ElasticSearch、Logstash和Kiabana均有开箱即用的版本，　也可以使用docker,就不用下载具体的包了

官网下载:https://www.elastic.co/cn/downloads/

1.ElasticSearch:

在config中 增加elasticsearch.yml文件如下内容:

```
network.host: 0.0.0.0   # 网络设置,表示大家都能连
```

执行 bin/elasticsearch 即可

在浏览器中输入 http://localhost:9200/

返回如下json表示成功:

```
{
  "name": "node-1",
  "cluster_name": "elasticsearch",
  "cluster_uuid": "qgDoFT0_Sa66sYTd_5ETug",
  "version": {
    "number": "5.5.3",
    "build_hash": "9305a5e",
    "build_date": "2017-09-07T15:56:59.599Z",
    "build_snapshot": false,
    "lucene_version": "6.6.0"
  },
  "tagline": "You Know, for Search"
}
```

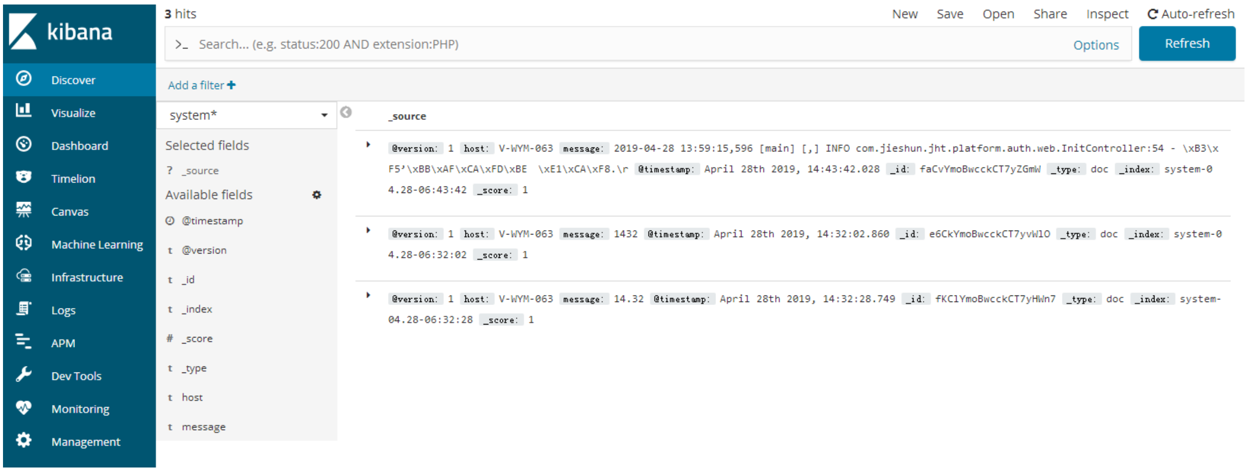
2.Kiabana:

在config中修改 kibana.yml ,如下内容:

```
elasticsearch.url: "http://localhost:9200"
```

执行 bin/kibana.bat

在浏览器中访问 http://localhost:5601 会出现页面,说明成功了



```
@version: 1 host: V-WYM-063 message: 2019-04-28 13:59:15,596 [main] [,] INFO com.jieshun.jht.platform.auth.web.InitController:54 - \xB3\xF5'\xBB\xAF\xCA\xFD\xBE  \xE1\xCA\xF8.\r @timestamp: April 28th 2019, 14:43:42.028 _id: faCvYmoBwcckCT7yZGmW _type: doc _index: system-04.28-06:43:42 _score: 1

@version: 1 host: V-WYM-063 message: 1432 @timestamp: April 28th 2019, 14:32:02.860 _id: e6CkYmoBwcckCT7yvW1O _type: doc _index: system-04.28-06:32:02 _score: 1

@version: 1 host: V-WYM-063 message: 14.32 @timestamp: April 28th 2019, 14:32:28.749 _id: fKC1YmoBwcckCT7yHWn7 _type: doc _index: system-04.28-06:32:28 _score: 1
```

3.Logstash

(有点像Flume,有接收数据,处理数据,输出数据.输入/输出都有各种选择)

