# Elk分布式日志系统

# 简介

Logstash+ElasticSearch+Kibana4+redis

Logstash：负责收集日志

ElasticSearch：负责检索分析日志

Kibana：负责展示ElasticSearch

Redis：做为队列传输

官网为：https://www.elastic.co/products

Elasticsearch:

https://www.elastic.co/downloads/elasticsearch  
<https://download.elastic.co/elasticsearch/elasticsearch/elasticsearch-2.3.0.tar.gz>

logstash:

https://www.elastic.co/downloads/logstash

<https://download.elastic.co/logstash/logstash/logstash-2.3.0.tar.gz>

kibana:

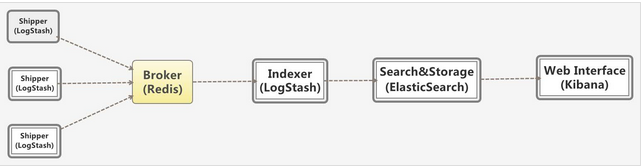
https://www.elastic.co/products/kibana

<https://download.elastic.co/kibana/kibana/kibana-4.5.0-linux-x64.tar.gz>

环境：centos6.5

服务器ip地址：10.10.57.87 Logstash+ElasticSearch+Kibana4

Redis 地址：clink-log.a9d2nt.0001.cnn1.cache.amazonaws.com.cn

客户端ip地址：clink所有主机 Logstash

# 安装

## 服务器配置10.10.57.87

下载软件包：

wget https://download.elastic.co/elasticsearch/elasticsearch/elasticsearch-2.3.0.tar.gz

wget https://download.elastic.co/logstash/logstash/logstash-2.3.0.tar.gz

wget https://download.elastic.co/kibana/kibana/kibana-4.5.0-linux-x64.tar.gz

无需下载，使用aws的redis

wget <http://download.redis.io/releases/redis-3.0.4.tar.gz>

依赖安装：

yum -y install vim unix2dos wget curl curl-devel expect expect-devel

yum -y install glibc glib2 libgcc glibc-devel glib2-devel java-1.8.0-openjdk

yum -y install m4 automake autoconf cmake cpp

yum -y install gcc-c++ libstdc++ libstdc++-devel libstdc++-docs

无需安装，使用aws的redis

#tar –xvzf redis-3.0.4.tar.gz

#cd redis-3.0.4

#make

#make install

#cd utils

#./install\_server.sh

这样redis就安装好了

#which redis-server在/usr/local/bin/redis-server

配置文件放在/etc/redis/6379.conf文件

#redis-cli

>set aa aavalue

>get aa 即可测试。

#cd

## 安装elasticsearch

#mkdir /usr/local/elk

useradd elasticsearch ##禁止root启动需要创建普通用户启动

#tar –xvzf elasticsearch-1.7.1.tar.gz –C /usr/local/elk/

## 安装logstash

#tar –xvzf logstash-1.5.4.tar.gz –C /usr/local/elk/

#### 编写监控logstash脚本，放在crontab里10分钟监控一次

touch /usr/local/bin/logstash\_check.sh

echo -e '#!/bin/bash \n\nvalue=`ps -ef | grep tcp2redis.conf | grep -v "grep"|wc -l` \n\nif [ $value -eq 0 ] \nthen \n /usr/local/elk/logstash-2.3.0/bin/logstash -f /usr/local/elk/logstash-2.3.0/conf/tcp2redis.conf & \nelse \n echo "logstash is ok" \nfi'> /usr/local/bin/logstash\_check.sh

chmod +x /usr/local/bin/logstash\_check.sh

echo "\*/10 \* \* \* \* /bin/bash /usr/local/bin/logstash\_check.sh" >> /var/spool/cron/root

#### 安装脚本

cat install\_logstash.sh

#!/bin/bash

mkdir /usr/local/elk

cd /usr/local/elk

wget -c http://10.10.16.29/test/logstash-2.3.0.tar.gz

tar -zxf logstash-2.3.0.tar.gz

mkdir /usr/local/elk/logstash-2.3.0/conf

touch /usr/local/elk/logstash-2.3.0/conf/tcp2redis.conf

echo -e "input {\n tcp {\n port => 4560\n codec => json\_lines\n }\n}\n \noutput {\n redis {\n host => \"clink-log.a9d2nt.0001.cnn1.cache.amazonaws.com.cn\"\n key => logstash\n data\_type => list\n }\n}" > /usr/local/elk/logstash-2.3.0/conf/tcp2redis.conf

