尚硅谷大数据项目之尚品汇（安全环境实战）

(作者：尚硅谷研究院)

版本：V4.0

# 第1章 概述

Hadoop启用Kerberos安全认证之后，之前的非安全环境下的全流程调度脚本和即席查询引擎均会遇到认证问题，故需要对其进行改进。

# 第2章 数仓全流程

## 2.1 改动说明

此处统一将数仓的全部数据资源的所有者设为hive用户，全流程的每步操作均认证为hive用户。

## 2.2 改动实操

### 2.2.1 用户准备

**1.在各节点创建hive用户，如已存在则跳过**

[root@hadoop102 ~]# useradd hive -g hadoop

[root@hadoop102 ~]# echo hive | passwd --stdin hive

[root@hadoop103 ~]# useradd hive -g hadoop

[root@hadoop103 ~]# echo hive | passwd --stdin hive

[root@hadoop104 ~]# useradd hive -g hadoop

[root@hadoop104 ~]# echo hive | passwd --stdin hive

**2.为hive用户创建Keberos主体**

1）创建主体

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"addprinc -randkey hive"

2）生成keytab文件

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"xst -k /etc/security/keytab/hive.keytab hive"

3）修改keytab文件所有者和访问权限

[root@hadoop102 ~]# chown hive:hadoop /etc/security/keytab/hive.keytab

[root@hadoop102 ~]# chmod 440 /etc/security/keytab/hive.keytab

4）分发keytab文件

[root@hadoop102 ~]# xsync /etc/security/keytab/hive.keytab

### 2.2.2 数据采集通道修改

**1.用户行为日志**

修改/opt/module/flume/conf/kafka-flume-hdfs.conf配置文件，增加以下参数

[root@hadoop104 ~]# vim /opt/module/flume/conf/kafka-flume-hdfs.conf

a1.sinks.k1.hdfs.kerberosPrincipal=hive@EXAMPLE.COM

a1.sinks.k1.hdfs.kerberosKeytab=/etc/security/keytab/hive.keytab

**2.业务数据**

修改sqoop每日同步脚本/home/atguigu/bin/mysql\_to\_hdfs.sh，

[root@hadoop102 ~]# vim /home/atguigu/bin/mysql\_to\_hdfs.sh

在顶部增加如下认证语句

kinit -kt /etc/security/keytab/hive.keytab hive

### 2.2.3 数仓各层脚本修改

数仓各层脚本均需在顶部加入如下认证语句

kinit -kt /etc/security/keytab/hive.keytab hive

修改语句如下

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' hdfs\_to\_ods\_log.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' hdfs\_to\_ods\_db.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' ods\_to\_dwd\_log.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' ods\_to\_dim\_db.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' ods\_to\_dwd\_db.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' dwd\_to\_dws.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' dws\_to\_dwt.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' dwt\_to\_ads.sh

[root@hadoop102 ~]# sed -i '1 a kinit -kt /etc/security/keytab/hive.keytab hive' hdfs\_to\_mysql.sh

注：

sed -i '1 a text' file

表示将text内容加入到file文件的第1行之后

### 2.2.4 修改HDFS特定路径所有者

1.认证为hdfs用户，执行以下命令并按提示输入密码

[root@hadoop102 ~]# kinit hdfs/hadoop

2.修改数据采集目标路径

[root@hadoop102 ~]# hadoop fs -chown -R hive:hadoop /origin\_data

3.修改数仓表所在路径

[root@hadoop102 ~]# hadoop fs -chown -R hive:hadoop /warehouse

4.修改hive家目录/user/hive

[root@hadoop102 ~]# hadoop fs -chown -R hive:hadoop /user/hive

5.修改spark.eventLog.dir路径

[root@hadoop102 ~]# hadoop fs -chown -R hive:hadoop /spark-history

### 2.2.5 全流程数据准备

**1.用户行为日志**

1）启动日志采集通道，包括Zookeeper，Kafka，Flume等

2）修改hadoop102，hadoop103两台节点的/opt/module/applog/application.yml文件，将模拟日期改为2020-06-16如下

