尚硅谷大数据技术之Hadoop（入门）

(作者：尚硅谷大数据研发部)

版本：V3.0

# 第1章 完全分布式运行模式（开发重点）

分析：

1）准备3台客户机（关闭防火墙、静态IP、主机名称）

2）安装JDK

3）配置环境变量

4）安装Hadoop

5）配置环境变量

**6）配置集群**

**7）单点启动**

**8）配置ssh**

**9）群起并测试集群**

## 1.1 Hadoop部署

**1）集群部署规划**

注意：NameNode和SecondaryNameNode不要安装在同一台服务器

注意：ResourceManager也很消耗内存，不要和NameNode、SecondaryNameNode配置在同一台机器上。

|  |  |  |  |
| --- | --- | --- | --- |
|  | hadoop102 | hadoop103 | hadoop104 |
| HDFS | NameNode  DataNode | DataNode | SecondaryNameNode  DataNode |
| YARN | NodeManager | ResourceManager  NodeManager | NodeManager |

**2）用SecureCRT工具将hadoop-3.1.3.tar.gz导入到opt目录下面的software文件夹下面**

切换到sftp连接页面，选择Linux下编译的hadoop jar包拖入，如图2-32所示

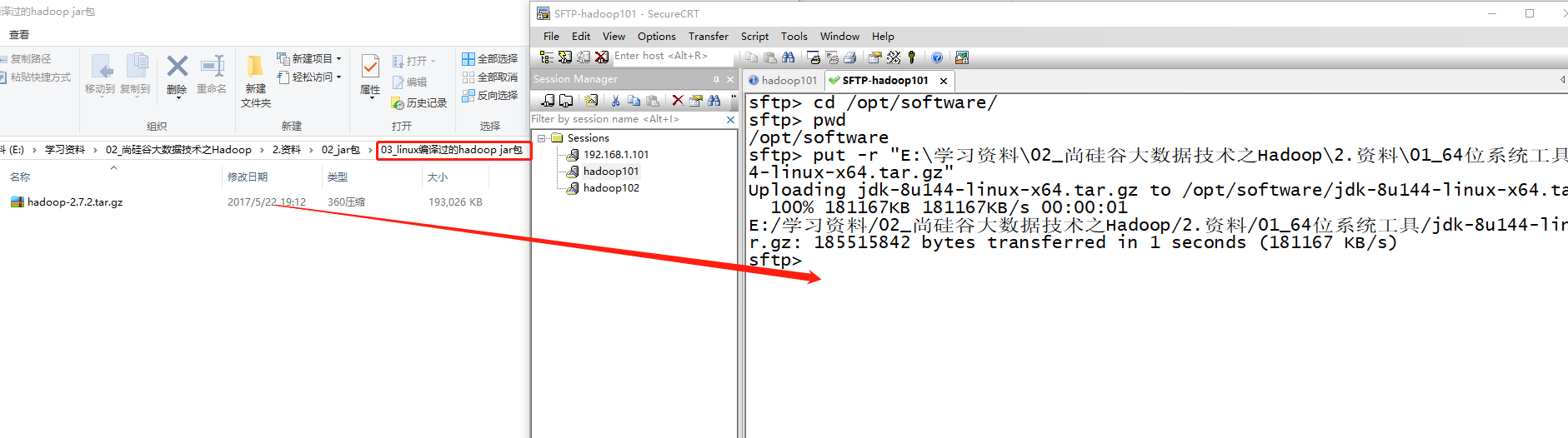


图 拖入hadoop的tar包

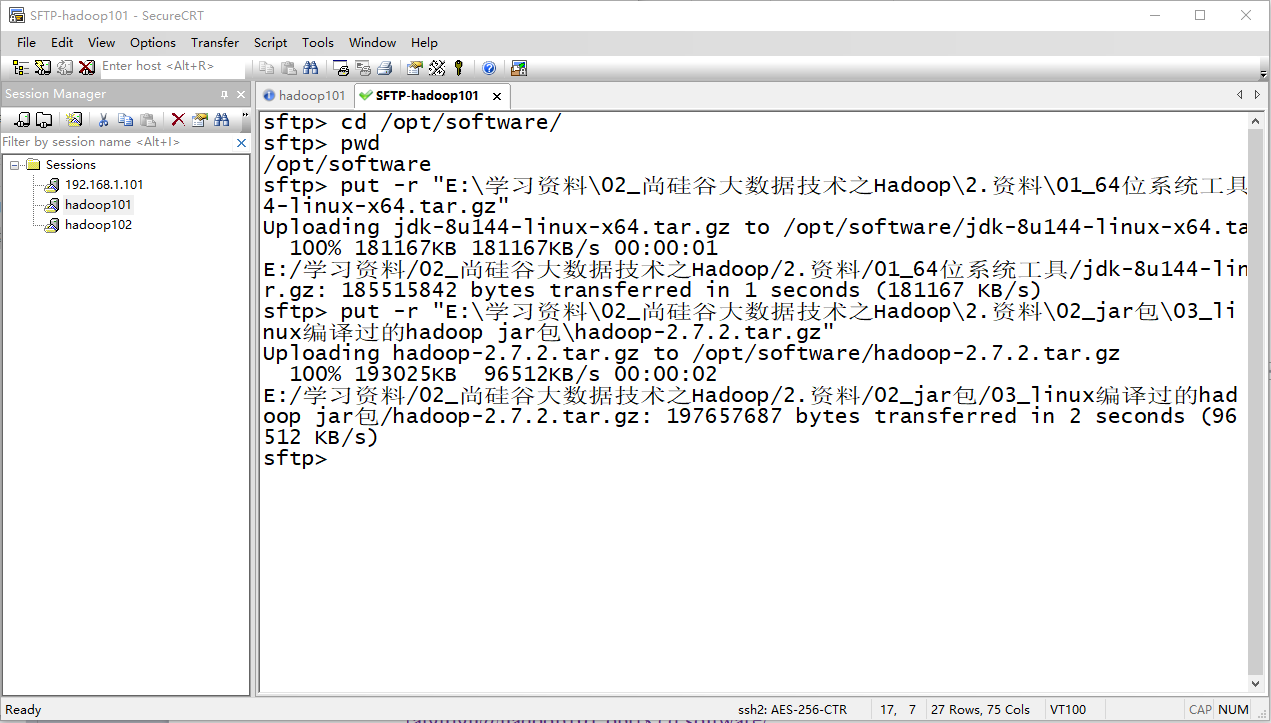


图2-33 拖入Hadoop的tar包成功