直接运行 start.bat 即可

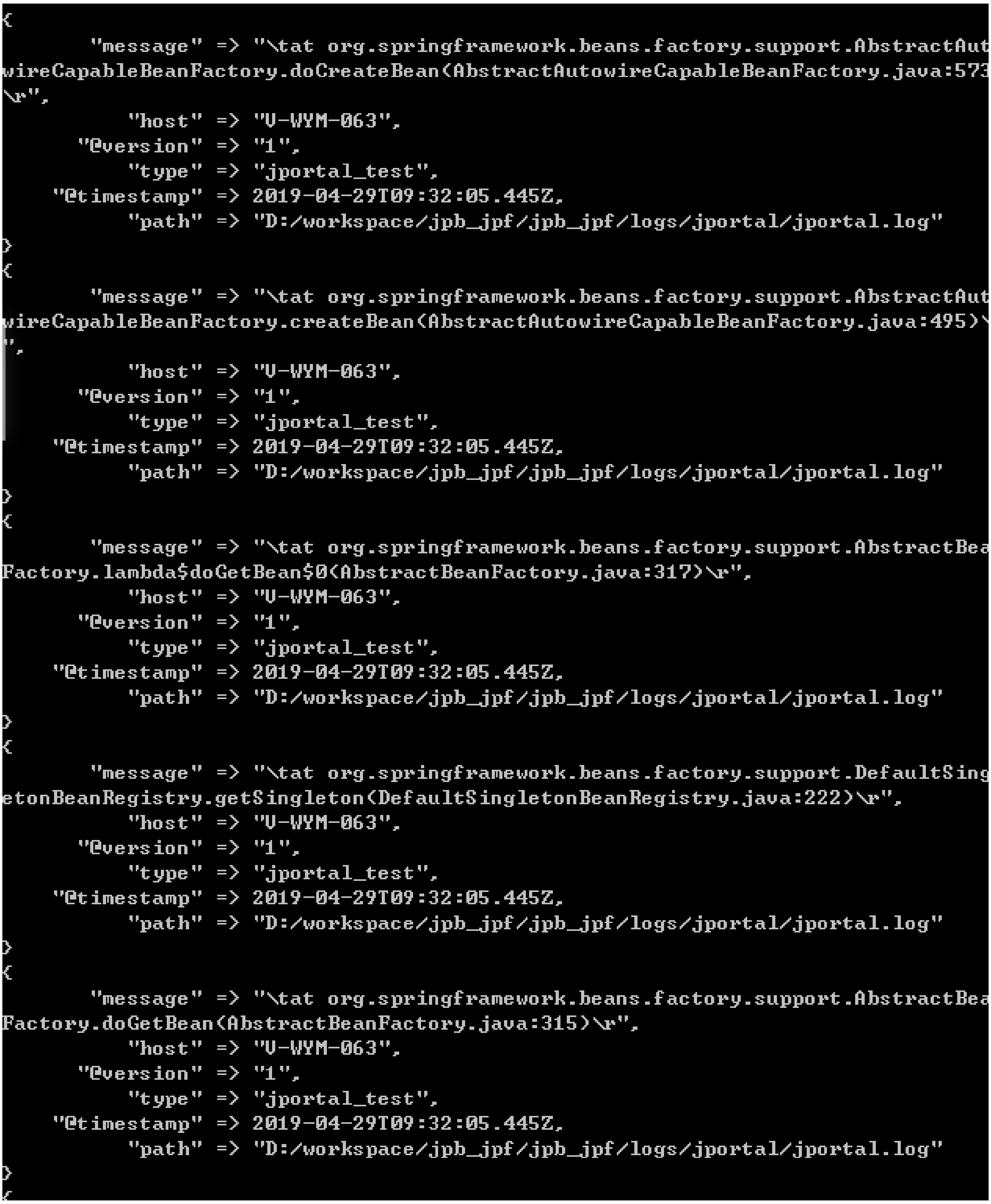
在config文件夹下,创建logstash.conf文件,写入以下内容:

```
input {// logstash的数据来源
   # 从控制台接收, 类型是test(可以不写),
  stdin{  type => "test" }
   #从文件中接收,
  file{
    path=>"/home/jht/jportal-license/jportal/logs/jportal/jportal.log" # 从这个文件中接收,允许写多个文件,写在[]中,用","隔开,数组的格式
    type => "jportal" # 类型是jportal
    start_position => "beginning" # 表示从文件的起始位置读
  }
}
# 过滤条件,可以处理数据的输出格式
#filter {
#     grok {
#     ##patterns_dir 是刚刚创建的patterns文件夹目录，根据创建具体路径配置
#       patterns_dir => "D:/software/ELK/logstash-6.5.0/patterns"
#       match => {
#               "message" => "%{JPORTAL}"
#       }
#  }
#
#}

#输出
output {
   #输出到控制台,codec(可以不写)表示类型,
  stdout {  codec => rubydebug  }
  # type是输入源里定义的
  if [type] == "system" {
      #输入到elasticsearch
     elasticsearch {
        hosts => ["localhost:9200"] # es的地址
        index => "system-%{+MM.dd-HH:mm:ss}" # 索引,kiabana会安装index分类,这里可以按照项目或者日志级别等分类
     }
   }

  if [type] == "jportal" or [type] == "test" {
   elasticsearch {
      hosts => ["localhost:9200"]
      index => "jportal-%{+MM.dd-HH:mm:ss}"
   }
 }
}
```

