**使用flume、elasticSearch、Kibana构建日志检索系统**

# 总体介绍

系统运行图，flume将数据归集到 flume agent server， flume agent Server再将数据写入到elasticSearch Server建立全文检索。用户通过浏览器访问Kibana，Kibana查找相应数据，返回查找结果。

Flume Agent

Flume Agent

Flume Agent

Flume agent Server

ElasticSearch Server

Kibana

IE / Chrome

# 部署模式

|  |  |  |
| --- | --- | --- |
| 组件 | 部署位置 | 说明 |
| flume Agent | 需要采集日志的机器。如XXX系统所在机器 | 必须 |
| flume agent server | 任意机器，如果服务器充裕建议独立服务器 | flume agent server不是必须，可以有flume agent直接将日志发往 elastic search server |
| elastic Search server | 任意机器，如果服务器充裕建议独立服务器 | 必须 |
| Kibana | 任意机器，如果服务器充裕建议独立服务器 | 必须 |

# 案例

鉴于环境机器限制，以下案例采用all in one部署。即 elastic search和 Kibana部署在同一台服务器，同时不部署 flume agent server。所有机器运行环境

# 环境准备

|  |  |
| --- | --- |
| 机器清单 |  |
| 192.168.1.158 | 需要采集日志的业务系统所在服务器 |
| 192.168.1.159 | elasticsearch + kibana |
| 软件清单 |  |
| flume 1.5.2 | apache-flume-1.5.2.bin.tar.gz |
| ealsticsearch 1.5.2 | elasticsearch-1.5.2.tar.gz |
| kibana-4.0.2 | kibana-4.0.2-linux-x64.tar.gz |
|  |  |
| OS + JDK |  |
| x86\_64 GNU/Linux |  |
| JDK1.7 |  |
|  |  |
|  |  |

# Flume安装

* 解压 Flume 安装包 apache-flume-1.5.2.bin.tar.gz

C:\Users\wangqiaodong581\AppData\Roaming\Tencent\Users\2242703215\QQ\WinTemp\RichOle\@SGYX04YT$G7T29EP@C$YVH.png

tar –xvf apache-flume-1.5.2.bin.tar.gz

* 设置 ，进入解压后的conf目录创建 index.conf 和 flume-env.sh

Index.conf 内容如下,

agent.sources.tail.channels = memoryChannel

agent.sources.tail.type = exec

agent.sources.tail.command = tail -F /wls/applogs/rtlog/mtplogs/mtp/web/MTP-WEB.log

agent.sources.tail.interceptors=i3 i1

agent.sources.tail.interceptors.i3.type=org.apache.flume.interceptor.HostInterceptor$Builder

agent.sources.tail.interceptors.i3.hostHeader = host

agent.sources.tailIntegration.channels = memoryChannel

agent.sources.tailIntegration.type = exec

agent.sources.tailIntegration.command = tail -F /wls/applogs/rtlog/mtplogs/mtp/integration/MTP-INTEGRATION.log

agent.sources.tailIntegration.interceptors=i3 i1

agent.sources.tailIntegration.interceptors.i3.type=org.apache.flume.interceptor.HostInterceptor$Builder

agent.sources.tailIntegration.interceptors.i3.hostHeader = host

agent.sinks = elasticsearch

agent.sinks.elasticsearch.channel = memoryChannel

agent.sinks.elasticsearch.type=org.apache.flume.sink.elasticsearch.ElasticSearchSink

agent.sinks.elasticsearch.batchSize=100

#agent.sinks.elasticsearch.hostNames=192.168.6.115:9300

agent.sinks.elasticsearch.indexType=web

agent.sinks.elasticsearch.indexName=mtp-test3

agent.sinks.elasticsearch.hostNames=192.168.1.159:9300

#agent.sinks.elasticsearch.indexType = external

##

agent.sources.tail.interceptors.i1.type = regex\_extractor

agent.sources.tail.interceptors.i1.regex = ^(?:\\n)?(\\d\\d\\d\\d-\\d\\d-\\d\\d\\s\\d\\d:\\d\\d:\\d\\d\\.\\d\\d\\d)

