CDH集群搭建简介

1一、安装前准备

PS：此安装文档适用于CDH5.12版本，操作系统版本7以上。

1.1 修改所有节点主机名

```
hostnamectl set-hostname cdh81-30
 ...
```

1.2 JDK安装

卸载系统自带的jdk：

查看：rpm -qa|grep -i jdk

卸载：rpm -e java-1.6.0-openjdk-1.6.0.0-1.66.1.13.0.el6.x86_64 --nodeps

安装jdk并配置环境变量，版本为1.8

1.3 修改所有节点主机名和IP映射关系

```
vim /etc/hosts
192.168.81.30  cdh81-30
192.168.81.31  cdh81-31
192.168.81.32  cdh81-32
192.168.81.33  cdh81-33
192.168.81.34  cdh81-34
192.168.81.35  cdh81-35
192.168.81.36  cdh81-36
192.168.81.37  cdh81-37
192.168.81.38  cdh81-38
192.168.81.39  cdh81-39
192.168.81.40  cdh81-40
192.168.81.41  cdh81-41
192.168.81.42  cdh81-42
192.168.81.43  cdh81-43
192.168.81.44  cdh81-44
192.168.81.45  cdh81-45
```

```
192.168.81.46  cdh81-46
192.168.81.47  cdh81-47
192.1686.81.48  cdh81-48
192.168.81.49  cdh81-49
192.168.81.50  cdh81-50
192.168.81.51  cdh81-51
192.168.81.52  cdh81-52
192.168.81.53  cdh81-53
192.168.81.54  cdh81-54
192.168.81.55  cdh81-55
192.168.81.56  cdh81-56
192.168.81.57  cdh81-57
192.168.81.58  cdh81-58
192.168.81.59  cdh81-59
```

1.4 配置ssh免登陆

在cdh81-30、cdh81-40、cdh81-50上分别生成一对钥匙

```
ssh-keygen -t rsa
...

#将公钥拷贝到其他节点，包括自己
ssh-copy-id cdh81-30
...
```

1.5 关闭防火墙

```
#查看防火墙状态
firewall-cmd  --state

#关闭防火墙
systemctl stop firewalld.service

#关闭防火墙开启启动
systemctl disable firewalld.service

#关闭SELINUX
```

setenforce 0 (临时生效)

修改/etc/selinux/config 下的 SELINUX=disabled （重启后生效）。

1.6 配置NTP服务

停止系统自带的ntp服务：systemctl stop chronyd

关闭开机自启动ntp服务：systemctl disable chronyd

查看系统自带ntp服务状态：systemctl status chronyd

所有节点安装相关组件： yum -y install ntp

配置开机启动： systemctl enable ntpd

检查是否设置成功： systemctl list-unit-files |grep ntpd

主节点配置（cdh81-50）

vi /etc/ntp.conf

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool
(http://www.pool.ntp.org/join.html).
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
#server 101.231.72.162 prefer
#server time3.aliyun.com iburst
server 192.168.23.108 iburst
```

启动ntp服务：systemctl start ntpd

配置ntp客户端（其他所有节点）

vi /etc/ntp.conf

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool
(http://www.pool.ntp.org/join.html).
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
```

```
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
server cdh81-50 prefer
server 192.168.23.108
```

ok保存退出，请求服务器前，请先使用ntpdate手动同步一下时间： ntpdate -u n1（主节点ntp服务器）

启动服务： systemctl start ntpd

查看服务状态：ntpstat

1.7 安装Mysql客户端

除主节点以外的其他节点

先卸载系统自带的mysql：yum remove -y mysql-community-common

安装mysql客户端：yum install -y mariadb

二、离线安装CM

下载对应CDH版本的CM，下载地址：http://archive.cloudera.com/cm5/redhat/

2.1 上传rpm文件到主节点（cdh-81-50）

cloudera-manager-agent-5.12.1-1.cm5121.p0.6.el7.x86_64.rpm

cloudera-manager-daemons-5.12.1-1.cm5121.p0.6.el7.x86_64.rpm

cloudera-manager-server-5.12.1-1.cm5121.p0.6.el7.x86_64.rpm

2.2 安装Cloudera Manager Server

在cdh81-50上运行：

```
yum -y localinstall cloudera-manager-daemons-5.12.1-
1.cm5121.p0.6.el7.x86_64.rpm
yum -y localinstall cloudera-manager-server-5.12.1-
1.cm5121.p0.6.el7.x86_64.rpm
yum -y localinstall cloudera-manager-agent-5.12.1-
1.cm5121.p0.6.el7.x86_64.rpm
```

2.3 安装Cloudera Manager Agent

除cdh81-50以外的其他节点

```
yum -y localinstall cloudera-manager-daemons-5.12.1-
1.cm5121.p0.6.el7.x86_64.rpm
yum -y localinstall cloudera-manager-agent-5.12.1-
1.cm5121.p0.6.el7.x86_64.rpm
```

配置/etc/cloudera-scm-agent/config.ini修改server_host= (Name of the host where Cloudera Manager Server is running.)

2.4 配置外置数据库

在cdh81-50上传mysql驱动包到/user/share/java/目录，并做软连接

```
cd /usr/share/java && ln -s /usr/share/java/mysql-connector-java-
5.1.44-bin.jar  mysql-connector-java.jar
```

指定CM外部数据源：

```
/usr/share/cmf/schema/scm_prepare_database.sh mysql -h
192.168.81.200 -uroot -p密码 --scm-host 192.168.81.50 baofoo_scm
scm 密码
```

三、离线安装CDH

3.1 上传文件

在cdh81-50上，将CHD5相关的Parcel包放到主节点的/opt/cloudera/parcel-repo/

相关文件如下：

```
CDH-5.12.1-1.cdh5.12.1.p0.3-el7.parcel
CDH-5.12.1-1.cdh5.12.1.p0.3-el7.parcel.sha1
manifest.json
```