/usr/local/elk/logstash-2.3.0/bin/logstash -f /usr/local/elk/logstash-2.3.0/conf/tcp2redis.conf &

echo "##elk##" >> /etc/rc.local

echo ". /etc/profile" >> /etc/rc.local

echo "/usr/local/elk/logstash-2.3.0/bin/logstash -f /usr/local/elk/logstash-2.3.0/conf/tcp2redis.conf &" >> /etc/rc.local

echo "/bin/sh /usr/local/bin/zabbix/restart\_zabbix.sh" >> /etc/rc.local

rm -rf /usr/local/elk/logstash-2.3.0.tar.gz

## 安装kibana

#tar –xvzf kibana-4.1.2-linux-x64.tar.gz –C /usr/local/elk/

程序安装全部完成。

由于elasticsearch和logstash是安装在一台机器上所以elasticsearch默认配置即可。

## 设置开机启动

##### elasticsearch #####

. /etc/profile

su - elasticsearch -c "/usr/local/elk/elasticsearch-2.3.0/bin/elasticsearch -d"

#####kibana#####

/usr/local/elk/kibana-4.5.0-linux-x64/bin/kibana -l /usr/local/elk/kibana-4.5.0-linux-x64/logs/kibana.log &

#####logstash#####

/usr/local/elk/logstash-2.3.0/bin/logstash -f /usr/local/elk/logstash-2.3.0/conf/logstash\_indexer.conf &

## 安装elasticsearch插件

方法1：

./bin/plugin -install lukas-vlcek/bigdesk 集群监控插件

./bin/plugin -install mobz/elasticsearch-head 集群管理插件

2.运行es

3.打开http://localhost:9200/\_plugin/bigdesk/

当然，也可以直接下载源码运行index.html

方法2：

1.https://github.com/lukas-vlcek/bigdesk下载zip 解压

2.建立elasticsearch-1.0.0\plugins\bigdesk\\_site文件

3.将解压后的bigdesk-master文件夹下的文件copy到\_site

4.运行es

5.打开http://localhost:9200/\_plugin/bigdesk/

## 启动elasticsearch

/usr/local/elasticsearch/bin/elasticsearch -d （以deamon方式启动elasticsearch）

/usr/local/elasticsearch/bin/elasticsearch -Xmx2g -Xms2g -Des.index.storage.type=memory -d

执行curl可以看到以下内容证明OK

# curl 172.16.203.76:9200

{

"status" : 200,

"name" : "Torgo of Mekka",

"cluster\_name" : "elasticsearch",

"version" : {

"number" : "1.7.1",

"build\_hash" : "b88f43fc40b0bcd7f173a1f9ee2e97816de80b19",

"build\_timestamp" : "2015-07-29T09:54:16Z",

"build\_snapshot" : false,

"lucene\_version" : "4.10.4"

},

"tagline" : "You Know, for Search"

}

创建logstash的etc配置文件目录

mkdir /usr/local/logstash/etc

编写index配置文件

cat /usr/local/logstash/etc/logstash\_indexer.conf

input {

redis {

host => "localhost"

data\_type => "list"

key => "logstash:redis"

type => "redis-input"

}

}

filter {

grok {

match => [

"message", "%{WORD:http\_host} %{URIHOST:api\_domain} %{IP:inner\_ip} %{IP:lvs\_ip} \[%{HTTPDATE:timestamp}\] \"%{WORD:http\_verb} %{URIPATH:baseurl}(?:\?%{NOTSPACE:request}|) HTTP/%{NUMBER:http\_version}\" (?:-|%{NOTSPACE:request}) %{NUMBER:http\_status\_code} (?:%{NUMBER:bytes\_read}|-) %{QS:referrer} %{QS:agent} %{NUMBER:time\_duration:float} (?:%{NUMBER:time\_backend\_response:float}|-)"

]

}

kv {

prefix => "request."

field\_split => "&"

source => "request"

}

urldecode {

all\_fields => true

}

date {

type => "log-date"

match => ["timestamp" , "dd/MMM/YYYY:HH:mm:ss Z"]

