cc

文档属性：内部资料，请勿外传

文档编号：

项目编号：

**日志集群建设方案**

**北京畅游天下网络技术有限公司**

D-media Communication Tech

2016年10月

文档说明

本文档所涉及到的文字、图表等，仅限于北京畅游天下网络技术有限公司及被呈送内部使用，未经北京畅游天下网络技术有限公司书面许可，请勿扩散到第三方。

**文档属性**

|  |  |
| --- | --- |
| **属性** | **内容** |
| 客户名称： |  |
| 项目名称： |  |
| 项目编号： |  |
| 文档主题： | 日志集群建设方案 |
| 文档编号： |  |
| 最后更新： | 2016-08-16 |
| 文档状态： | 提交 |
| 文档作者： | Zline |

**文档历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **文档版本** | **日期** | **修订人** | **描述** |
| V1.0 | 2016-08-16 | Zline | 文档创建 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1. **日志集群建设目标** 3](#_Toc459188655)

[1.1**目标** 3](#_Toc459188656)

[1.2. **ELK介绍** 4](#_Toc459188657)

[2. **架构设计** 6](#_Toc459188658)

[2.1**架构图** 6](#_Toc459188659)

[2.2**基础组建** 6](#_Toc459188660)

[3. **实施** 6](#_Toc459188661)

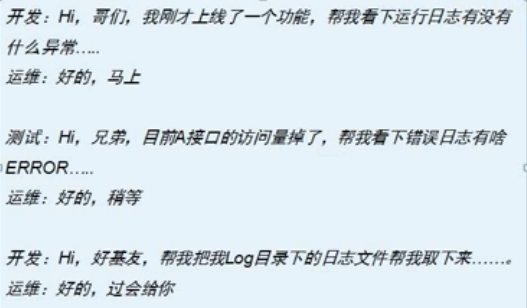
[3.1 **测试** 6](#_Toc459188662)

[3.2 **实施** 6](#_Toc459188663)

# 1. 日志集群建设目标

## 1.1目标

**⮚场景**



**⮚困扰**

⚫开发人员不能登录线上服务器查看详细日志

⚫日志数据分散难以查找

⚫日志数据量大，查询速度慢或者数据不够实时等

**⮚解决**

搭建实时日志查询、收集与分析系统

## 1.2. ELK介绍

⮚**介绍**

集中化管理日志后，日志的统计和检索又成为一件比较麻烦的事情，一般我们使用grep、awk 和 wc 等 Linux 命令能实现检索和统计，但是对于要求更高的查询、排序和统计等要求和庞大的机器数量依然使用这样的方法难免有点力不从心。

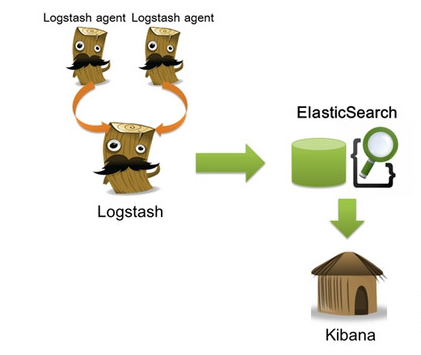
开源实时日志分析 ELK 平台能够完美的解决我们上述的问题， ELK 由 ElasticSearch 、 Logstash和Kiabana 三个开源工具组成。官方网站： <https://www.elastic.co/products>

**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、分析，并将其存储供以后使用（如，搜索）。

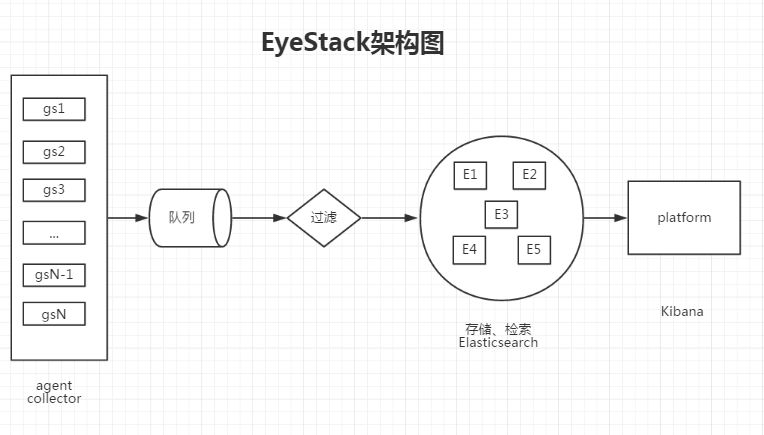
**kibana**也是一个开源和免费的工具，Kibana可以为Logstash 和 ElasticSearch 提供的日志分析友好的 Web 界面，可以帮助您汇总、分析和搜索重要数据日志。

