**elasticsearch 环境搭建说明**

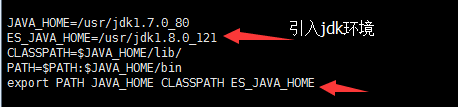
1. elasticsearch（简称es）至少需要在jdk1.8版本上运行，但是服务器上其他项目的运行环境要求在jdk1.7版本上，所以建立一个临时变量ES\_JAVA\_HOME专门存储jdk1.8路径

具体步骤:

1. 编辑etc下的profile文件

$vi /etc/profile

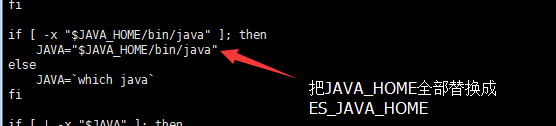
1. 如图所示，添加jdk1.8路径，注意只添加jdk路径，PATH和CLASSPATH不要管



1. 引入ES\_JAVA\_HOME到elasticsearch环境中，我把elasticsearch安装到/home文件夹下，确保环境运行空间充足，如果空间不足，运行elasticsearch的时候会报no space left on device 出现这个错误就换个空间充足的路径就行了（文件不能拷贝，换路径之后elasticsearch重新安装）。

vi /home/eshome/elasticsearch-5.4.3/bin/elasticsearch

找到此位置进行如下更改

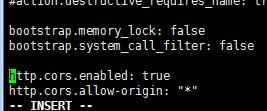


此时elasticsearch就是使用jdk1.8了。

1. 配置config下的elasticsearch.yml文件

添加如下配置：

前两个是内核安全支持，后两个是支持head插件能直接访问,在文件末尾直接添加即可



设置network.host 改配置设置成0.0.0.0为了让外网可以访问



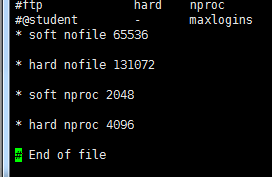
如果不设置的话，elasticsearch可以启动但是访问ip:port的时候会出现下面的提示



这就说明外网不能访问。前面的#一定要去掉，要不不生效。

1. 系统配置
2. [Linux](http://lib.csdn.net/base/linux)限制系统容量。包括进程持有的磁盘文件、网络连接；用户持有的进程数、等。配置在/etc/security/limits.conf中

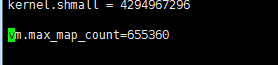
vi /etc/security/limits.conf 文件末尾添加



* 限制依据
  + hard-该限定值不可超过
  + soft-该限定值超过之后会警告
* 限制项目
  + nofile 单个用户持有最大文件数量（此处改此值即可）
  + nproc 可打开的文件描述符file descriptors数量

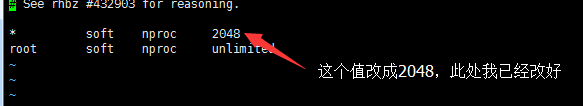
1. Linux限制虚拟内存

vi /etc/sysctl.conf 文件末尾添加



1. **max number of threads [1024] for user [elasticsearch] is too low, increase to at least [2048]** 如果出现这个错误去系统文件改

$vi /etc/security/limits.d/90-nproc.conf



文件全部改完执行下面命令

sysctl –p

如果报错 出现unknown key的提示

执行下面两条命令

modprobe bridge

lsmod |grep bridge

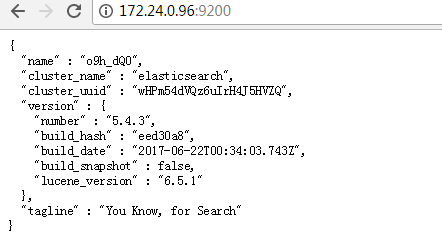
成功执行之后在执行 sysctl –p

1. 最重要的，特别注意 elasticsearch一定不要用root执行，要用非root组下面的用户去运行并把改文件夹的权限给新定义的用户。如果在过程用报出错误permission denied时候，查看elasticsearch-5.4.3文件夹内部还有属于root用户下的权限，如果还有就会报该错误，es访问不到root权限下的文件。 

这样我们切换es用户去执行他就可以了。运行bin/elasticsearch，成功运行后访问

<Http://ip:9200>即可

设置成功，访问界面如下图所示



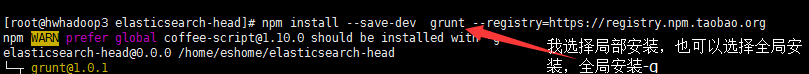
**Head插件安装**

head插件是集群管理插件，能够看到elasticsearch上的各种节点信息，但是elasticsearch5.X变化版本较大暂不支持直接安装

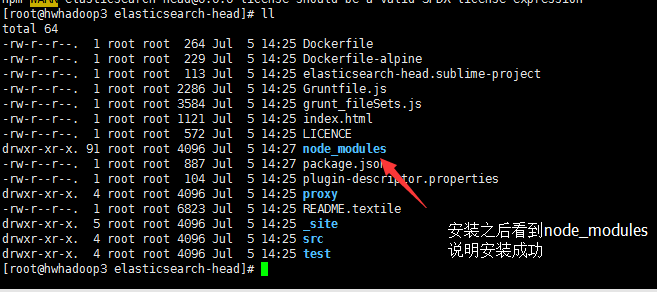
1. elasticsearch 插件

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

1. 安装grunt 主要是用来运行我们这个head插件用的



首先要进入到elasticsearch-head文件夹内出现图中所示，



说明安装成功。

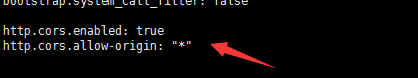
1. 修改head文件下的Gruntfile.js文件

vi Gruntfile.js

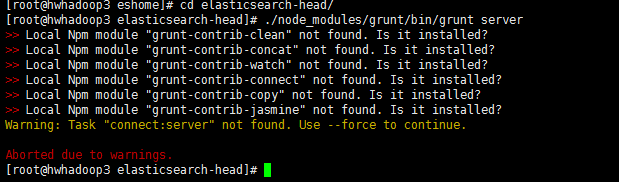


1. 设置跨域访问

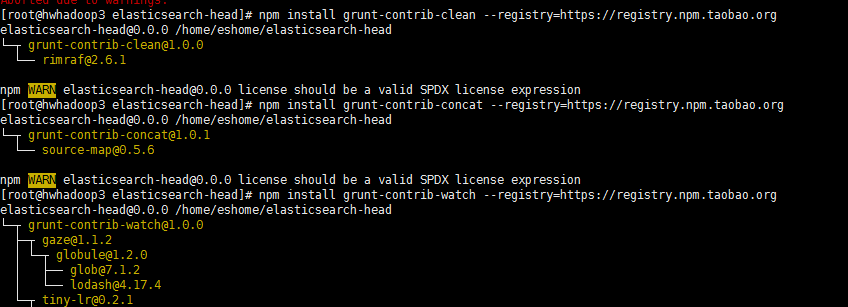
vi elasticsearch-5.4.3/config/elasticsearch.yml 在末尾添加如下两行，如果在上一环节已经配置，那么就不用重复添加。



