<https://www.cnblogs.com/linjiqin/p/12103645.html>

ELK是一套完整的日志解决方案，由ElasticSearch、Logstash、 Kibana这三款开源软件组成。

docker search logstash

docker search logstash --filter "is-official=true"

//接取了最新版,不要用最新的 使用7.5.1成功

docker pull logstash

docker pull logstash:7.5.1

**~~2、编辑logstash.yml配置文件~~** ~~logstash.yml配置文件放在宿主机/data/elk/logstash目录下，内容如下：~~

~~path.config: /usr/share/logstash/conf.d/\*.conf~~

~~path.logs: /var/log/logstash~~

**~~3、编辑test.conf文件~~** ~~test.conf文件放在宿主机/data/elk/logstash/conf.d目录下，内容如下：~~

~~[复制代码](javascript:void(0);)~~

~~input {~~

~~beats {~~

~~port => 9601~~

~~codec => "json"~~

~~}~~

~~}~~

~~output {~~

~~elasticsearch {~~

~~hosts => ["172.17.0.3:9200"]~~

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

~~}~~

~~stdout { codec => rubydebug }~~

~~}~~

~~//使用最新版的logstash~~

~~docker run -d --restart=always --log-driver json-file --log-opt max-size=100m --log-opt max-file=2 -p 5044:5044 --name logstash -v /data/elk/logstash/logstash.yml:/usr/share/logstash/config/logstash.yml -v /data/elk/logstash/conf.d/:/usr/share/logstash/conf.d/ logstash~~

docker run -d --name=logstash logstash:7.5.1

docker cp logstash:/usr/share/logstash /data/elk/

mkdir /data/elk/logstash/config/conf.d

chmod 777 -R /data/elk/logstash

vi /data/elk/logstash/config/logstash.yml

http.host: "0.0.0.0"

xpack.monitoring.elasticsearch.hosts: [ "**http://172.17.0.3:9200**" ]

path.config: /usr/share/logstash/config/conf.d/\*.conf

path.logs: /usr/share/logstash/logs

vi /data/elk/logstash/config/conf.d/syslog.conf

~~input {~~

~~file {~~

~~#标签~~

~~type => "systemlog-localhost"~~

~~#采集点~~

~~path => "/var/log/messages"~~

~~#开始收集点~~

~~start\_position => "beginning"~~

~~#扫描间隔时间，默认是1s，建议5s~~

~~stat\_interval => "5"~~

~~}~~

~~}~~

~~output {~~

~~elasticsearch {~~

~~hosts => ["172.17.0.3:9200"]~~

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

~~}~~

~~}~~

input {

tcp {

##host:port就是上面appender中的 destination，这里其实把logstash作为服务，开启9601端口接收logback发出的消息

host => "0.0.0.0"

port => 9601

#模式选择为server

mode => "server"

tags => ["tags"]

##格式json

codec => json\_lines

}

}

output {

elasticsearch {

hosts => ["172.17.0.3:9200"]

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

}

}

设置日志文件读取权限

chmod 644 /var/log/messages

docker run -d \

--name=logstash \

--restart=always \

-p 9601:9601 -p 5044:5044 \

--privileged=true \

-v /data/elk/logstash:/usr/share/logstash \

logstash:7.5.1

~~docker run -d \~~

~~--name=logstash \~~

~~--restart=always \~~

~~-p 9601:9601 \~~

~~-v /data/elk/logstash:/usr/share/logstash \~~

~~logstash:7.5.1~~

~~docker run -d \~~

~~--name=logstash \~~

~~--restart=always \~~

~~-p 9601:9601 \~~

~~-v /data/elk/logstash:/usr/share/logstash \~~

~~-v /var/log/messages:/var/log/messages \~~

~~logstash:7.5.1~~

*# 3.为logstash安装json\_lines插件*

/usr/share/logstash/bin/logstash-plugin install logstash-codec-json\_lines

/usr/share/logstash/bin/logstash-plugin install logstash-codec-json\_lines

https://www.cnblogs.com/balloon72/p/13177872.html

elasticsearch

//最新版

docker pull elasticsearch

mkdir -p /data/elasticsearch/config

mkdir -p /data/elasticsearch/data

echo "http.host: 0.0.0.0" >> /data/elasticsearch/config/elasticsearch.yml

docker run --name elasticsearch -p 9200:9200 -p 9300:9300 -e "discovery.type=single-node" -e ES\_JAVA\_OPTS="-Xms64m -Xmx128m" -v /data/elasticsearch/config/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml -v /data/elasticsearch/data:/usr/share/elasticsearch/data -v /data/elasticsearch/plugins:/usr/share/elasticsearch/plugins -d elasticsearch

其中elasticsearch.yml是挂载的配置文件，data是挂载的数据，plugins是es的插件，如ik，而数据挂载需要权限，需要设置data文件的权限为可读可写,需要下边的指令。

chmod -R 777 /data/elasticsearch/data

chmod -R 777 要修改的路径

-e "discovery.type=single-node" 设置为单节点

特别注意：

-e ES\_JAVA\_OPTS="-Xms256m -Xmx256m" \ 测试环境下，设置ES的初始内存和最大内存，否则导致过大启动不了ES

https://blog.csdn.net/shykevin/article/details/108272260

Kibana启动

docker pull kibana

配置文件

mkdir -p /data/elk/kibana/config/

vi /data/elk/kibana/config/kibana.yml

内容如下：

#

# \*\* THIS IS AN AUTO-GENERATED FILE \*\*

#

# Default Kibana configuration for docker target

server.name: kibana

server.host: "0.0.0.0"

#elasticsearch.hosts: [ "http://192.168.31.190:9200" ]

elasticsearch.url: "http://172.17.0.3:9200"

xpack.monitoring.ui.container.elasticsearch.enabled: true

查IP

docker exec -it elasticsearch /bin/bash

root@f5551b7056d7:/usr/share/elasticsearch# ip addr

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

71: eth0@if72: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc noqueue state UP group default

link/ether 02:42:ac:11:00:03 brd ff:ff:ff:ff:ff:ff link-netnsid 0

inet 172.17.0.3/16 brd 172.17.255.255 scope global eth0

valid\_lft forever preferred\_lft forever

启动

docker run -d \

--name=kibana \

--restart=always \

-p 5601:5601 \

-v /data/elk/kibana/config/kibana.yml:/usr/share/kibana/config/kibana.yml \

kibana

查看日志

docker logs -f kibana

等待30秒，如果出现以下信息，说明启动成功了。

{"type":"log","@timestamp":"2020-08-27T03:00:28Z","tags":["listening","info"],"pid":6,"message":"Server running at http://0:5601"}

{"type":"log","@timestamp":"2020-08-27T03:00:28Z","tags":["info","http","server","Kibana"],"pid":6,"message":"http server running at <http://0:5601>"}

网络的问题,将上面的 172.17.0.3 全部替换成 elasticsearch

docker network connect eureka\_net kibana

docker network connect eureka\_net logstash

docker network connect eureka\_net elasticsearch