
Flyve MDM Documentation

Flyve MDM

Dec 13, 2018

Contents:

| | | |
|----------|-------------------------------|----------|
| 1 | Table of Contents | 3 |
| 1.1 | Installation | 3 |
| 1.2 | Configuration | 5 |
| 1.3 | Update | 10 |
| 1.4 | Getting Started | 12 |
| 1.5 | How it works | 21 |
| 1.6 | MQTT Specifications | 22 |
| 1.7 | Use Cases | 29 |
| 1.8 | Rest API | 37 |
| 1.9 | Docker Environment | 42 |
| 1.10 | UML Diagrams | 42 |
| 1.11 | Join our community! | 43 |
| 1.12 | Translations | 49 |
| 1.13 | Amazon Web Services | 53 |
| 1.14 | Cloud Services | 60 |
| 1.15 | Demo Trial | 64 |



1.1 Installation

1.1.1 Server Recommendations

- A server running Linux, Apache, MySQL/MariaDB and PHP (a LAMP server) for the backend (GLPI and Flyve MDM for GLPI).
- An Ubuntu or Debian server running Mosquitto.
- A server running the web interface. It may run on the same server as GLPI.

1.1.2 Compatibility Matrix

| | | | |
|-------------------|---------|---------|---------|
| GLPI | 9.1.x | 9.2.x | 9.3.x |
| Flyve MDM | 1.x.x | 2.0.0 | 2.0.0 |
| FusionInventory | 9.1+1.0 | 9.2+1.0 | 9.3+1.0 |
| Web MDM Dashboard | - | 2.0.0 | 2.0.0 |

1.1.3 Downloads

GLPI installation

You must create your own instance of GLPI, in order to use the Flyve MDM plugin and the [Web MDM Dashboard](#).

Note: You can follow the proper [GLPI installation documentation](#).

- Go to `/var/www/`

- Download GLPI, you can get it with any of these methods:
 - From the [Download section](#) on their website.
 - From [GitHub releases](#). Be sure to select the `.tgz` file. As specified in the GLPI Installation documentation, the *Source code* files should not be used.
 - Using Git, for those who are familiar with it:

```
git clone https://github.com/glpi-project/glpi.git
```

Fusion Inventory

Flyve MDM depends on inventory features of FusionInventory for GLPI.

- Go to `glpi/plugins/`
- Download **FusionInventory for GLPI**, you can get it with any of these methods:
 - From the [Get it](#) section on their website.
 - From [GitHub Releases](#).

Keep in mind the type of file you download, to unpack a tarball you need the following command: `tar -xvzf file-name.tar.bz2`

- Using Git, for those who are familiar with it:

```
git clone https://github.com/fusioninventory/fusioninventory-for-glpi.git
git fusioninventory
```

Note: For more information, check the [Fusion Inventory Documentation](#).

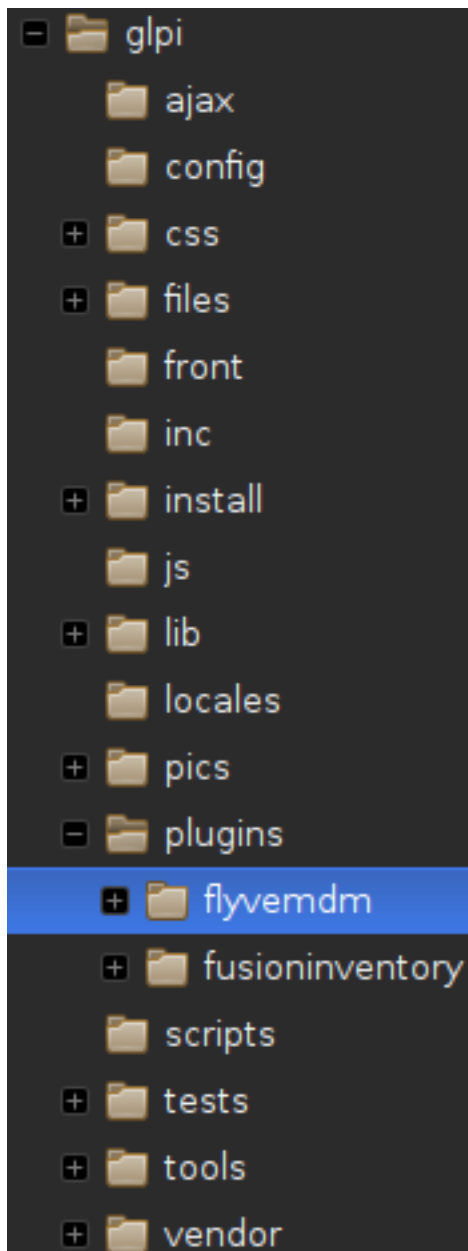
Flyve MDM

- Go to `glpi/plugins`
- Download Flyve MDM for GLPI, you can get it with any of these methods:
 - [Download zip file](#) from GitHub (The zip is from the develop branch, which is the bleeding edge version)
 - From [GitHub Releases](#).
 - Using Git, for those familiar with it:

```
git clone https://github.com/flyve-mdm/glpi-plugin.git flyvemdm
```

- Go in the directory `glpi/plugins/flyvemdm`
- Run `composer install --no-dev`

You should have a final structure like this:



1.2 Configuration

Flyve MDM counts with a Wizard to make sure everything is set up as it should, go to Flyve MDM configuration and select the tab Installation Wizard.

We believe the Administrator of the IT infrastructure must know every configuration that GLPI requires to work with Flyve MDM, therefore in the Wizard is an step by step guide to configure GLPI.

In case a step is missed or misconfigured in GLPI, a message will be shown in the plugin, warning you in order to avoid future problems.

Next, we'll be explaining every configuration tab of the plugin.

1.2.1 General

General configuration

Document types

Allow upload of APK and UPK files add document types

Device type

Computer type for the devices ----- i +

Agent account type

Account type for the devices ----- i +

Invitation deeplink

Invitation deeplink URL http://flyve.org/deeplink/ Base URL of the deeplink micro service

Restrictions

Default device limit per entity 0 No more devices than this quantity are allowed per entity by default (0 = no limitation)

Save

- Document types: add the types .apk and .upk required to upload uhuru and android apps.
- Device type: type of the computer for the devices, it can be added a new type with the “+” button.
- Agent account type: category of the user, it can be added a new type with the “+” button.
- Invitation deeplink: base url for the deeplink micro service.
- Restrictions: limit the number of devices allowed per entity.

1.2.2 Message Queue

| MQTT broker | | |
|--|---|---|
| Enabled | <input type="button" value="No"/> | |
| MQTT broker address for clients | <input type="text"/> | An IP address or a hostname used by the client devices to communicate with the broker |
| MQTT broker address for backend | <input type="text" value="127.0.0.1"/> | An IP address or a hostname used by the backend to communicate with the broker |
| MQTT broker port | <input type="text" value="1883"/> | A port number between 1025 and 65535, standard port is 1883 |
| MQTT broker port for TLS | <input type="text" value="8883"/> | A port number between 1025 and 65535, standard port is 8883 |
| Use TLS for clients | <input type="button" value="No"/> | Use TLS for MQTT communication between the broker and the client devices |
| Use TLS for backend | <input type="button" value="No"/> | Use TLS for MQTT communication between the broker and the backend |
| CA certificate | <div> File(s) (32 Mio max) i Drag and drop your file here, or <input type="button" value="Seleccionar archivo"/> <input type="button" value="No se eli... archivo"/> </div> | |
| Cipher suite (TLS enabled) | <input type="text" value="ECDHE-RSA-AES128-"/> | Mqtt broker cipher suite |
| test | <input type="button" value="test"/> | |
| Client certificate server (Broker MQTT with TLS enabled) | | |
| Use client certificates | <input type="button" value="No"/> | |
| Ssl certificate server for MQTT clients | <input type="text"/> | https://cert.domain.com/path/to/service |
| Firebase Cloud Messaging | | |
| Enabled | <input type="button" value="No"/> | This feature is in experimental phase |
| API token | <input type="text"/> | The server key token given on Firebase settings |
| <input type="button" value="Save"/> | | |

- MQTT Broker:
 - MQTT broker address for clients: an IP address or a hostname used by the clients to communicate with the broker.
 - MQTT broker internal address for backend: an IP address or a hostname used by the backend to communicate with the broker.
 - MQTT broker port: a port number between 1025 and 65535, standard port is 1883.
 - MQTT broker port for TLS: a port number between 1025 and 65535, standard port is 8883.
 - Use TLS for clients: enables TLS communication between broker and clients.
 - Use TLS for backend: enables TLS communication between broker and backend.
 - CA certificate: is the certificate of an authority to verify the MQTT server.
 - Cipher suite is used to limit the ciphers used with TLS.
- Client certificate server (Broker MQTT with TLS Enabled)

- Use client certificate: enables use of the client's certificate.
- SSL certificate server for MQTT clients: url to the certificate server for MQTT clients.
- Firebase Cloud Messaging
 - Enable this feature to use FCM, for devices with Android 8 or later.
 - API Token: the server key, you can find it in your Firebase project settings > Cloud Messaging

Important: After sending the certification request, the CA will very likely send back several files. One is the certificate, signed by the CA, and the others are intermediate certificates.

In Mosquitto **the certificate must be the concatenation of the certificate delivered + the intermediate certificates.**

The operating system must contain more certificates to establish a trust chain to the root certificate.

Important: The MQTT protocol and FCM can be enabled at the same time, to use FCM be sure to read [How to install the MDM Agent for Android 8 or later](#).

1.2.3 Debug

Debug

Enrollment

Enable explicit enrolment failures

No ▾

Disable token expiration on successful enrolment

Yes ▾

Bug collector

Android bug collector URL

https://bugreport.flyvemdm.com/path/to/service

Android bug collector user

Android bug collector password

Configuration wizard

Enable configuration wizard

Yes ▾

Warning: if the plugin is already configured, re-running the configuration wizard may break communication with already enrolled devices.

Save

- Enrollment:
 - Enable explicit enrollment failures: if enabled sends to devices the exact reason of an enrollment failure. For debug purposes only.
 - Disable token expiration on successful enrollment: the invitation token won't expire after successful enrollment.
- Bug collector:
 - Android bug collector URL: is the URL of a ACRA server. This server collects crash reports sent by Flyve MDM for Android.
 - Android bug collector user: username used by devices when they send a crash report.

- Android bug collector password: password used by devices when they send a crash report.
 - Configuration wizard: enables the wizard display.
- Also, learn how to customize the default template for the user invitations.

