ELK日志采集分析搭建

## JDK安装与配置

1. 确认系统是否已有JDK 1.8以上的环境 java –version,没有则重新安装
2. 在/usr目录下新建java目录
3. 下载jdk 1.8以上压缩包
4. 解压 tar -zxvf jdk-8u161-linux-x64.tar.gz
5. 配置环境变量 vi /etc/profile

向文件里面追加以下内容：

JAVA\_HOME=/usr/java/jdk1.8.0\_161

JRE\_HOME=/usr/java/jdk1.8.0\_161/jre

PATH=$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin

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

export JAVA\_HOME JRE\_HOME PATH CLASSPATH

1. 刷新配置文件 source /etc/profile
2. 测试 java –version javac –version

## ElasticSearch安装与配置

1. 下载elasticsearch压缩包 wget <https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-6.1.2.tar.gz>
2. 解压缩包 tar –zxvf elasticsearch-6.1.2.tar.gz
3. 新增es专用用户 useradd es –g es –p 123456
4. 将elasticsearch目录赋给es chown -R es:es elasticsearch-6.1.0
5. 使用es用户操作 su es
6. 进入elasticsearch目录 cd elasticsearch-6.1.2
7. 启动elasticsearch bin/elasticsearch
8. 查看启动无报错后测试 curl localhost:9200
9. 推出 crtl+c
10. 修改系统配置避免闪退
11. 切换root用户 su – root
12. 编辑配置文件 vi /etc/sysctl.conf

添加 vm.max\_map\_count=262144

1. 修改配置文件 vi /etc/security/limits.conf

在最后部分添加内容：

\* hard nofile 65536

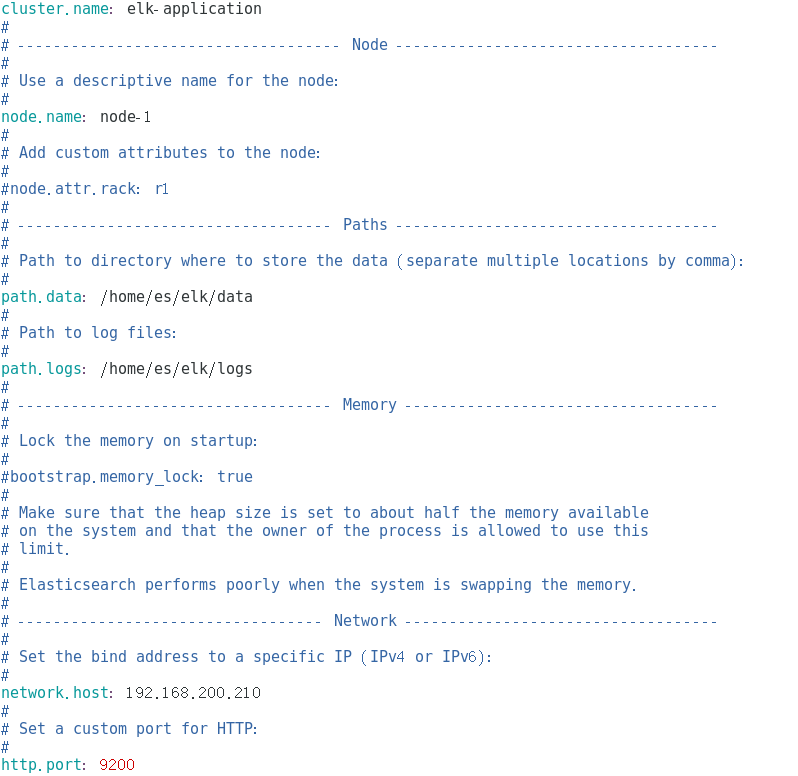
\* soft nofile 65536

\* soft nproc 2048

\* hard nproc 4096

# End of file

1. Sysctl –p 查看是否生效
2. 修改elastic配置文件 vi config/elasticsearch.yml

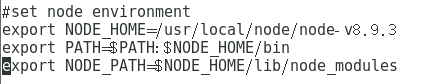


1. 重启elasticsearch bin/elasticsearch –d（后台运行模式）
2. 使用 jps命令查看是否有elasticsearch进程
3. 验证启动 curl 192.168.200.210:9200
4. 使用浏览器打开 192.168.200.210:9200

## ElasticSearch head插件安装

Head插件用于监视es集群节点运行状态及索引的查看

### Head插件运行环境nodejs安装

1. 切换用户为root
2. 下载nodejs wget https://npm.taobao.org/mirrors/node/v8.9.3/node-v8.9.3-linux-x64.tar.gz
3. 解压node.js tar –zxvf node-v8.9.3-linux-x64.tar.gz
4. 配置环境变量 vi /etc/profile 
5. 刷新环境变量 source /etc/profile
6. 安装grunt npm install -g grunt

npm install -g grunt-cli --registry=https://registry.npm.taobao.org --no-proxy

1. 版本确认 node –v nmp –v grunt –version

### Head插件安装

1. 下载head插件包 wget <https://github.com/mobz/elasticsearch-head/archive/master.zip>
2. 解压head包 unzip master.zip
3. 进入head目录 cd /opt/elk/ elasticsearch-head-master
4. 下载依赖 npm install 如果下载失败，使用

