

OCS NG-GLPI | Gestion de parc informatique

1. Présentation de la Solution et Architecture

1.1. Définition et Fonctionnalités

Dans le cadre de la gestion du parc informatique, nous avons retenu la solution **OCS Inventory NG** (Open Computer and Software Inventory Next Generation). Cette application permet de réaliser un inventaire automatisé sur la configuration des machines et du réseau, ainsi que sur les logiciels installés.

Ses fonctionnalités principales incluent :

- **L'inventaire** : Visualisation complète du parc via une interface web.
- **Le télédéploiement** : Capacité d'uploader des paquets (logiciels, commandes, fichiers) pour les installer à distance sur les postes clients en utilisant le protocole HTTP/HTTPS.

1.2. Architecture Technique

Le fonctionnement d'OCS Inventory repose sur un concept **Client/Serveur**.

Le serveur se compose de trois entités distinctes :

1. **Le Serveur de communication** : Il collecte, classe et archive les informations envoyées par les postes clients. Il s'appuie sur Apache, MariaDB et PHP, ainsi que sur des modules PERL et des scripts CGI.
2. **Le Serveur d'administration** : C'est l'interface Web (écrite en PHP) permettant aux techniciens de consulter l'inventaire et de gérer les droits.
3. **Le Serveur de déploiement** : Il assure la distribution centralisée des logiciels et mises à jour via Apache SSL (Secure Sockets Layer) pour sécuriser les échanges.

Côté poste client, un **Agent** est installé (disponible sous Windows, Linux, MacOS) pour remonter les informations au serveur.

2. Préparation de l'Environnement Serveur

2.1. Accès distant et Sécurisation

Afin d'administrer le serveur Linux (VM Ubuntu) à distance, le service SSH a été installé.

- **Commande :** `sudo apt install openssh-server`
- **Vérification :** Le service est actif (active/running).

```
thomas@apache2:~$ systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2026-01-26 14:11:49 CET; 22min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 3665 (sshd)
    Tasks: 1 (limit: 4542)
   Memory: 6.5M
      CPU: 716ms
   CGroup: /system.slice/ssh.service
           └─3665 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

janv. 26 14:11:49 apache2 systemd[1]: Starting OpenBSD Secure Shell server...
janv. 26 14:11:49 apache2 sshd[3665]: Server listening on 0.0.0.0 port 22.
janv. 26 14:11:49 apache2 systemd[1]: Started OpenBSD Secure Shell server.
janv. 26 14:11:49 apache2 sshd[3665]: Server listening on :: port 22.
janv. 26 14:19:26 apache2 sshd[3836]: Accepted password for thomas from 192.168.11.1 port 61230 ssh2
janv. 26 14:19:26 apache2 sshd[3836]: pam_unix(sshd:session): session opened for user thomas(uid=1000) by (uid=0)
janv. 26 14:19:27 apache2 sshd[3839]: Accepted password for thomas from 192.168.11.1 port 61231 ssh2
janv. 26 14:19:27 apache2 sshd[3839]: pam_unix(sshd:session): session opened for user thomas(uid=1000) by (uid=0)
thomas@apache2:~$
```

Figure 1 : Vérification de l'état du service SSH sur le serveur.

L'interopérabilité a été validée par une connexion réussie depuis un poste physique Windows via le client MobaXterm.

