Deploy ElasticSearch and Kibana using Docker-Compose

Peter Selva P · Follow
3 min read · Apr 30, 2024

> *Hello, Again!*

Today, I will be giving an insight into how to set up the **Elastic-Search** and **Kibana** stacks in Docker. For this tutorial, we will be making use of the docker-compose command to spin up our own containers running Elasticsearch and Kibana

Go ahead and create a **docker-compose.yml** file and paste the code below,

```yaml
version: '3'
services:
  elasticsearch:
    image: elasticsearch:7.8.1
    ports:
      - 9200:9200
    environment:
      discovery.type: 'single-node'
      xpack.security.enabled: 'true'
      ELASTIC_PASSWORD: '<your_password>'
      ES_JAVA_OPTS: '-Xmx2g -Xms2g'
  kibana:
    image: kibana:7.8.1
    volumes:
      - ./kibana.yml:/usr/share/kibana/config/kibana.yml
    ports:
      - 5601:5601
```

> *Note:*

This Docker compose recipe spins up 2 Docker containers on your machine. One container is running Elasticsearch 7.8.1, and the other is running Kibana 7.8.1. It maps the ElasticSearch API to port 9200 and Kibana to port 5601 on your machine. You can check the official Docker images for all available configuration options for Elasticsearch and Kibana. For this tutorial, we will be enabling security on Elasticsearch (duh!) and configuring a password (the default username is “elastic”).

The recipe also mounts the **kibana.yml** to the Kibana config directory, you will need to make a few changes to the file to make sure Kibana connects correctly to your container running Elasticsearch.

```yaml
# To allow connections from remote users, set this parameter to a non-loopback a
server.host: "0.0.0.0"

# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["http://elasticsearch:9200"]

# the username and password that the Kibana server uses to perform maintenance o
# index at startup. Your Kibana users still need to authenticate with Elasticsea
# is proxied through the Kibana server.
elasticsearch.username: "elastic"
elasticsearch.password: "<your_password>"
```

After both **yml** files are ready, open up a terminal, navigate to your directory, and execute the below command,

```
docker-compose up
```

Docker will download the images for Elasticsearch and Kibana and setup your containers. After successful completion you will be able to see your containers running,

Head on over to http://localhost:9200/ to verify that Elasticsearch is now running (key in your username and password, default username is “elastic”), you should see a response like below,


```
{
"name": "87eddc47c914",
"cluster_name": "docker-cluster",
"cluster_uuid": "pZwC9TD5TO-qG_XY7Flaag",
"version": {
"number": "8.12.0",
"build_flavor": "default",
"build_type": "docker",
"build_hash": "1665f706fd9354802c02146c1e6b5c0fbcddfbc9",
"build_date": "2024-01-11T10:05:27.953830042Z",
"build_snapshot": false,
"lucene_version": "9.9.1",
"minimum_wire_compatibility_version": "7.17.0",
"minimum_index_compatibility_version": "7.0.0"
},
"tagline": "You Know, for Search"
}
```


> *Notes*

Before verify the **Kibana** application, we need to create a **kibana_system** user and **password** for elasticsearch to access the Kibana login ,

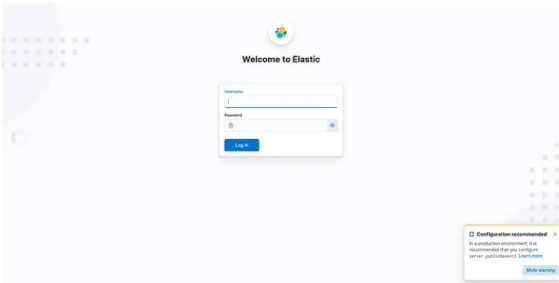
Login to the elastic container and create username and password

```
docker exec -it <elastic container id> bash

./bin/elasticsearch-reset-password --username kibana_system -i

<enter the kibana_system Password>
```

Once you generate username and password, please restart Docker Compose.

```
docker-compose restart
```

To verify that Kibana is successfully up and running on your machine, head on over to http://localhost:5601/ and key in your username and password,



After logging in, you can use the full potential of Kibana.

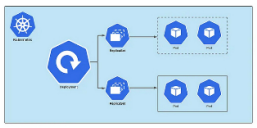
Elasticsearch Kibana Docker Compose Elastic Docker

No responses yet

DRucco

What are your thoughts?

More from Peter Selva P



Peter Selva P

Deploy NGINX on Kubernetes with Ingress with Custom Domain

Oct 15, 2024 51

Peter Selva P

Elastic-Search Data snap and restore using shared file system...

Hello All :

May 11, 2024 8

Peter Selva P

Deploy Liferay Portal with docker-compose

Apr 29, 2024

Peter Selva P

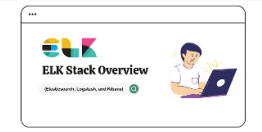
Deploying Self-Hosted Sentry for Application Monitoring

Introduction: In modern software development, application monitoring is...

Feb 9 2

See all from Peter Selva P

Recommended from Medium



Vishwas Acharya

ELK Stack Overview (Elasticsearch, Logstash, and Kibana)

The ELK Stack comprising Elasticsearch, Logstash, and Kibana offers a comprehensiv...

Nov 4, 2024

In DevOps Playbook by Shipa S Behani

🚀 Ultimate Guide to Installing and Configuring the ELK Stack...

🔹 Introduction

Mar 10 3

Vinoth Subbiah

🔨 Day 13: Elasticsearch Scaling and Performance Optimization

🚀 Learning Objective Master the techniques to scale Elasticsearch clusters and optimize...

Jan 16 2

In FAUN—Developer Community by D

How to run a local Elasticsearch using docker (in under 5 min)?

Read this for free here.

Jan 31 4

Ragesh Kanagaraj

ElasticSearch -The Fundamentals -01

What is ElasticSearch ?

Oct 8, 2024 21

Simon Reed

Elasticsearch Tutorial — Configuring Elasticsearch...

In a previous tutorial we had configured elasticsearch using self signed certificates. I...

Dec 8, 2024

See more recommendations