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

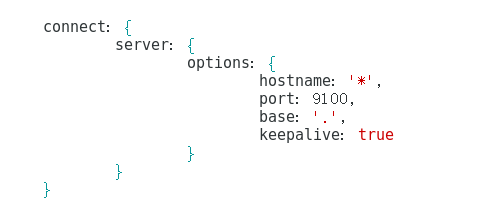
1. 停掉elasticsearch服务 jps列出进程 kill 进程号
2. 在es目录编辑elasticsearch配置 vi config/elasticsearch.yml

最后面添加内容：

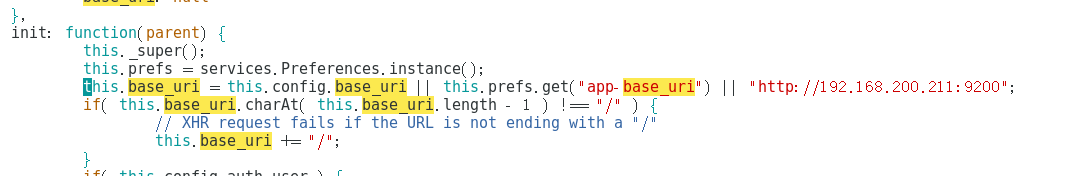
http.cors.enabled: true

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

1. 修改head插件配置 vi Gruntfile.js



1. 修改app.js vi \_site/app.js



1. Elasticsearch.yml最后添加以下配置：



1. 启动elasticsearch bin/elasticsearch –d
2. 启动head grunt server

后台无日志运行 nohup grunt server >/dev/null 2>&1 &

1. 通过浏览器访问 <http://192.168.200.211:9100>

## Logstash安装与配置

1. 下载压缩包 wget https://artifacts.elastic.co/downloads/logstash/logstash-6.1.2.tar.gz
2. 解压 tar –zxvf logstash-6.1.2.tar.gz
3. 进入目录 cd logstash-6.1.2
4. 创建启动配置文件 logstash-to-es.conf

Input：输入源

Filter：过滤器

Output：输出到

1. 启动 bin/logstash -f logstash-to-es.conf

后台无日志运行 nohup bin/logstash -f logstash-to-es.conf >/dev/null 2>&1 &

Logstash安装后需要配置自己的启动配置文件，主要配置输入源和输出源，以filebeat到logstash然后再到es为例beat\_to\_es.conf

Input: 输入源 默认beats的监听端口为5044，所有beats设置的上传输出源端口应与此端口一致

Filter：过滤器，内置各种过滤清洗插件

Grok： 匹配插件，使用ruby正则表达式，内置许多匹配模式

Match： 具体匹配的规则，根据自己的定义划分字段

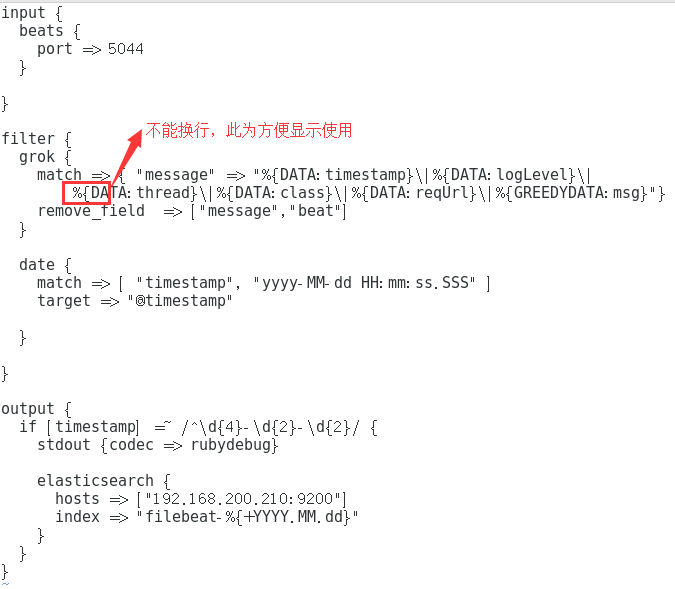
Remove\_field : 删除不需要的字段

Date： 日期转换， 用于个别字段专用处理，其他字段也可以通过其他规则处理，target => “@timestamp”将生产的时间戳配置为日志的时间戳

Output： 输出源，配置输出地址和方式

可使用if / when.contain /when equls等表达式，参考ruby语法

beat\_to\_es.conf参考配置：



## Kibana安装与配置

### 安装

1. 下载kibana安装包 wget https://artifacts.elastic.co/downloads/kibana/kibana-6.1.2-linux-x86\_64.tar.gz
2. 解压 tar –zxvf kibana-6.1.2-linux-x86\_64.tar.gz
3. 改名 mv kibana-6.1.2-linux-x86\_64.tar.gz kibana-6.1.2
4. 修改配置文件 vi config/kibana.yml

