G/On Server Manual

G/On 5.6

Document revision 2.1

2013-09-24

About this document

The document provides the basic information for setting up G/On, as well as in-depth reference to the configuration and management of the G/On servers.

G/On™

✖ Giritech A/S, 2013
Spotorno Allé 12, 2.
2630 Taastrup
Denmark
Phone +45 70.277.262

Contents

Configuration Reference

Management Reference

Extending G/On to support other applications

Setting up G/On

Introduction

A full functioning installation consists of the following steps:

1. Preparation of the installation
2. Installing the G/On software on the G/On server
3. Configuration of the two G/On service modules, G/On Gateway and G/On Management
4. Setting up basic policies and adding users and authentication tokens

This chapter describes all of these steps in turn.

Prepare installation

Supported Platforms

G/On Client

- Windows XP (32 bit), Windows Vista, Windows 7
- Apple Mac OS X 10.6 (Snow leopard) and 10.7 (Lion) on Intel based Macs
- Linux Fedora 17 with GTK+ GUI (32 bit)

G/On Management

- Windows XP (32 bit), Windows Vista, Windows 7
- Windows Server 2003 R2, Windows Server 2008, Windows Server 2008R2

G/On Server

- Windows Server 2003 R2, Windows Server 2008, Windows Server 2008R2

All platforms mentioned, have been tested with the latest Service Packs/updates at the time of the release of the this G/On version.

Software Dependencies

G/On Setup and Configuration requires Java run time in a version (32/64 bit) that is the same as the OS.

G/On Management also requires Java run time in a version (32/64 bit) that is the same as the OS. When installing G/On Management as a package on a token, it is possible also to install a package containing the Java Run time, on the token. When this has been done, G/On Management can be run from the token, no matter whether the PC has the correct JRE installed or not.

Client Installer. In order to be able to generate a Client Installer for field deployment, and a Gateway installer for separate Gateway installation, the Nullsoft scriptable install system must be installed on the G/On server. Get it here: http://nsis.sourceforge.net/

G/On Management requires Internet Explorer 8 or newer for some functions (reporting).

Java Runtime Environment

The G/On 5 Server Configuration and G/On 5 Management requires that Java Runtime Environment (JRE) is installed. Get it here: http://www.java.com/

Note: The Java run time must be in a version (32/64 bit) that is the same as the OS.

User Directory

G/On can connect to three types of user directories; Active Directory, a LDAP user directory or the local Windows users on the server machine. In order for Active Directory integration to work properly the installation must be done on a computer which is either on the Active Directory domain or on a domain with which a trust relationship has been established. See the Configuration Reference starting on page 50 for more details on how to choose and set up user directory.

Preparations

Before starting the installation please consider collecting the following information:

Which IP address and TCP port should the clients connect to, when connecting from outside the firewall? Symbolic/DNS name for the IP can also be used. The IP address is part of the G/On License File, and must be specified when ordering G/On. Please note, that IP address and port number(s) must be specified at ordering time – and is part of the license agreement.	IP/DNS address and Port that the G/On client should connect to. The default port number is 443, even though 3945 is the official IANA allocated port-number for G/On. If possible, port 80 – or 443 are normally good choices, as these ports are open outbound in most environments. So by selecting these ports, it is even more likely, that G/On clients will be able to connect to the G/On server

Which user directory should be used to authenticate users.	Connect to Active Directory through native Windows API or to eDirectory or Active Directory through LDAP

Installation

The installation is done using the installer, which at the end will start the G/On Configuration program. G/On Configuration includes all the basic, technical setup of server IP addresses etc.

Server Installation

Please download the latest installer from Giritech's website at

http://www.giritech.com/int/Support-Download/Product-Download

Store it on the server where you want to install G/On.

Installation is done in two steps: actual installation of the G/On Server software modules – and subsequently the initial configuration of the G/On main components.

The installation requires about 400MB free disk space.

Once the installation is completed, you can continue directly with the initial configuration (recommended). Once you press finish – and if you have checked the *Run G/On Configuration Wizard* – allow some time for the wizard to start. Alternatively, you can find the G/On Configuration program in the Windows Start Menu.

Note: On Windows Server 2008, you must run the G/On Configuration program, as Administrator: Find the program in the Windows Start Menu, right-click on it, and choose: “Run as Administrator”.

Configuration

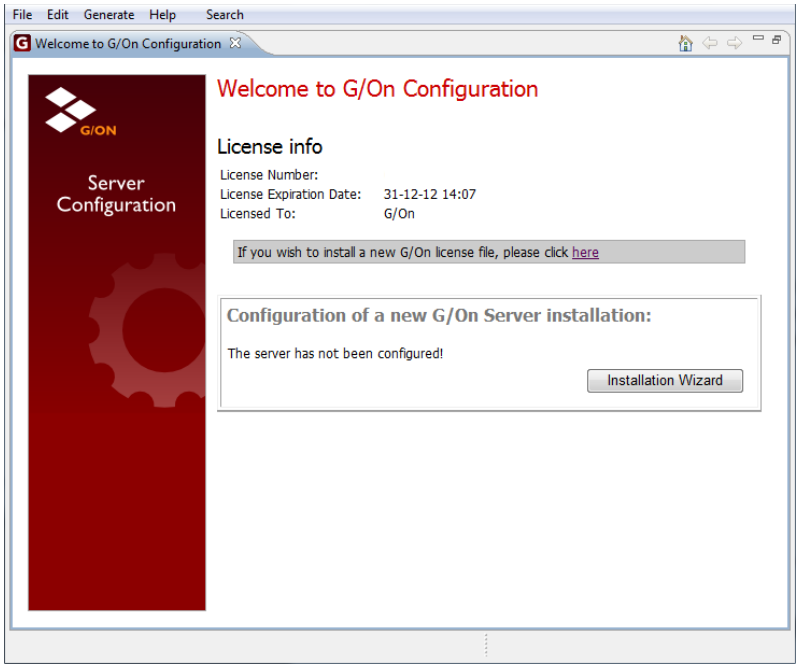
Initial server configuration

Initial Server Configuration is done through the configuration wizard. Before configuring the G/On server, you may wish to obtain a proper G/On license file. This file will determine the number of G/On users possible, as well as specifications regarding purchased options etc.

If you do not use a proper G/On license file, the installation will proceed with a so-called demo-license. This will allow you to test G/On, but not all options.

License Handling

All G/On installation options are specified in the license file. The license file is obtained as part of the purchase process for G/On. If you do need a proper G/On license file, please contact your Giritech Partner.

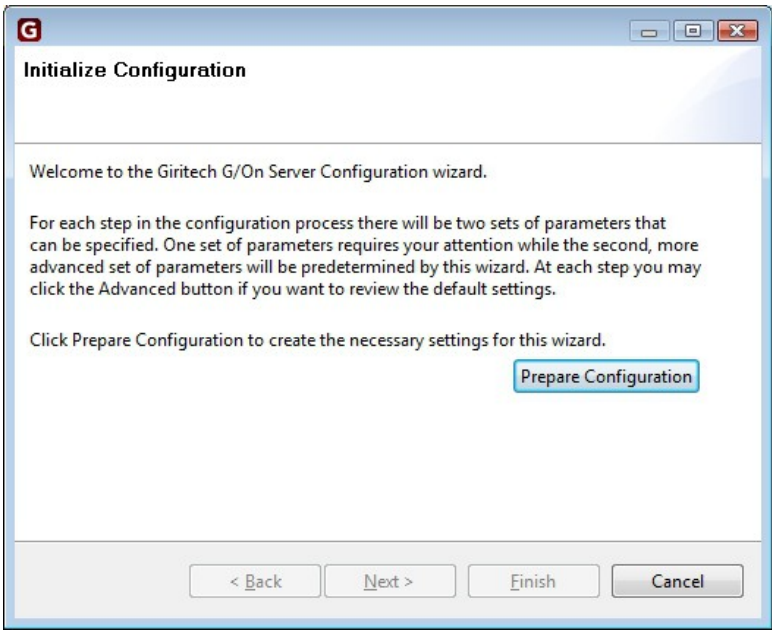


In the welcome screen to the right, you can see the current status of your license file:

If you have received a proper license file, you can install by clicking on the link in the welcome screen.

Using the Installation Wizard

Press the Installation Wizard button on the configuration welcome screen. You can also start the Wizard at a later time, from the Windows G-On program menu: ”G-On Configuration”. Once started the following instruction screen will be shown:



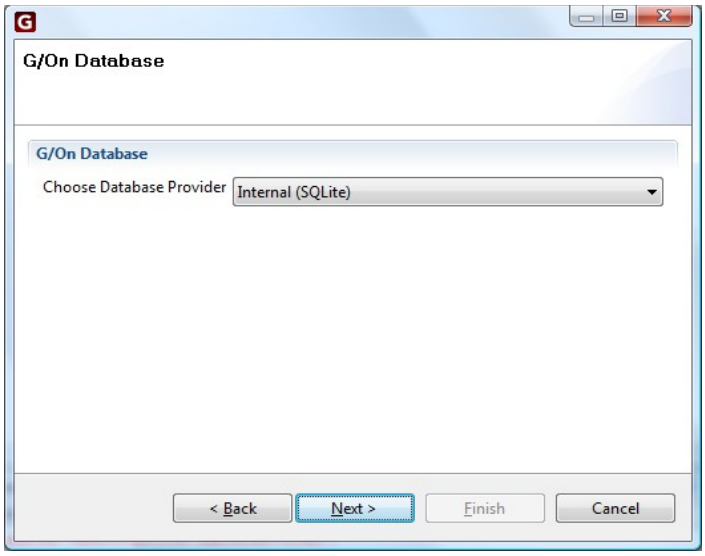
Please read, and continue by clicking Prepare Configuration - and the Configuration Wizard will generate an initial set of configuration data. Once done, click Next on the Initialize Configuration screen.

A G/On installation will normally have at least two services running:

1. **The G/On Management Server** – allowing Management of the solution (users, authentication and authorization policies)
2. **The G/On Gateway Server** – performing the actual tasks of connecting users with applications according to the policies specified.

Each server/service is configured separately as described below.

Database Configuration

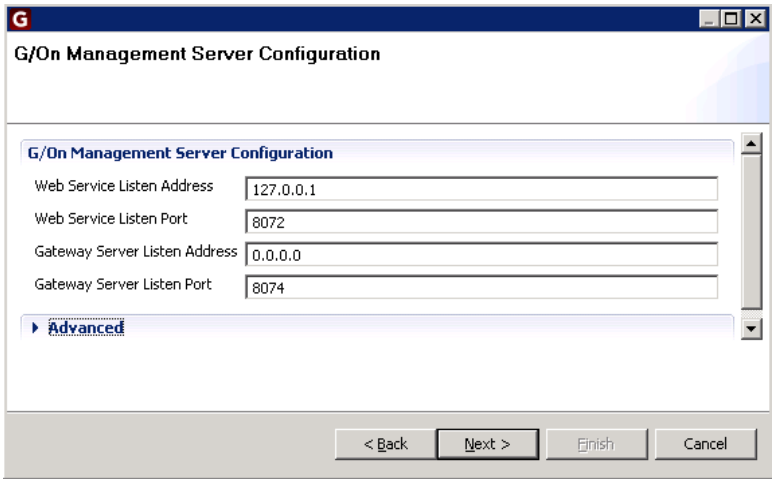


Choose the default Internal database and click Next.

You must select database provider. If the setup will have only a single Gateway server, you can use the default Internal (SQLite) database. This requires no further configuration.

For a setup that can handle multiple Gateway servers, select Microsoft SQL Server or MySQL. If you choose either of these, a new window for entering further configuration will open when you click Next. See *Database setup* on page 57 for more information.

Management Server Configuration



Web Service Listen Address: IP address that the Management Server should listen on. Default is 127.0.0.1 – which means that the G/On Management Server will only allow connections from the machine the Management server is running on.

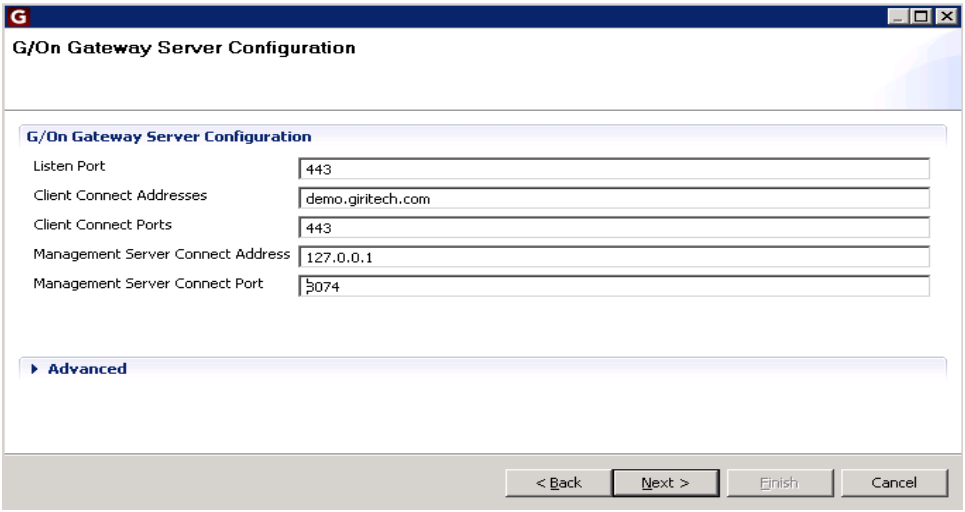
Web Service Listen Port: To enable the G/On Management Server, a port must be designated (default is 8072). This port is used for the G/On Management client to connect to the management Server.

Gateway Server Listen Address: IP address where the Management service should listen for connections from Gateway Servers.

Gateway Server Listen Port: TCP Port where the Management service should listen for connections from the Gateway Servers.

Note: In the default set-up, the Management Client must run on the same machine as the Management Server, in order to be able to connect to it. If you want to run the Management Client on a remote PC, define a G/On menu action for this purpose and run G/On Management through a G/On connection.

Gateway Server Configuration



Listen Port: The port that the Gateway Server listens on in order to accept connections from G/On Clients.

Client Connect Addresses: This is the IP address (DNS name or number), that the G/On clients will use to connect to the G/On server. Please note, that when using a proper license, this address is fixed, and must be determined at the time of ordering G/On, as the connection address is part of the license (file). If using the demo license, any address can be specified.

Client Connect Ports: The default port number is 443, even though 3945 is the official IANA allocated port-number for G/On. If possible, port 80 – or 443 are normally good choices, as these

ports are open outbound in most environments. The port must be specified at the time of ordering G/On, and is part of the license (file). If using the demo license, any port can be specified.

Management Server Connect Address: IP address or DNS name, which the Gateway servers should use for connecting to the management server.

Management Server Connect Port: TCP port number, which the Gateway servers should use for connecting to the management server.

Note: Remember to set up your firewall/router so it accepts connections on these ports and forwards the connections to the Listen port specified for the Gateway server (see above).

HTTP Encapsulation

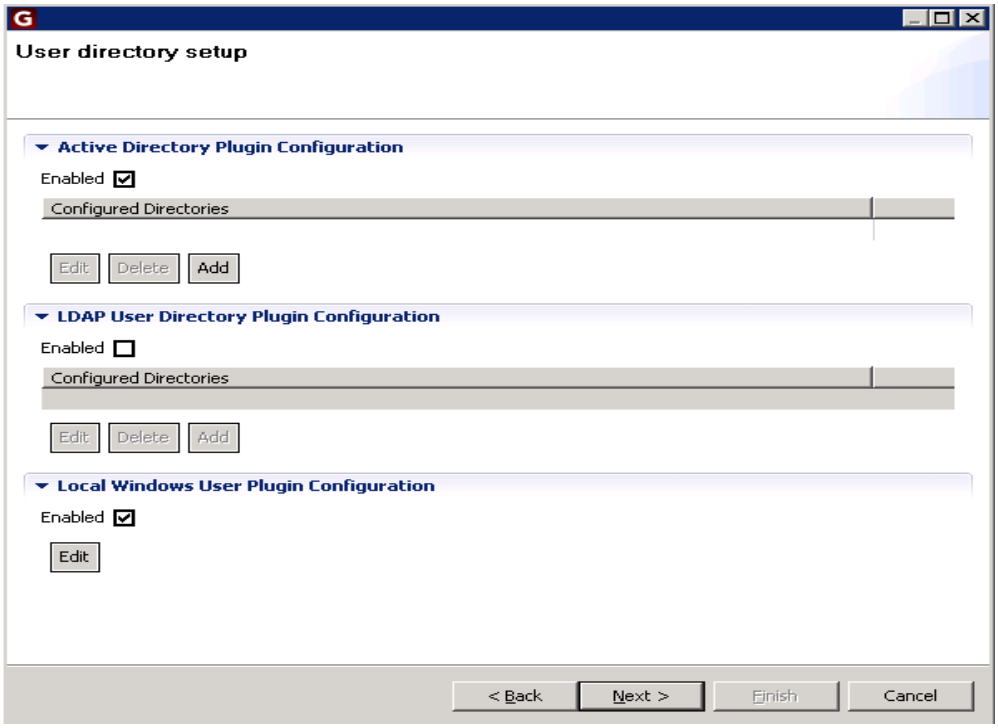
In some environments, a deep packet inspection firewall or other might block the communication between G/On clients and the G/On server. To allow connectivity in these situations, the **HTTP Encapsulation** option should be specified when ordering G/On.

If the feature is not included in your license, or you do not wish to configure it, you can skip this step.

For information on configuring this option, see the Configuration Reference chapter on page 50.

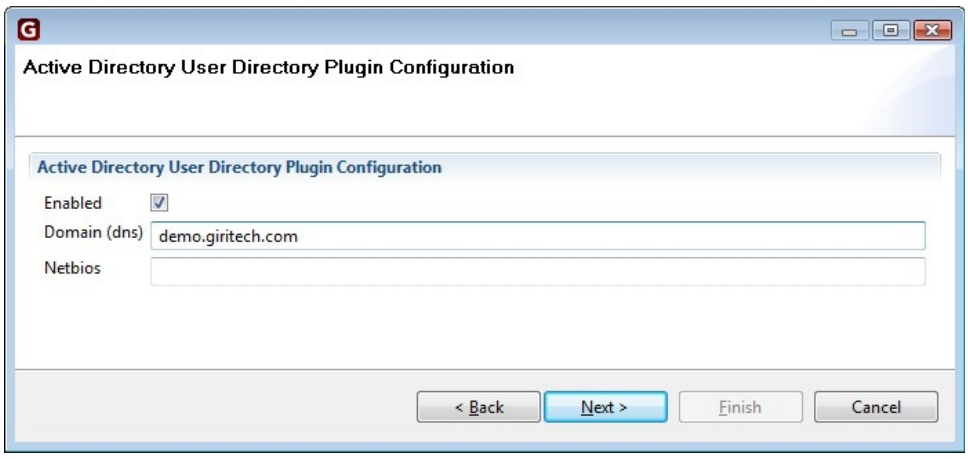
Directory Services Configuration

Identity management in G/On is through a directory service like Microsoft Active Directory (AD) – and/or LDAP (Lightweight Directory Access Protocol) – and/or Local Windows Users and groups on the G/On server machine.



Each user directory type can be enabled/disabled using the “Enabled” check boxes. For Active Directory and LDAP it is possible to add any number of different directory specifications, whereas there can be only one instance of the localy Windows user plugin. Choosing the “Add” or “Edit” buttons will open a new window with specifications for the user directory (type) in question.

Active Directory User Directory Plugin Configuration



On the Active Directory configuration screen, you must specify the **Domain (dns)** name of the Active Directory. Normally, the **Netbios** name is automatically filled in. If this does not happen, please fill in the Netbios name manually.

If the AD feature is not included in your license, or you do not wish to configure it, you can skip this step.

LDAP User Directory Plugin Configuration

As an alternative to Active Directory, an LDAP enabled user directory can be used. For information on configuring this option, see the G/On Set-up and Configuration Reference.

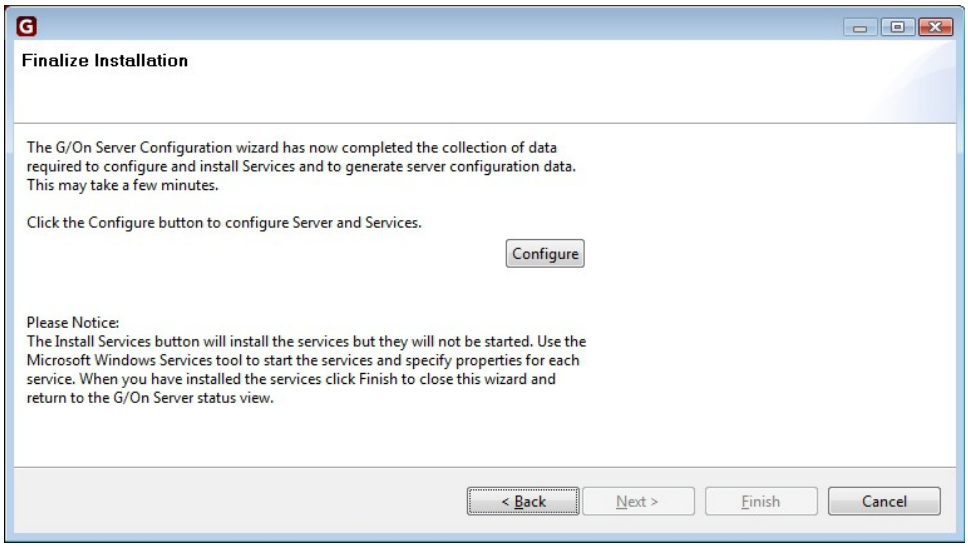
If the feature is not included in your license, or you do not wish to configure it, you can skip this step.

Local Windows User Plugin Configuration

As an alternative (or supplement) to Active Directory, G/On may be configured to use local users and groups that exist on the server machine where the G/On Gateway and Management Servers are running. If you do not wish to configure this feature, you can skip this step.

Finalize Configuration

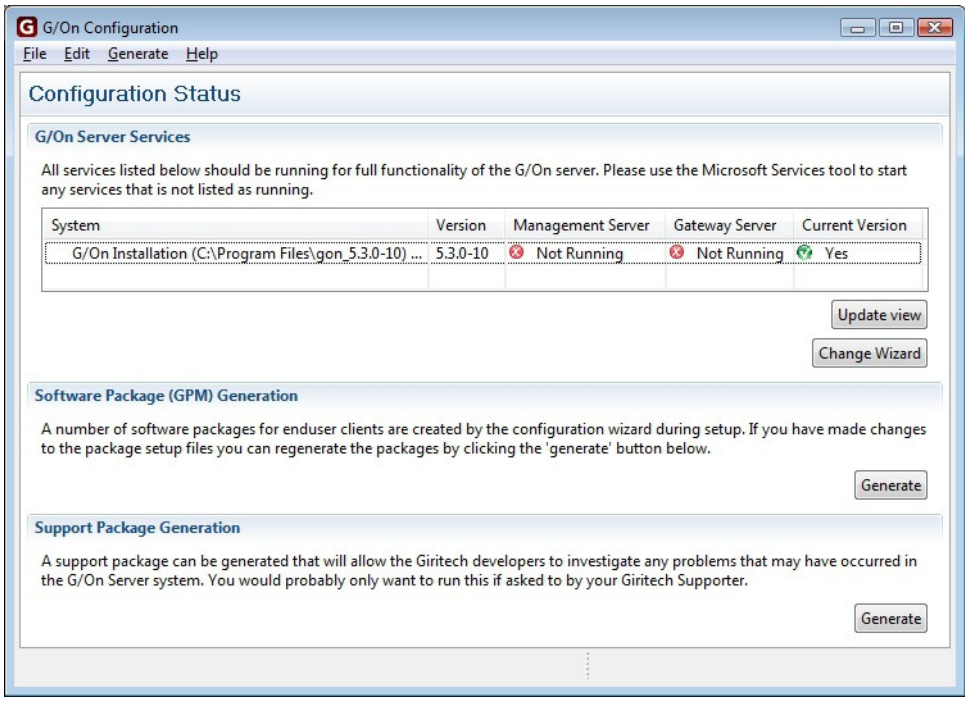
Last step in the initial configuration process of the G/On server is to configure the Management and Gateway services on the G/On server. Also, the system will use the supplied configuration data to generate the initial G/On Client Software Packages.



Click Configure to start configuration and services and generation of G/On Client Software packages.

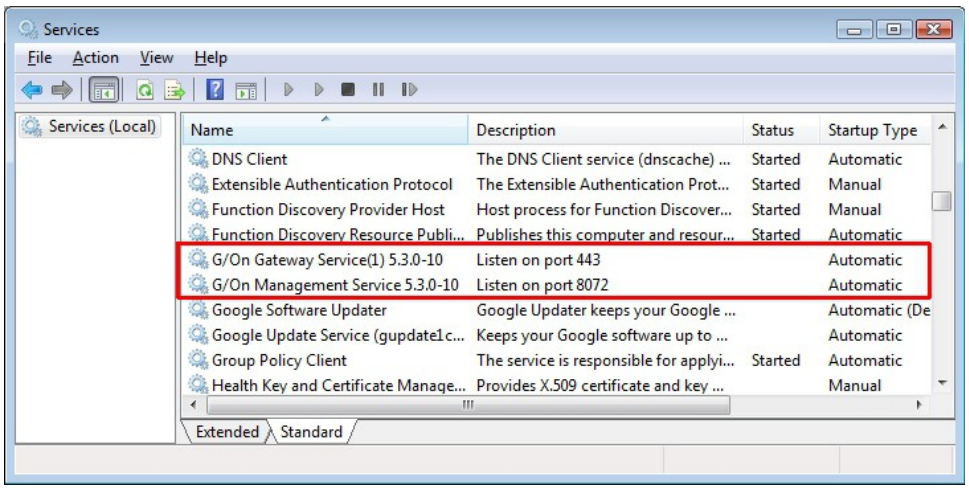
Once the configuration routine has finished, click Finish to go to the Configuration Status screen.

Configuration Status



The Configuration Status screen allows you to see the current status of the G/On Services. Note that the Management and Gateway services need to be started manually (first time).

You can start the G/On services by using the “Services” Management interface in windows. (All Programs > Administrative Tools > Services) – to see the following screen:



Locate the two installed G/On services (Management and Gateway) highlighted above – and right-click on each one in turn, and choose Start.

After starting the G/On services and clicking Update view on the G/On configuration status screen, the services will be listed as “running”.

Setting up G/On Management

This chapter describes a very basic setup using the G/On Management application, and will guide through setting up basic access policies. For more advanced setup, please read this chapter first, and then afterward proceed to *The Management Client* on page 115 for more information.

The G/On server uses a Rule Engine to decide who gets access to what. The G/On management application is primarily used for creating Rules for that engine. Each Rule has a number of premises and one conclusion. A Rules states, that if all premises hold, then the conclusion also holds. For instance, a Rule could say:

If the Token: micro_smart_0002 is being used, and the User is bob@giritech.com, then we conclude that in the current session, there is a known User with a Personal Token

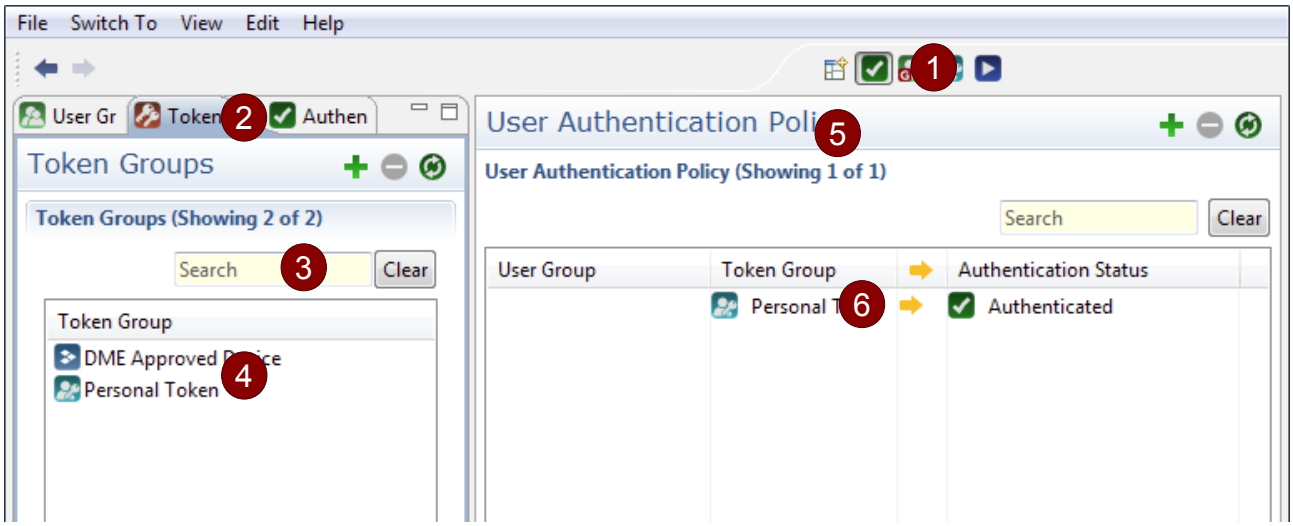
When you first open the G/On Management application, you will probably not have any Rules in the Rule Views. This guide will help you set up basic Rules for User access to the application of your choice.

Assumptions: In the following, *it is assumed that the G/On server is installed on a physical server machine, with a USB port*, which can be used for enrolling and deploying software to the first Token(s). **Note:** For demo/test installations, you may install on a Windows desktop OS (XP, Vista or 7). This usually works fine, even though it is not supported for production use. The only exception is the port scanning feature which does not work properly on the desktop operating systems.

More advanced topics, such as installing on a virtual server, are covered in the *Management Reference.*

Note: *On Windows Server 2008, Windows Vista and Windows 7, you must run the G/On Management program, as Administrator:*
Find the program in the Windows Start Menu, right-click it, and choose Run as Administrator.

Overview of the G/On Management Client

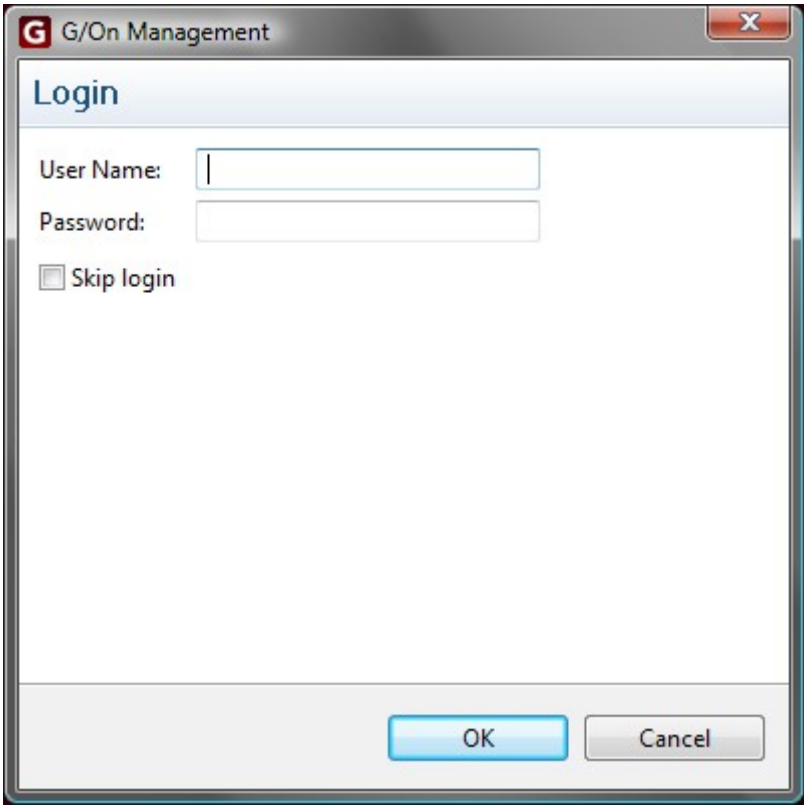


PARTS OF THE MANAGEMENT CLIENT

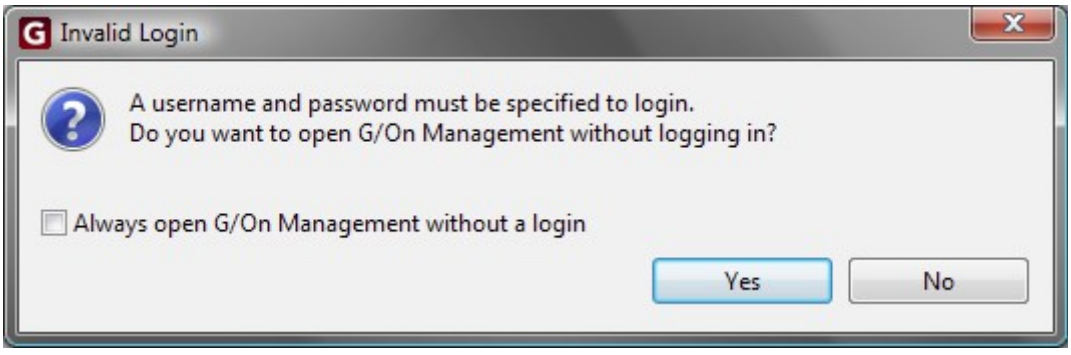
1. The **Perspective bar** is used for selecting between focus areas.
2. The **Element tabs** gives you access to the Elements that can be used in the perspective you are currently viewing. Every Element tab has a plus sign (+) in the top right, that will allow you to create new Elements. In the Element list you can see a list of all available Elements of a given type.
3. Search for specific elements.
4. Right-click in the view to get a **context menu** that will let you add/remove and edit Elements.
5. The **Rule list** shows all Rules in this perspective. In the top right corner of this view you will also find a plus sign (+) that will allow you to create new Rules.
6. You can right-click on existing Rules to get a **context menu** that will let you **add/remove** and **edit** Rules.

Starting G/On Management

When you start G/On Management it will connect to the management service and a login screen will be presented to you:



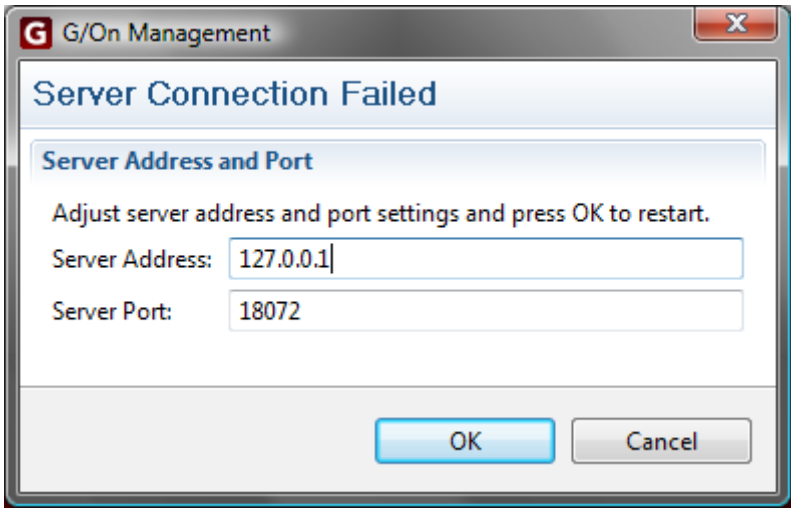
In the initial setup it is not necessary to log in, so you may either click the “Skip login” box and click “Ok” or just click Ok. In the latter case you will get a confirmation dialog:



Just choose “Yes” here. Access control can be added to G/On management using the Management Role Assignment view (see page 170). If you don't want to use access control, you can change the preferences so that the login screen is not shown at start-up (or by clicking the

Always open G/On Management without a login box in the message box above). See page 116 for further details.

In case a connection to the management service can not be established at start-up, you will be presented with a window, where you can edit the connection settings:



If the settings are wrong then change them and click “OK”. If the connection settings are ok, then you should check that the management service is running and, if it is, check the management service log file for possible errors.

Setting up Action Authorization Policies

Action Authorization Policies are used to give the Users access to specific Menu Actions when they have authenticated themselves. Click the Action Authorization Policy button in the Perspective bar to see the perspective. The perspective contains a number of Element tabs on the left, and a Rule list on the right.

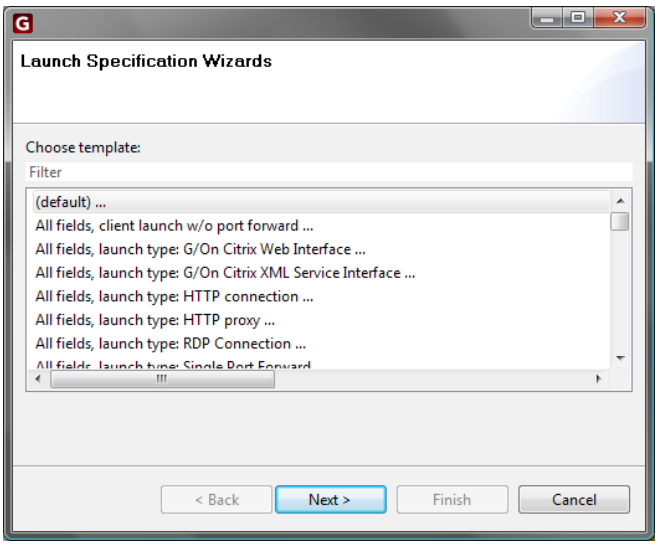
In this perspective you can use the Authentication Status, you specify in the Authentication Policy perspective.

You have the User Groups list to create Authorization Rules for separate groups of users. For instance managers may be authorized to use other applications than accountants.

The last tab contains the Menu Action list. These are the actions that will be listed in the Users menu if they are allowed to use them and if the computer they are using can handle it.

CREATE AN ACTION AUTHORIZATION POLICY

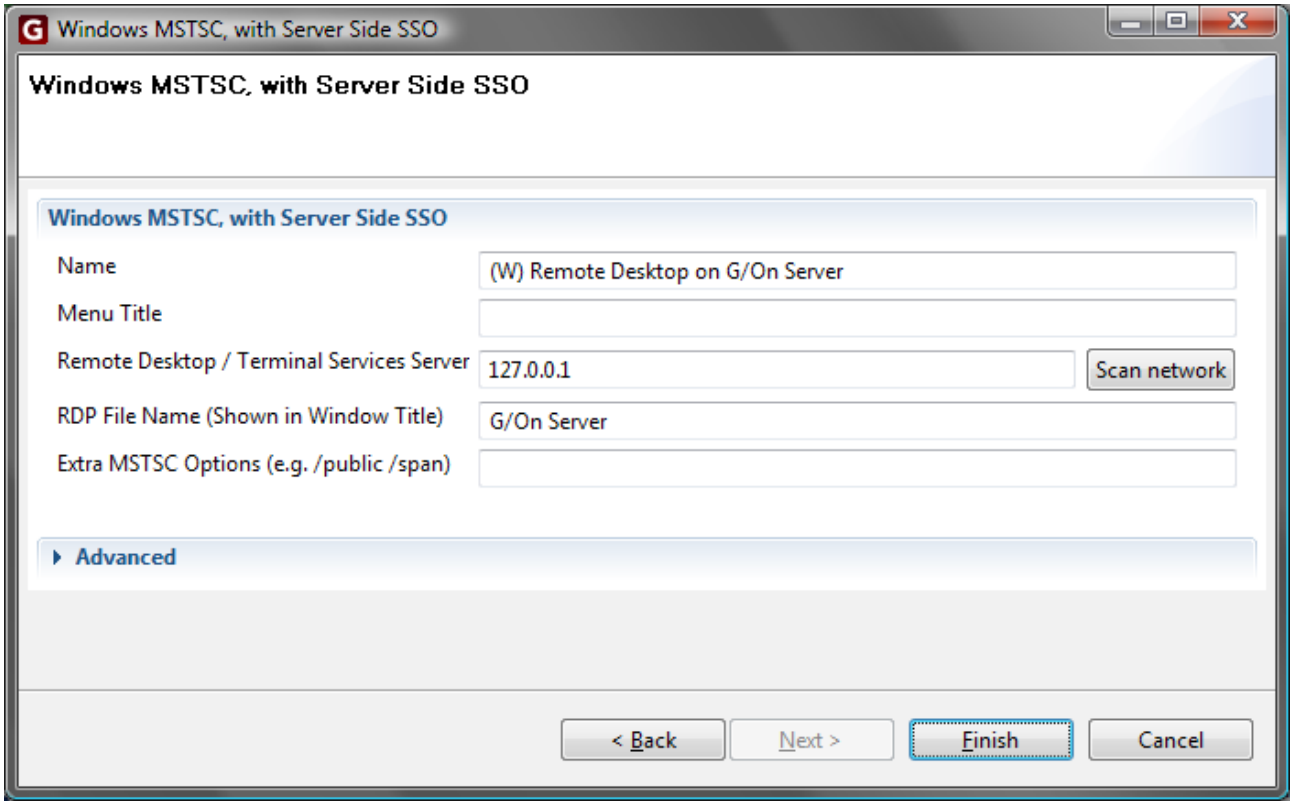
1. Open the Authorization Policy perspective.
2. From the Authentication Status list select the 'Authenticated' element and drag it into the Rule listing.
3. From the User Group list, select a group of users, and drag the Element representing them into the Rule editor.
4. From the Menu Actions list, select a Menu Action and add it to the Rule. The Menu Action will be added to the users menus if policies and Rules permit it.
5. Click Save and Close to save the new Rule.



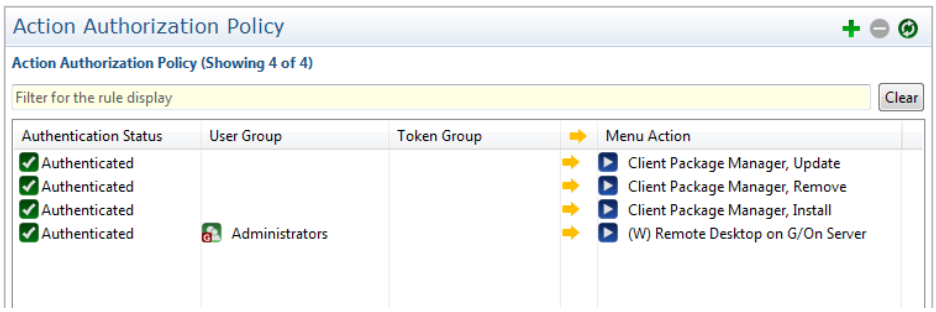
If a Menu Action for the action you want for your Users are not available already, you can use the Launch Specifications Wizard by clicking the green plus sign (+) in the Menu Actions tab. Just follow the instructions to create a new Menu Action and name it as you see fit.

To get started with something simple, we suggest to make a remote desktop

connection to a specific machine, e.g. the machine where the G/On server is installed. In order to enable this on Windows client PCs, use the template: “Windows MSTSC, with Server Side SSO ...”, and fill in the fields in the basic section:



In this example (in the first three Rules) a Menu Action provides access to the end-user package manager for installing, updating and removing packages, to any Users that are Authenticated.



The fourth Rule states that Authenticated Administrators are allowed access to the Remote Desktop connection.

You can create as many Authorization Rules as you like, to give access to various applications.

Changing Configuration

The configuration can be changed at a later point if needed. It is possible to either change the existing settings, or to reset them.

Changing the Server Configuration

Re-configuration of the system can be done by clicking Change Wizard in the Configuration Status Window. For details, see *Change Wizard* on page 75.

Restarting from Scratch

Cleaning and re-doing the configuration of the system, from scratch can be done by using the wizard in G/On Configuration. Start the G/On Server Configuration program. Choose Help > Welcome to the G/On Configuration. Click Start Wizard.

Warning: All configuration and management settings are reset.

Token Configuration and Enrollment

Introduction

Before any iOS device, G/On USB, or computer can be used with G/On, two actions must be performed:

- First, it must be **Configured** so it knows how to connect to the company G/On server
- Second, it must be **Enrolled**, so the company G/On server knows it and can allow it access.

G/On includes a feature called **Field Configuration and Enrollment** where users themselves can configure and enroll their iOS device, G/On USB, or computers. The administrator only needs to prepare the connection info and approve the requests for enrollment (G/On can even do this automatically), thereby minimizing the workload.

For G/On USB you can instead choose to use **Local Configuration and Enrollment**, described in the subsequent chapter. This method requires that you prepare every G/On USB yourself, so if you have many G/On USBs, use Field Configuration and Enrollment. But the advantage of the Local method is that when you hand over the G/On USB to the user, it is ready to use without the user having to do anything.

Hint: Outline of the instructions for each type of iOS device, USB, and computer can be found in *Instruction outlines* on page 48. You can use these as check lists.

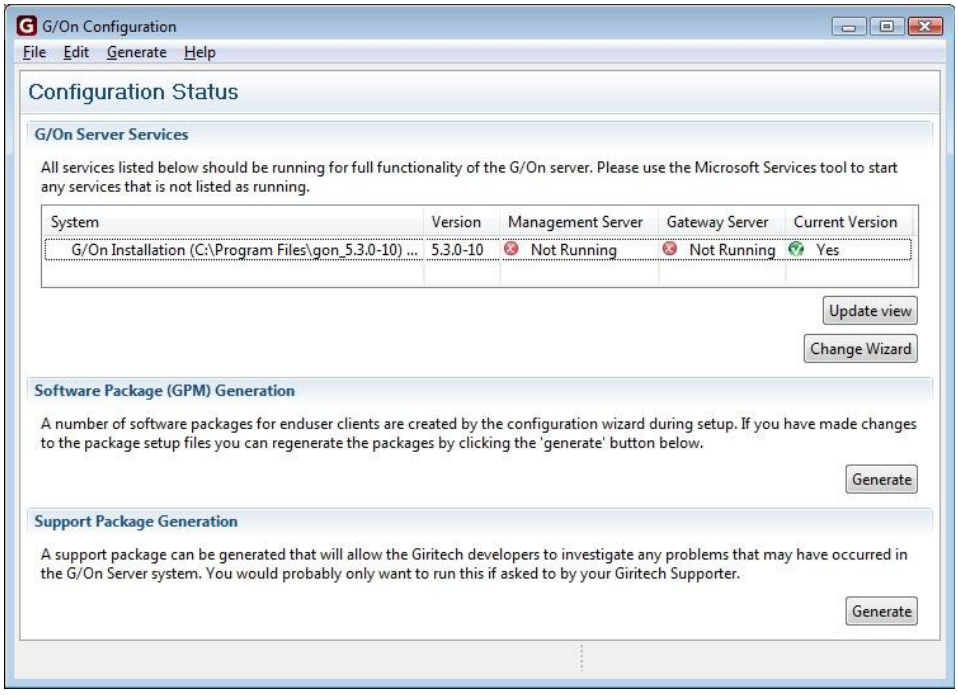
Important

Make sure that the setup of the G/On servers has been completed. See the *Setting up G/On* chapter for more information.

If you change the connect address or ports, make sure to regenerate the Software Packages and the Client Installer program. This will ensure that the updated Connection Info is included in the configuration.

You generate the Software Packages by

1. Open **G/On Configuration**



2. Click **Generate** under Software Package (GPM) Generation

You can see how to generate the Client Installation program in the chapter **G/On USB & Computer**.

Important: Always generate the Software Packages before you generate the Client Installation program.

