**[高可用Hadoop平台－Flume NG实战图解篇](http://www.cnblogs.com/smartloli/p/4468708.html)**

**1.概述**

　　今天补充一篇关于Flume的博客，前面在讲解高可用的Hadoop平台的时候遗漏了这篇，本篇博客为大家讲述以下内容：

* Flume NG简述
* 单点Flume NG搭建、运行
* 高可用Flume NG搭建
* Failover测试
* 截图预览

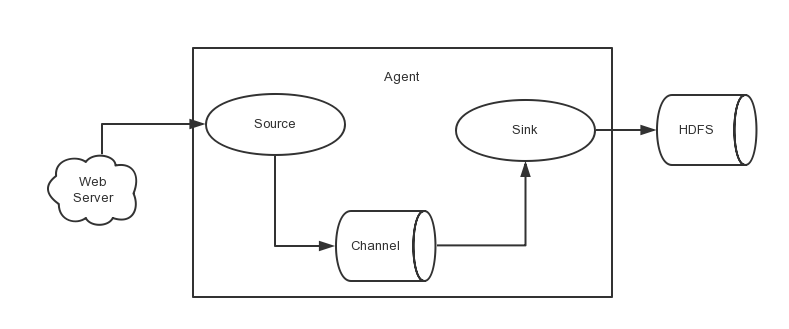
　　下面开始今天的博客介绍。

**2.Flume NG简述**

　　Flume NG是一个分布式，高可用，可靠的系统，它能将不同的海量数据收集，移动并存储到一个数据存储系统中。轻量，配置简单，适用于各种日志收集，并支持Failover和负载均衡。并且它拥有非常丰富的组件。Flume NG采用的是三层架构：Agent层，Collector层和Store层，每一层均可水平拓展。其中Agent包含Source，Channel和Sink，三者组建了一个Agent。三者的职责如下所示：

* Source：用来消费（收集）数据源到Channel组件中
* Channel：中转临时存储，保存所有Source组件信息
* Sink：从Channel中读取，读取成功后会删除Channel中的信息

　　下图是Flume NG的架构图，如下所示：



　　图中描述了，从外部系统（Web Server）中收集产生的日志，然后通过Flume的Agent的Source组件将数据发送到临时存储Channel组件，最后传递给Sink组件，Sink组件直接把数据存储到HDFS文件系统中。

**3.单点Flume NG搭建、运行**

　　我们在熟悉了Flume NG的架构后，我们先搭建一个单点Flume收集信息到HDFS集群中，由于资源有限，本次直接在之前的高可用Hadoop集群上搭建Flume。

　　场景如下：在NNA节点上搭建一个Flume NG，将本地日志收集到HDFS集群。

**3.1基础软件**

　　在搭建Flume NG之前，我们需要准备必要的软件，具体下载地址如下所示：

* Flume　　[《下载地址》](http://www.apache.org/dyn/closer.cgi/flume/1.5.2/apache-flume-1.5.2-bin.tar.gz)

　　JDK由于之前在安装Hadoop集群时已经配置过，这里就不赘述了，若需要配置的同学，可参考《[配置高可用的Hadoop平台](http://www.cnblogs.com/smartloli/p/4298430.html)》。

**3.2安装与配置**

* 安装

　　首先，我们解压flume安装包，命令如下所示：

[hadoop@nna ~]$ tar -zxvf apache-flume-1.5.2-bin.tar.gz

* 配置

　　环境变量配置内容如下所示：

export FLUME\_HOME=/home/hadoop/flume-1.5.2

export PATH=$PATH:$FLUME\_HOME/bin

　　flume-conf.properties

[复制代码](javascript:void(0);)