1. 启动服务



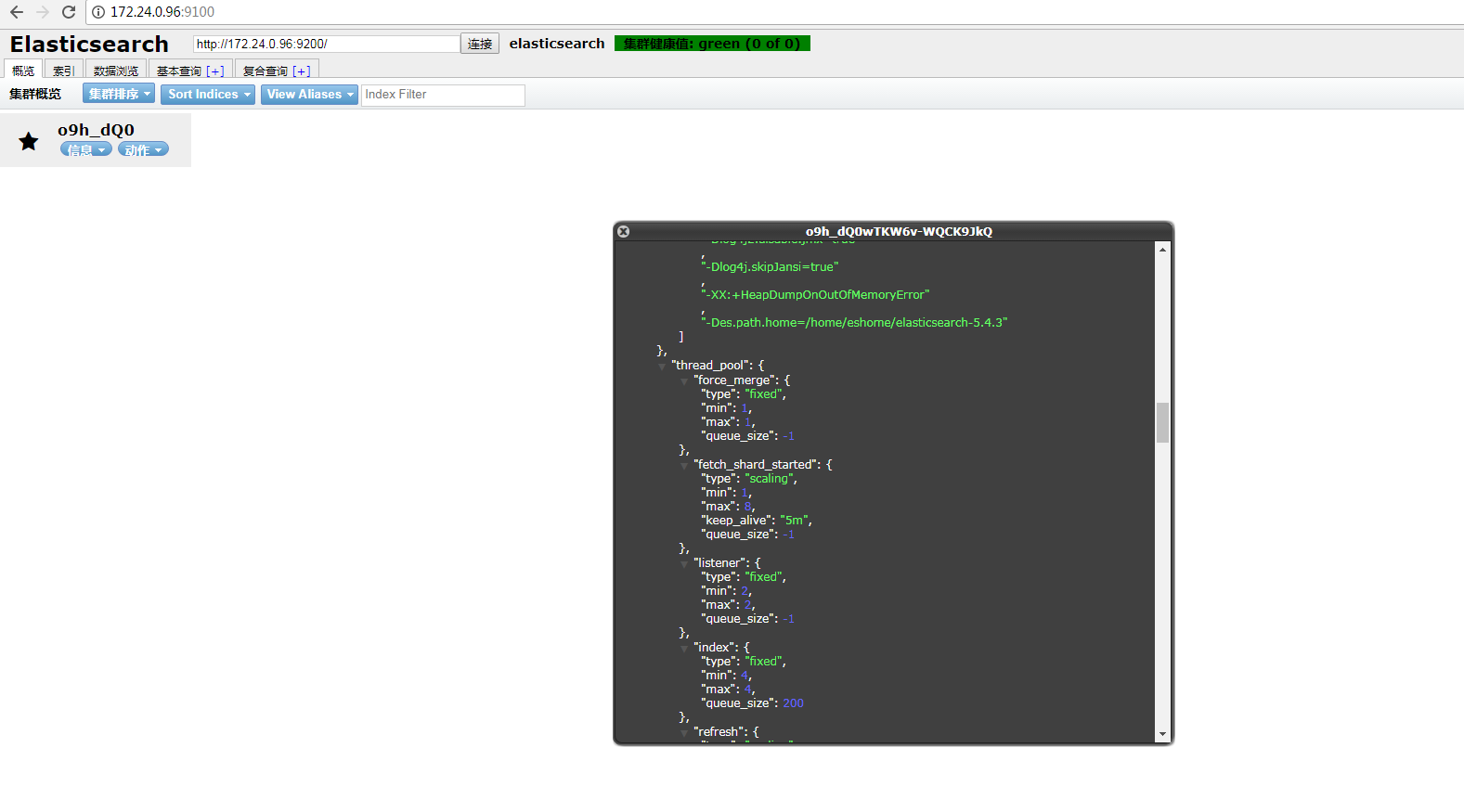
启动服务可能会出现如下警告提示，那么我们继续安装即可



依次把提示未安装的的组件安装上即可，一共6个。

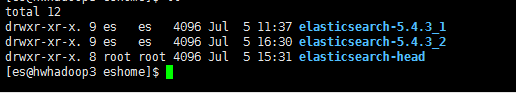
安装之后浏览器访问 <http://ip:9100>即可

成功访问界面如图所示

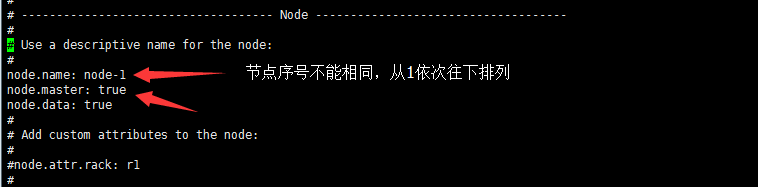


**elasticsearch添加子节点**

1. 在/home/eshome下建立两个文件夹，如下图所示，把之前的文件copy到两个文件夹下面



2. 在elasticsearch-5.4.3\_\*文件下config/elasticsearch.yml文件下修改

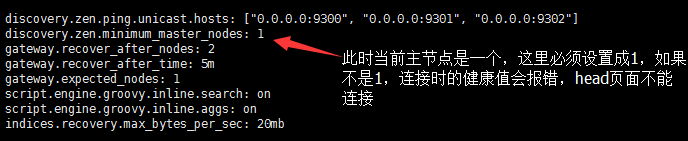


如下方红箭头所示 主节点，master 设置为true 子节点的master为false 其余的设置相同。

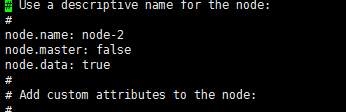
端口号设置如图所示

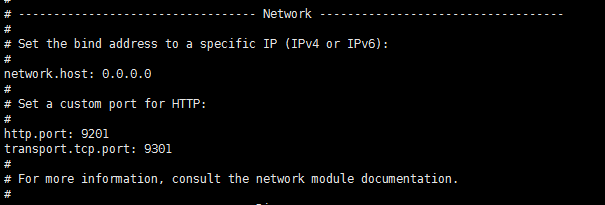


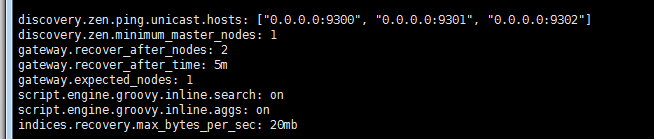
节点设置 如图所示，添加到空白处即可，



下面暂时的是子节点的设置图，前面已经一一介绍了，就直接放图





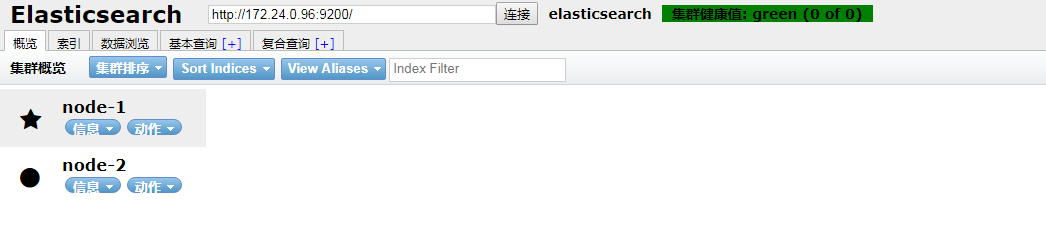


3. 受理权限，把elasticsearch-5.4.3\_\*的文件权限设置成es

chown -R es:es elasticsearch-5.4.3\_\*，并执行所有的./elasticsearch