}

}

output {

elasticsearch {

embedded => false

protocol => "http"

host => "localhost"

port => "9200"

index => "access-%{+YYYY.MM.dd}"

}

}

启动logstash

nohup ./bin/logstash -f /usr/local/logstash-1.5.4/etc/logstash\_agent.conf &

启动kibana

nohup /usr/local/kibana-4.0.2-linux-x86/bin/kibana &

## 客户端配置

#tar –xvzf logstash-1.5.4.tar.gz

#cp –a logstash-1.5.4 /usr/local

#cd /usr/local

#ln –s logstash-1.5.4 logstash

创建etc配置文件目录

mkdir /usr/local/logstash/etc

以监控apache日志为例配置：

编写agent配置文件

input {

tcp {

port => 5514

type => syslog

}

udp {

port => 5514

type => syslog

}

}

filter {

grok {

match => [ "message", "%{SYSLOGBASE}" ]

}

date {

match => ["timestamp" , "dd/MMM/YYYY:HH:mm:ss Z"]

}

}

output {

redis {

host => "172.16.203.76"

data\_type => "list"

key => "logstash:redis"

port => 6379

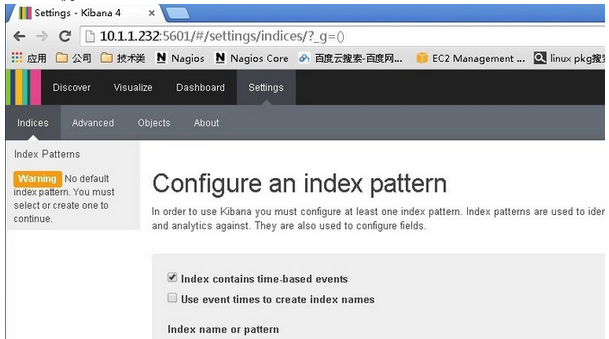
}

}

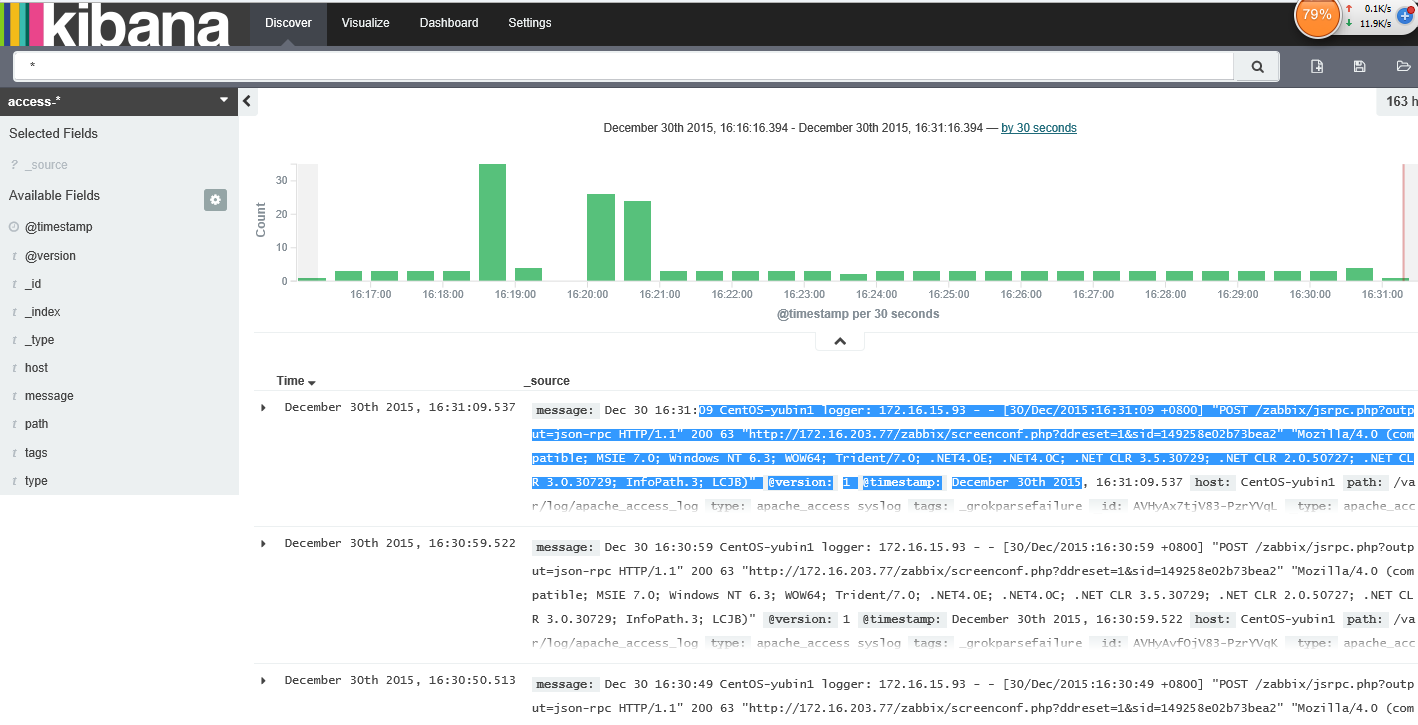
启动

nohup ./bin/logstash -f /usr/local/logstash-1.5.4/etc/logstash\_agent.conf &

打开浏览器<http://172.16.203.76:5601>进入



进来之后他要你创建一个默认索引，可是怎么办没有create啊，下面会灰色的,这个如果你装完kibana你那个日志文件要是一直没有新的日志产生呢这里就一直是灰的，这个时候你只需要去访问一下你的网站，然后就行了



# 三、后续研究

1.         Logstash索引的定时清理

2.         Kibana的深度使用

3.         Logstash与zabbix等的结合及报警

4.         Elasticsearch的集群部署

5.         Elasticsearch的管理插件

6.         不同日志格式的采集与解析

# logstash配置语法

## input配置

file：从文件获取

syslog：从syslog协议514端口获取

redis：从redis获取

stdin：从标准输入获取

配置事例：从/var/log/下读取日志

input {

file {

path => ["/var/log/\*.log", "/var/log/message"]

type => "system"

start\_position => "beginning"

}

}

## codec编码插件

## filter过滤器插件

这个目录下可以自定义正则变量

