

ELK stack install guideline

1. docker-compose로 ELK설치

- <https://github.com/deviantony/docker-elk>
- git clone <https://github.com/deviantony/docker-elk>

2. .env 파일 수정

```
C:\> elk > docker-elk > .env
1  ELASTIC_VERSION=8.13.4
2
3  ## Passwords for stack users
4  #
5
6  # User 'elastic' (built-in)
7  #
8  # Superuser role, full access to cluster management and data indices.
9  # https://www.elastic.co/guide/en/elasticsearch/reference/current/built-in-users.html
10 ELASTIC_PASSWORD='whs2project'
11
12 # User 'logstash_internal' (custom)
13 #
14 # The user Logstash uses to connect and send data to Elasticsearch.
15 # https://www.elastic.co/guide/en/logstash/current/ls-security.html
16 LOGSTASH_INTERNAL_PASSWORD='whs2project'
17
18 # User 'kibana_system' (built-in)
19 #
20 # The user Kibana uses to connect and communicate with Elasticsearch.
21 # https://www.elastic.co/guide/en/elasticsearch/reference/current/built-in-users.html
22 KIBANA_SYSTEM_PASSWORD='whs2project'
23
24 # Users 'metricbeat_internal', 'filebeat_internal' and 'heartbeat_internal' (custom)
25 #
26 # The users Beats use to connect and send data to Elasticsearch.
27 # https://www.elastic.co/guide/en/beats/metricbeat/current/feature-roles.html
28 METRICBEAT_INTERNAL_PASSWORD=''
29 FILEBEAT_INTERNAL_PASSWORD=''
30 HEARTBEAT_INTERNAL_PASSWORD=''
31
32 # User 'monitoring_internal' (custom)
33 #
34 # The user Metricbeat uses to collect monitoring data from stack components.
35 # https://www.elastic.co/guide/en/elasticsearch/reference/current/how-monitoring-works.html
36 MONITORING_INTERNAL_PASSWORD=''
37
38 # User 'beats_system' (built-in)
39 #
40 # The user the Beats use when storing monitoring information in Elasticsearch.
41 # https://www.elastic.co/guide/en/elasticsearch/reference/current/built-in-users.html
42 BEATS_SYSTEM_PASSWORD=''
43
```

각 패스워드 수정 ex) whs2project

ELASTIC_PASSWORD=' '

LOGSTASH_INTERNAL_PASSWORD=' '

KIBANA_SYSTEM_PASSWORD=' '

3. docker-compose.yml 파일 내 depends_on을 수정

```
C: > elk > docker-elk > docker-compose.yml
1  version: '3.7'
2
3  services:
4
5      # The 'setup' service runs a one-off script which initializes users inside
6      # Elasticsearch such as 'logstash_internal' and 'kibana_system' with the
7      # values of the passwords defined in the '.env' file. It also creates the
8      # roles required by some of these users.
9      #
10     # This task only needs to be performed once, during the *initial* startup of
11     # the stack. Any subsequent run will reset the passwords of existing users to
12     # the values defined inside the '.env' file, and the built-in roles to their
13     # default permissions.
14     #
15     # By default, it is excluded from the services started by 'docker compose up'
16     # due to the non-default profile it belongs to. To run it, either provide the
17     # '--profile=setup' CLI flag to Compose commands, or "up" the service by name
18     # such as 'docker compose up setup'.
19     setup:
20     profiles:
21     - setup
22     build:
23     context: setup/
24     args:
25     ELASTIC_VERSION: ${ELASTIC_VERSION}
26     init: true
27     volumes:
28     - ./setup/entrypoint.sh:/entrypoint.sh:ro,Z
29     - ./setup/lib.sh:/lib.sh:ro,Z
30     - ./setup/roles:/roles:ro,Z
31     environment:
32     ELASTIC_PASSWORD: ${ELASTIC_PASSWORD:-}
33     LOGSTASH_INTERNAL_PASSWORD: ${LOGSTASH_INTERNAL_PASSWORD:-}
34     KIBANA_SYSTEM_PASSWORD: ${KIBANA_SYSTEM_PASSWORD:-}
35     METRICBEAT_INTERNAL_PASSWORD: ${METRICBEAT_INTERNAL_PASSWORD:-}
36     FILEBEAT_INTERNAL_PASSWORD: ${FILEBEAT_INTERNAL_PASSWORD:-}
37     HEARTBEAT_INTERNAL_PASSWORD: ${HEARTBEAT_INTERNAL_PASSWORD:-}
38     MONITORING_INTERNAL_PASSWORD: ${MONITORING_INTERNAL_PASSWORD:-}
39     BEATS_SYSTEM_PASSWORD: ${BEATS_SYSTEM_PASSWORD:-}
40     networks:
41     - elk
42     depends_on:
43     - elasticsearch
44     - logstash
45     - kibana
46
```