Field Configuration

iOS

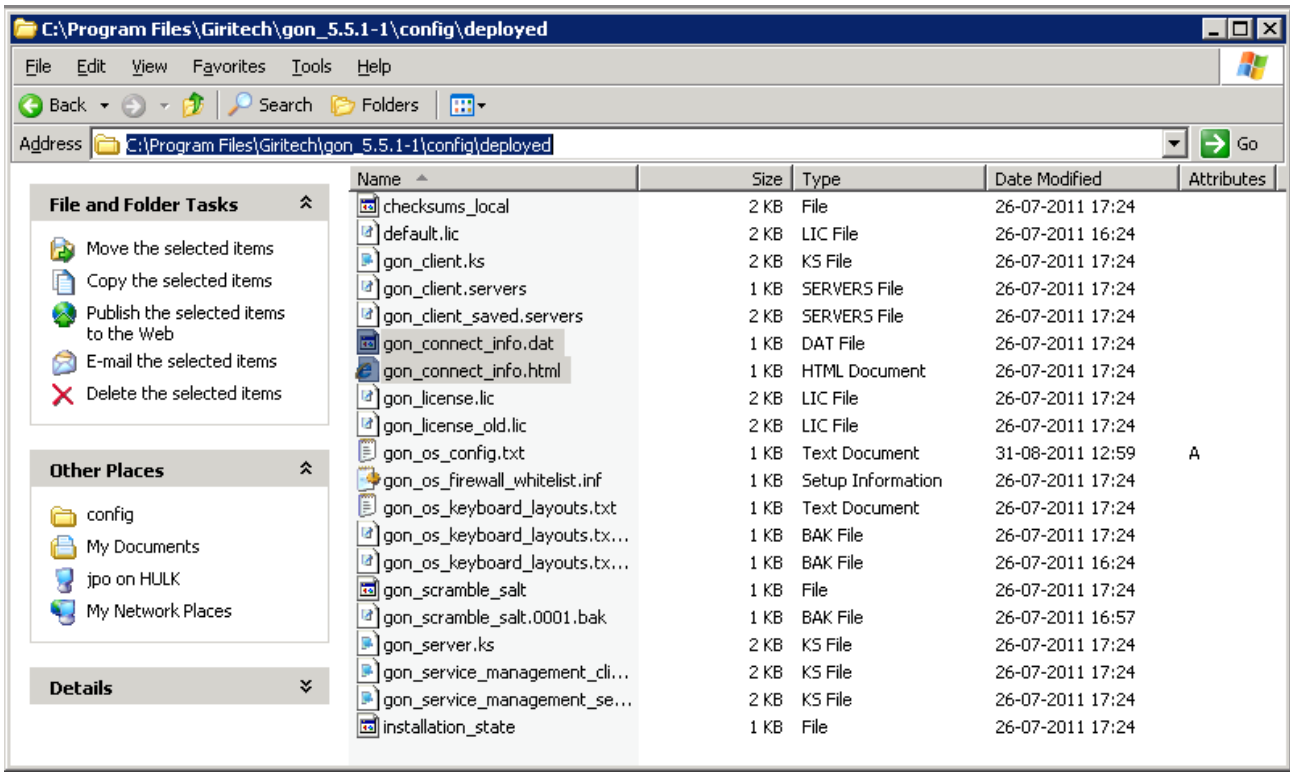
There are 3 ways to configure an iOS device to use G/On:

1. Opening a **Link** containing the G/On Connection Info
2. Scanning a 2-D barcode **Image** containing the G/On Connection Info
3. Importing a G/On Connection Info **File** through iTunes

You will only need to use of these options. You can read more on each option below.

When the G/On Server was initially installed and configured, two configuration files were generated. The files are located in

```
..\config\deployed\gon_connect_info.dat
..\config\deployed\gon_connect_info.html
```



Preparing for use of a G/On Connection Info Link

The IT administrator can make available a link like the following on a web page that can be reached from the built-in browser on the mobile device, e.g. at `http://my_server`

```
<a href="comgiritech://config/http://some_server/gon_connect_info.dat">G/On
Connection Info Link</a>
```

The url inside the above link (`http://some_server/gon_connect_indo.dat`) should also be reachable, and should point to the `gon_connect_info.dat` file generated during configuration.

Send the users a link to the webpage containing the Link.

You can also send the users the manual for their particular mobile device (iPhone or iPad) which can be found on the web page: www.giritech.com

Note: Microsoft's Internet Information Services servers (IIS) blocks files with the extension `.dat`

If the file is to be placed on an IIS server, rename the `.dat` extension to something else. Also remember to change the url in the Link accordingly.

Notes on using a Link

If you choose this option you will need to set up a homepage and a Link that the users can reach from their mobile device directly from the built-in browser.

Hint: If you do not want this, you can put the Link on the local network, making it only possible for the users to configure their mobile device while at work.

Preparing for use of a G/On Connection Info Image

The `gon_connect_info.html` file contains a QR code image (2-D barcode) with the G/On Connection Info needed by the users to configure their mobile devices.

This file can be distributed to the users through email, or made available for download from an intranet site or a file share.

You can also send the users the manual for their particular mobile device (iPhone or iPad) which can be found on the web page: www.giritech.com

Notes on using an Image

Low resolution cameras (iPhone 3GS) might not be able to scan the Image. This happens if the setup is very complex (e.g. many servers). You can check this yourself by opening the gon_connect_info.html file in a word processor. Count the characters in the line starting with `<img src=`

- If the number of characters are more than **700**, it can be difficult to scan with low resolution cameras
- If the number of characters are more than **900**, it will probably be impossible to scan the image with low resolution cameras.
- If the number of characters are more than **1200**, please open the gon_connection_info.html file in a browser. If you cannot see the image, the setup is too complex for an Image. Please choose one of the other options for distributing connection info.

Note: iPad 1 does not have a camera, and can therefore not be configured using an Image.

Preparing for use of a G/On Connection Info File

The `gon_connect_info.dat` file contains the G/On Connection Info, and can be distributed to the users through email, or made available for download from an intranet site or a file share.

You can also send the users the manual for their particular mobile device (iPhone or iPad) which can be found on the web page: www.giritech.com

Notes on using a File

It is recommended that you only use a File if it is not practically possible to use either a Link or an Image, as this method is not as easy for the users as the other options.

G/On USB & Computer

Note: Before you start, you need to install the Nullsoft scriptable install system on the G/On server. You can download it from here:

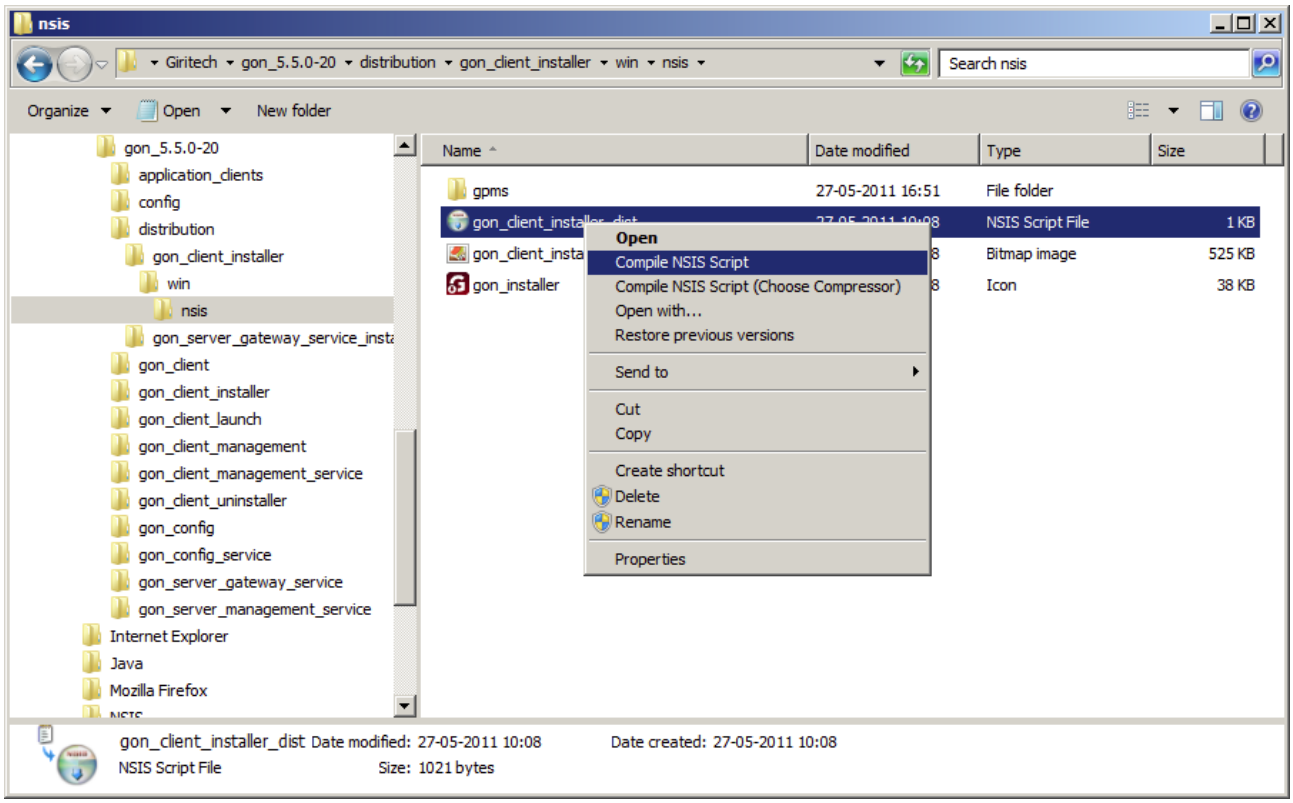
http://nsis.sourceforge.net/Download

Use the Nullsoft installer (NSIS) to generate the G/On Client Installation program, as follows:

On Windows Server 2003, do not start the NSIS program. Simply right-click on:

```
distribution\gon_client_installer\win\nsis\gon_client_installer_dist.nsi
```

and select Compile NSIS Script.



On Windows Server 2008, start the NSIS program with Run as administrator. Then choose Compile NSI scripts and File > Load Script... and then specify:

```
distribution\gon_client_installer\win\nsis\gon_client_installer_dist.nsi
```

The resulting Client Installer program is placed here:

```
distribution\gon_client_installer\win\G-On Client Installer.exe
```

Make the G/On Client Installer available for the users, e.g. from an intranet site or a file share.

You can also send the users the manual for either the Computer User Token or the G/On USB. This manual will show them how to install and use G/On, and can be found on the web page: www.giritech.com

Note: Computer User Token can only be installed on a Windows computer.

Note: The G/On USB can only be installed using a Windows computer.

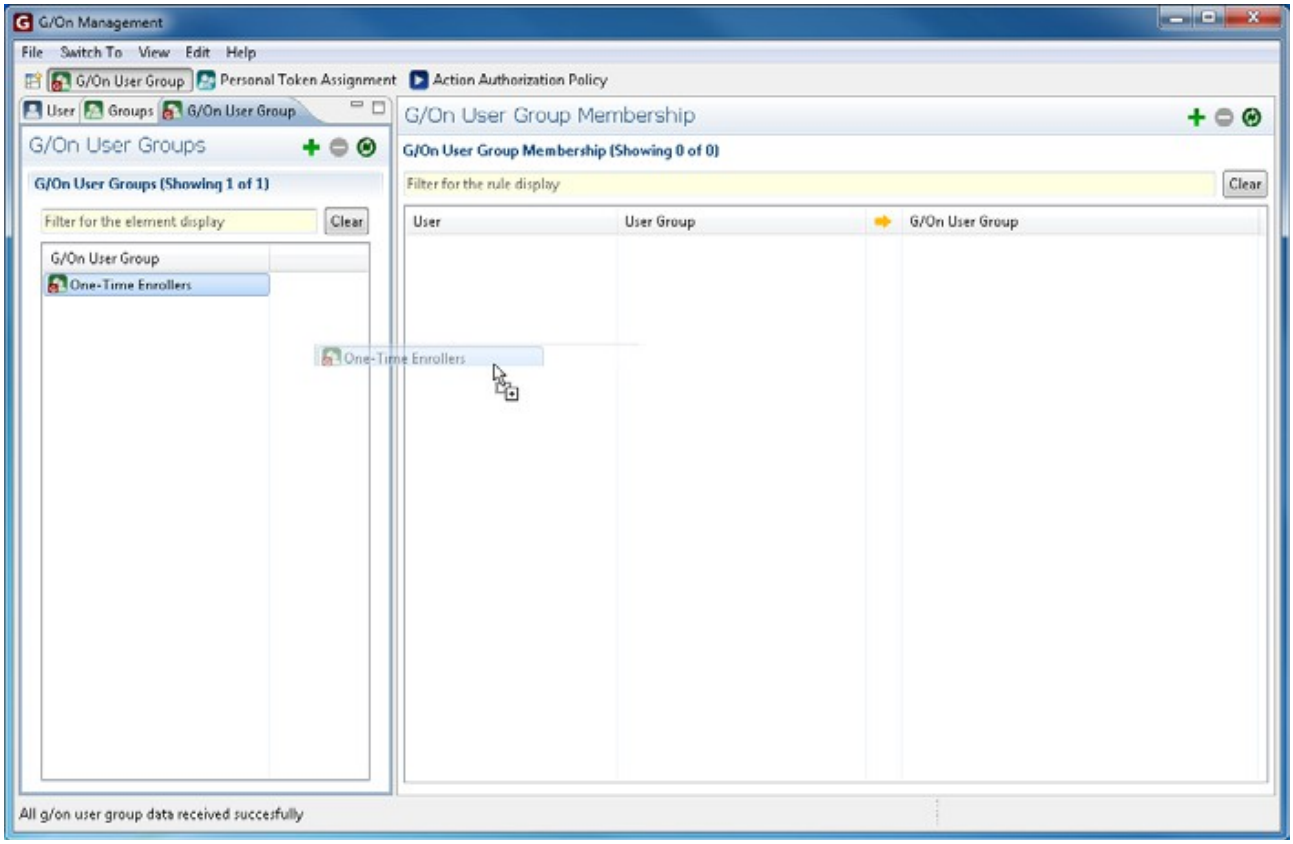
Note: It is possible to make two variants of the installation program: One will only offer to install on G/On Computer User Token, the other will only offer to install on a G/On USB. See the chapter *Creating Custom Client Installers* on page 94 for more information on how to make these variants.

Field Enrollment

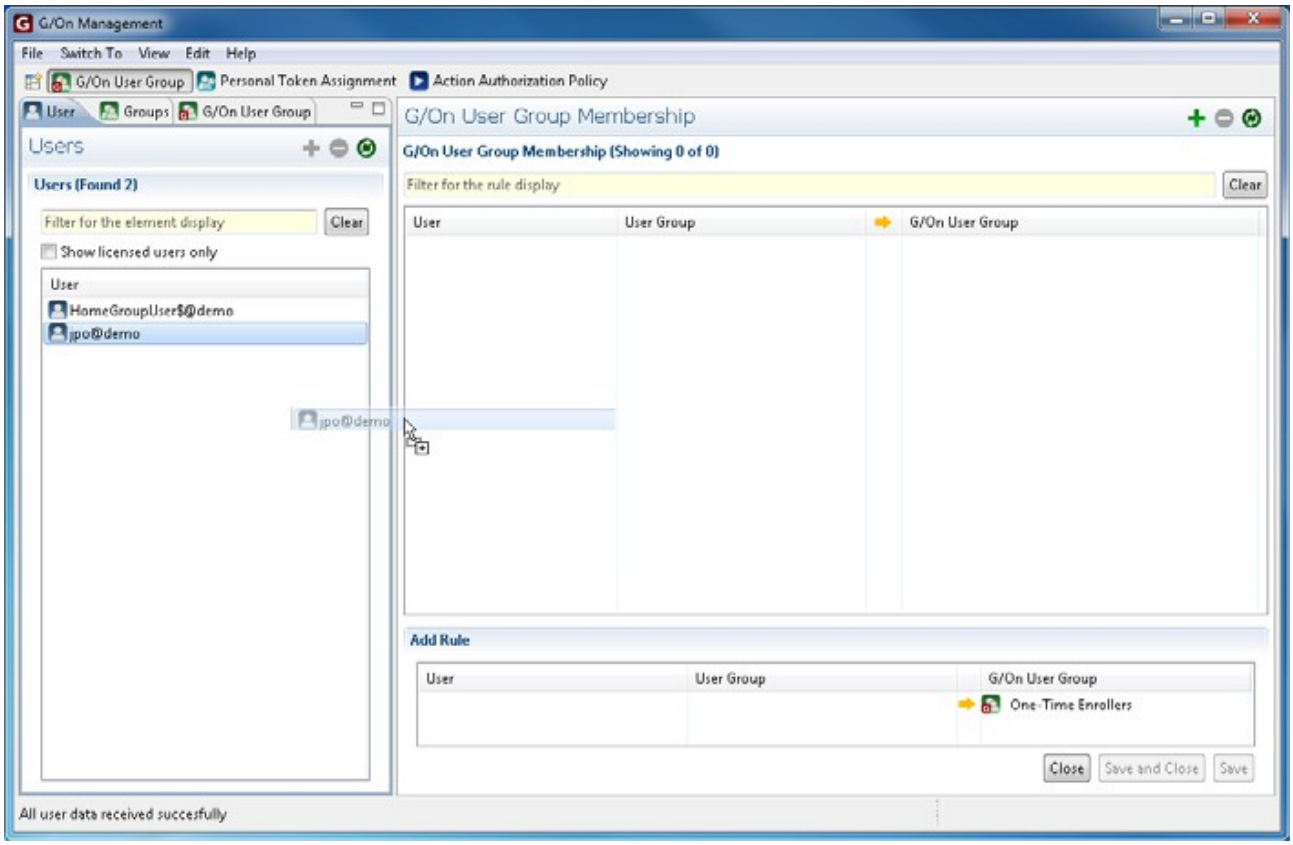
Preparing for Enrollment

Before the users try to configure, some setup is needed for them to be able to enroll their mobile devices, G/On USB, or computer. This is done by adding them to a special group called **One-Time Enrollers**.

1. Start the **G/On Management Client**
2. Open the **G/On User Group** perspective
3. From the **G/On User Group** list, drag **One-Time Enrollers** to anywhere in the Rule list on the right of the window



4. Then drag a user from the **User** list to the Rule list



5. Click **Save**

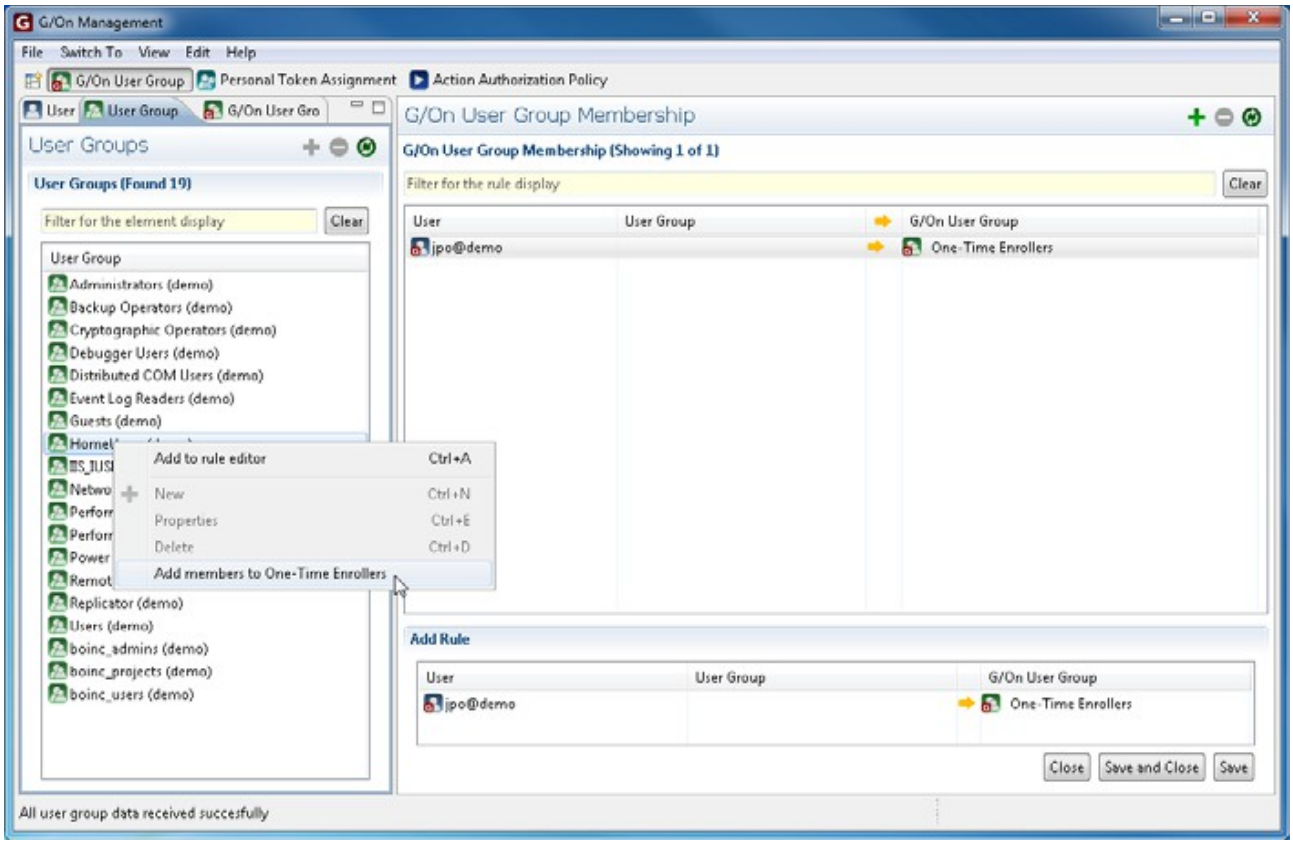
The user is now added to the **One-Time Enrollers** group.

6. Continue to drag users to the Rule list and click **Save** to add more users to the group
7. When done, click **Close**

Shortcut

An ease-of-use feature allows you to add all the members of a specific group to **One-Time Enrollers**. If you have a group that contains all the relevant users, this is the quickest way to add all of these users to **One-Time Enrollers**.

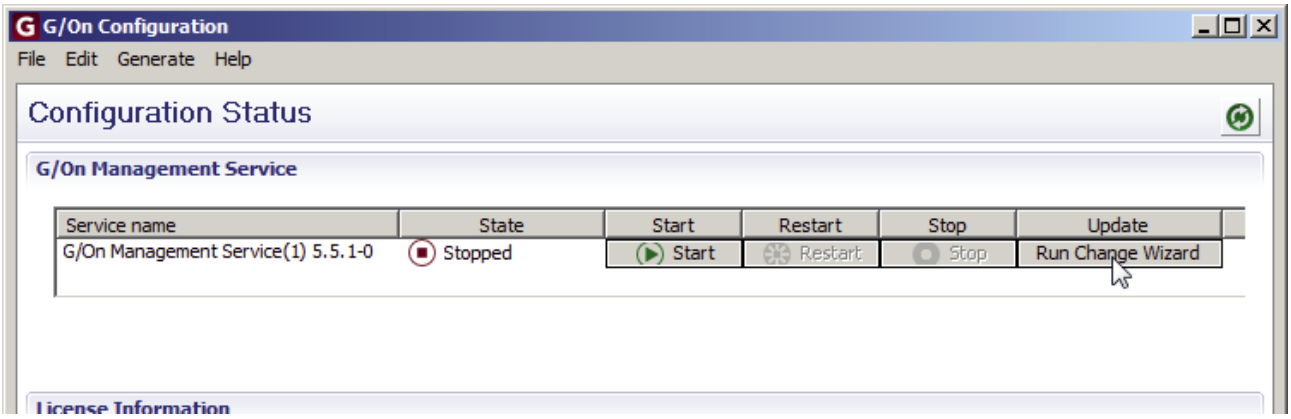
1. Select the **User Group** list
2. Right-click a group
3. Choose **Add members to One-Time Enrollers**



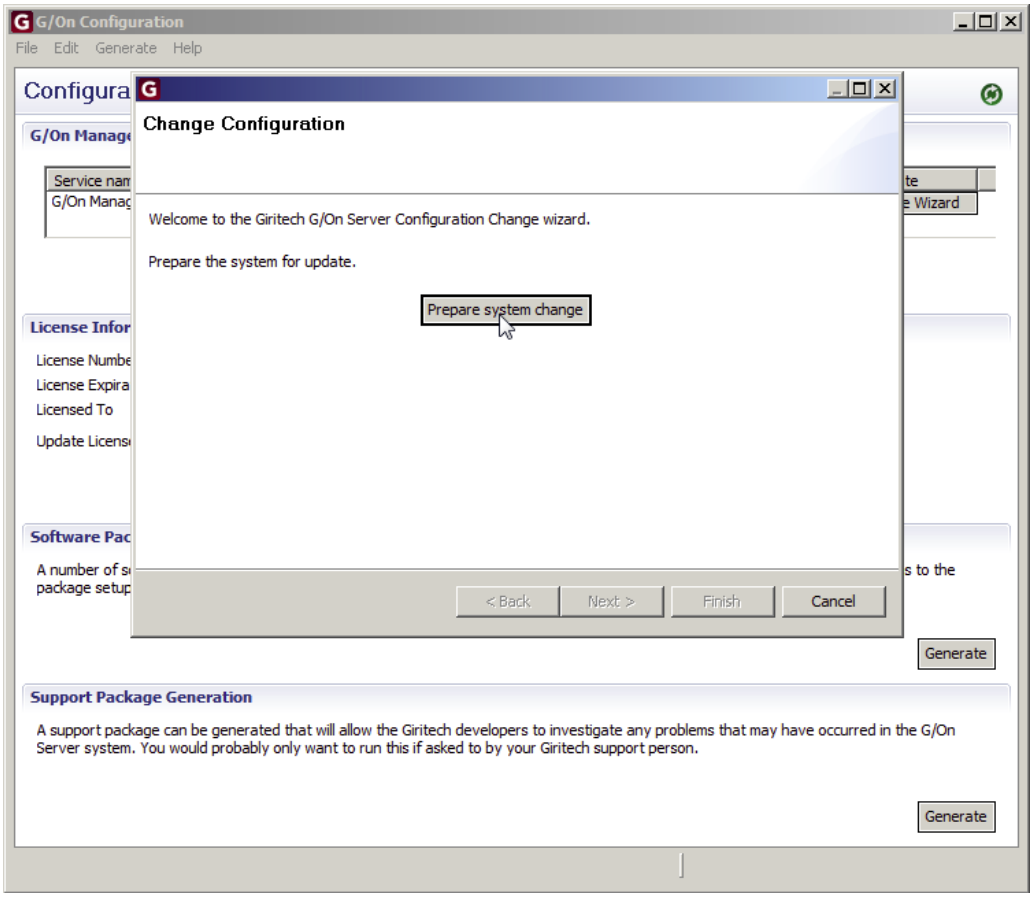
Automatic Approval of Enrollment

After the users enroll their mobile devices, G/On USB, or computers, you will need to approve them before they can use G/On (this is described below). This can also be done automatically, and can be enabled like this:

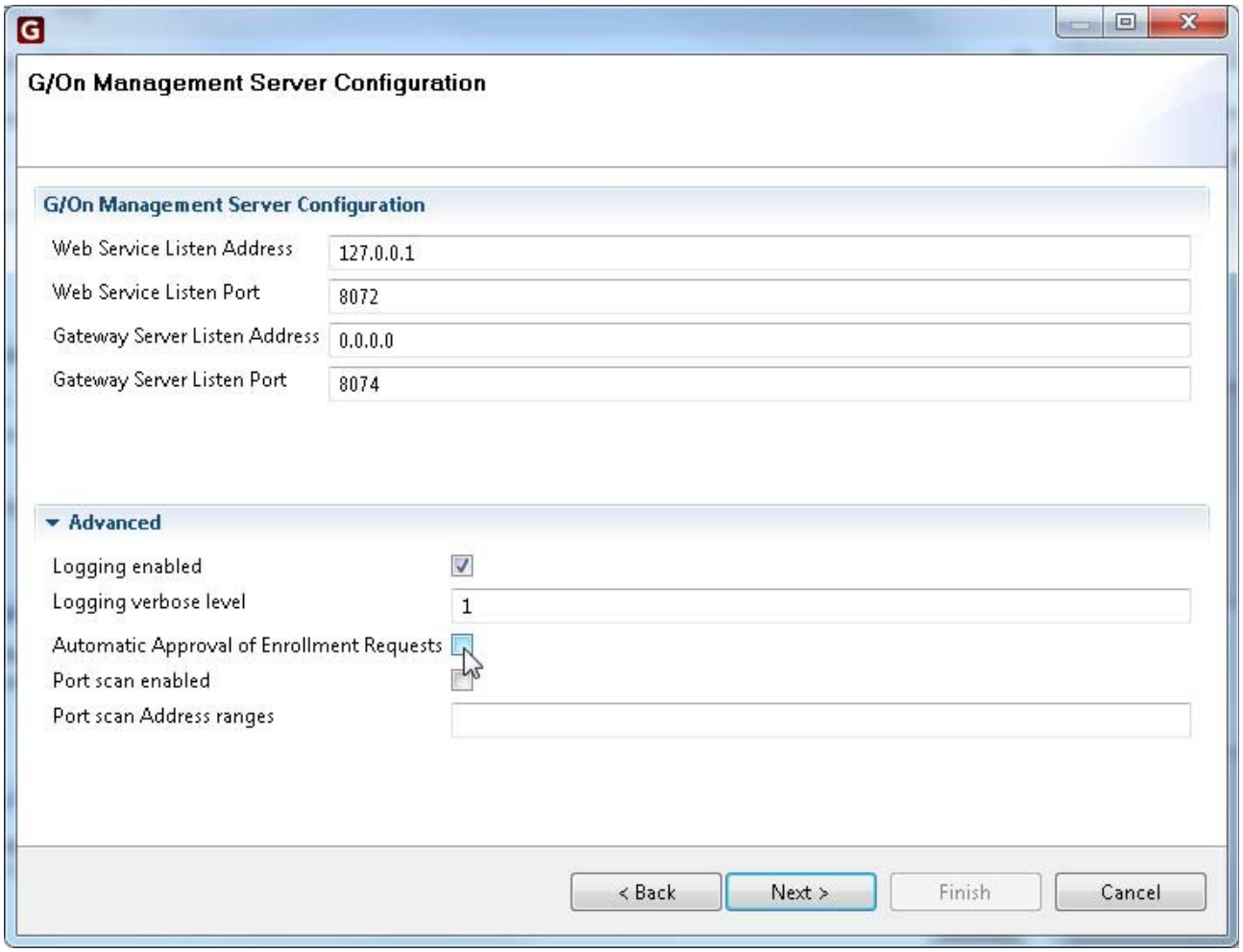
1. Start the **G/On Server Configuration**
2. Click **Run Change Wizard**



3. Click **Prepare system change**



4. Click **Next**
5. Click **Advanced** to unfold the Advanced section
6. Select **Automatic Approval of Enrollment Requests**



7. Click **Next** until you can click **Update**
8. Click **Finish**

Approval of Enrollment

Distribute the Link, Image, File, or Installer program to the users. Now they can configure and enroll their mobile device, G/On USB, or computer. If they need help to do this, manuals can be found on the homepage.

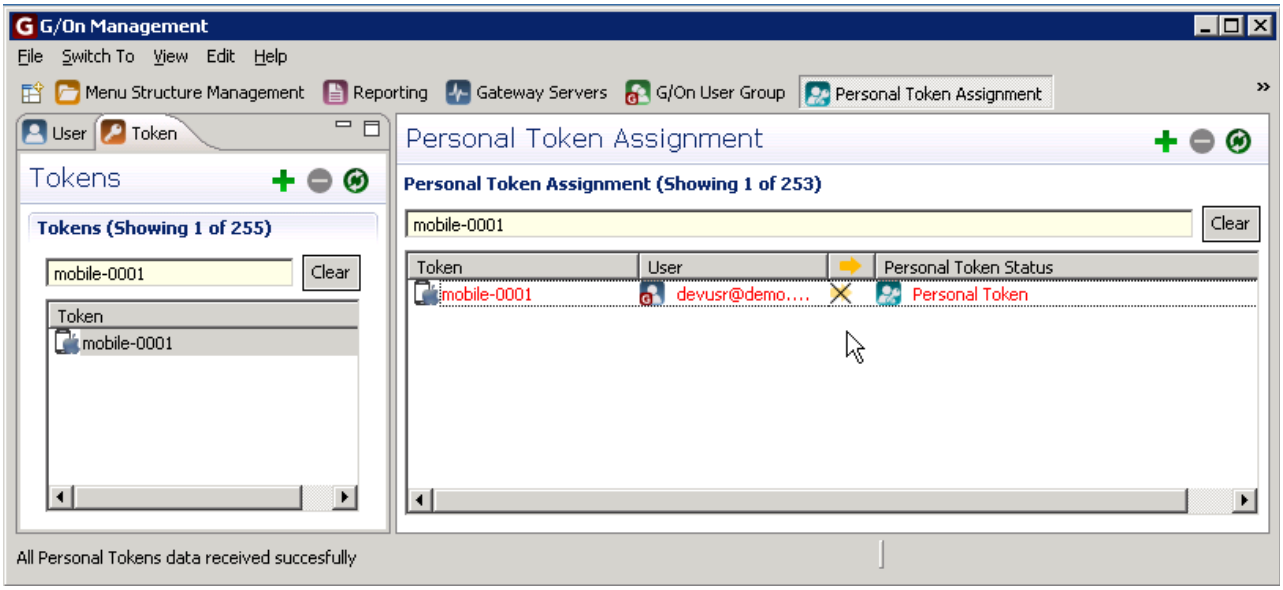
With Automatic Approval of Enrollment

If you have set Automatic Approval of Enrollment, you are done. As the users log on to G/On, their devices will be automatically enrolled, and are ready to use straight away.

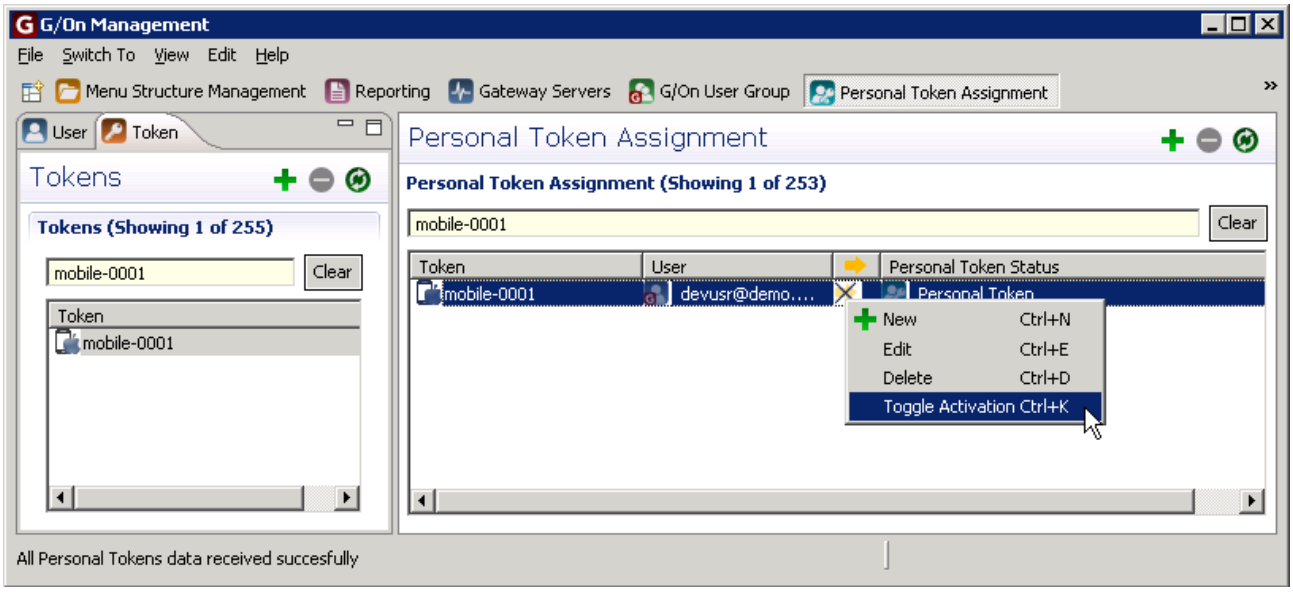
Without Automatic Approval of Enrollment

If you do not have automatic approval of enrollment, you need to Activate the relevant Personal Token Assignment:

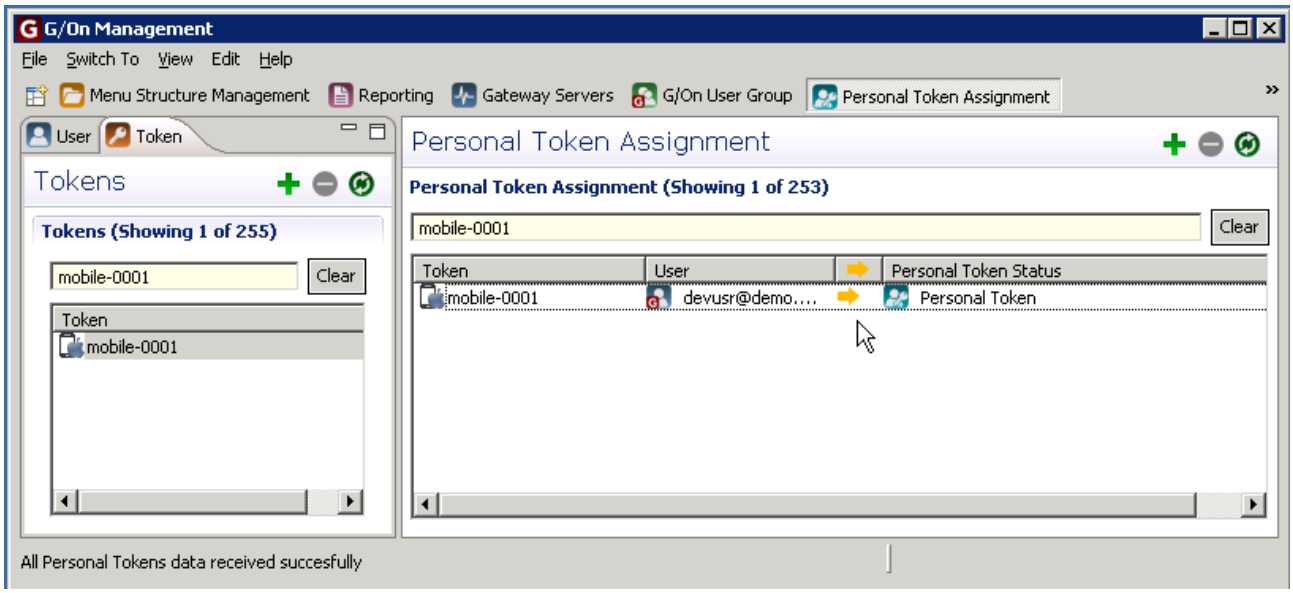
1. Start the **G/On Management Client**
2. Open **Personal Token Assignment** perspective
3. Find the rule with User and either the mobile device, G/On USB, or computer



4. Right-click the rule
5. Select **Toggle Activation**



The mobile device, G/On USB, or computer is now ready for use.

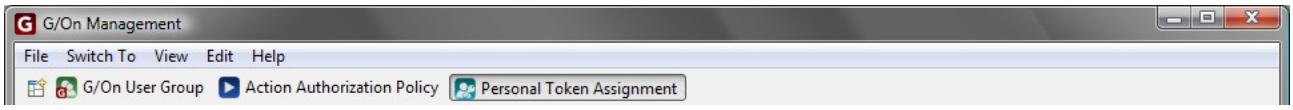


Local Configuration and Enrollment

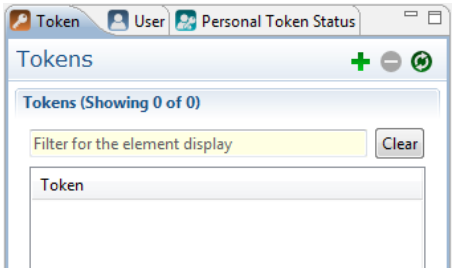
Enrollment

You can also prepare the G/On USB for the users, configuring and enrolling them using the G/On Management Client. This process start by enrolling the G/On USB and adding it as an personal token for a user. Afterwards it is configured by installing software packages.

1. Start the **G/On Management Client**
2. Open the **Personal Token Assignment** perspective

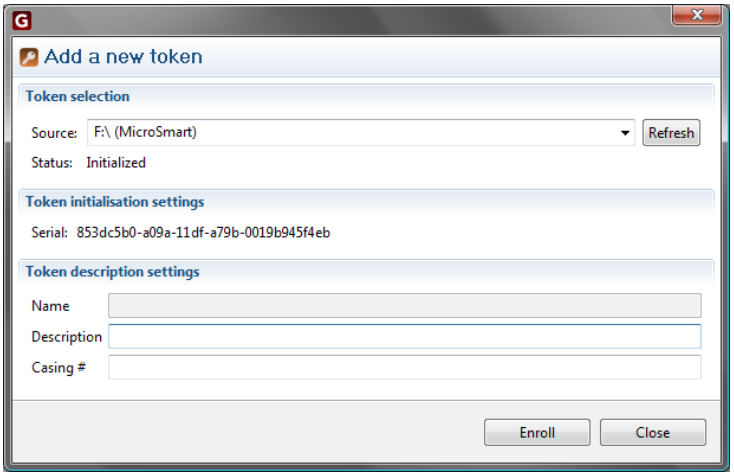


3. In the Token list, click the plus sign (+)



4. Insert the G/On USB into your local workstation
5. Click **Refresh**

Hint: For your own information, you can add a Description and/or Casing number as you see fit.



6. Click **Enroll**

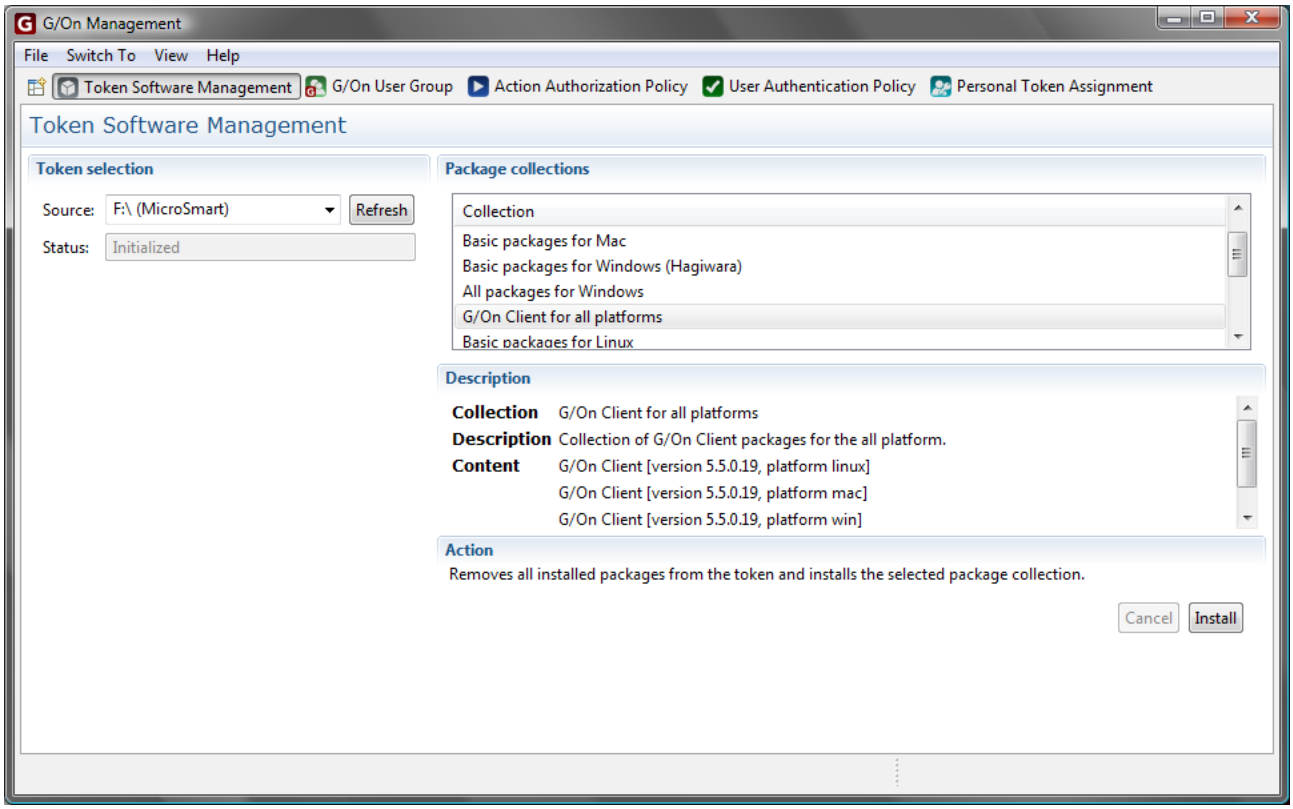
Now the G/On USB is enrolled, and you can assign it to a user, following the steps below.

This is important because we want each User to authenticate themselves by a Token that they carry and a password that they remember. This is what is called Two-Factor Authentication.

7. From the **Token** list, drag the new token to anywhere in the Rule list in the right of the window
8. From the **User** list, drag the relevant user to the Rule list
9. From the **Personal Token Status** list, drag **Personal Token** to the Rule list
10. Click **Save and Close**

Configuration

1. Insert the G/On USB into the computer
2. Start the **G/On Management Client**
3. Open **Token Software Management** perspective



4. Click **Refresh** until the G/On USB appears in the **Source** drop down list
5. Make sure to select the G/On USB in the list
6. In the right side, choose the packages you want to install on the G/On USB

Note: Adding packages to a G/On USB will erase all packages already installed on the G/On USB

Hint: If you select the collection called G/On packages for all platforms, your users can run the G/On client on Windows, Mac or Linux as they choose. If you are absolutely sure they will only use a certain platform, you need only select the relevant collection. This will save you time installing, as well as save space on the G/On USB.

7. Click **Install**

When installation is finished, you can hand the G/On USB to the user, who can use it right away.

Remember to make sure that they have access to Menu Actions. See the chapter *Management Reference* on page 101 for more information.

Instruction outlines

iOS

1. Use a Link, an Image or a File
2. Start G/On Management Client
3. Open G/On User Group perspective
4. Make a Rule that adds the Users to One-Time Enrollers group
5. Distribute the Link, Image or File to the Users
6. Approve enrollments

Computer User Token

1. Generate G/On Client installation program
2. Start G/On Management Client
3. Open G/On User Group perspective
4. Make a Rule that adds the Users to One-Time Enrollers group
5. Distribute G/On Client installation program to Users
6. Approve enrollments

G/On USB

Field Configuration and Enrollment

1. Generate G/On Client installation program
2. Start G/On Management Client
3. Open G/On User Group perspective
4. Make a Rule that adds the Users to One-Time Enrollers group
5. Distribute G/On Client installation program to Users
6. Approve enrollments

Local Configuration and Enrollment

1. Start G/On Management Client
2. Open Personal Token Assignment perspective
3. Add the G/On USB as a new Token
4. Make a Rule that connects G/On USB to the User as a Personal Token
5. Open Token Software Management perspective
6. Add software to the G/On USB
7. Distribute the G/On USB to the User

Configuration Reference

Introduction

Four different programs are used for installing, configuring and managing a G/On Server:

- **Windows Installer** creates the *G/On Master Installation* in a program folder and unpacks all the necessary files to this folder, and creates entries in the Windows start menu. The Master Installation contains the Management Server and also includes a Gateway Server. However, use of this Gateway Server may be supplemented by or replaced by one or more other Gateway servers, which are installed on separate machines by means of the Windows Gateway Installer (see below).
- **G/On Configuration** is used for basic configuration of a new G/On Master Installation (IP addresses etc.) and is also used for upgrading an existing version to a new version.
- **Windows Gateway Installer** is used for installing additional Gateway Servers, replacing or supplementing the Gateway Server in the G/On Master Installation.
- **G/On Management** is used for the management of authentication and authorization policies, and daily operation regarding users, tokens etc.

