ELK 课件

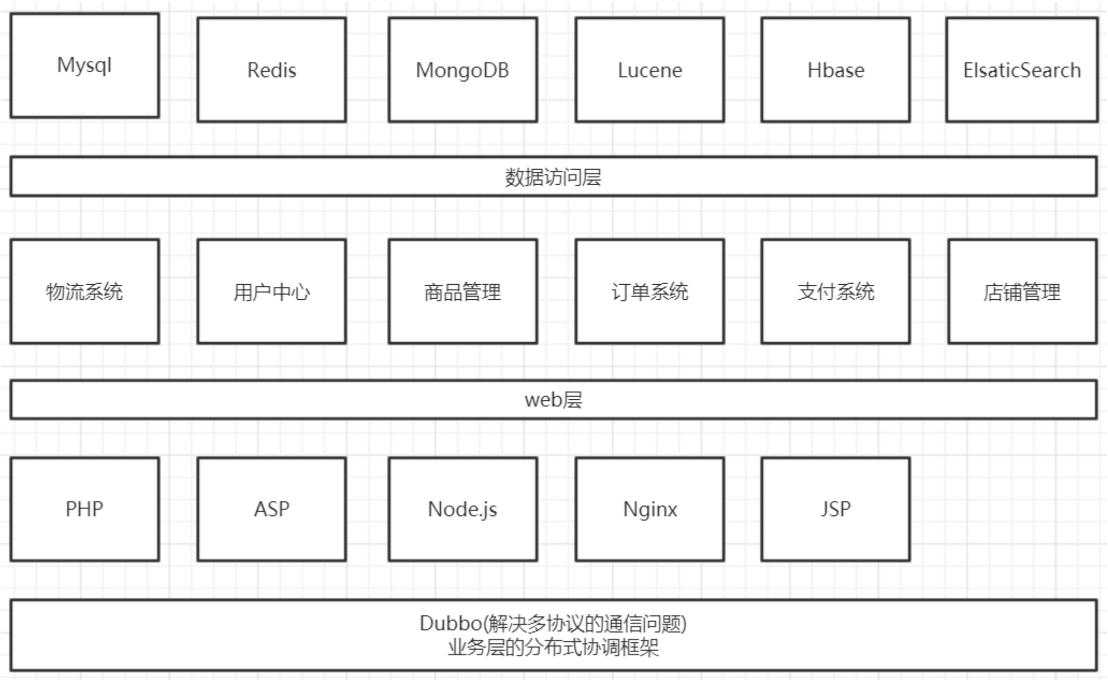
1、ELK 简介

1.1、ELK 是什么

- Elasticsearch 是个开源分布式搜索引擎，它的特点有：分布式，零配置，自动发现，索引自动分片，索引副本机制，restful 风格接口，多数据源，自动搜索负载等。
- Logstash 是一个完全开源的工具，他可以对你的日志进行收集、过滤，并将其存储供以后使用（如，搜索）。
- Kibana 也是一个开源和免费的工具，它 Kibana 可以为 Logstash 和 ElasticSearch 提供的日志分析友好的 Web 界面，可以帮助您汇总、分析和搜索重要数据日志。

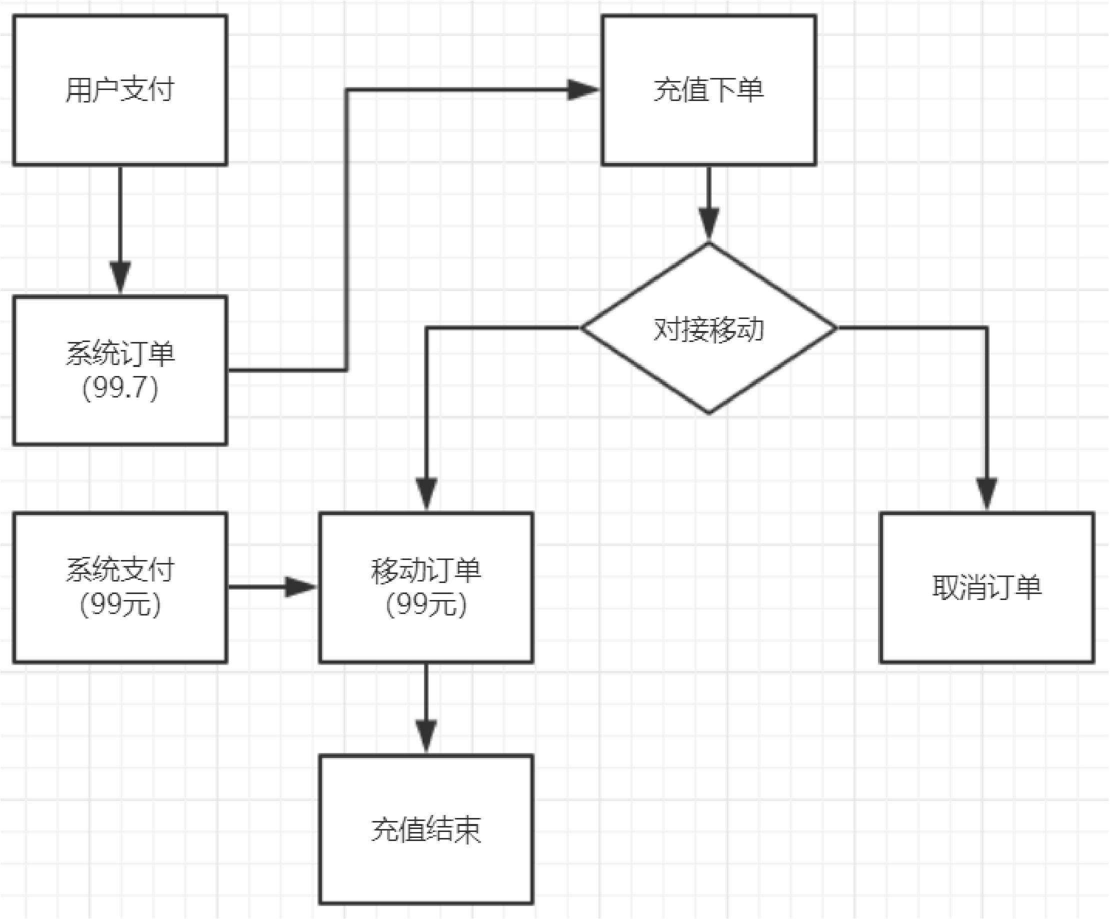
1.2、ELK 应用

- 电商体系架构



- 问题
  1、API 不一样，我们如何去整合？--》dubbo 定义统一的 api 规范
  2、各子系统之间会产生操作痕迹（用户行为轨迹）---》日志
  3、各个子系统都会生成各自的日志---日志整合--》logstash
  4、AOP 埋点，异步日志输出

- 具体场景 1
  通过第三方进行移动话费充值



日志输出：每次调用都会打印异步日志

分布式负载均衡：

很多太机器都可以充值（动态的去选择一台目前比较空闲的机器去执行这个任务）

问题：
A：兄弟，帮忙查一下今天手机号码 138001380000 充值日志记录（是否成功）
B：稍等
5 分钟后

A：怎么样了
B：稍等，还剩下 3 台机器没查完

结论：如果能把所有的日志整理在一起，就不会出现一台一台去查的问题

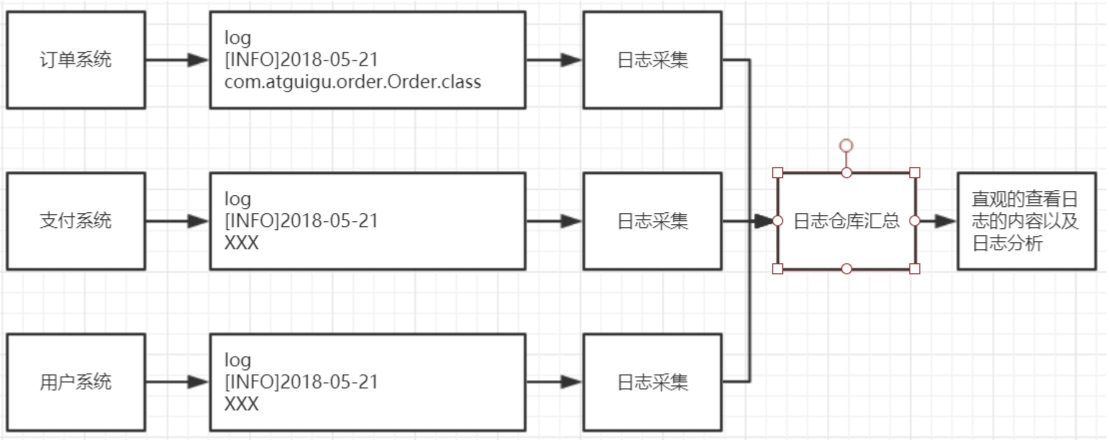
解决方案：
1、可不可以把日志放在数据库中。

数据量太大，且日志没有规范日志格式，数据库方案不太建议，且压力过大

2、采用大数据日志处理方案

成本太高，且分布式环境每个系统的日志规则不一样。

- 具体业务实践



日志收集：Logstash
日志存储：ElasticSearch
日志展示：Kibana
针对对台服务器日志不统一的问题，提供多种检索规则，方便可视化展示

- 案例总结

分布式带来的问题：多节点、负载均衡、日志分散、运维成本高（需要人为跟踪）

1.3、集中式日志管理系统

当前主流的一些集中日志管理系统
1、简单的：Rsyslog
2、商业化：Splunk
3、开源的：Scribe（FaceBook），Chukwa（Apache）
4、ELK 最广泛的（Elastic Stack）(java 语言编写)
www.elastic.co/cn

1.4、ELK

ElasticSearch	Java	实时的分布式搜索和分析引擎，他可以用于全文检索，结构化搜索以及分析，lucene。Solr
Logstash	JRuby	具有实时渠道能力的数据收集引擎，包含输入、过滤、输出模块，一般在过滤模块中做日志格式化的解析工作
Kibana	JavaScript	为 ElasticSerach 提供分析平台和可视化的 Web 平台。他可以 ElasticSerach 的索引中查找，呼唤数据，并生成各种维度的表图

