

T·FLEX DOCS 17

SERVER INSTALLATION GUIDE

for Linux OS



Copyright

Copyright

© Copyright 2024 Top Systems

All rights reserved. Any copying of this document in part or as a whole without a prior written permission obtained from Top Systems is expressly prohibited.

Top Systems assume no responsibility for any errors or omissions that may appear in this documentation. No claims are accepted for damages caused by using the information contained herein. The information contained in this document is subject to change without notice.

T-FLEX, T-FLEX DOCs, T-FLEX CAD and T-FLEX Parametric CAD trademarks are property of Top Systems Corp.

All other trademarks are the property of their respective companies.

Table of Contents

Copyright.....	2
Table of Contents	3
Introduction.....	5
Preparing OS to installation of T-FLEX DOCs Server and related components.....	6
Managing superuser rights on Debian.....	6
Superuser mode on Debian	6
Giving superuser rights to the current user on Debian	6
Installing .NET 6.0 Runtime.....	7
Method 1. Installing .NET 6.0 from internet-repository	7
Method 2. Installing .NET 6.0 without using internet-repository	8
Installing HASP License Manager.....	9
Installing PostgreSQL.....	11
Configuring PostgreSQL.....	12
Installing T-FLEX DOCs Server	14
Installing full-text search platform.....	15
Installing Elasticsearch platform.....	15
Installing Elasticsearch server	15
Configuring Elasticsearch server.....	16
Launching Elasticsearch server.....	17
Installing Kibana.....	18
Configuring Kibana.....	18
Launching Kibana.....	18
Configuring Elasticsearch cluster.....	19
Creating Elasticsearch index template.....	21
Installing OpenSearch platform.....	23
Installing OpenSearch server	23
Launching OpenSearch server.....	24
Installing Ingest Attachment plugin.....	24
Installing OpenSearch Dashboards	25
Configuring OpenSearch Dashboards.....	25
Launching OpenSearch Dashboards.....	26
Configuring OpenSearch cluster	27
Creating OpenSearch index template	30
Configuring OpenSearch.....	32
Configuring T-FLEX DOCs Server.....	33
Launching T-FLEX DOCs Server services.....	36
Method 1. Launching T-FLEX DOCs Server services as generic application.....	36
Method 2. Launching T-FLEX DOCs Server services using OS services management tools.....	37

Table of Contents

Stopping T-FLEX DOCs Server services.....	41
Method 1. Stopping T-FLEX DOCs Server services as generic applications.....	41
Method 2. Stopping T-FLEX DOCs Server services using OS services management tools.....	41
Updating T-FLEX DOCs Server.....	42
Uninstalling T-FLEX DOCs Server.....	43

Introduction

This document gives instructions on how to install T-FLEX DOCs Server using Wine on Linux operating systems.

Wine – is a set of free open-source programs and libraries, that allows to run 16-, 32- and 64-bit applications developed for Microsoft Windows on UNIX-based operating systems supporting CPU x86 and AMD64 architectures. Windows applications are running without any virtualization or emulation.

The instructions are provided for following Linux distributions

- Debian (version 11 and higher)
- Ubuntu (version 22.04 and higher)

Commands to be executed using terminal of operating system are marked as follows within this document:

- Command applicable to all aforementioned versions of operating system:

The text to be typed into the terminal

- Command applicable to particular versions only:

Version	The text to be typed into the terminal
---------	---

Instead of typing commands, you can just copy them to clipboard from the current document and then paste into the terminal. In case of long commands not fitting a single line of the document, it may be necessary to remove line breaks, upon pasting a command into the terminal, or copy lines one by one.

Upon inputting a command text, press **Enter** to execute the command.

Preparing OS to installation of T-FLEX DOCs Server and related components

Before installing T-FLEX DOCs Server on Debian, you should [provide superuser rights to the current user](#).

If you're using Ubuntu, skip this chapter and proceed to the [.NET 6.0 Runtime installation](#).

Managing superuser rights on Debian

In order to install or uninstall some of the operating systems components required to run T-FLEX DOCs Server, you need superuser rights.

Superuser mode on Debian

You can enter the superuser mode by executing the following command in the terminal:

Debian	<code>su root</code>
--------	----------------------

Upon executing this command, the system asks for a root password. Type the password into the terminal and press **Enter**.

When the superuser rights are no longer needed, you can exit the superuser mode by pressing **Ctrl+D** in the terminal.

However, some steps of T-FLEX DOCs Server and related components installation should be performed outside the superuser mode. Therefore, in order to avoid switching back and forth between the current user and the superuser, it is recommended to give superuser rights to the current user, as specified below.

Giving superuser rights to the current user on Debian

Superuser rights can be provided to the current user as follows:

1. Open the terminal and enter the superuser mode:

Debian	<code>su root</code>
--------	----------------------

Upon executing this command, the system asks for a root password. Type the password into the terminal and press **Enter**.

2. Open the `sudoers` file:

Debian	<code>nano /etc/sudoers</code>
--------	--------------------------------

3. Scroll down the file, until you find the following line:

```
root ALL=(ALL:ALL) ALL
```

Add the similar line below, but use the current username instead of `root`.

Then close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

4. Exit the superuser mode by pressing **Ctrl+D** in the terminal.

After installing or uninstalling T-FLEX DOCs Client and related components, when you no longer need superuser rights, open the `sudoers` file again and remove the previously added line. Then close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

Installing .NET 6.0 Runtime

If you're using Debian, instructions given below presume, that you have [provided superuser rights to the current user](#).

Method 1. Installing .NET 6.0 from internet-repository

This method is recommended, as it allows to update installed software automatically in future.

1. On Debian add the Microsoft package signing key to your list of trusted keys:

Debian 11	<code>wget https://packages.microsoft.com/config/debian/11/packages-microsoft-prod.deb -O packages-microsoft-prod.deb</code>
Debian 12	<code>wget https://packages.microsoft.com/config/debian/12/packages-microsoft-prod.deb -O packages-microsoft-prod.deb</code>

2. On Debian add the Microsoft package repository:

Debian	<code>sudo dpkg -i packages-microsoft-prod.deb</code>
--------	---

Upon executing the command, the system asks for a root password. Type the password into the terminal and press **Enter**.

3. Update the packaging index:

```
sudo apt-get update
```

4. Install .NET 6.0 runtime:

```
sudo apt-get install -y dotnet-runtime-6.0
```


Method 2. Installing .NET 6.0 without using internet-repository

This method should be used, if there's no internet connection on the target computer.

1. Put the following archive into the **Downloads** directory of the target computer:
[dotnet-sdk-6.0.418-linux-x64.tar.gz](#)

Download the archive on a computer connected to Internet and transfer it onto the target computer using physical media or local network connection.

2. Create the **dotnet** directory within the **/opt** directory:

```
sudo mkdir /opt/dotnet
```

Upon executing the command, the system asks for a root password. Type the password into the terminal and press **Enter**.

3. Unpack the archive into the created directory:

```
sudo tar xf ~/Downloads/dotnet-sdk-6.0.418-linux-x64.tar.gz -C /opt/dotnet
```

4. Open the environmental variables file:

Ubuntu	<code>nano ~/.profile</code>
Debian	<code>nano ~/.bashrc</code>

In the opened file, add following lines:

```
export DOTNET_ROOT=/opt/dotnet
```

```
export PATH=$PATH:/opt/dotnet
```

Then close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

5. Reboot the OS.

Installing HASP License Manager

If you're using Debian, instructions given below presume, that you have [provided superuser rights to the current user](#).

1. Put the following archive into the **Downloads** directory of the target computer:

[T-FLEX 17 Linux Components.zip](#)

If there is no internet connection on the target computer, download the archive on a computer connected to Internet and transfer it onto the target computer using physical media or local network connection.

2. Change working directory to **Downloads**:

```
cd ~/Downloads
```

3. Unpack the **T-FLEX_17_Linux_Components.zip** archive:

```
unzip -x T-FLEX_17_Linux_Components.zip
```

4. Change working directory to the **aksusbd** directory, previously unpacked from the **T-FLEX_17_Linux_Components.zip** archive:

```
cd HASP_Linux/aksusbd/
```

5. Make the **dinst** file executable:

```
sudo chmod +x dinst
```

Upon executing the command, the system asks for a root password. Type the password into the terminal and press **Enter**.

6. Install the **aksusbd** package:

```
sudo ./dinst
```

7. Change working directory to the **HASP_Linux** directory, previously unpacked from the **T-FLEX_17_Linux_Components.zip** archive:

```
cd ~/Downloads/HASP_Linux
```

8. Copy **haspvlib_46113.so** and **haspvlib_x86_64_46113.so** libraries into the HASP License Manager directory:

```
sudo cp haspvlib_46113.so /var/hasplm/haspvlib_46113.so
```

```
sudo cp haspvlib_x86_64_46113.so /var/hasplm/haspvlib_x86_64_46113.so
```

9. Make libraries copied on the previous step executable:

```
sudo chmod +x /var/hasplm/haspvlib_46113.so
```

```
sudo chmod +x /var/hasplm/haspvlib_x86_64_46113.so
```

10. Restart HASP License Manager service:

```
sudo systemctl restart hasplmd
```

11. Check that the HASP License Manager service is running:

```
systemctl status hasplmd
```

If the service is not running, try reinstalling the **aksusbd** package, as described in steps 5-9, and notice errors occurring during installation. Then check again that the service is running, and, if it isn't, contact Top Systems technical support and provide information on performed actions and detected errors.

Installing PostgreSQL

If you're using Debian, instructions given below presume, that you have [provided superuser rights to the current user](#).

Instructions given below describe installation of latest version of PostgreSQL. If you need a different version, please refer to following manuals:

- [for Debian](#)
- [for Ubuntu](#)

1. Update the packaging index:

```
sudo apt update
```

2. Install PostgreSQL package:

```
sudo apt install postgresql postgresql-contrib
```

After getting info on the packages to be installed, the system will ask to allow additional disk space usage. Type **Y** and press **Enter** to proceed the installation.

If the OS was installed from an installation media and such media was not removed from the list of sources, the system might request access to the media. In such case, either insert it into the disk drive and press **Enter**, or deny the request by pressing **Ctrl+D**.

3. Check that the PostgreSQL service is running:

```
systemctl status postgresql
```

If the service is not running, notice error messages displayed in the service status viewer and try to eliminate such errors using PostgreSQL manual and third-party resources.

Configuring PostgreSQL

Instructions given below presume, that you have [installed PostgreSQL](#).

If you're using Debian, instructions given below also presume, that you have [provided superuser rights to the current user](#).

1. Log into the **psql** client as **postgres** service user:

```
sudo -u postgres psql
```

Upon executing this command, the system asks for a root password. Type the password into the terminal and press **Enter**.

2. Set DBMS administrator password:

```
\password postgres
```

Upon executing the command, type the desired password, then press **Enter**, type same password again and press **Enter** again.

