

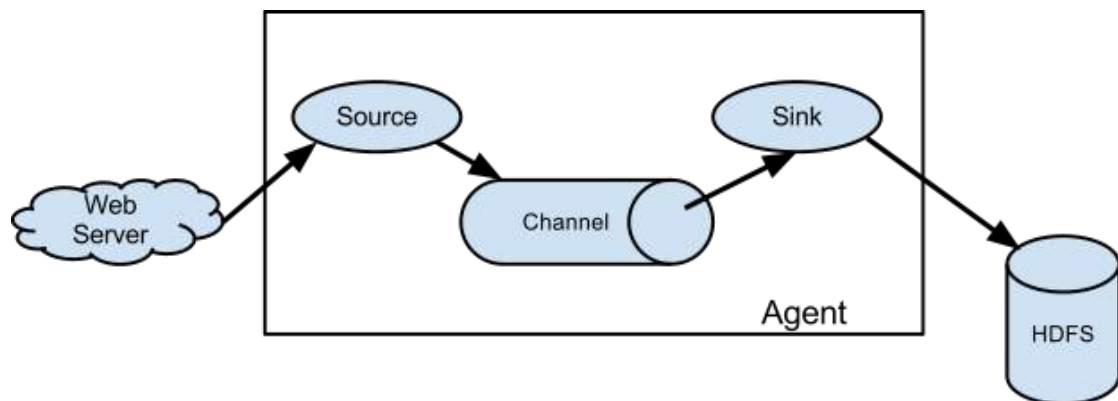
第 1 章 Flume 概述

1.1 Flume 概念

Flume 是一种分布式，可靠和可用的服务，用于高效收集，聚合和移动大量日志数据。它具有基于流数据流的简单灵活的架构。它具有可靠的可靠性机制和许多故障转移和恢复机制的强大和容错能力。它使用一个简单的可扩展数据模型，允许在线分析应用程序。

Flume 是 Cloudera 公司开发的用于实时收集服务器（apache/nginx 等）日志数据的框架，Flume 很多时候和 storm 以及 spark streaming 等流式处理框架结合使用。

1.2 Flume 组成架构



Agent

具有字节有效载荷和可选的一组字符串属性的数据流的单元。Flume 代理(Agent)是一个 (JVM) 进程，它承载事件从外部源传递到下一个目标（跳）的组件。是 Flume 数据传输的基本单元，以事件的形式将数据从源头送至目的地。

Agent 就是一个 java 进程，Agent 主要有 3 个部分组成，Source、Channel、Sink。

第 2 章 角色介绍

2.1 Source

source 组件是专门用来收集数据的,可以处理各种类型、各种格式的日志数据,包括 avro、thrift、exec、jms、spooling directory、netcat、sequence generator、syslog、http、legacy

2.2 Channel

source 组件把数据收集来以后,临时存放在 channel 中,即 channel 组件在 agent 中是专门用来存放临时数据的——对采集到的数据进行简单的缓存,可以存放在 memory、jdbc、file 等等。

2.3 Sink

sink 组件是用于把数据发送到目的地的组件,目的地包括 hdfs、logger、avro、thrift、ipc、file、null、Hbase、solr、自定义。

2.4 Event

传输单元, Flume 数据传输的基本单元,以事件的形式将数据从源头送至目的地。

第 3 章 快速入门

3.1 Flume 安装地址

- 1) Flume 官网地址

<http://flume.apache.org/>

- 2) 文档查看地址

<http://flume.apache.org/FlumeUserGuide.html>

- 3) 下载地址

<http://archive.apache.org/dist/flume/>

3.2 安装部署

- 1) 将 apache-flume-1.7.0-bin.tar.gz 上传到 linux 的 /opt/software 目录下

- 2) 解压 apache-flume-1.7.0-bin.tar.gz 到 /opt/module/ 目录下

```
[atguigu@hadoop102 software]$ tar -zxf apache-flume-1.7.0-bin.tar.gz -C /opt/module/
```

