**ELK 集群部署+ Grafana监控**

目 录

[1. 集群概述 4](#_Toc536106131)

[1.1 ELK集群部署 4](#_Toc536106132)

[1.2 集群概述 5](#_Toc536106133)

[2. Es集群部署详细 5](#_Toc536106134)

[2.1 部署文件建立 5](#_Toc536106135)

[2.1.1 IP设置 5](#_Toc536106136)

[2.1.2 公共文件夹 6](#_Toc536106137)

[2.1.3 目录建立 6](#_Toc536106138)

[2.1.4 用户创建并赋予root权限 6](#_Toc536106139)

[2.1.5 安装文件下载 7](#_Toc536106140)

[2.2 es配置文件更改 8](#_Toc536106141)

[2.2.1安装包解压 8](#_Toc536106142)

[2.2.2 修改ES配置文件 8](#_Toc536106143)

[2.3 es 程序启动 10](#_Toc536106144)

[2.3.1程序启动 11](#_Toc536106145)

[2.3.2命令查看 12](#_Toc536106146)

[2.4 head 插件安装及服务启动 12](#_Toc536106147)

[2.4.1 安装 nodejs 12](#_Toc536106148)

[2.4.2 安装grunt 13](#_Toc536106149)

[2.4.3 npm install 13](#_Toc536106150)

[2.4.4 修改 head 配置 14](#_Toc536106151)

[2.4.5 修改es配置 14](#_Toc536106152)

[2.4.6 启动 head插件 15](#_Toc536106153)

[3. Kibana部署安装 16](#_Toc536106154)

[3.1 安装部署 16](#_Toc536106155)

[3.1.1 包解压 16](#_Toc536106156)

[3.1.2 配置文件更改 16](#_Toc536106157)

[3.1.3 kibana启动 19](#_Toc536106158)

[3.2 x-pack权限配置 20](#_Toc536106159)

[3.2.1 初始化密码 20](#_Toc536106160)

[3.2.2 更改密码 21](#_Toc536106161)

[3.2.3 权限操作 22](#_Toc536106162)

[3.3 Kibana使用 23](#_Toc536106163)

[3.4 Kibana日志监控 24](#_Toc536106164)

[3.4.1 es规则建立 24](#_Toc536106165)

[3.4.2 dashbord 视图 26](#_Toc536106166)

[3.4.3 流量监控 27](#_Toc536106167)

[4. Logstash 部署安装 27](#_Toc536106168)

[4.1 解压包 27](#_Toc536106169)

[4.2 conf文件配置 27](#_Toc536106170)

[4.3 conf配置校验 29](#_Toc536106171)

[4.4 启动logStash 30](#_Toc536106172)

[4.5 MYSQL 数据同步至 ES 30](#_Toc536106173)

[4.6 logstash使用ES的template创建索引 35](#_Toc536106174)

[5. 插件IK+pinyin分词器安装 37](#_Toc536106175)

[5.1 下载ik+pinyin安装包 37](#_Toc536106176)

[5.2. 安装 38](#_Toc536106177)

[5.3 重新启动ES 39](#_Toc536106178)

[5.4 测试ik插件(通过head插件测试) 40](#_Toc536106179)

[5.5 版本差异分词器使用 40](#_Toc536106180)

[5.6 IK+Pinyin分词配置 41](#_Toc536106181)

[5.6.1 创建索引与分析器设置 41](#_Toc536106182)

[5.6.2 创建type并设置mapping 43](#_Toc536106183)

[6. 索引设置模板-template设置 45](#_Toc536106184)

[6.1 template 基础命令 45](#_Toc536106185)

[6.2 模板设置 45](#_Toc536106186)

[6.3 网址参考 50](#_Toc536106187)

[7. Grafana监控 51](#_Toc536106188)

[7.1 Grafana 获取下载 51](#_Toc536106189)

[7.2 启动安装 51](#_Toc536106190)

[7.3 配置日志数据源 52](#_Toc536106191)

[7.4 Graph添加 53](#_Toc536106192)

[7.5 用户权限设置 55](#_Toc536106193)

[7.6 设置Alert阈值和邮件报警 56](#_Toc536106194)

[7.6.1 notification channel 通知渠道设置 56](#_Toc536106195)

[7.6.2 Alert Config 配置 56](#_Toc536106196)

[7.7 网址参考 57](#_Toc536106197)

[8.部署问题 58](#_Toc536106198)

[8.1 问题01 58](#_Toc536106199)

[8.2 问题02 58](#_Toc536106200)

[8.3 问题03 59](#_Toc536106201)

[8.4 问题04 59](#_Toc536106202)

[8.5 问题05 59](#_Toc536106203)

[8.6 问题06 60](#_Toc536106204)

[8.7 问题07 61](#_Toc536106205)

[8.8 问题08 61](#_Toc536106206)

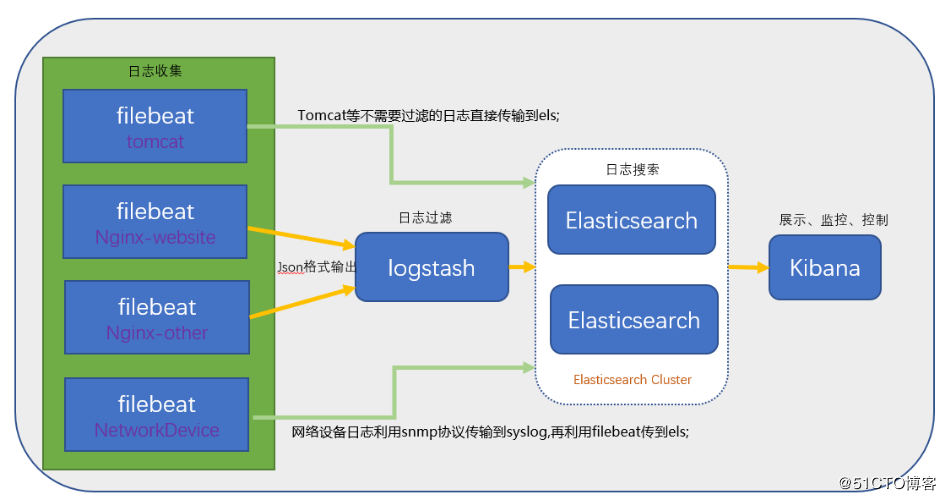
[8.9 问题09 61](#_Toc536106207)

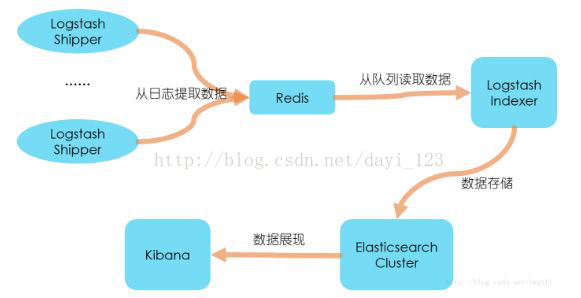
[8.10 问题10 62](#_Toc536106208)

# 1. 集群概述

## 1.1 ELK集群部署

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | type | port | V01- 10.1.2.105 | V02- 10.1.2.106 | V03- 10.1.2.107 |
| 01 | elasticSearch | 9200 |  |  |  |
| 02 | Head | 9100 |  |  |  |
| 03 | Kibana | 5601 |  |  |  |
| 04 | logstash |  |  |  |  |
| 版本信息：  elasticSearch : elasticsearch-6.3.0  Head: elasticsearch-head-master  Kibana: kibana-6.3.0-linux-x86\_64  Logstash: logstash-6.3.0 | | | | | |



s

## 1.2 集群概述

# 2. Es集群部署详细

## 2.1 部署文件建立

### 2.1.1 IP设置

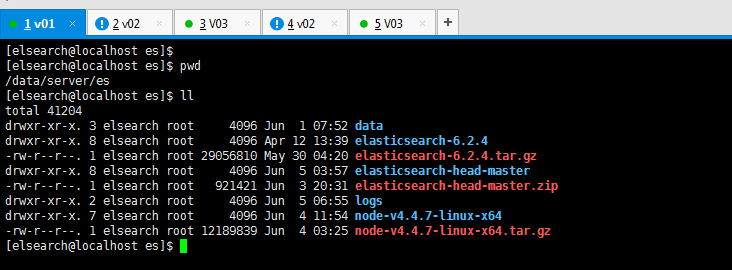
准备机器，IP分别设置为： v01/10.1.2.105、v02/10.1.2.106、v03/10.1.2.107

### 2.1.2 公共文件夹

分别在v01 、v02、 v03机器上建立 /data/server/es文件夹

### 2.1.3 目录建立

分别在机器v01 、v02、 v03 es文件建立data 、logs 两个目录。



### 2.1.4 用户创建并赋予root权限

因为安全问题elasticsearch 不让用root用户直接运行，所以要创建新用户，默认不允许root权限的用户允许elasticsearch，所以这里新建一个用户组elsearch及该组下的用户elsearch

建议创建一个单独的用户用来运行ElasticSearch, 创建elsearch用户组及elsearch用户

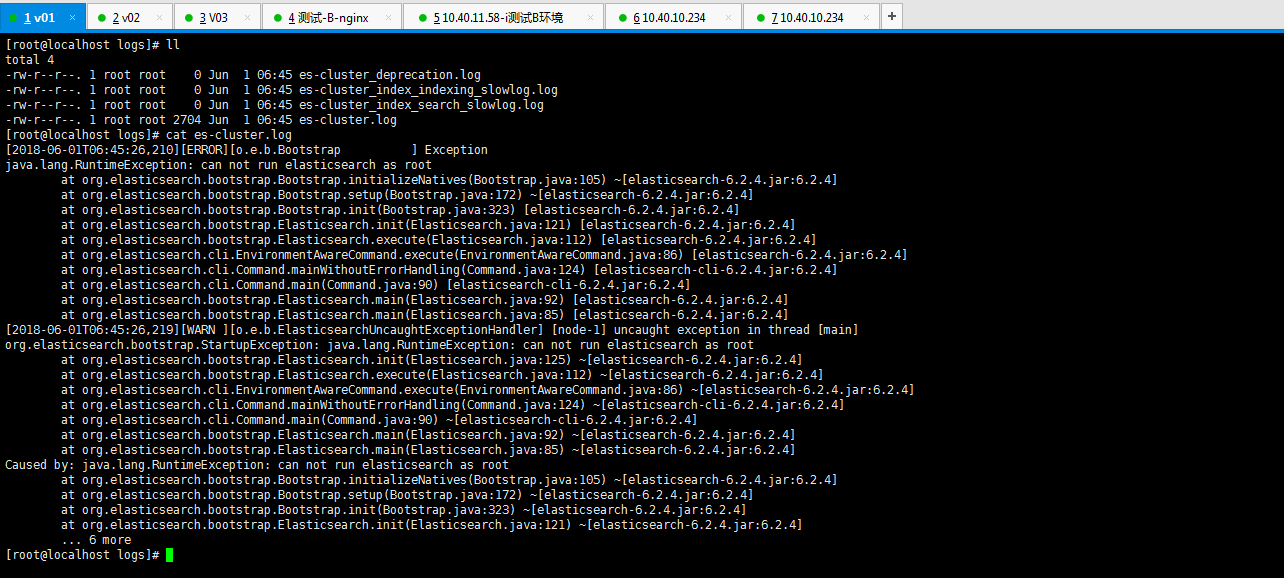
* 用户创建

|  |
| --- |
| groupadd elsearch  useradd elsearch -g elsearch -p elasticsearch  #赋予root权限  usermod -g root elsearch |

* 赋予Root权限

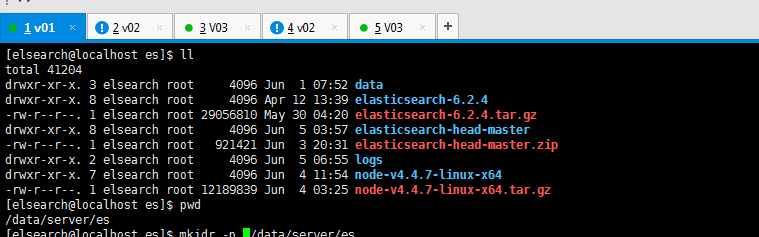
|  |
| --- |
| #在ROOT 用户申请以下权限  #root给elsearch赋权限，chown -R elsearch /你的elasticsearch安装目录  chown -R elsearch elasticsearch-6.2.4  #赋予root权限  usermod -g root elsearch |

如果采用ROOT权限来启动ES，报以下错误



### 2.1.5 安装文件下载

|  |
| --- |
| #es 源程序安装包，安装步骤见  elasticsearch-6.2.4.tar.gz  #es插件安装包，安装步骤见  elasticsearch-head-master.zip  #nodejs 安装包，安装步骤见  node-v4.4.7-linux-x64.tar.gz |



## 2.2 es配置文件更改

elasticsearch-6.3.0 版本己经集成了插件x-pack，不需要单独安装x-pack

### 2.2.1安装包解压

在机器 v01、v02、v03 三台机器解压es安装包

|  |
| --- |
| #解压ES安装包  tar -zxvf elasticsearch-6.3.0.tar.gz |

### 2.2.2 修改ES配置文件

切换到config目录下，对elasticsearch.yml 配置监控

|  |
| --- |
| # ============ Elasticsearch Configuration =============  #  # NOTE: Elasticsearch comes with reasonable defaults for most settings.  # Before you set out to tweak and tune the configuration, make sure you  # understand what are you trying to accomplish and the consequences.  #  # The primary way of configuring a node is via this file. This template lists  # the most important settings you may want to configure for a production cluster.  #  # Please consult the documentation for further information on configuration options:  # https://www.elastic.co/guide/en/elasticsearch/reference/index.html  #  # ---------------------------------- Cluster -----------------------------------  #  # Use a descriptive name for your cluster:  #  **cluster.name: es-cluster**  #  # ------------------------------------ Node ------------------------------------  #  # Use a descriptive name for the node:  #  **node.name: node-1**  **node.master: true**  **node.data: true**  #  # Add custom attributes to the node:  #  #node.attr.rack: r1  #  # ----------------------------------- Paths ------------------------------------  #  # Path to directory where to store the data (separate multiple locations by comma):  #  **path.data: /data/server/es/data**  #  # Path to log files:  #  **path.logs: /data/server/es/logs**  #  # ----------------------------------- Memory -----------------------------------  #  # Lock the memory on startup:  #  **bootstrap.memory\_lock: false**  **bootstrap.system\_call\_filter: false**  #  # Make sure that the heap size is set to about half the memory available  # on the system and that the owner of the process is allowed to use this  # limit.  #  # Elasticsearch performs poorly when the system is swapping the memory.  #  # ---------------------------------- Network -----------------------------------  #  # Set the bind address to a specific IP (IPv4 or IPv6):  #  **network.host: 10.1.2.105**  #  # Set a custom port for HTTP:  #  **http.port: 9200**  #  # For more information, consult the network module documentation.  #  # --------------------------------- Discovery ----------------------------------  #  # Pass an initial list of hosts to perform discovery when new node is started:  # The default list of hosts is ["127.0.0.1", "[::1]"]  #  discovery.zen.ping.unicast.hosts: ["10.1.2.105", "10.1.2.106","10.1.2.107"]  #  # Prevent the "split brain" by configuring the majority of nodes (total number of master-eligible nodes / 2 + 1):  #  **discovery.zen.minimum\_master\_nodes: 2**  #  # For more information, consult the zen discovery module documentation.  #  # ---------------------------------- Gateway -----------------------------------  #  # Block initial recovery after a full cluster restart until N nodes are started:  #  #gateway.recover\_after\_nodes: 3  #  # For more information, consult the gateway module documentation.  #  # ---------------------------------- Various -----------------------------------  #  # Require explicit names when deleting indices:  #  #action.destructive\_requires\_name: true  #  #  node.max\_local\_storage\_nodes: 256  #plugin head插件所需要配置属性  **http.cors.enabled: true**  **http.cors.allow-origin: "\*"**  ##增加 elasticsearch.yml 配置文件。用户支持 x-pack插件，见3.2 权限设置  ##如果 设以下参数 需要配置关注 kibana \logstash\elastic 用户名、密码  **xpack.security.enabled: true** |

在v01、V02、V03三台服务器配置文件 **network.host** 对应的IP值更改，其他保持 不变

## 2.3 es 程序启动

切换至目录 /data/server/es/elasticsearch-6.2.4/bin

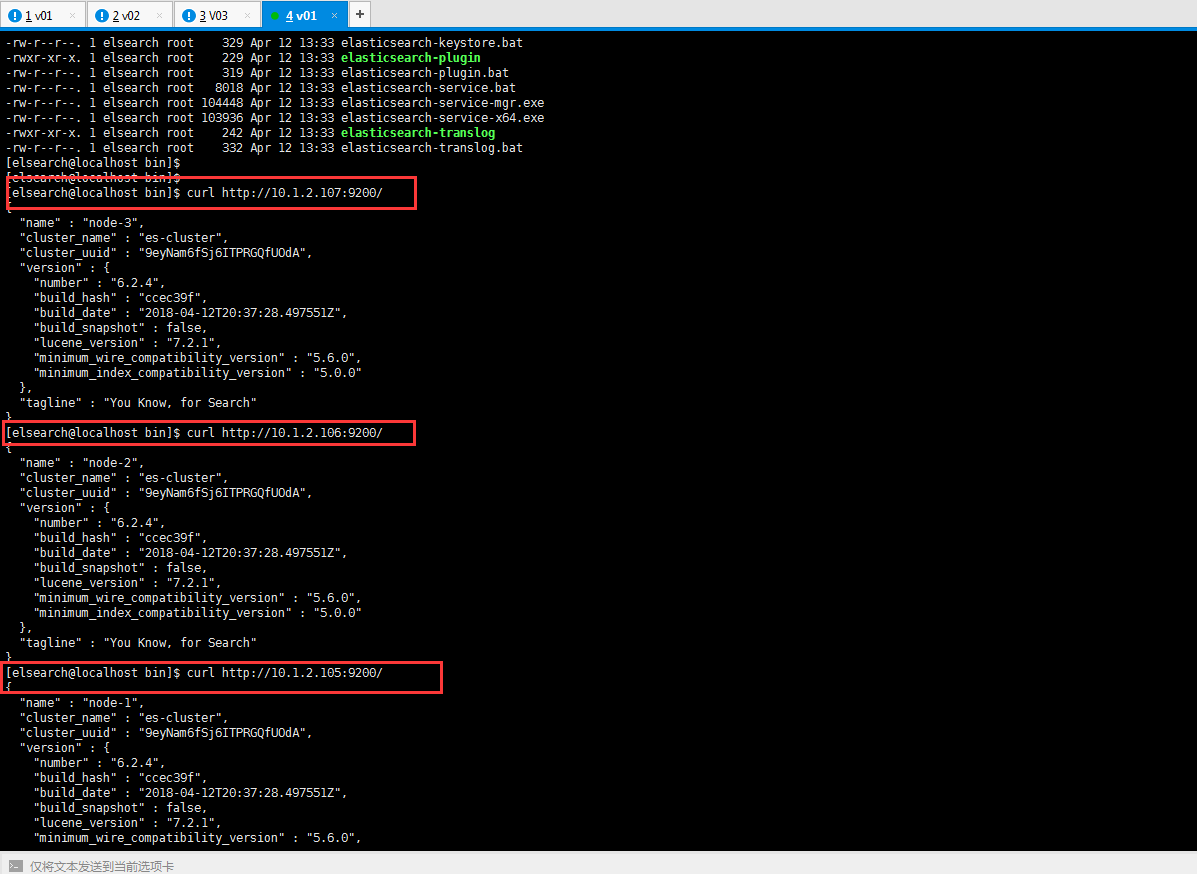
### 2.3.1程序启动

|  |
| --- |
| # -d 表示后台运行  ./elasticsearch -d |

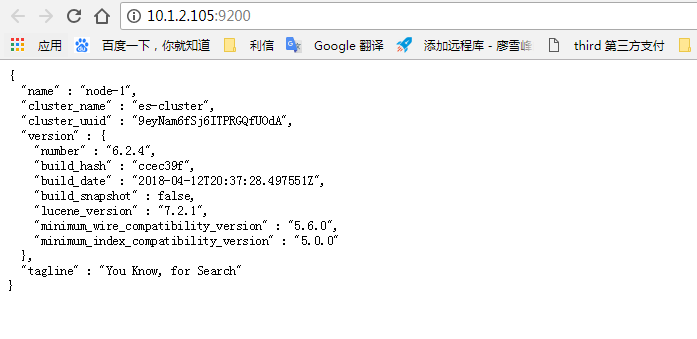
PS: 启动过程中异常信息见：部署问题01-08

Es 成功启动访问

命令控制crul访问



浏览器访问

****

### 2.3.2命令查看

|  |
| --- |
| 查看集群状态：curl -XGET http://10.1.2.105:9200/\_cat/health?v  查看集群节点：curl -XGET http://10.1.2.105:9200/\_cat/nodes?v  查询索引列表：curl -XGET http://10.1.2.105:9200/\_cat/indices?v |