3. Exit the **postgres** service user session:

```
\q
```

4. Find the location of the **postgresql.conf** file:

```
sudo -u postgres psql -c 'SHOW config_file'
```

5. Open the **postgresql.conf** file:

```
sudo nano /etc/postgresql/15/postgresql.conf
```

Use the path found on the previous step, instead of **/etc/postgresql/15/postgresql.conf**

6. In the opened file find the following line, in the **CONNECTIONS AND AUTHENTICATION** section in order to set remote access to database:

```
#listen_addresses = 'localhost'          # what IP address(es) to listen on;
```

Uncomment this line by removing the **#** symbol at its beginning and specify the desired value of the **listen_addresses** parameter. Following values can be used:

```
listen_addresses = 'localhost'
```

The postgresql service listens to localhost only, remote access is impossible.

```
listen_addresses = '*'
```

The postgresql service is connected to all network interfaces.

```
listen_addresses = '0.0.0.0'
```

The postgresql service is connected to all network interfaces IPv4.

```
listen_addresses = ':::'
```

The postgresql service is connected to all network interfaces IPv6.

```
listen_addresses = '192.168.1.2,10.0.0.2'
```

The postgresql service is connected to network interfaces with listed IP-addresses. Addresses are separated by commas.

Configuring PostgreSQL

7. In order to ensure the correct work of T-FLEX DOCs Server, it is recommended to increase the value of the **max_locks_per_transaction** parameter, which manages the average number of object locks provided for each transaction. To do so, scroll the opened file further down to the **LOCK MANAGEMENT** section and find the following line:

```
#max_locks_per_transaction = 64          # min 10
```

Uncomment the line by removing the # symbol at its beginning and increase the value from **64** to **1024**.

8. Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.
9. Restart the PostgreSQL service:

```
sudo systemctl restart postgresql
```

Installing T-FLEX DOCs Server

Instructions given below presume, that, before installing T-FLEX DOCs Server, you have [installed .NET 6.0 Runtime](#), [installed HASP License Manager](#), [installed PostgreSQL](#) and [configured PostgreSQL](#).

If you're using Debian, instructions given below also presume, that you have [provided superuser rights to the current user](#).

1. Put the T-FLEX DOCs Server distribution archive into the **Downloads** directory of the target computer. The name of the archive complies with the following pattern:

```
t-flex-docs-*.tar.gz
```

where * is replaced by a version number.

2. Create T-FLEX DOCs Server folder:

```
sudo mkdir /opt/t-flex-docs
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, add instance name to the name of the folder.

3. Unpack the T-FLEX DOCs Server installation archive into the created folder:

```
sudo tar -xzf ~/Downloads/t-flex-docs-*.tar.gz -C /opt/t-flex-docs
```

Use the name of the actual archive instead of **t-flex-docs-*.tar.gz**.

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of **/opt/t-flex-docs**.

4. Provide writing access to the T-FLEX DOCs Server directory for the server process launched by the current user:

```
sudo chown -R $USER:$USER /opt/t-flex-docs
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of **/opt/t-flex-docs**.

Installing full-text search platform

In T-FLEX DOCs on Linux full-text search may be based on one of the following platforms:

- Elasticsearch
- OpenSearch

Installing both platforms on the same computer is not recommended, as they may conflict with each other. T-FLEX DOCs Server will use the platform [specified in its configuration file](#).

Installing Elasticsearch platform

Installing Elasticsearch platform for T-FLEX DOCs full-text search includes:

- [Installing](#), [configuring](#) and [launching](#) Elasticsearch server;
- [Installing](#), [configuring](#) and [launching](#) Kibana platform;
- [Configuring](#) Elasticsearch cluster;
- [Creating](#) Elasticsearch index template;
- Setting up secure connection.

Setting up a secure connection is not a mandatory procedure for T-FLEX DOCs full-text search to work. This step is performed when necessary and is not covered within the current guide.

Installing Elasticsearch server

1. Add one of the following repositories:

- Elastic repository

```
echo "deb https://artifacts.elastic.co/packages/8.x/apt stable main" | tee  
-a /etc/apt/sources.list.d/elastic-8.x.list
```

Elastic repository may be unavailable in some countries. In such case, the following message appears in the terminal, upon trying to add the repository:

```
tee: /etc/apt/sources.list.d/elastic-8.x.list: Permission denied
```

If Elastic repository is not available, use Yandex repository.

- Yandex repository

```
echo "deb [trusted=yes] https://mirror.yandex.ru/mirrors/elastic/8/ stable  
main" | sudo tee /etc/apt/sources.list.d/elastic-8.x.list
```

2. Download and add the Elasticsearch repository key:

```
sudo wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch  
sudo apt-key add -
```

4. Update the packaging index:

```
sudo apt update
```

3. Install the Elasticsearch package:

```
sudo apt install elasticsearch
```


Configuring Elasticsearch server

1. Open the Elasticsearch configuration file:

```
sudo nano /etc/elasticsearch/elasticsearch.yml
```

2. Scroll the file down to the security features section and set `false` values for following parameters:

```
xpack.security.enabled  
xpack.security.enrollment.enabled  
xpacksecurity.http.ssl.enabled  
xpacksecurity.transport.ssl.enabled
```

```
#----- BEGIN SECURITY AUTO CONFIGURATION -----  
#  
# The following settings, TLS certificates, and keys have been automatically  
# generated to configure Elasticsearch security features on 13-02-2024 13:44:33  
#  
# -----  
  
# Enable security features  
xpack.security.enabled: false  
xpack.security.enrollment.enabled: false  
  
# Enable encryption for HTTP API client connections, such as Kibana, Logstash,  
xpack.security.http.ssl:  
  enabled: false  
  keystore.path: certs/http.p12  
  
# Enable encryption and mutual authentication between cluster nodes  
xpack.security.transport.ssl:  
  enabled: false  
  verification_mode: certificate  
  keystore.path: certs/transport.p12  
  truststore.path: certs/transport.p12  
# Enable remote clusters with the same security features
```

3. Add the following line at the end of the file:

```
indices.analysis.hunspell.dictionary.ignore_case: true
```

```
# Allow other nodes to join the cluster from anywhere  
# Connections are encrypted and mutually authenticated  
#transport.host: 0.0.0.0  
  
#----- END SECURITY AUTO CONFIGURATION -----  
  
indices.analysis.hunspell.dictionary.ignore_case: true
```

4. Close the file by pressing `Ctrl+X` and confirm changes by typing `Y` and pressing `Enter`.

Launching Elasticsearch server

1. Enable and launch the Elasticsearch service:

```
sudo systemctl enable --now elasticsearch
```

2. Check that Elasticsearch service is running:

```
sudo systemctl status elasticsearch
```

```
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/lib/systemd/system/elasticsearch.service; disabled; preset: enabled)
   Active: active (running) since Tue 2024-02-20 15:27:22 MSK; 10s ago
     Docs: https://www.elastic.co
   Main PID: 40018 (java)
    Tasks: 167 (limit: 9201)
   Memory: 4.1G
     CPU: 41.643s
   CGroup: /system.slice/elasticsearch.service
           └─40018 /usr/share/elasticsearch/jdk/bin/java -Xms4m -Xmx64m -XX:+UseSerialGC
           └─40092 /usr/share/elasticsearch/jdk/bin/java -Des.networkaddress.cache.ttl=60
           └─40131 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/bin/c

Feb 20 15:27:10 ubuntu-23 systemd[1]: Starting elasticsearch.service - Elasticsearch...
Feb 20 15:27:12 ubuntu-23 systemd-entrypoint[40018]: Feb 20, 2024 3:27:12 PM sun.util.local
Feb 20 15:27:12 ubuntu-23 systemd-entrypoint[40018]: WARNING: COMPAT locale provider will b
Feb 20 15:27:22 ubuntu-23 systemd[1]: Started elasticsearch.service - Elasticsearch.
```

3. Install curl:

```
sudo apt install curl
```

4. Check the availability of Elasticsearch server:

```
curl -XGET http://127.0.0.1:9200
```

If the server is running, the following response will be displayed:

```
roman@ubuntu-23:~$ curl -XGET http://localhost:9200
{
  "name" : "ubuntu-23",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "6w6QBqZTQj2VWkmJhXux7w",
  "version" : {
    "number" : "8.10.3",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "c63272efed16b5a1c25f3ce500715b7fddf9a9fb",
    "build_date" : "2023-10-05T10:15:55.152563867Z",
    "build_snapshot" : false,
    "lucene_version" : "9.7.0",
    "minimum_wire_compatibility_version" : "7.17.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

Installing Kibana

Kibana is a tool for visualizing and managing Elasticsearch server data.

1. Install Kibana:

```
sudo apt install kibana
```

Configuring Kibana

1. Open the Kibana configuration file:

```
sudo nano /etc/kibana/kibana.yml
```

2. Uncomment the `server.host` parameter by removing the `#` symbol at the start of the line and set the value to `"0.0.0.0"`:

```
# ===== System: Kibana Server =====  
# Kibana is served by a back end server. This setting specifies the port to use  
#server.port: 5601  
  
# Specifies the address to which the Kibana server will bind. IP addresses and  
# The default is 'localhost', which usually means remote machines will not be able  
# To allow connections from remote users, set this parameter to a non-loopback  
server.host: "0.0.0.0"  
  
# Enables you to specify a path to mount Kibana at if you are running behind a
```

3. Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

Launching Kibana

1. Launch Kibana service:

```
sudo systemctl start kibana
```

In order to avoid unnecessary usage of system resources, it is not recommended to enable the kibana service, because kibana is used only for initial configuring of server and is not required for further work.

2. Check that Kibana service is running:

```
sudo systemctl status kibana
```

```
roman@ubuntu-23:~$ sudo systemctl status kibana  
● kibana.service - Kibana  
   Loaded: loaded (/lib/systemd/system/kibana.service; disabled; preset: enabled)  
   Active: active (running) since Tue 2024-02-20 15:56:54 MSK; 5min ago  
     Docs: https://www.elastic.co  
    Main PID: 53001 (node)  
      Tasks: 11 (limit: 9201)  
     Memory: 417.2M  
        CPU: 28.139s  
    CGroup: /system.slice/kibana.service  
            └─53001 /usr/share/kibana/bin/../../node/bin/node /usr/share/kibana/bin/../../src/cl  
Feb 20 15:57:12 ubuntu-23 kibana[53001]: [2024-02-20T15:57:12.901+03:00][INFO ][plugins.obs  
Feb 20 15:57:12 ubuntu-23 kibana[53001]: [2024-02-20T15:57:12.927+03:00][INFO ][plugins.ale
```

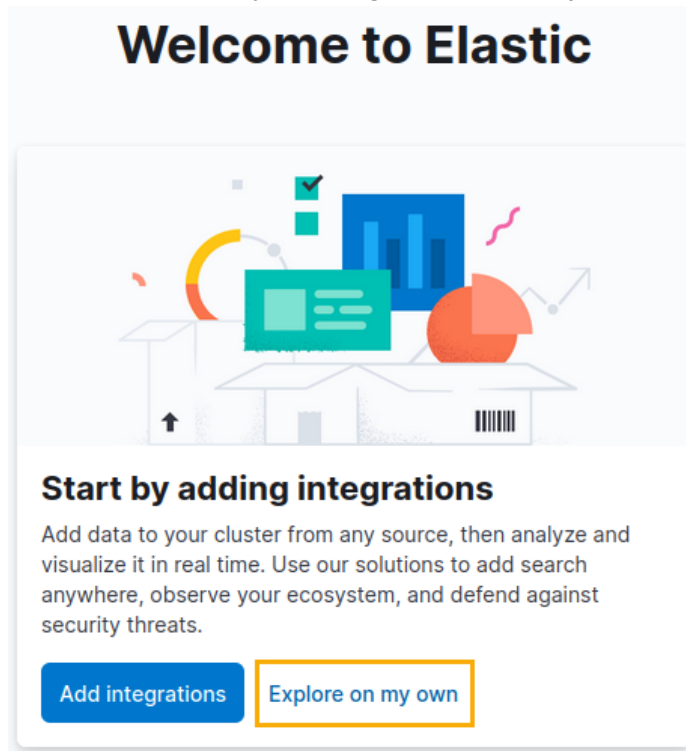
Configuring Elasticsearch cluster



Upon launching Kibana, you should reconfigure the Elasticsearch cluster, in order to disable the automatic index creation.

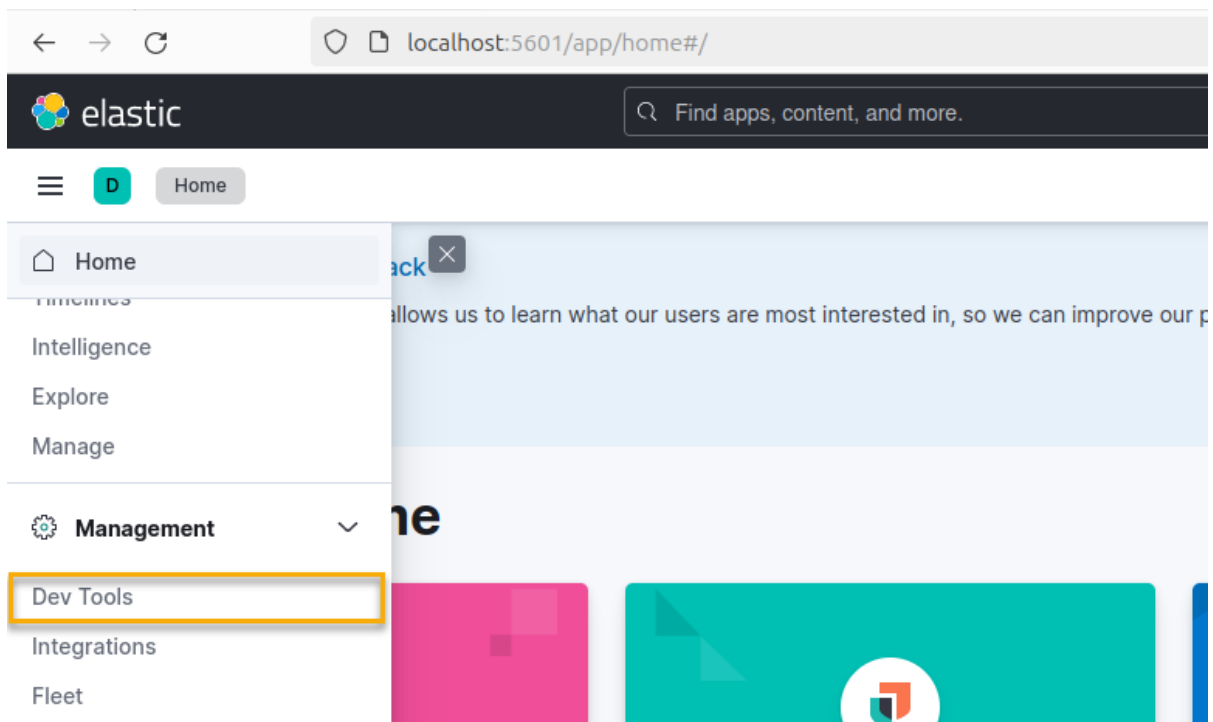
1. Open Kibana graphical interface using the following link: <http://localhost:5601>.

Launching kibana server takes some time after launching the service. If the graphical interface doesn't get open, despite the active service, wait for some time and try again. The required time depends on the performance of the system (usually under one minute).

2. Skip the Kibana initial set up screen by pressing **Explore on my own**.

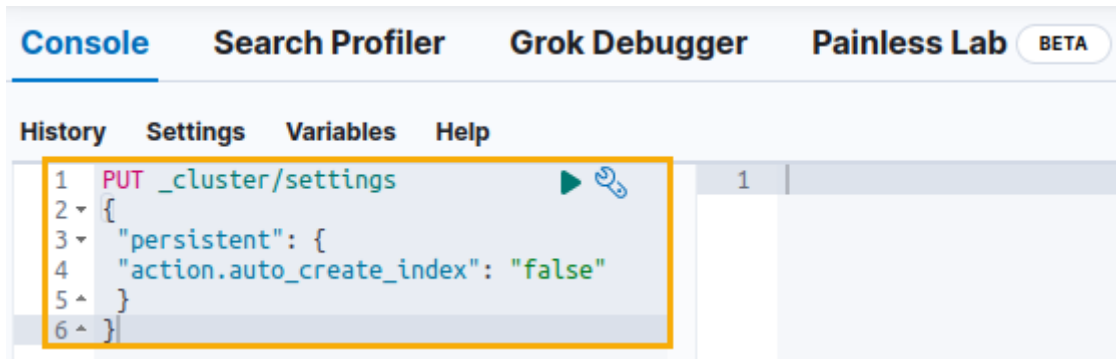


3. Invoke the main menu by pressing the  button in the upper left corner of the window and open the Kibana administrator console by selecting  **Management** > **Dev Tools**:

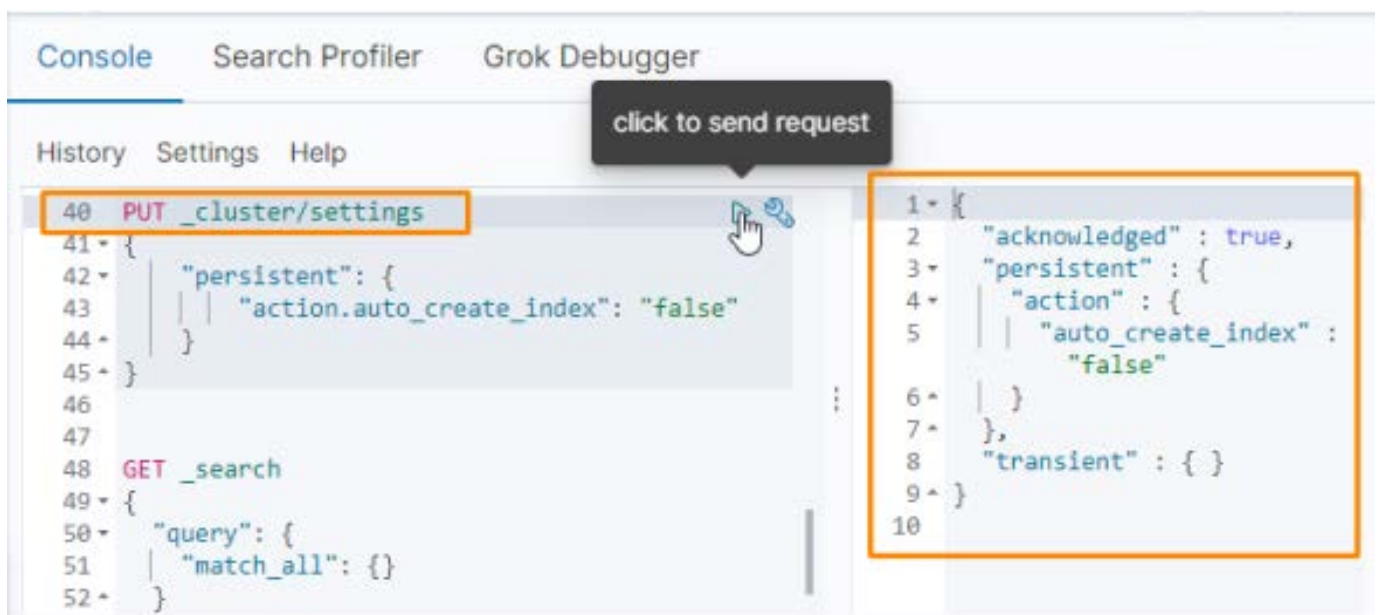


4. Input the following request into the left section of the Kibana administrator console window:

```
PUT _cluster/settings
{
  "persistent": {
    "action.auto_create_index": "false"
  }
}
```



5. Send the request by clicking the play button to the right of its first line. Upon sending the request, the response will appear in the right section of the Kibana console window. If the request is successful, the response looks like shown below:



If the request yields error, check for it being correctly inputted, and try sending again. If error still occurs, try to eliminate it using Kibana and Elasticsearch documentation or third-party resources.

6. Leave the Kibana administrator console open, as you'll need it later for creating the index template.

Creating Elasticsearch index template

An index template defines the settings according to which Elasticsearch indexes will be created. In the index template for T-FLEX DOCs full-text search, it is recommended to create a text analyzer ("lang_analyzer") taking into account the morphology of the language based on the **Hunspell** dictionary. To do this you should:

1. Change working directory to **Downloads**:


```
cd ~/Downloads
```

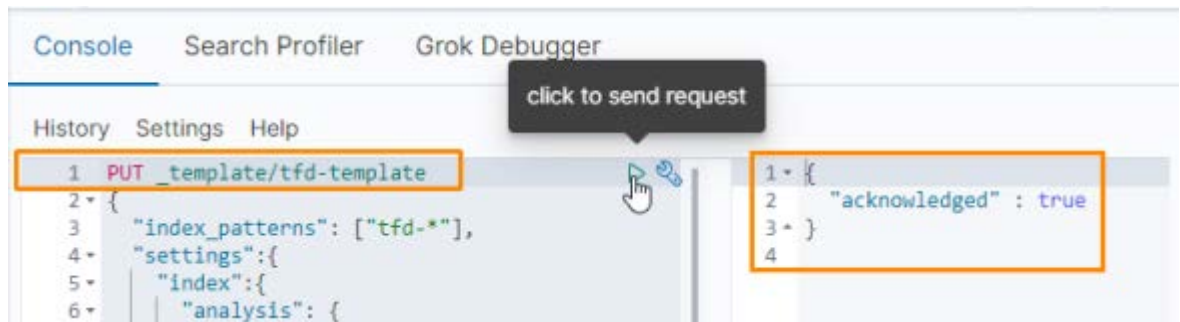
2. Move the **hunspell** folder, previously unpacked from the **T-FLEX_17_Linux_Components.zip** archive, into the **/etc/elasticsearch** directory:

```
sudo mv hunspell /etc/elasticsearch
```

3. Input a query that creates a text analyzer into the left section of the previously opened Kibana administrator console window:

```
PUT _template/tfd-template
{
  "index_patterns": ["tfd-*"],
  "settings": {
    "index": {
      "analysis": {
        "char_filter": {
          "yo_filter": {
            "type": "mapping",
            "mappings": [
            ]
          }
        },
        "filter": {
          "hunspell_filter": {
            "type": "hunspell",
            "locale": "en_US"
          }
        }
      },
      "analyzer": {
        "lang_analyzer": {
          "tokenizer": "standard",
          "filter": [
            "lowercase",
            "hunspell_filter"
          ],
          "char_filter": [
            "yo_filter"
          ]
        }
      }
    }
  }
}
```


4. Send the request by clicking the  button to the right of its first line. Upon sending the request, the response will appear in the right section of the Kibana console window. If the request is successful, the response looks like shown below:



If the request yields error, check for it being correctly inputted, and try sending again. If error still occurs, try to eliminate it using Kibana and Elasticsearch documentation or third-party resources.

5. Stop the Kibana service:

```
sudo systemctl stop kibana
```

ATTENTION! For the full-text search to work, you should also enable it in the T-FLEX DOCs Server configuration file `ServerSetup.Example.Linux.xml`, as described in the [Configuring T-FLEX DOCs Server](#) section.

Installing OpenSearch platform

Installing OpenSearch platform for T-FLEX DOCs full-text search includes:

- [Installing](#) and [launching](#) OpenSearch server;
- [Installing](#) Ingest Attachment plugin;
- [Installing, configuring](#) and [launching](#) OpenSearch Dashboards platform;
- [Configuring](#) OpenSearch cluster;
- [Creating](#) OpenSearch index template;
- [Configuring](#) OpenSearch server;
- Setting up secure connection.

Setting up a secure connection is not a mandatory procedure for T-FLEX DOCs full-text search to work. This step is performed when necessary and is not covered within the current guide.

Installing OpenSearch server

1. Go to the [OpenSearch downloads webpage](#).
2. Download the **x64/deb** OpenSearch package:

OpenSearch

OpenSearch is a distributed search and analytics engine based on Apache Lucene. After adding your data to OpenSearch, you can perform full-text searches on it with all of the features you might expect: search by field, search multiple indices, boost fields, rank results by score, sort results by field, and aggregate results.

Platform:

Package:

Downloaded file will be saved in the **Downloads** directory by default. If there is no internet connection on the target computer, download the archive on a computer connected to Internet and transfer it onto the target computer using physical media or local network connection.

3. Change working directory to **Downloads**:

```
cd ~/Downloads
```

4. Install OpenSearch:

```
sudo env OPENSEARCH_INITIAL_ADMIN_PASSWORD=1234 dpkg -i opensearch-*-linux-x64.deb
```

Instead of **1234**, make up and use an OpenSearch administrator password. Instead of *****, specify the version of the downloaded file.

If an error occurs during installation, check the information in the `/var/log/opensearch/opensearch.log` file and act accordingly, depending on the type of error. The most common case of errors is the password being too weak. Some combinations of symbols may not be recognized as a password, despite being strong enough; in such case, the log will state that password is not found.

```
Open | install_demo_configuration.log [Rea... | Save | - | □ | ×
/var/log/opensearch
1 ### OpenSearch Security Demo Installer
2 ### ** Warning: Do not use on production or public reachable systems **
3 OpenSearch install type: rpm/deb on Linux 6.5.0-26-generic amd64
4 OpenSearch config dir: /etc/opensearch/
5 OpenSearch config file: /etc/opensearch/opensearch.yml
6 OpenSearch bin dir: /usr/share/opensearch/bin/
7 OpenSearch plugins dir: /usr/share/opensearch/plugins/
8 OpenSearch lib dir: /usr/share/opensearch/lib/
9 Detected OpenSearch Version: 2.12.0
10 Detected OpenSearch Security Version: 2.12.0.0
11 Password 555 is weak. Please re-try with a stronger password.
```

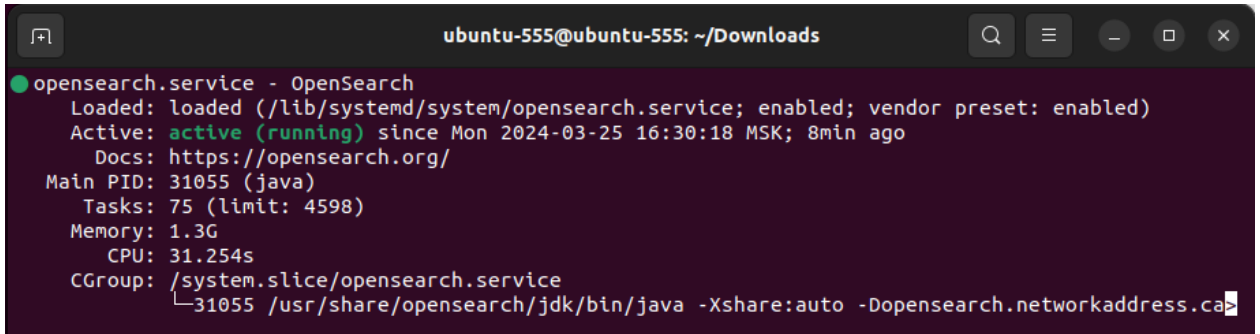
Launching OpenSearch server

1. Enable and launch the OpenSearch service:

```
sudo systemctl enable --now opensearch
```

2. Check that the OpenSearch service is running:

```
sudo systemctl status opensearch
```



```
ubuntu-555@ubuntu-555: ~/Downloads
● opensearch.service - OpenSearch
   Loaded: loaded (/lib/systemd/system/opensearch.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-03-25 16:30:18 MSK; 8min ago
     Docs: https://opensearch.org/
   Main PID: 31055 (java)
    Tasks: 75 (limit: 4598)
   Memory: 1.3G
      CPU: 31.254s
   CGroup: /system.slice/opensearch.service
           └─31055 /usr/share/opensearch/jdk/bin/java -Xshare:auto -Dopensearch.networkaddress.ca
```

3. Install curl:

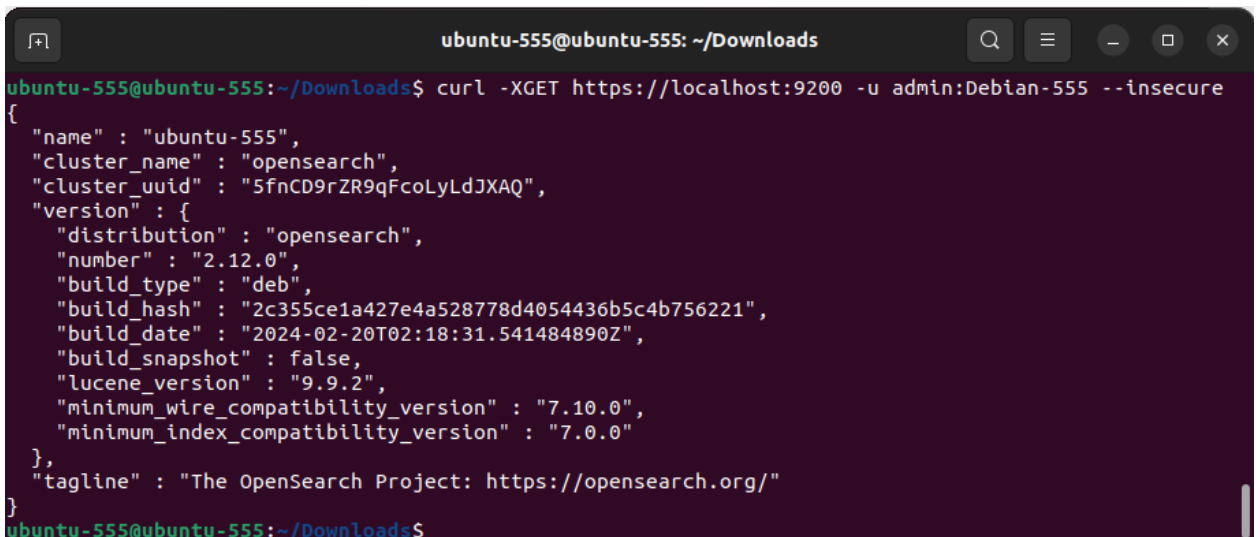
```
sudo apt install curl
```

4. Check the availability of OpenSearch server:

```
curl -XGET https://localhost:9200 -u admin:1234 --insecure
```

Instead of 1234 use the administrator password specified in the step 4 of the [OpenSearch server installation](#).

If the server is running, the following response will be displayed:



```
ubuntu-555@ubuntu-555: ~/Downloads
ubuntu-555@ubuntu-555:~/Downloads$ curl -XGET https://localhost:9200 -u admin:Debian-555 --insecure
{
  "name" : "ubuntu-555",
  "cluster_name" : "opensearch",
  "cluster_uuid" : "5fnCD9rZR9qFcoLyLdJXAQ",
  "version" : {
    "distribution" : "opensearch",
    "number" : "2.12.0",
    "build_type" : "deb",
    "build_hash" : "2c355ce1a427e4a528778d4054436b5c4b756221",
    "build_date" : "2024-02-20T02:18:31.541484890Z",
    "build_snapshot" : false,
    "lucene_version" : "9.9.2",
    "minimum_wire_compatibility_version" : "7.10.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "The OpenSearch Project: https://opensearch.org/"
}
ubuntu-555@ubuntu-555:~/Downloads$
```

Installing Ingest Attachment plugin

1. Install the Ingest-attachment plugin:

```
sudo ./bin/opensearch-plugin install ingest-attachment
```

Installing OpenSearch Dashboards

OpenSearch Dashboards is a tool for visualizing and managing OpenSearch server data.

1. Go to the [OpenSearch Dashboards downloads webpage](#).
2. Download the **x64/deb** OpenSearch Dashboards package:

OpenSearch Dashboards

OpenSearch Dashboards is the default visualization tool for data in OpenSearch. It also serves as a user interface for many of the OpenSearch plugins, including security, alerting, Index State Management, SQL, and more.

Platform:

Package:

Downloaded file will be saved in the **Downloads** directory by default. If there is no internet connection on the target computer, download the archive on a computer connected to Internet and transfer it onto the target computer using physical media or local network connection.

3. Change working directory to **Downloads**:

```
cd ~/Downloads
```

4. Install OpenSearch Dashboards:

```
sudo dpkg -i opensearch-dashboards-*-linux-x64.deb
```

Instead of * specify the version of the downloaded file.

Configuring OpenSearch Dashboards

1. Open the OpenSearch Dashboards configuration file:

```
sudo nano /etc/opensearch-dashboards/opensearch_dashboards.yml
```

2. Uncomment the **server.host** parameter by removing the # symbol at the start of the line and set the value to **"0.0.0.0"**:

```
ubuntu-555@ubuntu-555: ~/Downloads
GNU nano 6.2 /etc/opensearch-dashboards/opensearch_dashboards.yml *
---
# Copyright OpenSearch Contributors
# SPDX-License-Identifier: Apache-2.0

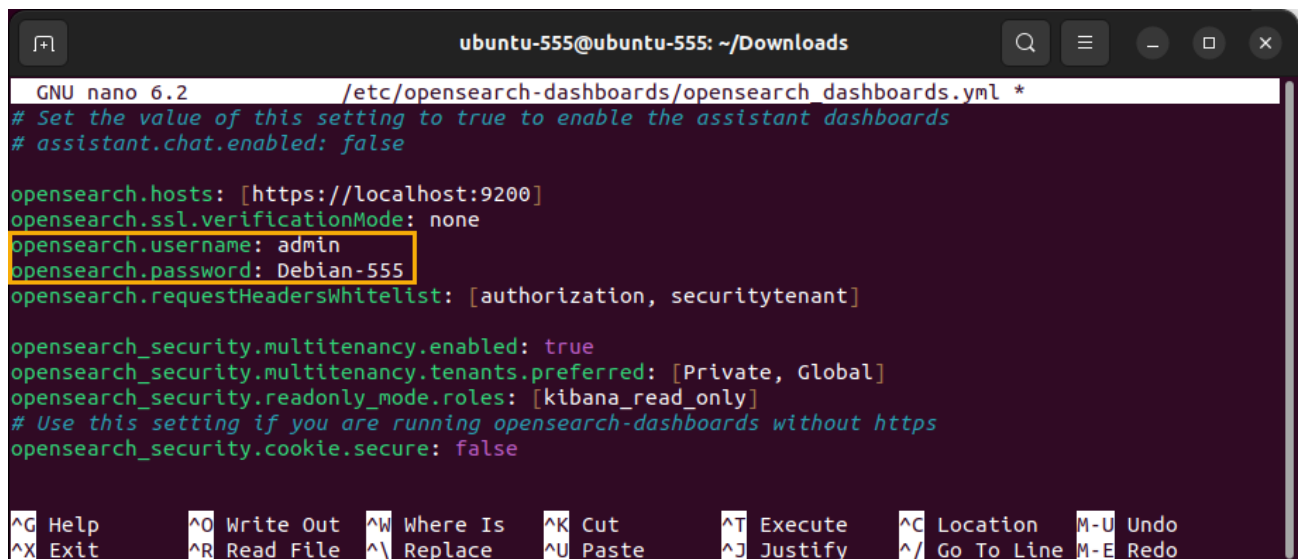
# Description:
# Default configuration for OpenSearch Dashboards

# OpenSearch Dashboards is served by a back end server. This setting specifies the port to use.
# server.port: 5601

# Specifies the address to which the OpenSearch Dashboards server will bind. IP addresses and host
# The default is 'localhost', which usually means remote machines will not be able to connect.
# To allow connections from remote users, set this parameter to a non-loopback address.
server.host: "0.0.0.0"

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location M-U Undo
^X Exit      ^R Read File ^\ Replace  ^U Paste     ^J Justify  ^/_ Go To Line M-E Redo
```

3. Set the value of the `opensearch.username` parameter to `admin`, and the value of the `opensearch.password` to the administrator password specified in the step 4 of the [OpenSearch server installation](#).