1.2.4 Customized Template

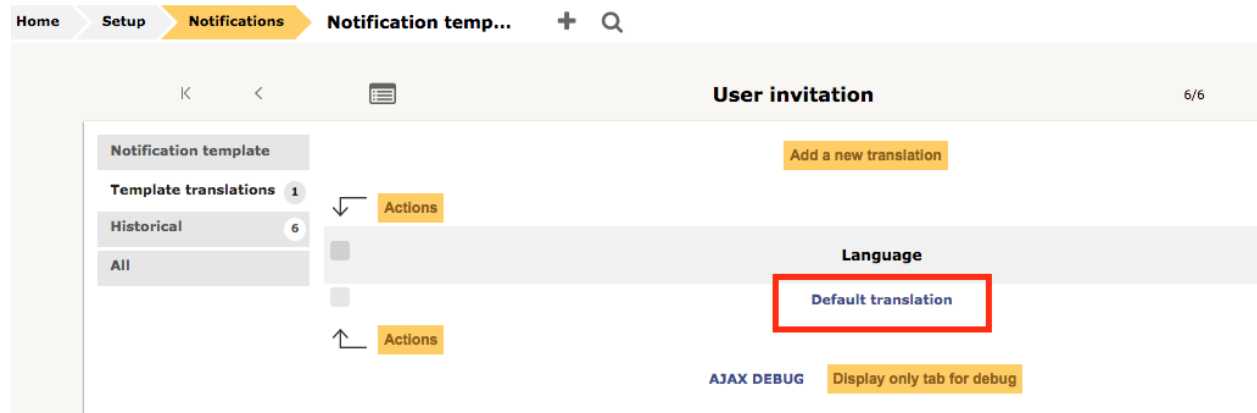
If you want to modify the default template for the invitations, you can change it directly from your GLPI instance.

1. Go to Setup > Notifications > Notifications Templates

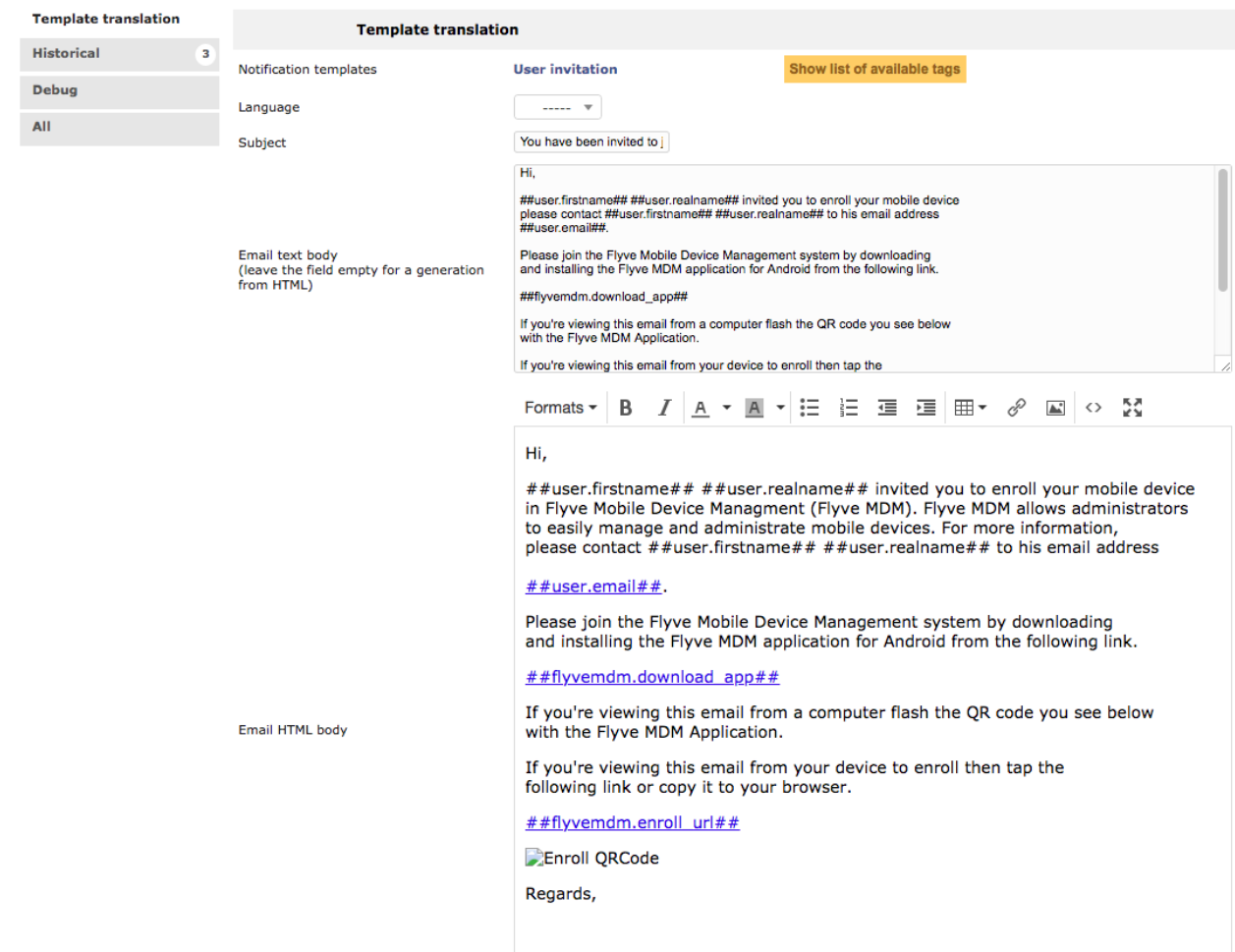
2. Find the User Invitation template

| Name | Type | Comments |
|---------------------|--------------|----------|
| Ticket Satisfaction | Tickets | |
| Tickets | Tickets | |
| Tickets (Simple) | Tickets | |
| Tickets Validation | Tickets | |
| Unlock Item request | Object Locks | |
| User invitation | Invitations | Hello |

3. Select Template translation > Default translation



4. Modify the email message according to your requirements



1.3 Update

Remember to:

- Check the compatibility matrix of your GLPI version with Flyve MDM Plugin before any update.

- Make a backup of your database.
- Make a backup of the files of GLPI.

1.3.1 Compatibility Matrix

| GLPI | 9.1.x | 9.2.x | 9.3.x |
|-------------------|---------|---------|---------|
| Flyve MDM | 1.x.x | 2.0.0 | 2.0.0 |
| FusionInventory | 9.1+1.0 | 9.2+1.0 | 9.3+1.0 |
| Web MDM Dashboard | - | 2.0.0 | 2.0.0 |

1.3.2 From one release to another

- Download the new version of the plugin.
- Go to `glpi/plugins` directory.
- Move the folder of Flyve MDM plugin out of the GLPI subtree (old version).
- Install the new version of the plugin (this one should be named `flyvemdm`).
- Change the current directory to the content of Flyve MDM plugin `cd glpi/plugins/flyvemdm`.
- Run `php tools/cli_install.php` in order to update your database.
- Enable the plugin again with the user interface of GLPI.
- Check in the file `glpi/plugins/flyvemdm/scripts/service.sh` that the user of the daemon is the same as your HTTP server (`www-data` for debian based systems).
- If you need to change the user, please run `update-rc.d flyvemdm defaults`.
- Restart the daemon `service flyvemdm restart`.

After you successfully confirm everything is as it should, delete the old version you moved out of GLPI.

1.3.3 Using Git

If you installed the Flyve MDM plugin through git:

- Go to the Flyve MDM plugin's directory.
- Use `git pull` to update your local branch.
- Run `php tools/cli_install.php` to update the database.
- Enable the plugin again with the user interface of GLPI.
- Check in the file `glpi/plugins/flyvemdm/scripts/service.sh` that the user of the daemon is the same as your HTTP server (`www-data` for debian based systems).
- If you need to change the user, please run `update-rc.d flyvemdm defaults`
- Restart the daemon `service flyvemdm restart`.

You can use this method to upgrade the plugin from a revision to another revision on the develop branch.

1.4 Getting Started

The Flyve MDM plugin for GLPI integrates the intuitive and outstanding features of Flyve MDM into GLPI platform providing you the security functionality for your IT infrastructure.

As GLPI is a Free Asset and IT Management Software package, we wanted to provide a plugin that helps you to keep control of your mobile devices, here we'll guide you in the steps to manage your mobile fleet from the basics.

1.4.1 Enrolling Agents

In order to add the devices to your fleet, you must invite the owner of the device to enroll by sending him an email.

- Go to the Invitations section
- Click on the “+” button
- Write the email of the user whose device you'll control
- Click on Add

The invitation will have the status pending until the user enrolls the device.

Note: Users must have installed in their devices the Android or iOS Agent, see these links for more information:

- [Android MDM Agent - Getting started.](#)
 - [iOS MDM Agent - Getting started.](#)
-

After enrollment, the device will be displayed on the Agent section.

1.4.2 Create your Fleet

You must create a new fleet since the “not managed fleet” is the default one and therefore comes locked.

Warning: Carefull, be sure to not delete it since it will bring you some problems.

In order to create it, you only have to follow these steps:

- Click in the “+” button on the Fleet section
- Name it and Click on add.

When you return to the Fleet section, you'll see your created fleet listed.

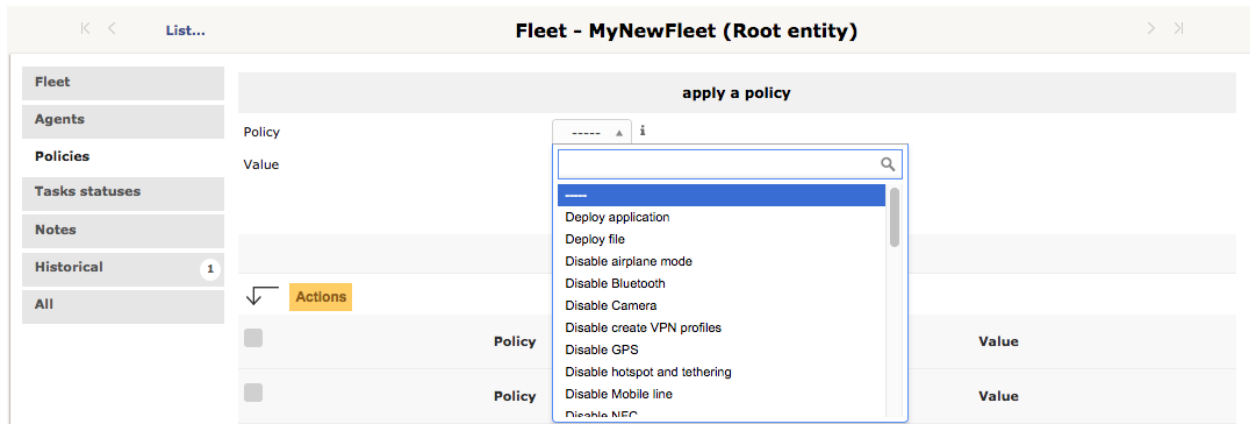
Add Policies

To add the policies:

- Go to the Fleets section
- Select your Fleet by clicking on it

- Select the Policies tab

There you can add all the policies your fleet requires.



Once you assign the devices to the Fleet, all the policies will be immediately applied!

Task Status

You can check the status of the policies deployed.

| Status | Description |
|--------------|---|
| Pending | Waiting for publication of the task, not yet received by the Agent |
| Received | Successfully received by the Agent, not yet applied |
| Done | Task successfully applied |
| Failed | Task could not be applied due to a failure |
| Canceled | The task was canceled |
| Incompatible | The task cannot be applied on the device because it is incompatible |
| Overriden | The task is overriden by a similar one applied on the device itself |
| Waiting | Only for deploy tasks which require time for application |

From the Fleets section

Select your fleet then click on the Task Status tab

| Fleet | Task status | | |
|---------------|------------------------------|------------------|-------|
| Agents 1 | Display (number of items) 20 | From 1 to 3 of 3 | |
| Tasks 3 | Policy | Status | Count |
| Task status 6 | Deploy application | pending | 2 |
| Notes | Disable Bluetooth | pending | 2 |
| Historical 4 | Disable GPS | pending | 2 |
| All | Policy | Status | Count |

From the Agent section.

