
OCS Inventory

Índex

1. Què és OCS Inventory	1
2. Requisits previs	2
3. Instal·lació de les dependències (LAMP + Perl)	3
4. Creació de la base de dades	4
5. Descàrrega i instal·lació del servidor OCS Inventory	4
6. Configuració d'Apache	14
6.1. Modifica la configuració de PHP	14
6.2. Propietaris i permisos	14
6.3. Certificat SSL autosignat per a ocs.thos.local	15
6.4. Crea la configuració HTTPS	15
6.5. Crea la configuració HTTP (redirecció)	16
6.6. Fitxers de VirtualHosts	16
6.7. Sintaxi, reinici i verificació	17
7. Assistent web d'instal·lació	18
8. Configuració del tallafocs	21
9. Instal·lació de l'agent a clients GNU/Linux (Debian/Ubuntu)	21
9.1. Dependències del client	22
9.2. Descarrega i compila l'agent	22
9.3. Primera execució manual de l'agent	24
9.4. Automatització amb cron	25
10. Instal·lació de l'agent a clients Windows	25
10.1. Instal·lació manual (interactiva)	26
10.2. Desplegament silencios (GPO / SCCM / script)	26
11. Comprovació i ús bàsic de la consola	27
12. Recomanacions per a producció	27
13. Documentació	27

Aquest document té com a finalitat descriure la instal·lació d'OCS Inventory Server sobre una pila LAMP (Apache + MariaDB + PHP) amb mòduls Perl addicionals, així com la instal·lació dels agents als clients GNU/Linux i Windows.

1. Què és OCS Inventory



Figura 1: OCS Inventory Logo

OCS Inventory NG (*Open Computers and Software Inventory Next Generation*) és una eina de codi obert per a la gestió d'actius informàtics que existeix des del 2001, centrada en dues grans funcionalitats:

Gestió d'actius

- Els agents instal·lats als equips recullen informació sobre la composició de maquinari i programari de cada ordinador o servidor.
- Per als dispositius que no poden tenir agent instal·lat (impressores, escàners, encaminadors, etc.), s'utilitza l'escaneig IP Discover.
- Des de la versió 2.0, s'ha afegit l'escaneig SNMP, que complementa la informació de la IP Discover i permet obtenir dades més detallades d'aquests dispositius de xarxa sense agent.

Desplegament

- OCS Inventory NG permet desplegar paquets de programari des del servidor central cap als clients, garantint que tots els equips de la xarxa tinguin els mateixos entorns de programari.
- Els paquets es descarreguen via HTTP/HTTPS i l'agent s'encarrega de llançar-los a l'ordinador client.
- El sistema està dissenyat per minimitzar l'impacte del desplegament sobre la xarxa, i s'ha utilitzat en infraestructures de més de 100.000 equips.

En resum: OCS combina **inventari automàtic** (via agents i escaneig de xarxa) amb **desplegament centralitzat de programari**, fent-lo escalable per a entorns grans.

L'arquitectura d'OCS Inventory es divideix en quatre components principals:

Component	Funció
Servidor de base de dades	Emmagatzema tota la informació recollida (MariaDB/MySQL)
Servidor de comunicació	Mòdul Perl sobre Apache que rep les dades enviades pels agents
Consola d'administració	Aplicació PHP (ocsreports) que mostra l'inventari via web
Servei de desplegament	Permet desplegar programari i paquets als clients de forma remota

Suporta agents per a Windows, GNU/Linux, BSD, macOS, Solaris, AIX, HP-UX i Android.

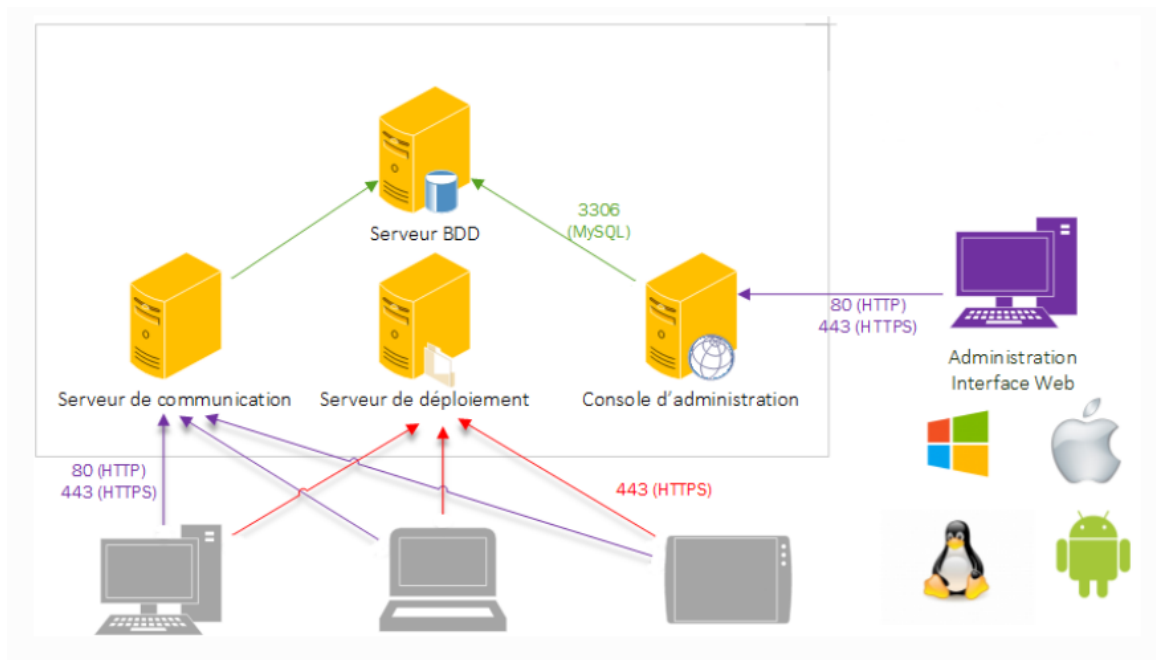


Figura 2: Arquitectura d'OCS Inventory

Font: <https://wiki.ocsinventory-ng.org/02.Newbie-documentation/OCS-Inventor-y-NG-Basics/>

2. Requisites previs

- Servidor amb **Ubuntu Server 26.04 LTS**, amb almenys 2 GB de RAM i 2 CPU.
- Accés root o usuari amb privilegis sudo.
- Adreça IP estàtica o nom DNS resoluble per al servidor.
- Connexió a Internet per descarregar paquets.
- Ports **80/TCP** (HTTP) i **443/TCP** (HTTPS) oberts al tallafocs.