修改elasticsearch.url项地址为自己的elasticsearch地址

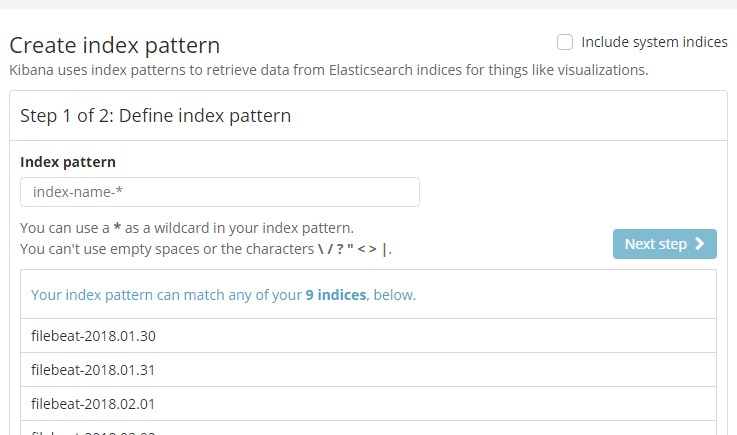
1. 启动 ./bin/kibana

后台无日志运行 nohup ./bin/kibana >/dev/null 2>&1 &

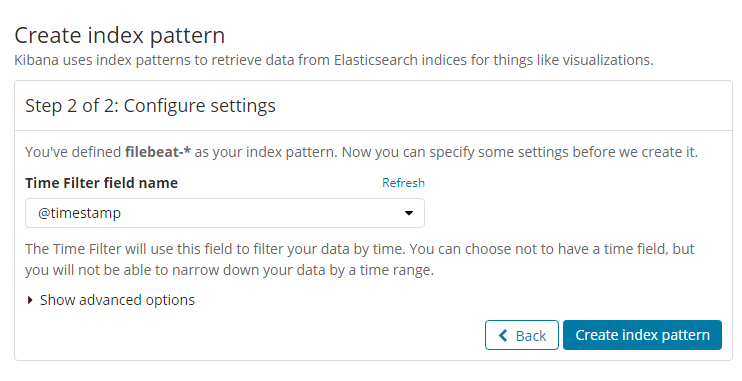
1. 测试 <http://localhost:5601>

### 使用

1. 首次登陆，需要新建索引模式，用于匹配查询，点击management->Index Patterns

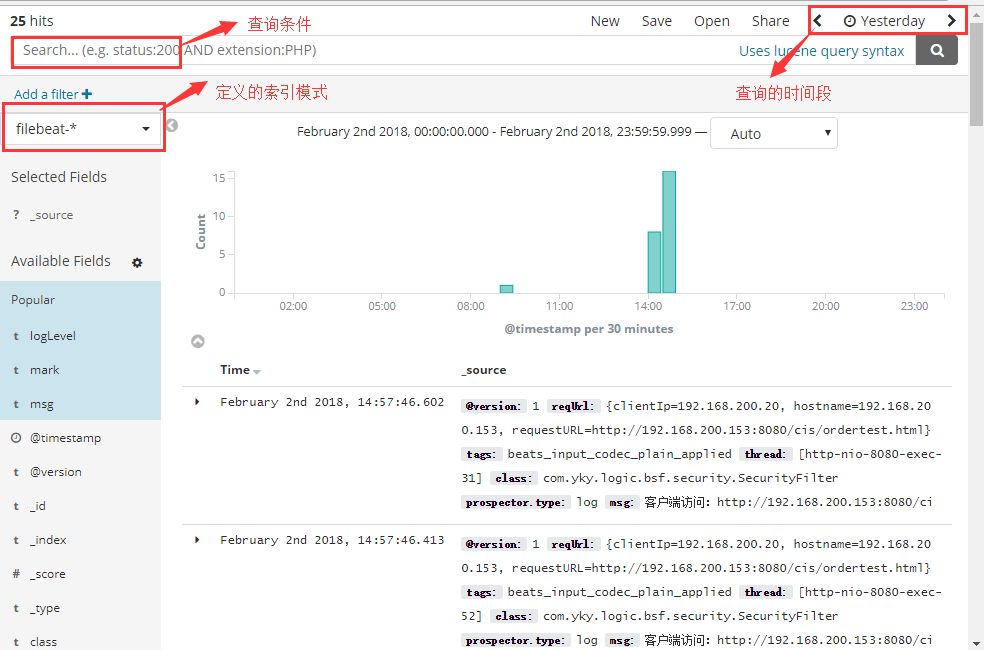


