# 服务器信息

## 基本信息

操作系统：CentOS release 6.8

cpu/内存：2核/8G

## 配置主机名

使用root用户

修改/etc/sysconfig/network文件，配置主机名

|  |  |
| --- | --- |
| IP | 主机名 |
| 192.168.4.244 | hdp-master |
| 192.168.4.228 | hdp-slave01 |
| 192.168.4.229 | hdp-slave02 |
| 192.168.4.230 | hdp-slave03 |
| 192.168.4.231 | hdp-slave04 |

每台服务器修改/etc/hosts文件，添加如下域名映射

|  |
| --- |
| 192**.**168**.**4**.**244 hdp-master  192**.**168**.**4**.**228 hdp-slave01  192**.**168**.**4**.**229 hdp-slave02  192**.**168**.**4**.**230 hdp-slave03  192**.**168**.**4**.**231 hdp-slave04 |

# 添加linux用户

使用root用户

每台服务器添加uhadoop用户

|  |
| --- |
| groupadd ghadoop  **useradd** uhadoop -g ghadoop -d **/**home**/**uhadoop  **passwd** uhadoop **(**uhadoop#123**)** |

# 配置yum源

使用root用户

每台服务器都需配置

## 安装wget

|  |
| --- |
| **yum** **-**y install wget |

## 增加网易源

进入/etc/yum.repos.d目录

|  |
| --- |
| **wget** http://mirrors.163.com/.help/CentOS6-Base-163.repo  **mv** CentOS-Base.repo CentOS-Base.repo\_backup  **yum** clean all  **yum** makecache |

# 配置ntp服务

使用root用户

hdp-master作为ntp服务器，其他服务器从hdp-master同步时间

## 安装ntp

每台服务器都需安装

|  |
| --- |
| **yum** **-**y install ntp |

## 配置ntp服务

hdp-master服务器上的/etc/ntp.conf配置文件中添加

|  |
| --- |
| restrict 192**.**168**.**4**.**0 mask 255**.**255**.**255**.**0 nomodify notrap |

启动服务并设置开机启动

|  |
| --- |
| **service** ntpd start  chkconfig ntpd on |

## 配置时间同步

hdp-master之外的其他服务器都需配置，/etc/ntp.conf配置文件中添加

|  |
| --- |
| server hdp-master prefer |

同步时间启动服务并设置开机启动

|  |
| --- |
| ntpdate hdp-master  **service** ntpd start  chkconfig ntpd on |

# 配置无密码登陆

## 安装相关命令

使用root用户

每台服务器都需安装，包含ssh-copy\_id、scp等命令

|  |
| --- |
| **yum** **-**y install openssh-clients |

## 配置

使用uhadoop用户

为uhadoop用户配置免密登陆

hdp-master服务器上面执行，一直回车到执行结束

|  |
| --- |
| ssh-keygen |

再给每台服务器配置无密码登陆

|  |
| --- |
| ssh-copy-id uhadoop**@**hdp-master  ssh-copy-id uhadoop**@**hdp-slave01  ssh-copy-id uhadoop**@**hdp-slave02  ssh-copy-id uhadoop**@**hdp-slave03  ssh-copy-id uhadoop**@**hdp-slave04 |

输入yes，回车，输入uhadoop登陆密码，回车

# 安装jdk

每台服务器都需安装

## 卸载openjdk

使用root用户

查看centos已安装的jdk

|  |
| --- |
| **rpm** –qa **|** **grep** gcj  **rpm** –qa **|** **grep** jdk |

删除查询到的openJDK相关的安装包

|  |
| --- |
| **rpm** -e –-nodeps **\*\*\*** |

## 安装oraclejdk

使用uhadoop用户

oracle官网下载安装包jdk-8u152-linux-x64.tar.gz

将安装包解压到指定目录

|  |
| --- |
| **tar** –zxvf jdk-8u152-linux-x64.tar.gz -C home**/**uhadoop**/**apps**/** |

修改home/uhadoop/.bash\_profile，添加如下配置

|  |
| --- |
| **export** JAVA\_HOME**=/**home**/**uhadoop**/**apps**/**jdk1.8.0\_152  **export** PATH**=$PATH:$JAVA\_HOME/**bin |

使配置生效，重新登陆或者执行命令

|  |
| --- |
| source home**/**uhadoop**/.**bash\_profile |

# 关闭selinux

使用root用户

每台服务器都需关闭

## 查看selinux状态

使用如下连个命令均可

|  |
| --- |
| sestatus **-**v #enabled开启，disabled关闭  getenforce #Enforcing开启，Permissive关闭 |

## 关闭selinux

永久关闭，重启后生效。修改/etc/selinux/config配置文件，设置

|  |
| --- |
| SELINUX**=**disabled |

临时关闭，重启后失效。使用如下命令

|  |
| --- |
| setenforce 0 |

# 关闭防火墙

使用root用户

每台服务器都需关闭

关闭防火墙并设置开机不启动

|  |
| --- |
| **service** iptables stop  chkconfig iptables off |

# 安装hadoop

使用uhadoop用户

每台服务器都需要安装

采用在hdp-master服务器上安装，scp同步到其他服务器的方式

## 设置环境变量

每台服务器修改home/uhadoop/.bash\_profile，添加如下配置

|  |
| --- |
| **export** HADOOP\_HOME**=/**home**/**uhadoop**/**apps**/**hadoop-2.7.5  **export** HADOOP\_COMMON\_LIB\_NATIVE\_DIR**=$HADOOP\_HOME/**lib**/**native  **export** HADOOP\_OPTS**=**"-Djava.library.path=$HADOOP\_HOME/lib:$HADOOP\_COMMON\_LIB\_NATIVE\_DIR"  **export** PATH**=$PATH:$HADOOP\_HOME/**bin**:$HADOOP\_HOME/**sbin |

