**Elasticsearch Linux环境部署文档**

|  |  |
| --- | --- |
| 文档名称 | elasticsearch环境部署文档 |
| 拟制 | 胡伟明 |
| 版本 | V0.7 |
| 审核 |  |

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 修订版本号 | 修改时间 | 拟制人/修改人 | 更新主要内容 |
| V0.1 | 2017-4-5 | 胡伟明 | 文档初稿 |
| V0.2 | 2017-4-6 | 胡伟明 | 修改Elasticsearch配置文件 |
| V0.3 | 2017-4-10 | 胡伟明 | 添加kibana配置说明 |
| V0.4 | 2017-4-11 | 胡伟明 | 添加logstash定时导入策略说明 |
| V0.5 | 2017-4-25 | 胡伟明 | 增加ES数据备份与恢复 |
| V0.6 | 2017-5-2 | 胡伟明 | 增加kibana监控license过期处理 |
| V0.7 | 2017-6-8 | 胡伟明 | 修改4.3节，elasticsearch.yml配置文件 |
| V0.8 | 2017-6-12 | 胡伟明 | 新增4.6节 |
| V0.9 | 2017-6-12 | 胡伟明 | 将配置文件中的IP替换为hostname |
| 备注：每次更新时需要填写此表 | | | |

目录

[1.0 elasticsearch的安装与配置 3](#_Toc484534695)

[1.1 elasticsearch 部署软件硬件兼容性说明 3](#_Toc484534696)

[1.2 elasticsearch安装包说明 3](#_Toc484534697)

[2.0 JDK安装配置 4](#_Toc484534698)

[2.1安装JDK 4](#_Toc484534699)

[4.0部署elasticsearch、 5](#_Toc484534700)

[4.1创建非root账户，并分配到root组 5](#_Toc484534701)

[4.2 修系统参数 6](#_Toc484534702)

[4.3 elasticsearch 安装 6](#_Toc484534703)

[4.4插件安装 9](#_Toc484534704)

[4.4 多节点部署 13](#_Toc484534705)

[4.5数据同步 15](#_Toc484534706)

[5.0 FAQ 18](#_Toc484534707)

[5.1 从oracle导入数据报错 18](#_Toc484534708)

[5.2 数据增量跟新，总是缺少数据 18](#_Toc484534709)

[5.3 添加端口白名单 19](#_Toc484534710)

[5.4 安装elasticsearch-head插件报错 19](#_Toc484534711)

[6. 维护 21](#_Toc484534712)

[6.1 数据备份与恢复 21](#_Toc484534713)

[6.2 备份快照迁移 25](#_Toc484534714)

[6.3 kibana 监控license过期问题 26](#_Toc484534715)

# 1.0 elasticsearch的安装与配置

## 1.1 elasticsearch 部署软件硬件兼容性说明

|  |  |  |
| --- | --- | --- |
| **规格描述** | **描述** | **备注** |
| **客户端软件规格** | Chrome浏览器 | 版本：54.0.2840.71  内核版本： |
| **系统硬件规格** | 型号：Dell R730  处理器：因特尔至强E5-2630 v3 2.4GHz  内存：128GB  硬盘：4.8TB | / |
| **系统软件规格** | CentOS7 | CentOS7 64位 |
| **数据库规格** | Oracle | win3 2\_11gR1\_database |

## 1.2 elasticsearch安装包说明

|  |  |
| --- | --- |
| 软件名称 | 描述 |
| Elasticsearch-5.2.2.tar.gz | Elasticsearch 主程序包 |
| Elasticsearch-analysis-ik-5.2.2.zip | Elasticsearch 中文分词插件包 |
| Elasticsearch-head.zip | Elasticsearch head 插件包 |
| Jdk-8u121-linux-x64.tar.gz | Java运行环境 |
| Kibana-5.2.2-linux-x86\_64.tar.gz | Elasticsearch 集成化管理工具 |
| Logstash-5.2.2.tar.gz | Elasticsearch 数据导入组件 |
| Node-v7.7.4-linux-x64.tar.gz | Node安装包 |
| ojdbc5.jar | Oracle驱动 |
| x-pack-5.2.2.zip | Elasticsearch 和 Kibana的功能扩展包 |

# 2.0 JDK安装配置

## 2.1安装JDK

Linux系统一般会自带OpenJDK，需要卸载，首先我们要卸载自带JDK

#java –version

Java version “1.4.2”

#rpm –qa|grep java 或者 rpm –qa|grep jdk 会列出系统所装的jdk信息

java-1.8.0-openjdk-1.8.0.102-4.b14.el7.x86\_64

tzdata-java-2016g-2.el7.noarch

javapackages-tools-3.4.1-11.el7.noarch

java-1.8.0-openjdk-headless-1.8.0.102-4.b14.el7.x86\_64

java-1.7.0-openjdk-headless-1.7.0.111-2.6.7.8.el7.x86\_64

java-1.7.0-openjdk-1.7.0.111-2.6.7.8.el7.x86\_64

python-javapackages-3.4.1-11.el7.noarch

# rpm -e --nodeps java-1.8.0-openjdk-1.8.0.102-4.b14.el7.x86\_64

# rpm -e --nodeps java-1.8.0-openjdk-headless-1.8.0.102-4.b14.el7.x86\_64

# rpm -e --nodeps java-1.7.0-openjdk-headless-1.7.0.111-2.6.7.8.el7.x86\_64

# rpm -e --nodeps java-1.7.0-openjdk-1.7.0.111-2.6.7.8.el7.x86\_64

再次查看java版本，提示不存在，说明删除成功

# java -version

-bash: /usr/bin/java: No such file or directory

JDK1.8安装(官方建议1.8或更高，一般建议1.8,更高版本需要市场检验)

# tar -zxvf jdk-8u121-linux-x64.tar.gz

# mv jdk1.8.0\_121 jdk8

配置jdk环境变量

#vim /etc/profile

export JAVA\_HOME=/usr/local/jdk

export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar

export PATH=$PATH:$HOME/bin:$JAVA\_HOME/bin

# source ~/.bash\_profile --使环境变量生效

# java -version

java version "1.8.0\_121"

Java(TM) SE Runtime Environment (build 1.8.0\_121-b13)

Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)