agent.sources.tail.interceptors.i1.serializers = s1

agent.sources.tail.interceptors.i1.serializers.s1.type = org.apache.flume.interceptor.RegexExtractorInterceptorMillisSerializer

agent.sources.tail.interceptors.i1.serializers.s1.name = timestamp

agent.sources.tail.interceptors.i1.serializers.s1.pattern = yyyy-MM-dd HH:mm:ss.SSS

agent.sources.tailIntegration.interceptors.i1.type = regex\_extractor

agent.sources.tailIntegration.interceptors.i1.regex = ^(?:\\n)?(\\d\\d\\d\\d-\\d\\d-\\d\\d\\s\\d\\d:\\d\\d:\\d\\d\\.\\d\\d\\d)

agent.sources.tailIntegration.interceptors.i1.serializers = s1

agent.sources.tailIntegration.interceptors.i1.serializers.s1.type = org.apache.flume.interceptor.RegexExtractorInterceptorMillisSerializer

agent.sources.tailIntegration.interceptors.i1.serializers.s1.name = timestamp

agent.sources.tailIntegration.interceptors.i1.serializers.s1.pattern = yyyy-MM-dd HH:mm:ss.SSS

#agent.sinks.elasticsearch.indexName=customer

#agent.sinks.elasticsearch.clusterName=elasticsearch

#agent.sinks.elasticsearch.clusterName=elasticsearch

agent.sinks.elasticsearch.serializer=org.apache.flume.sink.elasticsearch.ElasticSearchLogStashEventSerializer

agent.sinks.elasticsearch.ttl=2d

agent.sinks.elasticsearch.indexNameBuilder=org.apache.flume.sink.elasticsearch.SimpleIndexNameBuilder

从flume-env.sh.template 拷贝出flume-env.sh，修改最后一行文件内容如下，注意下面目前其实指定的elasticsearch 的包所在的目录，需要elasticsearch安装后才存在

FLUME\_CLASSPATH="/wls/logop/work/elasticsearch-1.5.2/lib/\*"

* 启动， 为了flume启动方便可以创建一个脚本文件，内容如下：

export JAVA\_HOME="/wls/logop/work/jre1.7.0\_51";bin/flume-ng agent --conf conf --conf-file conf/mtp.conf --name agent -Dflume.root.logger=INFO,console 1>logs/index.log 2>&1 &

# Elastic search 安装

* 解压

C:\Users\wangqiaodong581\AppData\Roaming\Tencent\Users\2242703215\QQ\WinTemp\RichOle\IAKSZTBK(0@4]N$N2M[Z4~M.png

tar –xvf elasticsearch-1.5.2.tar.gz

* 启动

直接在bin目录运行 ./elasticsearch 即可

* 检验

当elasticsearch启动后，如果flume配置正确，数据就会进入，可以通过下面命令查看索引情况

|  |  |
| --- | --- |
| 说明 | 命令 |
| 获取索引结构 | curl -XGET http://192.168.1.159:9200/mtp-test/web/\_mapping?pretty=true |
| 启用ttl，过期时间 | curl -XPUT 'http://192.168.1.159:9200/mtp-test/web/\_mapping?pretty' -d '{  "web": {  "\_ttl":{"enabled": true}  }  }' |
| 列索引 | curl 'localhost:9200/\_cat/indices?v' |
| 获取索引 | curl -XGET 'http://localhost:9200/twitter/' |
| put 数据 | put /indexname/type/id  PUT /megacorp/employee/1  {  "first\_name" : "John",  "last\_name" : "Smith",  "age" : 25,  "about" : "I love to go rock climbing",  "interests": [ "sports", "music" ]  } |
| 使用id查询 | GET /megacorp/employee/1  GET /megacorp/employee/\_search?q=last\_name:Smith |
| DSL query language 全文检索 | GET /megacorp/employee/\_search  {  "query" : {  "match" : {  "last\_name" : "Smith"  }  }  } |
| 短语检索 | GET /megacorp/employee/\_search  {  "query" : {  "match\_phrase" : {  "about" : "rock climbing"  }  }  } |
| 结果高亮 | GET /megacorp/employee/\_search  {  "query" : {  "match\_phrase" : {  "about" : "rock climbing"  }  },  "highlight": {  "fields" : {  "about" : {}  }  }  } |
| 删除文档 | DELETE /website/blog/123 |
| 索引管理，创建 | PUT /my\_index  {  "settings": {  "number\_of\_shards" : 1,  "number\_of\_replicas" : 0  },  "mappings": {  "type\_one": { ... any mappings ... },  "type\_two": { ... any mappings ... },  ...  }  } |
| 删除 | DELETE /my\_index  DELETE /index\_one,index\_two  DELETE /index\_\*  DELETE /\_all |
| 修改 | PUT /my\_temp\_index/\_settings  {  "number\_of\_replicas": 1  } |
|  |  |