**3）进入到Hadoop安装包路径下**

[atguigu@hadoop102 ~]$ cd /opt/software/

**4）解压安装文件到/opt/module下面**

[atguigu@hadoop102 software]$ tar -zxvf hadoop-3.1.3.tar.gz -C /opt/module/

**5）查看是否解压成功**

[atguigu@hadoop102 software]$ ls /opt/module/hadoop-3.1.3

**6）重命名**

[atguigu@hadoop102 software]$ mv /opt/module/hadoop-3.1.3 /opt/module/hadoop

**7）将Hadoop添加到环境变量**

（1）获取Hadoop安装路径

[atguigu@hadoop102 hadoop]$ pwd

/opt/module/hadoop

（2）打开/etc/profile.d/my\_env.sh文件

[atguigu@hadoop102 hadoop]$ sudo vim /etc/profile.d/my\_env.sh

在profile文件末尾添加JDK路径：（shitf+g）

#HADOOP\_HOME

export HADOOP\_HOME=/opt/module/hadoop

export PATH=$PATH:$HADOOP\_HOME/bin

export PATH=$PATH:$HADOOP\_HOME/sbin

（3）保存后退出

:wq

（4）分发环境变量文件

[atguigu@hadoop102 hadoop]$ sudo /home/atguigu/bin/xsync /etc/profile.d/my\_env.sh