根据提示输入自己定义的索引匹配，如“filebeat-\*”，点击下一步，选择系统时间戳



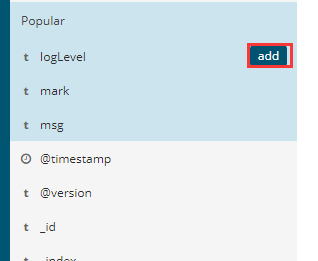
点击创建索引模式即可

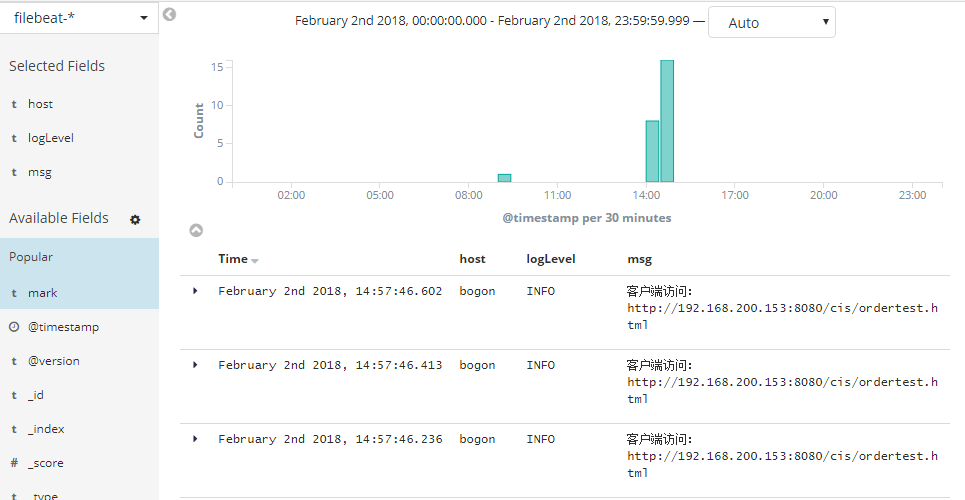
1. 点击Discovery菜单，选择自己定义的索引模式，选择查询日志的时间区域即可查询到相关日志



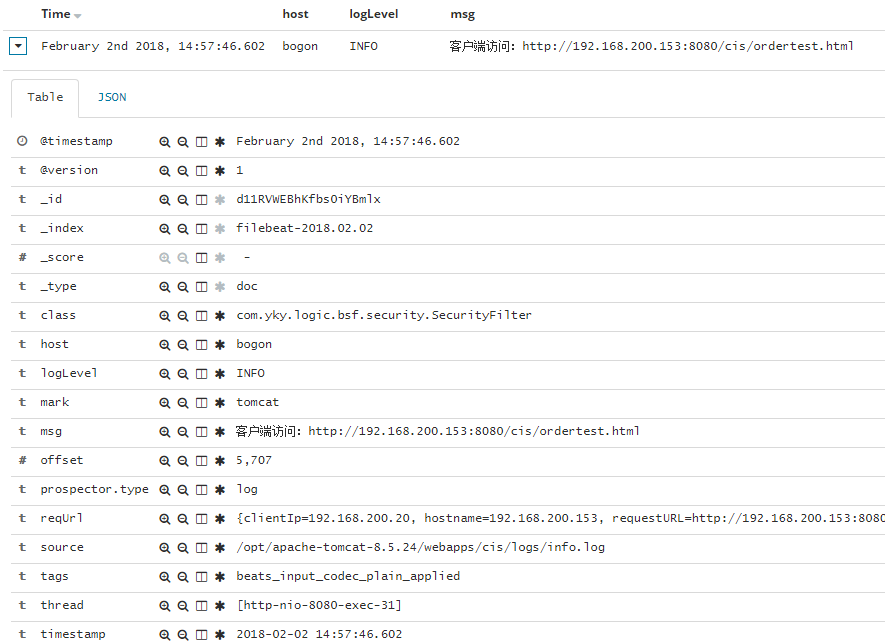
查询条件框可输入随意字段，也可以输入key: value匹配查询，key必须为模板能识别的字段

