# ELK安装实践

## 准备工作

下载elasticsearch,kibana,logstash

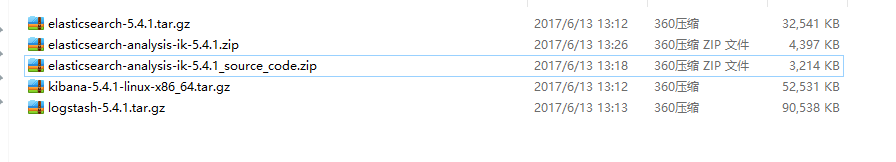
<https://www.elastic.co/cn/products>

ik分词器

从git上检出代码

<https://github.com/medcl/elasticsearch-analysis-ik>

用maven打包



## Elasticsearch安装&集群部署

### 配置

cluster.name: elasticsearch-cluster

node.name: node-master

path.data: /usr/local/service/elk/elastic\_master/data

path.logs: /usr/local/service/elk/elastic\_master/logs

network.host: 0.0.0.0

http.port: 9200

transport.tcp.port: 9300

transport.tcp.compress: true

#允许跨域访问

http.cors.enabled: true

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

discovery.zen.ping.unicast.hosts: ["172.16.2.175:9300","172.16.5.49:9301", "172.16.5.50:9302"]

discovery.zen.minimum\_master\_nodes: 1

### 安装ik分词器

将打包得到的 elasticsearch-analysis-ik-5.4.1.zip 放到es目录下/plugins/ik 目录下

解压缩即可

可用elasticsearch-plugin list命令查看插件安装情况

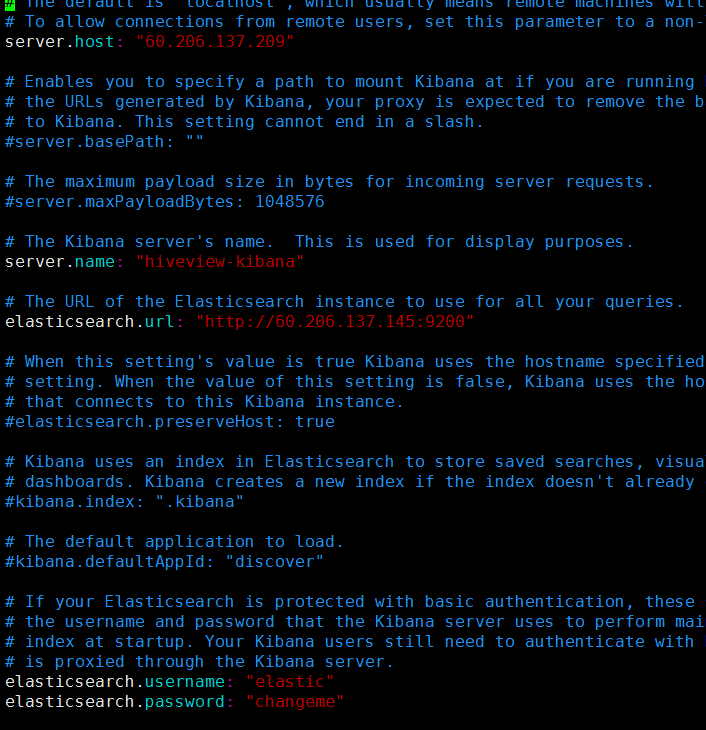
### 安装x-pack

./elasticsearch-plugin install x-pack

## Kibana安装

下载对应版本kibana

解压配置 对应elasticsearch url



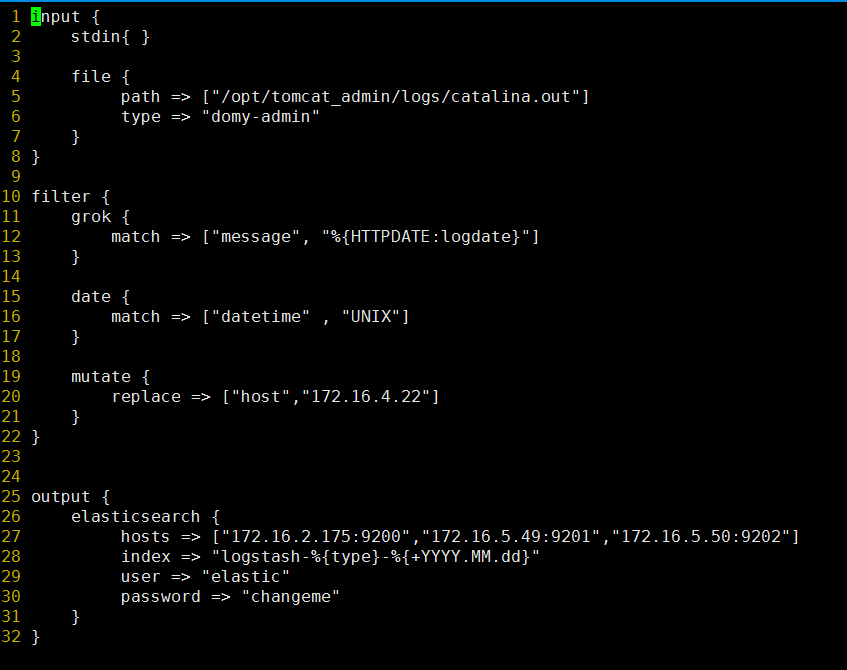
## Logstash安装

### 下载对应版本logstash

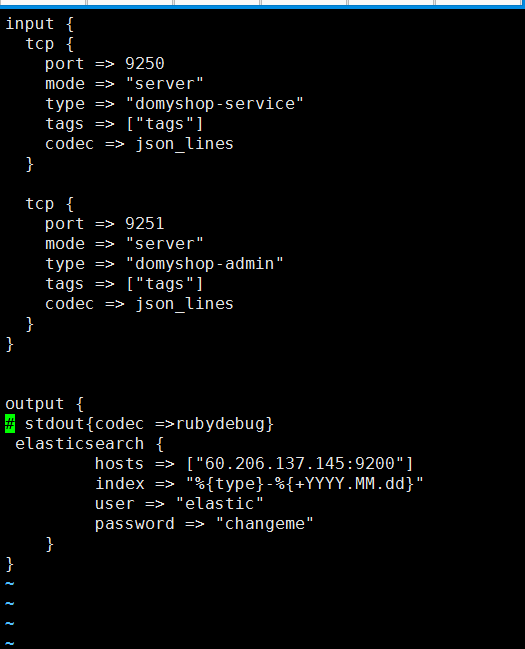
目前我试了两种方式

#### 监听文件的形式 收集log然后发送到elasticsearch (我感觉这种缺点就是每台应用服务器都需要安装一个logstash,但是可以保证收集日志的完整性)

##### logstash 配置



#### 监听tcp端口的形式 收集log 发送到elasticsearch 可以独立出来logstash服务, 对应应用程序需要 有log4j 或者 logback 能够发送log信息到logstash服务器.



##### 项目使用logback做日志管理

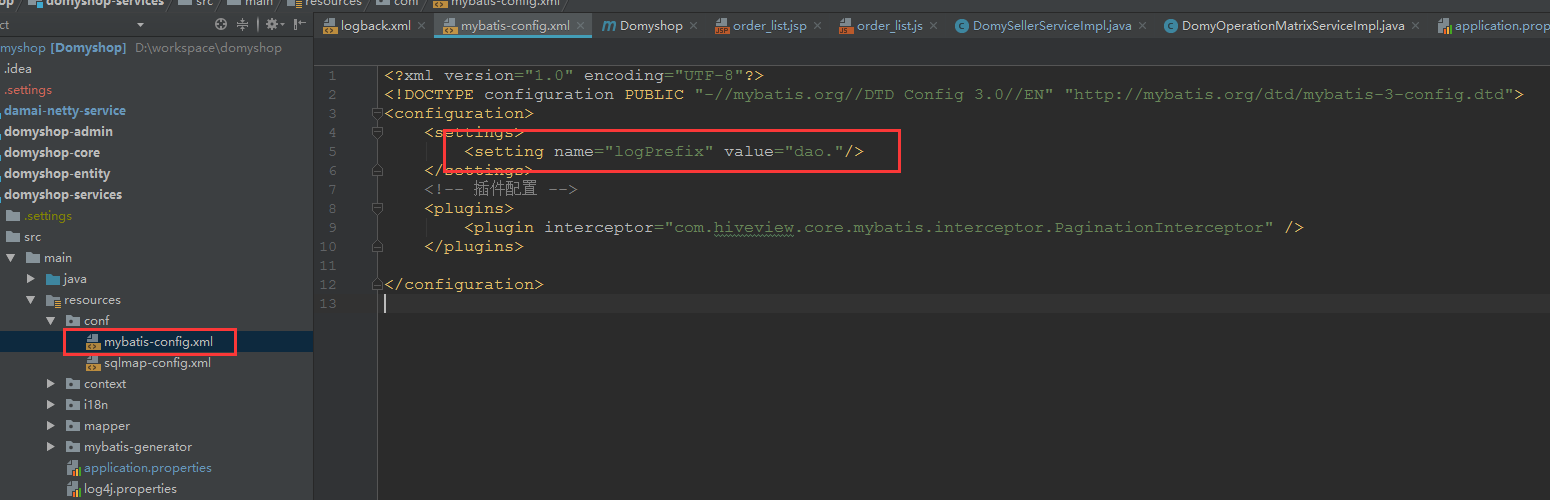
###### Maven 依赖

<logback.version>1.1.7</logback.version>  
<logstash.logback.version>4.11</logstash.logback.version>  