1.5、日志

日志：记录程序的运行轨迹---

级别：ERROR、INFO、DEBUG、WARN

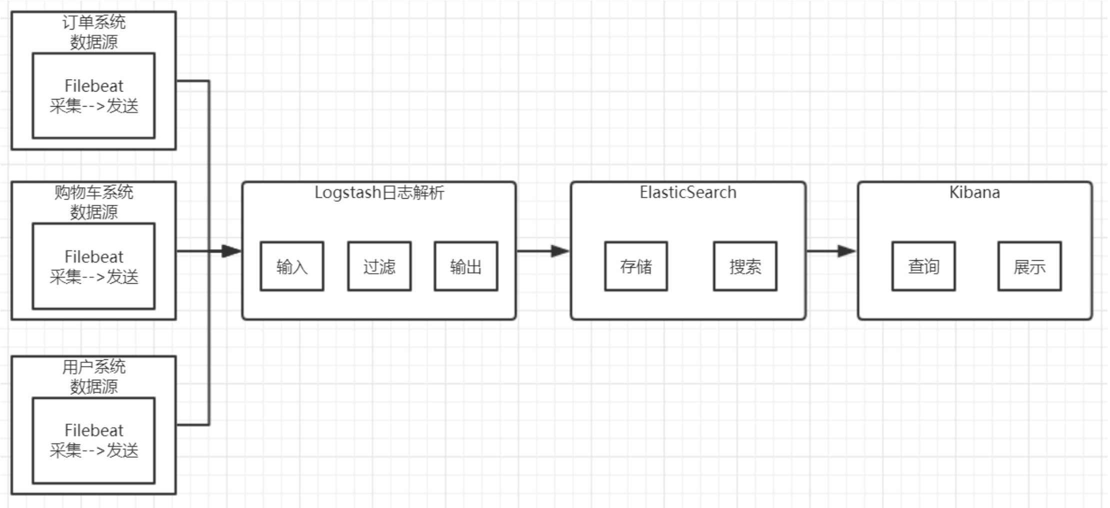
目的：方便定位和查找信息，记录除去业务外的附加的信息，链路

Filebeat 简介

当您要面对成百上千、甚至成千上万的服务器、虚拟机和容器生成的日志时，请告别 SSH 吧。Filebeat 将为您提供一种轻量型方法，用于转发和汇总日志与文件，让简单的事情不再繁杂。

当将数据发送到 Logstash 或 Elasticsearch 时，Filebeat 使用背压敏感协议，以考虑更多的数据量。如果 Logstash 正在忙于处理数据，则可以让 Filebeat 知道减慢读取速度。一旦拥堵得到解决，Filebeat 就会恢复到原来的步伐并继续运行。

无论在任何环境中，随时都潜伏着应用程序中断的风险。Filebeat 能够读取并转发日志行，如果出现中断，还会在一切恢复正常后，从中断前停止的位置继续开始。



2、准备工作

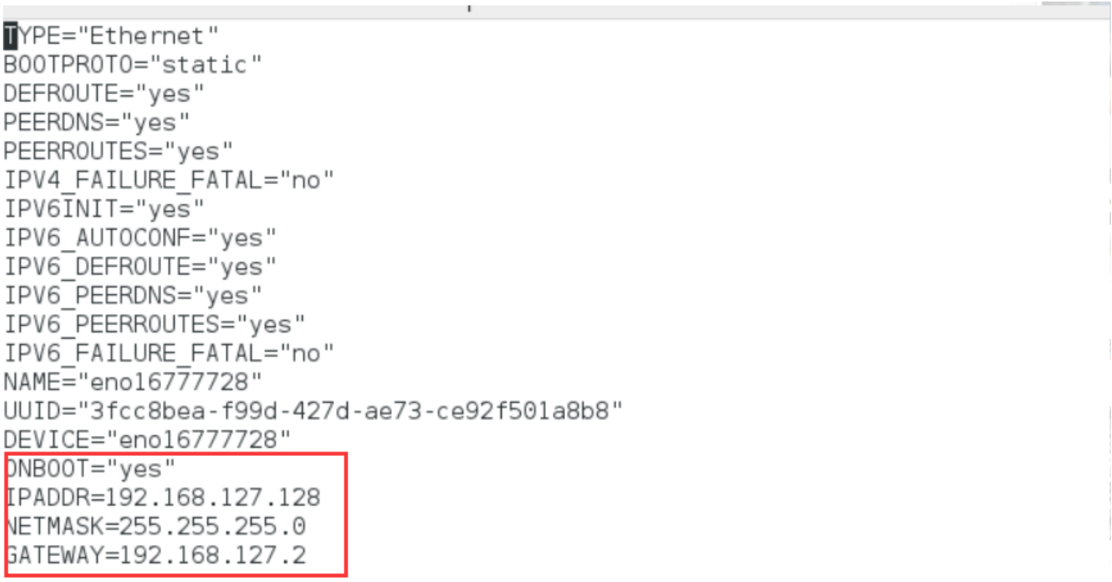
2.1、安装 Centos7

建议内存 2G 以上

2.2、基本配置

- 设置 IP 地址

vi /etc/sysconfig/network-scripts/ifcfg-eno33



```
TYPE="Ethernet"
BOOTPROTO="static"
DEFROUTE="yes"
PEERDNS="yes"
PEERROUTES="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_PEERDNS="yes"
IPV6_PEERROUTES="yes"
IPV6_FAILURE_FATAL="no"
NAME="eno16777728"
UUID="3fcc8bea-f99d-427d-ae73-ce92f501a8b8"
DEVICE="eno16777728"
ONBOOT="yes"
IPADDR=192.168.127.128
NETMASK=255.255.255.0
GATEWAY=192.168.127.2
```

service network restart

- 添加用户并授权

[root@localhost ~]# adduser elk1

[root@localhost ~]# passwd elk1

[root@localhost ~]# whereis sudoers

[root@localhost ~]# ls -l /etc/sudoers

[root@localhost ~]# chmod -v u+w /etc/sudoers

[root@localhost ~]# vi /etc/sudoers

```
## Allow root to run any commands anywher
root        ALL=(ALL)          ALL
linuxidc    ALL=(ALL)          ALL    #这个是新增的用户
```

[root@localhost ~]# chmod -v u-w /etc/sudoers

[root@localhost ~]# su elk1

3、ElasticSerach

3.1、Java 环境安装

- 解压安装包

[root@localhost jdk1.8]# tar -zxvf jdk-8u171-linux-x64.tar.gz

- 设置 Java 环境变量

[root@localhost jdk1.8.0_171]# vi /etc/profile

在文件最后添加

```
export JAVA_HOME=/home/elk1/jdk1.8/jdk1.8.0_171
export JRE_HOME=$JAVA_HOME/jre
export CLASSPATH=.:$JAVA_HOME/LIB:$JRE_HOME/LIB:$CLASSPATH
export PATH=$JAVA_HOME/bin:$JRE_HOME/bin:$PATH
```

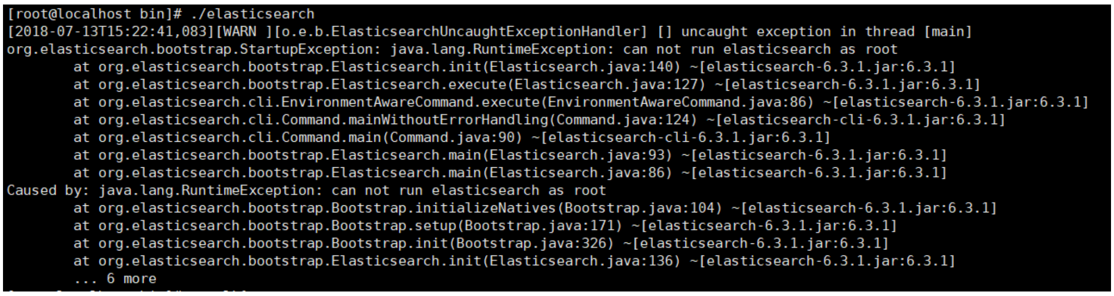
```
[root@localhost jdk1.8.0_171]# source /etc/profile
[root@localhost jdk1.8.0_171]# java -version
    java version "1.8.0_171"
    Java(TM) SE Runtime Environment (build 1.8.0_171-b11)
    Java HotSpot(TM) 64-Bit Server VM (build 25.171-b11, mixed mode)
```

3.2、ElasticSerach 单机安装

[root@localhost elasticserach]# tar -zxvf elasticsearch-6.3.1.tar.gz

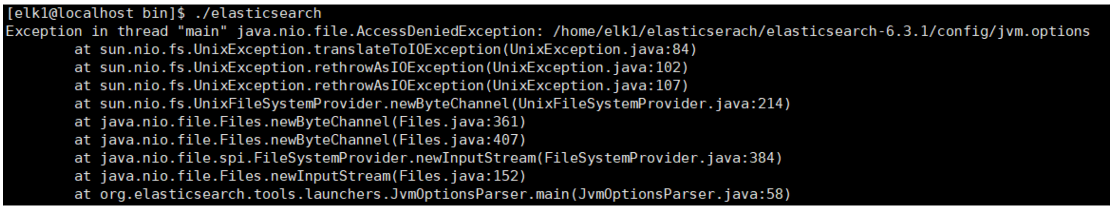
[root@localhost elasticserach]# cd elasticsearch-6.3.1/bin