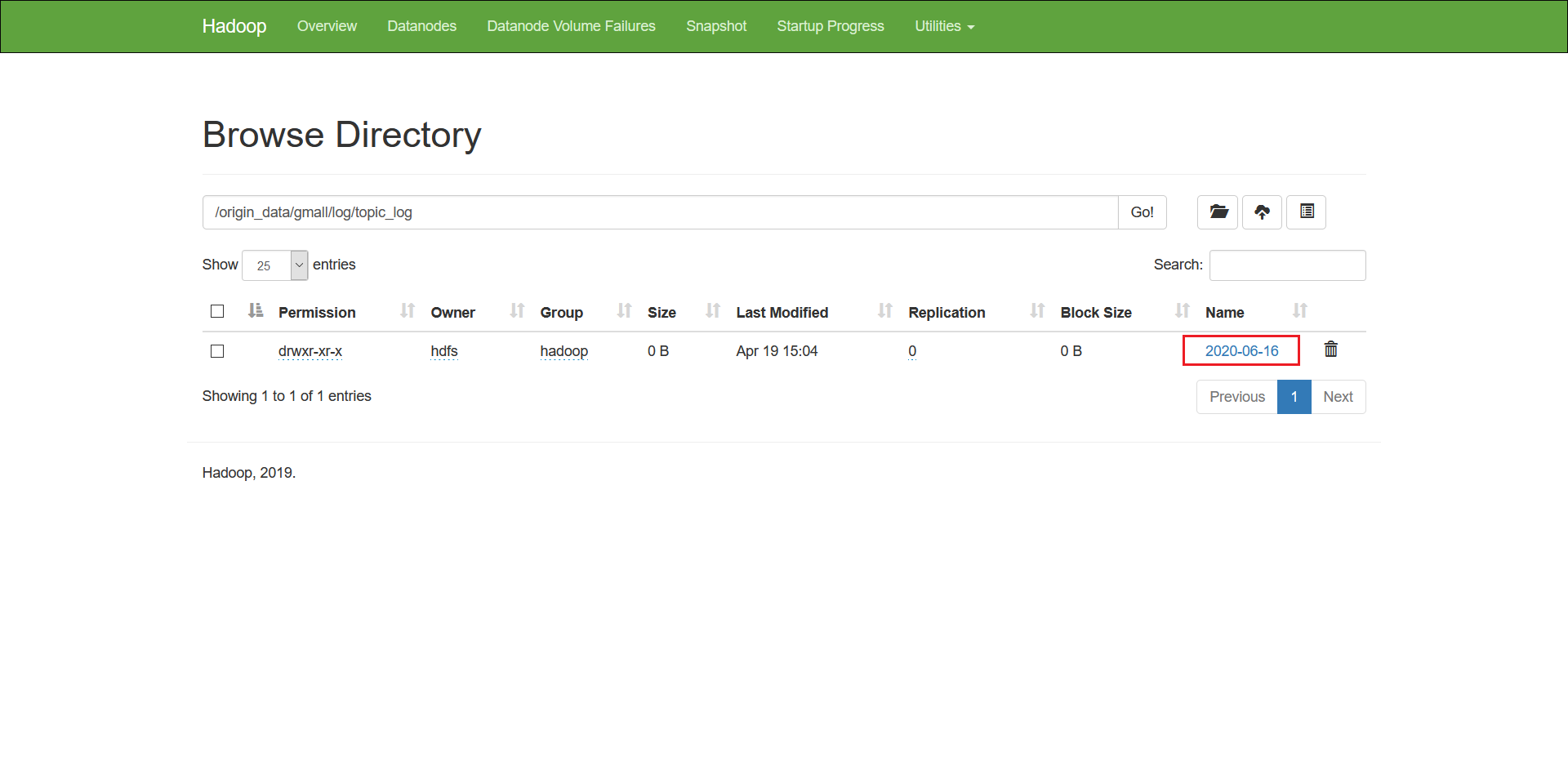
#业务日期

mock.date: "2020-06-16"