Select the Agent then click on the Task Status tab

| | | | |
|---------------|--------------------------------|--------------------|---------------|
| Agent | | | |
| Geolocation | Display (number of items) 20 ▾ | | |
| Danger zone ! | | | |
| Tasks 2 | | | |
| Task status 5 | | | |
| Notes | | | |
| Historical 8 | | | |
| All | | | |
| | ID | Policy | Status |
| | 2 | Disable Bluetooth | canceled |
| | 3 | Disable GPS | canceled |
| | 4 | Disable Bluetooth | pending |
| | 7 | Disable GPS | pending |
| | 14 | Deploy application | pending |
| | ID | Policy | Status |

Assign the Agent to a Fleet

For this you must go to the Agent section

- Select the Agent
- In the Fleet option, select your fleet
- Click on save

| Agent | | Root entity | |
|--------------|---|--|--|
| Name | <input type="text" value="email@teclib.com"/> | Version | <input type="text" value="2.0"/> |
| Computer | <input type="text" value="HTC"/> ⓘ | Fleet | <input type="text" value="not managed fleet"/> ⓘ |
| Last contact | 2018-02-22 10:24:55 | | |
| Actions | <input type="button" value="Ping"/> | <input type="button" value="Geolocate"/> | |
| | | <input type="button" value="Save"/> | |

- Add App
- App Deploy
- Multiple file
- My New Fleet**
- New Fleet

Note: An Agent can be assigned to only one Fleet.

1.4.3 Adding Files & Applications

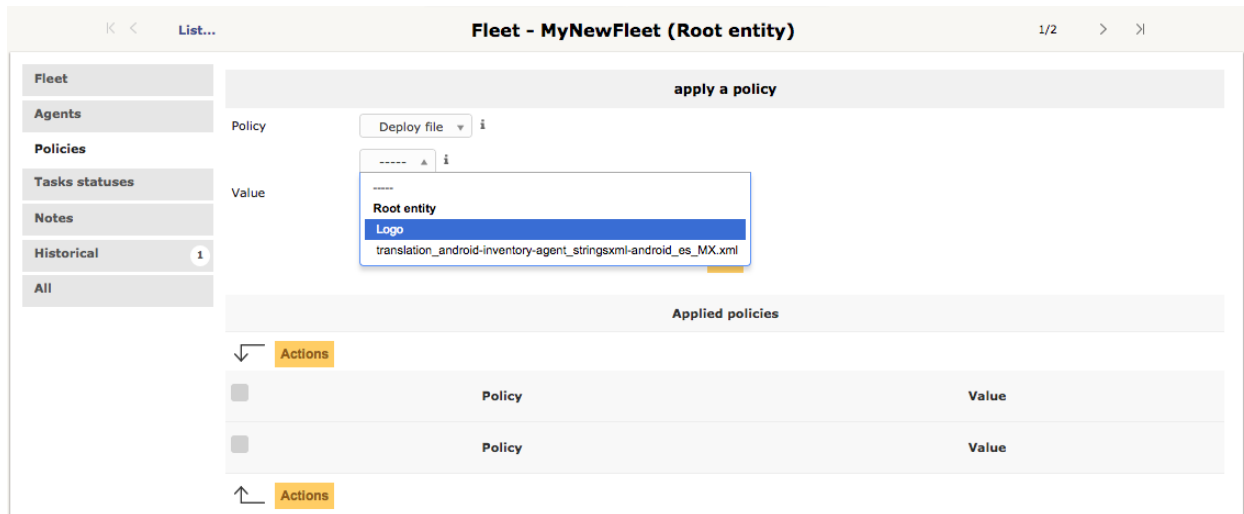
You must add in their respective sections the Files and Packages so you can deploy them on your Fleet.

Note: You must have previously added the APK and UPK types.

- Go to the Files section

- Click on the “+” button
- Name and select your file
- Click on Add

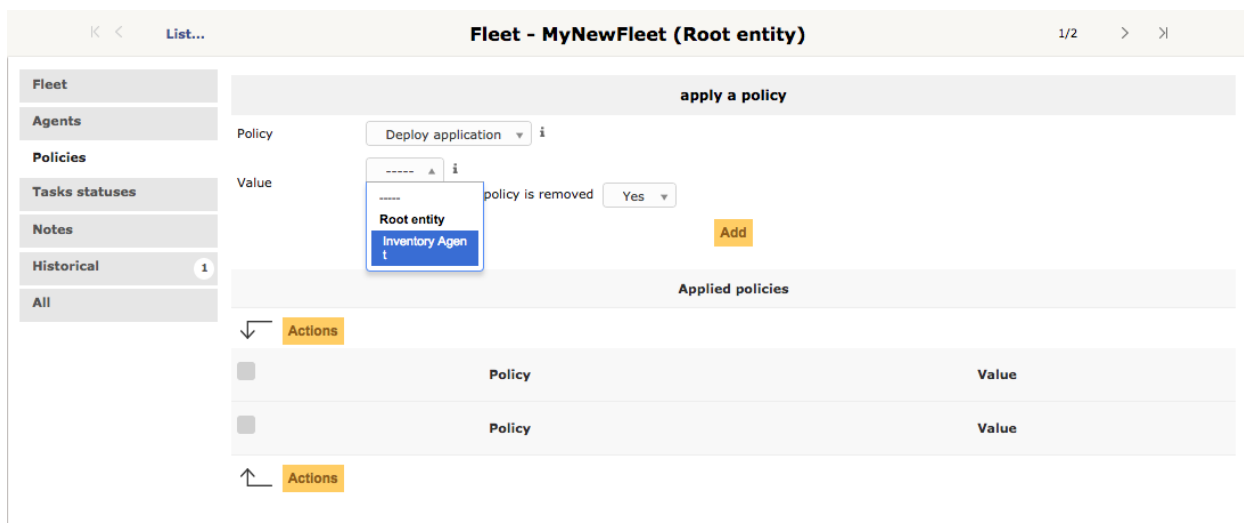
Now it will be available when you use the Deploy file policy.



The same procedure applies to the applications.

- Go to the Package section
- Click on the “+” button
- Name and select your application
- Click on Add

Now it will be available when you use the Deploy application policy.



1.4.4 Our Policies

The policies selected are sent in a JSON file to the MDM Agent with the values specified.

If the MDM Agent does not have the system privileges, it will require the confirmation of the user to implement the policies.

Note: Some policies are not available according to the version - API Level, for more information you can check the [Compatibility Matrix](#)

Disable

- Bluetooth: allows/forbids the use of Bluetooth.
Values: Yes or No.
- Airplane mode: allows/forbids the use of the airplane mode.
Values: Yes or No.
- Camera: allows/forbids the use of any camera on the phone.
Values: Yes or No.
- Create VPN profiles: allows/forbids the use to create VPN profiles. Available for devices with Api equal to or greater than 25.
Values: Yes or No.
- GPS: allows/forbids the use of the GPS.
Values: Yes or No.
- Hotspot and tethering: allows/forbids to configure the device as hotspot or tethering.
Values: Yes or No.
- Mobile line: allows/forbids the user to use the mobile line.
Values: Yes or No.
- NFC: allows/forbids the use of the Near Field Communication.
Values: Yes or No.
- Roaming: allows/forbids the use of Roaming. Available for devices with Api equal to or greater than 21.
Values: Yes or No.
- Screen capture: allows/forbids the user to make a screen capture. Available for devices with Api equal to or greater than 21.
Values: Yes or No.
- SMS and MMS: allows/forbids the user to send SMS and MMS.
Values: Yes or No.
- Speakerphone: allows/forbids the user to use speakerphone.
Values: Yes or No.
- Status bar: allows/forbids the user to use the status bar.
Values: Yes or No.

- USB ADB: allows/forbids the user to use the Android Debug Bridge through USB.
Values: Yes or No.
- USB MTP: allows/forbids the user to use the Media Transfer Protocol through USB.
Values: Yes or No.
- USB PTP: allows/forbids the use of the Picture Transfer Protocol through USB.
Values: Yes or No.
- Wifi: allows/forbids the user to connect to Wifi.
Values: Yes or No.
- Unknown sources: allows/forbids the installation of apps from unknown sources.
Values: Yes or No

Password

- Maximum failed password attempts for wipe: sets the number of failed attempts to unlock the device before wiping it.
Value: number of failed attempts.
- Maximum time to lock: the time in milisecond before to lock the device.
Values: number of miliseconds.
- Minimum letters required in password: minimum number of letters required.
Values: number of letters.
- Minimum lowercase letters required in password: minimum number of lowercase letters required.
Values: number of lowercase letters.
- Minimum non-letter characters required in password: minimum number of non-letter characters required.
Values: number of non-letter character.
- Minimum numerical digits required in password: minimum number of digits required.
Values: number of digits.
- Minimum password length: minimum length.
Values: number of minimum length.
- Minimum symbols required in password: minimum number of symbols required (@, %, =, &, *, etc).
Values: number of symbols.
- Minimum uppercase letters required in password: minimum uppercase letters.
Values: number of uppercase letters.
- Password enabled: enables/disables the password, if enabled it will request the password creation.
Values: Yes or No.
- Password quality: sets the complexity of the password.
Values:
 - Unspecified: no complexity specified.

- Something: requires a password but without a specific requirement.
 - Numeric: numbers only.
 - Alphabetic: letters or other symbols.
 - Alphanumeric: numbers and letters.
 - Complex: a combination of numbers, letters and symbols.
- Reset password: if a password is forgotten, it resets it to a new value.
Values: the new password (string).

Encryption

- Internal Storage encryption: encrypts the internal storage of the device.
Values: Yes or No.
- Use TLS: allows/forbids to use TLS protocol.
Values: Yes or No.

Apps & Files

Warning: To deploy a file or application, they must be previously uploaded to their respective sections.

- Deploy application: installs the application.
Value: The application.
Remove: if set to Yes, when the policy is removed, it will also remove the app by adding Remove application policy.
- Deploy file: downloads a file.
Values: the file.
Copy to: specifies the path where the file will be downloaded.
Remove: if set to Yes, when the policy is removed, it will also remove the file by adding Remove file policy.
- Remove application: uninstalls an application.
Value: the id of the APK.
Example: `com.remove.app`.
- Remove file: removes a file from the device.
Value: the name of the file with its format.
Example: `myfile.jpg`

Note: The Apps & Files policies can be used as many times as the Administrator requires.

Inventory

- Set an inventory frequency: set the frequency in which will be run the inventory.

Values: number of minutes.