配置环境变量有三种方式：

#vim /etc/profile 所有用户生效

#vim ~/.bash\_profile 所在用户组生效

#vim ~/.bashrc 当前用户生效

环境变量配置视情况而定

# 4.0部署elasticsearch、

## 4.1创建非root账户，并分配到root组

ES2.x后不允许使用root账户直接运行es服务

#useradd esserver –添加用户

#passwd esserver –设置密码

#gpasswd -a esserver root –吧esserver用户添加到root用户组（主要是权限问题）

如无特殊说明，下面的安装均在esserver用户下进行

## 4.2 修系统参数

Elasticsearch 5.x 版本较2.x版本改动较大，部分系统参数需要调整

root账户修改

# vim /etc/security/limits.conf

esserver hard nofile 65536

esserver soft nofile 65536

# 一下参数允许 es用户锁定内存

esserver soft memlock unlimited

esserver hard memlock unlimited

保存

es账户重新登录

修改该参数：

解决max file descriptors [4096] for elasticsearch process is too low, increase to at least [65536] 这个问题

root 账户登录

# sysctl -w vm.max\_map\_count=262144 -- 临时修改参数

# vim /etc/sysctl.conf

加入：

vm.max\_map\_count=262144

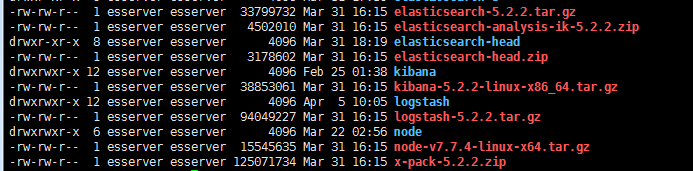
保存，永久修改

修改该参数：

解决max virtual memory areas vm.max\_map\_count [65530] is too low, increase to at least [262144] 这个问题

## 4.3 elasticsearch 安装

通过SSH工具将需要的软件上传到指定目录（/usr/local/es5），注意使用esserver账号上传



elasticsearch安装

$tar -zxvf elasticsearch-5.2.2.tar.gz

$mv elasticsearch-5.2.2 elasticsearch-1

$cd elasticsearch-1

$vim config/elasticsearch.yml

# ======================== Elasticsearch Configuration =========================

# ---------------------------------- Cluster -----------------------------------

#

# Use a descriptive name for your cluster:

#

cluster.name: zh-es5

#

# ------------------------------------ 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.rack: r1

#

# ----------------------------------- Paths ------------------------------------

#

# Path to directory where to store the data (separate multiple locations by comma):

#

path.data: /data/elasticsearch/instance1/data

#

# Path to log files:

#

path.logs: /data/elasticsearch/instance1/logs

#

# ----------------------------------- Memory -----------------------------------

#

# Lock the memory on startup:

# ES锁住内存

bootstrap.memory\_lock: true

#

# Make sure that the `ES\_HEAP\_SIZE` environment variable 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: es\_hostname

#

# Set a custom port for HTTP:

#

http.port: 9200

transport.tcp.port: 9300

# 下面配置，只在data node 节点配置，只提供9200端口对外提供服务

http.enabled: true

# For more information, see the documentation at:

# <http://www.elastic.co/guide/en/elasticsearch/reference/current/modules-network.html>

#

# --------------------------------- 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]"]

#集群初始化所有节点，以后新加节点可以发现的节点ip及端口

discovery.zen.ping.unicast.hosts: ["es\_hostname:9300"," es\_hostname:9301"," es\_hostname:9302"]

#

# Prevent the "split brain" by configuring the majority of nodes (total number of nodes / 2 + 1):

# 一般配置为 (节点数/2+1) 如：(3node/2 +1 ) =2 ,大型集群配置2-4个节点即可

discovery.zen.minimum\_master\_nodes: 1

#

# For more information, see the documentation at:

# <http://www.elastic.co/guide/en/elasticsearch/reference/current/modules-discovery.html>

#

# ---------------------------------- Gateway -----------------------------------

#

# Block initial recovery after a full cluster restart until N nodes are started:

#

# gateway.recover\_after\_nodes: 3

#

# For more information, see the documentation at:

# <http://www.elastic.co/guide/en/elasticsearch/reference/current/modules-gateway.html>

#

# ---------------------------------- Various -----------------------------------

#

# Disable starting multiple nodes on a single system:

#单机可启动的节点数，本机配置3个node

node.max\_local\_storage\_nodes: 3

cluster.routing.allocation.same\_shard.host: true

http.cors.enabled: true

http.cors.allow-origin: "\*"

http.cors.allow-headers: Authorization

#action.auto\_create\_index: .security,.monitoring\*,.watches,.triggered\_watches,.watcher-history\*

#xpack.security.enabled: false

action.auto\_create\_index: true

修改JVM内存参数

$ vim config/jvm.options

-Xms30g

-Xmx30g

最小和最大内存设置一样，另外，内存设置根据需要确定，本机内存128G；为提高内存利用率采用单机3节点方式安装，每个节点分配30g内存；根据官方建议，一个节点内存最好不要超过32G，这样内存利用率是最高的。

启动es服务

$bin/elasticsearch -d -- d 表示作为后台进程运行

$ps -ef|grep java --可以查看pid

$kill -sigterm pid --可以通过kill pid 结束服务，切忌直接使用kill -9 强杀进程

--sigterm命令可以让es优雅退出。

通过浏览器查看：

http://192.168.1.70:9200/

如果启动成功，但不能查看，请检查linux 防火墙添加9200端口白名单

## 4.4插件安装

#### 4.4.1 ik中文分词安装

需要修改的部分高亮显示

# unzip elasticsearch-analysis-ik-5.2.2.zip -d elasticsearch-analysis-ik-5.2.2

# mkdir elasticsearch-1/plugins/analysis-ik/

# cp -r elasticsearch-analysis-ik-5.2.2/\* elasticsearch-1/plugins/analysis-ik/

重启es服务，若出现DictLoading即为安装成功

也可以使用root账户安装，更改权限给esserver

#### 4.4.2 kibana插件安装

# tar -zxvf kibana-5.2.2-linux-x86\_64.tar.gz

# mv kibana-5.2.2-linux-x86\_64 kibana

# cd kibana

# vim config/kibana.yml

需要修改的地方有3个

server.port: 5601 -- 默认端口为5601

server.host: “es\_hostname” -- 如果配置为127.0.0.1 则只允许本地访问；后面的安全配置会讲到这一点

elasticsearch.url: "http:// es\_hostname:9200" -- es地址

#启动kibana (root账户启动)

nohup ./bin/kibana &

# 关闭kibana

fuser -n tcp 5601

kill -9 进程号

通过浏览器访问

http://192.168.142.3:5601

#### 4.4.3 x-pack安装

elasticsearch 安装x-pack扩展包

ES中安装x-pack

# bin/elasticsearch-plugin install <file:///usr/local/es/x-pack-5.2.2.zip> -- 离线安装

-- ES\_HOME/bin/elasticsearch-plugin install x-pack -- 在线安装 在5.2.2 bin目录下是没有 elasticsearch-plugin 目录的，按照敲就可以了，下载时间可能会比较长，耐心等待