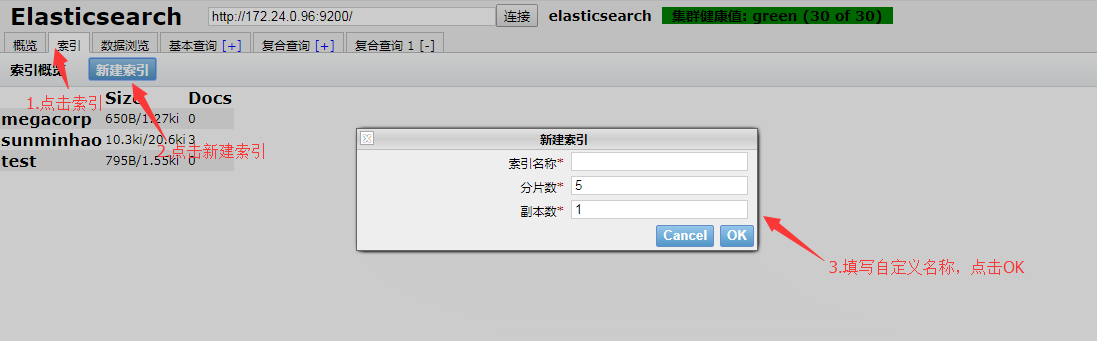
可能会报错，子节点不能加入集群，如果这样的问题出现，就把子节点文件夹下的data文件夹里的内容全部删去，但是不能把data删除。

4.启动各个节点，启动head之后，如果成功启动的话出现如下界面

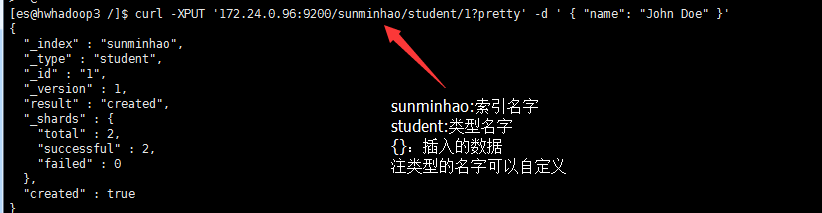


**数据操作**

1.简历索引



2.插入数据



当下方显示ceated：true时显示导入成功。

curl -XGET 'localhost:9200/sunminhao/external/1?pretty'get获取

curl -XDELETE 'localhost:9200/sunminhao?pretty' 删除索引

curl -XDELETE 'localhost:9200/sunminhao/student/\_query?pretty'-d ' { "query": { "match": { "name”:”sunmihnao”} 删除数据

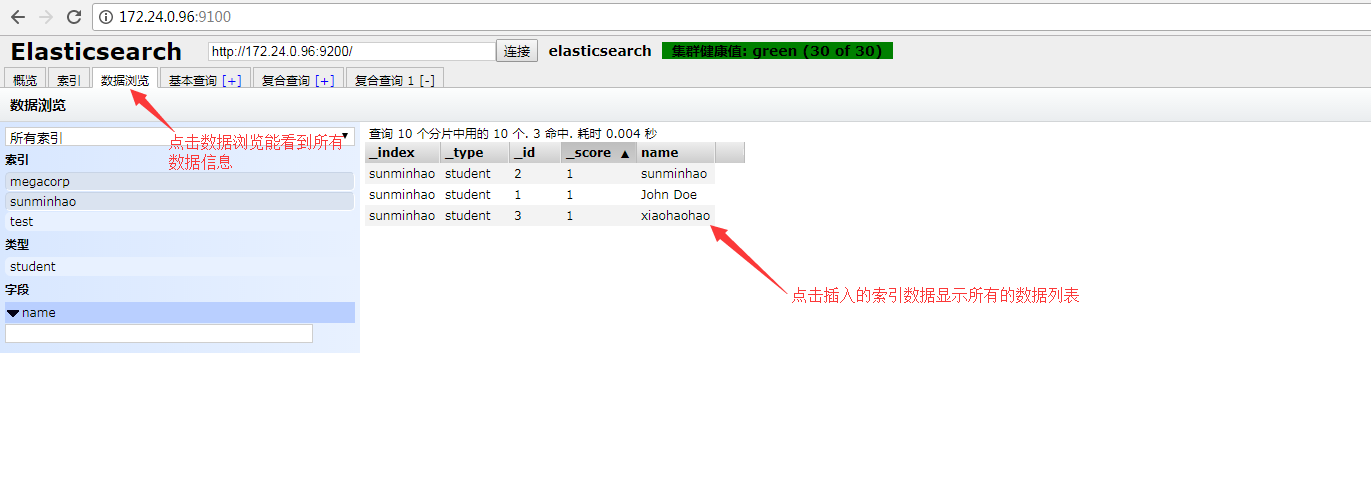
curl -XPOST 'localhost:9200/sunminhao/student/\_bulk?pretty'-d ' {"index":{"\_id":"5"}} {"name": "John Doe" } {"index":{"\_id":"6"}} {"name": "Jane Doe" } '

curl -XPOST 'localhost:9200/sunminhao/student/\_bulk?pretty'-d ' {"update":{"\_id":"1"}} {"doc": { "name": "John Doe becomes Jane Doe" } } {"delete":{"\_id":"2"}} '

