# 1 elasticStack安装

## 介绍

### 下载地址

|  |
| --- |
| elastic:  https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.5.1.zip  https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.5.1.tar.gz  kibana:  https://artifacts.elastic.co/downloads/kibana/kibana-5.5.1-linux-x86\_64.tar.gz  filebeat:  https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-5.5.1-linux-x86\_64.tar.gz  logstash:  https://artifacts.elastic.co/downloads/logstash/logstash-5.5.1.tar.gz  elasticsearch-hadoop:  http://download.elastic.co/hadoop/elasticsearch-hadoop-5.5.1.zip |

## Linux环境ElasticStack的安装(集群规模5个节点)

### 准备工作

Linux shell limits和sysctl内核设置

|  |
| --- |
| echo "elk - nofile 65536">>/etc/security/limits.conf  sysctl -w vm.max\_map\_count=262144  sysctl -p  echo "elk soft memlock unlimited">>/etc/security/limits.conf  echo "elk hard memlock unlimited">>/etc/security/limits.conf  注:否则报错Unable to lock JVM Memory: error=12, reason=Cannot allocate memory |

JDK安装

|  |
| --- |
| #卸载现有JDK  rpm -qa | grep jdk  yum erase java-1.7.0-openjdk  ##配置/etc/profile文件.否则同时配置.bash\_profile和.bahsrc文件  export JAVA\_HOME=/app/elk/jdk1.8.0\_121  export CLASS\_PATH=.:$JAVA\_HOME/lib/tools.jar:$JAVA\_HOME/lib/dt.jar  export PATH=$PATH:$JAVA\_HOME/bin:$JAVA\_HOME/jre/bin |

Linux的SSH互信

|  |
| --- |
| # 192.168.11.73/74/75/76/77配置互信  # elk用户:/app/elk/.ssh  # 发送所有shell  ssh-keygen -t rsa  cp ~/.ssh/id\_rsa.pub ~/.ssh/authorized\_keys2\_`hostname`  chmod 700 ~/.ssh  chmod 600 ~/.ssh/\*  #除了192.168.11.73, 单个节点执行  scp ~/.ssh/authorized\_keys2\_`hostname` 192.168.11.73:~/.ssh/  #192.168.11.73执行  cd ~/.ssh/  cat authorized\_keys2\_bjxhm-hadoopelk-73 >> authorized\_keys  cat authorized\_keys2\_bjxhm-hadoopelk-74 >> authorized\_keys  cat authorized\_keys2\_bjxhm-hadoopelk-75 >> authorized\_keys  cat authorized\_keys2\_bjxhm-hadoopelk-76 >> authorized\_keys  cat authorized\_keys2\_bjxhm-hadoopelk-77 >> authorized\_keys  #复制到其他节点  scp ~/.ssh/authorized\_keys 192.168.11.74:~/.ssh/  scp ~/.ssh/authorized\_keys 192.168.11.75:~/.ssh/  scp ~/.ssh/authorized\_keys 192.168.11.76:~/.ssh/  scp ~/.ssh/authorized\_keys 192.168.11.77:~/.ssh/  ##每个节点执行测试  ssh 192.168.11.73 'uptime'  ssh 192.168.11.74 'uptime'  ssh 192.168.11.75 'uptime'  ssh 192.168.11.76 'uptime'  ssh 192.168.11.77 'uptime' |

### 开始搭建Linux的elasticsearch环境

解压

|  |
| --- |
| mkdir elastic  mkdir -p /app/elk/elastic/elastic\_data  mkdir -p /app/elk/elastic/elastic\_log  tar -zxvf elasticsearch-5.5.1.tar.gz -C elastic |