<org.slf4j.version>1.7.21</org.slf4j.version>

<!--日志 start-->  
<dependency>  
 <groupId>ch.qos.logback</groupId>  
 <artifactId>logback-core</artifactId>  
 <version>${logback.version}</version>  
</dependency>  
<dependency>  
 <groupId>ch.qos.logback</groupId>  
 <artifactId>logback-classic</artifactId>  
 <version>${logback.version}</version>  
</dependency>  
<dependency>  
 <groupId>ch.qos.logback</groupId>  
 <artifactId>logback-access</artifactId>  
 <version>${logback.version}</version>  
</dependency>  
<dependency>  
 <groupId>org.slf4j</groupId>  
 <artifactId>slf4j-api</artifactId>  
 <version>${org.slf4j.version}</version>  
</dependency>  
<dependency>  
 <groupId>org.slf4j</groupId>  
 <artifactId>jcl-over-slf4j</artifactId>  
 <version>${org.slf4j.version}</version>  
</dependency>  
  
<dependency>  
 <groupId>org.logback-extensions</groupId>  
 <artifactId>logback-ext-spring</artifactId>  
 <version>0.1.2</version>  
</dependency>  
  
<dependency>  
 <groupId>net.logstash.logback</groupId>  
 <artifactId>logstash-logback-encoder</artifactId>  
 <version>${logstash.logback.version}</version>  
</dependency>  
<!--日志 end-->

###### Mybatis-confg配置

用于打印sql语句



###### logback.xml配置

<?xml version="1.0" encoding="UTF-8"?>  
<configuration scan="true" debug="false">  
 <!--加载外部属性文件-->  
 <property resource="application.properties" />  
 <property file="/etc/conf/service/application.properties" />  
 <!-- 控制台输出 -->  
 <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">  
 <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">  