This document describes in detail the options available when using the G/On Configuration program. Please refer to the chapter *Configuration* on page 13 for a quick introduction.

See the chapter *Management Reference* on page 101 for documentation regarding the G/On Management program.

The architecture of G/On Configuration is a client-server application where both client and server runs on the same computer (the server). The G/On Configuration client automatically starts up a G/On Configuration server process, which is the one that does the actual configuration. The G/On Configuration server program can also be used as a command line tool, for certain tasks.

Note: On Windows Server 2008, you must run the G/On Configuration program as Administrator: Find the program in the Windows Start Menu, right-click on it, and choose: “Run as Administrator”.

Preparing installation

For a list of supported platforms and software dependencies, please see *Prepare installation* on page 9.

Overview: Making New Installations and Upgrades

To make a new installation:

- Run the Windows Installer, G/On Configuration program and G/On Management program, in this order.

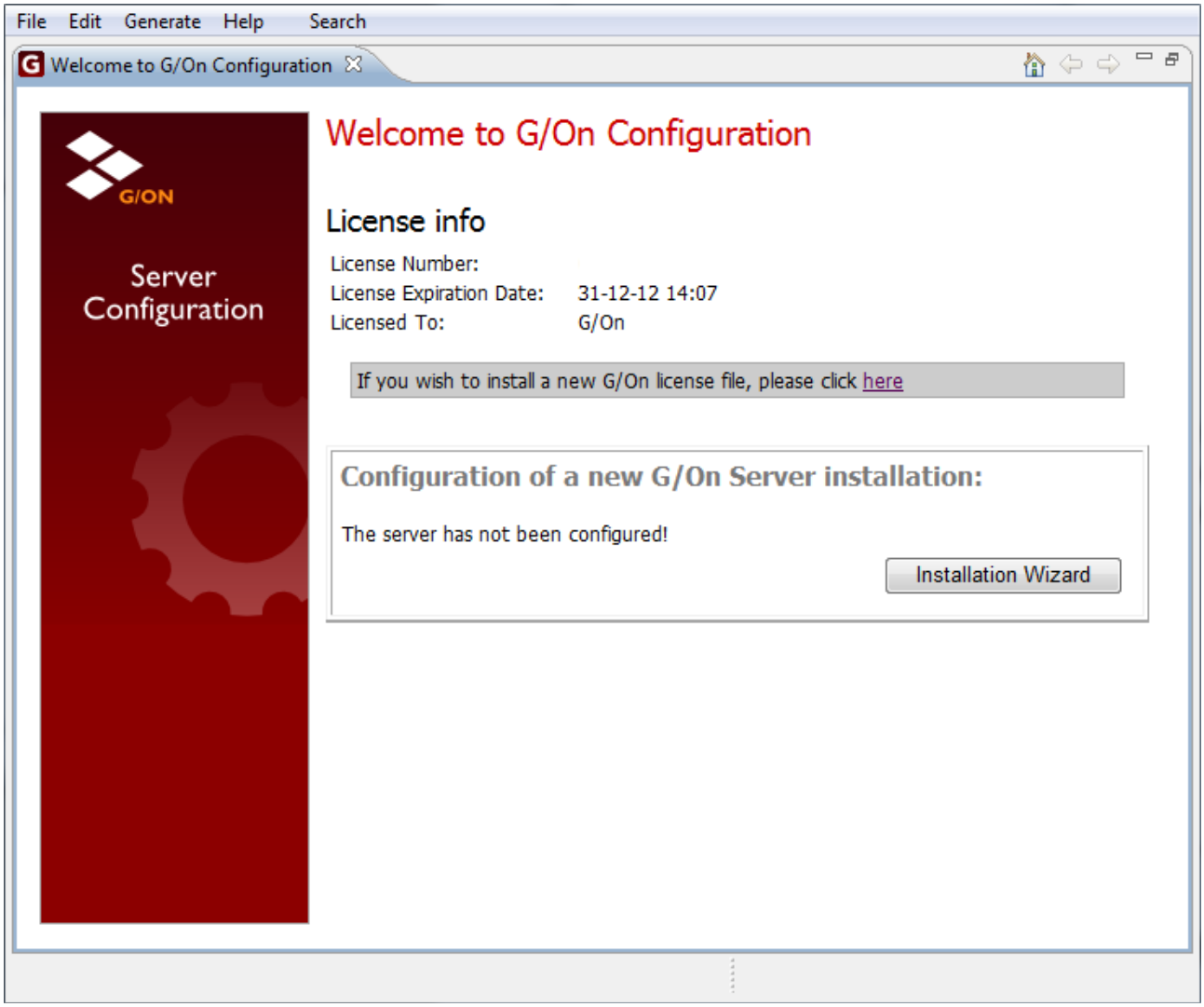
To make an upgrade of an existing installation:

8. Install the new version, by running the Windows Installer for that version. This will make a new program folder for the version, without affecting the already installed versions.
9. Run G/On Configuration for the new version. On the Welcome Screen, there will be a list of the already installed versions. Choose the one, which you want to upgrade from, and complete the steps that you are guided through.
10. Now, the services of the new version are ready to be started. But before doing that, stop (and disable) the services of the old version. This is necessary, because the services of the new version listens on the same ports as the old version, and two different services cannot listen on the same ports.

Note: Before starting any installation or upgrade, read the release notes, to see if there are special issues to consider.

G/On Configuration Welcome Screen

The first time you open G/On Configuration, you will be presented with a Welcome Screen like this:



This is because the G/On configuration utility has detected that the server has not yet been configured. Configuration is done using the installation wizard, which is described below.

The Welcome screen can also be opened from the Main Status Window by choosing Help > Welcome to the G/On Configuration in the menu.

In case one or more upgradable G/On system is already installed on the server, these systems will also be listed in the Welcome Screen.

To upgrade from a previously installed system press the “Upgrade Wizard” button for that system. The Upgrade Wizard is described below.

No License

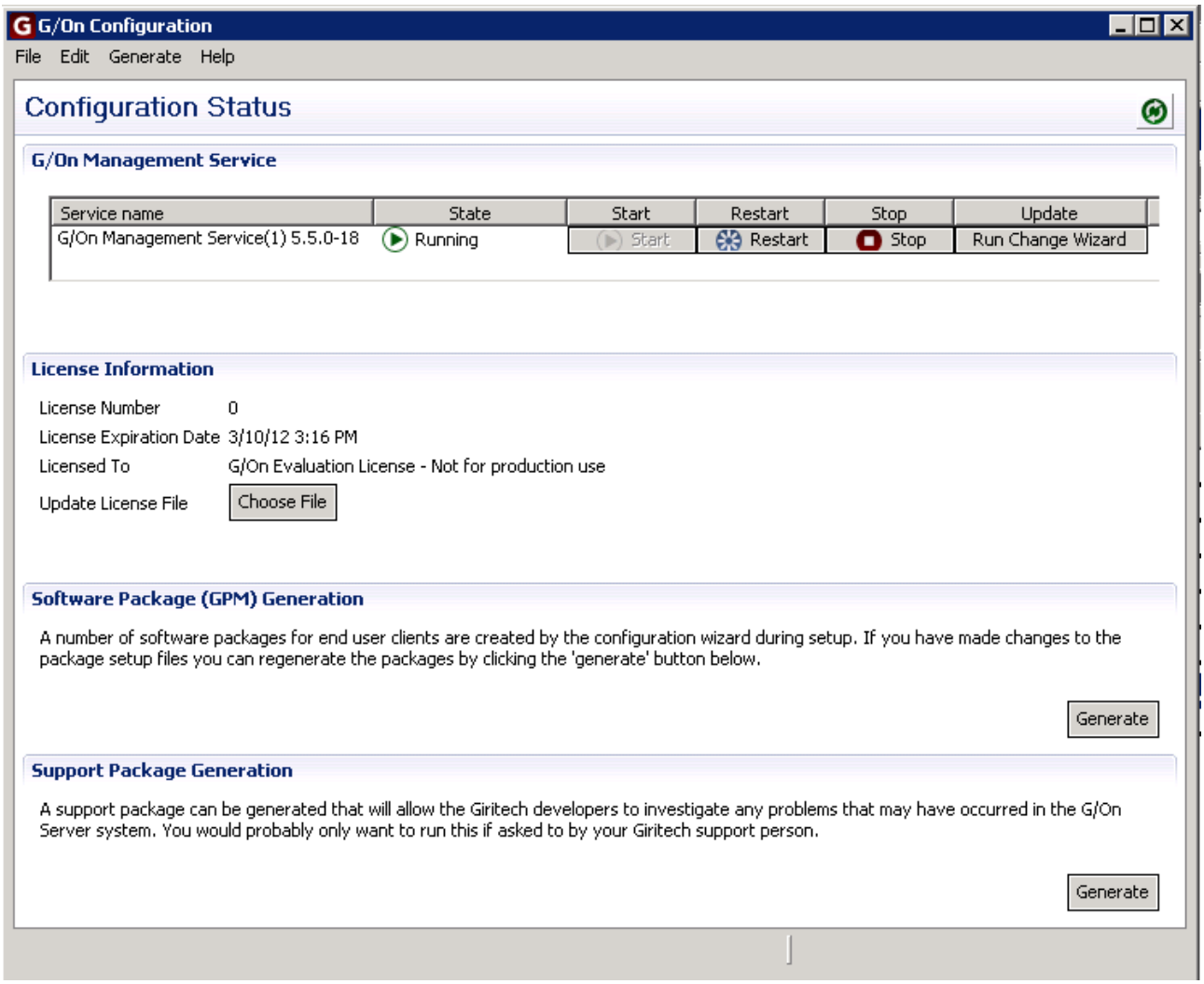
If you do not use a proper G/On license file, the installation will proceed with an evaluation license. If you have acquired a proper license file, you can place it in the folder

```
\config\deployed
```

or you can simply click on the link in the window. This will open a file chooser, in which you can choose the license file and install it.

Main Status Window

If the Installation Wizard has already been run successfully, G/On Configuration will open in the Main Status Window, which looks like this:



The Window is divided into three parts. Each part is described below.

G/On Management Service

In this section of the status window, you should see status of the installed G/On management service. The following information/functionality is availiable:

- **Service name:** The name of the management service.
- **State:** Current state; Running, Restarting or Stopped.
- **Start:** Starts the service.
- **Restart:** Restarts the service.
- **Stop:** Stops the service.
- **Run Change Wizard:** Starts the Change Wizard (see below).

Software Package (GPM) Generation

This section contains a description of the Software Package concept and a button which starts up the Software Package Generation Wizard. Terminology notes: GPM stands for G/On Package Management. Currently, most of the packages contain software to be deployed on the client side, e.g., application clients. These packages are also referred to as Client Packages, and the Client Package Management actions in the menu of the end-user client can be used for installing, updating and deleting client packages.

Support Package Generation

This section contains a description of the Support Package concept and a button for generating a Support Package. Support Packages can also be generated by choosing Generate > Generate Support Package in the menu.

A Support Package is a zip-file containing ini-files, log-files and more, that can be generated and send to Giritech Support. Notice that the database and the server part of the known secret are NOT included in the Support Package, because this information should only be shared in very special situations.

After completion a file chooser will open in which you can choose where to put the generated zip file.

Wizards

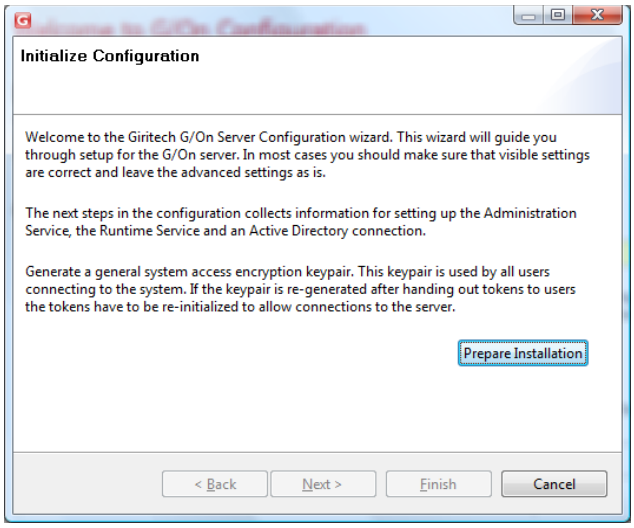
This section contains detailed information regarding the various Wizards in the G/On Configuration tool.

Installation Wizard

The installation Wizard is started automatically, the first time you run G/On Configuration. It can also be started by clicking the Start Wizard button In the “G/On Configuration Wizard” section in the Welcome Screen.

Note: the Installation Wizard should only be run once. Running it again on an installed system will erase system data and potentially invalidate the system.

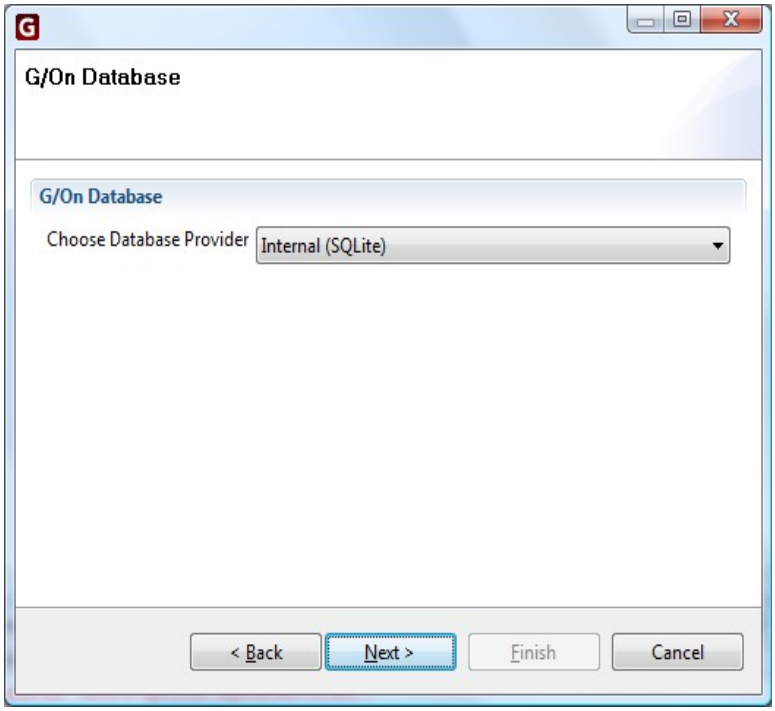
Initialize Configuration



Click Prepare Installation in order to run the preparation job. If no errors occur, you will be able to Click the Next button after the job finishes. If any errors occur they will be shown immediately after the Initialize Configuration title and you will not be able to continue the Wizard.

Database setup

A database is used for saving system setup.

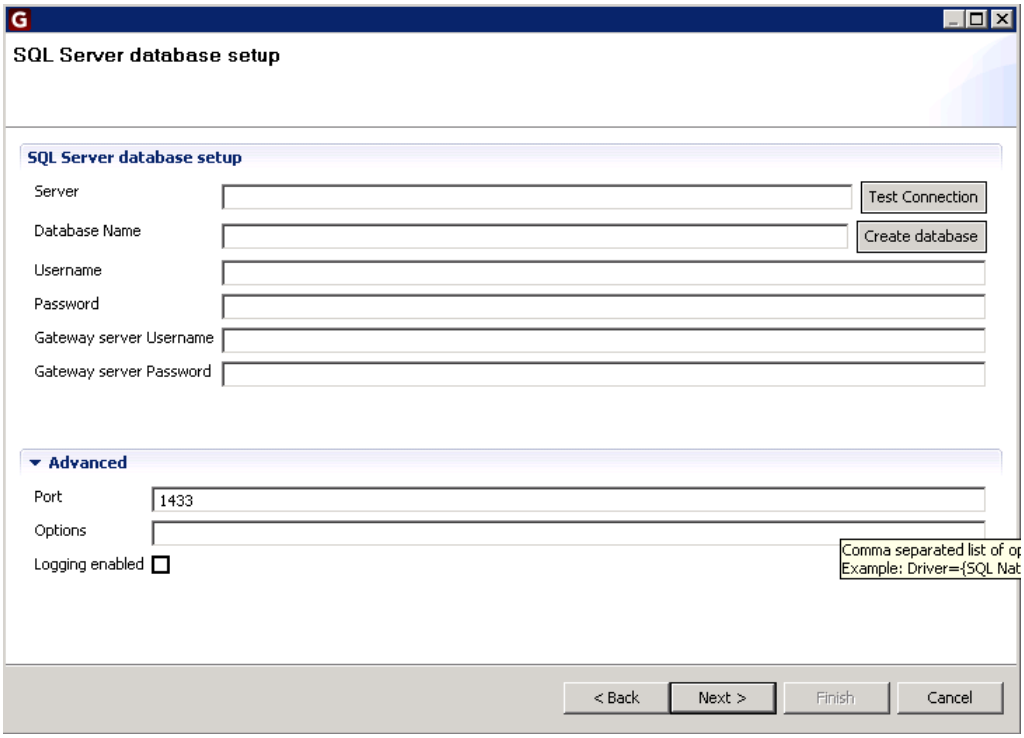


You must select database provider. If the setup will have only a single Gateway server, you can use the default Internal (SQLite) database. This requires no further configuration.

For a setup that can handle multiple Gateway servers, select Microsoft SQL Server or MySQL. If you choose either of these, a new window for entering further configuration will open when you click Next.

Database configuration

Both the SQL Server and MySQL configuration window looks like this:



Here, the database configuration can be entered:

Standard	
Server	Name or IP address and port number of SQL Server or MySQL host., e.g. myhost:1433
Database Name	The name of the database which will hold G/On data
Username	User name for a database administrator for the specified database. Leave blank if NT authentication should be used.
Password	The password for the specified user.
Gateway Server user name	Optional user login for the gateway server. The gateway server only needs read permission to the database, so it can make sense in some set-ups to use another account for database access on gateway servers.

	Leave blank in order to use same authentication as the config and management services.
Gateway Server password	The password for the Gateway server user account.

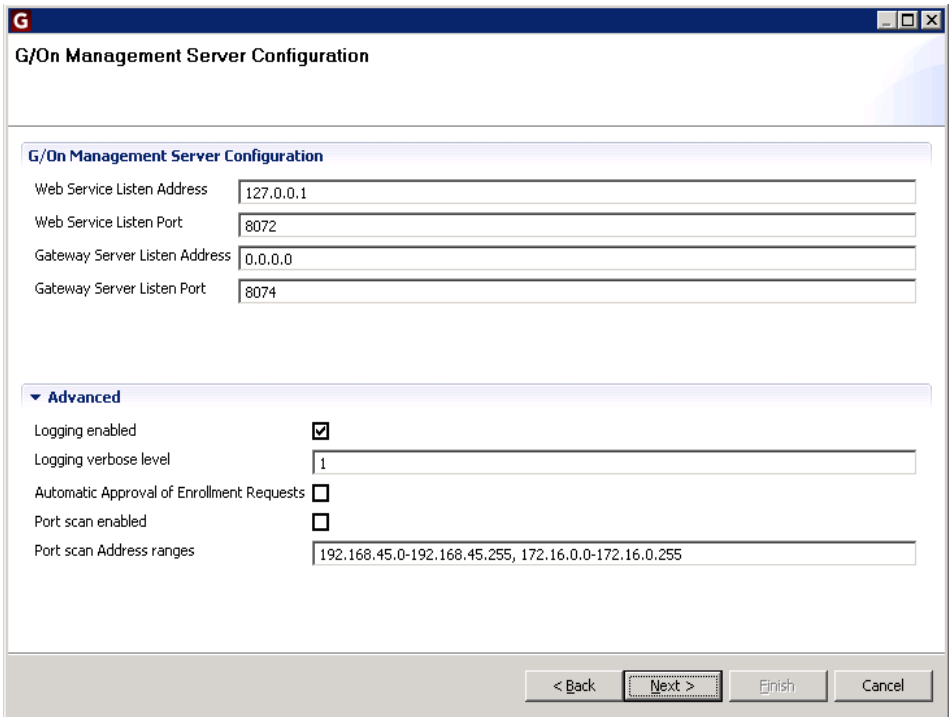
Advanced	
Port	The port number used for communication. Leave blank if you want the port number to be negotiated by the ODBC client.
Options	A comma separated list of extra options for the ODBC connection. Can be used for a failover setup. Example: “Driver={SQL Native Client}, Failover_Partner=SomeServerName”
Logging enabled	Enable logging

The button “Check connection” will check whether the connection to the database is ok and create and delete a temporary table in order to test that the connection has the rights necessary for creating the database.

The “Create database” button will try to create the database specified. Use this if the database has not already been created in the SQL Server Management tool.

Management Server Configuration

The Management server allows management of the solution (users, authentication and authorization policies. It accepts input from the G/On Management Client and stores the resulting policies etc. in a database, where the Gateway server can read it.



In this window, the Management Server configuration can be entered. Note that the Advanced pane is hidden initially, but in the screen shot above it has been opened in order to show the information fields.

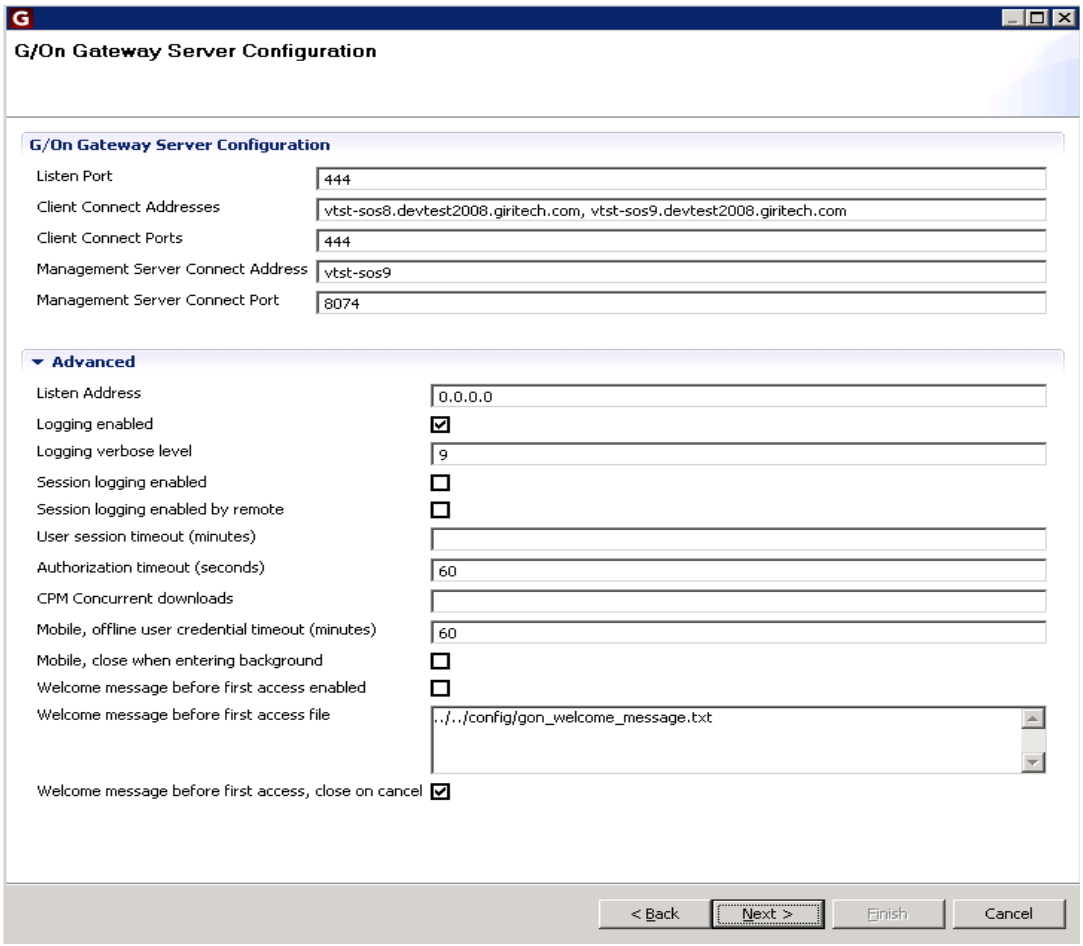
Standard	
Web service listen address	IP address that the Management Server should listen on. Default is 127.0.0.1 – which means that the G/On Management Server will only allow connections from the machine the Management server is running on. That way, the Management tool can only be accessed directly on the G/On server itself (through the console or a terminal server session) or through the G/On Gateway Service also running on the server. If for some reason the Management Server should allow connections from other machines, the Listen IP address can be specified as 0.0.0.0 – winch

	will allow access from all IP addresses on the local network. As stated above, authorization to use the G/On Management tool must then be enforced by other means, so this option should be selected carefully!
Web service listen port	TCP Port where the Management serrvice should listen for connections from the Management Client.
Gateway server listen address	IP address where the Management serrvice should listen for connections from Gateway Servers.
Gateway server listen port	TCP Port where the Management serrvice should listen for connections from Gateway Servers.

Advanced	
Logging enabled	Enable logging
Logging verbose level	The primary purpose of logging in this context is for support reasons. Currently, there are two logging levels defined: 0: All warnings, errors and critical errors will be logged 9: Very detailed logging level of all activities. Using this level will severely impact performance – and should not be used unless needed for support reasons (remember to deactivate).
Automatic Approval of Enrollment Requests	If checked, the personal token assignments created as a result of field enrollments are automatically activated. If not checked, these personal token assignments will be inactive until manually activated by an administrator.
Portscan enabled	Enable the possibility for port scanning when creating Menu Actions. Note that port scanning can violate local network security policies.
Portscan IP ranges	When port scanning is enabled, the ranges of ports to be scanned will be the ones defined here. A range is simply defined as <startPort>-<endPort>, and more ranges can be specified by separating them with a comma.

Gateway Server Configuration

The Gateway server does the actual “gate keeping”: it accepts connections from G/On clients, gets user names and passwords and tokens checked, and grants access to menu actions in accordance with the Authentication and Authorization policies specified in G/On Management.



Standard	
Listen Port	The port that the Gateway Server listens on in order to accept connections from G/On Clients. Only one port can be specified here. **Note:** The G/On clients can be configured to try connecting to several ports (see the field: “Port the client connects to”). In this case, there must be a firewall/router in front of the G/On Gateway server, which maps all these “external” ports to the port that the server is actually listening on.
Client Connect Addresses	This is the IP address (DNS name or number), that the G/On clients will use to connect to the G/On server. Please note, that when using a proper

	license, this address is fixed, and must be determined at the time of ordering G/On, as the connection address is part of the license (file). If using the demo license, any address can be specified.
Client Connect Ports	Although 3945 is the official IANA allocated port-number for G/On – other port-numbers can be used. Port 80 – or 443 are recommended, as these ports are open outbound in most environments. So by selecting these ports, the G/On clients will be able to connect to the G/On server under all normal circumstances. The port(s) must be specified at the time of ordering G/On, and is part of the license (file). If more ports are to be used, all ports must be specified at the time of ordering – and the ”Multiport” Option must be part of the license. If using the demo license, any port can be specified.
Management Server Connect Address	IP address or DNS name, which the Gateway servers should use for connecting to the management server.
Management Server Connect Port	TCP Port number, which the Gateway servers should use for connecting to the management server.

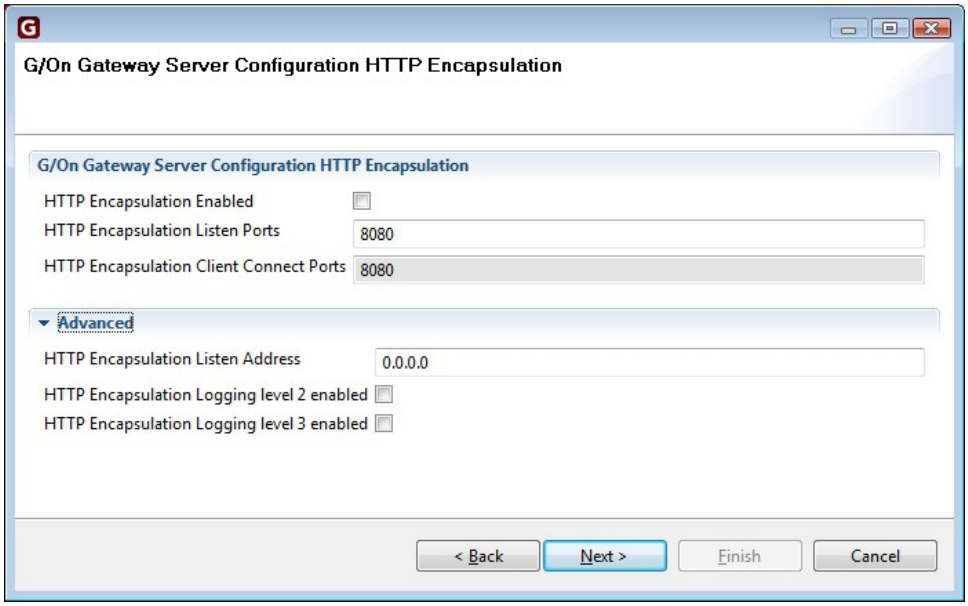
Advanced	
Listen address	This is the internal address that the G/On Gateway will listen on to accept connections from G/On clients. 0.0.0.0 will enable connections on all the network interfaces of the Gateway Server machine (default).
Logging enabled	Enable logging. The primary purpose of logging in this context is for support reasons.
Logging verbose level	Currently, there are two logging levels defined: 0: All warnings, errors and critical errors will be logged 9: Very detailed logging level of all activities. Using this level will severely impact performance – and should not be used unless needed for support reasons (remember to deactivate).

Session logging enabled	Log each user session in a separate file
Session logging enabled by remote	Enables possibility for session logging controlled by client. If this option is set you can get session logging for a specific client by specifying it in the client's configuration file.
ser session timeout (minutes)	Timeout to stop sessions, which has been idle for more than the specified period. Set to 0 in order to disable session timeout.
Authorization timeout (seconds)	This is the time users have to complete the authentication process (specify user-id and password) from a connection is established. If the user does not log on during the specified time, the connection is terminated.
GPM Concurrent Downloads	To avoid performance impact, the number of concurrent downloads of GPM packages can be limited by setting this field. This controls how many users can simultaneously do field updates or installs of the software on the tokens. If this limit is reached, the next user that attempts an installation or update will observe that the process is paused before download, and then automatically resumed at a later time, when fewer users are downloading.
Mobile, offline user credentials timeout (minutes)	The time period credentials are saved on mobile devices (iPad, iPhone). In order to improve usability, credentials are saved on mobile devices in the specified number of minutes.
Mobile, close when entering background	Controls whether G/On should disconnect from the server on mobile devices (iPad, iPhone), when the G/On app is entering background, i.e. when you switch to another app or to the main menu. Note that disconnecting means that port forwards (e.g. for mail clients) are disabled
Welcome message before first access enabled	This option can be used to acquire user acceptance of the terms and conditions under which access is granted.
Welcome message before first access message file	If the previous option is enabled, this file contains the message which the user must accept at the first time access is about to be granted. When using a relative path, note that the current working directory is: gon_server_gateway_service\win

Welcome message before first access, close-on-cancel	If this is checked, the G/On connection will be closed, unless the user clicks Accept, when shown the message.

HTTP Encapsulation

In some environments, a deep packet inspection firewall or other might block the communication between G/On clients and the G/On server. To allow connectivity in these situations, the **HTTP Encapsulation** option should be specified when ordering G/On. This optional feature enables the G/On client to encapsulate the G/On data stream in HTTP packages, thereby using G/On in ”web communications” mode. This will allow the G/On client to connect from virtually all environments, where a web browser can be started successfully.



If the HTTP Encapsulation option has been specified when ordering G/On, you can enable and configure this feature as follows:

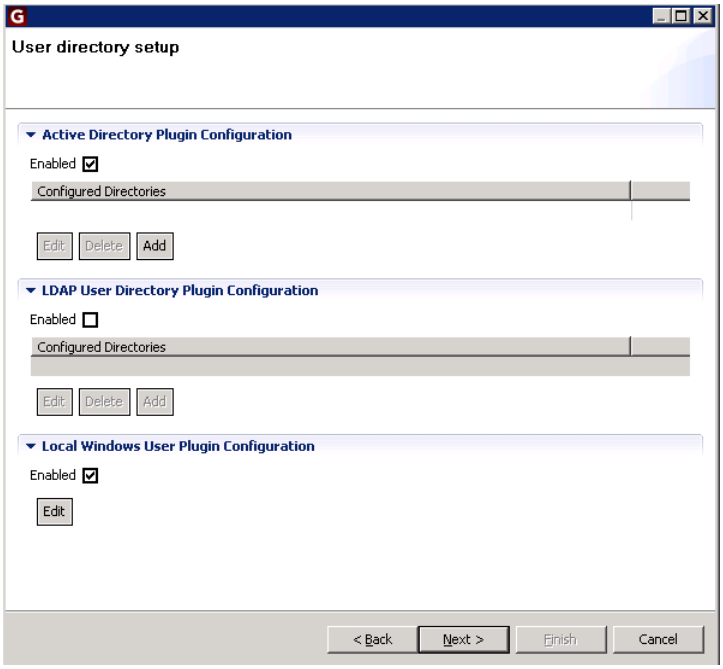
Standard	
HTTP Encapsulation Enabled	Enable or disable use of HTTP encapsulation.
HTTP Encapsulation Listen port	specifies the port on which the Gateway Server will listen for HTTP

	Encapsulated G/On traffic, on the inside of the firewall.
HTTP Encapsulation Client Connect Port	specifies the ports, that G/On clients will use on the outside when sending HTTP encapsulated data streams.

Advanced	
HTTP Listen Address	Specify the address from which HTTP Encapsulated traffic are accepted. 0.0.0.0 (default value) defines all addresses.
HTTP Encapsulation Logging level 2 enabled	Debug logging enabled.
HTTP Encapsulation Logging level 3 enabled	Very detailed logging level of all activities. Using this level will severely impact performance – and should not be used unless needed for support reasons (remember to deactivate).

User Directory Configuration

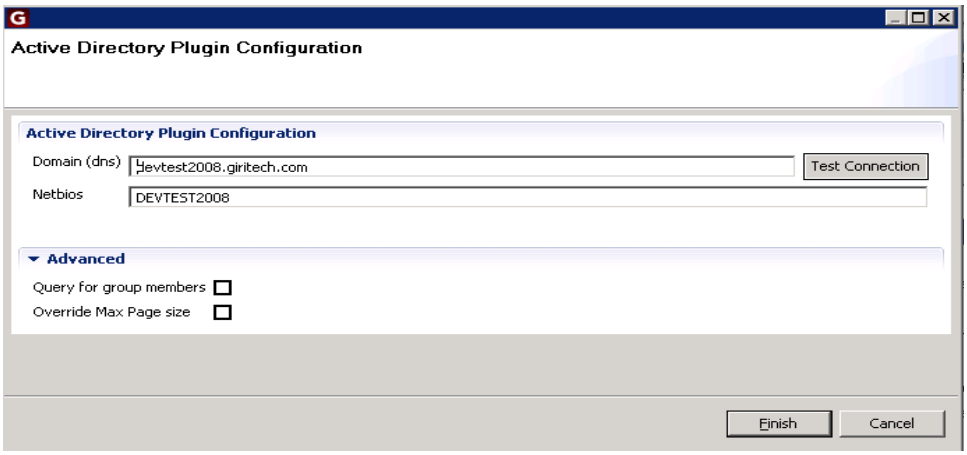
On this page it is possible to add (and remove) user directories used for user verification. There are currently three types of user directories: Active Directory, LDAP and Local Windows User (note that LDAP is a license feature and therefore may not be available).



Each user directory type can be enabled/disabled using the “Enabled” check boxes. For Active Directory and LDAP it is possible to add any number of different directory specifications, whereas there can be only one instance of the Local Windows User plugin. Choosing the “Add” or “Edit” buttons will open a new window with specifications for the user directory (type) in question. These windows are described below.

Active Directory User Directory Plugin Configuration

The Active Directory plugin uses the Windows API to connect to Active Directory. In order for Active Directory integration to work properly the installation must be done on a computer which is either on the Active Directory domain or on a domain with which a trust relationship has been established.



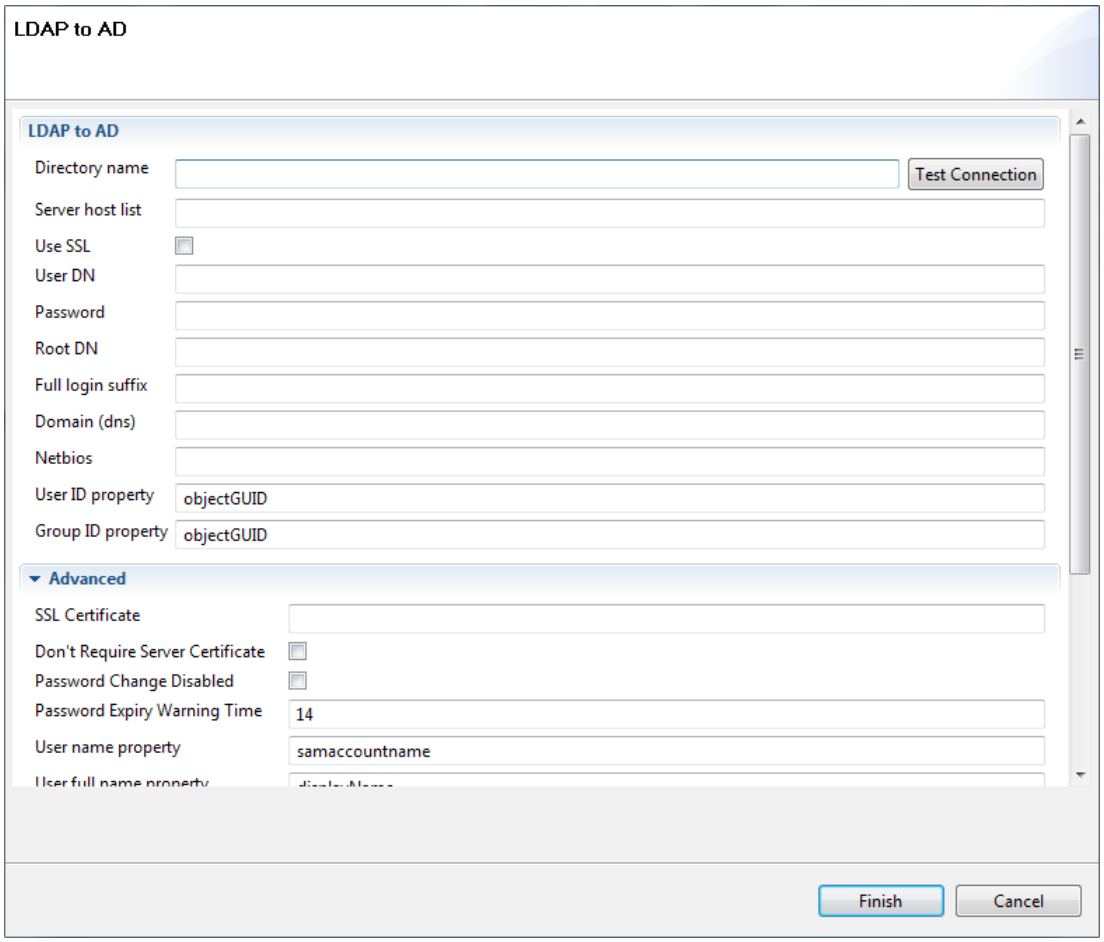
Standard	
Domain (dns)	Enter dns name of the AD domain, e. g. mycompany.com.
Netbios	The Netbios name of the AD domain.

Advanced	
Query for group members	Enables an alternative method for finding group memberships. The standard way does not include most domain local groups on remote trusted domains and most built-in domain local groups on the local domain (e.g. Remote Desktop Users). Note however, that the alternative method has been seen to fail on some installations, probably because of inadequate access rights – it has not been possible so far to pinpoint the exact reason.
Override Max Page Size	Overrides the Max Page Size limit so that all entries can be fetched in a query. Queries to Active Directory will by default only return a limited number of entries. The limit is controlled by a constant called MaxPageSize, which by default is 1000 (it can be changed using the “Ntdsutil.exe” utility program). The limit can be overridden in the query itself and enabling this option will cause G/On to do so.

LDAP Plugin Configuration

The LDAP plugin uses the LDAP protocol for user verification and for obtaining information about users and groups available. The three options are up against Active Directory, up against Domino, or up against another form of directory. It can, in principle, be used against any LDAP enabled User Directory, but has only been tested against Novell eDirectory and Active Directory.

So among the uses of the LDAP plugin is another way of connection to Active Directory. See page 86 for a discussion of issues related to Active Directory and LDAP and which plugin to use.



Standard	
Directory Name	Enter a name for the directory. This name will be used to identify users and groups from this LDAP Directory. This name should **never** be changed once users and groups have been entered in the system.

Server host list	A comma-separated list of servers for the LDAP directory. Add more servers to get fail-over if first server is down. Port number is assumed to be 389 unless specified. **Example**: firstserver:636, secondserver, thirdserver
Use SSL	Check if SSL communication should be used.
User DN	Name (dn) of user account used for connecting to LDAP in order to search for information. Leave blank if anonymous access is enabled in the User Directory. Note: AD does not allow anonymous access. **Example**: cn=myuser,ou=myorgunit,dc=mydomain,dc=com
Password	Password for the user specified account
Root DN	The root DN under which users, groups and ou's should be found. **Example**: dc=mydomain,dc=com
Full login suffix	The login suffix that distinguishes users from this directory from another one. If more than one user directory is specified, then there may be name clashes on the login names. If this is the case users must enter a *full* login, which is the normal login succeeded by a "@" and the Full login suffix (e.g. username@mydirectory). If left blank the Directory name is used as suffix.
Domain (dns)	The domain DNS name used when launching menu actions using the *user.domain* variable.
Netbios	The domain Netbios name used when launching menu actions using the *user.netbios* variable.
User ID property	To make sure that a user is always uniquely identified, the User ID property determines which property in the user directory to use as this unique identifier. Default value for LDAP to AD is: *objectGUID* Default value for LDAP to Domino is: *uid*
Group ID property	To make sure that a group is also always uniquely identified, the Group ID

	property determines which property in the user directory to use as this unique identifier. Default value for LDAP to AD is also: *objectGUID* Default value for LDAP to Domino is: *dn*

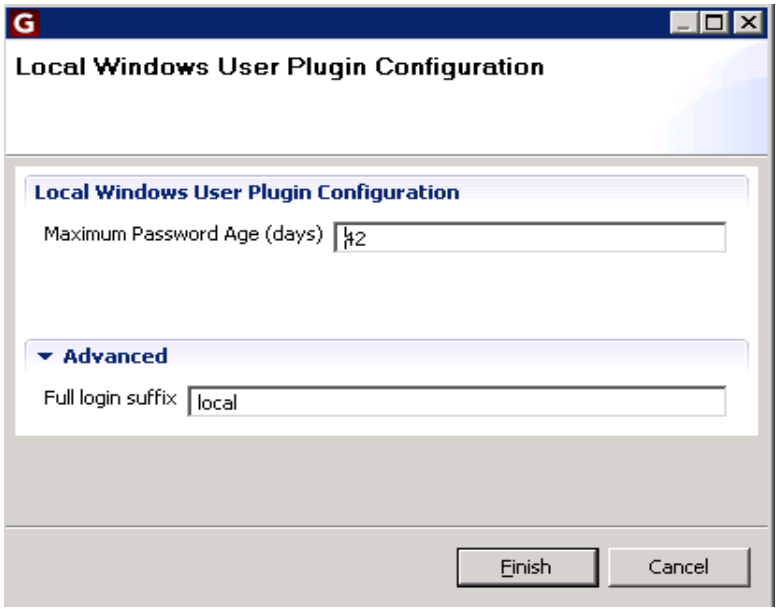
Advanced	
SSL Certificate	Full path to Certificate file used for SSL communication.
Don't Require Server Certificate	Set this if the Gateway server should **not** check the server certificate when connecting using SSL. In other words this enables SSL communication without server verification.
Password Change Disabled	Check if password change via G/On should be disabled.
Password Expiry Warning Time	Time (in number of days) before which the user is warned about password expiring. Enter '0' in order to disable warnings.
User name property	The property from the user directory that G/On shall display as the user's names.
User full name property	The property from the user directory that G/On shall display as the user's full names.
User query	The query used to fetch the users from the user directory to be used in G/On.
User login properties	The user directory property used as login If more than one property can be used, write them in a comma-separated list, example: cn, sn
User group membership property	What user directory group membership property G/On shall use when giving access via a group membership.

User password property	What user direcotury password users shall use to login to G/On.
Group name property	The property from the user directory that G/On shall display as the group's names.
Group full name property	The property from the user directory that G/On shall display as the group's full names.
Group query	The query used to fetch the groups from the user directory to be used in G/On.
Organizational Unit name property	The property from the user directory that G/On shall display as the organizational unit's names. Note that organizational units in G/On are displayed as if they were groups.
Organizational Unit full name property	The property from the user directory that G/On shall display as the organizational unit's full names.
Organizational Unit query	The query used to fetch the organizational units from the user directory to be used in G/On.

Local Windows User Plugin Configuration

The Local Windows User plugin is used for user verification and for obtaining information about local users and groups on the local server..

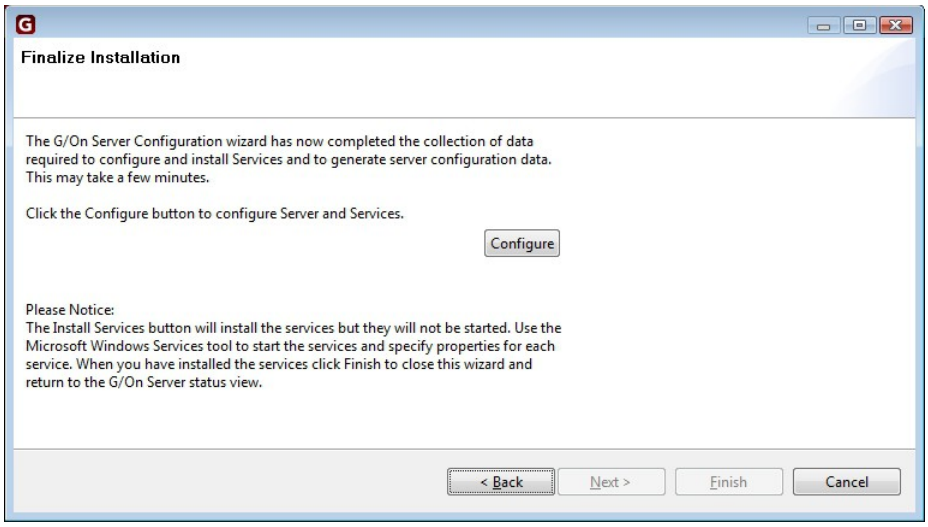
Note: In order for this to work correctly the G/On Management and all Gateway Servers should run on the same machine, so they will “see” the same local users and groups.