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ WARNING: plugin requires additional permissions @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@\* java.lang.RuntimePermission accessClassInPackage.com.sun.activation.registries

\* java.lang.RuntimePermission getClassLoader

\* java.lang.RuntimePermission setContextClassLoader

\* java.lang.RuntimePermission setFactory

\* java.security.SecurityPermission createPolicy.JavaPolicy\* java.security.SecurityPermission getPolicy

\* java.security.SecurityPermission putProviderProperty.BC

\* java.security.SecurityPermission setPolicy

\* java.util.PropertyPermission \* read,write

\* java.util.PropertyPermission sun.nio.ch.bugLevel write

\* javax.net.ssl.SSLPermission setEs\_hostnameVerifier

See http://docs.oracle.com/javase/8/docs/technotes/guides/security/permissions.htmlfor descriptions of what these permissions allow and the associated risks.

Continue with installation? [y/N]y

一定要输入y

修改es配置文件

http.cors.enabled: true

http.cors.allow-origin: "\*"

http.cors.allow-headers: Authorization

如果你禁用了ES的自动创建索引功能，则需要修改配置文件，在elasticsearch.yml中添加如下信息：

action.auto\_create\_index: .security,.monitoring\*,.watches,.triggered\_watches,.watcher-history\*

在访问es就需要密码

用户：elastic

密码：changeme

如果不希望密码可以再es和kibana的配置文件中添加如下配置：

xpack.security.enabled: false # 本次安装禁用密码

kibana 中安装x-pack扩展包

在Kibana 中安装x-pack插件

#bin/kibana-plugin install file:///usr/local/es/x-pack-5.2.2.zip --离线安装

-- bin/kibana-plugin install x-pack -- 在线安装

修改 kibana配置文件，添加如下配置：

xpack.security.enabled: false

启动 kibana

# 用root账户启动kibana，这样终端关闭后，可以后台挂起运行；

bin/kibana &

#### 4.4.4 head 插件安装

ES 5.x版本head 作为独立组件运行，其安装方式变化较大，需要安装node，grunt

root账号下安装node

tar -zxvf node-v7.7.4-linux-x64.tar.gz

mv node-v7.7.4-linux-x64 node

vim /etc/profile

export NODE\_HOME=/usr/local/es/node

export PAHT=$PATH:$NODE\_HOME/bin

source /etc/profile

查看版本

node -v

npm -v

root 账号下安装 grunt

#npm install -g grunt-cli

#npm install -g grunt

#grunt --version

root 账号下安装 head 组件

unzip elasticsearch-head.zip

npm install -g cnpm --registry=https://registry.npm.taobao.org #设置镜像路径为国内进行

cd elasticsearch-head

npm install # 网络原因，时间会比较长

修改 elasticsearch-head/Gruntfile.js

修改：elasticsearch-head/Gruntfile.js

新增：es\_hostname:'0.0.0.0',如下：

connect: {

server: {

options: {

es\_hostname:'0.0.0.0',

port: 9100,

base: '.',

keepalive: true

}

}

}

编辑：elasticsearch-head/\_site/app.js

#原配置

this.base\_uri = this.config.base\_uri || this.prefs.get("app-base\_uri") || "http://localhost:9200";

#将localhost 修改为ES的IP地址

this.base\_uri = this.config.base\_uri || this.prefs.get("app-base\_uri") || "http:// es\_hostname:9200";

启动：elasticsearch-head

#cd elasticsearch-head --必须要进到该目录才能启动

#./node\_modules/grunt/bin/grunt server & #注意也是用root账户启动

nohup grunt server &exit

关闭head

fuser -n tcp 9100

kill pid

# npm install

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.0.0 (node\_modules/chokidar/node\_modules/fsevents):

npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.1.1: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})

## 4.4 多节点部署

目前单台物理机上配置3个节点

cp –r elasticsearch-1 elasticsearch-2

cp –r elasticsearch-1 elasticsearch-3

修改elasticsearch-2/config/elasticsearch.yml

cluster.name: zh-es5

node.name: node-2

node.master: true

node.data: true

path.data: /data/elasticsearch/instance2/data

path.logs: /data/elasticsearch/instance2/logs

bootstrap.memory\_lock: true

network.host: es\_hostname

http.port: 9201

transport.tcp.port: 9301

discovery.zen.ping.unicast.hosts: ["es\_hostname:9300"," es\_hostname:9301"," es\_hostname:9302"]

discovery.zen.minimum\_master\_nodes: 1

node.max\_local\_storage\_nodes: 3

cluster.routing.allocation.same\_shard.host: true

http.cors.enabled: true

http.cors.allow-origin: "\*"

http.cors.allow-headers: Authorization

action.auto\_create\_index: true

xpack.security.enabled: false

修改elasticsearch-3/config/elasticsearch.yml

cluster.name: zh-es5

node.name: node-3

node.master: true

node.data: true

path.data: /data/elasticsearch/instance3/data

path.logs: /data/elasticsearch/instance3/logs

bootstrap.memory\_lock: true

network.host: es\_hostname

http.port: 9202

transport.tcp.port: 9302

discovery.zen.ping.unicast.hosts: ["es\_hostname:9300"," es\_hostname:9301"," es\_hostname:9302"]

discovery.zen.minimum\_master\_nodes: 1

node.max\_local\_storage\_nodes: 3

cluster.routing.allocation.same\_shard.host: true

http.cors.enabled: true

http.cors.allow-origin: "\*"