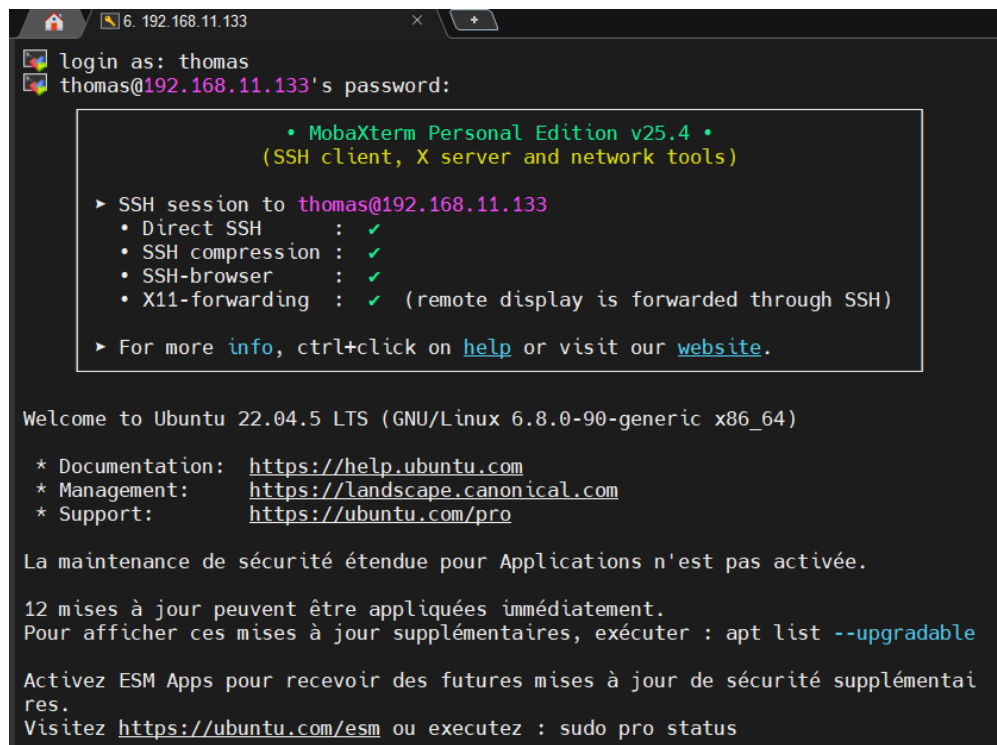


Figure 2 : Validation de la connexion distante SSH.

2.2. Installation de la pile LAMP (Linux Apache MariaDB PHP)

Conformément aux prérequis de l'architecture OCS, nous avons installé les composants suivants :

- **Serveur Web** : Apache2
- **Base de données** : MariaDB Server
- **Langage de script** : PHP

```
thomas@apache2:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2026-01-26 14:05:28 CET; 34min ago
     Docs: https://httpd.apache.org/docs/2.4/
    Main PID: 1020 (apache2)
      Tasks: 55 (limit: 4542)
     Memory: 8.1M
        CPU: 664ms
    CGroup: /system.slice/apache2.service
            └─1020 /usr/sbin/apache2 -k start
              └─1024 /usr/sbin/apache2 -k start
                └─1025 /usr/sbin/apache2 -k start

janv. 26 14:05:27 apache2 systemd[1]: Starting The Apache HTTP Server...
janv. 26 14:05:28 apache2 apachectl[1002]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name,
janv. 26 14:05:28 apache2 systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

Figure 3 : Vérification du bon fonctionnement du service Apache2.

OCS utilisant massivement le langage Perl pour son moteur de communication, l'installation de bibliothèques spécifiques (Perl et modules PHP associés) a été nécessaire (ex:

`libapache2-mod-perl2`, `php-soap`, `libxml-simple-perl`).

```
395 sudo apt install -y libapache2-mod-perl2
396 sudo apt install -y libxml-simple-perl
397 sudo apt install -y libnet-ip-perl
398 sudo apt install -y libsoap-lite-perl
399 sudo apt install -y libapache2-mod-perl2-dev
400 sudo apt install -y make
401 sudo apt install -y php-mysql
402 sudo apt install -y php-gd
403 sudo apt install -y php-mbstring
404 sudo apt install -y php-soap
405 sudo apt install -y php-xml
406 sudo apt install -y php-curl
407 sudo perl -MCPAN -e "install XML::Entities"
408 sudo cpan Apache2::SOAP
409 sudo cpan SOAP::Lite
410 sudo cpan Mojolicious::Lite
411 sudo cpan Switch
```

Figure 4 : Installation des dépendances Perl et PHP.

3. Installation et Configuration du Serveur OCS

3.1. Déploiement des sources

L'archive contenant les sources du serveur a été téléchargée (`wget`) puis décompressée (`tar xvzf`).