配置jvm option

|  |
| --- |
| (1)JVM heap size:  -Xms2g  -Xmx2g  (2)内存回收  ## GC configuration  -XX:+UseConcMarkSweepGC  -XX:CMSInitiatingOccupancyFraction=75  -XX:+UseCMSInitiatingOccupancyOnly |

elasticsearch 配置

|  |
| --- |
| (1)集群名称  cluster.name: unicompayment-bigdata  (2)节点名称**(五个节点配置分别为)**  node.name: node-1  node.name: node-2  node.name: node-3  node.name: node-4  node.name: node-5  (3)存储日志和数据,数据路径可以是separate multiple locations by comma  path.data: /app/elk/elastic/elastic\_data  path.logs: /app/elk/elastic/elastic\_log  (4)绑定IP地址,绑定多个IP地址,设置端口号**(五个节点配置分别为)**  network.host: ["192.168.11.73","127.0.0.1"]  network.host: ["192.168.11.74","127.0.0.1"]  network.host: ["192.168.11.75","127.0.0.1"]  network.host: ["192.168.11.76","127.0.0.1"]  network.host: ["192.168.11.77","127.0.0.1"]  http.port: 9200  (5)发现和初始化集群内的节点列表:  discovery.zen.ping.unicast.hosts: ["192.168.11.73:9300","192.168.11.74:9300","192.168.11.75:9300","192.168.11.76:9300","192.168.11.77:9300"]  (6)防止脑裂To avoid a split brain:  discovery.zen.minimum\_master\_nodes: 3  (7)小的集群,master节点,可以同时是数据节点, 如果是大型集群需要区分.  # master节点  node.master: true  node.data: false  #数据节点  node.master: false  node.data: true  (8)  bootstrap.memory\_lock: true  (9)redhat6安装,需要  bootstrap.system\_call\_filter: false  注:redhat7安装不需要. |

将安装包分发到其他节点

|  |
| --- |
| scp -r ~/\* 192.168.11.74:~/  scp -r ~/\* 192.168.11.75:~/  scp -r ~/\* 192.168.11.76:~/  scp -r ~/\* 192.168.11.77:~/ |

### 1.2.3 注意事项

redhat6安装,需要

bootstrap.system\_call\_filter: false

注:redhat7安装不需要.

否则报错如下:

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

启动验证状态

GET localhost:9200 /

GET localhost:9200 /\_cluster/state

GET localhost:9200 /\_stats

GET localhost:9200 /\_nodes/stats

GET localhost:9200 /\_nodes

GET localhost:9200 /\_cluster/health

GET localhost:9200 /\_cluster/state

GET localhost:9200 /\_all

### 1.2.4 安装elasticsearch -head(有问题未解决,待验证)

安装node

|  |
| --- |
| tar xvJf node-v6.11.2-linux-x64.tar.xz  export PATH=$PATH:/app/elk/node-v6.11.2-linux-x64/bin |

下载elasticsearch-head

|  |
| --- |
| git clone git://github.com/mobz/elasticsearch-head.git |

安装

|  |
| --- |
| cd elasticsearch-head  npm install  npm run start  http://192.168.11.73:9100/  http://192.168.11.73:9200/\_plugin/head/ |

Proxy代理

|  |
| --- |
| [elk@bjxhm-hadoopelk-74 elasticsearch-head]$ npm run start  > elasticsearch-head@0.0.0 start /app/elk/elasticsearch-head  > grunt server  Running "connect:server" (connect) task  Waiting forever...  Started connect web server on http://localhost:9100  [elk@bjxhm-hadoopelk-74 elasticsearch-head]$ npm run proxy  > elasticsearch-head@0.0.0 proxy /app/elk/elasticsearch-head  > node proxy/index.js  creating proxy localhost:9200  remote: http://localhost:9200  local: <http://localhost:9101>  npm config delete proxy |

### 1.2.5 kibana配置

#端口

server.port: 5601

#可以被访问的IP: To allow connections from remote users

server.host: "192.168.11.76"

# 所有的查询访问ES: The URL of the Elasticsearch instance to use for all your queries.

elasticsearch.url: <http://192.168.11.76:9200>

#如果ES有基本认证,ES的用户名和密码

#elasticsearch.username: "user"

#elasticsearch.password: "pass"

elasticsearch.preserveHost: true

# Kibana 使用ES的索引来保存searches, visualizations and dashboards.

kibana.index: ".kibana"

#暂时不使用SSL认证

server.ssl.enabled: false

关于sense,kibana自带不需要安装.如果手工安装类似,但是报错.

