2、部署Spark集群

Spark 分布式计算平台，主要承担实验引擎发送过来的计算任务，其中Worker实例可以横向扩展。

前置条件

需要使用Smartbi提供的Spark3.1安装包部署

数据挖掘数据量2000万以下时，无需单独部署spark节点，只需要提升数据挖掘服务器配置即可

文档环境

集群部署数据挖掘组件环境如下：

服务器IP	主机名	组件实例	部署目录
10.10.35.64	10-10-35-64	数据挖掘-1，Zookeeper-1，Python-1	/data
10.10.35.65	10-10-35-65	数据挖掘-2，**Spark-1**，Hadoop-1	/data
10.10.35.66	10-10-35-66	**Spark-2**，Zookeeper-2，Hadoop-2	/data
10.10.35.67	10-10-35-67	**Spark-3**，Zookeeper-3，Hadoop-3，Python-2	/data
10.10.204.250	10-10-204-250	Smartbi-Proxy	/data

1、系统环境准备

1.1 防火墙配置

为了便于安装，建议在安装前关闭防火墙。使用过程中，为了系统安全可以选择启用防火墙，但必须启用服务相关端口。

1.关闭防火墙

临时关闭防火墙（立即生效）

```
systemctl stop firewalld
```

永久关闭防火墙（重启后生效）

```
systemctl disable firewalld
```

查看防火墙状态

```
systemctl status firewalld
```

2.开启防火墙

相关服务及端口对照表：

服务名	需要开放端口
Spark	8080，8081，7077，[30000-65535]

如果确实需要打开防火墙安装，需要给防火墙放开以下需要使用到的端口
开启端口：8080，8081，7077，[30000-65535]

```
firewall-cmd --zone=public --add-port=8080/tcp --permanent
firewall-cmd --zone=public --add-port=8081/tcp --permanent
firewall-cmd --zone=public --add-port=7077/tcp --permanent
firewall-cmd --zone=public --add-port=30000-65535/tcp --permanent
```

配置完以后重新加载firewalld，使配置生效

```
firewall-cmd --reload
```

查看防火墙的配置信息

```
firewall-cmd --list-all
```

3.关闭selinux

临时关闭selinux，立即生效，不需要重启服务器。

```
setenforce 0
```

永久关闭selinux，修改完配置后需要重启服务器才能生效

```
sed -i 's/=enforcing/=disabled/g'  /etc/selinux/config
```

2、Spark集群安装

Spark集群节点说明

主机名	组件
10-10-35-65	Master，work-1
10-10-35-66	work-2
10-10-35-67	work-3

2.1 配置主机名映射

将数据挖掘组件中的服务器主机名映射到hosts文件中(所有节点均需执行此操作)

```
vi /etc/hosts
```

文件末尾添加(根据实际环境信息设置)：

```
10.10.35.64 10-10-35-64
10.10.35.65 10-10-35-65
10.10.35.66 10-10-35-66
10.10.35.67 10-10-35-67
```

2.2 配置系统免密登录

注意

Spark集群节点均需配置系统免密登陆

① 登陆服务器，生成密钥

```
ssh-keygen
```

输入ssh-keygen后，连续按三次回车，不用输入其它信息。

② 复制本机公钥到其它机器

假设当前的系统用户为root(注意，每台机器使用同一个用户来安装)，那命令如下:

```
ssh-copy-id -i ~/.ssh/id_rsa.pub root@10-10-35-65
ssh-copy-id -i ~/.ssh/id_rsa.pub root@10-10-35-66
ssh-copy-id -i ~/.ssh/id_rsa.pub root@10-10-35-67
```

测试是否设置成功

```
ssh root@10-10-35-65
ssh root@10-10-35-66
ssh root@10-10-35-67
```

如果不用输入密码，表示配置成功

2.3 安装JAVA环境

> **注意**
>
> Spark集群节点均需配置JAVA环境

解压jdk到指定目录：

```
tar -zxvf jdk-8u181-linux-x64.tar.gz -C /data
```

添加环境变量

```
vi /etc/profile
```

在文件末尾添加下面内容：

```
export JAVA_HOME=/data/jdk1.8.0_181
export JAVA_BIN=$JAVA_HOME/bin
export CLASSPATH=:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
export PATH=$PATH:$JAVA_BIN
```

让配置生效

```
source /etc/profile
```

验证安装

```
java -version
```

2.4 安装Spark

登陆10-10-35-65(Master)节点操作。

①解压Spark到指定目录

```
tar -zxvf spark-3.1.3-bin-hadoop3.2.tgz -C /data
```

②配置Spark从节点列表

```
cd /data/spark-3.1.3-bin-hadoop3.2/conf
cp workers.template workers
vi workers
```

把所有spark worker节点的机器名加到workers文件中，参考如下:

```
10-10-35-65
10-10-35-66
10-10-35-67
```

```
# A Spark Worker will be started on each of the machines listed below.
10-10-35-65
10-10-35-66
10-10-35-67
```

③将Spark安装包分发到Spark Work节点(10-10-35-66(work-1)、10-10-35-67(work-2))

假设当前的系统用户为root命令如下：

```
scp -r /data/spark-3.1.3-bin-hadoop3.2  root@10-10-35-66:/data/
scp -r /data/spark-3.1.3-bin-hadoop3.2  root@10-10-35-67:/data/
```