## 创建目录

在每台服务器本地创建目录：

|  |
| --- |
| **mkdir** –p **/**home**/**uhadoop**/**datas**/**hadoop**/**dfs**/**data  **mkdir** –p **/**home**/**uhadoop**/**datas**/**hadoop**/**dfs**/**name  **mkdir** –p **/**home**/**uhadoop**/**datas**/**hadoop**/**tmp |

## 安装

apache官网下载安装包：hadoop-2.7.5.tar.gz

将安装包解压到指定目录

|  |
| --- |
| **tar** –zxvf hadoop-2.7.5.tar.gz -C home**/**uhadoop**/**apps**/** |

## 配置slaves

修改hadoop安装目录下/etc/hadoop/slaves文件

添加DataNode主机名列表，每个主机名一行

|  |
| --- |
| hdp-slave01  hdp-slave02  hdp-slave03  hdp-slave04 |

## 配置core-site.xml

修改hadoop安装目录下/etc/hadoop/core-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**fs.defaultFS**</name>  <value>**hdfs://hdp-master**</value>  </property>  <property>  <name>**hadoop.tmp.dir**</name>  <value>**file:/home/uhadoop/datas/hadoop/tmp**</value>  </property>  <property>  <name>**io.file.buffer.size**</name>  <value>**131702**</value>  </property>  </configuration> |

## 配置hdfs-site.xml

修改hadoop安装目录下/etc/hadoop/hdfs-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**dfs.namenode.name.dir**</name>  <value>**file:/home/uhadoop/datas/hadoop/dfs/name**</value>  </property>  <property>  <name>**dfs.datanode.data.dir**</name>  <value>**file:/home/uhadoop/datas/hadoop/dfs/data**</value>  </property>  <property>  <name>**dfs.replication**</name>  <value>**3**</value>  </property>  <property>  <name>**dfs.namenode.secondary.http-address**</name>  <value>**hdp-master:50090**</value>  </property>  <property>  <name>**dfs.webhdfs.enabled**</name>  <value>**true**</value>  </property>  <property>  <name>**dfs.balance.bandwidthPerSec**</name>  <value>**10485760**</value>  </property>  </configuration> |

## 配置mapred-site.xml

进入hadoop安装目录下/etc/hadoop目录

复制模版配置文件

|  |
| --- |
| **cp** mapred-site.xml.template mapred-site.xml |

修改mapred-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**mapreduce.framework.name**</name>  <value>**yarn**</value>  </property>  </configuration> |

## 配置yarn-site.xml

修改hadoop安装目录下/etc/hadoop/yarn-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**yarn.resourcemanager.hostname**</name>  <value>**hdp-master**</value>  </property>  <property>  <name>**yarn.nodemanager.aux-services**</name>  <value>**mapreduce\_shuffle**</value>  </property>  <property>  <name>**yarn.nodemanager.vmem-check-enabled**</name>  <value>**false**</value>  </property>  </configuration> |

## 配置hadoop-env.sh

修改hadoop安装目录下/etc/hadoop/hadoop-env.sh文件

修改如下配置项

|  |
| --- |
| **export** JAVA\_HOME**=/**home**/**uhadoop**/**apps**/**jdk1.8.0\_152 |

## 配置yarn-env.sh

修改hadoop安装目录下/etc/hadoop/yarn-env.sh文件

修改如下配置项

|  |
| --- |
| **export** JAVA\_HOME**=/**home**/**uhadoop**/**apps**/**jdk1.8.0\_152 |

## 同步文件

将hadoop-2.7.5整个目录复制到其他机器上，放在**/**home**/**uhadoop**/**apps/目录下

|  |
| --- |
| **cd** **/**home**/**uhadoop**/**apps**/**  scp –r hadoop-2.7.5 hdp-slave01**:/**home**/**uhadoop**/**apps**/**  scp –r hadoop-2.7.5 hdp-slave02**:/**home**/**uhadoop**/**apps**/**  scp –r hadoop-2.7.5 hdp-slave03**:/**home**/**uhadoop**/**apps**/**  scp –r hadoop-2.7.5 hdp-slave04**:/**home**/**uhadoop**/**apps**/** |

## 启动Hadoop

在hdp-master服务器上，进入hadoop安装目录

格式化NameNode

|  |
| --- |
| bin**/**hdfs namenode –format |

启动NameNode、DataNode与ResourceManager、NodeManager

|  |
| --- |
| sbin**/**start-dfs.sh  sbin**/**start-yarn.sh |

## 验证Hadoop

在hdp-master服务器上执行jps命令，有如下进程，说明hadoop启动正常

|  |
| --- |
| 59024 Jps  153505 ResourceManager  152951 NameNode  153247 SecondaryNameNode |

