Elk 6.0 部署

卸载系统自带的openjdk

rpm -e java-1.7.0-openjdk-1.7.0.79-2.5.5.4.el6.x86\_64 –nodeps

安装jdk

tar xvf jdk-8u151-linux-x64.tar.gz -C /usr/local/

Vim /etc/profile （配置java环境变量）

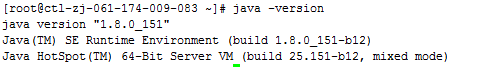
JAVA\_HOME=/usr/local/jdk1.8.0\_151

CLASSPATH=.:$JAVA\_HOME/lib/tools.jar:$JAVA\_HOME/lib/dt.jar

PATH=$JAVA\_HOME/bin:$PATH

export JAVA\_HOME CLASSPATH PATH

source /etc/profile



安装依赖

yum install -y autoconf make automake imake cmake gcc gcc-c++ libaio libaio-devel bzr bison expat-devel libtool ncurses5-devel gd-devel libjpeg-devel libpng-devel libxml2-devel bzip2-devel libcurl-devel pcre\* openssl\* epel-release

安装nginx

tar xvf nginx-1.13.4.tar.gz -C /usr/local/

mv nginx-1.13.4/ nginx

useradd www -s /sbin/nologin

cd /usr/local/nginx/

./configure --user=www --group=www --prefix=/usr/local/nginx --conf-path=/usr/local/nginx/nginx.conf --with-http\_stub\_status\_module --with-http\_sub\_module --with-http\_ssl\_module --with-pcre

make -j 8 && make install

安装es

Useradd elk

tar xvf elasticsearch-6.0.0.tar.gz -C /usr/local/

vim /usr/local/elasticsearch-6.0.0/config/elasticsearch.yml (修改以下内容)

cluster.name: "estest" #集群名称

node.name: xiaoniu-1 #该节点在集群内的名称

node.data: true

http.enabled: true

path.data: /data/elk/es/data #数据存放路径

path.logs: /data/elk/es/logs #日志存放路径

network.host: 61.174.9.83 #对外服务地址

http.port: 9200 #占用端口

http.cors.enabled: true

http.cors.allow-origin: "\*"

discovery.zen.ping.unicast.hosts: ["61.174.9.83"] #设置自动发现的节点

discovery.zen.minimum\_master\_nodes: 1 #设置这个参数来保证集群中的节点可以知道其它N个有master资格的节点

gateway.recover\_after\_nodes: 1 #在多少个节点启动后，允许数据恢复进程启动

xpack.security.enabled: false

mkdir –p /data/elk/es/logs

mkdir –p /data/elk/es/data

chown –R elk /data/elk （变更目录拥有人）

chown –R elk /usr/local/elasticsearch-6.0.0 （变更目录拥有人）

安装kibana

tar xvf kibana-6.0.0-linux-x86\_64.tar.gz -C /usr/local/

vim /usr/local/kibana-6.0.0-linux-x86\_64/config/kibana.yml (修改以下内容)

server.port: 56123 #服务运行端口

server.host: "61.174.9.83" #服务运行地址

server.maxPayloadBytes: 1048576 #最大请求负载

server.name: "xiaoniu-kibana"

elasticsearch.url: "<http://61.174.9.83:9200>" # es的连接地址

elasticsearch.preserveHost: true #当设置的值为true时使用server.host属性对应的hostname。当设置值为false,Kibana使用连接的Kibana实例对应的hostname

kibana.defaultAppId: "discover" #默认加载的应用

xpack.monitoring.elasticsearch.url: "http://61.174.9.83:9200" (安装xpack后需此配置)

xpack.monitoring.elasticsearch.username: "elastic"

xpack.monitoring.elasticsearch.password: "changeme"

chown –R elk /usr/local/kibana-6.0.0-linux-x86\_64 （变更目录拥有人）

安装logstash

tar xvf logstash-6.0.0.tar.gz -C /usr/local/

vim /usr/local/logstash-6.0.0/config/logstash.yml (修改以下配置)

path.data: /data/elk/logstash/data

path.config: /etc/logstash/conf.d

http.host: "127.0.0.1"

path.logs: /data/elk/logstash/logs

mkdir -p /etc/logstash/conf.d

mkdir –p /data/elk/logstash/data

mkdir –p /data/elk/logstash/logs

chown –R elk /usr/loca/logstash-6.0.0 （变更目录拥有人）

chown –R elk /etc/logstash （变更目录拥有人）

vim /etc/logstash/conf.d/logstash.conf (修改以下配置)

input {

beats {

port => 5044

client\_inactivity\_timeout => 60 多长时间之后关闭空闲的连接

codec => json {

charset => "UTF-8"

}

}

}

output {

if [type] == 'account\_errorlog'{

elasticsearch {

hosts => ["61.174.9.83:9200"]

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

}

}

if [type] == 'account\_log'{

elasticsearch {

hosts => ["61.174.9.83:9200"]

