<https://blog.51cto.com/u_15346267/3668927>

[spack安装使用\_yr\_linux的技术博客\_51CTO博客\_spack](https://blog.51cto.com/u_13323775/2061199)

名词解释

* Driver

该进程调用 Spark 程序的 main 方法, 并且启动 SparkContext

* Cluster Manager

该进程负责和外部集群工具打交道, 申请或释放集群资源

* Worker

该进程是一个守护进程, 负责启动和管理 Executor

* Executor

该进程是一个JVM虚拟机, 负责运行 Spark Task

总结

* Master 负责总控, 调度, 管理和协调 Worker, 保留资源状况等
* Slave 对应 Worker 节点, 用于启动 Executor 执行 Tasks, 定期向 Master汇报
* Driver 运行在 Client 或者 Slave(Worker) 中, 默认运行在 Slave(Worker) 中

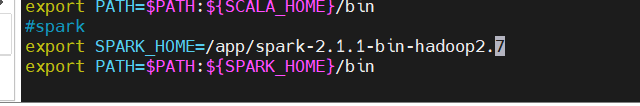
解压：tar -zxvf spark-2.1.1-bin-hadoop2.7

环境变量：

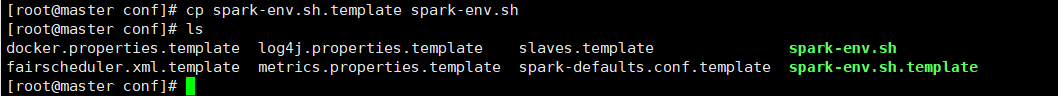
#spark

export SPARK\_HOME=/app/spark-2.1.1-bin-hadoop2.7

export PATH=$PATH:${SPARK\_HOME}/bin



cp spark-env.sh.template spark-env.sh



1）vim spark-env.sh 指定运行参数

# 指定 Java Home

export JAVA\_HOME=/usr/local/src/jdk1.8.0\_221

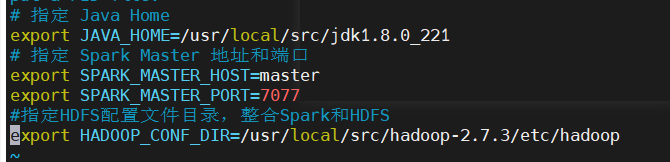
# 指定 Spark Master 地址和端口

export SPARK\_MASTER\_HOST=master

export SPARK\_MASTER\_PORT=7077

#指定HDFS配置文件目录，整合Spark和HDFS

export HADOOP\_CONF\_DIR=/usr/local/src/hadoop-2.7.3/etc/hadoop



export JAVA\_HOME=/usr/local/jdk

export SCALA\_HOME=/data/scala-2.12.4

export SPARK\_WORKER\_MEMORY=1g

export HADOOP\_CONF\_DIR=/data/hadoop/etc/hadoop/

2）修改配置文件 slaves, 以指定从节点为止, 从在使用 sbin/start-all.sh 启动集群的时候, 可以一键启动整个集群所有的 Worker

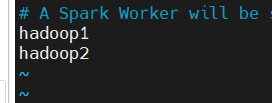
* 进入配置目录, 并复制一份新的配置文件, 以供在此基础之上进行修改

登录后复制

cp slaves.template slaves

vim slaves

配置所有从节点的地址



3.配置 HistoryServer

1.默认情况下, Spark 程序运行完毕后, 就无法再查看运行记录的 Web UI 了, 通过 HistoryServer 可以提供一个服务, 通过读取日志文件, 使得我们可以在程序运行结束后, 依然能够查看运行过程

2.复制 spark-defaults.conf, 以供修改

cp spark-defaults.conf.template spark-defaults.conf

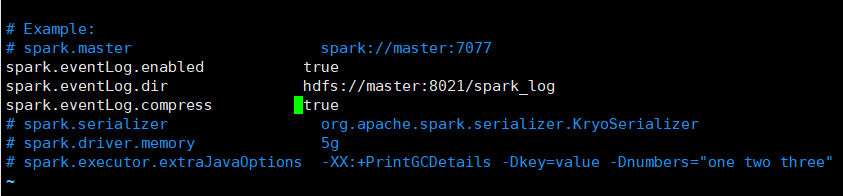
vim spark-defaults.conf

3.将以下内容复制到`spark-defaults.conf`末尾处, 通过这段配置, 可以指定 Spark 将日志输入到 HDFS 中

spark.eventLog.enabled true

spark.eventLog.dir hdfs://master:8021/spark\_log

spark.eventLog.compress true



4.vim spark-env.sh(配置 HistoryServer 启动参数, 使得 HistoryServer 在启动的时候读取 HDFS 中写入的 Spark 日志

# 指定 Spark History 运行参数

export SPARK\_HISTORY\_OPTS="-Dspark.history.ui.port=4000 -Dspark.history.retainedApplications=3 -Dspark.history.fs.logDirectory=hdfs://master:8021/spark\_log"