1. 鼠标移至左侧fields列表，点击add标签，将字段添加为表列查看

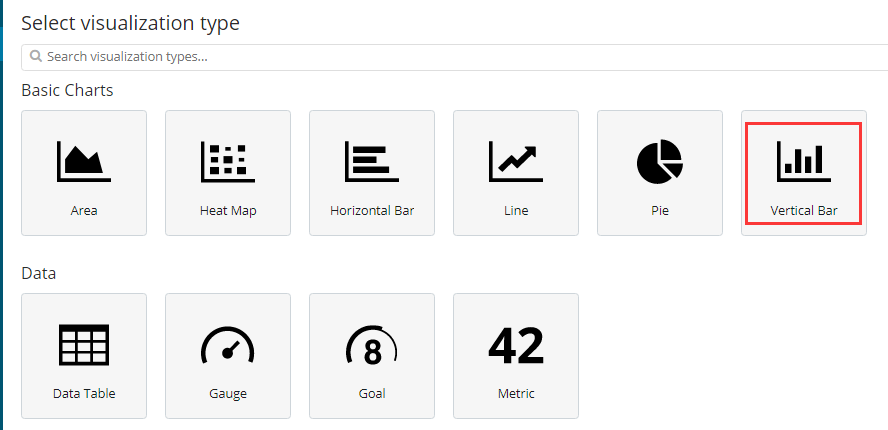




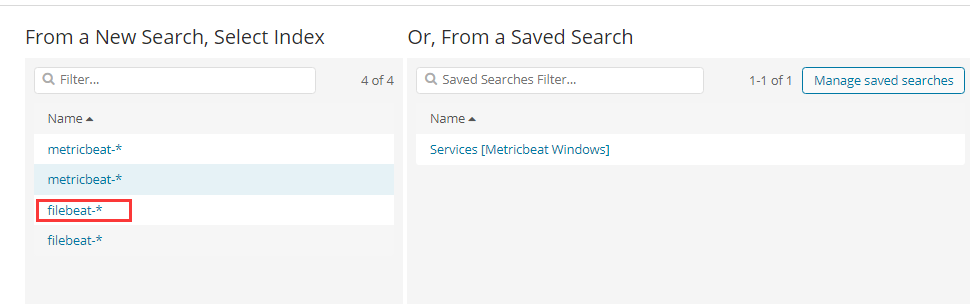
1. 点击Time列表下三角箭头，可查看该列详细信息



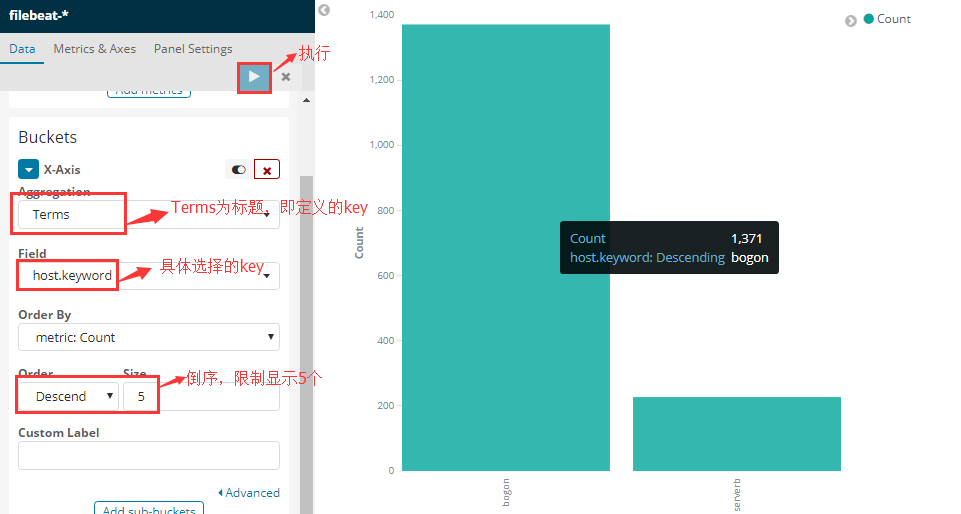
1. 点击visualize菜单，点击新建，选择图形模板，以柱状图为例



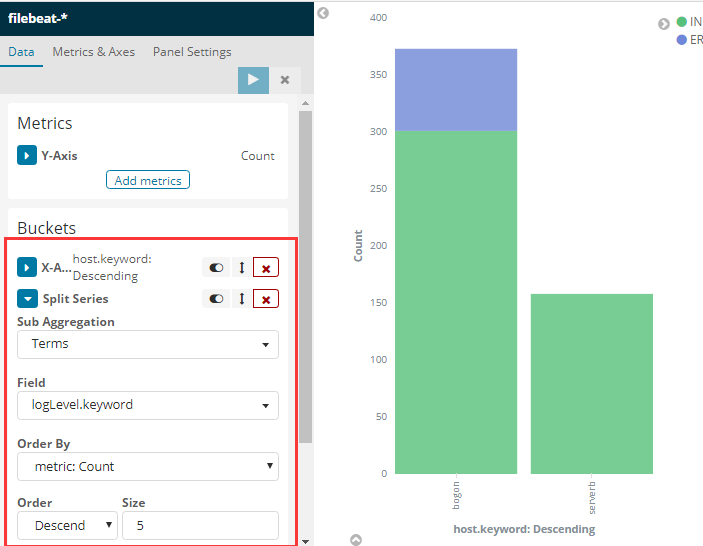
1. 选择要匹配的索引或保存过的查询