执行Hadoop自带的wordcount任务

在hdfs上创建测试目录

|  |
| --- |
| hdfs dfs **-**mkdir **/**test**/**wordcount**/**input |

上传一个英文文本文件到该目录

|  |
| --- |
| hdfs dfs **-**put inputwords **/**test**/**wordcount**/**input**/** |

进入hadoop安装路径的share/hadoop/mapreduce/目录下，执行任务，查看是否运行正常

|  |
| --- |
| hadoop jar hadoop-mapreduce-examples-2.7.5.jar wordcount **/**test**/**wordcount**/**input**/** **/**test**/**wordcount**/**output |

# 安装hive

使用uhadoop用户

在hdp-master服务器上安装

## 设置环境变量

在hdp-master服务器修改home/uhadoop/.bash\_profile，添加如下配置

|  |
| --- |
| **export** HIVE\_HOME**=**home**/**uhadoop**/**apps**/**apache-hive-2.2.0-bin  **export** PATH**=$HIVE\_HOME/**bin**:$PATH** |

## 创建目录

在hdp-master服务器创建本地目录

|  |
| --- |
| **mkdir** –p **/**home**/**uhadoop**/**datas**/**hive**/**log  **mkdir** –p **/**home**/**uhadoop**/**datas**/**hive**/**execlog  **mkdir** –p **/**home**/**uhadoop**/**datas**/**hive**/**tmp |

在hdfs 中创建目录

|  |
| --- |
| hdfs dfs **-**mkdir -p **/**user**/**hive**/**warehouse  hdfs dfs **-**mkdir -p **/**user**/**hive**/**tmp |

## 创建metastore库

在oracle数据库中执行

|  |
| --- |
| **create** tablespace hdpts  datafile '/data1/oracledbfiles/infadb/hdpts.dbf'  **size** 50m autoextend **on** **next** 50m maxsize 1024m  **create** **user** uhive **identified** **by** uhive  **default** tablespace hdpts  **temporary** tablespace temp |

## 安装

apache官网下载安装包：apache-hive-2.2.0-bin.tar.gz

将安装包解压到指定目录

|  |
| --- |
| **tar** –zxvf apache-hive-2.2.0-bin.tar.gz -C home**/**uhadoop**/**apps**/** |

删除hive安装目录下/lib/log4j-slf4j-impl-2.6.2.jar文件，否则会与hadoop安装目录下的log4j重复

在oracle官网下载jdbc驱动ojdbc6.jar，放到hive安装目录的/lib目录下

## 配置hive-env.sh

进入hive安装目录下/conf目录

复制模版配置文件

|  |
| --- |
| **cp** hive-env.sh.template hive-env.sh |

修改hive-env.sh文件以下配置项

|  |
| --- |
| **export** JAVA\_HOME**=/**home**/**uhadoop**/**apps**/**jdk1.8.0\_152  **export** HADOOP\_HOME**=/**home**/**uhadoop**/**apps**/**hadoop-2.7.5  **export** HIVE\_HOME**=/**home**/**uhadoop**/**apps**/**apache-hive-2.2.0-bin  **export** HIVE\_CONF\_DIR**=/**home**/**uhadoop**/**apps**/**apache-hive-2.2.0-bin**/**conf |

## 配置hive-site.xml

进入hive安装目录下/conf目录

复制模版配置文件

|  |
| --- |
| **cp** hive-default.xml.template hive-site.xml |

修改hive-site.xml文件以下配置项

|  |
| --- |
| <configuration>  <property>  <name>**hive.exec.scratchdir**</name>  <value>**/user/hive/tmp**</value>  </property>  <property>  <name>**hive.metastore.warehouse.dir**</name>  <value>**/user/hive/warehouse**</value>  </property>  <property>  <name>**hive.querylog.location**</name>  <value>**/home/uhadoop/datas/hive/tmp/${user.name}**</value>  </property>  <property>  <name>**javax.jdo.option.ConnectionURL**</name>  <value>**jdbc:oracle:thin:@192.168.4.209:1521:infadb**</value>  </property>  <property>  <name>**javax.jdo.option.ConnectionDriverName**</name>  <value>**oracle.jdbc.driver.OracleDriver**</value>  </property>  <property>  <name>**javax.jdo.option.ConnectionUserName**</name>  <value>**uhive**</value>  </property>  <property>  <name>**javax.jdo.option.ConnectionPassword**</name>  <value>**uhive**</value>  </property>  </configuration> |

将hive-site.xml中

所有${system:java.io.tmpdir}替换为/home/uhadoop/datas/hive/tmp/

所有{system:user.name}替换为{user.name}

## 配置hive-log4j2.properties

进入hive安装目录下/conf目录

复制模版配置文件

|  |
| --- |
| **cp** hive-log4j2.properties.template hive-log4j2.properties |

修改hive-log4j2.properties文件以下配置项

|  |
| --- |
| property.hive.log.dir **=** /home/uhadoop/datas/hive/log/${sys:user.name} |

## 配置hive-exec-log4j2.properties

进入hive安装目录下/conf目录

复制模版配置文件

|  |
| --- |
| **cp** hive-exec-log4j2.properties.template hive-exec-log4j2.properties |

修改hive-exec-log4j2.properties文件以下配置项

|  |
| --- |
| property.hive.log.dir **=** /home/uhadoop/datas/hive/execlog/${sys:user.name} |

