FusionInventory !

Contents

Introduction

Introduction

TODO

Links

Links of project to know

Internet site

Internet address of the project

You can find the projects' website on:

http://www.fusioninventory.org/

Development forge

Address of the forge

The development forge can be found under:

http://forge.fusioninventory.org/

Mailing lists

Address of mailing lists

FusionInventory users (*archive*)
FusionInventory developers (*archive*)
FusionInventory internationalization and translation (*archive*)

IRC-Channel

Address of the irc-Channel

Here are the informations to find the projects' IRC channel:

Server: freenode
Channel: #fusioninventory

Forums

Forums' address

You can find the projects' forums on:

http://forum.fusioninventory.org/

Versions

Versions used for this manual

Here is the versions list of each FusionInventory module and component described in this manual.

Agent FusionInventory

Version of the FusionInventory Agent

The version of the agent used for this manual is: **2.1.5**

Network discovery module

Version of the discovery module (NETDISCOVERY)

The version of the "NETDISCOVERY" module is: **1.2**

SNMP inventory module

Version hf the SNMP inventory module (SNMP (SNMPQUERY)

The version of the SNMP inventory module "SNMPQUERY" described in this manual is: **1.2**

P2P OCS deployment module

Version of the P2P OCS module (OCSDEPLOY)

The version of P2P OCS deployment module described in this manual is: **1.0.8**

FusionInventory plugin for GLPI

GLPI plugin version

The version of the GLPI plugin described in this manual is: **2.3.0**

Lib serveur php

Version de la lib serveur php

La lib serveur php n'a pas encore été releasé, elle est encore en cours de développement / test.

Part
I

FusionInventory-Agent

Topics:

Everything you need to know about the agent.

Learn his different capacities, how to install and configure it.

The agents' installation

Installation guide.

Tout ce qu'il faut savoir pour une installation confortable et sereine de l'agent FusionInventory.

In this chapter you will find in the first time all the requirements for the installation of the agent and then we describe the different manners how this can be done.

Requirements

Requirements for the agent's installation

The agent can be installed on the following operating systems:

- Windows (from 2000 up to 7 including Server editions), 32 and 64 bits ;
- Linux-based systems ;
- FreeBSD, OpenBSD, NetBSD, DragonFlyBSD ;
- MacOS X (from 10.3.9 PPC to 10.6 Intel)
- Solaris, OpenSolaris, OpenIndiana (SPARC and Intel)
- AIX
- HP-UX (Pa-RISC and IA64)

More informations are available here : *http://forge.fusioninventory.org/projects/fusioninventory-agent/wiki/Agent_supportedplateforms*

How to install FusionInventory-Agent

Learn how to install FusionInventory-Agent on various systems.

The are different ways to install FusionInventory-Agent depending on your system. This chapter will introduce the possible solutions.

Install with packages

How to install with help of packages

This is all informations to install with packages. It depend of operating systems

Using FusionInventory debian packages

Installation with the debian packages

Some distributions like Debian do already have versions of the FusionInventory agent in their distribution repositories - But they are more lilkely being older or even quite outdated.

The FusionInventory project maintains an up-to-date debian package of the most recent version, in a extra repository. To use it, edit the following file:

```
/etc/apt/sources.list
```

Add the following line:

```
deb http://debian.fusioninventory.org/debian/ stable main
```

Update the local package list:

```
apt-get update
or
aptitude update
```

List all the FusionInventory packages available:

```
apt-cache search fusioninventory
```

Simply run *apt-get install* fusioninventory to install the FusionInventory agent and his dependencies. If you want additional tasks to be performed by this agent (like network discovery and SNMP inventory), you'll have to install

additionnal packages : *libfusioninventory-agent-task-netdiscovery-perl* and *libfusioninventory-agent-task-snmpquery-perl*.

Using RPM packages

Installation with the rpm package

The RPM-Packages are mainained by Remi Collet.

First of all, you need to enable EPEL repository (*http://fedoraproject.org/wiki/EPEL*) and also the Remi repository (*http://blog.famillecollet.com/pages/Config*)

To install FusionInventory-Agent, simply execute the following command:

```
yum --enablerepo=remi install fusioninventory-agent
```

Important: Never use Perl packages provided by RPMForge.

Windows installer

How to install the agent under Windows

You have to download the installer package from the following FusionInventory page:*http://prebuilt.fusioninventory.org/stable/windows-i386/*

You will just have to launch the installer for an interactive installation process.

Note: The installer contains an embedded version of Perl that was specially made for usage with FusionInventory. It's not necessary to additionally install Perl for this. The embedded perl doesn't conflict with another version that may be installed on the system.

Note: If you want to use the network discovery and nmap feature, you will have to install the Windows-version of nmap. You can find the most recent version here: *http://nmap.org/download.html*.