- 3) 修改 apache-flume-1.7.0-bin 的名称为 flume

```
[atguigu@hadoop102 module]$ mv apache-flume-1.7.0-bin flume
```

- 4) 将 flume/conf 下的 flume-env.sh.template 文件修改为 flume-env.sh，并配置 flume-env.sh 文件

```
[atguigu@hadoop102 conf]$ mv flume-env.sh.template flume-env.sh
```

```
[atguigu@hadoop102 conf]$ vi flume-env.sh
```

```
export JAVA_HOME=/opt/module/jdk1.8.0_144
```

第 4 章 案例实操

4.1 监控端口数据

目标：Flume 监控一端 Console，另一端 Console 发送消息，使被监控端实时显示。

分步实现：

1) 先将 rpm 软件包(xinetd-2.3.14-40.el6.x86_64.rpm、telnet-0.17-48.el6.x86_64.rpm 和 telnet-server-0.17-48.el6.x86_64.rpm)拷入 Linux 系统。执行 RPM 软件包安装命令：

```
[atguigu@hadoop102 software]$ sudo rpm -ivh xinetd-2.3.14-40.el6.x86_64.rpm
```

```
[atguigu@hadoop102 software]$ sudo rpm -ivh telnet-0.17-48.el6.x86_64.rpm
```

```
[atguigu@hadoop102 software]$ sudo rpm -ivh telnet-server-0.17-48.el6.x86_64.rpm
```

2) 在 flume 目录下创建 job 文件夹，并在 job 文件夹下创建 Flume Agent 配置文件 flume_telnet.conf

```
# Name the components on this agent
a1.sources = r1
a1.sinks = k1
a1.channels = c1

# Describe/configure the source
a1.sources.r1.type = netcat
a1.sources.r1.bind = localhost
a1.sources.r1.port = 44444

# Describe the sink
a1.sinks.k1.type = logger

# Use a channel which buffers events in memory
a1.channels.c1.type = memory
a1.channels.c1.capacity = 1000
a1.channels.c1.transactionCapacity = 100

# Bind the source and sink to the channel
a1.sources.r1.channels = c1
a1.sinks.k1.channel = c1
```

3) 判断 44444 端口是否被占用

```
[atguigu@hadoop102 software]$ sudo netstat -tunlp | grep 44444
```

4) 先开启 flume 监听端口

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a1 --conf-file job/flume_telnet.conf -Dflume.root.logger==INFO,console
```

5) 使用 telnet 工具向本机的 44444 端口发送内容

```
[atguigu@hadoop102 software]$ telnet localhost 44444
```

4.2 实时读取本地文件到 HDFS

目标：实时监控 hive 日志，并上传到 HDFS 中

分步实现：

1) 拷贝 Hadoop 相关 jar 到 Flume 的 lib 目录下（要学会根据自己的目录和版本查找 jar 包）

hadoop-auth-2.7.2.jar

commons-configuration-1.6.jar

hadoop-hdfs-2.7.2.jar

hadoop-common-2.7.2.jar

htrace-core-3.1.0-incubating.jar

commons-io-2.4.jar

提示：标红的 jar 为 1.99 版本 flume 必须引用的 jar

2) 创建 flume_hdfs.conf 文件

```
# Name the components on this agent
a2.sources = r2
a2.sinks = k2
a2.channels = c2
# Describe/configure the source
a2.sources.r2.type = exec
a2.sources.r2.command = tail -F /opt/module/hive/hive.log
a2.sources.r2.shell = /bin/bash -c

# Describe the sink
a2.sinks.k2.type = hdfs
a2.sinks.k2.hdfs.path = hdfs://hadoop102:9000/flume/%Y%m%d/%H
#上传文件的前缀
a2.sinks.k2.hdfs.filePrefix = logs-
#是否按照时间滚动文件夹
a2.sinks.k2.hdfs.round = true
#多少时间单位创建一个新的文件夹
a2.sinks.k2.hdfs.roundValue = 1
#重新定义时间单位
```

