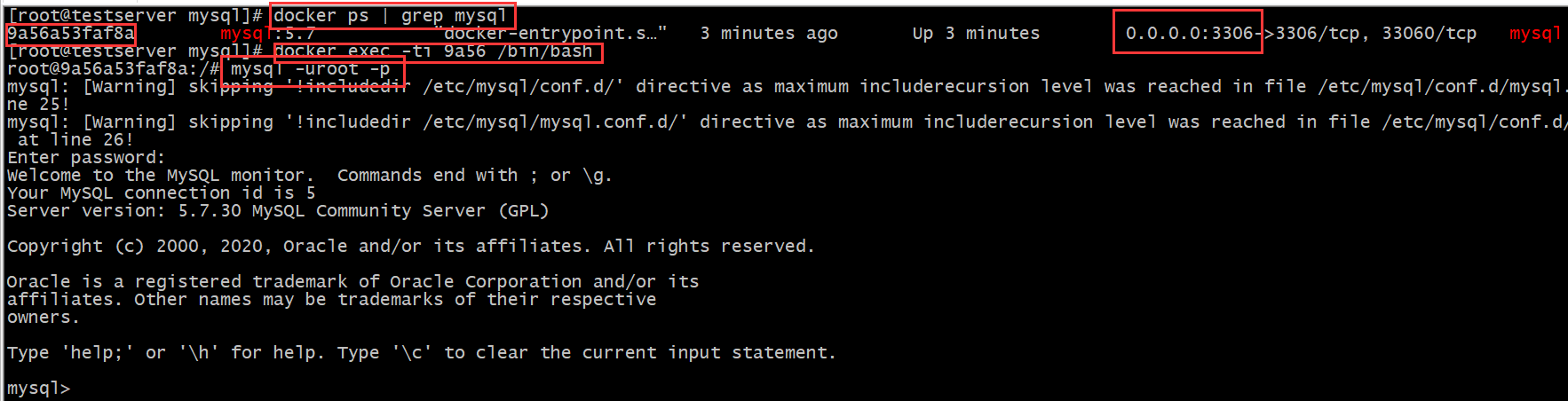
**登陆到检核应用的mysql中**

**docker ps | grep mysql**

**docker exec –ti 9a56 /bin/bash**

**mysql –uroot -p #输入mysql的密码（可在docker-compose.yml中配置）**

**use org\_auto;**



**然后输入** select \* from auto\_check\_result where org\_no='titic20200731' and scan\_date='20200903';

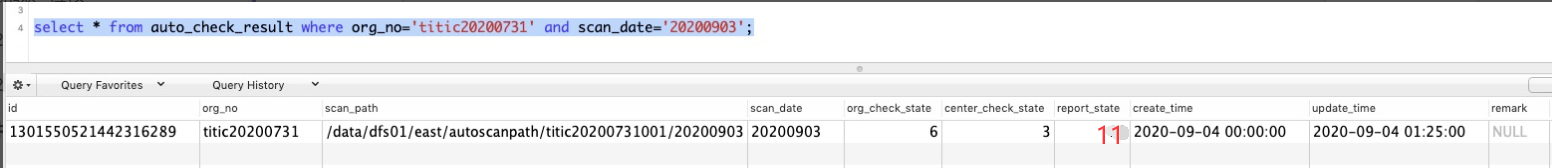
备注： org\_no修改为本机构的银行机构代码，scan\_date修改为检核日期

**如果扫描结果如下，则表示机构端自动检核成功，文件上报成功，并局端加载成功**

**org\_check\_state 为 6**

**center\_check\_state 为3**

**report\_state 为11**



**五、附录**

**附录一 安装包获取**

安装包获取方式：

方式1 请各机构准备容量大于100G的移动硬盘，到局里找胡洋老师拷贝安装包

方式2 通过ftp方式，从局端下载安装包到本地服务器

|  |  |
| --- | --- |
| 局端ftp服务器地址 | 19.116.250.81 |
| ftp账号/密码 | yinhang2/ Y123!dKLP |
| 文件存放目录 | Application:检核应用  mgp: 区块链安装包  cdh: 大数据安装包  iso: centos7.8 64bit镜像 |