## 配置core-site.xml

修改hadoop安装目录下/etc/hadoop/core-site.xml文件，给beeline登录用户开放权限

|  |
| --- |
| <property>  <name>**hadoop.proxyuser.uhadoop.groups**</name>  <value>**\***</value>  </property>  <property>  <name>**hadoop.proxyuser.uhadoop.hosts**</name>  <value>**\***</value>  </property> |

## 启动hive

使用如下命令初始化，自动生成hive数据库元数据信息

|  |
| --- |
| schematool **-**dbType oracle **-**initSchema |

启动hiveserver2，用于beeline等客户端连接hive

|  |
| --- |
| **nohup** hive **--**service hiveserver2 **>** **/**dev**/**null 2**>&**1 **&** |

查看hiveserver2启动状态

|  |
| --- |
| netstat **-**nptl **|** **grep** 10000 |

## 连接hive

本地连接，在命令行输入命令

|  |
| --- |
| hive |

beeline连接，在命令行输入命令

|  |
| --- |
| beeline |

进入beeline交互命令行，输入连接命令

|  |
| --- |
| **!**connect jdbc**:**hive2**://**hdp-master**:**10000 uhadoop uhadoop#123 |

# 安装tez

使用uhadoop用户

在hdp-master服务器上安装

## 安装

apache官网下载安装包：apache-tez-0.9.0-bin.tar.gz

将安装包解压到指定目录

|  |
| --- |
| **tar** –zxvf apache-tez-0.9.0-bin.tar.gz -C home**/**uhadoop**/**apps**/** |

删除tez安装目录下/lib/slf4j-log4j12-1.7.10.jar文件，否则会与hadoop安装目录下的log4j重复

删除tez安装目录下/lib/中以下两个文件

|  |
| --- |
| **rm** **-**rf hadoop-mapreduce-client-common-2.7.0.jar  **rm** **-**rf hadoop-mapreduce-client-core-2.7.0.jar |

将hadoop安装目录下/share/hadoop/mapreduce/中以下两个文件，复制到tez安装目录下/lib/目录中

|  |
| --- |
| **cp** hadoop-mapreduce-client-common-2.7.5.jar **~/**apps**/**apache-tez-0.9.0-bin**/**lib  **cp** hadoop-mapreduce-client-core-2.7.5.jar **~/**apps**/**apache-tez-0.9.0-bin**/**lib |

## 上传运行包

在tez 安装目录/share下有个tez.tar.gz压缩包，把它上传到hdfs上面

|  |
| --- |
| hadoop fs **-**mkdir -p **/**user**/**tez  hadoop fs **-**put tez.tar.gz **/**user**/**tez |

## 配置tez-site.xml

在hadoop安装目录/etc/hadoop/目录下新建tez-site.xml文件，添加配置

|  |
| --- |
| <configuration>  <property>  <name>**tez.lib.uris**</name>  <value>**/user/tez/tez.tar.gz**</value>  </property>  </configuration> |

## 配置mapred-site.xml

修改hadoop安装目录下/etc/hadoop/mapred-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**mapreduce.framework.name**</name>  <value>**yarn-tez**</value>  </property>  </configuration> |

## 配置hadoop-env.sh

修改hadoop安装目录下/etc/hadoop/hadoop-env.sh文件，增加配置

|  |
| --- |
| **export** TEZ\_HOME**=/**home**/**uhadoop**/**apps**/**apache-tez-0.9.0-bin  **for** jar **in** **`ls $TEZ\_HOME | grep jar`;** **do**  **export** HADOOP\_CLASSPATH**=$HADOOP\_CLASSPATH:$TEZ\_HOME/$jar**  **done**  **for** jar **in** **`ls $TEZ\_HOME/lib`;** **do**  **export** HADOOP\_CLASSPATH**=$HADOOP\_CLASSPATH:$TEZ\_HOME/**lib**/$jar**  **done** |

hive使用哪个引擎执行运算需要设置，如在hive客户端命令行设置

|  |
| --- |
| **set** hive.execution.engine**=**tez  **set** hive.execution.engine**=**mr |

# 安装kerberos

使用root用户

hdp-master服务器作为kdc服务器

## 安装

hdp-master服务器上安装

|  |
| --- |
| **yum** **-**y install krb5-libs krb5-workstation krb5-server |

hdp-master之外的其他服务器上安装

|  |
| --- |
| **yum** **-**y install krb5-devel krb5-workstation |

## 配置krb5.conf

修改hdp-master服务器上/etc/krb5.conf配置文件

|  |
| --- |
| **[logging]**  default **=** FILE:/var/log/krb5libs.log  kdc **=** FILE:/var/log/krb5kdc.log  admin\_server **=** FILE:/var/log/kadmind.log  **[libdefaults]**  default\_realm **=** HDP.HTJF  dns\_lookup\_realm **=** false  dns\_lookup\_kdc **=** false  ticket\_lifetime **=** 24h  renew\_lifetime **=** 7d  forwardable **=** true  **[realms]**  HDP.HTJF **=** {  kdc **=** hdp-master  admin\_server **=** hdp-master  }  **[domain\_realm]**  .hdp.htjf **=** HDP.HTJF  hdp.htjf **=** HDP.HTJF |