```
ubuntu-555@ubuntu-555: ~/Downloads
GNU nano 6.2 /etc/opensearch-dashboards/opensearch_dashboards.yml *
# Set the value of this setting to true to enable the assistant dashboards
# assistant.chat.enabled: false

opensearch.hosts: [https://localhost:9200]
opensearch.ssl.verificationMode: none
opensearch.username: admin
opensearch.password: Debian-555
opensearch.requestHeadersWhitelist: [authorization, securitytenant]

opensearch_security.multitenancy.enabled: true
opensearch_security.multitenancy.tenants.preferred: [Private, Global]
opensearch_security.readonly_mode.roles: [kibana_read_only]
# Use this setting if you are running opensearch-dashboards without https
opensearch_security.cookie.secure: false

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line  M-E Redo
```

4. Close the file by pressing `Ctrl+X` and confirm changes by typing `Y` and pressing `Enter`.

Launching OpenSearch Dashboards

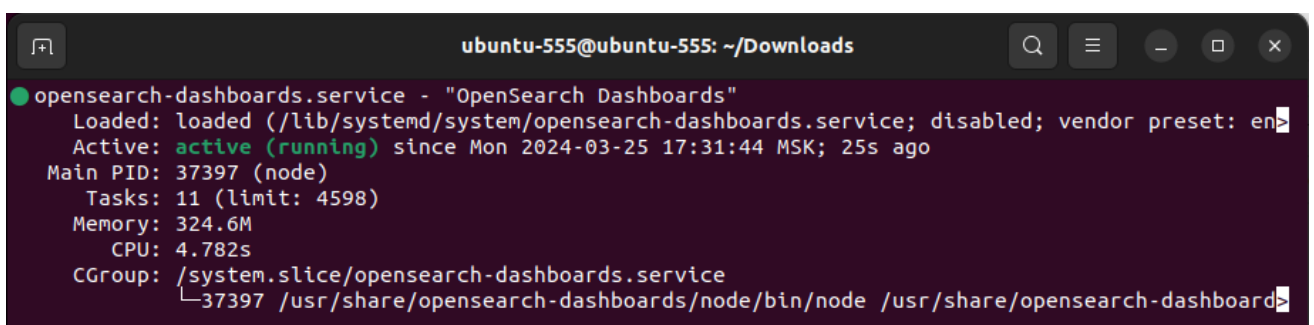
1. Launch the OpenSearch Dashboards service:

```
sudo systemctl start opensearch-dashboards
```

In order to avoid unnecessary usage of system resources, it is not recommended to enable the `opensearch-dashboards` service, because `opensearch-dashboards` is used only for initial configuring of server and is not required for further work.

2. Check that the OpenSearch Dashboards service is running:

```
sudo systemctl status opensearch-dashboards
```



```
ubuntu-555@ubuntu-555: ~/Downloads
● opensearch-dashboards.service - "OpenSearch Dashboards"
   Loaded: loaded (/lib/systemd/system/opensearch-dashboards.service; disabled; vendor preset: en
   Active: active (running) since Mon 2024-03-25 17:31:44 MSK; 25s ago
     Main PID: 37397 (node)
        Tasks: 11 (limit: 4598)
       Memory: 324.6M
          CPU: 4.782s
      CGroup: /system.slice/opensearch-dashboards.service
             └─37397 /usr/share/opensearch-dashboards/node/bin/node /usr/share/opensearch-dashboard
```

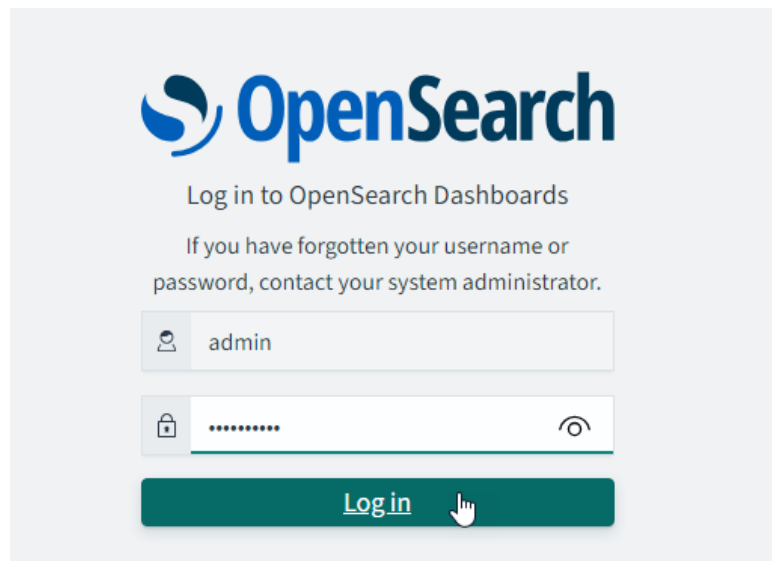
Configuring OpenSearch cluster

Upon launching OpenSearch Dashboards, you should reconfigure the OpenSearch cluster, in order to disable the automatic index creation.

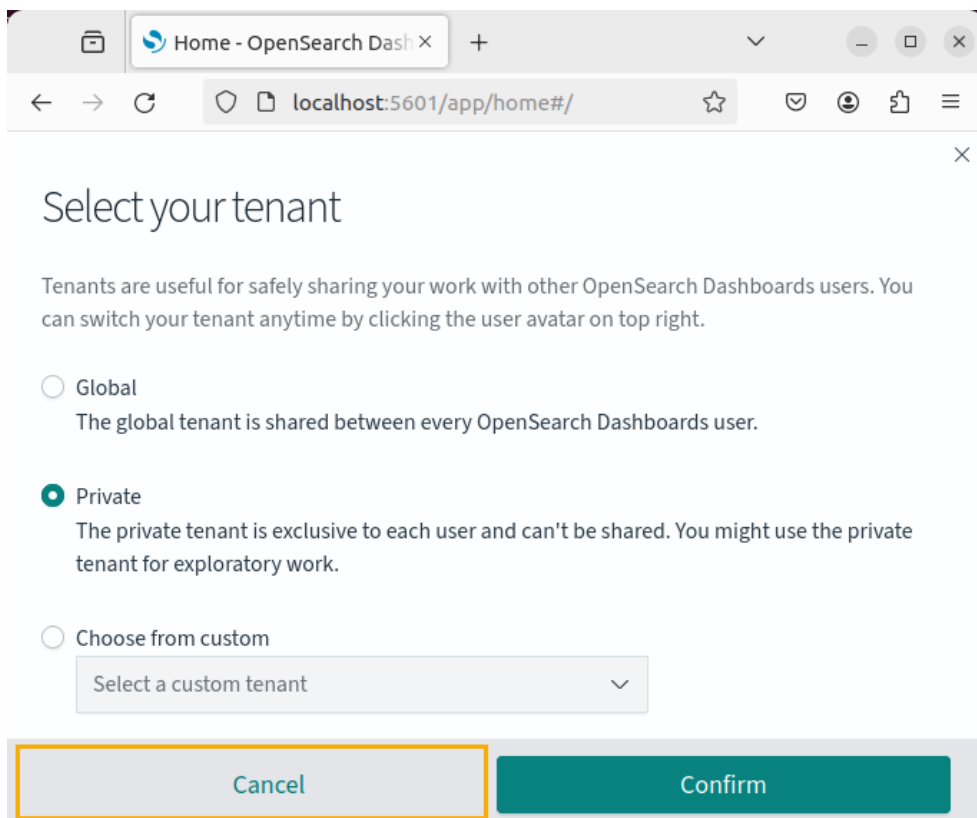
1. Open the OpenSearch Dashboards graphical interface using the following link:
<http://localhost:5601>.

Launching OpenSearch Dashboards server takes some time after launching the service. If the graphical interface doesn't get open, despite the active service, wait for some time and try again. The required time depends on the performance of the system (usually under one minute).

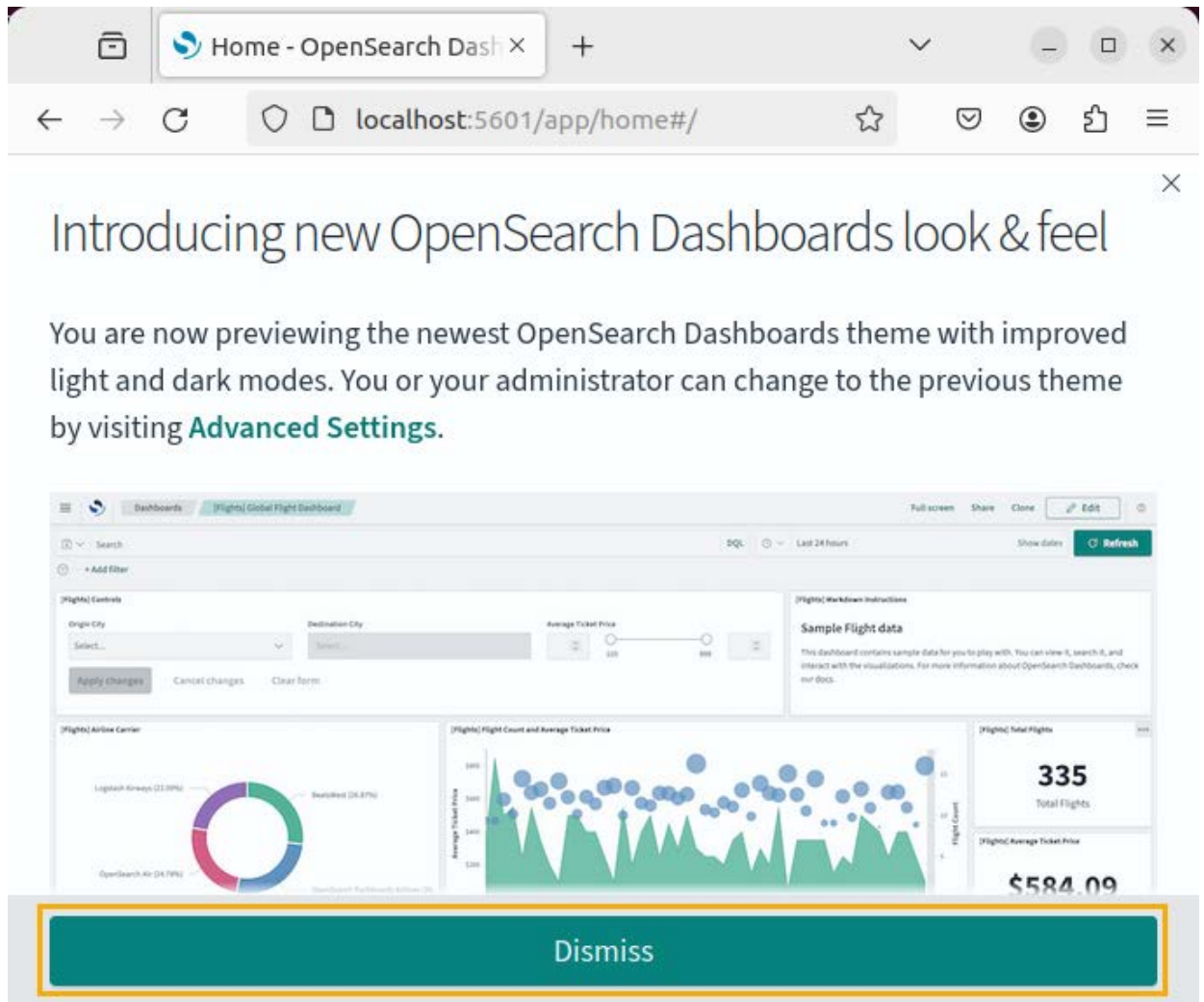
2. Input the **admin** username and password specified in the step 4 of the [OpenSearch server installation](#), then press the **Log in** button:




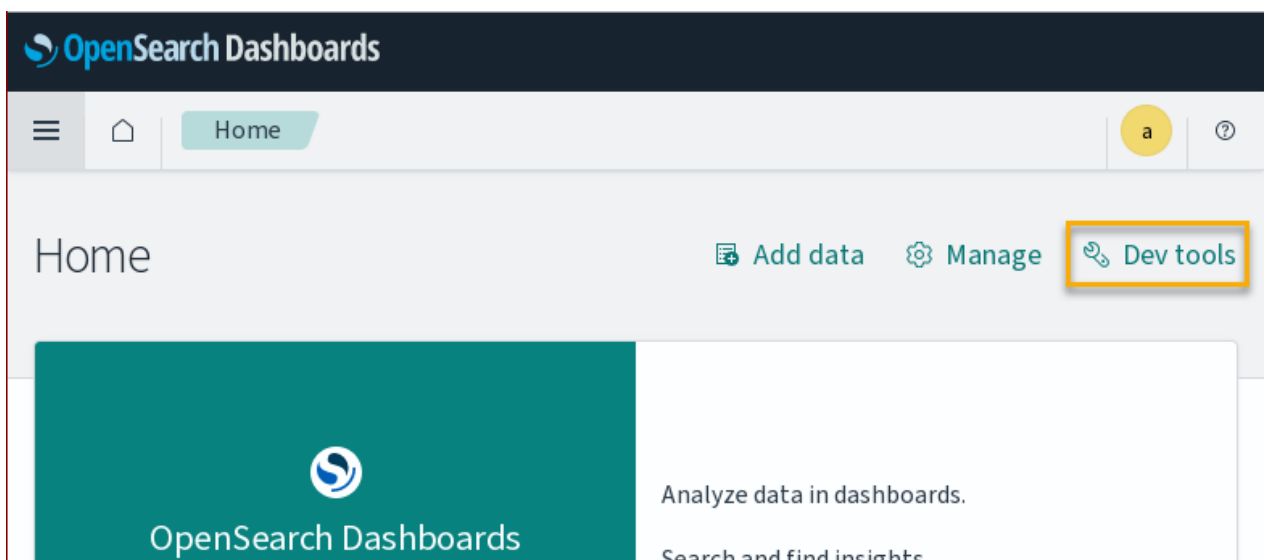
3. Skip the tenant selection by pressing **Cancel**, or select the desired tenant and press **Confirm**.



4. Skip the OpenSearch Dashboards new features overview by pressing **Dismiss**.

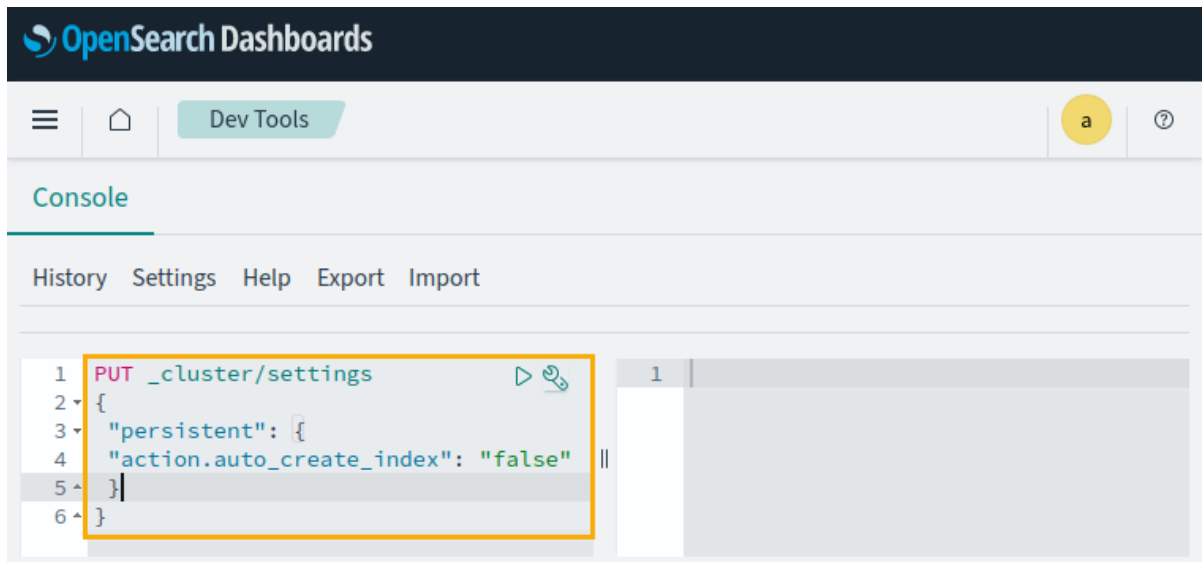



5. Open the OpenSearch Dashboards administrator console by pressing  **Dev Tools** in the upper right corner of the window:

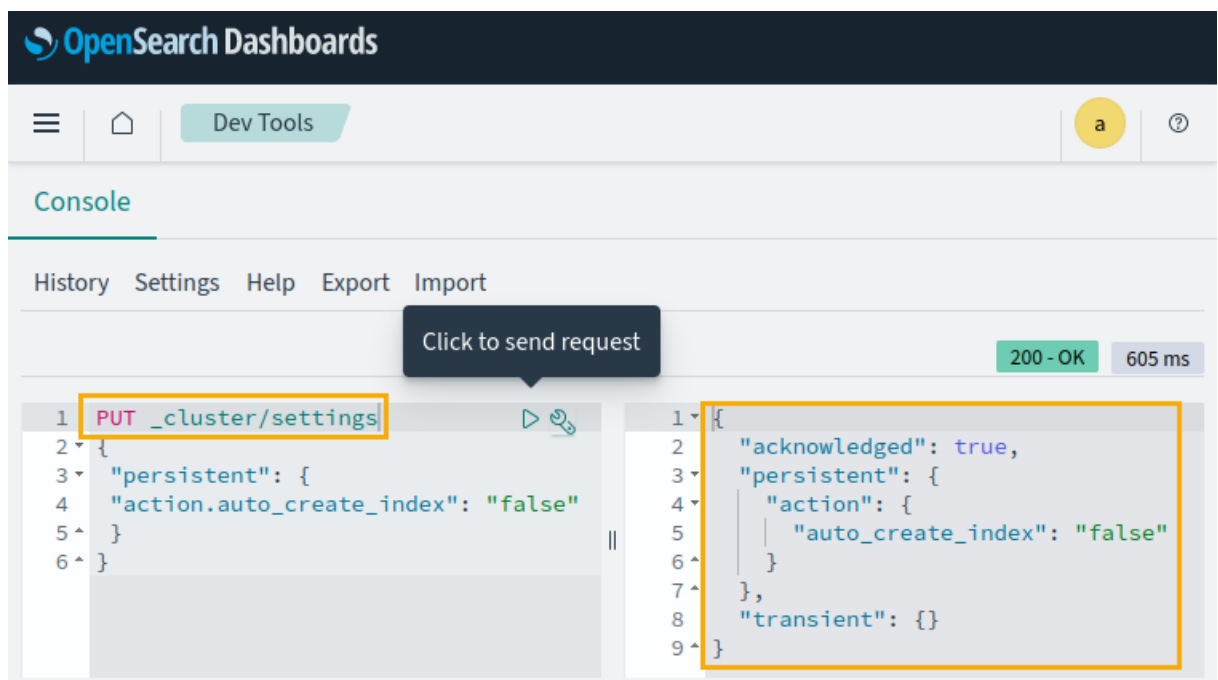


6. Input the following request into the left section of the OpenSearch Dashboards administrator console window:

```
PUT _cluster/settings
{
  "persistent": {
    "action.auto_create_index": "false"
  }
}
```



7. Send the request by clicking the  button to the right of its first line. Upon sending the request, the response will appear in the right section of the OpenSearch Dashboards console window. If the request is successful, the response looks like shown below:



If the request yields error, check for it being correctly inputted, and try sending again. If error still occurs, try to eliminate it using OpenSearch documentation or third-party resources.

8. Leave the OpenSearch Dashboards administrator console open, as you'll need it later for creating the index template.

Creating OpenSearch index template

An index template defines the settings according to which OpenSearch indexes will be created. In the index template for T-FLEX DOCs full-text search, it is recommended to create a text analyzer ("lang_analyzer") taking into account the morphology of the language based on the **Hunspell** dictionary. To do this you should:

1. Change working directory to **Downloads**:


```
cd ~/Downloads
```

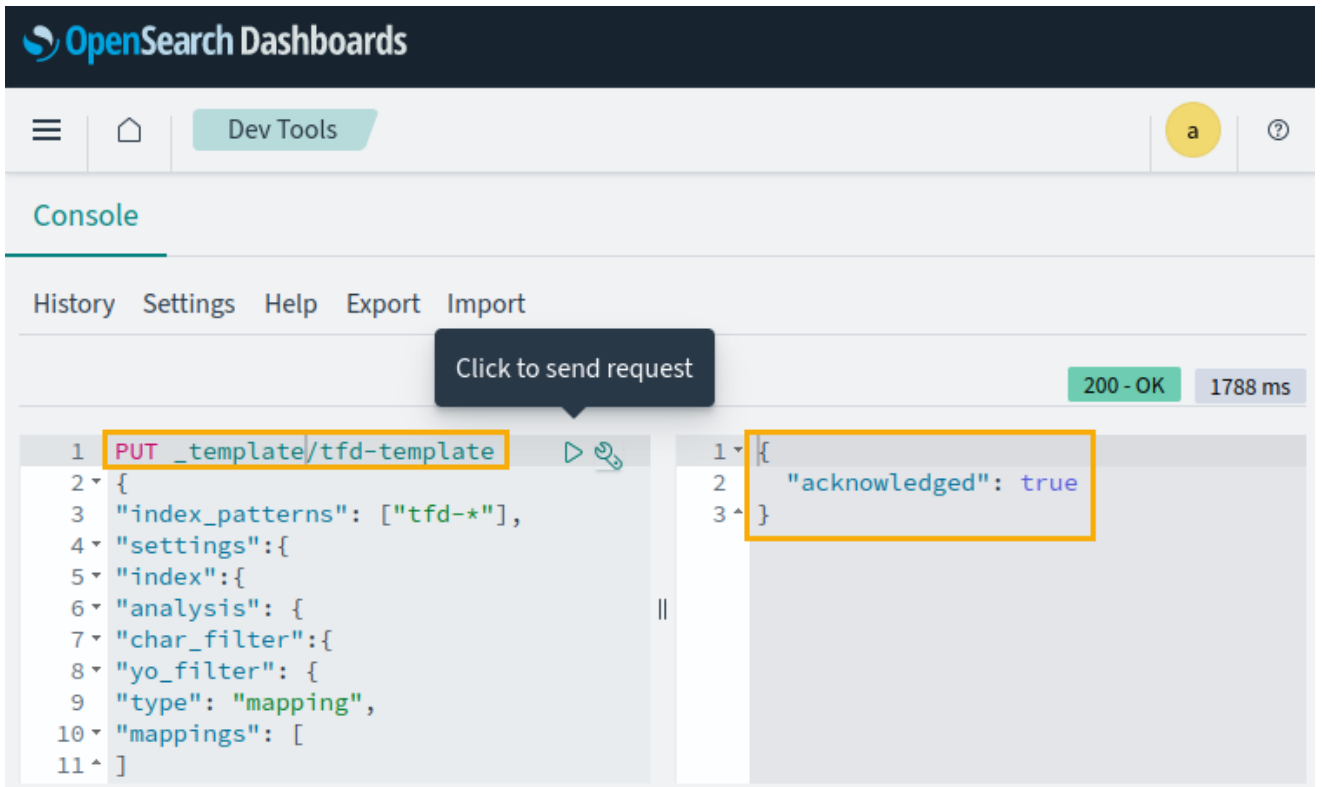
2. Move the **hunspell** folder, previously unpacked from the **T-FLEX_17_Linux_Components.zip** archive, into the **/etc/opensearch** directory:

```
sudo mv hunspell /etc/opensearch
```

3. Input a query that creates a text analyzer into the left section of the previously opened OpenSearch Dashboards administrator console window:

```
PUT _template/tfd-template
{
  "index_patterns": ["tfd-*"],
  "settings": {
    "index": {
      "analysis": {
        "char_filter": {
          "yo_filter": {
            "type": "mapping",
            "mappings": [
            ]
          }
        },
        "filter": {
          "hunspell_filter": {
            "type": "hunspell",
            "locale": "en_US"
          }
        }
      },
      "analyzer": {
        "lang_analyzer": {
          "tokenizer": "standard",
          "filter": [
            "lowercase",
            "hunspell_filter"
          ],
          "char_filter": [
            "yo_filter"
          ]
        }
      }
    }
  }
}
```

4. Send the request by clicking the  button to the right of its first line. Upon sending the request, the response will appear in the right section of the OpenSearch Dashboards console window. If the request is successful, the response looks like shown below:



If the request yields error, check for it being correctly inputted, and try sending again. If error still occurs, try to eliminate it using OpenSearch documentation or third-party resources.

5. Stop the OpenSearch Dashboards service:

```
sudo systemctl stop opensearch-dashboards
```

ATTENTION! For the full-text search to work, you should also enable it in the T-FLEX DOCs Server configuration file `ServerSetup.Example.Linux.xml`, as described in the [Configuring T-FLEX DOCs Server](#) section.

Configuring OpenSearch

1. Open the Opensearch configuration file:

```
sudo nano /etc/opensearch/opensearch.yml
```

2. In the **Network** section uncomment the **network.host** parameter by removing the # symbol at the start of the line and set the value to "0.0.0.0":

```
# ----- Network -----
#
# Set the bind address to a specific IP (IPv4 or IPv6):
#
network.host: 0.0.0.0
#
# Set a custom port for HTTP:
#
#http.port: 9200
#
# For more information, consult the network module documentation.
#
```

3. In the Discovery section uncomment the **cluster.initial_cluster_manager_nodes** parameter by removing the # symbol at the start of the line and set the value to the name or IP-address (in square brackets and quotes) of the computer the OpenSearch is installed on.

```
# ----- Discovery -----
#
# Pass an initial list of hosts to perform discovery when this node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
#discovery.seed_hosts: ["host1", "host2"]
#
# Bootstrap the cluster using an initial set of cluster-manager-eligible nodes:
#
cluster.initial_cluster_manager_nodes: ["ubuntu-555"]
#
# For more information, consult the discovery and cluster formation module documentation.
#
```

4. Add the following lines at the end of the file:

```
plugins.security.disabled: true
```

```
indices.analysis.hunspell.dictionary.ignore_case: true
```

```
node.max_local_storage_nodes: 3
##### End OpenSearch Security Demo Configuration #####
plugins.security.disabled: true
indices.analysis.hunspell.dictionary.ignore case: true
```

5. Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.
6. Restart the OpenSearch service:

```
sudo systemctl restart opensearch
```

ATTENTION! For the full-text search to work, you should also enable it in the T-FLEX DOCs Server configuration file **ServerSetup.Example.Linux.xml**, as described in the [Configuring T-FLEX DOCs Server](#) section.

Configuring T-FLEX DOCs Server

Instructions given below presume, that you have [installed T-FLEX DOCs Server](#) and performed all other actions required before its installation.

If you're using Debian, instructions given below also presume, that you have [provided superuser rights to the current user](#).

1. Open the T-FLEX DOCs server setup configuration file:

```
nano /opt/t-flex-docs/ServerSetup.Example.Linux.xml
```

In the opened file, specify desired parameters. Explanations of particular parameters are provided in comments within the file. The example of the minimal required data filling is provided below:

- 1.1. In the database connection settings section for the **<Database>** element set values of following attributes:

- **server**
Name or IP-address of a computer, where PostgreSQL is installed.
- **password**
PostgreSQL administrator password [specified earlier](#).

```
<Database
  platform="PostgreSql"
  server="localhost" port="5432"
  database="DOCs17"
  user="postgres"
  password="123"
  databasepassword="NewPassword.1"
  databaseFolder=""
  useWindowsIdentity="false"
  serverHomePath="/usr/bin/" />
```

- 1.2. In the applications server settings section for the **<AppServerConfig>** element set values of following attributes:

- **server**
Name or IP-address of a computer, where T-FLEX DOCs Server is installed.
- **password**
PostgreSQL administrator password [specified earlier](#).

```
<AppServerConfig
  platform="PostgreSql"
  startMode="Automatic"
  port="21321"
  http="false"
  httpPort="80"
  discovery="true"
  server="localhost"
  database="DOCs17"
  databasePort="5432"
  timeout="180"
  useWindowsIdentity="false"
  login="postgres"
  password="123"
  saveMailFileAttachmentsOnDisk="false"
  mailFileAttachmentsFolder=""
```

- 1.3. If a [full-text search platform is installed](#), in the same applications server settings section for the `<AppServerConfig>` element assign the `true` value to the `fullTextSearchingEnabled` attribute, and specify the Elasticsearch or OpenSearch server address in the `<ServerAddress>` element nested within the `<FullTextSearchConnectionSettings>` element:

```

uninstall="false"
fullTextSearchingEnabled="true">
<AuthorizationSettings type="LocalSystem" user="" password="" />
<AutoUpdate enabled="false" packagePath="" user="" password="" request="true" />
<Tasks days="false true true true true true false" tasks="RegenerateIndexes Shrink RefreshAcc
<FullTextSearchConnectionSettings>
  <ServerAddress>http://10.168.250.179:9200</ServerAddress>
  <BasicAuthenticationEnabled>>false</BasicAuthenticationEnabled>
  <BasicAuthenticationCredentials username="" password="" />
</FullTextSearchConnectionSettings>
</AppServerConfig>

```

- 1.4. If a [full-text search platform is installed](#), in the file server settings section for the `<FileServerConfig>` element, as well as for each of the `<Storage>` elements (it is one by default) nested within the `<Storages>` element, assign the `true` value to `fullTextSearchingEnabled` attributes, and specify the Elasticsearch or OpenSearch server address in the `<ServerAddress>` element nested within the `<FullTextSearchConnectionSettings>` element:

```

<!-- сервер файлов -->
<FileServerConfig
  startMode="Automatic"
  port="21322"
  http="false"
  install="false"
  start="false"
  uninstall="false"
  fullTextSearchIndexingEnabled="true">
  <AuthorizationSettings type="LocalSystem" user="" password="" />
  <Storages>
    <!-- разделы сервера файлов, нужно указать путь -->
    <Storage name="Основной" path="/opt/t-flex-docs/FileServer/Основной/" general="true" fullTextSearchIndexingEnabled="true" />
  </Storages>
  <FullTextSearchConnectionSettings>
    <ServerAddress>http://10.168.250.179:9200</ServerAddress>
    <BasicAuthenticationEnabled>>false</BasicAuthenticationEnabled>
    <BasicAuthenticationCredentials username="" password="" />
  </FullTextSearchConnectionSettings>
</FileServerConfig>

```

- 1.5. In the list of file reference storages, for each of the `<FileReferenceStorages>` elements nested within the `<FileReferenceStorages>` element set value of the following attribute:

- **address**

The value of this attribute consists of two parts separated by colon: first part is a name or IP-address of a computer, where T-FLEX DOCs Server is installed; second part is the number of the port to be used for connecting to the server.

```

<FileReferenceStorages>
  <FileReferenceStorage name="Основной" address="smolensk2:21322" storage="Основной"
  default="true" />
</FileReferenceStorages>

```

- 1.6. In the server components settings section, for each `<Configuration>` group of datasets within the `<Configurations>` list specify the following attribute:

- **included**

`true` value means datasets will be enabled, `false` – disabled.

- 1.7. Close the file by pressing `Ctrl+X` and confirm changes by typing `Y` and pressing `Enter`.

2. If you're installing multiple instances of T-FLEX DOCs Server on the same computer, create the instance configuration file:

```
nano /opt/t-flex-docs/Application.config
```

Use the current instance folder instead of `/opt/t-flex-docs`.

Fill the file with the following text:

```
<?xml version="1.0" encoding="utf-8"?>

<Application>

  <SETTINGS Language="Eng" Instance="InstanceName"/>

</Application>
```

where `InstanceName` should be replaced by the desired name of the instance.

Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

3. Change working directory to the T-FLEX DOCs Server folder:

```
cd /opt/t-flex-docs
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

4. Make T-FLEX DOCs Server Setup utility file executable:

```
sudo chmod +x TFlex.DOCs.ServerSetup.Console
```

Upon executing this command, the system asks for a root password. Type the password into the terminal and press **Enter**.

