日志中心（ELK）实现方案

# ELK简介

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。新增了一个FileBeat，它是一个轻量级的日志收集处理工具(Agent)，Filebeat占用资源少，适合于在各个服务器上搜集日志后传输给Logstash，官方也推荐此工具。

Elasticsearch是个开源分布式搜索引擎，提供搜集、分析、存储数据三大功能。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 主要是用来日志的搜集、分析、过滤日志的工具，支持大量的数据获取方式。一般工作方式为c/s架构，client端安装在需要收集日志的主机上，server端负责将收到的各节点日志进行过滤、修改等操作在一并发往elasticsearch上去。

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，可以帮助汇总、分析和搜索重要数据日志。

Filebeat隶属于Beats。目前Beats包含四种工具：

1. Packetbeat（搜集网络流量数据）
2. Topbeat（搜集系统、进程和文件系统级别的 CPU 和内存使用情况等数据）
3. Filebeat（搜集文件数据）

4 .Winlogbeat（搜集 Windows 事件日志数据）

# ELK架构详解

### 组件原理详解

#### Fliebeat

会先读取配置文件内配置的文件路径，然后将读取到的每个文件的文件状态记录在文件中（默认在/var/lib/filebeat/registry）。此状态可以记住Harvester收集文件的偏移量并将偏移量存储在内存中。对于每个文件，Filebeat存储唯一标识符以检测文件是否先前被收集，然后将文件增长的偏移量发送给输出设备若连接不上输出设备，如ES等，filebeat会记录发送前的最后一行(也就是没有发送成功的那一行)，并再可以连接的时候继续发送。

#### Logstash

分为input，fileter，output三阶段。Input主要是用于收集和接受数据，包括收集本地日志，redis中读取数据，beats接受数据。Fileter用于过滤数据，比如对数据格式，内容进行重构，添加，删除等。Output用于输出过滤后的数据，一般我们都输出给es存储。