./vendor/bundle/jruby/1.9/gems/logstash-patterns-core-2.0.2/patterns/

logstash/lib/logstash/filters/grok.rb

cat一下看看

## output配置

# 五elasticsearch常用操作命令

## 1、删除1月份的所有索引文件

curl -XDELETE 'http://127.0.0.1:9200/access-2016.01.\*'

## 2、查询索引文件 \_search?q=这个后面可以匹配条件

curl -XGET <http://127.0.0.1:9200/access-2016.03.10/_search?q=_id:AVNfY8utNswt_GKBzGxl>

### 查询索引为megacorp类型为employee的所有数据

curl -XGET 'http://127.0.0.1:9200/megacorp/employee/\_search

### 美化输出pretty

curl -XGET 'http://localhost:9200/website/blog/123?pretty'

### 检索文档的一部分

curl -XGET 'http://localhost:9200/website/blog/123?\_source=title,text'

只想得到\_source字段，其他都不要

curl -XGET 'http://localhost:9200/website/blog/123/\_source'

### 2.1、使用DSL语句查询，匹配last\_name=Smith的内容

curl -XGET 'http://127.0.0.1:9200/megacorp/employee/\_search' -d '

{

"query" : {

"match" : {

"last\_name" : "Smith"

}

}

}

'

### 2.2、****过滤器(filter)****

### 2.3、全文搜索

### 2.4、短语搜索（match\_phrase）

{

"query" : {

"match\_phrase" : {

"about" : "rock climbing"

}

}

}

### 2.5、高亮我们的搜索

{

"query" : {

"match\_phrase" : {

"about" : "rock climbing"

}

},

"highlight": {

"fields" : {

"about" : {}

}

}

}

## 3、插入索引为megacorp类型为employee的内容

curl -XPUT 'http://127.0.0.1:9200/megacorp/employee/3' -d '

{

"first\_name" : "Douglas",

"last\_name" : "Fir",

"age" : 35,

"about": "I like to build cabinets",

"interests": [ "forestry" ]

}

### 'id自增长 使用XPOST 后面不需要指定

curl -XPOST 'http://127.0.0.1:9200/megacorp/employee' -d '

{

"first\_name" : "DouglasSS",

"last\_name" : "FirR",

"age" : 36,

"about": "I like to build cabinets",

"interests": [ "forestry" ]

}

'