Parameter for Windows installer

A list of parameter are availaible to perform silent installation :

- */h*: Display the help
- */S*: Run setup in silent mode
- */server=url1,url2*: Set URL of the server
- */tag=tagame*: Set tag of agent
- */user=username*: Set user name
- */pass=password*: Set password
- */realm=realm*: Set realmi (optional)
- */ca-cert-file=filename*: set the file name of the certificate
- */no-ssl-check*: No certificate check when establishing SSL connection
- */proxy=http://proxy/*: Set proxy
- */debug*: Set debug mode
- */rpc-trust-localhost*: Allow local user to wake agent up
- */scan-homedirs*: Look for virtual machines in users home directories
- */runnow*: Turn this parameter on if you want a first inventory just after the installation

Example:

```
fusioninventory-agent_windows-i386_2.1-1.exe /S /debug /runnow /
server=https://myown-ocs-server.mydomain.de/ocsinventory"
```

Using Mac OS X Installer

Installation of the agent with the Mac OS X installer

The FusionInventory project provides 2 Mac OS X packages: one for powerpc computers and the other for Intel macs.

The package is a .pkg file, and is designed for silent installation. Agent's configuration needs to be done before install, by going into the package, and edit the *Content\Resources\agent.cfg* file.

By default, the agent runs as a daemon, and stores his files in /opt/fusioninventory-agent.

Please note that FusionInventory agent is not a Mac OS X application, and his invisible for users.

Once configured, the agent can be deployed either manually (by double clicking on the .pkg file), nor automatically using tools like Apple Remote Desktop.

Installation with Perl CPAN

Using perl CPAN to install FusionInventory-Agent

Perl language provides a centralize repository for modules called CPAN.

To install FusionInventory from CPAN, use this command:

```
cpan FusionInventory::Agent::Conf
```

Perl modules dependancies will be automatically computed and installed.

Using a prebuilt agent

How to use the pre-built versions

There are also pre-compiled or pre-built versions of the agent maintained by the project. The idea is to launch the agent without installing anything on your system. Here are the steps :

- Download the prebuilt version for your OS: *http://prebuilt.fusioninventory.org/stable/*
- Uncompress the downloaded archive
- Configure the agent by editing the agent.cfg file
- Launch the agent with the following command as root :

```
./fusioninventory-agent
```

Note: the FusionInventory project recommands usage of packages (deb, rpm, etc.) when it's possble. Prebuilts are interesting for testing, or when you have no possibility to install packages on your system.

TODO

TODO

TODO

Agent settings

How to configure the agent

The agent settings are based on a classic key/value mechanism.

It reads its settings differently depending on the operating system:

Windows: with the registry keys located at KEY_LOCAL_MACHINE/SOFTWARE/FusionInventory-Agent
MacOSX: the configuration file is located in /opt/fusionInventory-agent/agent.cfg
UNIX/Linux: on UNIX/Linux, the agent expect its configuration to be here /etc/fusioninventory/agent.cfg
prebuilt: if you use a prebuilt package, the agent.cfg configuration is in the packages' root directory.

TODO

TODO

TODO

Local machine inventory

How to do a local machine inventory

You can generate a local inventory of a machine in XML or HTML :

on Windows, you must first go in the FusionInventory-Agent installation directory and then in the perl/bin subdirectory:

```
perl fusioninventory-agent --local c:/tmp
```

on an UNIX systems:

```
fusioninventory-agent --local c:/tmp
```

To perform an HTML inventory, add the **--html** switch:

```
fusioninventory-agent --local c:/tmp --html
```

Send a local XML inventory file to an inventory server

You can push a local generated inventory file to a FusionInventory for GLPI server by using the fusioninventory-injector command:

: fusioninventory-injector is avalaible only since the 2.1.7 release of the agent.

```
     fusioninventory-injector -v -f /tmp/toto-2010-09-10-11-42-22.ocs --url
https://login:pw@yourserver/glpi/plugins/fusioninventory/
```

If you want to use a HTTP proxy, you can declare it with the http_proxy environment variable:

```
export http_proxy=http://www-proxy:3128
```

Please note that this scripts works with both FusionInventory for GLPI and OCS Inventory NG Server.

TODO

TODO

TODO

Network discovery

Netwok discovery

TODO

Description

Description

TODO

Scan réseau

Scan réseau

TODO

Nmap
Nmap scan

TODO

NetBIOS scan
Scan Netbios

TODO

SNMP
Scan SNMP

TODO

Network inventory

network inventory

Description

Description

Network inventory as for goal to get by SNMP informations of network devices like switch, network printers...

Informations retrieved

Informations retrieved

Description of inventoried data.