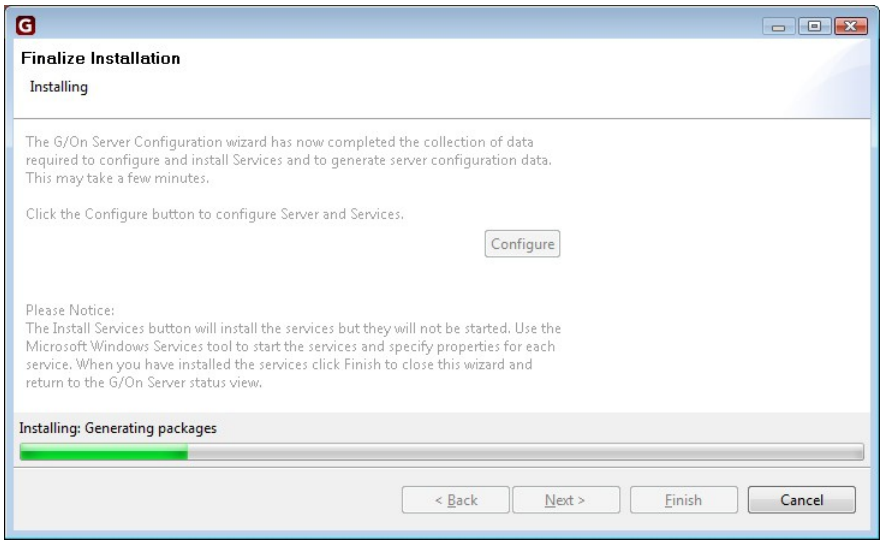
Standard	
Maximum Password Age (days)	When a user's password is older than this limit, G/On will ask the user to change the password.

Advanced	
Full login suffix	The login suffix that distinguishes users from this directory from another one. If more than one user directory is specified, then there may be name clashes on the login names. If this is the case users must enter a *full* login, which is the normal login succeeded by a “@” and the Full login suffix (e.g. username@local). If left blank the value “local” will be used.

Finalize Installation



Click Configure to save the configuration and generate database and G/On Client Software packages.



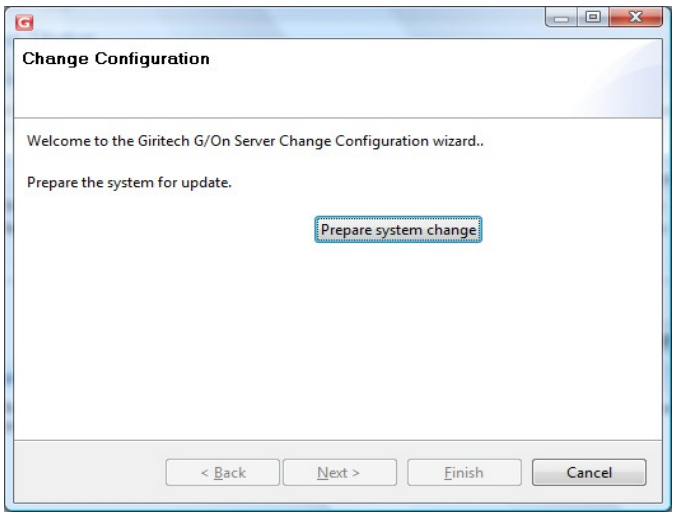
If no errors occur, you will be able to click Finish to exit the Wizard and go to the Configuration Status screen. If any errors occur they will be shown immediately after the “Finalize Installation” title.

Change Wizard

The Change Wizard is used for changing information for the currently installed system. The Wizard is started by clicking the Change Configuration button in the Main Status Window.

The Change Wizard has much the same structure as the Installation Wizard. On the first page, a “Prepare Change” job has to be run. The Prepare Change job reads the current settings from ini-files, so they can be presented in the following steps of the wizard.

On the following pages, configuration information can be entered and on the final page, the change is finalized. Here is a screen shot of the first page:



The following pages are the same as or similar to those of the Installation Wizard, so only differences from that wizard is described in this section. Please refer to the *Installation Wizard* chapter on page 56 for the remaining information.

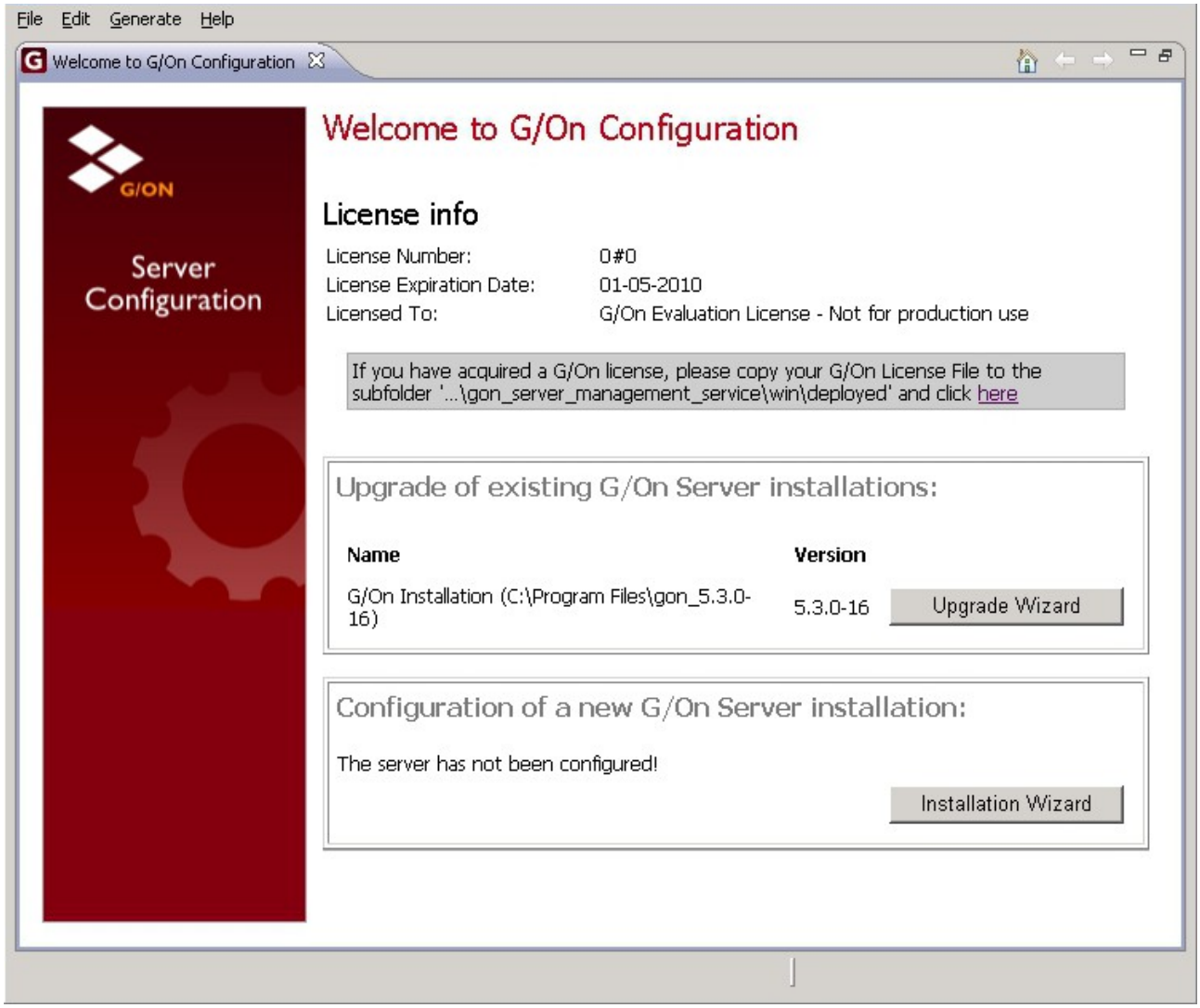
Database Setup

In the SQL Server and MySQL set-up an extra *encoding* field is available. The database encoding is detected and set automatically during installation but it can be changed here if necessary.

Note: it is not possible to create a new database using the Change Wizard. The database entered in the *database* field is assumed to be a previously installed G/On database. Entering the name of an empty database will result in a database missing essential set-up data.

Upgrade Wizard

The Upgrade Wizard is used for upgrading a previously installed G/On System to the same version as that of the G/On Server Configuration tool being used. When starting the Server Configuration tool, it scans the machine for existing G/On installations, and if any are found, they are presented on the Welcome page, each with a button to start an upgrade from that version:



The Upgrade Wizard has much the same structure as the Installation Wizard: On the first page a “Prepare Upgrade” job has to be run, on the following pages configuration information is entered and on the final page the upgrade is finalized. Note that depending on the upgrade there may not be any pages between the “Initialize” and “Finalize” page.

If the system being upgraded from uses an SQL Server or MySQL database, the default during upgrade is to make a new database instance, named after the G/On version, e.g. gon560. However, the upgrade wizard allows the administrator to choose another name or even the name

of the old database instance, which in this case will be overwritten.

Note: that during the upgrade, the system from which the upgrade is made will not be affected (except when it is deliberately chosen to re-use an existing database instance). You will need to stop the services of the “old” version manually, and uninstall it manually, if you desire to remove it.

Note: Note also, that additional GPM files that may have been added to the previous version after it was installed are not automatically copied to the new version during an upgrade. This includes, e.g., the package with the secure desktop linux image, and also other packages, which the customer or partner has added.

Upgrade memory issue

Upgrading to 5.5.1 or later from 5.5.0 or earlier may cause problems in some installations where the memory usage of the server during upgrade caused the server to crash. This problem was been fixed in 5.5.1, but upgrade from earlier versions may still cause problems, because the configuration server in the old version is used to make a backup of the system as part of the upgrade. If you have problems upgrading to 5.5.1 or newer versions from 5.5.0 or earlier versions, please use the following steps.

Try creating a backup of the old system manually

If it is possible to create a backup manually you can upgrade from that. Follow the procedure described in “G/On Setup and Configuration” to create a backup of the old version. If creation of the backup is successful, then you should copy or move the folder containing the backup from the backup folder of the old system to the backup folder of the new system. Then restart G/On configuration and the backup should appear as one of the systems usable for upgrade. Choose the backup and follow the standard procedures for an upgrade.

Prune the database

The main reason for the memory usage failure are database tables containing access log information. With the release of 5.5.0 we have also released an SQL script, which will remove access log entries older than a specified period of time (e.g. 3 months). If you want to keep these entries in the old system, then you can backup your database or create a copy before running the script. If you want to keep the entries in the upgraded version, then this solution is not applicable. Whether or not pruning the database will solve the problem is of course dependent on how much data is deleted, which again is dependent on the length of the period from which access log data are kept and the amount of user activity in that period.

From version 5.5.1 and forward it is possible to prune the database using a system command. See the Advanced Setup Topics section for details.

Create backup using the new system.

In 5.5.1 a special command line option has been added to the G/On configuration server, which enables the server to create a backup of a previous version without memory usage issues. However, in order for the backup to work as a foundation for an upgrade, some manual steps needs to be performed first. Use the following procedure in order to upgrade using this method:

1. Edit configuration files: In the old system some default settings needs to be specifically set in the configuration files in order for the new configuration server to use them. As a safety precaution you should create a copy of the files before editing them. Note that you may need administrative rights to edit the files. Open the files *gon_server_config.ini* and *gon_server_management.ini* in an editor. In each file you should uncomment settings not explicitly set. Here is an example:

   ```
   [log]
   # enabled = True
   enabled = True
   # rotate = True
   # type = text
   # verbose = 1
   verbose = 0
   # file = gon_server_management.log

   #[license]
   # filename = ./deployed/gon_license.lic
   ```

   All '#' and blank spaces at the beginning of lines should be removed, except for the settings which are already there, like e.g. the “verbose” setting in the example above. The example above should look like this after editing:

   ```
   [log]
   # enabled = True
   enabled = True
   rotate = True
   type = text
   # verbose = 1
   verbose = 0
   file = gon_server_management.log

   [license]
   filename = ./deployed/gon_license.lic
   ```

2. Backup old system: Open a command prompt (as administrator) and go to the subfolder *gon_config_service\win* in the new system. Here you should start the following command:

   ```
   gon_config_service.exe –backup_other_installation
   --backup_other_installation_path <path>
   ```

   where *path* is the path to the old system root folder, e.g. “*C:\Program Files\Giritech\gon_5.4.1-6”.* The command should produce a backup in the new system backup folder (*gon_config_service\win\backup).* Please check the log *backup_log.txt* in the backup folder to ensure that there were no errors during backup.

3. Start G/On Configuration in the new system. The backup should appear as one of the

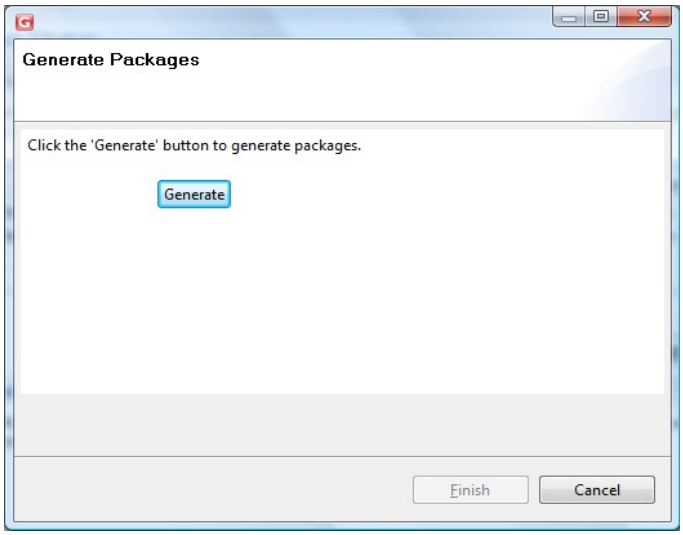
systems usable for upgrade. Choose this backup and follow the standard procedures for an upgrade.

Package Generation Wizard

This wizard generates GPM packages. Packages are also generated as part of the Installation Wizard, so this Wizard need only be run if package sources or definitions or package collections have been updated, added or removed.

The Wizard is started by either pushing the “Generate” button in the “Software Package (GPM) Wizard” section of the Main Status Window or by choosing Generate → Generate Software Packages (GPM) in the menu.

The Wizard consists of a single window:



Click Generate in order to start the task which generates the packages. If no errors occur, you will be able to click Finish to exit the Wizard. If an error occurs it will be shown immediately after the window title.

Menu

This section describes the options available in the menu.

File Menu

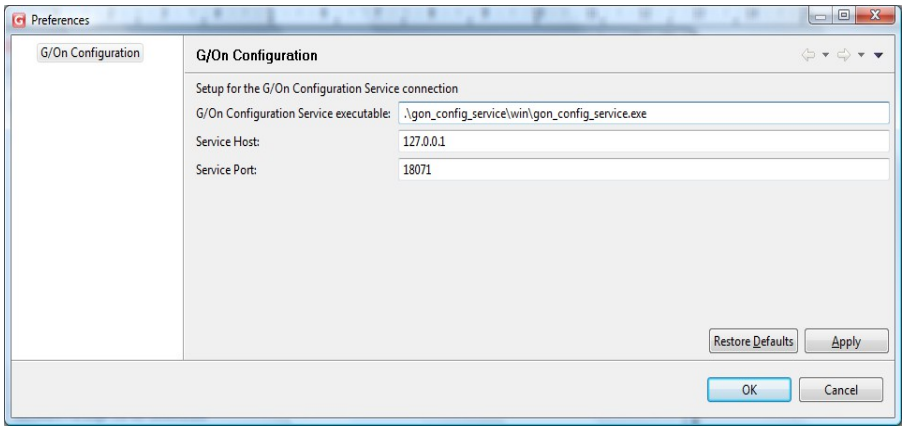
Quit G/On Configuration

Quits the program.

Edit Menu

Preferences

Opens the preferences window:



The following options are available:

G/ON CONFIGURATION SERVICE EXECUTABLE	**PATH TO THE UNDERLYING SERVER PROGRAM, WHICH DOES THE ACTUAL CONFIGURATION.**
SERVICE HOST	**THE SERVER NAME OR IP ADDRESS.**
SERVICE PORT	**THE PORT USED TO COMMUNICATE.**

Usually there is no need to change these settings, except perhaps the port number, if the default port number is unavailable for some reason.

Generate Menu

Generate Software Packages

Generates software packages. See Package Generation Wizard.

Generate Support Package

Generates a support package. See Support Package Generation

Help Menu

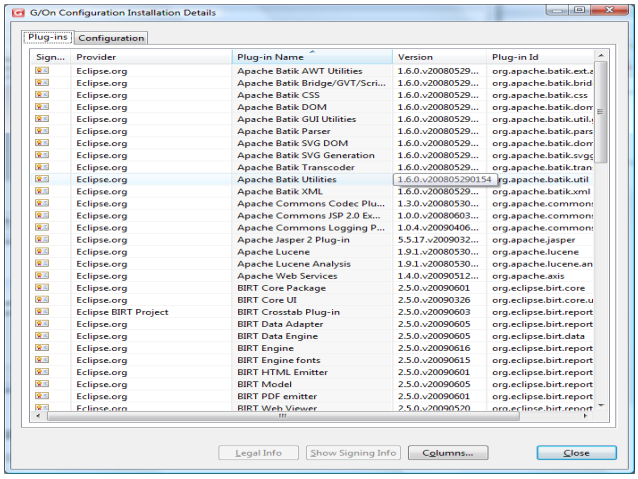
About G/On Configuration

Open the About Window. Apart from version and copyright information, you can access the Server Configuration client error log from here by following the steps below. Note that this log only pertains to the client (GUI) part of the Server Configuration Utility.

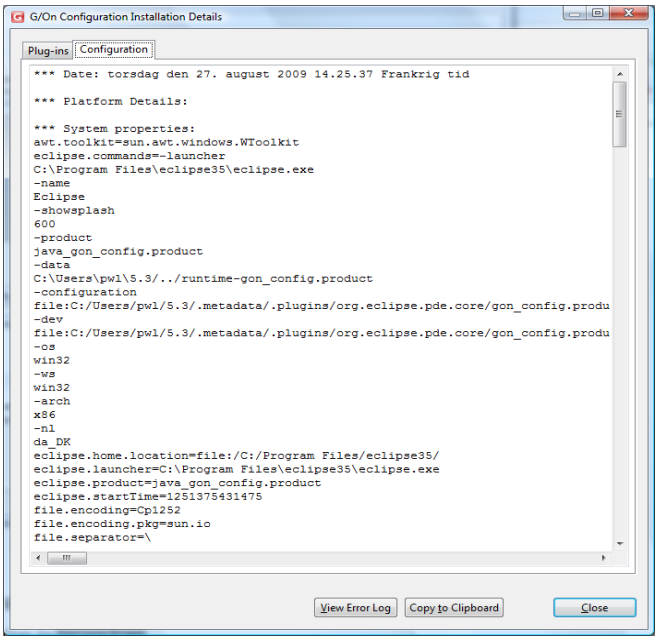
The error log is fetched like this:

1. In the About Window:

2. Click Installation Details. A window like this opens:



3. Select the Configuration tab. The Window changes to this:



4. Click View Error Log. The error log opens or you will get a window in which you can choose which program you want to use to open it. A browser like Internet Explorer or Firefox is usually a good choice for viewing. If you want to save the log, then open it in an editor like Notepad.

Welcome to the G/On Configuration

Opens the G/On Configuration Welcome Screen.

Advanced Setup Topics

Backup and Restore

All the configuration and operational data in a G/On installation can be backed up to a folder. This folder can then be used as input for restoring the G/On installation to the state that was backed up. It can also be used for moving the installation to a different location.

The backup folder includes both ini files and other configuration files.

The backup folder also includes xml dumps of the database tables.

Backup

To make a backup, run the command:

```
.\gon_config_service\win\gon_config_service.exe --backup
```

This will by default generate a folder like this, with all the backup files:

```
.\gon_config_service\win\backup\backup_5.4.0-16_2010-01-05_083639.507000
```

The name of the folder will indicate the G/On version, and data and time of the backup.

The following options can be used, together with the `--backup` option:

```
--backup_do_not_create_sub_folder
--backup_path=PATH
```

The first of these will place the backup files in `.\gon_config_service\win\backup` (not in a sub-folder). The second will place the backup files in the folder indicated (*PATH*).

Restore

To make a restore, run the command:

```
.\gon_config_service\win\gon_config_service.exe --restore
--restore_backup_path=PATH
```

where *PATH* is the full path to the folder containing the backup.

The following option can be used, together with the `--restore` option:

```
--restore_create_schema
```

This will force a restore of the database schema, in addition to restoring the data.

Initialization of Tokens

Initialization of Soft Tokens on USB-Key

Before the G/On Management Client can use a USB key as a soft token it has to be initialized. This can be done by creating the folder:

```
\gon_client\gon_init_soft_token
```

in the root of the USB-Key.

Initialization of Soft Tokens on HD

It is possible to prepare a soft token in a folder on the HD, and then copy it to a USB key. To do this, create a sub-folder of

```
.\gon_client_management_service\win\soft_token_root
```

containing the folders

```
gon_client\gon_init_soft_token
```

and it will appear in the G/On Management Client, just like a token, that can be enrolled, copied software packages to, etc.

The following example shows the folder structure needed for three soft tokens:

.\gon_client_management_service\win\soft_token_root\key_a\gon_client\gon_init_soft_token
.\gon_client_management_service\win\soft_token_root\key_b\gon_client\gon_init_soft_token
.\gon_client_management_service\win\soft_token_root\key_c\gon_client\gon_init_soft_token

When the soft token has been enrolled, and the desired software packages have been installed, it can be copied to the root of a USB key, and it will appear as if the token had been enrolled and installed directly.

Initialization of MicroSmart (USB) Tokens

Before the G/On Management Client can use a G/On MicroSmart (USB) token, it has to be initialized. This can be done by creating the folder

```
\gon_client\gon_init_micro_smart
```

in the root of the key.

Initialization of Computer User Tokens

Normally, Computer User Tokens are enrolled by using the procedure for field enrollment. But it is also possible to enroll a Computer User Token by running the G/On Management Client on the PC

where the Computer User Token has been installed.

Before the G/On Management Client can use a Computer User Token, the token has to be installed by using a G/On Client Installer program. How to generate such an installer program is explained in G/On USB & Computer on page 36. When the installer program has finished the installation task, it offers to options: “Launch” or “Exit” – choose “Exit”. At this point, the Computer User Token has been installed, so it will be recognized by the G/On Management Client.

No Initialization of Hagiwara Tokens

It is *not* necessary to initialize a Hagiwara token before the G/On Management Client can use it. However, the token must be formatted so it contains both a CDROM and a normal flash storage device.

Volume Label on Tokens

The linux “shortcut” (desktop icon) for starting the G/On client will only work if the volume label of the token is: G-ON. The same is true for the linux autostart feature.

Access notification by mail

A feature exists that sends a mail to the user's mailbox when he/she logins. This feature has been disabled by default, but can be configured and enabled by modifying the

```
[access_notification]
```

section in the ini-file:

```
.\gon_server_management_service\win\plugin_modules\ad\server_management\config.ini
```

The G/On Management Server need to be restarted in order for the configuration to be activated.

Advanced User Setup

Users are drawn from the user directory plugins, i.e. the Active Directory (AD), LDAP or Local Windows Users plugins. Each user must have a unique login in G/On. This fully qualified login is constructed as `<login>@<directory name>`, where `login` is the users login name or initials and `directory name` is a (unique) name from each plugin: For AD it is the domain DNS, for LDAP it is the specified directory name and for local Windows users it is always *local*. It is by default possible to log in using only the login part, provided that there is not a user in another directory with the

same login. In the latter case a fully qualified login is necessary for the user to log in.

User and group limit in G/On Management

In order to improve performance in G/On management there is a limit on how many users/groups are retrieved from the server. These limits can be configured manually by editing the `gon_server_management.ini` file. The options are called `users_returned_limit` and `groups_returned_limit` and has a default value of 500. Setting the limit to 0 is interpreted as no limit. Setting the limit to a negative number means that no users/groups are fetched initially, when the user/group pane is opened. When a search string is entered a the (positive) value of the setting is used as a limit for how many elements to show.

Note that the user directory may also have a limit on how many users/groups that can be fetched in one query (see the section on AD and LDAP plugins). The limits for G/On Management should always be set to something less than this limit in order for searches in G/On Management to work properly.

Require fully qualified login

It is possible to configure the Gateway server to always require a fully qualified login. This can be useful in a multi user directory setup, where user login in one directory should be independent of any users with the same name in other directories.

The option must be set manually by editing the *gon_server_gateway.ini* file. Add the line:

```
require_full_login = True
```

in the authorization section in order to require fully qualified login.

If the option is set and a user enters a login without a '@' in it, he/she will be presented with a new login prompt demanding a fully qualified login.

LDAP and Active Directory plugins

G/On supports User Directory connections using LDAP and Windows API to Active Directory (AD). In this section the requirements to supported User Directories are described along with a section regarding which plugin to use for connecting to AD.

LDAP to eDirectory

Requirements:

- IP or DNS address to an eDirectory server.
- User DN and password for an eDirectory user with browse rights OR
- Anonymous access set-up in eDirectory with a proxy user having browse rights
- Using SSL communication is highly recommended if the communication between the G/On and eDirectory server is visible from other machines. SSL communication requires a server certificate. A description on how to create a certificate can be found here: http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.itame.doc_6.0/rev/am60_install167.htm

LDAP to AD

Requirements

- IP or DNS address to an AD server.
- User DN and password for an AD user
- AD users should have permission to see their group memberships
- In order to enable password change, SSL communication must be used. SSL communication requires a server certificate. There are several descriptions from Microsoft on how to create such a certificate. One can be found here: http://support.microsoft.com/default.aspx?scid=kb;en-us;321051. Note even without password change, using SSL is highly recommended if the communication between the G/On and AD server is visible from other machines.

Limitations

- By default, a maximum of 1000 users/groups can be fetched by an LDAP query to AD. This means that a maximum of 1000 users/groups is available in G/On Management. You can however use the search functionality in G/On Server Management in order to find the users/groups you are looking for. The limitation is caused by the AD property "MaxPageSize", which can be altered using the `ntdsutil.exe` tool. See http://support.microsoft.com/?kbid=271088 for a description on how to change an AD property using this tool.
- Locking out a user after a number of failed login attempts (usually 3) does not work when logging in to AD using LDAP.

Native AD

Requirements:

- The server belongs to the domain OR
- The server belongs to another domain from which an outgoing trust has been set up to the domain. The trust type can be both forest or external.
- AD users should have permission to see their group memberships

Note that in order to create an outgoing trust, the trust has to be verified as an incoming trust in the other domain by a domain administrator. This can either be done by providing domain administrator credentials for the other domain during creation or by creating an incoming trust in the other domain using a shared trust password. Check Active Directory documentation for further details.

By default this plugin has the same 1000 users/groups limitations as described in the the LDAP to AD section. This limit can be overridden in the AD plugin configuration (see page 67).

LDAP to other directories

There are many LDAP directories apart from the ones described here (e.g. Apache, OpenLDAP, Siemens DirX,...), which probably work with G/On as well. We have, however, not conducted thorough testing against these directories, and therefore cannot include them in supported LDAP directories. Note also that property names and property usage can vary from one directory to another, so in order to connect to one these directories, some of the property names and queries used may have to be changed. This is also possible, but requires manual edit of LDAP configuration files. Please contact Giritech support for more information about this.

LDAP and SSL

In order to use SSL communication between the G/On server and LDAP directory server you only need to:

1. Make sure your LDAP server supports SSL communication (for AD this requires a certificate installation)
2. Check the “Use SSL” box in LDAP configuration.

However, if you want the G/On server to verify the LDAP server as well, you have to create a client certificate and specify the path to it in LDAP configuration. Whether this is necessary depends on how the servers connect, more specifically it depends on whether the G/On server can trust that it is talking to the right LDAP directory.

LDAP AD vs. Native AD

Since you can connect to AD using both an LDAP and a native plugin, the question of which one to use naturally arises. In order to help with this question we give a list of pros and cons of using the plugins.

LDAP pros and cons

Pros

- Server does not need to be on the domain
- Possible to connect to multiple unrelated AD's.
- Runs on Linux server

Cons

- Probably needs SSL communication, which may complicate the configuration phase.
- Default query limitation to 1000 users/groups
- Subject to changes by Microsoft of LDAP support in AD.

Native pros and cons

Pros

- Easily configured if G/On server is on the domain
- Performs "real" AD login using Windows API's, which we may be able to use for extending functionality in the future, like e.g. Kerberos Single Sign On.
- Better error messages.

Cons

- Requires that G/On server belongs to the domain.
- Dependent on trust relationships in order to support multiple AD's

Installing Additional Gateway Servers with a Gateway Installer

In this section we describe how to install additional Gateway Servers. Note that there is a license restriction on the number of Gateway servers and Client Connect Addresses,. Also note that this setup requires use of SQL Server database. The setup is partially manual, and thus requires some technical know-how.

Minimum Platform requirements

Server OS: Windows Server 2003

DBMS: SQL Server 2005

Before you start

First make sure that you have completed the following preparations:

1. Install the Nullsoft scriptable install system on the G/On server. Get it here: http://nsis.sourceforge.net/Download
2. Specify the client connect port and server addresses using the Change Wizard, if this has not already been done.
3. Make sure that SQL Server accepts remote connections and that it is reachable from the machine on which you want to make the installation. In order to check this you can try creating an ODBC connection to the server. In SQL Server 2005 remote connection is enabled by starting the “SQL Server Surface Area Configuration” tool, click on the “ Surface Area Configuration for Services and Connections” link and in the tool that opens enable remote connections for the server and possibly also start the “SQL Server Browser” service and set it to start up automatically, if this has not already been done. You should also check the SQL Server instance properties, on which remote connections can also be disabled.

Create Gateway Server installer

Use the Nullsoft installer (NSIS) to generate the G/On client installation program, as follows:

On Windows Server 2003, do not start the NSIS program. Simply right-click on:

```
distribution\gon_server_gateway_service_installer\win\nsis\gon_server_gateway
_service_installer.nsi
```

and select Compile NSIS Script.

On Windows Server 2008, start the NSIS program with Run as administrator. Then choose

Compile NSI scripts and File > Load Script... and then specify:

```
distribution\gon_server_gateway_service_installer\win\nsis\gon_server_gateway
_service_installer.nsi
```

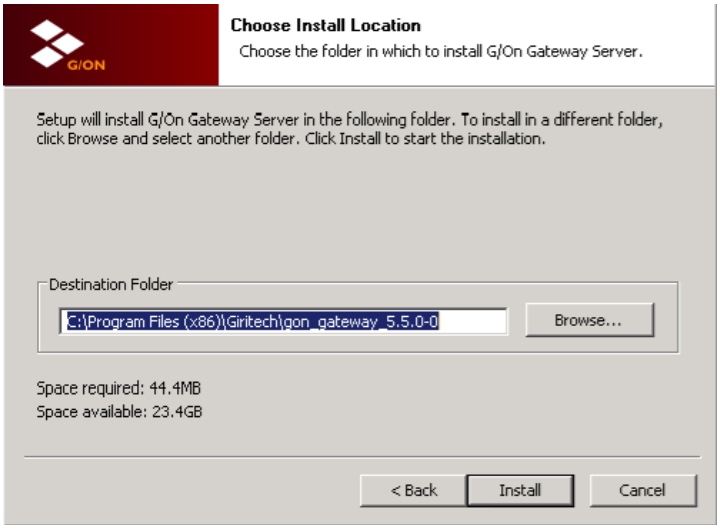
The resulting client installation program is placed here:

```
distribution\gon_server_gateway_service_installer\win\gon_server_gateway_serv
ice_installer.exe
```

Install Gateway Server

Copy the Gateway Server installer to the server machine on which it should be installed. Start the installer, and you will see this window:

Choosing “Next” brings you to a page where you should agree to the License Agreement. After that you can choose installation location:



Choose the destination folder of your choice and press “Install”. The installation will begin. When the installation is finished a final window will appear. Just click “Finish” to end installation.

Completing installation.

After running the Gateway Server installer there are a few steps (some optional) that needs to be performed manually.

1. **Edit the Port number:** If the gateway server has been installed on the same server machine as the standard Gateway Server or another installed Gateway Server you need to change the port number on which the server listens for client connections. The server is installed with the same port number as the standard Gateway Server. In order to change the port number you should open the file *gon_server_gateway_local.ini* in the folder *<installation folder>**gon_server_gateway_service**win* in Notepad or another editor of your choice. Depending on the operating system and installation folder you may need to run the editor as Adminstrator. Add the following two lines to the file:

   ```
   [service]
   port = <port number>
   ```

   and save the file.

2. **Name the server (optional):** In order better to distinguish servers in the “Gateway Servers” view in G/On Management it is possible to give each Gateway Server a name. In order to do so you should open the file *gon_server_gateway_local.ini* in the folder *<installation folder>**gon_server_gateway_service**win* in Notepad or another editor of your choice. Depending on the operating system and installation folder you may need to run the editor as Adminstrator. Add the following two lines to the file:

   ```
   [service]
   title = <my title>
   ```

   and save the file.

3. **Start the service**: Open the “Services” console in Windows. A new Gateway Server service should be available. Start the service. Open G/On Management and check that the new server appears in the “Gateway Servers” view.

Note that once the service is started it is possible to restart it using G/On Management. A stopped service must , however, be started using the Windows Services console.

Upgrade of Separately Installed Gateway servers

During upgrade of a G/On Master Installation to a newer version, only the Gateway Server in the Master Installation is upgraded. Separately installed Gateway Servers must be re-installed using the steps described above, with one exception: After the Gateway server has been installed using the Gateway installer, in stead of editing the *gon_server_gateway_local.ini* file, you should simply copy it from the old Gateway server installation. Also remember to stop the old service before starting the new one.

Set-up instructions when migrating from SQLite

The following is a description on how to migrate from using the internal SQLite database to SQL Server..

1. Create the database you want to migrate to in the SQL Server Management tool (or another tool of your preference).
2. Stop the G/On services, and do a backup (see the instructions regarding backup on page 83).
3. Open the file *.\gon_server_management_service\win\gon_server_management.ini* in an editor (e.g. Notepad). If the server is Windows 2008 or newer you have to run the editor as Administrator. Now, find the following:

   ```
   [db]
   …
   type = sqlite
   …
   ```

   and change it to:

   ```
   [db]
   …
   type = mssql
   …
   ```

4. Start G/On Server Configuration ans start the Change Wizard. On the first configuration page (after the “Prepare system” page) you should get a SQL Server database setup page.
5. Enter the database specification details. Note that you need to specify the database encoding. If you don't know what encoding to use then check the section below. Finish the wizard (without making further changes). Close G/On Server Configuration.
6. Create a new backup. Copy all the files in the folder *.\database* from the backup created in

step 2. to the new backup (overwrite the existing files).

7. Restore the new modified backup (see the instructions regarding restore on page 83).
8. Restart the G/On server services.

Finding database encoding

You can find the encoding of the database, by SQL queries like these:

First try:

```
SELECT databasepropertyex(<database name>, 'Collation')
```

If this returns the value “NULL” try:

```
SELECT SERVERPROPERTY('Collation')
```

You should get a collation name, e.g., “Danish_Norwegian_CI_AS”

Use the collation name in the query:

```
SELECT collationproperty(<collation name>, 'CodePage')
```

You should get a codepage number, e.g. “1252”.

For G/On, you must add “cp” before the code page number, to get the encoding name, e.g. “cp1252”.

Alternatively you can install a (temporary) G/On server and run the installation wizard. Aftere running the installation, start the Change Wizard. On the Database setup page you will find the encoding which has been automatically detected during installation.

Creating Custom Client Installers

On page 36 in the chapter *Token Configuration and Enrollment*, it is described how to make a basic client installer for installing G/On on a USB token or as a Computer User Token on a PC. The following describes some options available to the client installer.

Changing which Packages are included in the Client Installer

When the installer is generated, all the gpm packages in the following folder are automatically included in the installer:

```
distribution/gon_client_installer/win/nsis/gpms
```

So in order to get a package included in the installer, you can simply copy the gpm file to this folder. The gpm packages can be found in another folder:

```
config/gpm/gpms
```

Alternatively, it is possible to make sure that the newest versions of the desired packages are always copied automatically to the nsis/gpms folder, so they are included when the installer is generated next time. This is done by adding the names of the packages to this package collection:

```
config/gpm/gpmcdefs/dist_client_installer_win.gpmcdef.xml
```

Then all packages in this collection are automatically copied to the nsis/gpms subfolder, whenever packages are generated.

Options for the Client Installer

The Nullsoft program includes the following ini-file when it builds the client installer:

```
gon_client_installer\win\gon_client_installer.ini
```

When a users starts the client installer, it behaves according to options in this ini-file. The options and their default values are:

```
[discover]
desktop_enabled = True
token_enabled = True


[keys]
generate_keypair_enabled = True
```

By default, the client installer offers the user a choice of installation destination: either on a (new) Computer User Token, or (if inserted in the PC) on a hardware token (G/On USB). However, it is possible to make installer that only offers to install on either Computer User Token or on G/On USB:

The option `desktop_enabled` controls whether the user will be presented with a choice of installing (and re-installing) Computer User Tokens.

The option `token_enabled` controls whether the user will be presented with a choice of installing (and re-installing) G/On USB tokens (MicroSmart, Hagiwara, Soft Tokens).

Furthermore, the option `generate_keypair_enabled` controls whether a new keypair is generated when (re-)installing a token. Set this to False, if you want to make an installer for updating the software and connect information on a token, without giving the token a new identity. This can, e.g., be used for upgrading the software (and connect info) of a selected group of tokens.

Changing which address the installed client connects to (the Connection Info)

The Nullsoft program includes the following file when it builds the client installer:

```
config\deployed\gon_client.servers
```

This file describes the addresses that the installed client will connect to. If you follow the steps described in the following, you can change this file temporarily, and generate a client installer including this changed file. *But you must complete all the steps, as described.* Otherwise you run the risk of other tokens also getting their connect addresses updated to match the changed file, and you will loose the automatic synchronization of client connect addresses with the contents of the license file. The steps are:

1. Stop the management server service
2. If there is a gateway server service running on the same machine as the management service, stop this gateway server service. Note that this will mean that any user session on this gateway service will be terminated, and no users can connect to this gateway service.
3. Make a copy of the file `gon_client.servers`
4. Make the desired changes to the file `gon_client.servers, using a text editor`
5. Generate the client installer
6. Restore the file `gon_client.servers` from the copy that you made in step 3
7. Start the management server service and the gateway server service

Special Settings for the G/On Gateway Server

Settings governing the reaction to DOS attacks

G/On can be configured to address two kinds of attacks: "Hanging connections" and "failed key exchanges", as described briefly in the following. The behavior under these types of attacks is governed by settings in the gon_server_gateway.ini file (with default values as indicated):

```
[security]
dos_attack_keyexchange_phase_one_high = 20
dos_attack_keyexchange_phase_one_low = 5
dos_attack_security_closed_high = 4
dos_attack_security_closed_low = 2
dos_attack_ban_attacker_ips = True
```

Hanging connections: If at least a given number of TCP connections as defined by `dos_attack_keyexchange_phase_one_high` have been established within a window of 10 seconds, and the clients have not communicated anything on these TCP connections, DOS attack mode is initiated.

Failed Key Exchanges: If at least a given number of TCP connections as defined by `dos_attack_security_closed_high` have been established within a window of 10 seconds, and the clients have communicated erroneously on these TCP connections, DOS attack mode is initiated.

DOS attack mode terminates when there in a window of 10 seconds have been less than a given number of hanging connections (`dos_attack_keyexchange_phase_one_low`) and less that a given number of failed key exchanges (`dos_attack_security_closed_low`).

With the setting `dos_attack_ban_attacker_ips = True`, all the client IP addresses that were "involved" in the connections that led to the initiation (or continuation) of DOS attack mode get banned until DOS attack mode is terminated.

Pruning the database

The database contains a lot of access log information, which is not erased automatically. Normally this does not influence the system performance, but it can be inconvinient in connection with e.g. automatic backup or system upgrades being very time consuming. It is possible to remove all access log information prior to a specific data using the ”prune_access_log” system command:

```
.\gon_config_service\win\gon_config_service.exe --prune_access_log <date>
```

This will remove all access log information dated before the specified date. The date must be of the form YYYY-MM-DD.

Example: The command

```
.\gon_config_service\win\gon_config_service.exe --prune_access_log 2013-09-07
```

will remove all access log information dated before 7th of September, 2013.

Troubleshooting

Error “Unable to connect to local service” shown at start-up	The underlying server program has not been started correctly. This typically happens on Windows server 2008, if G/On Configuration has not been started with “Run as administrator”. Also check that the preferences (Edit → Preferences) are set up correctly. Check the log file `.\gon_config_service\win\gon_config_service.log` for any errors.
"Error: Unable to generate checksum for ..."	If the G/On Management client is running on the server, Software Packages (GPM) Generation in the G/On Configuration client will give the following error: "Error: Unable to generate checksum for ..." It happens because one of the packages to be generated contains the Management client files, and one of these gets locked, when the Management client is running. Workaround: Exit G/On Management client (if running on the server), before generating packages.

FAQ

How to change the external address or port of the G/On Gateway Server?

Question: I have set up the server using the wizard in the G/On Server Configuration program. But the client connect address/port that I specified for the G/On Gateway Server was not correct. How can I change that?

Answer: If you are using a demo license, the fields are open so you can change these settings. If you are using a proper license, please obtain a new license with the desired address and port.
Note: G/On version 5.5 and later can push out changes in client connect addresses to the client, which can then update the token. However, this requires that the client can still connect to the old address. So the Gateway Server should be kept listening on the original address, until all tokens have been updated.

How to install a changed license?

If you have acquired a changed license file, you should place it in the folder

```
.\config\deployed
```

Thereafter, you may want to start and complete the Change Wizard, in order to take advantage of the changes in your new license file, e.g., to use new/changed client connect addresses.

Note: If you have changed client connect addresses and/or ports, you also need to:

1. Generate packages
2. Regenerate the Client installer and Gateway installer (if you are using these)

Management Reference

Basic Concepts

The purpose of G/On is to securely connect authenticated users to authorized applications. To prepare for this, the manager of a G/On system must define:

1. Which applications can be authorized,
2. Which Authentication Factors are sufficient for establishing the identity of a User
3. Which groups of Users can be authorized to use which applications, and
4. Under which circumstances an authorized application can be allowed to be used.

An application, which has been authorized for a given user, may appear in the menu for that user. In G/On terminology, the specification of an application is therefore called a *Menu Action*. Menu Actions are introduced below.

Which Authentication Factors are sufficient for establishing the identity of a User, and which groups of Users can be authorized to use which applications under which circumstances are defined in G/On in terms of so-called *Decision Rules*. Decision Rules are introduced below, after the introduction to Menu Actions.

Menu Actions

G/On Menu Actions are divided into the following types:

Port Forward. Creates one or more Port Forwards from the client side to the server side, and may start a client side command.

RDP Connection. Port forward with built-in RDP protocol inspection. Does single sign-on on the server side, and reacts to re-direction messages from Remote Desktop Connection Broker (Terminal Services Session Broker).

Citrix XML Interface. Enables Citrix applications, published through the Citrix XML interface, and makes them available as individual menu items in the G/On Menu, without having to install anything on the client PC.

Citrix Web Interface. Creates a http proxy connection between a browser on the client side and a Citrix Web server on the server side. The http proxy implements single sign-on the server side and launch of the Citrix client on the client side, without having to install anything on the client PC or browser.

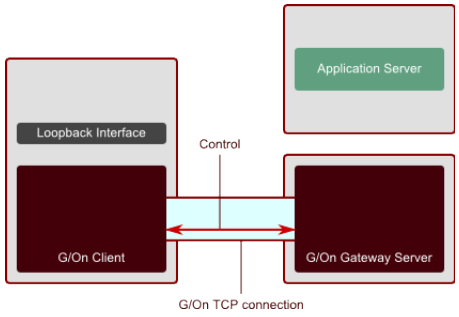
HTTP and SOCKS Proxy. Establishes a connection to the built-in proxy in the Gateway server. The proxy can handle HTTP and SOCKS proxy protocols, and can also function as a transparent

HTTP proxy. Menu actions of this type include specification of a server whitelist and may include configuration data for HTTP single sign-on.

G/On Internal. Starts the build-in G/On actions for installing, updating or removing Packages, or doing field enrollment of Tokens, or changing password.

Wake-on-LAN. Sends wake-on-LAN packets from the G/On server to wake up a machine with a given network MAC address.

Each of the types is described more in detail in the following. In each description, we assume that the User has already established a session, and has been authorized to carry out the Menu Action. This means that the G/On client and G/On Gateway server have established a TCP connection between them, and through this TCP connection, the client and server exchanges control data.

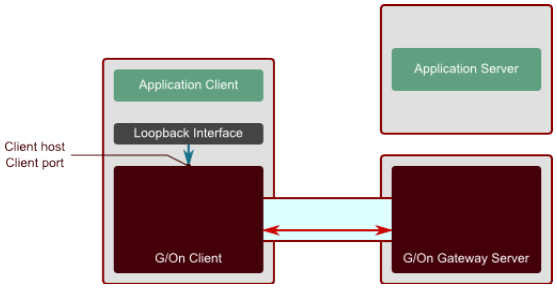


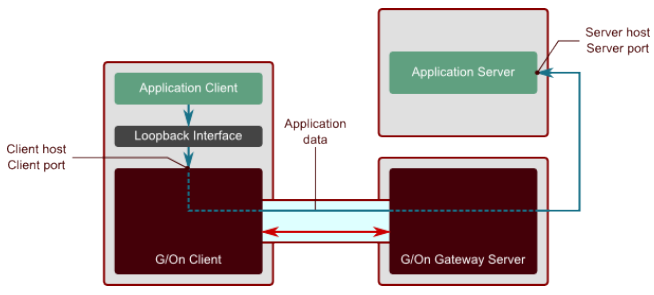
Port Forward Menu Actions

When the User selects the Menu Action, the Menu Action name is communicated from the client to the server, as control data. The server then looks up the definition of this specific Menu Action and instructs the client to do its part:

- To start listening on a given address and port (Client host, Client port).
- To start a given application client with given parameters. The parameters can, e.g., include the address and port which the client must communicate on,
  to reach the application server through G/On.

When the application client communicates on the given address and port, the G/On client forwards this communication through its TCP connection to the G/On server, and the G/On

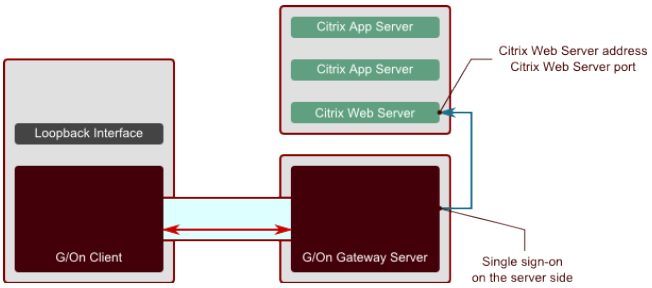




server forwards the communication through a connection to the application server at an address and port, that was also defined by the Menu Action (Server host, Server port).

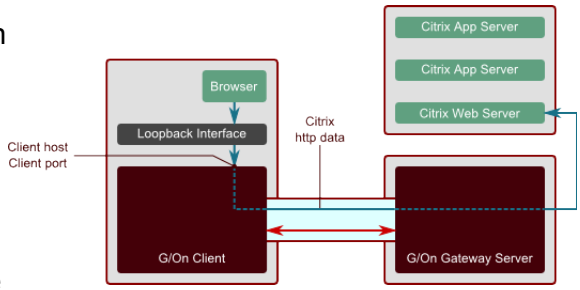
Citrix Web Interface Menu Actions

When the User chooses the Menu Action, the Menu Action name is communicated from the client to the server, as control data. The server then looks up the definition of this specific menu action, and contacts the Citrix Web Server at the specified address and port. The Web server responds by sending a web page with a login form, and the G/On server fills in the User Name and Password, and posts the form back to the web server.



