

Opentheso

Opentheso v25.01.01

Miled Rousset

Contenu :

1 – Installation

2 – Mise à jour

1. Installation sous Debian (installation effectuée avec Debian 12)

Prérequis pour installer Opentheso :

- Java 17 et plus
- Postgresql 15 et plus

Installation de Java

1. Installez la dernière version Openjdk

- `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)

Installation de Postgresql

2. Installez le serveur Postgresql

- `apt-get install postgresql`
- `psql --version`

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

3. Changer le mot de passe de Postgres

- `passwd postgres`
- Nouveau mot de passe :
- Saisissez-le à nouveau :

1. Créer le rôle et la base de données : *(ne pas saisir le texte en vert)*

- `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

Si Apache2 n'est pas installé, il faut l'installer avec la commande :

- `apt install apache2`

Parametrage de Apache (Virtual Host)

(ceci est un exemple de paramétrage dans /etc/apache2/sites-enabled/)

Il faut créer un fichier de conf propre à Opentheso, voici un exemple suivant l'installation décrite ici :

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

```
<VirtualHost *:443>
    ServerAdmin moi@mondomaine.fr
    ServerName opentheso.mondomaine.fr

    ProxyPass /opentheso <http://localhost:8099/opentheso> timeout=3600
    ProxyPassReverse /opentheso <http://localhost:8099/opentheso>
    timeout=3600

    # Ajout des en-têtes nécessaires pour HTTPS
    RequestHeader set X-Forwarded-Proto https
    RequestHeader set X-Forwarded-Port 443

    ProxyPreserveHost On
    ProxyRequests Off
```

```

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>

```

Activer les modules proxy, SSL et rewrite :

- [a2enmod proxy_http ssl headers rewrite](#)

Relancer apache

- [systemctl restart apache2](#)

Installation d'Opentheso :

Version 25.01.01 (c'est la version actuelle à la rédaction de la doc)

- Créer le dossier opentheso dans /opt/opentheso :
 - [mkdir /opt/opentheso](#)
- Récupérer les deux fichiers nécessaires pour le fonctionnement de Opentheso : <https://github.com/miledrousset/Opentheso/releases/>
 - **opentheso.jar**
 - **application-prod.yaml**
- Copier les deux fichiers dans /opt/opentheso



Vérifiez que le fichier application-prod.yaml contient la configuration correcte pour votre environnement, notamment les paramètres de base de données et les ports. Ce fichier est crucial pour le bon fonctionnement d'Opentheso.

- Créer un dossier pour les certificats pour Handle :
 - [mkdir /opt/opentheso/certificats](#)



Si vous utilisez des identifiants pérennes Handle basés sur un serveur, il faut copier dans ce dossier le fichier `admpriv.bin`

- Ajuster les droits sur le dossier /opt/opentheso :
 - `cd /opt`
 - `chown miled -R opentheso/`
 - `chgrp miled -R opentheso/`
- Modification du fichier « application-prod.yaml » :

```
server:
  port: 8099
  forward-headers-strategy: native
  servlet:
    context-path: /
    session:
      timeout: 125m
  error:
    path: /errorPages/error500.xhtml
    whitelabel:
      enabled: false

primefaces:
  uploader: native

spring:
  servlet:
    multipart:
      enabled: true
      max-file-size: 100MB #Définit la taille max, fichier à télécharger
      max-request-size: 100MB # Taille totale max de la requête
  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

settings:
  workLanguage: fr

certificats:
  admpriv: ./certificats/admpriv.bin
  cacerts2: ./certificats/cacerts2
  key: ./certificats/key.pl2

smtp:
  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

```

- Démarrer Opentheso :

- `java -jar opentheso.jar --spring.profiles.active=prod --spring.config.additional-location=application-prod.yaml`



*Si tout va bien, on verra des messages de log qui expliquent ce qui se passe. A la fin, la base de données sera créée automatiquement et le programme se lance.
Compte par défaut : user=admin pass=admin*

- Se connecter à Opentheso avec le navigateur Web :

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

Créer un fichier de service systemd

Pour pouvoir lancer et arrêter Opentheso automatiquement, il faut créer un fichier de service.

1. Ouvrir un terminal et exécuter la commande suivante pour créer un nouveau fichier de service dans `/etc/systemd/system/` :
 - `sudo nano /etc/systemd/system/opentheso.service`
2. Ajouter les lignes suivantes dans le fichier `opentheso.service` :

```

[Unit]
Description=Opentheso Java Application

```

```

After=network.target
Service]
# Utilisateur sous lequel le service sera exécuté
User=miled
# Répertoire de travail
WorkingDirectory=/opt/opentheso
# Commande à exécuter
ExecStart=java -jar opentheso.jar --spring.profiles.active=prod --
spring.config.additional-location=application-prod.yaml
# Redémarrer automatiquement si le service plante
Restart=always
RestartSec=10
[Install]
WantedBy=multi-user.target

```

3. User : Remplacez nom_utilisateur par l'utilisateur sous lequel on souhaite exécuter le service. Si on veut que ce soit l'utilisateur root, on peut laisser cette ligne de côté, mais il est souvent préférable d'utiliser un utilisateur dédié pour des raisons de sécurité.
4. WorkingDirectory : Remplacez /chemin/vers/le/dossier/de/votre/application par le chemin où se trouve le fichier .jar et le fichier application-prod.yaml.

Actualiser et activer le service

1. Sauvegardez le fichier et fermez l'éditeur (pour nano, appuyez sur Ctrl + X, puis Y et Enter).
2. Rechargez systemd pour qu'il prenne en compte le nouveau fichier de service :
 - `sudo systemctl daemon-reload`
3. Activez le service pour qu'il démarre automatiquement au démarrage du système :
 - `sudo systemctl enable opentheso.service`
4. Démarrez le service immédiatement pour tester s'il fonctionne correctement :
 - `sudo systemctl start opentheso.service`

Vérifier le statut du service

Vérifiez si le service fonctionne correctement avec cette commande :

- `sudo systemctl status opentheso.service`

Si tout se passe bien, on devrait voir :

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

Autres commandes utiles

- **Arrêter le service :**
 - `sudo systemctl stop opentheso.service`
- **Redémarrer le service :**
 - `sudo systemctl restart opentheso.service`
- **Désactiver le démarrage automatique :**
 - `sudo systemctl disable opentheso.service`

2. Mise à jour

Pour mettre à jour Opentheso :

Télécharger la dernière version du JAR sur GitHub :

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

Arrêter le service Opentheso :

- `sudo systemctl stop opentheso.service`

Remplacer l'ancien fichier opentheso.jar par le nouveau fichier opentheso.jar

Relancer le service Opentheso :

- `sudo systemctl start opentheso.service`