## 配置kdc.conf

修改hdp-master服务器上/var/kerberos/krb5kdc/kdc.conf配置文件

|  |
| --- |
| **[kdcdefaults]**  kdc\_ports **=** 88  kdc\_tcp\_ports **=** 88  **[realms]**  HDP.HTJF **=** {  acl\_file **=** /var/kerberos/krb5kdc/kadm5.acl  dict\_file **=** /usr/share/dict/words  admin\_keytab **=** /var/kerberos/krb5kdc/kadm5.keytab  max\_renewable\_life **=** 7d  supported\_enctypes **=** aes256-cts:normal aes128-cts:normal des3-hmac-sha1:normal arcfour-hmac:normal des-hmac-sha1:normal des-cbc-md5:normal des-cbc-crc:normal  } |

## 配置kadm5.acl

修改hdp-master服务器上/var/kerberos/krb5kdc/kadm5.acl配置文件

|  |
| --- |
| \*/admin@HDP.HTJF \* |

## 同步配置文件

将hdp-master服务器上/etc/krb5.conf配置文件复制到集群的其他服务器上

|  |
| --- |
| scp **/**etc**/**krb5.conf hdp-slave01**:/**etc**/**krb5.conf  scp **/**etc**/**krb5.conf hdp-slave02**:/**etc**/**krb5.conf  scp **/**etc**/**krb5.conf hdp-slave03**:/**etc**/**krb5.conf  scp **/**etc**/**krb5.conf hdp-slave04**:/**etc**/**krb5.conf |

## 创建数据库

在hdp-master上运行初始化数据库命令，会在 /var/kerberos/krb5kdc/ 目录下创建 principal 数据库

|  |
| --- |
| kdb5\_util create -r HDP.HTJF -s |

出现Loading random data的时候，另开个终端执行消耗cpu的命令加快随机数采集

|  |
| --- |
| **cat** **/**dev**/**sda **>** **/**dev**/**urandom |

设置KDC database密码为：kdcdb#123

## 启动服务

在hdp-master上启动kerberos相关服务器，并设置开机启动

|  |
| --- |
| **service** krb5kdc start  **service** kadmin start  chkconfig krb5kdc on  chkconfig kadmin on |

## 创建管理员

在hdp-master上执行kadmin.local命令，进入交互命令行

执行如下命令创建管理员用户，密码为：admin!123

|  |
| --- |
| addprinc admin**/**admin**@**HDP.HTJF |

执行如下命令查看用户

|  |
| --- |
| getprinc admin**/**admin**@**HDP.HTJF |

# hadoop添加kerberos验证

使用uhadoop用户

## 安装jce

使用centos5.6及以上的系统，默认使用AES-256来加密的，需要集群中的所有节点安装jce

oracle官网下载jdk对应版本的jce压缩包jce\_policy-8.zip

将压缩包解压，然后将需要的文件复制到jdk先关目录下

|  |
| --- |
| **unzip** jce\_policy-8.zip  **cd** UnlimitedJCEPolicyJDK8  **cp** local\_policy.jar **~/**apps**/**jdk1.8.0\_152**/**jre**/**lib**/**security**/**  **cp** US\_export\_policy.jar **~/**apps**/**jdk1.8.0\_152**/**jre**/**lib**/**security**/** |

## 创建本地目录

在所有节点上创建目录，用户存放keytab文件和证书

|  |
| --- |
| **mkdir** -p **~/**apps**/**security |

## 添加principal

使用root用户

在kdc服务器hdp-master上执行kadmin.local命令，进入交互命令行

执行如下命令为集群所有节点添加principal

|  |
| --- |
| addprinc **-**randkey uhadoop**/**hdp-master**@**HDP.HTJF  addprinc **-**randkey uhadoop**/**hdp-slave01**@**HDP.HTJF  addprinc **-**randkey uhadoop**/**hdp-slave02**@**HDP.HTJF  addprinc **-**randkey uhadoop**/**hdp-slave03**@**HDP.HTJF  addprinc **-**randkey uhadoop**/**hdp-slave04**@**HDP.HTJF  addprinc **-**randkey HTTP**/**hdp-master**@**HDP.HTJF  addprinc **-**randkey HTTP**/**hdp-slave01**@**HDP.HTJF  addprinc **-**randkey HTTP**/**hdp-slave02**@**HDP.HTJF  addprinc **-**randkey HTTP**/**hdp-slave03**@**HDP.HTJF  addprinc **-**randkey HTTP**/**hdp-slave04**@**HDP.HTJF  addprinc **-**randkey host**/**hdp-master**@**HDP.HTJF  addprinc **-**randkey host**/**hdp-slave01**@**HDP.HTJF  addprinc **-**randkey host**/**hdp-slave02**@**HDP.HTJF  addprinc **-**randkey host**/**hdp-slave03**@**HDP.HTJF  addprinc **-**randkey host**/**hdp-slave04**@**HDP.HTJF |

