**安装--**elasticsearch

官方文档：<https://www.elastic.co/guide/cn/elasticsearch/guide/current/index.html>

yum -y install java

rpm --import https://packages.elastic.co/GPG-KEY-elasticsearch

vim /etc/yum.repos.d/elasticsearch.repo

[elasticsearch-2.x]  
name=Elasticsearch repository for 2.x packages  
baseurl=http://packages.elastic.co/elasticsearch/2.x/centos  
gpgcheck=1  
gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch  
enabled=1

mkdir -p /data/es-data

chown elasticsearch:elasticsearch /data/es-data/

yum install -y elasticsearch

cd /etc/elasticsearch/

rpm -ql elasticsearch

vim elasticsearch.yml

[root@localhost /etc/elasticsearch]# grep "^[a-z]" elasticsearch.yml

cluster.name: myes

node.name: linux-node1

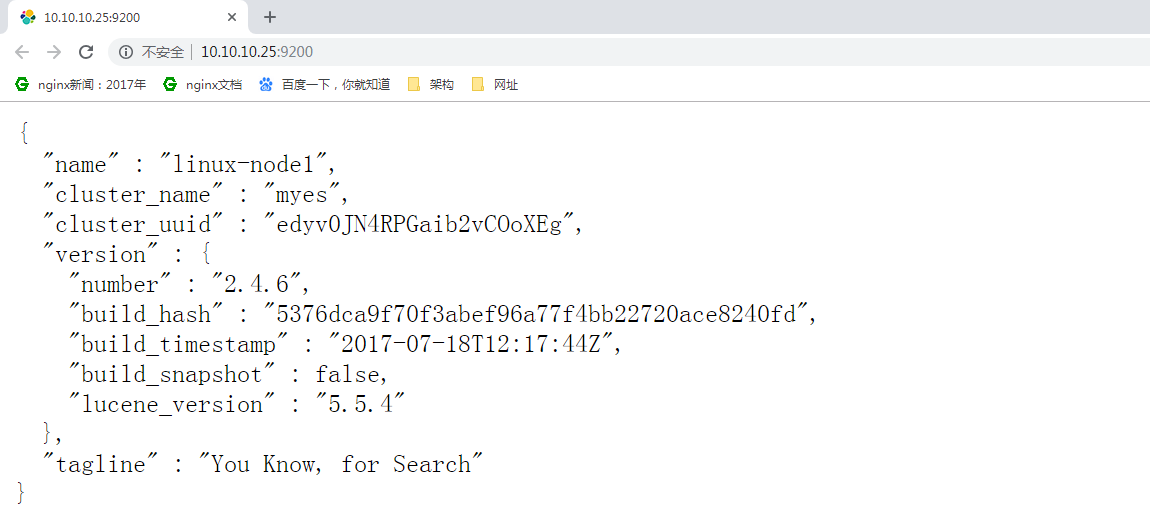
path.data: /data/es-data

path.logs: /var/log/elasticsearch

bootstrap.memory\_lock: true

network.host: 10.10.10.25

http.port: 9200



命令路径：/usr/share/elasticsearch/bin/plugin

插件路径：/usr/share/elasticsearch/plugins

安装 marvel-agent插件

/usr/share/elasticsearch/bin/plugin install marvel-agent

安装head插件

github地址：<https://github.com/mobz/elasticsearch-head/>

安装：/usr/share/elasticsearch/bin/plugin install mobz/elasticsearch-head

or：

wget <https://github.com/mobz/elasticsearch-head/archive/master.zip>