```
thomas@apache2:~$ tar xvzf OCSNG_UNIX_SERVER-2.10.0.tar.gz
OCSNG_UNIX_SERVER-2.10.0/
OCSNG_UNIX_SERVER-2.10.0/Apache/
OCSNG_UNIX_SERVER-2.10.0/Apache/Makefile.PL
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/INSTALL
OCSNG_UNIX_SERVER-2.10.0/Apache/MANIFEST
OCSNG_UNIX_SERVER-2.10.0/Apache/LICENSE
OCSNG_UNIX_SERVER-2.10.0/Apache/Changes
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Plugins/
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Plugins/Modules.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Interface.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/SOAP.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Plugins.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Constants.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Communication.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Groups.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Software.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Update/
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Update/Hardware.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Update/AccountInfos.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Cache.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Filter.pm
OCSNG_UNIX_SERVER-2.10.0/Apache/Ocsinventory/Server/Inventory/Export.pm
```

Figure 5 : Extraction de l'archive d'installation.

3.2. Exécution du script d'installation (Gestion d'incident)

L'installation est réalisée via le script interactif `setup.sh`. Ce script configure le serveur de communication et l'interface d'administration.

Incident rencontré : Erreur de chemin de configuration Lors de la question sur le répertoire de configuration d'Apache, une erreur de saisie a été commise : `/etc/apache2/confenabled` au lieu de `/etc/apache2/conf-available`.

- **Conséquence** : Le script a retourné l'erreur "not a directory", ne pouvant écrire le fichier de configuration.
- **Résolution** : Le chemin a été corrigé lors de la ressaisie, permettant au script de finaliser la configuration.

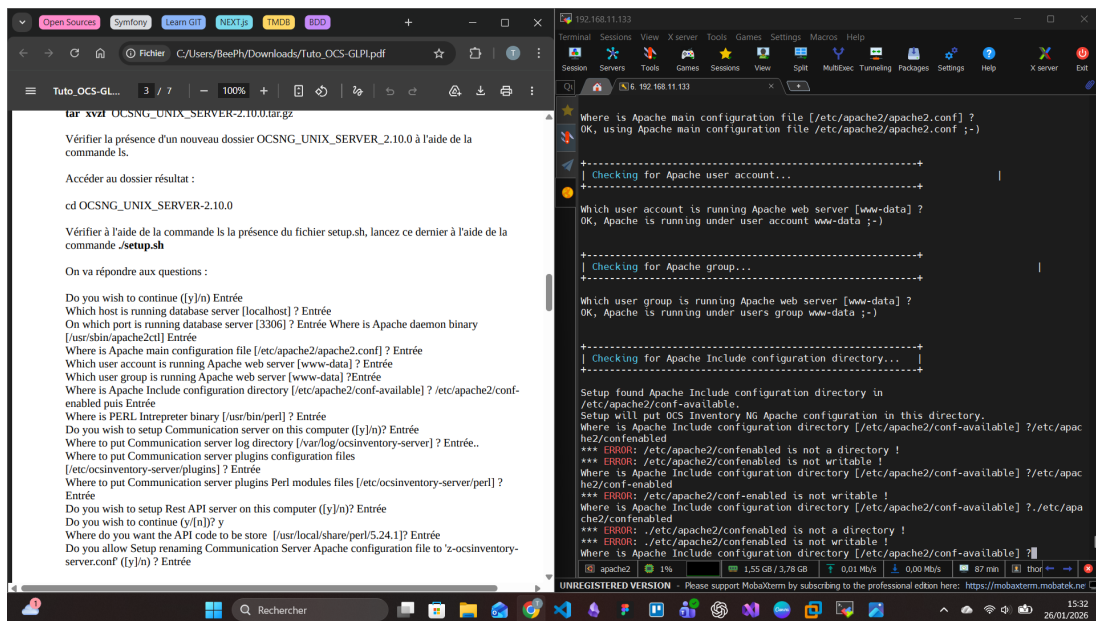


Figure 6 : Erreur de chemin lors de l'exécution du script setup.sh et correction.

Le script confirme la fin de l'installation et rappelle la nécessité de redémarrer le démon Apache.

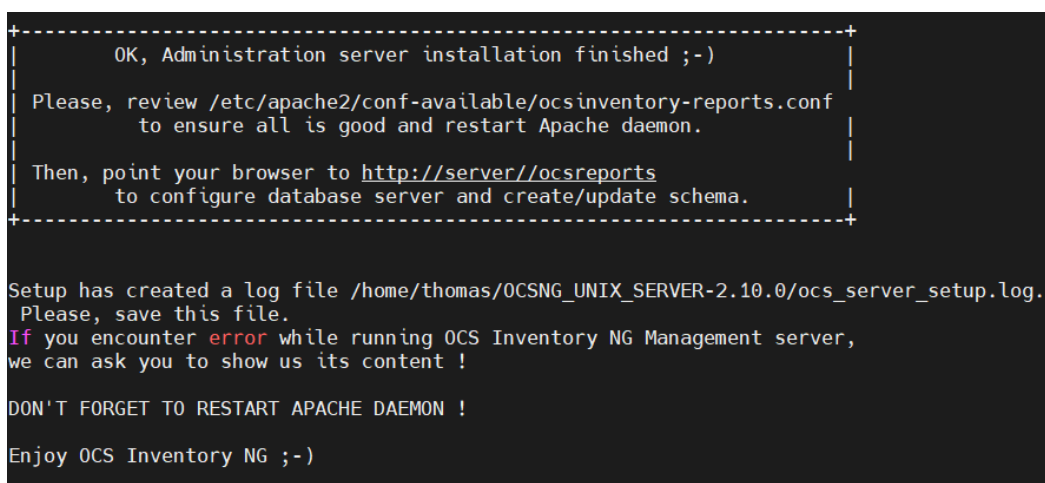


Figure 7 : Succès de l'installation du moteur OCS.

4. Configuration de la Base de Données

Le serveur nécessite une base de données nommée `ocsweb` et un utilisateur dédié `ocs`.

Difficulté technique : Authentification Root Une tentative de connexion via `mysql -u root` a échoué (Erreur 1698). Sur les distributions récentes, l'utilisateur root utilise le plugin d'authentification `unix_socket`, nécessitant des privilèges système.

- **Solution :** Utilisation de la commande `sudo mysql -u root`.

Une fois connecté, la base de données et l'utilisateur ont été créés avec les privilèges appropriés (`GRANT ALL PRIVILEGES`).

