Update GestióIP LDAP configuration after an update from a GestióIP version <= 3.5.3

GestióIP version 3.5.4 introduces the possibility to manage local and LDAP Users via the GUI. The new feature is only for new installations available, because it requires some changes in the Apache configuration. The actualization mechanism of GestióIP does not update the Apache configuration automatically, because of the danger of breaking it in the case that it was manually edited.

The concept of GestióIP users has changed in version 3.5.4. In older versions, GestióIP users were only used for the authorization process. Up from version 3.5.4, GestióIP users are used for authentication (and with “User Management” enabled, also for authorization like in older versions). You find more information about the “User Management” in the User Guide (https://www.gestioip.net/docu/Documentation_GestioIP_35_en.pdf).

To enable the new feature it is necessary to execute a couple of manual steps. Read first the *entire* manual before you start working.

Note: There is also an alternative way to the here described method available. It requires the installation on a new server, but is more easy.

1. Update the actual installation to the latest version of GestióIP
2. Install the latest version of GestióIP on a new server
3. Export the database from the old server:
   $ mysqldump -u root -p gestioip > gestioip_bck.sql
4. Import the database on the new server
   $ mysql -u root -p gestioip < gestioip_bck.sql

1) Download the tar-ball gip_apache_config_354.tar.gz

$ wget https://www.gestioip.net/files/gip_apache_config_354.tar.gz

2) untar it

$ tar zxf gip_apache_config_354.tar.gz

3) change to the new directory gip_new_apache_config

$ cd gip_apache_config_354

4) Execute the script create_config_354.sh

$ sudo ./create_config_354.sh

This will prepare the file apache/gestioip.conf_354 and create the following new files in /usr/share/gestioip/etc/apache:

apache-groups - Local users which are members “GestioIPGroup” will have access
apache_ldap.conf - Apache LDAP configuration
apache_ldap_require.conf - Require directives

5) Make a backup of you actual GestióIP Apache configuration file and copy the new configuration over the old one:

Debian/Ubuntu:
$ sudo cp /etc/apache2/sites-available/gestioip.conf /etc/apache2/sites-available/gestioip.conf.old
$ sudo cp apache/gestioip.conf_354 /etc/apache2/sites-available/gestioip.conf

RH/CentOS:
$ sudo cp /etc/httpd/conf.d/gestioip.conf /etc/httpd/conf.d/gestioip.conf.old
$ sudo cp apache/gestioip.conf_354 /etc/httpd/conf.d/gestioip.conf

Suse:
$ sudo cp /etc/httpd/conf.d/gestioip.conf /etc/httpd/conf.d/gestioip.conf.old
$ sudo cp apache/gestioip.conf_354 /etc/httpd/conf.d/gestioip.conf

6) Add local users to the new Apache groups file

Local users are defined in the file “users-gestioip”.

Ubuntu/Debian: /etc/apache2/users-gestioip
RH/CentOS: /etc/httpd/conf.d/users-gestioip
Suse: /etc/httpd/conf.d/users-gestioip

Display the content of the file:
$ cat users-gestioip

gipadmin:$asdBSMzkdafoadsknkadklaalXBBrQQQ1
user1:$aprsadas1$34dshL$b/m10E715Fl/ExO7nw/
user1:$adss2FE4aWpasdadaFWGodab2qjcpiRoj41

and add the users which appear in the file to /usr/share/gestioip/etc/apache/apache-groups.

In this example the content of the file apache-groups consist is the line:

GestioIPGroup: *gipadmin user1 user2*

NOTE: if you are using a different default user account as “gipadmin” use this user instead.

7) LDAP users and groups

The LDAP configuration is passed from the GestióIP Apache configuration file gestioip.conf to the new file /usr/share/gestioip/etc/apache/apache_ldap.conf.

If you already have LDAP authentication configured, update the file apache_ldap.conf with your LDAP server configuration. Edit the existing lines, do not add lines.

For example:

```
AuthFormProvider file ldap
AuthLDAPBindDN "ad_user@example.edu"
AuthLDAPBindPassword XXXXXX
AuthLDAPUrl "ldap://ad.example.com:389/dc=example,dc=edu?sAMAccountName?sub?(objectClass=*)" NONE

AuthLDAPGroupAttribute member uniqueMember memberUid

# Default values
#    AuthLDAPGroupAttributeIsDN off
#    AuthLDAPMaxSubGroupDepth 10
#    AuthLDAPSubGroupAttribute memberUid
#    AuthLDAPSubGroupClass posixgroup
```

Add the LDAP related “Require” directives, which you configured in the old configuration to the file /usr/share/gestioip/etc/apache/apache_ldap_require.conf

For example:
Require ldap-user ldap_user1
Require ldap-group cn=gestioip,dc=example,dc=edu

8) Access to a CLI of the MySQL database

$ mysql -u gestioip -p
mysql> use gestioip;

9) Add the LDAP server to GestióIP

Use the values you configured in /usr/share/gestioip/etc/apache/apache_ldap.conf. “type” must be either “MS Active Directory” or “LDAP”.