1.4.5 Features

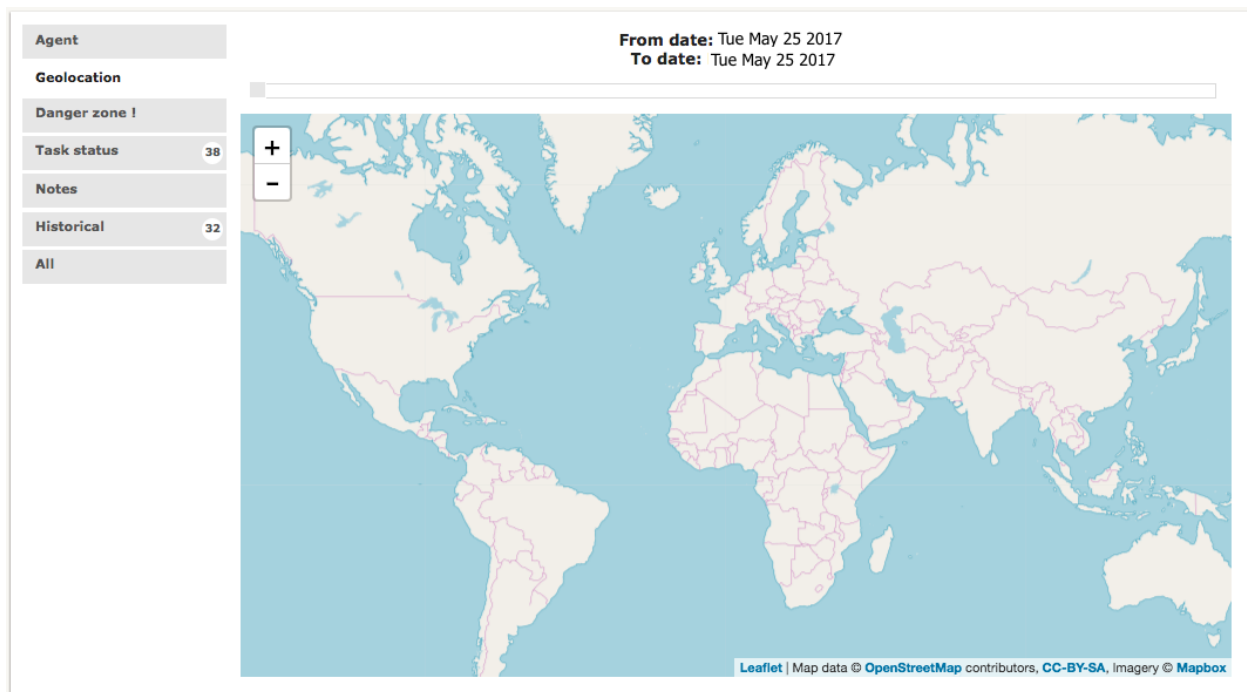
These are particular actions that can be taken only from the Agent section, some are accessible from the Agent tab and others from the Danger Zone! tab.

Ping

Allows to check the connectivity with the device.

Geolocate

Geolocates the device and shows its location on the map.



Inventory

Get the current inventory of the device.

Lock

If enabled, it will lock the device.

Wipe

Erase all the information in the device without deleting the Agent from the database.

Warning: After wipe there is no going back.

1.4.6 Unenrolling Agents

The Unenrollment will leave the device at the current state, this means the Flyve MDM Agent won't be uninstalled from the device, however the policies will be unapplied.

- Go to the Agent's section
- Select the Agent to Unenroll
- Select the tab Danger Zone!
- Click on Unenroll.



Note: To uninstall the App you must go to Security > Device Administrator and uncheck Flyve MDM first.

1.4.7 Delete an Agent

Deleting the agent will make all the policies unapplied.

- Go to the Agent's section
- Select the Agent to delete
- Select the tab Agent
- Click on Delete permanently.

| Agent | Agent | | Root entity | |
|----------------|---|--|--|--|
| Geolocation | Name | <input type="text" value="email@example.com"/> | Version | <input type="text" value="2.0.1-beta"/> |
| Danger zone ! | Computer | <input type="text" value="HTC"/> ⓘ | Fleet | <input type="text" value="not managed fleet"/> ⓘ |
| Task status 38 | Last contact | 2018-01-20 13:24:15 | | |
| Notes | Actions | <input type="button" value="Ping"/> | <input type="button" value="Geolocate"/> | <input type="button" value="Inventory"/> |
| Historical 19 | <input type="text" value="text"/> <input type="button" value="Save"/> | | | |
| All | <input type="button" value="Delete permanently"/> | | | |

Important: Apps & Files already deployed won't be removed after deleting or unenrolling an Agent.

Warning: When deleting or unenrolling the Agent, you can't go back. You will have to re-enroll the device again.

1.5 How it works

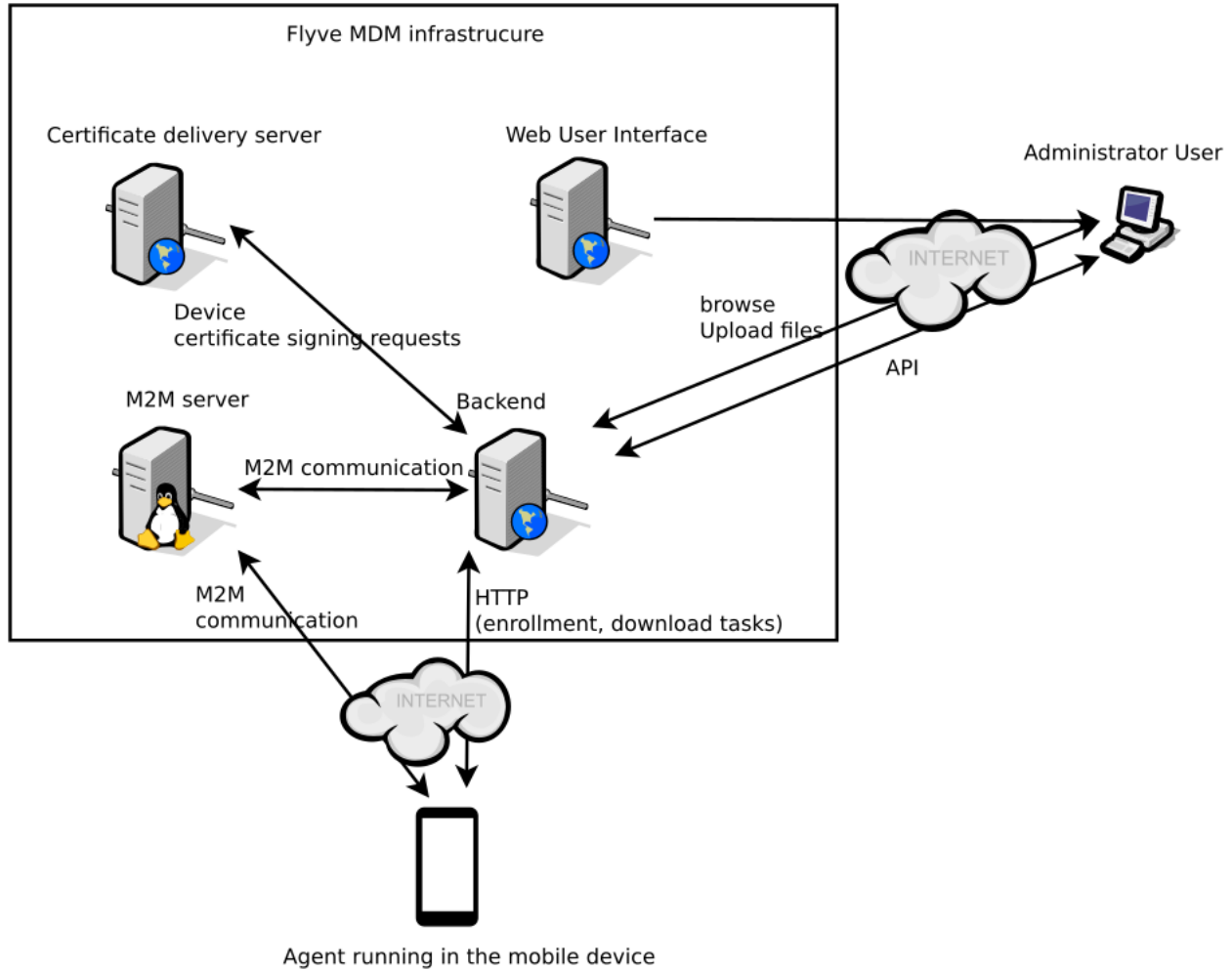
1.5.1 General architecture

The Mobile Device Management is composed of:

- a user interface server for the administrator
- a backend server
- a M2M server
- an agent installed in the managed device.

Any connection through an untrusted network is encrypted.

A mobile device cannot guarantee a permanent connection with the backend. The M2M protocol provides features to handle loss of connectivity and guarantee delivery of important messages in both directions. The agent takes control of the device to maintain a minimal connectivity with the backend server via the M2M protocol and execute requests from the backend.



The certificate delivery server needs a private key to complete its role. It must communicate only with the backend and no communication is allowed from internet or any other untrusted network. It must run on a distinct server from the backend, the M2M server and the web User Interface.

All communications must be TLS encrypted.

The M2M server is a gateway between the devices and the backend, providing some helpful features to handle the unstable connectivity with devices. These features are available in a messaging queue protocol.

1.6 MQTT Specifications

1.6.1 Introduction

MQTT messages are JSON strings.

1.6.2 MQTT Topic hierarchy

```
<FlyvemdmManifest>
+- Status
```

(continues on next page)

(continued from previous page)

```

|   +- Version

<1st entity-ID>
+- agent
|   +- <1st Device's serial>
|   |   +- Command/Subscribe
|   |   +- Command/Ping
|   |   +- Command/Geolocate
|   |   +- Command/Lock
|   |   +- Command/Wipe
|   |   +- Command/Inventory
|   |   +- Command/Unenroll
|   |
|   |   +- Status/Ping
|   |   +- Status/Geolocation
|   |   +- Status/Inventory
|   |   +- Status/Install
|   |   +- Status/Unenroll
|   |   +- Status/Task
|   |   +- Status/Online
|   |
|   +- <2nd Device's serial ...>
|   +- <Nth Device's serial >
|
+- fleet
  +- <1st fleet ID>
    +- <Policy>
      +- <1st policySymbol>
        +- Task
        +- <task ID>
      +- <2nd policySymbol>
        +- Task
        +- <task ID>
      +- <Nth policySymbol>
        +- Task
        +- <task ID>
    +- <2nd fleet ID ...>
    +- <Nth fleet ID>

+- <2nd Entity-ID...>
+- <Nth entity-ID>

```

1.6.3 MQTT message for policy deployment

There are many policies available. Some may be applied, some not.

When the backend needs to notify a fleet or an agent about new policy settings, the backends send all policies actually applied, in a single message.

Example:

```

{
  "storageEncryption":"false",

```

(continues on next page)

(continued from previous page)

```
{  
  "taskId":37  
}
```

1.6.4 MQTT messages sent by the backend

Subscription to topics

Subscription to a fleet occurs when a device enrolls, and when an administrator moves a device from a fleet to another.

The database model makes sure a device is assigned to one and only one fleet. However the JSON format in the message allows a possible removal of this constraint in the future.

Sub topic /Command/Subscribe

```
{  
  "subscribe" : [  
    {"topic": "topic_1"},  
    {"topic": "topic_2"},  
    {"topic": "topic_3"}  
  ]  
}
```

QoS of the message = 1

Device status policies

Ping query

Sub topic Command/Ping

```
{  
  "query" : "Ping"  
}
```

Expected answer:

Sub topic Status/Ping !