3）执行生成日志的脚本

[root@hadoop102 ~]# lg.sh

4）等待片刻，观察HDFS是否出现2020-06-16的日志文件



**2.业务数据**

1）修改/opt/module/db\_log/application.properties，将模拟日期修改为2020-06-16，如下

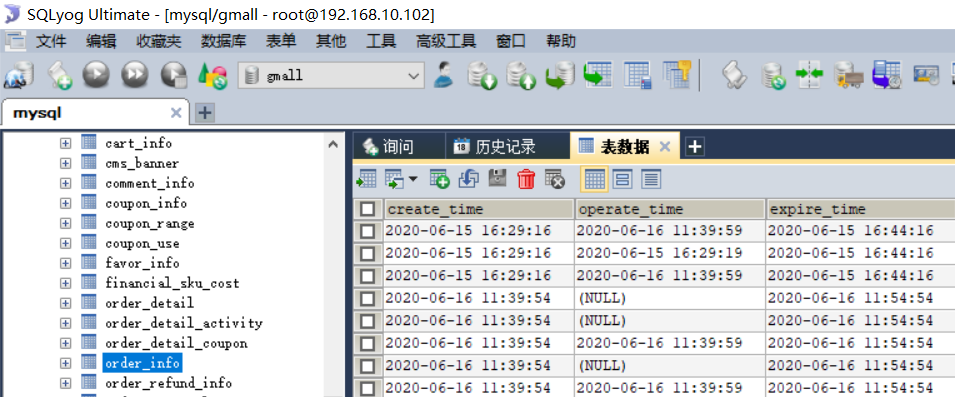
#业务日期

mock.date=2020-06-16

2）进入到/opt/module/db\_log路径，执行模拟生成业务数据的命令，如下

[root@hadoop102 ~]# java -jar gmall2020-mock-db-2021-01-22.jar

3）观察mysql的gmall数据中是否出现2020-06-16的数据



### 2.2.6 启动Azkaban

1.在各节点创建azkaban用户

[root@hadoop102 ~]# useradd azkaban -g hadoop

[root@hadoop102 ~]# echo azkaban | passwd --stdin azkaban

[root@hadoop103 ~]# useradd azkaban -g hadoop

[root@hadoop103 ~]# echo azkaban | passwd --stdin azkaban

[root@hadoop104 ~]# useradd azkaban -g hadoop

[root@hadoop104 ~]# echo azkaban | passwd --stdin azkaban

2.将各节点Azkaban安装路径所有者改为azkaban用户

[root@hadoop102 ~]# chown -R azkaban:hadoop /opt/module/azkaban

[root@hadoop103 ~]# chown -R azkaban:hadoop /opt/module/azkaban

[root@hadoop104 ~]# chown -R azkaban:hadoop /opt/module/azkaban

3.使用azkaban用户启动Azkaban

1）启动Executor Server

在各节点执行以下命令，启动Executor

[root@hadoop102 ~]# sudo -i -u azkaban bash -c "cd /opt/module/azkaban/azkaban-exec;bin/start-exec.sh"

[root@hadoop103 ~]# sudo -i -u azkaban bash -c "cd /opt/module/azkaban/azkaban-exec;bin/start-exec.sh"

[root@hadoop104 ~]# sudo -i -u azkaban bash -c "cd /opt/module/azkaban/azkaban-exec;bin/start-exec.sh"

2）激活Executor Server，任选一台节点执行以下激活命令即可

[root@hadoop102 ~]# curl http://hadoop102:12321/executor?action=activate

[root@hadoop102 ~]# curl http://hadoop103:12321/executor?action=activate

[root@hadoop102 ~]# curl http://hadoop104:12321/executor?action=activate

3）启动Web Server

[root@hadoop102 ~]# sudo -i -u azkaban bash -c "cd /opt/module/azkaban/azkaban-web;bin/start-web.sh"

