# Docker部署ELK

ELK是指Elasticsearch、Kibana、Logstash这三种服务搭建的日志收集系统，官方网站：https://www.elastic.co/cn/downloads/。

ELK中各个服务的作用：

- Elasticsearch：用于存储收集到的日志信息；
- Logstash：用于收集日志，SpringBoot应用整合了Logstash以后会把日志发送给Logstash,Logstash再把日志转发给Elasticsearch；
- Kibana：通过Web端的可视化界面来查看日志。

安装参考：

- https://www.elastic.co/guide/en/elasticsearch/reference/7.6/docker.html
- https://www.elastic.co/guide/en/kibana/7.6/docker.html
- https://www.elastic.co/guide/en/logstash/7.6/docker.html

## 1 准备目录

在home目录下面创建`elk`目录

```
mkdir /home/elk
cd /home/elk
mkdir logstash elasticsearch elasticsearch/data elasticsearch/plugins
# 修改数据目录访问权限
chmod 777 elasticsearch/data/
```

将资源目录下面的`logstash.conf`上传到`logstash`目录下面。

在生产环境中在Docker中运行Elasticsearch时适用以下要求和建议，参考下面链接：

https://www.elastic.co/guide/en/elasticsearch/reference/7.6/docker.html#docker-prod-prerequisites

## 2 开放端口

```
# 开放端口
firewall-cmd \
--add-port 9200/tcp \
--add-port 9300/tcp \
--add-port=4560-4563/tcp \
--add-port=5601/tcp \
--permanent
# 重新加载防火墙
firewall-cmd --reload
```

## 3 服务编排

执行命令行`vi docker-compose.yml`创建服务编排文件，然后在文件中写入下列内容后保存。

当然你也可以使用资源目录中提供`yml`文件直接上传到`elk`目录即可，然后跳过本步骤。

```yaml
version: '3'
services:
  elasticsearch:
    image: docker.elastic.co/elasticsearch/elasticsearch:7.6.2
    container_name: elasticsearch
    environment:
      - "cluster.name=elasticsearch"
      - "discovery.type=single-node"
      - "ES_JAVA_OPTS=-Xms256m -Xmx256m"
      - TZ=Asia/Shanghai
    volumes:
      - /home/elk/elasticsearch/plugins:/usr/share/elasticsearch/plugins
      - /home/elk/elasticsearch/data:/usr/share/elasticsearch/data
    ports:
      - 9200:9200
      - 9300:9300
  kibana:
    image: docker.elastic.co/kibana/kibana:7.6.2
    container_name: kibana
    links:
      - elasticsearch:es
    depends_on:
      - elasticsearch
    environment:
      - "elasticsearch.hosts=http://es:9200"
      - "JAVA_OPTS=-Xms256m -Xmx256m"
      - TZ=Asia/Shanghai
    ports:
      - 5601:5601
  logstash:
    image: docker.elastic.co/logstash/logstash:7.6.2
    container_name: logstash
    environment:
      - "LS_JAVA_OPTS=-Xms256m -Xmx256m"
      - TZ=Asia/Shanghai
    volumes:
      - /home/elk/logstash/logstash.conf:/usr/share/logstash/pipeline/logstash.conf
    depends_on:
      - elasticsearch
    links:
      - elasticsearch:es
    ports:
      - 4560:4560
      - 4561:4561
      - 4562:4562
      - 4563:4563
```