```
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo mysql -u root
[sudo] Mot de passe de thomas :
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.6.23-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database ocsweb;
Query OK, 1 row affected (0,001 sec)

MariaDB [(none)]>
MariaDB [(none)]> ls
-> ls
-> ^C
MariaDB [(none)]> clear
MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO ocs@'localhost' IDENTIFIED BY 'ocs';
Query OK, 0 rows affected (0,007 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,009 sec)

MariaDB [(none)]> exit;
Bye
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ service mariadb restart
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication requise pour redémarrer « mariadb.service ».
Authenticating as: Thomas Montout,,, (thomas)
Password:
==== AUTHENTICATION COMPLETE ====
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ service apache2 restart
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication requise pour redémarrer « apache2.service ».
Authenticating as: Thomas Montout,,, (thomas)
Password:
==== AUTHENTICATION COMPLETE ====
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ █
```

Figure 8 : Création de la base de données ocsweb et de l'utilisateur.

5. Résolution d'Incident Majeur : Configuration Apache/PHP

Lors de la première tentative d'accès à l'interface web (<http://IP/ocsreports>), le navigateur a affiché le code source PHP brut au lieu de la page web attendue.

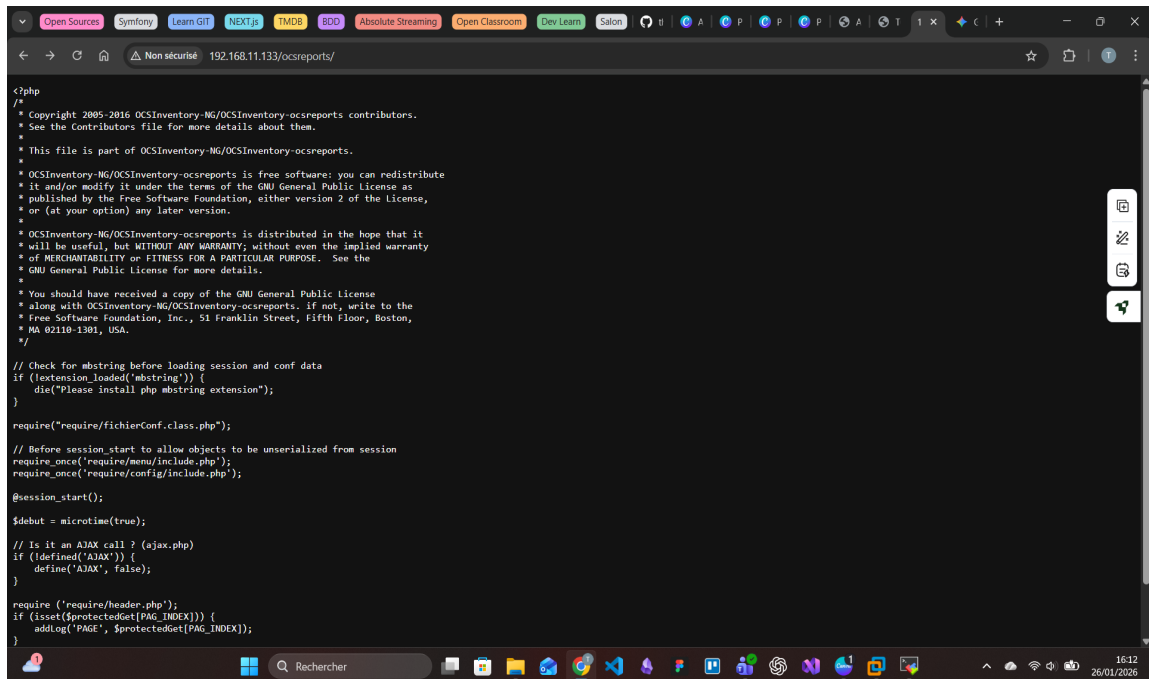


Figure 9 : Symptôme de la panne - Le code PHP n'est pas interprété par le serveur.

Diagnostic et Résolution : Le problème provenait d'un conflit entre le module MPM (Multi-Processing Module) par défaut d'Apache (`mpm_event`) et le module PHP nécessaire à OCS.

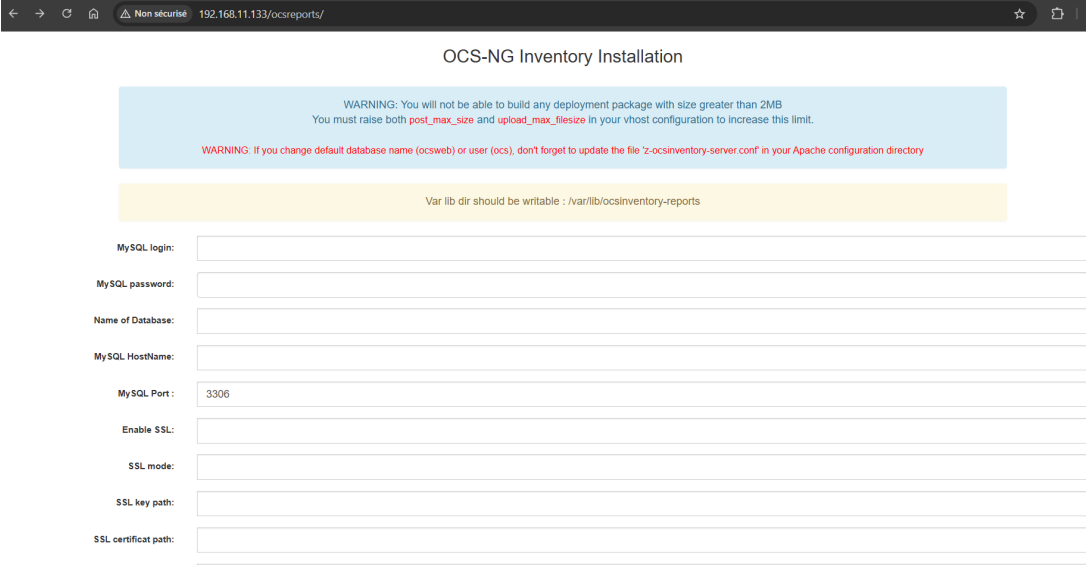
1. Désactivation du module conflictuel : `a2dismod mpm_event`
2. Activation du module compatible : `a2enmod mpm_prefork`
3. Activation du module PHP : `a2enmod php8.1`
4. Redémarrage du service Apache.