[root@localhost bin]# ./elasticsearch



```
[root@localhost bin]# ./elasticsearch
[2018-07-13T15:22:41,083][WARN ][o.e.b.ElasticsearchUncaughtExceptionHandler] [] uncaught exception in thread [main]
org.elasticsearch.bootstrap.StartupException: java.lang.RuntimeException: can not run elasticsearch as root
        at org.elasticsearch.bootstrap.Elasticsearch.init(Elasticsearch.java:140) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.bootstrap.Elasticsearch.execute(Elasticsearch.java:127) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.cli.EnvironmentAwareCommand.execute(EnvironmentAwareCommand.java:86) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.cli.Command.mainWithoutErrorHandling(Command.java:124) ~[elasticsearch-cli-6.3.1.jar:6.3.1]
        at org.elasticsearch.cli.Command.main(Command.java:90) ~[elasticsearch-cli-6.3.1.jar:6.3.1]
        at org.elasticsearch.bootstrap.Elasticsearch.main(Elasticsearch.java:93) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.bootstrap.Elasticsearch.main(Elasticsearch.java:86) ~[elasticsearch-6.3.1.jar:6.3.1]
Caused by: java.lang.RuntimeException: can not run elasticsearch as root
        at org.elasticsearch.bootstrap.Bootstrap.initializeNatives(Bootstrap.java:104) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.bootstrap.Bootstrap.setup(Bootstrap.java:171) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.bootstrap.Bootstrap.init(Bootstrap.java:326) ~[elasticsearch-6.3.1.jar:6.3.1]
        at org.elasticsearch.bootstrap.Elasticsearch.init(Elasticsearch.java:136) ~[elasticsearch-6.3.1.jar:6.3.1]
        ... 6 more
```

```
[root@localhost bin]# su elk1
[elk1@localhost bin]$ ./elasticsearch
```

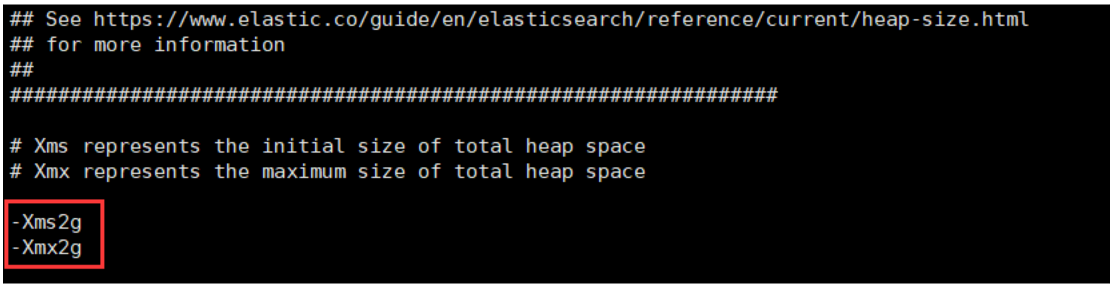


```
[elk1@localhost bin]$ ./elasticsearch
Exception in thread "main" java.nio.file.AccessDeniedException: /home/elk1/elasticserach/elasticsearch-6.3.1/config/jvm.options
        at sun.nio.fs.UnixException.translateToIOException(UnixException.java:84)
        at sun.nio.fs.UnixException.rethrowAsIOException(UnixException.java:102)
        at sun.nio.fs.UnixException.rethrowAsIOException(UnixException.java:107)
        at sun.nio.fs.UnixFileSystemProvider.newByteChannel(UnixFileSystemProvider.java:214)
        at java.nio.file.Files.newByteChannel(Files.java:361)
        at java.nio.file.Files.newByteChannel(Files.java:407)
        at java.nio.file.spi.FileSystemProvider.newInputStream(FileSystemProvider.java:384)
        at java.nio.file.Files.newInputStream(Files.java:152)
        at org.elasticsearch.tools.launchers.JvmOptionsParser.main(JvmOptionsParser.java:58)
```

[root@localhost bin]# chown -R elk1:elk1 /home/elk1/elasticsearch

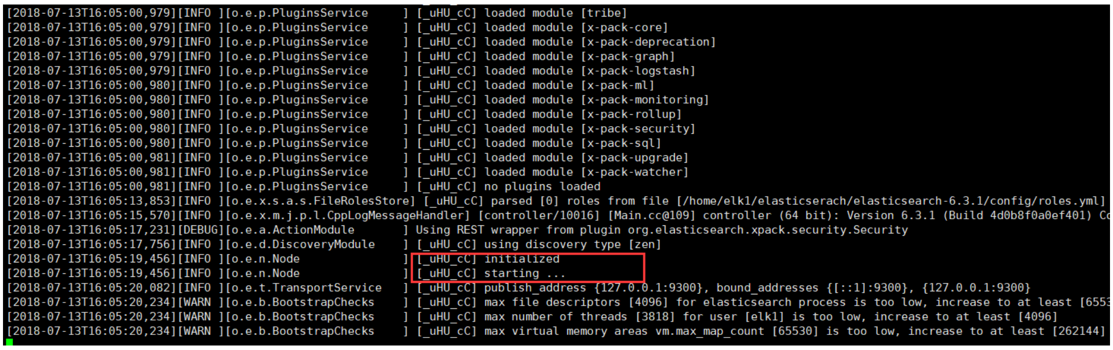
[elk1@localhost bin]$ ./elasticsearch

[elk1@localhost config]$ vi jvm.options



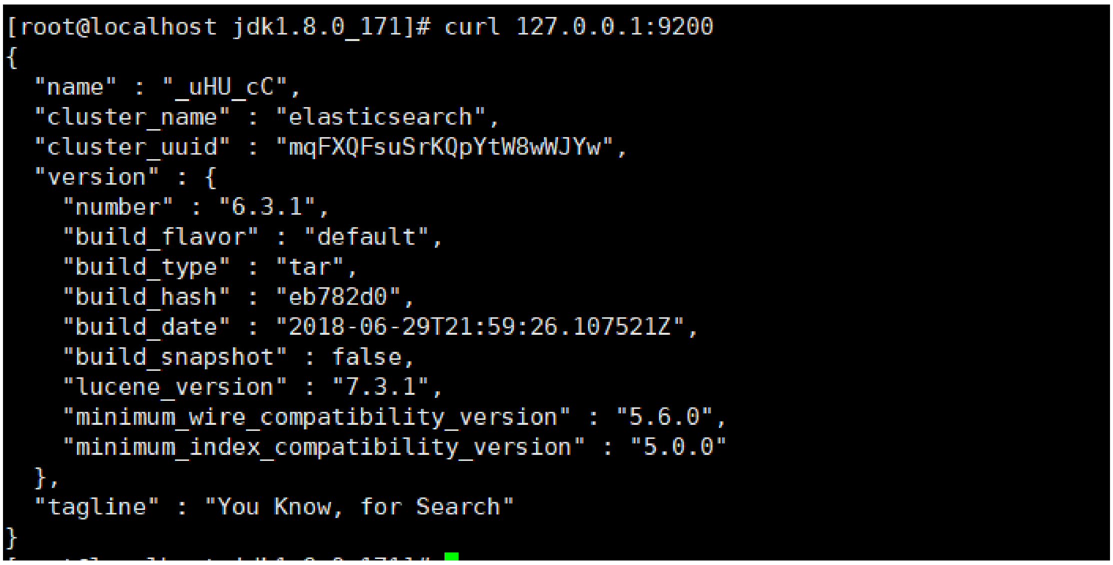
```
## See https://www.elastic.co/guide/en/elasticsearch/reference/current/heap-size.html
## for more information
##
################################################################

# Xms represents the initial size of total heap space
# Xmx represents the maximum size of total heap space

-Xms2g
-Xmx2g
```

[elk1@localhost bin]$ ./elasticsearch



```
[2018-07-13T16:05:00,979][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [tribe]
[2018-07-13T16:05:00,979][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-core]
[2018-07-13T16:05:00,979][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-deprecation]
[2018-07-13T16:05:00,979][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-graph]
[2018-07-13T16:05:00,979][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-logstash]
[2018-07-13T16:05:00,980][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-ml]
[2018-07-13T16:05:00,980][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-monitoring]
[2018-07-13T16:05:00,980][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-rollup]
[2018-07-13T16:05:00,980][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-security]
[2018-07-13T16:05:00,980][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-sql]
[2018-07-13T16:05:00,981][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-upgrade]
[2018-07-13T16:05:00,981][INFO ][o.e.p.PluginsService     ] [_uHU_cC] loaded module [x-pack-watcher]
[2018-07-13T16:05:00,981][INFO ][o.e.p.PluginsService     ] [_uHU_cC] no plugins loaded
[2018-07-13T16:05:13,853][INFO ][o.e.x.s.a.s.FileRolesStore] [_uHU_cC] parsed [0] roles from file [/home/elk1/elasticserach/elasticsearch-6.3.1/config/roles.yml]
[2018-07-13T16:05:15,570][INFO ][o.e.x.m.j.p.l.CppLogMessageHandler] [controller/10016] [Main.cc@109] controller (64 bit): Version 6.3.1 (Build 4d0b8f0a0ef401) C
[2018-07-13T16:05:17,231][DEBUG][o.e.a.ActionModule       ] Using REST wrapper from plugin org.elasticsearch.xpack.security.Security
[2018-07-13T16:05:17,756][INFO ][o.e.d.DiscoveryModule    ] [_uHU_cC] using discovery type [zen]
[2018-07-13T16:05:19,456][INFO ][o.e.n.Node               ] [_uHU_cC] initialized
[2018-07-13T16:05:19,456][INFO ][o.e.n.Node               ] [_uHU_cC] starting ...
[2018-07-13T16:05:20,082][INFO ][o.e.t.TransportService   ] [_uHU_cC] publish_address {127.0.0.1:9300}, bound_addresses {[::1]:9300}, {127.0.0.1:9300}
[2018-07-13T16:05:20,234][WARN ][o.e.b.BootstrapChecks    ] [_uHU_cC] max file descriptors [4096] for elasticsearch process is too low, increase to at least [655
[2018-07-13T16:05:20,234][WARN ][o.e.b.BootstrapChecks    ] [_uHU_cC] max number of threads [3818] for user [elk1] is too low, increase to at least [4096]
[2018-07-13T16:05:20,234][WARN ][o.e.b.BootstrapChecks    ] [_uHU_cC] max virtual memory areas vm.max_map_count [65530] is too low, increase to at least [262144]
```