## 2.4 head 插件安装及服务启动

ES 5.X 后， 插件 elasticsearch-head-master 不能放在 elasticsearch-/plugins 插件文件下，一般情况下与elasticsearch保持相同一目录

在机器 10.1.2.105/v01 安装head插件

### 2.4.1 安装 nodejs

* 下载

nodejs源码包 node-v4.4.7-linux-x64.tar.gz

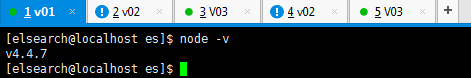
head 源码包 elasticsearch-head-master.zip

* 解压源码包

|  |
| --- |
| #解压node源码包  tar –node-v4.4.7-linux-x64.tar.gz  #解压head 源码包  unzip elasticsearch-head-master.zip |

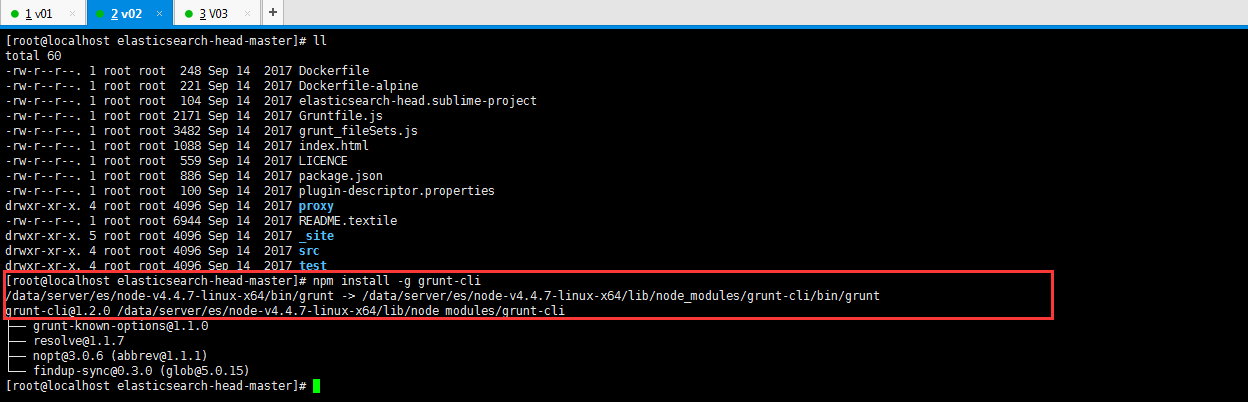
* 环境变量配置

|  |
| --- |
| **NODE\_HOME=/data/server/es/node-v4.4.7-linux-x64**  JAVA\_HOME=/data/server/jdk/jdk1.8.0\_161  JRE\_HOME=/data/server/jdk/jdk1.8.0\_161/jre  CLASS\_PATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar:$JRE\_HOME/lib  PATH=$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin:$NODE\_HOME/bin  **NODE\_PATH=$NODE\_HOME/lib/node\_modules**  export JAVA\_HOME JRE\_HOME CLASS\_PATH NODE\_HOME NODE\_PATH PATH |



### 2.4.2 安装grunt

|  |
| --- |
| npm install -g grunt-cli  grunt -version |

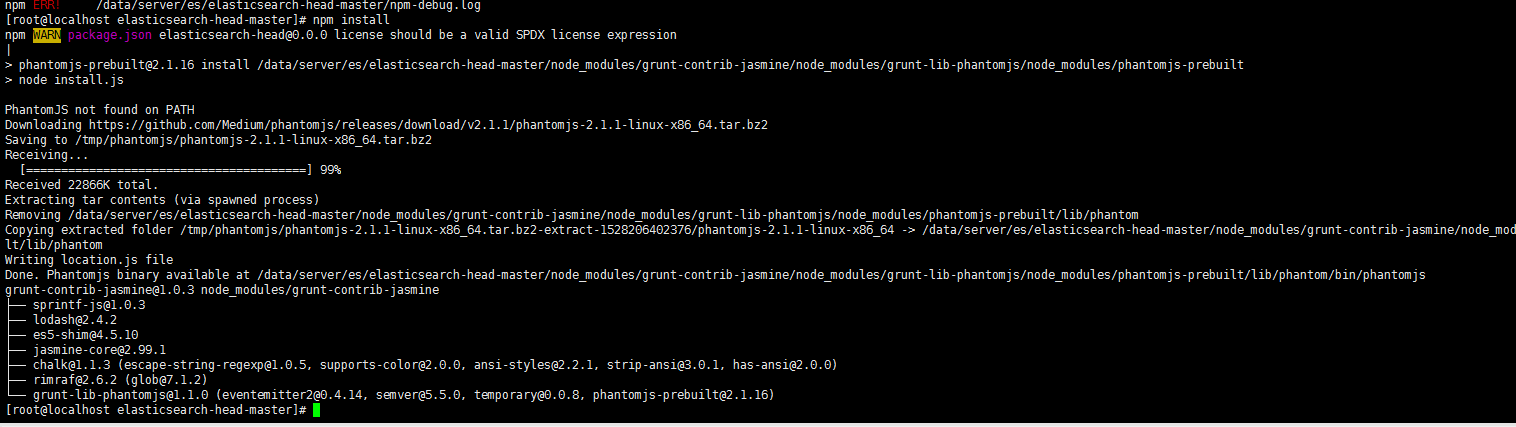


### 2.4.3 npm install

切换目录至 elasticsearch-head-master目录

cd elasticsearch-head-master

>npm install



### 2.4.4 修改 head 配置

打开elasticsearch-head-master/Gruntfile.js，找到下面connect属性，

新增hostname: ‘0.0.0.0’:

|  |
| --- |
| connect: {  server: {  options: {  **hostname: '0.0.0.0',**  port: 9100,  base: '.',  keepalive: true  }  }  } |

### 2.4.5 修改es配置

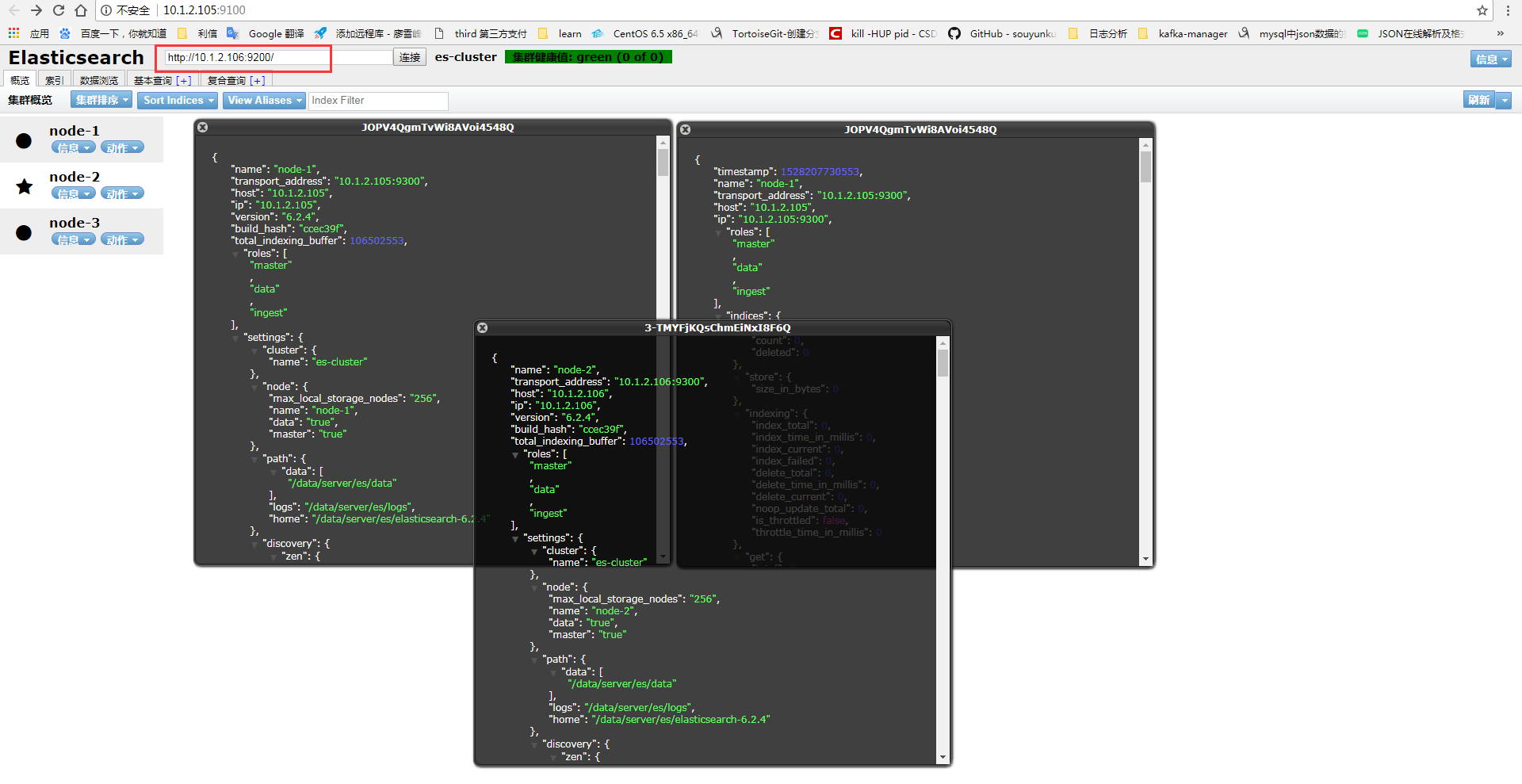
修改 ES 的配置，增加跨域的配置，使得head 插件能够访问到 es

|  |
| --- |
| ##修改elasticsearch.yml文件，增加如下内容：  　　http.cors.enabled: true  　　http.cors.allow-origin: "\*" |

### 2.4.6 启动 head插件

|  |
| --- |
| #切换目录  cd /data/server/es/elasticsearch-head-master  #以grunt 方式启动  grunt server & |

Es 启动成功见下图



连接访问es header

|  |
| --- |
| http://10.1.2.105:9100/ |

ps:head安装参见配置

|  |
| --- |
| https://www.imooc.com/article/22709?block\_id=tuijian\_wz  <https://www.cnblogs.com/techroad4ca/p/7748293.html> |

# 3. Kibana部署安装

## 3.1 安装部署

kibana-6.3.0 己经集成插件x-pack,所以该 版本不需要单独x-pack插件

PS: X-pack监控组件使您可以通过Kibana轻松地监控ElasticSearch。您可以实时查看集群的健康和性能，以及分析过去的集群、索引和节点度量。此外，您可以监视Kibana本身性能。当你安装X-pack在群集上，监控代理运行在每个节点上收集和指数指标从Elasticsearch。安装在X-pack在Kibana上，您可以查看通过一套专门的仪表板监控数据。

### 3.1.1 包解压

在机器 10.1.2.105/V01 部署kibana

|  |
| --- |
| #解压kibana安装包  tar –zxvf kibana-6.3.0-linux-x86\_64.tar.gz |

### 3.1.2 配置文件更改

切换于 config目录，编缉更改 文件 kibana.yml 配置如下

|  |
| --- |
| # Kibana is served by a back end server. This setting specifies the port to use.  **server.port: 5601**  # Specifies the address to which the Kibana server will bind. IP addresses and host names are both valid values.  # The default is 'localhost', which usually means remote machines will not be able to connect.  # To allow connections from remote users, set this parameter to a non-loopback address.  **server.host: "10.1.2.106"**  # Enables you to specify a path to mount Kibana at if you are running behind a proxy.  # Use the `server.rewriteBasePath` setting to tell Kibana if it should remove the basePath  # from requests it receives, and to prevent a deprecation warning at startup.  # This setting cannot end in a slash.  #server.basePath: ""  # Specifies whether Kibana should rewrite requests that are prefixed with  # `server.basePath` or require that they are rewritten by your reverse proxy.  # This setting was effectively always `false` before Kibana 6.3 and will  # default to `true` starting in Kibana 7.0.  #server.rewriteBasePath: false  # The maximum payload size in bytes for incoming server requests.  #server.maxPayloadBytes: 1048576  # The Kibana server's name. This is used for display purposes.  #server.name: "your-hostname"  # The URL of the Elasticsearch instance to use for all your queries.  **elasticsearch.url: "http://10.1.2.106:9200"**  # When this setting's value is true Kibana uses the hostname specified in the server.host  # setting. When the value of this setting is false, Kibana uses the hostname of the host  # that connects to this Kibana instance.  #elasticsearch.preserveHost: true  # Kibana uses an index in Elasticsearch to store saved searches, visualizations and  # dashboards. Kibana creates a new index if the index doesn't already exist.  kibana.index: "wecompany"  # The default application to load.  #kibana.defaultAppId: "home"  # If your Elasticsearch is protected with basic authentication, these settings provide  # the username and password that the Kibana server uses to perform maintenance on the Kibana  # index at startup. Your Kibana users still need to authenticate with Elasticsearch, which  # is proxied through the Kibana server. 用户名、密码配置参见 x-pack权限配置，  ##如果 没有用户密码设置则不需要 参下参数  **elasticsearch.username: "elastic"**  **elasticsearch.password: "elastic"**  # Enables SSL and paths to the PEM-format SSL certificate and SSL key files, respectively.  # These settings enable SSL for outgoing requests from the Kibana server to the browser.  #server.ssl.enabled: false  #server.ssl.certificate: /path/to/your/server.crt  #server.ssl.key: /path/to/your/server.key  # Optional settings that provide the paths to the PEM-format SSL certificate and key files.  # These files validate that your Elasticsearch backend uses the same key files.  #elasticsearch.ssl.certificate: /path/to/your/client.crt  #elasticsearch.ssl.key: /path/to/your/client.key  # Optional setting that enables you to specify a path to the PEM file for the certificate  # authority for your Elasticsearch instance.  #elasticsearch.ssl.certificateAuthorities: [ "/path/to/your/CA.pem" ]  # To disregard the validity of SSL certificates, change this setting's value to 'none'.  #elasticsearch.ssl.verificationMode: full  # Time in milliseconds to wait for Elasticsearch to respond to pings. Defaults to the value of  # the elasticsearch.requestTimeout setting.  #elasticsearch.pingTimeout: 1500  # Time in milliseconds to wait for responses from the back end or Elasticsearch. This value  # must be a positive integer.  #elasticsearch.requestTimeout: 30000  # List of Kibana client-side headers to send to Elasticsearch. To send \*no\* client-side  # headers, set this value to [] (an empty list).  #elasticsearch.requestHeadersWhitelist: [ authorization ]  # Header names and values that are sent to Elasticsearch. Any custom headers cannot be overwritten  # by client-side headers, regardless of the elasticsearch.requestHeadersWhitelist configuration.  #elasticsearch.customHeaders: {}  # Time in milliseconds for Elasticsearch to wait for responses from shards. Set to 0 to disable.  #elasticsearch.shardTimeout: 30000  # Time in milliseconds to wait for Elasticsearch at Kibana startup before retrying.  #elasticsearch.startupTimeout: 5000  # Logs queries sent to Elasticsearch. Requires logging.verbose set to true.  #elasticsearch.logQueries: false  # Specifies the path where Kibana creates the process ID file.  #pid.file: /var/run/kibana.pid  # Enables you specify a file where Kibana stores log output.  #logging.dest: /data/server/es/kibana-6.3.0-linux-x86\_64/logs/  # Set the value of this setting to true to suppress all logging output.  #logging.silent: false  # Set the value of this setting to true to suppress all logging output other than error messages.  #logging.quiet: false  # Set the value of this setting to true to log all events, including system usage information  # and all requests.  #logging.verbose: false  # Set the interval in milliseconds to sample system and process performance  # metrics. Minimum is 100ms. Defaults to 5000.  #ops.interval: 5000  # The default locale. This locale can be used in certain circumstances to substitute any missing  # translations.  #i18n.defaultLocale: "en" |