下载地址：http://archive.cloudera.com/cdh5/parcels/

最后将CDH-5.12.1-1.cdh5.12.1.p0.3-el7.parcel.sha1

，重命名为CDH-5.12.1-1.cdh5.12.1.p0.3-el7.parcel.sha

，这点必须注意，否则，系统会重新下载CDH-5.7.0-1.cdh5.7.0.p0.45-

el6.parcel.sha1

文件。

3.2 启动进程

启动进程，主节点(cdh81-50)：

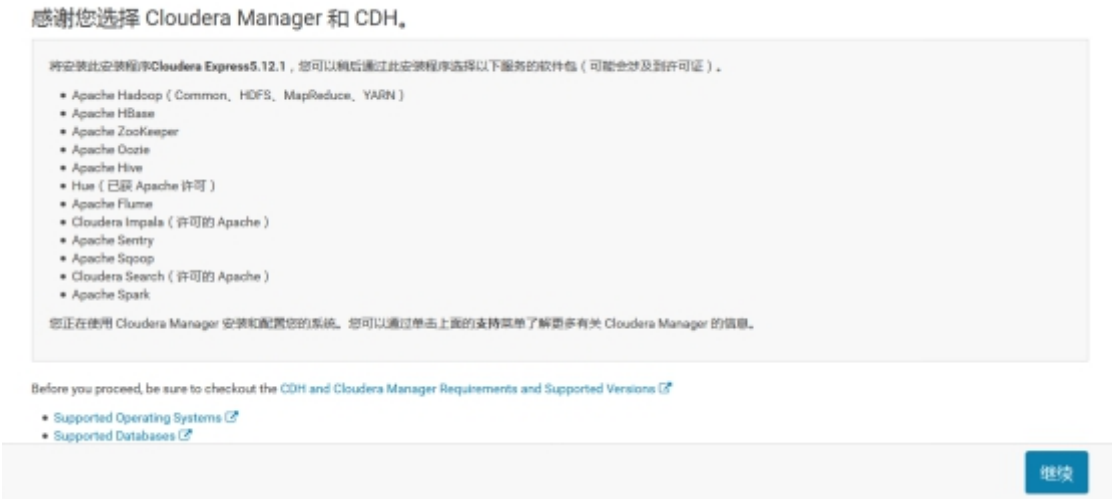
```
systemctl start cloudera-scm-server
systemctl start cloudera-scm-agent
```

启动进程，各子节点(除cdh81-50以外的所有节点)：

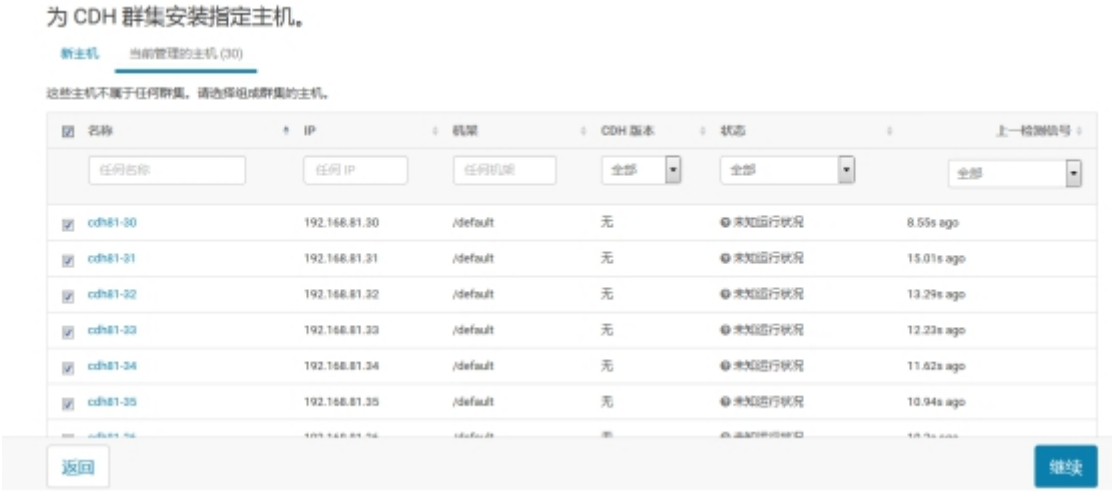
```
systemctl start cloudera-scm-agent
```

3.3 登录管理界面

等待cm元数据库初始化完成，大概需要几分钟，随后浏览器访问：192.168.81.50:7180，默认帐号admin，默认密码admin



点击继续

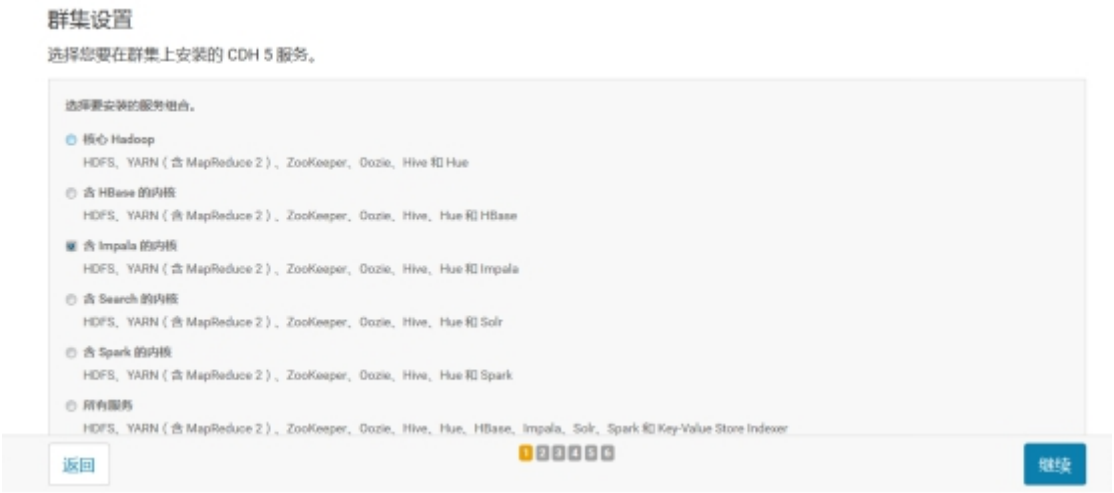


勾选主机，点击继续

点击继续，等待安装，安装完成如下图：

点击继续

根据提示修改，点击完成



我们这里选含Impala的内核安装，点击继续

3.4 配置服务角色

HDFS:

NameNode cdh81-30

SecondaryNameNode cdh81-40 暂时先这样，安装好后启用HA

DataNode 除30、40、50以外的其它主机

Balancer cdh81-50

HttpFs cdh81-50

HIVE:

Gateway cdh81-50

HiveMetastoreServer cdh81-50

HiveServer2 cdh81-50

HUE:

Hue Server cdh81-51、52、53

Load Balancer cdh81-52

Impala：

ImpalaStateStore cdh81-30

ImpalaCatalogServer cdh81-40

Impala Daemon 除30、40、50以外的其它主机

Cloudera Management Service：

cdh81-50

Oozie:
Oozie Server cdh81-50

Yarn：
ResourceManager cdh81-30
JobHistory cdh81-50
NodeManager 除30、40、50以外的其它主机

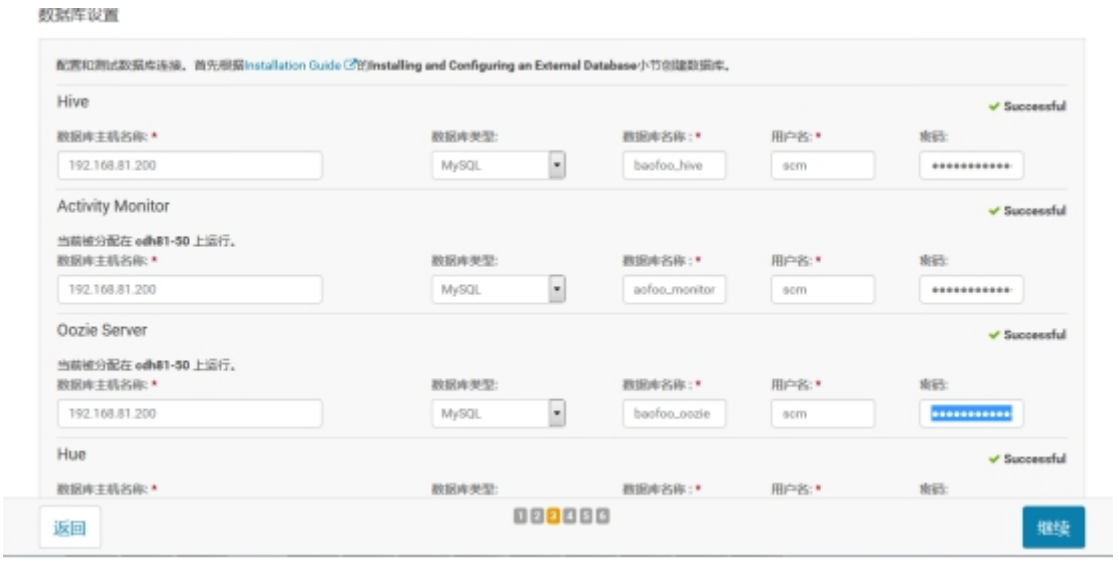
Zookeeper：
Zookeeper Server cdh81-30、40、50

创建hive、oozie、sentry、monitor、hue库并授权：baofoo_hive、baofoo_oozie、baofoo_sentry、baofoo_monitor、baofoo_hue

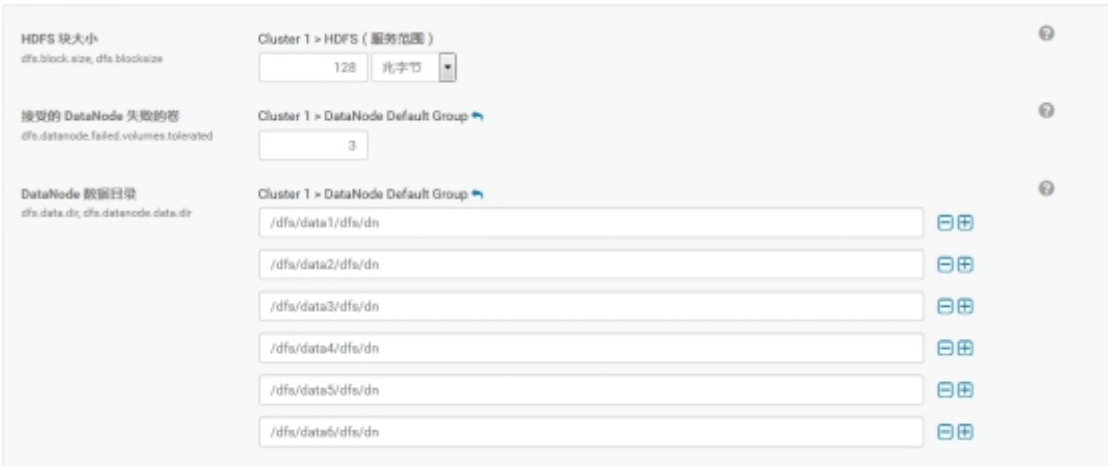
拷贝mysql驱动到hive jar包（cdh81-50）
cd /opt/cloudera/parcels/CDH/lib/hive/lib && ln -s /usr/share/java/mysql-connector-java-5.1.44-bin.jar mysql-connector-java.jar

拷贝mysql驱动到oozie jar包（cdh81-50）
cd /opt/cloudera/parcels/CDH/lib/oozie/libtools && ln -s /usr/share/java/mysql-connector-java-5.1.44-bin.jar mysql-connector-java.jar

