

## Introduction

### A- Mise en place d'un serveur LAMP

- 1- Mettre à jour la machine
- 2- Renommez la machine en ocs
- 3- Mise en place du serveur LAMP
- 4- Test de connexion de notre serveur LAMP
- 5- Restriction de l'accès à la base de données mariadb

### B- Installation et configuration d'ocs

- 1- Création d'une base de données mariadb pour ocs
- 2- Installer les librairies perl et les module PHP
- 3- Installation d'OCS Inventory NG Linux
- 4- Mise à jour des noms de la base de données dbocs et de l'utilisateur ocuser dans les fichiers de configuration ci-dessous :  
`z-ocsinventory-server.conf`      `zz-ocsinventory-restapi.conf`
- 5- Activations des fichiers de configurations :
  - z-ocsinventory-server.conf
  - zz-ocsinventory-restapi.conf
  - ocsinventory-reports.conf
- 6- La finition de l'installation d'ocs par l'interface web
- 7- Accès à ocs par nom de domaine
- 8- Inventaire des machines
  - a- Sur une machine linux
  - b- Sur une machine windows
  - c- Sur un système androïde
  - d- Vérification des machines remontées

## Introduction

Le parc informatique d'une organisation est un assemblage, parfois hétéroclite de matériels et de logiciels accumulés tout au long des années. On y trouve des :

- matériels différents (téléphones, portables, pc, imprimantes, éléments d'interconnexion, etc)
- logiciels et systèmes d'exploitation variés (Linux, Windows, Mac OS)
- applications utilisées dans différentes versions.
- niveaux de sécurité disparates.

De plus, la quantité de matériels et de logiciels à gérer, leur éclatement au sein de l'organisation souvent très étendue dans l'espace, les exigences de performance et de réactivité font que la gestion de parc est devenue un processus global, complet et indispensable.

La gestion du parc informatique recouvre non seulement la fonction d'inventaire de ces éléments mais aussi celles concernant le suivi et l'évolution :

- gestion de l'emplacement du matériel ;
- gestion des licences ;
- le télé-déploiement ;
- gestion financière des éléments d'inventaire ;
- gestion du cycle de vie de chaque élément ;
- gestion de la documentation informatique ;
- gestion des partenaires (fabricants, fournisseurs, transporteurs, prestataires, ...) et des contrats associés ;
- gestion statistique (nombre d'inventaire, coût des consommables,)
- prévision des besoins (matériel, logiciel, formation)

Cette gestion permet, d'une part, de répondre aux multiples questions quotidiennes posées à l'administrateur réseau (quelles sont les versions de Windows installées et sur quels postes ? y a-t-il des disques durs proches de la saturation ? Tel matériel est-il bien connecté au commutateur ? A quel endroit se trouve tel élément ? Quelle est la valeur actuelle de tel autre composant ? Quelle sont les postes encore sous garantie ?.....).

Actuellement, la tendance des DSI (Direction des Systèmes d'Information) est l'utilisation du référentiel de "bonnes pratiques" **ITIL** (Information Technology Infrastructure Library)

OCS Inventory est basé sur un concept Client/serveur. Le serveur est composé de 3 parties :

### - Serveur de communication

Collecte ; classe et archive les informations relatives aux postes clients

Ce dernier fonctionne sous **Apache** (serveur Web), **MYSQL** (SGBD : Système Gestion de Base de Données) et **PHP**. Il peut s'installer sous n'importe quel OS (Microsoft, Linux). Il utilise quelques modules PERL et des CGI (Common Gateway Interface, bibliothèques)

L'agent : Programme qui s'installe sur le client pour remonter les informations de la machine au serveur. Les agents sont disponibles pour Windows, Linux et MacOS.

### - Le serveur d'administration

L'interface web écrite en PHP qui offre des services tels que consulter des inventaires, manipuler les droits des utilisateurs etc.

### - Le serveur de déploiement

Il permet le déploiement des logiciels, des MAJ sur les postes de manière centralisé. Basé sur Apache SSL (Secure Sockets Layer : sécurisation des échanges sur internet.

### 1- Mise à jour de la machine

```
root@debian:~# apt update && apt upgrade
```

### 2- Modification du nom de la machine

```
root@debian:~# hostnamectl set-hostname ocs
```

### 3- Mise en place du serveur LAMP

L=linux

A=apache2

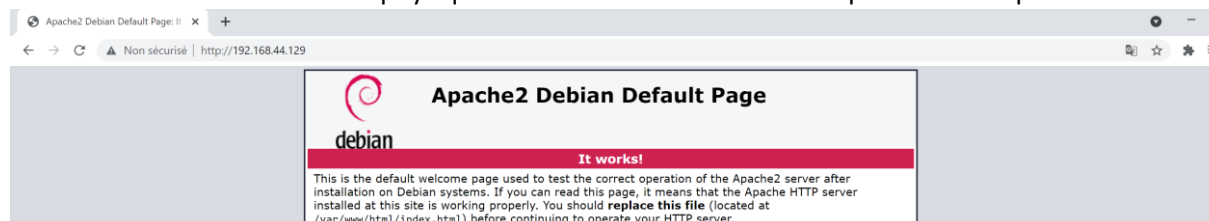
M=mariadb

P=PHP

```
root@ocs:~# apt install apache2 php mariadb-server -y
```

### 4- Test de connexion de notre serveur LAMP

Sur votre machine physique ou un machine sur vmware tapez l'adresse ip de votre machine ocs