#agent1 name

agent1.sources=source1

agent1.sinks=sink1

agent1.channels=channel1

#Spooling Directory

#set source1

agent1.sources.source1.type=spooldir

agent1.sources.source1.spoolDir=/home/hadoop/dir/logdfs

agent1.sources.source1.channels=channel1

agent1.sources.source1.fileHeader = false

agent1.sources.source1.interceptors = i1

agent1.sources.source1.interceptors.i1.type = timestamp

#set sink1

agent1.sinks.sink1.type=hdfs

agent1.sinks.sink1.hdfs.path=/home/hdfs/flume/logdfs

agent1.sinks.sink1.hdfs.fileType=DataStream

agent1.sinks.sink1.hdfs.writeFormat=TEXT

agent1.sinks.sink1.hdfs.rollInterval=1

agent1.sinks.sink1.channel=channel1

agent1.sinks.sink1.hdfs.filePrefix=%Y-%m-%d

#set channel1

agent1.channels.channel1.type=file

agent1.channels.channel1.checkpointDir=/home/hadoop/dir/logdfstmp/point

agent1.channels.channel1.dataDirs=/home/hadoop/dir/logdfstmp

[复制代码](javascript:void(0);)

　　flume-env.sh

JAVA\_HOME=/usr/java/jdk1.7