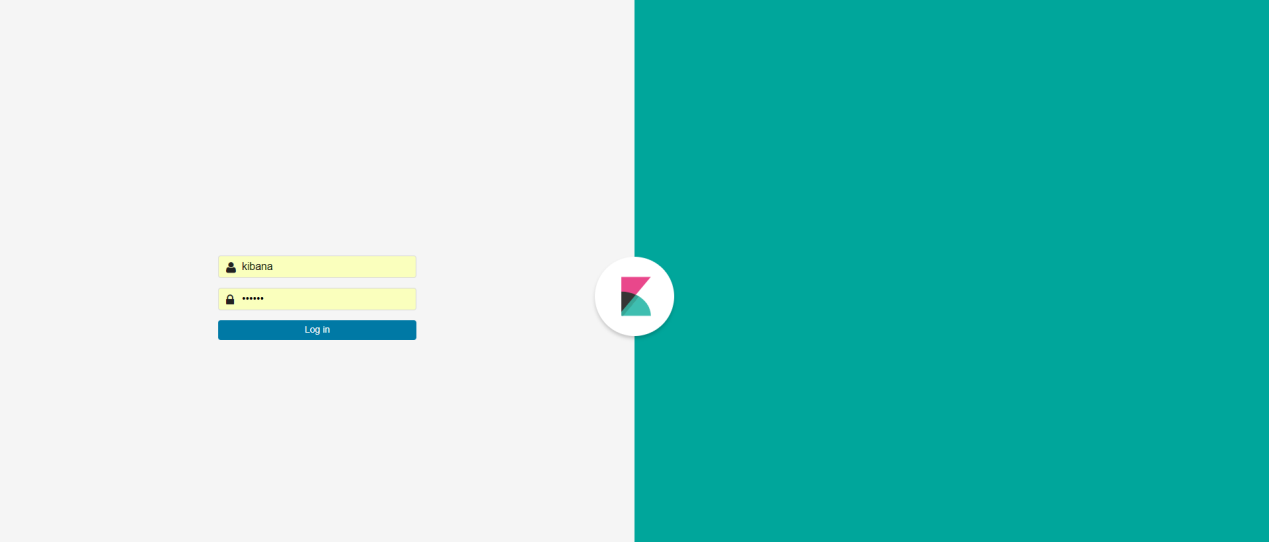
### 3.1.3 kibana启动

切换至目录 bin，启动kibana

|  |
| --- |
| #后台运行方式  ./kibana & |

在浏览器访问

|  |
| --- |
| http://10.1.2.105:5601/login#?\_g=() |



用户登录名/密码 见3.3 x-pack 权限配置

## 3.2 x-pack权限配置

如果es 配置文件中设置 elasticsearch.yml 配置需要将参数加上

##默认值为false

**xpack.security.enabled: true**

如果不配置该参数则 elastic 、logstash、kibana则不需要用户名、密码权限配置，则不需要x-pack 权限配置

### 3.2.1 初始化密码

如果对ES操作出现以下信息提示错误

|  |
| --- |
| {     **"error"**:{         **"root\_cause"**:[             {                 **"type"**:**"security\_exception"**,                 **"reason"**:**"failed to authenticate user [elastic]"**,                 **"header"**:{                     **"WWW-Authenticate"**:**"Basic realm="security" charset="UTF-8""**                 }             }         ],         **"type"**:**"security\_exception"**,         **"reason"**:**"failed to authenticate user [elastic]"**,         **"header"**:{             **"WWW-Authenticate"**:**"Basic realm="security" charset="UTF-8""**         }     },     **"status"**:**401** } |

输入默认用户和密码，就报这个错，猜想就是用户名和密码不对，可切换到x-pack目录重新生成目录

|  |
| --- |
| #后来我执行下面命令重新生成密码  bin/x-pack/setup-passwords auto |

|  |
| --- |
| [elsearch@localhost x-pack]$ ./setup-passwords auto  Initiating the setup of passwords for reserved users elastic,kibana,logstash\_system.  The passwords will be randomly generated and printed to the console.  Please confirm that you would like to continue [y/N]y  Changed password for user kibana  PASSWORD kibana = WpLRxuuoijfBYQRHj0uL  将密码更改为 kibana  Changed password for user logstash\_system  PASSWORD logstash\_system = Dd8don8K2VHj2U8arbAW  将密码更改为 logstash  Changed password for user elastic  PASSWORD elastic = AGsHeKgxzDYxQnDCWDJQ  将密码更改为 elastic |

参见网址：

<https://blog.csdn.net/u013066244/article/details/78698340>

### 3.2.2 更改密码

* 更改elasticsearch密码

|  |
| --- |
| curl -H "Content-Type: application/json" -X PUT -u elastic '10.1.2.105:9200/\_xpack/security/user/elastic/\_password' -d '{  "password" : "elastic"  }'; |

输入确认的密码为3.2.1 生成的后的密码

* 更改kibana密码

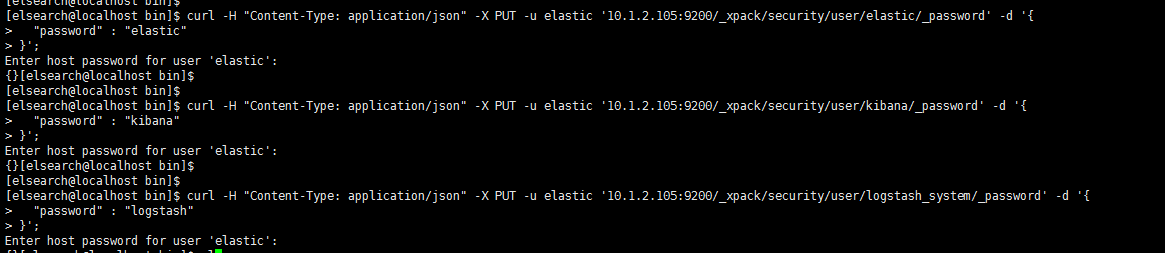
修改之前需要在kibana.yml中配置elasticsearch的用户名和密码后才能需改密码，否则会报错。

|  |
| --- |
| curl -H "Content-Type: application/json" -X PUT -u elastic '10.1.2.105:9200/\_xpack/security/user/kibana/\_password' -d '{  "password" : "kibana"  }' |

输入确认的密码为3.2.2 elasticsearch密码 更改后的密码elastic

* 更改logstash\_system 密码

|  |
| --- |
| curl -H "Content-Type: application/json" -X PUT -u elastic '10.1.2.105:9200/\_xpack/security/user/logstash\_system/\_password' -d '{  "password" : "logstash"  }' |



参见网址：

<https://blog.csdn.net/pistolove/article/details/53838138>

### 3.2.3 权限操作

* 查询所有User

|  |
| --- |
| curl -H "Content-Type: application/json" -XGET -u elastic '10.1.2.105:9200/\_xpack/security/user' |

返回结果

|  |
| --- |
| {  "elastic":{  "username":"elastic",  "roles":[  "superuser"  ],  "full\_name":null,  "email":null,  "metadata":{  "\_reserved":true  },  "enabled":true  },  "kibana":{  "username":"kibana",  "roles":[  "kibana\_system"  ],  "full\_name":null,  "email":null,  "metadata":{  "\_reserved":true  },  "enabled":true  },  "logstash\_system":{  "username":"logstash\_system",  "roles":[  "logstash\_system"  ],  "full\_name":null,  "email":null,  "metadata":{  "\_reserved":true  },  "enabled":true  }  } |

* 增加User

|  |
| --- |
| curl -XPOST -u elastic '10.1.2.105:9200/\_xpack/security/user/demo' -d  '{  "password":"123456",  "full\_name":" demo",  "email":"demo@163.com",  "roles":[  "clicks\_admin"  ]  }' |

* 查询具体User

|  |
| --- |
| curl -XGET -u elastic '10.1.2.105:9200/\_xpack/security/user/demo' |

* 删除具体User

|  |
| --- |
| curl -XDELETE -u elastic '10.1.2.105:9200/\_xpack/security/user/demo' |

Kibana安装参考网址：

<https://blog.csdn.net/cb2474600377/article/details/78963247>

<https://blog.csdn.net/mydeman/article/details/54743893>

<https://www.cnblogs.com/zhangs1986/p/7325504.html>

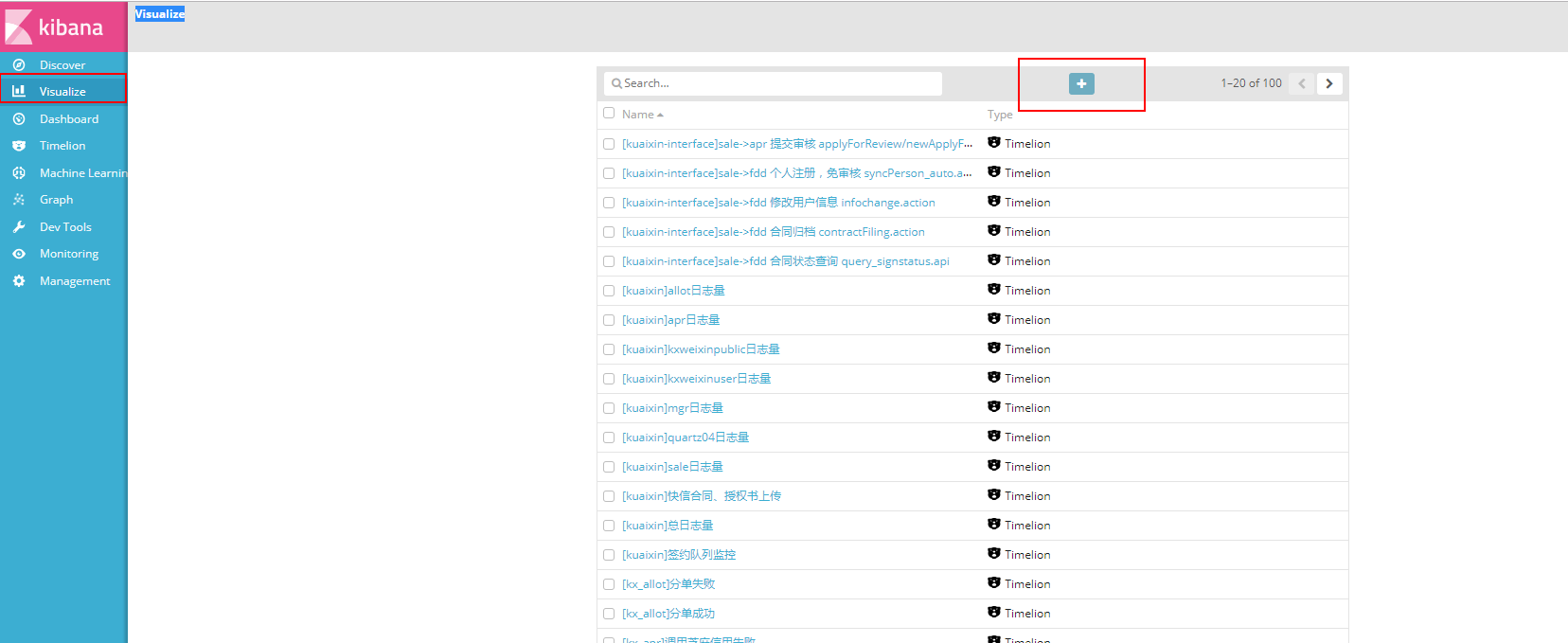
<https://www.cnblogs.com/zhangs1986/p/7325504.html>

## 3.3 Kibana使用

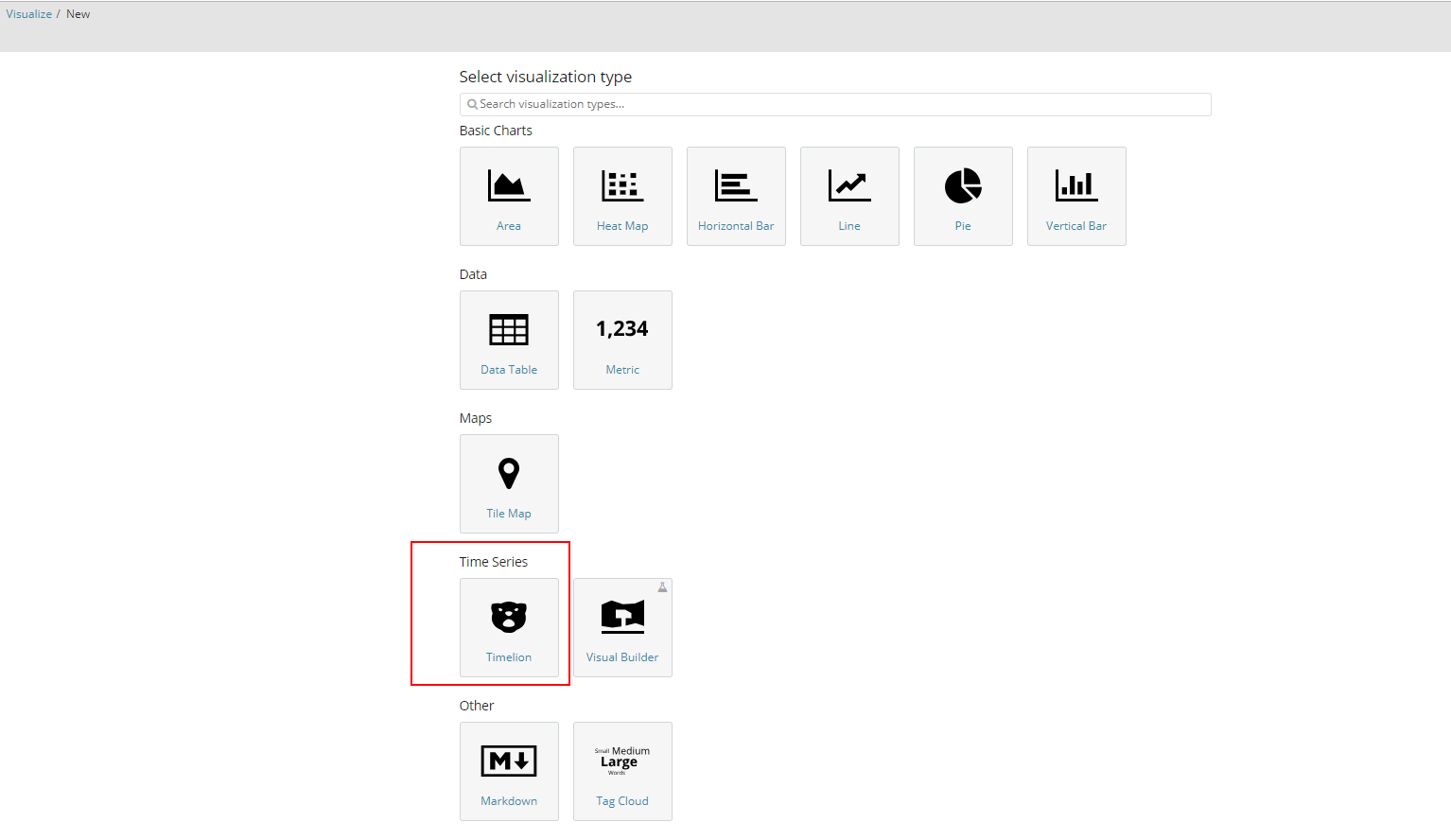
## 3.4 Kibana日志监控

### 3.4.1 es规则建立

点击左侧菜单 button [**Visualize**]



点击【】进入es 监控表达式



选择【Time Series】->【Timelion】 点击进去



[es 表达式]

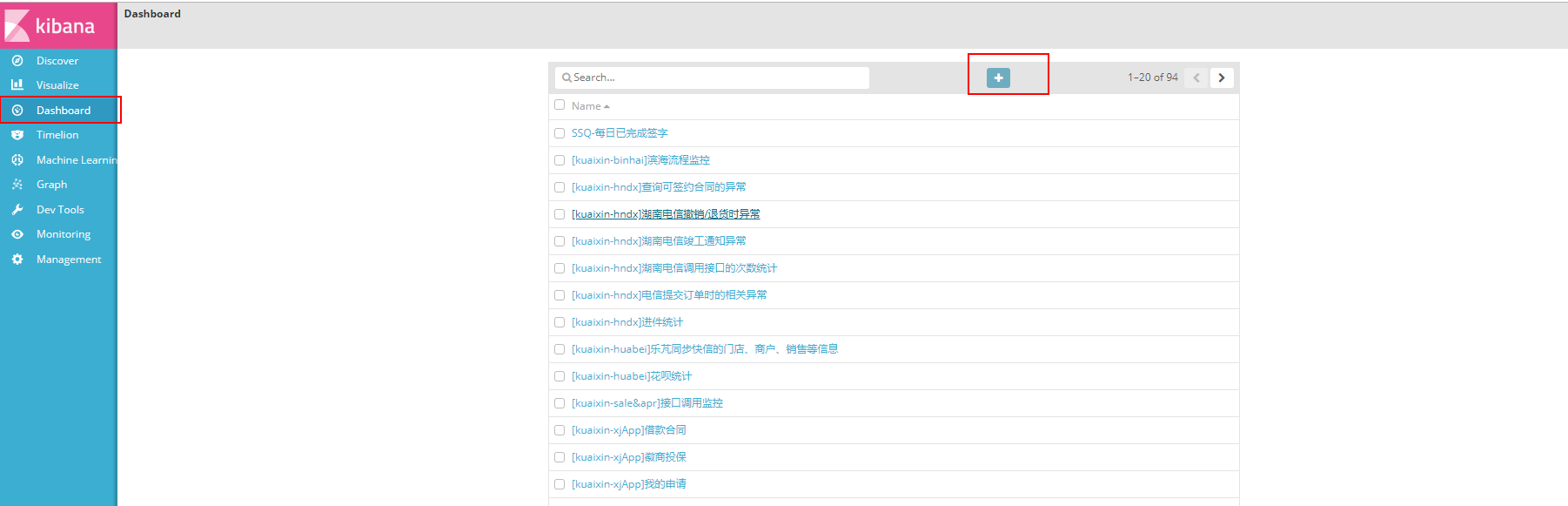
|  |
| --- |
| 表达式1  .es(timefield=time,index=kuaixin,q='projectName:sale AND logDetail:"for monitor:[createBHNormalLoan]"')  表达式2  .es(timefield=time,index=kuaixin,q='projectName:sale AND (logDetail:"invokeZaJudgeFundChannel end use ZADG" OR logDetail:"judgeFundChannel end use ZADG")'),.es(timefield=time,index=kuaixin,q='projectName:sale AND logDetail:"posZaJudgeFundChannel za not pass"'),.es(timefield=time,index=kuaixin,q='projectName:sale AND logDetail:"for monitor:[invokeSwitchFundChannelYL] "') |