⮚**工作原理**



# 2. 架构设计

## 2.1架构图



## 2.2基础组建

⮚日志收集客户端(logstash、rsyslog、filebeat等)

⚫Logstash过于负重，对于日志的格式和类型没有限制

⚫Rsyslog较轻量级，配置过程相比之下较笨重，不灵活

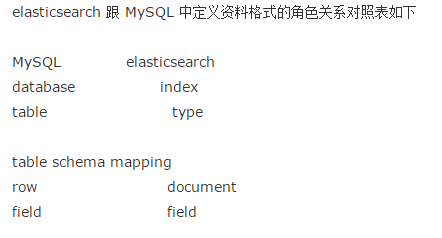
⚫Filebeat是logstash升级轻量级logstash-forwarder的替代品

⮚消息队列（redis、kafka等）

集群较小可不用中间件，添加消息队列是为了防止网络堵塞、集群过大ES集群处理应接不暇考虑

⮚存储检索（elasticsearch等）

ElasticSearch是一个基于Lucene构建的开源，分布式，RESTful搜索引擎。设计用于云计算中，能够达到实时搜索，稳定，可靠，快速，安装使用方便。支持通过HTTP使用JSON进行数据索引

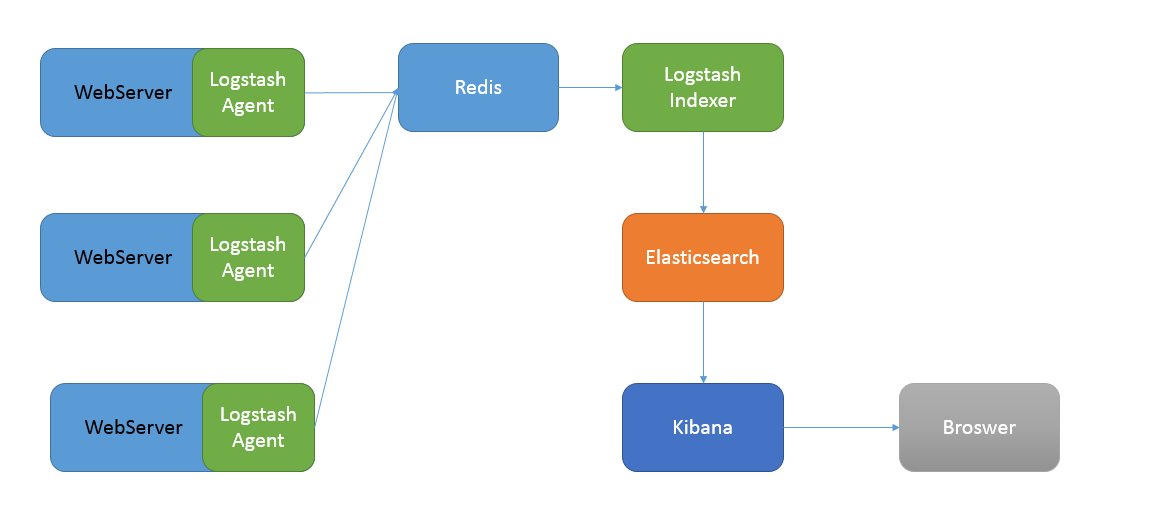


⮚统计分析可视化（kibana等）

# 3. 实施

## 3.1 测试

1.搭建基础设施并出测试报告（待定）



从左边看起，每一台webserver上都会部署一个logstash-agent,它的作用是用类似tailf的方式监听日志文件，然后把新添加的日志发送到redis队列里面，logstash-indexer负责从redis相应的队列里面取出日志，对日志进进行加工后输出到elasticsearch中，elasticsearch会根据要求对日志进行索引归集，最后用户可以通过kibana来查看和分析日志。

# 开始搭建日志系统

**准备工作**

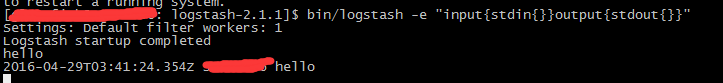
1. 安装JDK8
2. 安装Redis2.8

这里不涉及怎样安装JDK和Redis，请自行查阅资料

使用**logstash2.3.1**, **elasticsearch2.3.1**, **kibana2.3.1**

首先去ELK官网下载相应的压缩包   
<https://www.elastic.co/downloads>

## 在webserver上安装logstash-agent

1.解压 tar -zxvf logstash2.3.1   
2.检验安装是否成功 bin/logstash -e "input{stdin{}}output{stdout{}}",然后在终端输入hello，看看有没有内容返回   
   
3.编写配置文件logstash\_agent.conf   
在logstash安装目录下新建conf文件夹，在里面新建配置文件logstash\_agent.conf

input {

file {

type => "customer\_service"

#需要收集的日志文件

path => ["/home/java/logs/cust/customer-service-\*.log"]

tags => ["customer-service", "log4j2"]