On vérifie l'activation des module **php7.4** et de **mpm\_prefork** ainsi que les syntaxes des fichiers de configuration apache2

```
root@ocs:~# a2query -m mpm_prefork
mpm_prefork (enabled by maintainer script)

root@ocs:~# a2query -m php7.4
php7.4 (enabled by maintainer script)

root@ocs:~# apache2ctl -t
AH00557: apache2: apr_sockaddr_info_get() failed for ocs
AH00558: apache2: Could not reliably determine the server's fully qualified domain name using 127.0.0.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
```

On remarque mpm\_prefork et le module **php7.4** sont démarrés dans le cas contraire il faut les démarrer manuellement avec la commande **a2enmod** comme indiqué ci-dessous

```
root@ocs:~# a2enmod mpm_prefork
```

```
root@ocs:~# a2enmod php7.4
```

Une petite erreur de nom de domaine signalée qu'on va régler en créant un fichier fqdn.conf et en l'activant.

```
root@ocs:~# echo "ServerName www.ocs.local" > /etc/apache2/conf-available/fqdn.conf
root@ocs:~# a2enconf fqdn
Enabling conf fqdn.
To activate the new configuration, you need to run:
    systemctl reload apache2
root@ocs:~# systemctl reload apache2.service
root@ocs:~# apache2ctl -t
Syntax OK
```

## 5- Restriction de l'accès à la base de données

```
root@ocs:/usr/bin# ls mysql*
mysql      mysqlcheck      mysqldump      mysqlimport      mysqlreport      mysql_tzinfo_to_sql
mysqlaccess  mysql_convert_table_format  mysqldumpslow  mysql_install_db  mysql_secure_installation  mysql_upgrade
mysqladmin  mysql_multi      mysql_find_rows  mysql_optimize    mysql_setpermission  mysql_waitpid
mysqlanalyze  mysql_safe      mysql_fix_extensions  mysql_plugin      mysqlshow
mysqlbinlog  mysql_safe_helper  mysqlhotcopy      mysqlrepair      mysqlslap
```

On lance le script de sécurité **mysql\_secure\_installation** pour restreindre l'accès au serveur

```
root@ocs:~# mysql_secure_installation
```

On va devoir répondre à la multitude de questions qui vont s'afficher.

On définit le mot de passe root :

On tape entrée

Enter current password for root (enter for none) : entrée

On nous demande si on veut créer un mot de passe pour le compte root de la base de données. Il faut choisir N. Le compte root de MariaDB est lié à la maintenance du système, nous ne devons pas modifier les méthodes d'authentification configurées pour ce compte.

le compte root de la base de données configuré pour s'authentifier à l'aide du plugin **unix\_socket**

Switch to unix\_socket authentication [Y/n] n

Change the root password? [Y/n] Y

New password:

Re-enter new password:

Password updated successfully!

On supprime les utilisateurs anonymes, de root, etc...

Remove anonymous **users**? [Y/n] Y

les connexions distantes

Disallow root **login** remotely? [Y/n] Y

La base de test

Remove **test** database and access to it? [Y/n] Y

Recharger les tables de privilèges maintenant

Reload privilege tables now? [Y/n] Y

... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB

Installation should now be secure.

Thanks for using MariaDB!

## B- Installation et configuration d'ocs

## 1- Création d'une base de données mariadb pour ocs.

```
root@ocs:~# mysql -u root
```

Je crée une base de données qui s'appelle **dbocs**

```
MariaDB [(none)]> create database dbocs;
```

Query OK, 1 row affected (0.001 sec)

Je crée un utilisateur ocuser et je lui donne tous les privilèges sur toutes les bases dbocs

```
MariaDB [(none)]> grant all privileges on dbocs.* to userocs@'localhost' identified by 'userocs';
```

Query OK, 0 rows affected (0.002 sec)

Je recharge les droits

```
MariaDB [(none)]> flush privileges;
```

Query OK, 0 rows affected (0.001 sec)

### Vérification de mes requêtes

J'affiche ma base de données

```
MariaDB [(none)]> show databases;
```

```
MariaDB [dbocs]> show databases;
+-----+
| Database |
+-----+
| dbocs    |
| information_schema |
| mysql    |
| performance_schema |
| phpmyadmin |
+-----+
5 rows in set (0.001 sec)
```