The Citrix Web Server now initiates an User session, and sends a web page with icons for the Citrix enabled applications.

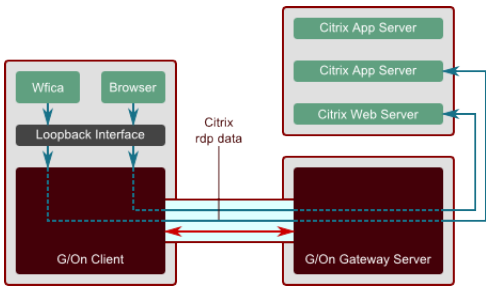
The G/On server forwards this page to the G/On client, which starts a browser. The start URL points to the G/On client itself, and allows the G/On client to serve the web page to the browser, so it can display the page to the User.



When the User clicks on one of the icons on the web page, the browser sends a request to get the .ica file, which describes how to start the corresponding application through Citrix.

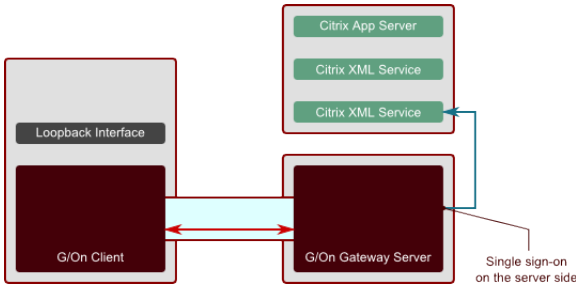
The request is forwarded through the G/On client and server to the web server, which responds by sending the .ica file.

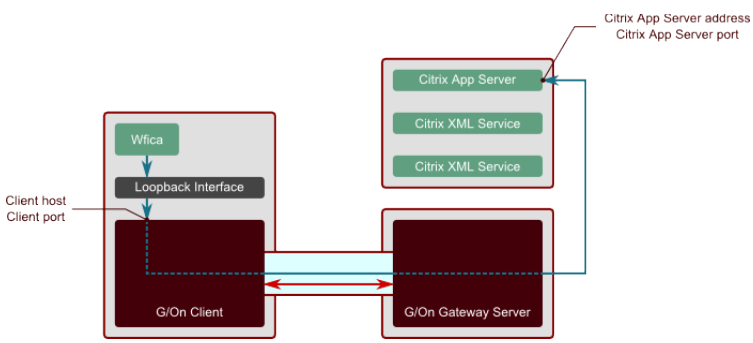
The .ica file is inspected by the G/On server in order to identify the address and port of the Citrix application server. The .ica file is then forwarded to the G/On client, which starts a Citrix client (wfica), and gives it the .ica file, however with an modified address and port, so the Citrix client will contact the Citrix Application server through G/On rather than directly.
When the Citrix client communicates on the modified address and port, the G/On client and server forwards this communication to the original address and port of the Citrix application server.



Citrix XML Interface Menu Actions

When a User is authorized for a Menu Action of this type, the G/On server creates a connection to a Citrix XML service and logs in as this User. Each of the Citrix applications published for the User is then presented as a G/On Menu item. When the User chooses one of these Menu items, the Menu item name is communicated from the client to the server, as control data. The server then looks up the definition of this specific Menu Action, and contacts the specified Citrix XML service, to get input for generating an .ica file that describes how to start the corresponding application through Citrix. The template for the .ica file is specified as part of the menu action. The .ica file is then forwarded to the G/On client, which starts a Citrix client/Citrix receiver (wfica), and gives it the .ica file. The .ica file includes information about a local address and port, so the Citrix client will contact the Citrix Application server through G/On rather than directly.





When the Citrix client communicates on the local address and port, the G/On client and server forwards this communication to the original address and port of the Citrix application server, obtained from the Citrix XML service.

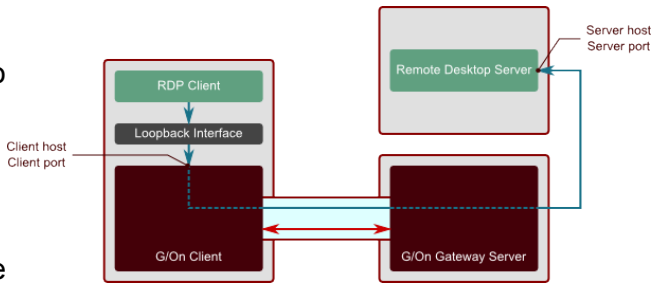
RDP Connection Menu Actions

When the User chooses the Menu Action, the Menu Action name is communicated from the client to the server, as control data. The server then looks up the definition of this specific Menu Action and connects to the specified Remote Desktop server (Terminal server).

The G/On server then instructs the client to start listening on a given address and port, and also to start the specified RDP client. When the RDP client communicates on the given address and port, the G/On client forwards this communication through its TCP connection to the G/On server, and the G/On server forwards the communication to the Remote Desktop server.

If the Remote Desktop server uses the Remote Desktop Connection Broker (Terminal Services Session Broker), it may respond that another server should be used. In that case, the G/On server connects to the other server.

When the Remote Desktop server asks for User Login, the G/On server provides User Name and Password on behalf of the User.



HTTP and SOCKS Proxy Menu Actions

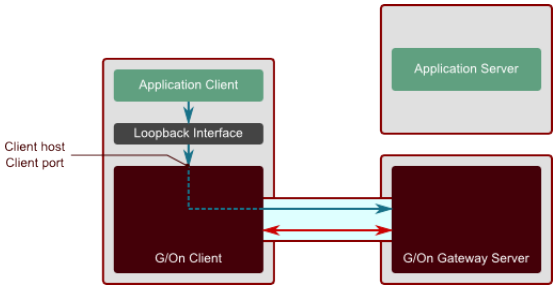
When the User chooses the Menu Action, the Menu Action name is communicated from the client to the server, as control data. The server then looks up the definition of this specific Menu Action and instructs the client to do its part:

- To start listening on a given address and port (Client host, Client port).
- To start a given application client with given parameters. The parameters can, e.g., include the address and port which the client must communicate on, to reach the HTTP/SOCKS proxy built into the G/On server.

When the application client communicates on the given address and port, the G/On client forwards this communication through its TCP connection to the G/On server.

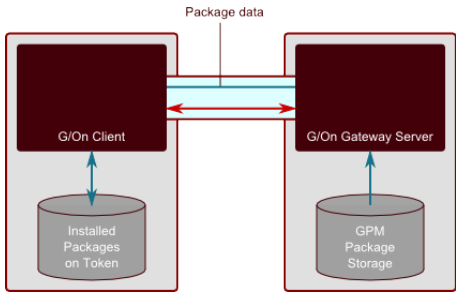
If the client communicates using plain HTTP, the communication will then be routed through the transparent HTTP proxy in the G/On Server, and forwarded to an application server address and port specified in the Menu Action.

If the client communicates using the HTTP proxy protocol or the SOCKS proxy protocol, the communication will be routed to the built-in HTTP or SOCKS proxy in the server. The proxy will carry out the commands in the proxy protocol for establishing HTTP or TCP connections to given addresses and ports – however only if the addresses and ports are included in the whitelist specified in the Menu Action.



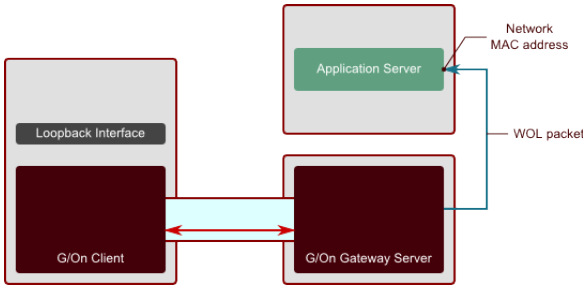
G/Update Menu Actions

When the User selects the Menu Action, the Menu Action name is communicated from the client to the server, as control data.
The server inspects the GPM package storage to find out which packages are available, and this is compared with the information about which packages are currently installed on the token.
Depending on the definition of the G/Update Menu Action, the User is then presented with a wizard for either installing, updating or removing packages, and if needed, packages are downloaded from the server to the client.



Wake-on-LAN Menu Actions

When the User selects the Menu Action, the Menu Action name is communicated from the client to the server, as control data.
The server then looks up the definition of this specific menu action and then sends Wake-on-LAN packets to the device with the network MAC address, which is specified in the definition of the Menu Action.



Rules and Elements

A G/On Decision Rule states that if given premises hold, then a given conclusion also holds. The Rules are written in this form (the number of premises can be 0, 1 or more):

PREMISE 1, **PREMISE 2** ✱ **CONCLUSION**

where both premises and conclusions have the form:

ELEMENT TYPE: *ELEMENT*

As an example, consider the following Rule:

TOKEN: *MICRO_SMART_0002*, **USER:** *BOB@GIRITECH.COM* ✱ **PERSONAL TOKEN STATUS:** *PERSONAL TOKEN*

In short, it can be read as: “If the micro_smart_0002 token is being used, and the User is bob@giritech.com, then we conclude that we have some known user with a Personal Token”.

Put differently, we can say that the rule registers the fact, that micro_smart_0002 is a Personal Token of bob@giritech.com.

Technically, the premise: Token: micro_smart_0002 is true, if the token micro_smart_0002 has been verified as being plugged into the client computer.

And the premise: User: bob@giritech.com is true, if the User of the client computer has entered a Name and Password that, according to the User Directory (e.g. Active Directory), establishes that the User is in fact: bob@giritech.com.

So, technically, the conclusion of the rule expresses that the current User has been authenticated with two factors: Password verified by the User Directory and Personal Token verified by G/On.

Combining Rules to make more complex decisions

The conclusion of one Rule can be used as premise for other Rules. For example, consider these two Rules:

TOKEN: *MICRO_SMART_0002*, **USER:** *BOB@GIRITECH.COM* ✱ **PERSONAL TOKEN STATUS:** *PERSONAL TOKEN*

USER GROUP: *EMPLOYEES*, **PERSONAL TOKEN STATUS:** *PERSONAL TOKEN*
✱ **AUTHENTICATION STATUS:** *AUTHENTICATED*

Assuming that the conclusion of the first Rule holds, this can be used “as input” to the second Rule. If we also assume that bob@giritech.com is a member of the User Group *Employees*, the second Rule then allows us to conclude that the current User is *Authenticated*.

Overview of the types of Elements in G/On

Token. Elements of this type are things that can be given to a User, and which the User can then present at a later time in order to confirm his or her identity. Some Tokens also have a capacity to hold client side software, such as the G/On client, application clients, and even a whole client side operating system.

User. Elements of this type are Users, registered in a User Directory.

Personal Token Status. There is only one, fixed Element of this type. The Element is called: Personal Token. It represents the fact that a known User (from a User Directory) has presented (one of) his Personal Tokens.

DME Approved Device. There is only one, fixed Element of this type. This Element is called: DME Approved Device. It represents the fact that a known User (from a User Directory) has logged in from a DME client, on a device which approved for use by this user, in DME.

User Group. Elements of this type are User Groups, registered in a User Directory

Authentication Status. There is one, pre-defined Element of this type. The Element is called: Authenticated. It represents the fact that a User has been properly authenticated. Other Elements can be defined in G/On, if needed.

Token Group. Elements of this type are groups of Tokens. They are defined in G/On.

G/On User Group. Elements of this type are groups of Users, defined in G/On (not in the User Directory)

Menu Action. An element of this type is a specification of an application, that may appear in the user's menu. See the overview of menu actions, above.

IP Range. An element of this type represents a range of IP addresses, as they may be observed by the G/On Gateway Server when a G/On Client connects. The range may concern either the client side or the server side address.

Operating System State. An element of this type represents observed properties of the state of the operating system, where the G/On Client is running.

Login Interval. An element of this type represents a time of day/day of week interval.

Zone. Elements of this type represent circumstances of a user session, that may be required before the user can be allowed to use given menu actions

Management Role. An element of this type represents a role that a G/On Manager may have, and the (limited) set of privileges that are needed for carrying out the management tasks of that role.

Overview of the types of Rules in G/On

Personal Token Assignment Rules register the fact, that a given Token is a Personal Token of a given User. The Rules have the type:

TOKEN **USER** ✱ **PERSONAL TOKEN STATUS**

User Authentication Policy Rules register policies for authentication. The Rules have one of the types:

UG **TG** ✱ **AUTHENTICATION STATUS**

where **UG** is either absent or one of:

- **USER GROUP**
- **G/ON USER GROUP**

and **TG** is either absent or one of:

- **PERSONAL TOKEN STATUS**
- **TOKEN GROUP**
- **DME APPROVED DEVICE**

Action Authorization Policy Rules register policies for giving access to Menu Actions. The Rules have one of the types:

AS **UG** **TG** ✱ **MENU ACTION**

where **AS** is either absent or:

- **AUTHENTICATION STATUS**

and **UG** is either absent or one of:

- **USER GROUP**
- **G/ON USER GROUP**
- **DME APPROVED DEVICE**

and **TG** is either absent or:

- **TOKEN GROUP**

Token Group Membership Rules register the fact, that a given Token is a member of a Token Group. The Rules have the type:

TOKEN * **TOKEN GROUP**

G/On User Group Membership Rules register the fact, that a User or all the Users of a group from the User Directory are members of a G/On User Group. The Rules have one of the types:

USER * **G/ON USER GROUP**

USER GROUP * **G/ON USER GROUP**

Management Role Assignments Rules register the fact, that a User or all the Users of a group from the User Directory have a given management role. The Rules have one of the types:

USER * **MANAGEMENT ROLE**

USER GROUP * **MANAGEMENT ROLE**

Zone Detection Rules register the conditions for being in a given zone. The Rules have one of the types:

IR **OSS** **LI** * **ZONE**

where **IR** is either absent or:

- **IP RANGE**

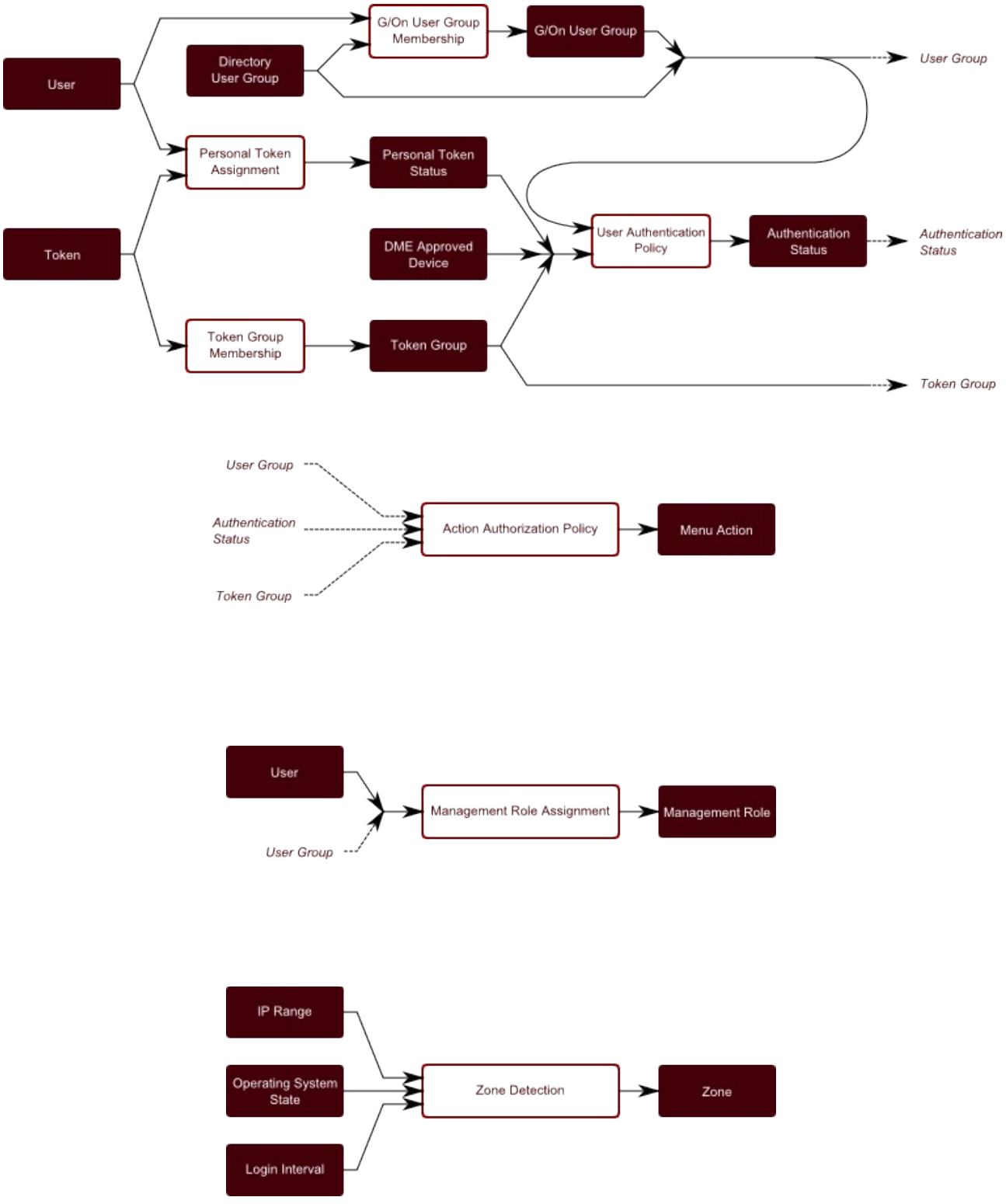
and **OSS** is either absent or:

- **OPERATING SYSTEM STATE**

and **LI** is either absent or:

- **LOGIN INTERVAL**

Overview of the way Rules fit together in G/On



The Rule Engine

User Sessions – Deciding which Menu Actions are Authorized and Active

The G/On Rule engine is used in the Gateway Server, where it starts each time a new User session is created.

First the engine examines all the Rules, which have been entered in the G/On Management interface, and checks all the basic premises, which occur in the Rules. For instance, when there are Rules with premises of type *User,* it asks the User management layer in G/On to present a Login dialog and verify User Name and Password in the appropriate User Directory.

Having checked the basic premises, the Rule engine then checks if any Rule has fulfilled *all* its premises. In this case, the engine registers the conclusion of the Rule. This conclusion may be the premise of other Rules, which now have all premises fulfilled, leading to new conclusions being registered, etc.

The process stops when no more new conclusions can be found. At this time, the Rule engine:

- Collects all the conclusions of type *Menu Action*, and registers them as the *Authorized Menu Actions* for the current User session.
- Collects all the conclusions of type *Zone*, and registers them as the *Active Zones* for the current User session.

Each authorized menu action is then considered:

- If it is not restricted to any Zones, the Menu Actions is marked as an *Active Menu Action*
- If it is restricted to one or more Zones, and at least one of these is an Active Zone, the Menu Actions is marked as an *Active Menu Action*
- If it is restricted to one or more Zones, and none of these is an Active Zone, the Menu Actions is marked as an *Inactive Menu Action*

All the authorized menu actions will be shown to the user, but the inactive ones will be marked as such, and the server will only carry out a menu action if it is active.

Management Sessions – Deciding which Management Roles are Active

The G/On Rule engine is also used in the Management Server, where it starts each time a new Management session is created. Here, it only works on the specified Management Role Assignments Rules, in order to determine which roles the current manager has.

The Management Client

The management client is a tool that, among other things, lets an administrator add Rules to the Rule engine and create Tokens for Users. After installation there should be a sub-menu in the windows start menu called 'G-On'. Navigate to the menu item labeled G-On Management and use this menu item to launch the Management Client.

If there are no G-On menu items, the Management Client program: gon_client_management.exe can be found in subfolders (win, win64) of the folder:

```
Program Files (x86)/Giritech/gon_<version>/gon_client_management
```

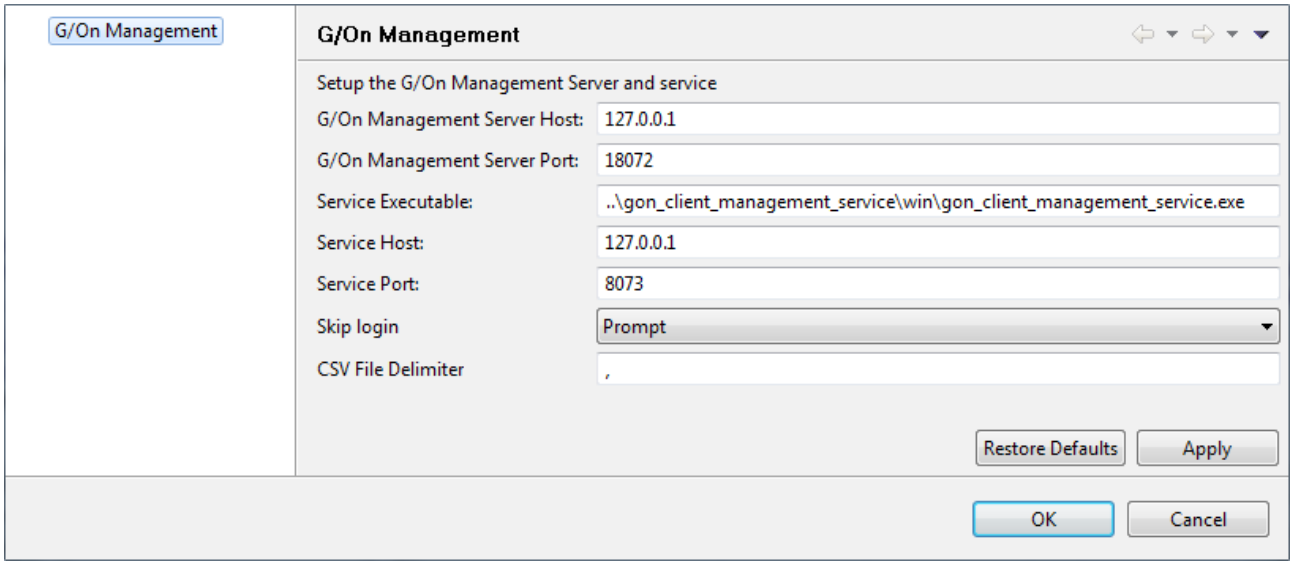
Choose win if you are on a 32 bit machine, and win64 if you are on a 64 bit machine.

Note: On Windows Server 2008, you must run the G/On Management Client as Administrator: Find the program in the Windows Start Menu, right-click on it, and choose: “Run as Administrator”.

The Management Client is separated into a number of different perspectives. Some of these perspectives are used for creating Rules. The perspectives used for creating Rules all have the same basic functionality. Adding Elements to the Rule engine always takes place in a Rule creation perspective. Special perspectives have been created for tasks that do not create Rules for the Rule engine. For example adding software to a Token or getting Reports on system usage.

Preferences

Preferences for the Management Client are used for setting connection and login settings.



The settings are:

- **G/On Management Server Host:** IP address or DNS name of management service host
- **G/On Management Server Port:** Port number management service listens on.
- **Service Executable:** Path to local management server tool executable.
- **Service Host:** IP address for the local management service. This should only be if, for some reason, it is not possible to set up a service on 127.0.0.1.
- **Service Port:** The port number used to communicate with the local service.
- **Skip login:** Setting for the login dialog. There are four possible values:
  - Always: The login dialog will not be shown at start-up.
  - Default: The “Skip login” box in the login dialog will be checked by default.
  - Prompt: Default setting.
  - Never: It is not possible to skip login.
- **CSV File Delimiter:** What delimiters the G/On Management should use when *Export Selected Rule(s) to CSV* is used.

Note: If, for some reason, it is not possible to access G/On Management to edit the preferences, it is possible to restore the default settings by removing the folder “workspace” located in the folder containing the G/On Management executable (<installation folder>\gon_client_management\win – or \win64 on the server).

Introduction to Perspectives

Perspectives are defined as a full window with a specific purpose. Most perspectives are used for defining Rules for the Rule engine. Additional perspectives are used for adding software to Tokens or display Reports on system usage.

The perspectives included in the Management Client are:

- **G/On User Group Perspective** adds User or Groups to local G/On User Groups via Rules.
- **Action Authorization Policy Perspective** sets up Authorization Policies via Rules.
- **User Authentication Policy Perspective** sets up Authentication Policies via Rules.
- **Personal Token Assignment Perspective** sets up User and Token links via Rules.
- **Token Software Management Perspective** adds software to a Token.
- **Token Group Management Perspective** adds Tokens to Token Groups via Rules.
- **Zone Management Perspective** defines special circumstances that may be used to restrict access to otherwise authorized actions.
- **Management Role Assignment Perspective** defines who can do what in the Management Client.
- **Menu Structure Management Perspective** orders User Menus via Tags.
- **Gateway Servers Perspective** manages running Gateway Servers and User Sessions.
- **License Management Perspective** shows information on license usage.
- **Reporting perspective** gets information on system usage.

Use the perspective bar to select another perspective. The perspective bar is by default set to include the most used perspectives when first using the management client. If the wanted perspective is not in the perspective bar, use the open perspective button, in the far left of the perspective bar, to open other perspectives.

The Perspective bar

The Perspective bar is used for changing the current perspective. When the Management Client first launches it will display buttons for the three most used perspectives. These are:

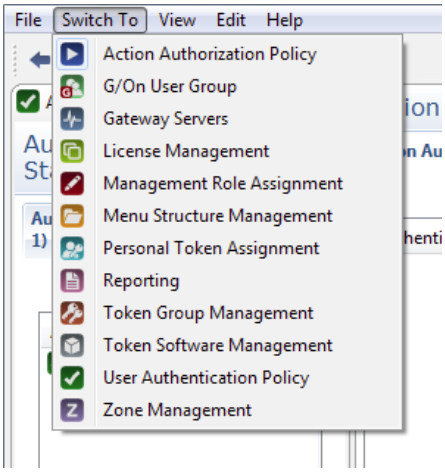
- **G/On User Group** – for defining the group of Users who are allowed to enroll a

Personal Token “in the field”

- **Personal Token Management** – for activating Personal Token assignments and thereby approving field enrollment requests
- **Action Authorization Policy** - for defining which Menu Actions are authorized for use by which User Groups, under which which circumstances

Selecting other perspectives

Choose the menu Switch To and then choose the relevant perspective.

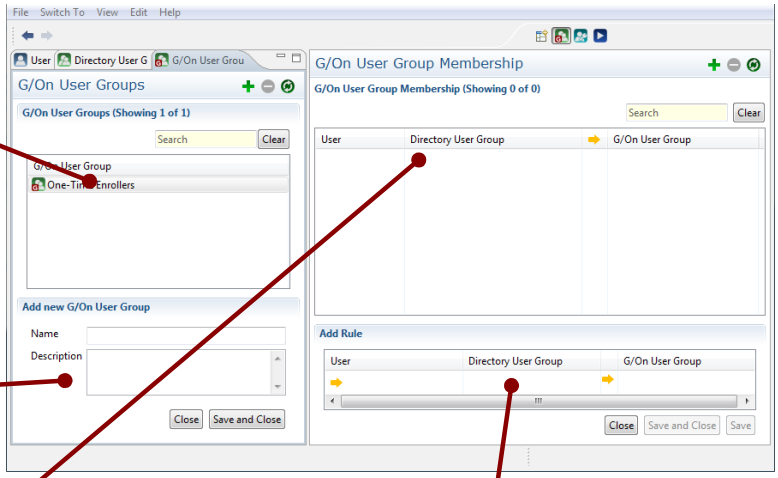


When a Perspective has been chosen in the menu, it will appear and stay in the Perspective bar. Buttons in the Perspective bar can be removed by right-clicking and choosing Close.

Perspective layout

Underneath the Perspective bar on the left hand side is number of Element tabs. The Element tabs holds a list of the existing Elements of specific types. A search field is used for locating Elements by name.

When choosing to add or edit an Element the Add/Edit Element area appears at the bottom.



At the right hand side is a listing of Rules related to the selected perspective.

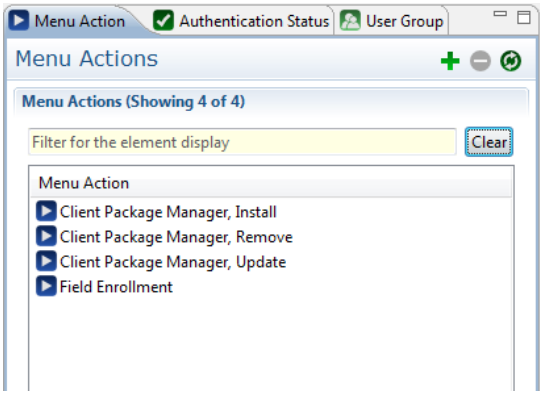
When adding or editing a Rule the Add/Edit Rule area will appear at the bottom.

Resetting the perspective

If the perspective seems to be out of order somehow, it is possible to reset the different parts of the perspective. In most cases this should not be needed. However to reset the perspective go to the menu and choose View > Reset Window.

Introduction to Element lists

The Elements in the Element lists are used for creating Rules that are located in the Rule lists. For each type of Element there is an Element list. The Element list available in any perspective reflects the Elements that can be used in Rule creation in that specific perspective.



Listing Elements

The list should show all the available Elements of the selected type. It is possible to refresh the Element list, thereby ensuring that all Elements are displayed:

- Click anywhere in the list area. Then choose View > Refresh in the menu.
- Click anywhere in the list area. Then press the keyboard short cut F5.

Editing Elements

Most Elements can be edited.

There are several ways of editing existing Elements:

- Double-click the Element to start the editor.
- Select the Element that should be edited and press Enter.
- Right-click on the Element the should be edited and choose Edit from the context menu.
- Select the Element that should be edited. Then press the keyboard short cut Ctrl-E.

Note that not all Element types can be edited (For instance groups retrieved from a User Directory).

The editing possibilities depend on the Element type. Please refer to the subsections on the individual Elements for details.

Creating new Elements

There are several ways of adding new Elements:

- Click the plus sign (+) in the upper right corner.
- Right-click anywhere in the list area. Then choose New from the the context menu. For some Element types, it is also possible to choose Create Copy.
- Click anywhere in the list area. Then choose Edit > New.
- Click anywhere in the list area. Then use the keyboard short cut Ctrl-N.

Click the 'Close' button to close the editor without saving. Click the 'Save and close' button to save the changes and close the editor.

Deleting Elements

There are several ways of deleting an Element:

- Right-click the Element you wish to delete. Then choose Delete in the context menu.
- Select the Element you wish to delete. Then choose Edit > Delete.
- Select the Element you wish to delete. Then press the keyboard short cut Ctrl-D.

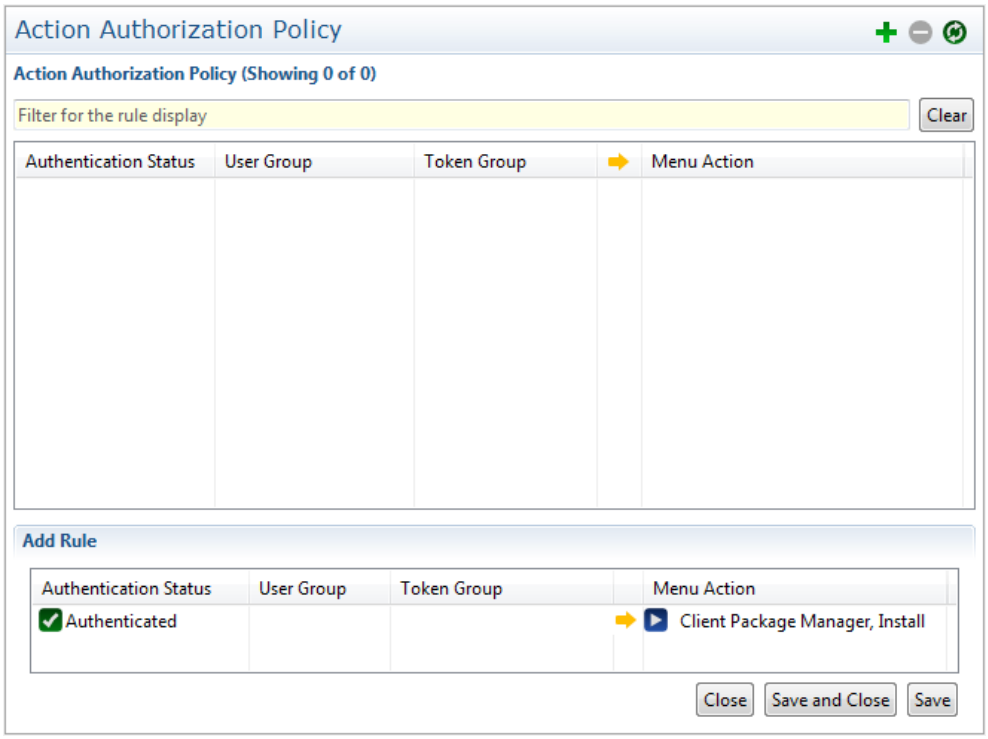
You will be asked to confirm the deletion of the selected Element.

Searching Elements

The Element search is a live search. This means that while typing in the search input area, the list adjusts to display the relevant Elements. Use the Clear button to clear the search and display all available Elements.

Introduction to Rule lists

The Rule list shows all the Rules that correspond to a specific perspective.



Listing Rules

The Rule list should show the Rules related to the selected perspective. It is possible to refresh the Rule list, thereby ensuring that all Rules are displayed:

- Click anywhere in the Rules list. Then press the keyboard short cut F5.
- Click anywhere in the Rules list. Then choose View > Refresh.

Creating new Rules

In any of the Rule based perspectives it is possible to create new Rules. There are several ways of starting to create new Rules.

- Click the green plus sign (+) at the top right.
- Click anywhere in the Rules list. Then press the keyboard short cut Ctrl-N.
- Click anywhere in the Rules list. Then choose Edit > New.
- Right-click anywhere in the Rules list. Then choose New in the context menu.

Any of these should result in the Add/Edit Rule area appearing at the bottom of the perspective.

Editing Rules

Rules can be edited. There are several ways of editing a Rule:

- Double-click the Rule.
- Select the Rule. Then press Enter.
- Right-click on the Rule. Then choose Edit in the context menu.
- Select the Rule. Then press the keyboard short cut Ctrl-E
- Select the Rule. Then choose Edit > Edit.

Single elements can be removed from an existing rule by right-clicking the element and selecting remove from the context menu.

Deleting Rules

Rules can be deleted. There are several ways of deleting a Rule:

- Select the Rule. Then press the keyboard short cut Ctrl-D
- Select the Rule. Then choose Edit > Delete.
- Right-click the Rule. The choose Delete from the context menu.

Searching Rules

The Rule search is a live search. This means that while typing in the search input area, the list adjusts to display the relevant Rules. Use the Clear button to clear the search and display all available Rules. The search considers all Elements in a Rule to see if something matches.

Adding Elements to a Rule

Elements are selected from the Element lists. There are several ways of adding an Element to a Rule:

- Select the Element and drag it onto the Add/Edit Rule area.
- Select the Element and drag it into the Rule list.
- Select the Element. Then press the keyboard short cut Ctrl-A
- Right-click the Element and choose Add to rule editor from the context menu.
- Select the Element. Then choose Edit > Add to rule editor.

Adding an Element to a Rule results in that Element appearing in the Add/Edit Rules area at the

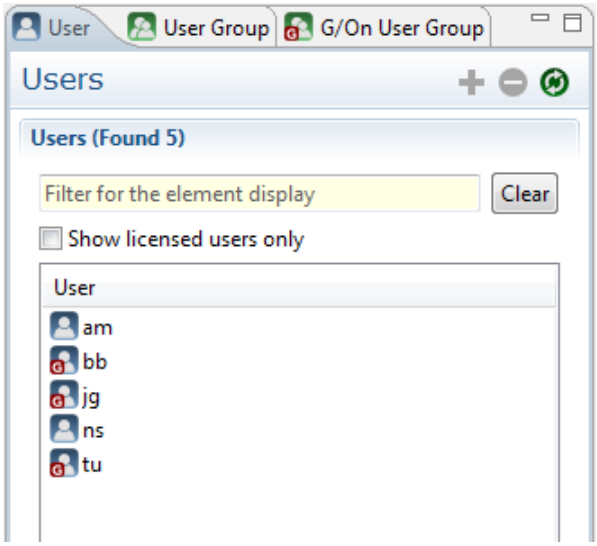
location provided for that specific Element type. Adding another Element of the same type removes the existing Element and adds the new one instead.

Export Rules

It is possible to export rules to a CSV file:

- Select the Rules to be exported.
- Right-click, and choose Export Selected Rule(s) to CSV
- Choose where to save the file, and click Save

Element: User



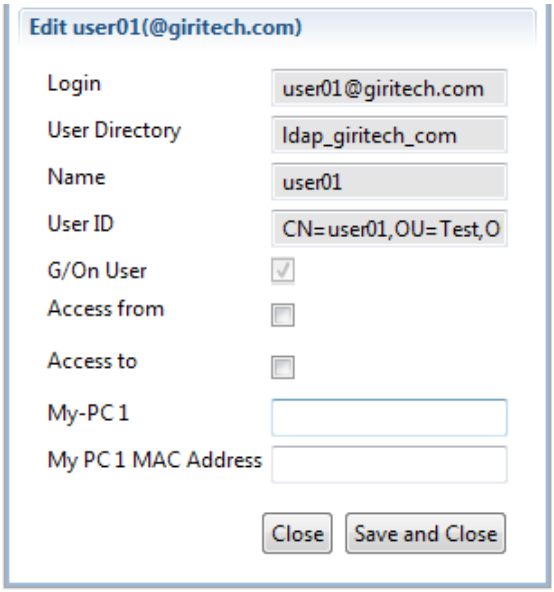
User Elements represents an actual User on the G/On server.

User Elements come from the User Directories, which the server is set up to connect to. This may also include local Users on the machine where the G/On Management and G/On Gateway Servers are running. User Elements may be used in the Personal Token Assignment perspective and the G/On User Group perspective. Note that there is a limit of 500 on the number of users shown in the list by default. This limit has been introduced in order for G/On Management to work with large user directories. The limit size can be changed in G/On Configuration.

New

It is not possible to add new Users directly through G/On. Users should be created in one of the external User Directories.

Edit



The User Elements can be edited. See page 120 for information on how to start editing. Note that once settings have been saved, the user will be registered as a G/On user and therefore count as a licensed user. If the number of licensed users have been reached already it will not be possible to save settings for an unregistered user.

The settings that can be changed for a User are the access period and the User's personal workstation settings.

An access period can be set for a user, so that the user will only be considered a valid user during that period. Checking the “Access from” will reveal a date and a time field in which the start date of an access period can be entered. Likewise an end date and time can be entered by checking the

“Access to” box. If only one of the fields is checked it means that the access period is open-ended at one end. Leaving both unchecked means that there is no time restriction on the User's access.

The personal workstation settings (the My PCx fields) allows you to set up actions that will allow a User direct and secure access to his/her personal workstation from anywhere. Note that any number of workstations can be set up. In order to set up more than one workstations, first enter the data for the first one, save it and then edit the User again. It is now possible to add information on workstation 2 and save it. After that you can add workstation 3 and so on.

It is possible to create a Menu Action that will wake up a User's personal workstation. For this Menu Action to work properly the workstations mac address needs to be set. The Menu Actions are created in the Menu Action list. See page 135 for more information on the Menu Action list.

It is not possible to change any User Directory related settings through the G/On management client.

Delete

It is not possible to delete Users directly through G/On. Users should be deleted in one of the external User Directories.

Add as G/On user

This is a shortcut for adding a user as a licensed G/On user. A user can also be added by editing user settings or by creation of a rule containing the user.

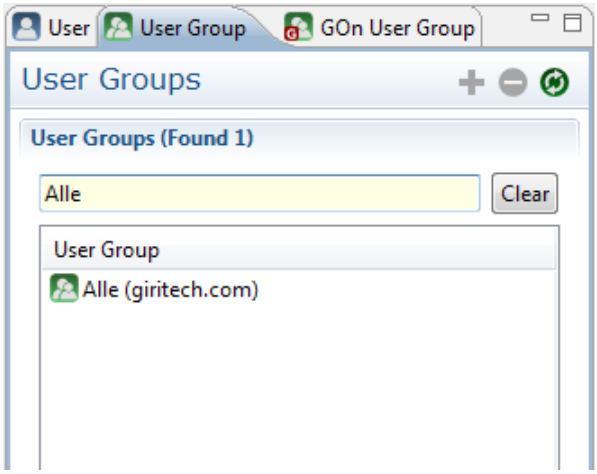
Remove from G/On users

This action will remove the user from the G/On database. If the user has been assigned a personal token, it is not possible to remove. The token assignment rules containing the user must be deleted first. Memberships of G/On user groups will, however, be deleted automatically when the user is been removed.

Show licensed users only

Checking this box will change the User view so that only users registered as licensed G/On users are shown in the list. Note that a user which has been registered as a G/On user, but has been deleted from the user directory, will only appear in this list. Also note that there is no limit on the number of users shown in this list.

Element: Directory User Group



Directory User Group Elements come from the User Directories, which the server is set up to connect to. This may also include local groups on the Machine where the G/On Management and G/On Gateway Servers are running. It is not possible to add or remove Users from Directory User Groups from within the G/On Management Client. Directory User Groups are used in the Authorization and the Authentication Policy Perspectives.

Note: There is a limit of 500 on the number of users shown in the list by default. This limit has been introduced in order for G/On Management to work with large user directories. The limit size can be changed in G/On Configuration.

Note: In the perspectives: User Authentication Policy and Action Authorization Policy, the User Group list is actually *a mix* of groups from the Directory User Groups and G/On User Groups (see below).

New

Creating a new User Group defaults to creating a new G/On User Group. If you create a new G/On User Group you should go to the G/On User Group Management Perspective to manage which Users should be part of your new group. See page 121 for information on how to create new Elements.

Edit

It is not possible to edit any settings for Directory User Groups directly through G/On. These groups should be edited in one of the external User Directories.

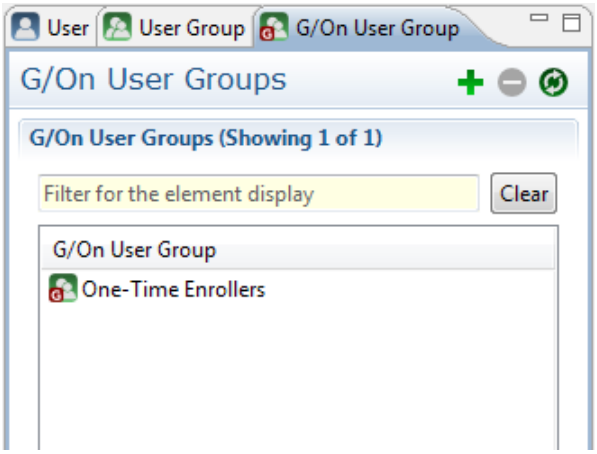
In the G/On User Group Management perspective it is possible to edit both the title of G/On User Groups and which Users are member of the group.

Delete

It is not possible to delete Directory User Groups directly through G/On. These User Groups should be deleted in one of the external User Directories.

It is possible to delete G/On User Groups in the G/On User Group Management perspective. But only if they are not used in any Rule.

Element: G/On User Group



G/On User Groups can be used as an extension of the User Directory Users and Groups. If you have a number of Directory User Groups and individual Users that you wish to combine into one Group, you can use G/On User Groups for that purpose. This can significantly simplify Authorization and/or Authentication Policies.

Note: There is a special, built-in G/On User Group: One-Time Enrollers. It is intended to be used in connection with field enrollment: Users in this Group can be authorized to enroll a Token, as a action in the end-user G/On client.

New

It is possible to create new G/On User Group Elements. See page 121 for information on how to create new Elements.

Edit

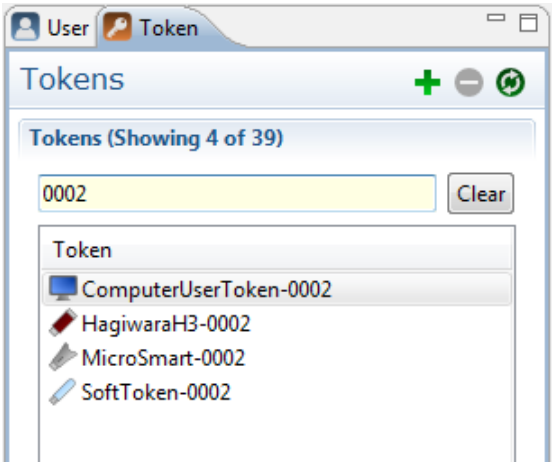
You can edit the name of any G/On User Group that is not built-in. See page 120 for information on how to start editing Elements.

Delete

It is possible to delete G/On User Groups that are not built-in or in use in any Rules. See page 121 for general information on how to delete Elements.

Element: Token

A Token is a hardware or software device that can serve as an authentication factor of the kind: "Something You Have". So a Token can be distributed to a User, and the User can then present the Token at a later time in order to confirm his or her identity. Some Tokens also have a capacity to hold client side software, such as the G/On Client, application clients, and even a whole client side operating system.

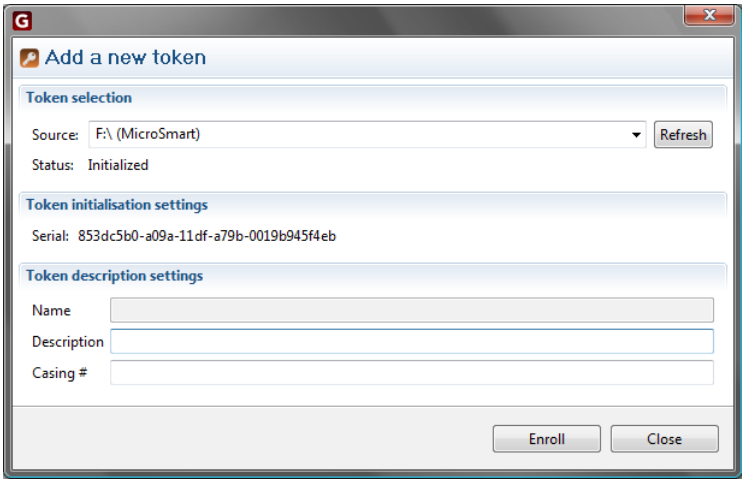


New

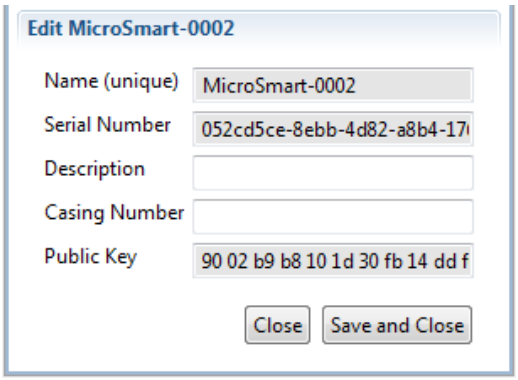
Token Elements are added to the system by enrolling them. See page 121 for information on how to start creating new Elements.

Creating a new Token Element results in opening the 'Add new token' dialogue.

The source drop-down shows any valid Token inserted into the local workstation.



Note: Some tokens must be initialized before they can be enrolled. See the G/On Setup and Configuration Reference, section: Advanced Setup Topics - Initialization of Tokens.