点击【】生效

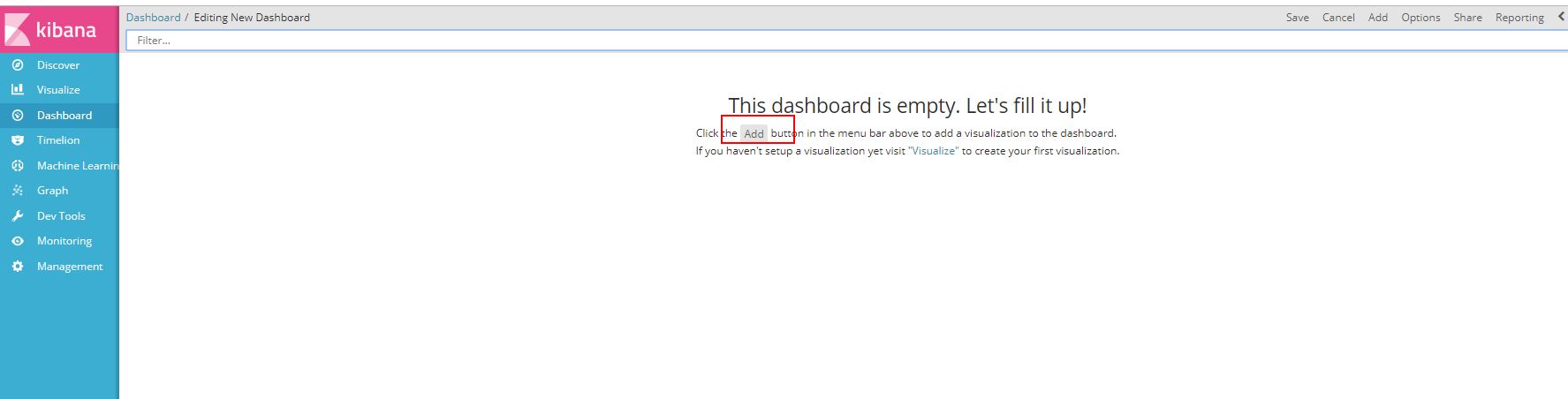
点击【save】保存

可以建多个  New Visualization  ，然后加至同一个dashbord中

### 3.4.2 dashbord 视图

点击左侧【Dashbord】

点击 [] 增加 Visualization



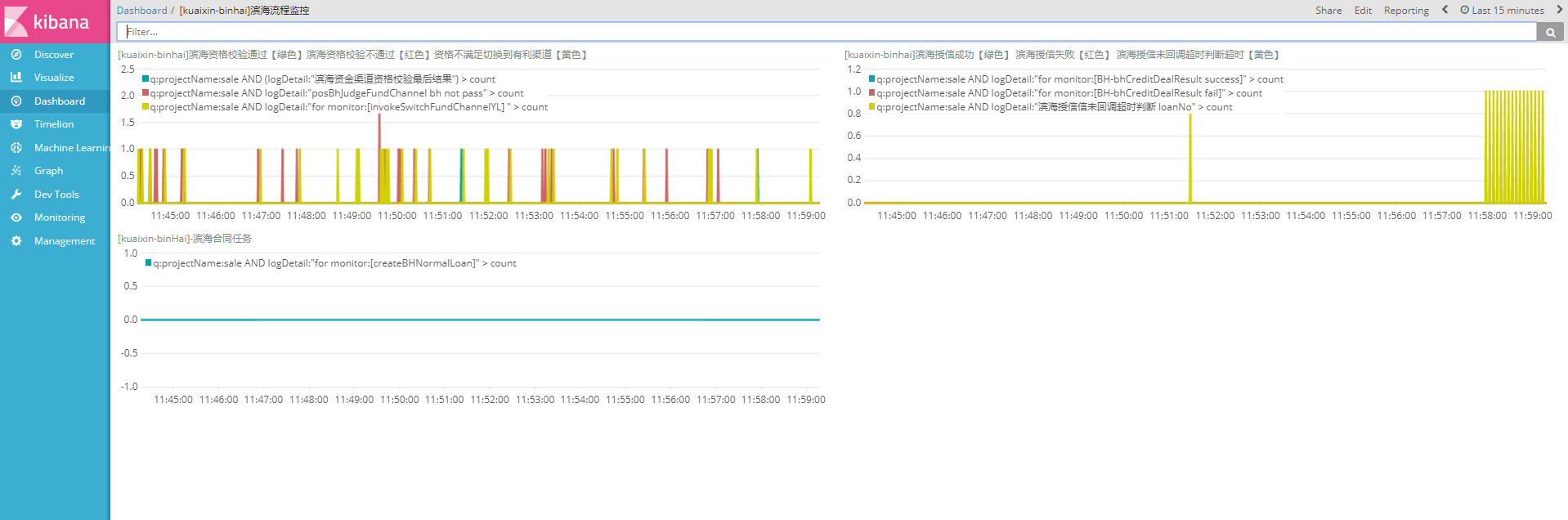
Click [Add] 点击之前定义的visualization，



可将多个  New Visualization  ，然后加至同一个dashbord中

然后点击右上方 【save】保存

### 3.4.3 流量监控



# 4. Logstash 部署安装

由于机器配置限制，将logstash-6.3.0.tar.gz 安装在V03机器 上

## 4.1 解压包

|  |
| --- |
| #目录切换  cd /data/server/es  ##解包  tar –zxvf logstash-6.3.0.tar.gz |

## 4.2 conf文件配置

|  |
| --- |
| cd /data/server/es/logstash-6.3.0/bin  mkdir conf  cd conf  vi logstash.conf |

此conf文件主要用于流程elk测试测，简单配置

|  |
| --- |
| input{  ##源数据 于从test.log，test.log主要是shell文件模拟追加生成数据  file{  # 要读取的日志文件，可用\*通配符匹配 例如["/tmp/log4j/\*.log"]  path=>["/data/server/es/logstash-6.3.0/bin/conf/test.log"]  # 从文件读取日志的间隔，单位：秒  stat\_interval=>5  }  file{  # 要读取的日志文件，可用\*通配符匹配 例如["/tmp/log4j/\*.log"]  path=>["/data/server/es/logstash-6.3.0/logs/\*.log"]  # 从文件读取日志的间隔，单位：秒  stat\_interval=>5  }  }  ##日志格式  filter{  grok{  match=>{  "message"=>"%{DATA:clientIp} - - \[%{HTTPDATE:accessTime}\] \"%{DATA:method} %{DATA:requestPath} %{DATA:httpversion}\" %{DATA:retcode} %{DATA:size} \"%{DATA:fromHtml}\" \"%{DATA:useragent}\""  }  remove\_field=>["message"]  }  date{  match=>["accessTime","dd/MMM/yyyy:HH:mm:ss Z"]  }  }    # 输出日志  output{  # file{  # # 要读取的日志文件，可用\*通配符匹配 例如["/tmp/log4j/\*.log"]  # path=>["/data/server/es/logstash-6.3.0/test/monitor2.log"]  # message\_format => "%{message}"  # }    ##写入es  elasticsearch{  hosts => ["10.1.2.107:9200"]  ##如果设置了用户名、密码就配置以下参数  #user => ["elastic"]  #password => ["elastic"]  index => "logstash-tomcat-log-%{+YYYY.MM.dd}"  }  ##写入redis  **#redis {**  **# host => '192.168.1.104'**  **# data\_type => 'list'**  **# key => 'logstash:redis'**  **# }**  # 输出到终端  stdout{  codec => rubydebug  }  } |

PS:

如果ES elasticSearch.yml配置文件设置了以下参数

**xpack.security.enabled: true**

而将日志写入ES时，没有配置用户名和密码

##如果设置了用户名、密码就配置以下参数

#user => ["elastic"]

#password => ["elastic"]

启动时报以下异常信息： 见问题5.10 问题10

Conf配置参见网址：

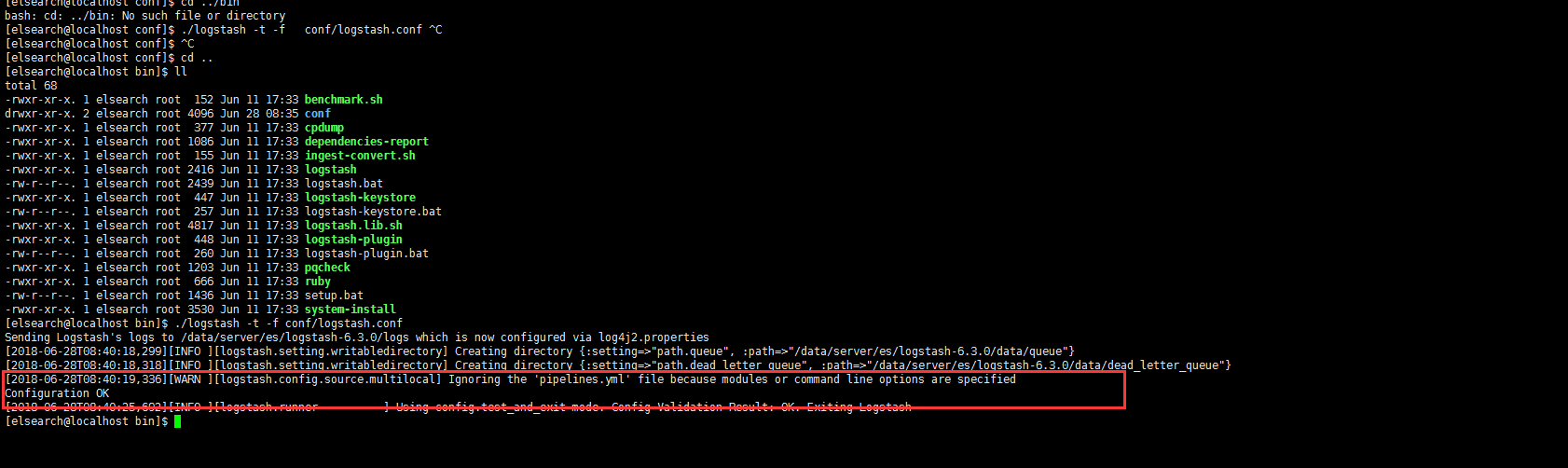
<https://blog.csdn.net/u010246789/article/details/52086799>

<https://blog.csdn.net/buqutianya/article/details/72019264?utm_source=itdadao&ut>

**https://doc.yonyoucloud.com/doc/logstash-best-practice-cn/index.html**

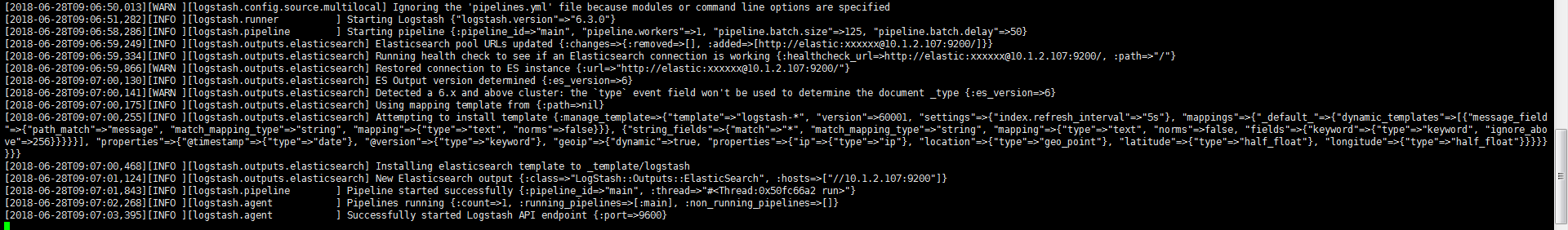
## 4.3 conf配置校验

|  |
| --- |
| cd /data/server/es/logstash-6.3.0/bin  ./logstash –t –f conf/ logstash.conf  ##结果如下图 |



## 4.4 启动logStash

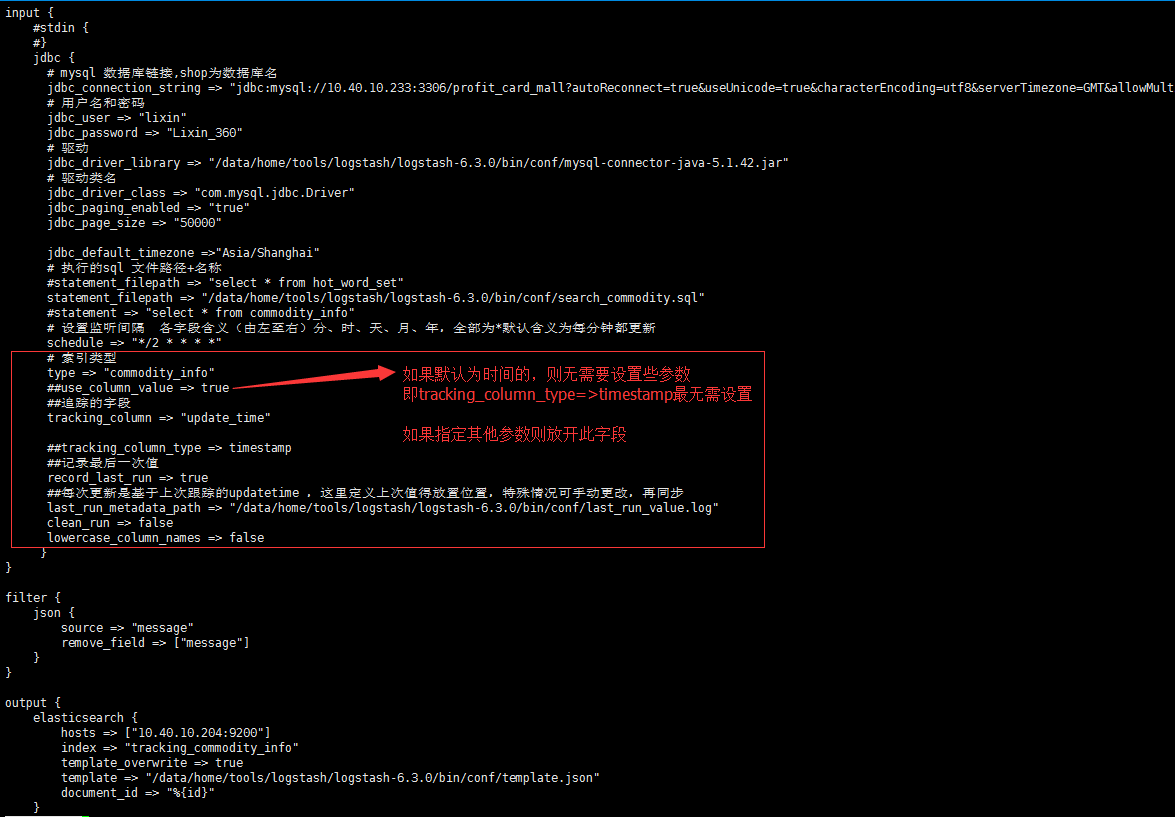
|  |
| --- |
| cd /data/server/es/logstash-6.3.0/bin  ./logstash –f conf/ logstash.conf &  ##启动日志如下 |



## 4.5 MYSQL 数据同步至 ES

将mysql 数据同步至ES，先配置mysql.conf文件

|  |
| --- |
| input {  stdin {  }  input {  #stdin {  #}  jdbc {  # mysql 数据库链接,shop为数据库名  jdbc\_connection\_string => "jdbc:mysql://10.40.10.233:3306/profit\_card\_mall?autoReconnect=true&useUnicode=true&characterEncoding=utf8&serverTimezone=GMT&allowMultiQueries=true"  # 用户名和密码  jdbc\_user => "lixin"  jdbc\_password => "Lixin\_360"  # 驱动  jdbc\_driver\_library => "/data/home/tools/logstash/logstash-6.3.0/bin/conf/mysql-connector-java-5.1.42.jar"  # 驱动类名  jdbc\_driver\_class => "com.mysql.jdbc.Driver"  jdbc\_paging\_enabled => "true"  jdbc\_page\_size => "50000"  jdbc\_default\_timezone =>"Asia/Shanghai"  # 执行的sql 文件路径+名称  #statement\_filepath => "select \* from hot\_word\_set"  statement\_filepath => "/data/home/tools/logstash/logstash-6.3.0/bin/conf/search\_commodity.sql"  #statement => "select \* from commodity\_info"  # 设置监听间隔 各字段含义（由左至右）分、时、天、月、年，全部为\*默认含义为每分钟都更新  schedule => "\*/5 \* \* \* \*"  # 索引类型  type => "commodity\_info"  ##use\_column\_value => true  ##追踪的字段  tracking\_column => "update\_time"  ##tracking\_column\_type => timestamp  ##记录最后一次值  record\_last\_run => true  ##每次更新是基于上次跟踪的updatetime ，这里定义上次值得放置位置，特殊情况可手动更改，再同步  last\_run\_metadata\_path => "/data/home/tools/logstash/logstash-6.3.0/bin/conf/last\_run\_value.log"  clean\_run => false  lowercase\_column\_names => false  }  }  filter {  json {  source => "message"  remove\_field => ["message"]  }  }  output {  elasticsearch {  hosts => ["10.40.10.204:9200"]  index => "tracking\_commodity\_info"  template\_overwrite => true  template => "/data/home/tools/logstash/logstash-6.3.0/bin/conf/template.json"  document\_id => "%{id}"  } |



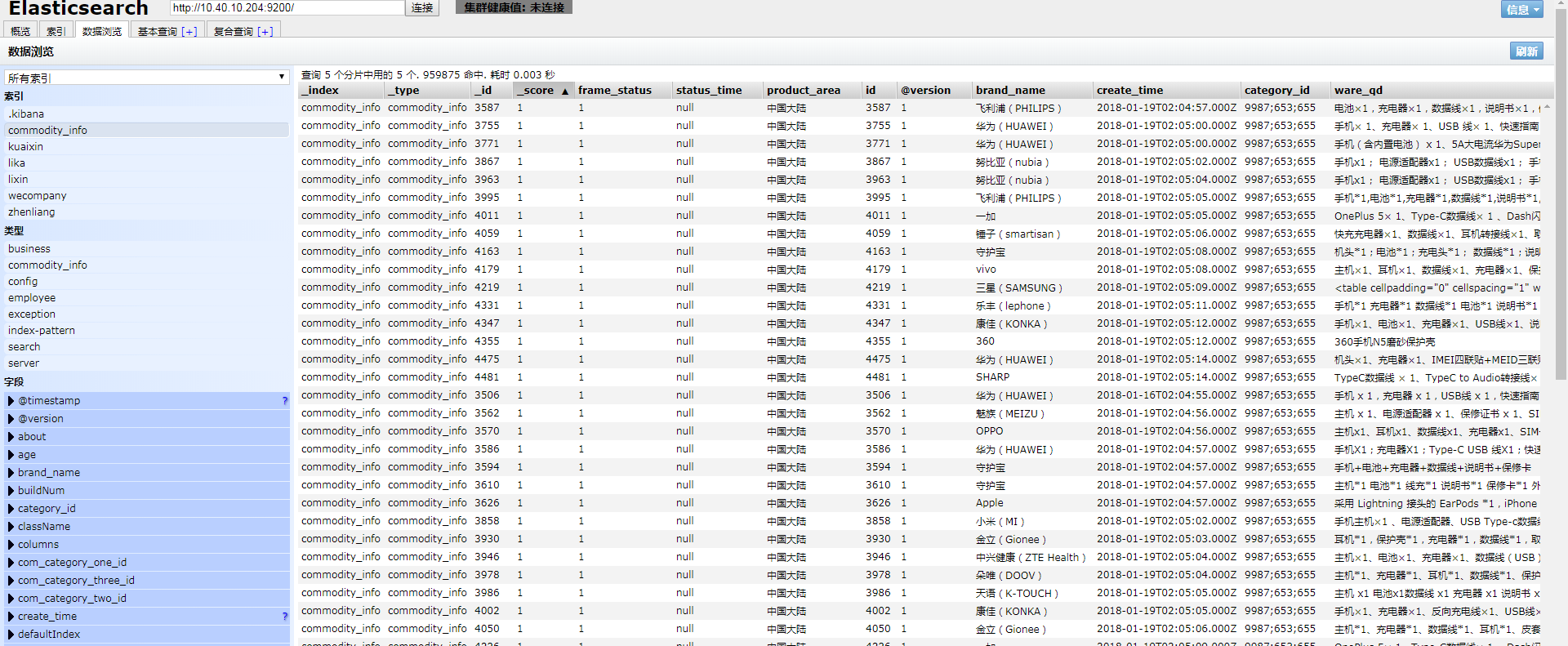
配置明细见：



<附见配置>

后续启动验证启动参照 4.4 步骤

商品数据同步至ES，通过head查找如下：



网址参考：

<https://blog.csdn.net/q15150676766/article/details/75949679>

<https://blog.csdn.net/plei_yue/article/details/78317980>

<https://blog.csdn.net/yeyuma/article/details/50240595>

https://my.oschina.net/xiaowangqiongyou/blog/1812708

数据增量同步

* 采用时间 作为追踪字段属性

##数据增加设置，以ID 或者 时间 作为追踪属性，将其存last\_run\_value.log文件中，下次再取出

<https://blog.csdn.net/opera95/article/details/78553743>

SQL demo:

|  |
| --- |
| SELECT  ci.\*, cs.sales\_count  FROM  commodity\_info\_bak ci  LEFT JOIN commodity\_sales cs ON ci.skuid = cs.sku\_id  where **ci.update\_time > date\_sub(:sql\_last\_value,INTERVAL 8 hour)** |

实际进一步研究发现，在配置文件中有use\_column\_value字段决定，是否需要记录某个column 的值,如果 record\_last\_run 为真,可以自定义我们需要 track 的 column 名称，此时该参数就要为 true. 否则默认 track 的是 timestamp 的值.