1. 选择自己的x轴要显示的内容（y轴通常为统计数）



可点击下面的Add sub-buckets添加额外的标题显示

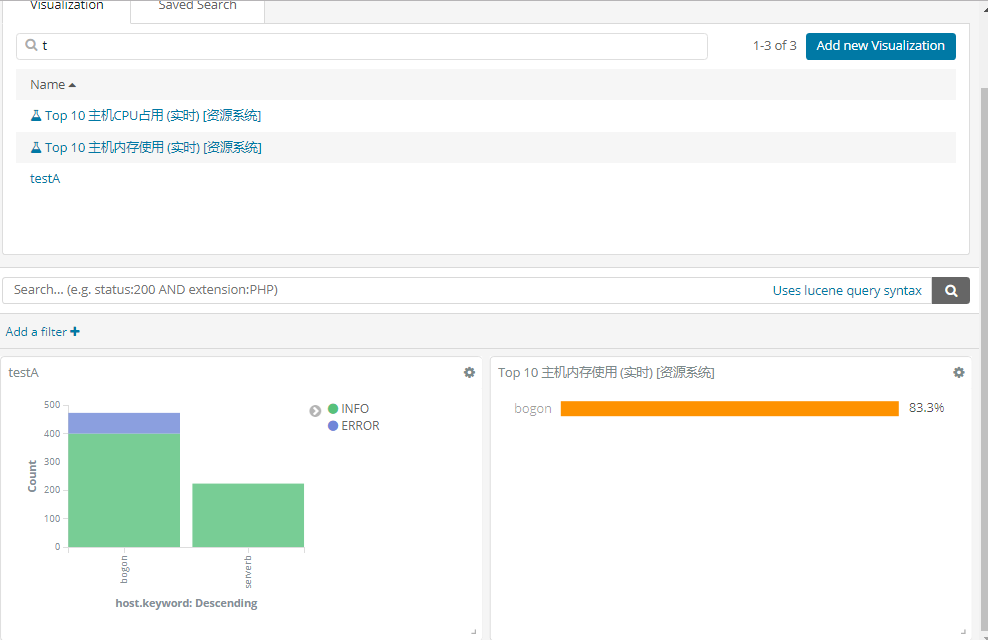


1. 右上可选择保持图表，图表刷新频率和查询时间段

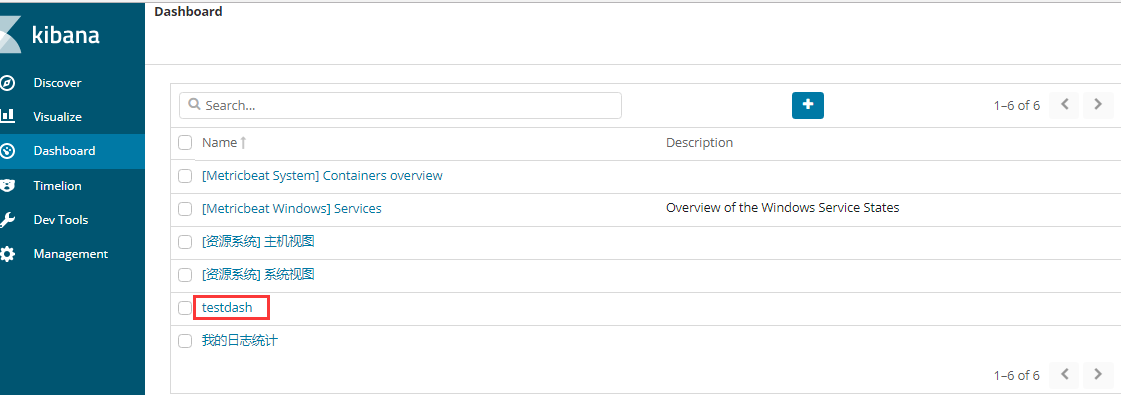


选择好后保存自己的图表

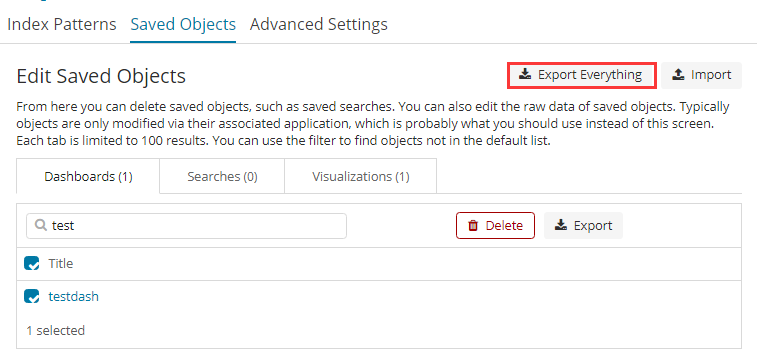
1. 进入Dashboard菜单，点击新增（dashboard），选择要展示的图表，这里选择刚自己定义好的图表，可以选择多个，自动排序显示，可拖动顺序