4.修改数仓各层脚本访问权限，确保azkaban用户能够访问到

[root@hadoop102 ~]# chown -R atguigu:hadoop /home/atguigu

[root@hadoop102 ~]# chmod 770 /home/atguigu

[root@hadoop103 ~]# chown -R atguigu:hadoop /home/atguigu

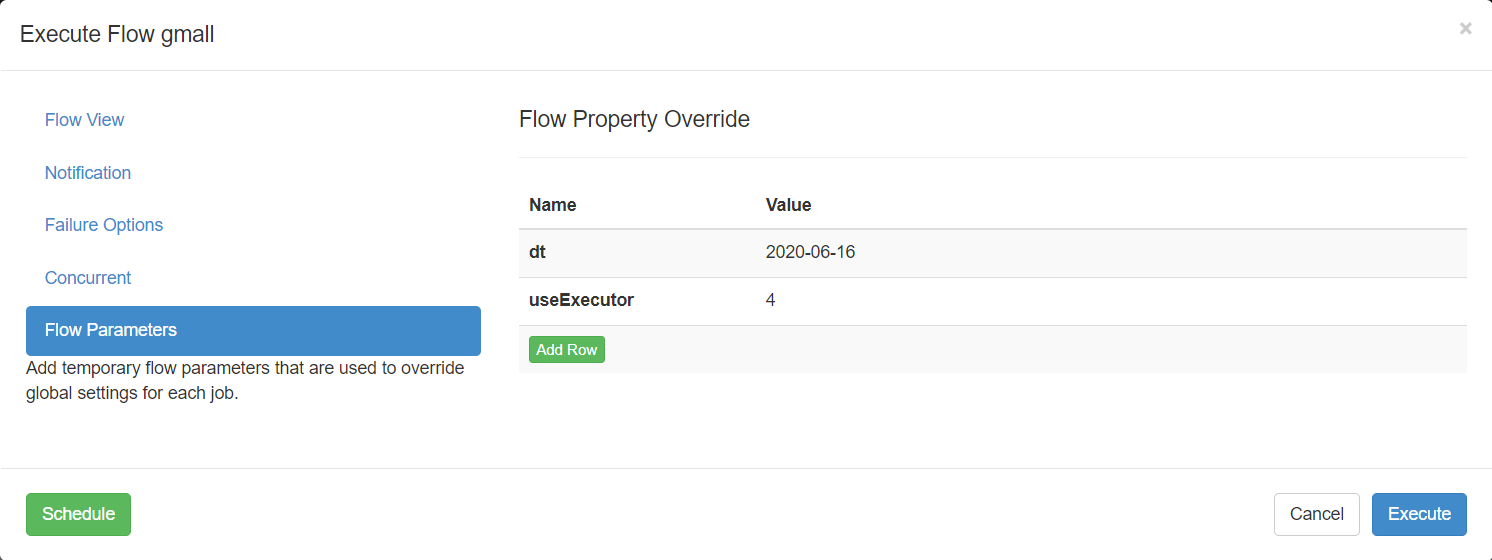
[root@hadoop103 ~]# chmod 770 /home/atguigu

