



**Universidad
Europea**

INSTRUCCIONES EPANET

**Andrea Braojos, Laura Poveda,
Ignacio Torres y Victor Rodríguez**

Configuración Elasticsearch En Docker

Estas instrucciones configuran Elasticsearch en Docker, creando un entorno aislado y accesible para búsquedas y análisis de datos. A continuación, se explican los pasos para configurar la red, descargar la imagen de Elasticsearch y ejecutar el contenedor. Para la configuración de Elastic en docker, abre el terminal y escribe los siguientes comandos indicados en los pasos:

- 1) **Crear red de Docker:** crea una red llamada elastic para permitir la comunicación entre contenedores que usen esa misma red.
Comando: `docker network create elastic`
- 2) **Descargar la imagen de Elasticsearch:** descarga la imagen oficial de Elasticsearch versión 8.15.3 desde el repositorio de Elastic.
Comando: `docker pull docker.elastic.co/elasticsearch/elasticsearch:8.15.3`
- 3) **Ejecutar Elasticsearch en un contenedor:** inicia un contenedor llamado pbd-es01 conectado a la red elastic, abre el puerto 9200 y asigna 2 GB de memoria al servicio Elasticsearch.
Comando: `docker run --name pbd-es01 --net elastic -p 9200:9200 -itd -m 2GB docker.elastic.co/elasticsearch/elasticsearch:8.15.3`

Una vez creado el contenedor, se deben mirar los logs en docker desktop en el propio contenedor llamado “pbd-es01” para ver qué contraseña tiene el usuario “elastic”. Si el contenedor no está ejecutándose, dale al play para que empiece a hacerlo. Se te abrirá una pestaña de internet, y te saldrá un error. Para continuar, se debe pulsar el botón de configuración avanzada, y, posteriormente, escribir en el navegador <https://localhost:9200>. Después te pedirá un usuario y contraseña que son:

Usuario: elastic

Contraseña: la contraseña se encuentra en los logs del contenedor.

Para comprobar si está funcionando debe aparecer algo como esto:

```
{
  "nombre": "d0b97ffbc911",
  "nombre_clúster": "clúster docker",
  "uuid_del_clúster": "a2DrJKrKQl28APIY0YFYPA",
  "versión": {
    "número": "8.15.3",
    "build_flavor": "predeterminado",
    "tipo_de_compilación": "docker",
    "hash_de_compilación": "f97532e680b555c3a05e73a74c28afb666923018",
    "fecha_de_compilación": "2024-10-09T22:08:00.328917561Z",
    "build_snapshot": falso,
    "lucene_version": "9.11.1",
    "versión_de_compatibilidad_de_cable_mínima": "7.17.0",
    "versión_de_compatibilidad_de_índice_mínimo": "7.0.0"
  },
  "tagline": "Ya sabes, para buscar"
}
```

Cargar y ejecutar el archivo Python en Docker:

El archivo ProyectoEsios.py primeramente lee los datos de la API de esios de la red eléctrica española generando un csv para cada petición, y posteriormente, se vuelcan en elastic creando un índice para cada csv. Para ejecutarlo en docker, se deben seguir estos pasos:

- 1) Descargar y descomprimir la carpeta entregada del trabajo, en dicha carpeta tenemos que tener el archivo Python llamado ProyectoEsios.py y el Docker file llamado "Proyecto.dockerfile" que es un archivo sin extensión.
- 2) Antes de empezar a ejecutar los archivos, en el archivo "ProyectoEsios.py", debemos escribir la contraseña que viene en los logs, y que hemos usado antes para ejecutar elastic. El lugar donde se debe cambiar es en la función "insercionElastic()" en la línea 552 en "basic_auth". En ella viene el usuario "elastic" y otra contraseña, la cual se debe cambiar por la que hemos utilizado antes para correr elastic.
- 3) Abrimos la línea de comandos del ordenador y navegamos hasta encontrarnos dentro de la carpeta del proyecto donde se encuentran los dos archivos mencionados anteriormente. Para ello usa el comando `cd` "*nombre_ruta_archivo*"
- 4) Escribimos el comando para crear la imagen en Docker:
Comando: **`docker build -t epanet_imagen -f Proyecto.dockerfile . -->`**
IMPORTANTE incluir el punto al final del comando.
- 5) Una vez ejecutado esto creamos el contenedor con este comando:
Comando: **`docker run -d --name epanet_contenedor epanet_imagen`**
- 6) Comprobamos que se ha ejecutado correctamente el código en Docker desde la parte de files del contenedor, se nos tienen que haber creado y cargado los csv de datos dentro de la carpeta "app" de files (esto lleva un poco de tiempo

porque son muchos datos y varios archivos). Nota: Tarda aproximadamente media hora en ejecutarse todo.