

01 March 2025

# Opentheso

Opentheso v25.03.01

Miled Rousset

#### **Continue**

- 1 Installation
- 2 Update

# 1. Installation under Debian (installation carried out with Debian 12)

#### **Prerequisites for installing Opentheso:**

- Java 17 and above
- · Postgresql 15 and above

#### Java installation

- 1. Install the latest version of
  - apt install default-jre
  - java -version

openjdk version "17.0.8" 2023-07-18 OpenJDK Runtime Environment (build 17.0.8+7-Debian-1deb12u1) OpenJDK 64-Bit Server VM (build 17.0.8+7-Debian-1deb12u1, mixed mode, sharing)

#### **Installing Postgresql**

- 2. Install the server
  - apt-get install postgresql
  - psql --version

psql (PostgreSQL) 15.3 (Debian 15.3-0+deb12u1)

- 3. Changing the Postgres password
  - · passwd postgres
  - New password:
  - Enter it again:
- 1. Create the role and database: (do not enter the text in green)
  - su postgres
  - psql
  - postgres=# CREATE USER opentheso PASSWORD 'opentheso';
  - postgres=# create Database opentheso OWNER opentheso;
  - postgres=# ALTER USER opentheso WITH SUPERUSER;
  - \q
  - Ctrl + d

#### Apache2

If Apache2 is not installed, you need to install it with the command:

• apt install apache2

#### **Apache settings (Virtual Host)**

(this is an example of the settings in /ect/apache2/sites-enabled/)

You need to create a conf file specific to Opentheso. Here is an example following the installation described here:

- cd /etc/apache2/sites-enabled/
- · vi opentheso.conf

```
SSLEngine On
   <Directory /opentheso>
       Order allow, deny
       Allow from all
   </Directory>
         ProxyPreserveHost On
         <Location "/META-INF/">
       deny from all
   </Location>
   <Location "/WEB-INF/">
       deny from all
   </Location>
        <IfModule mod rewrite.c>
               RewriteEngine On
                RewriteRule ^/$ /opentheso/$1 [R]
   </IfModule>
</VirtualHost>
```

Activate the proxy, SSL and rewrite modules

· a2enmod proxy\_http ssl headers rewrite

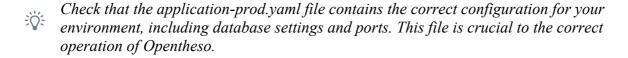
#### Restart apache

· systemctl restart apache2

#### **Installing Opentheso:**

*Version 25.01.01 (this is the current version at the time of writing)* 

- Create the opentheso folder in /opt/opentheso:
  - mkdir /opt/opentheso
- Retrieve the files needed to run Opentheso: https://github.com/miledrousset/Opentheso/releases/
  - opentheso.jar
  - application-prod.yaml
  - logback-spring.xml
- Copy files to



- Create a folder for Handle certificates:
  - mkdir /opt/opentheso/certificates
- If you are using perennial Handle identifiers based on a server, you need to copy the admpriv.bin file to this folder.