unzip master.zip

mv elasticsearch-head-master head

chown -R elasticsearch:elasticsearch /usr/share/elasticsearch/plugins

**访问：****http://10.10.10.25:9200/\_plugin/head/**



目前只有一个master节点：健康值为5/10

安装bigdesk 目前github支持elasticsearch的最高版本为1.30-1.3x

github地址：https://github.com/lukas-vlcek/bigdesk

安装方式： /usr/share/elasticsearch/bin/plugin install lukas-vlcek/bigdesk

安装kopf：

/usr/share/elasticsearch/bin/plugin install lmenezes/elasticsearch-kopf

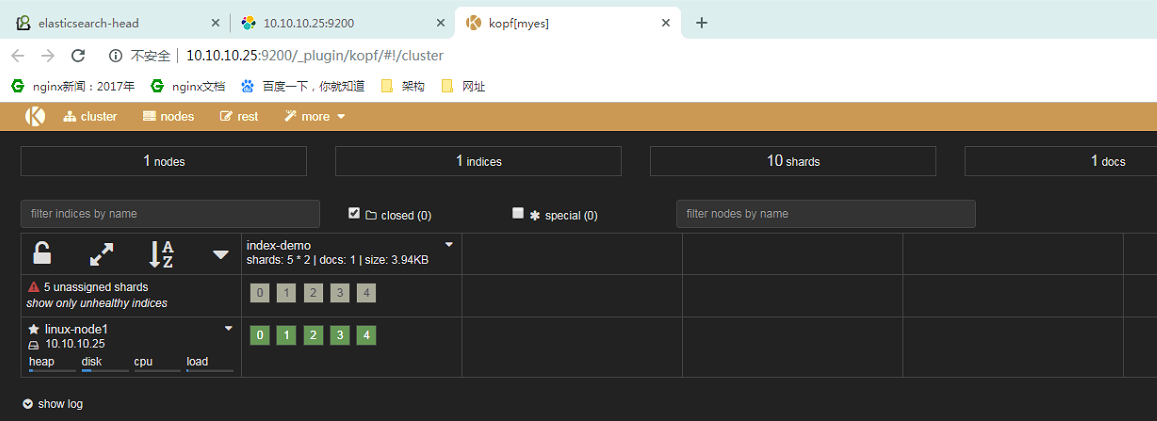
or

wget <https://github.com/lmenezes/elasticsearch-kopf/archive/master.zip>

unzip master.zip

mv elasticsearch-kopf-master kopf

访问：http://10.10.10.25:9200/\_plugin/kopf/



slave：安装===================================================================================================

[root@localhost /etc/elasticsearch]# grep "^[a-Z]" /etc/elasticsearch/elasticsearch.yml

cluster.name: myes

node.name: linux-node2

path.data: /data/es-data

path.logs: /var/log/elasticsearch

bootstrap.memory\_lock: true

network.host: 10.10.10.26

http.port: 9200

discovery.zen.ping.unicast.hosts: ["10.10.10.25", "10.10.10.26"]

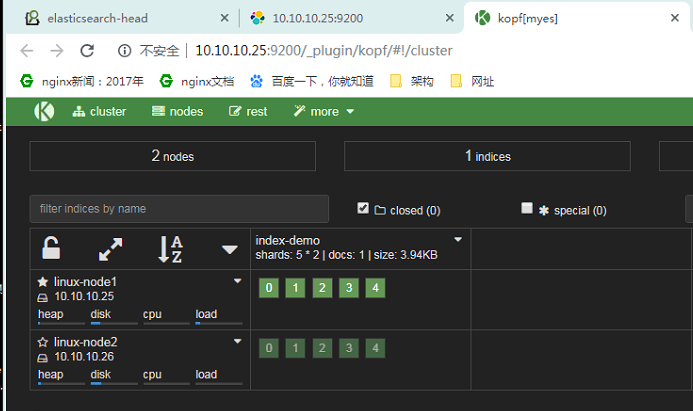
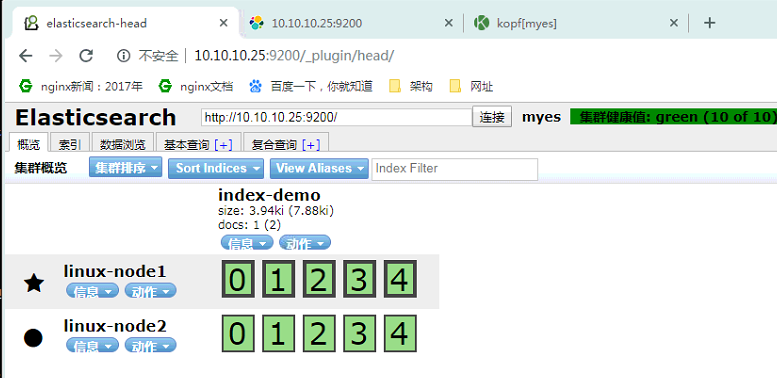
带实验：工作中是否要开启单播模（开启此参数，把IP加入，机器中只有一台加入即可），才能集群、