　　注：配置中的目录若不存在，需提前创建。

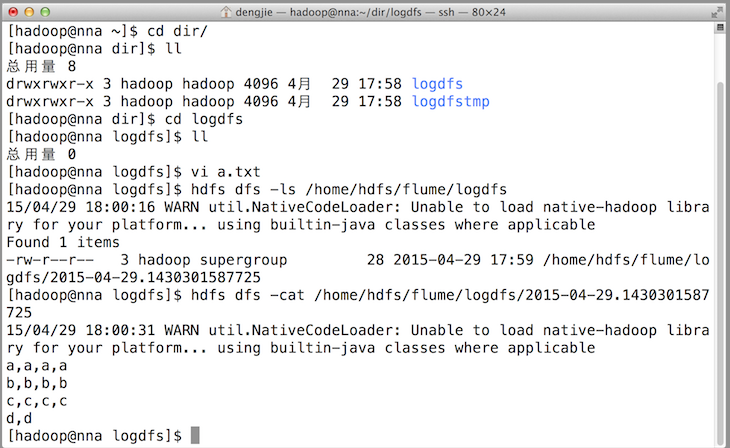
**3.3启动**

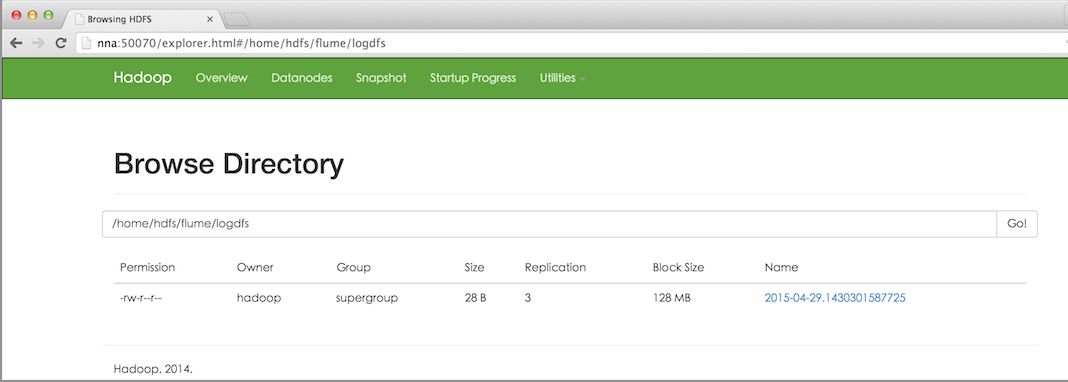
　　启动命令如下所示：

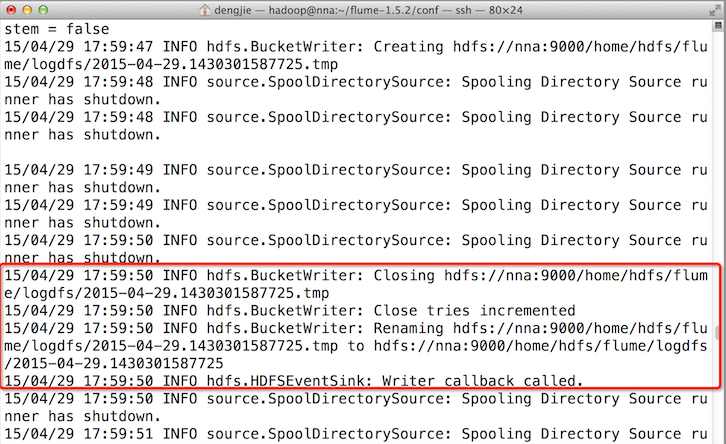
flume-ng agent -n agent1 -c conf -f flume-conf.properties -Dflume.root.logger=DEBUG,console

　　注：命令中的agent1表示配置文件中的Agent的Name，如配置文件中的agent1。flume-conf.properties表示配置文件所在配置，需填写准确的配置文件路径。

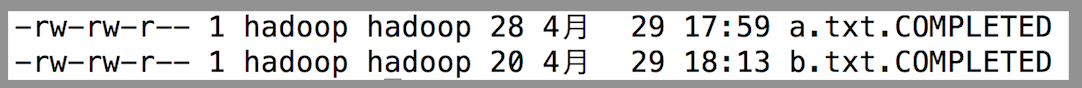
**3.4效果预览**





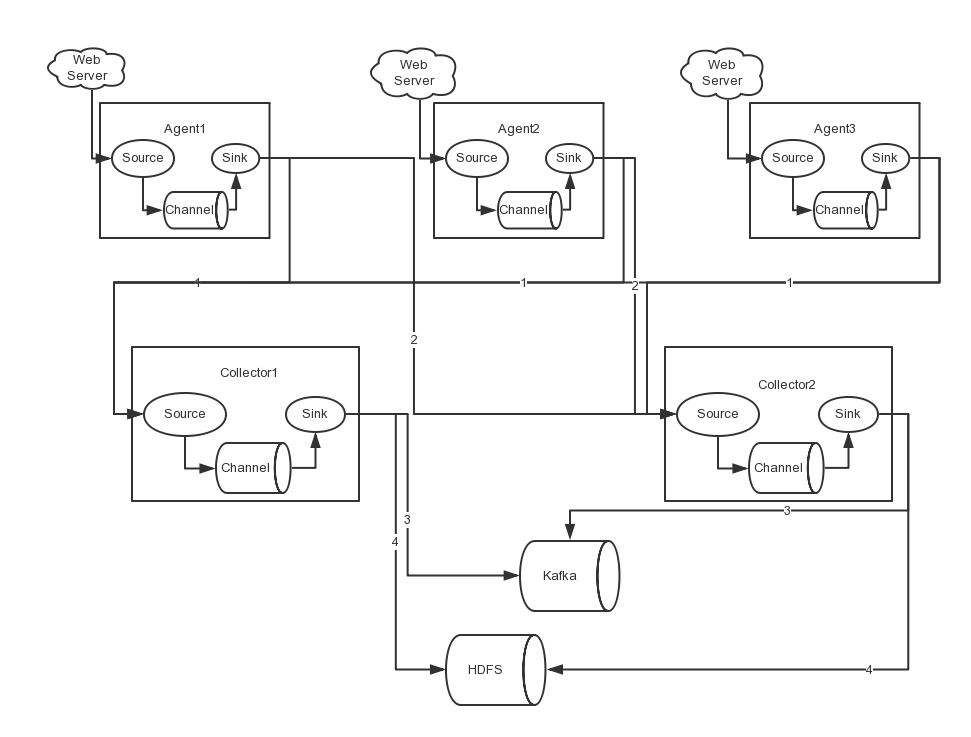


　　之后，成功上传后本地目的会被标记完成。如下图所示：



**4.高可用Flume NG搭建**

　　在完成单点的Flume NG搭建后，下面我们搭建一个高可用的Flume NG集群，架构图如下所示：



　　图中，我们可以看出，Flume的存储可以支持多种，这里只列举了HDFS和Kafka（如：存储最新的一周日志，并给Storm系统提供实时日志流）。

**4.1节点分配**

　　Flume的Agent和Collector分布如下表所示：

|  |  |  |
| --- | --- | --- |
| 名称 | HOST | 角色 |
| Agent1 | 10.211.55.14 | Web Server |
| Agent2 | 10.211.55.15 | Web Server |
| Agent3 | 10.211.55.16 | Web Server |
| Collector1 | 10.211.55.18 | AgentMstr1 |
| Collector2 | 10.211.55.19 | AgentMstr2 |