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

}

}

}

Logstash 完整配置

input {

beats {

port => 5044

client\_inactivity\_timeout => 60

codec => json {

charset => "UTF-8"

}

}

}

filter {

if [clientip] !~ "^127\.|^192\.168\.|^172\.1[6-9]\.|^172\.2[0-9]\.|^172\.3[01]\.|^10\." { #排除私网地址

geoip {

source => "clientip" #设置解析IP地址的字段

target => "city" #将geoip数据保存到一个字段内

database => "/usr/local/share/GeoIP/GeoLite2-City.mmdb" #IP地址数据库

fields => ["country\_name", "city\_name", "latitude", "longitude", "region\_name"] #输出哪些字段

}

}

mutate { # 添加字段

add\_field => {

"blockid" => "%{ruleid}"

}

}

if [blockid] == "12001" { mutate { replace => { "blockid" => "0" } } }

if [blockid] == "12002" { mutate { replace => { "blockid" => "0" } } }

if [blockid] == "12003" { mutate { replace => { "blockid" => "0" } } }

if [blockid] == "12004" { mutate { replace => { "blockid" => "0" } } }

if [ruleid] == "0" { mutate { replace => { "ruleid" => "normal" } } }

if [ruleid] == "11037" { mutate { replace => { "ruleid" => "sql" } } }

if [ruleid] == "11040" { mutate { replace => { "ruleid" => "sql" } } }

if [ruleid] == "11041" { mutate { replace => { "ruleid" => "sql" } } }

if [ruleid] == "11042" { mutate { replace => { "ruleid" => "sql" } } }

if [ruleid] == "2" { mutate { replace => { "ruleid" => "xss" } } }

if [ruleid] == "30001" { mutate { replace => { "ruleid" => "rce" } } }

if [ruleid] == "4" { mutate { replace => { "ruleid" => "cmd" } } }

if [ruleid] == "5" { mutate { replace => { "ruleid" => "scan" } } }

if [ruleid] == "50020" { mutate { replace => { "ruleid" => "scan" } } }

if [ruleid] == "55003" { mutate { replace => { "ruleid" => "scan" } } }

if [ruleid] == "6" { mutate { replace => { "ruleid" => "dir" } } }

if [ruleid] == "7" { mutate { replace => { "ruleid" => "blackip" } } }

if [ruleid] == "8" { mutate { replace => { "ruleid" => "cc" } } }

if [ruleid] == "9" { mutate { replace => { "ruleid" => "other" } } }

if [ruleid] == "12001" { mutate { replace => { "ruleid" => "white" } } }

if [ruleid] == "12002" { mutate { replace => { "ruleid" => "white" } } }

if [ruleid] == "12003" { mutate { replace => { "ruleid" => "white" } } }

if [ruleid] == "12004" { mutate { replace => { "ruleid" => "white" } } }

}

output {

elasticsearch {

hosts => ["61.174.9.83:9200"]

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

loadbalance: true

user => elastic

password => changeme

manage\_template => true

}

}

安装filebeat

rpm -ivh filebeat-6.0.0-x86\_64.rpm

vim /etc/filebeat/filebeat.yml （修改以下配置）

filebeat.prospectors:

- input\_type: log

paths:

- /cache/logs/\*access.log

document\_type: accesslog

output.logstash:

hosts: ["61.174.9.83:5044","61.174.9.83:5045"]

loadbalance: true

如果设置为false输出插件发送所有事件到随机的一台主机上如果选择的不可达将切换到另一台主机。默认是false。如果设置为TRUE和配置了多台logstash主机输出插件将负载均衡的发布事件到所有logstash主机。

#loadbalance: true

安装过程报错

ERROR: [3] bootstrap checks failed

[1]: max number of threads [1024] for user [elk] is too low, increase to at least [4096]

[2]: max virtual memory areas vm.max\_map\_count [65530] is too low, increase to at least [262144]

http://blog.csdn.net/liangzhao\_jay/article/details/56840941

[3]: system call filters failed to install; check the logs and fix your configuration or disable system call filters at your own risk

报错解决方法

<http://blog.csdn.net/liangzhao_jay/article/details/56840941>

/usr/local/logstash-6.0.0/bin/logstash --path.settings /usr/local/logstash-6.0.0/conf/ -f /etc/logstash/conf.d/logstash.conf --path.data /data/elk/logstash/data1 --path.log /data/elk/logstash/logs/logstash1 (启动)