4.使用

如果三者改了配置都需要重新启动, 不会动态获取配置

启动Logstash, 如果监控的文件中有数据, 会输出:



```
{
       "message" => "\tat org.springframework.beans.factory.support.AbstractAut
wireCapableBeanFactory.doCreateBean(AbstractAutowireCapableBeanFactory.java:573
\r",
          "host" => "V-WYM-063",
      "@version" => "1",
          "type" => "jportal_test",
    "@timestamp" => 2019-04-29T09:32:05.445Z,
          "path" => "D:/workspace/jpb_jpf/jpb_jpf/logs/jportal/jportal.log"
}
{
       "message" => "\tat org.springframework.beans.factory.support.AbstractAut
wireCapableBeanFactory.createBean(AbstractAutowireCapableBeanFactory.java:495)\
",
          "host" => "V-WYM-063",
      "@version" => "1",
          "type" => "jportal_test",
    "@timestamp" => 2019-04-29T09:32:05.445Z,
          "path" => "D:/workspace/jpb_jpf/jpb_jpf/logs/jportal/jportal.log"
}
{
       "message" => "\tat org.springframework.beans.factory.support.AbstractBea
Factory.lambda$doGetBean$0(AbstractBeanFactory.java:317)\r",
          "host" => "V-WYM-063",
      "@version" => "1",
          "type" => "jportal_test",
    "@timestamp" => 2019-04-29T09:32:05.445Z,
          "path" => "D:/workspace/jpb_jpf/jpb_jpf/logs/jportal/jportal.log"
}
{
       "message" => "\tat org.springframework.beans.factory.support.DefaultSing
etonBeanRegistry.getSingleton(DefaultSingletonBeanRegistry.java:222)\r",
          "host" => "V-WYM-063",
      "@version" => "1",
          "type" => "jportal_test",
    "@timestamp" => 2019-04-29T09:32:05.445Z,
          "path" => "D:/workspace/jpb_jpf/jpb_jpf/logs/jportal/jportal.log"
}
{
       "message" => "\tat org.springframework.beans.factory.support.AbstractBea
Factory.doGetBean(AbstractBeanFactory.java:315)\r",
          "host" => "V-WYM-063",
      "@version" => "1",
          "type" => "jportal_test",
    "@timestamp" => 2019-04-29T09:32:05.445Z,
          "path" => "D:/workspace/jpb_jpf/jpb_jpf/logs/jportal/jportal.log"
}
{
```