使用listprincs列出kdc数据库中所有principal名称

如果不需要某个principal，可以使用如下命令删除

|  |
| --- |
| delprinc **-**force uhadoop**/**hdp-master**@**HDP.HTJF |

执行如下命令将上面新增的principal添加到keytab文件

|  |
| --- |
| ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab uhadoop**/**hdp-master**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab uhadoop**/**hdp-slave01**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab uhadoop**/**hdp-slave02**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab uhadoop**/**hdp-slave03**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab uhadoop**/**hdp-slave04**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab HTTP**/**hdp-master**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab HTTP**/**hdp-slave01**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab HTTP**/**hdp-slave02**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab HTTP**/**hdp-slave03**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab HTTP**/**hdp-slave04**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab host**/**hdp-master**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab host**/**hdp-slave01**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab host**/**hdp-slave02**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab host**/**hdp-slave03**@**HDP.HTJF  ktadd **-**norandkey -k **/**home**/**uhadoop**/**apps**/**security**/**hadoop.keytab host**/**hdp-slave04**@**HDP.HTJF |

退出交互命令行，将hadoop.keytab的owner修改为uhadoop

|  |
| --- |
| **chown** uhadoop.ghadoop hadoop.keytab |

使用如下命令查看keytab文件内包含的principal

|  |
| --- |
| klist -k hadoop.keytab |

## 同步keytab文件

将hadoop.keytab文件同步到集群中其他节点

|  |
| --- |
| scp hadoop.keytab hdp-slave01**:~/**apps**/**security**/**  scp hadoop.keytab hdp-slave02**:~/**apps**/**security**/**  scp hadoop.keytab hdp-slave03**:~/**apps**/**security**/**  scp hadoop.keytab hdp-slave04**:~/**apps**/**security**/** |

修改hadoop.keytab文件的权限

|  |
| --- |
| **chmod** 400 hadoop.keytab |

在集群中每个节点上初始化其对应的principal

|  |
| --- |
| kinit **-**kt hadoop.keytab uhadoop**/**hdp-master**@**HDP.HTJF  kinit **-**kt hadoop.keytab uhadoop**/**hdp-slave01**@**HDP.HTJF  kinit **-**kt hadoop.keytab uhadoop**/**hdp-slave02**@**HDP.HTJF  kinit **-**kt hadoop.keytab uhadoop**/**hdp-slave03**@**HDP.HTJF  kinit **-**kt hadoop.keytab uhadoop**/**hdp-slave04**@**HDP.HTJF |

可以使用klist查看、kdestroy清除当前获取的principal

## 配置sasl证书

在hdp-master上创建根证书，用于给节点证书签名

进入~/apps/security/目录

|  |
| --- |
| **openssl** req **-**new **-**x509 **-**keyout hdfs\_ca\_key **-**out hdfs\_ca\_cert **-**days 9999 **-**subj '/C=CN/ST=gd/L=gz/O=htjf/OU=cy/CN=hdp-master' |

输入密码ssl123456

将生成的文件同步到集群中其他节点

|  |
| --- |
| scp hdfs\_ca**\*** hdp-slave01**:~/**apps**/**security  scp hdfs\_ca**\*** hdp-slave02**:~/**apps**/**security  scp hdfs\_ca**\*** hdp-slave03**:~/**apps**/**security  scp hdfs\_ca**\*** hdp-slave04**:~/**apps**/**security |

在每个节点服务器上面执行如下的操作，生成证书及信任库

|  |
| --- |
| keytool **-**keystore keystore **-**alias localhost **-**validity 9999 **-**keysize 2048 **-**dname "CN=hdp-master,OU=cy,O=htjf,L=gz,ST=gd,C=CN" **-**genkey **-**keyalg RSA  keytool **-**keystore truststore **-**alias CARoot **-**import **-**file hdfs\_ca\_cert  keytool **-**certreq **-**alias localhost **-**keystore keystore **-**file cert  **openssl** **-**req x509 **-**CA hdfs\_ca\_cert **-**CAkey hdfs\_ca\_key **-**in cert **-**out cert\_signed **-**days 9999 **-**CAcreateserial  keytool **-**keystore keystore **-**alias CARoot **-**import **-**file hdfs\_ca\_cert  keytool **-**keystore keystore **-**alias localhost **-**import **-**file cert\_signed |

其中-dname参数中的CN需要修改为每个节点的主机名，过程中需要填写密码的地方都为ssl123456，所有是否判断的地方都填“是”

重命名证书，删除过程文件

|  |
| --- |
| **mv** keystore keystore.jks  **mv** truststore truststore.jks  **rm** **-**rf cert**\*** hdfs\_ca\_**\*** |

## 配置ssl-client.xml

进入hadoop安装目录下/etc/hadoop目录

复制模版配置文件

|  |
| --- |
| **cp** ssl-client.xml.example ssl-client.xml |

修改ssl-client.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**ssl.client.truststore.location**</name>  <value>**/home/uhadoop/apps/security/truststore.jks**</value>  </property>  <property>  <name>**ssl.client.truststore.password**</name>  <value>**ssl123456**</value>  </property>  <property>  <name>**ssl.client.keystore.location**</name>  <value>**/home/uhadoop/apps/security/keystore.jks**</value>  </property>  <property>  <name>**ssl.client.keystore.password**</name>  <value>**ssl123456**</value>  </property>  <property>  <name>**ssl.client.keystore.keypassword**</name>  <value>**ssl123456**</value>  </property>  </configuration> |