http.cors.allow-headers: Authorization

action.auto\_create\_index: true

xpack.security.enabled: false

分别启动 elasticsearch-1，elasticsearch-2，elasticsearch-3

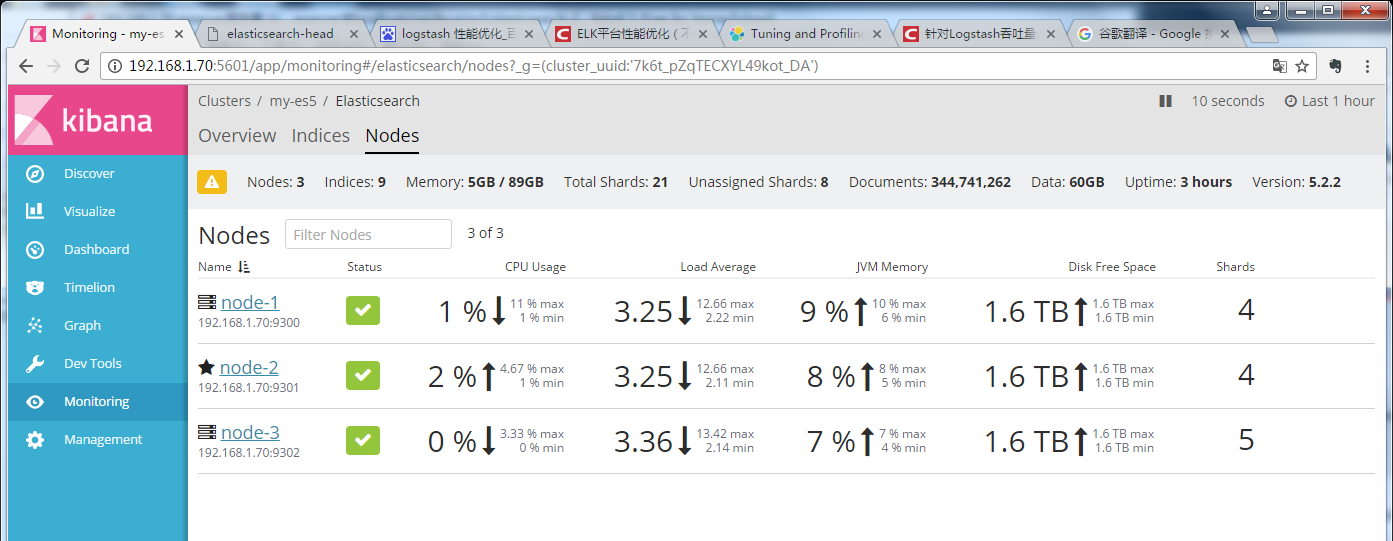
elasticsearch-1/bin/elasticsearch.yml -d

elasticsearch-2/bin/elasticsearch.yml -d

elasticsearch-3/bin/elasticsearch.yml -d

通过浏览器访问，查看启动情况：

<http://192.168.1.70:5601>



## 4.5数据同步

#### 4.5.1 logstash 安装

wget https://artifacts.elastic.co/downloads/logstash/logstash-5.2.1.tar.gz

tar -zxvf logstash-5.2.1.tar.gz

安装x-pack

./bin/logstash-plugin install <file:///usr/local/es/x-pack-5.2.2.zip>

修改 config/logstash.yml

# 指定es地址

xpack.monitoring.elasticsearch.url: “es\_hostname:9200”

#### 4.5.2 gem安装

gem 安装（root账户安装）

# yum install gem

替换国内的镜像

# gem sources --add https://ruby.taobao.org/ --remove https://rubygems.org/

查看是否替换成功

# gem sources -l

\*\*\* CURRENT SOURCES \*\*\*

https://ruby.taobao.org/ #这里为taobao站点即为成功

#### 4.5.3 logstash-input-jdbc插件安装

进入logstash安装目录（root）

vim Gemfile

将第一行数据替换为中国站点

https://ruby.taobao.org/

然后开始安装，时间可能会比较长

# ./bin/logstash-plugin install logstash-input-jdbc

Validating logstash-input-jdbc

Installing logstash-input-jdbc

Installation successful

#### 4.5.4 导入配置文件修改

进入logstash-input-jdbc插件安装目录

新建jdbc.conf 文件，内容如下

input {

jdbc {

# 数据库连接配置

jdbc\_connection\_string => "jdbc:oracle:thin:@//192.168.1.170:1521/ZHTD"

jdbc\_user => "ZHWD"

jdbc\_password => "ZHWD"

jdbc\_driver\_library => "/usr/local/es/logstash/ojdbc5.jar"

jdbc\_driver\_class => "Java::oracle.jdbc.driver.OracleDriver"

# 需要执行的sql文件

statement\_filepath => "jdbc.sql"

# 每秒执行一次

schedule => "\* \* \* \* \* \*"

#type 可以增加外字段，仅仅用于标识用

#type => "detect\_data"

sql\_log\_level => "debug"

jdbc\_default\_timezone => "Asia/Shanghai"

#use\_column\_value => true

#tracking\_column => ser\_time

# 用于记录上一次执行的元信息，如上次执行时间，可利用这一点做增量更新

last\_run\_metadata\_path => "/usr/local/es/logstash/detect\_data.json"

}

}

output {

stdout {

# 输出格式为json，调试时使用

codec => json\_lines

}

elasticsearch {

hosts => " es\_hostname:9200"

# 索引类型，导入前先建好索引

index => "detect\_data"

# 文档类型

document\_type => "detect\_data"

# 文档id，如果不指定系统会自动生成

document\_id => "%{data\_id}"

}

}

新建jdbc.sql文件，内容如下

select \* from detect\_data where (ser\_time + 2/(24\*60\*60)) >= :sql\_last\_value and city is not null

说明 :sql\_last\_value 取值就是 last\_run\_metadata\_path 配置的文件内容

./bin/logstash -f jdbc.conf

可以通过查看detect\_data.json 中的日期是否定时跟新，验证定时任务是否执行

关于schedule简单说明：

schedule => "10,20,30 \* \* \* \* \*"

以上配置，将在每分钟第10秒，第20秒，第30秒执行计划

## 4.6 镜像恢复后修改相应配置文件

该步骤仅适用于通过镜像文件恢复的ES

#### 4.6.1 修改 ES配置文件ES\_HOME/config/elasticsearch.yml

# -------------------------------cluster name------------------------------

cluster.name: zh-es5

# -------------------------------node info------------------------------

node.name: node-1

node.master: true

node.data: true

# -------------------------------log and data path------------------------------

path.data: /data/elasticsearch/instance1/data

path.logs: /data/elasticsearch/instance1/logs

# backup path

path.repo: ["/data/elasticsearch/backups"]