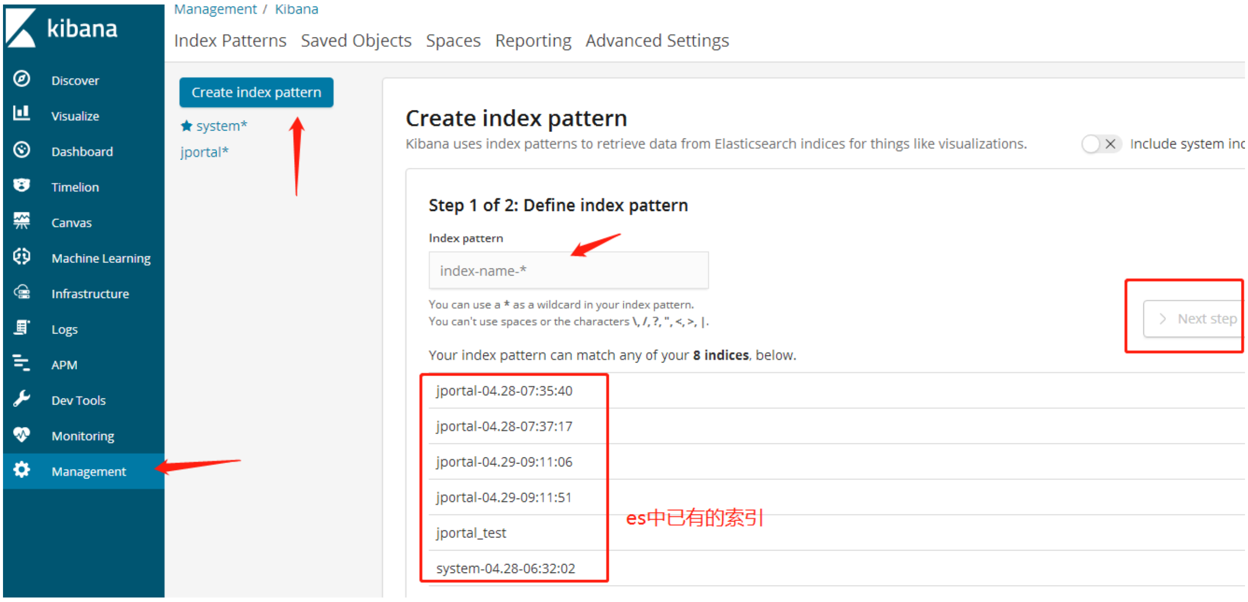
启动Logstash后,控制会等待输入， 输入 hello:

控制台会输出:

```
{
          "type" => "test",
      "@version" => "1",
    "@timestamp" => 2019-04-29T09:11:51.533Z,
       "message" => "hello\r",
          "host" => "V-WYM-063"
}
```

在kiabana中能查询的es中的数据,默认按索引分类.

先新建个索引,



Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

Step 1 of 2: Define index pattern

Index pattern

index-name-*

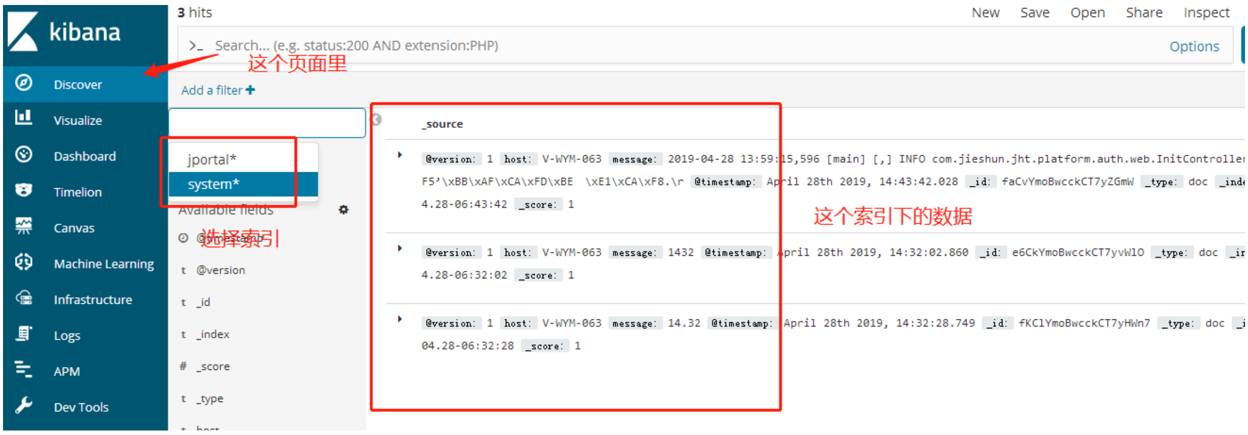
You can use a * as a wildcard in your index pattern.
You can't use spaces or the characters \, /, ?, ", <, >, |.