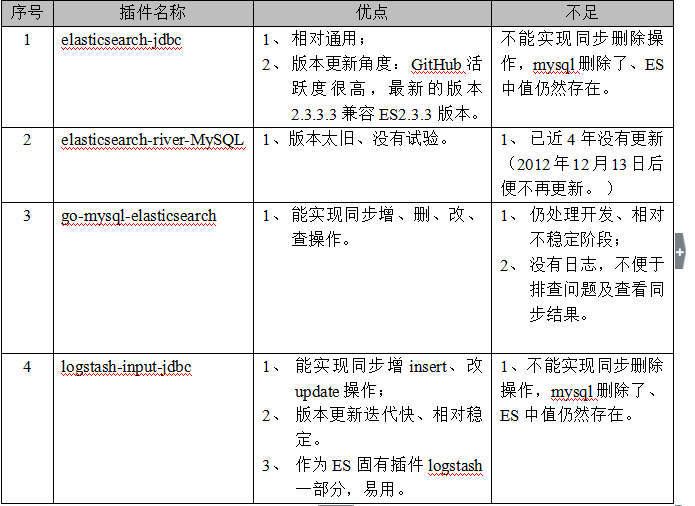
* 采用ID 作为追踪字段属性

|  |
| --- |
| select  \*  from  cc  where cc.id >= :sql\_last\_value |

我们可以指定文件,来记录上次执行到的 tracking\_column 字段的值 比如上次数据库有 12 条记录,查询完后该文件中就会有数字 12 这样的记录,下次执行 SQL 查询可以从 13 条处开始.



几种常用数据库MYSQL-ES同步插件优点缺点：



参考网址：

https://www.cnblogs.com/licongyu/p/5383334.html

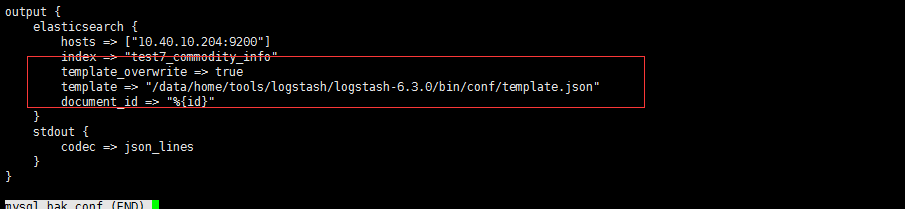
<https://my.oschina.net/xiaowangqiongyou/blog/1812708>

<https://blog.csdn.net/laoyang360/article/details/51793301>

<https://www.cnblogs.com/a-du/p/7611620.html>

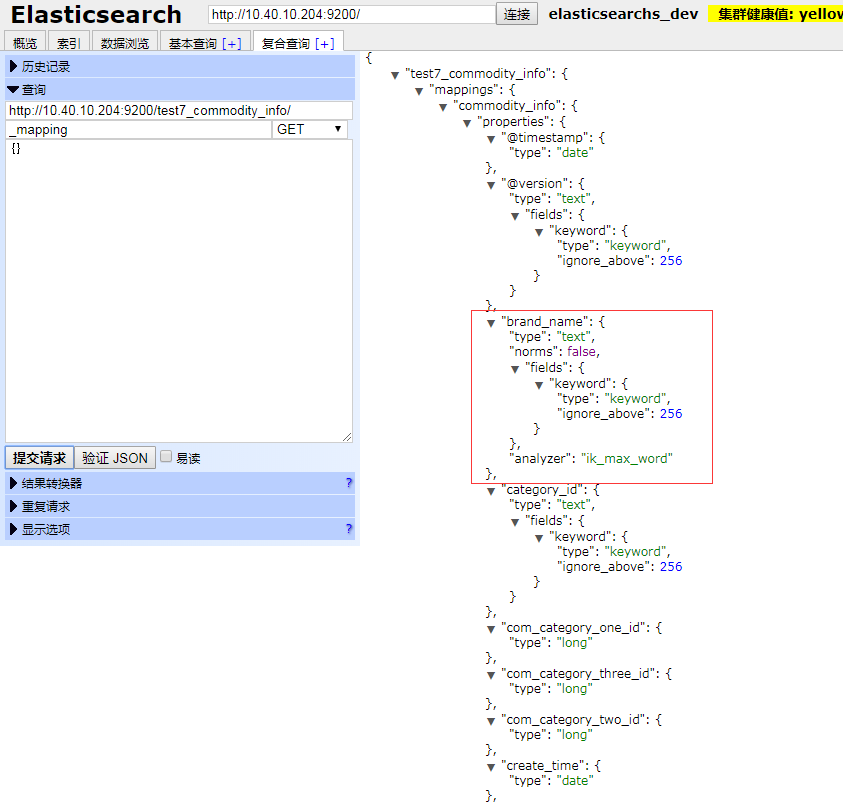
## 4.6 logstash使用ES的template创建索引

希望logstash在自动创建索引的时候使用ES中定义好的template，配置好之后，logstash自动创建的索引都不是我定义的template，请问要如何让logstash使用ES中定义的template创建索引



**template.json 文本如下**

可通过以下查到原始index mappings,在原有mapping添加以分词器



将name 、brandname 设置了分词器 "analyzer":"ik\_max\_word"

****

参考网址：

<https://www.cnblogs.com/jstarseven/p/7707499.html>

<https://www.cnblogs.com/cocowool/p/elk_dynamic_templates.html>

<https://blog.csdn.net/u012516166/article/details/75106184>

<https://elasticsearch.cn/question/2726>

# 5. 插件IK+pinyin分词器安装

## 5.1 下载ik+pinyin安装包

Ik分词器

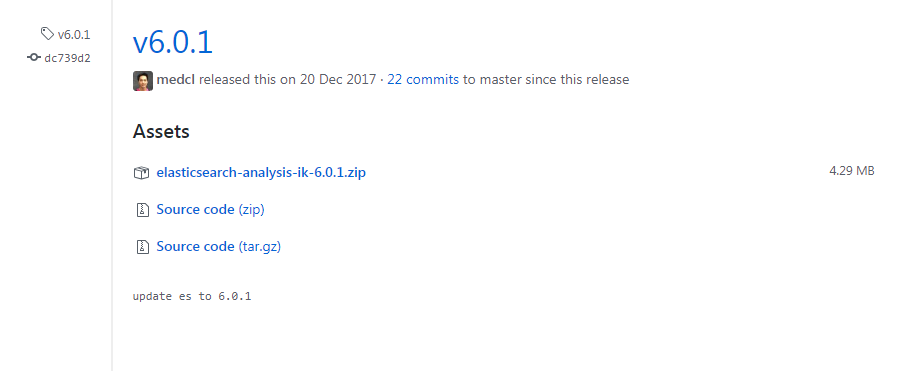
<https://github.com/medcl/elasticsearch-analysis-ik/releases>

pinyin分词器

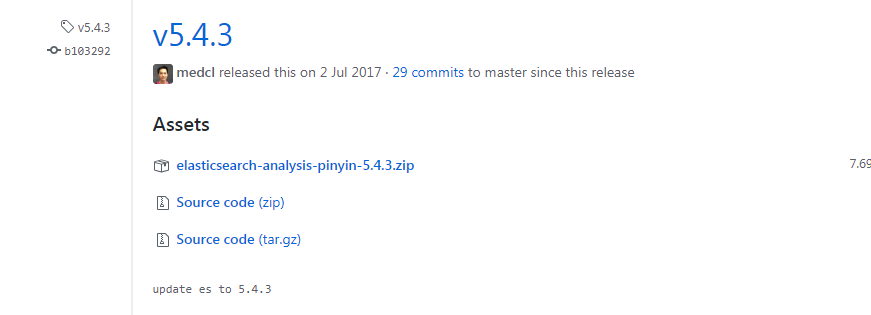
<https://github.com/medcl/elasticsearch-analysis-pinyin/releases>

注：一定要与ES版本保持一致(我ES用的是6.0.0所以ik也要下载6.0.0)，

如下图：



<ik>



<pinyin>

如果没有下载到己打包直接可以使用的zip， 可采用源码的形式，用maven命令进行打包

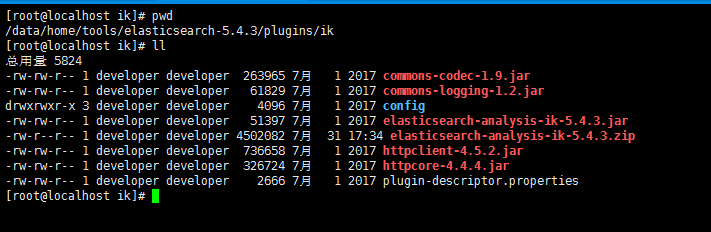
## 5.2. 安装

在目录 /data/home/tools/elasticsearch-5.4.3/plugins 新建文件夹 ik,pinyin

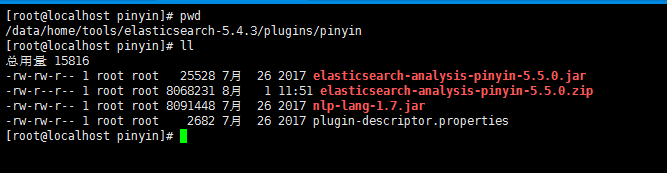
将下载文件上传至 ik 目录下，并解压

|  |
| --- |
| ##中文分词器  mkdir ik  cd ik  rz elasticsearch-analysis-ik-5.4.3.zip  unzip elasticsearch-analysis-ik-5.4.3.zip  ## 拼音分词器  mkdir pinyin  cd pinyin  rz elasticsearch-analysis-pinyin-5.5.0.zip  unzip elasticsearch-analysis-pinyin-5.5.0.zip |

如果为ES 集群，每个ES 节点都需按装插件分词器



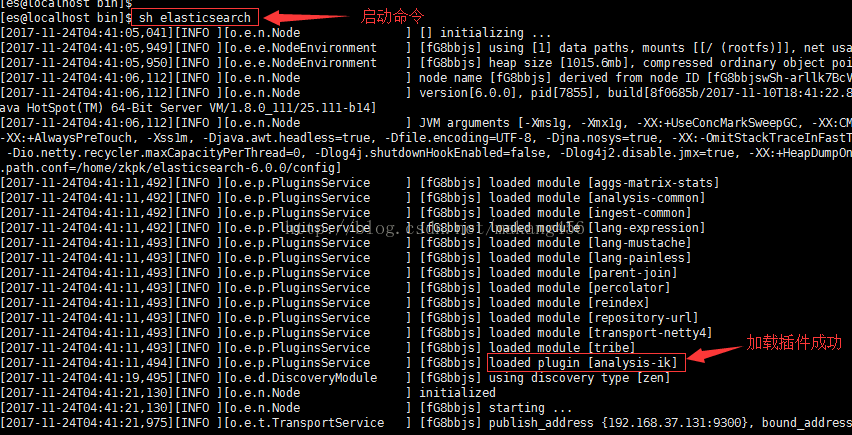
<elasticsearch-analysis-ik>

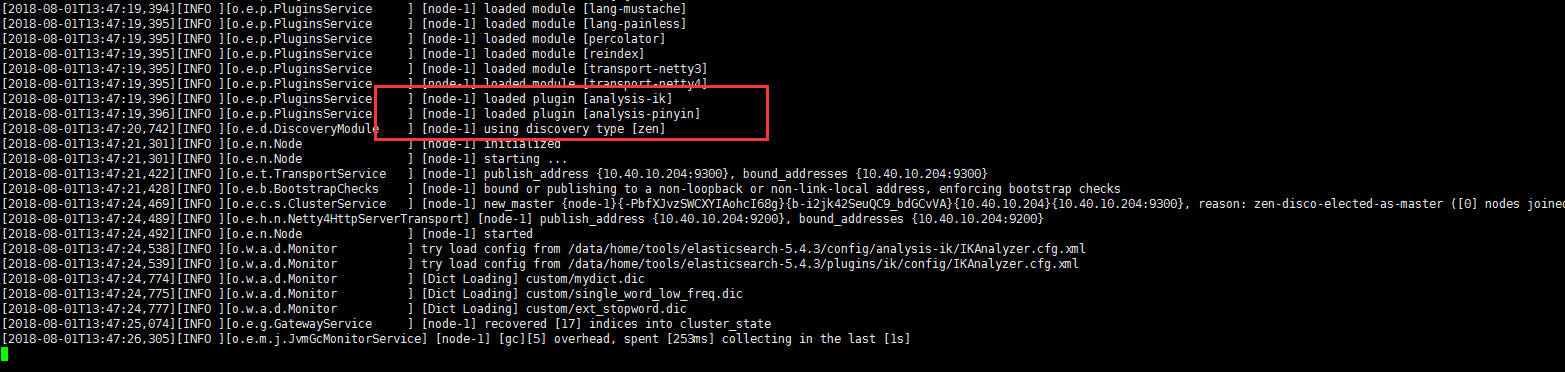


<elasticsearch-analysis-pinyin>

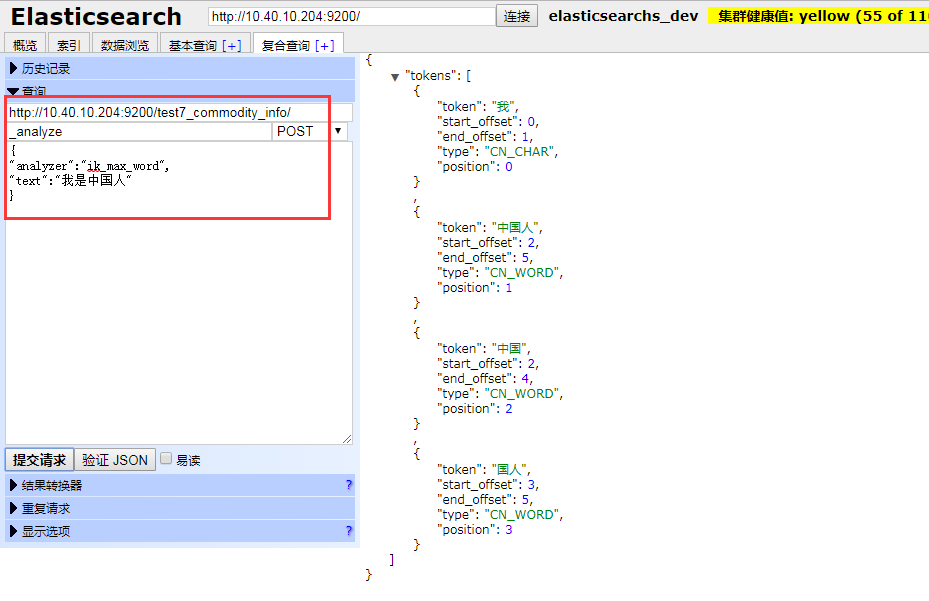
## 5.3 重新启动ES

若为ES 为集群，每个节点都需要重启

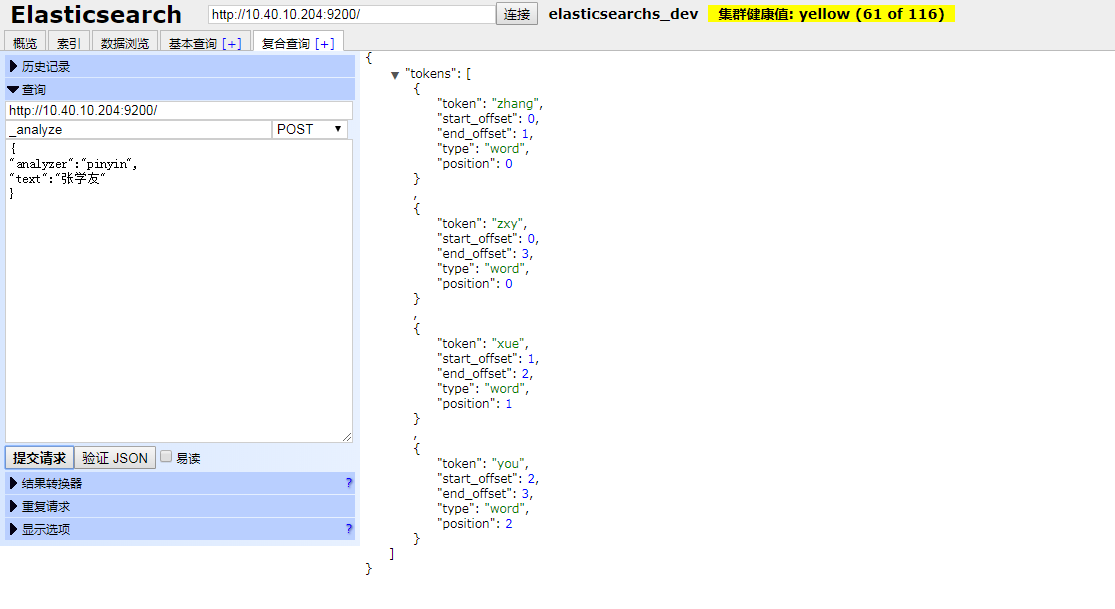




## 5.4 测试ik插件(通过head插件测试)



<ik>



<pinyin>

## 5.5 版本差异分词器使用

ES2.0版本与ES5.0以后的版本对比以及注意事项

* 5.0以后移除名为 ik 的analyzer和tokenizer,请分别使用 ik\_smart 和 ik\_max\_word(也就是5.0之前进行测试的时候需要将上面中的ik\_max\_word换成ik关键字)
* ES6.0与ik6.0进行集成安装之后，只能通过head工具来进行测试，通过url进行测试的时候会报错，这是一个bug已经提交到gethub上了，希望社区管理能及时更正(亲测5.5.1版本没问题)。
* ES5.0之后的ik\_smart和ik\_max\_word说明