```
Considering conflict mpm_event for mpm_prefork:
ERROR: Module mpm_event is enabled - cannot proceed due to conflicts. It needs to be disabled first!
Considering conflict mpm_worker for mpm_prefork:
ERROR: could not enable dependency mpm_prefork for php8.1, aborting
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo phenmod mbstring
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo a2enmod php*
Considering dependency mpm_prefork for php8.1:
Considering conflict mpm_event for mpm_prefork:
ERROR: Module mpm_event is enabled - cannot proceed due to conflicts. It needs to be disabled first!
ERROR: could not enable dependency mpm_prefork for php8.1, aborting
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo a2enmod php8.1
Considering dependency mpm_prefork for php8.1:
Considering conflict mpm_event for mpm_prefork:
ERROR: Module mpm_event is enabled - cannot proceed due to conflicts. It needs to be disabled first!
Considering conflict mpm_worker for mpm_prefork:
ERROR: could not enable dependency mpm_prefork for php8.1, aborting
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo systemctl restart apache2
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ ^C
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo a2dismod mpm_event
Module mpm_event disabled.
To activate the new configuration, you need to run:
  systemctl restart apache2
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo a2enmod mpm_prefork
Considering conflict mpm_event for mpm_prefork:
Considering conflict mpm_worker for mpm_prefork:
Removing dangling link /etc/apache2/mods-enabled/mpm_prefork.confenabling module mpm_prefork.
To activate the new configuration, you need to run:
  systemctl restart apache2
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo a2enmod php8.1
Considering dependency mpm_prefork for php8.1:
Considering conflict mpm_event for mpm_prefork:
Considering conflict mpm_worker for mpm_prefork:
Module mpm_prefork already enabled
Considering conflict php5 for php8.1:
Enabling module php8.1.
To activate the new configuration, you need to run:
  systemctl restart apache2
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$ sudo systemctl restart apache2
thomas@apache2:~/OCSNG_UNIX_SERVER-2.10.0$
```

Figure 10 : Intervention technique sur les modules Apache pour rétablir le service PHP.

6. Finalisation de l'Installation

Une fois le serveur web opérationnel, l'accès à l'installateur Web a permis de renseigner les informations de connexion à la base de données MariaDB (Utilisateur : **ocs**, Base : **ocsweb**).



The screenshot shows a web browser window with the address bar displaying "192.168.11.133/ocsreports/". The page title is "OCS-NG Inventory Installation". The main content area contains several warning messages and a form for database configuration. The warnings are:

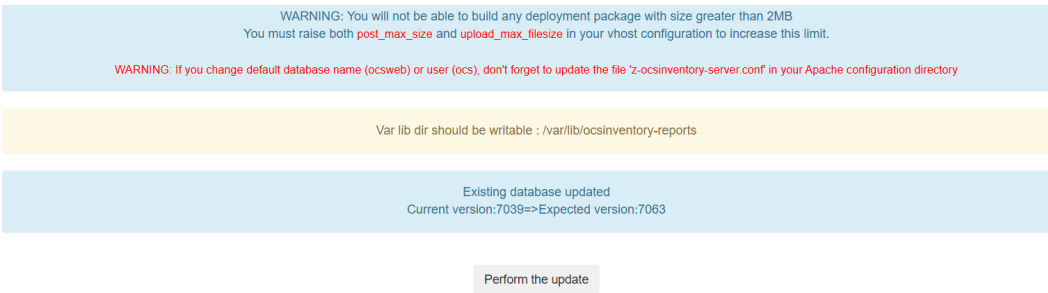
- WARNING: You will not be able to build any deployment package with size greater than 2MB. You must raise both `post_max_size` and `upload_max_filesize` in your vhost configuration to increase this limit.
- WARNING: If you change default database name (ocsweb) or user (ocs), don't forget to update the file 'z-ocsinventory-server.conf' in your Apache configuration directory.
- Var lib dir should be writable : /var/lib/ocsinventory-reports

The form fields are:

- MySQL login:
- MySQL password:
- Name of Database:
- MySQL HostName:
- MySQL Port:
- Enable SSL: ☐
- SSL mode:
- SSL key path:
- SSL certificat path:

Figure 11 : Configuration de la connexion BDD via l'interface Web.

L'installateur a ensuite peuplé la base de données.



The screenshot shows the same web browser window as Figure 11, but with the database configuration fields filled out. The main content area now displays the following messages:

- WARNING: You will not be able to build any deployment package with size greater than 2MB. You must raise both `post_max_size` and `upload_max_filesize` in your vhost configuration to increase this limit.
- WARNING: If you change default database name (ocsweb) or user (ocs), don't forget to update the file 'z-ocsinventory-server.conf' in your Apache configuration directory.
- Var lib dir should be writable : /var/lib/ocsinventory-reports
- Existing database updated
Current version:7039=>Expected version:7063

At the bottom of the page, there is a button labeled "Perform the update".

Figure 12 : Validation de la mise à jour du schéma de base de données.

Validation finale : L'accès au tableau de bord d'administration est fonctionnel après authentification (admin/admin).

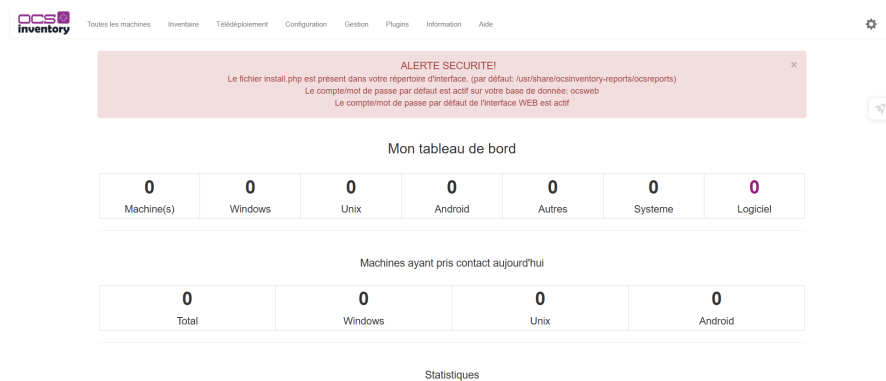


Figure 13 : Accès réussi au Dashboard OCS Inventory.