J'affiche les utilisateurs dans mariadb

```
MariaDB [dbocs]> select user,host from mysql.user;
```

```
MariaDB [dbocs]> select user,host from mysql.user;
+-----+-----+
| User | Host |
+-----+-----+
| mariadb.sys | localhost |
| mysql | localhost |
| phpmyadmin | localhost |
| root | localhost |
| userocs | localhost |
+-----+-----+
5 rows in set (0.007 sec)
```

J'affiche les droits de l'utilisateur userocs

```
SHOW GRANTS FOR userocs@localhost;
```

```
MariaDB [(none)]> SHOW GRANTS FOR userocs@localhost;
+-----+
| Grants for userocs@localhost |
+-----+
| GRANT USAGE ON *.* TO 'userocs'@'localhost' IDENTIFIED BY PASSWORD '*300C62D5B575E2ED1CC2922CAA453A315FE05B54' |
| GRANT ALL PRIVILEGES ON 'dbocs'.* TO 'userocs'@'localhost' |
+-----+
2 rows in set (0.001 sec)
```

## 2- Installer les librairies perl et les module PHP

### - Module Perl ?

Un module est un script Perl possédant l'extension (.pm) il a été élaboré pour effectuer certaines tâches.

Un script peut contenir beaucoup de lignes de codes et donc ça devient compliqué de s'y retrouver.

C'est pour cette raison on sectionne notre programme en modules afin de le rendre plus court et donc plus facilement réutilisable.

### - Les modules CPAN

*CPAN (Comprehensive Perl Archive Network)* est le réseau complet d'archives de Perl. Il permet de collecter des milliers de modules prêts à être utilisés. Perl intègre une partie de ces modules sans lesquels il ne fonctionnera pas. La plupart des modules du CPAN sont testés régulièrement. Ils sont fiables et c'est à vous de lire attentivement la documentation et de les installer afin de les tester et en tirer tous les bénéfices.

Tous les développeurs recherchent des modules (librairies) afin de ne pas tout refaire. Pour le Perl, tout est sur le CPAN. Il existe un moteur de recherche à cet effet : <http://search.cpan.org/>.

La commande **cpan** permet d'installer des **modules de CPAN** pour le langage de programmation **Perl**.

OCS inventory a besoin de nombreuses dépendances référencées dans la documentation qui se trouve dans le lien ci-dessous

<https://wiki.ocsinventory-ng.org/01.Prerequisites/Libraries-version/>

Pour notre ocs on va installer les modules suivants

```
libapache2-mod-perl2
libapache2-mod-perl2-dev
libxml-simple-perl
libapache-dbi-perl
libnet-ip-perl
libarchive-zip-perl
libdbd-mysql-perl
libsoap-lite-perl
apache2-dev
make
php-{mysql,gd,curl,mbstring,soap,xml}
```

```
root@ocs:~# apt install libapache2-mod-perl2 libapache2-mod-perl2-dev libxml-simple-perl libapache-dbi-perl
libarchive-zip-perl libdbd-mysql-perl libnet-ip-perl libsoap-lite-perl make php-{mysql,gd,curl,mbstring,soap,xml} -y
root@ocs:~# apt install libapache2-mod-perl2 libapache2-mod-perl2-dev libxml-simple-perl libapache-dbi-perl libnet-ip-perl libsoap-lite-perl make php-{mysql,gd,curl} -y
```

Installation des modules PERL il faut respecter la casse

CPAN (Comprehensive Perl Archive Network)

Cet URL <https://metacpan.org/> permet de référencier tous les modules existants

```
perl -MCPAN -e "install XML::Entities"
```

A la question ci-dessous on tape y ou entrée

```
Would you like to configure as much as possible automatically? [yes]
```

```
cpan SOAP::Lite
```

```
cpan Mojolicious::Lite
```

```
cpan Switch
```

```
cpan Plack::Handler
```

```
cpan Apache2::SOAP
```

### 3- Installation d'OCS Inventory NG Linux

Récupération du lien de téléchargement de la dernière version d'OCS sur le site

[https://ocsinventory-ng.org/?page\\_id=1235&lang=fr](https://ocsinventory-ng.org/?page_id=1235&lang=fr)



## Téléchargements OCS Inventory

Nous vous remercions de l'intérêt que vous portez au projet OCS Inventory.

Vous trouverez ci-dessous les différents produits téléchargeables.