* 设置ttl

由于flume中设置了数据过期时间，默认情况elasticSearch不会启用，需要通过命令行启动

curl -XPUT 'http://192.168.1.159:9200/mtp-test/web/\_mapping?pretty' -d '{

"web": {

"\_ttl":{"enabled": true}

}

}'

# Kibana 安装

* 解压

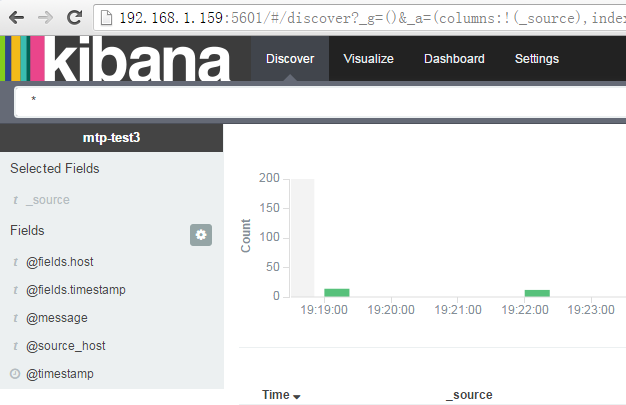
C:\Users\wangqiaodong581\AppData\Roaming\Tencent\Users\2242703215\QQ\WinTemp\RichOle\6M`VIL}Q)V_40ZTHZB0L5NY.png