　　图中所示，Agent1，Agent2，Agent3数据分别流入到Collector1和Collector2，Flume NG本身提供了Failover机制，可以自动切换和恢复。在上图中，有3个产生日志服务器分布在不同的机房，要把所有的日志都收集到一个集群中存储。下面我们开发配置Flume NG集群

**4.2配置**

　　在下面单点Flume中，基本配置都完成了，我们只需要新添加两个配置文件，它们是flume-client.properties和flume-server.properties，其配置内容如下所示：

* flume-client.properties

[复制代码](javascript:void(0);)

#agent1 name

agent1.channels = c1

agent1.sources = r1

agent1.sinks = k1 k2

#set gruop

agent1.sinkgroups = g1

#set channel

agent1.channels.c1.type = memory

agent1.channels.c1.capacity = 1000

agent1.channels.c1.transactionCapacity = 100

agent1.sources.r1.channels = c1

agent1.sources.r1.type = exec

agent1.sources.r1.command = tail -F /home/hadoop/dir/logdfs/test.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

agent1.sinks.k1.hostname = nna

agent1.sinks.k1.port = 52020

# set sink2

agent1.sinks.k2.channel = c1

agent1.sinks.k2.type = avro

agent1.sinks.k2.hostname = nns

agent1.sinks.k2.port = 52020

#set sink group

agent1.sinkgroups.g1.sinks = k1 k2

#set failover

agent1.sinkgroups.g1.processor.type = failover

agent1.sinkgroups.g1.processor.priority.k1 = 10

agent1.sinkgroups.g1.processor.priority.k2 = 1

agent1.sinkgroups.g1.processor.maxpenalty = 10000

[复制代码](javascript:void(0);)

　　注：指定Collector的IP和Port。

* flume-server.properties

[复制代码](javascript:void(0);)

#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

a1.sources.r1.type = avro

a1.sources.r1.bind = nna

a1.sources.r1.port = 52020

a1.sources.r1.interceptors = i1

a1.sources.r1.interceptors.i1.type = static

a1.sources.r1.interceptors.i1.key = Collector

a1.sources.r1.interceptors.i1.value = NNA

a1.sources.r1.channels = c1

#set sink to hdfs

a1.sinks.k1.type=hdfs

a1.sinks.k1.hdfs.path=/home/hdfs/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

[复制代码](javascript:void(0);)

　　注：在另一台Collector节点上修改IP，如在NNS节点将绑定的对象有nna修改为nns。

**4.3启动**

　　在Agent节点上启动命令如下所示：

flume-ng agent -n agent1 -c conf -f flume-client.properties -Dflume.root.logger=DEBUG,console

　　注：命令中的agent1表示配置文件中的Agent的Name，如配置文件中的agent1。flume-client.properties表示配置文件所在配置，需填写准确的配置文件路径。