Serveur :

- [Serveur pour Linux/Unix 2.9.1](#)
- [Serveur pour Linux/Unix 2.9.1 \(nightly\)](#)



**Agents :**

- [Agent pour Windows 2.9.0.0 \(64 bits\)](#)
- [Agent pour Windows 2.9.0.0 \(32 bits\)](#)
- [Agent pour Windows 2.1.1.1 \(XP & 2003R2 seulement\)](#)
- [Agent pour Unix/Linux 2.8.1](#)
- [Dépôt Unix/Linux Redhat](#)
- [Agent pour MacOS X 2.6.0](#)
- [Agent pour Android 2.7](#)

**Outils :**

- [Windows Packager 2.8 \(64 bits uniquement\)](#)
- [Windows Packager 2.3 \(32 bits\)](#)
- [Unix Packager 1.0](#)
- [Agent Deployment Tool 2.3](#)

**Plugins :**

- [Plugins disponibles](#)

**Rejoignez la communauté Open Source !**

- Lisez [la documentation](#) d'OCS Inventory
- Posez vos questions sur notre [Forum](#)
- Suivez le développement sur [Github](#)
- Discutez sur [IRC](#) avec les développeurs et des utilisateurs

Sur le terminal avec la commande wget et le lien de téléchargement on télécharge ocs :

```
#wget https://github.com/OCSInventory-NG/OCSInventory-ocsreports/releases/download/2.9.1/OCSNG_UNIX_SERVER-2.9.1.tar.gz
```

```
root@ocs:~# mkdir tmp
root@ocs:~# cd tmp
root@ocs:~/tmp# wget https://github.com/OCSInventory-NG/OCSInventory-ocsreports/releases/download/2.9.1/OCSNG_UNIX_SERVER-2.9.1.tar.gz
OCSNG_UNIX_SERVER-2.9.1.tar.gz 100%[=====] 5,42M 7,04MB/s ds 0,8s
2021-11-13 13:17:17 (7,04 MB/s) - « OCSNG_UNIX_SERVER-2.9.1.tar.gz » sauvegardé [5683654/5683654]
```

Vérifier la présence du fichier téléchargé (.tar.gz)

```
root@ocs:~/tmp# ls
OCSNG_UNIX_SERVER-2.9.1.tar.gz
```

On décompresse le fichier téléchargé avec la commande tar, en suite on lance l'exécutable setup.sh pour installer ocs :

x : permet d'extraire certains fichiers d'une archive

z : décompacte l'archive avec l'utilitaire gzip

f : extrait un fichier donné (ici le fichier est **OCSNG\_UNIX\_SERVER-2.9.1.tar.gz**)

```
root@ocs:~/tmp# tar xzf OCSNG_UNIX_SERVER-2.9.1.tar.gz
root@ocs:~/tmp# ls
OCSNG_UNIX_SERVER-2.9.1  OCSNG_UNIX_SERVER-2.9.1.tar.gz
root@ocs:~/tmp# cd OCSNG_UNIX_SERVER-2.9.1/
root@ocs:~/tmp/OCSNG_UNIX_SERVER-2.9.1# ls
Apache  App  binutils  cpanfile  Std  Sys  INSTALL  LICENSE  ocsreports  README.md  setup.sh
```

Lancer le script setup.sh à l'aide de la commande : `./setup.sh` et on répond à la série de questions :

```
root@ocs:~/tmp/OCSNG_UNIX_SERVER-2.9.1# ./setup.sh
```

Il faut toujours cliquer sur entrée



Ce qui est indiqué entre crochet est choisit par défaut si on fait entrée



Do you wish to continue ([y]/n)?



Which host is running database server [localhost] ?



On which port is running database server [3306] ?



Where is Apache daemon binary [/usr/sbin/apache2ctl] ?



Where is Apache main configuration file [/etc/apache2/apache2.conf] ?



Which user account is running Apache web server [www-data] ?



Which user group is running Apache web server [www-data] ?



Where is Apache Include configuration directory [/etc/apache2/conf-available] ?



Where is PERL interpreter binary [/usr/bin/perl] ?



Do you wish to setup Communication server on this computer ([y]/n)?



Where to put Communication server log directory [/var/log/ocsinventory-server] ?



Where to put Communication server plugins configuration files [/etc/ocsinventory-server/plugins] ?



Where to put Communication server plugins Perl modules files [/etc/ocsinventory-server/perl] ?



```
+-----+
| Checking for required Perl Modules... |
+-----+
```

```
Checking for DBI PERL module...
Found that PERL module DBI is available.
Checking for Apache::DBI PERL module...
Found that PERL module Apache::DBI is available.
Checking for DBD::mysql PERL module...
Found that PERL module DBD::mysql is available.
Checking for Compress::Zlib PERL module...
Found that PERL module Compress::Zlib is available.
Checking for XML::Simple PERL module...
Found that PERL module XML::Simple is available.
Checking for Net::IP PERL module...
Found that PERL module Net::IP is available.
Checking for Archive::Zip Perl module...
Found that PERL module Archive::Zip is available.
```

Do you wish to setup Rest API server on this computer ([y]/n)?