[root@hadoop104 ~]# chown -R atguigu:hadoop /home/atguigu

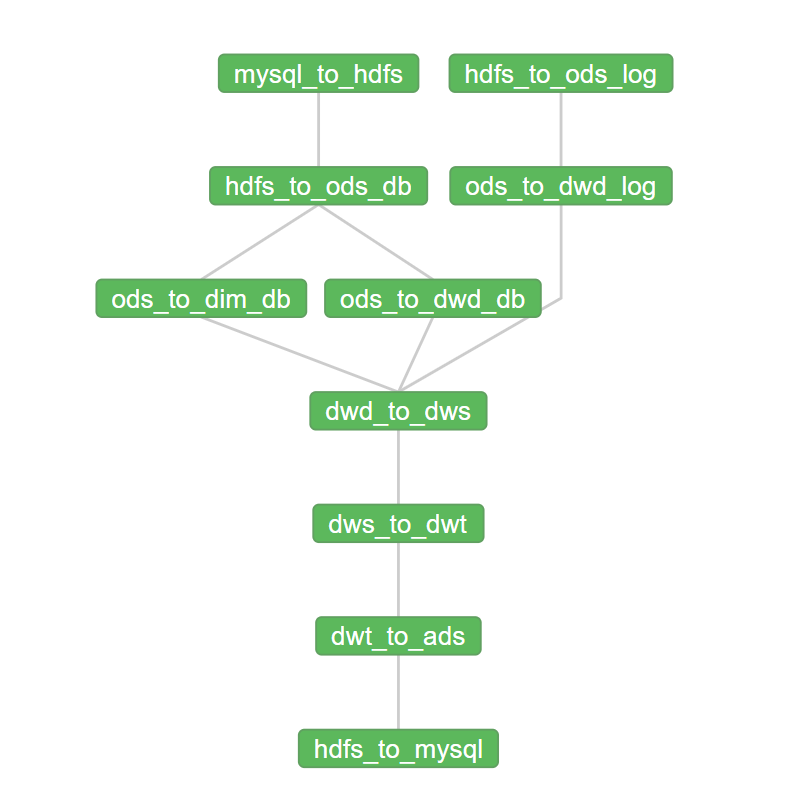
[root@hadoop104 ~]# chmod 770 /home/atguigu

### 2.2.7 全流程调度

1）工作流参数



2）运行结果



# 第3章 即席查询之Presto

## 3.1 改动说明

Presto集群开启Kerberos认证可只配置Presto Coordinator和Presto Cli之间进行认证，集群内部通讯可不进行认证。Presto Coordinator和Presto Cli之间的认证要求两者采用更为安全的HTTPS协议进行通讯。

若Presto对接的是Hive数据源，由于其需要访问Hive的元数据和HDFS上的数据文件，故也需要对Hive Connector进行Kerberos认证。

## 3.2 改动实操

### 3.2.1 用户准备

**1.在所有节点创建presto系统用户**

[root@hadoop102 ~]# useradd presto -g hadoop

[root@hadoop102 ~]# echo presto | passwd --stdin presto

[root@hadoop103 ~]# useradd presto -g hadoop

[root@hadoop103 ~]# echo presto | passwd --stdin presto

[root@hadoop104 ~]# useradd presto -g hadoop

[root@hadoop104 ~]# echo presto | passwd --stdin presto

**2.为Hive Connector创建Kerberos主体**

1）创建presto用户的Kerberos主体

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"addprinc -randkey presto"

2）生成keytab文件

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"xst -k /etc/security/keytab/presto.keytab presto"

3）修改keytab文件的访问权限

[root@hadoop102 ~]# chown presto:hadoop /etc/security/keytab/presto.keytab

4）分发keytab文件

[root@hadoop102 ~]# xsync /etc/security/keytab/presto.keytab

**3.为Presto Coordinator创建Kerberos主体**

1）创建presto用户的Kerberos主体

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"addprinc -randkey presto/hadoop102"

2）生成keytab文件

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"xst -k /etc/security/keytab/presto.service.keytab presto/hadoop102"

3）修改keytab文件的访问权限

[root@hadoop102 ~]# chown presto:hadoop /etc/security/keytab/presto.service.keytab

### 3.2.2 创建HTTPS协议所需的密钥对

**注意**：

（1）alias（别名）需要和Presto Coordinator的Kerberos主体名保持一致

（2）名字与姓氏 需要填写Coordinator所在的主机名

1）使用Java提供的keytool工具生成密钥对

[root@hadoop102 ~]# keytool -genkeypair -alias **presto** -keyalg RSA -keystore /etc/security/keytab/keystore.jks

输入密钥库口令:

再次输入新口令:

您的名字与姓氏是什么?