# ----------------------------------- Memory -----------------------------------

bootstrap.memory\_lock: true

# ---------------------------------- Network -----------------------------------

network.host: 192.168.1.22

http.port: 9200

transport.tcp.port: 9300

# --------------------------------- Discovery ----------------------------------

discovery.zen.ping.unicast.hosts: ["es\_hostname:9300"," es\_hostname:9301"," es\_hostname:9302"]

discovery.zen.minimum\_master\_nodes: 2

# ---------------------------------- Various -----------------------------------

node.max\_local\_storage\_nodes: 3

cluster.routing.allocation.same\_shard.host: true

http.cors.enabled: true

http.cors.allow-origin: "\*"

http.cors.allow-headers: Authorization

xpack.security.enabled: false

action.auto\_create\_index: true

注意：elasticsearch-1，elasticsearch-2，elasticsearch-3全部要修改

#### 4.6.2 修改kibana配置文件，KIBANA\_HOME/config/kibana.yml

server.port: 5601

server.host: " es\_hostname "

elasticsearch.url: "http:/es\_hostname:9200"

xpack.security.enabled: false

#### 4.6.3 修改elasticsearch-head 配置文件，ELASTICSEARCH\_HEAD\_HOME/\_site/app.js



将此处ip更换为实际ES 服务所在IP即可。

# 5.0 FAQ

## 5.1 从oracle导入数据报错

Caused by: org.elasticsearch.index.mapper.MapperParsingException: malformed mapping no root object found

在2.x以后需要先建好索引，在导入数据

ES启动正常，但是无法导入数据，报错：

org.elasticsearch.client.transport.NoNodeAvailableException: no cluster nodes available,

检查jdbc导入对应的sh文件，检查集群名称，ip，端口是否正确

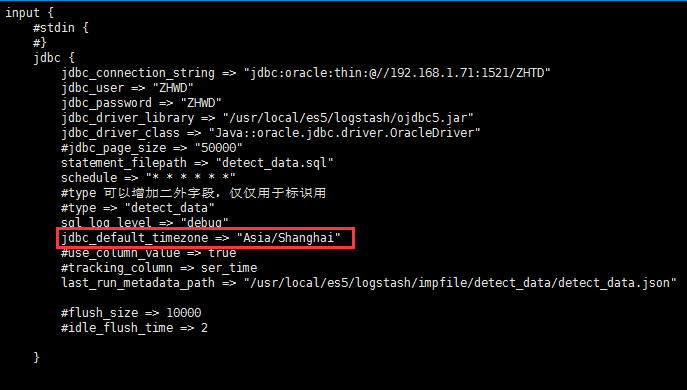
数据库驱动明明添加了，但还是报错，说找不到驱动

可能是驱动版本与当前数据库不匹配，需要寻找合适的驱动

## 5.2 数据增量跟新，总是缺少数据

1、在生产环境中，一般是将数据库和ES部署在不同的机器上面，导致增量更新数据缺失的问可能是机器之间时间不同步导致的，因此需要做ntp时间同步。

2、可能是时区问题导致，需要在导入脚本中添加时区配置，如下：



## 5.3 添加端口白名单

# firewall-cmd --zone=public --add-port=9200/tcp –permanent

出现success表明添加成功

--zone #作用域

--add-port=9200/tcp #添加端口，格式为：端口/通讯协议

--permanent #永久生效，没有此参数重启后失效

# 查询防火墙状态

# firewall-cmd –state

#关闭防火墙

systemctl stop firewalld.service

#开启防火墙

systemctl start firewalld.service

重启防火墙

# systemctl restart firewalld.service

#禁止firewall开机启动

systemctl disable firewalld.service

#开启firewall 开机启动

# systemctl enable firewalld.service

## 5.4 安装elasticsearch-head插件报错

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.0.0 (node\_modules/chokidar/node\_modules/fsevents):

npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.1.1: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})

npm WARN elasticsearch-head@0.0.0 license should be a valid SPDX license expression

npm ERR! Linux 3.10.0-514.el7.x86\_64

npm ERR! argv "/usr/local/es5/node/bin/node" "/usr/local/es5/node/bin/npm" "install"

npm ERR! node v7.7.4

npm ERR! npm v4.1.2

npm ERR! code ELIFECYCLE

npm ERR! phantomjs-prebuilt@2.1.14 install: `node install.js`

npm ERR! Exit status 1

npm ERR!

npm ERR! Failed at the phantomjs-prebuilt@2.1.14 install script 'node install.js'.

npm ERR! Make sure you have the latest version of node.js and npm installed.

npm ERR! If you do, this is most likely a problem with the phantomjs-prebuilt package,

npm ERR! not with npm itself.

npm ERR! Tell the author that this fails on your system:

npm ERR! node install.js

npm ERR! You can get information on how to open an issue for this project with:

npm ERR! npm bugs phantomjs-prebuilt

npm ERR! Or if that isn't available, you can get their info via:

npm ERR! npm owner ls phantomjs-prebuilt

npm ERR! There is likely additional logging output above.

npm ERR! Please include the following file with any support request:

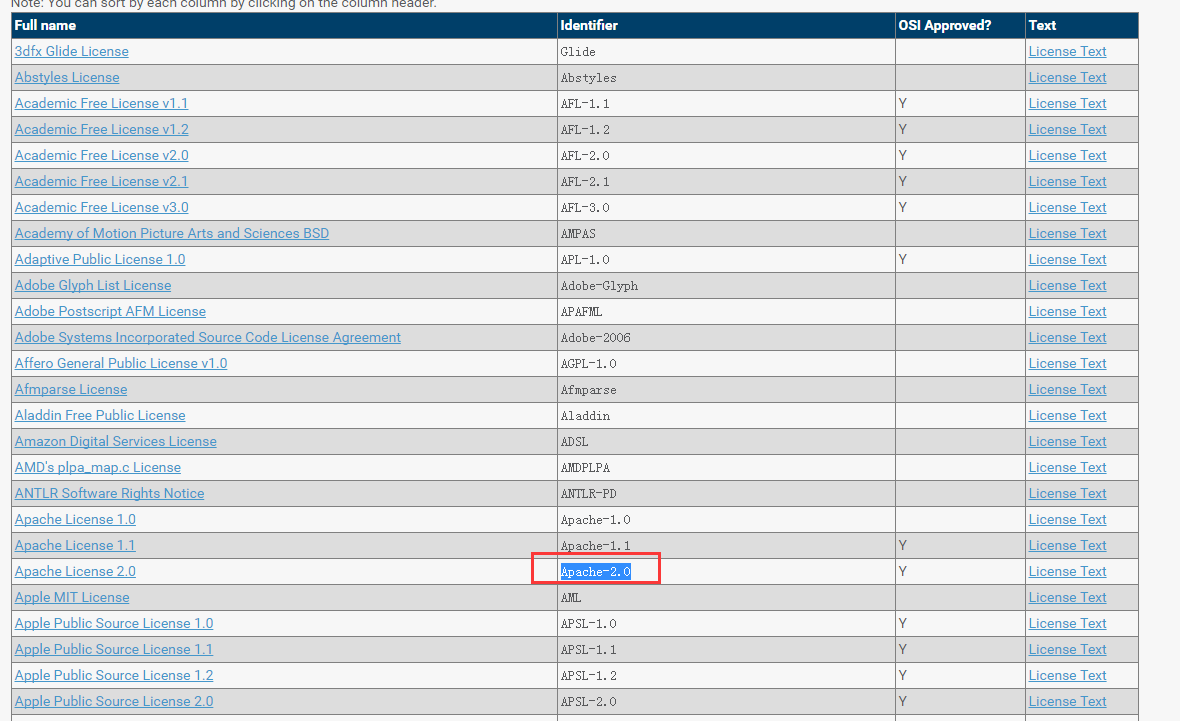
npm ERR! /usr/local/es5/elasticsearch-head/npm-debug.log

关于spdx license 详情可以参考官网：<https://spdx.org/licenses/>

简单来说就是一个开源软件使用声明。

既然知道spdx是个啥，解决就容易了；打开elasticsearch-head目录下的package.json

文件，找到license位置，修改为上面网站上存在的Identifier就可以了，把原来的Apache内容修改为Apache-2.0，如下图，吧：



5.4 number of documents in the index cannot exceed 2147483519

分片上文档数达到20亿上限，无法插入新文档

重建索引，增加分片；也可以增加节点；

# 6. 维护

## 6.1 数据备份与恢复

#### 6.1.1 数据备份

ES是通过Snapshot（快照）的方式来实现数据备份，并且是以增量的方式，因此第一次备份花费的时间相对较长。为了做快照，就需要注册快照厂库，告诉ES快照存储在哪里。

ES的快照仓库支持以下几种形式：

* 共享的文件系统
* Amazon S3
* HDFS（Hadoop Distributed File System）
* Azure Cloud

我们选择第一种形式，使用简单，也符合我们的使用场景。

**修改配置文件**

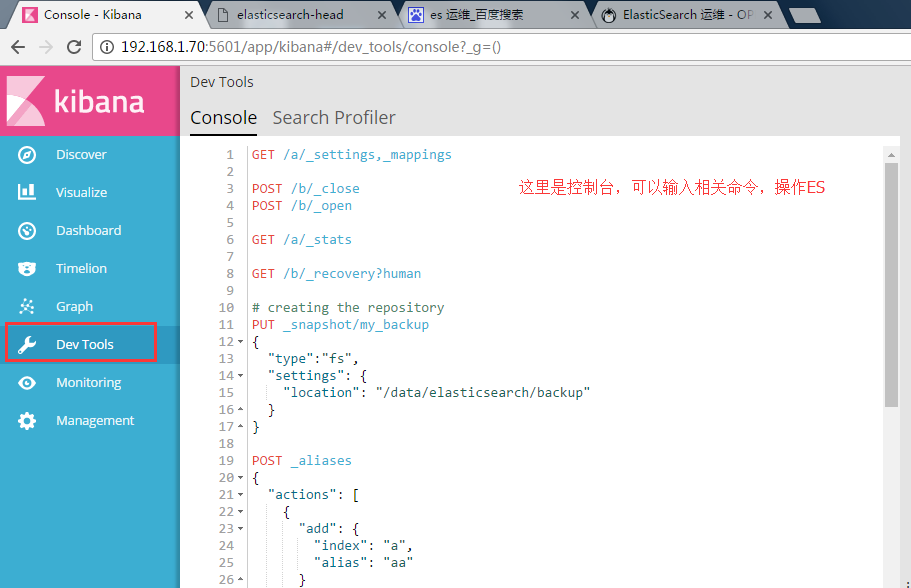
修改ES配置文件config/elasticsearch.yml，每个节点都要修改

path.repo: [“/data/elasticsearch/backups”]

该目录一般不能与ES的数据节点在同一硬盘中，分开存储能降低数据丢失概率，最大化保障数据完整性。

**注册快照仓库**

为了方便操作，我们选择在Kibana平台进行操作：



PUT \_snapshot/my\_backup

{

"type":"fs",

"settings": {

"location": "/data/elasticsearch/backups"

}

}

这样就注册了一个名为my\_backup的仓库，这里的location 要注意，必须是每个节点都能访问并且有写权限的共享目录。

除了location外，还有一些额外选项可以设置：

* compress 是否压缩
* max\_snapshot\_bytes\_per\_sec 制作快照的速度，默认40mb/s
* max\_restore\_bytes\_per\_sec 快照恢复的速度，默认40mb/s

**更新仓库设置**：

POST \_snapshot/my\_backup

{

"type":"fs",

"settings":{

"location":"/data/elasticsearch/backups",

"max\_snapshot\_bytes\_per\_sec" : "50mb",

"max\_restore\_bytes\_per\_sec" : "50mb",

"compress" : true

}

}

注意，注册仓库用的是PUT，而更新仓库用的是POST

**检查注册仓库的信息**

GET \_snapshot/my\_backup

**开始备份**

指定快照名称为snapshot\_detect\_data\_20170424

PUT \_snapshot/my\_backup/snapshot\_detect\_data\_20170424

{

"indices": "detect\_data"

}

需要备份多个索引，可以采用以下方式：

PUT \_snapshot/my\_backup/ snapshot\_detect\_data\_20170424

{

"indices": "detect\_data,detect\_data2"

}

**备份状态查看**

GET \_snapshot/my\_backup/snapshot\_detect\_data\_20170424/\_status

主要有如下几种状态：

* INITIALIZING 集群状态检查，检查当前集群是否可以做快照
* IN\_PROCESSING 正在转移数据到仓库
* FINALIZING 数据转移完成，正在转移元信息
* DONE 完成
* FAILED 备份失败