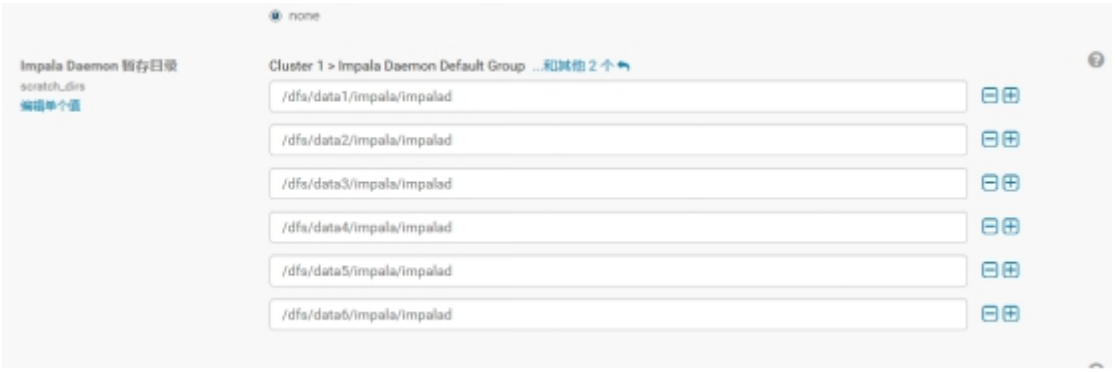
cd /opt/cloudera/parcels/CDH/lib/oozie/libserver && ln -s /usr/share/java/mysql-connector-java-5.1.44-bin.jar mysql-connector-java.jar

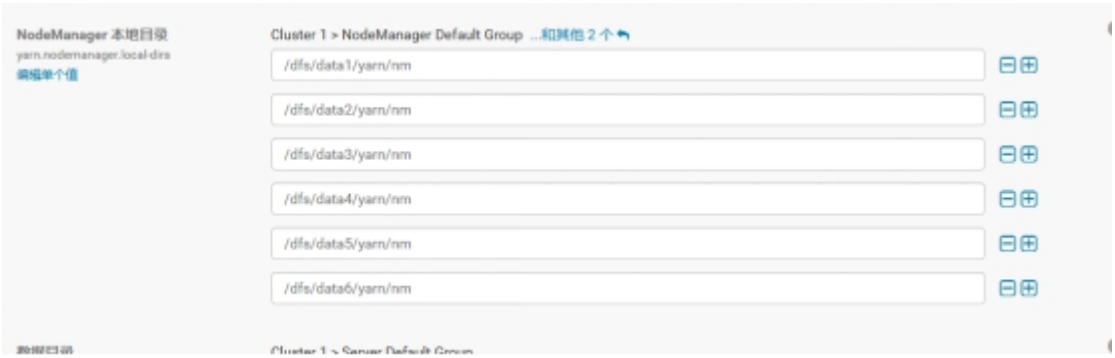

数据库设置
配置和测试数据库连接。首先根据Installation Guide 的Installing and Configuring an External Database小节创建数据库。
Hive
Successful
数据库主机名称: *
数据库类型:
数据库名称: *
用户名: *
密码:
192.168.81.200
MySQL
baofoo_hive
scm
Activity Monitor
Successful
当前被分配在 cdh81-50 上运行。
192.168.81.200
MySQL
aofoo_monitor
scm
Oozie Server
Successful
当前被分配在 cdh81-50 上运行。
192.168.81.200
MySQL
baofoo_oozie
scm
Hue
Successful
返回
继续

点击继续

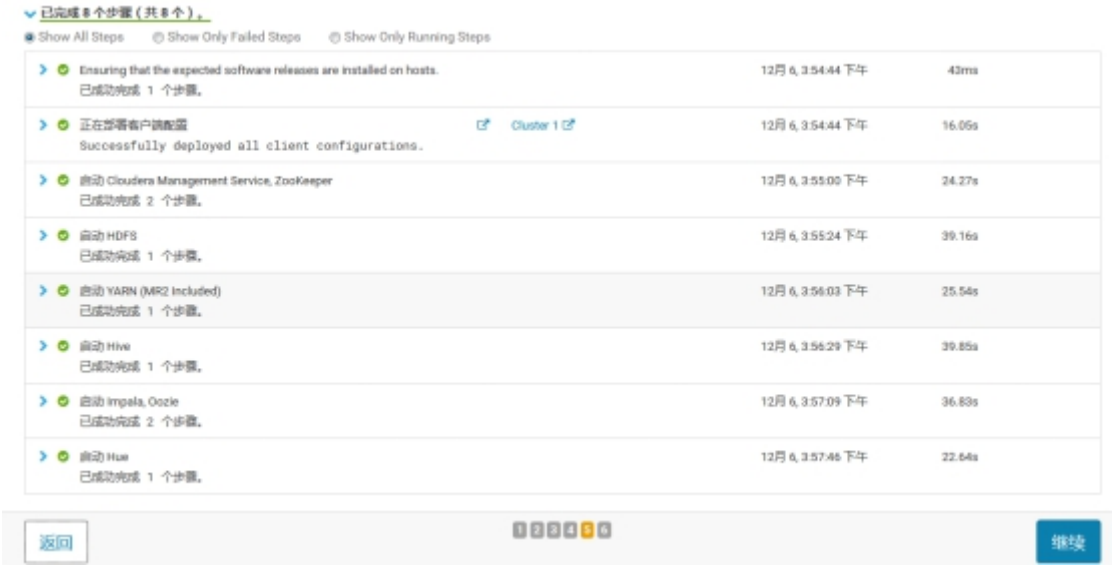

HDFS 块大小
dfs.block.size, dfs.blocksize
Cluster 1 > HDFS (服务范围)
128
兆字节
接受的 DataNode 失败的卷
dfs.datanode.failed.volumes.tolerated
Cluster 1 > DataNode Default Group
3
DataNode 数据目录
dfs.data.dir, dfs.datanode.data.dir
Cluster 1 > DataNode Default Group
/dfs/data1/dfs/dn
/dfs/data2/dfs/dn
/dfs/data3/dfs/dn
/dfs/data4/dfs/dn
/dfs/data5/dfs/dn
/dfs/data6/dfs/dn