Your index pattern can match any of your **8 indices**, below.

jportal-04.28-07:35:40
jportal-04.28-07:37:17
jportal-04.29-09:11:06
jportal-04.29-09:11:51
jportal_test
system-04.28-06:32:02

es中已有的索引

查询数据:



这个页面里

选择索引

_source

```
@version: 1 host: V-WYM-063 message: 2019-04-28 13:59:15,596 [main] [,] INFO com.jieshun.jht.platform.auth.web.InitControlle
F5’\xBB\xAF\xCA\xFD\xBE  \xE1\xCA\xF8.\r @timestamp: April 28th 2019, 14:43:42.028 _id: faCvYmoBwcckCT7yZGmW _type: doc _inde
4.28-06:43:42 _score: 1

@version: 1 host: V-WYM-063 message: 1432 @timestamp: April 28th 2019, 14:32:02.860 _id: e6CkYmoBwcckCT7yvW1O _type: doc _ir
4.28-06:32:02 _score: 1

@version: 1 host: V-WYM-063 message: 14.32 @timestamp: April 28th 2019, 14:32:28.749 _id: fKC1YmoBwcckCT7yHWn7 _type: doc _i
04.28-06:32:28 _score: 1
```

这个索引下的数据

安装head插件,可以管理ES中的数据

a)插件安装方法一

/usr/share/elasticsearch/bin/plugin install mobz/elasticsearch-head（新版本的plugin命令可能被替代了,自己找找类似的)

b）插件安装方法二

首先下载head插件，下载到/usr/loca/src目录下

下载地址：https://github.com/mobz/elasticsearch-head

==

head插件包百度云盘下载：https://pan.baidu.com/s/1boBE0qj

提取密码：ifj7

==

unzip elasticsearch-head-master.zip

在/usr/share/elasticsearch/plugins目录下创建head目录

然后将上面下载的elasticsearch-head-master.zip解压后的文件都移到/usr/share/elasticsearch/plugins/head下

修改es 的配置文件: elasticsearch.yml配置文件,添加以下内容,

```
http.cors.enabled: true
http.cors.allow-origin: "*"
http.cors.allow-credentials: true
```

