

# RHadoop 云计算集群服务器架构指南

## 一、认识 Rhadoop——了解概况

RHadoop 是运行 R 语言的 Hadoop 分布式计算平台的简称。要认识 Rhadoop 首先我们分别来认识 R 语言和 Hadoop。

Hadoop 已经成名好多年了，它是 Apache 软件基金会旗下的一个 JAVA 开源分布式计算平台，现已被各大互联网企业（包括 Yahoo 和 Facebook 等）用于大规模数据分布式存储与分布式计算。近年来随着云计算、大数据处理、数据挖掘等概念和应用越来越火，Hadoop 更是名声大噪，各大企业对熟悉 Hadoop 体系架构和性能优化的人才需求也相当旺盛，人才缺口也相对加大。

R 语言可能对大多数人来说比较陌生，但是做数理统计、数据分析的人不会不知道 R。R 是一门主要运用于统计分析与绘图的编程语言和操作工具。和 Matlab 类似，R 是一门高级脚本化编程语言，直接在命令行输入指令即可运行，无需编译链接。它可以进行高效的向量化运算，效率远高于传统的循环运算。另外，R 是开源的，有大批杰出的工程师和程序员为 R 开发了许多好用的函数库和图形化工具，尤其在数理统计方面有大量成熟的开源的程序包。所以 R 是做数据挖掘、数据分析的廉价高效的方案。

所以如果能 R 运行在 Hadoop 平台上可以说是如虎添翼，可以让数据分析人员，即能用到 R 丰富的程序包，又能借助 Hadoop 分布式运算、分布式存储的优势做海量数据的处理。

现在，这个如果已经可以成为结果。

## 二、走近 Rhadoop——安装并运行

安装配置这样的集群服务器就像搭积木，须要一步一步按部就班，可能会有些繁琐，但是在安装之前如果先理清一下思路，并且学习一下 Hadoop 的一些基本概念，会让安装过程显得更仅仅有条。安装过程我们分两大步来做：1、安装 Java 环境、R 环境、ssh 等各种必需品；2、安装并配置 Hadoop 和 Rhadoop 程序包。如果各种“必需品”恰好已经在服务器上安装过下文所述的相应版本，那么就可以直接进行第二步了。

### 必需品 1：JRE6、JDK6 版本

首先要知道 Hadoop 是用 JAVA 编写的一款分布式计算平台——用 JAVA 写，用 JAVA 编程开发。所以肯定要为服务器网络中的所有服务器安装上 JRE 和 JDK，然后为每台服务器安装上 Hadoop 平台且配置参数要相同。（服务器最好是 Linux 操作系统，Hadoop 在 Linux 上较为稳定。这里我仅以 Ubuntu Linux 为例。）

JRE,JDK 版本需要是 1.6 的。1.7 版本和有些 Linux 系统自带的 OpenJDK 都会不兼容。所以要先删除 OpenJDK，然后可以到 Oracle 官网上下载所需版本的 JDK，也可以用 apt-get 直接用命令行安装。

```
#删除 openjdk
sudo apt-get purge openjdk*
```

```
#安装 java6
```

```
sudo apt-get install sun-java6-jre sun-java6-jdk sun-java6-plugin
```

安装完需要设置 JAVA 环境变量，在/etc/environment 文件中加入以下两行即可：

```
JAVA_HOME=/usr/lib/jvm/java-6-sun
```

```
CLASSPATH=./usr/lib/jvm/java-6-sun/lib
```

装完后检查一下 java 命令是不是已经可以用了。

## 必需品 2: ssh、openssh-sever、密钥登录

像所有组织都有管理者一样，服务器集群也有主节点和从节点，Hadoop 中通常由主节点上的服务器控制启动和停用其他从节点上的 Hadoop，这种启停服务是通过 ssh 来实现的。这很自然，因为如果你想启动远程服务器的某项服务，最朴素的方法就是通过 ssh 用某账户登录它然后启动某项服务。然后你又不想每次登录都要输入密码，所以你会特别希望无密码地用密钥登录。所以要想实现主节点服务器可以“畅通无阻”地启停 Hadoop 服务器集群，就须要使其可以无密码地用 ssh 密钥登录其他从节点服务器。我们需要在每个服务器上都建立一个 Hadoop 用户（这主要是为了方便管理，这个 Hadoop 用户将来有处理 Hadoop 平台的权限），然后要求主节点服务器的 Hadoop 用户生成一个 ssh 私钥公钥对，然后把该公钥复制给其他从节点服务器的 Hadoop 用户的 ~/.ssh 就可以了。具体过程没有任何特殊之处，所以不再赘述。