CONSELL

En un entorn d'aula, és recomanable assignar una IP fixa al servidor (per exemple 192.168.1.10) perquè els agents dels clients sempre puguin trobar-lo.

3. Instal·lació de les dependències (LAMP + Perl)

Actualitza la llista de paquets

```
sudo apt update
```

Actualitza el sistema

```
sudo apt upgrade
```

Instal·la Apache, MariaDB, PHP i les extensions necessàries:

```
sudo apt install apache2 mariadb-server php php-mysql php-gd  
↔ php-mbstring \  
    php-xml php-curl php-soap php-zip php-pclzip libapache2-mod-php \  
    libapache2-mod-perl2 libapache2-mod-perl2-dev
```

Instal·la els mòduls Perl requerits pel servidor de comunicació:

```
sudo apt install perl libxml-simple-perl libdbi-perl  
↔ libdbd-mysql-perl \  
    libapache-dbi-perl libnet-ip-perl libsoap-lite-perl  
↔ libarchive-zip-perl \  
    libcompress-zlib-perl libio-compress-perl make build-essential
```

Alguns mòduls Perl no estan disponibles als dipòsits d'Ubuntu i cal instal·lar-los via CPAN:

```
sudo cpan install XML::Entities Apache2::SOAP Mojolicious::Lite  
↔ Switch Plack::Handler
```

AVÍS

La primera vegada que s'executa cpan, pot demanar configuració inicial. Es pot acceptar la configuració automàtica prement Enter a totes les preguntes.

Comprova que Apache i MariaDB s'han iniciat correctament (a Ubuntu arrenquen automàticament després de la instal·lació):

```
systemctl is-active apache2 mariadb
```

Ambdós haurien de mostrar active.

4. Creació de la base de dades

Primer, executa l'script de seguretat de MariaDB per establir una contrasenya de root i eliminar configuracions insegures:

```
sudo mariadb-secure-installation
```

Respon afirmativament a totes les preguntes (establir contrasenya root, eliminar usuaris anònims, desactivar accés remot de root, eliminar la base de dades de test i recarregar privilegis).

Ara accedeix a la consola de MariaDB:

```
sudo mariadb -u root -p
```

I crea la base de dades i l'usuari dedicat per a OCS:

```
CREATE DATABASE ocsweb;  
GRANT ALL PRIVILEGES ON ocsweb.* TO ocsuser@localhost IDENTIFIED BY  
↳ 'ContrasenyaSegura123!';  
FLUSH PRIVILEGES;  
EXIT;
```

IMPORTANT

Substitueix `ContrasenyaSegura123!` per una contrasenya forta i guarda-la, ja que la necessitaràs al punt 6.6. El nom `ocsweb` és el que espera OCS Inventory per defecte.

5. Descàrrega i instal·lació del servidor OCS Inventory

Mou-te al directori temporal

```
cd /tmp
```

Descarrega l'última versió estable des del dipòsit oficial de GitHub (en el moment d'escriure aquest document, la 2.12.4):

```
wget -c https://github.com/OCSInventory-NG/OCSInventory-ocsreports/releases/download/2.12.4/OCSNG_UNIX_SERVER-2.12.4.tar.gz
```

NOTA

Comprova si hi ha una versió més recent a la [pàgina de releases del projecte](#) abans de descarregar.

Descomprimeix

```
tar -xzf OCSNG_UNIX_SERVER-2.12.4.tar.gz
```

Accedeix a la carpeta creada

```
cd OCSNG_UNIX_SERVER-2.12.4
```

Edita l'script per corregir alguns errors que hi conté.

```
sudo nano -l setup.sh
```

- A la línia 154 canvia -f6 per -f5
- A la línia 155 canvia -f6 per -f5
- A la línia 158 canvia -f4 per -f5
- A la línia 159 canvia -f4 per -f5

Les línies han de quedar així:

```
DB_CLIENT_MAJOR_VERSION=$(eval mysql -V | cut -d' ' -f5 | cut -d'.'
↪ -f1) >>$SETUP_LOG 2>&1
DB_CLIENT_MINOR_VERSION=$(eval mysql -V | cut -d' ' -f5 | cut -d'.'
↪ -f2) >>$SETUP_LOG 2>&1

if [ "$DB_CLIENT_MAJOR_VERSION" = "Linux" ]; then
    DB_CLIENT_MAJOR_VERSION=$(eval mysql -V | cut -d' ' -f5 | cut
↪ -d'.' -f1) >>$SETUP_LOG 2>&1
    DB_CLIENT_MINOR_VERSION=$(eval mysql -V | cut -d' ' -f5 | cut
↪ -d'.' -f2) >>$SETUP_LOG 2>&1
```

Executa l'script d'instal·lació, que és interactiu:

```
sudo bash ./setup.sh
```

```
+-----+
|       Welcome to OCS Inventory NG Management server setup !       |
+-----+

Trying to determine which OS or Linux distribution you use
+-----+
| Checking for Apache web server binaries !                          |
+-----+

CAUTION: If upgrading Communication server from OCS Inventory NG 1.0
↪ RC2 and
previous, please remove any Apache configuration for Communication
↪ Server!

Do you wish to continue ([y]/n)?
Assuming Communication server 1.0 RC2 or previous is not installed
```

on this computer.

Starting OCS Inventory NG Management server setup from folder

↪ /tmp/OCSNG_UNIX_SERVER-2.12.4

Storing log in file /tmp/OCSNG_UNIX_SERVER-2.12.4/ocs_server_setup.log

```
+-----+
| Checking for database server properties...      |
+-----+
```

Your MySQL client seems to be part of MySQL version 15.2.

Your computer seems to be running MySQL 4.1 or higher, good ;-)

Which host is running database server [localhost] ?

OK, database server is running on host localhost ;-)

On which port is running database server [3306] ?