 <!--格式化输出：%d表示日期，%thread表示线程名，%-5level：级别从左显示5个字符宽度%msg：日志消息，%n是换行符-->  
 <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{50} - %msg%n</pattern>  
 </encoder>  
 </appender>  
 <!-- 按照每天生成日志文件 -->  
 <appender name="FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">  
 <file>${log.path}/${application.name}.log</file>  
 <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">  
 <!--日志文件输出的文件名-->  
 <FileNamePattern>${log.path}/${application.name}.%d{yyyy-MM-dd}.log</FileNamePattern>  
 <!--日志文件保留天数-->  
 <MaxHistory>30</MaxHistory>  
 </rollingPolicy>  
 <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">  
 <!--格式化输出：%d表示日期，%thread表示线程名，%-5level：级别从左显示5个字符宽度%msg：日志消息，%n是换行符-->  
 <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{50} - %msg%n</pattern>  
 </encoder>  
 <!--日志文件最大的大小-->  
 <triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">  
 <MaxFileSize>10MB</MaxFileSize>  
 </triggeringPolicy>  
 </appender>  
 <!--输出到logstash-->  
 <appender name="stash" class="net.logstash.logback.appender.LogstashTcpSocketAppender">  
 <destination>${logstash.host}</destination>  
  
 <!-- encoder is required -->  
 <encoder charset="UTF-8" class="net.logstash.logback.encoder.LogstashEncoder" />  
 </appender>  
 <!--打印mybatis sql语句-->  
 <logger name="dao" level="DEBUG"/>  
 <!-- 日志输出级别 -->  
 <root level="INFO">  
 <!--控制台输出-->  
 <appender-ref ref="STDOUT" />  
 <!--输出到日志文件-->  
 <appender-ref ref="FILE" />  
 <!--输出到logstash-->  
 <appender-ref ref="stash" />  
 </root>  
</configuration>

###### Logstash配置

