# ELK 搭建

Elasticsearch 重点，存储空间 分布式集群

Logstash 根据压力负载情况，多节点

Kibana 单节点

ELK 是一个实时分布式的日志分析平台

ELK 是一整套的解决方案

（E）lasticsearch -- 数据库

（L）ogstash -- 收集日志、标准化的程序

（K）ibana -- 图形的展示工具

# Elasticsearch 组建

1 禁用 selinux， 卸载 firewalld (所有主机都要做)

[root@room9pc19 doc]# cat /etc/selinux/config

SELINUX=disabled

禁用防火墙 (所有主机都要做)

systemctl stop firewalld

systemctl mask firewalld

2 配置 /etc/hosts (所有主机都要做)

192.168.1.11 es1

192.168.1.12 es2

192.168.1.13 es3

192.168.1.14 es4

192.168.1.15 es5

3 安装 java 运行环境 java-1.8.0-openjdk (所有主机都要做)

yum install java-1.8.0-openjdk -y

4 安装 elasticsearch (所有主机都要做)

yum install elasticsearch-2.3.4.rpm

5 修改配置文件 /etc/elasticsearch/elasticsearch.yml (所有主机都要做)

cluster.name: nsd1804

node.name: 当前主机名称

network.host: 0.0.0.0

discovery.zen.ping.unicast.hosts: ["es1", "es2", "es3"]

启动并验证 (所有主机都要做)

systemctl start elasticsearch

systemctl enable elasticsearch

浏览器访问

http://192.168.1.ip:9200/

http://192.168.1.ip:9200/\_cluster/health?pretty

# 集群插件

/usr/share/elasticsearch/bin

下载插件文件到本地，必须使用 file:// 绝对路径安装(如果下载到本地安装的话)

./plugin install file:///usr/share/elasticsearch/bin/bigdesk-master.zip

插件安装

/usr/share/elasticsearch/bin/plugin install ftp://192.168.1.254/public/bigdesk-master.zip

/usr/share/elasticsearch/bin/plugin install ftp://192.168.1.254/public/elasticsearch-head-master.zip

/usr/share/elasticsearch/bin/plugin install ftp://192.168.1.254/public/elasticsearch-kopf-master.zip

安装完成以后使用 ./plugin list 查看

访问插件 head

http://192.168.1.11:9200/\_plugin/head

访问插件 kopf

http://192.168.1.11:9200/\_plugin/kopf

访问插件 bigdesk

http://192.168.1.11:9200/\_plugin/bigdesk

集群 api 查询地址

http://192.168.1.15:9200/\_cat

http://192.168.1.15:9200/\_cat/nodes?v 显示详细信息

http://192.168.1.15:9200/\_cat/nodes?help 显示帮助信息

ES 数据库基本操作，使用 curl 命令

**创建索引**

curl -XPUT 'http://192.168.1.13:9200/tarena/' -d '{

"settings":{

"index":{

"number\_of\_shards": 5,

"number\_of\_replicas": 1

}

}

}'

**增加数据 PUT**

curl -XPUT 'http://192.168.1.11:9200/nsd1804/teacher/1' -d '{

"title": "阶段1",

"name":{"first": "小逗比", "last": "牛犇"},

"age": 25

}'

curl -XPUT 'http://192.168.1.11:9200/nsd1804/teacher/2' -d '{

"title": "阶段2",

"name":{"first": "老逗比", "last": "丁丁"},

"age": 52

}'

curl -XPUT 'http://192.168.1.11:9200/nsd1804/teacher/3' -d '{

"title": "阶段3",

"name":{"first": "漂亮姐", "last": "静静"},

"age": 20

}'

curl -XPUT 'http://192.168.1.11:9200/nsd1804/teacher/4' -d '{

"title": "阶段4",

"name":{"first": "老司机", "last": "欣欣"},

"age": 35

}'

**更改数据 POST**

curl -XPOST 'http://192.168.1.11:9200/nsd1804/teacher/3/\_update' -d '{

"doc":{

"age": 18

}

}'

查询

curl -XGET 'http://192.168.1.14:9200/nsd1804/teacher/1'

[root@es1~]# curl 'http://192.168.1.14:9200/nsd1804/teacher/1?pretty'

#**?pretty**表示转化一个容易看懂的格式

删除

curl -XDELETE "http://192.168.1.14:9200/nsd1804/teacher/1?pretty"