OK, database server is running on port 3306 ;-)

```
+-----+
| Checking for Apache web server daemon...        |
+-----+
```

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

OK, using Apache daemon /usr/sbin/apache2ctl ;-)

```
+-----+
| Checking for Apache main configuration file...  |
+-----+
```

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

OK, using Apache main configuration file /etc/apache2/apache2.conf ;-)

```
+-----+
| Checking for Apache user account...            |
+-----+
```

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

OK, Apache is running under user account www-data ;-)

```
+-----+
| Checking for Apache group...                   |
+-----+
```

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

OK, Apache is running under users group www-data ;-)

```
+-----+
```

```

| Checking for Apache Include configuration directory... |
+-----+

Setup found Apache Include configuration directory in
/etc/apache2/conf-available.
Setup will put OCS Inventory NG Apache configuration in this
↪ directory.
Where is Apache Include configuration directory
↪ [/etc/apache2/conf-available] ?
OK, Apache Include configuration directory
↪ /etc/apache2/conf-available found ;-)

+-----+
| Checking for PERL Interpreter... |
+-----+

Found PERL interpreter at </usr/bin/perl> ;-)
Where is PERL interpreter binary [/usr/bin/perl] ?
OK, using PERL interpreter /usr/bin/perl ;-)

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

+-----+
|           Checking for Make utility...           |
+-----+

OK, Make utility found at </usr/bin/make> ;-)

+-----+
|           Checking for Apache mod_perl version... |
+-----+

Checking for Apache mod_perl version 1.99_22 or higher
Found that mod_perl version 1.99_22 or higher is available.
OK, Apache is using mod_perl version 1.99_22 or higher ;-)

+-----+
|   Checking for Communication server log directory...   |
+-----+

Communication server can create detailed logs. This logs can be
↪ enabled
by setting integer value of LOGLEVEL to 1 in Administration console
menu Configuration.
Where to put Communication server log directory
↪ [/var/log/ocsinventory-server] ?
OK, Communication server will put logs into directory
↪ /var/log/ocsinventory-server ;-)

```

```

+-----+
↪ -----+
|   Checking for Communication server plugins configuration
↪ directory...   |
+-----+
↪ -----+

Communication server need a directory for plugins configuration files.
Where to put Communication server plugins configuration files
↪ [/etc/ocsinventory-server/plugins] ?
OK, Communication server will put plugins configuration files into
↪ directory /etc/ocsinventory-server/plugins ;- )

+-----+
|   Checking for Communication server plugins perl directory...   |
+-----+

Communication server need a directory for plugins Perl modules files.
Where to put Communication server plugins Perl modules files
↪ [/etc/ocsinventory-server/perl] ?
OK, Communication server will put plugins Perl modules files into
↪ directory /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)?

+-----+
| Checking for REST API Dependencies ...                               |
+-----+

Found that PERL module Mojolicious::Lite is available.
Found that PERL module Switch is available.
Found that PERL module Plack::Handler is available.

```

```

+-----+
| Configuring REST API Server files ... |
+-----+

Where do you want the API code to be store
↪ [/usr/local/share/perl/5.40.1] ?
Copying files to /usr/local/share/perl/5.40.1

+-----+
| Configuring REST API Server configuration files ... |
+-----+

+-----+
| OK, looks good ;- ) |
| Configuring Communication server Perl modules... |
+-----+

Checking if your kit is complete...
Looks good
Generating a Unix-style Makefile
Writing Makefile for Apache::Ocsinventory
Writing MYMETA.yml and MYMETA.json

+-----+
| OK, looks good ;- ) |
| Preparing Communication server Perl modules... |
+-----+

+-----+
| OK, prepare finshed ;- ) |
| Installing Communication server Perl modules... |
+-----+

+-----+
| OK, Communication server Perl modules install finished;- ) |
| Creating Communication server log directory... |
+-----+

Creating Communication server log directory
↪ /var/log/ocsinventory-server.

Fixing Communication server log directory files permissions.
Configuring logrotate for Communication server.
Removing old communication server logrotate file
↪ /etc/logrotate.d/ocsinventory-NG

```

```

Writing communication server logrotate to file
↳ /etc/logrotate.d/ocsinventory-server

+-----+
↳ --+
|      OK, Communication server log directory created ;-)|
|      Creating Communication server plugins configuration directory...|
↳ |
+-----+
↳ --+

Creating Communication server plugins configuration directory
↳ /etc/ocsinventory-server/plugins.

+-----+
↳ --+
| OK, Communication server plugins configuration directory created ;-)|
↳ |
|      Creating Communication server plugins Perl directory...|
↳ |
+-----+
↳ --+

Creating Communication server plugins Perl directory
↳ /etc/ocsinventory-server/perl.

+-----+
↳ --+
|      OK, Communication server plugins Perl directory created ;-)|
↳ |
|      Now configuring Apache web server...|
+-----+
↳ --+

To ensure Apache loads mod_perl before OCS Inventory NG Communication
↳ Server,
Setup can name Communication Server Apache configuration file
'z-ocsinventory-server.conf' instead of 'ocsinventory-server.conf'.
Do you allow Setup renaming Communication Server Apache configuration
↳ file
to 'z-ocsinventory-server.conf' ([y]/n) ?
OK, using 'z-ocsinventory-server.conf' as Communication Server Apache
↳ configuration file
Removing old communication server configuration to file
↳ /etc/apache2/conf-available/ocsinventory.conf
Writing communication server configuration to file
↳ /etc/apache2/conf-available/z-ocsinventory-server.conf

```

```

+-----+
↪ --+
|      OK, Communication server setup successfully finished ;-)      |
|
| Please, review
↪ /etc/apache2/conf-available/z-ocsinventory-server.conf |
|      to ensure all is good. Then restart Apache daemon.      |
+-----+
↪ --+

Do you wish to setup Administration Server (Web Administration
↪ Console)
on this computer ([y]/n)?

+-----+
|      Checking for Administration Server directories...      |
+-----+

CAUTION: Setup now install files in accordance with Filesystem
↪ Hierarchy
Standard. So, no file is installed under Apache root document
↪ directory
(Refer to Apache configuration files to locate it).
If you're upgrading from OCS Inventory NG Server 1.01 and previous,
↪ YOU
MUST REMOVE (or move) directories 'ocsreports' and 'download' from
↪ Apache
root document directory.
If you choose to move directory, YOU MUST MOVE 'download' directory to
Administration Server writable/cache directory (by default
/var/lib/ocsinventory-reports), especially if you use deployment
↪ feature.

Do you wish to continue ([y]/n)?
Assuming directories 'ocsreports' and 'download' removed from
Apache root document directory.

Where to copy Administration Server static files for PHP Web Console
[/usr/share/ocsinventory-reports] ?
OK, using directory /usr/share/ocsinventory-reports to install static
↪ files ;-)

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

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