**取消备份**

DELETE \_snapshot/my\_backup/snapshot\_detect\_data\_20170424

#### 6.1.2 恢复备份

一般情况下，执行备份恢复都是在原有索引无法对外正常提供服务的情况下发生的，因此需要删除对应索引，并重新建立该索引，关闭后才能进行恢复，具体操作如下：

**删除损坏的索引**：

DELETE /detect\_data

**新建索引**

PUT detect\_data

{

"settings": {

"number\_of\_replicas": 0,

"number\_of\_shards": 3

}

}

number\_of\_replicas: 副本数，一般在分布式集群环境下使用，默认值为1

number\_of\_shards: 分片数量，默认为5

**关闭索引**

POST /detect\_data/\_close

**打开索引，恢复之后需要打开索引**

POST /detect\_data/\_open

**查看所有备份备份信息**

GET \_snapshot/my\_backup/\_all

**恢复特定备份下的所有索引**

POST \_snapshot/my\_backup/ snapshot\_detect\_data\_20170424/\_restore

**恢复指定索引**

POST \_snapshot/my-backup/ snapshot\_detect\_data\_20170424/\_restore

{

"indices": "detect\_data"

}

**查看恢复进度**

# 查看索引的恢复进度

GET /\_snapshot/my\_backup/sp\_2017-04-21\_15\_29\_04/\_status

**取消恢复**

只需要删除索引，即可恢复

DELETE /detect\_data

#### 6.1.3 编写脚本定时备份

编写detect\_data\_backup.sh

#!/bin/sh

ls\_date=`date +%Y-%m-%d\_%H\_%M\_%S`

curl -XPUT http://192.168.1.70:9200/\_snapshot/my\_backup/sp\_detect\_data\_${ls\_date} -d '{"indices":"detect\_data"}'

ls\_date: 当前备份的时间，精确到秒

赋予脚本执行权限

chomd 755 detect\_data\_backup.sh

配置crontab

crontab -e

插入数据：

5 0 \* \* \* /data/elasticsearch/detect\_data\_backup.sh

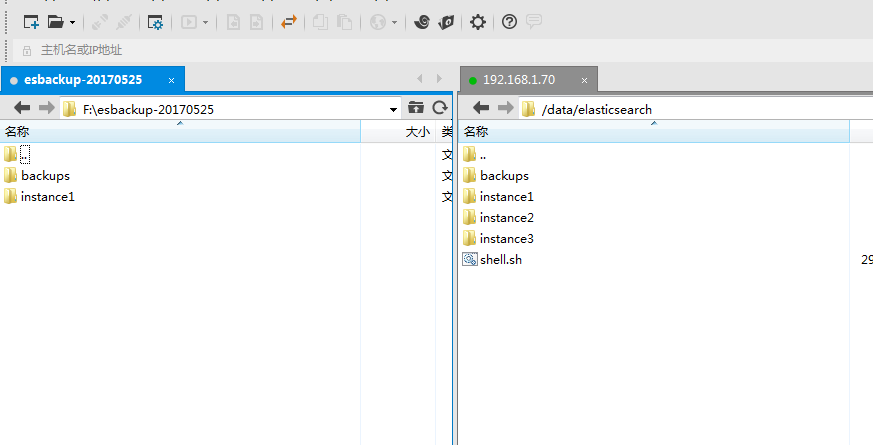
以上配置表示每天00:05 开始执行备份

关于crontab 更多用法请参考：<http://yangqijun.iteye.com/blog/1173016>

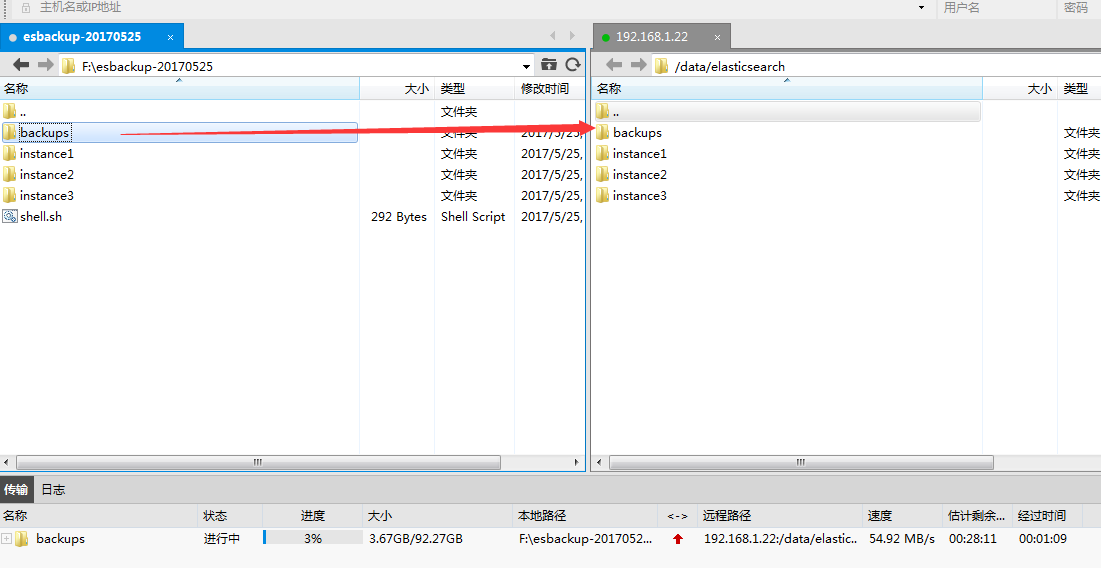
## 6.2 备份快照迁移

如果需要将快照迁移到另一个集群，只需要将备份文件全部拷贝到需要迁移的机器上面，然后在新的集群上注册一个快照仓库，设置location的位置为备份文件所在地址，最后执行恢复备份命令即可。