Hardware and software informations
Hardware and software informations

We can get informations like :

- Device name
- Device model
- Firmware (version)
- Serial number
- Memory
- Manufacturer name
- IP
- MAC

Switch and router informations
Switch and router informations

Network inventory get more informations of switch and routers.

Network ports informations
Network ports informations

Network inventory get each physical port and its informations :

- *Name* : port name
- *MTU* :
- *Speed* : Port speed (10, 100, 1000... Mbps)
- *Internal status* : port is active or not
- *Last changement* : time since last modification of port
- *Number of octets received*

- *Number of errors in input*
- *Number of octets sent*
- *Number of errors in output*
- *Duplex* : Port is in half-duplex or full duplex
- *Internal MAC* : MAC address of the port (can be different of switch mac address)
- *VLAN* :
- *Connexion* : List of MAC connected to the port
- *Description port* :

Devices connected on each port

Devices connected on each port

This module can get MAC addresses of devices connected on each ports.

Important: These MAC addresses can be get only on switch of the manufacturers Cisco, Hewlett-Packard and Nortel.

Attention: The MAC addresses of these devices are stored in cache some minutes (see configuration of switch) and if no network trames go on switch, this MAC address will not found in switch.

Printers informations

Printers informations

TODO

Network ports informations

Network ports informations

TODO

Cartridges state

Informations of cartridges state

TODO

Pages counter

Pages counter informations

TODO

Wake on LAN

Wake on LAN: Remote wake up of computer

This chapter treats the remote wake up of computers - with FusionInventory.

Description

Description

Wake on LAN (WoL) allows remote wake-up of computers and workstation over the network by sending a signal over the network that is then received by the clients' network adapter which then powers up the motherboard.

Sometimes it can be useful e.g. for installing updates outside working hours in the night so the users aren't disturbed.

It's with this focus that this module has been developed.

Wake on LAN on OSI Layer 2

The OSI Layer 2

This method is the best. The agent send magic packet directly on layer 2 of OSI on the network.

Important: Yet this method only works with agents installed on Linux distributions (Debian, Fedora, RedHat, Ubuntu ...).

Wake on LAN over UDP

Wake on LAN over UDP

This metod is a substitute for the method that works on OSI Layer 2. This method sends a magic-packet over UDP (usually Port 7 or 9). This is using a broadcast and won't work normally in routed networks. Sometimes you may need to used so-called directed broadcasts which normally requires configuration steps on your switches / routers.

Important: This method won't always work. Il peut y avoir un nombre important d'échecs de démarrage des machines.

OCS Deploy P2P

OCS deploy P2P: Application deployment with peer to peer data exchange

This module is designed to work with an OCS Inventory server and especially with the OCS Reports web console. Please read OCS Inventory documentaton to go further. *http://wiki.ocsinventory-ng.org/index.php/Main_Page*

Description

Description

This mode allows (unattended) deployment of applications, execution of commands and allows also simple file transfer.

How peer 2 peer deployment feature works

How the peer 2 peer feature works

Peer 2 peer (P2P) allows:

- Reduce bandwidth between central server (useful when have remote site) and the computer
- Get files very quickly

How it works:

- The first client downloads the fragments of a pckage from the central server.
- The other computers check the other clients' agents on the local site whether they have these fragments. Depending on the answers the agent gets:
  - if the fragments are found on other computers, then they are downloaded (they will then only be downloaded from the local site)
  - if the fragments are not found on other agents / systems, then they will be downloaded from the central server.

During a deployment on serveral machines who are on the same site, you can see a significant better bandwith usage.

Note: With the P2P feature you can e.g. avoid having local server with synchronization to a remote master server. Etc. In this case you can start automated application deployment without changing the architecture of an already present network.

Plugins FusionInventory for GLPI

Installation guide and users manual for the plugins FusionInventory for GLPI

Plugin for GLPI : fusioninventory (core)

Installation guide and users manual for the plugin for GLPI : fusioninventory (core).

Description

Description of plugin for GLPI : fusioninventory (core)

This plugin is the core of different other plugins (fusinv....).

It gives many functions and have the communication part with the agents.

Requirements

Requirements for using this plugin for GLPI : fusioninventory

In order to install and use the plugin FusionInventory for GPLI on your server, the following requirements must be verified:

GLPI 0.78.x (0.72 and older is not supported)
PHP 5 or greater
PHP-ZLIB
PHP-GD
MySQL 5 or greater

Features

Features list

This is features list :

Communication in HTTP and/or HTTPS between agent and plugin (server).
Agent wake up features.
Agents management with actions (enable / disable), link with computers.
Management of unknown devices.
Complete scheduled task system.

Installation

Plugin installation

Procedure list of installation on different cases.