## 配置ssl-server.xml

进入hadoop安装目录下/etc/hadoop目录

复制模版配置文件

|  |
| --- |
| **cp** ssl-server.xml.template ssl-server.xml |

修改ssl-server.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**ssl.server.truststore.location**</name>  <value>**/home/uhadoop/apps/security/truststore.jks**</value>  </property>  <property>  <name>**ssl.server.truststore.password**</name>  <value>**ssl123456**</value>  </property>  <property>  <name>**ssl.server.keystore.location**</name>  <value>**/home/uhadoop/apps/security/keystore.jks**</value>  </property>  <property>  <name>**ssl.server.keystore.password**</name>  <value>**ssl123456**</value>  </property>  <property>  <name>**ssl.server.keystore.keypassword**</name>  <value>**ssl123456**</value>  </property>  </configuration> |

## 配置core-site.xml

修改hadoop安装目录下/etc/hadoop/core-site.xml文件，添加如下配置

|  |
| --- |
| <configuration>  <property>  <name>**hadoop.security.authorization**</name>  <value>**true**</value>  </property>  <property>  <name>**hadoop.security.authentication**</name>  <value>**kerberos**</value>  </property>  </configuration> |

## 配置hdfs-site.xml

修改hadoop安装目录下/etc/hadoop/hdfs-site.xml文件，添加如下配置

|  |
| --- |
| <configuration>  <property>  <name>**dfs.block.access.token.enable**</name>  <value>**true**</value>  </property>  <property>  <name>**dfs.namenode.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**dfs.namenode.keytab.file**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**dfs.datanode.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**dfs.datanode.keytab.file**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**dfs.datanode.data.dir.perm**</name>  <value>**700**</value>  </property>  <property>  <name>**dfs.secondary.namenode.keytab.file**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**dfs.secondary.namenode.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**dfs.web.authentication.kerberos.principal**</name>  <value>**HTTP/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**dfs.web.authentication.kerberos.keytab**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**dfs.http.policy**</name>  <value>**HTTPS\_ONLY**</value>  </property>  <property>  <name>**dfs.data.transfer.protection**</name>  <value>**integrity**</value>  </property>  </configuration> |

## 配置mapred-site.xml

修改hadoop安装目录下/etc/hadoop/mapred-site.xml文件，添加如下配置

|  |
| --- |
| <configuration>  <property>  <name>**mapreduce.jobtracker.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**mapreduce.jobtracker.keytab.file**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**mapreduce.tasktracker.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**mapreduce.tasktracker.keytab.file**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  </configuration> |

## 配置yarn-site.xml

修改hadoop安装目录下/etc/hadoop/yarn-site.xml文件，添加如下配置

|  |
| --- |
| <configuration>  <property>  <name>**yarn.resourcemanager.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**yarn.resourcemanager.keytab**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**yarn.nodemanager.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**yarn.nodemanager.keytab**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  </configuration> |

## 同步配置文件

将以下配置文件同步到集群中其他节点

|  |
| --- |
| core-site.xml  hdfs-site.xml  mapred-site.xml  yarn-site.xml  ssl-client.xml  ssl-server.xml |

# hive添加kerberos验证

使用uhadoop用户

在hdp-master服务器上配置

## 配置hive-site.xml

修改hive安装目录下/conf/hive-site.xml文件以下配置项

|  |
| --- |
| <configuration>  <property>  <name>**hive.server2.authentication**</name>  <value>**KERBEROS**</value>  </property>  <property>  <name>**hive.server2.authentication.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  <property>  <name>**hive.server2.authentication.kerberos.keytab**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**hive.metastore.sasl.enabled**</name>  <value>**true**</value>  </property>  <property>  <name>**hive.metastore.kerberos.keytab.file**</name>  <value>**/home/uhadoop/apps/security/hadoop.keytab**</value>  </property>  <property>  <name>**hive.metastore.kerberos.principal**</name>  <value>**uhadoop/\_HOST@HDP.HTJF**</value>  </property>  </configuration> |

# 客户端beeline连接hive

使用普通用户

## 添加principal

使用root用户

在kdc服务器hdp-master上执行kadmin.local命令，进入交互命令行

执行如下命令为集群所有节点添加principal

|  |
| --- |
| addprinc **-**randkey hutest**/**learn-ubuntu**@**HDP.HTJF |

执行如下命令将上面新增的principal添加到keytab文件

|  |
| --- |
| ktadd **-**norandkey -k **~/**hutest.keytab hutest**/**learn-ubuntu**@**HDP.HTJF |

## 初始化principal

将hutest.keytab文件上传到客户机上，使用如下命令初始化principal

|  |
| --- |
| kinit **-**kt hutest.keytab hutest**/**learn-ubuntu**@**HDP.HTJF |

## 安装jce

参照13.1节

## 修改目录权限

修改hdfs上的/tmp目录的权限

|  |
| --- |
| hdfs dfs **-**chmod -R 777 **/**tmp |

## 安装hadoop

将安装包hadoop-2.7.5.tar.gz解压到指定目录

|  |
| --- |
| **tar** –zxvf hadoop-2.7.5.tar.gz -C **/**home**/**ljwork**/**apps**/** |

设置环境变量，添加如下配置

|  |
| --- |
| **export** HADOOP\_HOME**=/**home**/**ljwork**/**apps**/**hadoop-2.7.5 |

配置hadoop安装目录下/etc/hadoop/core-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**fs.defaultFS**</name>  <value>**hdfs://hdp-master**</value>  </property>  <property>  <name>**hadoop.security.authorization**</name>  <value>**true**</value>  </property>  <property>  <name>**hadoop.security.authentication**</name>  <value>**kerberos**</value>  </property>  </configuration> |