[Unknown]: **hadoop102**

您的组织单位名称是什么?

[Unknown]:

您的组织名称是什么?

[Unknown]:

您所在的城市或区域名称是什么?

[Unknown]:

您所在的省/市/自治区名称是什么?

[Unknown]:

该单位的双字母国家/地区代码是什么?

[Unknown]:

CN=hadoop102, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown是否正确?

[否]: y

输入 <presto> 的密钥口令

(如果和密钥库口令相同, 按回车):

2）修改keystore文件的所有者和访问权限

[root@hadoop102 ~]# chown presto:hadoop /etc/security/keytab/keystore.jks

[root@hadoop102 ~]# chmod 660 /etc/security/keytab/keystore.jks

### 3.2.3 修改Presto Coordinator配置文件

在/opt/module/presto/etc/config.properties文件中**增加**以下参数

[root@hadoop102 ~]# vim /opt/module/presto/etc/config.properties

http-server.authentication.type=KERBEROS

http.server.authentication.krb5.service-name=presto

http.server.authentication.krb5.keytab=/etc/security/keytab/presto.service.keytab

http.authentication.krb5.config=/etc/krb5.conf

http-server.https.enabled=true

http-server.https.port=7778

http-server.https.keystore.path=/etc/security/keytab/keystore.jks

http-server.https.keystore.key=123456

### 3.2.4 修改Hive Connector配置文件

1.在/opt/module/presto/etc/catalog/hive.properties中增加以下参数

[root@hadoop102 ~]# vim /opt/module/presto/etc/catalog/hive.properties

hive.metastore.authentication.type=KERBEROS

hive.metastore.service.principal=hive/hadoop102@EXAMPLE.COM

hive.metastore.client.principal=presto@EXAMPLE.COM

hive.metastore.client.keytab=/etc/security/keytab/presto.keytab

hive.hdfs.authentication.type=KERBEROS

hive.hdfs.impersonation.enabled=true

hive.hdfs.presto.principal=presto@EXAMPLE.COM

hive.hdfs.presto.keytab=/etc/security/keytab/presto.keytab

hive.config.resources=/opt/module/hadoop-3.1.3/etc/hadoop/core-site.xml,/opt/module/hadoop-3.1.3/etc/hadoop/hdfs-site.xml

2.分发/opt/module/presto/etc/catalog/hive.properties文件

[root@hadoop102 ~]# xsync /opt/module/presto/etc/catalog/hive.properties

### 3.2.5 配置客户端Kerberos主体到用户名之间的映射规则

1.新建/opt/module/presto/etc/access-control.properties配置文件，内容如下

[root@hadoop102 ~]# vim /opt/module/presto/etc/access-control.properties

access-control.name=file

security.config-file=etc/rules.json

2. 新建/opt/module/presto/etc/rules.json文件，内容如下

[root@hadoop102 ~]# vim /opt/module/presto/etc/rules.json

{

"catalogs": [

{

"allow": true

}

],

"user\_patterns": [

"(.\*)",

"([a-zA-Z]+)/?.\*@.\*"

]