* 启动

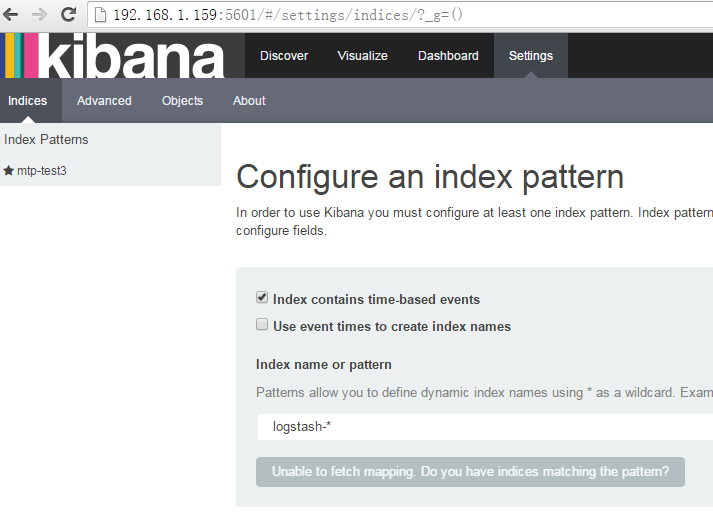
解压后的bin目录运行 Kibana即可

* 设置

由于缺省情况Kibana是西方风格，需要修改配置，并设置查询的索引。打开浏览器chrome， 输入http://ip:5601



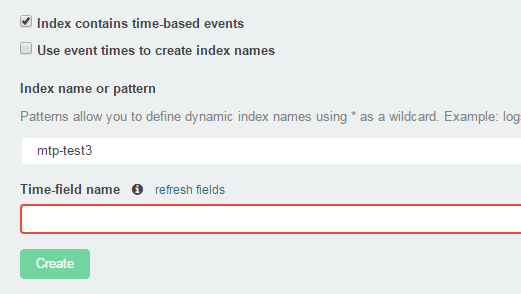
如上图选择Settings，



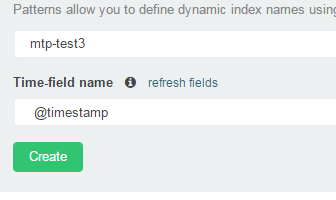
在上面位置输入日志名称，这个名称是 flume.conf配置文件中指定的。

agent.sinks.elasticsearch.indexName=mtp-test3

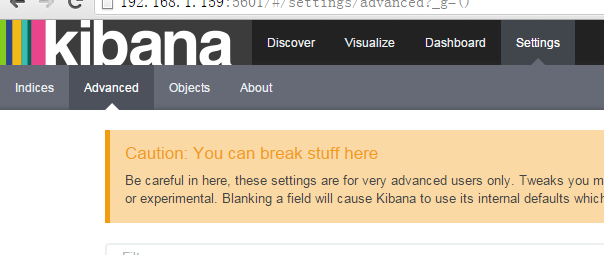
如上这个名称就是 mtp-test3



输入完成后，会显示时间Field 选择框，按照默认的选择即可。



接下来设置时间格式及显示项



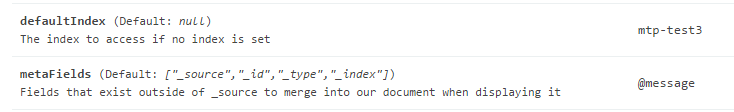
设置时间显示格式，选项是

C:\Users\wangqiaodong581\AppData\Roaming\Tencent\Users\2242703215\QQ\WinTemp\RichOle\O~`%$21]V03SU664FPTRCIL.png

设置时间格式为

C:\Users\wangqiaodong581\AppData\Roaming\Tencent\Users\2242703215\QQ\WinTemp\RichOle\$G~D$0]IJ){WF)%TUMVKIDP.png

设置默认索引及显示字段，选项是

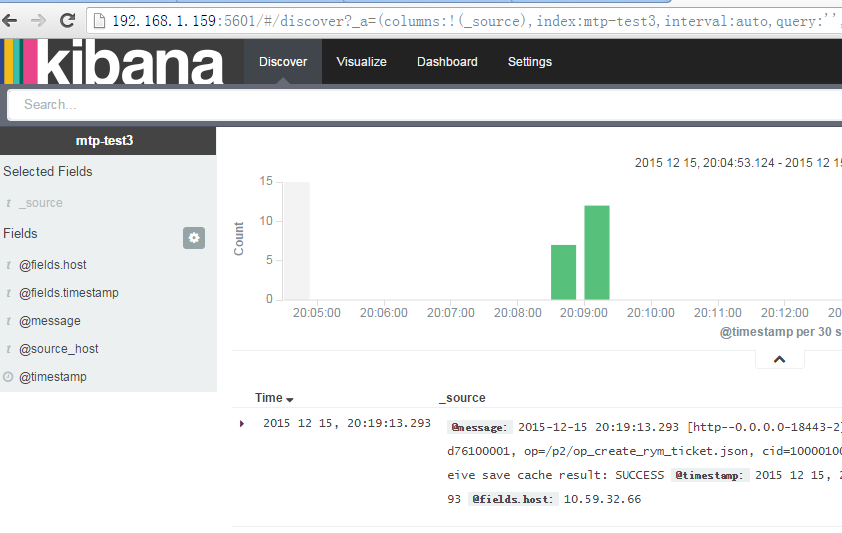


值分别如下

![C:\Users\wangqiaodong581\AppData\Roaming\Tencent\Users\2242703215\QQ\WinTemp\RichOle\K79{DE]KP38YZ{1I](Y1CAF.png](data:image/png;base64,)

# 大功告成

在方框中输入要收索的内容，点击查找



|  |  |
| --- | --- |
| 查找特定 | 查找方式 |
| 关键字有空格 | 通过”” 双引号，引注关键字, 例如 “beijing city” |
| 多个条件查找 | AND 关键字,例如 city AND province |
| 或条件查找 | 通过空格分隔即可 city province |
|  |  |

* 限定时间