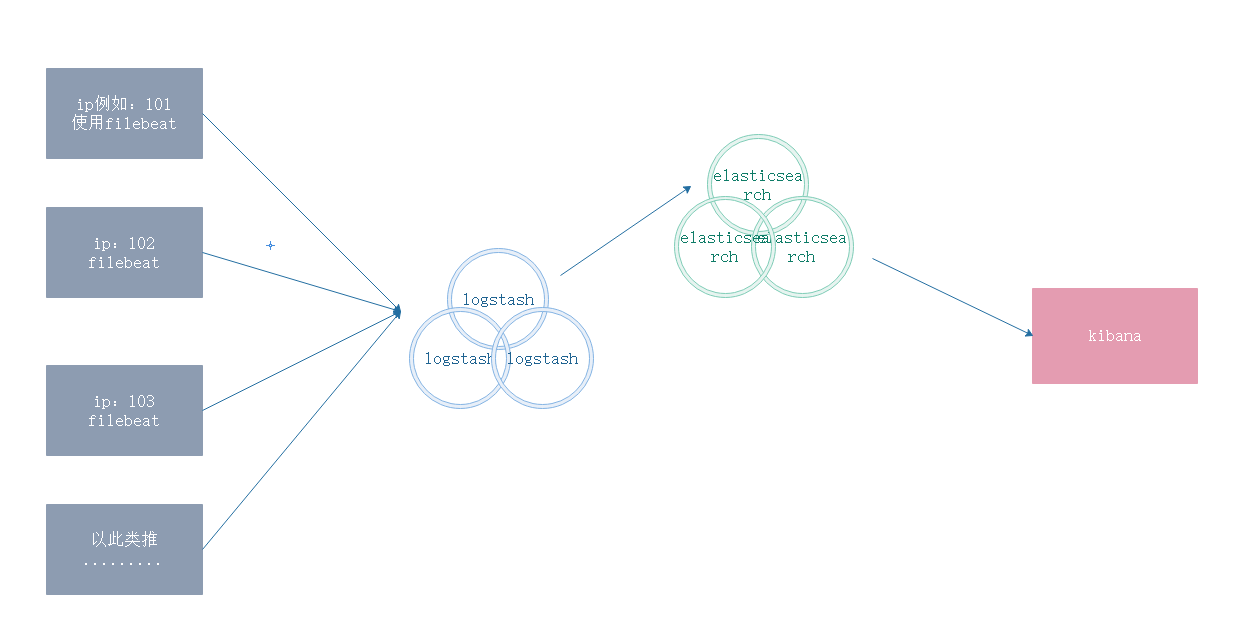
#### Elasticsearch

Elasticsearch 是一个分布式可扩展的实时搜索和分析引擎，接受并存储从logstash发送过来的数据，并将每一个字段都编入索引，使其可以被搜索。

#### Kibana

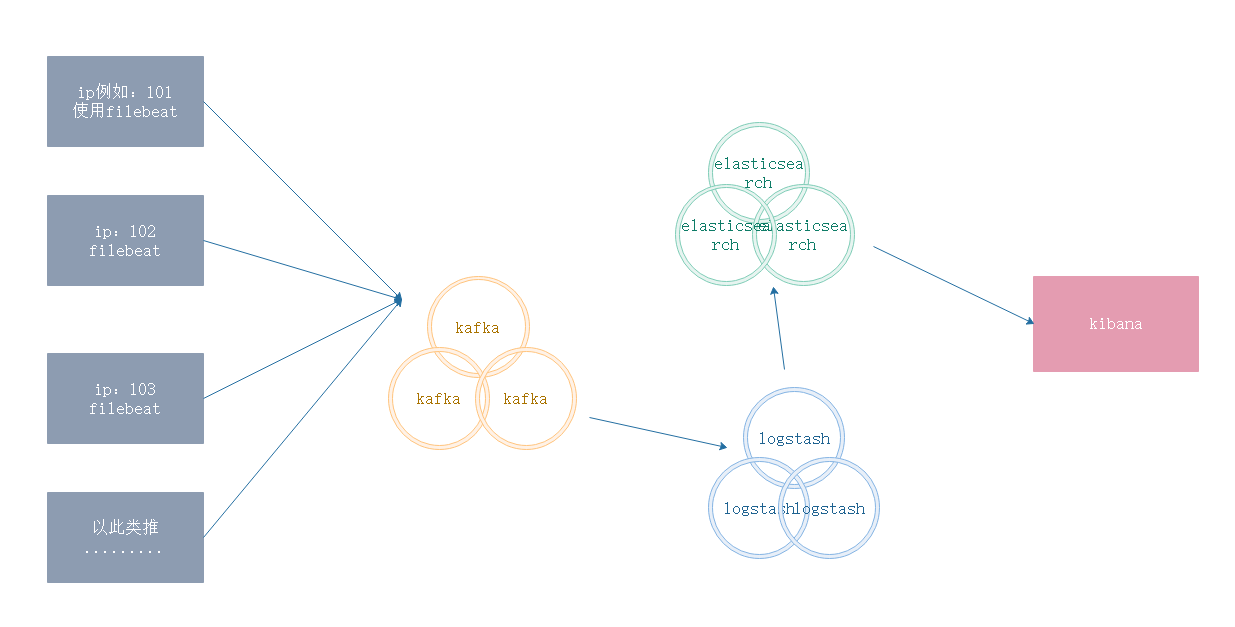
Kibana从es中读取数据及索引并进行展示，主要从web界面进行展示我们的日志内容，增长量等，kibana还可以用于监控es集群

### 基本架构图及图解



1. 首先我们在每台需要收集日志的应用机上安装filebeat，然后filebeat配置完成后会收集日志传输给logstash（ps：这里因为filebeat占用资源较少所以用轻量级的filebeat收集日志数据，logstash占用资源非常之大所以此处不在应用机上分别使用logstash）
2. Logstash（或集群）接受到日志后解析并构造任意文本，即过滤日志，比如我们需要加一行内容或修改对应键值（ps：格式内容可自行定义或调换等等.......），解析完成后交给elasticsearch
3. Elasticsearch（或集群）主要用于存储日志数据，从logstash接受到数据并存储，连接kibana
4. Kibana将会展示elasticsearch中存储的数据，以web可视化的方式进行展示（ps：其中的绘图功能类似于grafana）

### 高级架构图解



1. 此架构我们在filebeat和logstash之间加入了kafka集群
2. Kafka集群加入后，filebeat将数据push到kafka，但是这里logstash开始从kafka集群pull数据，这里kafka起到了大量数据及压力缓冲的作用，对于集群性能优化起到了非常大的作用（ps：此处也是为了防止在网络堵塞的时候数据丢失，因为此时数据还未到elasticsearch内存储）
3. 其他功能与基本架构图解中相同

# 各类服务安装方法

### 注意事项

除jdk外所有服务的版本需一致，这里我用的是6.7.1版本，而且都不要用root用户去启动，这里我都是用的web用户去启动

这里所有的服务都是java写的，所以要依赖jvm虚拟机，我们给所有机器都下载并安装jdk，下载网址：<https://www.oracle.com/technetwork/java/javase/downloads/index.html>

