注:**Flume**框架对**hadoop**和**zookeeper**的依赖只是在**jar**包上，并不要求**flume**启动时必须将**hadoop**和**zookeeper**服务也启动

来自 <http://www.cnblogs.com/oubo/archive/2012/05/25/2517751.html>

启动flume服务器,需要用一个服务端的配置文件

```
flume-ng agent --conf conf --conf-file /mysoftware/flume-1.7.0/conf/flume-server.properties --name a1 -Dflume.root.logger=INFO,console > /mysoftware/flume-1.7.0/logs/flume-server.log  2>&1 &
```

启动flume客户端,需要用一个客户端的配置文件

```
flume-ng agent --conf conf --conf-file /mysoftware/flume-1.7.0/conf/flume-client.properties --name agent1 -Dflume.root.logger=INFO,console > /mysoftware/flume-1.7.0/logs/flume-client.log  2>&1 &
```

这里的用到了avro,貌似必须先启动服务器,后启动客户端(这里是从客户端发消息到服务端)

以下是两个例子:

> 注:文件均需自己创建!,两个配置文件

1. 从网络端口接收,存入HDFS

flume-server.properties: 服务端　　　　　　(从avro端口到hdfs)

```
#set Agent name
a1.sources = r1
a1.channels = c1
a1.sinks = k1
#set channel
a1.channels.c1.type = memory
a1.channels.c1.capacity = 1000
a1.channels.c1.transactionCapacity = 100
# other node,nna to nns  source
a1.sources.r1.type = avro
a1.sources.r1.bind = master
a1.sources.r1.port = 52020
a1.sources.r1.interceptors = i1         #指定传输中event的head(头信息)，常用timestamp
a1.sources.r1.interceptors.i1.type = static
a1.sources.r1.interceptors.i1.key = Collector
a1.sources.r1.interceptors.i1.value = master
a1.sources.r1.channels = c1
#set sink to hdfs
a1.sinks.k1.type=hdfs
a1.sinks.k1.hdfs.path=hdfs://master:9000/flume/logdfs
a1.sinks.k1.hdfs.fileType=DataStream
a1.sinks.k1.hdfs.writeFormat=TEXT
a1.sinks.k1.hdfs.rollInterval=1
```

```
a1.sinks.k1.channel=c1
a1.sinks.k1.hdfs.filePrefix=%Y-%m-%d
a1.sinks.k1.hdfs.fileSuffix=.txt
```

flume-client.properties: 客户端（从命令结果中avro端口）

```
#agent1 name
 agent1.channels = c1
 agent1.sources = r1
 agent1.sinks = k1
 #set gruop
 #agent1.sinkgroups = g1
 #set channel
 agent1.channels.c1.type = memory
 agent1.channels.c1.capacity = 1000
 agent1.channels.c1.transactionCapacity = 100
 #set source
 agent1.sources.r1.channels = c1
 agent1.sources.r1.type = exec         #指定源的类型,为一个命令
 agent1.sources.r1.command = tail -F /mysoftware/flume-1.7.0/logdfs/flumetest.log
 agent1.sources.r1.interceptors = i1 i2
agent1.sources.r1.interceptors.i1.type = static
agent1.sources.r1.interceptors.i1.key = Type
agent1.sources.r1.interceptors.i1.value = LOGIN
agent1.sources.r1.interceptors.i2.type = timestamp
# set sink1
agent1.sinks.k1.channel = c1
agent1.sinks.k1.type = avro         #作为中间传递者,因为它具有很快捷的传输
agent1.sinks.k1.hostname = master
agent1.sinks.k1.port = 52020
# set sink2   #(可以有多个sink)
#agent1.sinks.k2.channel = c1
#agent1.sinks.k2.type = avro
#agent1.sinks.k2.hostname = slave1
#agent1.sinks.k2.port = 52020
#set sink group
#agent1.sinkgroups.g1.sinks = k1 k2
#set failover  #设置容错
#agent1.sinkgroups.g1.processor.type = failover         #load_balance 还有个是负载均衡
#agent1.sinkgroups.g1.processor.priority.k1 = 10
#agent1.sinkgroups.g1.processor.priority.k2 = 1
#agent1.sinkgroups.g1.processor.maxpenalty = 10000
```

2. 从监听文件中接收,存入Kafka

从命令结果到kafka(启动kafka消费者来查看结果)

```
#agent  客户端
agent.sources = origin
agent.channels = memorychannel
agent.sinks = target

agent.sources.origin.type = exec
agent.sources.origin.command = tail -F /home/hadoop/orderslog/app.log
agent.sources.origin.channels = memorychannel
agent.sources.origin.interceptors = i1
agent.sources.origin.interceptors.i1.type = static
agent.sources.origin.interceptors.i1.key = topic
agent.sources.origin.interceptors.i1.value = ordersInfo

agent.sinks.loggerSink.type = logger
agent.sinks.loggerSink.channel = memorychannel

agent.channels.memorychannel.type = memory
agent.channels.memorychannel.capacity = 10000

agent.sinks.target.type = avro
agent.sinks.target.channel = memorychannel
agent.sinks.target.hostname =master
agent.sinks.target.port = 4545


#collect      服务器
agent.sources = origin
agent.channels = memorychannel
agent.sinks = target

agent.sources.origin.type = avro
agent.sources.origin.channels = memorychannel
agent.sources.origin.bind = master
agent.sources.origin.port = 4545

agent.sinks.loggerSink.type = logger
agent.sinks.loggerSink.channel = memorychannel

agent.channels.memorychannel.type = memory
agent.channels.memorychannel.capacity = 5000000
agent.channels.memorychannel.transactionCapacity = 1000000

agent.sinks.target.type = org.apache.flume.sink.kafka.KafkaSink
agent.sinks.target.kafka.topic = ordersInfo          #数据的主题(kafka)
agent.sinks.target.kafka.bootstrap.servers=master:9092,slave01:9092,slave02:9092,slave03:9092

agent.sinks.target.metadata.broker.list=master:9092,slave01:9092,slave02:9092,slave03:9092
agent.sinks.target.producer.type=sync
#agent.sinks.target.kafka.producer.value.serializer=kafka.serializer.DefaultEncoder
agent.sinks.target.kafka.producer.acks=1
agent.sinks.target.flumeBatchSize=100
agent.sinks.target.channel = memorychannel
```