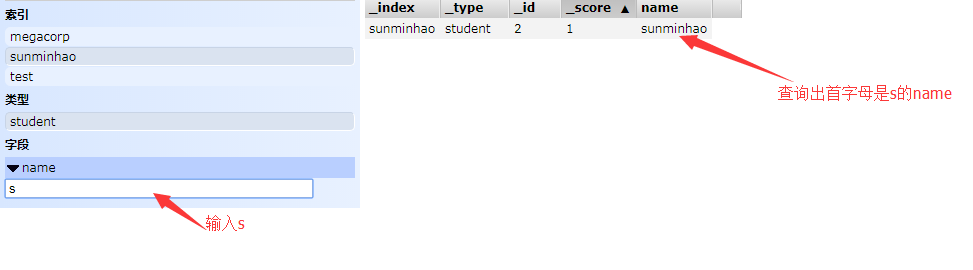
curl -XPOST 'localhost:9200/sunminhao/student/\_bulk?pretty' --data-binary @accounts.json 批量操作

查询操作在下方演示

head界面显示如下：

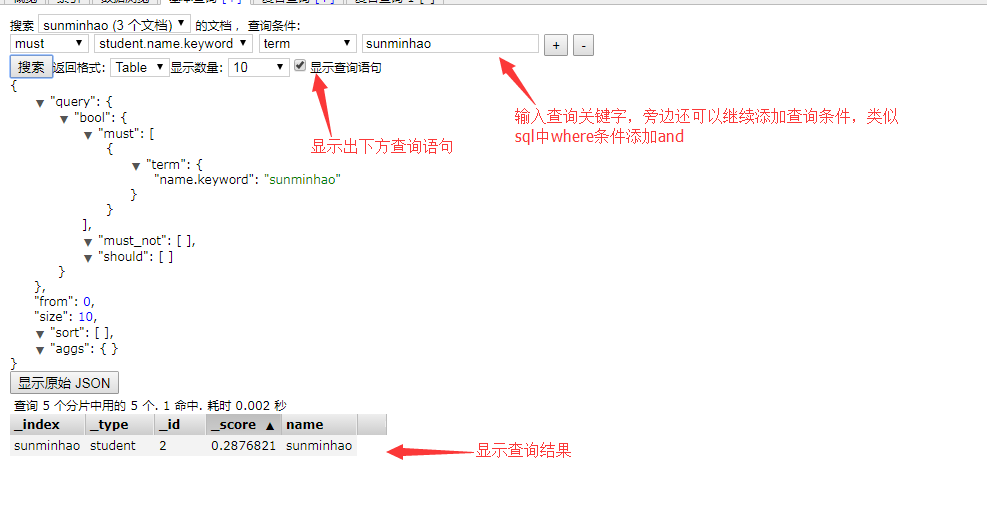


左边栏依次现实的是索引名称，点击查询的索引，下方显示的是索引的类型和字段，在name中输入查询的关键字可以模糊查询，如下图所示

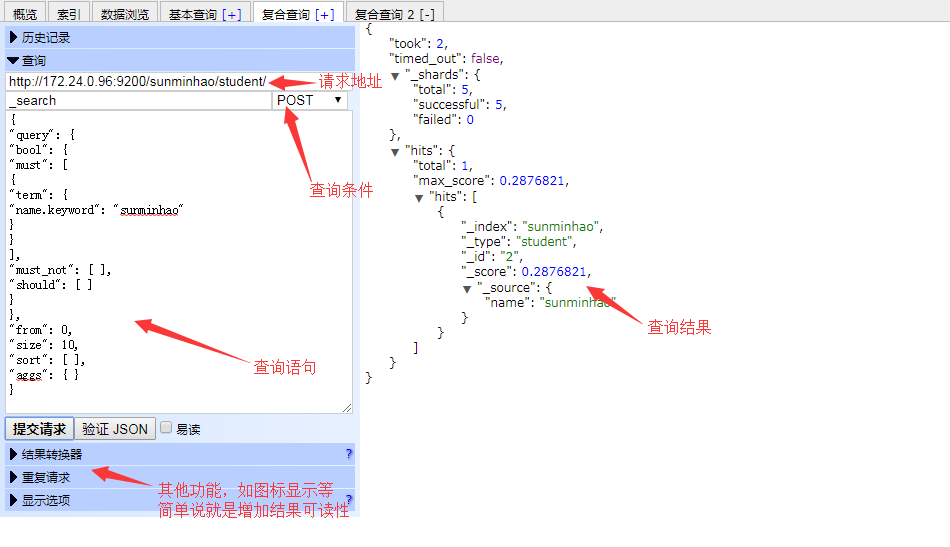


3.查询数据

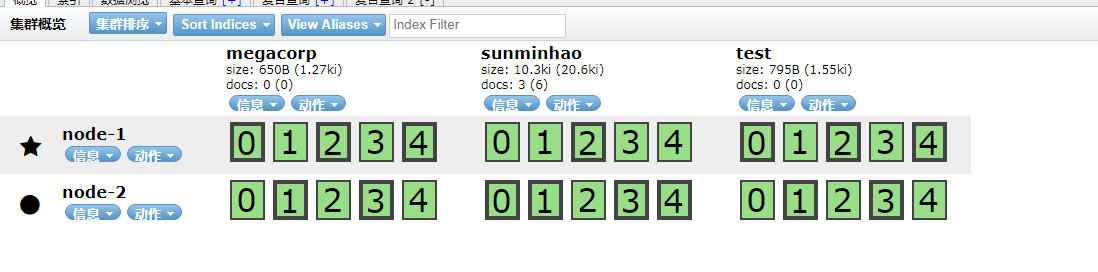
点击基本查询出现如下界面，根据需求自己定义查询语句，本例子举出按照关键字查询（不是模糊查询）



点击复合查询如下图所示



4.其他信息

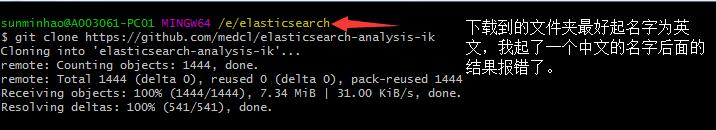


点击信息可以查看索引的状态的节点的信息，动作可以删除节点，还可以对节点排序等等，根据需求可以自行查看。

右上角的信息和刷新可以查看集群的信息，设定每隔多少秒刷新一次。

# 中文分词插件安装

1.下载中文分词插件ik



下载插件之后进入到插件文件夹中

$ cd elasticsearch-analysis-ik/

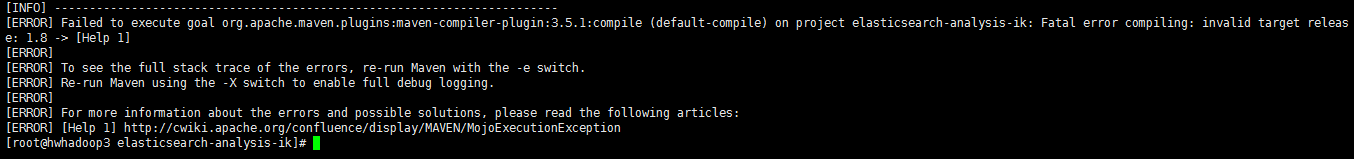
2.编译打包



在第二步的时候遇到了很多坑，调试了很久，最后占春哥给了一个提示，问题解决了。

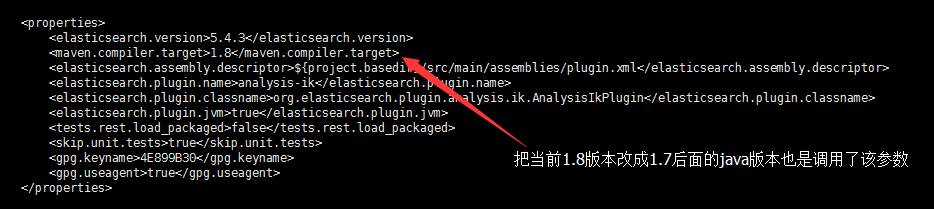
最新的版本ik在jdk1.8版本上运行，但是当前96系统上的JAVA\_HOME是jdk1.7版本，版本不想同导致编译错误。