　　在Collector节点上启动命令如下所示：

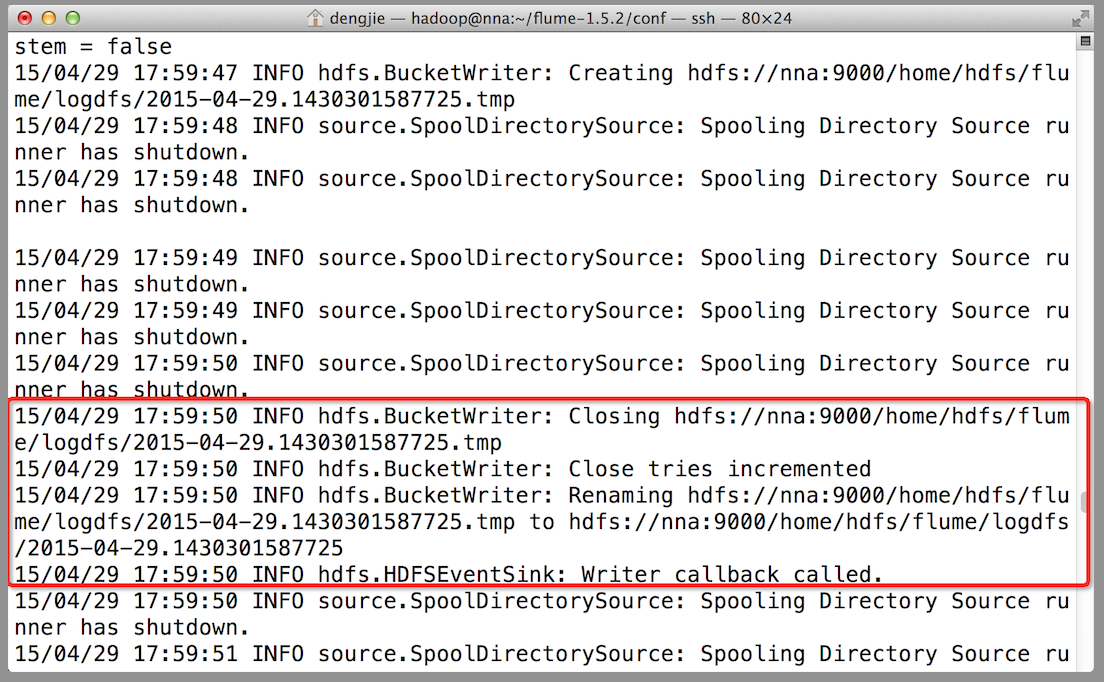
flume-ng agent -n a1 -c conf -f flume-server.properties -Dflume.root.logger=DEBUG,console

　　注：命令中的a1表示配置文件中的Agent的Name，如配置文件中的a1。flume-server.properties表示配置文件所在配置，需填写准确的配置文件路径。