```

```
Checking for DBI PERL module...
Found that PERL module DBI is available.
Checking for DBD::mysql PERL module...
Found that PERL module DBD::mysql 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.
```

```
+-----+
|           Installing files for Administration server...           |
+-----+
```

```
Creating PHP directory /usr/share/ocsinventory-reports/ocsreports.
Copying PHP files to /usr/share/ocsinventory-reports/ocsreports.
Fixing permissions on directory
↳ /usr/share/ocsinventory-reports/ocsreports.
Creating database configuration file
↳ /usr/share/ocsinventory-reports/ocsreports/dbconfig.inc.php.
Creating IPDiscover directory /var/lib/ocsinventory-reports/ipd.
Fixing permissions on directory /var/lib/ocsinventory-reports/ipd.
Creating packages directory /var/lib/ocsinventory-reports/download.
Fixing permissions on directory
↳ /var/lib/ocsinventory-reports/download.
Creating snmp mibs directory /var/lib/ocsinventory-reports/snmp.
Fixing permissions on directory /var/lib/ocsinventory-reports/snmp.
Creating Administration server log files directory
↳ /var/lib/ocsinventory-reports/logs.
Fixing permissions on directory /var/lib/ocsinventory-reports/logs.
Creating Administration server temporary files directory
↳ /var/lib/ocsinventory-reports/tmp_dir.
Fixing permissions on directory /var/lib/ocsinventory-reports/tmp_dir.
Creating Administration server scripts log files directory
↳ /var/lib/ocsinventory-reports/scripts.
Fixing permissions on directory /var/lib/ocsinventory-reports/scripts.
Configuring IPDISCOVER-UTIL Perl script.
Installing IPDISCOVER-UTIL Perl script.
Fixing permissions on IPDISCOVER-UTIL Perl script.
Writing Administration server configuration to file
↳ /etc/apache2/conf-available/ocsinventory-reports.conf
```

```
+-----+
↳ --+
|           OK, Administration server installation finished ;-)           |
|                                                                           |
| Please, review                                                           |
↳ /etc/apache2/conf-available/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
↪ /tmp/OCSNG_UNIX_SERVER-2.12.4/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 ;-)

```

Durant la instal·lació es faran diverses preguntes. Les respostes recomanades per a Ubuntu són:

Pregunta	Resposta
Host del servidor de base de dades	localhost
Port de la base de dades	3306
Directori de configuració d'Apache	/etc/apache2/conf-available
Usuari d'Apache	www-data
Grup d'Apache	www-data
Instal·lar el servidor de comunicació	y
Instal·lar la REST API	y
Instal·lar la consola d'administració	y
Directori de logs	/var/log/ocsinventory-server

L'instal·lador crea els directoris de configuració i logs, copia els mòduls Perl i desplega els fitxers de la consola web. El registre complet de la instal·lació queda guardat a `ocs_server_setup.log` en el directori actual, útil per detectar errors si alguna cosa falla.

6. Configuració d'Apache

6.1. Modifica la configuració de PHP

```
sudo nano -l /etc/php/8.5/apache2/php.ini
```

Modifica les línies 693 i 851

```
post_max_size = 64M
upload_max_filesize = 64M
```

6.2. Propietaris i permisos

Canvia el propietari de la carpeta `/var/lib/ocsinventory-reports/`

```
sudo chown -R www-data:www-data /var/lib/ocsinventory-reports/
```

Canvia el propietari de la carpeta `/usr/share/ocsinventory-reports/ocsreports`

```
sudo chown -R www-data:www-data
↪ /usr/share/ocsinventory-reports/ocsreports
```

Canvia els permisos de la carpeta `/usr/share/ocsinventory-reports/ocsreports`

```
sudo chmod -R 755 /usr/share/ocsinventory-reports/ocsreports
```

Si consultes amb l'ordre `systemctl cat apache2` el contingut dels **fitxers d'unitat**¹ observaràs que Apache té una configuració `ProtectSystem=full`, que fa que `systemd` munti `/usr`, `/boot` i `/etc` en només lectura dins del namespace del procés Apache, excepte les rutes que s'indiquin explícitament a `ReadWritePaths`. Com que `/usr/share/ocsinventory-reports/ocsreports` viu sota `/usr`, i només hi ha `ReadWritePaths` per als directoris de logs i cache d'Apache, Apache no pot escriure-hi encara que els permisos Unix (`www-data:www-data`, `755`) siguin correctes.

Per poder escriure des del navegador a la carpeta `/usr/share/ocsinventory-reports/ocsreports` ho has de configurar a Apache:

```
sudo systemctl edit apache2
```

```
[Service]
ReadWritePaths=/usr/share/ocsinventory-reports/ocsreports
```

I com que has modificat el fitxer d'unitat d'Apache cal llegir la nova configuració.

¹Un fitxer d'unitat (unit file) és el fitxer de configuració que `systemd` fa servir per saber com gestionar un servei, dispositiu, muntatge o qualsevol altre recurs del sistema. És l'equivalent modern als antics scripts `/etc/init.d/` de SysVinit, però en format declaratiu (`clau=valor`) en lloc d'scripts bash.

```
sudo systemctl daemon-reload
```

6.3. Certificat SSL autosignat per a ocs.thos.local

Genera un certificat autosignat vàlid durant 825 dies (límit habitual acceptat per navegadors moderns) amb el Common Name (CN) corresponent al nom del servidor.

```
sudo openssl req -x509 -nodes -days 825 -newkey rsa:4096 \  
-keyout /etc/ssl/private/ocs.thos.local.key \  
-out /etc/ssl/certs/ocs.thos.local.crt \  
-subj "/C=ES/ST=Barcelona/L=Mataro/O=IES Thos i  
↪ Codina/OU=ASIX/CN=ocs.thos.local" \  
-addext "subjectAltName=DNS:ocs.thos.local"
```

L'extensió subjectAltName (SAN) és necessària perquè els navegadors moderns (Chrome, Firefox) ja no validen certificats que només especifiquin el nom al camp CN; cal que el SAN inclogui el mateix nom de domini.