If no Token is shown in the Source drop-down try inserting another Token and click Refresh. For any Token you can add a Description and a Casing Number. Click Enroll to add the selected Token to the G/On Server. The new Tokens Serial number should now appear in the Token list. Click Close to close the editor without saving. Click Save and close to save the changes and close the editor.

Edit

The Tokens that are enrolled into the server can be edited. See page 120 for information on how to start editing Elements.

It is possible to edit the Description and the Casing Number fields. The Serial Number is used be the G/On system and needs to be unique. Therefore it can not be changed.

Click Close to close the editor without saving. Click Save and close to save the changes and close the editor.

Delete

It is possible to delete Tokens. But only if the Token is not used in any Rules. See page 121 for general information on how to delete Elements.

Types of Tokens

Soft Token is a top level folder on a removable device with G/On software and a private key file for authentication.

MicroSmart Token is a MicroSD card with a flash drive with G/On software and a built-in Smart Card with a private key for authentication.

MicroSmart USB Token is a USB adapter with a MicroSmart Token (see above). Software which accesses the Token cannot distinguish it from Tokens of the type: MicroSmart.

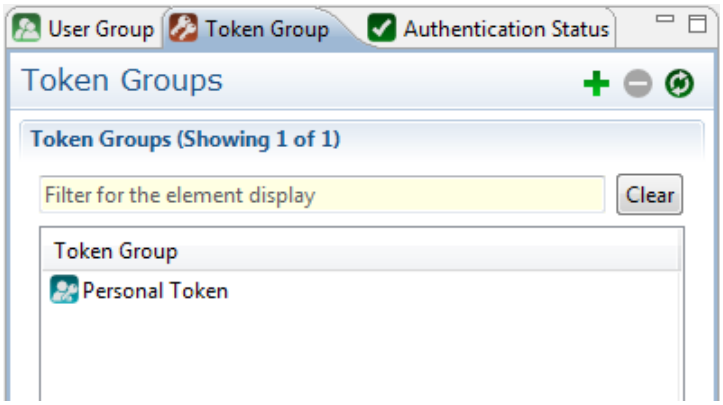
Hagiwara H2/H3 USB Token has both a CD and a flash drive with G/On software and a hidden Unique ID and private key file for authentication.

Computer User Token is a computer (PC or other computing device) where a (possibly non-admin) User has "installed" a private key and the G/On software in the Users home directory/registry on a computer. The key file is typically locked to a specific computer by a screening check of the MAC addresses.

Mobile Token is a mobile device, e.g. an iPhone, where a (possibly non-admin) User has "installed" a private key and the G/On software on the device. The key file is locked to a device by a screening check of the unique ID of the device.

Smart Card Token is a private key authentication factor which can be inserted in a Smart Card Reader in a PC running the G/On software.

Element: Token Group



Token Groups are collections of Tokens that can be used when formulating User Authentication Policies or action Authorization Policies. Which Tokens are members of which Token Groups is defined in the Token Group Management perspective.

Note: In addition, there is a built-in, "dynamic" Token group called Personal Token which is used for identifying the Personal Tokens of the User of the current session. This is defined in the Personal Token Assignment perspective.

New

It is possible to add new Token Group Elements. See page 121 for general information on how to create new Elements.

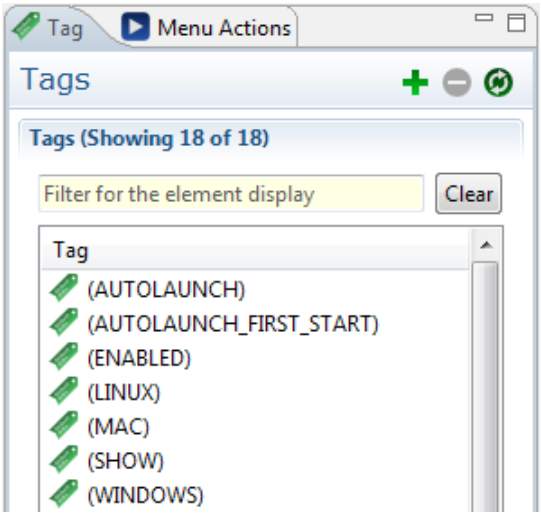
Edit

It is possible to change the name of Token Groups – but only in the Token Group Management perspective. See page 120 for more information on how to start editing Elements.

Delete

Any Token Groups that is not built-in can be deleted when they are not used in any Rules - but only in the Token Group Management perspective. See page 121 for general information on how to delete Elements.

Element: Tag



The Tag Elements can be used to categorize your Menu Actions. All Menu Actions are assigned a number of Tags. A Tag can be placed in a Menu structure and then any Menu Action that has that Tag will be added in the Tag's location (But only if the User has access to that specific Menu Action). Any personalization of menu actions placement in the user menu is done by manipulating tags. Because of the dynamic nature of when Menu Actions are available to Users, the location of Menu Actions can not be done more precisely.

Note that several predefined Tags exist. See page 143.

New

It is possible to add new Tags. See page 121 for general information on how to create new Elements. Note also that new Tags can be introduced by adding them to a Menu Action – see page 143.

Edit

It is possible to edit Tags. See page 120 for more information on how to start editing Elements.

Each Tag has several settings that can be changed.

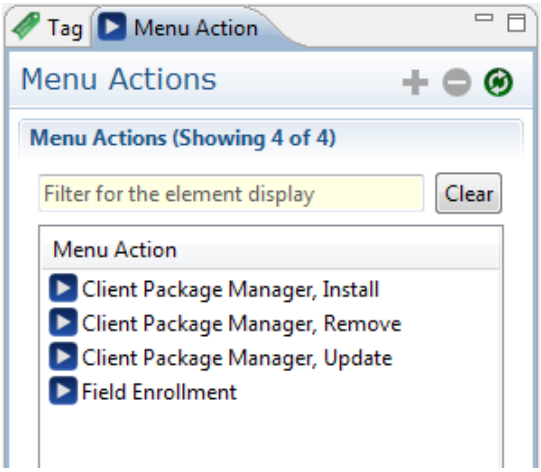
- **Name** is the Tag's name. This is the name which is used as referral in Menu Action specifications. The Tag name can only consist of alphanumerical characters and is always in upper case (lower case letters entered will be converted when the Tag is saved)
- **Caption** is used for naming a folder in the User menu that will contain Menu Actions with this Tag.
- **Show in menu** is used for deciding whether or not this Tag should be shown as a folder in the User menu. Some Tags should not. For instance tags can be used to decide whether some Menu Actions should be displayed at all.
- **Parent Tags** are listing the parents to this Tag in the menu structure. A Tag can have any number of parent Tags. Note that parent Tags can only be edited by dragging the Tag onto the menu tree.

- **Max items to show** are used for limiting the number of items displayed in the menu folder with Menu Actions with this Tag. This is useful for e.g. creating top 3 most used menu folders. If the value is 0, all items are displayed.
- **Sort option** is used alone or in combination with the “Max items to show” functionality to find the order of items shown. Possible values are most used, last used or plain alphabetically.
- **Override item show** can be set in order to always see all Menu Actions in the menu even though other factors (e.g. client platform) prevents it from being shown. Useful for checking that a Menu Action has been authorized for a User.
- **Automatically add to all items** is used for adding this tag dynamically to all Menu Actions. This is used for creating an “All Programs” menu or a “Top <X> Most Used” folder for the Users.

Delete

It is possible to delete Tags. See page 121 for general information on how to delete Elements.

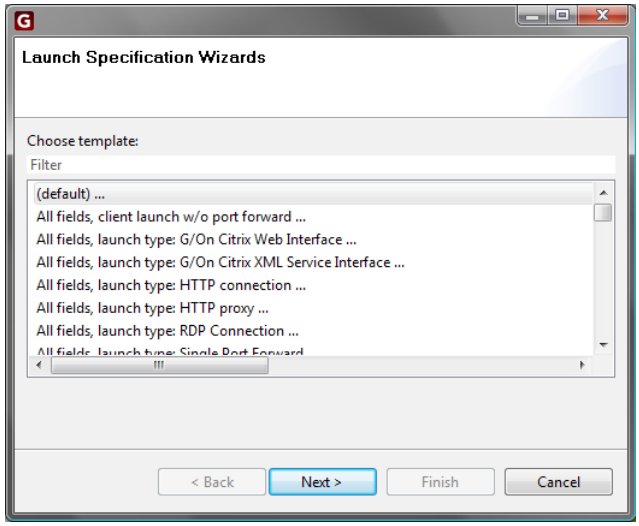
Element: Menu Action



The Menu Action Elements are the Elements that correspond to the Menu items that may end up in the end-users menu if they are authorized to use it. See the introduction in the section: “Menu Actions“, on page 102.

New

It is possible to add new Menu Actions. See page 121 for general information on how to create new Elements. Note that it is possible to start creating a new Menu Action, based on a copy of an existing Menu Action: right-click and choose Create New.



This will start a Menu Action creation wizard that will guide you through the set up of the most commonly used Menu Actions. If you need to create a Menu Action that is more specialized you can use the default wizard.

Click next and fill in the required information until you can click Finish and the Menu Action appears in the Menu Action list.

Some Menu Actions require information about specific User's workstation settings. These settings are added in the User's information editor. See page 125 for more information on editing User settings.

Some Menu Actions require a special setup of an application server. These are documented on page 188.

Properties

It is possible to edit a Menu Actions. The editor resembles the wizard pages for the specific Menu Actions. See page 120 for information on how to start editing Elements.

Delete

It is possible to delete Menu Actions that are not used in any Rules. See page 121 for general information on how to delete Elements.

Create Copy

Create a new Menu Action based on an existing one.

View

View the Menu Action using the default wizard in read-only mode.

Add/Remove zone restrictions

Add or remove Zone restrictions to/from a Menu Action. See page 150 for further description of the Zone concept.

General Features of Menu Actions

The different Menu Action templates have different fields, but there is a basic set of fields, which are used in many templates. These fields and their meanings are described in the following.

Note: For an introduction to the notion and types of Menu Actions, see the section: “Menu Actions“, on page 102.

Fields for identifying the Menu Action

Each Menu Action has a unique name and a title and optionally an image (icon). The unique name is internal, only visible to the administrators. The title is used in the menu presented to end-users; it need not be unique across all menu actions. For instance, the title: “Remote desktop” can be used both for a menu action for Windows client, and a menu action for the Mac client, which provide access to the same terminal server:

- Name
- Menu Title
- Menu Image ID (currently only used on iPad/iPhone)

Fields for defining port forward(s)

Menu Actions of the type: “Port Forward” can define 1 or more port forwards. Each port forward is defined by the listening address and port on the client side and the address and port on the server side:

- Client Host (default value: 127.0.0.1)
- Client Port (default value: 0, which means: pick any unused port)
- Server Host
- Server Port

This works much like a port/address translation in a router:

1. An application on the client PC can connect to the *client port* at the *client host* address
2. This connection will be translated (forwarded) to the *server port* at the *server host* address, on the network where the G/On server is located.

Fields for defining client-server connectivity with RDP and Citrix connections

When a menu action of type: “RDP Connection”, “Citrix XML Interface” or “Citrix Web Interface” has been started, the G/On client will listen for connections from the application client (RDP client, ICA client or browser) on a specified address and port:

- Client Host
- Client Port

On the server side, connection to an RDP or Citrix server is specified by server address and port:

- Server Host
- Server Port

In order to support fail-over and load sharing functionality, a list of hosts can be specified in the Server Host field. If the initial connection to one host fails, there will be fail-over to another one in the list. Note, however: there is no "hot" fail-over - connected users must reconnect after failure.

Entries in the list of hosts must be separated by space. Optionally, a port can be added to each entry, separated by a colon. If there is no port added to an entry, the port specified in the separate field: Server Port is used.

Special directives can be specified at the start of the list of hosts, separated by space. The directives controls which server will be tried first, and what will happen if that fails:

DIRECTIVE	ACTION
#random	Chooses a random starting point in the list of hosts. If connection to this host fails, the next hosts will be tried, one after the other, with the first following the last. This will share the load among the hosts. #random is default.
#failover	Chooses the first host in the list of hosts. If connection to this host fails, the next host will be tried. This will never use the remaining hosts in the list, unless all previous hosts have failed. If both #failover and #random is specified, that last one takes precedence.
#timeout=n	Specifies the number of seconds before a host should time out and considered a failure. The default is 3.

Examples:

- `#failover #timeout=2 primary.server.com secondary.server.com`
- `#random fw.com:112 fw.com:911`

In addition to the explicitly defined addresses and ports, menu actions of types: “RDP Connection”, “Citrix XML Interface” and “Citrix Web Interface” may automatically create other connections as described above in the section: “Menu Actions“, on page 102.

Fields for defining client-server connectivity with HTTP and SOCKS proxy connections

When a menu action of type: “HTTP and SOCKS Proxy” has been started, the G/On client will listen for connections from the application client (e.g. a browser) on a specified address and port:

- Client Host
- Client Port

The menu action can be set up to allow either plain HTTP or proxy communication. In the case of plain HTTP, the client's communication will be routed through the transparent HTTP proxy in the G/On Gateway Server, and forwarded to the specified application server address and port (multiple hosts and control tags for fail-over and load sharing can be specified as described above, on page 138). Note that the selected host and port will be the target of all communication no matter which HTTP commands are issued:

- Server Host
- Server Port

If the menu action has been set up to allow communication using the HTTP proxy protocol or the SOCKS proxy protocol, the communication will be routed to the built-in HTTP or SOCKS proxy. The proxy will receive the commands in the proxy protocol for establishing HTTP or TCP connections to the requested addresses and ports, and will react to these commands depending on a specified white list:

- 1. Permitted Server Address
- 1. Permitted Server Port
- 2. Permitted Server Address
- 2. Permitted Server Port
- etc.

A permitted server address field must start with a server address specification that has one of these forms:

- A DNS name
- A specific IP number (example: 192.168.1.10)
- An IP range specified by IP/bitcount (example: 10.7.0.0/16)
- An IP range specified by IP/ bitmask (example: 10.7.1.0/255.255.255.0)
- An IP range specified by to and from (example: 192.168.1.10-192.168.1.19)
- 0/0, which matches all IP numbers

A permitted server port field must contain a port number to be matched. The port number 0 is treated as a wild card that matches all ports.

When a client tries to communicate to a given address, the white list is searched in two rounds: first searching for a matching DNS name, and then searching for a matching IP or IP range. In each round the white list is searched in the order specified and the first match is used. If no match is found, the communication attempt is blocked.

Four different matching rules will be used, depending on whether the client uses SOCKS or HTTP proxy communication (usually governed by the command field of the menu action), and whether the client tries to communicate to an IP number or a DNS name:

PROTOCOL	IP / DNS	MATCHING RULE
HTTP	IP	Matches whitelist entries that either mention this given IP number, or mention an IP range which includes it
HTTP	DNS	Matches whitelist entries that mention this given DNS name
SOCKS	IP	Matches whitelist entries that either mention this given IP number, or mention an IP range which includes it, or a DNS name that resolves to this IP number, on the server side
SOCKS	DNS	Matches whitelist entries that are matched by the IP number which this given DNS name resolves to, on the server side

When the first matching white list entry has been found, an action it taken without considering the following entries. The action will depend on directives that may be specified in the permitted server address field after the IP/range/DNS:

DIRECTIVE	ACTION
#deny	If the #deny directive has been specified in the matching white list entry, the communication attempt is blocked. Otherwise, the communication attempt is allowed (however subject to other directives that may be specified in the given white list entry). Applicable to both SOCKS and HTTP proxy protocol.
#httpproxy=proxyserver:port	Allowed communication is forwarded to another HTTP proxy server at the given address and port. Applicable only to the HTTP proxy protocol.
#auth #auth=basic #auth=ntlm	User credentials will be injected into allowed communication, as described below under the heading “Fields for defining server side single sign-on credentials”. Applicable only to the HTTP proxy protocol.

Example This example allows connection to my.site.local with single sign-on using basic authentication, and denies access to all other internal addresses and forwards all remaining requests to an HTTP proxy that runs on the same machine as the G/On Gateway Server:

```
1. Permitted Server Address: my.site.local #auth=basic
   ------------------- Port: 80
2. Permitted Server Address: 192.168.0.0/16 #deny
   ------------------- Port: 80
3. Permitted Server Address: 10.0.0.0/24 #deny
   ------------------- Port: 80
4. Permitted Server Address: 172.16.0.0/16 #deny
   ------------------- Port: 80
5. Permitted Server Address: 0/0 #httpproxy=127.0.0.1:8080
   ------------------- Port: 80
```

Caution:

- Giving access to a web server's IP number/port will give access to all name based virtual hosts on that IP number/port.
- Care must be taken to ensure that credentials don't get forwarded to untrusted servers.

Note:

- In the whitelist, it is recommended to place entries with DNS names before entries with IP numbers and ranges.
- The syntax for directives is similar to the one used for load sharing and fail over, but it is not well-defined how the two classes of directives should interoperate and mixing them is therefore not supported.

Fields for defining server side single sign-on credentials

- SSO login
- SSO password
- SSO domain

The credentials specified here may be used for single sign-on by injecting the credentials on the server side, in the application protocol. This is only supported for Citrix and RDP proxy and HTTP and HTTP proxy connections.

For Citrix and RDP proxy connections, SSO is automatically attempted unless the fields are empty.

Leaving the fields empty for a Citrix connection may render the menu action useless, as it may not be possible for Citrix to present a login dialog for manual login.

For HTTP and HTTP proxy connections, SSO is only attempted towards a server if one of the following authentication directives has been specified in the corresponding Server Address field:

- #auth=basic
- #auth=ntlm
- #auth

The directive must appear after the server address specification, separated by a blank space. If #auth is specified in the server host field then the proxy will silently try to respond to server requests for authentication using NTLM or Basic authentication. #auth=basic or #auth=ntlm can be used to initiate a specific kind of authentication before it has been requested by the server. Note that limitations in the HTTP standards do that it cannot be guaranteed that this will work in all cases.

Caution: Care must be taken to ensure that credentials don't get forwarded to untrusted servers.

Fields for defining the client side program to start

A Menu Action can start a program on the client PC, and can also generate a parameter file with data for the program. Both are optional. The parameter file is automatically deleted after the specified parameter file life time has expired, or when the port forward closes, whichever comes first. A parameter file life time 0 means: Remove the file when the port forward closes, or if there is no port forward, remove the file when the G/On client exits. A parameter file life time -1 means: Do not remove the parameter file.

- Command
- Working directory
- Parameter file name
- Parameter file lifetime
- Parameter file template

Field for defining what to do when ending a menu action

When a menu action ends, either because of “Close with process” (see below) or because the G/On Client exits, a command may be executed:

- Close Command

This can in some cases be used to perform some kind of clean-up.

Fields for defining ties between port forwards and client side programs

The following fields are used for specifying which programs can use a port forward, and what to do if the port forward closes, or the program exits:

- Close with process (closes the port forwards, if the program exits)
- Kill process on close (kills the program, if one of the port forwards closes; currently this can only happen if the G/On client closes)
- Lock to process PID (Only the launched command may use the port forward)
- - or its sub processes (Also allow subprocesses of the launched command to use the port forward - requires lock_to_process_pid)
- Lock to process name (Only processes with this name are allowed to use the port forward - conflicts with lock_to_process_pid)

Note: Some programs behave in a way which makes it impossible to use the above fields. For instance, some applications hand over control to another process immediately after they have been started, and then exits. This is the case for commonly used browsers. It is also the case for the Microsoft Terminal Services Client (mstsc), when used on 64 bit versions of Windows Vista and Windows 7.

Fields for specifying User convenience properties by means of Tags

The following fields are used for controlling the appearance of the Menu Action in the menu, and whether the Menu Action should be automatically started, when first appearing in the menu:

- Dialog Tags
- Dialog Tag generators

Any Tag can be put on a Menu Action by simply adding the Tag to the field: Dialog Tags. Thereafter, it will be available a basis for defining menus and sub-menus. See page 133 and 172.

For instance, you can put a tag “ERP” on some menu actions – this will enable that the menu actions can be presented in a separate sub-menu.

In addition, the following Tags have special meaning:

_MENU_ROOT

Enables that a menu action can be shown in the root of the user's menu.

SHOW

Must be present for the Menu Action to be shown. Is automatically added to all Menu Actions that have (or get) the Tag ENABLED. Can also be added manually, in order to show Menu Actions that are not enabled.

ENABLED

Is automatically added to all Menu Actions that have (or get) both the Tags: CLIENTOK and SERVEROK.

CLIENTOK

Must be present for the Menu Action to get the Tag ENABLED. Can be generated dynamically by a Tag generator of the form:

```
client_ok::IfPlatformIs("...")
```

SERVEROK

Must be present for the Menu Action to get the Tag ENABLED. In future versions, there may be Tag generators for automatically generating this, e.g. based on the availability of a server etc.

AUTOLAUNCH

If specified, the Menu Action will be automatically started, when it becomes available in a User's menu – provided that it also has the Tag ENABLED.

AUTOLAUNCH_FIRST_START

If specified, the Menu Action will be automatically started, when it becomes available in a User's menu – provided that it also has the Tag ENABLED, and provided that this is this first time the client is running after it was installed.

Tag generators

There are currently two types of Tag generators:

```
client_ok::IfPlatformIs("os")
```

where os is either win, mac, linux, iOS, iOS-iPhone, iOS-iPad, or iOS-iPod.

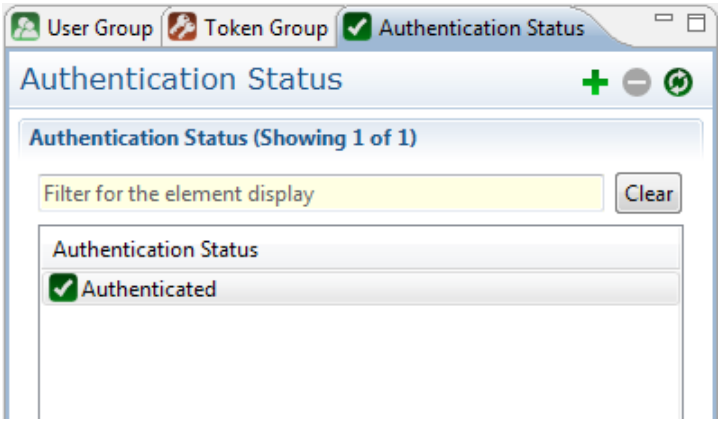
If this is specified ,and the G/On Client is running on a computer with the given OS, the Tag CLIENTOK is automatically generated.

```
package::CheckPackage("name", "os")
```

If this is specified, the Tag: PACKAGE_CHECK is automatically generated. Moreover, if the G/On Client is running in an environment where the given package is installed, in the highest version available from the server, the Tag: PACKAGE_INSTALLED is also automatically generated. If the package is not installed in the highest version available, the Tag: Package("name", "os") is generated. Currently, this is used for providing feedback, when a User chooses a Menu Action, where the necessary package has not been installed.

Element: Authentication Status

The Authentication Status list has a built-in Element called Authenticated. This can be used to indicate when proper authentication has been achieved. For a simple set-up use this as the only indicator for proper authentication.



New

It is possible to add new Authentication Status Elements. See page 121 for general information on how to create new Elements.

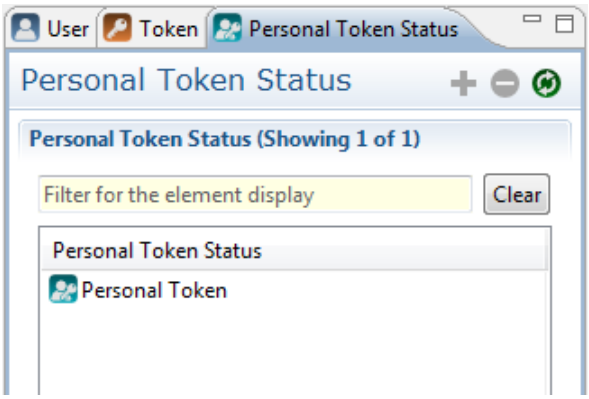
Edit

It is possible to edit the name of the Authentication Status Elements. See page 120 for information on how to start editing Elements.

Delete

It is possible to delete Authentication Status Elements that are not built-in or in use in any Rules. See page 121 for general information on how to delete Elements.

Element: Personal Token Status



The Personal Token Status Element is used as an indicator of when a User can be said to be using a Personal Token. In the Personal Token Status list is a built-in Element named Personal Token. This Element is used as a result Element in the Personal Token Assignment Rules. The Token Assignment Rules register individual Tokens to be authentication factors for individual Users. So if it follows from evaluation of the Rules, that Personal Token Status is Personal Token, we know that a known User with a Personal Token is using the system..

The Personal Token Status Element can also be viewed as a dynamic Token Group, which depends on the current User: for a given User, the Token Group, Personal Token, contains the Personal Token(s) of that User. Therefore, in the Authentication Policy perspective, the Token Group list also contains “Personal Token” as a special Token Group.

New

It is *not* possible to add new Personal Token Status Elements.

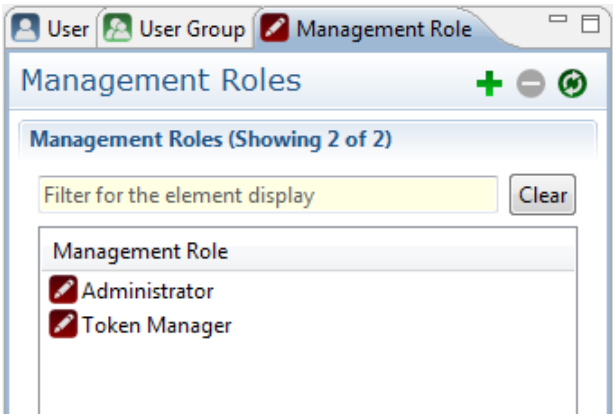
Edit

The built-in Personal Token Status Element called Personal Token can not be edited.

Delete

The built-in Personal Token Status Element can not be deleted.

Element: Management Role



A Management Role Element represents a role that a G/On Manager may have, and the (limited) set of privileges that are needed for carrying out the management tasks of that role. By using Management Roles it is possible to define exactly which access a user or user group should have to information and functions in the Management Server. The assignment of roles to users and groups is made in the perspective: Management Role Assignment.

There are two built-in Management Roles: Administrator and Token Manager. The Administrator role has access to all functionality, whereas the Token Manager role is a sample role dedicated to the management of tokens and their assignment to users.

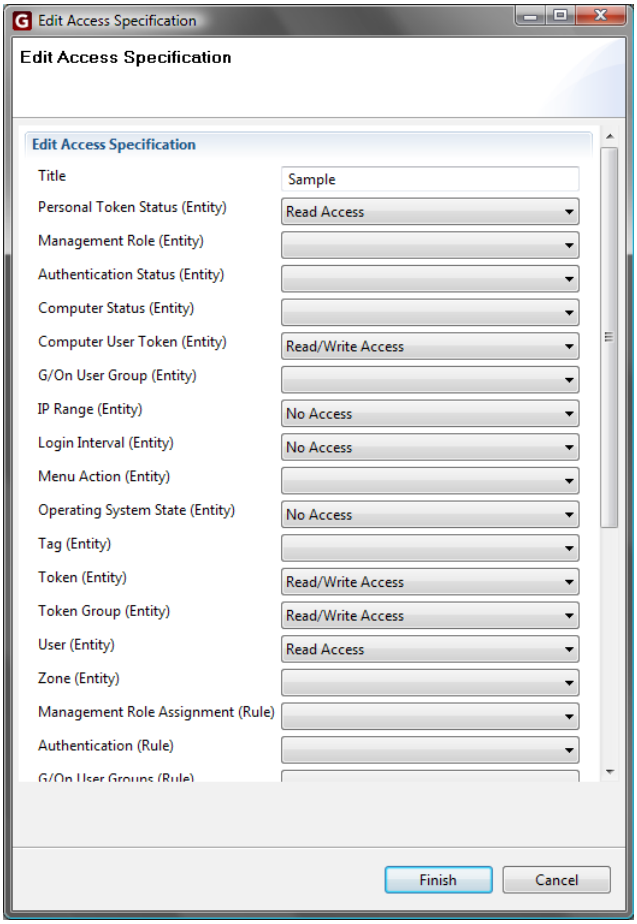
Each role specifies access rights to elements, rules, reports and special functionality.

For each Entity type (Users, Tokens, Menu Actions, etc). it is possible to give Read or Read/Write access.

For each Rule type (Personal Token Assignment, Action Authorization Policy, etc.) it is possible to give Read or Read/Write access.

For each Report it is possible to give Read access.

A special Gateway server configuration access right can be given in order to enable the Gateway Servers perspective. This also controls whether a user can install a new license.



New

Create a new Management Role

Properties

View/Edit a Management Role. Note that the built-in roles cannot be edited.

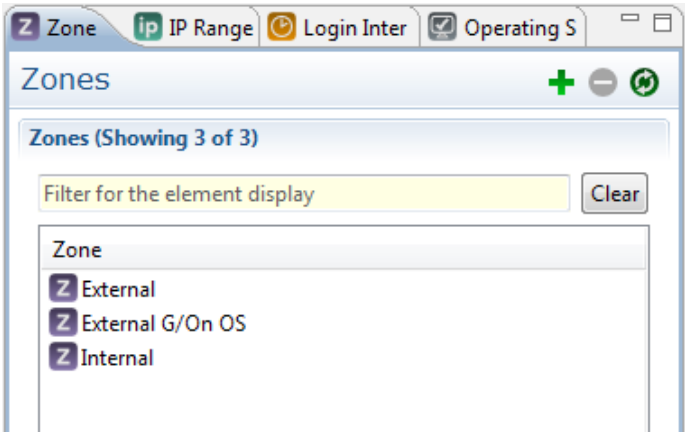
Delete

Delete a Management Role. Note that the built-in roles cannot be deleted.

Create Copy

Create a new Management Role based on an existing role. This means that the new role will be pre-filled with the access rights from the existing role. Note that for built-in roles, the copy will have "Built-in" set when the window is opened. That will change once the rule has been saved.

Element: Zone



The Zone elements each represent a set of circumstances which may be detected at the start of a user session. Zone detection rules are defined in the Zone Management perspective. A set of allowed Zones can be specified for a Menu Action, and this has the consequence that the Menu Action can only be launched if at least one of the allowed Zones has been detected for the current user session. If none of the allowed zones have been detected, the Menu Action will still appear in the client menu, but it will be marked with a special “disabled” icon and launching it will result in an error message specifying the reason why it cannot be launched.

New

It is possible to add new Zone Elements. See page 121 for general information on how to create new Elements. To create a Zone a unique name must be specified. It is also possible to specify a User Message. If the User Message is filled in, then this message will be shown to users trying to launch a Menu Action, which has been disabled by this zone. If the User Message is left blank the user message will be “Zone restriction <zone name> not met”

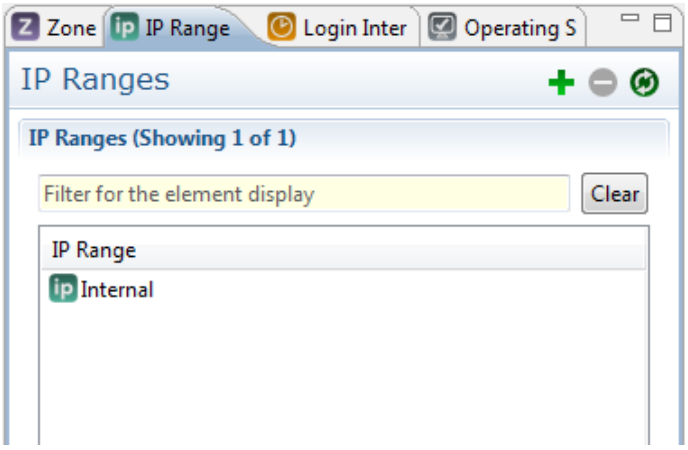
Properties

It is possible to edit Zone Elements. See page 120 for information on how to start editing Elements.

Delete

It is possible to delete Zone Elements that are not used in any Rules. See page 121 for general information on how to delete Elements.

Element: IP Range



An IP Range Element represents a range of IP addresses, as they may be observed by the G/On Gateway Server when a G/On Client connects. The range may concern either the client side or the server side address.

Both IPv4 and IPv6 addresses can be specified.

An IP address range may consist of just a single IP address.

New

It is possible to add new IP Ranges. See page 121 for general information on how to create new Elements.

Each IP Range has the following settings.

- **Name** is the IP Range name. This is the name which is used as referral in Zone Detection Rule specifications.
- **Description** is a description which can be used for reference.
- **Client IP Ranges** is a list of IP ranges for the client side of the connection between the G/On Client and G/On Gateway Server. IP Ranges should be separated by a comma, e.g. “127.0.0.1 – 127.0.0.2, 192.168.0.0/24.
  Note: If the client is behind a NAT router, these ranges relate to the externally observable address of the router.
  Exception: In G/On 5.5.0, the actual client side IP is always reported as: 127.0.0.1, if the client has connected by use of HTTP encapsulation.
- **Server IP Ranges** is a list of IP ranges for the network interfaces on the server, specified in the same way as Client addresses. This is only relevant when the G/On Gateway Server(s) have more than one network interface.

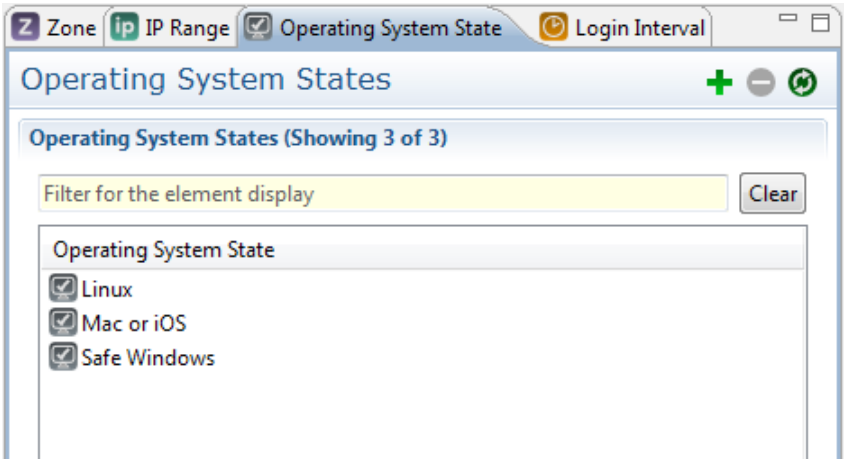
Properties

It is possible to edit IP Ranges. See page 120 for more information on how to start editing Elements.

Delete

It is possible to delete IP Ranges, which are not used in any rules. See page 121 for general information on how to delete Elements.

Element: Operating System State



An Operating System State element represents observed properties of the state of the operating system, where the G/On Client is running.

The properties may concern the client Operating System type, version and security status. Currently version and security settings check is available for Windows only.

New

It is possible to add new Operating System States. See page 121 for general information on how to create new Elements.

Each Operating System State has the following settings.

- **Name:** This is the name which is used as referral in Zone Detection Rule specifications.
- **Description:** This is a description which can be used for reference.
- **Linux Allowed:** Connection from linux is allowed
- **Windows Allowed:** Connection from Windows is allowed
- **G/On OS Allowed:** Connection from G/On OS is allowed
- **Mac Allowed:** Connection from a Mac is allowed
- **iOS Allowed:** Connection from iPhone or iPad is allowed

If Windows OS is allowed it is also possible to specify version and security checks on a separate page. Click Next in the wizard to reveal this page. The following options are available:

Click Check Version if version check should be in effect. The following versions are available:

- **Windows XP**
- **Windows Vista**
- **Windows 7**

and for each version a service pack requirement can be added. In the drop-down list it is possible

to choose the service packs available at the release of the current G/On version. In order to specify a service pack which has been released since the G/On release, the version number can be entered in the field manually. Check Giritech support for further details on what should be entered for a given service pack.

Click Check Security if security should be checked. The following checks are available:

- **Firewall**
- **Anti virus**
- **Windows Auto Update**
- **All important updates installed**

All except the last check are gathered from Windows Security Center. The possible values are “Poor” and “Good” relating to the values returned by Windows Security Center. Please check “http://msdn.microsoft.com/en-us/library/bb432506%28v=vs.85%29.aspx” for further details on the precise definition of the used API. Testing shows that the “Poor” value is always met, so setting the check for an option to be “Poor” is in practice the same as checking whether Windows Security Center returns a valid value. The value “Good” means the following:

- **Firewall:** a firewall is installed and enabled
- **Anti virus:** Anti virus is installed and up-to-date.
- **Windows Auto Update:** Windows is set to download and install updates automatically. Any other setting will result in the value “Poor”

Note that on Windows XP there is no Windows Security Center API. Therefore G/On uses WMI and registry lookup on Windows XP, in order to get the information necessary to behave in a similar manner as on the newer operating systems.

The “All important updates installed” check performs a call to Windows Update to check if there are any updates available, which are marked as required. Put in another way it checks for updates which will be automatically installed if Windows Auto Update is set to install automatically. Note that this check can be time consuming (normally in the range of 5-15 seconds, but it can be more), which means that Menu Actions depending on this check may be disabled at start-up and then enabled later when the check has passed.

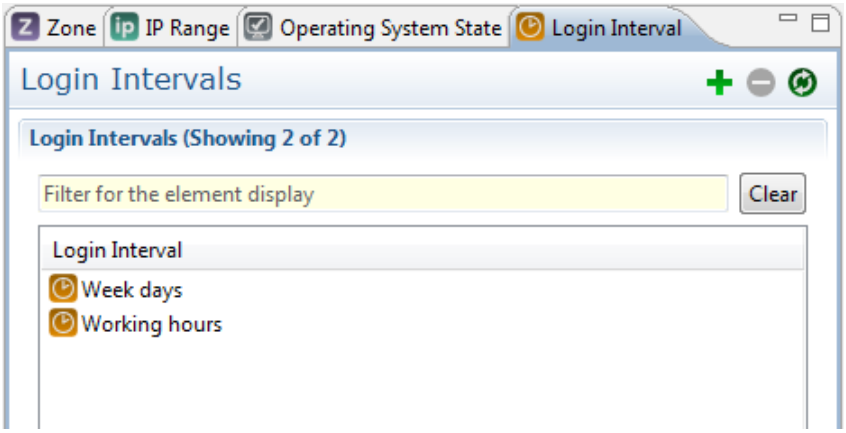
Properties

It is possible to edit Operating System States. See page 120 for more information on how to start editing Elements.

Delete

It is possible to delete Operating System States, which are not used in any rules. See page 121 for general information on how to delete Elements.

Element: Login Interval



Login Interval Elements are definitions of allowed login intervals on a weekly basis. It is possible to specify login hours on each day of the week. The following limitations apply:

- The login time is the server time. Time zone is the server time zone.
- The check is only made at login time. So if a user session expands outside any given login interval it has no consequences.

Note: For individual users, it is also possible to specify a (possibly open-ended) date/time interval where the user is considered valid. See page 125 for further details.

New

It is possible to add new Login Intervals. See page 121 for general information on how to create new Elements.

The following general settings are available.

- **Name:** This is the name which is used as referral in Zone Detection Rule specifications.
- **Description:** is a description which can be used for reference.

Furthermore there are settings for each week day:

- **All:** Access all day, i.e. from 0:00 to 24:00
- **None:** No access that day
- **Time:** A time interval in which access is allowed.

Note that it is allowed for the time interval to pass midnight, e.g. go from 9 PM to 9 AM. Also note that this means that it is possible to enter contradictory information. If, for example, access on Monday is set to be from 9 PM to 9 AM and access on Tuesday is disabled, then access on Tuesday morning is both allowed and forbidden. In such a case the interpretation will be that the positive access definition wins, i.e. logging in Tuesday morning before 9 AM is allowed.

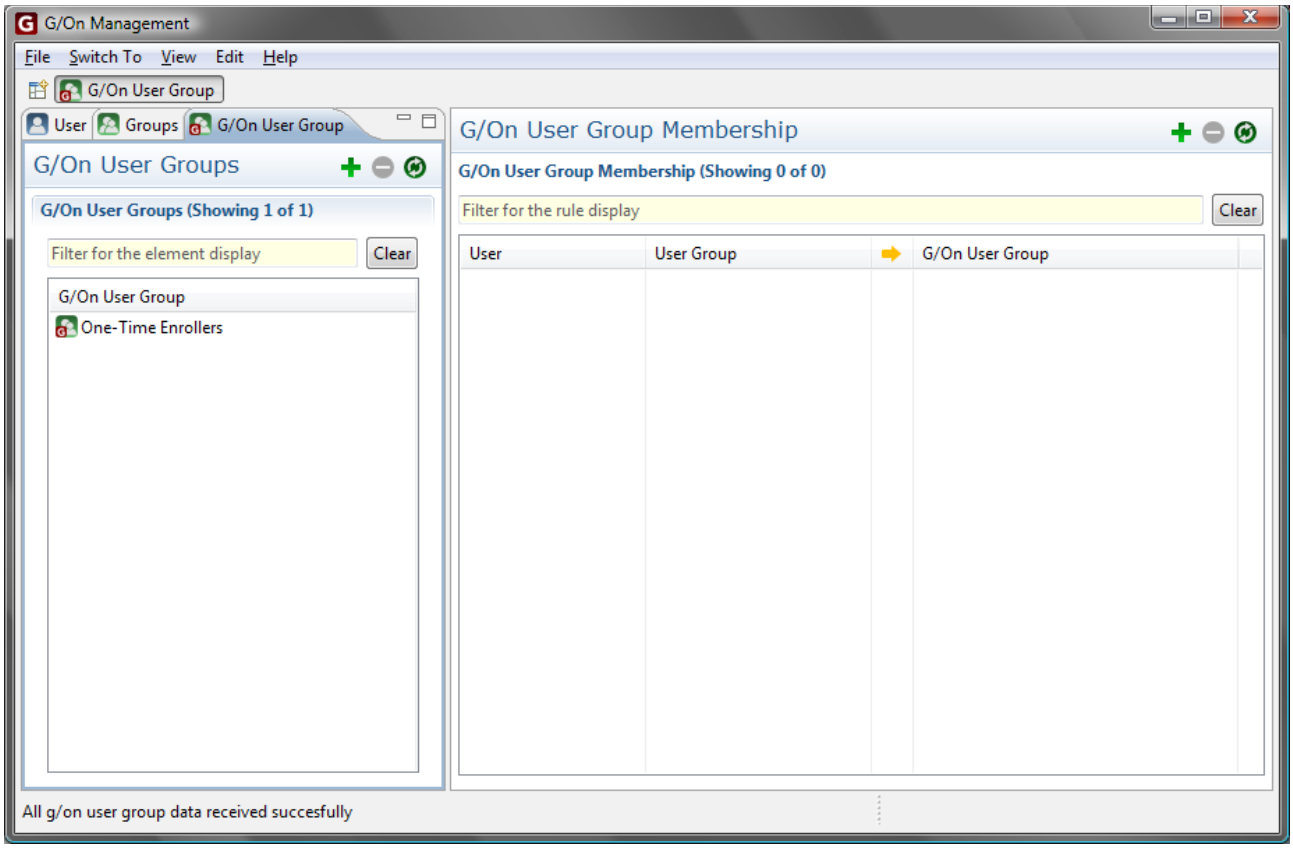
Properties

It is possible to edit Login Intervals. See page 120 for more information on how to start editing Elements.

Delete

It is possible to delete Login Intervals, which are not used in any rules. See page 121 for general information on how to delete Elements.

Perspective: G/On User Group



The G/On User Group perspective is used for adding Users that are retrieved from a central User Directory to a G/On User Group that can be created for that purpose. It is also possible to add entire groups from a User Directory to G/On User Groups. G/On User Groups adds a convenient way off creating local groups for use with the G/On services.

Note: There is a built-in G/On User Group: One-Time Enrollers, which has special properties:

1. By default, there is an Action Authorization Rule which authorizes members of One-Time Enrollers to do Field Enrollment
2. When a User has succeeded in doing a field enrollment, this User is automatically removed from the group One-Time Enrollers. However, for this to work, Users have to be added *individually* to One-Time Enrollers, i.e., there must be one Rule in the G/On user group perspective for each User. The automatic removal will not work, if the User is *indirectly* a member of One-Time Enrollers, i.e. through a Rule that adds an entire group from a User Directory to One-Time Enrollers.
3. To make this easier, right-click the User Group and choose Add members to One-Time-

Enrollers. This will add all the Users in the Directory User Group to the One-Time Enrollers group, *as individual members.*

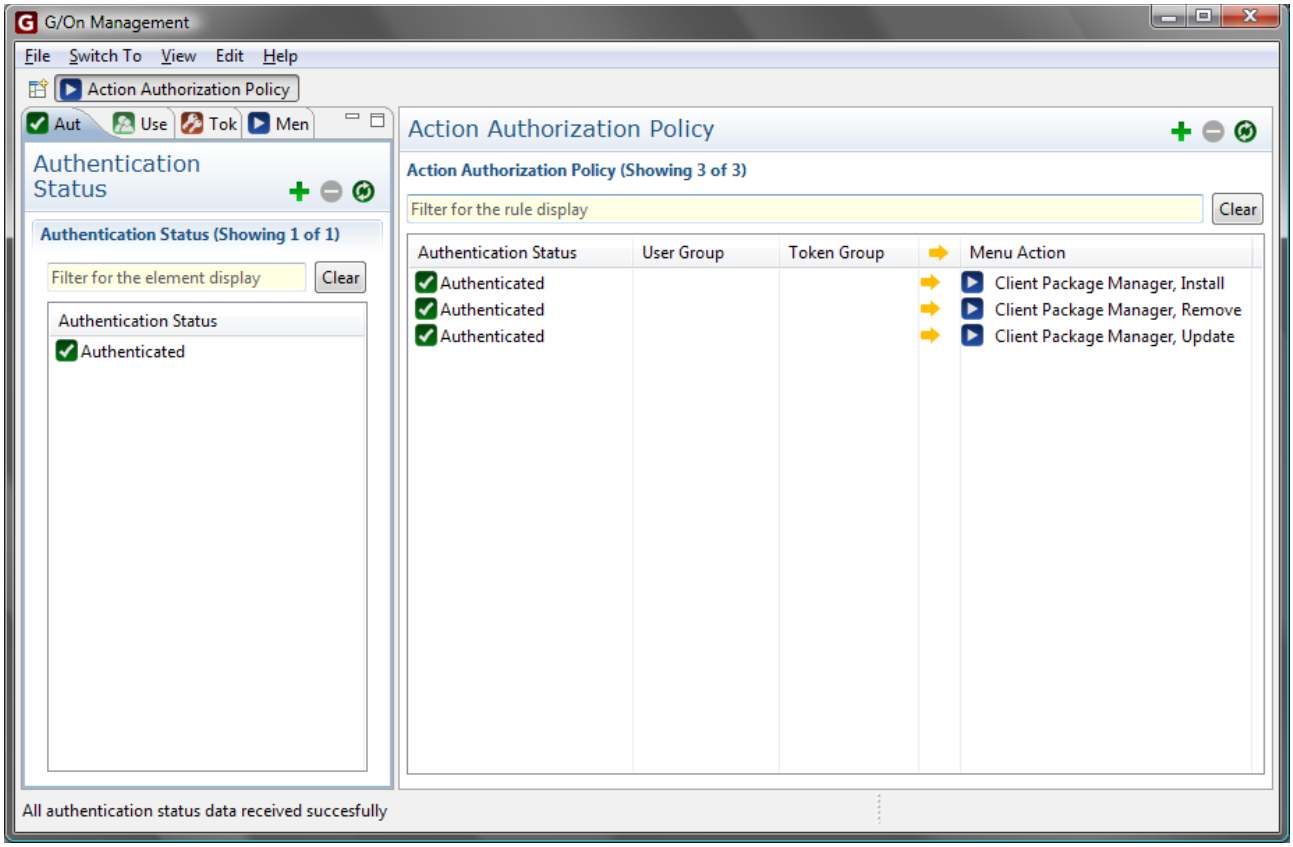
Rule Elements

The User and Directory User Group Elements come from a central User Directory. The G/On User Groups are the result Elements of the Rules in this perspective. This means that for each Rule any User or Directory User Group on the left hand side means they are placed in the G/On User Group at the right hand side.