New installation

How to install the plugin fusioninventory

To install it, you must do :

Download archive here : *http://forge.fusioninventory.org/projects/fusioninventory-for-glpi/files*
Uncompress the file into the plugin folder of GLPI. File list seems to be like :

- glpi

--- plugins

-----fusioninventory

-------index.php

------- hook.php

------- front

------- inc

etc...
Connect on GLPI on super-admin
Go into the menu *Configuration > Plugins*
Install FusionInventory
Activate FusionInventory

These operations will create tables in database MySQL and rights on the plugin.

Update plugin FusionInventory for GLPI

This describe how to update plugin FusionInventory for GLPI

To update, you must :

Download archive of the plugin here : *http://forge.fusioninventory.org/projects/fusioninventory-for-glpi/files*
Delete the folder "fusioninventory" in glpi/plugins
Uncompress the file into the plugin folder of GLPI. File list seems to be like :

- glpi

--- plugins

-----fusioninventory

-------index.php

------- hook.php

------- front

------- inc

etc...
Connect to GLPI on super-admin
Go into the menu *Configuration > Plugins*
Install FusionInventory. The plugin will update. Update process can be short (few seconds) to long (many minutes or one/two hours).
Activate FusionInventory

These operations will modify tables and datas in database MySQL and rights on the plugin.

Update from Tracker plugin

How to update Tracker plugin to FusionInventory plugin

Tracker is the old name of the plugin FusionInventory for GLPI.

To update, you must :

Delete the folder "tracker" in glpi/plugins
Download archive of the plugin here : *http://forge.fusioninventory.org/projects/fusioninventory-for-glpi/files*
Uncompress the file into the plugin folder of GLPI. File list seems to be like : :

- glpi

--- plugins

-----fusioninventory

-------index.php

------- hook.php

------- front

------- inc

etc...
Connect to GLPI on super-admin
Go into the menu *Configuration > Plugins*
Install FusionInventory. The plugin will update the plugin. Update process can be short (few seconds) to long (many minutes or one/two hours) if you have a very old version of Tracker and if you have lots of data in the plugin.
Activate FusionInventory

These operations will modify tables and datas in database MySQL and rights on the plugin.

Configuration

Configuration of plugin for GLPI : fusioninventory

In this topic, the general configuration is describe.

This configuration page is available in menu *Plugins > FusionInventory* and after by the icon with a *key* in the top of the page.

General configuration

General configuration

This is available in the tab *General configuration*.

SSL only for agent : activate or not communication only in HTTPS between plugin and agents.
Inventory frequency (in hours) : parameter for the PROLOG function of agent. With it, agent must contact plugin (server) always between 0 and xx hours (configured here).
Delete tasks after : Delete tasks logs xx hours after they are finished.
Agent port : it's the port of agent to wake it from the plugin. (by default, port is 62354).
Extra-debug : plugin write log files to debug, with one file for communication trace, another for the rules usage...

Agents modules

Agents modules

This is available in the tab *Agents modules*.

It is possible to define for an agent what it is allowing to do.

Each plugin *fusinv[...]* can add one or more modules in this menu; by default, you have only WAKEONLAN because it is integrated in this plugin.

For each module you can configure :

Activation by default : default activation of this module for all agents (it's activated or not).
Agents exceptions : It depends on *Activation by default* but invert activation for each agent selected.

Example: if WAKEONLAN is not activated by default and we put an exception of agent "computer01-2010-11-24", this agent will be able to get order "WAKEONLAN" from plugin.

.

Agents management

Agents management

This is available in the menu *Plugins > FusionInventory > Agents management*.

It's possible to manage and configure agents here.

Agents creation

Agents creation

How agents is created here?

It is so simple : Creation is created itself!

When installation of agent is done on a computer, agent run and contact for the first time the server (plugin fusioninventory). The plugin register the agent with some informations in this menu.

Agents information

Agents information

Some informations are get from agent dynamically and can't be modified :

device_id : it's the identifier of the agent generated it is running first time.
Token : it's the key used to wake agent remotely. It is regenerated each time agent begin to run and sent to server.

Version : it's the versions of agent (version core and for each modules). This version is only get when agent send informations of each modules to plugin.
Last contact : it's the last time agent have contacted plugin.

Other informations can be modified :

Name : it's the name of agent. By default, name is *device_id*.
Locked : if it's set to YES, when agent contact plugin to get list of actions, server answer to do nothing. With this, agent is temporary disabled.
Computer link: It's the link with a GLPI computer. The goal is to have IP of computer to wake up the agent. This link can be made automatically if you use Inventory function (inventory of the computer where agent is installed).

Agents modules

Agents modules

This is available in the tab *Agents modules*.

In this tab, is displaying the fusioninventory modules and if they are activated or not for this agent.