Do you allow Setup renaming Communication Server Apache configuration file

to 'z-ocsinventory-server.conf' ([y]/n) ?



Do you wish to setup Administration Server (Web Administration Console)

on this computer ([y]/n)?

Administration Server writable/cache directory (by default /var/lib/ocsinventory-reports), especially if you use deployment feature.

Do you wish to continue ([y]/n)?

Where to copy Administration Server static files for PHP Web Console

[/usr/share/ocsinventory-reports] ?

Where to create writable/cache directories for deployment packages, administration console logs, IPDiscover and SNMP [/var/lib/ocsinventory-reports] ?

A la fin de l'installation vous devez avoir le résultat suivant, vous indiquant comment se connecter sur votre serveur OCS

```

OK, Administration server installation finished ;-)
Please, review /etc/apache2/conf-enabled/ocsinventory-reports.conf
to ensure all is good and restart Apache daemon.
Then, point your browser to http://server//ocsreports
to configure database server and create/update schema.

Setup has created a log file /root/tmp/OCSNG_UNIX_SERVER-2.9.1/ocs_server_setup.log. Please, save this file.
If you encounter error while running OCS Inventory NG Management server,
we can ask you to show us its content !

DON'T FORGET TO RESTART APACHE DAEMON !
Enjoy OCS Inventory NG ;-)
```

#### 4- Mise à jour des noms de la base de données dbocs et de l'utilisateur ocsuser dans les fichiers de configuration **z-ocsinventory-server.conf** et **zz-ocsinventory-restapi.conf**

- On renseigne dans le fichier **zz-ocsinventory-restapi.conf** Le nom de notre base et de l'utilisateur créé

```
root@debian:~# vim /etc/apache2/conf-available/zz-ocsinventory-restapi.conf
```

```
$ENV{OCS_DB_LOCAL} = 'dbocs';
$ENV{OCS_DB_USER} = 'userocs';
$ENV{OCS_DB_PWD} = 'userocs';
```

- On fait de même dans le fichier **z-ocsinventory-server.conf** Le nom de notre base et de l'utilisateur créé

```
root@debian:~# vim /etc/apache2/conf-available/z-ocsinventory-server.conf
```

```
PerlSetEnv OCS_DB_NAME dbocs
PerlSetEnv OCS_DB_LOCAL dbocs
# User allowed to connect to database
PerlSetEnv OCS_DB_USER userocs
# Password for user
PerlSetVar OCS_DB_PWD userocs
```

- Même chose dans le fichier dbconfig.inc.php

```
?php
$_SESSION["SERVEUR_SQL"]="localhost";
$_SESSION["COMPTE_BASE"]="userocs";
$_SESSION["PSWD_BASE"]="userocs";
```

#### 5- Activations des fichiers de configurations :

z-ocsinventory-server.conf  
zz-ocsinventory-restapi.conf  
ocsinventory-reports.conf

- On démarre avec la commande a2enmod les confs suivantes :

```
root@debian:~# a2enconf z-ocsinventory-server
Enabling conf z-ocsinventory-server.
To activate the new configuration, you need to run:
systemctl reload apache2

root@debian:~# a2enconf zz-ocsinventory-restapi
Enabling conf zz-ocsinventory-restapi.
To activate the new configuration, you need to run:
systemctl reload apache2
```

- Je déplace le fichier **ocsinventory-reports.conf** vers le répertoire **sites-available**

```
root@ocs:~# mv /etc/apache2/conf-available/ocsinventory-reports.conf /etc/apache2/sites-available/
```

J'active après cette configuration dans sites-enabled avec la commande **a2ensites**

```
root@ocs:~# a2ensite ocsinventory-reports.conf
Enabling site ocsinventory-reports.
To activate the new configuration, you need to run:
systemctl reload apache2
```

On redémarrer le service apache2

```
root@ocs:~# systemctl restart apache2
```

#### 6- La finition de l'installation d'ocs par l'interface web

Depuis la machine physique, taper l'url de la machine adresse <http://adresse IP/ocsreports/>

OCS-NG Inventory Installation

WARNING: You will not be able to build any deployment package with size greater than 100MB  
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

MySQL login:

MySQL password:

Name of Database:

MySQL HostName:

MySQL Port:

Enable SSL:

MySQL login: **userocs**  
MySQL password: **userocs**  
Name of Database: **dbocs**  
MySQL HostName: **localhost**

La base de données est validée on continue notre installation

OCS-NG Inventory Installation

WARNING: You will not be able to build any deployment package with size greater than 100MB  
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

Your database is OK.  
No action taken.

[Click here to enter OCS-NG GUI](#)

On nous propose une mise à jour donc on l'effectue

AVERTISSEMENT: Vous ne serez pas en mesure de construire un paquet de déploiement d'une taille plus grande que 100Mo  
Vous devez modifier **post\_max\_size** et **upload\_max\_filesize** dans la configuration du vhost, pour augmenter cette limite.