具体错误如下图所示



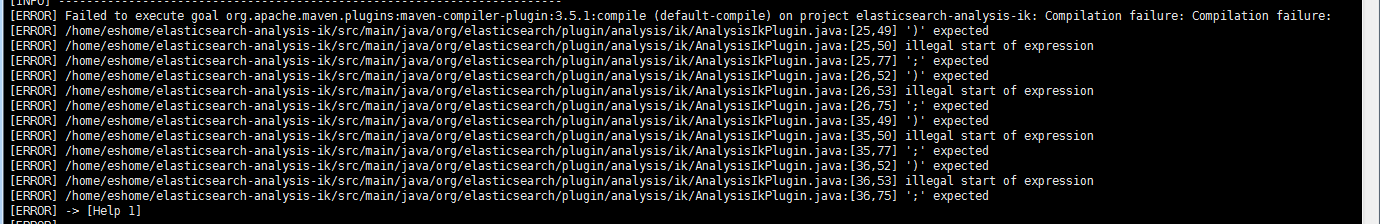
此错误出现就是jdk当前版本和ik支持jdk版本不相同

该配置文件

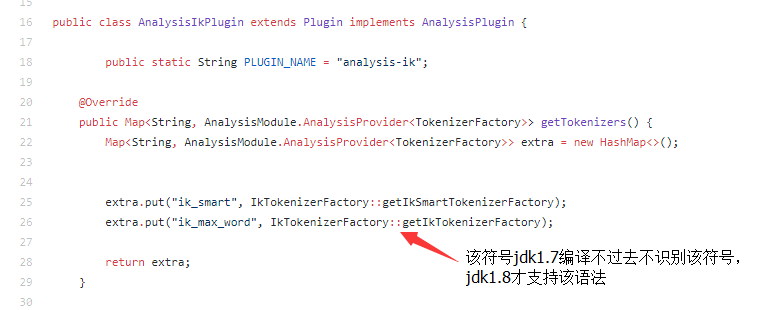


改完再次编译

$mvn compile



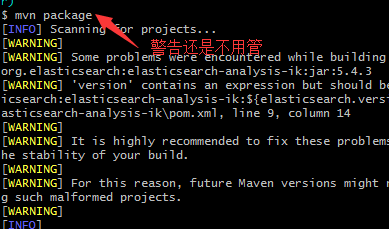
出现了无法该段java代码的错误，找到该段代码查看



下一步想到去找ik支持jdk的路径，没找到，这段懂得配置的可以一起讨论一下

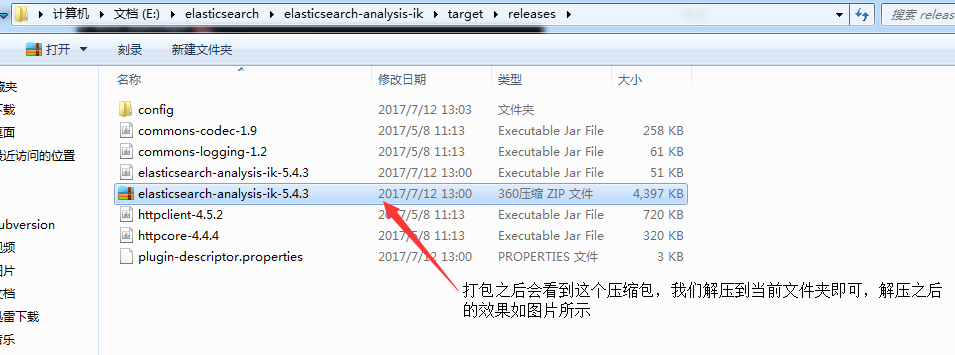
换一种一种想法，本地的jdk版本是1.8，我再本地clean 和 compile 编译之后就不会报错，像第二步图片所示就是我再本地上的运行结果。

