
toc: true date: 2018-10-27 title: 《从0到1学习Flink》—— Flink 配置文件详解
tags:

- Flink
 - 大数据
 - 流式计算
-



前面文章我们已经知道 Flink 是什么东西了，安装好 Flink 后，我们再来看下安装路径下的配置文件吧。

```
# server.z=host:peer-port:leader-port
zhisheng@zhisheng > /usr/local/Cellar/apache-flink/1.6.0/libexec/conf 11
total 112
-rw-r--r--@ 1 zhisheng admin 9.6K 8 6 23:11 flink-conf.yaml
-rw-r--r-- 1 zhisheng admin 2.1K 8 6 23:11 log4j-cli.properties
-rw-r--r-- 1 zhisheng admin 1.8K 8 6 23:11 log4j-console.properties
-rw-r--r-- 1 zhisheng admin 1.7K 8 6 23:11 log4j-yarn-session.properties
-rw-r--r-- 1 zhisheng admin 1.9K 8 6 23:11 log4j.properties
-rw-r--r-- 1 zhisheng admin 2.2K 8 6 23:11 logback-console.xml
-rw-r--r-- 1 zhisheng admin 1.5K 8 6 23:11 logback-yarn.xml
-rw-r--r-- 1 zhisheng admin 2.3K 8 6 23:11 logback.xml
-rw-r--r-- 1 zhisheng admin 15B 8 6 23:11 masters
-rw-r--r-- 1 zhisheng admin 10B 8 6 23:11 slaves
-rw-r--r-- 1 zhisheng admin 3.2K 8 6 23:11 sql-client-defaults.yaml
-rw-r--r-- 1 zhisheng admin 1.4K 8 6 23:11 zoo.cfg
zhisheng@zhisheng > /usr/local/Cellar/apache-flink/1.6.0/libexec/conf
```

安装目录下主要有 flink-conf.yaml 配置、日志的配置文件、zk 配置、Flink SQL Client 配置。

flink-conf.yaml

基础配置

```
# jobManager 的IP地址
jobmanager.rpc.address: localhost

# JobManager 的端口号
jobmanager.rpc.port: 6123

# JobManager JVM heap 内存大小
jobmanager.heap.size: 1024m

# TaskManager JVM heap 内存大小
taskmanager.heap.size: 1024m

# 每个 TaskManager 提供的任务 slots 数量大小

taskmanager.numberOfTaskSlots: 1

# 程序默认并行计算的个数
parallelism.default: 1

# 文件系统来源
# fs.default-scheme
```

高可用性配置

```
# 可以选择 'NONE' 或者 'zookeeper'.
# high-availability: zookeeper

# 文件系统路径, 让 Flink 在高可用性设置中持久保存元数据
# high-availability.storageDir: hdfs:///flink/ha/

# zookeeper 集群中仲裁者的机器 ip 和 port 端口号
# high-availability.zookeeper.quorum: localhost:2181

# 默认是 open, 如果 zookeeper security 启用了该值会更改成 creator
# high-availability.zookeeper.client.acl: open
```

容错和检查点 配置

```
# 用于存储和检查点状态
# state.backend: filesystem

# 存储检查点的数据文件和元数据的默认目录
# state.checkpoints.dir: hdfs://namenode-host:port/flink-checkpoints

# savepoints 的默认目标目录(可选)
# state.savepoints.dir: hdfs://namenode-host:port/flink-checkpoints

# 用于启用/禁用增量 checkpoints 的标志
# state.backend.incremental: false
```

web 前端配置

```
# 基于 Web 的运行时监视器侦听的地址.
#jobmanager.web.address: 0.0.0.0

# Web 的运行时监视器端口
rest.port: 8081

# 是否从基于 Web 的 jobmanager 启用作业提交
# jobmanager.web.submit.enable: false
```

高级配置

```
# io.tmp.dirs: /tmp

# 是否应在 TaskManager 启动时预先分配 TaskManager 管理的内存
# taskmanager.memory.preallocate: false

# 类加载解析顺序，是先检查用户代码 jar (“child-first”) 还是应用程序类路径
# (“parent-first”)。默认设置指示首先从用户代码 jar 加载类
# classloader.resolve-order: child-first

# 用于网络缓冲区的 JVM 内存的分数。这决定了 TaskManager 可以同时拥有多少流数
# 据交换通道以及通道缓冲的程度。如果作业被拒绝或者您收到系统没有足够缓冲区的警告，
# 请增加此值或下面的最小/最大值。另请注意，“taskmanager.network.memory.min”和“taskmanager.network.memory.max”
# 可能会覆盖此分数

# taskmanager.network.memory.fraction: 0.1
# taskmanager.network.memory.min: 67108864
# taskmanager.network.memory.max: 1073741824
```

Flink 集群安全配置

```
# 指示是否从 Kerberos ticket 缓存中读取
# security.kerberos.login.use-ticket-cache: true

# 包含用户凭据的 Kerberos 密钥表文件的绝对路径
# security.kerberos.login.keytab: /path/to/kerberos/keytab

# 与 keytab 关联的 Kerberos 主体名称
# security.kerberos.login.principal: flink-user

# 以逗号分隔的登录上下文列表，用于提供 Kerberos 凭据（例如，`Client`,
# `KafkaClient` 使用凭证进行 ZooKeeper 身份验证和 Kafka 身份验证）
# security.kerberos.login.contexts: Client,KafkaClient
```