<https://www.linuxea.com/1717.html> x-pack 破解

<https://www.elastic.co/guide/en/x-pack/current/installing-xpack.html> 安装x-pack

bin/logstash-plugin install <file:///path/to/file/x-pack-6.0.0.zip>

xpack.monitoring.elasticsearch.url: "http://61.174.9.83:9200"

xpack.monitoring.elasticsearch.username: "elastic"

xpack.monitoring.elasticsearch.password: "changeme"

bin/kibana-plugin install <file:///path/to/file/x-pack-6.0.0.zip>

bin/elasticsearch-plugin install <file:///path/to/file/x-pack-6.0.0.zip>

xpack.security.enabled: false

如果不需要该插件，也可以通过非常简单的方式删除：  
bin/elasticsearch-plugin remove x-pack

# ELK利用GeoIP进行地理定位

http://dev.maxmind.com/geoip/legacy/geolite/

wget <http://geolite.maxmind.com/download/geoip/database/GeoLiteCity.dat.gz>

gzip -d GeoLiteCity.dat.gz

logstash-plugin install logstash-filter-geoip (默认已安装)

修改配置文件 /usr/local/etc/GeoIP.conf 中 ProductIds GeoLite2-City ，然后直接执行 geoipupdate ，便会自动下载并校验数据库文件。默认数据库文件目录为： /usr/local/share/GeoIP ，可以通过配置项 DatabaseDirectory /etc/logstash/ 更改数据库文件目录。

0 12 \* \* 3 /usr/local/bin/geoipupdate (设置计划任务自动更新ip库)

curl -XGET http://61.174.14.194:9200/\_cat/shards | grep UNASSIGNED 查看不可用的索引

curl -XGET <http://61.174.14.194:9200/_cat/nodes> 查看集群内节点数量

## x-pack 破解

**创建LicenseVerifier.java**文件

[root@linuxea.com-Node61 /elk/]# cat LicenseVerifier.java

package org.elasticsearch.license;

import java.nio.\*;

import java.util.\*;

import java.security.\*;

import org.elasticsearch.common.xcontent.\*;

import org.apache.lucene.util.\*;

import org.elasticsearch.common.io.\*;

import java.io.\*;

public class LicenseVerifier

{

public static boolean verifyLicense(final License license, final byte[] encryptedPublicKeyData) {

return true;

}

public static boolean verifyLicense(final License license) {

return true;

}

}

**编译class文件**

[root@linuxea.com-Node49 ~/elk]# javac -cp "/usr/share/elasticsearch/lib/elasticsearch-5.5.1.jar:/usr/share/elasticsearch/lib/lucene-core-6.6.0.jar:/usr/share/elasticsearch/plugins/x-pack/x-pack-5.5.1.jar" LicenseVerifier.java

**解压**

[root@linuxea.com-Node49 /usr/share/elasticsearch/plugins/x-pack/test]# jar xvf x-pack-5.5.1.jar

**替换class**

[root@linuxea.com-Node49 /usr/share/elasticsearch/plugins/x-pack/test]# cd org/elasticsearch/license

**打包**

[root@linuxea.com-Node49 /usr/share/elasticsearch/plugins/x-pack/test/org/elasticsearch/license]# cd /usr/share/elasticsearch/plugins/x-pack/test/

[root@linuxea.com-Node49 /usr/share/elasticsearch/plugins/x-pack/test]# jar cvf x-pack-5.5.1.jar .

**将打包好的文件放回x-pack目录下**

[root@linuxea.com-Node49 /usr/share/elasticsearch/plugins/x-pack/test]# cp x-pack-5.5.1.jar ../

**申请license**

<https://license.elastic.co/registration>

**申请一个license后会发到邮箱，然后修改下即可**

{"license":{"uid":"d13W1FM-ef9XWi-45eAKLH6-afT5b4-b8erC7460","type":"platinum","issue\_date\_in\_millis":11042324000000,"expiry\_date\_in\_millis":2535123399999,"max\_nodes":100,"issued\_to":"sean wang (alibaba)","issuer":"Web Form","signature":"","start\_date\_in\_millis":1504224000000}}

**输入密码进行修改：**

[root@linuxea.com-Node49 ~/elk]# curl -XPUT -u elastic 'http://10.0.1.49:9200/\_xpack/license' -H "Content-Type: application/json" -d @license.json Enter host password for user 'elastic': {"acknowledged":true,"license\_status":"valid"}

**修改完成后查看**

[root@linuxea.com-Node49 ~/elk]# curl -XPUT -u elastic 'http://10.0.1.49:9200/\_xpack/license' -H "Content-Type: application/json"curl -XGET -u elastic:linuxea 'http://10.0.1.49:9200/\_license'

{

"license" : {

"status" : "active",

"uid" : "d13W1FM-ef9XWi-45eAKLH6-afT5b4-b8erC7460",

"type" : "platinum",

"issue\_date" : "2017-09-01T00:00:00.000Z",

"issue\_date\_in\_millis" : 11042324000000,

"expiry\_date" : "2050-05-11T01:46:39.999Z",

"expiry\_date\_in\_millis" : 2535123399999,

"max\_nodes" : 100,

"issued\_to" : "sean wang (alibaba)",

"issuer" : "Web Form",

"start\_date\_in\_millis" : 11042324000000

}

}