Geolocation query

Sub topic Command/Geolocate

```
{  
  "query" : "Geolocate"  
}
```

Expected answer:

Sub topic Status/Geolocation

```
{"latitude":48.1054276,"longitude":-1.67820699,"datetime":1476345332}
```

Note: the datetime is in Unix time format, and *must* be on UTC timezone for proper save in DB by the backend.

Unenroll query

Sub topic Command/Unenroll

```
{
  "unenroll": "Now"
}
```

Expected answer:

Subtopic Status/Unenroll

```
{
  "unenroll": "unenrolled"
}
```

Password settings policies

There are several password policies to setup the type of password required on a device and the complexity of the challenge.

Topic: 0/fleet/1/Policy/passwordEnabled/Task/2

```
{ "passwordEnabled": "true|false", "taskId": "2"},
```

Topic: 0/fleet/1/Policy/passwordQuality/Task/3

```
{ "passwordQuality" : "PASSWORD_QUALITY_NUMERIC|PASSWORD_QUALITY_ALPHABETIC|PASSWORD_
→QUALITY_ALPHANUMERIC|PASSWORD_QUALITY_COMPLEX|PASSWORD_QUALITY_SOMETHING|PASSWORD_
→QUALITY_UNSPECIFIED", "taskId": "3"},
```

Topic: 0/fleet/1/Policy/passwordMinLetters/Task/4

```
{ "passwordMinLetters" : "0|1|2|..", "taskId": "4"},
```

Topic: 0/fleet/1/Policy/passwordMinLowerCase/Task/5

```
{ "passwordMinLowerCase" : "0|1|2|..", "taskId": "5"},
```

Topic: 0/fleet/1/Policy/passwordMinUpperCase/Task/6

```
{ "passwordMinUpperCase" : "0|1|2|..", "taskId": "6"},
```

Topic: 0/fleet/1/Policy/passwordMinNonLetter/Task/7

```
{ "passwordMinNonLetter" : "0|1|2|..", "taskId": "7"},
```

Topic: 0/fleet/1/Policy/passwordMinNumeric/Task/8

```
{ "passwordMinNumeric" : "0|1|2|..", "taskId": "8"},
```

Topic: 0/fleet/1/Policy/passwordMinLength/Task/9

```
{ "passwordMinLength" : "0|1|2|..", "taskId": "9"},
```

Topic: 0/fleet/1/Policy/MaximumFailedPasswordsForWipe/Task/10

```
{ "MaximumFailedPasswordsForWipe" : "0|1|2|..", "taskId": "10"},
```

Topic: 0/fleet/1/Policy/MaximumTimeToLock/Task/11

```
{ "MaximumTimeToLock" : "time in MS", "taskId": "11"},
```

Topic: 0/fleet/1/Policy/passwordMinSymbols/Task/12

```
{ "passwordMinSymbols" : "0|1|2|..", "taskId": "12"}
```

Application deployment policies

There are two application deployment policies. One policy actually deploys an application, the other one removes an application. These policies may both apply multiple times on the same fleet target.

The deployment policy retains a `remove_on_delete` flag. If this flag is set, removal of the deployment policy will create a policy in charge of the deletion of the same application, applied to the same fleet target.

Example

Three deployment policies are applied to a single fleet target

Topic: 0/fleet/1/Policy/deployApp/Task/11

```
{"deployApp" : "org.fdroid.fdroid", "id" : "1", "version": "18", "taskId": "11"},
```

Topic: 0/fleet/1/Policy/deployApp/Task/14

```
{"deployApp" : "com.domain.application", "id" : "42", "version": "2", "taskId": "14"},
```

Topic: 0/fleet/1/Policy/deployApp/Task/19

```
{"deployApp" : "com.domain.application", "id" : "5", "version": "42", "taskId": "19"}
```

One application removal policies is applied to a fleet target

Topic: 0/fleet/1/Policy/removeApp/Task/16

```
{  
  "removeApp" : "org.fdroid.fdroid",  
  "taskId": "16"  
}
```

File deployment policies

Example of file deployment policy

Topic: 0/fleet/1/Policy/deployFile/Task/23

```
{
  "deployFile": "%SDCARD%/path/to/file.ext",
  "version": "18",
  "taskId": "23"
}
```

Peripheral related policies

Topic: 0/fleet/1/Policy/disableCamera/Task/25

```
{
  "disableCamera" : "true|false",
  "taskId": "25"
}
```

Device access policies

Lock a device

To lock a device as soon as possible

```
{
  "lock": "now"
}
```

Unlock a device

To unlock a device

```
{
  "lock": "unlock"
}
```

Wipe a device

Sub topic /Command/Wipe

```
{
  "wipe" : "now"
}
```

QoS of the message = 2

Connectivity policies

3 policies are available, a registered user can choose to apply only some of them. This means the array in the JSON may contain a subset of the JSON array below.

Topic: 0/fleet/1/Policy/disableWifi/Task/25

```
{
  "disableWifi" : "true|false",
  "taskId": "25"
}
```

Topic: 0/fleet/1/Policy/disableGPS/Task/27

```
{
  "disableGPS" : "true|false",
  "taskId": "27"
}
```

Topic: 0/fleet/1/Policy/disableBluetooth/Task/28

```
{
  "disableBluetooth" : "true|false",
  "taskId": "28"
}
```

(Uhuru Mobile) Applications available from the launcher

(specification only, not implemented)

```
{
  "launcher":
  { "code" : "update|start|unlock",
    "data" : [
      { "name" : "com.android.contacts" },
      { "name" : "com.android.mms" },
      { "name" : "com.android.settings" }
    ]
  }
}
```

Property:

- **code**: command identifier
- **_start_**: starts the application launcher
- **_update_**: updates the application launcher
- **_unlock_**: unlocks the 'screen pinning'
- **data**: applications' list
- **_name_**: application package to autorise on terminal

Ps1: In the case only one applications is referenced, it will be executed automatically (self-launch)

Ps2: In the case it is an applications' list, they will be displayed on desk.

1.6.5 MQTT messages sent by a device

FlyveMDM version manifest

This subtopic contains metadata about Flyve MDM published to each device. This is the current version of the backend.

Sub topic /FlyvemdmManifest/Status/Version

```
{
  "version": "0.6.0"
}
```

Task status

This subtopic is used by agents to feedback the progress of a policy deployment.

Sub topic /Status/Task/<task ID>

```
{
  "status": "in progress"
},
```

The status value may be any string up to 255 chars except the reserved statuses (see below). The status should be a short string. In the future, statuses will be normalized.

Reserved statuses:

- queued (when a task is created, this value is used to initialize the task status)
- pushed (when a message is sent by the backend, this value is used to update the status)

1.6.6 Sources

- [Spec MQTT 3.1.1](#)

1.7 Use Cases

1.7.1 Enrollment process

Enrollment requests

The guest user enrolls himself using a QR code or a deeplink sent by email. They both contain the same useful data to build several HTTP requests.

The user token is a token to authenticate against the REST API. When an administrator invites someone to enroll his devices, an account is created. The login is the email address provided by the administrator.

The user API token is unique among all user accounts. Using it alone is sufficient to get a session, as described in GLPI documentation.

The invitation token is a hash with limited lifetime to identify the invitation being used at enrollment time.

After the enrollment the agent must connect to the MQTT broker. The serial of the device and the password sent by the backend in the enrollment process are the credentials.

Deeplink

Flyve MDM Deeplink, is a microservice hosted by us, that generates a deeplink with the information of the invitation to perform the enrollment of the device.

The organizations have the option to set up their own instance of Flyve MDM Deeplink, in order to manage by themselves this microservice.

Content

The deeplink contains some base64 encoded semicolon separated fields like a CSV format:

- the URL of the backend for REST API requests
- a user API token belonging to a human user account
- an invitation token
- the name of the company's helpdesk
- the phone number of the company's helpdesk
- the website of the company's helpdesk
- the email of the company's helpdesk

All fields related to the helpdesk may not be populated by the administrators. The fields are ordered.

```
http://api.domain.com/deeplink/?  
→data=aHR0cDovL2FwaS5kb21haW4uY29tLzs0NWVyamJlZGtscTU4NjVzZGtqaGprcztsa2hqZmtnc2RmNTQ2NjM0cztjb21wYV
```

Uses

- Email account

When the invitation is opened from the email account of the user in the respective device to enroll, the email will display the deeplink with the information of the invitation encoded. Also the QR code image is attached.

After tapping the link, the MDM Agent will recognize it and start the enrollment.

- Computer browser

The deeplink can also be opened in a computer's browser, in this case the deeplink will decode part of the information and display the Helpdesk information and the QR code, so the user can scan it from the MDM Agent and make the enrollment.

- Mobile device browser

In case the invitation is opened with the browser of the device, it will display a button that contains the deeplink so the MDM Agent can recognize it and start the enrollment.

Obtaining a session token

The agent must first acquire a session token issuing a request like this:

```
GET http://api.domain.com/initSession?user_token=45erjbudklq5865sdkjhjks  
Content-Type: application/json
```

Note: the header is required

Answer:

```
200 OK
{
  "session_token": "83af7e620c83a50a18d3eac2f6ed05a3ca0bea62"
}
```

Note: The URL of the backend is the base for requests built by the client, which must be suffixed by the endpoint to reach. See the [GLPI documentation](#) to know more about the rest API of GLPI.

Check the current profile

GLPI requires users have at least one profile to allow them to login. When users have several profiles, they need to send a request to switch between them. This may happen if a user enrolls his own device.

To ensure the device always use the guest profile of the user, the device must run this request

```
GET http://api.domain.com/getFullSession?session_
→token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Content-Type: application/json
```

Note: the header is required

Answer (truncated):

```
200 OK
{
  "glpi_plugins": {
    "1": "flyvemdm",
    "2": "fusioninventory"
  },
  "snip": "many content here, not shown",
  "glpiactiveprofile": {
    "id": 9,
    "name": "Flyve MDM guest users",
  },
  "snip": "many content here, not shown",
  "plugin_flyvemdm_guest_profiles_id": 9
}
```

Many data are returned. The agent must read inside `glpiactiveprofile` the value **id**. This is the ID of the current profile. If it differs from the value of **plugin_flyvemdm_guest_profiles_id** then the device needs to change its profile. If they match, the device may skip the next request.

Switch to guest profile

```
POST http://api.domain.com/changeActiveProfile?profiles_id=9&session_
↪token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Content-Type: application/json
```

Note: **the header is required**

Note the value **profiles_id** in the query string. The value must match **plugin_flyvemdm_guest_profiles_id** found in the previous request

Answer if the request fails

| HTTP Status | Error Message | Cause |
|-------------|----------------|--|
| 400 | Bad Request | The request is probably malformed. |
| 404 | Item not found | The requested profile is not available or does not exists. |

In both cases, the enrollment cannot continue.

