

# Flink官方文档翻译：安装部署（集群模式）

📅 2016-02-26 | 📁 [分布式系统](#) | 👁 1042

本文主要介绍如何将Flink以分布式模式运行在集群上（可能是异构的）。

## 环境准备

Flink 运行在所有类 UNIX 环境上，例如 Linux、Mac OS X 和 Cygwin（对于Windows），而且要求集群由一个 **master** 节点和一个或多个 **worker** 节点组成。在安装系统之前，确保每台机器上都已经安装了下面的软件：

- **Java 1.7.x**或更高版本
- **ssh**（Flink的脚本会用到sshd来管理远程组件）

如果你的集群还没有完全装好这些软件，你需要安装/升级它们。例如，在 Ubuntu Linux 上，你可以执行下面的命令安装 ssh 和 Java：

```
sudo apt-get install openjdk-7-jre
```

## SSH免密码登录

译注：安装过Hadoop、Spark集群的用户应该对这段很熟悉，如果已经了解，可跳过。\*

为了能够启动/停止远程主机上的进程，master节点需要能免密登录所有worker节点。最方便的方式就是使用ssh的公钥验证了。要安装公钥验证，首先以最终会运行Flink的用户登录master节点。所有的**worker**节点上也必须要有同样的用户（例如：使用相同用户名的用户）。本文会以 flink 用户为例。非常不建议使用 root 账户，这会有很多的安全问题。

当你用需要的用户登录了master节点，你就可以生成一对新的公钥/私钥。下面这段命令会在 ~/.ssh 目录下生成一对新的公钥/私钥。

```
ssh-keygen -b 2048 -P '' -f ~/.ssh/id_rsa
```

接下来，将公钥添加到用于认证的authorized\_keys文件中：

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

最后，将authorized\_keys文件分发给集群中所有的worker节点，你可以重复地执行下面这段命令：

```
scp ~/.ssh/authorized_keys <worker>:~/.ssh/
```

将上面的<worker>替代成相应worker节点的IP/Hostname。完成了上述拷贝的工作，你应该就可以从master上免密登录其他机器了。

```
ssh <worker>
```

## 配置JAVA\_HOME

Flink 需要master和worker节点都配置了 JAVA\_HOME环境变量。有两种方式可以配置。

一种是，你可以在conf/flink-conf.yaml中设置env.java.home配置项为Java的安装路径。

另一种是，sudo vi /etc/profile，在其中添加JAVA\_HOME：

```
export JAVA_HOME=/path/to/java_home/
```

然后使环境变量生效，并验证 Java 是否安装成功

```
$ source /etc/profile    #生效环境变量
$ java -version          #如果打印出版本信息，则说明安装成功
java version "1.7.0_75"
Java(TM) SE Runtime Environment (build 1.7.0_75-b13)
Java HotSpot(TM) 64-Bit Server VM (build 24.75-b04, mixed mode)
```

## 安装 Flink

进入[下载页面](#)。请选择一个与你的Hadoop版本相匹配的Flink包。如果你不打算使用Hadoop，选择任何版本都可以。

在下载了最新的发布包后，拷贝到master节点上，并解压：

```
tar xzf flink-*.tgz
cd flink-*
```

## 配置 Flink

在解压完之后，你需要编辑`conf/flink-conf.yaml`配置Flink。

设置`jobmanager.rpc.address`配置项为你的master节点地址。另外为了明确 JVM 在每个节点上所能分配的最大内存，我们需要配置`jobmanager.heap.mb`和`taskmanager.heap.mb`，值的单位是 MB。如果对于某些 worker 节点，你想要分配更多的内存给Flink系统，你可以在相应节点上设置`FLINK_TM_HEAP`环境变量来覆盖默认的配置。

最后，你需要提供一个集群中worker节点的列表。因此，就像配置HDFS，编辑`conf/slaves`文件，然后输入每个 worker 节点的 IP/Hostname。每一个worker结点之后都会运行一个 TaskManager。

每一条记录占一行，就像下面展示的一样：

```
192.168.0.100
192.168.0.101
.
.
```

```
.  
192.168.0.150
```

译注：`conf/master`文件是用来做JobManager HA的，在这里不需要配置

每一个worker节点上的 Flink 路径必须一致。你可以使用共享的 NSF 目录，或者拷贝整个 Flink 目录到各个worker节点。

```
scp -r /path/to/flink <worker>:/path/to/
```

请查阅[配置页面](#)了解更多关于Flink的配置。

特别的，这几个

- TaskManager 总共能使用的内存大小（`taskmanager.heap.mb`）
- 每一台机器上能使用的 CPU 个数（`taskmanager.numberOfTaskSlots`）
- 集群中的总 CPU 个数（`parallelism.default`）
- 临时目录（`taskmanager.tmp.dirs`）

是非常重要的配置项。

## 启动 Flink

下面的脚本会在本地节点启动一个 JobManager，然后通过 SSH 连接所有的worker节点（`slaves`文件中所列的节点），并在每个节点上运行 TaskManager。现在你的 Flink 系统已经启动并运行了。跑在本地节点上的 JobManager 现在会在配置的 RPC 端口上监听并接收任务。

假定你在master节点上，并在Flink目录中：

```
bin/start-cluster.sh
```

要停止Flink，也有一个 *stop-cluster.sh* 脚本。

## 添加 JobManager/TaskManager 实例到集群中

你可以使用 *bin/jobmanager.sh* 和 *bin/taskmanager* 脚本来添加 JobManager 和 TaskManager 实例到你正在运行的集群中。

### 添加一个 JobManager

```
bin/jobmanager.sh (start cluster)|stop|stop-all
```

### 添加一个 TaskManager

```
bin/taskmanager.sh start|stop|stop-all
```

确保你是在需要启动/停止相应实例的节点上运行的这些脚本。

©著作权归作者所有

#Flink    #Flink官方文档翻译    #部署

分享到： 新浪微博 微信 Twitter Facebook 更多 0

◀ Flink官方文档翻译：安装部署（本地模式）

Flink 原理与实现：如何处理反压问题 ▶

0条评论 Jark's Blog

1 登录 ▼

♥ 推荐

🔗 分享

按评分高低排序 ▼



开始讨论.....

来做第一个留言的人吧！

在 JARK'S BLOG 上还有

使用Redis和SQLAlchemy对Scrapy Item去重并存储

11条评论 • 1年前•



musa — <http://doc.scrapy.org/en/1....>改动参考这里~

Flink 原理与实现：Session Window

2条评论 • 6个月前•



Jark — 应该会OOM的，可以再设置一个trigger，当窗口的length达到一定的程度的时候强制fire。

2013年终总结

5条评论 • 1年前•



黑桃夹克 — 呵呵，下次可以给你看看照片

Git 常用技能

4条评论 • 1年前•



爱奇艺分享网 — 秒赞不是偶然,是一种态度!

© 2013 - 2017 ♥ WuChong

由 [Hexo](#) 强力驱动 | 主题 - [NexT.Mist](#)