Restringeix els permisos de la clau privada.

```
sudo chmod 600 /etc/ssl/private/ocs.thos.local.key
```

Canvia el propietari de les claus.

```
sudo chown root:root /etc/ssl/private/ocs.thos.local.key
```

```
sudo chown root:root /etc/ssl/certs/ocs.thos.local.crt
```

6.4. Crea la configuració HTTPS

```
sudo nano /etc/apache2/sites-available/ocs-ssl.conf
```

```
<VirtualHost *:443>  
    ServerAdmin webmaster@localhost  
    ServerName ocs.thos.local  
    DocumentRoot /var/www/html  
  
    ErrorLog ${APACHE_LOG_DIR}/error.log  
    CustomLog ${APACHE_LOG_DIR}/access.log combined  
  
    SSLEngine on  
  
    SSLCertificateFile      /etc/ssl/certs/ocs.thos.local.crt  
    SSLCertificateKeyFile  /etc/ssl/private/ocs.thos.local.key
```

```
<FilesMatch "\.(?:cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>
<Directory /usr/lib/cgi-bin>
    SSLOptions +StdEnvVars
</Directory>
</VirtualHost>
```

6.5. Crea la configuració HTTP (redirecció)

Crea un fitxer de configuració de VirtualHost específic per a OCS que redirigeixi a HTTPS

```
sudo nano /etc/apache2/sites-available/ocs.conf
```

```
<VirtualHost *:80>
    ServerName ocs.thos.local
    Redirect permanent / https://ocs.thos.local/
</VirtualHost>
```

6.6. Fitxers de VirtualHosts

L'script d'instal·lació genera automàticament els fitxers de VirtualHosts d'Apache per al servidor de comunicació i la consola web. Cal editar-los per indicar les credencials de la base de dades creada al punt 4.

Obre el fitxer de configuració:

```
sudo nano /etc/apache2/conf-available/z-ocsinventory-server.conf
```

I ajusta (o comprova) els paràmetres de connexió:

```
PerlSetEnv OCS_DB_HOST localhost
PerlSetEnv OCS_DB_PORT 3306
PerlSetEnv OCS_DB_NAME ocsweb
PerlSetEnv OCS_DB_USER ocsuser
PerlSetVar OCS_DB_PWD ContrasenyaSegura123!
PerlSetEnv OCS_OPT_SNMP_LINK_TAG 0
```

Edita la configuració del fitxer zz-ocsinventory-restapi.conf

```
sudo nano /etc/apache2/conf-available/zz-ocsinventory-restapi.conf
```

```
$ENV{OCS_DB_USER} = 'ocsuser';
$ENV{OCS_DB_PWD} = 'ContrasenyaSegura123!';
```

Mou els fitxers de VirtualHosts a la carpeta de llocs disponibles

```
sudo mv /etc/apache2/conf-available/ocsinventory-reports.conf
↪ /etc/apache2/sites-available/
sudo mv /etc/apache2/conf-available/z-ocsinventory-server.conf
↪ /etc/apache2/sites-available/
sudo mv /etc/apache2/conf-available/zz-ocsinventory-restapi.conf
↪ /etc/apache2/sites-available/
```

Habilita els mòduls d'Apache necessaris

```
sudo a2enmod perl rewrite ssl
```

Habilita els fitxers de configuració d'OCS

```
sudo a2ensite ocsinventory-reports.conf z-ocsinventory-server.conf
↪ zz-ocsinventory-restapi.conf ocs-ssl.conf ocs.conf
```

6.7. Sintaxi, reinici i verificació

Comprova que Apache no té errors de sintaxi:

```
sudo apache2ctl configtest
```

Hauria de mostrar Syntax OK.

Reinicia Apache

```
sudo systemctl restart apache2
```

Verifica l'estat

```
sudo systemctl status apache2
```

Sortida esperada:

```
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled;
   ↪ preset: enabled)
   Active: active (running) since Sat 2026-07-04 19:48:59 UTC; 15s
   ↪ ago
  Invocation: cddd287c5ebf496f88fff93feb10b1f5
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 23162 (/usr/sbin/apach)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec:
   ↪ 0; Bytes served/sec: 0 B/sec"
     Tasks: 6 (limit: 1718)
    Memory: 47M (peak: 47M)
       CPU: 374ms
    CGroup: /system.slice/apache2.service
```

```
|23162 /usr/sbin/apache2 -k start -DFOREGROUND
|23165 /usr/sbin/apache2 -k start -DFOREGROUND
|23166 /usr/sbin/apache2 -k start -DFOREGROUND
|23167 /usr/sbin/apache2 -k start -DFOREGROUND
|23168 /usr/sbin/apache2 -k start -DFOREGROUND
|23169 /usr/sbin/apache2 -k start -DFOREGROUND
```

```
de jul. 04 19:48:59 glpi systemd[1]: Starting apache2.service - The
↳ Apache HTTP Server...
de jul. 04 19:48:59 glpi systemd[1]: Started apache2.service - The
↳ Apache HTTP Server.
```

NOTA

Si apareixen errors relacionats amb mòduls Perl que falten, cal instal·lar-los amb `cpan` (com al punt 3) i tornar a reiniciar Apache.

7. Assistent web d'instal·lació

Amb Apache i MariaDB funcionant, obre un navegador i accedeix a:

<https://ocs.thos.local/ocsreports/install.php>

L'assistent web et guiarà per tres passos:

1 - **Connexió a la base de dades:** introduir l'usuari (`ocsuser`), la contrasenya, l'amfitrió (`localhost`), el port (`3306`) i el nom de la base de dades (`ocsweb`).

OCS-NG Inventory Installation

WARNING: You will not be able to build any deployment package with size greater than 64MB
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 `2-ocsinventory-server.conf` in your Apache configuration directory

MySQL login:

MySQL password:

Name of Database:

MySQL HostName:

MySQL Port:

Enable SSL:

SSL mode:

SSL key path:

SSL certificat path:

CA certificat path:

Figura 3: Connexió a la base de dades

2 - **Inicialització de la base de dades:** l'assistent crea automàticament totes les taules necessàries dins `ocsweb`.

3 - Resum de la configuració: es revisen els paràmetres i es finalitza la instal·lació.