## 4 启动服务

执行命令`docker-compose up`。

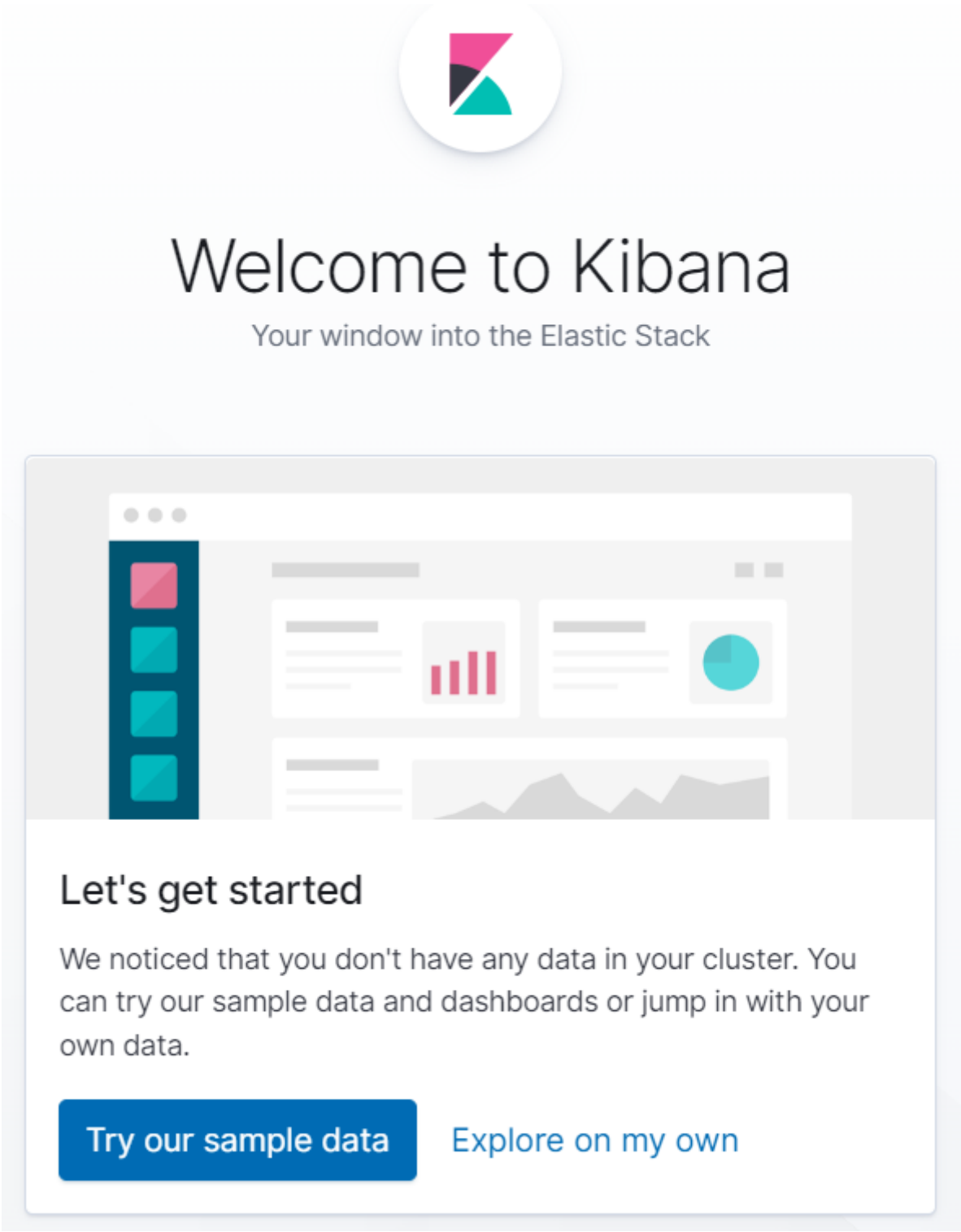
首次启动会拉取镜像，效果如下图所示。

```
[root@localhost elk]# pwd
/home/elk
[root@localhost elk]# ls
docker-compose.yml  elasticsearch  logstash
[root@localhost elk]# docker-compose up
[+] Running 0/10
 ⠿ elasticsearch Pulling
   ⠿ c808caf183b6 Pulling fs layer
   ⠿ d6caf8e15a64 Pulling fs layer
   ⠿ b0ba5f324e82 Pulling fs layer
   ⠿ d7e8c1e99b9a Waiting
   ⠿ 85c4d6c81438 Waiting
   ⠿ 3119218fac98 Waiting
   ⠿ 914accf214bb Waiting
 ⠿ kibana Pulling
 ⠿ logstash Pulling
```

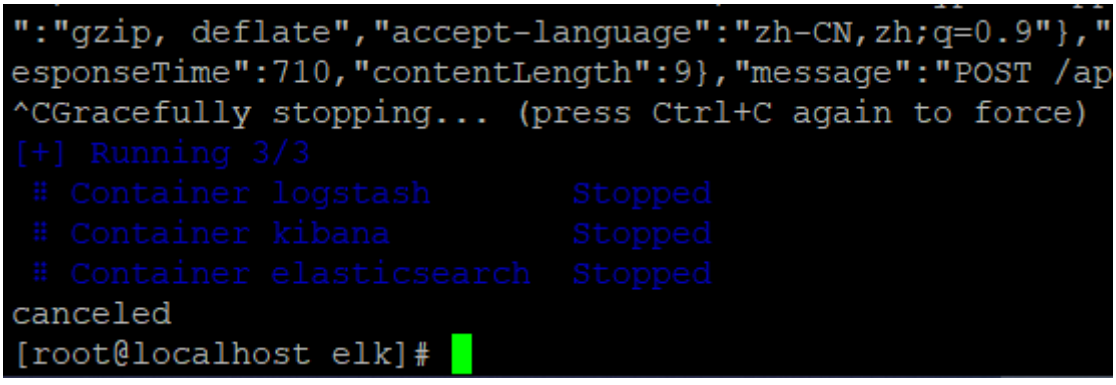
如果启动过程没有看到明显的错误，那么表示服务器启动成功，接下来访问一下elasticsearch和kibana。

首先访问elasticsearch，访问地址：http://ip:9200 如果看了类似下图所示的结果表示elasticsearch启动成功。


接下来在访问kibana，访问地址：http://ip:5601 如果看了类似下图所示的结果表示kibana启动成功。




然后`Ctrl + C`结束前台启动，成功停止效果如下图所示




执行后台启动，使用命令`docker-compose up -d`，启动成功如下图所示

```
canceled
[root@localhost elk]# docker-compose up -d
[+] Running 3/3
 # Container elasticsearch  Started
 # Container kibana         Started
 # Container logstash       Started
[root@localhost elk]#
```

