YARN配置Kerberos认证

2014.11.05 | Comments

关于 Kerberos 的安装和 HDFS 配置 kerberos 认证，请参考 HDFS配置kerberos认证 (/2014/11/04/config-kerberos-in-cdh-hdfs.html)。

1. 环境说明

系统环境：

- 操作系统：CentOs 6.6
- Hadoop版本： CDH5.4
- JDK版本： 1.7.0_71
- 运行用户：root

集群各节点角色规划为：

```
192.168.56.121        cdh1     NameNode、ResourceManager、HBase、Hive metastore、Impala Catalog、Impala statestore、Sentry
192.168.56.122        cdh2     DataNode、SecondaryNameNode、NodeManager、HBase、Hive Server2、Impala Server
192.168.56.123        cdh3     DataNode、HBase、NodeManager、Hive Server2、Impala Server
```

cdh1作为master节点，其他节点作为slave节点，hostname 请使用小写，要不然在集成 kerberos 时会出现一些错误。

2. 生成 keytab

在 cdh1 节点，即 KDC server 节点上执行下面命令：

```
cd /var/kerberos/krb5kdc/

kadmin.local -q "addprinc -randkey yarn/cdh1@JAVACHEN.COM "
kadmin.local -q "addprinc -randkey yarn/cdh2@JAVACHEN.COM "
kadmin.local -q "addprinc -randkey yarn/cdh3@JAVACHEN.COM "

kadmin.local -q "addprinc -randkey mapred/cdh1@JAVACHEN.COM "
kadmin.local -q "addprinc -randkey mapred/cdh2@JAVACHEN.COM "
kadmin.local -q "addprinc -randkey mapred/cdh3@JAVACHEN.COM "

kadmin.local -q "xst  -k yarn.keytab  yarn/cdh1@JAVACHEN.COM "
kadmin.local -q "xst  -k yarn.keytab  yarn/cdh2@JAVACHEN.COM "
kadmin.local -q "xst  -k yarn.keytab  yarn/cdh3@JAVACHEN.COM "

kadmin.local -q "xst  -k mapred.keytab  mapred/cdh1@JAVACHEN.COM "
kadmin.local -q "xst  -k mapred.keytab  mapred/cdh2@JAVACHEN.COM "
kadmin.local -q "xst  -k mapred.keytab  mapred/cdh3@JAVACHEN.COM "
```

拷贝 yarn.keytab 和 mapred.keytab 文件到其他节点的 `/etc/hadoop/conf` 目录

```
$ scp yarn.keytab mapred.keytab cdh1:/etc/hadoop/conf
$ scp yarn.keytab mapred.keytab cdh2:/etc/hadoop/conf
$ scp yarn.keytab mapred.keytab cdh3:/etc/hadoop/conf
```

并设置权限，分别在 cdh1、cdh2、cdh3 上执行：

```
$ ssh cdh1 "cd /etc/hadoop/conf/;chown yarn:hadoop yarn.keytab;chown mapred:hadoop mapred.keytab ;chmod 400 *.keytab"
$ ssh cdh2 "cd /etc/hadoop/conf/;chown yarn:hadoop yarn.keytab;chown mapred:hadoop mapred.keytab ;chmod 400 *.keytab"
$ ssh cdh3 "cd /etc/hadoop/conf/;chown yarn:hadoop yarn.keytab;chown mapred:hadoop mapred.keytab ;chmod 400 *.keytab"
```

由于 keytab 相当于有了永久凭证，不需要提供密码(如果修改 kdc 中的 principal 的密码，则该 keytab 就会失效)，所以其他用户如果对该文件有读权限，就可以冒充 keytab 中指定的用户身份访问 hadoop，所以 keytab 文件需要确保只对 owner 有读权限(`0400`)

3. 修改 YARN 配置文件

修改 yarn-site.xml，添加下面配置：

```
<property>
  <name>yarn.resourcemanager.keytab</name>
  <value>/etc/hadoop/conf/yarn.keytab</value>
</property>
<property>
  <name>yarn.resourcemanager.principal</name>
  <value>yarn/_HOST@JAVACHEN.COM</value>
</property>

<property>
  <name>yarn.nodemanager.keytab</name>
  <value>/etc/hadoop/conf/yarn.keytab</value>
</property>
<property>
  <name>yarn.nodemanager.principal</name>
  <value>yarn/_HOST@JAVACHEN.COM</value>
</property>
<property>
  <name>yarn.nodemanager.container-executor.class</name>
  <value>org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor</value>
</property>
<property>
  <name>yarn.nodemanager.linux-container-executor.group</name>
  <value>yarn</value>
</property>
```