```
a2.sinks.k2.hdfs.roundUnit = hour
#是否使用本地时间戳
a2.sinks.k2.hdfs.useLocalTimeStamp = true
#积攒多少个 Event 才 flush 到 HDFS 一次
a2.sinks.k2.hdfs.batchSize = 1000
#设置文件类型，可支持压缩
a2.sinks.k2.hdfs.fileType = DataStream
#多久生成一个新的文件
a2.sinks.k2.hdfs.rollInterval = 600
#设置每个文件的滚动大小
a2.sinks.k2.hdfs.rollSize = 134217700
#文件的滚动与 Event 数量无关
a2.sinks.k2.hdfs.rollCount = 0
#最小冗余数
a2.sinks.k2.hdfs.minBlockReplicas = 1

# Use a channel which buffers events in memory
a2.channels.c2.type = memory
a2.channels.c2.capacity = 1000
a2.channels.c2.transactionCapacity = 100

# Bind the source and sink to the channel
a2.sources.r2.channels = c2
a2.sinks.k2.channel = c2
```

3) 执行监控配置

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a2 --conf-file
job/flume_hdfs.conf
```

4) 开启 hive 或者操作 hive 使其产生日志

4.3 实时读取目录文件到 HDFS

目标：使用 flume 监听整个目录的文件

分步实现：

1) 创建配置文件 flume-dir.conf

```
a3.sources = r3
a3.sinks = k3
a3.channels = c3

# Describe/configure the source
a3.sources.r3.type = spooldir
```

```
a3.sources.r3.spoolDir = tail -F /opt/module/flume/upload
a3.sources.r3.fileSuffix = .COMPLETED
a3.sources.r3.fileHeader = true
#忽略所有以.tmp 结尾的文件，不上传
a3.sources.r3.ignorePattern = ([^ ]*\tmp)

# Describe the sink
a3.sinks.k3.type = hdfs
a3.sinks.k3.hdfs.path = hdfs://hadoop102:9000/flume/upload/%Y%m%d/%H
#上传文件的前缀
a3.sinks.k3.hdfs.filePrefix = upload-
#是否按照时间滚动文件夹
a3.sinks.k3.hdfs.round = true
#多少时间单位创建一个新的文件夹
a3.sinks.k3.hdfs.roundValue = 1
#重新定义时间单位
a3.sinks.k3.hdfs.roundUnit = hour
#是否使用本地时间戳
a3.sinks.k3.hdfs.useLocalTimeStamp = true
#积攒多少个 Event 才 flush 到 HDFS 一次
a3.sinks.k3.hdfs.batchSize = 100
#设置文件类型，可支持压缩
a3.sinks.k3.hdfs.fileType = DataStream
#多久生成一个新的文件
a3.sinks.k3.hdfs.rollInterval = 600
#设置每个文件的滚动大小大概是 128M
a3.sinks.k3.hdfs.rollSize = 134217700
#文件的滚动与 Event 数量无关
a3.sinks.k3.hdfs.rollCount = 0
#最小冗余数
a3.sinks.k3.hdfs.minBlockReplicas = 1

# Use a channel which buffers events in memory
a3.channels.c3.type = memory
a3.channels.c3.capacity = 1000
a3.channels.c3.transactionCapacity = 100

# Bind the source and sink to the channel
a3.sources.r3.channels = c3
a3.sinks.k3.channel = c3
```