[root@localhost jdk1.8.0_171]# curl 127.0.0.1:9200



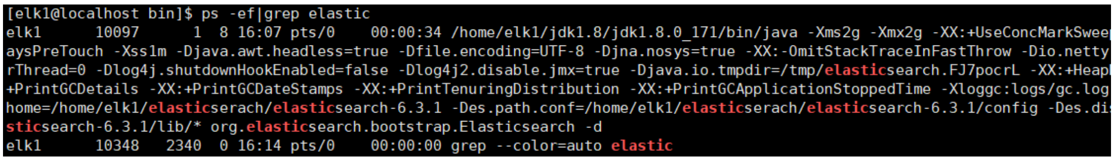
```
[root@localhost jdk1.8.0_171]# curl 127.0.0.1:9200
{
  "name" : "_uHU_cC",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "mqFXQFsuSrKQpYtW8wWJYw",
  "version" : {
    "number" : "6.3.1",
    "build_flavor" : "default",
    "build_type" : "tar",
    "build_hash" : "eb782d0",
    "build_date" : "2018-06-29T21:59:26.107521Z",
    "build_snapshot" : false,
    "lucene_version" : "7.3.1",
    "minimum_wire_compatibility_version" : "5.6.0",
    "minimum_index_compatibility_version" : "5.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

#后台启动

[elk1@localhost bin]$ ./elasticsearch -d

#关闭程序

[elk1@localhost bin]$ ps -ef|grep elastic



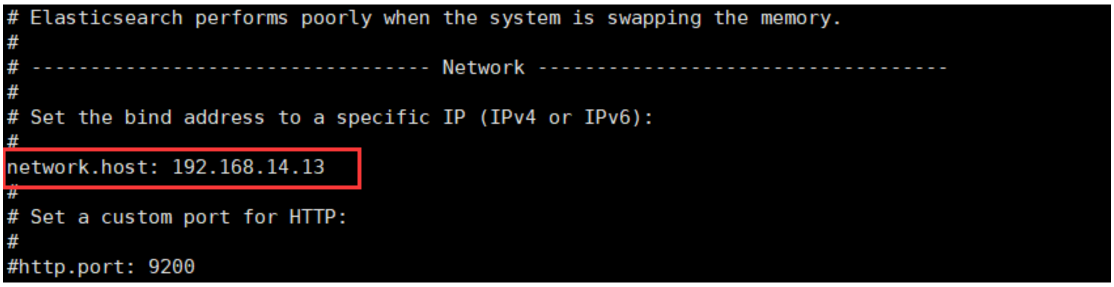
```
[elk1@localhost bin]$ ps -ef|grep elastic
elk1      10097      1  8 16:07 pts/0    00:00:34 /home/elk1/jdk1.8/jdk1.8.0_171/bin/java -Xms2g -Xmx2g -XX:+UseConcMarkSwee
aysPreTouch -Xss1m -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djna.nosys=true -XX:-OmitStackTraceInFastThrow -Dio.netty
rThread=0 -Dlog4j.shutdownHookEnabled=false -Dlog4j2.disable.jmx=true -Djava.io.tmpdir=/tmp/elasticsearch.FJ7pocrL -XX:+Heap
+PrintGCDetails -XX:+PrintGCDateStamps -XX:+PrintTenuringDistribution -XX:+PrintGCApplicationStoppedTime -Xloggc:logs/gc.log
home=/home/elk1/elasticserach/elasticsearch-6.3.1 -Des.path.conf=/home/elk1/elasticserach/elasticsearch-6.3.1/config -Des.di
sticsearch-6.3.1/lib/* org.elasticsearch.bootstrap.Elasticsearch -d
elk1      10348   2340  0 16:14 pts/0    00:00:00 grep --color=auto elastic
```

[elk1@localhost bin]$ kill 10097

#设置浏览器访问

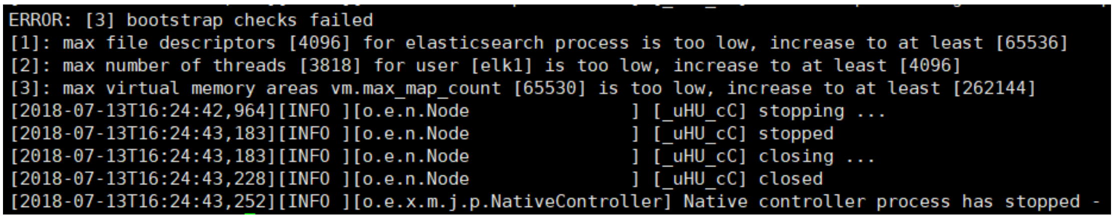
[root@localhost bin]systemctl stop firewalld

[root@localhost bin]vi config/elasticsearch.yml



```
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ---------------------------------- Network -----------------------------------
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: 192.168.14.13
#
# Set a custom port for HTTP:
#
#http.port: 9200
```

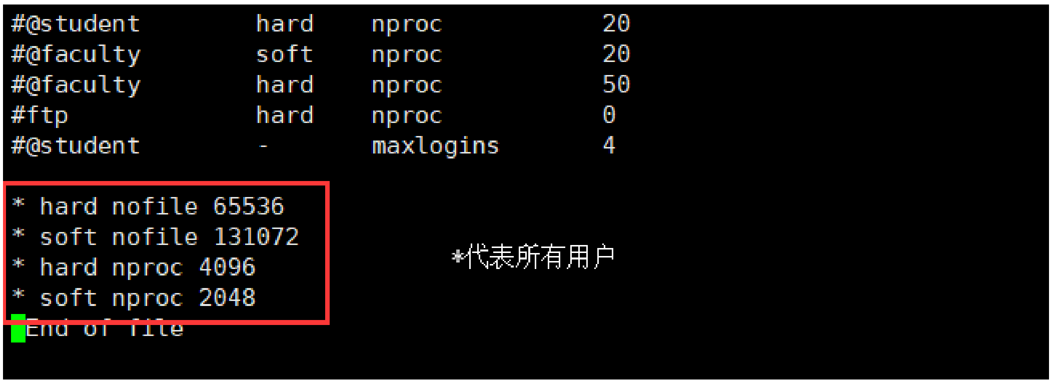
安装问题：



```
ERROR: [3] bootstrap checks failed
[1]: max file descriptors [4096] for elasticsearch process is too low, increase to at least [65536]
[2]: max number of threads [3818] for user [elk1] is too low, increase to at least [4096]
[3]: max virtual memory areas vm.max_map_count [65530] is too low, increase to at least [262144]
[2018-07-13T16:24:42,964][INFO ][o.e.n.Node               ] [_uHU_cC] stopping ...
[2018-07-13T16:24:43,183][INFO ][o.e.n.Node               ] [_uHU_cC] stopped
[2018-07-13T16:24:43,183][INFO ][o.e.n.Node               ] [_uHU_cC] closing ...
[2018-07-13T16:24:43,228][INFO ][o.e.n.Node               ] [_uHU_cC] closed
[2018-07-13T16:24:43,252][INFO ][o.e.x.m.j.p.NativeController] Native controller process has stopped -
```

[1] [2]解决方案

[root@localhost bin]# vi /etc/security/limits.conf

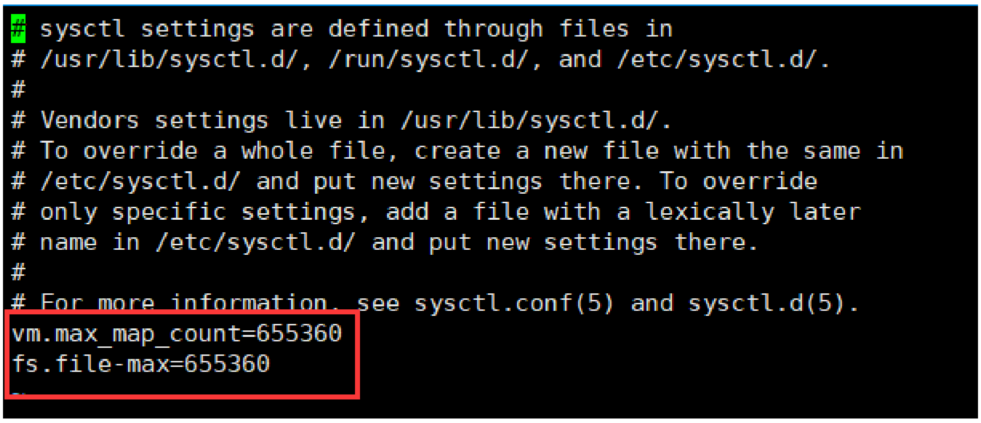


```
#@student        hard    nproc           20
#@faculty        soft    nproc           20
#@faculty        hard    nproc           50
#ftp             hard    nproc           0
#@student        -       maxlogins       4

* hard nofile 65536
* soft nofile 131072
* hard nproc 4096
* soft nproc 2048
End of file
```

*代表所有用户

[3] 解决方案

```
[root@localhost bin]# vi /etc/sysctl.conf
[root@localhost bin]# sysctl -p
```



```
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
#
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
#
# For more information, see sysctl.conf(5) and sysctl.d(5).
vm.max_map_count=655360
fs.file-max=655360
```

3.3、ElasticSerach 集群安装

- 修改配置文件 elasticserach.yml
  vim /elasticsearch.yml

```
cluster.name: aubin-cluster#必须相同
# 集群名称（不能重复）
node.name: els1（必须不同）
# 节点名称，仅仅是描述名称，用于在日志中区分（自定义）
#指定了该节点可能成为 master 节点，还可以是数据节点
node.master: true
node.data: true
path.data: /var/lib/elasticsearch
# 数据的默认存放路径（自定义）
path.logs: /var/log/elasticsearch
# 日志的默认存放路径
network.host: 192.168.0.1
# 当前节点的 IP 地址
http.port: 9200
# 对外提供服务的端口
transport.tcp.port: 9300
#9300 为集群服务的端口
discovery.zen.ping.unicast.hosts: ["172.18.68.11", "172.18.68.12","172.18.68.13"]
# 集群个节点 IP 地址，也可以使用域名，需要各节点能够解析
discovery.zen.minimum_master_nodes: 2
# 为了避免脑裂，集群节点数最少为 半数+1
```

注意：清空 data 和 logs 数据

192.168.14.12:9200/_cat/nodes?v

3.4、安装 head 插件

- 下载 head 插件

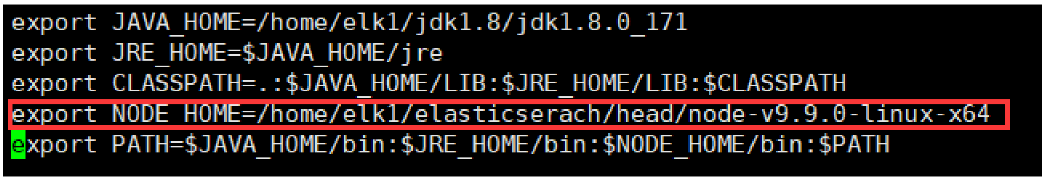
  `wget https://github.com/mobz/elasticsearch-head/archive/elasticsearch-head-master.zip`

  也可以用 git 下载，前提 yum install git

  unzip elasticsearch-head-master.zip
- 安装 node.js

  `wget https://npm.taobao.org/mirrors/node/latest-v4.x/node-v4.4.7-linux-x64.tar.gz`

  tar -zxvf   node-v9.9.0-linux-x64.tar.gz
- 添加 node.js 到环境变量