如果想要 YARN 开启 SSL，则添加：

```
<property>
  <name>yarn.http.policy</name>
  <value>HTTPS_ONLY</value>
</property>
```

修改 mapred-site.xml，添加如下配置：

```
<property>
  <name>mapreduce.jobhistory.keytab</name>
  <value>/etc/hadoop/conf/mapred.keytab</value>
</property>
<property>
  <name>mapreduce.jobhistory.principal</name>
  <value>mapred/_HOST@JAVACHEN.COM</value>
</property>
```

如果想要 mapreduce jobhistory 开启 SSL，则添加：

```
<property>
  <name>mapreduce.jobhistory.http.policy</name>
  <value>HTTPS_ONLY</value>
</property>
```

在 `/etc/hadoop/conf` 目录下创建 container-executor.cfg 文件，内容如下：

```
#configured value of yarn.nodemanager.linux-container-executor.group
yarn.nodemanager.linux-container-executor.group=yarn
#comma separated list of users who can not run applications
banned.users=bin
#Prevent other super-users
min.user.id=0
#comma separated list of system users who CAN run applications
allowed.system.users=root,nobody,impala,hive,hdfs,yarn
```

设置该文件权限：

```
$ chown root:yarn container-executor.cfg
$ chmod 400 container-executor.cfg

$ ll container-executor.cfg
-r-------- 1 root yarn 354 11-05 14:14 container-executor.cfg
```

注意：

- `container-executor.cfg` 文件读写权限需设置为 `400`，所有者为 `root:yarn`。
- `yarn.nodemanager.linux-container-executor.group` 要同时配置在 yarn-site.xml 和 container-executor.cfg，且其值需要为运行 NodeManager 的用户所在的组，这里为 yarn。
- `banned.users` 不能为空，默认值为 `hfds,yarn,mapred,bin`
- `min.user.id` 默认值为 1000，在有些 centos 系统中，用户最小 id 为500，则需要修改该值
- 确保 `yarn.nodemanager.local-dirs` 和 `yarn.nodemanager.log-dirs` 对应的目录权限为 `755` 。

设置 /usr/lib/hadoop-yarn/bin/container-executor 读写权限为 `6050` 如下：

```
$ chown root:yarn /usr/lib/hadoop-yarn/bin/container-executor
$ chmod 6050 /usr/lib/hadoop-yarn/bin/container-executor

$ ll /usr/lib/hadoop-yarn/bin/container-executor
---Sr-s--- 1 root yarn 333 11-04 19:11 container-executor
```

测试是否配置正确：

```
$ /usr/lib/hadoop-yarn/bin/container-executor --checksetup
```

如果提示错误，则查看 NodeManger 的日志，然后对照 YARN ONLY: Container-executor Error Codes (http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cdh_sg_other_hadoop_security.html?scroll=topic_18_unique_2) 查看错误对应的问题说明。

关于 LinuxContainerExecutor 的详细说明，可以参考 http://hadoop.apache.org/docs/r2.5.0/hadoop-project-dist/hadoop-common/SecureMode.html#LinuxContainerExecutor (http://hadoop.apache.org/docs/r2.5.0/hadoop-project-dist/hadoop-common/SecureMode.html#LinuxContainerExecutor)。

记住将修改的上面文件同步到其他节点：cdh2、cdh3，并再次一一检查权限是否正确。

```
$ cd /etc/hadoop/conf/

$ scp yarn-site.xml mapred-site.xml container-executor.cfg  cdh2:/etc/hadoop/conf/
$ scp yarn-site.xml mapred-site.xml container-executor.cfg  cdh3:/etc/hadoop/conf/

$ ssh cdh2 "cd /etc/hadoop/conf/; chown root:yarn container-executor.cfg ; chmod 400 container-executor.cfg"
$ ssh cdh3 "cd /etc/hadoop/conf/; chown root:yarn container-executor.cfg ; chmod 400 container-executor.cfg"
```