2) 执行测试：执行如下脚本后，请向 upload 文件夹中添加文件

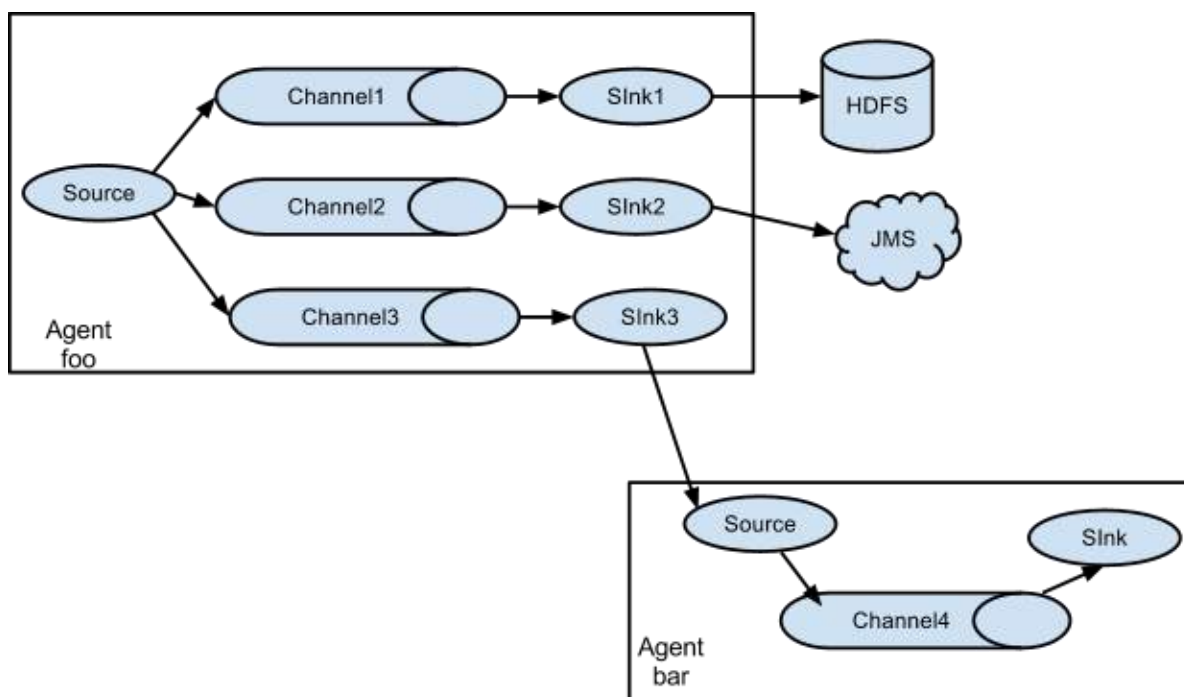
```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a3 --conf-file
```

job/flume-dir.conf

说明： 在使用 Spooling Directory Source 时

- a. 不要在监控目录中创建并持续修改文件
- b. 上传完成的文件会以.COMPLETED 结尾
- c. 被监控文件夹每 600 毫秒扫描一次文件变动

4.4 单 Flume 多 Channel、Sink



目标：使用 flume-1 监控文件变动，flume-1 将变动内容传递给 flume-2，flume-2 负责存储到 HDFS。同时 flume-1 将变动内容传递给 flume-3，flume-3 负责输出到 local filesystem。

分步实现：

- 1) 创建 flume-1.conf，用于监控 hive.log 文件的变动，同时产生两个 channel 和两个 sink 分别输送给 flume-2 和 flume3：

```
# Name the components on this agent
a1.sources = r1
a1.sinks = k1 k2
a1.channels = c1 c2
# 将数据流复制给多个 channel
a1.sources.r1.selector.type = replicating

# Describe/configure the source
a1.sources.r1.type = exec
a1.sources.r1.command = tail -F /opt/module/hive/hive.log
```



```
a1.sources.r1.shell = /bin/bash -c

# Describe the sink
a1.sinks.k1.type = avro
a1.sinks.k1.hostname = hadoop102
a1.sinks.k1.port = 4141

a1.sinks.k2.type = avro
a1.sinks.k2.hostname = hadoop102
a1.sinks.k2.port = 4142

# Describe the channel
a1.channels.c1.type = memory
a1.channels.c1.capacity = 1000
a1.channels.c1.transactionCapacity = 100

a1.channels.c2.type = memory
a1.channels.c2.capacity = 1000
a1.channels.c2.transactionCapacity = 100

# Bind the source and sink to the channel
a1.sources.r1.channels = c1 c2
a1.sinks.k1.channel = c1
a1.sinks.k2.channel = c2
```