ik\_max\_word: 会将文本做最细粒度的拆分，比如会将“我是中国人”拆分为“我，是，中国人，中国，国人”，会穷尽各种可能的组合。

ik\_smart: 会做最粗粒度的拆分，比如会将“我是中国人”拆分为“我，是，中国人”。

参考网址

<https://blog.csdn.net/makang456/article/details/78623397>

<https://www.cnblogs.com/youran-he/p/7381291.html>

<https://www.cnblogs.com/zlslch/p/6440373.html>

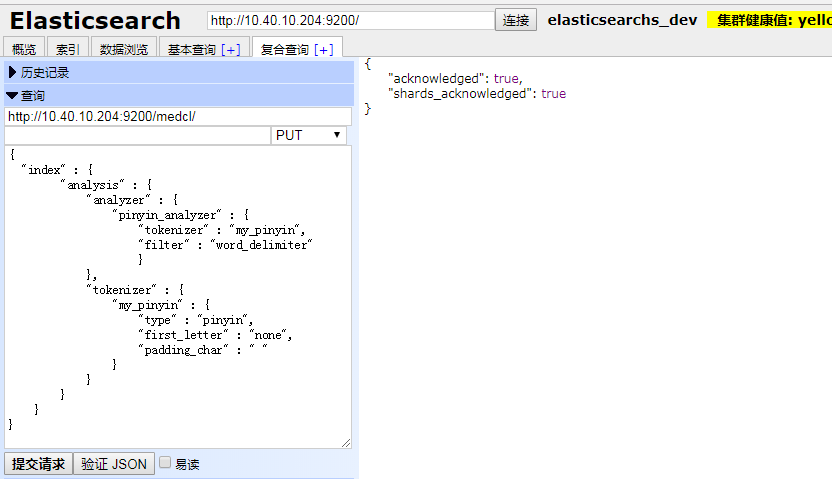
<https://github.com/medcl/elasticsearch-analysis-ik/releases?after=v5.6.4>

## 5.6 IK+Pinyin分词配置

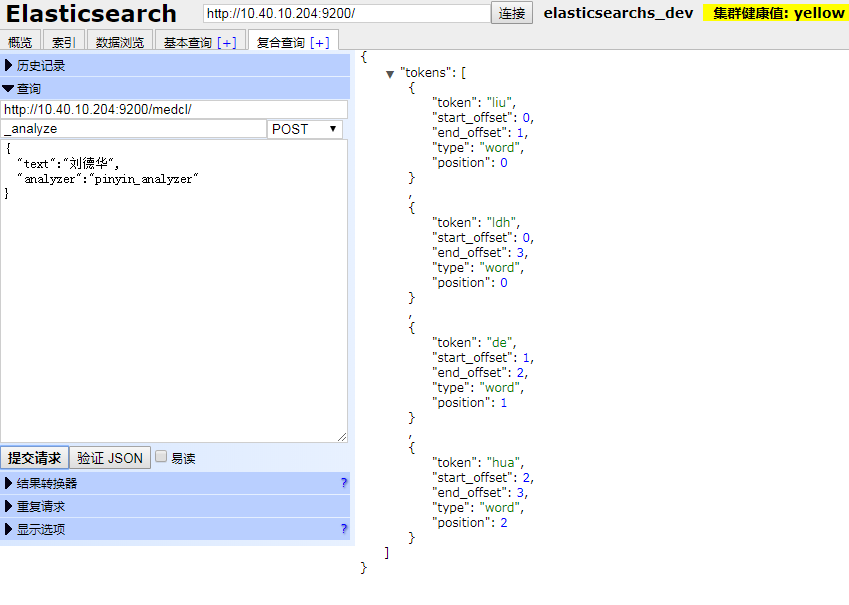
### 5.6.1 创建索引与分析器设置

创建一个索引，并设置index分析器相关属性（可在模板里一并设置见 template\_pinyin.json）:

|  |
| --- |
| {     **"index"**:{         **"analysis"**:{             **"analyzer"**:{                 **"pinyin\_analyzer"**:{                     **"tokenizer"**:**"my\_pinyin"**                 }             },             **"tokenizer"**:{                 **"my\_pinyin"**:{                     **"type"**:**"pinyin"**,                     **"keep\_separate\_first\_letter"**:**false**,                     **"keep\_full\_pinyin"**:**true**,                     **"keep\_original"**:**true**,                     **"limit\_first\_letter\_length"**:**16**,                     **"lowercase"**:**true**,                     **"remove\_duplicated\_term"**:**true**                 }             }         }     } } |



然后使用analyze api，进行测试



如果分词器设置为pinyin\_first\_letter，则分析的结果为：

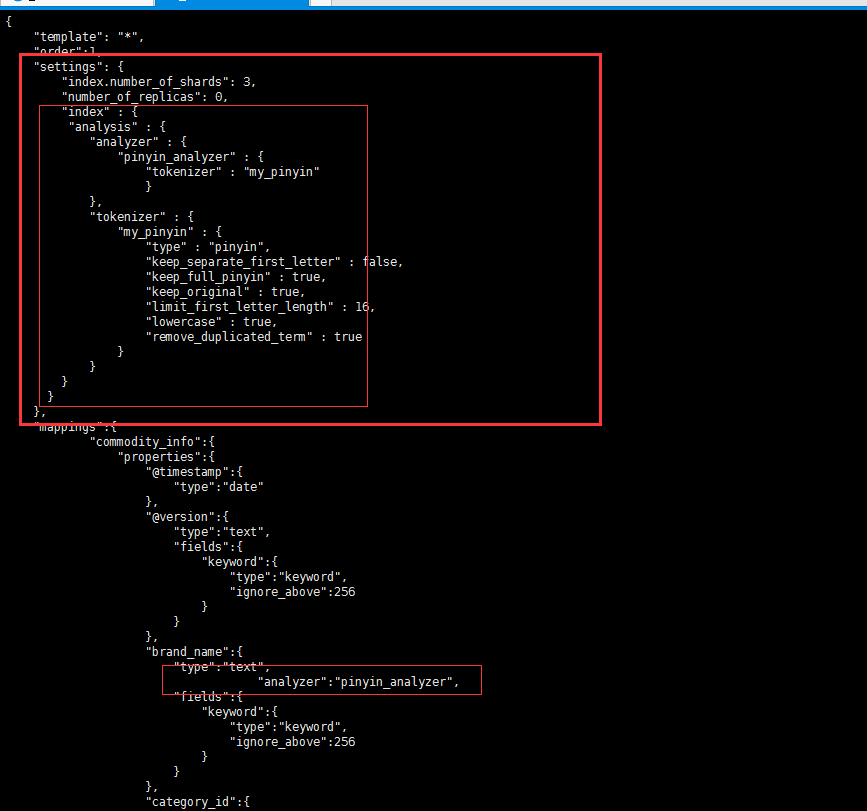
|  |
| --- |
| {     **"tokens"**:[         {             **"token"**:**"ldh"**,             **"start\_offset"**:**0**,             **"end\_offset"**:**3**,             **"type"**:**"word"**,             **"position"**:**0**         }     ] } |

### 5.6.2 创建type并设置mapping

创建一个type并设置mapping,针对属性值设置对应的分词器

在设置mapping 时，可指定es词组解析器，对于属性值可单独指定

**"analyzer":"pinyin\_analyzer"##pinyin**





参考网址：

<https://github.com/medcl/elasticsearch-analysis-pinyin>

<https://www.cnblogs.com/Leo_wl/p/6100077.html>

<https://blog.csdn.net/q7867401/article/details/78815799>

<https://blog.csdn.net/opera95/article/details/78594949>

# 6. 索引设置模板-template设置

## 6.1 template 基础命令

|  |
| --- |
| ##获取所有模板信息  curl -XGET <http://10.40.10.204:9200/_template>  #删除模板信息，logstash 值为模板名称  curl -XDELETE <http://10.40.10.204:9200/_template/logstash>  ##模板设置  curl -XPUT <http://10.40.10.204:9200/_template/template_4> -d ‘  {  #json格式 模板设置  }  ’ |

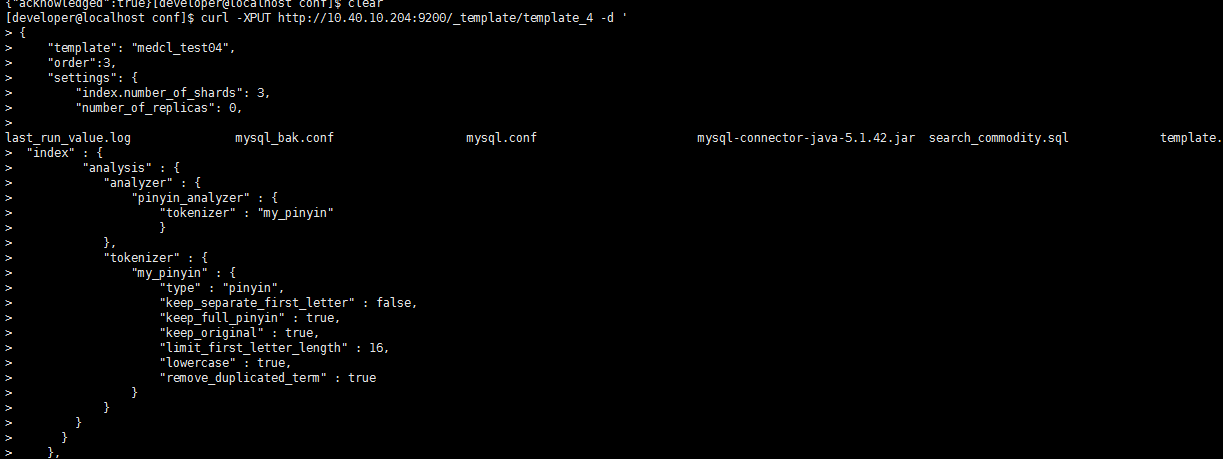
## 6.2 模板设置

|  |
| --- |
| curl -XPUT http://10.40.10.204:9200/\_template/template\_4 -d '  {  "template": "medcl\_test04",##此值最匹配index 索引值，在创建索引时，会自动匹配  "order":3,  "settings": {  "index.number\_of\_shards": 3,  "number\_of\_replicas": 0,  "index" : {  "analysis" : {  "analyzer" : {  "pinyin\_analyzer" : {  "tokenizer" : "my\_pinyin"  }  },  "tokenizer" : {  "my\_pinyin" : {  "type" : "pinyin",  "keep\_separate\_first\_letter" : false,  "keep\_full\_pinyin" : true,  "keep\_original" : true,  "limit\_first\_letter\_length" : 16,  "lowercase" : true,  "remove\_duplicated\_term" : true  }  }  }  }  },  "mappings":{  "commodity\_info":{  "properties":{  "@timestamp":{  "type":"date"  },  "@version":{  "type":"text",  "analyzer":"pinyin\_analyzer",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "brand\_name":{  "type":"text",  "analyzer":"pinyin\_analyzer",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "category\_id":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "com\_category\_one\_id":{  "type":"long"  },  "com\_category\_three\_id":{  "type":"long"  },  "com\_category\_two\_id":{  "type":"long"  },  "create\_time":{  "type":"date"  },  "frame\_status":{  "type":"long"  },  "id":{  "type":"long"  },  "image\_path":{  "type":"text",  "analyzer":"pinyin\_analyzer",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "is\_can\_vat":{  "type":"boolean"  },  "is\_delete":{  "type":"long"  },  "is\_update\_price":{  "type":"long"  },  "lower\_limit":{  "type":"long"  },  "min\_month\_repay":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "name":{  "type":"text",  "analyzer":"pinyin\_analyzer",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "product\_area":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "profit\_retail\_price":{  "type":"float"  },  "profit\_state":{  "type":"long"  },  "property":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "purchase\_price":{  "type":"float"  },  "reserve":{  "type":"long"  },  "retail\_price":{  "type":"float"  },  "sale\_state":{  "type":"long"  },  "sale\_unit":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "sku":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "skuid":{  "type":"long"  },  "status\_time":{  "type":"date"  },  "supplier":{  "type":"long"  },  "tax\_code":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "type":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "upc":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "update\_time":{  "type":"date"  },  "upper\_limit":{  "type":"long"  },  "ware\_qd":{  "type":"text",  "fields":{  "keyword":{  "type":"keyword",  "ignore\_above":256  }  }  },  "weight":{  "type":"float"  }  }  }  }  }' |



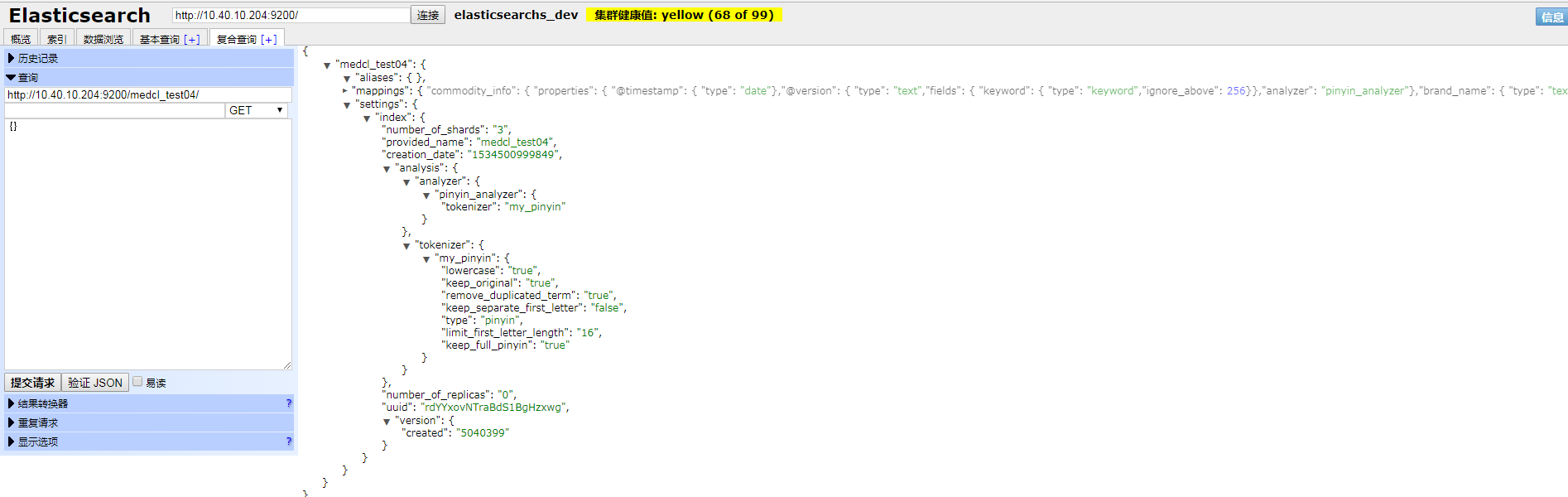
如：查看所模板信息

模板名称：medcl\_test04 刚新增模板，



Template 所命名的值，此值最匹配index 索引值，在创建索引时，会自动匹配

如创建锁引值为：medcl\_test04 ， 就会自动去匹配值设置模板值



## 6.3 网址参考

<https://www.cnblogs.com/mikeluwen/p/8031760.html>

# 7. Grafana监控

## 7.1 Grafana 获取下载

Grafana官网：<https://grafana.com/grafana/download>

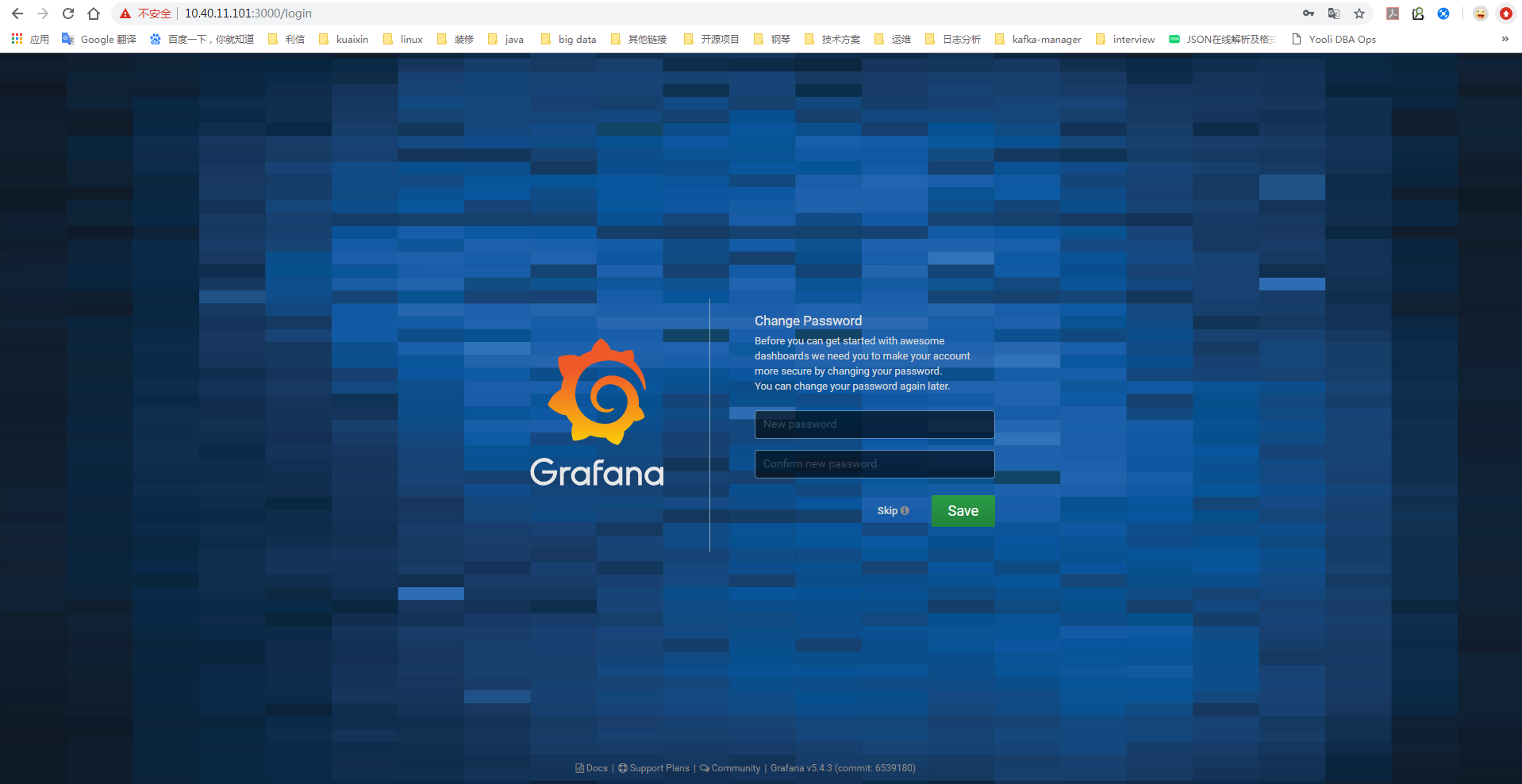
|  |
| --- |
| mkdir /data/server/grafana  cd /data/server/grafana  wget <https://dl.grafana.com/oss/release/grafana-5.4.3.linux-amd64.tar.gz> |