3.$ mvn package

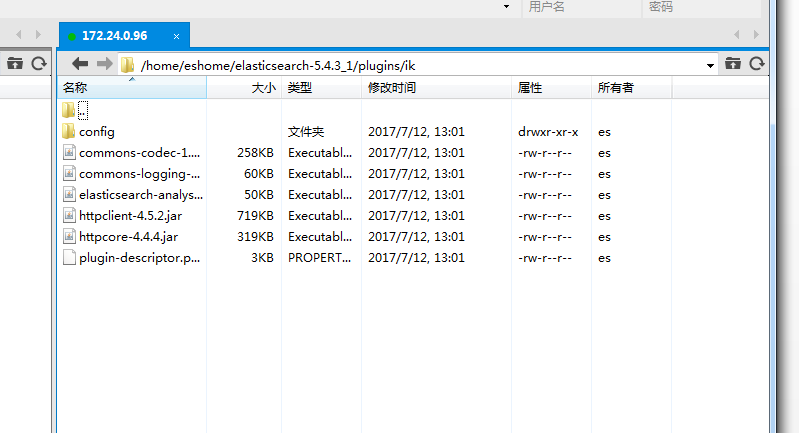


最后显示BUILD SUCCESS说明打包成功

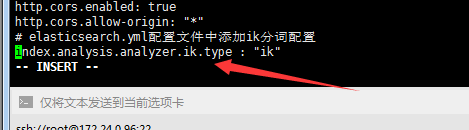
进入到target/releases文件夹下，如下图所示：



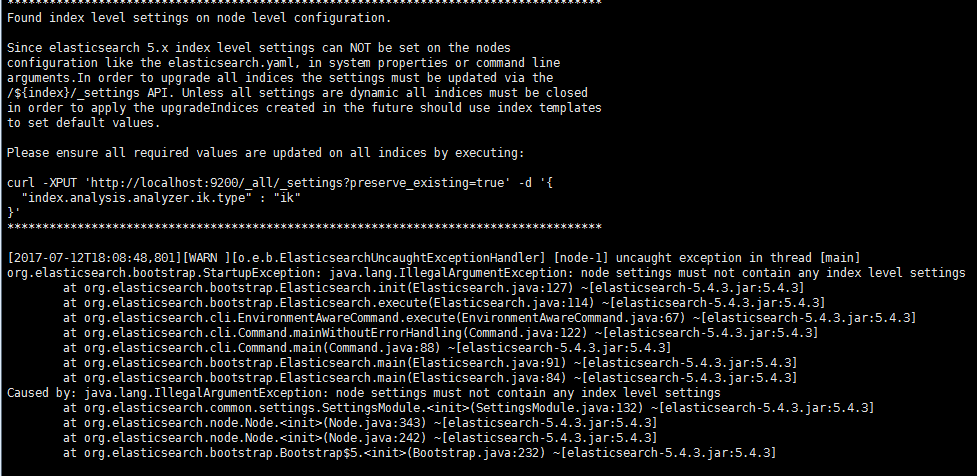
把解压后的文件拷贝到es文件夹中plugins/ik 如果没有ik路径新建一个即可



\*\*\*\*\*\*我试了一下在config/elasticsearch.yml文件下添加了ik的配置，结果报错



重启elasticsearch之后，看到错误如下



6. 5.x版本以上不用配置ik信息。

7.添加分词设置（分词器可选ik\_smart或ik\_max\_word），可以自定义需求。



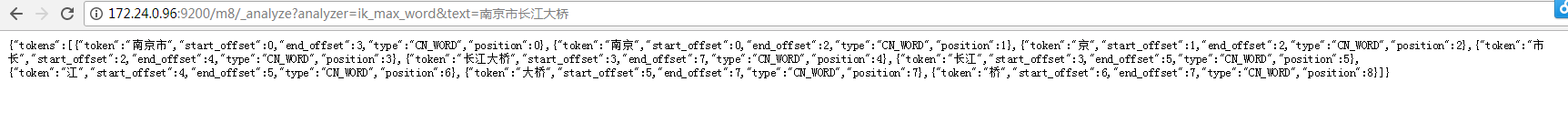
其中：

ik\_max\_word: 会将文本做最细粒度的拆分，比如会将“中华人民共和国国歌”拆分为“中华人民共和国,中华人民,中华,华人,人民共和国,人民,人,民,共和国,共和,和,国国,国歌”，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“中华人民共和国国歌”拆分为“中华人民共和国,国歌”。

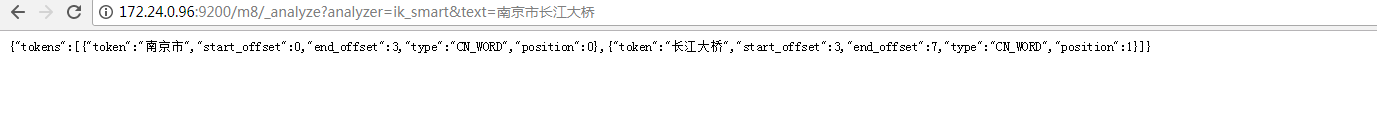
8.访问分词结果

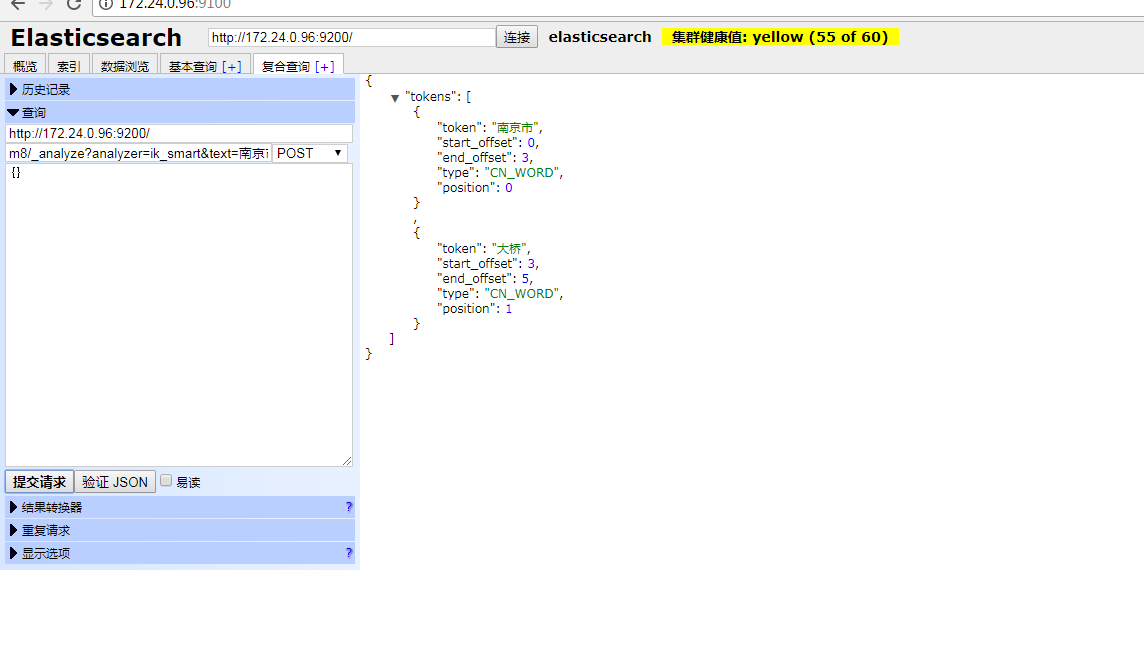
<http://172.24.0.96:9200/m8/_analyze?analyzer=ik_max_word&text=南京市长江大桥>