2) 创建 flume-2.conf, 用于接收 flume-1 的 event, 同时产生 1 个 channel 和 1 个 sink, 将数据输送给 hdfs:

```
# Name the components on this agent
a2.sources = r1
a2.sinks = k1
a2.channels = c1

# Describe/configure the source
a2.sources.r1.type = avro
a2.sources.r1.bind = hadoop102
a2.sources.r1.port = 4141

# Describe the sink
a2.sinks.k1.type = hdfs
a2.sinks.k1.hdfs.path = hdfs://hadoop102:9000/flume2/%Y%m%d/%H
#上传文件的前缀
a2.sinks.k1.hdfs.filePrefix = flume2-
#是否按照时间滚动文件夹
```

```
a2.sinks.k1.hdfs.round = true
#多少时间单位创建一个新的文件夹
a2.sinks.k1.hdfs.roundValue = 1
#重新定义时间单位
a2.sinks.k1.hdfs.roundUnit = hour
#是否使用本地时间戳
a2.sinks.k1.hdfs.useLocalTimeStamp = true
#积攒多少个 Event 才 flush 到 HDFS 一次
a2.sinks.k1.hdfs.batchSize = 100
#设置文件类型，可支持压缩
a2.sinks.k1.hdfs.fileType = DataStream
#多久生成一个新的文件
a2.sinks.k1.hdfs.rollInterval = 600
#设置每个文件的滚动大小大概是 128M
a2.sinks.k1.hdfs.rollSize = 134217700
#文件的滚动与 Event 数量无关
a2.sinks.k1.hdfs.rollCount = 0
#最小冗余数
a2.sinks.k1.hdfs.minBlockReplicas = 1

# Describe the channel
a2.channels.c1.type = memory
a2.channels.c1.capacity = 1000
a2.channels.c1.transactionCapacity = 100

# Bind the source and sink to the channel
a2.sources.r1.channels = c1
a2.sinks.k1.channel = c1
```

3) 创建 flume-3.conf，用于接收 flume-1 的 event，同时产生 1 个 channel 和 1 个 sink，将数据输送给本地目录：

```
# Name the components on this agent
a3.sources = r1
a3.sinks = k1
a3.channels = c1

# Describe/configure the source
a3.sources.r1.type = avro
a3.sources.r1.bind = localhost
a3.sources.r1.port = 4142

# Describe the sink
```

```
a3.sinks.k1.type = file_roll
a3.sinks.k1.sink.directory = /home/atguigu/flume3

# Describe the channel
a3.channels.c1.type = memory
a3.channels.c1.capacity = 1000
a3.channels.c1.transactionCapacity = 100

# Bind the source and sink to the channel
a3.sources.r1.channels = c1
a3.sinks.k1.channel = c1
```