组播：

单播：

slave启动后对比单台效果：

如果很多的话，打开很慢，慢慢等就行了，木办法



集群的状态：

**curl -XGET "http://10.10.10.25:9200/\_cluster/health?pretty=true"**

官方帮助文档：https://www.elastic.co/guide/cn/elasticsearch/guide/current/index.html

**安装--------logstash**

官方文档：https://www.elastic.co/guide/en/logstash/current/index.html

rpm --import <https://packages.elastic.co/GPG-KEY-elasticsearch>

cat > /etc/yum.repos.d/logstash.repo <<EOF

[logstash-2.1]

name=Logstash repository for 2.1.x packages

baseurl=http://packages.elastic.co/logstash/2.1/centos

gpgcheck=1

gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch

enabled=1

EOF

yum install -y logstash

**官方网站**：https://www.elastic.co/guide/en/logstash/current/index.html

标准输入和输出：

/opt/logstash/bin/logstash -e 'input { stdin{} } output { stdout{} }'

使用rubydebug详细输出

/opt/logstash/bin/logstash -e 'input { stdin{} } output { stdout{ codec => rubydebug} }'

Settings: Default filter workers: 1

Logstash startup completed

**hello word**

{

"message" => "hello word",

"@version" => "1",

"@timestamp" => "2019-03-07T09:56:21.039Z",

"host" => "localhost.localdomain"

}

错误：

[root@localhost ~]# /opt/logstash/bin/logstash -e 'input { stdin{} } output { stdout{} }'

OpenJDK 64-Bit Server VM warning: If the number of processors is expected to increase from one, then you should configure the number of parallel GC threads appropriately using -XX:ParallelGCThreads=N

Settings: Default filter workers: 1

Logstash startup completed

解决：修改虚拟机cpu数量即可

**写入数据到elasticsearch**

简单点写： /opt/logstash/bin/logstash -e 'input { stdin{} } output { elasticsearch { hosts => ["10.10.10.25:9200"]} }'

/opt/logstash/bin/logstash -e 'input { stdin{} } output { elasticsearch { hosts => ["10.10.10.25:9200"] index => "logstash-%{+YYYY.MM.dd}" } }'

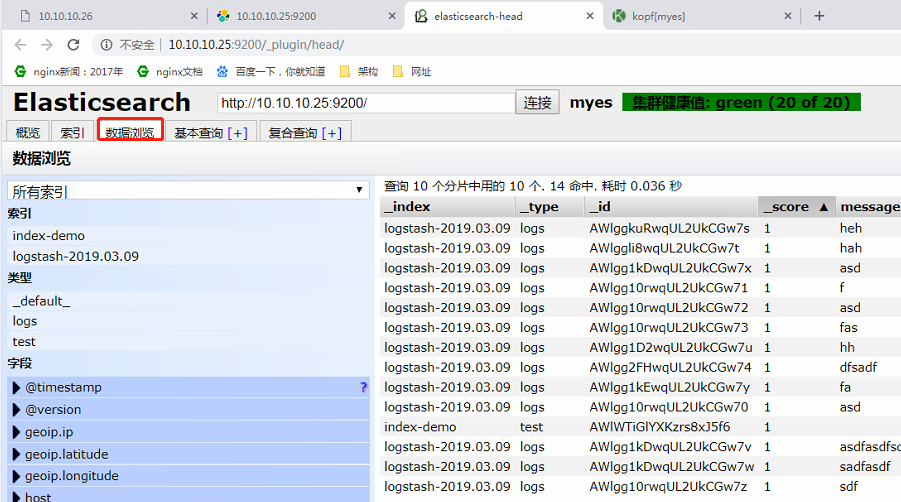
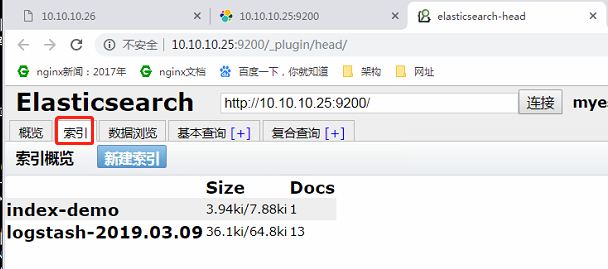
Settings: Default filter workers: 1

Logstash startup completed

heh

hah

hh



既标准输出到屏幕、同时写入到elasticsearch：

 /opt/logstash/bin/logstash -e 'input { stdin{} } output { elasticsearch { hosts => ["10.10.10.25:9200"]} stdout{ codec => rubydebug}}'