NameNode 数据目录
dfs.name.dir, dfs.namenode.name.dir
Cluster 1 > NameNode Default Group
/opt/dfs/nn
HDFS 检查点目录
fs.checkpoint.dir,
dfs.namenode.checkpoint.dir
Cluster 1 > SecondaryNameNode Default Group
/opt/dfs/snn
Hive 仓库目录
hive.metastore.warehouse.dir
Cluster 1 > Hive (服务范围)
/user/hive/warehouse
Hive Metastore 服务器端口
hive.metastore.port
Cluster 1 > Hive Metastore Server Default Group
9083
Kudu 服务
Cluster 1 > Impala (服务范围)
none





点击继续，等待启动。



点击继续即可完成。

四、Kerberos安装

4.1 Kerberos服务端安装

在cdh81-50上安装：

```
yum install -y krb5-server krb5-libs krb5-workstation
```

修改**/etc/krb5.conf**

```
[logging]
  default = FILE:/var/log/krb5libs.log
  kdc = FILE:/var/log/krb5kdc.log
  admin_server = FILE:/var/log/kadmind.log

[libdefaults]
  dns_lookup_realm = false
  dns_lookup_kdc = false
  ticket_lifetime = 24h
  renew_lifetime = 7d
  forwardable = true
  default_realm = master
  renewable = true

[realms]
  master = {
   kdc = cdh81-50
   admin_server = cdh81-50
  }

[domain_realm]
  .master = master
  master = master
```

修改**/var/kerberos/krb5kdc/kdc.conf**

```
[kdcdefaults]
  kdc_ports = 88
  kdc_tcp_ports = 88

[realms]
  master = {
   #master_key_type = aes256-cts
   acl_file = /var/kerberos/krb5kdc/kadm5.acl
```

```
   dict_file = /usr/share/dict/words
max_renewable_life = 7d
   max_life = 1d
   admin_keytab = /var/kerberos/krb5kdc/kadm5.keytab
   supported_enctypes = aes256-cts:normal aes128-cts:normal des3-
hmac-sha1:normal arcfour-hmac:normal camellia256-cts:normal
camellia128-cts:normal des-hmac-sha1:normal des-cbc-md5:normal
des-cbc-crc:normal
  }
```

修改**/var/kerberos/krb5kdc/kadm5.acl**

```
*/admin@master  *
```

以上三个文件配置完毕后，只需拷贝krb5.conf到集群中其他机器上即可。

```
scp /etc/krb5.conf cdh81-30:/etc/
...
```

4.2 Kerberos客户端安装

在其他几点上安装：

```
yum install -y krb5-libs krb5-workstation
```

或者在cdh81-50上调用脚本：

```
sh /app/shell/exe_command_on_all_nodes_1.sh "scp /app/krb5-*" "yum -
y localinstall krb5*.rpm && rm -rf ~/krb5*.rpm"
```

4.3 关于**AES-256**加密

oracle官网下载jce_policy-8.zip，解压，将local_policy.jar、US_export_policy.jar拷贝 到$JAVA_HOME/jre/lib/security目录下

4.4 创建数据库

在 cdh81-50 上运行初始化数据库命令。其中 -r 指定对应 realm，初始密码*** $

```
kdb5_util create -r master -s
```

出现 Loading random data 的时候另开个终端执行点消耗CPU的命令如 cat /dev/sda > /dev/urandom 可以加快随机数采集。该命令会

在 /var/kerberos/krb5kdc/ 目录下创建 principal 数据库。

如果遇到数据库已经存在的提示，可以把 /var/kerberos/krb5kdc/ 目录下的 principal 的相关文件都删除掉。默认的数据库名字都是 principal。可以使用 -d 指定数据库名字。

4.5 启动服务

在cdh81-50上执行如下命令：

```
systemctl enable krb5kdc.service
systemctl enable kadmin.service
systemctl start krb5kdc
systemctl start kadmin
```

4.6 创建Kerveros管理员

关于 kerberos 的管理，可以使用 kadmin.local 或 kadmin，至于使用哪个，取决于账户和访问权限：

如果有访问 kdc 服务器的 root 权限，但是没有 kerberos admin 账户，使用 kadmin.local

如果没有访问 kdc 服务器的 root 权限，但是用 kerberos admin 账户，使用 kadmin

在 cdh81-50 上创建远程管理的管理员：

在KDC server主机上，创建一个名为『hadoop』的principal，并将其密码设为『***』。执行命令：

```
[root@cdh81-50 /]# kadmin.local
Authenticating as principal root/admin@master with password.
kadmin.local: addprinc -pw *** hadoop/admin@master
```

通过执行kadmin.local中的listprincs命令可以看到创建了一个名为『hadoop/admin@master』的principal：

```
kadmin.local:  listprincs
K/M@master
hadoop/admin@master
kadmin/admin@master
kadmin/cdh81-50@master
kadmin/changepw@master
```