使用`docker ps`指令查看进程

```
[root@localhost elk]# docker ps
CONTAINER ID   IMAGE                                                  COMMAND                  CREATED          STATUS
                  NAMES
6cdb48e6609a   docker.elastic.co/kibana/kibana:7.6.2                  "/usr/local/bin/dumb…"   37 seconds ago   Up 35 seconds
                  kibana
7f7d3a428846   docker.elastic.co/logstash/logstash:7.6.2              "/usr/local/bin/dock…"   37 seconds ago   Up 34 seconds
-4563/tcp, 9600/tcp      logstash
3b533a0963d6   docker.elastic.co/elasticsearch/elasticsearch:7.6.2    "/usr/local/bin/dock…"   37 seconds ago   Up 36 seconds
tcp, :::9300->9300/tcp   elasticsearch
```

# 5 安装插件

## 5.1 安装IK分词插件

进入elasticsearch容器

```
1  docker exec -it elasticsearch /bin/bash
```

执行插件安装命令

```
1  elasticsearch-plugin install https://release.infinilabs.com/analysis-
   ik/stable/elasticsearch-analysis-ik-7.6.2.zip
```

安装完成效果如下图所示

```
[root@3b533a0963d6 elasticsearch]# elasticsearch-plugin install https://github.com/medcl
-> Installing https://github.com/medcl/elasticsearch-analysis-ik/releases/download/v7.6.
-> Downloading https://github.com/medcl/elasticsearch-analysis-ik/releases/download/v7.6
[=================================================] 100%??
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@     WARNING: plugin requires additional permissions     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
* java.net.SocketPermission * connect,resolve
See http://docs.oracle.com/javase/8/docs/technotes/guides/security/permissions.html
for descriptions of what these permissions allow and the associated risks.

Continue with installation? [y/N]y
-> Installed analysis-ik
[root@3b533a0963d6 elasticsearch]#
```

执行exit退出容器

```
Continue with installation? [y/N]y
-> Installed analysis-ik
[root@3b533a0963d6 elasticsearch]# exit
exit
[root@localhost elk]#
```

## 5.2 安装json_lines插件

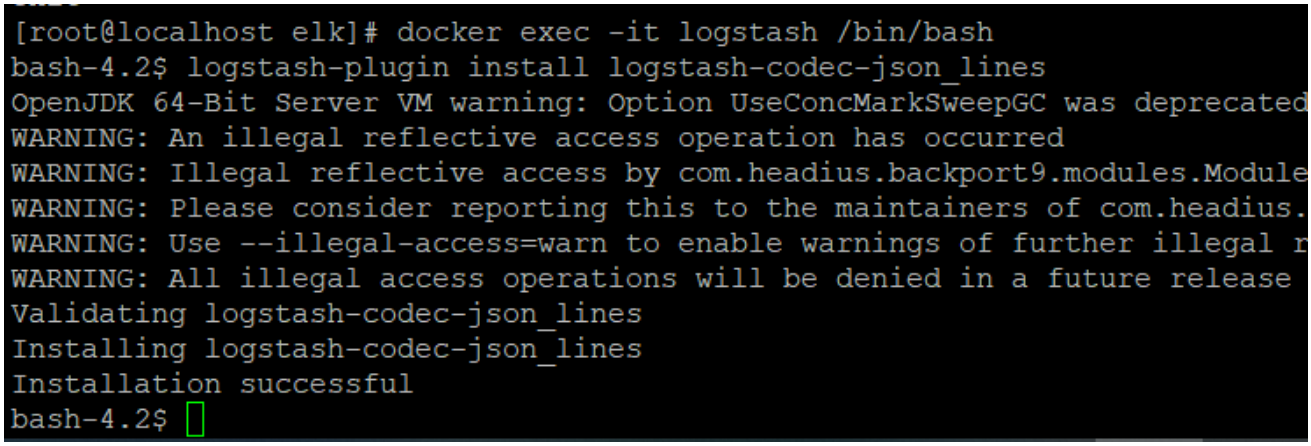
进入logstash容器

```
docker exec -it logstash /bin/bash
```

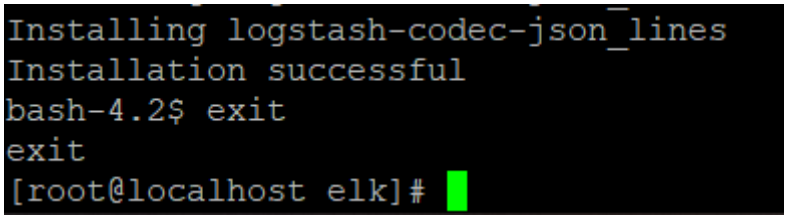
执行插件安装命令

```
logstash-plugin install logstash-codec-json_lines
```

安装成功可以看到如下图所示的效果




执行exit退出容器。




## 5.3 重启服务

所有插件安装完成统一重启一下服务。