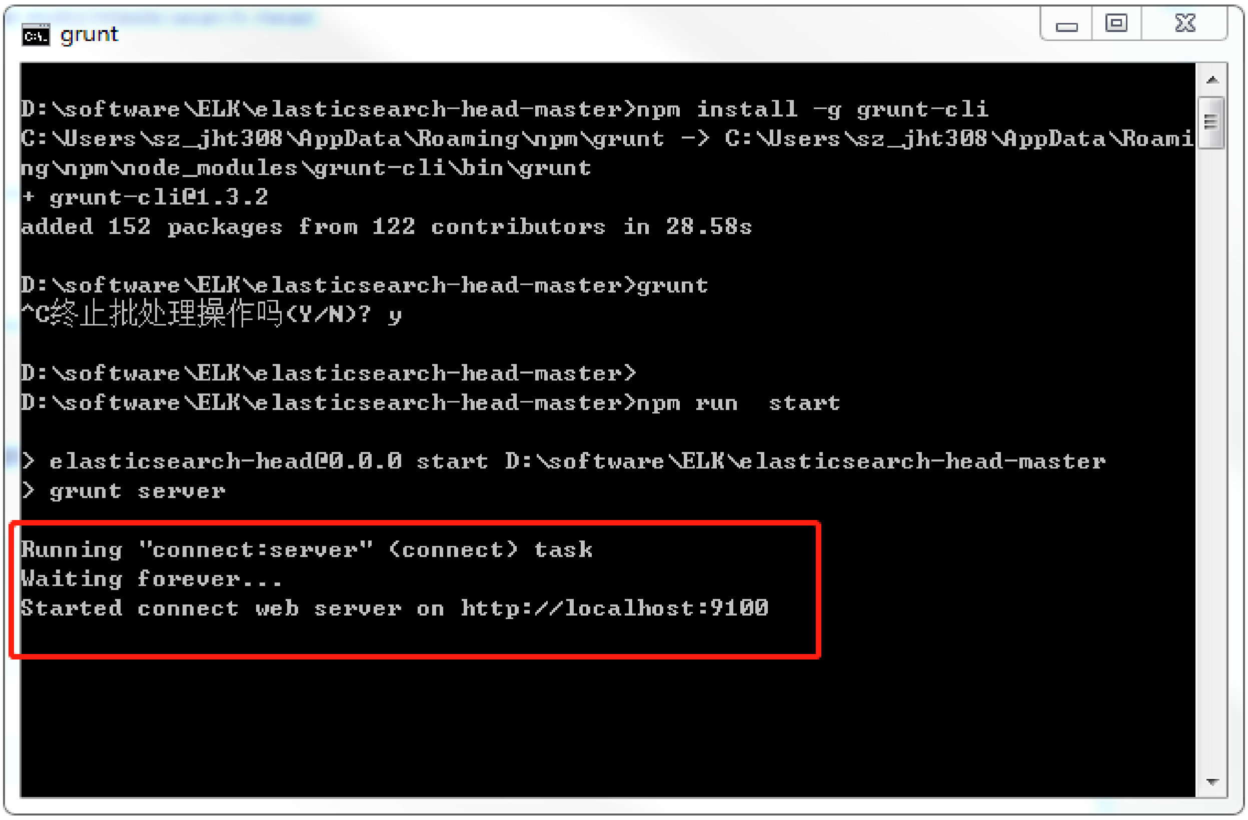
启动elasticsearch-head,在elasticsearch-head文件下

```
npm run  start
```

npm基于node.js,如果没有则需先装node.js,

> 其中还需要grunt命令,如果没有则需安装: npm install -g grunt-cli

出现以下内容表示成功:

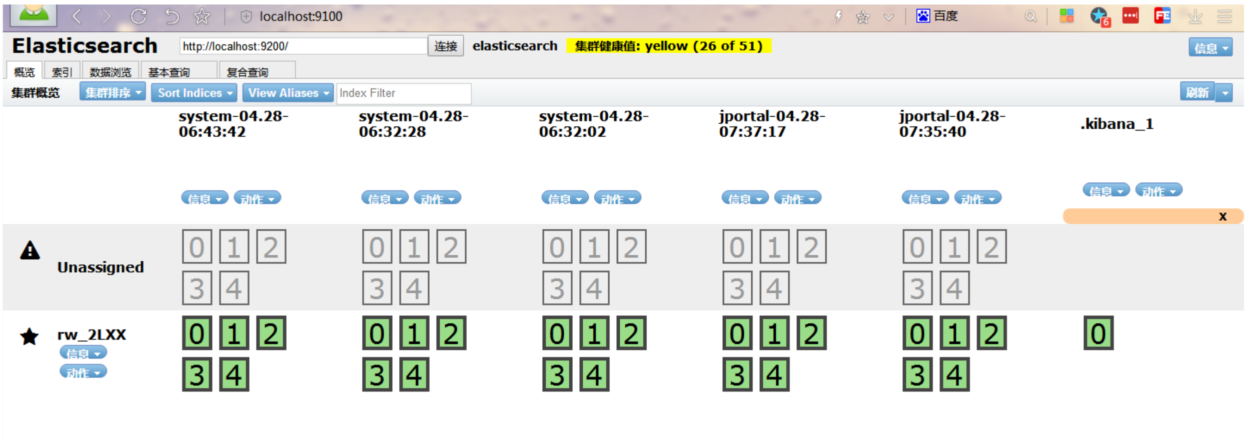


```
D:\software\ELK\elasticsearch-head-master>npm install -g grunt-cli
C:\Users\sz_jht308\AppData\Roaming\npm\grunt -> C:\Users\sz_jht308\AppData\Roami
ng\npm\node_modules\grunt-cli\bin\grunt
+ grunt-cli@1.3.2
added 152 packages from 122 contributors in 28.58s

D:\software\ELK\elasticsearch-head-master>grunt
^C终止批处理操作吗(Y/N)? y

D:\software\ELK\elasticsearch-head-master>
D:\software\ELK\elasticsearch-head-master>npm run  start

> elasticsearch-head@0.0.0 start D:\software\ELK\elasticsearch-head-master
> grunt server

Running "connect:server" (connect) task
Waiting forever...
Started connect web server on http://localhost:9100
```

接着重启elasticsearch服务即可!

