ELK入门及如何搭建

1. ELK是什么？

在项目初期的时候，大家都是赶着上线，一般来说对日志没有过多的考虑，当然日志量也不大，所以用log4j就够了，随着应用的越来越多，日志散落在各个服务器的logs文件夹下，确实有点不大方便。

当我们需要日志分析的时候你大概会这么做：直接在日志文件中 grep、awk 就可以获得自己想要的信息。

那你们想过这种方式的问题吗？

1.日志量太大如何归档、文本搜索太慢怎么办、如何多维度查询

2.应用太多，面临数十上百台应用时你该怎么办

3.随意登录服务器查询log对系统的稳定性及安全性肯定有影响

4.如果使用人员对Linux不太熟练那面对庞大的日志简直要命

ELK因此就应运而生，那么为什么要用ELK呢？ELK又能给我们解决哪些问题呢？

1.日志统一收集，管理，访问。查找问题方便安全

2.使用简单，可以大大提高定位问题的效率

3.可以对收集起来的log进行分析

4.能够提供错误报告，监控机制

1. ELK架构设计

ELK是三个开源软件的缩写，分别表示：Elasticsearch , Logstash, Kibana , 它们都是开源软件

**1. LogStash**

     它可以流放到各自的服务器上收集Log日志，通过内置的ElasticSearch插件解析后输出到ES中

**2.ElasticSearch**

   这是一个基于Lucene的分布式全文搜索框架，可以对logs进行分布式存储，有点像hdfs。此为ELK的核心组件，日志的分析以及存储全部由es完成，因此在后面的课程中我们还会对此进行重点讲解。

**3.FileBeat**

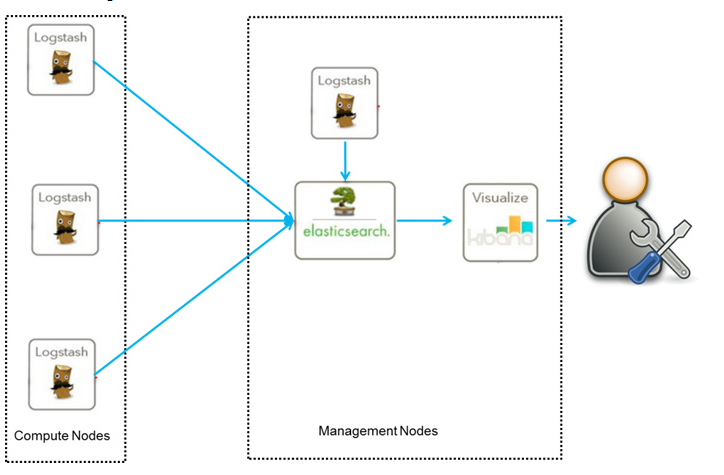
收集日志的组件，跟logstash相比他是一个轻量级的框架，占用极小的系统资源，但同样它也没有logstash那么多丰富的组件，所以我建议纯粹做日志收集的系统就用filebeat来替代logstash，即部署EFK

**4. Kibana**

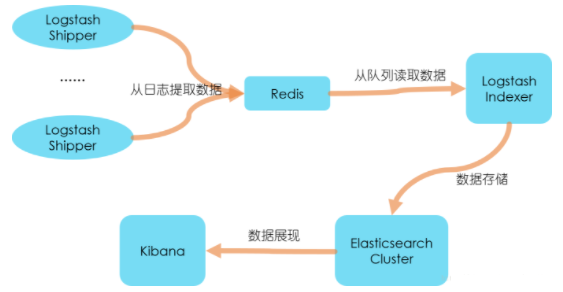
   它可以多维度的展示es中的数据。这也解决了用mysql存储带来了难以可视化的问题。他提供了丰富的UI组件，简化了使用难度，数据在es中的展示是比较让人蛋疼的，后面再讲es的时候让大家看看。

目前ELK主要有两种框架：

1.普通框架



2.个性化扩展框架（针对日志数据需要二次处理以及多方使用的场景）



三、如何快速搭建一套稳定的ELK环境

1.官方下载 :https://www.elastic.co/cn/products

2.版本：elasticsearch 6.6.0，kibana6.6.0 ，logstash6.6.0三个版本（注意 JDK需要1.8）

3.安装logstash

（1）解压，在config目录建：my-logstash.conf

（2）做好input ，filter，output三大块， 其中input是吸取logs文件下的所有log后缀的日志文件，filter是一个过滤函数，这里不用配置，output配置了导入到

hosts为127.0.0.1:9200的elasticsearch中，每天一个索引

|  |
| --- |
| input {  file {  type => "log"  path => ["/apps/svr/server/\*/log.file"]  start\_position => "end"  ignore\_older => 0  codec=> multiline {  pattern => "^\d{4}-\d{1,2}-\d{1,2}\s\d{1,2}:\d{1,2}:\d{1,2}"  negate => true  auto\_flush\_interval => 5  what => "previous"  }  }  beats {  port => 5044  }  }  output {  if [type] == "log" {  elasticsearch {  hosts => ["http://127.0.0.1:19200"]  index => "logstash-%{+YYYY.MM}"  #user => es  #password => es2018  }  }  } |

（3）bin目录下启动logstash了，配置文件设置为my-logstash.conf(大家最好直接拿我的配置文件，不要从这里copy，容易加特殊字符进去导致报错)

启动命令：./bin/logstash -f my-logstash.conf

FileBeat:安装

解压我给的包，配置文件我已经给你们放好了。

你们只需要修改下log收集的路径以及自己的es集群地址就可以了

./filebeat -e -d filebeat.yml

启动命令