So you can see in this tab what modules are allowed to be running for this agent.

The default activation (or not activation) is controlled by the general configuration page (*Agents modules*). If you inverse check, the plugin add an exception that you can see in general configuration page.

Tasks management

Tasks management

This is available in the menu *Plugins* > *FusionInventory* > *Task management*.

It is possible to manage tasks and configure here.

Tasks description

Tasks description

The tasks are defined by 2 things :

Tasks : task are great definition with *name*, *activation*, *communication*...
Actions : defined in a task, theses actions are designed to make wakeonlan, inventory, netdiscovery... on agents. You can have many actions on one task.

These tasks are used to order to agents what to do. They can be scheduled and have periodicity parameter to run for example every hour.

Tasks usage

Tasks usage

The tasks must be defined by different fields :

Name : used to identify task. Think to put a complete name because if there is lots of tasks, it must be better to find task you want to display or configure.
Active : enable or disable a task.
Communication type :

Push (server wake agent, agent contact server and give task to it)
Pull (server wait agent contact server itself and get tasks to it)

Permanent : some tasks may be permanent. these tasks are defined in other menu (like in IP range of fusinvsnmp plugin). Permanent mean task can be controlled and is attached to an other GLPI object.
Scheduled date: You define the date / time of next execution of the task. This date will be updated after each running if periodicity is set.
Periodicity : define time between 2 runs of this task.

Task actions

Task actions

A tasks is defined by one or more action.

Depends of plugins *fusinv[...]* installed, for example, you can have a wakeonlan of a computer at 1 hour am and a local inventory at 2 hours am.

There are 2 types of tabs :

New action: form used to add a new action.
Action [number]: it's an action defined. Number is incremental, so if there are 4 actions, 4 tabs are displayed : *Action 1*, *Action 2*, *Action 3* and *Action 4*. These tabs are sorted by time of execution (*Action 1* at 0 seconds after task begin , *Action 2* at 30 minutes after the first action etc...).

Action description

Action description

An action is defined by fields:

Name: name for identify this action.
Time between task start and start this action: It's the time when action will start after the task is beginning. If are set 10 minutes, task start at 1 hour am and this action begin at 1 hour and 10 minutes am.

: Always have at minimum, an action of each task with this value set to 0.
Number of trials: It's possible to define number of trials if there are errors on execution of the task.
Time between 2 trials (in minutes): Number of minutes between 2 trials.
Module: Module defined the action (WakeOnLan, Inventory, netdiscovery...). These modules come from each plugins *fusinv[...]* installed.

Action definition

Action definition

Definition in action describe what definition of action like a computer to wake up (WakeOnLan module), IP range to netdiscovery (network discovery) ...

You can select a type defined for each module and after an object of this type. Click on *Add* button to add it in selection list. You can have many objects in this definition list.

Action action

Action action

Action in an action witch describe with what agent or computer you can to this tack action.

One or more agents can be selected.

For some method, you can have 2 other choices for agent actions :

Auto management dynamic of agents: can be used only in *push* communication. Task will detect online and not running agent are allowed to run this method and give definitions to these agents.
Auto management dynamic of agents (same subnet): can be used only in *push* communication. Task will detect online and not running agent in the same subnet of definition (for SNMP inventory switchs for example) are allowed to run this method and give definitions to these agents.

Rules (Equipment import and link rules)

Rules system for import and link equipments

Rules description

Rules description

These rules are used to search and link the device (insert or update datas) get from agent with devices in GLPI.

The engine rules will stop when a rule is checked.

Rules working

Rules working

This section will describe how rules works.

For each import device from agent, device informations pass on rules to try to discover is device exist in GLPI inventory. If find, this device will be updated, else device will be created.

Rules criteria

Rules criteria

Many possibilities are possible to have a criteria.

This is the different operators :

is : the criteria is a value
is not : the criteria is not a value
contains : the criteria have this value in the name
does not contain : the criteria don't have this value in the name
start with : the criteria have is a name begin with this value
finished by : the criteria have is a name finish with this value
regular expression checks : the criteria check this regular expression
regular expression does not check : the criteria doesn't check this regular expression
exist : the criteria is defined
not exist : the criteria is not defined
is present in GLPI : this criteria value is find in GLPI
is empty : this criteria is empty

With all of these operators, it is possible to make really powerfull rules.

Rules actions

Rules actions

These are 2 possibilities for actions.

link if possible : the rule will link a device enter in rules with a GLPI device if find it or continue this next rule
link if possible, else import denied : the rule will link a device enter in rules with a GLPI device if find it, else end rules here and device not created (so device import ignored).

Unknown devices

Management of temporary area between real devices and devices managed in GLPI

Unknown devices is a temporary area between real devices and devices managed into GLPI.

What does it mean?