```
export JAVA_HOME=/home/elk1/jdk1.8/jdk1.8.0_171
export JRE_HOME=$JAVA_HOME/jre
export CLASSPATH=.:$JAVA_HOME/LIB:$JRE_HOME/LIB:$CLASSPATH
export NODE_HOME=/home/elk1/elasticserach/head/node-v9.9.0-linux-x64
export PATH=$JAVA_HOME/bin:$JRE_HOME/bin:$NODE_HOME/bin:$PATH
```

  source /etc/profile
- 测试

  node -v

  npm -v
- 安装 grunt（grunt 是一个很方便的构建工具，可以进行打包压缩、测试、执行等等的工作）

  进入到 elasticsearch-head-master

  npm install -g grunt-cli

  npm install

  (npm install -g cnpm --registry=https://registry.npm.taobao.org)

- 修改 Elasticsearch 配置文件

  编辑 elasticsearch-6.3.1/config/elasticsearch.yml,加入以下内容：

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

- 修改 Gruntfile.js（注意’，’）

打开 elasticsearch-head-master/Gruntfile.js，找到下面 connect 属性，新增 hostname:’*’:

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

- 启动 elasticsearch-head

  进入 elasticsearch-head 目录，执行命令：grunt server
- 后台启动 elasticsearch-head

  nohup grunt server &exit
- 关闭 head 插件

  ps -aux|grep head

  kill 进程号

3.5、ElasticSerach API

- elasticsearch rest api 遵循的格式为：

  curl -X<REST Verb> <Node>:<Port>/<Index>/<Type>/<ID>
- 检查 es 版本信息

  curl IP:9200
- 查看集群是否健康

  http://IP:9200/_cat/health?v
- 查看节点列表

  http://IP:9200/_cat/nodes?v
- 列出所有索引及存储大小

  http://IP:9200/_cat/indices?v
- 创建索引

  curl -XPUT 'IP:9200/XX?pretty'
- 添加一个类型

  ```
  curl -XPUT 'IP:9200/XX/external/2?pretty' -d '
  {
    "gwyy": "John"
  }'
  ```
- 更新一个类型

  ```
  curl -XPOST 'IP:9200/XX/external/1/_update?pretty' -d '
  {
    "doc": {"name": "Jaf"}
  }'
  ```
- 删除指定索引

  curl -XDELETE 'IP:9200/_index?pretty'

3.6、配置详情

- ElasticSearch.yml
  ES 的相关配置

```
# 集群的名字，以此作为是否同一集群的判断条件
cluster.name: elasticsearch
# 节点名字，以此作为集群中不同节点的区分条件
node.name: node-1
#设置当前节点既可以为主节点也可以为数据节点
node.master: true
node.data: true
# 索引分片个数，默认为 5 片
#index.number_of_shards: 5
# 索引副本个数，默认为 1 个副本
#index.number_of_replicas: 1
# 数据存储目录（多个路径用逗号分隔）
discovery.zen.ping.unicast.hosts: ["192.168.14.14","192.168.14.15"]
discovery.zen.minimum_master_nodes: 2
#数据目录
path.data: /home/elk1/elasticserach/data
# 日志目录
path.logs: /home/elk1/elasticserach/logs
# 修改一下 ES 的监听地址，这样别的机器才可以访问
network.host: 192.168.14.13
# 设置节点间交互的 tcp 端口（集群）,默认是 9300
transport.tcp.port: 9300
# 监听端口（默认的就好）
http.port: 9200
# 增加新的参数，这样 head 插件才可以访问 es
http.cors.enabled: true
http.cors.allow-origin: "*"
```

- Jvm.options
  JVM 的相关配置

- Log4j2.properties
  日志相关配置

3.7、Elasticserach 模式

- 分为 Development 和 Production 两种模式
  - 区分方式

以 transport 的地址是否绑定在 localhost 为标准（实际地址）
即：elasticserach.yml 文件中的 network.host 配置

- 模式区别

（1）Development 模式下启动时会以 warning 的方式提示配置检查异常
（2）Production 模式下在启动时会以 error 的方式提示配置检查异常并推出

3.8、elasticserach 操作

- 基本概念
  - ■ Document:文档对象
  - ■ Index:索引（库）
  - ■ Type:索引中的数据类型（表）
  - ■ Field:字段，文档的属性（字段）
  - ■ Query DSL:查询语法（sql）
- CRUD 操作
  - ■ 创建文档

请求：

```
POST /atguigu/student/1
{
	"name":"zhangsan",
	"clazz":"0115bigdata",
	"description":"we are family"
}
```

返回：

```
{
  "_index": "atguigu",
  "_type": "student",
  "_id": "1",
  "_version": 1,
  "result": "created",
  "_shards": {
    "total": 2,
    "successful": 2,
    "failed": 0
  },
  "_seq_no": 0,
  "_primary_term": 1
}
```

  - ■ 获取文档

请求：

```
GET atguigu/student/1
```

返回：

```
{
  "_index": "atguigu",
  "_type": "student",
  "_id": "1",
  "_version": 1,
  "found": true,
  "_source": {
    "name": "zhangsan",
    "clazz": "0115bigdata",
    "description": "we are family"
  }
}
```

- 更新文档

请求：

```
POST /atguigu/student/1/_update
{
    "doc":{
        "description":"hello world"
    }
}
```

返回：

```
{
  "_index": "atguigu",
  "_type": "student",
  "_id": "1",
  "_version": 2,
  "result": "updated",
  "_shards": {
    "total": 2,
    "successful": 2,
    "failed": 0
  },
  "_seq_no": 1,
  "_primary_term": 1
}
```

- 删除文档

请求：

```
DELETE atguigu/student/1
```

查询结果：

```
{
  "_index": "atguigu",
  "_type": "student",
  "_id": "1",
  "found": false
}
```

- Elasticserach Query
    - Query String

      GET /atguigu/student/_sea'rch?q=关键字

      返回：

      ```
      {
        "took": 8,
        "timed_out": false,
        "_shards": {
          "total": 5,
          "successful": 5,
          "skipped": 0,
          "failed": 0
        },
        "hits": {
          "total": 1,
          "max_score": 0.2876821,
          "hits": [
            {
              "_index": "atguigu",
              "_type": "student",
              "_id": "1",
              "_score": 0.2876821,
              "_source": {
                "name": "zhangsan",
                "clazz": "0115bigdata",
                "description": "we are family"
              }
            }
          ]
        }
      }
      ```

    - Query DSL

      ```
      GET atguigu/student/_search
      {
        "query":{
          "term":{
            "name":{
                "value":"zhangsan"
      ```

```
                }
              }
            }
          }
```

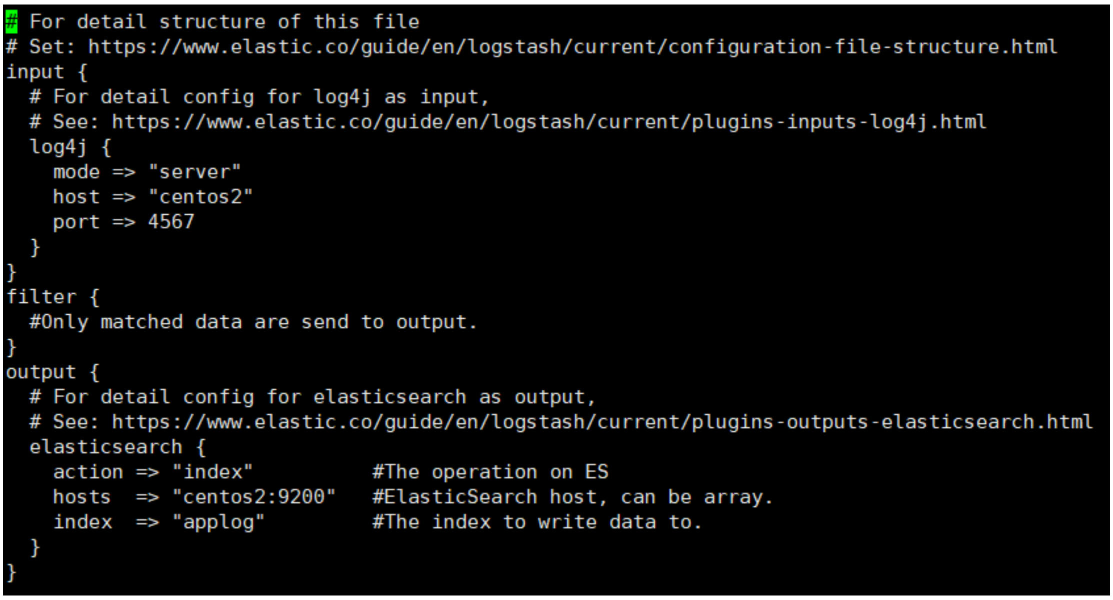
4、Logstash

4.1、安装 logstash

[root@localhost logstash]# tar -zxvf logstash-6.3.1.tar.gz

[root@localhost logstash-6.3.1]# cd config

[root@localhost config]# vi log4j_to_es.conf