出现了所有分词结果

ik\_smart结果如图所示



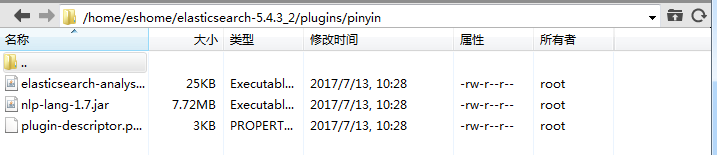


在符合查询也可以操作，如上图所示。但需要注意的是第二行的输入框不要写ip：port了，再写就重复了，访问不到了。

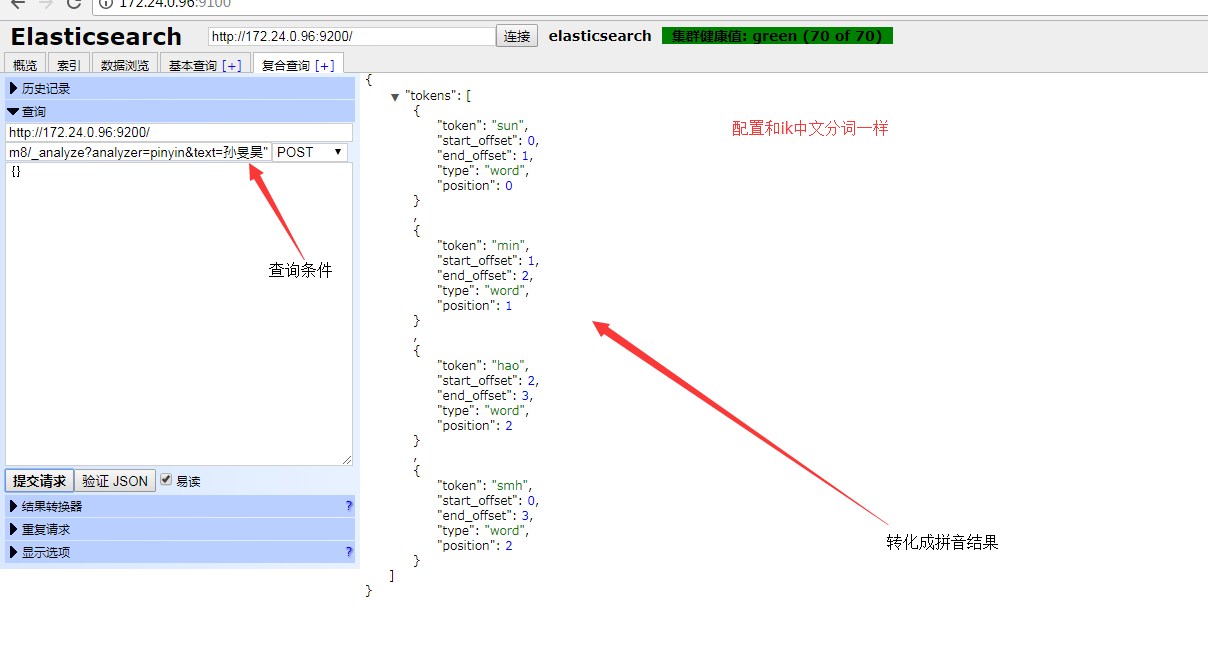
# 拼音分词

拼音分词和中文分词一模一样，不再叙述，重复上面的操作就能实现，这里这是介绍一下需要的资源路径

pinyin分词器的下载地址:   
<https://github.com/medcl/elasticsearch-analysis-pinyin> 编译之后复制到该处，如果没有pinyin路径同样的自己创建一个。



测试结果如下图所示：

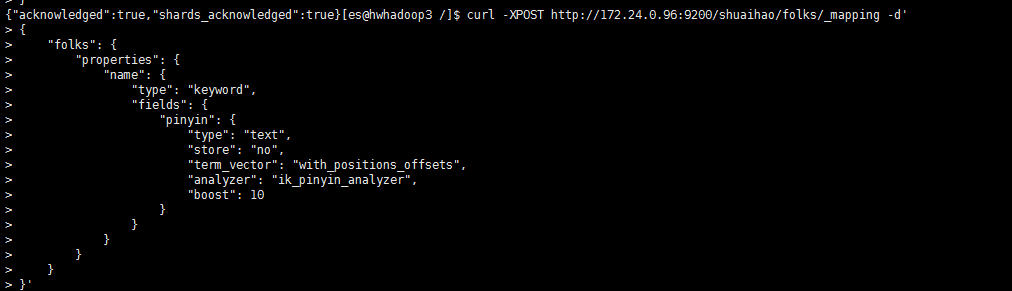


# ik+pinyin测试结果

1.创建一个索引，并设置相关属性



2.添加mapping



详情如下：

curl -XPOST http://172.24.0.96:9200/shuaihao/folks/\_mapping -d'

{

"folks": {

"properties": {

"name": {

"type": "keyword",

"fields": {

"pinyin": {

"type": "text",

"store": "no",

"term\_vector": "with\_positions\_offsets",

"analyzer": "ik\_pinyin\_analyzer",

"boost": 10

}

}

}

}

}

}'

3.插入数据

curl -XPOST http://172.24.0.96:9200/shuaihao/folks/andy -d'{"name":"刘德华"}'

curl -XPOST http://172.24.0.96:9200/shuaihao/folks/tina -d'{"name":"中华人民共和国国歌"}'

4.测试1：pinyin分词

curl-XPOST<http://localhost:9200/medcl/folks/_search?q=name.pinyin:hua>

界面测试



5.测试2：ik分词



curl -XPOST "http:// 172.24.0.96:9200/shuaihao/\_search?pretty" -d'

{

"query": {

"match": {

"name.pinyin": "国歌"

}

},

"highlight": {

"fields": {

"name.pinyin": {}

}

}

}'

6.测试3：pinyin+ik分词

curl -XPOST "http://172.24.0.96:9200/shuaihao/\_search?pretty" -d'

{

"query": {

"match": {

"name.pinyin": "zhonghua"