Answer if the request succeeds

```
200 OK
```

Instantiation of the agent

After a session token is acquired, the agent must create itself in the backend. It must send its serial number or its UUID. Sending both is recommended to avoid inventory problems.

```
POST http://api.domain.com/PluginFlyvemdmAgent?session_
↪token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Content-Type: application/json
```

Payload:

```
{ "input":
{
  "_email": "emailaddress@domain.com",
  "_invitation_token": "lkhjfkgsdf546634s",
  "_serial": "0123456ATDJ-045",
  "_uuid": "49D53434-0200-9D08-9000-01DEA9028055",
  "csr": "",
  "firstname": "my first name",
  "lastname": "my lastname",
  "version": "0.99.0",
  "type": "android"
}
}
```

- **_email**: email address of the invited user
- **_invitation_token**: invitation token found in the invitation message
- **_serial**: serial number of the device (serial or uuid required, both is better)
- **_uuid**: UUID of the device (serial or uuid required, both is better)

- `csr`: certificate signing request (not implemented yet)
- `firstname`: first name of the user (optional)
- `lastname`: last name of the user (optional)
- `version`: version of the MDM agent
- `type`: type of MDM used (must be one of the MDM types supported by the backend)

Answer if the request fails

If the enrollment fails, a JSON array is returned

```
400 Bad Request
[
  "ERROR_GLPI_ADD",
  "wrong email address"
]
```

The first string is a software error code. The second string is an error message displayable to the user.

If the request fails, the enrollment cannot continue.

Answer if the request succeeds

```
201 OK
{
  "id": "17",
  "message": "XML has been imported succesfully!"
}
```

`id` is the ID of the agent in the database. It is needed for the next request.

Get agent's initial settings (only if enrollment succeeded)

```
GET http://api.domain.com/PluginFlyvemdmAgent/17?session_
token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Content-Type: application/json
```

Answer if the request succeeds

```
200 OK
{
  "id": 65,
  "name": "user5@teclib.com",
  "version": "1.0.0",
  "computers_id": 65,
  "wipe": 0,
  "lock": 0,
  "enroll_status": "enrolled",
  "entities_id": 76,
  "plugin_flyvemdm_fleets_id": 103,
```

(continues on next page)

(continued from previous page)

```
"last_report": null,
"last_contact": null,
"certificate": "",
"topic": "/76/agent/AZERTY",
"mqttpasswd": "YTQUJagG7iBas45thrbwDxtzaQxZIhp4g",
"api_token": "2qanibblaoi16d7edqm6wcl14y6f3c24vtjupk4r"
"broker": "demo.flyve.org",
"port": 8883,
"tls": 1,
"android_bugcollector_url": "",
"android_bugcollector_login": "",
"android_bugcollector_passwd": "",
"links": [
  {
    "rel": "Computer",
    "href": "https://demo.flyve.org/glpi/apirest.php/Computer/65"
  },
  {
    "rel": "Entity",
    "href": "https://demo.flyve.org/glpi/apirest.php/Entity/76"
  },
  {
    "rel": "PluginFlyvemdmFleet",
    "href": "https://demo.flyve.org/glpi/apirest.php/PluginFlyvemdmFleet/103"
  }
]
```

Note: The property certificate in the JSON payload is the certificate delivered to the agent if the client certificate feature is enabled.

- **api_token:** to consume API from GLPI. Used to download files and applications from HTTP(S).
- **mqttpasswd:** password to access MQTT. Login is the serial of the device.

The api_token delivered by this request replaces the user_token used in the first request **initSession**. The agent must forget the user_token and save for later use the api_token received from this request.

Logout after enrollment

```
GET http://api.domain.com/killSession?session_
→token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Content-Type: application/json
```

The answer should contain an empty body.

Answer if the request succeeds

```
200 OK
```

Answer if the request fails

```
400 Bad Request
```

1.7.2 Files and Applications Download

Prerequisites

The device needs to be logged with a user account. Refer to enrollment process to know how to log in.

Download a file

```
GET http://api.domain.com/PluginFlyvemdmFile/1?session_
↪token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Accept: application/octet-stream
Content-Type: application/json
range: bytes=0-
```

The headers **Accept** and **Content-Type** are mandatory. The header **Range** is optional and allows to download a range of bytes.

The body of the answer is the file or a part of the file, depending on the usage of the **Range** header.

Download an application

```
GET http://api.domain.com/PluginFlyvemdmPackage/1?session_
↪token=83af7e620c83a50a18d3eac2f6ed05a3ca0bea62
Accept: application/octet-stream
Content-Type: application/json
range: bytes=0-
```

The headers **Accept** and **Content-Type** are mandatory. The header **Range** is optional and allows to download a range of bytes.

The body of the answer is the file or a part of the file, depending on the usage of the **Range** header.

Using ranges

In a nutshell:

- bytes=0- means bytes [0; +infinite[(in other words: the whole file)
- bytes=10-100 means bytes [10; 100]
- bytes=-100 means bytes [0; 100] It is similar to bytes=0-100

Answered headers

- Accept-Ranges: bytes
- Access-Control-Expose-Headers: content-type, content-range, accept-range
- Cache-Control: private, must-revalidate

- Connection: keep-alive
- Content-Length: 504750
- Content-Range: bytes 0-504750/504750
- Content-Transfer-Encoding: binary
- Content-Type: application/octet-stream
- Content-disposition: attachment; filename="b_1_q_0_p_0.jpg"
- Date: Fri, 02 Sep 2016 14:15:23 GMT
- Expires: Mon, 26 Nov 1962 00:00:00 GMT
- Last-Modified: Fri, 02 Sep 2016 13:57:41 +0200

1.7.3 Agent Online Status

Flyve MDM takes benefit from the will and testament concept of MQTT protocol. When a client connects to the broker it can send a message to the broker. When the client goes offline, the broker sends the message on behalf of the disconnected client.

Online status

The online status relies on will and testament feature of MQTT. When a device goes online, it must send

Sub topic /Status/Online

```
{"online": "true"}
```

When the device disconnects it must set a will and testament message which will fire to the backend from the Broker on behalf of the agent.

Sub topic /Status/Online

```
{"online": "false"}
```

This way, when the Broker detects loss of connectivity with an agent, it will inform the backend that the devices are gone offline.

1.7.4 Update user

This works only when a user has UPDATE right on users, a guest profile cannot update himself from the API currently, see [GLPI issue #2568](#).

The agent may update user information like phone number, emails, name.

This can be done with the endpoint `User`. Only fields requiring an update should be specified.

PUT `apiREST.php/User/:id`

```
{
  "input": {
    "firstname": "new firstname",
    "realname": "new realname",
    "phone": "0123456789",
```

(continues on next page)

(continued from previous page)

```

    "phone2": "0987654321",
    "_useremails": {
      "0": "email1@example.com",
      "1": "email2@example.com"
    },
    "_default_email": "1",
  }
}

```

1.7.5 Agent deletion

There are two cases where an Agent must be deleted from the database:

- After succesful enrollment
- After explicit deletion by an administrator

Unenrollment

When an administrator unenrolls an agent, a MQTT message is sent to the device running this agent to process unenrollment. Once the unenrollment is done, the agent sends a MQTT message to acknowledge the unenrollment request.

The agent won't be removed from the database since the device is under the assumption that is still owned by a legitimate user and is used in normal operations.

Forced deletion

When an enrolled device is lost or stolen, and it appears impossible to retrieve it, the administrator may want to delete its data from the database. In such case we cannot unenroll th device as it is very likely offline forever. We need to delete the agent and maybe the associated computer by force.

To do this the administrator deletes the agent from his user interface. Next he is free to delete or keep the associated computer. This means that deleting the agent does not cascade a computer deletion. This is an extra and manual action taken by the administrator.

1.8 Rest API

All the information concerning the Api Endpoints and Error Messages.

1.8.1 API Endpoints Specifications

Flyve MDM relies in GLPI's REST API for communications. The plugin provides additional endpoints for its item-types.

Agent

This endpoint allows the API consumer to read or modify agents.

- Name: PluginFlyvemdmAgent

File

This endpoint allows the API consumer to read or modify files to deploy to devices.

- Name: PluginFlyvemdmFile

Geolocation

This endpoint allows the API consumer to read or modify geographical positions acquired from GPS.

- Name: PluginFlyvemdmGeolocation

Invitation

This endpoint allows the API consumer to read or modify invitations.

- Name: PluginFlyvemdmInvitation

InvitationLog

This endpoint allows the API consumer to read invitation events (errors only when a device failed to enroll).

- Name: PluginFlyvemdmInvitationLog

Package

This endpoint allows the API consumer to read or modify packages (aka applications for mobile devices).

- Name: PluginFlyvemdmPackage

Policy

This endpoint allows the API consumer to read policies handled by Flyve MDM.

- Name: PluginFlyvemdmPolicy

PolicyCategory

This endpoint allows the API consumer to read policy categories handled by Flyve MDM.

- Name: PluginFlyvemdmPolicyCategory

WellKnownPath

This endpoint allows the API consumer to read well known path in devices.

- Name: PluginFlyvemdmWellKnownPath

1.8.2 Error Messages

GLPI

- `initSession`

| HTTP Status | Error Symbol | Error Message |
|-------------|---------------------------------------|--|
| 200 | | |
| 400 | ERROR_LOGIN_PARAMETERS_MISSING | Parameter(s) login, password or user_token missing |
| 400 | ERROR_LOGIN_WITH_CREDENTIALS_DISABLED | Session with credentials is disabled |
| 401 | ERROR_GLPI_LOGIN | Parameter user_token seems invalid |

- `killSession`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `changeActiveEntities`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `getMyEntities`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `getActiveEntities`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `changeActiveProfile`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `getMyProfiles`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `getActiveProfile`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- `getFullSession`

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---------------|
| 200 | | |

- getItem

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------|---|
| 200 | | |
| 401 | | You don't have permission to perform this action. |

- getItem

| HTTP Status | Error Symbol | Error Message |
|-------------|---|--|
| 200 | | |
| 400 | ERROR_ITEMTYPE_NOT_FOUND_NOR_COMMONDBTM | Itemtype is not found or an instance of CommonDBTM |

- listSearchOptions

| HTTP Status | Error Symbol | Error Message |
|-------------|---|--|
| 200 | | |
| 400 | ERROR_ITEMTYPE_NOT_FOUND_NOR_COMMONDBTM | Itemtype is not found or an instance of CommonDBTM |

- searchItems

| HTTP Status | Error Symbol | Error Message |
|-------------|--------------------------|--|
| 200 | | |
| 400 | ERROR_RANGE_EXCEED_TOTAL | Provided range exceed total count of data: placeholder |
| 500 | ERROR_SQL | Unexpected SQL Error: placeholder |

- createItems

| HTTP Status | Error Symbol | Error Message |
|-------------|------------------------|---------------|
| 200 | | |
| 207 | ERROR_GLPI_PARTIAL_ADD | |
| 400 | ERROR_GLPI_ADD | |

- updateItems

| HTTP Status | Error Symbol | Error Message |
|-------------|---------------------------|---------------|
| 200 | | |
| 207 | ERROR_GLPI_PARTIAL_UPDATE | |
| 400 | ERROR_GLPI_UPDATE | |

- deleteItems

| HTTP Status | Error Symbol | Error Message |
|-------------|---------------------------|---------------|
| 200 | | |
| 207 | ERROR_GLPI_PARTIAL_DELETE | |
| 400 | ERROR_GLPI_DELETE | |

Flyve MDM

All itemtypes may produce the error messages from GLPI, depending on their HTTP verb. Additional errors messages are below.

