Deploying ELK stack in Docker Container:

Steps and Commands:

1. Change the **mmap counts** of ELK stack to not run out of virtual memory during installation and use

```
sudo sysctl -w vm.max_map_count=262144
```

2. pull the ELK image from the Docker registry

```
sudo docker pull sebp/elk
```

3. clone the Git repository and enter the directory

a. `git clone https://github.com/spujadas/elk-docker.git`
b. `Cd elk-docker`

4. build the Docker image with the **docker build** command

```
sudo docker build -t elk-docker .
```

5. use the Dockerfile to install plugins and build the image to run the installation

a) `sudo nano Dockerfile`
b) Add the following at the end of the Dockerfile:

```
FROM sebp/elk
ENV ES_HOME /opt/elasticsearch
WORKDIR ${ES_HOME}
RUN yes | CONF_DIR=/etc/elasticsearch gosu elasticsearch
bin/elasticsearch-plugin \
install -b <plugin name or link>
```

6. Build the image using either **docker build** or **docker-compose**

7. Run ELK stack container using **docker run command**

```
sudo docker run -p 5601:5601 -p 9200:9200 -p 5044:5044 -it --name elk sebp/elk
```

8. The command publishes the following ports:

- **5601** serves the Kibana web interface.
- **9200** for Elasticsearch JSON interface.
- **5044** for Logstash Beats interface.

9. Additionally, the following ports are exposed but not published:

- **9300** for the transport interface of Elasticsearch (expose with **-p 9300:9300**).
- **9600** for the Logstash monitoring API (expose with **-p 9600:9600**).

10. Open browser and go to http://localhost:5601. The ELK stack page will open.

11. Check that Elasticsearch is running with a curl request:

```
curl localhost:9200
```