1. 同样可以设置查询，查询条件和查询时间的，然后点击保存，以后再进入dashboard菜单时就可以直接点击显示自己的dashboard了



1. timelion菜单，此功能供配置实时的监控的绘图功能，通过专用的语法显示，具体语法参考官网或查询相关资料
2. Dev tools此功能通过es专用的结构化语句对es的数据进行管理，增删改查，结果为json格式
3. 配置好自己的图表，查询和dashboard后，回到management菜单，点击saved objects菜单，查询，并选择自己的dashboard，然后点击export everything，保存自己的kibana模板



1. 在新搭建的环境中就可以在savedobjects菜单点击import功能导入自己的模板，因为导出的模板只有索引模式时没有，此时只需要配置自己的索引，并刷新即可使用了

## Filebeat安装与配置

1. 下载filebeat压缩包 wget https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-6.1.2-linux-x86\_64.tar.gz
2. 解压缩filebeat包 tar -zxvf filebeat-6.1.2-linux-x86\_64.tar.gz
3. 修改filebeat.yml文件 vi filebeat.yml
4. 启动./filebeat -e -c filebeat.yml

后台无日志运行 nohup ./filebeat -e -c filebeat.yml >/dev/null 2>&1 &

1. 导入模板（如果需要, filebeat.yml配置kibana地址） ./filebeat setup

Filebeat.yml配置介绍：

Filebeat.prospectors: 日志探测器，查找要监控的日志，每一个“-”标记表示一个探测器，每一个被监视的文件，都会分配一个harvester，监控该文件的新增内容，并传送到spooler，然后spooler汇总这些event送到logstash或es

Type: 监控的日志类型 log/slowlog/mysql/reids等等

Paths: 监控的文件路径，可模糊匹配

Fileds: 自定义自己的字段，可多个，可在此设置日志区分标志

Fields\_under\_root: true 自定义字段显示的位置为标准字段，false时为fields字段内字段

Multiline： 多行日志合并显示，用于异常日志

Set kibana: 需要通过命令上传模板时必须配置

Output.logstash: 输出到logstash 可改为redis或elasticsearch

filebeat.yml参考配置：



## Metricbeat安装配置

1. 下载 wget https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-6.1.2-linux-x86\_64.tar.gz
2. 解压 tar –zxvf metricbeat-6.1.2-linux-x86\_64.tar.gz
3. 修改metribeat.yml配置
4. 导入dashboards模板 ./metricbeat setup

此操作会导入所有的模板，索引，仪表和查询，只供测试使用，生产环境使用已配置好手工导出的模板，然后在kibana端手工导入即可

1. 启动 ./metricbeat –e –c metricbeat.yml

后台无日志运行 nohup ./metricbeat –e –c metricbeat.yml >/dev/null 2>&1 &

metricbeat.yml参考配置：



Tips:

1. 通过logstash过滤显示出来的自定义字段，默认是不做为模板字段的，所以在绘图时不可用，需要到index patterns中找到该索引匹配，刷新后就可以正常使用了
2. Discovery是查看数据的
3. Visualize 是选择模板绘制图表的，可选择图表的数据时间范围，以及刷新频率
4. Dashboard 是展示已经绘制好的图表的，同一个board可以放置多个图表
5. Timelion是绘制实时数据统计图表的，需要使用专用描述语言配置
6. Dev Tools 对于es相当于pl/sql相对于数据库， 在dev tools中可以查询，新增，修改，删除数据