/opt/logstash/bin/logstash -e 'input { stdin{} } output { stdout {codec =>rubydebug } elasticsearch { hosts => ["10.10.10.25:9200"] index => "logstash-%{+YYYY.MM.dd}" } }'

Settings: Default filter workers: 1

Logstash startup completed

111111111111111111111111

{

"message" => "111111111111111111111111",

"@version" => "1",

"@timestamp" => "2019-03-09T03:42:44.650Z",

"host" => "nginx"

logstash的配置和文件的编写：

根据启动脚本：cat /etc/init.d/logstash LS\_CONF\_DIR=/etc/logstash/conf.d 配置写到此目录下（可以修改）

官方文档：

<https://www.elastic.co/guide/en/logstash/current/plugins-inputs-file.html#plugins-inputs-file-path>

<https://www.elastic.co/guide/en/logstash/current/configuration-file-structure.html>

<https://www.elastic.co/guide/en/logstash/current/event-dependent-configuration.html>

output插件：

https://www.elastic.co/guide/en/logstash/current/output-plugins.html

/opt/logstash/bin/logstash -f -t /etc/logstash/conf.d/file2.conf -t 测试配置文件是否正确

sincdb ：大约意思 logstash读取日志的记录点

sincedb\_path：可以设置一个路径 默认 /root or /var/lib/logstash/ 貌似都有，用到再说吧

start\_position => "beginning" "end"

[root@nginx ~]# cat /var/lib/logstash/.sincedb\_

.sincedb\_1fb922e15ccea4ac0d028d33639ba3ea .sincedb\_a9b9fed7edff6fd888ffe131a05b5397

.sincedb\_452905a167cf4509fd08acb964fdb20c .sincedb\_d883144359d3b4f516b37dba51fab2a2

1、必须包含input和outpu

2、input codec filter output

测试一、**标准输出** 与 **写入elasticsearch**

[root@nginx conf.d]# cat demo.conf

input {

stdin {}

}

#input{stdin{}}

output {

elasticsearch

{

hosts => ["10.10.10.25:9200"]

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

}

stdout

{

codec => rubydebug

}

}

[root@nginx conf.d]# /opt/logstash/bin/logstash -f /etc/logstash/conf.d/demo.conf

Settings: Default filter workers: 1

Logstash startup completed

test

{

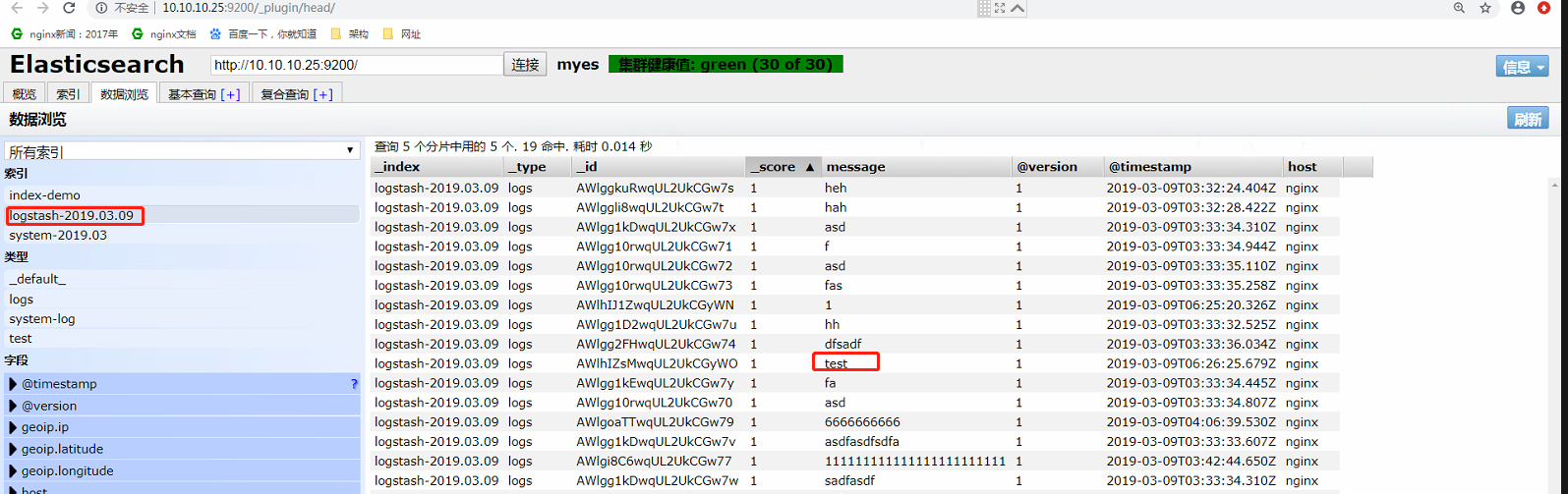
"message" => "test",

"@version" => "1",

"@timestamp" => "2019-03-09T06:26:25.679Z",

"host" => "nginx"