}

### 3.2.6 配置Presto代理用户

**1.修改Hadoop配置文件**

修改$HADOOP\_HOME/etc/hadoop/core-site.xml配置文件，增加如下内容

[root@hadoop102 ~]# vim $HADOOP\_HOME/etc/hadoop/core-site.xml

<property>

<name>hadoop.proxyuser.presto.hosts</name>

<value>\*</value>

</property>

<property>

<name>hadoop.proxyuser.presto.groups</name>

<value>\*</value>

</property>

<property>

<name>hadoop.proxyuser.presto.users</name>

<value>\*</value>

</property>

**2.分发修改的文件**

[root@hadoop102 ~]# xsync $HADOOP\_HOME/etc/hadoop/core-site.xml

**3.重启Hadoop集群**

[root@hadoop102 ~]# stop-dfs.sh

[root@hadoop103 ~]# stop-yarn.sh

[root@hadoop102 ~]# start-dfs.sh

[root@hadoop103 ~]# start-yarn.sh

### 3.2.7 重启Presto集群

**1.关闭集群**

[root@hadoop102 ~]# /opt/module/presto/bin/launcher stop

[root@hadoop103 ~]# /opt/module/presto/bin/launcher stop

[root@hadoop104 ~]# /opt/module/presto/bin/launcher stop

**2.修改Presto安装路径所有者为presto**

[root@hadoop102 ~]# chown -R presto:hadoop /opt/module/presto

[root@hadoop103 ~]# chown -R presto:hadoop /opt/module/presto

[root@hadoop104 ~]# chown -R presto:hadoop /opt/module/presto

**3.使用hive用户启动MetaStore服务**

[root@hadoop102 ~]# sudo -i -u hive hive --service metastore

**4.使用presto用户启动Presto集群**

[root@hadoop102 ~]# sudo -i -u presto /opt/module/presto/bin/launcher start

[root@hadoop103 ~]# sudo -i -u presto /opt/module/presto/bin/launcher start

[root@hadoop104 ~]# sudo -i -u presto /opt/module/presto/bin/launcher start

### 3.2.8 客户端认证访问Presto集群

[root@hadoop102 presto]# ./prestocli \

--server https://hadoop102:7778 \

--catalog hive \

--schema default \

--enable-authentication \

--krb5-remote-service-name presto \

--krb5-config-path /etc/krb5.conf \

--krb5-principal atguigu@EXAMPLE.COM \

--krb5-keytab-path /home/atguigu/atguigu.keytab \

--keystore-path /etc/security/keytab/keystore.jks \

--keystore-password 123456 \

--user atguigu

# 第4章 即席查询之Kylin

## 4.1 改动说明

从Kylin的架构，可以看出Kylin充当只是一个Hadoop客户端，读取Hive数据，利用MR或Spark进行计算，将Cube存储至HBase中。所以在安全的Hadoop环境下，Kylin不需要做额外的配置，只需要具备一个Kerberos主体，进行常规的认证即可。

但是Kylin所依赖的HBase需要进行额外的配置，才能在安全的Hadoop环境下正常工作。

## 4.2 改动实操

### 4.2.1 HBase开启Kerberos认证

**1.用户准备**

1）在各节点创建hbase系统用户

[root@hadoop102 ~]# useradd -g hadoop hbase

[root@hadoop102 ~]# echo hbase | passwd --stdin hbase

[root@hadoop103 ~]# useradd -g hadoop hbase

[root@hadoop103 ~]# echo hbase | passwd --stdin hbase

[root@hadoop104 ~]# useradd -g hadoop hbase

[root@hadoop104 ~]# echo hbase | passwd --stdin hbase

2）创建hbase Kerberos主体

（1）在hadoop102节点创建主体，生成密钥文件，并修改所有者

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"addprinc -randkey hbase/hadoop102"

[root@hadoop102 ~]# kadmin -padmin/admin -wadmin -q"xst -k /etc/security/keytab/hbase.service.keytab hbase/hadoop102"

[root@hadoop102 ~]# chown hbase:hadoop /etc/security/keytab/hbase.service.keytab

（2）在hadoop103节点创建主体，生成密钥文件，并修改所有者

[root@hadoop103 ~]# kadmin -padmin/admin -wadmin -q"addprinc -randkey hbase/hadoop103"

[root@hadoop103 ~]# kadmin -padmin/admin -wadmin -q"xst -k /etc/security/keytab/hbase.service.keytab hbase/hadoop103"