## 必需品 3: R 语言 2.15 版本

要用 R 的 Hadoop 平台，自然要装 R 了，在 R 环境下使用 Hadoop 的思路可以简单理解为用 R 语言方便的调用 Hadoop 平台提供的 API，幸运地是已经有人为 R 开发了这样的程序包，只要为 R 装上这些程序包就可以了。但是这仅在 R2.15 版本中能够完美支持。其实主要是由于这些程序包出现的较晚，甚至 R 的官方程序包下载源中还没有。至于具体是哪些包，和他们的作用和意义，下文再详细描述。

如果以上三个必需品已经具备，那么现在就可以开始装 RHadoop 平台了。安装 Hadoop 很容易，但配置好 Hadoop、协调好 Hadoop 是须要学习的。在 apache 官网提供的镜像 <http://www.apache.org/dyn/closer.cgi/hadoop/common/> 可以下载到各个版本的 Hadoop 的压缩包，直接解压到本地目录即可。在这里不建议下载最新版本的，目前推荐 hadoop-0.20.x 比较稳定。其实对于所有开发用的开源软件平台都不建议用最新版的，最新版都是服务商用来试错的，稳定性很难保证。

下载完成后可以用 tar 命令解压成一个名为 hadoop 文件夹到任何路径，现在假设解压到/usr/local 这个路径：

```
cd /usr/local
```

```
#解压
```

```
usrsudo tar -zxf hadoop 压缩包名.tar.gz
```

```
#改名
```

```
sudo mv 解压后的 hadoop 文件夹名 hadoop
```

为了方便管理，让所有机器上的 hadoop 用户拥有 hadoop 安装目录的管理权：

```
chown -R hadoop /usr/local/hadoop
```

然后需要设置 hadoop 中本机 JDK 的路径，在 hadoop/conf/hadoop-env.sh 后加上：

```
export JAVA_HOME=/usr/lib/jvm/java-6-sun
```

到这里安装的工作可以说画上了一个逗号，接下来是最重要的配置阶段。在配置之前需要了解 Hadoop 的基本结构，不然等于瞎配置，运行起来如果出错也不看不懂提示信息是什么意思。

Hadoop 有三个最主要的主从关系：Masters 与 Slaves(主节点与从节点服务器)、Namenode 与 Datanode(分布式存储的主从)和 Jobtracker 与 Tasktracker（分布式运算的主从），一个重要的文件系统 hdfs(hadoop distributed file system)，它们可以实现一个容灾性较高的服务器集群。

假如用户上传了一个 1TB 的大文件给服务器，Hadoop 集群的 Namenode（通常就是作为 Master 的服务器）会以默认每片 64MB 分片给各个 DataNode(通常就是作为 Slaves 的服务器),而且每个分片默认有 3 个备份分开存储，所以任何一个服务器可能都没有完整的 1TB 文件而任何一个服务器宕机也不会导致文件的损坏，事实上，在每个分片 3 个备份的默认情况，任何两个服务器宕机也不会导致文件损坏。

Jobtracker 和 Tasktracker 的关系相当于工头和工人的关系，用户提交的程序会经由 Jobtracker 分配给 Tasktracker 并行运算。

以上提到的两种任务分配都通过 TCP/IP 网络通讯来进行，所以对网络要求较高，一般千兆网络较为理想。用户提交的文件和任务都将提交给 Master 服务器的特定端口，即 Jobtracker 和 Namenode 的网络端口,这需要我们在配置文件中设置。

在 conf/core-site.xml 文件中设置 Namenode 的端口号，在<configuration>标签中加入以下属性：

```
<property>
<name>fs.default.name</name>
<value>hdfs://master:9000</value>
<!--其中 Master 是主节点机器的 ip-->
<name>hadoop.tmp.dir</name>
<value>/usr/local/hadoop/tmp</value>
<!--这个目录即 hdfs 文件系统在每台服务器上分布式存储的目录,
需事先创建-->
</property>
```