提示：输出的本地目录必须是已经存在的目录，如果该目录不存在，并不会创建新的目录。

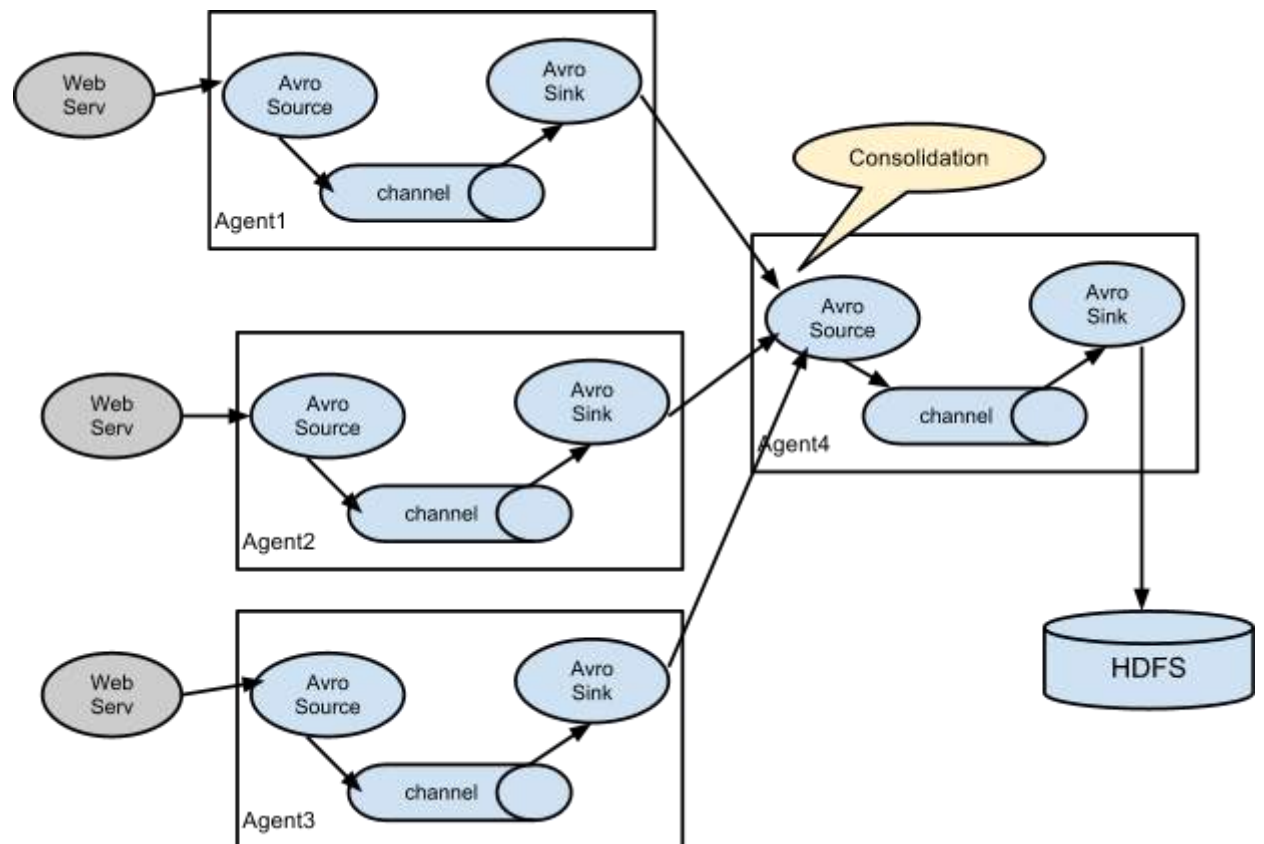
4) 执行测试：分别开启对应 flume-job（依次启动 flume-3, flume-2, flume-1），同时产生文件变动并观察结果：

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a3 --conf-file
job/group-job1/flume-3.conf
```

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a2 --conf-file
job/group-job1/flume-2.conf
```

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a1 --conf-file
job/group-job1/flume-1.conf
```

4.5 多 Flume 汇总数据到单 Flume



目标：flume-1 监控文件 hive.log，flume-2 监控某一个端口的数据流，flume-1 与 flume-2 将数据发送给 flume-3，flume3 将最终数据写入到 HDFS。

分步实现：

1) 创建 flume-1.conf，用于监控 hive.log 文件，同时 sink 数据到 flume-3:

```
# Name the components on this agent
a1.sources = r1
a1.sinks = k1
a1.channels = c1

# Describe/configure the source
a1.sources.r1.type = exec
a1.sources.r1.command = tail -F /opt/module/hive/hive.log
a1.sources.r1.shell = /bin/bash -c