Figura 4: Instal·lació

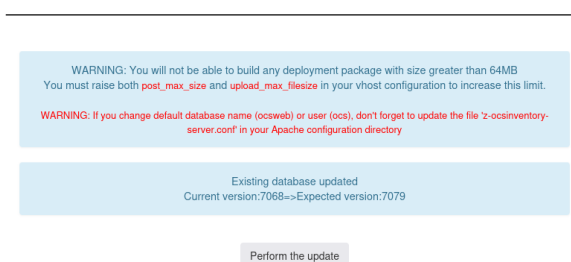


Figura 5: Actualització

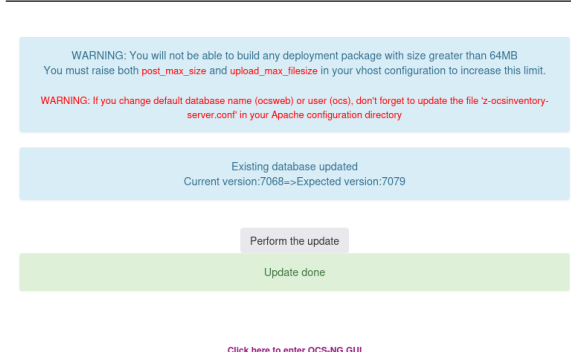


Figura 6: Actualització completada

Un cop completat, **per seguretat cal eliminar l'script d'instal·lació**, ja que qualsevol persona amb accés a l'URL podria tornar a inicialitzar la base de dades:

```
sudo rm -f /usr/share/ocsinventory-reports/ocsreports/install.php
```

Finalment, accedim a la consola d'OCS Inventory:

```
https://ocs.thos.local/ocsreports
```

OCSS
inventory

IDIOMA
Español

Usuario :
admin

Contraseña :

Aceptar

Figura 7: Login

Amb les credencials per defecte:

- **Usuari:** admin
- **Contrasenya:** admin

PRECAUCIÓ

És imprescindible canviar la contrasenya per defecte just després del primer accés, des de l'apartat de configuració d'usuaris de la consola.

OCSS
inventory

Todos los dispositivos Inventario Distribución software Configurar Gestión Complementos/Plugins Información Ayuda

¡ALERTA DE SEGURIDAD!
El inicio de sesión/contraseña predeterminado se activa en la GUI de OCS

MI CUENTA
DEPURAR / DEBUG
SALIR

ID Usuario : admin

Tipo : Super administradores

Grupo :

Nombre : admin

Apellido : admin

E-mail : admin

Comentarios : Default administrator account

Contraseña : *

OK Cancelar

Figura 8: Canvi de contrasenya

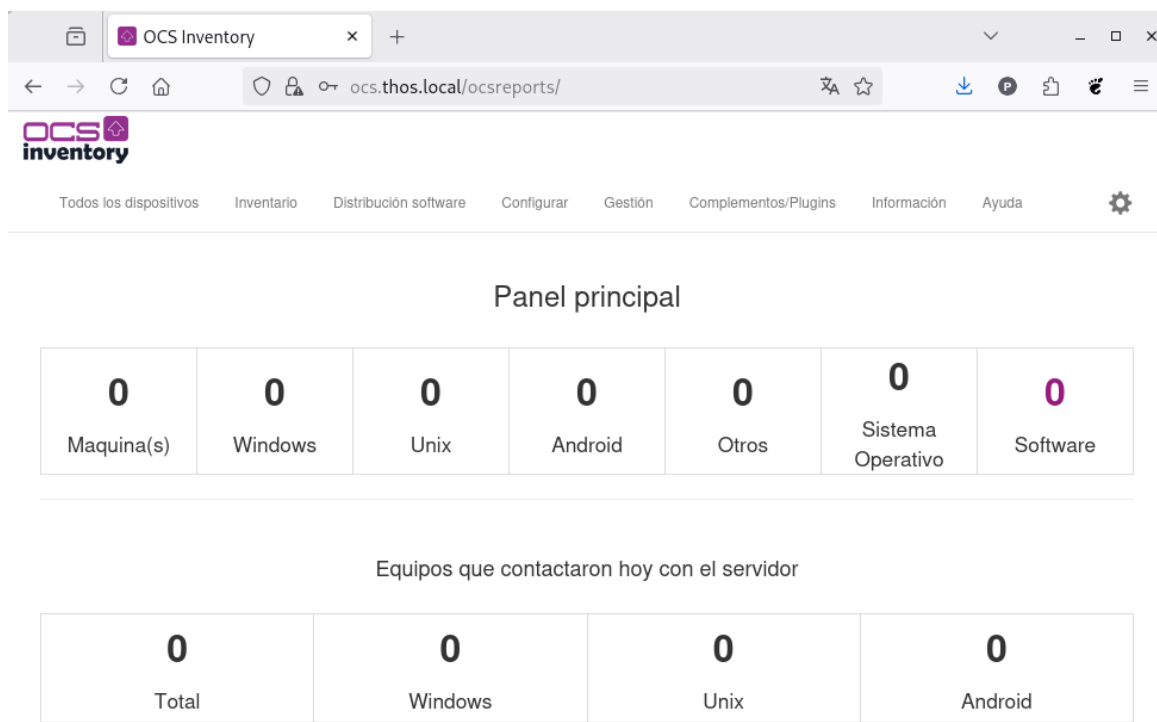


Figura 9: Tauler de control

8. Configuració del tallafocs

Si el servidor té UFW actiu, cal obrir els ports HTTP i HTTPS:

```
sudo ufw allow 80/tcp
sudo ufw allow 443/tcp
sudo ufw reload
sudo ufw status
```

9. Instal·lació de l'agent a clients GNU/Linux (Debian/Ubuntu)

L'agent Unix d'OCS Inventory recull la informació de maquinari i programari del client i l'envia al servidor de comunicació.

9.1. Dependències del client

Actualitza la llista de paquets

```
sudo apt update
```

Instal·la les dependències

```
sudo apt install libmodule-install-perl dmidecode libxml-simple-perl \  
libcompress-zlib-perl libnet-ip-perl libdigest-md5-perl \  
↪ libdata-uuid-perl \  
libcrypt-ssleay-perl libnet-ssleay-perl liblwp-protocol-https-perl \  
libproc-pid-file-perl libproc-daemon-perl libfile-pid-perl \  
↪ net-tools \  
libsys-syslog-perl pciutils ipmitool nmap libnet-snmp-perl \  
↪ libnet-netmask-perl
```

9.2. Descarrega i compila l'agent

Mou-te a la carpeta temporal

```
cd /tmp
```

Descarrega l'agent

```
wget -c https://github.com/OCSInventory-NG/UnixAgent/releases/download/  
↪ ad/v2.10.5/Ocsinventory-Unix-Agent-2.10.5.tar.gz
```

Descomprimeix

```
tar -xzf Ocsinventory-Unix-Agent-2.10.5.tar.gz
```

Mou-te a la carpeta creada

```
cd Ocsinventory-Unix-Agent-2.10.5
```

Compila i instal·la

```
sudo perl Makefile.PL  
sudo make  
sudo make install
```

Durant l'execució de Makefile.PL, l'instal·lador demanarà alguna informació rellevant:

- **URL del servidor OCS:** `http://IP_DEL_SERVIDOR/ocsinventory`
- **Tag (opcional):** una etiqueta per agrupar l'equip (per exemple, el nom de l'aula o del departament: `Aula-Informatica-01`)

```
Do you want to configure the agent?
Please enter 'y' or 'n'?> [y]
Where do you want to write the configuration file?
 0 -> /etc/ocsinventory
 1 -> /usr/local/etc/ocsinventory
 2 -> /etc/ocsinventory-agent
?> 2
Do you want to create the directory /etc/ocsinventory-agent?
Please enter 'y' or 'n'?> [y]
Should the old unix_agent settings be imported?
Please enter 'y' or 'n'?> [y] n
[info] The config file will be written in
  ↪ /etc/ocsinventory-agent/ocsinventory-agent.cfg,
What is the address of your ocs server?> 10.0.2.10
Do you need credential for the server? (You probably don't)
Please enter 'y' or 'n'?> [n]
Do you want to apply an administrative tag on this machine?
Please enter 'y' or 'n'?> [y]
tag?> client01
Do you want to install the cron task in /etc/cron.d?
Please enter 'y' or 'n'?> [y]
Where do you want the agent to store its files? (You probably don't
  ↪ need to change it)?> [/var/lib/ocsinventory-agent]
Do you want to create the /var/lib/ocsinventory-agent directory?

Please enter 'y' or 'n'?> [y]
Should I remove the old unix_agent?
Please enter 'y' or 'n'?> [n]
Do you want to activate debug configuration option?
Please enter 'y' or 'n'?> [y]
Do you want to use OCS Inventory NG UNix Unified agent log file?
Please enter 'y' or 'n'?> [y]
Specify log file path you want to use?> /var/log/ocs_agent.log
Do you want disable SSL CA verification configuration option (not
  ↪ recommended)?
Please enter 'y' or 'n'?> [n]
Do you want to set CA certificates file path?
Please enter 'y' or 'n'?> [y] n
Do you want disable software inventory?
Please enter 'y' or 'n'?> [n]
Do you want to use OCS-Inventory software deployment feature?
Please enter 'y' or 'n'?> [y]
Do you want to use OCS-Inventory SNMP scans feature?
Please enter 'y' or 'n'?> [y]
Do you want to send an inventory of this machine?
Please enter 'y' or 'n'?> [y]
Setting OCS Inventory NG server address...
Looking for OCS Inventory NG Unix Unified agent installation...
ocsinventory agent presents: /usr/local/bin/ocsinventory-agent
Setting crontab...
Creating /var/lib/ocsinventory-agent directory...
Creating /etc/ocsinventory-agent directory...
Writing OCS Inventory NG Unix Unified agent configuration
```

```
Creating /var/lib/ocsinventory-agent/http:__10.0.2.10_ocsinventory
↳ directory...
Creating
↳ /var/lib/ocsinventory-agent/http:__10.0.2.10_ocsinventory/snmp
↳ directory...
Copying SNMP MIBs XML files...
Activating modules if needed...
Launching OCS Inventory NG Unix Unified agent...
-> Success!
New settings written! Thank you for using OCS Inventory
```

9.3. Primera execució manual de l'agent

Per enviar el primer inventari i comprovar que tot funciona:

```
sudo ocsinventory-agent --server=http://IP_DEL_SERVIDOR/ocsinventory
↳ --force
```

Si tot va bé, l'agent mostrarà un missatge confirmant l'enviament correcte de l'inventari, i l'equip apareixerà al cap de pocs segons a la consola web (apartat **Tots els ordinadors**).

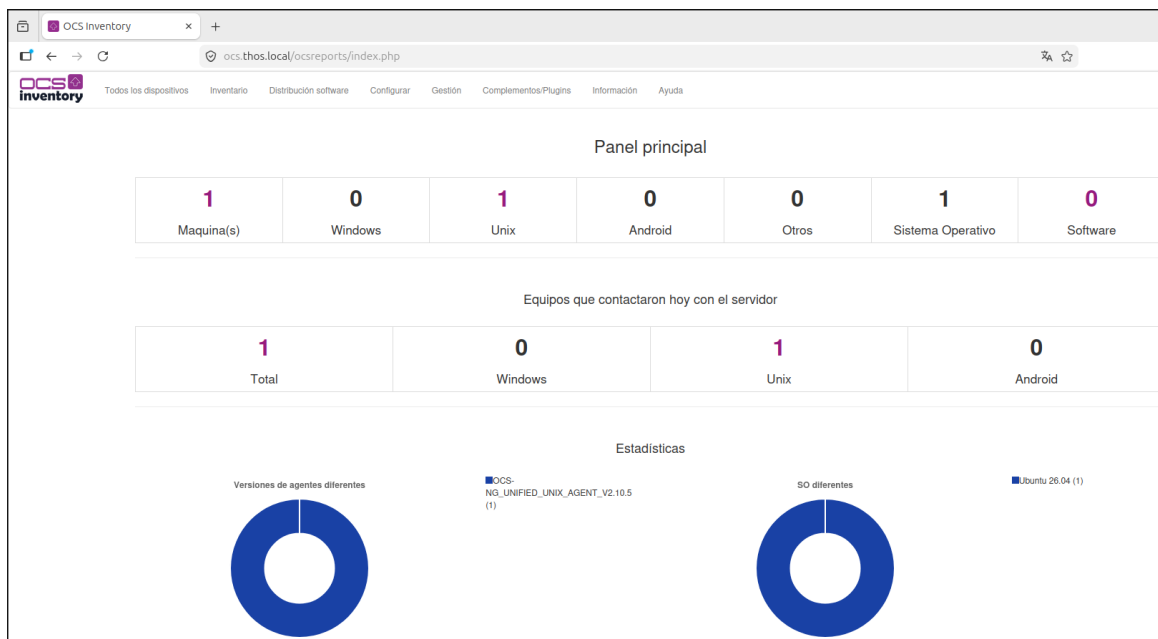


Figura 10: Informació del client GNU/Linux al tauler de control

9.4. Automatització amb cron

Per evitar haver d'executar l'agent manualment, es pot programar una tasca periòdica (per exemple, cada dia a les 2:00):

```
echo "0 2 * * * root /usr/local/bin/ocsinventory-agent  
↪ --server=http://IP_DEL_SERVIDOR/ocsinventory" | sudo tee  
↪ /etc/cron.d/ocsinventory-agent
```

NOTA

En distribucions modernes, l'instal·lador de l'agent sol crear ja un servei o tasca programada per defecte; cal comprovar-ho amb `systemctl status ocsinventory-agent` abans de duplicar la programació.