LDAP server (Microsoft AD):

mysql> INSERT INTO ldap_server (name, server, type, protocol, port, bind_dn, bind_password, base_dn, user_attribute, user_filter, comment, enabled, client_id) VALUES ("LDAP_AD", "ad.example.com", "MS Active Directory", "LDAP", "389", "ad_user@example.edu", "XXXXXX", "dc=example,dc=edu", "sAMAccountName", "objectClass=*", "optional comment", 1, 1);

LDAP server (none Microsoft AD):

mysql> INSERT INTO ldap_server (name, server, type, protocol, port, bind_dn, bind_password, base_dn, user_attribute, user_filter, comment, enabled, client_id) VALUES ("LDAP", "ldap.example.com", "LDAP", "LDAP", "389", "ldap_ro_user", "XXXXXX", "dc=example,dc=edu", "uid", "", "optional comment", 1, 1);

10) Add the local users to the database

Add the local users, which you added before to the user-group-file /usr/share/gestioip/etc/apache/apache-groups, to the table “gip_users”.

Display the content of the table gip_users;

```
mysql> select * from gip_users;
+----+---------------+----------+-----------------+----------+--------------------------+-------+
| id | name          | group_id | email           | phone    | comment                  | type  |
+----+---------------+----------+-----------------+----------+--------------------------+-------+
|  1 | gipadmin      |        1 |                 |          | user automatically created | NULL  |
+----+---------------+----------+-----------------+----------+--------------------------+-------+
1 rows in set (0.00 sec)
```

The user with the id = 1 is the user which was created during the installation. The default username is “gipadmin”. If you choose a different username during the installation you can also use this user in the following steps.

If there does not exist a user with the id = 1, create it:

mysql> INSERT INTO gip_users (name, group_id, phone, email, comment, type) VALUES ("gipadmin",1,"","","","", "local");

if it exist, set the “type” of the user to “local”:

mysql> UPDATE gip_users SET type="local" where name="gipadmin";

Insert the local users you found in users-gestioip (see 6) to the database (you can change the group, comment, … later via GUI)

mysql> INSERT INTO gip_users (name, group_id, phone, email, comment, type) VALUES ("user1",1,"","","","", "local");
INSERT INTO gip_users (name, group_id, phone, email, comment, type) VALUES ("user2",1,"","","","", "local");

11) Add the LDAP users to the database:
If you have any “Require ldap-user” directives in your old Apache configuration defined, add this users to the table gip_users:

mysql> INSERT INTO gip_users (name, group_id, phone, email, comment, type) VALUES ("ldap_user1",1,"","","", "LDAP");

12) Add the LDAP groups to the database
If you have any “Require ldap-group” directive defined, add the groups to the table ldap_group:

mysql> INSERT INTO ldap_group (name, dn, user_group_id, ldap_server_id, group_attrib_is_dn, comment, enabled, client_id) VALUES ("group_1”, "cn=gestioip,dc=example,dc=edu", 1, **1**, 1, "optional comment", 1, 1);

- The “user_group_id” is only relevant, if the “user Management” feature is enabled. You can set in now to “1” and change the “User Group” of the LDAP Group later from the web Gui.

- The “ldap_server_id” is the ID of the newly created LDAP server. Check the ID with the statement

```
mysql> select  * from ldap_server;
+----+---------------+-----------+------+----------+------+-----------------------------------+--------------+
--------------------+---------------+-------------+---------+---------+-----------+
| id | name          | server    | type | protocol | port |
bind_dn     | bind_password | base_dn             | user_attribute |
user_filter | comment | enabled | client_id |
+----+---------------+-----------+------+----------+------+-----------------------------------+--------------+
--------------------+---------------+-------------+---------+---------+-----------+
|  1 | LDAP_AD | ad.example.com | MS Active Directory | LDAP     |  389 |
ad_user@example.edu | XXXXXX       | dc=example,dc=edu | sAMAccountName   |
objectClass=* | optinal comment |       1 |         1 |
+----+---------------+-----------+------+----------+------+-----------------------------------+--------------+
--------------------+---------------+-------------+---------+---------+-----------+
1 row in set (0.00 sec)
```

- If you do not have clients defined, “client_id” is 1.

13) Restart the Apache web server

Ubuntu/Debian:
sudo service apache2 restart

RH/CentOS:
sudo systemctl restart httpd

Suse
sudo service apache2 restart

If the Apache web server does not start correctly, check the Apache error log for related messages.

14) Login with a local or a LDAP account

If it is not possible to log in, check the Apache error log for related messages. If there are no messages, raise the LogLevel of the Apache web server. Search the line starting with “LogLevel” in the Apache main configuration file, change it to “LogLevel debug” and restart the Apache web server.