# Describe the sink
a1.sinks.k1.type = avro
a1.sinks.k1.hostname = hadoop102
a1.sinks.k1.port = 4141
```

```
# Describe the channel
a1.channels.c1.type = memory
a1.channels.c1.capacity = 1000
a1.channels.c1.transactionCapacity = 100

# Bind the source and sink to the channel
a1.sources.r1.channels = c1
a1.sinks.k1.channel = c1
```

2) 创建 flume-2.conf, 用于监控端口 44444 数据流, 同时 sink 数据到 flume-3:

```
# Name the components on this agent
a2.sources = r1
a2.sinks = k1
a2.channels = c1

# Describe/configure the source
a2.sources.r1.type = netcat
a2.sources.r1.bind = hadoop102
a2.sources.r1.port = 44444

# Describe the sink
a2.sinks.k1.type = avro
a2.sinks.k1.hostname = hadoop102
a2.sinks.k1.port = 4141

# Use a channel which buffers events in memory
a2.channels.c1.type = memory
a2.channels.c1.capacity = 1000
a2.channels.c1.transactionCapacity = 100

# Bind the source and sink to the channel
a2.sources.r1.channels = c1
a2.sinks.k1.channel = c1
```

3) 创建 flume-3.conf, 用于接收 flume-1 与 flume-2 发送过来的数据流, 最终合并后 sink 到 HDFS:

```
# Name the components on this agent
a3.sources = r1
a3.sinks = k1
a3.channels = c1

# Describe/configure the source
```

```
a3.sources.r1.type = avro
a3.sources.r1.bind = hadoop102
a3.sources.r1.port = 4141

# Describe the sink
a3.sinks.k1.type = hdfs
a3.sinks.k1.hdfs.path = hdfs://hadoop102:9000/flume3/%Y%m%d/%H
#上传文件的前缀
a3.sinks.k1.hdfs.filePrefix = flume3-
#是否按照时间滚动文件夹
a3.sinks.k1.hdfs.round = true
#多少时间单位创建一个新的文件夹
a3.sinks.k1.hdfs.roundValue = 1
#重新定义时间单位
a3.sinks.k1.hdfs.roundUnit = hour
#是否使用本地时间戳
a3.sinks.k1.hdfs.useLocalTimeStamp = true
#积攒多少个 Event 才 flush 到 HDFS 一次
a3.sinks.k1.hdfs.batchSize = 100
#设置文件类型，可支持压缩
a3.sinks.k1.hdfs.fileType = DataStream
#多久生成一个新的文件
a3.sinks.k1.hdfs.rollInterval = 600
#设置每个文件的滚动大小大概是 128M
a3.sinks.k1.hdfs.rollSize = 134217700
#文件的滚动与 Event 数量无关
a3.sinks.k1.hdfs.rollCount = 0
#最小冗余数
a3.sinks.k1.hdfs.minBlockReplicas = 1

# Describe the channel
a3.channels.c1.type = memory
a3.channels.c1.capacity = 1000
a3.channels.c1.transactionCapacity = 100

# Bind the source and sink to the channel
a3.sources.r1.channels = c1
a3.sinks.k1.channel = c1
```

4) 执行测试：分别开启对应 flume-job（依次启动 flume-3, flume-2, flume-1），同时产生文件变动并观察结果：

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a3 --conf-file
job/group-job2/flume-3.conf
```

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a2 --conf-file  
job/group-job2/flume-2.conf
```

```
[atguigu@hadoop102 flume]$ bin/flume-ng agent --conf conf/ --name a1 --conf-file  
job/group-job2/flume-1.conf
```

提示：测试时记得启动 hive 产生一些日志，同时使用 telnet 向 44444 端口发送内容，如：

```
[atguigu@hadoop102 hive]$ bin/hive
```

```
[atguigu@hadoop102 flume]$ telnet hadoop102 44444
```