4. 启动服务

启动 ResourceManager

resourcemanager 是通过 yarn 用户启动的，故在 cdh1 上先获取 yarn 用户的 ticket 再启动服务：

```
$ kinit -k -t /etc/hadoop/conf/yarn.keytab yarn/cdh1@JAVACHEN.COM
$ service hadoop-yarn-resourcemanager start
```

然后查看日志，确认是否启动成功。

启动 NodeManager

resourcemanager 是通过 yarn 用户启动的，故在 cdh2 和 cdh3 上先获取 yarn 用户的 ticket 再启动服务：

```
$ ssh cdh2 "kinit -k -t /etc/hadoop/conf/yarn.keytab yarn/cdh2@JAVACHEN.COM ;service hadoop-yarn-nodemanager start"
$ ssh cdh3 "kinit -k -t /etc/hadoop/conf/yarn.keytab yarn/cdh3@JAVACHEN.COM ;service hadoop-yarn-nodemanager start"
```

启动 MapReduce Job History Server

resourcemanager 是通过 mapred 用户启动的，故在 cdh1 上先获取 mapred 用户的 ticket 再启动服务：

```
$ kinit -k -t /etc/hadoop/conf/mapred.keytab mapred/cdh1@JAVACHEN.COM
$ service hadoop-mapreduce-historyserver start
```

5. 测试

检查 web 页面是否可以访问：http://cdh1:8088/cluster

运行一个 mapreduce 的例子：

```
$ klist
  Ticket cache: FILE:/tmp/krb5cc_1002
  Default principal: yarn/cdh1@JAVACHEN.COM

  Valid starting     Expires            Service principal
  11/10/14 11:18:55  11/11/14 11:18:55  krbtgt/cdh1@JAVACHEN.COM
    renew until 11/17/14 11:18:55


  Kerberos 4 ticket cache: /tmp/tkt1002
  klist: You have no tickets cached

$ hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar pi 10 10000
```

如果没有报错，则说明配置成功。最后运行的结果为：

```
Job Finished in 54.56 seconds
Estimated value of Pi is 3.14120000000000000000
```

如果出现下面错误，请检查环境变量中 `HADOOP_YARN_HOME` 是否设置正确，并和 `yarn.application.classpath` 中的保持一致。

```
14/11/13 11:41:02 INFO mapreduce.Job: Job job_1415849491982_0003 failed with state FAILED due to: Application applicati
on_1415849491982_0003 failed 2 times due to AM Container for appattempt_1415849491982_0003_000002 exited with  exitCod
e: 1 due to: Exception from container-launch.
Container id: container_1415849491982_0003_02_000001
Exit code: 1
Stack trace: ExitCodeException exitCode=1:
  at org.apache.hadoop.util.Shell.runCommand(Shell.java:538)
  at org.apache.hadoop.util.Shell.run(Shell.java:455)
  at org.apache.hadoop.util.Shell$ShellCommandExecutor.execute(Shell.java:702)
  at org.apache.hadoop.yarn.server.nodemanager.LinuxContainerExecutor.launchContainer(LinuxContainerExecutor.java:281)
  at org.apache.hadoop.yarn.server.nodemanager.containermanager.launcher.ContainerLaunch.call(ContainerLaunch.java:299)
  at org.apache.hadoop.yarn.server.nodemanager.containermanager.launcher.ContainerLaunch.call(ContainerLaunch.java:81)
  at java.util.concurrent.FutureTask$Sync.innerRun(FutureTask.java:303)
  at java.util.concurrent.FutureTask.run(FutureTask.java:138)
  at java.util.concurrent.ThreadPoolExecutor$Worker.runTask(ThreadPoolExecutor.java:886)
  at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:908)
  at java.lang.Thread.run(Thread.java:662)

Shell output: main : command provided 1
main : user is yarn
main : requested yarn user is yarn


Container exited with a non-zero exit code 1
.Failing this attempt.. Failing the application.
14/11/13 11:41:02 INFO mapreduce.Job: Counters: 0
Job Finished in 13.428 seconds
java.io.FileNotFoundException: File does not exist: hdfs://cdh1:8020/user/yarn/QuasiMonteCarlo_1415850045475_708291630/
out/reduce-out
```