OpenSearch

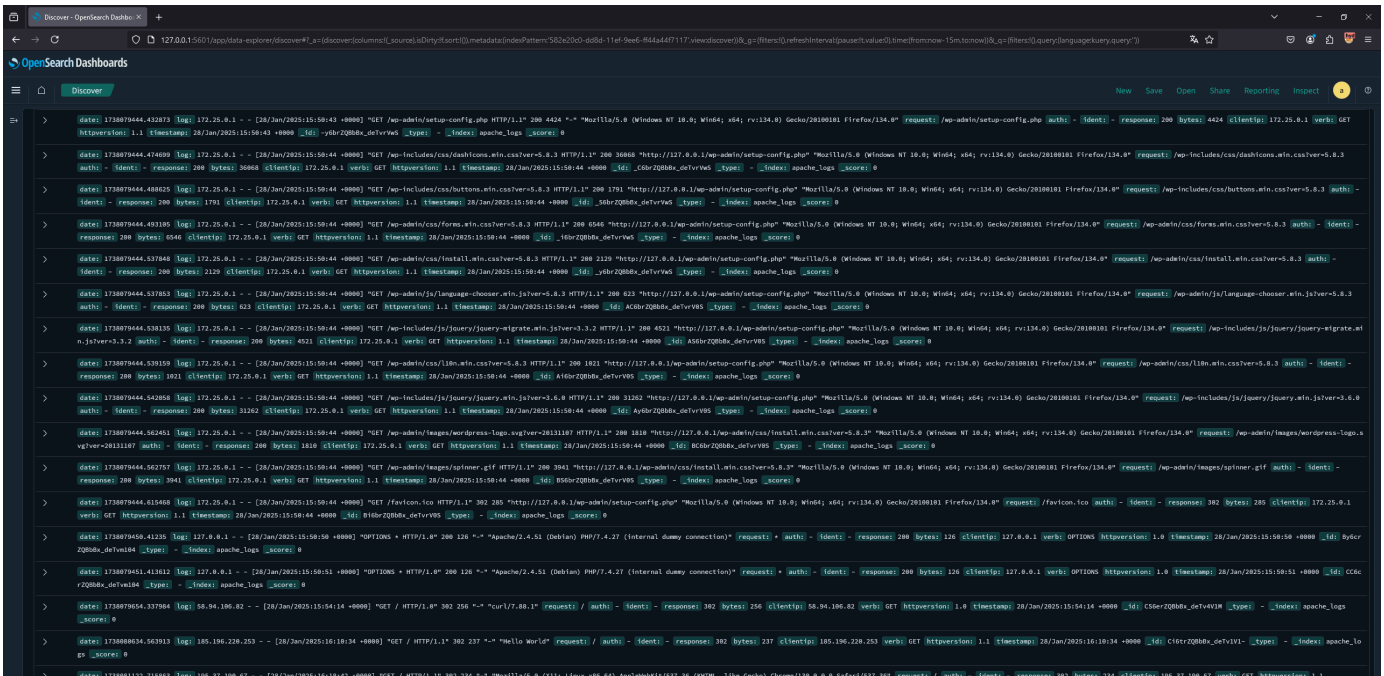
OpenSearch is an open-source dashboard that we use to automatically ingest multiple log files to monitor Apache and Scada-LTS access logs.

Screenshots

For example, the `webserver/log/apache2/access.log` log file is read and then visualized in our OpenSearch dashboard through a log ingestion workflow.

- The access.log file from the Apache web server

- The ingested log file visualization in OpenSearch Dashboard



Docker configuration

It comprises of the following containers:

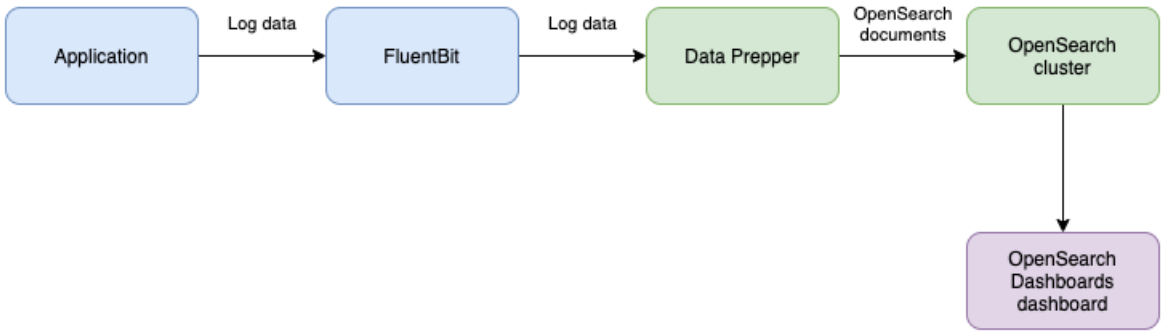
- `opensearch-node` - The OpenSearch core
- `opensearch-dashboards` - The OpenSearch web interface

For log ingestion it also requires:

- `fluent-bit` - Log collector that collects log data and sends it to Data Prepper
- `data-prepper` - Receives and transforms the log data into a structure format and indexes it OpenSearch

Data flow diagram

This diagram shows the log ingestion workflow before it can be visualized in the OpenSearch Dashboard.