}



测试二、**写入到elasticsearch**：

[root@nginx conf.d]# cat file.conf

input{

file{

path => ["/var/log/messages","/var/log/secure" ]

type => "system-log"

start\_position => "beginning"

}

}

filter{

}

output {

elasticsearch

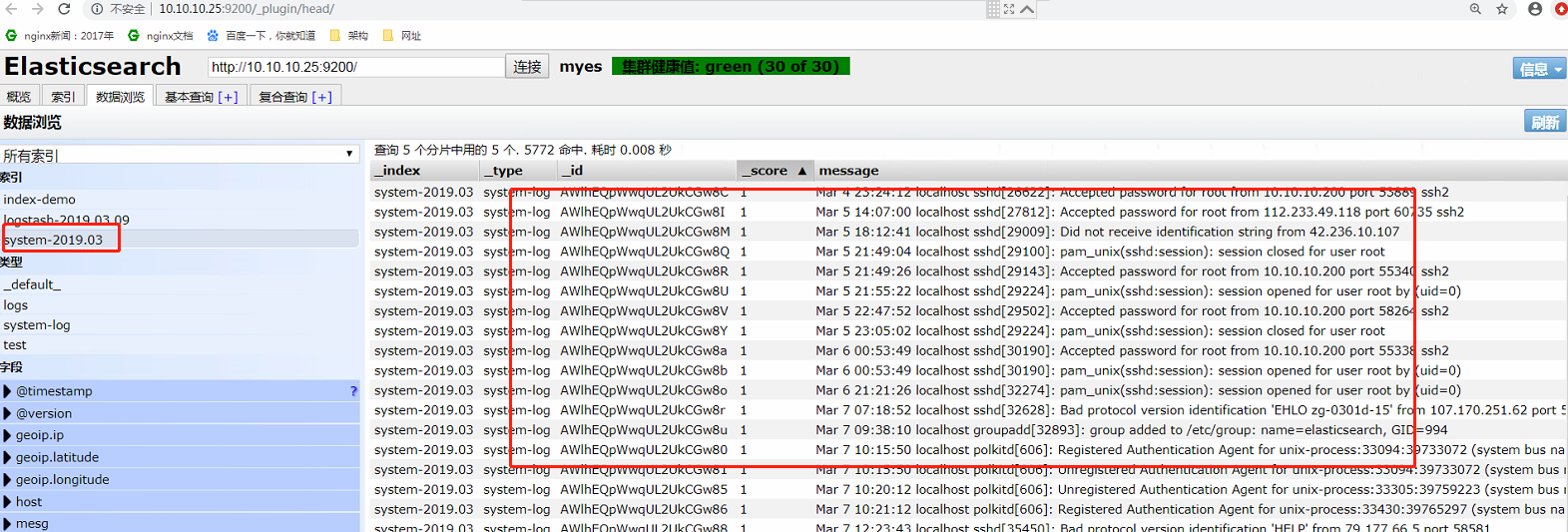
{

hosts => ["10.10.10.25:9200"]

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

}

}



测试三、**写入到elasticsearch**：

注意：  
如果你的日志中有type字段 那你就不能在conf文件中使用type

[root@nginx conf.d]# cat file1.conf

input {

file {

path => "/var/log/messages"

type => "messages"

start\_position => "beginning"

}

}

input {

file {

path => "/var/log/elasticsearch/myes.log"

type => "es-error"

start\_position => "beginning"

}

}

output {

if [type] == "messages"{

elasticsearch {

hosts => ["10.10.10.25:9200"]

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

}

}

if [type] == "es-error"{

elasticsearch {

hosts => ["10.10.10.25:9200"]