kiprop/cdh81-50@master

krbtgt/master@master

principal的名字的第二部分是admin,那么该principal就拥administrative privileges

这个账号将会被CDH用来生成其他用户/服务的principal

4.7 CDH启用**Kerberos**

在CM的界面上点击启用Kerberos，启用的时候需要确认几个事情：

1.KDC已经安装好并且正在运行

2.将KDC配置为允许renewable tickets with non-zerolifetime

- 在之前修改kdc.conf文件的时候已经添加了kdc_tcp_ports、max_life和max_renewable_life这个三个选项

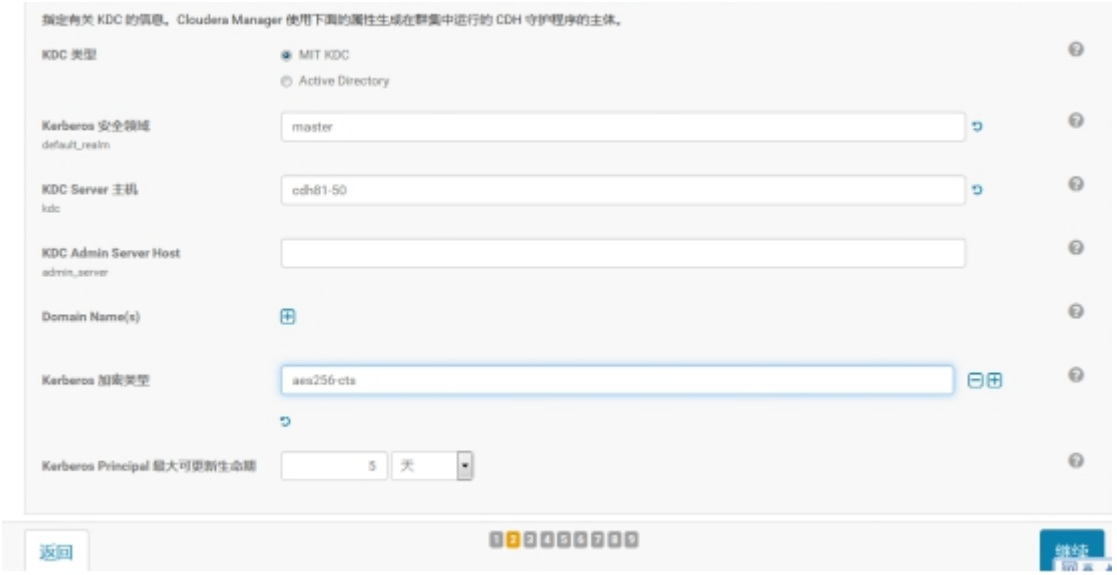
3.在Cloudera Manager Server上安装openldap-clients

PS:为了使Kerberos能够绑定到OpenLDAP服务器，需要创建一个管理员用户和一个principal，并生成keytab文件,所以这里先安装openldap-clients

4.为Cloudera Manager创建一个principal，使其能够有权限在KDC中创建其他的principals，就是上面创建的Kerberos管理员账号.

确定完了之后点击continue，进入下一页进行配置，要注意的是：这里的『Kerberos Encryption Types』必须跟KDC实际支持的加密类型匹配（即kdc.conf中的值）

这里使用了默认的aes256-cts



注意，如果勾选了这个选项就可以通过CM的管理界面来部署krb5.conf，但是实际操作过程中发现有些配置仍然需要手动修改该文件并同步

点击continue，进入下一页，输入Cloudera Manager Principal的管理员账号和密码，注意输入账号的时候要使用@前要使用全称，hadoop/admin

点击continue，进入下一页，导入KDC Account Manager Credentials

点击continue，进入下一页，restart cluster并且enable Kerberos

之后CM会自动重启集群服务，启动之后会会提示Kerberos已启用
这个过程中CM会自动在Kerberos的数据库中创建各个节点中各个账户对应的principle
可以使用 kadmin.local -q "list_principals"查看，，格式为
username/hostname@XIAOHEI.INFO，例如hdfs/hadoop-10-0-8-124@XIAOHEI.INFO

在CM上启用Kerberos的过程中，CM会自动做以下的事情：
1.集群中有多少个节点，每个账户都会生成对应个数的principal
2.为每个对应的principal创建keytab
3.部署keytab文件到指定的节点中
4.在每个服务的配置文件中加入有关Kerberos的配置

其中包括Zookeeper服务所需要的jaas.conf和keytab文件都会自动设定并读取，如果用户仍然手动修改了Zookeeper的服务，要确保这两个文件的路径和内容正确性

keytab是包含principals和加密principal key的文件

keytab文件对于每个host是唯一的，因为key中包含hostname

keytab文件用于不需要人工交互和保存纯文本密码，实现到kerberos上验证一个主机上的principal

启用之后访问集群的所有资源都需要使用相应的账号来访问，否则会无法通过Kerberos的authenticatin

4.8 创建HDFS超级用户

我们使用yarn作为hadoop集群的超级用户，在集群所有节点上创建supergroup组并加入yarn用户。

此时直接用CM生成的principal访问HDFS会失败，因为那些自动生成的principal的密码是随机的，用户并不知道，而通过命令行的方式访问HDFS需要先使用kinit来登录并获得ticket

用户可以通过创建一个yarn@master的principal并记住密码从命令行中访问HDFS

```
# 需要输入两遍密码 kadmin.local -q "addprinc yarn"
```

先使用

```
kinit yarn@master
```

登录之后就可以通过认证并访问HDFS

```
# 查看principals
$ kadmin: list_principals

# 添加一个新的 principal
kadmin: addprinc user1
WARNING: no policy specified for user1@JAVACHEN.COM; defaulting to no policy Enter password for principal "user1@JAVACHEN.COM": Re-enter password for principal "user1@JAVACHEN.COM": Principal "user1@JAVACHEN.COM" created.

# 删除 principal
kadmin: delprinc user1
```

Are you sure you want to delete the principal "user1@JAVACHEN.COM"? (yes/no): yes Principal "user1@JAVACHEN.COM" deleted. Make sure that you have removed this principal from all ACLs before reusing.

kadmin: exit

4.9 确认HDFS可以正常使用

登录到某一个节点后，用kinit来获取yarn用户的credentials
现在用'hadoop fs -ls /'应该能正常输出结果
用kdestroy销毁credentials后，再使用hadoop dfs -ls /会发现报错

获取了yarn的证书后，提交一个PI程序，如果能正常提交并成功运行，则说明Kerberized Hadoop cluster在正常工作

```
hadoop jar /opt/cloudera/parcels/CDH/jars/hadoop-examples.jar pi 10 1000
```

beeline连接hive、impala

hive：beeline -u 'jdbc:hive2://cdh81-50:10000/;principal=hive/cdh81-50@master'

impala：beeline -u 'jdbc:hive2://cdh81-51:21050/;principal=impala/cdh81-51@master'