10. Instal·lació de l'agent a clients Windows

OCS Inventory proporciona un instal·lador natiu per a Windows (en el moment d'escriure aquest document, la versió 2.11.0.1).

10.1. Instal·lació manual (interactiva)

1. Descarregar l'instal·lador des de la [pàgina de releases de l'agent Windows](#), triant OCS-Windows-Agent-Setup-x64.exe per a sistemes de 64 bits.
2. Executar l'instal·lador com a administrador.
3. Durant l'assistent, configurar:
 - **URL del servidor:** `http://IP_DEL_SERVIDOR/ocsinventory`
 - **Autenticació del servidor:** deixar en blanc, tret que s'hagi configurat autenticació HTTP a Apache
 - **Proxy:** configurar-lo només si la xarxa ho requereix
 - **Llançar l'inventari immediatament:** marcar aquesta opció per enviar l'inventari just després d'instal·lar
 - **TAG:** etiqueta opcional (per exemple, l'aula o el departament)

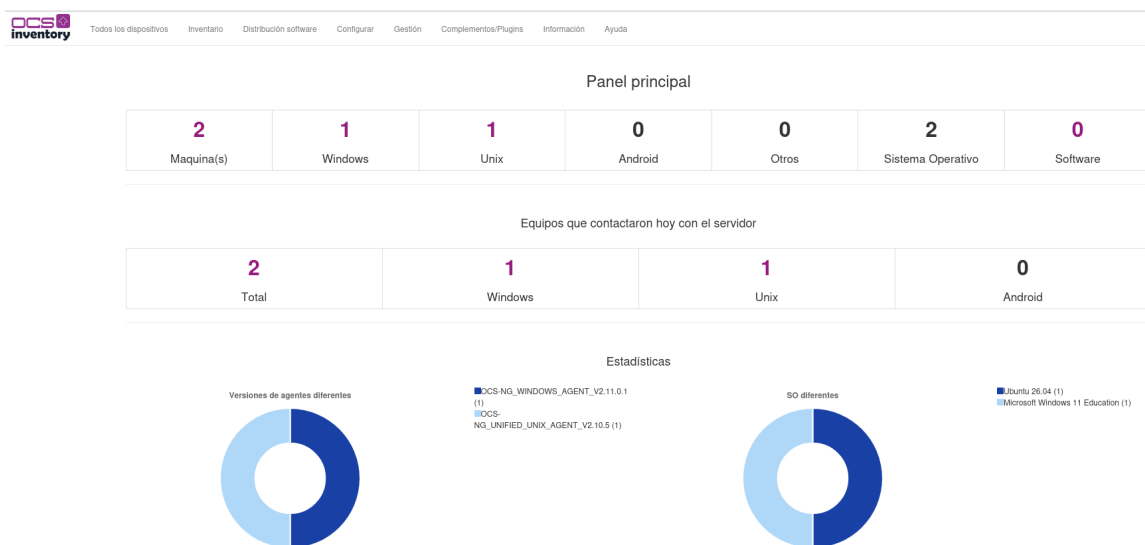


Figura 11: Informació del client Windows al tauler de control

L'agent s'instal·la com a servei de Windows (**OCS Inventory Service**) i s'inicia automàticament en cada arrencada del sistema. Per defecte, envia l'inventari cada 24 hores (interval configurable des del servidor).

10.2. Desplegament silencios (GPO / SCCM / script)

Per a desplegaments massius en una aula o organització, es pot instal·lar l'agent en mode silencios des de la línia d'ordres:

```
OCS-Windows-Agent-Setup-x64.exe /S  
↪ /SERVER=http://IP_DEL_SERVIDOR/ocsinventory /NOW
```

Paràmetre	Funció
/S	Instal·lació silenciosa (sense interfície)
/SERVER=	URL del servidor de comunicació OCS

Aquesta ordre es pot incorporar fàcilment a un script d'inici de sessió, a una GPO de l'Active Directory o a una eina de desplegament com SCCM, ideal per a un entorn d'aula amb molts equips Windows.

11. Comprovació i ús bàsic de la consola

Un cop els agents comencen a reportar, des de la consola web (http://IP_DEL_SERVIDOR/ocsreports) es pot:

- Consultar **Tots els ordinadors** per veure cada equip que ha reportat inventari: CPU, RAM, placa base, BIOS, discs, particions, targetes de xarxa, programari instal·lat amb versions, processos i serveis en execució.
- Fer cerques avançades per filtrar equips segons maquinari, programari instal·lat, versió del sistema operatiu o camps personalitzats.
- Crear **grups dinàmics** basats en criteris de cerca (per exemple, tots els equips amb menys de 4 GB de RAM, o tots els que tenen una versió concreta d'un programa).
- Ampliar la funcionalitat mitjançant **extensions/plugins**, instal·lant-les al directori `/usr/share/ocsinventory-reports/ocsreports/extensions/` i activant-les des de **Extensions** → **Gestionar extensions**.

12. Recomanacions per a producció

- Activa **HTTPS** amb un certificat SSL vàlid per xifrar la comunicació entre agents i servidor.
- Programa **còpies de seguretat regulars** de la base de dades amb `mysqli_dump`.
- Si el volum d'equips és gran, considera situar el servidor OCS darrere d'un **proxy invers** per repartir millor la càrrega.
- **Canvia sempre la contrasenya** per defecte de l'usuari `admin` just després de la instal·lació.
- **Esborra l'script** `/usr/share/ocsinventory-reports/ocsreports/install.php` just després de la instal·lació.

13. Documentació

Lloc web - <https://ocsinventory-ng.org/?lang=en>

Documentació - <https://wiki.ocsinventory-ng.org/>

GitHub d'OCS Inventory Server - <https://github.com/OCSInventory-NG/OCSInventory-ocsreports>

GitHub de l'Agent Unix - <https://github.com/OCSInventory-NG/UnixAgent/tree/master>

GitHub de l'Agent Windows - <https://github.com/OCSInventory-NG/WindowsAgent>

Versions d'aquest document

- HTML - [ocs.html](#)
- PDF - [ocs.pdf](#)
- ODT - [ocs.odt](#)
- MD - [ocs.md](#)

[Domini Públic \(CC0\)](#)