Want agent find a device and not be in GLPI inventory, FusionInventory create this device in unknown devices. A user will decide if this device must be imported in GLPI inventory (devices managed by company) or not.

For example: A computer can be found and it's a computer of someone outside of company but this computer is not computer of our company. We can manage it computer (to known where is connected, IP, MAC...) in unknown devices but we don't manage it into our GLPI inventory.

Many informations can be managed on these devices :

Informations of these devices

Name: Name of the component
Entity: Entity of the device
Type: GLPI type of device (computer, printer, network device...)
Alternate username : Name of the user uses this device
Location :
Domain:
Approuved devices : If this device is approuved, you know if this unknown devices is allowed to be in your network.
Serial number:
Network hub : use by fusinvsnmp plugin
Inventory number:

Comments :

Tab connections

Tab import

Used to import this unknown device into GLPI inventory. In fact, if this device have a type *Computer*, when we clic on import button, the plugin will create a computer with the informations and delete this unknown device. It will get the network port of unknown device to set to the computer to keep history and relations between network ports.

: Importation canwork only if the type is defined.

Tab FusInv XML

In this tab, you can see XML of all data get by agent (if creation has been made by agent).

Plugin for GLPI : fusinvinventory

Installation guide and users manual for the plugin for GLPI : fusinvinventory.

Description

Description of plugin for GLPI : fusinvinventory

The goal of this plugin is to get inventory (hardware, software) of computers and display it in GLPI.

Requirements

Requirements for using this plugin for GLPI : fusinvinventory

In order to install and use the plugin *fusinvinventory* on your server, you have to meet the following requirements:

GLPI 0.78.x (0.72 and older is not supported)
Plugin for GLPI : fusioninventory (core)

Features

Features list

This is features list :

Get many informations of computers :

Hardware informations (serial number, computer model, manufacturer, components like processor, memory, network card...)
Software informations (all softwares installed in the computer)
Antivirus informations (for Windows only)

Installation

Plugin installation

Procedure list of installation on different case.

New installation

How to install the plugin *fusinvinventory*

To install it, you must do :

Download file here :

http://forge.fusioninventory.org/projects/pluginfusinvinventory/files
Uncompress the file into the plugin folder of GLPI. File list seems to be like :

- glpi

--- plugins

-----fusinvinventory

-------index.php

------- hook.php

------- front

------- inc

etc...
Connect on GLPI on super-admin
Go into the menu *Configuration* > *Plugins*
Install *FusionInventory INVENTORY*.
Activate *FusionInventory INVENTORY*.

These operations will create tables in database MySQL and rights on the plugin.

: Link for install / activate this plugin is hidden if plugin *Plugin for GLPI : fusioninventory (core)* isn't installed and activated.

How it works

How to plugin *fusinvinventory* works?

This is a description of the processus to inventory computers.

[Agent] contact server (plugin *fusioninventory*).
[Plugin fusioninventory] create agent the first time (see *Agents management*).
[Plugin fusinvinventory] get information if this agent is allowed to execute local inventory module. If no, Agent return order with no inventory request. If yes, go on next step.
[Agent] get all informations of computer.
[Agent] send XML of inventory to plugin.
[Plugin fusioninventory] get XML file and send it to plugin *fusinvinventory*.
[Plugin fusinvinventory] receive XML file, parse it and remove fields that are blacklisted.
[Plugin fusinvinventory] Data are sent to criteria rules.
[Plugin fusinvinventory] Criteria rules check, and send to the *libserver fusioninventory*.
[Libserver fusioninventory] Lib create a file and send all data to plugin *fusinvinventory*. In case of computer is yet created, it send only modifications of inventory to plugin *fusinvinventory*.
[Plugin fusinvinventory] Get data, create computer, send to entity rules to get the right entity.
[Plugin fusinvinventory] All data of XML file will be created in GLPI (for some fields, before write in GLPI, dictionnaries are used to rewrite them).
Your computer is in GLPI !

: To make inventory, you don't need use task management they are bypassed).

Plugin for GLPI : fusinvsnmp

Installation guide and users manual for the plugin for GLPI : fusinvsnmp.

Description

Description of plugin for GLPI : fusinvsnmp

The goal of this plugin is to get remotly inventory of network equipment (switch, printers...) and to discovery all devices connected to the network.

Requirements

Requirements for using this plugin for GLPI : fusinvsnmp

In order to install and use the plugin *fusinvsnmp* on your server, you have to meet the following requirements:

GLPI 0.78.x (0.72 and older is not supported)
Plugin for GLPI : fusioninventory (core)

Features

Features list

This is features list :

Get many informations of switch:

Hardware informations (serial number, model, manufacturer, firmware, mac caddress...)
Ports informations (name, description, MTU, erors...)
Connections on each port (what mac is connected on this port)

Get many informations of printer:

Hardware informations (serial number, model, manufacturer, firmware, memory...)
Cartridges state (empty, full, percent of ink)
Page counter (counter registred all days and possibility to display graph of these datas, get counter total, color counter, black and white counter...)

