 nginx的日志文件没有rotate功能。如果你不处理，日志文件将变得越来越大，还好我们可以写一个nginx日志切割脚本来自动切割日志文件。  
第一步就是重命名日志文件，不用担心重命名后nginx找不到日志文件而丢失日志。在你未重新打开原名字的日志文件前，nginx还是会向你重命名的文件写日志，linux是靠文件描述符而不是文件名定位文件。  
第二步向nginx主进程发送USR1信号。  
nginx主进程接到信号后会从配置文件中读取日志文件名称，重新打开日志文件(以配置文件中的日志名称命名)，并以工作进程的用户作为日志文件的所有者。  
重新打开日志文件后，nginx主进程会关闭重名的日志文件并通知工作进程使用新打开的日志文件。  
工作进程立刻打开新的日志文件并关闭重名名的日志文件。  
然后你就可以处理旧的日志文件了

**Nginx\_log\_division.sh代码：**

 Shell代码

#!/bin/bash

#设置日志文件存放目录

logs\_path="/usr/local/nginx/nginxlog/"

#设置pid文件

pid\_path="/usr/local/nginx/nginx-1.7.3/logs/nginx.pid"

#日志文件

filepath=${logs\_path}"access.log"

# Source function library.

#重命名日志文件

mv ${logs\_path}access.log ${logs\_path}access\_$(date -d '-1 day' '+%Y-%m-%d').log

#向nginx主进程发信号重新打开日志

kill -USR1 `cat ${pid\_path}`

**Oracle\_log\_division.sh代码：**

 Shell代码

#!/bin/bash

#设置日志文件存放目录

logs\_path="/usr/local/nginx/nginxlog/"

#设置pid文件

pid\_path="/usr/local/nginx/nginx-1.7.3/logs/nginx.pid"

#日志文件

filepath=${logs\_path}"access.log"

# Source function library.

#重命名日志文件

mv ${logs\_path}access.log ${logs\_path}access\_$(date -d '-1 day' '+%Y-%m-%d').log

#向nginx主进程发信号重新打开日志

kill -USR1 `cat ${pid\_path}`

**flumeng配置：**

Xml代码  [IMG_256](https://www.cnblogs.com/tonychai/p/javascript:void())

# A single-node Flume configuration

# Name the components on this agent

agent1.sources = source1

agent1.sinks = sink1

agent1.channels = channel1

# Describe/configure source1

agent1.sources.source1.type = exec

agent1.sources.source1.command = tail -n +0 -F /logs/access.log

agent1.sources.source1.channels = channel1

# Describe sink1

agent1.sinks.sink1.type = file\_roll

agent1.sinks.sink1.sink.directory=/var/log/data

# Use a channel which buffers events in memory

agent1.channels.channel1.type = file

agent1.channels.channel1.checkpointDir=/var/checkpoint

agent1.channels.channel1.dataDirs=/var/tmp

agent1.channels.channel1.capacity = 1000

agent1.channels.channel1.transactionCapactiy = 100

# Bind the source and sink to the channel

agent1.sources.source1.channels = channel1

agent1.sinks.sink1.channel = channel1

flume-ng agent -n a1 -c conf -f conf/example.file -Dflume.root.logger=DEBUG,console

参数说明：

-n 指定agent名称(与配置文件中代理的名字相同)

-c 指定flume中配置文件的目录

-f 指定配置文件

-Dflume.root.logger=DEBUG,console 设置日志等级

agent1.channels.channel1.transactionCapactiy = 100 sink端收集到100条数据后提交事务

Sink.batchsize sink端一次性从channel取的数据量(sink和channel使用一个双向队列维护事务，进行回滚。LinkedBlockingQuene)