## 安装hive

将安装包apache-hive-2.2.0-bin.tar.gz解压到指定目录

|  |
| --- |
| **tar** –zxvf apache-hive-2.2.0-bin.tar.gz -C home**/**ljwork**/**apps**/** |

## 使用beeline连接

进入hive安装路径/bin/目录，执行如下命令进入交互命令行

|  |
| --- |
| **./**beeline |

执行如下命令连接hive

|  |
| --- |
| **!**connect jdbc**:**hive2**://**hdp-master**:**10000**/;**principal**=**uhadoop**/**hdp-master**@**HDP.HTJF |

# hdfs权限控制

使用uhadoop用户

## hdfs访问权限

基本权限使用chown/chmod命令进行配置，如

|  |
| --- |
| hdfs dfs **-**chown hutest **/**hu**/**hutest  hdfs dfs **-**chmod -R 700 **/**hu**/**hutest |

acl（access control list）权限配置

修改hadoop安装目录下/etc/hadoop/hdfs-site.xml文件，添加如下配置用于支持acl权限控制

|  |
| --- |
| <configuration>  <property>  <name>**dfs.permissions.enabled**</name>  <value>**true**</value>  </property>  <property>  <name>**dfs.namenode.acls.enabled**</name>  <value>**true**</value>  </property>  </configuration> |

将hdfs-site.xml同步到集群中其他节点上

使用setfacl设置权限

|  |
| --- |
| hdfs dfs **-**setfacl -R **-**m user**:**hutest**:**rw- **/**hu**/**hutest  hdfs dfs **-**setfacl -x user**:**hutest **/**hu**/**hutest |

使用getfacl查看权限

|  |
| --- |
| hdfs dfs **-**getfacl **/**hu**/**hutest  hdfs dfs **-**getfacl -R **/**hu**/**hutest |

## 用户存储配额

通过配置目录的配额，实现对用户存储空间的控制

设置配额，如设置/hu/hutest目录的配额为100MB

|  |
| --- |
| hdfs dfsadmin **-**setquota |

如果设置的配额大小不是正的长整型数、目录不存在或者是文件、或者目录已有内容超过配额，则会产生错误报告

清除配额，如清除目录/hu/hutest上设置的配额

|  |
| --- |
| hdfs dfsadmin **-**clrquota **/**hu**/**hutest |

如果目录不存在或者是文件，则会产生错误报告

如果目录原来没有设置配额不会报错

查看配额，如查看目录/hu/hutest上设置的配额

|  |
| --- |
| hdfs dfs **-**count **-**q **/**hu**/**hutest |

报告每个目录设置的配额，以及剩余配额

如果目录没有设置配额，会报告none和inf

## 用户计算资源配额

使用[Capacity Scheduler](https://han-zw.iteye.com/blog/2315893)配置用户计算资源配额

修改hadoop安装目录下/etc/hadoop/yarn-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**yarn.resourcemanager.scheduler.class**</name>  <value>**org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler**</value>  </property>  </configuration> |

修改hadoop安装目录下/etc/hadoop/capacity-scheduler.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**yarn.scheduler.capacity.root.queues**</name>  <value>**default,outward**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.default.acl\_administer\_jobs**</name>  <value>**uhadoop**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.default.acl\_submit\_applications**</name>  <value>**uhadoop**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.default.capacity**</name>  <value>**50**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.default.maximum-capacity**</name>  <value>**100**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.default.user-limit-factor**</name>  <value>**8**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.default.state**</name>  <value>**RUNNING**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.outward.acl\_administer\_jobs**</name>  <value>**uhadoop**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.outward.acl\_submit\_applications**</name>  <value>**\***</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.outward.capacity**</name>  <value>**50**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.outward.maximum-capacity**</name>  <value>**100**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.outward.user-limit-factor**</name>  <value>**8**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.outward.state**</name>  <value>**RUNNING**</value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.acl\_submit\_applications**</name>  <value></value>  </property>  <property>  <name>**yarn.scheduler.capacity.root.acl\_administer\_queue**</name>  <value></value>  </property>  </configuration> |

将yarn-site.xml、capacity-scheduler.xml文件同步到集群其他节点上

# hive权限控制

使用uhadoop用户

## hive用户操作权限

修改hive安装目录下/conf/hive-site.xml文件

|  |
| --- |
| <configuration>  <property>  <name>**hive.security.authorization.enabled**</name>  <value>**true**</value>  </property>  <property>  <name>**hive.security.authorization.createtable.owner.grants**</name>  <value>**ALL**</value>  </property>  </configuration> |

可使用基于命令行的权限管理功能，对hive用户的权限进行管理，如

|  |
| --- |
| **grant** **select** **on** **table** table\_name **to** **user** hutest  **grant** **select** **on** database database\_name **to** **user** hutest |

具体参考hive官网文档

|  |
| --- |
| https://cwiki.apache.org/confluence/display/Hive/SQL+Standard+Based+Hive+Authorization#SQLStandardBasedHiveAuthorization-RestrictionsonHiveCommandsandStatements |

hive客户端执行任务时，需要指定hadoop的执行队列

|  |
| --- |
| **set** mapreduce.job.queuename**=**outward |

## hive存储权限

在hdfs配置hive连接用户访问文件的权限

参照16.1