Discovery all devices connected on the network

Scan by NMAP (if installed, get mac address and dns name)
Scan by NETBIOS (get netbios name, workgroup or domain, user connected)
Scan by SNMP (get serial number, mac address, name)

Installation

Plugin installation

Procedure list of installation on different case.

New installation

How to install the plugin *fusinvsnmp*

To install it, you must do :

Download file here :

http://forge.fusioninventory.org/projects/pluginfusinvsnmp/files
Uncompress the file into the plugin folder of GLPI. File list seems to be like :

- glpi

--- plugins

-----fusinvsnmp

-------index.php

------- hook.php

------- front

------- inc

etc...
Connect on GLPI on super-admin
Go into the menu *Configuration* > *Plugins*
Install *FusionInventory SNMP*.
Activate *FusionInventory SNMP*.

These operations will create tables in database MySQL and rights on the plugin.

: Link for install / activate this plugin is hidden if plugin *Plugin for GLPI : fusioninventory (core)* isn't installed and activated.

SNMP models

SNMP models

The SNMP models are maintened and generated by FusionInventory team.

They are usefull to get right informations in the right SNMP oid because depend on device, manufacturer, device and sometimes firmware.

This list is updated for each release of plugin *fusinvsnmp*.

There are some things to know of the models :

If SNMP device have no model associated, we can create the model by sending us the snmpwalk. See *http://forge.fusioninventory.org/projects/fusioninventory/wiki/Sending_snmpwalk*
To see if your device have model, in stable or in devel version, you can see *http://forge.fusioninventory.org/projects/pluginfusinvsnmp/wiki/switch_list* (only for switch for the moment)

Add a SNMP model

There are many possibilities to add a model when clic on + button:

Import a model with a file generated by FusionInventory team.
Import all models (mass importation) in folder *plugins/fusinvsnmp/models/*.
Create a model yourself (not very recommanded).

Content of a model

A model is defined by :

Name : Name of the model (number randomly generated)
Type : Type of devices designed to work (network equipment, printers...
Comments : List of SNMP *sysdescr* compatible with this model
List of oids : List of the different oids associated to a GLPI information

SNMP authentication

SNMP authentication

This is the list of authentication for permit to read SNMP of the devices.

FusionInventory support SNMP v1, v2c and v3 (no crypt, half crypt and full crypt).

Authentication is defined by :

Name : Name of the authentication
SNMP version : Version of SNMP (v1, v2c or v3)

For versions 1 and 2c only :

Community : Used by versions 1 and 2c (it's like a password)

For version 3 only :

User : Name of user allowed to read SNMP used only for communication
Encryption protocol for authentication : Set encryption protocol (MD5 or SHA) used only for communication
Password : password used only for communication
Encryption protocol for data : set encryption protocol (DES, AES128, AES192 or AES256) used only to get SNMP data.
Password : password used only to get SNMP data.

IP ranges

Management of IP ranges

The range IP must be defined for network discovery and SNMP inventory by the agents.

Name : Name of the range
Start of IP range : First IP address of the range
End of IP range: Last IP address of the range
Entity : Associate this range IP to an entity

: When validate, a verification of IP is made and a message appear if there is a problem.

Tab Permanent task - discovery (SNMP)

This tab is used to create a netdiscovery task easily.

Clic on link *Create task easily* to create the task. After configuration is required. For configuration, see *Tasks management*.

In this tab, history and complete logs of last executions is displayed.

Tab Permanent task - inventory (SNMP)

This tab is used to create an inventory (SNMP) task easily. See section before *Tab Permanent task - discovery (SNMP).*

Network ports

Port visualization of network components (switches, routers)

This page is available on from of a network equipment and go on tab *FusInv SNMP*.

SNMP Informations

Data seen here are :

Sysdescr : it's the sysdescr of the switch (SNMP)
SNMP models : it's the SNMP model associated with this switch. The button *Load the correct model* can be used to set the right model with help of the *sysdescr* and if is present in the SNMP definitions (*SNMP models*).
SNMP authentication : Authentication used to access query SNMP oid.
Last inventory : date of last inventory of this switch
CPU usage (in %) : % of CPU when have made last inventory
Memory usage (in %) : % of memory used (in most case of switch this value is the storage of operating system (firmware)
Uptime : Uptime since last reboot of switch

IP

A switch can have many IP. This array display all of these IP addresses.