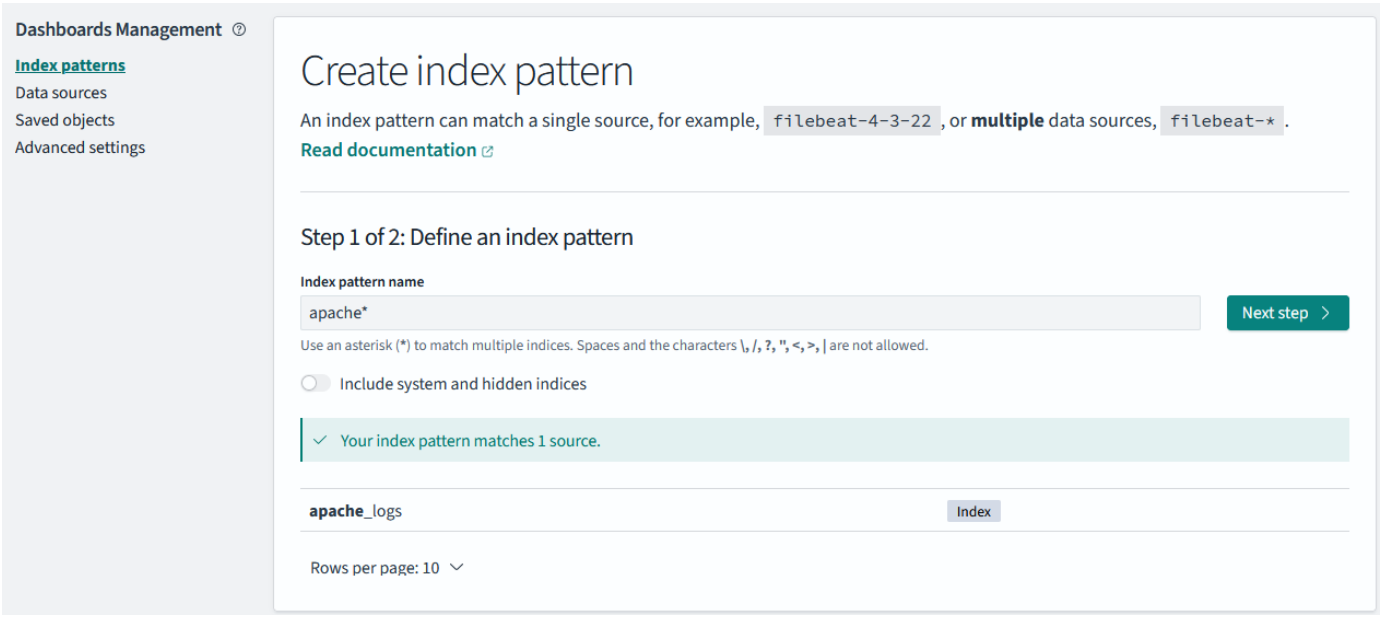


Credentials

The credentials for the OpenSearch dashboard are set in `opensearch/opensearch.env`, in our case `admin:Patat123!`. Note that when setting one it requires at least a lowercase character, an uppercase character, a number and a symbol, otherwise OpenSearch will refuse to run.

Index pattern

Before the log files can be ingested in OpenSearch Dashboard they need to be indexed first, which requires adding an index pattern. This can be done via `Discovery -> Create index pattern` and adding the `apache*` index pattern.



Log ingestion workflow

The log file paths are defined in the `docker-compose.yaml`. For example The `/var/log/apache2` folder in the webserver container is mapped to `./webserver/log/apache2` on the host machine, which then allows the fluent-bit container to ingest the `./webserver/log/apache2/access.log` log file to `/var/log/test.log` in the OpenSearch container.

```
webserver:
  container_name: webserver
  volumes:
    - ./webserver/log/apache2:/var/log/apache2
```

```
  fluent-bit:
    container_name: fluent-bit
    volumes:
      - ./webserver/log/apache2/access.log:/var/log/test.log
```

Testing

When visiting the (WordPress) web server on http://127.0.0.1:80 the `access.log` will be updated and should be automatically ingested to the OpenSearch dashboard.

This can also be manually tested by adding a line to the log file, which should show in the dashboard when refreshing it. This will require first elevating to root with `sudo su` or otherwise editing with `sudo nano`, since the log files are supposed to be read-only.

```
echo '63.173.168.120 - - [04/Nov/2021:15:07:25 -0500] "GET /search/tag/list HTTP/1.0" 200 5003' >>
./webserver/log/apache2/access.log
```

The OpenSearch API also allows querying raw documents which can be used to troubleshoot the connection.

```
curl -X GET -u 'admin:Patat123!' -k 'https://localhost:9200/apache_logs/_search?pretty&size=1'
```

Which should return the following JSON response.


```
{
  "took" : 966,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 114,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "apache_logs",
        "_id" : "-i6brZQBbBx_deTvrVwS",
        "_score" : 1.0,
        "_source" : {
          "date" : 1.738079443556357E9,
          "log" : "172.25.0.1 - - [28/Jan/2025:15:50:43 +0000] \"GET / HTTP/1.1\" 302 286 \"-\"
\"Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:134.0) Gecko/20100101 Firefox/134.0\"",
          "request" : "/",
          "auth" : "-",
          "ident" : "-",
          "response" : "302",
          "bytes" : "286",
          "clientip" : "172.25.0.1",
          "verb" : "GET",
          "httpversion" : "1.1",
          "timestamp" : "28/Jan/2025:15:50:43 +0000"
        }
      }
    ]
```

```
    }
  }
```