Zookeeper 安全配置

```
# 覆盖以下配置以提供自定义 ZK 服务名称
# zookeeper.sasl.service-name: zookeeper

# 该配置必须匹配 "security.kerberos.login.contexts" 中的列表（含有一个）
# zookeeper.sasl.login-context-name: Client
```

HistoryServer

```
# 你可以通过 bin/historyserver.sh (start|stop) 命令启动和关闭 HistoryServer

# 将已完成的作业上传到的目录
# jobmanager.archive.fs.dir: hdfs:///completed-jobs/

# 基于 Web 的 HistoryServer 的地址
# historyserver.web.address: 0.0.0.0

# 基于 Web 的 HistoryServer 的端口号
# historyserver.web.port: 8082

# 以逗号分隔的目录列表，用于监视已完成的作业
# historyserver.archive.fs.dir: hdfs:///completed-jobs/

# 刷新受监控目录的时间间隔（以毫秒为单位）
# historyserver.archive.fs.refresh-interval: 10000
```

查看下另外两个配置 slaves / master

```
zhisheng@zhisheng > /usr/local/Cellar/apache-flink/1.6.0/libexec/conf cat masters
localhost:8081
zhisheng@zhisheng > /usr/local/Cellar/apache-flink/1.6.0/libexec/conf cat slaves
localhost
zhisheng@zhisheng > /usr/local/Cellar/apache-flink/1.6.0/libexec/conf
```

2、slaves

里面是每个 worker 节点的 IP/Hostname，每一个 worker 结点之后都会运行一个 TaskManager，一个一行。

```
localhost
```

3、masters

host:port

```
localhost:8081
```

4、zoo.cfg

```
# 每个 tick 的毫秒数
tickTime=2000

# 初始同步阶段可以采用的 tick 数
initLimit=10

# 在发送请求和获取确认之间可以传递的 tick 数
syncLimit=5

# 存储快照的目录
# dataDir=/tmp/zookeeper

# 客户端将连接的端口
clientPort=2181

# ZooKeeper quorum peers
server.1=localhost:2888:3888
# server.2=host:peer-port:leader-port
```

5、日志配置

Flink 在不同平台下运行的日志文件

```
log4j-cli.properties
log4j-console.properties
log4j-yarn-session.properties
log4j.properties
logback-console.xml
logback-yarn.xml
logback.xml
```

sql-client-defaults.yaml

```
execution:
  # 'batch' or 'streaming' execution
  type: streaming
  # allow 'event-time' or only 'processing-time' in sources
  time-characteristic: event-time
  # interval in ms for emitting periodic watermarks
  periodic-watermarks-interval: 200
  # 'changelog' or 'table' presentation of results
  result-mode: changelog
  # parallelism of the program
  parallelism: 1
  # maximum parallelism
  max-parallelism: 128
  # minimum idle state retention in ms
  min-idle-state-retention: 0
  # maximum idle state retention in ms
  max-idle-state-retention: 0

deployment:
  # general cluster communication timeout in ms
  response-timeout: 5000
  # (optional) address from cluster to gateway
  gateway-address: ""
  # (optional) port from cluster to gateway
  gateway-port: 0
```

Flink sql client：你可以从官网这里了解 <https://ci.apache.org/projects/flink/flink-docs-stable/dev/table/sqlClient.html>

总结

本文拿安装目录文件下的配置文件讲解了下 Flink 目录下的所有配置。

你也可以通过官网这里学习更多：<https://ci.apache.org/projects/flink/flink-docs-stable/ops/config.html>

关注我

本篇文章地址是：<http://www.54tianzhisheng.cn/2018/10/27/flink-config/>

另外我自己整理了些 Flink 的学习资料，目前已经全部放到微信公众号了。你可以加我的微信：zhisheng_tian，然后回复关键字：Flink 即可无条件获取到。



Github 代码仓库

<https://github.com/zhisheng17/flink-learning/>

以后这个项目的所有代码都将放在这个仓库里，包含了自己学习 flink 的一些 demo 和博客

相关文章

- 1、[《从0到1学习Flink》—— Apache Flink 介绍](#)
- 2、[《从0到1学习Flink》—— Mac 上搭建 Flink 1.6.0 环境并构建运行简单程序入门](#)
- 3、[《从0到1学习Flink》—— Flink 配置文件详解](#)
- 4、[《从0到1学习Flink》—— Data Source 介绍](#)
- 5、[《从0到1学习Flink》—— 如何自定义 Data Source ?](#)
- 6、[《从0到1学习Flink》—— Data Sink 介绍](#)
- 7、[《从0到1学习Flink》—— 如何自定义 Data Sink ?](#)
- 8、[《从0到1学习Flink》—— Flink Data transformation\(转换\)](#)
- 9、[《从0到1学习Flink》—— 介绍Flink中的Stream Windows](#)
- 10、[《从0到1学习Flink》—— Flink 中的几种 Time 详解](#)
- 11、[《从0到1学习Flink》—— Flink 写入数据到 ElasticSearch](#)