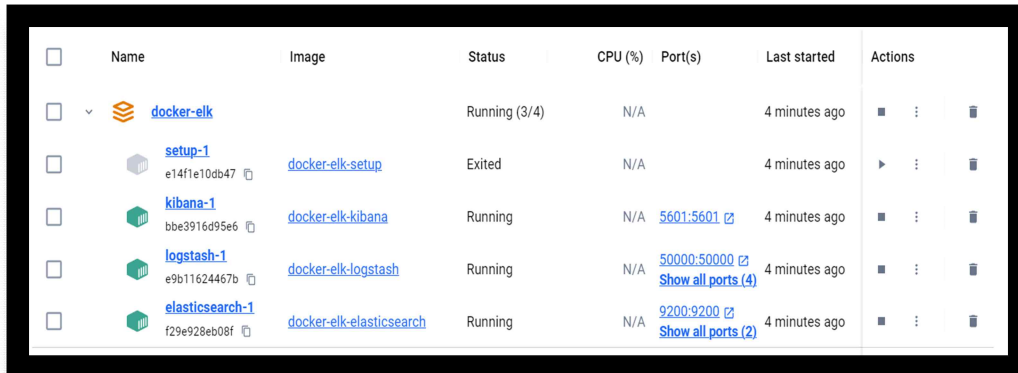
depends_on에 elasticsearch, logstash, kibana 추가




























depends_on:

- elasticsearch
- logstash
- kibana

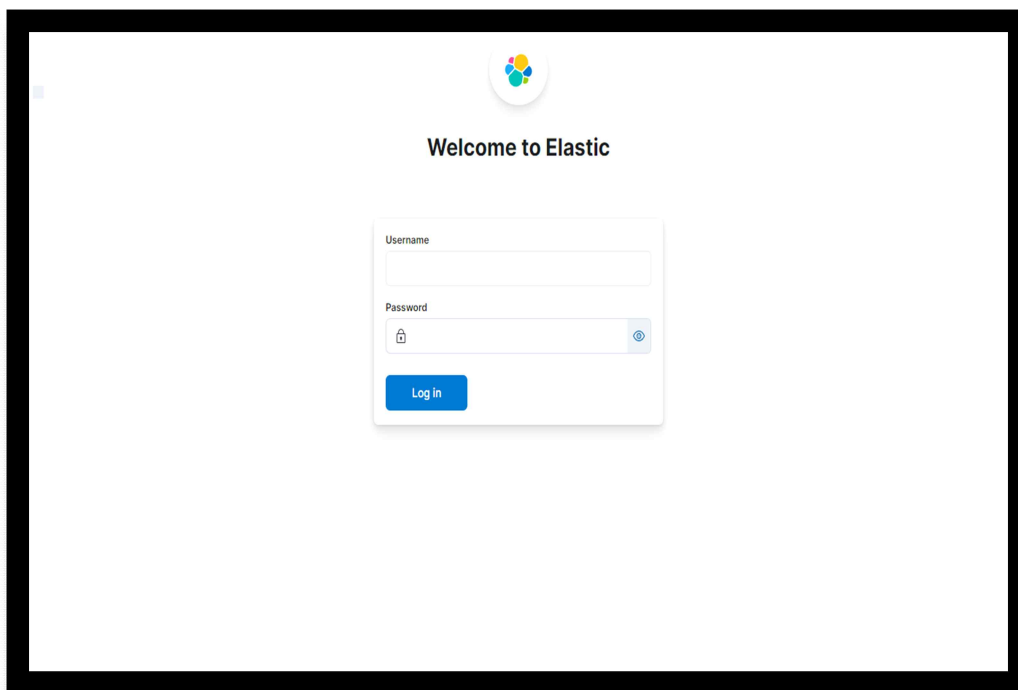
4. docker-compose up

- docker-compose up setup
- docker-compose up -d (백그라운드 실행)



<input type="checkbox"/>	Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
<input type="checkbox"/>	 docker-elk		Running (3/4)	N/A		4 minutes ago	  
<input type="checkbox"/>	 setup-1 e14f1e10db47 	docker-elk-setup	Exited	N/A		4 minutes ago	  
<input type="checkbox"/>	 kibana-1 bbe3916d95e6 	docker-elk-kibana	Running	N/A	5601:5601 	4 minutes ago	  
<input type="checkbox"/>	 logstash-1 e9b11624467b 	docker-elk-logstash	Running	N/A	50000:50000  Show all ports (4)	4 minutes ago	  
<input type="checkbox"/>	 elasticsearch-1 f29e928eb08f 	docker-elk-elasticsearch	Running	N/A	9200:9200  Show all ports (2)	4 minutes ago	  

docker-elk-kibana 포트 주소 클릭



Username = elastic(기본)

Password = whs2project(설정한 값)

5. 접속