## X-Pack安装配置

如果需要使用，则在集群中每个es节点都需要安装

./bin/elasticsearch-plugin install x-pack

Kibana下载也需要安装

./bin/kibana-plugin install x-pack

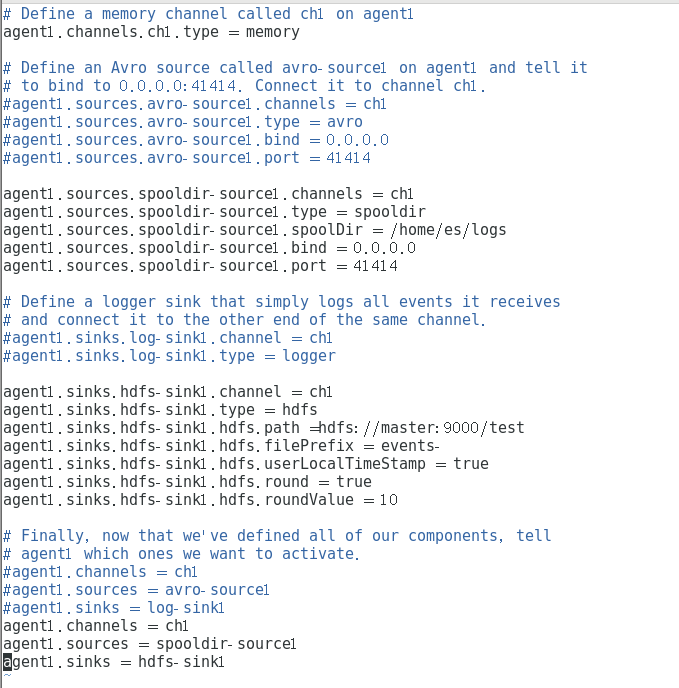
Logstash可选择性安装

## Flume安装与配置

1. 下载 wget http://mirrors.tuna.tsinghua.edu.cn/apache/flume/1.8.0/apache-flume-1.8.0-bin.tar.gz
2. 解压 tar –zxvf apache-flume-1.8.0-bin.tar.gz
3. 修改文件夹名称 mv apache-flume-1.8.0-bin flume-1.8.0
4. 进入conf目录 cd flume-1.8.0/conf
5. 新建配置文件 cp flume-env.sh.template flume-env.sh

cp flume-conf.properties.template flume.conf

1. 修改配置文件flume.conf,设置相应的source，channel和sink

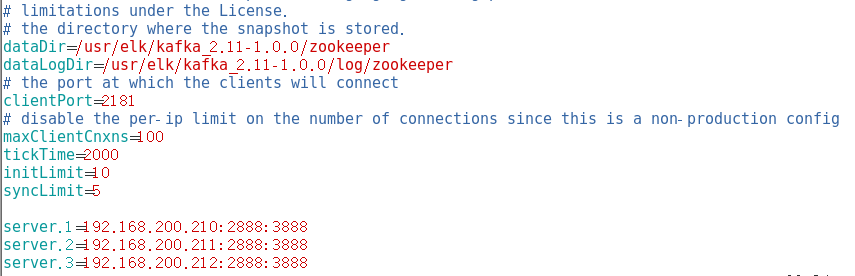


1. 启动flume bin/flume-ng agent --conf ./conf/ -f conf/flume.conf -Dflume.root.logger=DEBUG,console -n agent1

## Kafka集群安装与配置

1. 下载 wget <http://mirrors.shu.edu.cn/apache/kafka/1.0.0/kafka_2.11-1.0.0.tgz>
2. 解压 tar –zxvf kafka\_2.11-1.0.0.tgz
3. 进入目录 cd kafka\_2.11-1.0.0
4. 准备集权部署机器，210，211，212三台服务器为例
5. 先建立zk集群，直接使用kafka自带的zookeeper建立zk集群，修改zookeeper.properties文件： vi zookeeper.properties

三个机器上的zookeeper.properties文件配置相同，需要注意的是日志保存的路径，不会自动生成，需要自己手动建立相关的路径， dataLogDir是我自己加的，日志文件太多，把日志文件区分开；



1. 创建myid文件，进入/usr/local/kafka\_2.11-1.0.0/zookeeper，创建myid文件，将三个服务器上的myid文件分别写入1，2，3, myid是zk集群用来发现彼此的标识，必须创建，且不能相同
2. 进入kafka目录 执行启动zookeeper命令：
3. ./bin/zookeeper-server-start.sh config/zookeeper.properties &
4. 三台机器都执行启动命令，查看zookeeper的日志文件，没有报错就说明zookeeper集群启动成功了

## 测试flume + zookeeper + kafka

1. 根据配置启动zookeeper

bin/zookeeper-server-start.sh config/zookeeper.properties

1. 根据配置启动kafka

bin/kafka-server-start.sh config/server.properties

1. 创建测试主题

./bin/kafka-topics.sh --create --zookeeper 192.168.200.210:2181 --replication-factor 1 --partitions 1 --topic logstest

1. 查看主题是否创建成功

./bin/kafka-topics.sh --list --zookeeper 192.168.200.210:2181

1. 创建测试生产者

bin/kafka-console-consumer.sh --zookeeper 192.168.200.210:2181 --topic logstest --from-beginning

1. 创建测试消费者

./bin/flume-ng agent --conf conf -f ./conf/flume-kafka.conf -n agent -Dflume.root.logger=INFO,console

1. 根据配置启动flume

./bin/flume-ng agent --conf conf -f ./conf/flume-kafka.conf -n agent -Dflume.root.logger=INFO,console

1. 修改flume监视文件查看效果