我使用的版本是jdk-8u91,大家也可以使用openjdk

# tar -xf /data/install/logstash-6.7.1.tar.gz -C /data/server

# echo ‘export JAVA\_HOME=/data/server/jdk1.8.0\_191’>> ~/.bash\_profile

# echo ‘export PATH=$JAVA\_HOME/bin:$PATH’>> ~/.bash\_profile

# source ~/.bash\_profile

这样就能读到jdk的环境变量了

### Logstash

下载地址：<https://www.elastic.co/cn/downloads/logstash>

这里下载tar.gz文件，下载的是编译好的文件压缩包，直接解压即可

# tar -xf /data/install/logstash-6.7.1.tar.gz -C /data/server/

# vim /data/server/logstash-6.7.1/config/logstash.yml

找到下列字段并修改：

path.data: /data/server/logstash-6.7.1/data

path.config: /data/server/logstash-6.7.1/conf.d

path.logs: /data/logs/logstash

然后创建相关目录

# mkdir /data/server/logstash-6.7.1/conf.d

# mkdir /data/logs/logstash

编辑子配置文件，也就是过滤日志的配置文件

# vim /data/server/logstash-6.7.1/conf.d/test.conf

input {

beats {

port => 5044 #这里指定接受数据的端口

ssl => false

}

}

filter {

mutate {

remove\_field => ["@version"] #过滤掉不需要的参数，这个大家可以根据实际情况来做

remove\_field => ["beat"]

remove\_field => ["host"]

remove\_field => ["prospector"]

remove\_field => ["tags"]

remove\_field => ["\_id"]

remove\_field => ["\_type"]

remove\_field => ["input"]

remove\_field => ["offset"]

remove\_field => ["\_score"]

remove\_field => ["log"]

}

}

output { #发送到es，判断日志类型指定索引

if [fields][logtype] == "246log" {

#if [fields] == "246log" {

elasticsearch {

hosts => ["10.0.1.191:9200"]

manage\_template => false

index => "246-log-%{+YYYY.MM}"

}

}

if [fields][logtype] == "245log" {

elasticsearch {

hosts => ["10.0.1.191:9200"] #写es服务器地址

manage\_template => false

index => "245-log-%{+YYYY.MM}"

}

}

}

#chown -R web. /data/server/logstash-6.7.1/

启动logstash并让程序后台运行

# nohup /data/server/logstash-6.7.1/bin/logstash -f /data/server/logstash-6.7.1/conf.d/test.conf &

### Elasticsearch

下载地址：<https://www.elastic.co/cn/downloads/elasticsearch>

这里下载LINUX的选项，也是被编译好的包，解包即可

# tar -xf /data/install/elasticsearch-6.7.1.tar.gz -C /data/server

# vim /data/server/elasticsearch-6.7.1/config/elasticsearch.yml

找到下列字段并添加或修改：

bootstrap.memory\_lock: false

bootstrap.system\_call\_filter: false

network.host: 0.0.0.0

http.port: 9200

# vim /etc/security/limits.conf

在文件末尾加上：  
web soft nofile 65536

web hard nofile 65536

web soft nproc 4096

web hard nproc 4096

# vim /etc/security/limits.d/20-nproc.onf

修改成下面样子  
web soft nproc 4096 #这里为你启动es的用户

root soft nproc unlimited

# vim /etc/sysctl.conf

修改或添加下面内容

vm.max\_map\_count = 655360

# sysctl -p

# reboot

创建相关文件夹

# mkdir /data/server/elasticsearch-6.7.1/data

# chown -R web. /data/server/elasticsearch-6.7.1

启动es

# nohup /data/server/elasticsearch-6.7.1/bin/elasticsearch &

### Kibana

下载地址：<https://www.elastic.co/cn/downloads/kibana>

这里下载LINUX 64-BIT选项，也是编译好的包

# tar -xf /data/install/kibana-6.7.1.tar.gz -C /data/server

# vim /data/server/kibana-6.7.1-linux-x86\_64/config/kibana.yml

修改文件内容为:

server.port: 5601

server.host: "0.0.0.0"

elasticsearch.hosts: ["http://10.0.1.191:9200"] #填es主机ip

启动kibana

# nohup /data/server/kibana-6.7.1-linux-x86\_64/bin/kibana &

### Filebeat

下载地址：<https://www.elastic.co/downloads/beats/filebeat>

# tar -xf /data/install/filebeat-6.7.1-linux-x86\_64.tar.gz -C /data/server

# vim /data/server/filebeat-6.7.1-linux-x86\_64/filebeat.yml

修改或添加下列内容

- type: log

# Change to true to enable this input configuration.

enabled: true

# Paths that should be crawled and fetched. Glob based paths.

paths:

- /var/log/\*.log

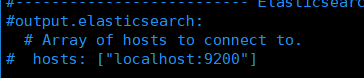
document\_type: "245log"

fields:

logsource: Test-245

logtype: 245log

将以下内容都注释掉



在output.logstash处解开注释

output.logstash:

# The Logstash hosts

hosts: ["10.0.1.190:5044"] #填logstash的主机ip

启动filebeat

# nohup /data/server/filebeat-6.7.1-linux-x86\_64/filebeat &