**http://192.168.1.18:5601/app/kibana#/**

Kibana配置的页面地址

进入后输入 apachelog 点击 Create确认,点击Discover、Visualize（在Pie cheart 里配图）、Dashboard分别有不同的功能

kibana 配置

vim /opt/kibana/config/kibana.yml

2 server.port: 5601 #这个端口被写死，不能更改！

5 server.host: "0.0.0.0"

15 elasticsearch.url: "http://es1:9200" #值修改这一项，其他的去掉#

23 kibana.index: ".kibana"

26 kibana.defaultAppId: "discover"

53 elasticsearch.pingTimeout: 1500

57 elasticsearch.requestTimeout: 30000

64 elasticsearch.startupTimeout: 5000

[root@kibana ~]# systemctl start kibana #启动

## 数据批量导入

[root@kibana ~]# lftp 192.168.1.254 #将数据下载到本地

lftp 192.168.1.254:~> cd /public

lftp 192.168.1.254:/public> ls

lftp 192.168.1.254:/public> get shakespeare.json.gz

lftp 192.168.1.254:/public> get accounts.json.gz

lftp 192.168.1.254:/public> get logs.jsonl.gz

[root@kibana ~]# gzip -d shakespeare.json.gz #解压数据

[root@kibana ~]# gzip -d logs.jsonl.gz

[root@kibana ~]# gzip -d accounts.json.gz

**-X** 导入使用的方法 POST

**--data-binary**  导入数据的格式

**@urfile**  导入数据的文件名

**\_bulk** 导入关键字

**有 index, type, id 的导入**

curl -XPOST http://192.168.1.13:9200/\_bulk --data-binary @shakespeare.json **#数据导入**

**如果没有 index 和 type ，我们需要自己指定一下 index 和 type**

[root@kibana ~]# curl -XPOST http://192.168.1.13:9200/00/xx/\_bulk --data-binary @accounts.json

**有 多个index,type 无 id 的导入**

curl -XPOST http://192.168.1.13:9200/\_bulk --data-binary @logs.jsonl

## 数据批量查询

**查询一条数据**

curl -X "GET" "http://192.168.1.12:9200/haha/xixi/1"

**查询多条数据，使用 \_mget**

curl -XGET http://192.168.1.12:9200/\_mget?pretty -d '

{

docs:[

{"\_index": "oo",

"\_type": "xx",

"\_id": 99

},

{"\_index": "shakespeare",

"\_type": "act",

"\_id": 80730

},

{"\_index": "logstash-2015.05.20",

"\_type": "log",

"\_id": "AWTo2xXm16RGslV6jxJR"

}

]

}'

**在kibana上安装一个apache**

[root@kibana ~]# yum -y install httpd

[root@kibana ~]# cd /var/www/html/

[root@kibana html]# curl http://www.baidu.com -o index.html

#下载百度页面

[root@kibana html]# vim /etc/httpd/conf/httpd.conf

ServerName localhost #修改这一行

[root@kibana html]# apachectl -t

Syntax OK

[root@kibana html]# apachectl start

## logstash 的安装

**安装依赖包 openjdk**

yum install java-1.8.0-openjdk -y

yum install logstash-2.3.4-1.noarch.rpm -y

**创建logstash配置文件**

[root@logstash ~]# cd /etc/logstash/

[root@logstash logstash]# touch logstash.conf

[root@logstash logstash]# /opt/logstash/bin/logstash-plugin list #查看插件

**ELK 工作结构模型**

+-----------------logstash-------------------+

+--------+ | +--------+ +---------+ +-----------+ | +---------+ +----------+

| 数据源 | --->| INPUT | -->| FILTER | --> | OUTPUT | ---> | ES 集群 | -->| KIBANA |

+--------+ | +--------+ +---------+ +-----------+ | +---------+ +----------+

+------------------------------------------------+

**logstash.conf 初始配置**

input{

stdin{}

}

filter{

}

output{

stdout{}

}

**codec 插件**

stdout{ codec => "rubydebug" }

**file 插件**

file{

sincedb\_path => "/var/lib/logstash/since.db"

start\_position => "beginning"

path => ["/var/tmp/a.log", "/tmp/b.log"]

type => "filelog"