ATTENTION: Si vous changez le nom de la base (ocsweb), pensez à modifier vos fichiers de conf moteur (file z-ocsinventory-server.conf)

Var lib dir should be writable : /var/lib/ocsinventory-reports

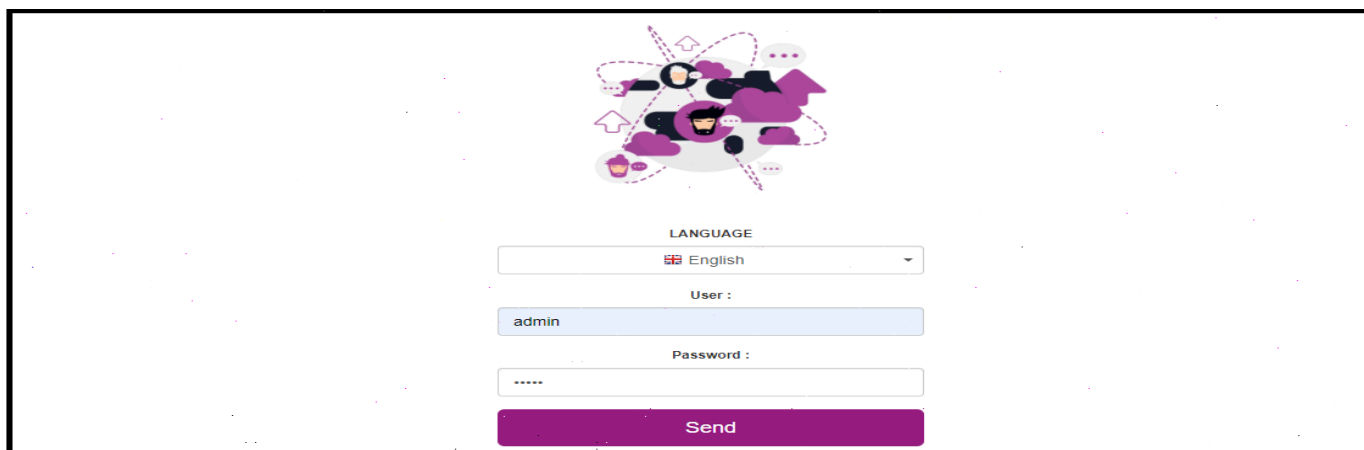
mise à jour de la base de données existante  
Version actuelle:7039=>Version attendue:7051

[Effectuer la mise à jour](#)

On se connecte avec le comptes et mot de passe par défaut : admin



admin



LANGUAGE  
English

User :  
admin

Password :  
\*\*\*\*\*

Send

**SECURITY ALERT!**

Your `install.php` exists in your installation directory (by default : `/usr/share/ocsinventory-reports/ocsreports`).

Pour cette alerte de sécurité, renommez le fichier `install.php` en `.install.php`

```
root@ocs: /usr/share/ocsinventory-reports/ocsreports# mv install.php .install.php
```

Vérifier que vous n'avez plus d'alertes de sécurité sur l'interface Web.

**7- Accès à ocs par nom de domaine**

On va tenter d'accéder à notre serveur ocs par un nom de domaine : **ocs.sitka.local**

a- Pour cela il faut créer un enregistrement ocs sur notre serveur DNS

 ocs	Hôte (A)	192.168.65.132
---	----------	----------------

b- Modifier le fichier `ocsinventory-reports.conf`  
Dans ce fichier on rajoute un bloc Virtual host

```
<virtualhost *:80>  
ServerName ocs.sitka.local  
DocumentRoot /usr/share/ocsinventory-reports/ocsreports/  
</virtualhost>
```



## 8- Testez l'inventaire de la machine ocs et d'une machine windows

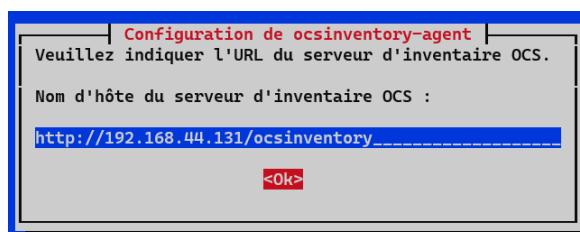
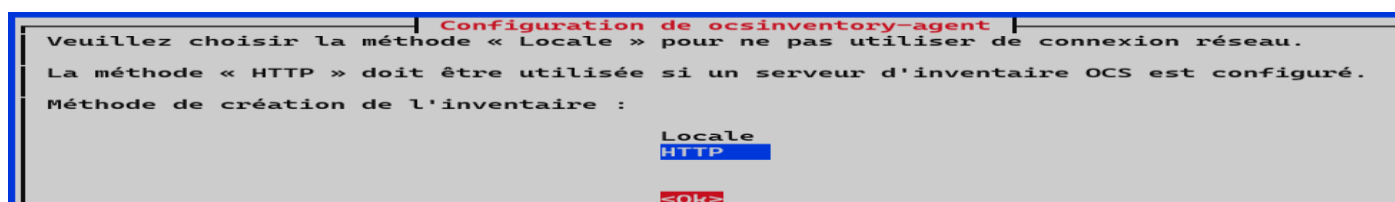
- a- Sur une machine linux
- c- Sur un système androïde