Ports array

This is list of columns / informations of each port :

Name: Shops the name of the network port
MTU: Shows the maximum packet size traversing this port
Speed: Show the speed the port is operating at

: The speed is also shown at 10 Mbps when there is no device connected (and power up) to this port.
Internal status : Display with a colored dot if the port is active or not
Last change : Display time since last modification of the port (not works very well on some manufacturer switch)
Number of bytes received : Display number of bytes the port have received
Number of input errors : Display number of errors when port received data. If more than 0, this case have red background color to identify quickly
Number of bytes sent : Display number of bytes port have sent
Number of errors in reception : Display number of errors when port send data. If more than 0, this case have red background color to identify quickly
Duplex : Display if the port is in half-duplex or full duplex
Internal MAC : Display MAC address of the port (can be different than switch MAC)
VLAN: Show the VLAN of this port
Connected to : Display on witch GLPI device this port is connected
Connection : Display with colored dot if a device is connected and power on

: A device can be connected but power off (so colored dot is red)
Description du port : Display description of the port

There are many colors line (Background color) :

Grey: This port has no connection with any device
Orange with the wording "Trunk port": A connected port that is connected with a switch or server using port trunking / link aggregation (or is tagged)
Orange with the wording "Connected port": A port connected to equipment without usage of trunks.

The history of each port can be visualized by clicking on the + which gen be found in the first cell. An array will display all modifications of informations and all connections and disconnections of devices.

Cartridge state of printers

See cartridge state of printers

TODO

Printer page counter

Visualization of page counters from printers

TODO

Exemple de mise en place

Exemple de mise en place de FusionInventory

Nous souhaitons :

- Faire une découverte de nos switchs
- Faire un inventaire de nos switchs découverts

Voici les données utilisées dans cet exemple :

- Plage IP de notre réseau : *192.168.0.1* à *192.168.0.254*
- Serveur GLPI : *192.168.0.10*
- GLPI est accessible via l'url http://192.168.0.10/glpi/
- L'agent sera installé sur le serveur 192.168.0.11 (mais il pourrait être installé sur le même serveur que le serveur GLPI)
- Communauté utilisée : versions 1 & 2c "public"

Configuration

Configuration

Il faut commencer à configurer le plugin FusionInventory pour GLPI.

Dans la *configuration du plugin*, nous allons paramétrer les *critères d'existence* suivants :

Critères 1 :

IP : Non
Nom : Non
Numéro de série : Oui
Adresse Mac : Non

Critères 2 :

IP : Non
Nom : Non
Numéro de série : Non
Adresse Mac : Oui

Dans les *modules* nous allons activer uniquement :

Snmp
Découverte d'équipements

Configuration de l'agent

Configuration de l'agent

Il faut *installer l'agent* sur le serveur 192.168.0.11

Il faut configurer l'agent avec cette variable :

server=http://192.168.0.10/glpi/plugins/fusioninventory/front/plugin_fusioninventory.communication.php

Lancer l'agent une fois.

Configuration de l'agent dans GLPI

Configuration de l'agent dans GLPI

L'agent a été créé dans GLPI.

Il faut activer les modules nécessaires pour faire la découverte ainsi que l'inventaire réseau.

Dans *Plugins* > *FusionInventory* > *Gestion des agents*, cliquer sur le nom de l' agent et mettre à *Oui* les champs suivants :

Découverte d'équipements
Interrogation SNMP

Valider les modifications.

Configuration d'une plage IP

Configuration d'une plage IP

Il faut maintenant *créer notre plage IP* avec les paramètres suivants :

Nom : réseau interne
Début de la plage IP : 192.168.0.1
Fin de la plage IP : 192.168.0.254
Agent découverte : L'agent qui a dû se créer (on ne doit en avoir qu'un seul dans la liste)
Agent interrogation : L'agent qui a dû se créer (on ne doit en avoir qu'un seul dans la liste) IP

Découverte : Oui
Interrogation : Non

Découverte de matériels

Découverte de matériel

Nous pouvons désormais découvrir nos matériels réseau.

On lance l'agent une nouvelle fois pour qu'il découvre les matériels.

Une fois que l'agent a terminé, on passe au chapitre suivant.

Gestion du matériel inconnu découvert

Gestion du matériel inconnu découvert

Dans ce *menu*, on va gérer les matériels que l'agent a découvert.

On va importer les switchs découverts.

Inventaire du matériel réseau

Inventaire du matériel réseau

Pour inventorier nos switch, il faut modifier notre *configuration de plages IP* pour les champs :

Découverte : Non
Interrogation : Oui

On relance l'agent et quand il a fini, toutes les données sont dans la fiche de chaque switch.

Appendix
A

Glossary

Topics:

Glossary

[Work in progress]

Addresse MAC

Addresse MAC

[En cours d'écriture]