拷贝备份文件



将备份数据上传到指定目录



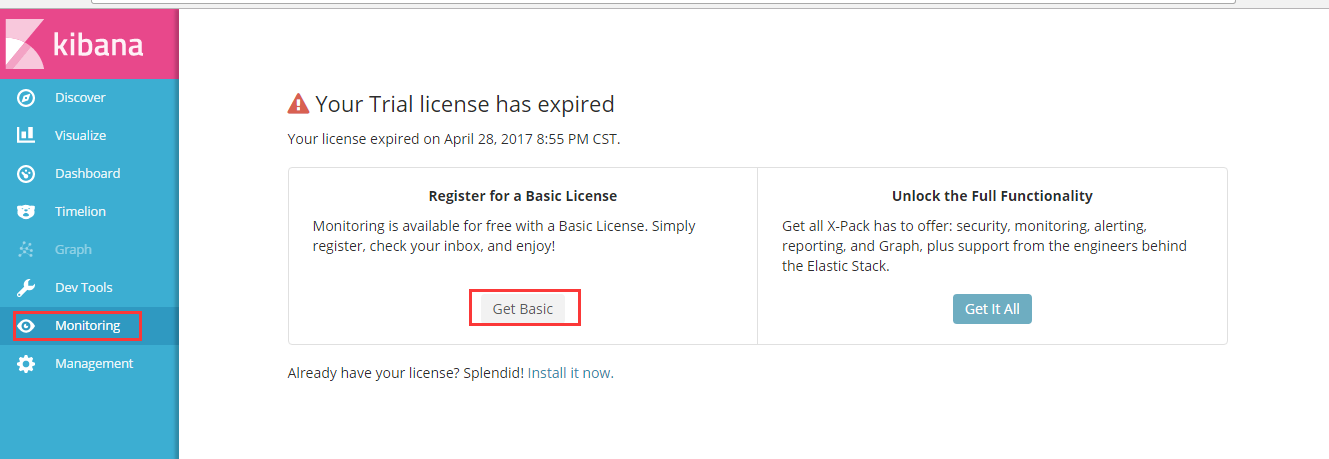
在需要导入数据的ES各个节点修改配置，参考【6.1.1】数据备份

注册快照仓库，参考【6.1.1】

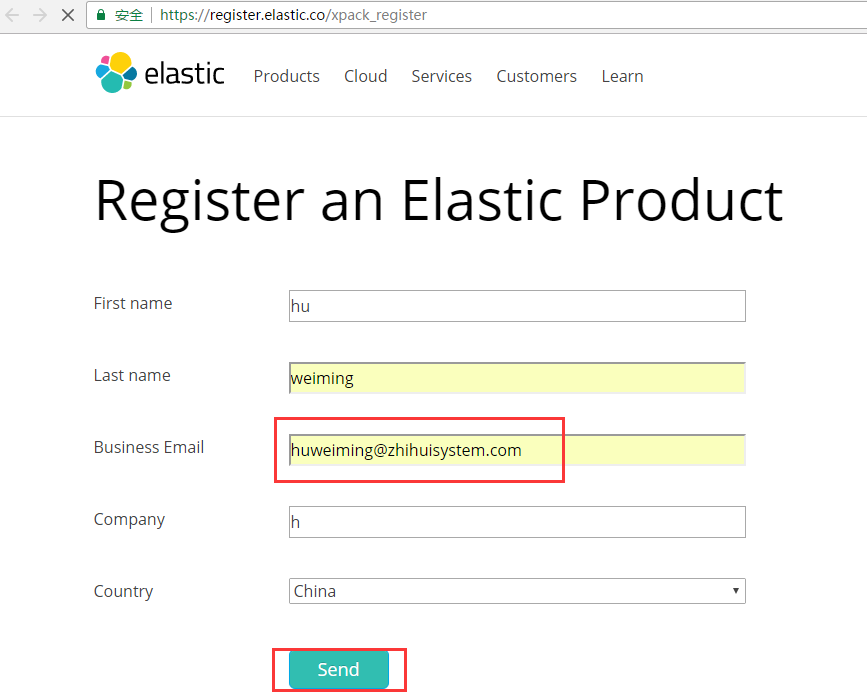
进行数据恢复

## 6.3 kibana 监控license过期问题

初始化安装x-pack的试用期为1个月，包含全部监控指标；一个月后使用license授权文件过期，如果需要继续使用监控功能，可以申请免费版本（相比试用版license缺少部分监控和技术支持）

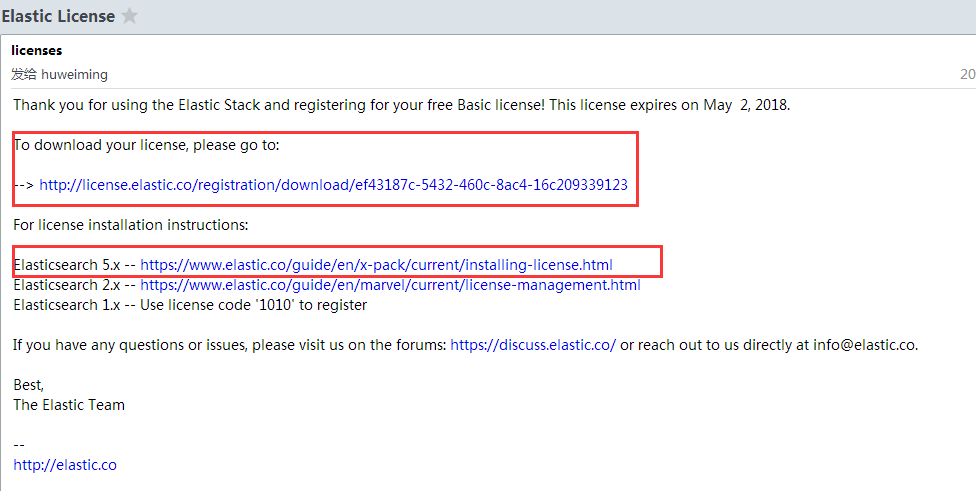


点击【Get Basic】



Business Email ：接收新license的邮箱地址

查看收到的邮件



首先下载license，然后安装。

curl -XPUT -u elastic 'http://<host>:<port>/\_xpack/license?acknowledge=true' -H "Content-Type: application/json" -d @license.json

查看当前license的详细信息

curl -XGET -u elastic:changeme 'http://<host>:<port>/\_xpack/license'

{

"license" : {

"status" : "active",

"uid" : "0a98411f-73f4-4c67-954c-724874ed5488",

"type" : "basic",

"issue\_date" : "2015-10-13T18:18:20.709Z",

"issue\_date\_in\_millis" : 1444760300709,

"expiry\_date" : "2015-11-12T18:18:20.709Z",

"expiry\_date\_in\_millis" : 1447352300709,

"max\_nodes" : 1000,

"issued\_to" : "elasticsearch",

"issuer" : "elasticsearch"

}

}

status：active 表示激活

expiry\_date：过期时间