}

**logstash 配置文件** logstash.conf

input{

stdin{ codec => "json" }

}

filter{

}

output{

stdout{ codec => "rubydebug" }

}

[root@logstash logstash]# /opt/logstash/bin/logstash -f logstash.conf

**#解析测试**

源码地址 https://github.com/logstash-plugins

文档地址 https://www.elastic.co/guide/en/logstash/current/index.html

**file 模块**

input {

... ...

file {

path => ["/tmp/a.log", "/tmp/b.log"]

sincedb\_path => "/var/lib/logstash/sincedb.log"#声明读取文件的位置

start\_position => "beginning" #配置第一次读取文件从什么地方开始

type => "filelog"

}

}

[root@logstash logstash]# rm -f /var/lib/logstash/sincedb.log

#删除这个文件才会重新读取

[root@logstash logstash]# /opt/logstash/bin/logstash -f logstash.conf

**tcp & udp 模块**

input {

... ...

tcp {

mode => "server"

host => "0.0.0.0"

port => 8888

type => "tcplog"

}

udp {

port => 9999

type => "udplog"

}

}

测试命令(**网络重定向**)

echo "test udp log" >/dev/udp/192.168.1.20/9999 #发送 udp 数据

echo "test tcp log" >/dev/tcp/192.168.1.20/8888 #发送 tcp 数据

**syslog 模块**

input {

... ...

syslog {

host => "0.0.0.0"

port => 514

type => "syslog"

}

}

[root@logstash logstash]# /opt/logstash/bin/logstash -f logstash.conf

**#等待客户端测试结果**

客户机配置 @@(tcp) @(udp)

[root@kibana log]# vim /etc/rsyslog.conf

local0.info /var/log/info.log

local0.info @@192.168.1.20:514

[root@kibana log]# systemctl restart rsyslog.service

命令

logger -p local0.info -t "testlog" "hello world" #测试

[root@kibana log]# cat info.log

**正则表达式分组匹配 (?<name>reg)**

**正则宏路径**

/opt/logstash/vendor/bundle/jruby/1.9/gems/logstash-patterns-core-2.0.5/patterns

**filter 配置，解析标准 apache 日志**

filter{

grok{

match => ["message", "%{COMBINEDAPACHELOG}"] #apache日志的宏

}

}

**output 输出到 Elasticsearch**

增加类型判断，细化区分日志

output{

if [type] == "apachelog"{

elasticsearch {

hosts => ["192.168.1.15:9200", "192.168.1.11:9200"]

index => "apachelog"

flush\_size => 2000

idle\_flush\_time => 10

}}

}

**使用 filebeat 收集日志，发送到 logstash**

**logstash beats 配置**

input{

... ...

beats{

port => 5044

}

}

**客户端配置**

[root@kibana ~]# yum -y install filebeat

[root@kibana ~]# vim /etc/filebeat/filebeat.yml

15 - /var/log/httpd/access\_log #apache配置文件路径

72 document\_type: apachelog #声明文档类型

183 #elasticsearch:

188 #hosts: ["localhost:9200"]

278 logstash:

280 hosts: ["192.168.1.20:5044"]

[root@kibana ~]# grep -Pv "^\s\*(#|$)" /etc/filebeat/filebeat.yml

#查看有效行

[root@kibana ~]# systemctl start filebeat #启动服务

**最终版**

input{

stdin{ codec => "json" }

beats{

port => 5044

}

file {

path => ["/tmp/a.log"]

sincedb\_path => "/var/lib/logstash/sincedb.log"

start\_position => "beginning"

type => "filelog"

}

tcp {

mode => "server"

host => "0.0.0.0"

port => 9999

type => "tcplog"

}

udp {

port => 8888

type => "udplog"

}

syslog {

host => "0.0.0.0"

port => 514

type => "syslog"

}

}

filter{

if [type] == "apachelog"{

grok{

match => ["message", "%{COMBINEDAPACHELOG}"]

}}

}

output{

stdout{ codec => "rubydebug" }

if [type] == "apachelog"{

elasticsearch {

hosts => ["192.168.1.15:9200", "192.168.1.11:9200"]

index => "apachelog"

flush\_size => 2000

idle\_flush\_time => 10

}}

}

[root@kibana~]#curl -XDELETE [http://192.168.1.11:9200/\*](http://192.168.1.11:9200/*) #删除（可不做！）

[root@kibana ~]# systemctl restart kibana

在192.168.1.18的百度页面多次刷新访问，然后在http://192.168.1.11:9200/\_plugin/head/ 上查看数据（可能要等一会）