Sur une machine linux, installez l'agent ocsinventory-agent :

```
root@ocs:~# apt install ocsinventory-agent
```

Au moment de l'installation, vous devez choisir la méthode http et l'URL : **http://ip de votre serveur ocs/ocsinventory**  
 Depuis le terminal, lancez la commande ocsinventory-agent, cette dernière doit vous faire apparaître dans l'interface Web d'OCS la machine ocs-gipi

- 1- Sur la machine ocs-gipi, installez l'agent ocsinventory-agent :



Depuis le terminal, lancez la commande **ocsinventory-agent**, cette dernière doit vous faire apparaître dans l'interface Web d'OCS la machine.

```
root@ocs:~# ocsinventory-agent
```

```
root@debian:/etc# ocsinventory-agent
[info] [download] Download is off.
[info] [download] Beginning work. I am 25103.
[info] [download] Option turned off. Exiting.
```

En cas de problème de configuration de l'agent à cause d'une erreur on peut reconfigurer l'agent avec la commande **dpkg-reconfigure ocsinventory-agent**

```
root@ocs:~# dpkg-reconfigure ocsinventory-agent
```

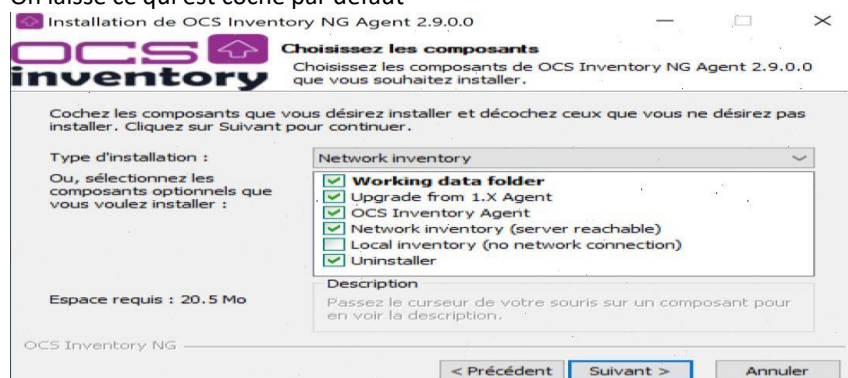
b- Sur une machine Windows

Sur une machine Windows on installe l'agent ocs à partir du lien ci-dessous on télécharge l'agent pour windows

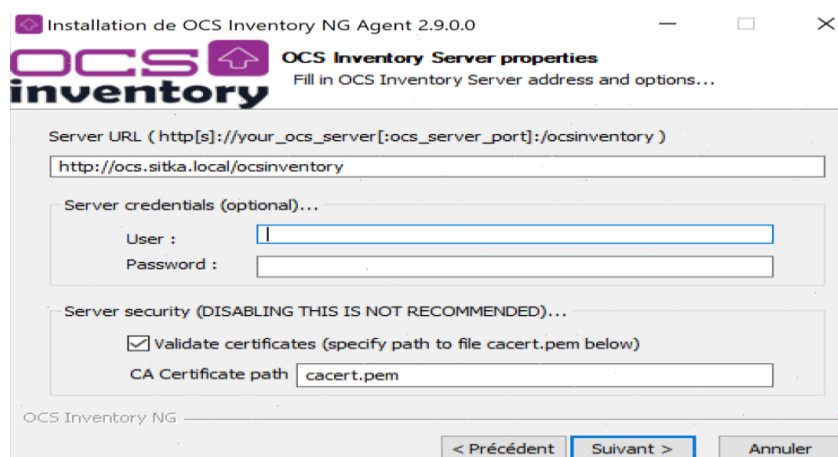
[https://github.com/OCSInventory-NG/WindowsAgent/releases/download/2.9.0.0/OCS-Windows-Agent-2.9.0.0\\_x64.zip](https://github.com/OCSInventory-NG/WindowsAgent/releases/download/2.9.0.0/OCS-Windows-Agent-2.9.0.0_x64.zip)

On décompresse le fichier  OCS-Windows-Agent-2.9.0.0\_x64 et on lance l'exécutable  OCS-Windows-Agent-Setup-x64

On laisse ce qui est coché par défaut



On renseigne l'adresse de notre serveur ocs



On laisse vide car on n'a pas de proxy



Installation de OCS Inventory NG Agent 2.9.0.0

**OCS inventory** **Proxy Server properties**  
If needed, specify proxy server to use...

Proxy type : None

Address :

Port :

Proxy credentials (optional)...