Usage

Rules can be added, edited and deleted. See page 122 and onwards for general information on how to do this. For general information on how to add Elements to a new Rule or to an existing Rule, see page 123.

Perspective: Action Authorization Policy



The Action Authorization Policy perspective is used for creating Rules specifying when to authorize the use of specific Menu Actions. The authorization may depend on the Authentication Status and the User Group membership of the current User. It may also depend on the presence of a Token in a given Token Group.

Rule Elements

The Authentication Status Elements are possible results of User Authentication Policy Rules, that have been set up in the User Authentication Policy perspective.

The User Group Elements come from one of the User Directories that G/On has been configured to work with - or the G/On User Groups defined in the G/On User Group Management perspective. The Groups can be used for giving different groups of Users access to different Menu Actions. For example, management-users may need access to different applications than guest-users.

The Token Group Elements are defined in the Token Group Management perspective. In Authorization Rules, Token Groups can be useful as a way of identifying groups of PCs which must be used, in order for certain actions to be authorized. For instance, the PCs could each have a MicroSmart Token or a smart card inserted in a build-in reader or they could each have a

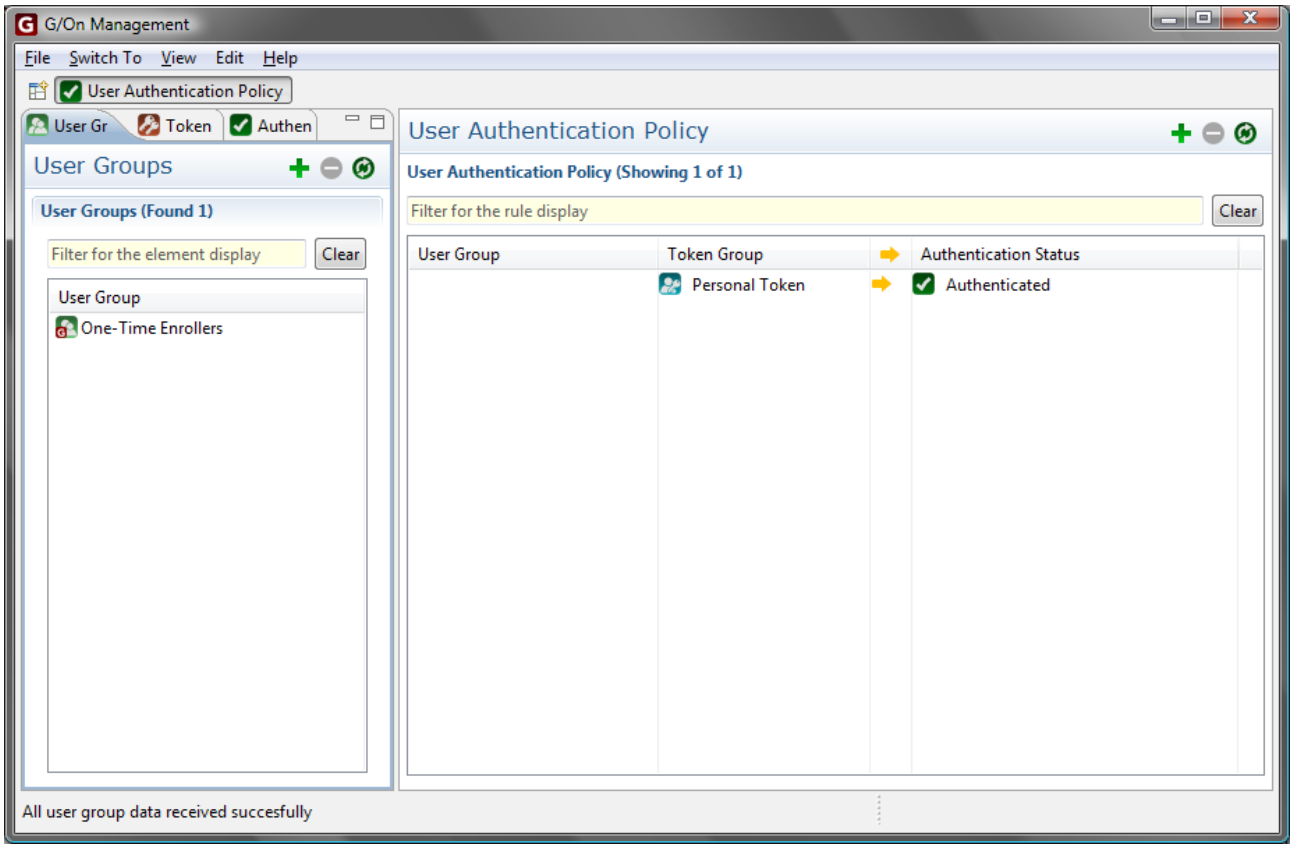
Computer User Token installed, which could then be used to identify them.

Menu Actions are the result Element in the Rules in this perspective. This means that if all the specified parameters on the left hand side of a Rule are true, then the User will get access to the Menu Action on the right hand side of the Rule.

Usage

Rules can be added, edited and deleted. See page 122 and onwards for general information on how to do this. For general information on how to add Elements to a new Rule or to an existing Rule, see page 123.

Perspective: User Authentication Policy



The User Authentication Policy perspective is used for creating Rules specifying when to conclude that the current User has a given Authentication Status. The conclusion may depend on the User Group membership of the current User. It may also depend on the presence of a Token in a given Token Group.

For example a Rule can say that all Users are properly authenticated if they are using a Personal Token. The use of a Personal Token implies that the User has also logged in. So this Rule says that a User who is logged in and is using his/her Personal Token is authenticated. Another Rule could say that any User in the “Production” User Group is authenticated if using any Token from the “Production” Token Group.

Rule Elements

User Group Elements are either from a User Directory or are G/On User Groups created in the G/On User Group Management perspective. These Elements can be used to give different groups of Users different means of authentication.

Token Group Elements are either the built-in Element Personal Token or any Token Group created in the Token Group Management perspective. The Personal Token group is dynamic, in the sense

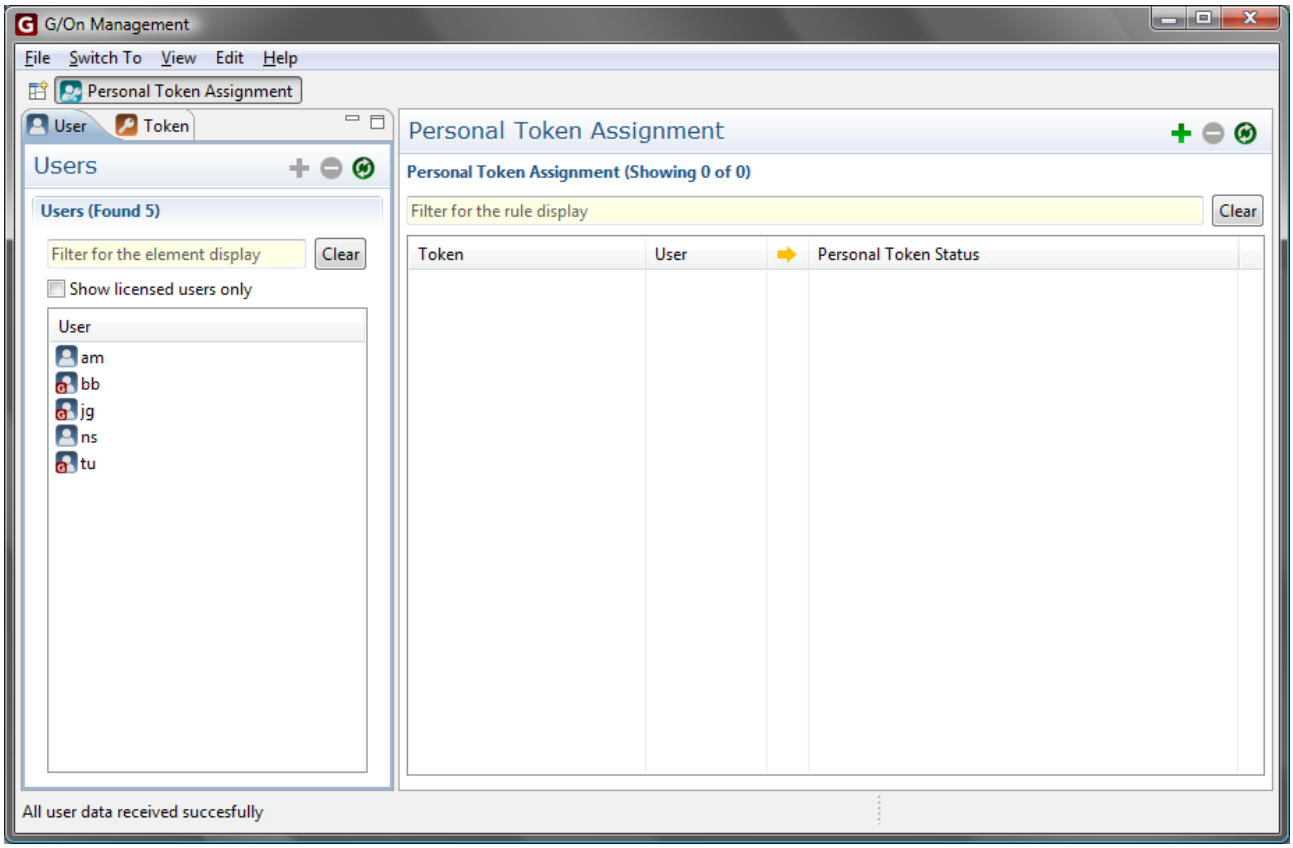
that it depends on the current User.

Authentication Status Elements are the result Elements in the Rules in this perspective. If all the specified Elements on the left hand side of a Rule are true, then the Authentication Status Element on the right hand side is also considered to to be true. It is possible to create new Authentication Status Elements to get a more fine grained notion of authentication. However, in most cases this would be an unnecessary complication, because it would result in a combinatorial explosion of the number of Action Authorization Rules.

Usage

Rules can be added, edited and deleted. See page 122 and onwards for general information on how to do this. For general information on how to add Elements to a new Rule or to an existing Rule, see page 123.

Perspective: Personal Token Assignment



The Personal Token Assignment perspective is used for creating Rules that link a Token to a unique User so that it becomes the User's Personal Token.

Rule Elements

Token Elements are created by enrolling each individual Token. After enrollment, the Token may be entrusted with a specific User, for use as a second authentication factor. For the Rule engine to know which User has which Token, a Personal Token assignment Rule has to be created. A Token cannot be the Personal Token of more than one User (then it would not be personal).

User Elements come from a User Directory.

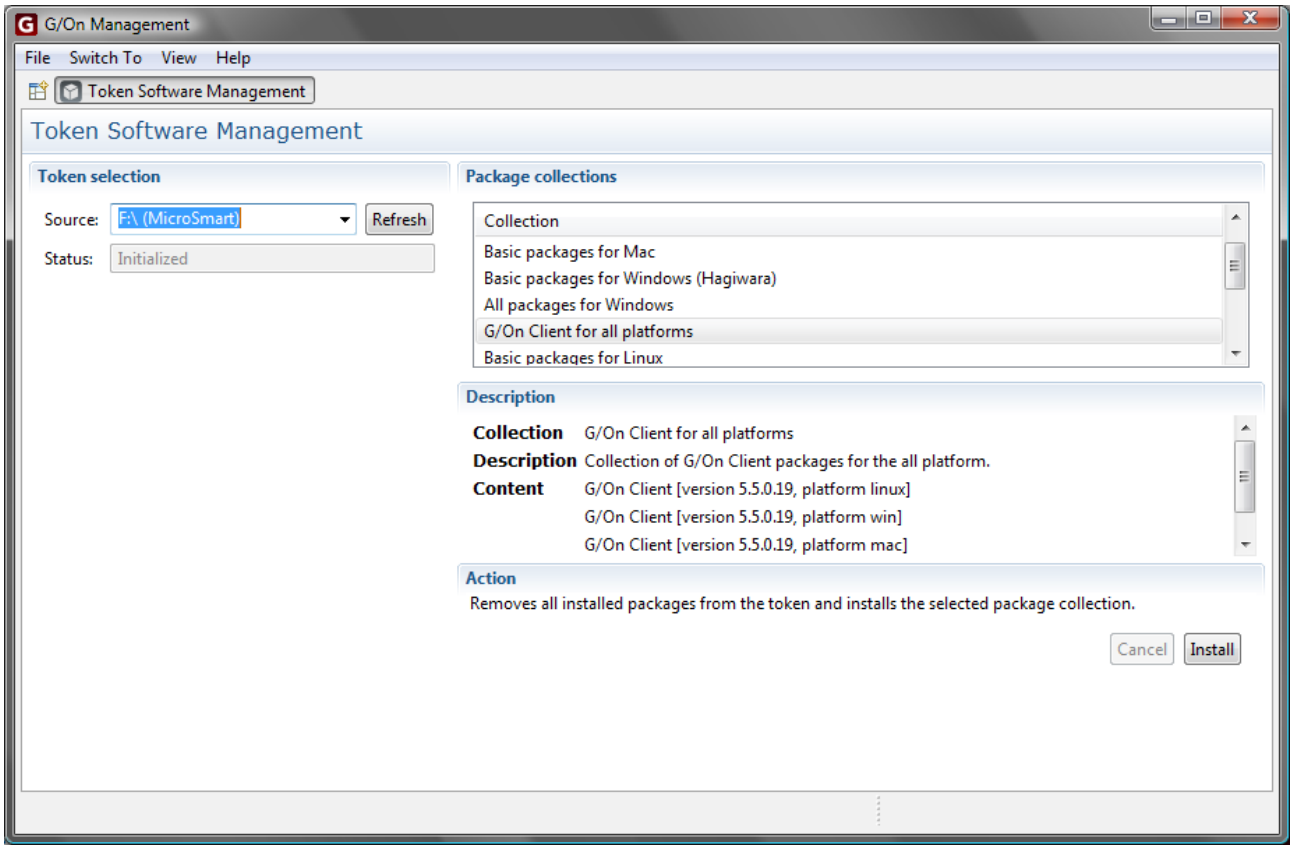
Personal Token Status Elements are the result Element of the Rules in this perspective. It is *not* possible to create other Personal Token Status Elements.

It is not possible to have empty fields in these kind of Rules.

Usage

Rules can be added, edited and deleted. See page 122 and onwards for general information on how to do this. For general information on how to add Elements to a new Rule or to an existing Rule, see page 123.

Perspective: Token Software Management

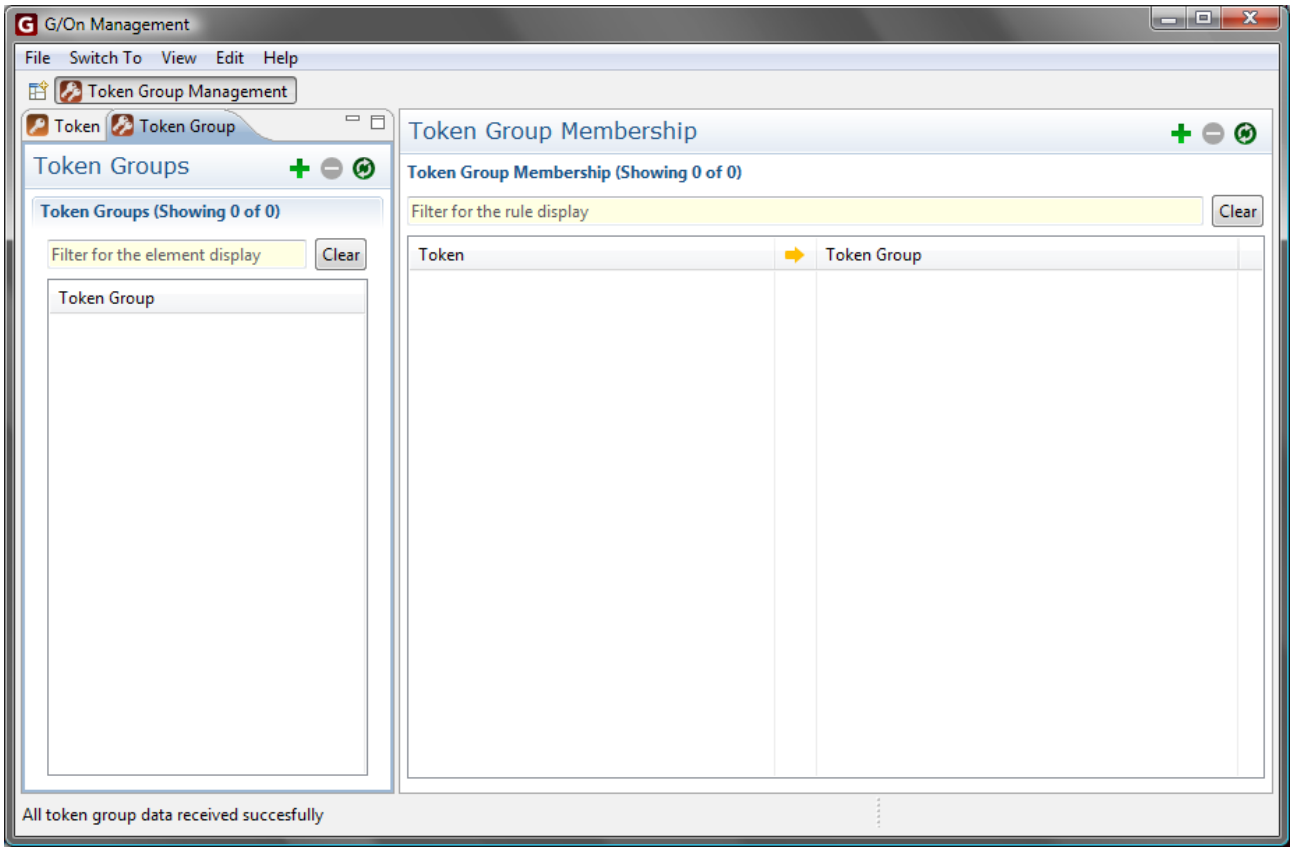


The Token Software Management perspective is used for installing software package collections to Tokens before handing them to the Users. The Tokens in the source list are the Tokens inserted in the USB ports of the local workstation.

Note: Installing a package collection will overwrite the contents of the Token: Existing files will not be deleted but may be replaced. Files unrelated to G/On will be untouched.

Also note: After the installation of a package collection, it will appear to G/On client as if only the packages in the given package collection are on the Token – even if there are in fact files on the Token, from packages that were installed earlier.

Perspective: Token Group Management



The Token Group Management perspective is used for adding Tokens to Token Groups. A newly created Token Group is empty and the way to add Tokens to that Token Group is by creating a Rule for each Token saying that it is a part of a particular Token Group.

Token Groups can be useful for identifying specific sets of PCs, assuming that they each have a fixed Token in/on them. They can also be used for identifying, e.g., a set of guest Tokens that are not personal.

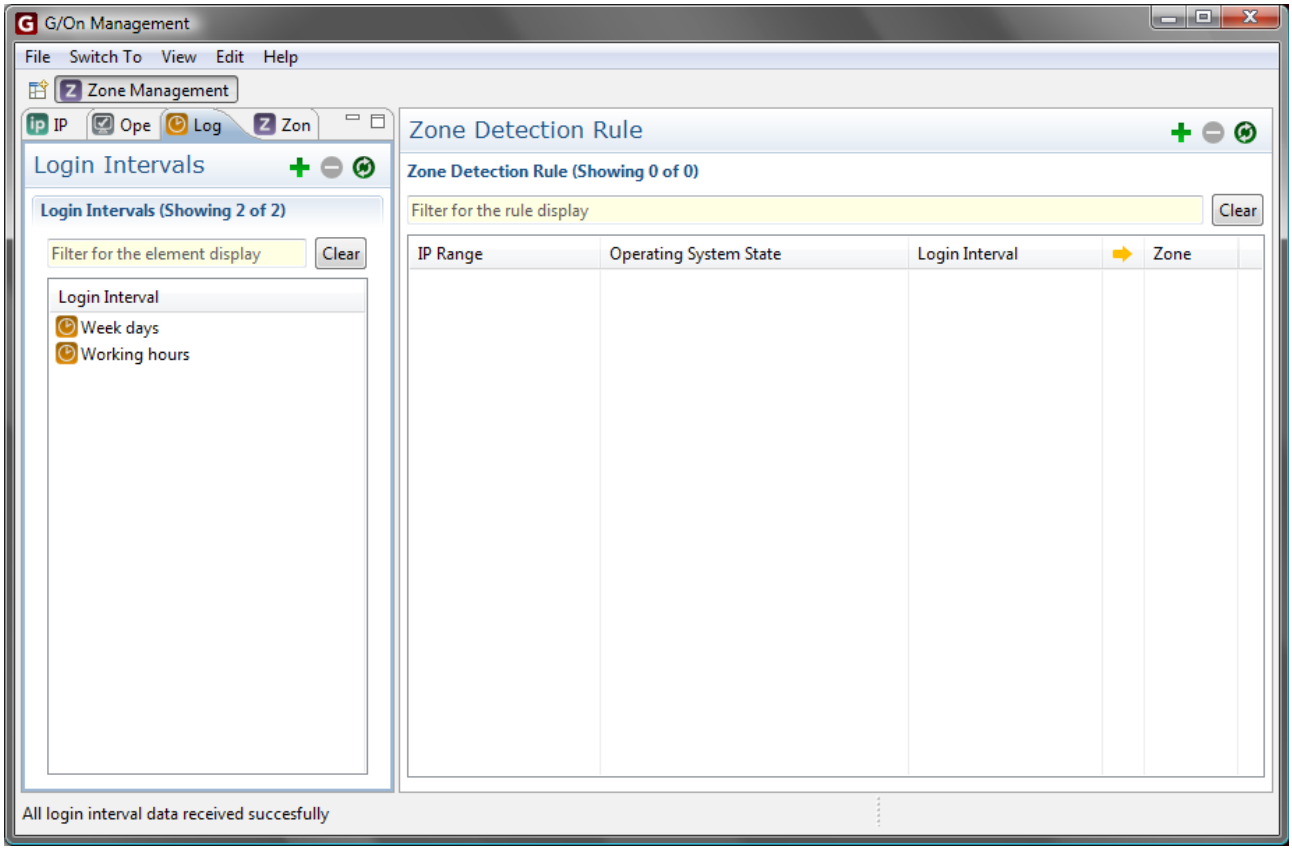
Rule Elements

Tokens are created by enrolling each individual Token into the G/On server. Once a Token has been enrolled, it can be added to an existing Token Group. Token Group Elements are the result Elements of the Rules in this perspective. Each Rule says that a given Token is a member of a given Token Group.

Usage

Rules can be added, edited and deleted. See page 122 and onwards for general information on how to do this. For general information on how to add Elements to a new Rule or to an existing Rule, see page 123.

Perspective: Zone Management



The Zone Management perspective is used for creating Zones and Zone detection rules.

Zone detection rules is defined using IP ranges, Operating System States and Login Intervals. A Zone can be attached to a Menu Action, thereby creating the restriction that the Menu Action can only be launched if the circumstances of the Zone are fulfilled.

Zone restrictions is a more soft type of authorization, where access to the menu is not denied, but access to launching Zone restricted actions may be denied. The user is therefore able to see that under the right circumstances the action would be available.

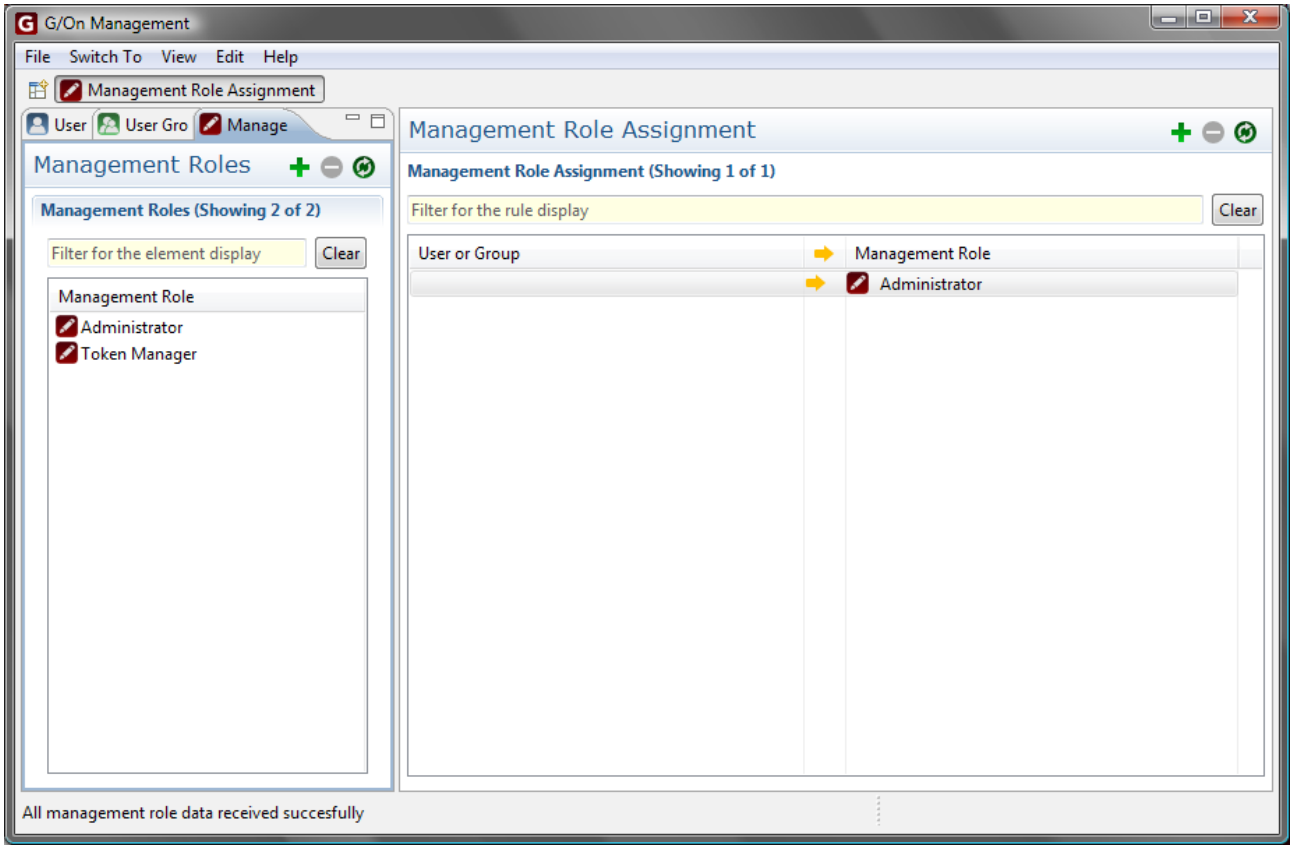
Rule Elements

Rule elements are IP Ranges (page 151), Operating System State (page 153) and Login Interval (page 156).

Usage

Rules can be added, edited and deleted. See page 122 and onwards for general information on how to do this. For general information on how to add Elements to a new Rule or to an existing Rule, see page 123.

Perspective: Management Role Assignment



The Management Role Assignment perspective is used for creating management Access Roles and assigning them to users. Each Management Role define a set of access rights for management functionality. Using the Management Role Management it is possible to delegate responsibility for different tasks in G/On Management and only giving people access to the functionality they actually need for the task. A sample Token Manager role is provided, which provides access to the perspectives and views necessary to manage tokens, i.e. enrollment, assignment and software deployment of tokens.

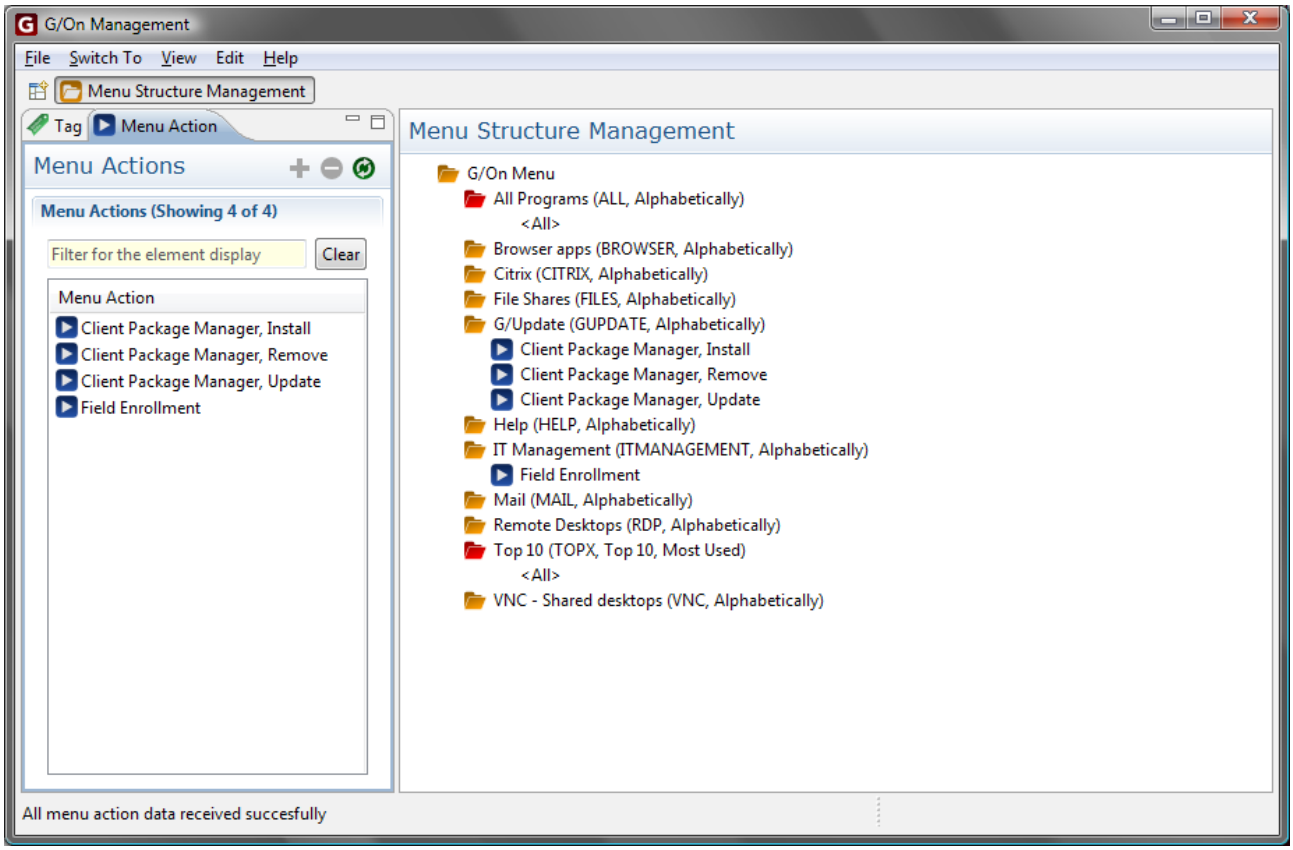
Rule Elements

The role members can be individual users or user groups, including G/On User Groups. The users, which can log in to the G/On Management are drawn from the same user directories which are defined for access via G/On Client. A Role condition can also be left blank, meaning that anyone has that role. The initial setup is that everyone has the Administrator Role, thus it is not necessary to login. If a user is member of more than one role, the user will get the union of access from both roles, i.e. if access to e.g. Users is read-only in one role and read/write in the other then the user will have read/write access to Users.

Usage

Rules can be added, edited and deleted. Note however that as a special precaution there must always be at least one active rule defining access to the Administrator Role. Thus it is not possible to delete or change the result of a rule giving access to the Administrator Role if this rule is the only one doing so. The reason for this restriction is to minimize the risk for accidentally locking yourself out of editing Management Roles. See page 122 and onwards for general information on how to edit and delete rules.

Perspective: Menu Structure Management



The Menu Structure Management perspective is used for structuring the content of end-user's menus. Each Menu Action has a number of Tags associated to it. For example, a Tag named “BROWSER” may be associated with the Menu Action “Mercurial intranet site” that launches an intranet site using Windows Explorer. If this Tag is added to the menu structure, the Tag acts like a folder containing any Menu Action that is associated with the Tag. Because a Menu Action can have any number of Tags associated with it, the Menu Action can appear several places in the menu structure.

Elements

Tags can be created by entering them in the Tags field of a Menu Action. They can also be created by the New operation in the Tag list. Tags have a number of settings. One of the settings decide whether the Tag generates a menu folder that can be used as a container for Menu Actions. See the Tag Elements description for more information.

Usage

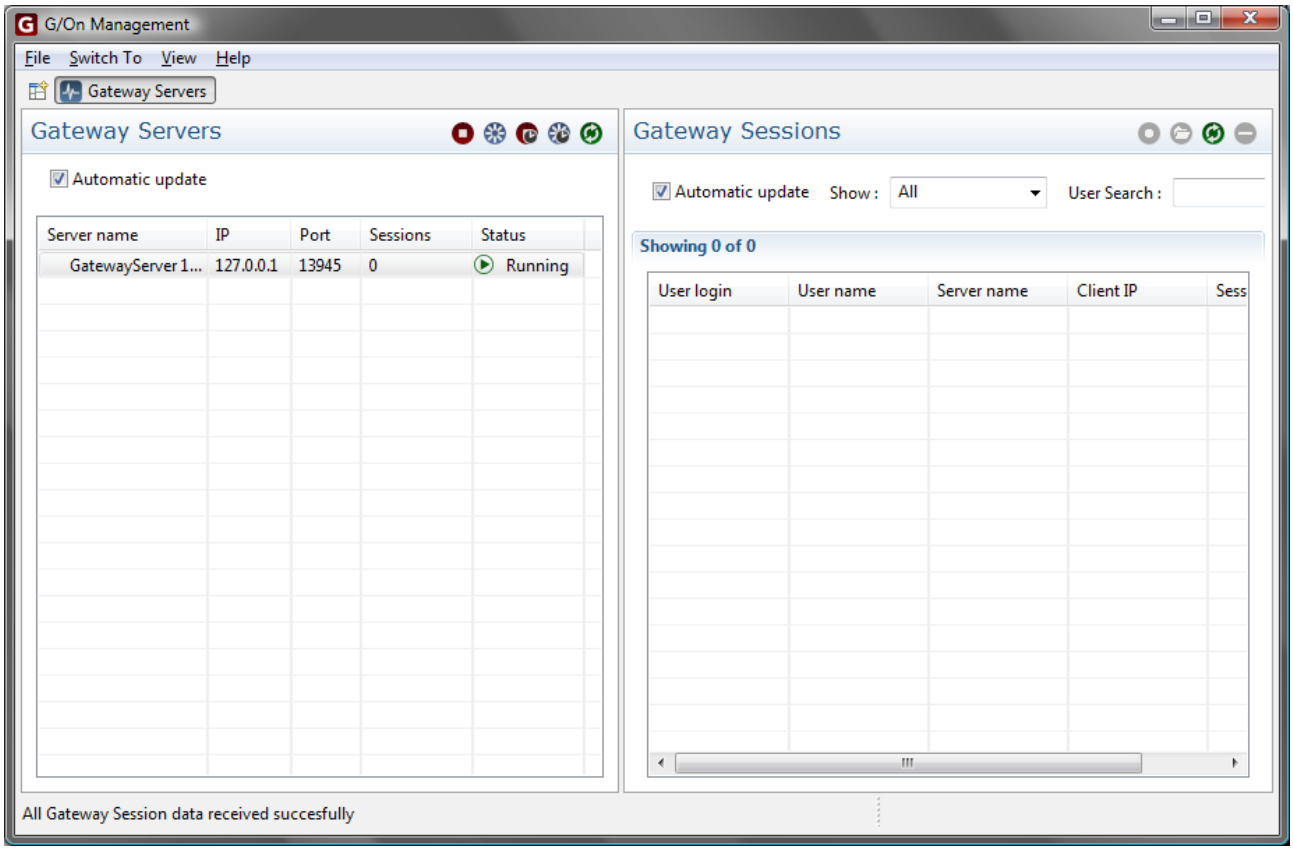
Tags are added to the folder structure by dragging and dropping. Tags can be dragged onto other Tags in the menu structure in order to create sub-menus.

Menu Actions can be created and edited in the Action Authorization Policy perspective. Do not try to add Menu Actions to a specific location in the menu structure. Instead, you should add Tags to the individual menu actions in the Action Authorization perspective and then let the Tag system handle locations.

To create a new folder in the menu structure start by creating a new Tag. The new Tag has a name and that name should be added to the Tag list in the Menu Actions that should go into the folder that the Tag generates. Notice that Tags can be added to Menu Actions in the Action Authorization Policy perspective.

Note: Menu Actions cannot be created in the Element list in this perspective. Use the Action Authorization perspective when you want to create Menu Actions.

Perspective: Gateway Servers



In the Gateway Servers perspective it is possible to monitor and control Gateway servers and the users sessions running on them. The perspective consists of two parts: Left is a view of Gateway Servers and their status and right is a view of Gateway Sessions. These views are described below.

Gateway Servers

This view contains a list of running gateway servers and their status. The following fields are shown for each server:

- **Server name** is the title of the gateway server defined in the *gon_server_gateway_local.ini* file or “Not defined” if no title has been defined.
- **IP** is the IP address of the gateway server host.
- **Port** is the port on which the Gateway server listens.
- **Sessions** is the current number of user sessions.
- **Status** is the server status, which would normally be “Running”. It can be changed by choosing server actions, see below for details

Above the list is a check box titled “Automatic update”. If this box is checked then the gateway server list will be automatically updated each 30 seconds.

In the “Gateway Servers” header there is a number of buttons which can manipulate the servers and the view:

- **Stop** will send a signal to the chosen gateway to stop immediately. Any user sessions on the server in question will be disconnected.
- **Restart** will send a signal to the chosen gateway to restart immediately. Any user sessions on the server in question will be disconnected.
- **Stop when no users** will send a signal to the chosen gateway to stop as soon all current user sessions has stopped. This will also block the chosen gateway server from receiving any new client connections.
- **Restart when no users** will send a signal to the chosen gateway to restart as soon all current user sessions has stopped. This will also block the chosen gateway server from receiving any new client connections.
- **Refresh** will refresh the gateway servers view.

The first four actions will result in the gateway server Status field changing to “Stopping”, “Restarting”, “Stop when ready” and “Restart when ready” respectively. The actions are also available in the right-click menu.

Gateway Sessions

This view contains a list of gateway (user) sessions.. The following fields are shown for each session:

- **User login** is the (full) user login of the user logged in to this session or blank if no user has logged in yet.
- **User name** is the registered name of the user logged in to this session or blank if no user has logged in yet.
- **Server name** is the name of the Gateway server the session belongs to (see description under Gateway Servers above).
- **Client IP** is the IP address of the G/On client
- **Session start** is the date and time the session was initiated.

Above the list are som options:

- **Automatic update:** If checked the session list will be automatically updated each 30

seconds.

- **Show:** Choose to show all sessions or only those belonging to the selected Gateway Server in the Gateway Server list.
- **User search:** Find sessions for which the user name or login contains the content of the search field.

In the Gateway Sessions header the following actions are available:

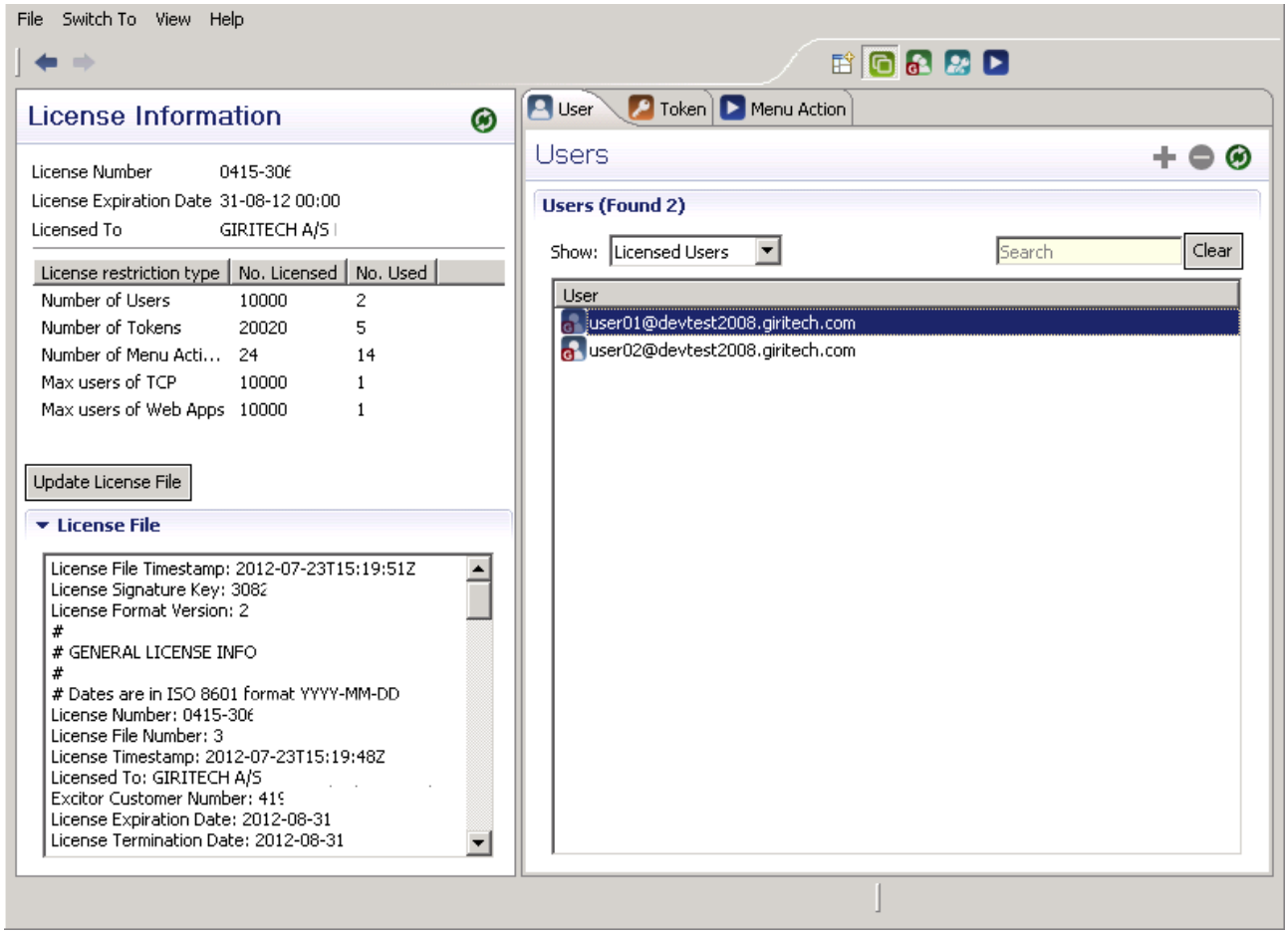
- **Stop Session** will send a signal to stop the selected session immediately.
- **Recalculate Menu** will cause the client menu to be recalculated. This functionality is described below.
- **Refresh** will refresh the list
- **Cancel Updating** will stop fetching more sessions from server (only available if there are more than 50 sessions).

These actions are also available in the right-click menu.

Menu Recalculation

The menu recalculation feature makes it possible to force a recalculation of the menu for a running G/On session. It is mainly a tool for an administrator to be able to test authorization and menu action setup, without having to restart a G/On client and log in constantly. Therefore the recalculation is not a full recalculation in the respect that all the basic authentication factors like user login, token identity, computer properties, IP address, etc. are assumed to be the same. It is the possible new authorization based on these premises, which is calculated. For example, if a new rule is added giving access to a new Menu Action for a user, then a recalculation should result in that the new Menu Action is part of the users menu. However if a new token has been enrolled and assigned to a user, a recalculation will not give further access, since the token identity was not established at login time.

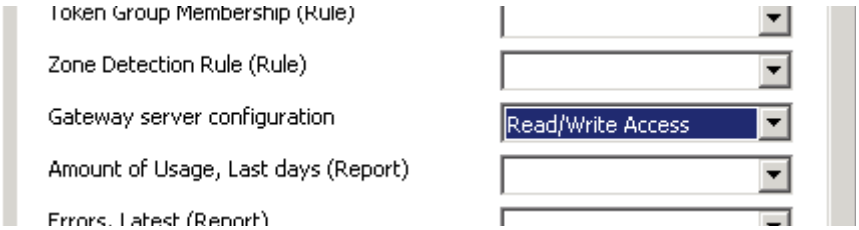
Perspective: License Management



The License Management perspective is used to give an overview of the number of licenses available and used. The perspective can also be used for updating the License File.

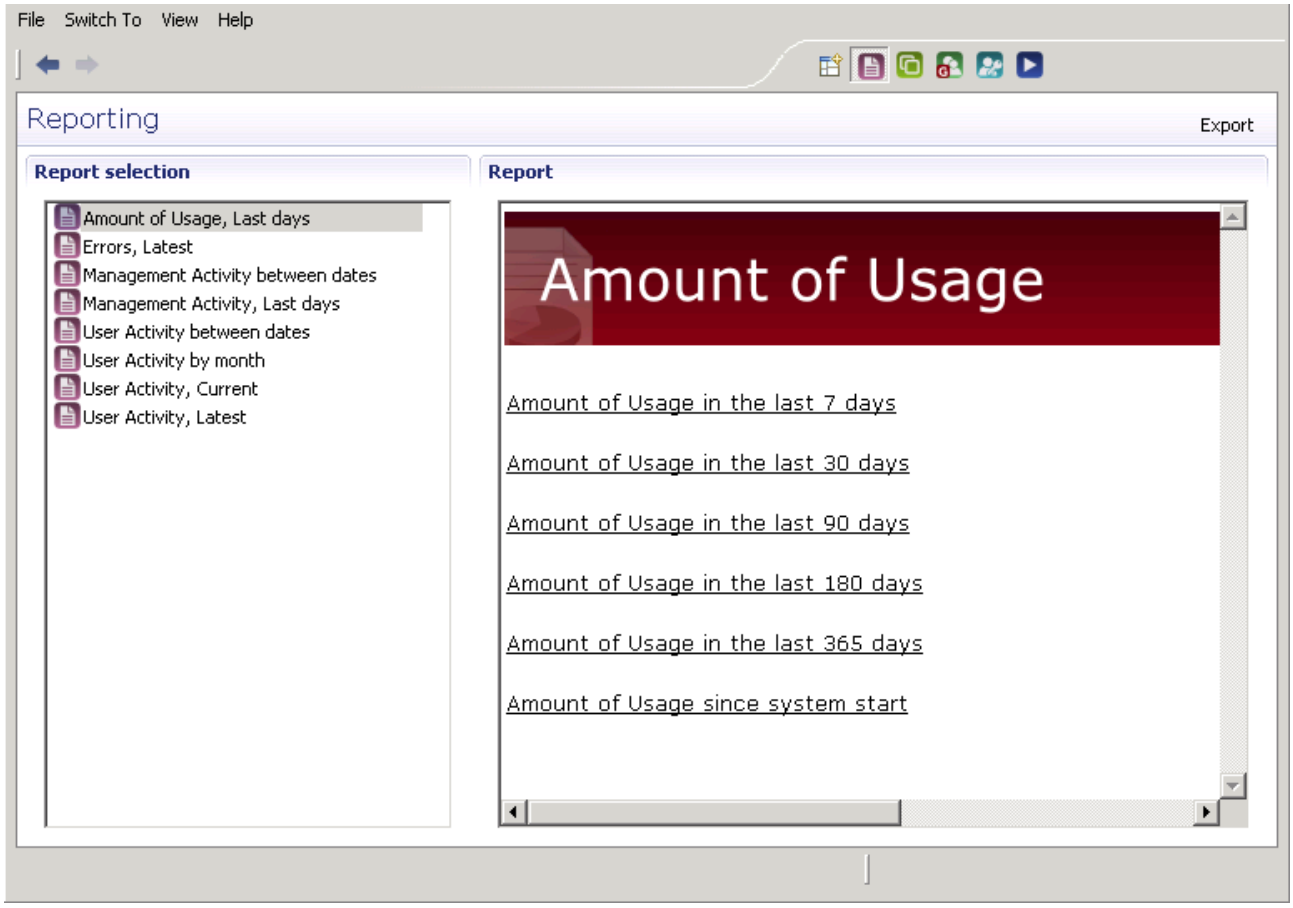
The User, Token, and Menu Action tabs in the right are there for a quick view of what elements are used.