启动head插件

访问：http://localhost:9100，出现如下界面，并能连接表示成功



http://localhost:9200/

elasticsearch 集群健康值: yellow (26 of 51)

	system-04.28-06:43:42	system-04.28-06:32:28	system-04.28-06:32:02	jportal-04.28-07:37:17	jportal-04.28-07:35:40	.kibana_1
Unassigned	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	
★ rw_2LXX	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0 1 2 3 4	0

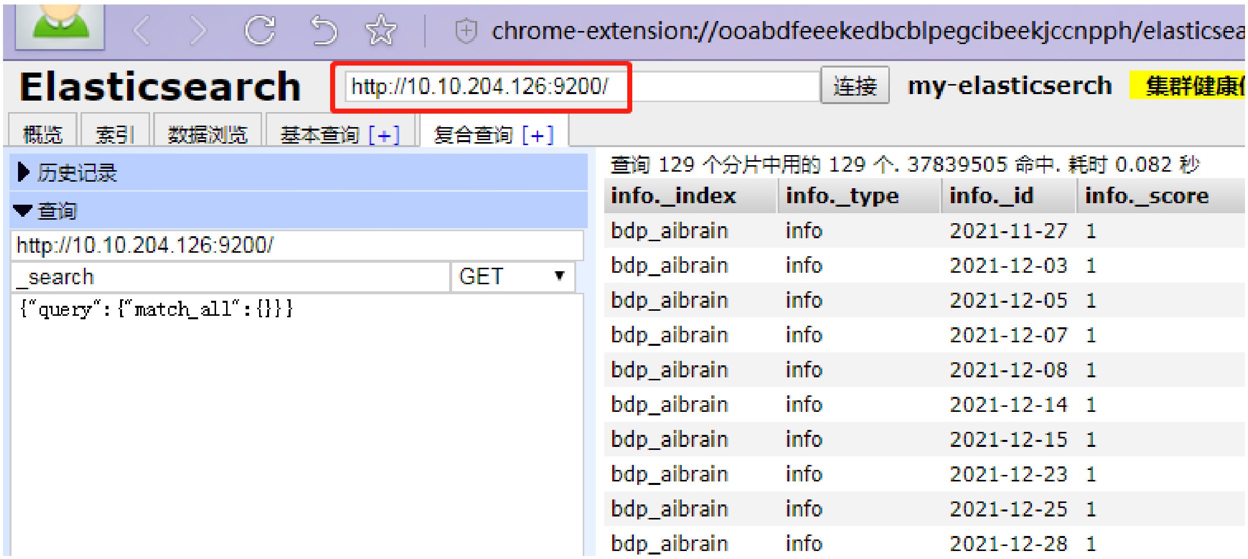
c）插件安装方法三（推荐）

下载浏览器插件，输入es地址，即可访问es

插件地址：

> https://github.com/liufengji/es-head/blob/master/elasticsearch-head.crx

输入IP地址：



http://10.10.204.126:9200/

http://10.10.204.126:9200/

_search

GET

```
{"query":{"match_all":{}}}
```

查询 129 个分片中用的 129 个. 37839505 命中. 耗时 0.082 秒

info._index	info._type	info._id	info._score
bdp_aibrain	info	2021-11-27	1
bdp_aibrain	info	2021-12-03	1
bdp_aibrain	info	2021-12-05	1
bdp_aibrain	info	2021-12-07	1
bdp_aibrain	info	2021-12-08	1
bdp_aibrain	info	2021-12-14	1
bdp_aibrain	info	2021-12-15	1
bdp_aibrain	info	2021-12-23	1
bdp_aibrain	info	2021-12-25	1
bdp_aibrain	info	2021-12-28	1

> **好像插件有问题,在复合查询中,仅支持get方式**