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

}

}

}

暂停一段落：

**收集TCP日志：**

https://www.elastic.co/guide/en/logstash/current/plugins-inputs-tcp.html

使用nc发送数据：

25端logstash配置：

[root@nginx conf.d]# cat vim tcp.conf

input{

tcp{

type => "tcp"

port => "6666"

mode => "server"

}

}

output{

stdout{

codec => "rubydebug"

}

}

**IP26写入数据 25即可收到**

echo "hehe" |nc 10.10.10.25 6666

nc 10.10.10.25 6666 < /etc/hostname

echo "hehehehehe" >/dev/tcp/10.10.10.25/6666

[Filter plugins](https://www.elastic.co/guide/en/logstash/current/filter-plugins.html)

**使用filter 的grok 插件 对收集的事件，进行过滤，字段拆分等**

<https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html>

例如：收集apaceh日志

使用说明：/opt/logstash/vendor/bundle/jruby/1.9/gems/logstash-patterns-core-2.0.2/patterns

[root@elk-node1 ~]# cat grok.conf

input {

    stdin{}

}

filter {

  grok {

    match => { "message" => "%{IP:client} %{WORD:method} %{URIPATHPARAM:request} %{NUMBER:bytes} %{NUMBER:duration}" }

  }4

}

output {

    stdout{

        codec => "rubydebug"

    }

}

**执行检测：**

[root@elk-node1 ~]# /opt/logstash/bin/logstash -f grok.conf

Settings: Default filter workers: 1

Logstash startup completed

55.3.244.1 GET /index.html 15824 0.043                    #输入这个，下面就会自动形成字典的形式

{

       "message" => "55.3.244.1 GET /index.html 15824 0.043",

      "@version" => "1",

    "@timestamp" => "2019-0113T11:45:47.882Z",

          "host" => "elk-node1",

        "client" => "55.3.244.1",

        "method" => "GET",

       "request" => "/index.html",

         "bytes" => "15824",

      "duration" => "0.043"

}

ELK中使用redis作为中间件，缓存日志采集内容

如果使用redis作为ELKstack消息队列，需要对所有的list key的长度进行监控

根据实际情况：例如超过10万报警

**Kibanna安装与使用**

使用手册：https://www.elastic.co/guide/cn/kibana/current/index.html

版本查看：https://www.elastic.co/downloads/past-releases

安装：<https://www.elastic.co/guide/cn/kibana/current/rpm.html>

rpm --import <https://artifacts.elastic.co/GPG-KEY-elasticsearch>

cat > /etc/yum.repos.d/kibana.repo <<EOF

[kibana-6.x]

name=Kibana repository for 6.x packages

baseurl=https://artifacts.elastic.co/packages/6.x/yum

gpgcheck=1

gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch

enabled=1

autorefresh=1

type=rpm-md

EOF

说明一下：想安装5.x把上面的6改为5即可

由于elasticsearch安装的为2.X 我擦的用kibana5x不行的。

4.5.4版本安装配置源如下

cat > /etc/yum.repos.d/kibana.repo <<EOF

[kibana-4.5]

name=Kibana repository for4.5.x packages

baseurl=http://packages.elastic.co/kibana/4.5/centos

gpgcheck=1

gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch

enabled=1

EOF

yum -y install kibana

[root@nginx conf.d]# grep "^[a-Z]" /etc/kibana/kibana.yml

server.port: 5601

server.host: "10.10.10.25"

elasticsearch.url: "http://10.10.10.26:9200"

kibana.index: ".kibana"

<http://10.10.10.25:5601/>

如下摘自<https://www.cnblogs.com/kevingrace/p/5919021.html>

暂未学习

**==============logstash配置java环境===============**  
由于新版的ELK环境要求java1.8，但是有些服务器由于业务代码自身限制只能用java6或java7。  
这种情况下，要安装Logstash，就只能单独配置Logstas自己使用的java环境了。

|  |
| --- |
| 操作如下：  0) 使用rpm包安装logstash    1）安装java8，参考：http://www.cnblogs.com/kevingrace/p/7607442.html    2）在/etc/sysconfig/logstash文件结尾添加下面两行内容：  [root@cx-app01 ~]# vim /etc/sysconfig/logstash  .......  JAVA\_CMD=/usr/local/jdk1.8.0\_172/bin  JAVA\_HOME=/usr/local/jdk1.8.0\_172    3）在/opt/logstash/bin/logstash.lib.sh文件添加下面一行内容：  [root@cx-app02 ~]# vim /opt/logstash/bin/logstash.lib.sh  .......  export JAVA\_HOME=/usr/local/jdk1.8.0\_172    4) 然后使用logstash收集日志，就不会报java环境错误了。 |