```
# For detail structure of this file
# Set: https://www.elastic.co/guide/en/logstash/current/configuration-file-structure.html
input {
  # For detail config for log4j as input,
  # See: https://www.elastic.co/guide/en/logstash/current/plugins-inputs-log4j.html
  log4j {
    mode => "server"
    host => "centos2"
    port => 4567
  }
}
filter {
  #Only matched data are send to output.
}
output {
  # For detail config for elasticsearch as output,
  # See: https://www.elastic.co/guide/en/logstash/current/plugins-outputs-elasticsearch.html
  elasticsearch {
    action => "index"          #The operation on ES
    hosts  => "centos2:9200"   #ElasticSearch host, can be array.
    index  => "applog"         #The index to write data to.
  }
}
```

```
input {
        file {
                path=>[""]
                type=>""
                start_position=>"beginning"
        }
}
output {
        stdout {
                codec=>rubydebug
        }
}
```

[root@localhost logstash-6.3.1]# ./bin/logstash -f config/log4j_to_es.conf

4.2、输入、输出、过滤

- 输入
  input{file{path=>”/tomcat/logs/abc.log”}}
- 输出
  output{stdout{codec=>rubydebug}}
- 过滤插件
  - Grok
    1、基于正则表达式提供了丰富可重用的模式（pattern）
    2、基于此可以将非结构化数据作结构化处理
  - Date
    将字符串类型的时间字段转换为时间戳类型，方便后续数据处理
  - Mutate
    进行增加、修改、删除、替换等字段相关处理

4.3、logstash 格式化 nginx 日志内容

- 创建 nginx_logstash.conf 文件

```
input {
  stdin { }
}

filter {
  grok {
    match => {
      "message" => '%{IPORHOST:remote_ip} - %{DATA:user_name} \[%{HTTPDATE:time}\]
"%{WORD:request_action} %{DATA:request}
HTTP/%{NUMBER:http_version}" %{NUMBER:response} %{NUMBER:bytes} "%{DATA:referrer}"
"%{DATA:agent}"'
    }
  }

  date {
    match => [ "time", "dd/MMM/YYYY:HH:mm:ss Z" ]
    locale => en
  }

  geoip {
    source => "remote_ip"
    target => "geoip"
  }
  useragent {
    source => "agent"
    target => "user_agent"
  }
}

output {
stdout {
 codec => rubydebug
 }
}
```

- Logstash 启动解析 nginx 文件

```
head -n 2 /home/elk1/nginx_logs|./logstash -f ../config/nginx_logstash.conf
```

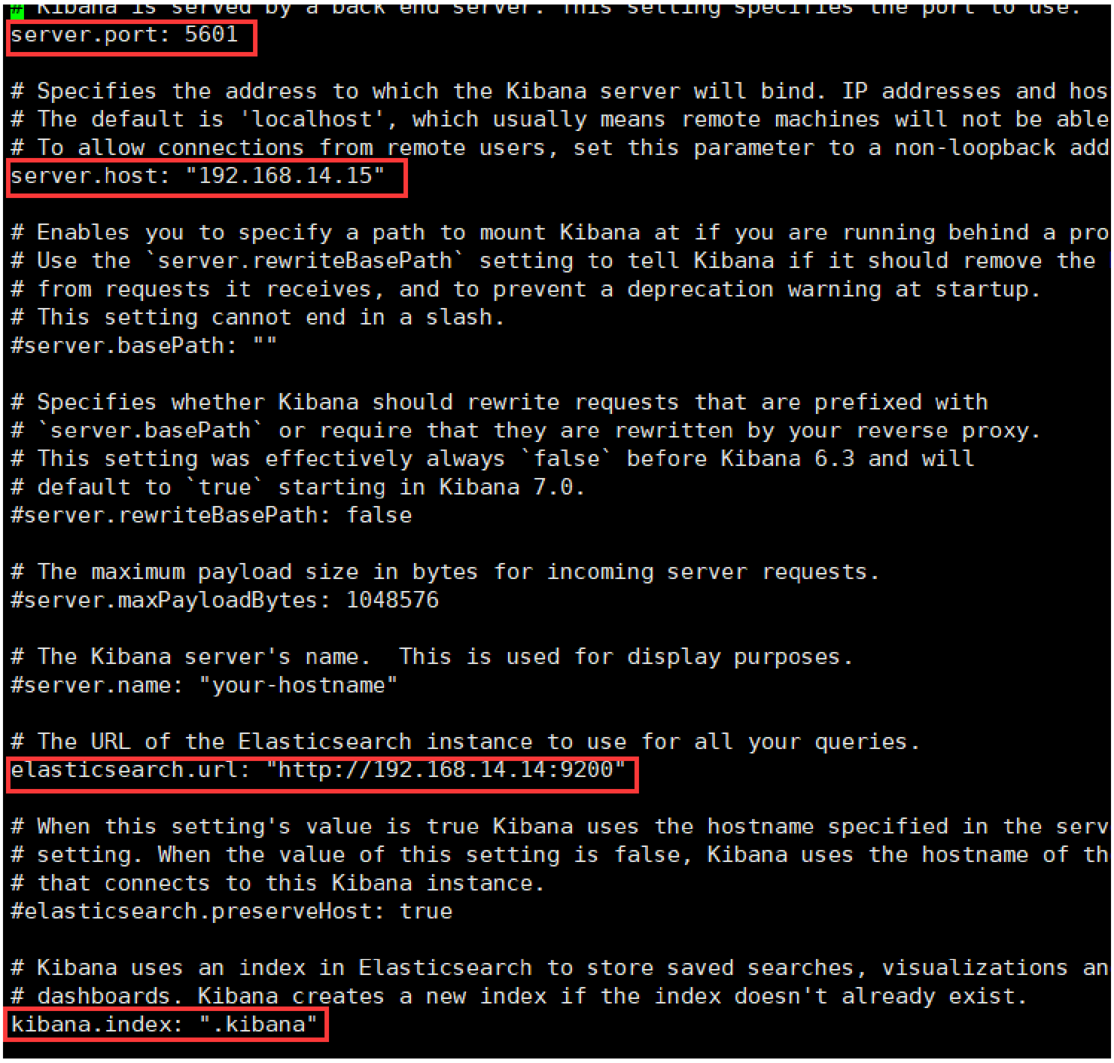
- 结果

```
{
         "user_name" => "-",
          "referrer" => "-",
        "@timestamp" => 2015-05-17T08:05:32.000Z,
           "request" => "/downloads/product_1",
              "time" => "17/May/2015:08:05:32 +0000",
             "geoip" => {
         "country_code3" => "NL",
             "longitude" => 4.8995,
        "continent_code" => "EU",
              "latitude" => 52.3824,
              "timezone" => "Europe/Amsterdam",
         "country_code2" => "NL",
                    "ip" => "93.180.71.3",
          "country_name" => "Netherlands",
              "location" => {
            "lat" => 52.3824,
            "lon" => 4.8995
        }
    },
          "@version" => "1",
      "http_version" => "1.1",
         "remote_ip" => "93.180.71.3",
           "message" => "93.180.71.3 - - [17/May/2015:08:05:32 +0000] \"GET
/downloads/product_1 HTTP/1.1\" 304 0 \"-\" \"Debian APT-HTTP/1.3
(0.8.16~exp12ubuntu10.21)\"",
             "bytes" => "0",
        "user_agent" => {
          "minor" => "3",
             "os" => "Debian",
           "name" => "Debian APT-HTTP",
        "os_name" => "Debian",
          "build" => "",
          "major" => "1",
         "device" => "Other"
    },
             "agent" => "Debian APT-HTTP/1.3 (0.8.16~exp12ubuntu10.21)",
              "host" => "localhost.localdomain",
          "response" => "304",
    "request_action" => "GET"
}
```

5、Kibana

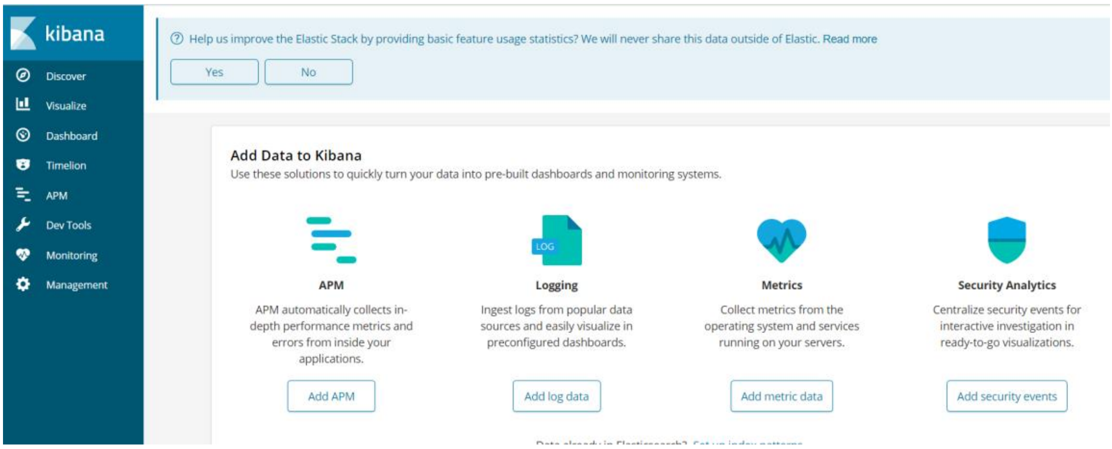
5.1、Kibana 安装

```
[root@localhost kibana]# tar -zxvf kibana-6.3.1-linux-x86_64.tar.gz
[root@localhost kibana]# cd kibana-6.3.1-linux-x86_64/config
[root@localhost config]# vi kibana.yml
```



```
# 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 hos
# The default is 'localhost', which usually means remote machines will not be able
# To allow connections from remote users, set this parameter to a non-loopback add
server.host: "192.168.14.15"

# Enables you to specify a path to mount Kibana at if you are running behind a pro
# Use the `server.rewriteBasePath` setting to tell Kibana if it should remove the 
# 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://192.168.14.14:9200"

# When this setting's value is true Kibana uses the hostname specified in the serv
# setting. When the value of this setting is false, Kibana uses the hostname of th
# that connects to this Kibana instance.
#elasticsearch.preserveHost: true