}

},

"highlight": {

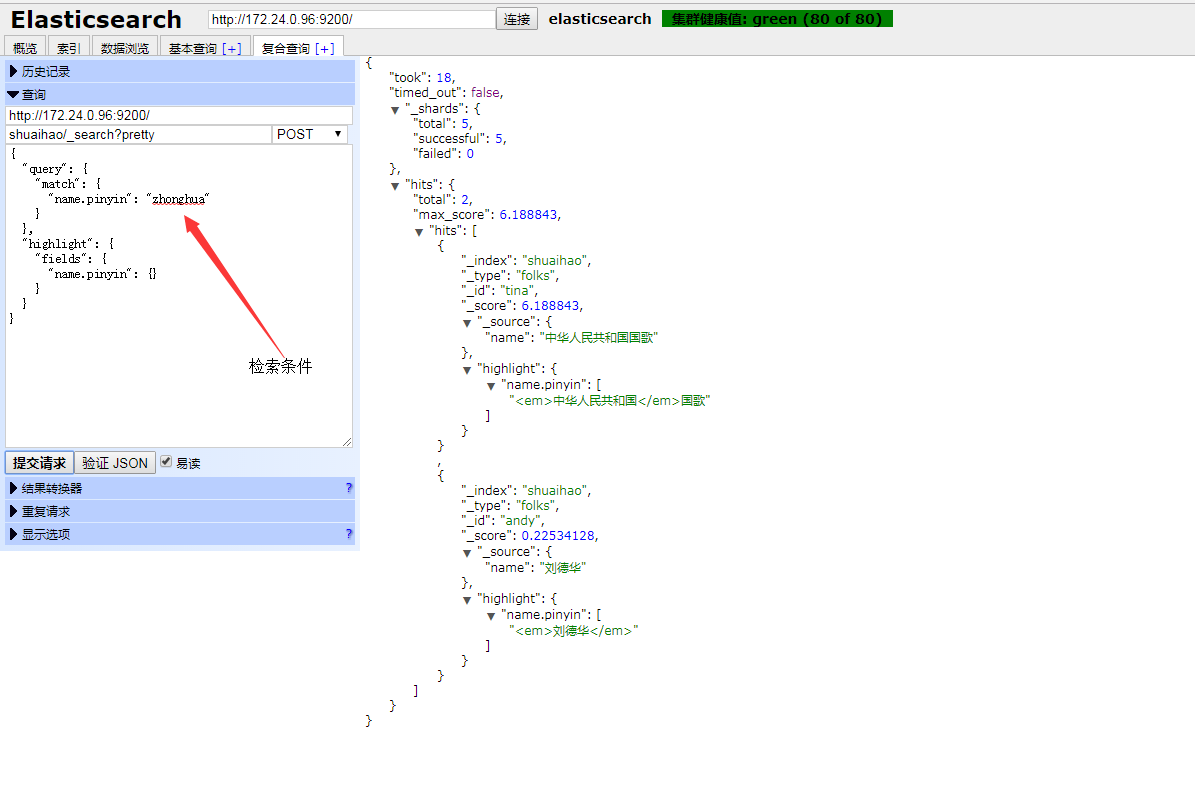
"fields": {

"name.pinyin": {}

}

}

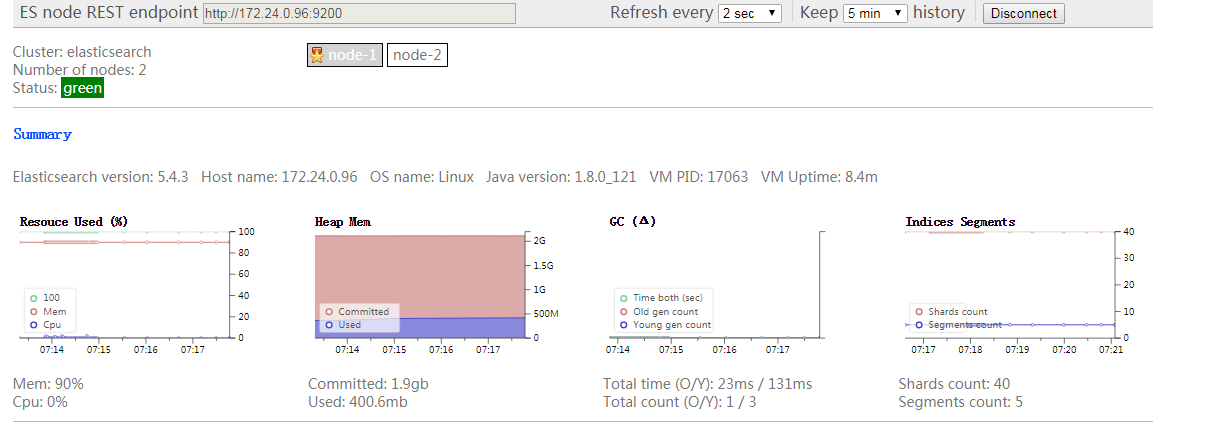
}'

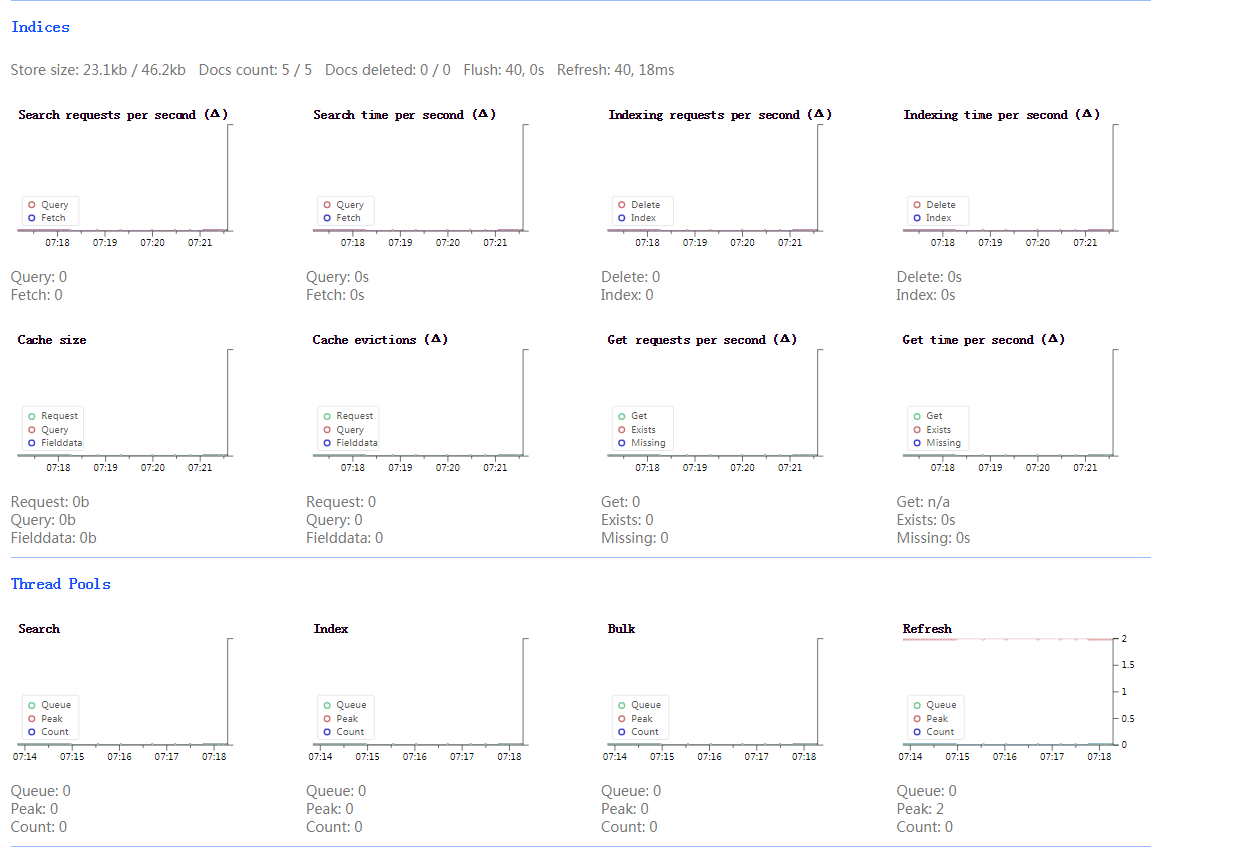


其余检索条件可以根据自己的需求去做相应的查询

# bigdesk 系统监测安装

1. git clone https://github.com/hlstudio/bigdesk
2. $ cd bigdesk/\_site/
3. $ python -m SimpleHTTPServer
4. 访问ip:8000访问界面如下





具体的细节可以自己登陆查看，还有jvm 线程池等信息。