五、LDAP安装

5.1 服务端安装

在cdh81-50上安装

```
$ yum install -y db4 db4-utils db4-devel cyrus-sasl* krb5-server-ldap
$ yum install -y openldap openldap-servers openldap-clients  openldap-devel
```

有两个文件要复制：slapd的配置文件和数据库文件，将openldap-servers自带的example复制到相应目录：

PS: centos7 slapd.conf.obsolete 并不存在，所以我从centos6 里拷贝了一个过来

```
cp /usr/share/openldap-servers/slapd.conf.obsolete
/etc/openldap/slapd.conf
cp /usr/share/openldap-servers/DB_CONFIG.example
/var/lib/ldap/DB_CONFIG
```

5.2 服务端配置

使用slappasswd创建LDAP管理员密码，这个命令不会直接将密码写入配置，运行slappasswd后输入两次密码，会返回一串密文，复制下这个密文。

```
[root@cdh81-50 ~]# slappasswd
New password:
Re-enter new password:
{SSHA}N3tGSYS8PGYjV66pcqgKNuHj5GuGnX9v
```

编辑/etc/openldap/slapd.conf，找到"database bdb"，按照自己的需求更改下面的：

```
suffix "dc=baofoo,dc=com" rootdn "cn=admin,dc=baofoo,dc=com" //管理员为admin rootpw {SSHA}N3tGSYS8PGYjV66pcqgKNuHj5GuGnX9v //复制的管理员的密码，也支持明文
```

添加一些基本配置，并引入 kerberos 和 openldap 的 schema：

```
$ cp /usr/share/doc/krb5-server-ldap-1.15.1/kerberos.schema
/etc/openldap/schema/
```

在/etc/openldap/slapd.conf加入

```
include /etc/openldap/schema/kerberos.schema
```

更改目录权限：

```
chown -R ldap:ldap /etc/openldap
chown -R ldap:ldap /var/lib/ldap
```

5.3 测试并生成配置文件

```
rm -rf /etc/openldap/slapd.d/* //删除原文件
```

```
systemctl start slapd //生成bdb文件
slaptest -f /etc/openldap/slapd.conf -F /etc/openldap/slapd.d //生成配置文件
chown -R ldap:ldap /etc/openldap/slapd.d
```

5.4 配置完成重启服务

```
systemctl restart slapd
systemctl enable slapd //设置开机启动
systemctl list-unit-files slapd.service //查看开机启动状态
```

经过上面的配置后，openldap server就配置好了。

查看状态，验证服务端口：

```
[root@cdh81-50 openldap]# ps aux | grep slapd | grep -v grep
ldap      58094  0.0  0.0 533648  9748 ?        Ssl  10:11   0:00 /usr/sbin/slapd -u ldap -h ldapi:/// ldap:///
[root@cdh81-50 openldap]# ss -tunlp  | grep :389
tcp    LISTEN     0      128       *:389                   *:*                   users:(("slapd",pid=58094,fd=8))
tcp    LISTEN     0      128      :::389                  :::*                   users:(("slapd",pid=58094,fd=9))
```

查看LDAP数据库结构：

```
[root@cdh81-50 openldap]# ldapsearch -x -H ldap://127.0.0.1 -b 'dc=baofoo,dc=com'
# extended LDIF
#
# LDAPv3
# base <dc=baofoo,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# search result
search: 2
result: 32 No such object
```

```
# numResponses: 1
```

5.5 Kerberos和Ldap集成

为了使Kerberos能够绑定到OpenLDAP服务器，需要创建一个管理员用户和一个principal，并生成keytab文件

设置该文件的权限为LDAP服务运行用户可读（一般为ldap）：

```
kadmin.local -q "addprinc -randkey ldap/cdh81-50@master"

kadmin.local -q "ktadd -k /etc/openldap/ldap.keytab ldap/cdh81-50@master"

chown ldap:ldap /etc/openldap/ldap.keytab && chmod 640 /etc/openldap/ldap.keytab
```

确保LDAP启动时使用上一步中创建的keytab文件，在/etc/sysconfig/ldap增加KRB5_KTNAME配置：

PS: centos7 /etc/sysconfig/ldap 并不存在，所以我从centos6 里拷贝了一个过来

```
export KRB5_KTNAME=/etc/openldap/ldap.keytab
```

重启 systemctl restart slapd

5.6 配置并迁移系统用户

配置好的LDAP数据库是空的，需要将系统上的用户导入到LDAP数据库中。需要用migrationtools将系统用户转换为LDAP能识别的ldif文件。

安装migrationtools：

```
yum install -y migrationtools
```

配置migrationtools：

编辑/usr/share/migrationtools/migrate_common.ph ，按需更改下面两行：

```
$DEFAULT_MAIL_DOMAIN = "baofoo.com";
$DEFAULT_BASE = "dc=baofoo,dc=com";
```

生成模板文件：

```
/usr/share/migrationtools/migrate_base.pl > /opt/base.ldif
```

生成ldif文件：

```
/usr/share/migrationtools/migrate_passwd.pl /etc/passwd
>/opt/passwd.ldif /usr/share/migrationtools/migrate_group.pl
/etc/group >/opt/group.ldif
```

将生成的ldif导入到LDAP数据库：

```
ldapadd -x -D "cn=admin,dc=baofoo,dc=com" -W -f /opt/base.ldif
ldapadd -x -D "cn=admin,dc=baofoo,dc=com" -W -f /opt/passwd.ldif
ldapadd -x -D "cn=admin,dc=baofoo,dc=com" -W -f /opt/group.ldif
```

5.7 LDAP客户端配置

在其他节点上运行

```
yum install openldap-clients -y
```

或者在cdh81-50上调用脚本：

```
sh /app/shell/exe_command_on_all_nodes_1.sh "scp /app/  openldap-
clients-2.4.44-5.el7.x86_64.rpm" "yum -y localinstall openldap-
clients-2.4.44-5.el7.x86_64.rpm && rm -rf ~/  openldap-clients-
2.4.44-5.el7.x86_64.rpm"
```