（5）source 是之生效（3台节点）

[atguigu@hadoop102 module]$ source /etc/profile.d/my\_env.sh

[atguigu@hadoop103 module]$ source /etc/profile.d/my\_env.sh

[atguigu@hadoop104 module]$ source /etc/profile.d/my\_env.sh

## 1.2 配置集群

1）核心配置文件

配置core-site.xml

[atguigu@hadoop102 ~]$ cd $HADOOP\_HOME/etc/hadoop

[atguigu@hadoop102 hadoop]$ vim core-site.xml

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- 指定NameNode的地址 -->

<property>

<name>fs.defaultFS</name>

<value>hdfs://hadoop102:8020</value>

</property>

<!-- 指定hadoop数据的存储目录 -->

<property>

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

<value>/opt/module/hadoop/data</value>

</property>

<!-- 配置HDFS网页登录使用的静态用户为atguigu -->

<property>

<name>hadoop.http.staticuser.user</name>

<value>atguigu</value>

</property>

<!-- 配置该atguigu(superUser)允许通过代理访问的主机节点 -->

<property>

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

<value>\*</value>

</property>

<!-- 配置该atguigu(superUser)允许通过代理用户所属组 -->

<property>

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

<value>\*</value>

</property>

<!-- 配置该atguigu(superUser)允许通过代理的用户-->

<property>

<name>hadoop.proxyuser.atguigu.users</name>

<value>\*</value>

</property>

</configuration>

2）HDFS配置文件

配置hdfs-site.xml

[atguigu@hadoop102 hadoop]$ vim hdfs-site.xml

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- nn web端访问地址-->

<property>

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

<value>hadoop102:9870</value>

</property>

<!-- 2nn web端访问地址-->

<property>

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

<value>hadoop104:9868</value>

</property>

<!-- 测试环境指定HDFS副本的数量1 -->

<property>

<name>dfs.replication</name>

<value>3</value>

</property>

</configuration>

3）YARN配置文件

配置yarn-site.xml

[atguigu@hadoop102 hadoop]$ vim yarn-site.xml

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- 指定MR走shuffle -->

<property>

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

<value>mapreduce\_shuffle</value>

</property>

<!-- 指定ResourceManager的地址-->

<property>

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

<value>hadoop103</value>

</property>

<!-- 环境变量的继承 -->

<property>

<name>yarn.nodemanager.env-whitelist</name>

<value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>

</property>

<!--yarn单个容器允许分配的最大最小内存 -->

<property>

<name>yarn.scheduler.minimum-allocation-mb</name>

<value>512</value>

</property>

<property>

<name>yarn.scheduler.maximum-allocation-mb</name>

<value>4096</value>

</property>

<!-- yarn容器允许管理的物理内存大小 -->

<property>

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

<value>4096</value>

</property>

<!-- 关闭yarn对物理内存和虚拟内存的限制检查 -->

<property>

<name>yarn.nodemanager.pmem-check-enabled</name>

<value>true</value>

</property>

<property>

<name>yarn.nodemanager.vmem-check-enabled</name>

<value>false</value>

</property>

</configuration>

4）MapReduce配置文件

配置mapred-site.xml

[atguigu@hadoop102 hadoop]$ vim mapred-site.xml

文件内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<configuration>

<!-- 指定MapReduce程序运行在Yarn上 -->

<property>

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

<value>yarn</value>

</property>

</configuration>

5）配置workers

[atguigu@hadoop102 hadoop]$ vim /opt/module/hadoop/etc/hadoop/workers

在该文件中增加如下内容：

hadoop102

hadoop103

hadoop104

注意：该文件中添加的内容结尾不允许有空格，文件中不允许有空行。

## 1.3 配置历史服务器

为了查看程序的历史运行情况，需要配置一下历史服务器。具体配置步骤如下：

**1）配置mapred-site.xml**

[atguigu@hadoop102 hadoop]$vim mapred-site.xml

在该文件里面增加如下配置。

<!-- 历史服务器端地址 -->

<property>

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

<value>hadoop102:10020</value>

</property>

<!-- 历史服务器web端地址 -->

<property>

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

<value>hadoop102:19888</value>

</property>

## 1.4 配置日志的聚集

日志聚集概念：应用运行完成以后，将程序运行日志信息上传到HDFS系统上。

日志聚集功能好处：可以方便的查看到程序运行详情，方便开发调试。

注意：开启日志聚集功能，需要重新启动NodeManager 、ResourceManager和HistoryManager。

开启日志聚集功能具体步骤如下：

**1）配置yarn-site.xml**

[atguigu@hadoop102 hadoop]$ vim yarn-site.xml

在该文件里面增加如下配置。

<!-- 开启日志聚集功能 -->

<property>

<name>yarn.log-aggregation-enable</name>

<value>true</value>

</property>

<!-- 设置日志聚集服务器地址 -->

<property>

<name>yarn.log.server.url</name>

<value>http://hadoop102:19888/jobhistory/logs</value>

</property>

<!-- 设置日志保留时间为7天 -->

<property>

<name>yarn.log-aggregation.retain-seconds</name>

<value>604800</value>

</property>

## 1.5 分发Hadoop

[atguigu@hadoop102 hadoop]$ xsync /opt/module/hadoop/

## 1.6 群起集群

**1）启动集群**

（1）**如果集群是第一次启动**，需要在hadoop102节点格式化NameNode（注意格式化之前，一定要先停止上次启动的所有namenode和datanode进程，然后再删除data和log数据）