5.为 Spark 创建 HDFS 中的日志目录

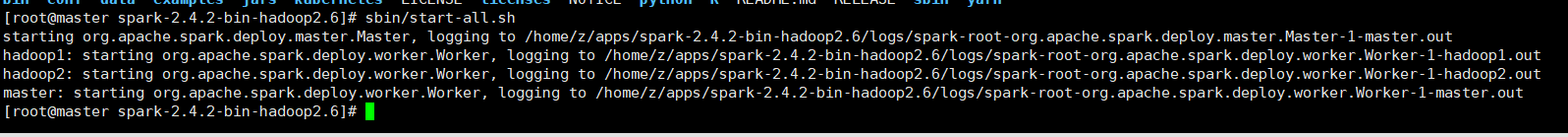
hdfs dfs -mkdir -p /spark\_log

将 Spark 安装包分发给集群中其它机器



6.启动 Spark Master 和 Slaves, 以及 HistoryServer

sbin/start-all.sh



sbin/start-history-server.sh

[Spark Master at spark://master:7077](http://192.168.120.50:8080/)

**2.3. Spark 集群高可用搭建**

sbin/stop-all.sh

进入 spark-env.sh 所在目录, 打开 vi 编辑

vim spark-env.sh

export JAVA\_HOME=/home/z/apps/jdk1.8.0\_221

# 指定 Spark Master 地址

#export SPARK\_MASTER\_HOST=master

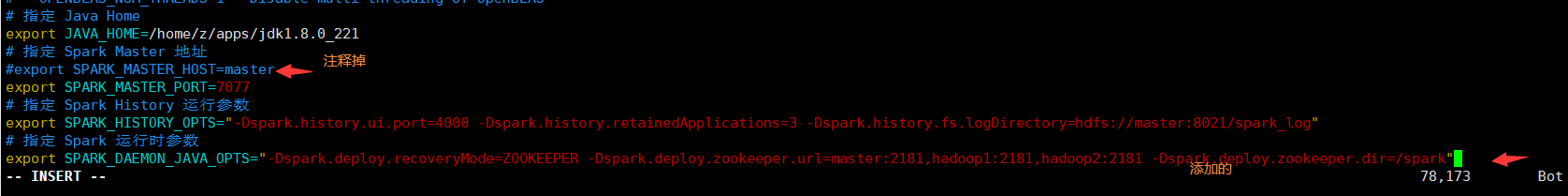
export SPARK\_MASTER\_PORT=7077

# 指定 Spark History 运行参数

export SPARK\_HISTORY\_OPTS="-Dspark.history.ui.port=4000 -Dspark.history.retainedApplications=3 -Dspark.history.fs.logDirectory=hdfs://master:8021/spark\_log"

# 指定 Spark 运行时参数

export SPARK\_DAEMON\_JAVA\_OPTS="-Dspark.deploy.recoveryMode=ZOOKEEPER -Dspark.deploy.zookeeper.url=master:2181,hadoop1:2181,hadoop2:2181 -Dspark.deploy.zookeeper.dir=/spark"



分发配置文件到整个集群:

scp spark-env.sh hadoop1:$PWD

scp spark-env.sh hadoop2:$PWD

1. 在 node01 上启动整个集群

sbin/start-all.sh

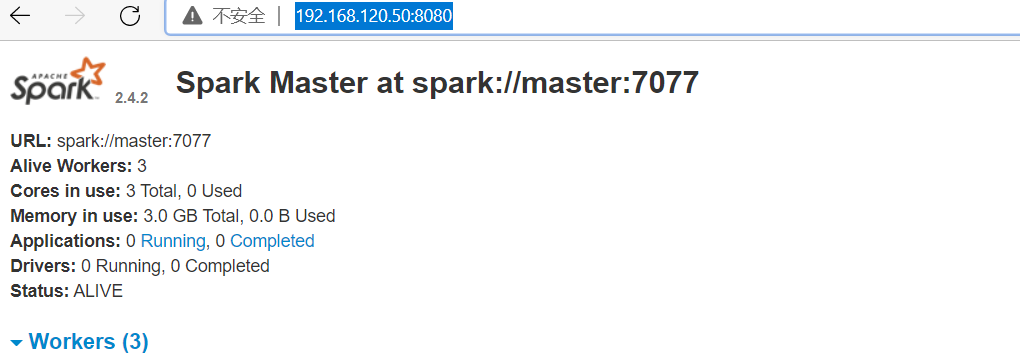
sbin/start-history-server.sh

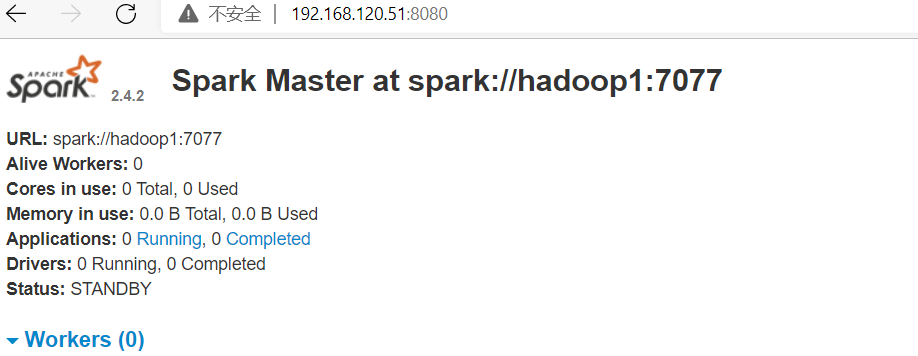
1. 在 node02 上单独再启动一个 Master

sbin/start-master.sh

查看 node01 master 和 node02 master 的 WebUI

1. 你会发现一个是 ALIVE(主), 另外一个是 STANDBY(备)





如果关闭一个, 则另外一个成为`ALIVE`, 但是这个过程可能要持续两分钟左右