## 7.2 启动安装

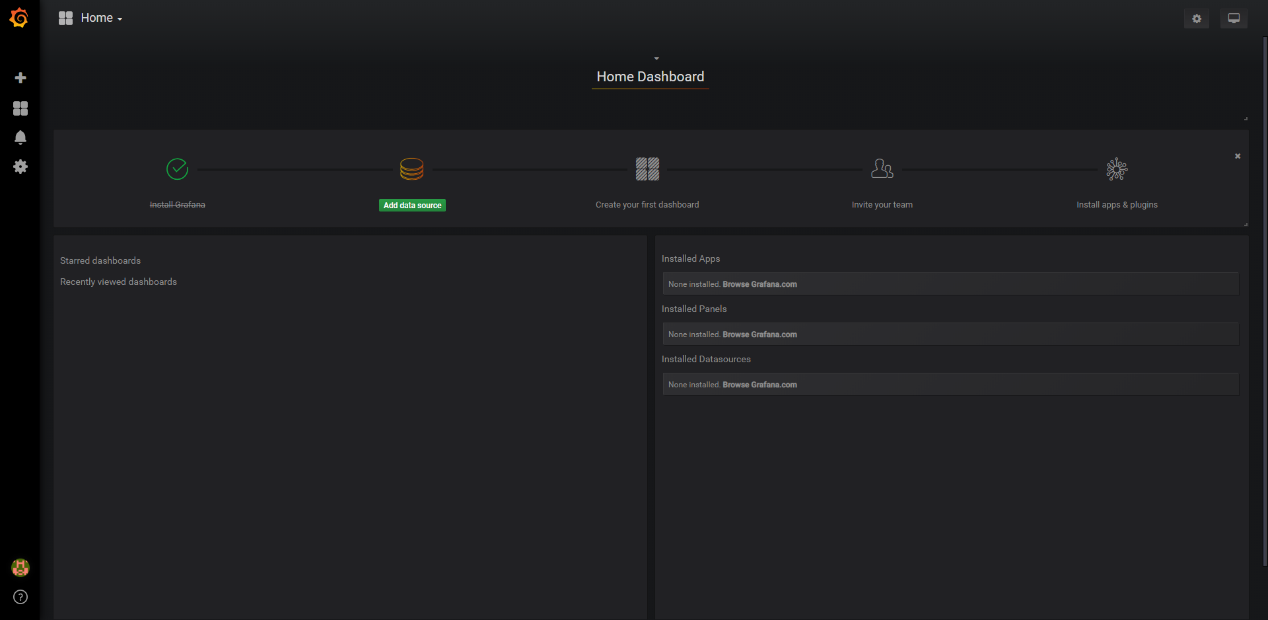
|  |
| --- |
| cd /data/server/grafana  tar -zxvf grafana-5.4.3.linux-amd64.tar.gz  cd /data/server/grafana/grafana-5.4.3/bin  ## 启动 默认端口3000  ./grafana-server & |

IP 访问：

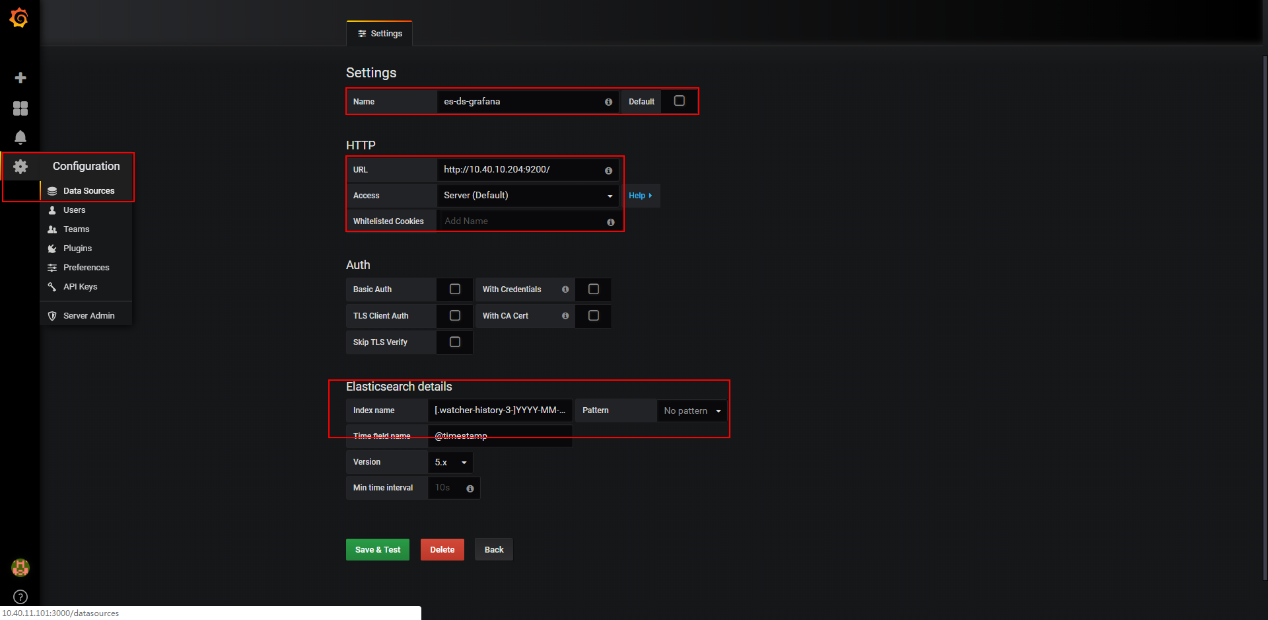
<http://10.40.11.101:3000/login>



默认 用户名/密码为 admin/admin登录后信息提示更改密码，设置新密码为 aabb1122



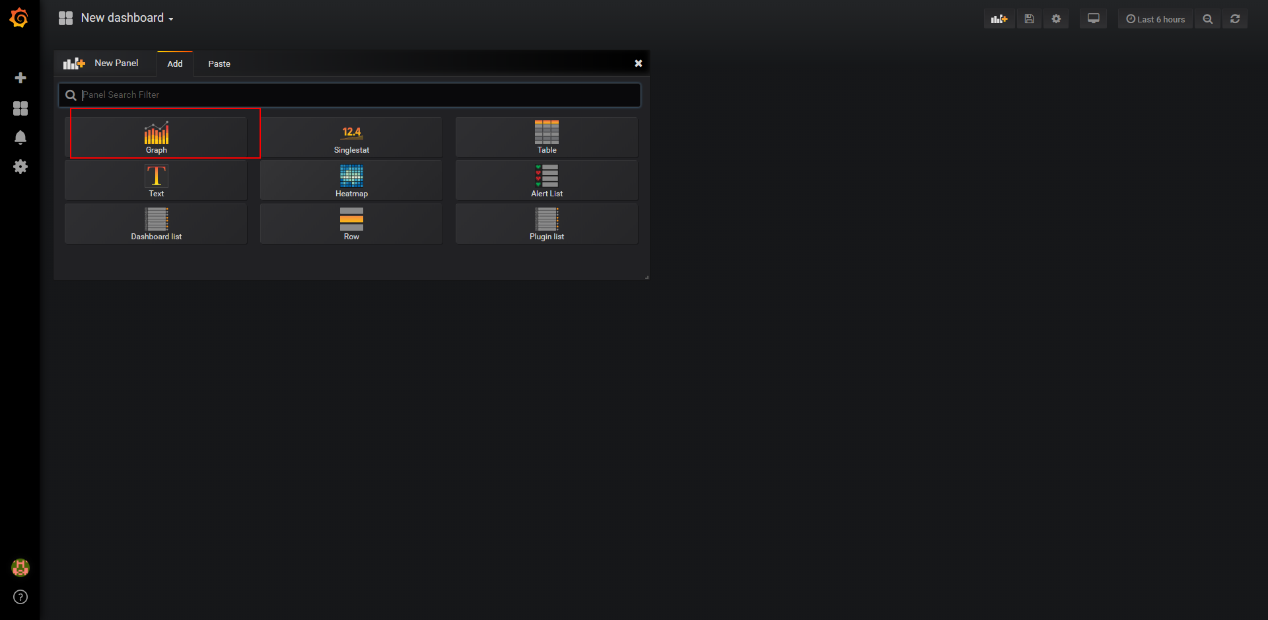
## 7.3 配置日志数据源

Grafana 支持多种数据源，因为我司现在是基于ES 存储日志数据，所以本次配置数据源为 ES ，地址 <http://10.40.10.204:9200/>

|  |
| --- |
| 字段说明：  name：数据源名称。本样例值：es-ds-grafana。  Type：数据源类型，这里选择Elasticsearch。  URL：是Elasticsearch服务器的URL。本样例值：http://10.40.10.204:9200/  Access：访问方式。选择默认方式  Index name：索引名称。  本样例：[.watcher-history-3-]YYYY.MM.DD  Pattern：选择hourly。因为我们的索引是按照小时分的，所以这里选择houry。  Version：选择你安装的相应的版本。 |

## 7.4 Graph添加

New dashboard->Add->Graph



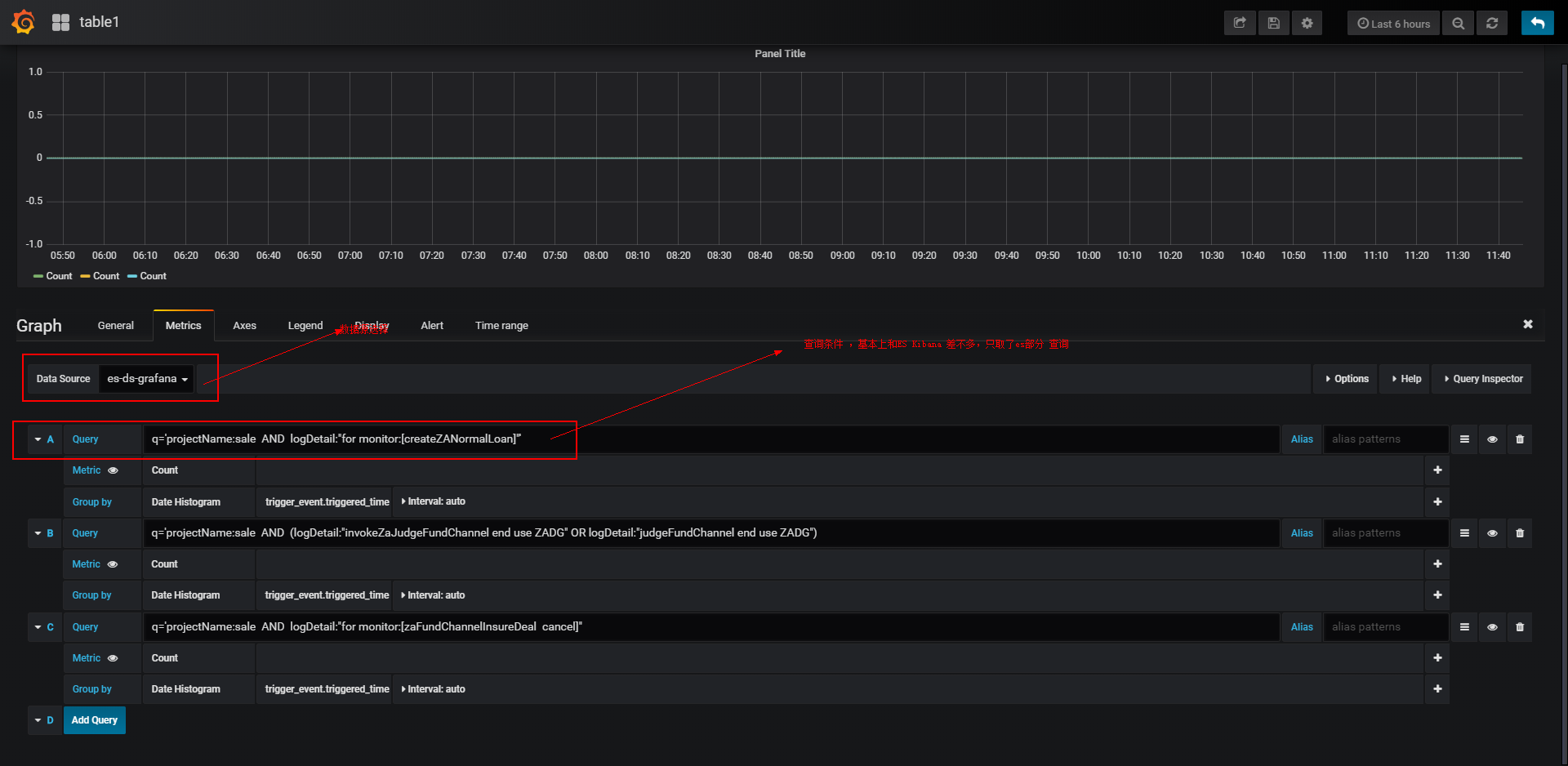
点击编缉Edit

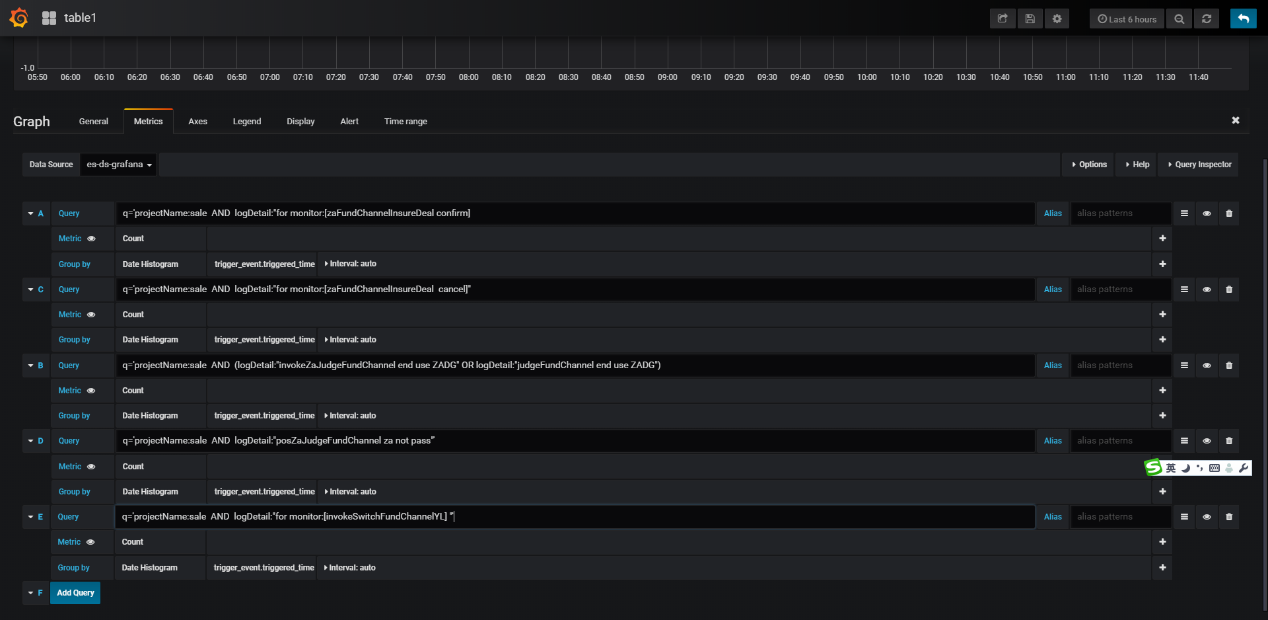


选择数据源 es-ds-grafana （7.1 步骤己设置）

添加检索查询条件，查询表达式与3.4 Kibana日志监控 ES 表达式几乎一致，表达式规则为

|  |
| --- |
| Query1:  q='projectName:sale AND (logDetail:"invokeZaJudgeFundChannel end use ZADG" OR logDetail:"judgeFundChannel end use ZADG")  Query2:  q='projectName:sale AND logDetail:"for monitor:[zaFundChannelInsureDeal confirm]"' |