Note: If Management Roles are used (see *Perspective: Management Role Assignment* on page 170 and *Element: Management Role* on page 148), the person updating the license file needs to have read/write access to Gateway server configuration:



Note: If the current license is invalid, it will not be possible to start G/On Management.

Perspective: Reporting



The report perspective can be used for retrieving information about the usage of the system. Double click on any of the reports listed in the report selection list too see the selected report in the report viewer on the right hand side. Click on hyperlinks in reports to see more information.

Reports

A number of reports are available to the G/On management client user:

- **Amount of usage** provides a list of users and their complete online time.
  - *Last days*: Choose from the last 7, 30, 90, 180, 360 days, or usage since the system was started.
- **Errors** provided a list and details of errors. This can be useful if, for example, a user can't log in and don't know why.
  - *Latest*: Provides data from the last 3 days.
- **Management Activity** provides a list of the changes made in the Management client.
  - *Between dates*: Choose two dates and get the changes made between these dates

(both inclusive).

- *Last days*: Choose from the last 7, 30, 90, 180, 360 days, or usage since the system was started.

- **User Activity** provides a list of user activity.
  - *Between dates*: Choose two dates and get the user sessions overlapping the period between these dates (both inclusive).
  - *By month*: The same as Between dates, except this quickly provides the list for a full month.
  - *Current*: Shows users currently online.
  - *Latest*: Show a list of users and their last online session (including currently online sessions).

Export reports

It is possible to export any of the reports to a document formatted as a PDF file. At the top right of the window is a button the says 'export'. Click this button to start exporting the currently visible document.

Best Practices

Tokens

What is the best practice for handing out Tokens?

Users should not share Tokens. For the most secure setup, Tokens should be personal, so a User has to present a Personal Token, in order to prove his or her identity – not just any token, that could have been used by a number of other people. This practice also provides grounds for better usage reporting.

That being said, it is possible to create a Token that should be used by a group of Users. This feature should primarily be for “guest tokens” or Tokens handed out to a group of Users with less strict requirements for secure, individual authentication, such as a group of Users working for a contractor. Create a Token Group and call it, for example, 'guest tokens' and add the Token to this Token Group. Then create an Authentication Policy stating that Users from the Active Directory group 'guests' can be authenticated using the Tokens from the 'guest tokens' Token Group.

Elements

What is the best practice for using the built-in Personal Token Status Element called Personal Token?

The Personal Token defines a special Token Group that evaluates to true when a specific Token and a specific User is active in using the G/On client. In most cases it is practical to link each Token to a specific User, so it becomes a Personal Token for this User. In the Authentication Policy perspective a Rule can be created, saying that when a Personal Token Element is validated, then User is properly authenticated.

The reason for this seemingly extra step is to allow for the creation of Authentication Policy Rules using Token Groups which contain specific Tokens. This will allow for a slightly looser authentication concept combining Token pools and User Groups from a central User and Groups Directory.

What is the best practice for using the built-in Authentication Status Element called Authenticated?

Use the built-in Authentication Status Element called Authenticated for labeling when a User can be considered to be properly authenticated. Do not introduce different level of authentication strength unless it is really needed. Set up a policy for when Users are properly authenticated and

use this notion in Authorization Policies. The Authorization Policies are the proper place for distinguishing which Menu Actions different groups of Users should be allowed to use. Introducing different levels of authentication in many cases adds unnecessary complexity to the Rule engine and in particular to the Authorization Policy Rules.

FAQ

General

How is it enforced that traffic from a client-side application hits the right server on the right ports?

The explanation is quite simple: The menu items shown to the User contain nothing but a name. So when a User chooses a Menu Action, the name is sent to the server, which decides what application server addresses and ports to connect to.

Going into more details, this is an overview of G/On Server behavior with focus on how the server controls access from the G/On Client to application servers:

1. The G/On client connects to the G/On server and authenticates the server and they establish an encrypted communication channel
2. At this point, the server accepts nothing from the client, except info regarding authentication and authorization factors: username, password, responses to challenges to authenticate Tokens, etc.
3. Based on the authentication and authorization information received from the client, the server computes which actions the user is authorized to do, and sends out a menu to the client. Each menu item is simply a title and the name of the associated action – there is no information sent out to the client about the meaning of the Menu Actions.
4. At this point the server now also accepts information about the names of Menu Actions, that the User has chosen – but only the ones that were authorized by the server itself.
5. The User chooses a Menu Action, and the client sends the name of that Menu Action to the server.
6. Based on the Menu Action name, the server looks up the definition of the Menu Action, and establishes a port forward as specified in the definition. The port forward is established by setting up a component on the G/On server and one on the G/On client, with a communication channel between them. The port forward component on the server will – when receiving traffic from the client part – open a connection to the application server address and port, as specified in the Menu Action definition. So the client has no control over which addresses and ports that the server connects to.

How to run the G/On Management Client remotely?

In many cases it is not convenient to run the G/On Management client on the server machine. For instance, it is often not practical to enroll tokens, by physically having to plug them into a USB port on the server machine. If you wish to run the G/On Management client remotely, through a G/On connection, do the following:

1. Initially, you need to enroll *at least one* Token and install the G/On Windows client on it, using the server machine. If you cannot access a USB port on the server machine, you can create a "soft Token" on the hard disk and then copy it to a location where you can put it on an ordinary USB flash drive. See the section Initialization of Soft Tokens on HD in the Advanced Setup Topics section of the G/On Setup and Configuration Reference. Alternatively, make a field installation of a Computer User Token, as described in the separate document: G/On Field Deployment.
2. On the Management server machine use G/On Management and create a menu item for starting the Management Client and authorize it for the Users who are going to use it remotely.
3. Start the G/On client on the Token that you created in step 1. In the G/On menu, select G/Update – Install, and install the G/On Management and Management service packages.
4. In the G/On menu, choose the menu item that you created in step 2.

Will the end-user client automatically reconnect?

It is not currently possible to have the client reconnect if the network temporarily fails. Security concerns makes this a non trivial task. So the user will have to restart the client to reconnect.

Rules

Why can't I make Rules for individual Users in the policy perspectives?

The policy perspectives are meant as general policies that should not be changed often. The day to day management of the G/On server should be about assigning new Tokens to Users and handing them out. The Users should in most cases fall into a User Group from the central User Directory that already has the proper authentications.

Elements

I see a tab called Rule Elements?

Click the tab to refresh the tabs header.

Menu Actions

How to specify multiple port forwards

Use the template called (default), when creating the Menu Action: First enter information in the Server Host field, then save, and open again, and enter information in the Server Host 1 field, etc.

Tokens

How to enroll Tokens without having to plug them into the server machine?

The G/On Management client can be run remotely through a G/On connection. See the FAQ entry above, on How to run the G/On Management Client remotely. So you can run it on a PC of your choice with a USB port where you can plug in the Tokens to be enrolled.

There are no Tokens listed in the Source drop down list?

Tokens should be listed in the source drop down. Make certain that you have inserted a proper Token into the local machine. Click Refresh to refresh the drop down list.

Note: however, that Tokens from which a G/On client has been started (and is still running) will not be listed in the drop down list.

The Enroll button is inactive (grey)?

If a Token is selected in the drop down list, and the Enroll button is inactive (grey), this indicates that the Token is already enrolled in the system. It must be deleted from the Token Element list, before it can be re-enrolled.

Can a Token be used by a group of Users?

Yes it can but it is not the best practice. Best practice is to have each Token identifying a specific user. However it is possible to create a Token Group and add a token to that group. In the User Authentication Policy, link the created Token Group and a User Group to say that those Users can use the Tokens from the newly created Token Group and then be properly authenticated.

Users

Can I create G/On Users?

It is not currently possible to create local individual Users in the G/On management. It is possible to create local G/On User Groups. This means that it is possible to collect groups and individuals from the User Directory into one User Group and use that to simplify the policies and other Rules. This should also be a help if it is not possible or viable for the G/On administrator to create new groups in the central User Directory.

Can I force an update of the user menu?

Yes, it is possible to update a user menu from G/On Management. See page 176 for further details.

When will a login window appear to the Users?

If there is a Rule where the system needs to know who the user is then the login window will appear. For example if a Rule says that a User is properly authenticated if he is using a Personal Token. Then the system needs to know who the User is and which Token he is using.

Messages

I get the error: 'Unable to connect to server'?

If you get a message saying: “Unable to connect to server”, when starting the G/On Management client, try adjusting the preferences (by choosing View > Preferences).

I get the error: 'This element cannot be deleted'?

Certain Element lists have predefined and built-in Elements intended for use in best practice based set-up. These Elements can not be deleted.

I get the error: 'Unknown element type [type]'?

This can happen if the perspective has Element lists that hold types that should not be added to the current Add/Edit Rule area. The perspective has probably not been properly refreshed. Choose View > Reset Window. This should reset all the perspectives views and lists.

I get the error: 'Token has already been used in another rule'?

Tokens should identify one User only. If you want a Token to identify a different User, first delete the Rule where the Token is currently used.

An end-user gets a notification: 'Insufficient authorization' in the G/On client?

If it turns out that the User is not authorized to do any Menu Action, this notification will be displayed, and the G/On client will terminate. This may happen for different reasons, depending on how the authentication and authorization policies have been set up. These two scenarios are often seen:

1. The User has not presented the right User Name/Password/Token, so he is not properly authenticated, and all Authorization Policies require proper authentication.
2. The administrator has forgotten to set up some Action Authorization Policies. So this User is not authorized to do any Menu Action, even when he is properly authenticated.

Menus

What are menus in G/On 5?

The menu which is shown when you log into G/On is created from the *Tags*, which are attached to the Menu Actions you have access to. In other words, when you log into G/On, the system first calculates which Menu Actions you have access to, then it builds the menu based on the Tags attached to these Menu Actions. The Tags are both used for sorting out or disabling irrelevant actions (e.g. a Linux based application when running Windows) and as building blocks for the menu tree. On each Tag you can choose whether it should be shown as a menu or not. If the Tag is set to be shown as a menu and you have access to one or more Menu Actions with this Tag attached, then it will be shown in the menu tree containing the Menu Actions in question. The Tag can also have parent Tags, in which case it will be shown as a sub menu in all the menu folders created from these parent Tags.

What does the Menu Structure Management view show?

The Menu Structure Management view shows the menu tree derived from all the Menu Actions which have been created in the system. In other words it shows you how the menu would look for a User who has access to, and is able to run all Menu Actions in the system. So for most Users the menu will not look like the one you see in Menu Structure Management. But it would however always be a *subset* of the menu tree shown, in that the User's menu would consist of the Menu Actions available to the User at the same position(s) in the menu tree.

Why can I not create a new Menu Action?

You can create Menu Actions and specify who are allowed to use them in the Action Authorization Policy perspective. This automatically puts the Menu Actions into the menus. So you do not need to go into Menu Structure Management perspective. You only need to open the Menu Structure Management perspective, if you have some special requirements regarding the structure of the menu (new sub-menus, or sub-sub menus etc.). So the Menu Structure Management perspective, is only for the final “polishing” of how the menu will appear. The main part of the work: deciding who get access to which Menu Actions, you do in the Action Authorization Policy perspective. In order to promote this new work flow, the creation of Menu Actions has been disabled in the Menu Structure Management perspective.

How do I create a personal menu?

The short answer: You don't! Or more precisely: You don't have to – the menu is already personal.

In G/On 5, the personal menu is created in two steps. First, access to an application is given in the Authorization Policy view, in which you can give access to a certain application for a specific group of authenticated Users. As a special case a User Group could consist of just one person, but we recommend not to name the group after the person, because we find that almost all authorization is given to people because of their *role* rather than because of who they are. As an example, most G/On administrators like to have a personal menu, in which they can add administrative applications. In G/On 5 this would be solved by creating a G/On Administrator group either in the User Directory (e.g. AD) or in G/On Management. Then access to the applications can be given to this User Group (along with the “Authenticated” condition). But if you really want a personal User Group, it is possible to create it in the G/On User Group view.

How should I manage menus in G/On 5

One of the objectives of using tags for creating menus is to diminish the amount of work regarding menu management. Consider the task of adding a new application to G/On: In G/On 3 you would create the application string, and then you would manually enter the new application into the menus of the users or groups who should be able to access it. In G/On 5 you will normally create the application (Menu Action) using a template. Then you have to decide who should have access to it (and under what circumstances) and create corresponding rules. And in most cases that's it - you don't have to explicitly add the application to the menu, it will already be there underneath the menu folders derived from its tags. You can of course change this, either by changing the tags directly on the Menu Action element or by using Menu Structure Management.

Predefined Menu Action Templates

The G/On system comes with a number of predefined Menu Action templates. Most of these are self explanatory, but a few need a specific setup of the server, or have other prerequisites. These are documented in the following.

FileZilla Template

FileZilla is an FTP client, which connects to an FTP server using the FTP protocol. It can operate in two modes: Active or Passive.

The FileZilla client starts by "telling" the FTP server whether to use active or passive mode.

In active mode the server will try to initiate connections back to the client, based on information that the client has supplied about its address and a port. Opening connections from the server to the client is not supported by G/On, so this mode cannot be used with G/On.

In passive mode, the server will dynamically select a new port for data traffic, and send information to the client that it should connect to this port. However, this is not possible if the G/On connection is simply one port forward from the FTP client to the FTP server: G/On does not "know" that it has to open a new port forward, for the data traffic. The solution is to configure FileZilla so it uses the SOCKS protocol, and then set up a SOCKS server on the server side.

Setting up the Server Side

Install the GSOCKS service, e.g. on the same server as the one running the G/On Servers, and set up the gsocks.ini file, so it allows access to the desired FTP server.

Using the template to define menu actions

Notes regarding selected fields in the template:

Server Folder The folder on the FTP server, which is shown in the FileZilla client, when connected. Unfortunately, Filezilla uses a very special syntax for specifying the path to this folder. Assuming that the server is a windows or linux/unix server, the syntax is as follows:

```
1 0 l1 name1 l2 name2 ....
```

where *l1* is the number of characters in name1

and name1 is the name of the top level folder

Likewise for *l2* and name2, etc.

For example the Windows path: \Documents and Settings\abc should be written as follows:

```
1 0 22 Documents and Settings 3 abc
```

Notes regarding usage of menu actions generated from the template

Due to the fixed port being used for the SOCKS connection on the client side, it is not possible to have two instances of FileZilla running at the same time through G/On. When trying to launch the second instance is you get an error: "Unable to create port forward - address is in use".

Citrix Web Interface Template

G/On supports a special type of menu actions for creating a http proxy connection between a browser on the client side and a Citrix Web server on the server side. The http proxy implements single sign-on the server side and launch of the Citrix client on the client side, without having to install anything on the client PC or browser.

Setting up the Server Side

We assume that Citrix Web Interface has been configured with:

- "Authentication Point" "At Web Interface"
- "Authentication Method" "Explicit"
- No "secure client access" (just "Direct" Access Method)
- No special "client-side proxy" (just "User's browser setting")
- "Access Method" Allow Users to access published resources using
- browser bookmarks

When configuring or debugging G/On Citrix and Web Interface integration then first try to log on to the web interface directly from a browser, preferably running on the Gateway server.

Verify that you get a login.aspx html page and verify that the User Name and Password you are using for testing G/On works. Note whether a domain must be specified or not.

Verify that the Web Interface application menu contains the expected applications. Check the URL and verify that the Citrix Web Server Address and Citrix Metaframe Path is configured correctly in G/On.

Internet Explorer sometimes hides important debug information, so if it shows an error page instead of the Web Interface through G/On then try to open the same URL in Firefox.

Extending G/On to support other applications

Introduction

In G/On it is possible to include support for new kinds of applications that are not already covered by the existing Menu Action Templates:

1. The software (e.g. the application client) that must be run on the client PCs can be packaged and easily distributed to the client PCs. The format for defining such client software *packages* is described in the following.
2. When preparing a token before giving it to a user, it is possible to put a whole *collection* of software packages on the token. The format for defining special collections with the packages of your choice, is also described below.
3. When introducing a new application, it is necessary to define how this should be started: Which client side program to start, which parameters to supply, and which communication connections to allow to the application server, through G/On. Usually, some of this information will be fixed for all uses of the application, while some depends on the specifics of the actual IT infrastructure. Below, we will introduce a format for describing the so-called *Menu Action Templates*. The templates capture both the fixed and the parameterized parts of the application start-up information, and each template appears in the G/On Management Client as a Menu Action Wizard.

Note: Customizations cannot be upgraded automatically

If you make manual changes to *any* of the files that are supplied by Giritech, it will no longer be possible to automatically upgrade these files. So, every time an upgrade is installed, all manually changed files must be upgraded by hand.

The same goes for all new files that are manually introduced for the purpose of customization.

Packages and Package Collection Specifications

This section describes the syntax and semantics of Packages Specifications and Package Collection Specifications. Packages Specifications are XML files, which define individual client software packages containing e.g. the files of a G/On client or an application client, or some other set of files. Packages Collection Specifications are also XML files; they define often used “bundles” of packages which can be put on a token when initializing it. First we give a description on where the specifications are found and how to create, change or delete them. After that, follow two subsections with descriptions of the details of Package Specifications and Package Collection Specifications.

Creating new or revised package and collection specifications

Package and Package Collection Specifications are located in the folders:

```
.\config\gpm\gpmdefs
.\config\gpm\gpmcdefs
```

The folder already contains pre-defined specifications. If you want to change these, you can do so, but we recommend that you create a copy and edit that instead.

The reason for this recommendation is that **customized files cannot be upgraded automatically.** So, every time an upgrade is installed in the future, all manually changed files must be upgraded by hand.

In particular, it is recommended not to change the package specifications for the G/On clients. If you do this, the package specifications will not be automatically upgraded, which means that the version numbers will be unchanged. This in turn means that the new clients will not be pushed out to or offered to the users after the upgrade – unless you manually upgrade the changed package specification files.

You can create new specifications. The easiest way to make a new specification is simply to copy one of the existing ones, and change it. Use, e.g. Wordpad for editing, or an editor that is good for editing XML. Note that all packages and package collections must have a unique name (see the following subsections).

After making changes to package or collection specifications, always click the button: “Generate” in the section: “Software Package (GPM) generation”, in the G/On Configuration program. This will generate the actual packages and recompute cached information about the collections.

The results are placed in the folder:

```
.\config\gpm\gpms
```

To remove a package or collection, remove the appropriate XML file, and also remove any unwanted gpm package file from the folder:

```
.\config\gpm\gpms
```

Note: After changes, you may need to restart G/On Management (the client – not the Management Server), in order to use new/changed package collections. And you will need to restart G/On clients, in order that new/changed packages can be installed/updated, using the G/Update menu actions.

Package Specifications (gpmdef.xml)

This section describes the package specification format **gpmdef**. From a package specification, a package(gpm-file) can be generated.

The package specification contains meta information about a package e.g. version and description, and it contains the selection of which files should be included, and where they should be placed, when the package is installed.

The following example is the package specification of the gon_client package for the mac platform:

```
<?xml version="1.0" encoding="UTF-8"?>
<gpmdef>
  <header name="gon_client">
    <version main="5" minor="5" bugfix="0" build_id="21" />
    <version_release main="1"/>
    <arch>mac</arch>
    <description lang="en"
       summary="G/On Client">This package contains the G/On Client for the Mac platform.
    </description>
    <update type="enhancement" timing="connect"/>
  </header>

  <filedefs_ro>
   <include source="${gpm_build_root}/gon_client/mac" dest="/gon_client/mac"/>
   <include
    source="${gpm_build_root}/gon_client/mac/gon_client.app/Contents/Resources/shortcut/G-On Mac"
    dest="/G-On Mac"/>
  </filedefs_ro>
</gpmdef>
```

The main element *gpmdef* contains a *header* element which holds the meta information. The *version* element is used for holding the version of the included software or included files, and the *version_release* element is the version for the package specification itself. The *version_release* should be increased if something is changed in the package e.g. an extra file is added, or the description is changed. The combined version number of the *version* element and the *version_release* element constitutes the full version number of the generated package, and is used by the update functionality to detect packages that should be updated.

The *update* element specifies how the system should support updating of the package, if it exists in a newer version on the server, than the version on the client. Currently, the following three choices are supported:

1. No update element specified in the gpmdef file
   The user is not notified about the availability of an update, but he/she may view the list of available updates and install them by selecting the menu item: 'G/Update – Client Package manager, Update'
2. <update type="enhancement" timing="connect"/> specified in the gpmdef file
   At connect time, the user is notified about the availability of an update and may choose to install the updated package. If choosing not to install at this time, the user will be prompted again, at next connect. The update can also be manually installed by the user by selecting the menu item: 'G/Update – Client Package manager, Update'.
3. <update type="security" timing="connect"/> specified in the gpmdef file
   At connect time, the user is forced to install the updated package.

The main element *gpmdef* can contain a *filedefs_ro* element and/or a *filedefs_rw* element. Both elements contains the specification of which files to include, and where to put them when the package is installed. The *filedef_ro* specify the files to be installed to the Read-Only part, and *filedef_rw* specify the files to be installed to the Read-Write part of the destination token. How the Read-Only part and Read-Write part are interpreted depends on the token. Notice that the Read-Only part and Read-Write part could be the same, e.g. on a SoftToken. In order not to introduce errors, this means that the destination of files included by the *filedefs_ro* element and *filedefs_rw* must not overlap. The *filedefs_ro* element and *filedefs_rw* element can contain a number of *include* and/or *ignore* elements. The *include* element can be used to include the content of a folder, but it can also be used to include a single file, see the example above. The *ignore* element excludes files that would otherwise be included.

The main element *gpmdef* can contain a *dependency* element which defines the relation to other packages. The following example specify that the package 'my_package' is to be installed, and that the 'old_package' should be removed.

```
...
<dependency>
  <requires>
    <packageref name="my_package" arch="win"/>
  </requires>
  <obsolutes>
    <packageref name="old_package" arch="win"/>
  </obsolutes>
</dependency>
...
```

Package Collection Specifications (gpmcdef.xml)

This section describes the package collection format **gpmcdef**, in order to enable you to create your own collections. A Package Collection is an XML definition of a collection of packages, which can be used in the “Token Software Management” view in G/On Management.

A number of package collections are included in the standard G/On system. Here is an example:

```
<?xml version="1.0" encoding="UTF-8"?>
<gpmcdef>
  <header name="all_clients">
    <description lang="en" summary="GOn Client for all platforms">Collection of GOn Client packages
all platforms.</description>
  </header>
  <packages>
     <packageref name="gon_client" arch="win"/>
     <packageref name="gon_client" arch="mac"/>
     <packageref name="gon_client" arch="linux"/>
  </packages>
</gpmcdef>
```

The main element *gpmcdef* consists of two child elements; *header* and *packages*.

The *header* element has an attribute *name*, which is the unique identifier for the collection. The child element, *description,* contains the long description, which will appear in the Description section in Token Software Management when a collection is chosen. The content of the attribute *summary* is the one that will be shown in the list of collections in Token Software Management. The attribute *lang* is not used in this version. In the future there may be a number of descriptions in different languages, but in the current version the first *description* child element is always used.

The *packages* element is the one containing references to the packages, which should be included in the collection. It contains a number of *packageref* child elements, each of which must have a *name* and *arch* attribute. The *name* and *arch* attributes refer to the same entries for the package in question. Here is an example of how a package definition looks like:

```
<?xml version="1.0" encoding="UTF-8"?>
<gpmdef>
  <header name="gon_client">
    <version main="5" minor="3" bugfix="0" build_id="8" />
    <version_release main="1"/>
    <arch>win</arch>
    <description lang="en" summary="G/On Client">This package contains the G/On Client for the
Windows platform.</description>
  </header>
  ...
</gpmdef>
```

In this example, notice the header element with a *name* attribute and an *arch* child element. These are the values, which should be used if wanting to include this package in a collection.

Menu Action templates

This section describes the syntax and semantics of Menu Action templates. Menu Action templates are XML files, which define templates for the Menu action Wizard. First we give a description on where templates are found and how to create, change or delete them. After that, a description of the template syntax is given. Finally lists of fields and variables, which can be used in templates, is presented along with a short description of their meaning. There is more information on the meaning of fields in the G/On Management Reference.

Creating new or revised templates

Templates are located in the folder:

```
.\config\templates
```

The folder already contains pre-defined templates. If you want to change these templates you can do so, but we recommend that you create a copy and edit that instead.

The reason for this recommendation is that **customized files cannot be upgraded automatically.** So, every time an upgrade is installed in the future, all manually changed files must be upgraded by hand.

You can create new templates. The easiest way to make a new template is simply to copy one of the existing ones, and change it. Use, e.g. Wordpad for editing, or an editor that is good for editing XML. Note that all templates must have a unique name (see Syntax). If two templates have the same name only one of them will appear in the management client.

To remove a wizard template, simply remove the appropriate XML file from templates folder.

After making changes to the templates folder, restart the management client, to use the new set of wizard templates.

Syntax

In this section we describe the elements that make up a Menu Action template. We will use examples to show the various elements and attributes which can be used. The XML schema for templates is provided in the end of this section.

Basic features

The template consists of a header element *ConfigurationTemplate*, which contains a number of

field elements and possibly a description element. The *ConfigurationTemplate* element has two parameters:

- **name** : The template name. This must be unique.
- **title** : The template title. This is the title which will appear in the administration client

The *ConfigurationTemplate* element can contain a *description* child element. The content of this element will appear next to the template title on the wizard chooser page.

The central part of the template are the fields, which are also children of the con element. There must always be at least one *field* element in the template. Each *field* has a mandatory attribute *field_type*, which can be either *edit*, *hidden* or *custom_template*. Fields of type *edit*, will appear as input fields in the Menu Action wizard in the order specified in the template. Fields with attribute *advanced* set, will appear in the Advanced section of the wizard. Fields of type *hidden* will not be shown. Fields of type *custom_template* are described in the Custom fields section below. Each field must contain a *name* element. The content of this should be the name of a Menu Action field. Please refer to the Field specification section, for a list of the fields available.

The following elements are allowed in a *field*:

- **name -** Unique field name referring to Menu Action field (except for custom fields).
- **default_value** - Initial value for the field.
- **title** - Field title shown in the client. If not specified, a default title is used.
- **tooltip** - Text shown as tool tip in the administration client.
- **description** - A description of the field. Not used at the moment.
- **selection** - Drop-down option – see below.
- **action** - Add action button next to field – see below.

And these are the possible attributes for a *field*:

- **field_type** - Can be *edit*, *hidden* or *custom_template*
- **advanced** - Boolean. True if field should appear in advanced section
- **max_length** - Integer. Maximum number of characters which can be entered.

- **secret** - Boolean. If set the typed value will not be shown (password field).
- **read_only** - Boolean. Should field be read-only
- **mandatory** - Boolean. Should this field always be set.

Example of a simple template:

```
<?xml version="1.0" encoding="utf-8"?>
<ConfigurationTemplate name="example" title="Template Example" xmlns="http://giritech.com/admin_ws">
  <description>Sample template</description>

  <field field_type="edit">
    <name>label</name>
  </field>

  <field field_type="edit">
    <title>Server</title>
    <name>portforward.0.server_host</name>
  </field>

  <field field_type="hidden">
    <name>launch_type</name>
    <default_value>0</default_value>
  </field>

  <field field_type="edit" advanced="true">
    <title>Server Port</title>
    <name>portforward.0.server_port</name>
    <default_value>80</default_value>
  </field>

  <field field_type="edit" advanced="true">
    <name>portforward.0.client_host</name>
    <default_value>127.0.0.2</default_value>
  </field>

</ConfigurationTemplate>
```

The example template has 5 fields: 2 normal edit fields, 2 advanced edit fields and 1 hidden field. The hidden field sets a value for launch_type, which cannot be changed by the user (as you would almost always do).

Custom fields

Fields of type *custom_template* are special editable fields, the values of which are expanded into other fields. In other words you can refer to the value of a custom fields in the default value of other (non-custom) fields, like this:

```
 <field field_type="hidden">
    <name>command</name>
    <default_value>command -width=%(custom_template.width)</default_value>
  </field>

  <field field_type="custom_template">
    <title>Window Width</title>
    <name>width</name>
    <default_value>40</default_value>
  </field>
```

In the example, the default value of the “command” field contain the string

```
%(custom_template.width)
```

which refers to the value of the custom field “width” specified underneath. The value is expanded when the Menu Action is saved, so the resulting value of “command” would be

```
command -width=40
```

assuming that the default value was used in the “width” field.

Entering default field values

Default field values are entered in the *default_value* element of a field. However some field values requires special notation:

- Boolean values: Enter 1 for True, 0 for False
- String values containing special characters : If the value contains special characters (e.g. “<” or “&”) it should be defined as CDATA in order for the XML parser not to parse it. For example, in order to add the XML string

  ```
  <sender>John Smith</sender>
  ```

  as a value you must enter it like this:

  ```
  <default_value>![CDATA[<sender>John Smith</sender>]]</default_value>
  ```

Selection (drop-down)

It is possible to create a drop-down input field in the client by adding the selection element to the field. Here is an example:

```
<field field_type="custom_template">
  <name>redirectsmartcards</name>
  <title>Redirect SmartCards</title>
  <default_value>0</default_value>
  <selection>
    <selection_choice title="No" value="0" />
    <selection_choice title="Yes" value="1"/>
  </selection>
</field>
```

If a field contains a *selection* element it will be shown as a drop-down. The *selection* element must contain at least one *selection_choice*. Each *selection_choice* must have a *value* attribute, which contains a value which can be chosen for the field. If a *title* attribute is specified that is what will be shown in the drop-down (like “Yes” and “No” in the example). If no title is specified the value itself will be shown in the drop-down. If *default_value* is set then that value is shown initially in the drop-down.

Action

An action can be added to a field, enabling the user to perform a specific action on it. In the current version only the port scan action is available:

```
<action>
  <portscan port_field="port"/>
</action>
```

The port scan action performs a scan for servers listening on a given port. The port scan action has a single required attribute *port_field*, which should contain the name of the field, from which the port number value should be taken.

XML schema

```
 <xsd:complexType name="value_selection_choice">
    <xsd:attribute name="value" type="xsd:string" use="required" />
    <xsd:attribute name="title" type="xsd:string" use="optional" />
  </xsd:complexType>

  <xsd:complexType name ="value_selection">
    <xsd:sequence>
      <xsd:element minOccurs="0" maxOccurs="unbounded" name="selection_choice"
type="tns:value_selection_choice"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="portscan_action">
    <xsd:attribute name="title" type="xsd:string" use="optional" default="Scan network"/>
    <xsd:attribute name="port_field" type="xsd:string" use="required"/>
  </xsd:complexType>

  <xsd:complexType name="field_action">
    <xsd:choice maxOccurs="1" minOccurs="1">
      <xsd:element name="portscan" type="tns:portscan_action"/>
    </xsd:choice>
  </xsd:complexType>

  <xsd:complexType name="field_element">
    <xsd:sequence>
      <xsd:element name="name" type="xsd:string" />
      <xsd:element minOccurs="0" name="default_value" type="xsd:string" />
      <xsd:element minOccurs="0" name="title" type="xsd:string" />
      <xsd:element minOccurs="0" name="tooltip" type="xsd:string" />
      <xsd:element minOccurs="0" name="description" type="xsd:string" />
      <xsd:element name="selection" minOccurs="0" type="tns:value_selection"/>
      <xsd:element name="action" minOccurs="0" type="tns:field_action"/>
    </xsd:sequence>
    <xsd:attribute name="field_type" use="required">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="edit" />
          <xsd:enumeration value="custom_template" />
          <xsd:enumeration value="hidden" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="value_type" default="string_value_type">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="string_value_type" />
          <xsd:enumeration value="text_value_type" />
          <xsd:enumeration value="integer_value_type" />
          <xsd:enumeration value="float_value_type" />
          <xsd:enumeration value="boolean_value_type" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="advanced" use="optional" type="xsd:boolean" default="false"/>
    <xsd:attribute name="max_length" use="optional" type="xsd:integer"/>
    <xsd:attribute name="secret" use="optional" type="xsd:boolean" default="false"/>
    <xsd:attribute name="read_only" use="optional" type="xsd:boolean" default="false"/>
    <xsd:attribute name="mandatory" use="optional" type="xsd:boolean" default="false"/>
  </xsd:complexType>

  <xsd:complexType name="ConfigurationTemplate">
    <xsd:sequence>
      <xsd:element maxOccurs="unbounded" minOccurs="1" name="field" type="tns:field_element">
      </xsd:element>
      <xsd:element name="description" type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="name" type="xsd:string" use="required" />
    <xsd:attribute name="title" type="xsd:string" use="required" />
```

Field specification

In this subsection, we list all fields available in Menu Actions and a short description of their meaning. Each field has a Type, which define its value type and also defines how the input field is shown in the administration client. The types are:

String(X)	A single line string field of maximum length X
String	A single line string field of unlimited length
Text	A multi line string field of unlimited length
Integer	Single line text field for integer input
Boolean	A check box

Launch type 0: Port Forward

The traditional way to enable applications for G/On is to use port forwards listening on a localhost loopback network interface. The client side application must be configured to connect to the port forward instead of directly to the server, and the G/On client will forward the connection securely to the G/On server which will forward the connection to the real server if the user is properly authorized.

Field	Type	Description
label	String(40)	The name of the Menu Action
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	0 for Port Forward
command	Text	The command to launch after the port forwards has been established
close_command	Text	The command to launch when the menu action ends.
working_directory	Text	If specified then this directory will be used for launching the command

Field	Type	Description
close_with_process	Boolean	The port forwards will be closed when the launched command returns
kill_on_close	Boolean	The launched process will be killed when the port forward is closed
param_file_name	String	An alternative name to use for the parameter file
param_file_lifetime	Integer	The parameter file will be removed this many seconds after the menu action has been launched. For more details, see the complete description on page 142.
param_file_template	Text	Create a parameter file containing this template with variables expanded
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action
dialog_tag_generators	Text	A newline-separated list of tag generators
portforward.0.server_host	String	The server side IP address or name to connect to
portforward.0.client_host	String	The client side IP address or name to listen on
portforward.0.server_port	Integer	The server side port number to connect to
portforward.0.client_port	Integer	The client side port number to listen on. If 0 (zero) is specified, the G/On client automatically chooses an unused port. The port that has been chosen can be observed in the variable %(portforward.port) – see the next section on variables.
portforward.0.lock_to_process_pid	Boolean	Only the launched command may use the port forward

Field	Type	Description
portforward.0.sub_processes	Boolean	Also allow subprocesses of the launched command to use the port forward (requires lock_to_process_pid)
portforward.0.lock_to_process_name	String	Only processes with this name are allowed to use the port forward (conflicts with lock_to_process_pid)

Note that additional port forwards can be added by changing the '0' in the port forward fields to '1', '2', '3', etc. The numbers must be consecutive, i.e. specifying port forwards with numbers 0 and 2 but not 1 is not allowed.

Note that many Mac OS/X applications launches and uses other processes, so features like *close_with_process*, *kill_on_close*, *lock_to_process_pid* and *sub_processes* often doesn't work as expected.

Launch type 1: Citrix Web Interface

G/On has special support for connecting to a Citrix farm through the Citrix Web Interface.

Field	Type	Description
label	String(40)	The name of the Menu Action
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	1 for Citrix Web Interface
command	Text	The command to launch a browser for the Web Interface as a portforward on %(portforward.host):%(portforward.port) – or the end of a launch URL starting with “?”
citrix_command	Text	The command to launch wfica – it should reference the ICA file in %(launch.param_file)

Field	Type	Description
sso_login	String	The login to use on the Web Interface – probably %(user.login). If this field is empty, no single sign-on will be made. This may render the menu action useless.
sso_password	String	The password to use on the Web Interface – probably %(user.password)
sso_domain	String	The domain to use on the Web Interface
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action
dialog_tag_generators	Text	A newline-separated list of tag generators
portforward.0.server_host	String	Hostname or IP address of Web Interface server
portforward.0.server_port	Integer	Port number on Web Interface server
portforward.0.lock_to_process_pid	Boolean	Only the launched wfica command may connect to the Citrix server
portforward.0.sub_processes	Boolean	Also allow subprocesses of the launched wfica command to connect to the Citrix server (requires lock_to_process_pid)
portforward.0.lock_to_process_name	String	Only processes with this name are allowed to connect to the Citrix server (conflicts with lock_to_process_pid)
close_with_process	Boolean	The ICA connection will be closed when the launched wfica command returns

Note that Citrix client programs sometimes cooperate, so that two Citrix windows use the same wfica process and the same connection.

Launch type 2: G/On Internal

This launch type is reserved for internal G/On commands.

Launch type 3: Wake on LAN

G/On can send Wake On LAN packages to sleeping machines on the server network.

Field	Type	Description
label	String(40)	The name of the Menu Action
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	3 for Wake on Lan
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action
dialog_tag_generators	Text	A newline-separated list of tag generators
portforward.0.server_host	String	UDP Target IP
portforward.0.server_port	Integer	UDP Target Port
command	Text	MAC address in the format 01:23:45:67:89:ab

Launch type 4: Citrix XML Interface

G/On has special support for connecting to a Citrix farm through the Citrix XML Interface.

Field	Type	Description
label	String(40)	The name of the Menu Action

Field	Type	Description
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	4 for Citrix XML Interface
command	Text	Name of a XenApp Application (optional). If not specified, individual menu items will be created for all XenApp applications available for the user through the Citrix XML interface.
param_file_name	String	An alternative name to use for the ica parameter file
param_file_lifetime	Integer	The parameter file will be removed this many seconds after the menu action has been launched. For more details, see the complete description on page 142.
param_file_template	Text	Create an ica parameter file containing this template with variables expanded
sso_login	String	The login to use on the XML Interface – probably %(user.login)
sso_password	String	The password to use on the XML Interface – probably %(user.password). If this field is empty, no single sign-on will be made. This most probably will render the menu action useless.
sso_domain	String	The domain to use on the XML Interface
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action
dialog_tag_generators	Text	A newline-separated list of tag generators

Field	Type	Description
portforward.0.server_host	String	A server list for load sharing and/or fail-over, as described in section “Fields for defining client-server connectivity with RDP and Citrix connections“ on page 137.
portforward.0.server_port	Integer	Port number on XML Interface server
portforward.0.client_host	String	The client side IP address or name to listen on
portforward.0.lock_to_process_pid	Boolean	Only the launched wfica command may connect to the Citrix server
portforward.0.sub_processes	Boolean	Also allow subprocesses of the launched wfica command to connect to the Citrix server (requires lock_to_process_pid)
portforward.0.lock_to_process_name	String	Only processes with this name are allowed to connect to the Citrix server (conflicts with lock_to_process_pid)
close_with_process	Boolean	The ICA connection will be closed when the launched wfica command returns

Note that Citrix client programs sometimes cooperate, so that two Citrix windows use the same wfica process and the same connection.

Launch types 5-7: Unused

These launch types are reserved for future use.

Launch type 8: HTTP connection, and HTTP and SOCKS proxy

G/On has special support both for transparent proxying of plain HTTP and for proxying using the explicit HTTP proxy protocol. G/On also has support for the SOCKS proxy protocol. All of these are handled by launch type 8:

Field	Type	Description
label	String(40)	The name of the Menu Action
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	8 for connecting through a plain HTTP connection (transparent HTTP proxy) or through an HTTP proxy or through a SOCKS proxy.
command	Text	The command to launch client side program, after the port forward to the server side proxy has been established
close_command	Text	The command to launch when the menu action ends.
working_directory	Text	If specified then this directory will be used for launching the command
close_with_process	Boolean	The port forwards will be closed when the launched command returns
kill_on_close	Boolean	The launched process will be killed when the port forward is closed
param_file_name	String	An alternative name to use for the parameter file
param_file_lifetime	Integer	The parameter file will be removed this many seconds after the menu action has been launched. For more details, see the complete description on page 142.
param_file_template	Text	Create a parameter file containing this template with variables expanded
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action

Field	Type	Description
dialog_tag_generators	Text	A newline-separated list of tag generators
portforward.0.server_host	String	A server list for load sharing and/or fail-over, as described on page 138. If this is specified, the menu action will allow plain HTTP traffic to be forwarded to a single one of the specified servers. If the field is empty, the menu action will allow HTTP proxy traffic and SOCKS proxy traffic limited by a white list that is specified in the fields *portforward.n.server_host*, and portforward.*n*.server_port, where *n* is 1, 2, etc The details of white lists are described in the section “Fields for defining client-server connectivity with HTTP and SOCKS proxy connections” on page 138 and onwards.
portforward.0.client_host	String	The client side IP address or name to listen on.
portforward.0.server_port	Integer	The server side port number to connect to. May be empty, if the menu action is to allow HTTP proxy traffic and SOCKS proxy traffic – see the decription of field portforward.0.server_host, above.
portforward.0.client_port	Integer	The client side port number to listen on. If 0 (zero) is specified, the G/On client automatically chooses an unused port. The port that has been chosen can be observed in the variable %(portforward.port) – see the next section on variables.
portforward.0.lock_to_process_pid	Boolean	Only the launched command may use the port forward
portforward.0.sub_processes	Boolean	Also allow subprocesses of the launched command to use the port forward (requires lock_to_process_pid)
portforward.0.lock_to_process_name	String	Only processes with this name are allowed to use the port forward (conflicts with lock_to_process_pid)

Field	Type	Description
portforward.*n*.server_host portforward.*n*.server_port	String	For *n* = 1, 2, … The server and port specifications in these fields constitute a white list, as described on page 139 and onwards. They only have effect if the field portforward.0.server_host is empty
sso_login	String	The login to use for automatic HTTP authentication. See complete details in the section “Fields for defining server side single sign-on credentials“ on page 141.
sso_password	String	The password to use for automatic HTTP authentication
sso_domain	String	The domain to use for automatic HTTP authentication

Launch type 10: RDP Connection

G/On has special support for connecting to a Remote Desktop server.

Field	Type	Description
label	String(40)	The name of the Menu Action
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	10 for connecting through an RDP connection.
command	Text	The command to launch the Remote Desktop client, after the port forward has been established
close_command	Text	The command to launch when the menu action ends.
working_directory	Text	If specified then this directory will be used for launching the command

Field	Type	Description
close_with_process	Boolean	The port forwards will be closed when the launched command returns
kill_on_close	Boolean	The launched process will be killed when the port forward is closed
param_file_name	String	An alternative name to use for the parameter file
param_file_lifetime	Integer	The parameter file will be removed this many seconds after the menu action has been launched. For more details, see the complete description on page 142.
param_file_template	Text	Create an rdp parameter file containing this template with variables expanded
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action
dialog_tag_generators	Text	A newline-separated list of tag generators
portforward.0.server_host	String	A server list for load sharing and/or fail-over, as described in section “Fields for defining client-server connectivity with RDP and Citrix connections“ on page 137.
portforward.0.client_host	String	The client side IP address or name to listen on. OBS: for Terminal server 2008 farm redirection to work, this *must* be 127.0.0.1.
portforward.0.server_port	Integer	The server side port number to connect to
portforward.0.client_port	Integer	The client side port number to listen on. If 0 (zero) is specified, the G/On client automatically chooses an unused port. The port that has been chosen can be observed in the variable %(portforward.port) – see the next section on variables.

Field	Type	Description
portforward.0.lock_to_process_pid	Boolean	Only the launched command may use the port forward
portforward.0.sub_processes	Boolean	Also allow subprocesses of the launched command to use the port forward (requires lock_to_process_pid)
portforward.0.lock_to_process_name	String	Only processes with this name are allowed to use the port forward (conflicts with lock_to_process_pid)
sso_login	String	The login to use on the XML Interface – probably %(user.login) If this field is empty, no single sign-on will be made.
sso_password	String	The password to use on the XML Interface – probably %(user.password)
sso_domain	String	The domain to use on the XML Interface

Launch type 11: DME

G/On has special support for connecting the AppBox browser in an Excitor DME client to web servers, in accordance with a specified white list. This is handled by launch type 11:

Field	Type	Description
label	String(40)	The name of the Menu Action
menu_title	Text	The menu item name, that is used when the action appears in the menu of an end-user
launch_type	Integer	11 for connecting DME AppBox through an HTTP proxy
command	Text	The DME AppBox browser command to launch, after the

Field	Type	Description
		port forward to the server side proxy has been established
dialog_tags	String	A comma-separated list of the tags which should be attached to the Menu Action
dialog_tag_generators	Text	A newline-separated list of tag generators
portforward.*n*.server_host portforward.*n*.server_port	String	For *n* = 1, 2, … The server and port specifications in these fields constitute a white list, as described on page 139 and onwards.
sso_login	String	The login to use for automatic HTTP authentication. See complete details in the section “Fields for defining server side single sign-on credentials“ on page 141.
sso_password	String	The password to use for automatic HTTP authentication
sso_domain	String	The domain to use for automatic HTTP authentication

Variables

A number of pre-defined values, such as loop back host address or default browser, can be added to Menu Actions fields. These variables are expanded when the Menu Action is launched. Here is a list of the variables:

Field	Description
%(portforward.host)	The loop back host address
%(portforward.port)	The loop back port address
%(launch.param_file)	The full file name of a parameter file, if there is one

Field	Description
%(launch.cwd)	The current working directory, when the command is launched
%(launch.browser)	Full file name for the default browser
%(launch.ie)	Full file name for the IE browser
%(reg.<regkey>)	Value of a registry key Example of <regkey>: HKEY_CLASSES_ROOT\Applications\iexplore.exe\shell\open\command
%(env.<env.variable>)	Value of an env variable
%(cpm. ro_root)	Location where files intended for the R/O partition are placed
%(cpm. rw_root)	Location where files intended for the R/W partition are placed
%(run_env.temp)	Location of G/On temp folder
%(user.login)	User name
%(user.password)	Password in clear text
%(user.password_base64)	Password in base 64 encoding
%(user.domain)	Authentication domain
%(user.netbios)	NetBIOS name for AD Domain
%(user.my_pc_<x>)	DNS or IP address of user's PC no. x (x=1,2,3,..). Example: %(user.my_pc_1)
%(user.mac_address_<x>)	MAC address of user's PC no. x (x=1,2,3,..). Example: %

Field	Description
	(user.mac_address_1)
%(user.<property>)	The given property of the user entry, as defined in the user directory where the user is registered

Index