User :

Password :

OCS Inventory NG

< Précédent Suivant > Annuler

Installation de OCS Inventory NG Agent 2.9.0.0

**OCS inventory** **OCS Inventory Agent for Windows properties**  
If needed, specify OCS Inventory Agent options...

General options...

☒ Enable verbose log

☐ Do not scan for installed Software

☐ Never ask for TAG

Specify TAG value :

Setup options...

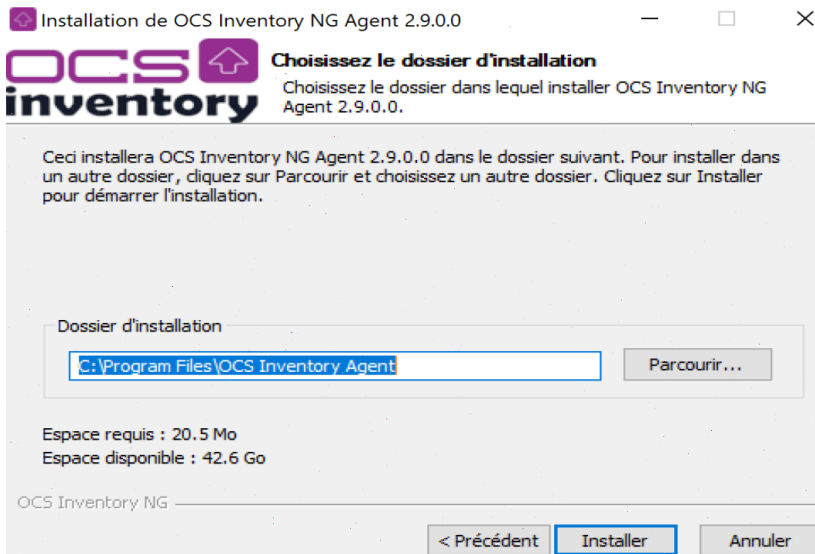
☐ Do not register service - agent must be launched manually (= /NO\_SERVICE)

☐ Do not register Systray applet to automatically start (= /NO\_SYSTRAY)

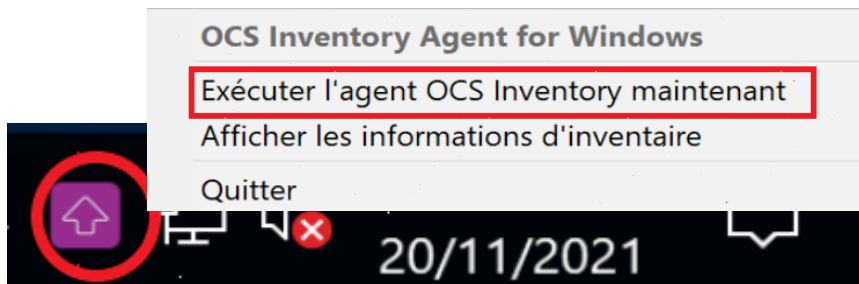
☒ Immediately launch inventory (= /NOW)

OCS Inventory NG

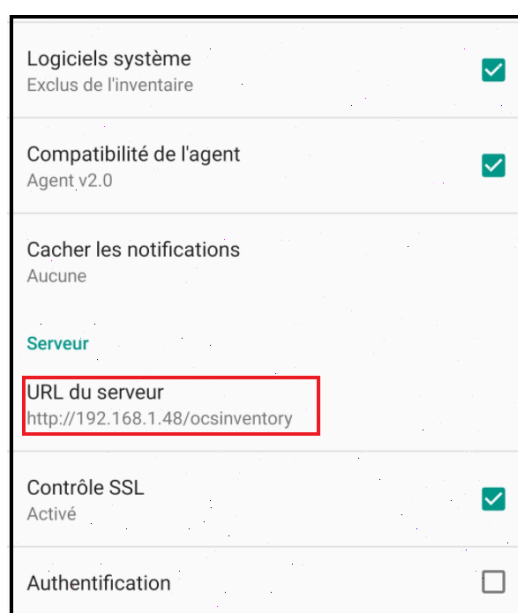
< Précédent Suivant > Annuler



Clique droit sur l'icône ocs puis on choisit Exécuter l'agent OCS Inventory maintenant



- c- Sur un système androïde
  - Mettez une deuxième carte en bridge pour pouvoir accéder au serveur ocs à partir du smartphone
  - Sur un smartphone ou tablette téléchargez l'agent ocs pour androïde [Agent pour Android 2.7](#)
  - Entrez dans l'agent ocs l'adresse de ocs puis envoyez l'inventaire



#### d- Vérification des machines remontées

En accédant à l'interface web d'ocs on vérifie la monté des machines