5. Launch T-FLEX DOCs Server Setup utility and wait for it to complete the setup:

```
./TFlex.DOCs.ServerSetup.Console ServerSetup.Example.Linux.xml /GO
```

Setup may take several minutes.

6. If you're using a firewall, open following ports listened by the T-FLEX DOCs Server:

- UDP port 21320
- TCP ports specified in the `ServerSetup.Example.Linux.xml` file for app server, event server and file server.

Ubuntu uses UFW firewall by default, so ports may be opened as follows:

```
sudo ufw allow 21320/udp

sudo ufw allow 21321,21322/tcp
```

where `21321` and `21322` may need to be replaced by your port numbers, if you have changed them from defaults in the `ServerSetup.Example.Linux.xml` file.

On other operating systems ports are open by default.

If you're using a different firewall, refer to its documentation.

Launching T-FLEX DOCs Server services

Instructions given below presume, that you have [installed](#) and [configured](#) T-FLEX DOCs Server and performed all other actions required before its installation.

T-FLEX DOCs Server services should be launched under the same user that performed [the server configuring](#).

Method 1. Launching T-FLEX DOCs Server services as generic application

In this method each of the three services is launched in its own separate terminal window. The windows should stay open as long as services are running. If you close a window, the corresponding service will be terminated.

1. Open the first terminal window and perform following actions:

- 1.1. Change working directory to the T-FLEX DOCs Server directory:

```
cd /opt/t-flex-docs
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

- 1.2. Make T-FLEX DOCs Server services files executable:

```
sudo chmod +x TFlex.DOCs.Server TFlex.DOCs.FileServer  
TFlex.DOCs.EventService
```

- 1.3. Launch the applications server service:

```
./TFlex.DOCs.Server
```

2. Open the second terminal window and perform following actions:

- 2.1. Change working directory to the T-FLEX DOCs Server directory:

```
cd /opt/t-flex-docs
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

- 2.2. Launch the file server service:

```
./TFlex.DOCs.FileServer
```

3. Open the third terminal window and perform following actions:

- 3.1. Change working directory to the T-FLEX DOCs Server directory:

```
cd /opt/t-flex-docs
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

- 3.2. Launch the event server service:

```
./TFlex.DOCs.EventService
```

Method 2. Launching T-FLEX DOCs Server services using OS services management tools

If you're using Debian, instructions given below presume, that you have [provided superuser rights to the current user](#).

1. Create the applications server service file in the home directory:

```
nano ~/docs-app-server.service
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, add instance name before `.service`.

Fill the file with the following text:

- If .NET 6.0 Runtime was installed [from internet-repository](#):

```
[Unit]
Description=T-FLEX DOCs Server
[Service]
User=*
ExecStart=/opt/t-flex-docs/TFlex.DOCs.Server --service

SuccessExitStatus=143
TimeoutStopSec=10
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

- If .NET 6.0 Runtime was installed [without using internet-repository](#):

```
[Unit]
Description=T-FLEX DOCs Server
[Service]
User=*
ExecStart=/opt/t-flex-docs/TFlex.DOCs.Server -service
Environment="DOTNET_ROOT=/opt/dotnet"
Environment="PATH=/opt/dotnet"

SuccessExitStatus=143
TimeoutStopSec=10
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

In both cases, `*` should be replaced by the name of the current user.

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

2. Create the file server service file in the home directory:

```
nano ~/docs-file-server.service
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, add instance name before `.service`.

Fill the file with the following text:

- If .NET 6.0 Runtime was installed [from internet-repository](#):

```
[Unit]
Description=T-FLEX DOCs FileServer
[Service]
User=*
ExecStart=/opt/t-flex-docs/TFlex.DOCs.FileServer --service

SuccessExitStatus=143
TimeoutStopSec=10
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

- If .NET 6.0 Runtime was installed [without using internet-repository](#):

```
[Unit]
Description=T-FLEX DOCs FileServer
[Service]
User=*
ExecStart=/opt/t-flex-docs/TFlex.DOCs.FileServer -service
Environment="DOTNET_ROOT=/opt/dotnet"
Environment="PATH=/opt/dotnet"

SuccessExitStatus=143
TimeoutStopSec=10
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

In both cases, `*` should be replaced by the name of the current user.

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

3. Create the event server service file in the home directory:

```
nano ~/docs-event-service.service
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, add instance name before `.service`.

Fill the file with the following text:

- If .NET 6.0 Runtime was installed [from internet-repository](#):

```
[Unit]
Description=T-FLEX DOCs EventService
[Service]
User=*
ExecStart=/opt/t-flex-docs/TFlex.DOCs.EventService -service

SuccessExitStatus=143
TimeoutStopSec=10
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

- If .NET 6.0 Runtime was installed [without using internet-repository](#):

```
[Unit]
Description=T-FLEX DOCs EventService
[Service]
User=*
ExecStart=/opt/t-flex-docs/TFlex.DOCs.EventService -service
Environment="DOTNET_ROOT=/opt/dotnet"
Environment="PATH=/opt/dotnet"

SuccessExitStatus=143
TimeoutStopSec=10
Restart=on-failure
RestartSec=5

[Install]
WantedBy=multi-user.target
```

In both cases, * should be replaced by the name of the current user.

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use the current instance folder instead of `/opt/t-flex-docs`.

Close the file by pressing **Ctrl+X** and confirm changes by typing **Y** and pressing **Enter**.

4. Copy files created on previous steps into the `/etc/systemd/system` directory:

```
sudo cp ~/docs-app-server.service /etc/systemd/system/docs-app-
server.service

sudo cp ~/docs-file-server.service /etc/systemd/system/docs-file-
server.service

sudo cp ~/docs-event-service.service /etc/systemd/system/docs-event-
service.service
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use names of services of the current instance.

5. Enable and launch services:

```
systemctl enable --now docs-app-server

systemctl enable --now docs-file-server

systemctl enable --now docs-event-service
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use names of services of the current instance.

Upon executing each of these commands, the system asks for a root password. Type the password and press **Enter**.

6. Check that the services are running:

```
systemctl status docs-app-server  
systemctl status docs-file-server  
systemctl status docs-event-service
```

If you're installing multiple instances of T-FLEX DOCs Server on the same computer, use names of services of the current instance.

If the services are not running, contact Top Systems technical support and provide information on performed actions and detected errors.

Stopping T-FLEX DOCs Server services

The method of stopping server services depends on how they were launched.

Method 1. Stopping T-FLEX DOCs Server services as generic applications

If T-FLEX DOCs server services [were launched as generic applications](#), perform following actions:

1. Switch to the terminal window, where `TFlex.DOCs.EventService` is running, and press **Ctrl+E**.
2. Switch to the terminal window, where `TFlex.DOCs.Server` is running, and press **Ctrl+E**.
3. Switch to the terminal window, where `TFlex.DOCs.FileServer` is running, and press **Ctrl+E**.

Method 2. Stopping T-FLEX DOCs Server services using OS services management tools

If T-FLEX DOCs server services [are launched using OS services management tools](#), perform following actions:

1. Stop the event server service:

```
systemctl stop docs-event-service
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use name of service of the desired instance instead of **docs-event-service**.

Upon executing the command, the system asks for a root password. Type the password and press **Enter**.

2. Stop the applications server service:

```
systemctl stop docs-app-server
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use name of service of the desired instance instead of **docs-app-service**.

Upon executing the command, the system asks for a root password. Type the password and press **Enter**.

3. Stop the file server service:

```
systemctl stop docs-file-server
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use name of service of the desired instance instead of **docs-file-service**.

Upon executing the command, the system asks for a root password. Type the password and press **Enter**.

Updating T-FLEX DOCs Server

1. Put the T-FLEX DOCs Server installation archive of the new version into the **Downloads** directory of the target computer. The name of the archive complies with the following pattern:

```
t-flex-docs-*.tar.gz
```

where ***** is replaced by a version number.

2. Copy the server setup configuration file into the home directory:

```
cp /opt/t-flex-docs/ServerSetup.Example.Linux.xml  
~/ServerSetup.Example.Linux.xml
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use folder of the instance to be updated instead of `/opt/t-flex-docs`.

3. Stop T-FLEX DOCs Server services, as described [in the previous section](#).
4. Unpack the T-FLEX DOCs Server installation archive of the new version into the directory of the installed version:

```
sudo tar -xzf ~/Downloads/t-flex-docs-*.tar.gz -C /opt/t-flex-docs
```

Use the name of the actual archive instead of `t-flex-docs-*.tar.gz`.

If there are multiple instances of T-FLEX DOCs Server on the same computer, use folder of the instance to be updated instead of `/opt/t-flex-docs`.

Upon executing the command, the system asks for a root password. Type the password and press **Enter**.

5. Copy the `ServerSetup.Example.Linux.xml` file from the home directory into the T-FLEX DOCs Server directory:

```
cp ~/ServerSetup.Example.Linux.xml /opt/t-flex-  
docs/ServerSetup.Example.Linux.xml
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use folder of the instance to be updated instead of `/opt/t-flex-docs`.

6. Change working directory to the T-FLEX DOCs Server folder:

```
cd /opt/t-flex-docs
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use folder of the instance to be updated instead of `/opt/t-flex-docs`.

7. Make T-FLEX DOCs Server Setup utility file executable:

```
sudo chmod +x TFlex.DOCs.ServerSetup.Console
```

8. Launch T-FLEX DOCs Server Setup utility and wait for it to complete the setup:

```
./TFlex.DOCs.ServerSetup.Console ServerSetup.Example.Linux.xml /GO
```

Setup may take several minutes.

Uninstalling T-FLEX DOCs Server

1. Stop T-FLEX DOCs Server services, as described [in the corresponding section](#).
2. If actions described in the [Launching T-FLEX DOCs Server services using OS services management tools](#) section were performed, disable automatic launch of server services:

```
systemctl disable docs-app-server  
  
systemctl disable docs-file-server  
  
systemctl disable docs-event-service
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use names of services of the instance to be uninstalled.

Upon executing each of these commands, the system asks for a root password. Type the password and press **Enter**.

3. If actions described in the [Launching T-FLEX DOCs Server services using OS services management tools](#) section were performed, remove files of server services:

```
sudo rm /etc/systemd/system/docs-app-server.service  
  
sudo rm /etc/systemd/system/docs-file-server.service  
  
sudo rm /etc/systemd/system/docs-event-service.service
```

If there are multiple instances of T-FLEX DOCs Server on the same computer, use names of services of the instance to be uninstalled.

Upon executing the first command, the system asks for a root password. Type the password and press **Enter**.

4. Remove T-FLEX DOCs Server directory:

```
sudo rm -R /opt/t-flex-docs
```

5. Log into **psql** as **postgres** service user:

```
sudo -u postgres psql
```

6. Delete database:

```
DROP DATABASE "DOCs17";
```

where **DOCs17** is the name of the database specified in the [ServerSetup.Example.Linux.xml](#) file

7. Exit **psql**:

```
\q
```

Contact us to request information about T-FLEX PLM software,
to reach our Support, or to come up with your ideas on
cooperating with Top Systems

Contact Us

www.tflex.com/company/contactus/



www.tflex.com

ADVANCED INTEGRATED CAD/CAM/CAE/PLM SOLUTIONS

+7 (499) 973-20-34
+7 (499) 973-20-35

marketing@topsystems.ru