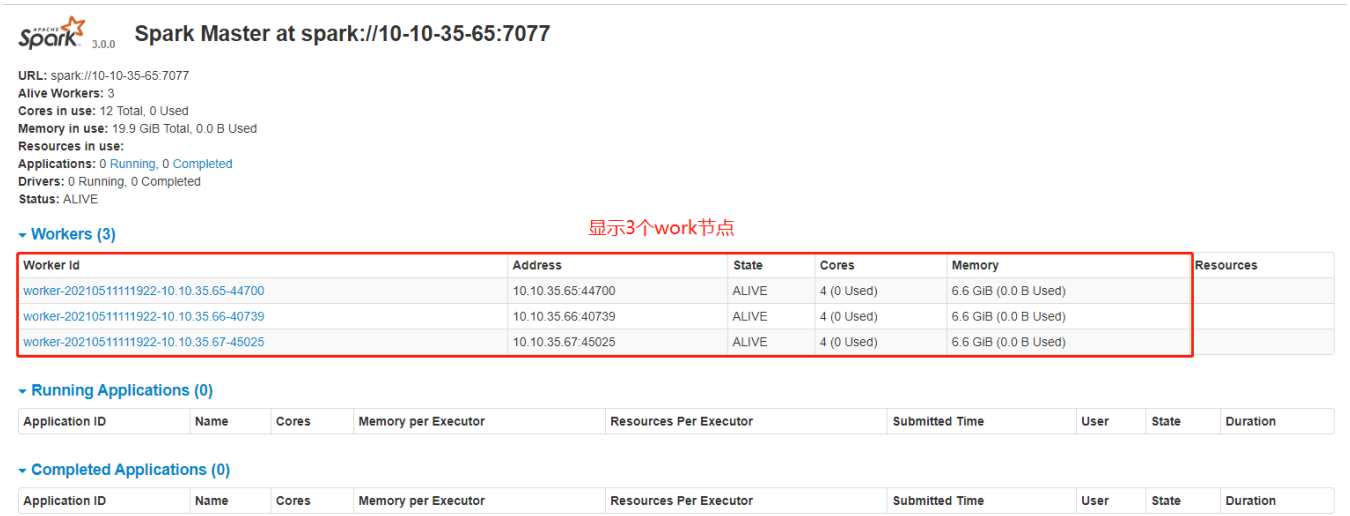
④在Spark Master节点(10-10-35-65(Master))启动Spark集群

```
cd /data/spark-3.1.3-bin-hadoop3.2/sbin
./start-all.sh
```

```
[root@10-10-35-65 sbin]# ./start-all.sh
starting org.apache.spark.deploy.master.Master, logging to /data/spark-3.0.0-bin-hadoop3.2/logs/spark-root-org.apache.spark.deploy.master.Master-1-10-10-35-65.out
10-10-35-65: starting org.apache.spark.deploy.worker.Worker, logging to /data/spark-3.0.0-bin-hadoop3.2/logs/spark-root-org.apache.spark.deploy.worker.Worker-1-10-10-3
5-65.out
10-10-35-67: starting org.apache.spark.deploy.worker.Worker, logging to /data/spark-3.0.0-bin-hadoop3.2/logs/spark-root-org.apache.spark.deploy.worker.Worker-1-10-10-3
5-67.out
10-10-35-66: starting org.apache.spark.deploy.worker.Worker, logging to /data/spark-3.0.0-bin-hadoop3.2/logs/spark-root-org.apache.spark.deploy.worker.Worker-1-10-10-3
5-66.out
[root@10-10-35-65 sbin]#
```

2.5 检查Spark

在浏览器中输入：http://master节点的IP:8080，查看集群状态



在spark节点提交任务测试进入/data/spark-3.1.3-bin-hadoop3.2/bin目录，执行以下命令(注意将”Spark-MasterIP”替换对应的IP或主机名)

```
./spark-submit --class org.apache.spark.examples.SparkPi --master spark://Spark-MasterIP:7077 /data/spark-3.1.3-bin-hadoop3.2/examples/jars/spark-examples_2.12-3.1.3.jar 100
```

```
21/05/11 11:23:43 INFO TaskSchedulerImpl: Killing all running tasks in stage 0: Stage finished
21/05/11 11:23:43 INFO DAGScheduler: Job 0 finished: reduce at SparkPi.scala:38, took 3.495777 s
Pi is roughly 3.1413727141372716
21/05/11 11:23:43 INFO SparkUI: Stopped Spark web UI at http://10-10-35-65:4040
21/05/11 11:23:43 INFO StandaloneSchedulerBackend: Shutting down all executors
21/05/11 11:23:43 INFO CoarseGrainedSchedulerBackend$DriverEndpoint: Asking each executor to shut down
21/05/11 11:23:43 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
21/05/11 11:23:43 INFO MemoryStore: MemoryStore cleared
21/05/11 11:23:43 INFO BlockManager: BlockManager stopped
21/05/11 11:23:43 INFO BlockManagerMaster: BlockManagerMaster stopped
21/05/11 11:23:43 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
21/05/11 11:23:43 INFO SparkContext: Successfully stopped SparkContext
21/05/11 11:23:43 INFO ShutdownHookManager: Shutdown hook called
21/05/11 11:23:43 INFO ShutdownHookManager: Deleting directory /tmp/spark-bffa7ed5-b5fc-4076-b0c9-3fa030040108
21/05/11 11:23:43 INFO ShutdownHookManager: Deleting directory /data/spark-3.0.0-bin-hadoop3.2/tmp/executor/spark-d813d90b-027e-48c7-885a-165b40245d0b
```

运行得出圆周率Pi的近似值3.14即部署成功。

2.6 运维操作

登陆10-10-35-65(Master)节点操作。

启动/停止spark服务

```
cd /data/spark-3.1.3-bin-hadoop3.2/sbin
./start-all.sh    #spark
./stop-all.sh     #spark
```

查看日志
Spark的日志路径：/data/spark-3.1.3-bin-hadoop3.2/logs
安装部署或者使用中有问题，可能需要根据日志来分析解决。