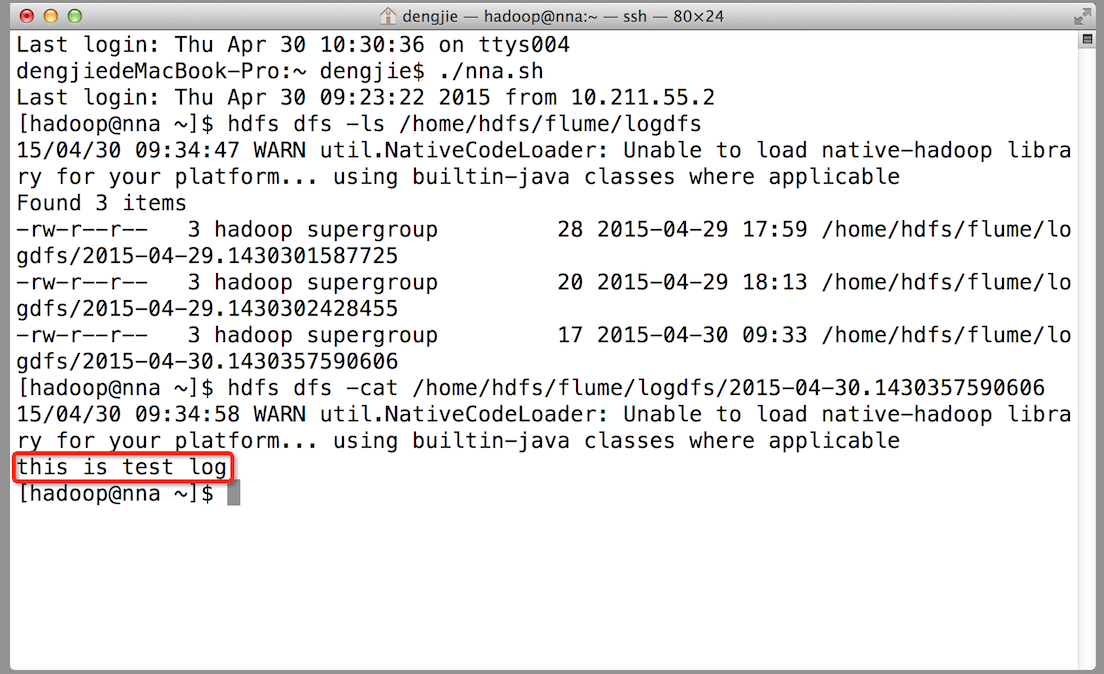
**5.Failover测试**

　　下面我们来测试下Flume NG集群的高可用（故障转移）。场景如下：我们在Agent1节点上传文件，由于我们配置Collector1的权重比Collector2大，所以Collector1优先采集并上传到存储系统。然后我们kill掉Collector1，此时有Collector2负责日志的采集上传工作，之后，我们手动恢复Collector1节点的Flume服务，再次在Agent1上次文件，发现Collector1恢复优先级别的采集工作。具体截图如下所示：