#

codec => multiline { #

pattern => "^%{TIMESTAMP\_ISO8601}"

negate => true

what => "previous"

}

}

}

output {

redis {

host => "192.168.235.195"

data\_type => "list"

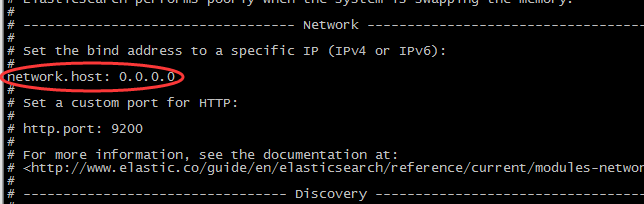
key => "logstash:redis:customer"

}

}

4.后台启动   
nohup ./bin/logstash -f conf/logstash\_agent.conf &

## 在日志server上安装elasticsearch

1.解压tar -zxvf elasticsearch-2.3.1.tar.gz   
2.修改安装目录下的配置文件config/elasticsearch.yml   
把network.host字段给反注释掉，把地址改为0.0.0.0（官方并没明确说要去改这配置，默认配置应该就可以了，不过实测的时候发现如果不做这修改，elasticsearch访问不了）   
   
3.后台启动elasticsearch   
nohup ./bin/elasticsearch &

## 在日志server上安装logstash-indexer

与在webserver上安装logstash-agent一节基本相同，只是配置文件不一样，这里使用logstash-indexer.conf

input {

redis {

host => "localhost" #redis地址

data\_type => "list"

key => "logstash:redis:customer"

type => "redis-input"

}

}

filter {

grok {

match => {

"message" => "%{TIMESTAMP\_ISO8601:date} \[(?<thread\_name>.+?)\] (?<log\_level>\w+)\s\*(?<content>.\*)"

}

}

date {

match => ["timestamp", "dd/MMM/YYYY:HH:mm:ss Z"]

}

}

output {

if[type] == "customer\_service" {

elasticsearch {

#embedded => false

#protocol => "http"

hosts => "localhost:9200"

index => "customer-%{+YYYY.MM.dd}"

}

}else if[type] == "其他类型" {

elasticsearch {

#embedded => false

#protocol => "http"

hosts => "localhost:9200"

index => "其他类型索引名字-%{+YYYY.MM.dd}"