[root@hadoop103 ~]# chown hbase:hadoop /etc/security/keytab/hbase.service.keytab

（3）在hadoop104节点创建主体，生成密钥文件，并修改所有者

[root@hadoop104 ~]# kadmin -padmin/admin -wadmin -q"addprinc -randkey hbase/hadoop104"

[root@hadoop104 ~]# kadmin -padmin/admin -wadmin -q"xst -k /etc/security/keytab/hbase.service.keytab hbase/hadoop104"

[root@hadoop104 ~]# chown hbase:hadoop /etc/security/keytab/hbase.service.keytab

**2.修改HBase配置文件**

修改$HBASE\_HOME/conf/hbase-site.xml配置文件，增加以下参数

[root@hadoop102 ~]# vim $HBASE\_HOME/conf/hbase-site.xml

<property>

<name>hbase.security.authentication</name>

<value>kerberos</value>

</property>

<property>

<name>hbase.master.kerberos.principal</name>

<value>hbase/\_HOST@EXAMPLE.COM</value>

</property>

<property>

<name>hbase.master.keytab.file</name>

<value>/etc/security/keytab/hbase.service.keytab</value>

</property>

<property>

<name>hbase.regionserver.kerberos.principal</name>

<value>hbase/\_HOST@EXAMPLE.COM</value>

</property>

<property>

<name>hbase.regionserver.keytab.file</name>

<value>/etc/security/keytab/hbase.service.keytab</value>

</property>

<property>

<name>hbase.coprocessor.region.classes</name>

<value>org.apache.hadoop.hbase.security.token.TokenProvider</value>

</property>

**3.分发配置文件**

[root@hadoop102 ~]# xsync $HBASE\_HOME/conf/hbase-site.xml

**4.修改hbase.rootdir路径所有者**

1）使用hdfs/hadoop用户进行认证

[root@hadoop102 ~]# kinit hdfs/hadoop

2）修改所有者

[root@hadoop102 ~]# hadoop fs -chown -R hbase:hadoop /hbase

**5.启动HBase**

1）修改各节点HBase安装目录所有者

[root@hadoop102 ~]# chown -R hbase:hadoop /opt/module/hbase

[root@hadoop103 ~]# chown -R hbase:hadoop /opt/module/hbase

[root@hadoop104 ~]# chown -R hbase:hadoop /opt/module/hbase

2）配置hbase用户从主节点（hadoop102）到所有节点的ssh免密

3）使用hbase用户启动HBase

[root@hadoop102 ~]# sudo -i -u hbase start-hbase.sh

**6.停止HBase**

启用Kerberos认证之后，关闭HBase时，需先进行Kerberos用户认证，认证的主体为hbase。

1）认证为hbase主体

[root@hadoop102 ~]# sudo -i -u hbase kinit -kt /etc/security/keytab/hbase.service.keytab hbase/hadoop102

2）停止hbase

[root@hadoop102 ~]# sudo -i -u hbase stop-hbase.sh

### 4.2.2 Kylin进行Kerberos认证

**1.用户准备**

创建kylin系统用户

[root@hadoop102 ~]# useradd -g hadoop kylin

[root@hadoop102 ~]# echo kylin | passwd --stdin kylin

**2.修改kylin.env.hdfs-working-dir路径所有者为kylin**

1）使用hdfs/hadoop用户进行认证

[root@hadoop102 ~]# kinit hdfs/hadoop

2）修改所有者

[root@hadoop102 ~]# hadoop fs -chown -R hive:hadoop /kylin

**3.修改/opt/module/kylin所有者为kylin**

[root@hadoop102 ~]# chown -R kylin:hadoop /opt/module/kylin

**4.启动kylin**

1）在kylin用户下认证为hive主体

[root@hadoop102 ~]# sudo -i -u kylin kinit -kt /etc/security/keytab/hive.keytab hive

2）以kylin用户的身份启动kylin

[root@hadoop102 ~]# sudo -i -u kylin /opt/module/kylin/bin/kylin.sh start