SPARK + HADOOP Cluster Install CentOS

* Packages:

1. Pre-build for Hadoop Spark spark-1.6.1-bin-hadoop2.6.tgz
2. HADOOP version according to spark pre-build package
3. JDK
4. Python
5. //SSH

* Clustersu

1. Master : 196.168.1.62
2. Slave :
3. Slave :

* Steps

1. SSH ( client + servic

# rpm -qa |grep ssh

#sudo ps -e |grep ssh ­­­\_\_\_\_\_查看是否安装了ssh

# systemctl status sshd.service ­­­\_\_\_\_\_查看ssh状态

/etc/ssh/sshd\_config \_\_­­\_\_\_config file

* port

# yum install ssh

# sudo apt-get install openssh-server \_\_\_\_\_\_安装ssh

# sudo service sshd start

/ stop / restart

1. SYSTEM SETTING
2. Environment variables

/etc/profile \_\_\_\_config file

:

${}

export JAVA\_HOME=/opt/jdk1.7.0\_79

export JRE\_HOME=${JAVA\_HOME}/jre

export PYTHON\_HOME=/opt/Python-3.4.4

export HADOOP\_HOME=/opt/hadoop-2.6.4

export SPARK\_HOME=/opt/spark-1.6.1-bin-hadoop2.6

export CLASSPATH=.:${JAVA\_HOME}/lib:${JRE\_HOME}/lib

exportPATH=${JAVA\_HOME}/bin:${PYTHON\_HOME}/bin:${SPARK\_HOME}/bin:

${HADOOP\_HOME}/bin:$PATH

# source /etc/profile ­\_\_\_\_\_\_编译修改后文件

1. Static IP address 固定IP地址

/etc/sysconfig/network-scripts/ifcfg-eno16777736

BOOTPROTO="static"

ONBOOT="yes"

IPADDR0="192.168.1.7"

PREFIX0="255.255.255.0"

GATEWAY0="192.168.1.1"

DNS1="61.147.37.1"

DNS2="101.226.4.6"

1. Host

/etc/hosts

192.168.1.7 master

192.168.1.8 slave1

192.168.1.9 slave2

# hostnamectl set-hostname master

1. Hadoop Setting

*Dicts： /opt/hadoop-2.6.4/etc/hadoop/*

* **hadoop-env.sh / yarn-env.sh**

export JAVA\_HOME=/opt/jdk1.7.0\_79

export HADOOP\_ROOT\_LOGGER=DEBUG,console

* **slaves (增加slaves节点)**

slave1

slaves2

* **core-site.xml**

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://master:9000</value>

</property>

<property>

<name>io.file.buffer.size</name>

<value>131702</value>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>file:/opt/hadoop-2.6.4/tmp</value>

</property>

<!--

<property>

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

<value>\*</value>

</property>

<property>

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

<value>\*</value>

</property>

<property>

<name>hadoop.native.lib</name>

<value>false</value>

</property>

-->

</configuration>

* **hdfs-site.xml**

<configuration>

<property>

<name>dfs.namenode.name.dir</name>

<value>file:/opt/hadoop-2.6.4/dfs/name</value>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value>file:/opt/hadoop-2.6.4/dfs/data</value>

</property>

<property>

<name>dfs.replication</name>

<value>3</value>

</property>

<property>

<name>dfs.namenode.secondary.http-address</name>

<value>master:9001</value>

</property>

<property>

<name>dfs.webhdfs.enabled</name>

<value>true</value>

</property>

</configuration>

* **mapred-site.xml (cp from template)**

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

<property>

<name>mapreduce.jobhistory.address</name>

<value>master:10020</value>

</property>

<property>

<name>mapreduce.jobhistory.webapp.address</name>

<value>master:19888</value>

</property>

</configuration>

* **yarn-site.xml**

<configuration>

<!-- Site specific YARN configuration properties -->

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.resourcemanager.hostname</name>

<value>master</value>

</property>

<property>

<name>yarn.resourcemanager.address</name>

<value>master:8032</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>master:8030</value>

</property>

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>master:8031</value>

</property>

<property>

<name>yarn.resourcemanager.admin.address</name>

<value>master:8033</value>

</property>

<property>

<name>yarn.resourcemanager.webapp.address</name>

<value>master:8088</value>

</property>

<!--

<property>

<name>yarn.nodemanager.resource.memory-mb</name>

<value>3000</value>

</property>

-->

</configuration>

将配置好的hadoop 文件cp到 slaves机器上，配置路径与master一样

格式化namenode

$HADOOP\_HOME/bin/hadoop namenode –format

只需要在master上执行一次就行

10、启动hdfs

$HADOOP\_HOME/sbin/start-dfs.sh

$HADOOP\_HOME/sbin/start-yarn.sh

11、检查启动情况

http://192.168.1.7:8088

<http://192.168.1.7:50070>

启动后，jps 查看进程，通常有：

NameNode

ResourceManager

DataNode

Jps

NodeManager

SecondaryNameNode

1. SPARK SETTING

Dicts: ./conf

* spark-env.template

# cp spark-env.template speak-env.sh

export SPARK\_MASTER\_IP=192.168.1.17

export MASTER=SPARK://master:7077

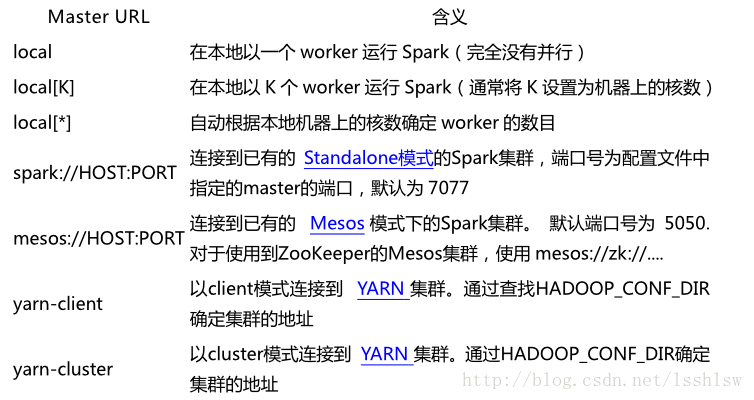
export SPARK\_HOME=/opt/spark-1.6.1-bin-hadoop2.6

* slaves.template

# cp slaves.template slaves (添加节点)

Master

Slaves



#./bin/pyspark --master local[2] \_\_\_\_\_ shell by local

#./bin/spark-submit --master local[2] examples/src/main/python/pi.py 10 \_\_\_<class> [params]

http://my.oschina.net/u/1428349/blog/313646?fromerr=Z1L4NtS4#OSC\_h2\_1