# Kibana uses an index in Elasticsearch to store saved searches, visualizations an
# dashboards. Kibana creates a new index if the index doesn't already exist.
kibana.index: ".kibana"
```

[root@localhost bin]# ./kibana



Help us improve the Elastic Stack by providing basic feature usage statistics? We will never share this data outside of Elastic. Read more

Add Data to Kibana

Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Logging

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Metrics

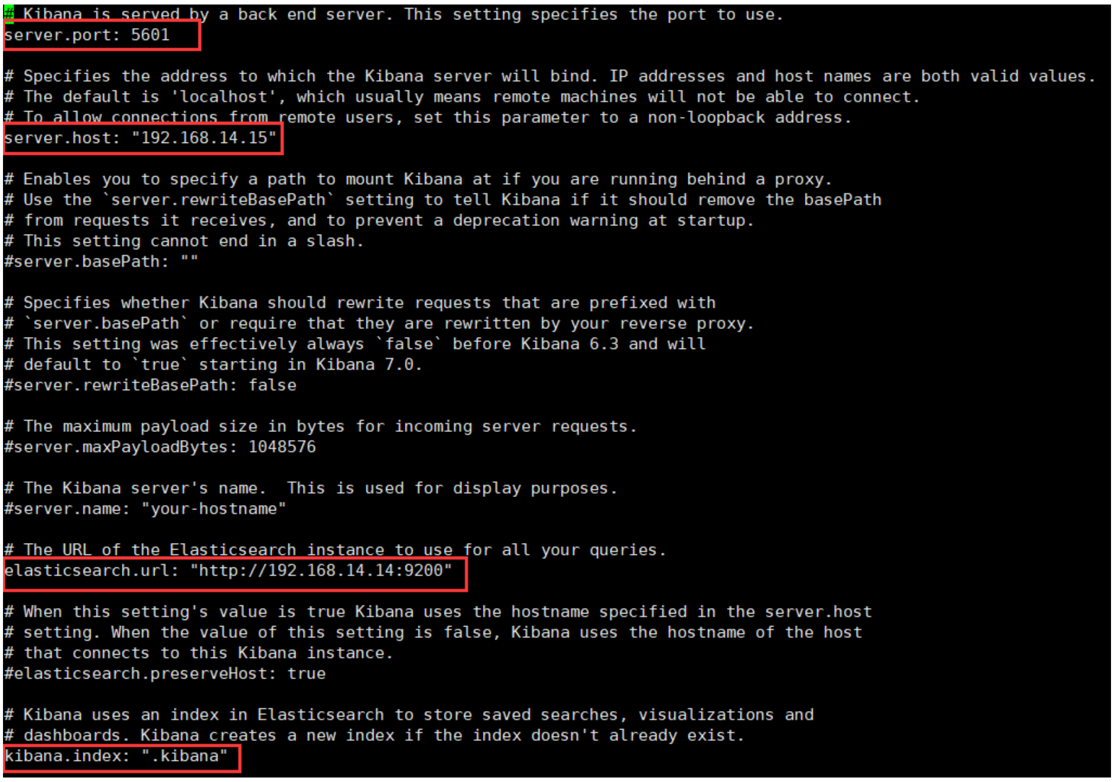
Collect metrics from the operating system and services running on your servers.

Security Analytics

Centralize security events for interactive investigation in ready-to-go visualizations.

5.2、kibana 配置

- 配置文件在 config 文件夹下
- Kibana.yml 常用配置说明

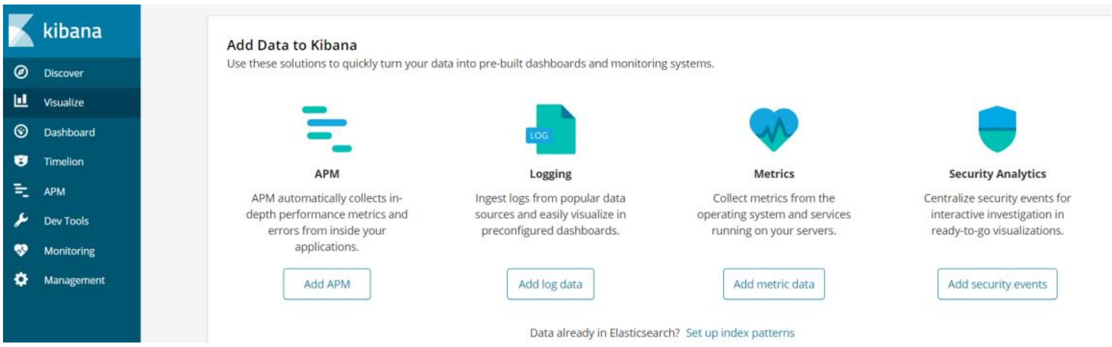


```
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: 5601

# 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: "192.168.14.15"

# 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://192.168.14.14:9200"

# 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: ".kibana"
```

Server.host/server.port:访问的端口号和地址(地址设置后才能被外网访问)

Elasticsearch.url:访问 elasticserach 的地址

5.3、kibana 功能简介



Add Data to Kibana

Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Logging

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Metrics

Collect metrics from the operating system and services running on your servers.

Security Analytics

Centralize security events for interactive investigation in ready-to-go visualizations.

Data already in Elasticsearch? Set up index patterns

Discover:数据搜索查看

Visualize:图标制作

Dashboard:仪表盘制作

Timeline:时序数据的高级可视化分析

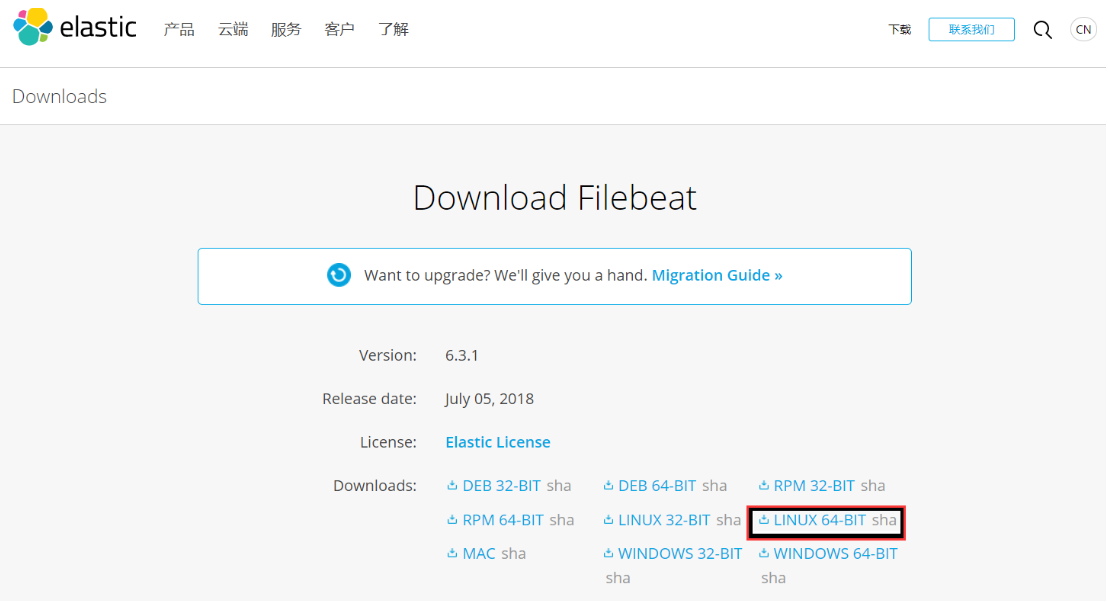
DevTools:开发者工具

Management:kibana 相关配置

6、Filebeat 和 packetbeat

2.1、Filebeat

- 下载 Filebeat
  https://www.elastic.co/cn/downloads/beats/filebeat
  查看系统位数：getconf LONG_BIT



Download Filebeat

Want to upgrade? We'll give you a hand. Migration Guide »

Version: 6.3.1

Release date: July 05, 2018

License: Elastic License

Downloads: DEB 32-BIT sha DEB 64-BIT sha RPM 32-BIT sha RPM 64-BIT sha LINUX 32-BIT sha LINUX 64-BIT sha MAC sha WINDOWS 32-BIT sha WINDOWS 64-BIT sha

2.2、Packetbeat

- Packetbeat 简介
  （1）实时抓取网络包
  （2）自动解析应用层协议（抓包）
  DNS、Http、Redis、Mysql 等
- Packetbeat 抓取 elasticserach 请求数据
  （1）进入 packetbeat 目录，创建 es.yml 文件
  （2）编辑 es.yml 文件

```
packetbeat.interfaces.device: ens33#网卡

packetbeat.protocols.http:
    ports: [9200]#es 端口
    send_request: true#抓取请求信息
    include_body_for: ["application/json", "x-www-form-urlencoded"]#包含内容
output.console:
    pretty: true#控制台输出
```

（3）启动 packetbeat

```
sudo ./packetbeat -e -c es.yml -strict.perms=false
```

7、Nginx

- 安装 nginx

```
#安装依赖环境
yum install gcc-c++
yum install pcre-devel
yum install zlib zlib-devel
yum install openssl openssl-deve
#//一键安装上面四个依赖
#yum -y install gcc zlib zlib-devel pcre-devel openssl openssl-devel