在 conf/mapred-site.xml 文件中设置 Jobtracker 的端口号，在<configuration>标签中加入以下属性：

```
<property>
<name>mapred.job.tracker</name>
```

```
<value>master:9001</value>
</property>
```

在 `conf/hdfs-site.xml` 文件中设置文件分片的备份数，在 `<configuration>` 标签中设置以下属性：

```
<configuration>
<property>
<name>dfs.replication</name>
<value>3</value>
</property>
```

在 `conf/masters` 中写入主节点服务器的 ip 地址（一行一个），在 `conf/slaves` 中写入从节点服务器的 ip 地址（一行一个）即可。

至此，单台服务器配置基本完成。把配置完的这个 `hadoop` 文件夹复制到集群中的所有服务的相同目录下，即可实现所有服务器相同的配置，可以使用 `scp` 命令进行复制。

最后我们可以测试运行 `hadoop` 服务器集群了，首次运行时需要格式化 `hdfs` 文件系统：

```
bin/hadoop namenode -format
```

然后启动所有服务

```
bin/start-all.sh
```

用 `java` 的 `jps` 命令，即查看现在运行的 `java` 进程如果在主节点服务器上看到 `namenode`, `datanode`, `jobtracker`, `tasktracker`, `secondarynamenode` 和 `jps` 六个进程那么说明服务都成功启动了。另外，还 Hadoop 提供了两个用 `jsp` 写的监控网页。通过 `master` 主节点的 50030 的 web 端口（`http://master:50030`）直接通过浏览器查看 `jobtracker` 的运行情况，通过 50070 的 web 端口（`http://master:50070`）可以查看 `namenode` 的运行情况。

安装 R 的程序包 `rmr2`, `rhdfs`。这两个包可以到 `github` 下载：

<https://github.com/RevolutionAnalytics/RHadoop/wiki/Downloads>

`rhdfs` 是用 R 语言操作 `hdfs` 文件系统的程序包，`rmr2` 是 R `map/reduce 2` 的缩写。其中 `map/reduce` 是 Hadoop 分布式计算的基本程序设计模式。

这两个包有很多依赖包，首先需要通过 R 自带的 `install.packages` 函数来获取程序包，比如：

```
install.packages("rJava")
```

这是 `rhdfs` 的依赖包。`rmr2` 需要如下包：

```
Rcpp、RJSONIO、bitops、digest、functional、stringr、
plyr、reshape2
```

现在安装 `rhdfs` 和 `rmr2` 包，先把 `HADOOP_CMD` 和 `HADOOP_STREAMING` 设置到环境变量,这是给 `rhdfs` 做准备的，因为用 `r` 对 `hdfs` 文件系统本质上还是调用了 Hadoop 提供的 `JAVA` 操控 `hdfs` 文件系统的 `API`

命令，所以把 `hadoop` 命令和 `hadoop` 的文件流操作加入环境变量更好。

```
HADOOP_CMD=/usr/local/hadoop/bin/hadoop
```

```
HADOOP_STREAMING=/usr/local/hadoop/contrib/streaming/hadoop-streaming-1.0.3.jar
```

用以下命令安装 `rhdfs` 和 `rmr2` 包

```
R CMD INSTALL rhdfs_x.x.x.tar.gz
```

```
R CMD INSTALL rmr2_x.x.x.tar.gz
```

注意，要用 `root` 用户安装，这样就会装到 `/usr/local/lib/R/site-packages/` 路径下，集群中的每台机器都一样。对于第三方的包，`hadoop` 会从每个从机中的这个目录下寻找库函数，如果在这个目录下没有，当你在 `map` 函数中调用第三方包的时候 `Hadoop` 通常会报 “`map tasks exceed allowed limit`” 错误。

之后在 `R` 中运行 `library(rmr2)` 和 `library(rhdfs)` 如果没有报错，说明已经安装完成了。如果有错，无非就是提示还没有安装某某依赖包，那接着安装就是了。

现在就可以用 `R` 在 `hadoop` 服务器集群上运行 `map/reduce` 程序了！