**nginx + flume + kafka + hdfs**

安装nginx:

* 1: 安装：缺少GCC编辑器：

yum -y install pcre\*

yum -y install pcre\* openssl\* gcc make

yum -y install gcc

yum -y install gcc-c++

yum -y install make

* 2:拷贝 nginx 安装包到 nginx 服务器中并解压。进入nginx目录。
* 3: 编译nginx:

进入nginx目录。

3.1： ./configure

3.2:  make

3.3: make install(nginx 安装在本机里，/usr/local/nginx)

4nginx的启动，停止，重新加载命令。

启动nginx

Cd /usr/local/nginx/sbin

./nginx

停止nginx

./nginx -s stop

重新加载配置文件：

./nginx -s reload

|  |
| --- |
| 如果发现80端口被占用。  #fuser -n tcp 80 查看端口占用情况  not found  [root@namenode-1 sbin]# kill -9 6677  杀死所有进程。 |

配置nginx的conf 目录下的nginx.conf文件。

|  |  |
| --- | --- |
| nginx 日志采集  配置如下：  user root;  worker\_processes 1;  #error\_log logs/error.log;  #error\_log logs/error.log notice;  #error\_log logs/error.log info;  #pid logs/nginx.pid;  events {  worker\_connections 1024;  }  http {  include mime.types;  default\_type application/octet-stream;  log\_format main '$remote\_addr,$remote\_user,[$time\_local],"$request" '  '$status,$body\_bytes\_sent,"$http\_referer" '  '"$http\_user\_agent","$http\_x\_forwarded\_for"'  '$msec,$http\_host,$request\_uri';  #access\_log logs/access.log main;  sendfile on;  #tcp\_nopush on;  #keepalive\_timeout 0;  keepalive\_timeout 65;  #gzip on;    server {  listen 80;  server\_name localhost;  #charset koi8-r;  #access\_log logs/host.access.log main;  location / {  access\_log /home/huaqiang/nginx/access.log main;  root /home/huaqiang/www ;    }    }  } |  |

Flume配置

|  |
| --- |
| flume 配置kafka sink  a1.sources = r1  a1.sinks = k1 k2  a1.channels = c1  # Describe/configure the source  a1.sources.r1.type = exec  a1.sources.r1.command = tail -F /home/flumelog.log  a1.sources.r1.shell = /bin/bash -c  a1.sinks.k1.type = org.apache.flume.sink.kafka.KafkaSink  a1.sinks.k1.kafka.topic = dahuang  a1.sinks.k1.kafka.bootstrap.servers = namenode-1:9092  a1.sinks.k1.kafka.flumeBatchSize = 20  a1.sinks.k1.kafka.producer.acks = all  a1.sinks.k1.kafka.producer.linger.ms = 1  a1.sinks.k1.kafka.producer.compression.type = snappy  a1.sinks.k2.type = hdfs  a1.sinks.k2.hdfs.path = hdfs://namenode-1:8020/flume-access/%Y%m%d/%H  #上传文件的前缀  a1.sinks.k2.hdfs.filePrefix = flume2-  #是否按照时间滚动文件夹  a1.sinks.k2.hdfs.round = true  #多少时间单位创建一个新的文件夹  a1.sinks.k2.hdfs.roundValue = 1  #重新定义时间单位  a1.sinks.k2.hdfs.roundUnit = hour  #是否使用本地时间戳  a1.sinks.k2.hdfs.useLocalTimeStamp = true  #积攒多少个Event才flush到HDFS一次  a1.sinks.k2.hdfs.batchSize = 100  #设置文件类型，可支持压缩  a1.sinks.k2.hdfs.fileType = DataStream  #多久生成一个新的文件  a1.sinks.k2.hdfs.rollInterval = 600  #设置每个文件的滚动大小大概是128M  a1.sinks.k2.hdfs.rollSize = 134217700  #文件的滚动与Event数量无关  a1.sinks.k2.hdfs.rollCount = 0  #最小冗余数  a1.sinks.k2.hdfs.minBlockReplicas = 1  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  a1.sinks.k2.channel = c1 |