[基于docker搭建elasticsearch集群](https://www.cnblogs.com/lz0925/p/12011026.html)

### es集群的搭建

- 基于单机搭建elasticsearch集群见官网 https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html

#### 本文旨在三台不同的服务器，搭建elasticsearch集群，版本为7.1.1

* 1、服务器列表即配置

172.168.50.40（8G)， 172.168.50.41(16G)， 172.168.50.240(8G)

服务器内存尽量不要低于4G

选用172.168.50.41作为master节点，

* 2、建立存放yml文件的目录

建立文件夹，/root/elasticsearch（随意即可)，用于存放启动elasticsearch容器的yml文件以及es的配置文件

mkdir /root/elasticsearch

* 3、创建docker-compose.yml文件

cd /root/elasticsearch

touch docker-compose.yml

* 4、docker-compose.yml 的文件内容如下

version: '3'

services:

elasticsearch: # 服务名称

image: elasticsearch:7.1.1 # 使用的镜像

container\_name: elasticsearch # 容器名称

restart: always # 失败自动重启策略

environment:

- node.name=node-41 # 节点名称，集群模式下每个节点名称唯一

- network.publish\_host=172.168.50.41 # 用于集群内各机器间通信,对外使用，其他机器访问本机器的es服务，一般为本机宿主机IP

- network.host=0.0.0.0 # 设置绑定的ip地址，可以是ipv4或ipv6的，默认为0.0.0.0，即本机

- discovery.seed\_hosts=172.168.50.40,172.168.50.240,172.168.50.41 # es7.0之后新增的写法，写入候选主节点的设备地址，在开启服务后，如果master挂了，哪些可以被投票选为主节点

- cluster.initial\_master\_nodes=172.168.50.40,172.168.50.240,172.168.50.41 # es7.0之后新增的配置，初始化一个新的集群时需要此配置来选举master

- cluster.name=es-cluster # 集群名称，相同名称为一个集群， 三个es节点须一致

# - http.cors.enabled=true # 是否支持跨域，是：true // 这里设置不起作用，但是可以将此文件映射到宿主机进行修改，然后重启，解决跨域

# - http.cors.allow-origin="\*" # 表示支持所有域名 // 这里设置不起作用，但是可以将此文件映射到宿主机进行修改，然后重启，解决跨域

- bootstrap.memory\_lock=true # 内存交换的选项，官网建议为true

- "ES\_JAVA\_OPTS=-Xms512m -Xmx512m" # 设置内存，如内存不足，可以尝试调低点

ulimits: # 栈内存的上限

memlock:

soft: -1 # 不限制

hard: -1 # 不限制

volumes:

- /root/elasticsearch/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml # 将容器中es的配置文件映射到本地，设置跨域， 否则head插件无法连接该节点

- esdata:/usr/share/elasticsearch/data # 存放数据的文件， 注意：这里的esdata为 顶级volumes下的一项。

ports:

- 9200:9200 # http端口，可以直接浏览器访问

- 9300:9300 # es集群之间相互访问的端口，jar之间就是通过此端口进行tcp协议通信，遵循tcp协议。

volumes:

esdata:

driver: local # 会生成一个对应的目录和文件，如何查看，下面有说明。

* 另外两台服务器也照着这个配置进行配置，但 publish\_host，以及 node.name 需要改一下即可。
* 内存设置，三台服务器都需要进行设置

两种设置内存的方式：

1、设置简单，但机器重启后需再次设置

sysctl -w vm.max\_map\_count=262144

2、直接修改配置文件， 进入sysctl.conf文件添加一行（解决容器内存权限过小问题）

vi /etc/sysctl.conf

sysctl vm.max\_map\_count=262144 # 添加此行

退出文件后，执行命令： sysctl -p 立即生效

* 然后三台服务器依次执行 docker-compose up -d
* 然后访问 <http://172.168.50.41:9200/_cluster/health?pretty> 查看是否集群正常运行, 正常运行会返回如下信息

{

"cluster\_name" : "es-cluster",

"status" : "green",

"timed\_out" : false,

"number\_of\_nodes" : 3,

"number\_of\_data\_nodes" : 3,

"active\_primary\_shards" : 0,

"active\_shards" : 0,

"relocating\_shards" : 0,

"initializing\_shards" : 0,

"unassigned\_shards" : 0,

"delayed\_unassigned\_shards" : 0,

"number\_of\_pending\_tasks" : 0,

"number\_of\_in\_flight\_fetch" : 0,

"task\_max\_waiting\_in\_queue\_millis" : 0,

"active\_shards\_percent\_as\_number" : 100.0

}

* elasticsearch.yml 文件内容如下：

network.host: 0.0.0.0

http.cors.enabled: true # 是否支持跨域

http.cors.allow-origin: "\*" # 表示支持所有域名

* 上面说到的顶级volumes， 如何查看挂载卷在宿主机的位置呢？

# docker volume create elk\_data // 创建一个自定义容器卷，本教程内不需要执行，我们的docker-compose.yml会帮我们自动执行改命令。

# docker volume ls // 查看所有容器卷，

# docker volume inspect elk\_data // 查看指定容器卷详情信息, 包括真实目录

注意： 如果你要删除一个挂载卷，或者重新生成，请执行删除卷操作

# docker volume rm elk\_data // 直接执行这个命令，同时会删除文件，但是先删除文件的话，必须再次执行此命令，否则可能导致该节点无法加入集群。

* 以上就是es集群的搭建，如果出现权限不足等简单问题，可以百度自行解决。

知止而后有定，定而后能静，静而后能安，安而后能虑，虑而后能得。 所谓诚其意者，毋自欺也。