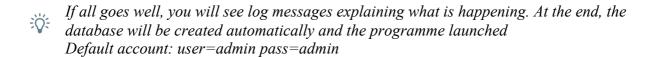
- Adjust permissions on the /opt/opentheso folder :
  - cd /opt
  - chown miled -R opentheso/
  - chgrp miled -R opentheso/
- Modification of the "application-prod.yaml" file :

```
server:
 port: 8099
  forward-headers-strategy: native
  servlet:
   context-path: /
   session:
      timeout: 120m
  error:
   path: /errorPages/error500.xhtml
   whitelabel:
      enabled: false
primefaces:
 uploader: native
spring:
  servlet:
   multipart:
      enabled: true
     max-file-size: 100MB #Defines the maximum size of the file to be
downloaded
      max-request-size: 100MB # Total max request size
  datasource:
   url: jdbc:postgresql://localhost:5432/opentheso
   password: opentheso
   username: opentheso
 jpa:
 show-sql: true
 generate-ddl: true
   hibernate:
     ddl-auto: validate
   properties:
      hibernate:
        format sql: true
        dialect: org.hibernate.dialect.PostgreSQLDialect
  liquibase:
   enabled: true
   change-log: classpath:/changelog/db.changelog.xml
graphql:
 path: /graphql # GraphQL API path
  graphigl:
   enabled: true # Enables the graphiql interface for testing queries
   path: /graphiql # Path to the graphiql interface
  schema:
   printer:
      enabled: false # Enables schema printing at startup
settings:
 workLanguage: fr
```

```
certificates:
  admpriv: ./certificates/admpriv.bin
  cacerts2: ./certificates/cacerts2
 key: ./certificates/key.p12
smpt:
 protocol: smtp
 hostname: smtprelay.mondomaine.fr
 portNumber: 25
 authorization: false
 mailFrom: opentheso@ mondomaine.fr
 transportMail: smtp
neo4j:
  serverName: localhost
  serverPort: 7687
 databaseName: neo4j
 user: neo4j
 password: neo4j1234
ldap:
  security:
   authentication: simple
  server:
   url: ldapr3.mondomaine.fr
  initial:
   context:
     factory: com.sun.jndi.ldap.LdapCtxFactory
  key:
   store:
     password: changeit
     path: /usr/lib/jvm/java-11-openjdk-amd64/lib/security/cacerts
   trust:
     password: changeit
     path: /usr/lib/jvm/java-11-openjdk-amd64/lib/security/cacerts
```

#### • Start Opentheso:

• java -jar -Dlogging.config=./logback-spring.xml opentheso.jar --spring.profiles.active=prod --spring.config.additional-location=application-prod.yaml



• Connecting to Opentheso using a Web browser:

https://opentheso.mondomaine.fr:9099/opentheso https://opentheso.mondomaine.fr/opentheso

#### Create a systemd service file

To start and stop Opentheso automatically, you need to create a service file.

- 1. Open a terminal and run the following command to create a new service file in /etc/systemd/system/:
  - sudo nano /etc/systemd/system/opentheso.service
- 2. Add the following lines to the opentheso.service file:

```
[Unit]
Description=Opentheso Java Application
After=network.target
Servicel
# User under which the service will be run
User=miled
# Work directory
WorkingDirectory=/opt/opentheso
# Command to be executed
ExecStart=java -jar -Dlogging.config=./logback-spring.xml opentheso.jar
--spring.profiles.active=prod --spring.config.additional-
location=application-prod.
# Restart automatically if the service crashes
Restart=always
RestartSec=10
[Install]
WantedBy=multi-user.target
```

- 3. User: Replace username with the user under which you wish to run the service. If you want this to be the root user, you can leave this line out, but it is often preferable to use a dedicated user for security reasons.
- 4. WorkingDirectory: Replace /path/to/the/folder/of/your/application with the path to the .jar file and the application-prod.yaml file.

#### Update and activate the service

- 1. Save the file and close the editor (for nano, press Ctrl + X, then Y and Enter).
- 2. Reload systemd so that it takes the new service file into account:
  - sudo systemctl daemon-reload
- 3. Enable the service so that it starts automatically when the system is booted:
  - sudo systemctl enable opentheso.service
- 4. Start the service immediately to check that it is working correctly:
  - sudo systemctl start opentheso.service

#### **Check service status**

Check that the service works correctly with this command:

• sudo systemctl status opentheso.service

If all goes well, we should see:

• opentheso.service - Opentheso Java Application Loaded: loaded (/etc/systemd/system/opentheso.service; enabled; vendor preset: enabled) Active: active (running) since ... Main PID: ... Tasks: ... Memory: ... ...

#### Other useful commands

- Stop the service :
  - sudo systemctl stop opentheso.service
- Restart the service :
  - sudo systemctl restart opentheso.service
- Disable automatic start :
  - sudo systemctl disable opentheso.service

## 2. Update

### To update Opentheso:

Download the latest version of the JAR on GitHub:

https://github.com/miledrousset/Opentheso/releases/

Stop the Openheso service:

• sudo systemctl stop opentheso.service

Replace the old opentheso.jar file with the new opentheso.jar file

Restart the Opentheso service:

• sudo systemctl start opentheso.service