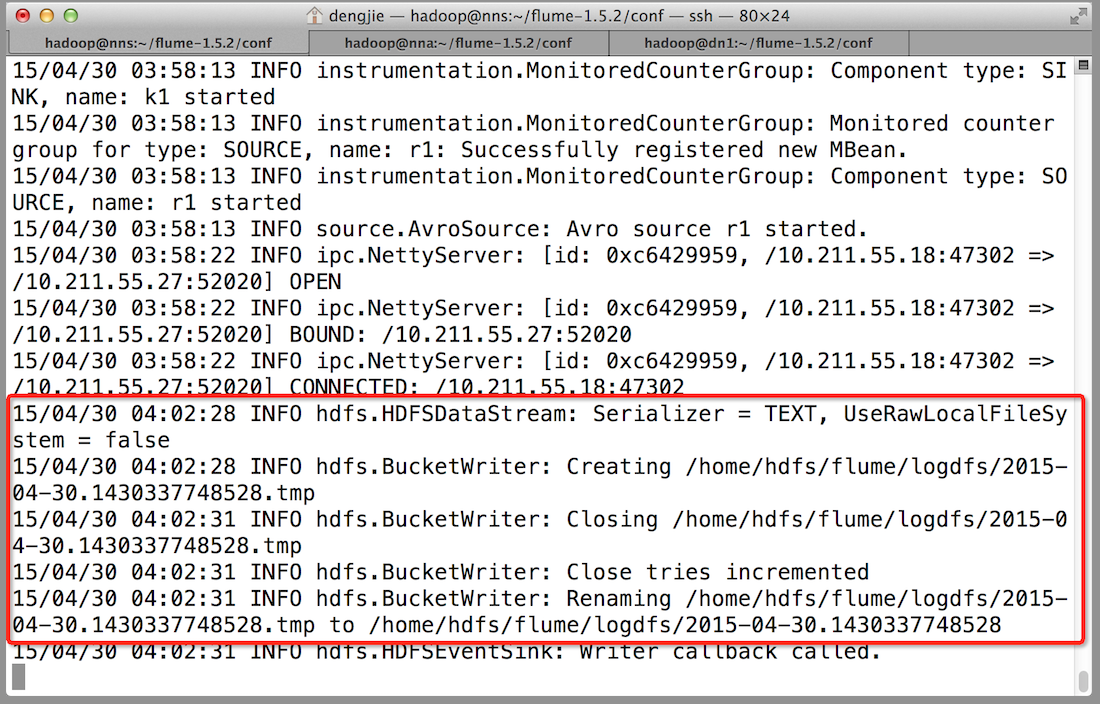
* Collector1优先上传



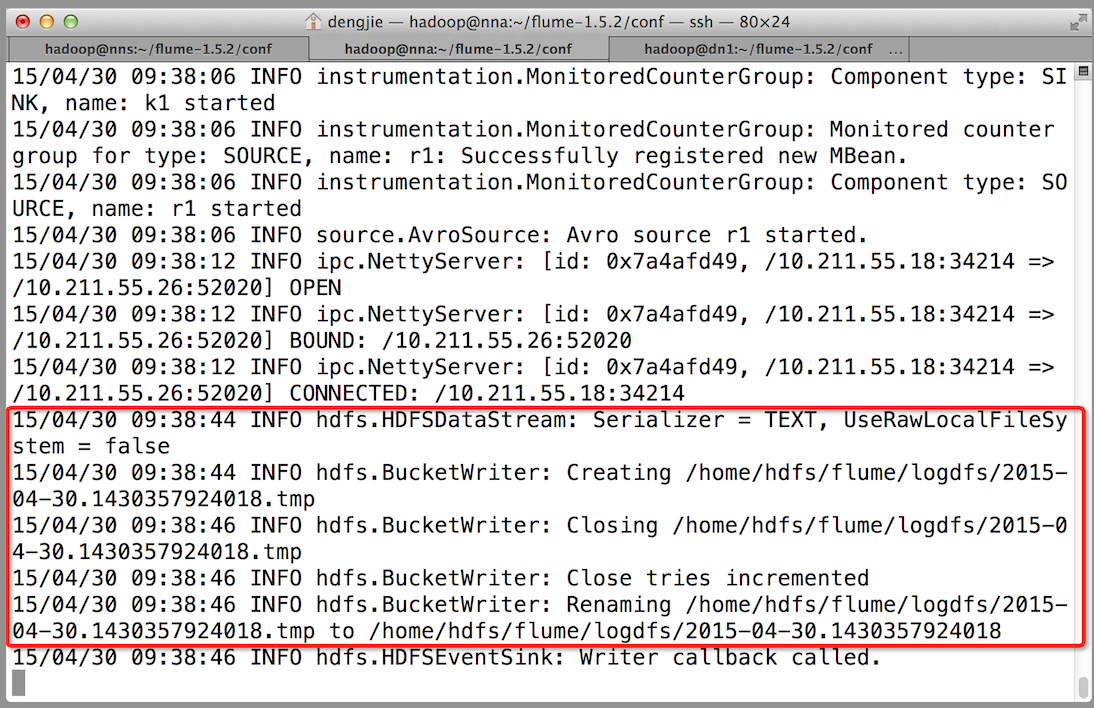
* HDFS集群中上传的log内容预览



* Collector1宕机，Collector2获取优先上传权限



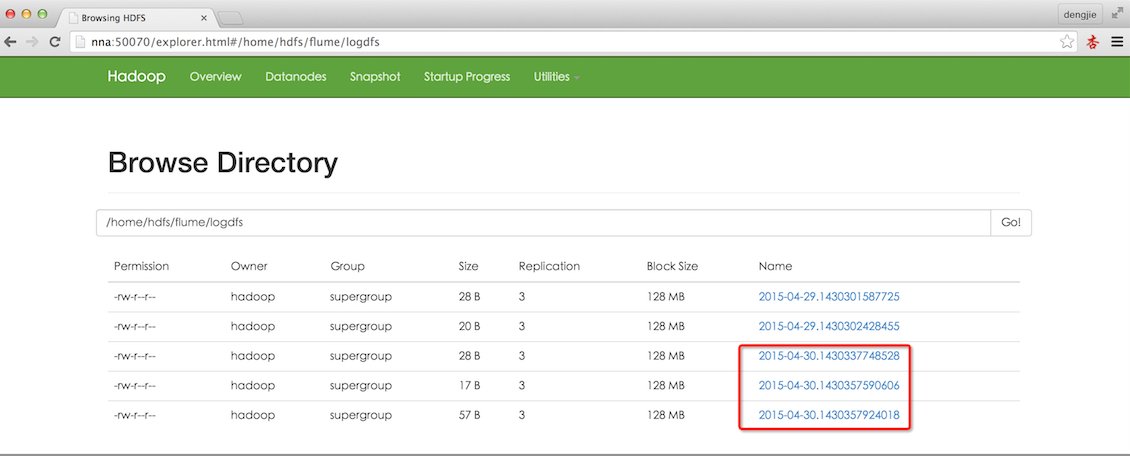
* 重启Collector1服务，Collector1重新获得优先上传的权限



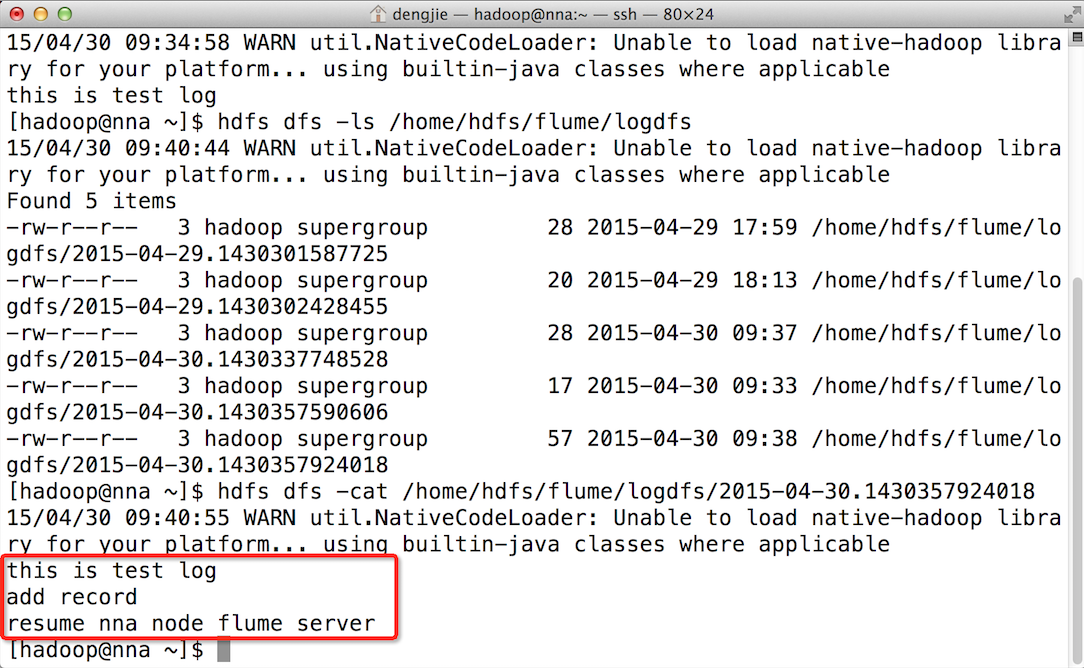
**6.截图预览**

　　下面为大家附上HDFS文件系统中的截图预览，如下图所示：

* HDFS文件系统中的文件预览



* 上传的文件内容预览



**7.总结**

　　在配置高可用的Flume NG时，需要注意一些事项。在Agent中需要绑定对应的Collector1和Collector2的IP和Port，另外，在配置Collector节点时，需要修改当前Flume节点的配置文件，Bind的IP（或HostName）为当前节点的IP（或HostName），最后，在启动的时候，指定配置文件中的Agent的Name和配置文件的路径，否则会出错。

**8.结束语**

　　这篇博客就和大家分享到这里，如果大家在研究学习的过程当中有什么问题，可以加群进行讨论或发送邮件给我，我会尽我所能为您解答，与君共勉！

联系方式：   
邮箱：smartdengjie@gmail.com   
QQ群（Hadoop - 董的博客2）：306184597 （已满）   
QQ群（Hadoop - 交流社区1）：424769183   
温馨提示：请大家加群的时候写上加群理由（姓名＋公司/学校），方便管理员审核，谢谢！

**热爱生活，享受编程，与君共勉！**