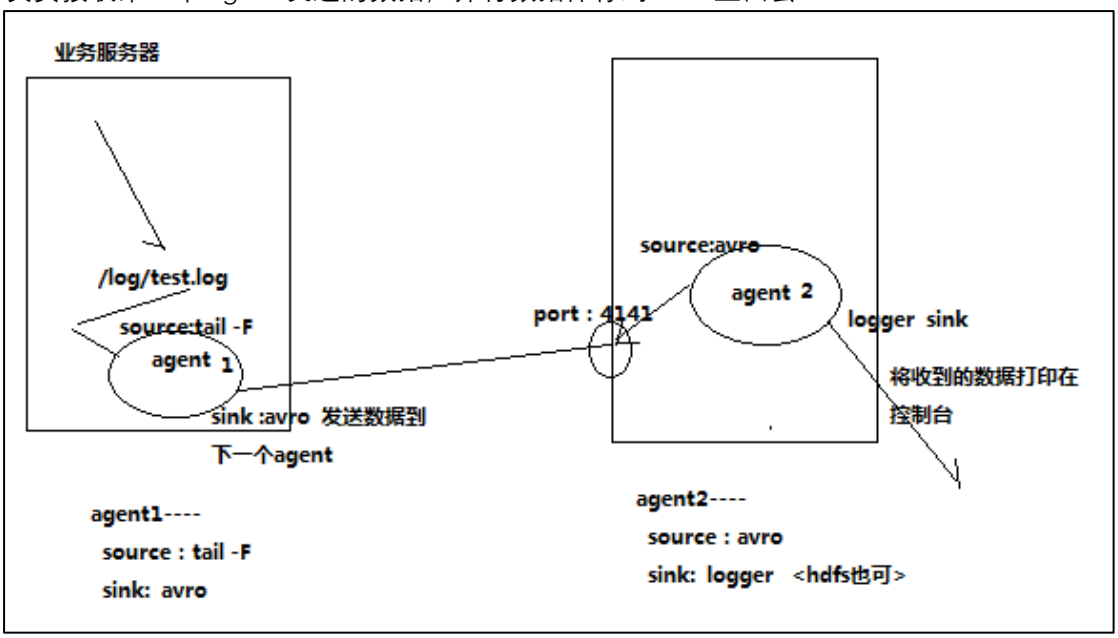
两个 agent 级联

需求分析：

第一个 agent 负责收集文件当中的数据，通过网络发送到第二个 agent 当中去，第二个 agent 负责接收第一个 agent 发送的数据，并将数据保存到 hdfs 上面去



第一步：node02 安装 flume

将 node03 机器上面解压后的 flume 文件夹拷贝到 node02 机器上面去

```
cd  /export/servers
scp -r apache-flume-1.6.0-cdh5.14.0-bin/ node02:$PWD
```

```
[root@node03 servers]# cd /export/servers/
[root@node03 servers]# scp -r apache-flume-1.6.0-cdh5.14.0-bin/ node02:$PWD
```

第二步：node02 配置 flume 配置文件

在 node02 机器配置我们的 flume

```
cd /export/servers/apache-flume-1.6.0-cdh5.14.0-bin/conf
vim tail-avro-avro-logger.conf
```

```
[root@node02 servers]# cd /export/servers/apache-flume-1.6.0-cdh5.14.0-bin/conf
You have new mail in /var/spool/mail/root
[root@node02 conf]# ll
total 24
-rw-r--r--. 1 root root 1661 Jul 12 10:44 flume-conf.properties.template
-rw-r--r--. 1 root root 1455 Jul 12 10:44 flume-env.ps1.template
-rw-r--r--. 1 root root 1568 Jul 12 10:44 flume-env.sh
-rw-r--r--. 1 root root 1565 Jul 12 10:44 flume-env.sh.template
-rw-r--r--. 1 root root 3107 Jul 12 10:44 log4j.properties
-rw-r--r--. 1 root root  323 Jul 12 10:44 netcat-logger.conf
[root@node02 conf]# vim tail-avro-avro-logger.conf
```

```
##################
# 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 /export/servers/taillogs/access_log
a1.sources.r1.channels = c1
# Describe the sink
##sink avro is a sender
a1.sinks = k1
a1.sinks.k1.type = avro
a1.sinks.k1.channel = c1
a1.sinks.k1.hostname = 192.168.52.120
a1.sinks.k1.port = 4141
a1.sinks.k1.batch-size = 10
# 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
```

第三步：node02 开发定脚本文件往写入数据

```
mkdir -p /export/servers/shells/
cd    /export/servers/shells/
vim tail-file.sh
```

```
#!/bin/bash
```

```
while true
do
  date >> /export/servers/taillogs/access_log;
    sleep 0.5;
done
```

创建文件夹

```
mkdir -p /export/servers/taillogs
```

第四步：node03 开发 flume 配置文件

在 node03 机器上开发 flume 的配置文件

```
cd /export/servers/apache-flume-1.6.0-cdh5.14.0-bin/conf
vim avro-hdfs.conf
```

```
# Name the components on this agent
a1.sources = r1
a1.sinks = k1
a1.channels = c1
# Describe/configure the source
##source avro is a receiver
a1.sources.r1.type = avro
a1.sources.r1.channels = c1
a1.sources.r1.bind = 192.168.52.120
a1.sources.r1.port = 4141
# Describe the sink
a1.sinks.k1.type = hdfs
a1.sinks.k1.hdfs.path = hdfs://node01:8020/avro/hdfs/%y-%m-%d/%H%M/
a1.sinks.k1.hdfs.filePrefix = events-
a1.sinks.k1.hdfs.round = true
a1.sinks.k1.hdfs.roundValue = 10
a1.sinks.k1.hdfs.roundUnit = minute
a1.sinks.k1.hdfs.rollInterval = 3
a1.sinks.k1.hdfs.rollSize = 20
a1.sinks.k1.hdfs.rollCount = 5
a1.sinks.k1.hdfs.batchSize = 1
a1.sinks.k1.hdfs.useLocalTimeStamp = true
# Sequencefile,DataStream
a1.sinks.k1.hdfs.fileType = DataStream
# 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
```

第五步：顺序启动

node03 机器启动 flume 进程

```
cd /export/servers/apache-flume-1.6.0-cdh5.14.0-bin
bin/flume-ng agent -c conf -f conf/avro-hdfs.conf -n a1 -Dflume.root.logger=INFO,console
```

node02 机器启动 flume 进程

```
cd /export/servers/apache-flume-1.6.0-cdh5.14.0-bin/
bin/flume-ng agent -c conf -f conf/tail-avro-avro-logger.conf -n a1 -Dflume.root.logger=INFO,console
```

node02 机器启 shell 脚本生成文件

```
cd  /export/servers/shells
sh tail-file.sh
```