**==================配置范例===================**

|  |
| --- |
| 如下的配置范例：  192.168.10.44为elk的master节点，同时也是redis节点    [root@client-node01 opt]# pwd  /opt  [root@client-node01 opt]# cat redis-in.conf  input {      file {         path => "/usr/local/tomcat8/logs/catalina.out"         type => "tomcat8-logs"         start\_position => "beginning"         codec => multiline {             pattern => "^\["           //表示收集以"["开头的日志信息             negate => true             what => "previous"         }      }  }    output {      if [type] == "tomcat8-logs"{         redis {            host => "192.168.10.44"            port => "6379"            db => "1"            data\_type => "list"            key => "tomcat8-logs"         }       }  }    [root@client-node01 opt]# cat redis-input.conf  input {    file {          path => "/var/log/messages"          type => "systemlog"          start\_position => "beginning"          stat\_interval => "2"    }  }    output {    if [type] == "systemlog" {          redis {                  data\_type => "list"                  host => "192.168.10.44"                  db => "2"                  port => "6379"                  key => "systemlog"          }    }    }  [root@client-node01 opt]# cat file.conf  input {       redis {          type => "tomcat8-logs"          host => "192.168.10.44"          port => "6379"          db => "1"          data\_type => "list"          key => "tomcat8-logs"       }           redis {            type => "systemlog"            host => "192.168.10.44"            port => "6379"            db => "2"            data\_type => "list"            key => "systemlog"         }    }      output {        if [type] == "tomcat8-logs"{          elasticsearch {             hosts => ["192.168.10.44:9200"]             index => "elk-node2-tomcat8-logs-%{+YYYY.MM.dd}"          }      }        if [type] == "systemlog"{          elasticsearch {             hosts => ["192.168.10.44:9200"]             index => "elk-node2-systemlog-%{+YYYY.MM.dd}"          }      }  }  [root@client-node01 opt]# /opt/logstash/bin/logstash -f /opt/redis-in.conf --configtest  Configuration OK  [root@client-node01 opt]# /opt/logstash/bin/logstash -f /opt/redis-input.conf --configtest  Configuration OK  [root@client-node01 opt]# /opt/logstash/bin/logstash -f /opt/file.conf --configtest  Configuration OK    启动logstash  [root@client-node01 opt]# /opt/logstash/bin/logstash -f /opt/redis-in.conf &  [root@client-node01 opt]# /opt/logstash/bin/logstash -f /opt/redis-input.conf &  [root@client-node01 opt]# /opt/logstash/bin/logstash -f /opt/file.conf &    这时候，当/usr/local/tomcat8/logs/catalina.out和/var/log/messages文件里有新日志信息写入时，就会触发动作，  在redis里就能查看到相关信息，并查看写入到es里。    =========================================================================================================  温馨提示：  当客户机的日志信息收集后，经过redis刚读到es数据库里后，如果没有新数据写入，则默认在es的访问界面里是看不到  数据的，只有当日志文件里有新的日志写入后才会触发数据展示的动作，即es的访问界面（http://192.168.10.44:9200/\_plugin/head/）  里才能看到日志数据的展示效果。  ==========================================================================================================    假设想上面两个文件里写入测试数据  [root@client-node01 opt]# echo "hellohellohellohello" >> /var/log/messages  [root@client-node01 opt]# echo "[hahahahahahhahahahahahahahahahahahah]" >> /usr/local/tomcat8/logs/catalina.out    到redis里发现有相关的key，很快就会读到es里。可以配置到kibana里观察。    可以先测试下日志信息是否写到redis里？然后再测试下数据是否从redis读到es里？一步步确定数据去向。    注意上面redis-in.conf文件中的下面设置，使用正则匹配，收集以哪些字符开头的日志信息：  pattern => "^\["                    表示收集以"["开头的日志信息  pattern => "^2018"                  表示收集以"2018"开头的日志信息  pattern => "^[a-zA-Z0-9]"           表示收集以字母（大小写）或数字开头的日志信息  pattern => "^[a-zA-Z0-9]|[^ ]+"     表示收集以字母（大小写）或数字或空格的日志信息 |