[atguigu@hadoop102 hadoop]$ bin/hdfs namenode -format

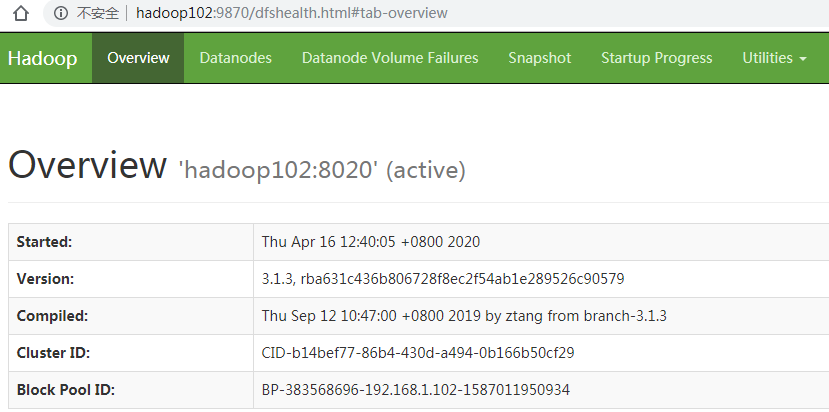
（2）启动HDFS

[atguigu@hadoop102 hadoop]$ sbin/start-dfs.sh

（3）**在配置了ResourceManager的节点（hadoop103）**启动YARN

[atguigu@hadoop103 hadoop]$ sbin/start-yarn.sh

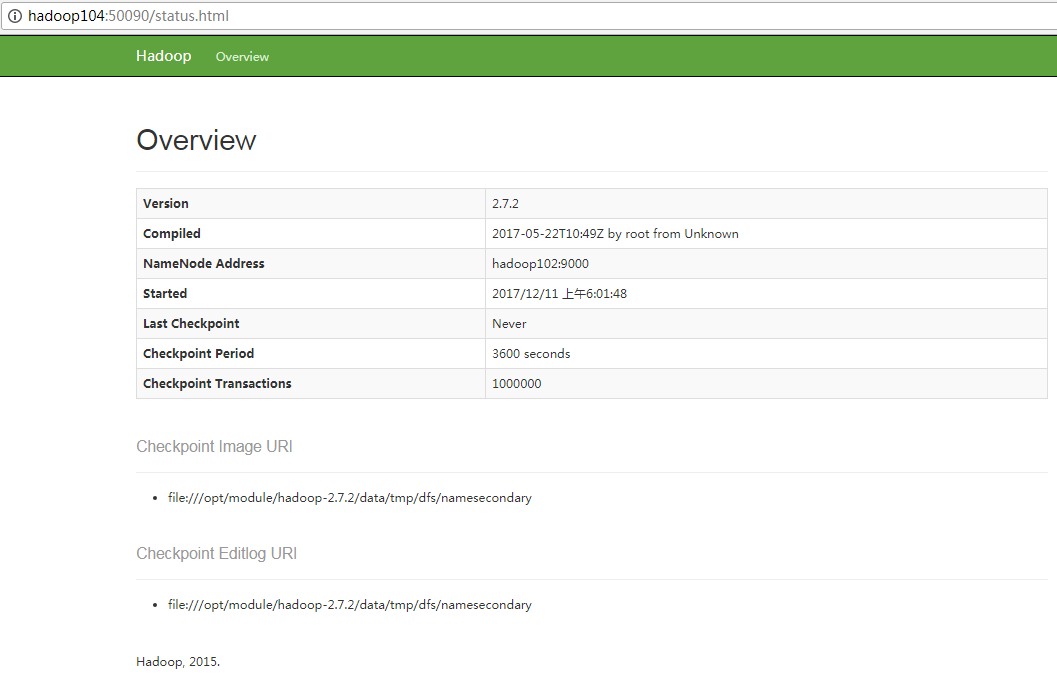
（4）Web端查看HDFS的Web页面：<http://hadoop102:9870/>



（5）Web端查看SecondaryNameNode

（a）浏览器中输入：<http://hadoop104:9868/status.html>

（b）查看SecondaryNameNode信息



## 1.7 Hadoop群起脚本

[atguigu@hadoop102 bin]$ pwd

/home/atguigu/bin

[atguigu@hadoop102 bin]$ vim hdp.sh

输入如下内容：

#!/bin/bash

if [ $# -lt 1 ]

then

echo "No Args Input..."

exit ;

fi

case $1 in

"start")

echo " =================== 启动 hadoop集群 ==================="

echo " --------------- 启动 hdfs ---------------"

ssh hadoop102 "/opt/module/hadoop/sbin/start-dfs.sh"

echo " --------------- 启动 yarn ---------------"

ssh hadoop103 "/opt/module/hadoop/sbin/start-yarn.sh"

echo " --------------- 启动 historyserver ---------------"

ssh hadoop102 "/opt/module/hadoop/bin/mapred --daemon start historyserver"

;;

"stop")

echo " =================== 关闭 hadoop集群 ==================="

echo " --------------- 关闭 historyserver ---------------"

ssh hadoop102 "/opt/module/hadoop/bin/mapred --daemon stop historyserver"

echo " --------------- 关闭 yarn ---------------"

ssh hadoop103 "/opt/module/hadoop/sbin/stop-yarn.sh"

echo " --------------- 关闭 hdfs ---------------"

ssh hadoop102 "/opt/module/hadoop/sbin/stop-dfs.sh"

;;

\*)

echo "Input Args Error..."

;;

esac

[atguigu@hadoop102 bin]$ chmod 777 hdp.sh