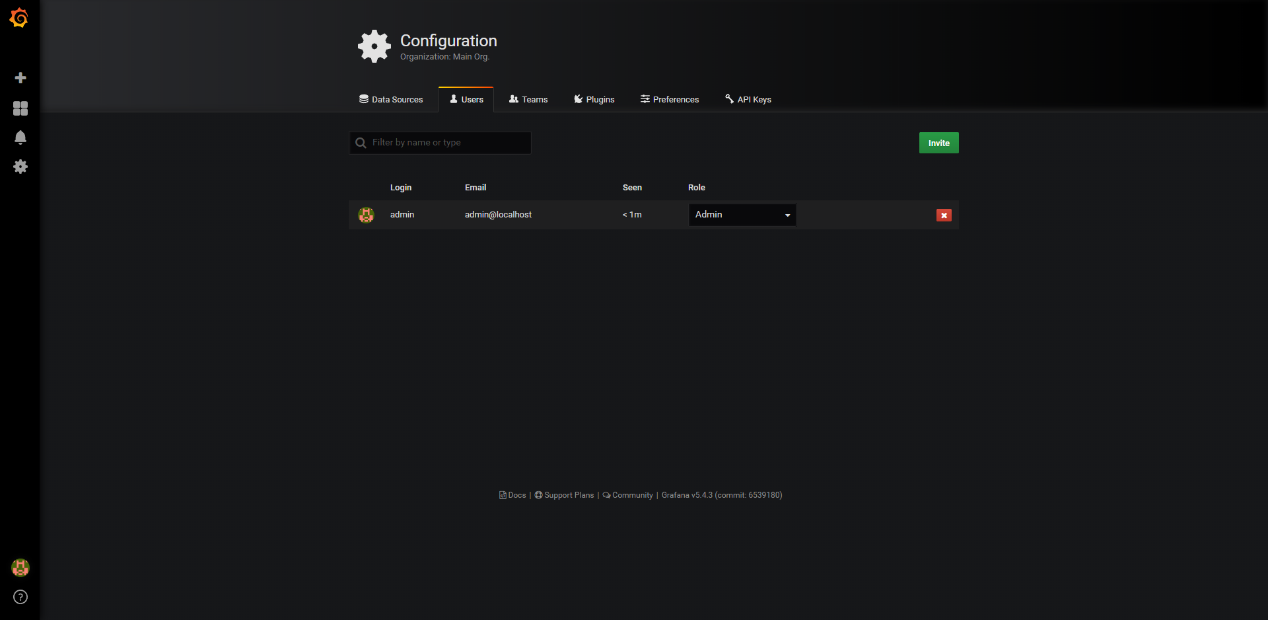
点击 右上角 保存

## 7.5 用户权限设置

graph 用户权限 有3个角色 Viewer、Editor 、Admin

其中 Viewer 只可查看 不可编缉 、Editor 可编缉状态、Admin 管理员

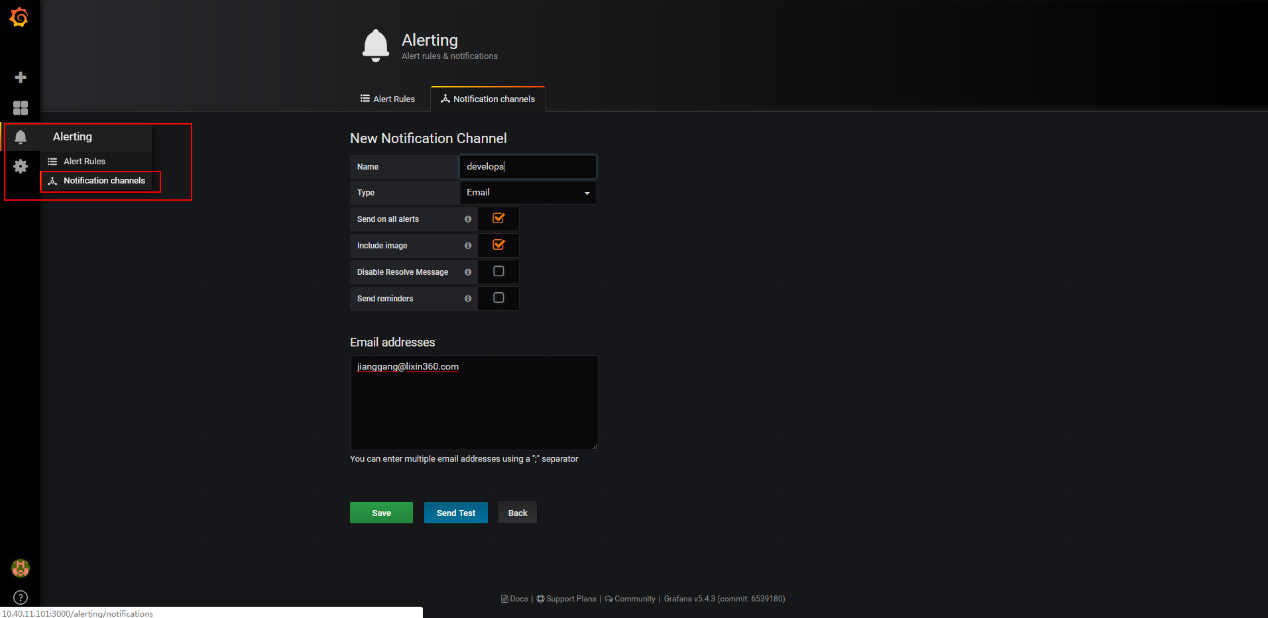
注册用户一般权限只有查看功能 即角色权限：Viewer ，亦可设置分组



## 7.6 设置Alert阈值和邮件报警

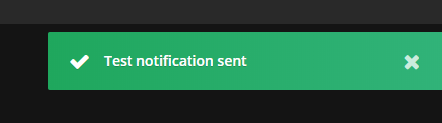
### 7.6.1 notification channel 通知渠道设置

grafana页面添加一个notification channel，即通知渠道，grafana支持众多的报警通知方式



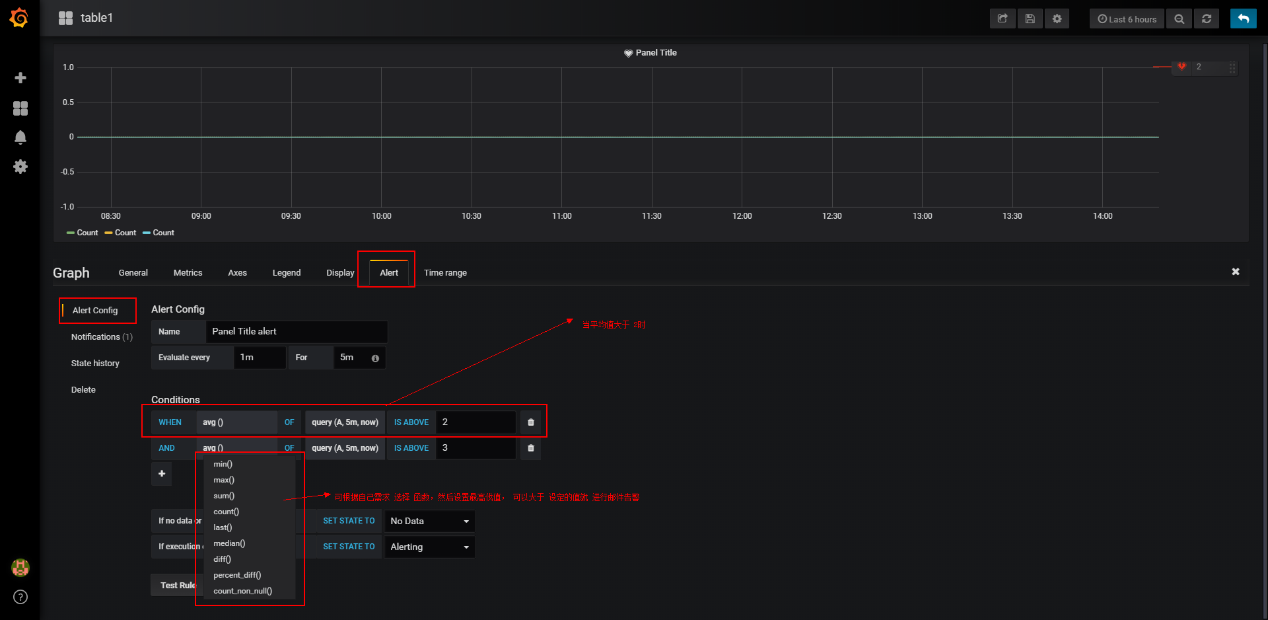
点击Sent Test

右上角发送成功提示，不成功请检查配置或网络

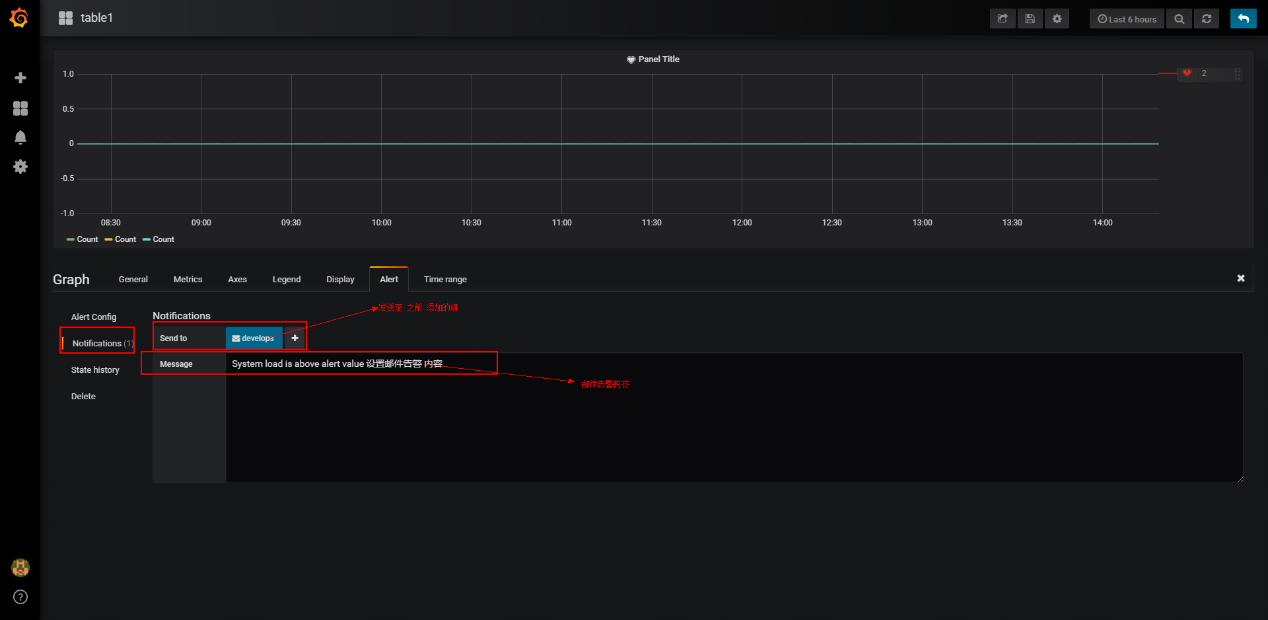


### 7.6.2 Alert Config 配置

面板设置Alert 框->Alert Config ，可根据自己业务监控需求设置函数 以及所对应的伐值



面板设置Alert 框->Notification 可设置邮件告警消息



## 7.7 网址参考

<https://www.cnblogs.com/wxwall/p/9642621.html>

<https://blog.csdn.net/xiaobai66073/article/details/81539086>

<https://blog.csdn.net/jailman/article/details/78920166> (设置Alert阈值和邮件报警)

# 8.部署问题

## 8.1 问题01

|  |
| --- |
| [elsearch@10.1.2.105 bin]$ ./elasticsearch -d  [elsearch@10.1.2.105 bin]$ 2018-06-01 07:49:06,951 main ERROR RollingFileManager (/data/server/es/logs/es-cluster.log) java.io.FileNotFoundException: /data/server/es/logs/es-cluster.log (Permission denied) java.io.FileNotFoundException: /data/server/es/logs/es-cluster.log (Permission denied)  at java.io.FileOutputStream.open0(Native Method)  at java.io.FileOutputStream.open(FileOutputStream.java:270)  at java.io.FileOutputStream.<init>(FileOutputStream.java:213)  at java.io.FileOutputStream.<init>(FileOutputStream.java:133) |

原因分析：

日志目录没有权限

解决方案：

对相关目录授予权限

|  |
| --- |
| chown -R elsearch data/ logs/ |

## 8.2 问题02

|  |
| --- |
| [2018-06-01T07:53:58,382][WARN ][o.e.b.JNANatives ] unable to install syscall filter:  java.lang.UnsupportedOperationException: seccomp unavailable: requires kernel 3.5+ with CONFIG\_SECCOMP and CONFIG\_SECCOMP\_FILTER compiled in  at org.elasticsearch.bootstrap.SystemCallFilter.linuxImpl(SystemCallFilter.java:328) ~[elasticsearch-6.2.4.jar:6.2.4]  at org.elasticsearch.bootstrap.SystemCallFilter.init(SystemCallFilter.java:616) ~[elasticsearch-6.2.4.jar:6.2.4] |

解决方案：

|  |
| --- |
| vi /etc/security/limits.conf  \* soft nofile 65536  \* hard nofile 131072  \* soft nproc 2048  \* hard nproc 4096 |

参见网址：

<https://blog.csdn.net/u014659211/article/details/51038299>

## 8.3 问题03

|  |
| --- |
| [4] bootstrap checks failed  [1]: memory locking requested for elasticsearch process but memory is not locked |

解决方案：

|  |
| --- |
| #切换到root用户，编辑limits.conf配置文件， 添加类似如下内容：  sudo vim /etc/security/limits.conf  #添加如下内容:  \* soft memlock unlimited  \* hard memlock unlimited  #备注：\* 代表Linux所有用户名称 |

## 8.4 问题04

|  |
| --- |
| max number of threads [1024] for user [elsearch] is too low, increase to at least [4096] |

原因分析：

无法创建本地线程问题,用户最大可创建线程数太小

解决方案：

切换到root用户，进入limits.d目录下，修改90-nproc.conf 配置文件。

|  |
| --- |
| vim /etc/security/limits.d/90-nproc.conf  #找到如下内容：  \* soft nproc 1024  #修改为  \* soft nproc 2048 |

## 8.5 问题05

|  |
| --- |
| [3]: max virtual memory areas vm.max\_map\_count [65530] is too low, increase to at least [262144] |

解决方案

|  |
| --- |
| sudo vim /etc/sysctl.conf  ##添加下面配置：  vm.max\_map\_count=655360  ##并执行命令：  sysctl -p |

## 8.6 问题06

|  |
| --- |
| system call filters failed to install; check the logs and fix your configuration or disable system call filters at your own risk |

原因分析：

因为Centos6不支持SecComp

SecComp是Linux kernel （自从2.6.23版本之后）所支持的一种简洁的sandboxing机制。它能使一个进程进入到一种“安全”运行模式，该模式下的进程只能调用4种系统调用（system calls），即read(), write(), exit()和sigreturn()，否则进程便会被终止。

而ES5.2以后的版本默认bootstrap.system\_call\_filter为true进行检测，所以导致检测失败，失败后直接导致ES不能启动。

参见网址：

<https://github.com/elastic/elasticsearch/issues/22899>

解决方案：

解决方法：在elasticsearch.yml中配置bootstrap.system\_call\_filter为false，注意要在Memory下面:

|  |
| --- |
| bootstrap.memory\_lock: false  bootstrap.system\_call\_filter: false |

## 8.7 问题07

|  |
| --- |
| [o.e.d.z.ZenDiscovery ] [node-1] not enough master nodes discovered during pinging (found [[Candidate{node={node-1}{JOPV4QgmTvWi8AVoi4548Q}{t9\_kKd11STCJYdNF2jby8g}{10.1.2.105}{10.1.2.105:9300}, clusterStateVersion=-1}]], but needed [2]), pinging again |

解决方案：

|  |
| --- |
| ##三个节点都须成功启动，并且防火墙都己关闭（保证可拼通的状态），  ## elasticsearch.yml 设置以下参数：  node.master: true  node.data: true  discovery.zen.minimum\_master\_nodes: 3 |

## 8.8 问题08

|  |
| --- |
| 执行 ./elasticsearch 时 - elasticsearch start more than one nodes on the same data folder  [2018-06-04T04:08:27,822][WARN ][o.e.b.ElasticsearchUncaughtExceptionHandler] [node-1] uncaught exception in thread [main] |

解决方案：

当我在开启多个elasticsearch 实例时，遇到上述异常，在 config/elasticsearch.yml文件中新增一个配置变量：

|  |
| --- |
| node.max\_local\_storage\_nodes: 256  #replace 256 with another num that greater than 1 |

## 8.9 问题09

|  |
| --- |
| kibana提示“Your Kibana index is out of date, reset it or use the X-Pack upgrade assistant.” |

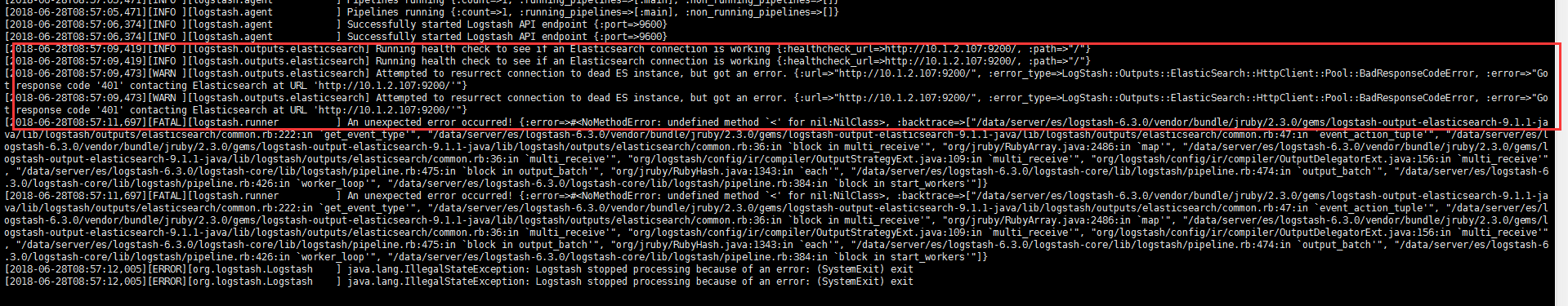
解决方案：

参照以下网址

<http://www.bubuko.com/infodetail-2429568.html>

## 8.10 问题10

|  |
| --- |
| [2018-06-28T08:57:09,473][WARN ][logstash.outputs.elasticsearch] Attempted to resurrect connection to dead ES instance, but got an error. {:url=>"http://10.1.2.107:9200/", :error\_type=>LogStash::Outputs::ElasticSearch::HttpClient::Pool::BadResponseCodeError, :error=>"Got response code '401' contacting Elasticsearch at URL 'http://10.1.2.107:9200/'"} |



解决方案：

链接 es时，配置用户名和密码

|  |
| --- |
| output{  # file{  # # 要读取的日志文件，可用\*通配符匹配 例如["/tmp/log4j/\*.log"]  # path=>["/data/server/es/logstash-6.3.0/test/monitor2.log"]  # message\_format => "%{message}"  # }  elasticsearch{  hosts => ["10.1.2.107:9200"]  **user => ["elastic"]**  **password => ["elastic"]**  index => "logstash-tomcat-log-%{+YYYY.MM.dd}"  }  # 输出到终端  stdout{  codec => rubydebug  }  } |