|  |
| --- |
| mkdir sense  cp /app/elk/package/sense-2.0.0-beta7.tar.gz .  bin/kibana-plugin install <file:///app/elk/kibana-5.5.1-linux-x86_64/sense/sense-2.0.0-beta7.tar.gz>  Retrieving metadata from plugin archive  Error: end of central directory record signature not found  at /app/elk/kibana-5.5.1-linux-x86\_64/node\_modules/yauzl/index.js:179:14  at /app/elk/kibana-5.5.1-linux-x86\_64/node\_modules/yauzl/index.js:539:5  at /app/elk/kibana-5.5.1-linux-x86\_64/node\_modules/fd-slicer/index.js:32:7  at FSReqWrap.wrapper [as oncomplete] (fs.js:682:17)  Plugin installation was unsuccessful due to error "Error retrieving metadata from plugin archive" |

### 1.2.6 logstash的安装

配置.conf的文件

测试

bin/logstash -f config/full\_pipeline.conf --config.test\_and\_exit

启动

nohup bin/logstash -f config/webtends\_pipeline.conf --config.reload.automatic 2>&1 &

### 1.2.7 filebeat安装

#部署服务器

webtrends sdc服务器

# 端口占用情况

netstat -aon|findstr "5044"

配置config文件:

|  |
| --- |
| #输出,制定index名称  output.logstash:  # The Logstash hosts  hosts: ["192.168.11.77:5044"]  index: "webtrends "  #输出  - input\_type: log  # Paths that should be crawled and fetched. Glob based paths.  paths:  - D:\ProgramFiles(x86)\Webtrends\SmartSourceDataCollector\sdc\weblog\dcs4zpvp31v0kiag42qwkqc9o\_4w2v\_sdc\*.log  #添加域  fields:  project\_name: "webtrends " |

## Windows环境ElasticStack的安装

### 简介

filebeat服务监听 172.28.11.167:5044

filebeat启动 ./filebeat -c filebeat.yml -e

/filebeat -e -c filebeat.yml -d "publish"

logstash API endpoint 172.28.11.167:9600

logstash启动 bin\logstash -f filter-pipeline.conf --config.reload.automatic

elasticsearch服务监听 172.28.11.167:9200

elasticsearch启动 %ES\_HOME%\bin\elasticsearch.bat

kibana网页端口 172.28.11.167:5601

kibana启动 %KIBANA\_HOME%\bin\kibana.bat

elasticsearch-head

**file:///D:/installtools/ELK\_ElasticStack/elasticsearch-head/index.html?auth\_user=elastic&auth\_password=changeme**

### packetbeat的安装

准备

|  |
| --- |
| ############################  # 问题1：由于找不到wpcap.dll  安装Win10Pcap-v10.2-5002.msi  ############################ |

查看网卡设备

|  |
| --- |
| # 查看device  .\packetbeat.exe -devices  0: {} (VMware Virtual Ethernet Adapter) (192.168.13.1)  1: {} (Microsoft) (192.168.1.11)  2: {} (Sangfor SSL VPN CS Support System VNIC) (0.0.0.0)  3: {} (VMware Virtual Ethernet Adapter) (192.168.139.1)  4: {} (Microsoft) (0.0.0.0) |

Edit the packetbeat.yml configuration file

|  |
| --- |
| packetbeat.interfaces.device: 1  output.elasticsearch:  hosts: ["localhost:9200"]  template.name: "packetbeat"  template.path: "packetbeat.template.json"  template.overwrite: false |

启动

|  |
| --- |
| .\packetbeat -c .\packetbeat.yml -e |

To load the dashboards for packetbeat into Kibana, run:

|  |
| --- |
| .\scripts\import\_dashboards |

### winlogbeat.sh

|  |
| --- |
| # To load the dashboards for winlogbeat into Kibana, run:  ./scripts/import\_dashboards  # https://www.elastic.co/guide/en/beats/winlogbeat/5.5/index.html  # 日志  # application events  # hardware events  # security events  # system events  winlogbeat.event\_logs:  - name: Application  ignore\_older: 72h  - name: Security  - name: System  # output  output.elasticsearch:  hosts: ["localhost:9200"]  template.name: "winlogbeat"  template.path: "winlogbeat.template.json"  template.overwrite: false  # 启动  .\winlogbeat -c winlogbeat.yml -e |