修改 /etc/openldap/ldap.conf 以下两个配置

```
BASE dc=baofoo,dc=com
URI ldap://cdh81-50
```

然后，运行下面命令测试：

```
#先删除 ticket $ kdestroy
[root@cdh81-50 shell]# ldapsearch -b 'dc=baofoo,dc=com'
SASL/GSS-SPNEGO authentication started
ldap_sasl_interactive_bind_s: Local error (-2)
        additional info: SASL(-1): generic failure: GSSAPI Error:
Unspecified GSS failure.  Minor code may provide more information
(SPNEGO cannot find mechanisms to negotiate)
```

重新获取 ticket：

```
[root@cdh81-50 shell]# kinit yarn
Password for yarn@master:
[root@cdh81-50 shell]# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: yarn@master

Valid starting       Expires              Service principal
12/08/2017 13:33:58  12/09/2017 13:33:58  krbtgt/master@master
        renew until 12/15/2017 13:33:58

$ ldapsearch -x -b 'dc=baofoo,dc=com'
#没有报错
# numEntries: 128
```

5.8 配置HIVE集成LDAP

在hive-site.xml中加入以下配置：

```
<property>
<name>hive.server2.authentication</name>
<value>LDAP</value>
</property>
<property>
<name>hive.server2.authentication.ldap.url</name>
<value>ldap://cdh81-50</value>
</property>
<property>
<name>hive.server2.authentication.ldap.baseDN</name>
<value>ou=people,dc=baofoo,dc=com</value>
</property>
```

重启Hive和Yarn服务,进入beeline测试：

LDAP认证：

```
[root@cdh81-50 ~]# beeline
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option
MaxPermSize=512M; support was removed in 8.0
```

```
Java HotSpot(TM) 64-Bit Server VM warning: Using incremental CMS
is deprecated and will likely be removed in a future release
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option
MaxPermSize=512M; support was removed in 8.0
Beeline version 1.1.0-cdh5.12.1 by Apache Hive
beeline> !connect jdbc:hive2://cdh81-50:10000/default
scan complete in 1ms
Connecting to jdbc:hive2://cdh81-50:10000/default
Enter username for jdbc:hive2://cdh81-50:10000/default: yarn
Enter password for jdbc:hive2://cdh81-50:10000/default:
**********
Connected to: Apache Hive (version 1.1.0-cdh5.12.1)
Driver: Hive JDBC (version 1.1.0-cdh5.12.1)
Transaction isolation: TRANSACTION_REPEATABLE_READ
0: jdbc:hive2://cdh81-50:10000/default>
```

Kerberos认证：

```
[root@cdh81-50 ~]# kinit yarn
Password for yarn@master:
[root@cdh81-50 ~]# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: yarn@master

Valid starting       Expires              Service principal
12/08/2017 15:18:53  12/09/2017 15:18:53  krbtgt/master@master
        renew until 12/15/2017 15:18:53
[root@cdh81-50 ~]# beeline -u 'jdbc:hive2://cdh81-
50:10000/;principal=hive/cdh81-50@master'
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option
MaxPermSize=512M; support was removed in 8.0
Java HotSpot(TM) 64-Bit Server VM warning: Using incremental CMS
is deprecated and will likely be removed in a future release
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option
MaxPermSize=512M; support was removed in 8.0
scan complete in 2ms
```

```
Connecting to jdbc:hive2://cdh81-50:10000/;principal=hive/cdh81-50@master
Connected to: Apache Hive (version 1.1.0-cdh5.12.1)
Driver: Hive JDBC (version 1.1.0-cdh5.12.1)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.1.0-cdh5.12.1 by Apache Hive
0: jdbc:hive2://cdh81-50:10000/>
```

5.9 配置IMPALA集成LDAP

Impala中可以同时使用Kerberos+LDAP的认证方式，所以在已经启用Kerberos的情况下启用LDAP可以正常工作

在Impala配置页中：

- 启用 LDAP 身份验证选项设置为true
- 启用 LDAP TLS 选项设置为true
- LDAP URL 设置为ldap://cdh81-50
- LDAP BaseDN 设置为ou=people,dc=baofoo,dc=com

重启Impala服务

在chd81-51上执行，使用impala-shell测试LDAP账号：

```
[root@cdh81-51 ~]# impala-shell -l -u yarn --auth_creds_ok_in_clear
Starting Impala Shell using LDAP-based authentication
LDAP password for yarn:
Connected to cdh81-51:21000
Server version: impalad version 2.9.0-cdh5.12.1 RELEASE (build 5131a031f4aa38c1e50c430373c55ca53e0517b9)
*********************************************************************************
Welcome to the Impala shell.
(Impala Shell v2.9.0-cdh5.12.1 (5131a03) built on Thu Aug 24 09:27:32 PDT 2017)
To see more tips, run the TIP command.
*********************************************************************************
\nLDAP authentication is enabled, but the connection to Impala is not secured by TLS.
ALL PASSWORDS WILL BE SENT IN THE CLEAR TO IMPALA.
[cdh81-51:21000] >
```

使用beeline测试LDAP账号：

```
beeline -u "jdbc:hive2://cdh81-51:21050/default;" -n yarn -p 密码
```

5.10 配置HUE集成LDAP

在Hue中配置LDAP可以让Hue直接使用LDAP所管理的账号而不必在Hue中重新管理

在Hue的配置页面中修改

- 身份验证后端/backend为desktop.auth.backend.LdapBackend
- 登录时创建 LDAP 用户/create_users_on_login 设置为True
- 修改ldap_url=ldap://cdh81-50, ldap_username_pattern=uid=<username>,ou=people,dc=baofoo,dc=com
- 使用搜索绑定身份验证/search_bind_authentication 设置为False