#解压
tar -xvf nginx-1.13.7.tar.gz

#进入 nginx 目录
cd /usr/local/nginx    #执行命令

./configure

#执行 make 命令 make//执行 make install 命令
make
make install
//启动命令
nginx/sbin/nginx
//停止命令
nginx/sbin/nginx -s stop 或者 : nginx -s quit
//重启命令
nginx -s reload
```

8、数据可视化演示实战

8.1、实战说明

- 需求：
  收集 Elasticserach 集群的查询语句
  分析查询语句的常用语句、响应时长等
- 方案
  数据收集：Packetbeat+logstash
  数据分析：Kibana+Elasticsearch

8.2、前期准备

- Production Cluster(生产环境)

  1、Elasticsearch 192.168.14.13:9200

  2、Kibana 192.168.14.15:5601
- Monitoring Cluster(监控环境)

  1、Elasticsearch 192.168.14.16:8200

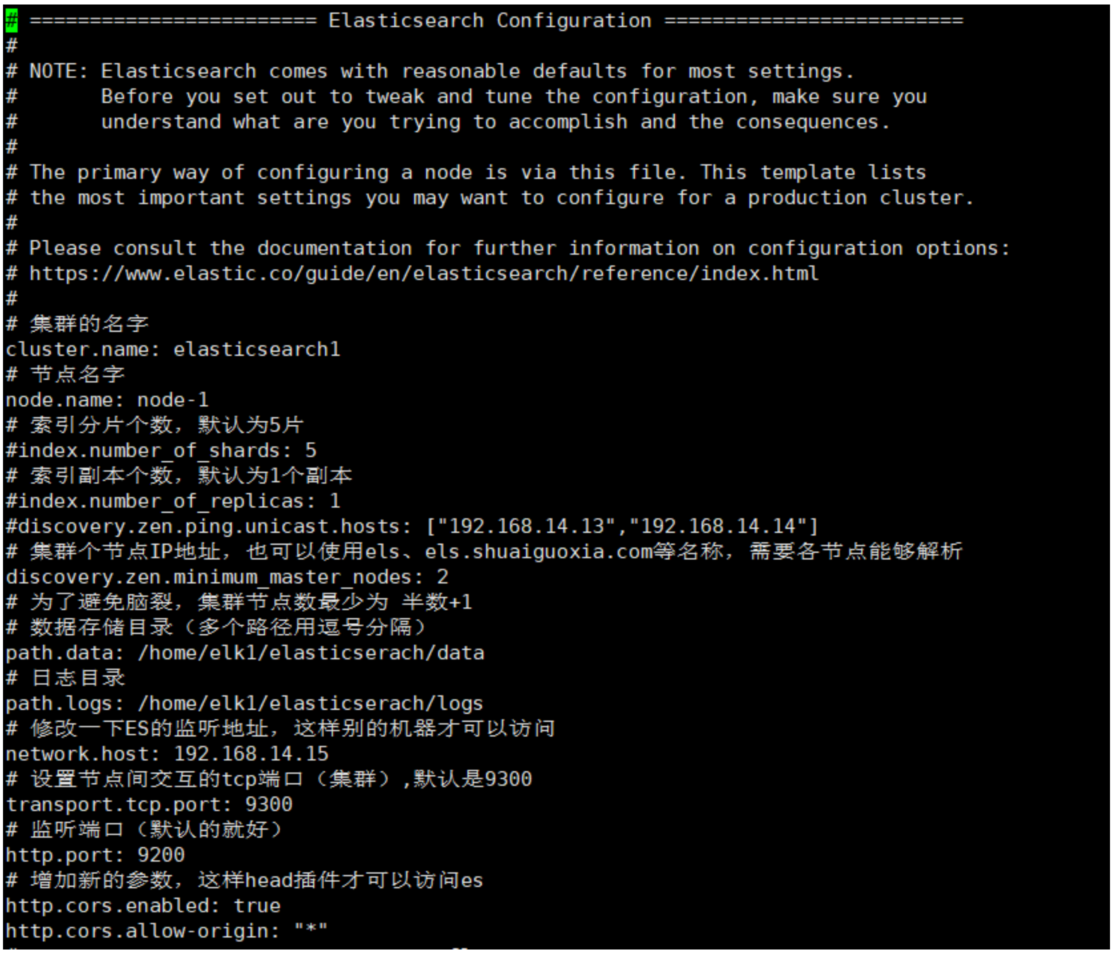
  2、Kibana 192.168.14.16:8601
- Logstash\packetbeat

8.3、实战

- 启动数据采集集群

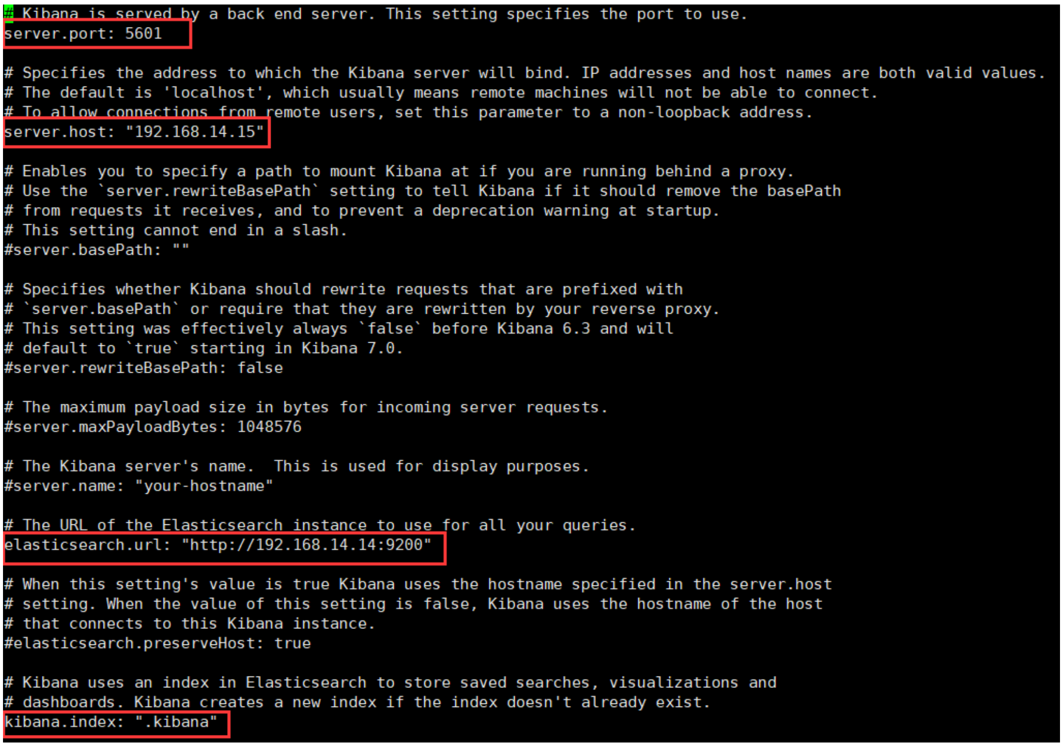
  启动 ES：

  ./elasticsearch



```
# ======================== 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
#
# 集群的名字
cluster.name: elasticsearch1
# 节点名字
node.name: node-1
# 索引分片个数，默认为5片
#index.number_of_shards: 5
# 索引副本个数，默认为1个副本
#index.number_of_replicas: 1
#discovery.zen.ping.unicast.hosts: ["192.168.14.13","192.168.14.14"]
# 集群个节点IP地址，也可以使用els、els.shuaiguoxia.com等名称，需要各节点能够解析
discovery.zen.minimum_master_nodes: 2
# 为了避免脑裂，集群节点数最少为 半数+1
# 数据存储目录（多个路径用逗号分隔）
path.data: /home/elk1/elasticserach/data
# 日志目录
path.logs: /home/elk1/elasticserach/logs
# 修改一下ES的监听地址，这样别的机器才可以访问
network.host: 192.168.14.15
# 设置节点间交互的tcp端口（集群），默认是9300
transport.tcp.port: 9300
# 监听端口（默认的就好）
http.port: 9200
# 增加新的参数，这样head插件才可以访问es
http.cors.enabled: true
http.cors.allow-origin: "*"
```

修改 kibana 配置



```
# 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: "192.168.14.15"

# 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://192.168.14.14:9200"

# 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: ".kibana"
```

./kibana　　#启动

- 启动数据分析集群

（1）启动 ES

同上

（2）启动 logstash

```
    input {
        beats {
            port => 5044
        }
    }
    filter {
        if "search" in [request]{
            grok {
                match => { "request" => ".*\n\{(?<query_body>.*)"}
            }
            grok {
                match => { "path" => "\/(?<index>.*)\/_search"}
            }
         if [index] {
          } else {
                mutate {
                  add_field   => { "index" => "All" }
            }
          }


          mutate {
```

```
                        update    => { "query_body" => "{%{query_body}"}}
            }

    #       mutate {
    #               remove_field => [ "[http][response][body]" ]
    #       }
}

output {
    #stdout{codec=>rubydebug}

    if "search" in [request]{
            elasticsearch {
            hosts => "127.0.0.1:9200"
            }
     }
}
```

（3）启动

./bin/logstash -f config/log4j_to_es.conf

附录：防火墙配置

1、firewalld 的基本使用

启动： systemctl start firewalld

关闭： systemctl stop firewalld

查看状态： systemctl status firewalld

开机禁用 ： systemctl disable firewalld

开机启用 ： systemctl enable firewalld

2.systemctl 是 CentOS7 的服务管理工具中主要的工具，它融合之前 service 和 chkconfig 的功能于一体。

启动一个服务：systemctl start firewalld.service

关闭一个服务：systemctl stop firewalld.service

重启一个服务：systemctl restart firewalld.service

显示一个服务的状态：systemctl status firewalld.service

在开机时启用一个服务：systemctl enable firewalld.service

在开机时禁用一个服务：systemctl disable firewalld.service

查看服务是否开机启动：systemctl is-enabled firewalld.service

查看已启动的服务列表：systemctl list-unit-files|grep enabled

查看启动失败的服务列表：systemctl --failed

3.配置 firewalld-cmd

查看版本： firewall-cmd --version

查看帮助： firewall-cmd --help

显示状态： firewall-cmd --state

查看所有打开的端口： firewall-cmd --zone=public --list-ports

更新防火墙规则： firewall-cmd --reload

查看区域信息: firewall-cmd --get-active-zones

查看指定接口所属区域： firewall-cmd --get-zone-of-interface=eth0

拒绝所有包：firewall-cmd --panic-on

取消拒绝状态： firewall-cmd --panic-off

查看是否拒绝： firewall-cmd --query-panic

4.那怎么开启一个端口呢

添加

firewall-cmd --zone=public --add-port=80/tcp --permanent （--permanent 永久生效，没有此参数重启后失效）

重新载入

firewall-cmd --reload

查看

firewall-cmd --zone= public --query-port=80/tcp

删除

firewall-cmd --zone= public --remove-port=80/tcp --permanent