}

}else {

//做其他处理

}

stdout{ #输出到标准输出，可以去掉

}

}

启动logstash-indexer   
nohup ./bin/logstash -f conf/logstash-indexer.conf &

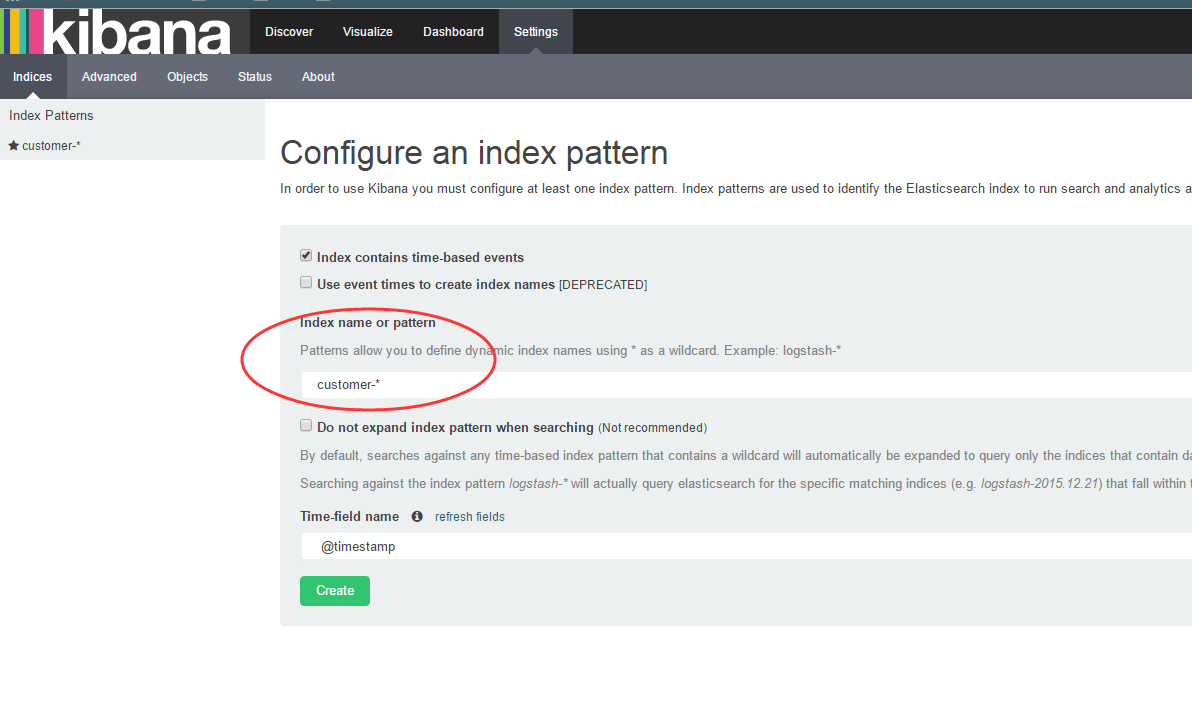
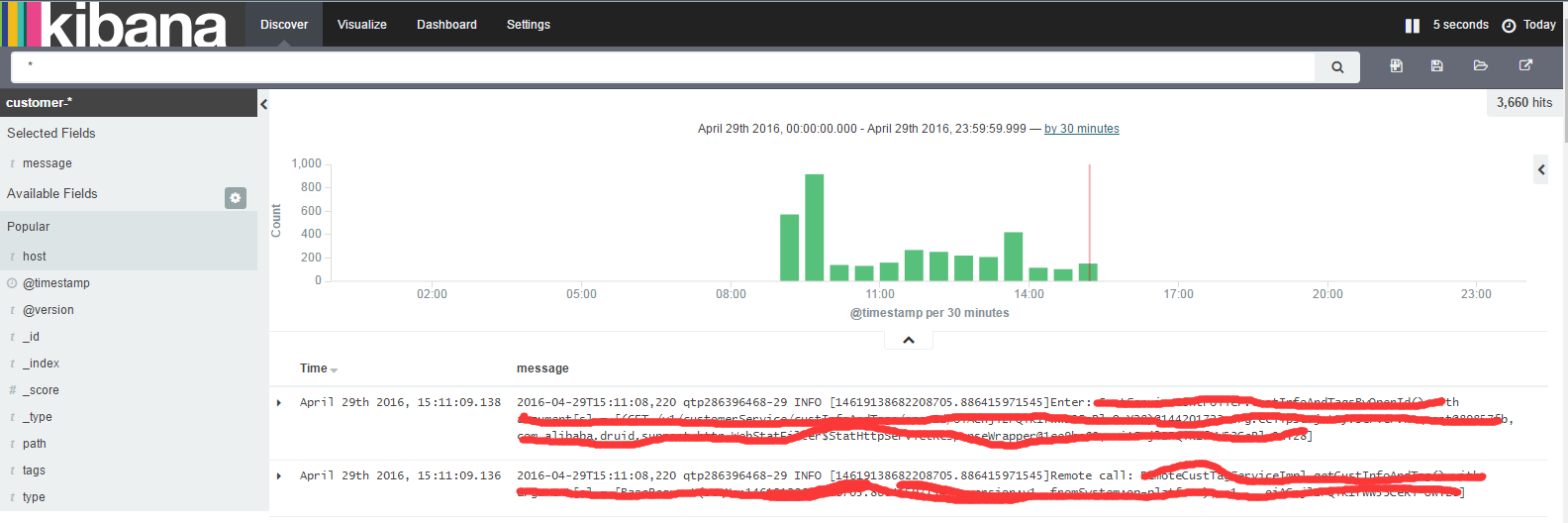
## 在日志server上安装kibana

1.解压tar -zxvf kibana-4.5.0-linux-x64.gz   
2.后台启动kibana nohup ./bin/kibana &   
这里没有特别去配置kibana要访问的elasticsearch地址，默认它会找本地的，如果需要，可以到conf/kibana.yml修改

## 配置log4j2的日志格式

<PatternLayout pattern="%d{ISO8601} %t %level %msg%n"/>

## 访问kibana主页

1.打开kibana主页<http://your-kibana-ip:5601/>   
2.输入索引customer-\*，并点击create   
   
3.在Discover页面观察日志   


# ToDo:

需要了解elasticsearch日志存放在哪里，定期清理日志

# 清理Elasticsearch索引

索引放久了需要清理，清理所以可以使用Elasticsearch的API。   
例如我需要删除某个索引2016年5月份的所有索引，可以使用下面的命令：

curl -XDELETE 'http://localhost:9200/customer-2016.05.\*'

更多API可以查阅Elastic官网   
<https://www.elastic.co/guide/en/elasticsearch/reference/current/docs-delete.html>

# 参考资料

<http://www.cnblogs.com/yjf512/p/4199105.html>   
<http://www.csdn123.com/html/topnews201408/34/15334.htm>   
<http://www.cnblogs.com/xing901022/p/4805586.html>   
<https://github.com/chenryn/logstash-best-practice-cn/blob/master/codec/multiline.md>   
<http://www.open-open.com/lib/view/open1451801542042.html>

## 3.2 实施

1.上线生产环境（待定）