PluginFlyvemdmAgent

- addItem

| HTTP Status | Error Symbol | Error Message |
|-------------|----------------|-------------------------------------|
| 400 | ERROR_GLPI_ADD | Unable to find an enrollment method |

- updateItem

| HTTP Status | Error Symbol | Error Message |
|-------------|-------------------|---|
| 400 | ERROR_GLPI_UPDATE | The fleet of the device does not longer exist |
| 400 | ERROR_GLPI_UPDATE | The target fleet does not exist |
| 400 | ERROR_GLPI_UPDATE | The device is not enrolled yet |
| 400 | ERROR_GLPI_UPDATE | Timeout querying the device |
| 400 | ERROR_GLPI_UPDATE | Timeout querying the device inventory |

- deleteItem

| HTTP Status | Error Symbol | Error Message |
|-------------|-------------------|--|
| 400 | ERROR_GLPI_DELETE | Failed to find the guest user profile |
| 400 | ERROR_GLPI_DELETE | Failed to remove guest habilitation for the user of the device |
| 400 | ERROR_GLPI_DELETE | Failed to delete the device |
| 400 | ERROR_GLPI_DELETE | Failed to delete documents attached to the device |
| 400 | ERROR_GLPI_DELETE | Timeout querying the device inventory |

PluginFlyvemdmFile

- addItem

| HTTP Status | Error Symbol | Error Message |
|-------------|----------------|-----------------------|
| 400 | ERROR_GLPI_ADD | Could not upload file |

- updateItem

| HTTP Status | Error Symbol | Error Message |
|-------------|-------------------|----------------------------|
| 200 | ERROR_GLPI_UPDATE | File successfully uploaded |
| 207 | ERROR_GLPI_UPDATE | Could not upload file |
| 400 | ERROR_GLPI_UPDATE | Could not save file |

PluginFlyvemdmFleet_Policy

- addItem

| HTTP Status | Error Symbol | Error Message |
|-------------|----------------|--|
| 400 | ERROR_GLPI_ADD | Fleet and policy must be specified |
| 400 | ERROR_GLPI_ADD | Policy not found |
| 400 | ERROR_GLPI_ADD | Incorrect value for this policy |
| 400 | ERROR_GLPI_ADD | Cannot find the target fleet |
| 400 | ERROR_GLPI_ADD | Cannot apply a policy on a not managed fleet |
| 400 | ERROR_GLPI_ADD | The requirements for this policy are not met |
| 400 | ERROR_GLPI_ADD | Failed to apply the policy |

- updateItem

| HTTP Status | Error Symbol | Error Message |
|-------------|-------------------|--|
| 400 | ERROR_GLPI_UPDATE | Policy not found |
| 400 | ERROR_GLPI_UPDATE | Incorrect value for this policy |
| 400 | ERROR_GLPI_UPDATE | Cannot find the target fleet |
| 400 | ERROR_GLPI_UPDATE | Cannot apply a policy on a not managed fleet |
| 400 | ERROR_GLPI_UPDATE | The requirements for this policy are not met |
| 400 | ERROR_GLPI_UPDATE | Failed to apply the policy |

- deleteItem

| HTTP Status | Error Symbol | Error Message |
|-------------|-------------------|--|
| 400 | ERROR_GLPI_DELETE | Policy not found |
| 400 | ERROR_GLPI_DELETE | Fleet not found |
| 400 | ERROR_GLPI_DELETE | Incorrect value for this policy |
| 400 | ERROR_GLPI_DELETE | Cannot find the target fleet |
| 400 | ERROR_GLPI_DELETE | Cannot apply a policy on a not managed fleet |
| 400 | ERROR_GLPI_DELETE | The requirements for this policy are not met |
| 400 | ERROR_GLPI_DELETE | Failed to apply the policy |

1.9 Docker Environment

We count with a docker environment to easily set up Flyve MDM, allowing you to have a development instance ready to work on, to know where to start, just follow our [documentation](#).

1.10 UML Diagrams

Here we present you the sequence diagrams of the main interactions between the GLPI Plugin and Flyve MDM Agent.

1.10.1 Enrollment MQTT

1.10.2 Enrollment FCM

1.10.3 Status Online/Offline MQTT

1.10.4 Status Online/Offline FCM

The Offline status is only sent to the backend when there is an unenrollment or wipe, if the Agent loses connectivity in any other form, it won't send the status

1.10.5 Implementation of policies MQTT

1.10.6 Implementation of policies FCM

1.10.7 File & App download

1.10.8 Ping request

1.10.9 Geolocation

1.10.10 Wipe

1.10.11 Lock/Unlock

1.10.12 Inventory

1.10.13 Unenrollment MQTT

1.10.14 Unenrollment FCM

1.11 Join our community!

Welcome to our awesome team!

We are more than happy to accept external contributions to the project in the form of feedback, translations, bug reports, and even better, pull requests!

Starting from the basics, here we'll be explaining everything you need to know to contribute in Flyve MDM plugin for GLPI.

1.11.1 Download & New Accounts

You'll require to install and open new accounts in:

- [Download Git](#)
- [Download Visual Studio Code](#)
- [Download GPG command line tools](#) -> Mac OSX users download GPG suite
- [Telegram account](#)

- [GitHub account](#)
- [Download Keybase](#) if you don't have an account already.

1.11.2 Set things up

Git

After installing git, run:

```
git config --global user.name "First.Name Last.Name"
git config --global user.email "my.email@email.com"
```

Tip: Git provides the [Pro Git book](#), available in several languages, and [Try Git](#), a 15 min tutorial, both very helpful if you're starting with SCM.

GitHub

For security reasons, we demand our members to have enabled the following Authentication measures. The instructions to configure each one are available from the [GitHub Help Documentation](#), this assures us that any change added to any of our repositories comes from an authorized member of our team.

Tip: We strongly recommend to configure your GitHub account and Git locally, if you are a collaborator, it will increase security to your account and work.

Two Factor Authentication

This is the easiest step, you only need to follow the [GitHub Documentation](#), [Securing your account with 2FA](#), there are several methods to do this, for example by [SMS Text](#) or a [TOTP app](#).

Test it!

Sign out then sign in again, GitHub will ask you the Authentication code.

SSH

Follow the [GitHub guide](#), [Connecting to GitHub with SSH](#), there is everything you need to know to successfully add your SSH key.

Test it!

1. [Create a Test Repository on GitHub](#)
2. Create a **test** directory on your work environment
3. From the terminal go to your **test** directory and run: `git clone git@github.com:MY-USER-NAME/MY-REPO-NAME.git`

4. Run `ls -a` to list all directories in `test/`

If there is a new folder with your repo name, then the setup of SSH was successful!

GPG signing

This configuration requires some patience.

1. Open a Terminal.
2. Use the following command to create your GPG key: `keybase pgp gen`
 - 2.1. Add your information: Name, Email(s)
 - 2.2. Make sure to push an encrypted copy of your secret key to keybase.io
 - 2.3. Add a passphrase to encrypt the key, be sure to remember it since the keychain will request it to save your keys.
3. Run the following command `keybase pgp export` and copy everything that starts and ends with:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
-----END PGP PUBLIC KEY BLOCK-----
```

4. Add it to your GitHub account, go to settings > SSH and GPG keys > New GPG key
5. Get the ID of your PGP key:
 - Run `gpg --list-secret-keys --keyid-format LONG`
 - Check your keybase profile

```
➔ ~ gpg --list-secret-keys --keyid-format LONG
/Users/mac5/.gnupg/pubring.gpg
-----
sec  rsa4096/B344E73DA95715F4 2017-08-09 [SC] [caduca: 2033-08-05]
    B50ADC2811C5819FE6A20D67B340FF9DFE2F6D
uid  [desconocida] User name <my.email@email.com>
ssb  rsa4096/1C0830E1D03AB 2017-08-09 [E] [caduca: 2033-08-05]
```

6. Tell git about your GPG key, run:

```
git config --global user.signingkey B344E73DA95715F4
```

Also run the following commands to sign all commits by default in your current local repository:

```
git config --global commit.gpgsign true
```

- Only for Windows:

```
git config --global gpg.program "C:\Program Files
(x86)\GnuPG\bin\gpg.exe"
```

I already have a GPG key on Keybase

1. Sign in from a browser to your Keybase Account.
2. Click on the ID of your PGP key.
3. Copy and paste the command to import your public GPG key on the terminal:

```
curl https://keybase.io/MY_USER_NAME/pgp_keys.asc | gpg --import
```

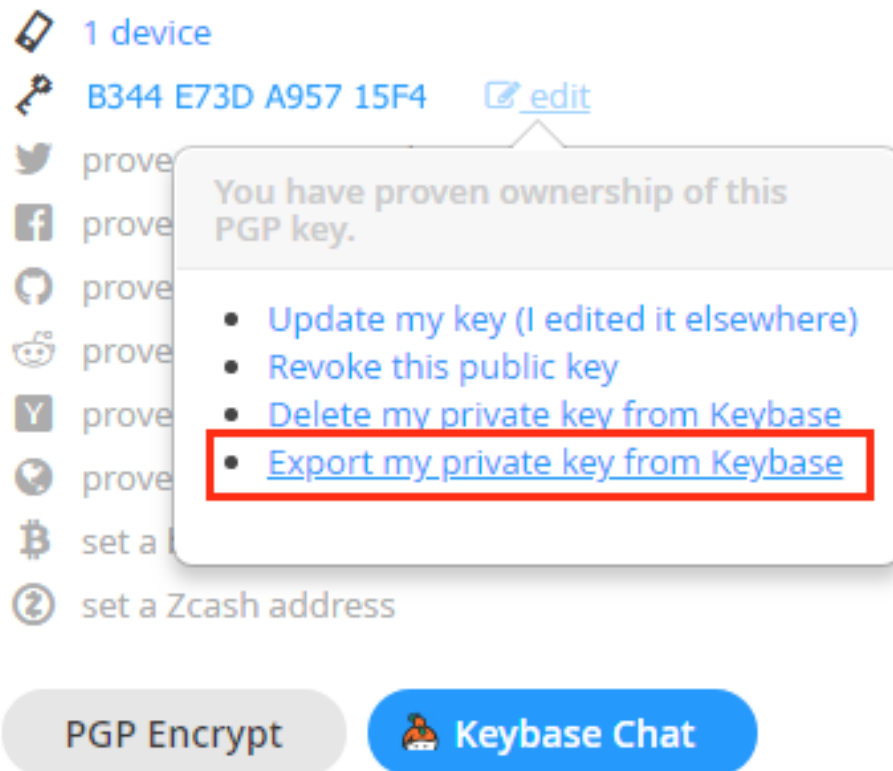
4. Add it to your GitHub account, go to settings > SSH and GPG keys > New GPG key

```
-----BEGIN PGP PUBLIC KEY BLOCK-----  
-----END PGP PUBLIC KEY BLOCK-----
```

5. Import your private key to your PC:

- 5.1. Go to your keybase account on your browser

- 5.2. Next to your key ID, click on edit and select export private key



- 5.3. Copy and paste your private key in a txt editor, and save it with the name `private.key`

- On Windows make sure it is on your user folder.

- 5.4. Go to command line and run:

```
gpg --import private.key
```

- 5.5. Check the key was imported by running:

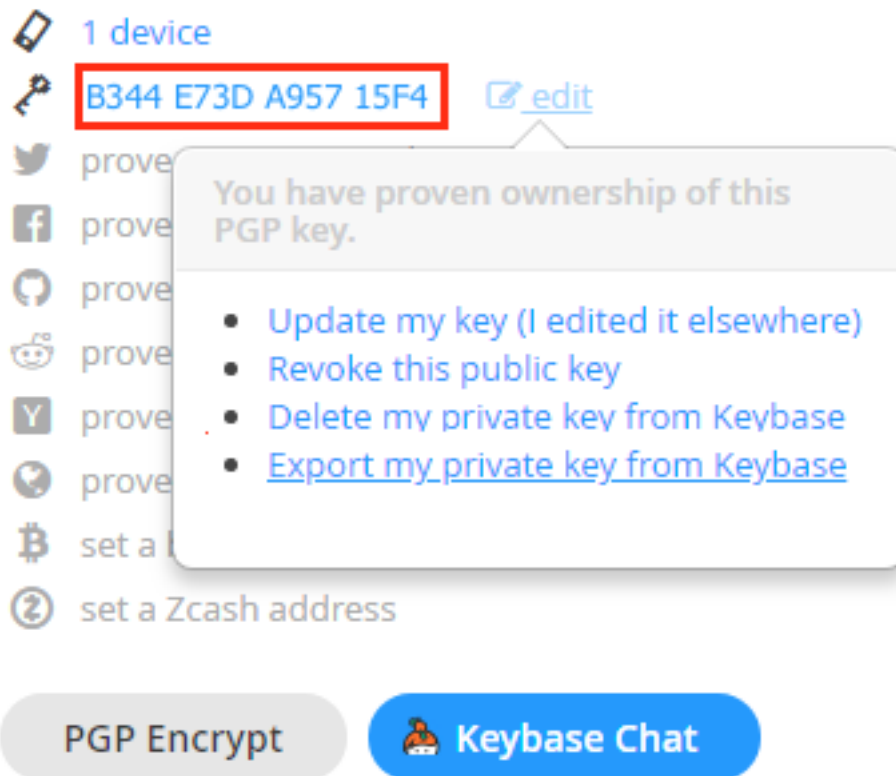
```
gpg --list-secret-keys --keyid-format LONG
```

Here should be listed your key, check the ID from keybase is the same on the sec line.

```

➔ ~ gpg --list-secret-keys --keyid-format LONG
/Users/mac5/.gnupg/pubring.gpg
-----
sec  rsa4096/B344E73DA95715F4 2017-08-09 [SC] [caduca: 2033-08-05]
     B50ADC2811C5819FE6A20D67B340FF9DFE2F6D
uid                          [desconocida] User name <my.email@email.com>
ssb  rsa4096/1C0830E1D03AB 2017-08-09 [E] [caduca: 2033-08-05]

```



6. Tell git about your GPG key, run:

```
git config --global user.signingkey B344E73DA95715F4
```

Also run the following commands to sign all commits by default in your current local repository:

```
git config --global commit.gpgsign true
```

- Only for Windows:

```
git config --global gpg.program "C:\Program Files
(x86)\GnuPG\bin\gpg.exe"
```

Test it!

1. With your Visual Studio Code open the folder of your cloned repo
2. Open the Readme and add a new line to it, for example: Hello World

3. Save changes and open the terminal on VS Code:

- On Windows use: `ctrl + ñ`
- On OSX use: ```

4. Run: `git add . && git commit -s -m "my first commit" && git push`

5. Now go to your repo and click on commits, you should see the Verified label.



• Git explanation:

- `git add .` -> Adds the changes to be committed
- `git commit -s -m "message"` -> commits the changes, -s is for signing the commit and -m “message”, the message describing the changes
- `git push` -> pushes the local changes to your remote repo (the repo on GitHub)

Tip: Don't forget you can learn more about these git commands with the [Pro Git book](#), available in several languages, and [Try Git](#), tutorial.

Note: Make sure that the email address in git, the Primary email in your GitHub account and the one in your GPG key are all the same.

In case you want to add an email account to your GPG, follow this guide [Associating an email account with your GPG key](#)

After adding your email account, remember to update your GPG on keybase, to do that, follow these steps:

1. Sign in from a browser to your Keybase Account.
2. Next to your key ID, click on edit and select *Update my key (I edited it elsewhere)*.
3. Run again `gpg --armor --export B344E73DA95715F4`
4. Copy the output and paste it where indicated in keybase.

1.11.3 Now that everything is set up

From the [Contributing Guidelines](#) you'll learn how to make an issue, a PR with your contribution, using the [Git Flow](#), [Conventional Commits](#) and the tools we implement.

Attention: The guidelines are mandatory for Contributors, Maintainers and Collaborators.

1.11.4 Inside our repos

Branches

We use the [git flow branching model](#).

We have three main branches in every repository, each one has a specific function:

- Our `master` branch is for the Long Term Support version.
- The `develop` branch is for our Bleeding edge version, and finally
- The `gh-pages` branch, where the project site is build using [Jekyll](#).

Continuous Integration

To speed things up and as part of our quality control, we count with automatic processes to run tests for the Pull Requests in our projects and to generate/update the information in the Development section on our `gh-pages` branches.

Merges in `Develop` branch

Every time a merge is made in `develop` the following documentation is generated or updated.

- Code Documentation: the description of every class and function of the source code.
- Coverage: the coverage of the code according to the tests.
- Test Report: the report thrown after all the tests are run.

Merges in `Master` branch

But when the merge is in `master`:

- Changelog: the list of commits that either add a fix or a feture, since the last stable version released.
- Releases: create a GitHub release.

Important:

- We follow the [Standard version](#) for changelogs and the [Semantic versioning](#) for the releases.
 - According to the projects they might have other deployment tasks besides the ones previously described, you should read the `README.md` in every `ci` directory to know their specific workflow.
-

1.12 Translations

1.12.1 I know a lot of languages

Great! We are currently translating to Korean, Spanish, Catalan, French, Portuguese, Russian, Japanese and more.

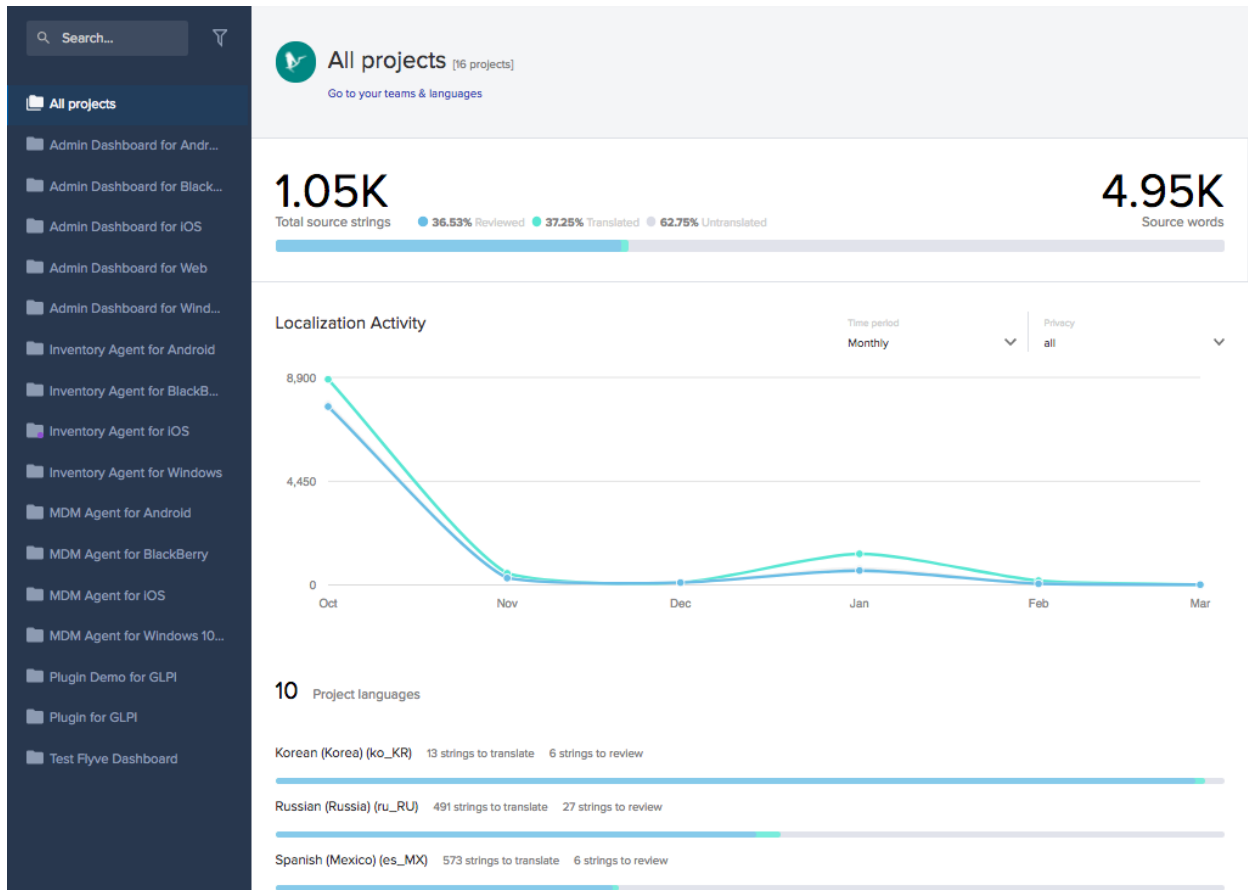
1.12.2 But wait! I'm an expert in a different language

If you are an expert in a language we are not currently translating, you can request this translation in Transifex and help us to reach more customers in their native language.

1.12.3 Let's Translate

We use [Transifex](#) to translate our projects. To start contributing with any of our projects' translations you require an account and to request to join the team of Flyve MDM.

After you are accepted you can now go to the Flyve MDM organization where you'll see all the projects listed to your left and a graphic with the statistics of Source Strings translations.



Under the graphic you can see all the languages in which the projects are being translated to, select the one you will translate, and click on View Projects.

10 Project languages

| | | | |
|-----------------------------|--------------------------|----------------------|-------------------------------|
| Korean (Korea) (ko_KR) | 13 strings to translate | 6 strings to review | |
| Russian (Russia) (ru_RU) | 491 strings to translate | 27 strings to review | |
| Spanish (Mexico) (es_MX) | 573 strings to translate | 6 strings to review | View projects |
| Catalan (Spain) (ca_ES) | 442 strings to translate | 0 strings to review | |
| French (France) (fr_FR) | 737 strings to translate | 0 strings to review | |
| Spanish (Spain) (es_ES) | 740 strings to translate | 1 string to review | |
| Japanese (ja) | 573 strings to translate | 15 strings to review | |
| Portuguese (Brazil) (pt_BR) | 805 strings to translate | 0 strings to review | |

















[show all languages](#)

All the projects under translation in that language will be displayed, you will also be able to see how many strings are required to translate. Select a project.

8 Projects

| Name | Progress | Last Update | |
|-----------------------------|-------------------------------|---------------------|-------------------|
| Plugin for GLPI | <div><div></div></div> 99.17% | 🕒 Oct 18th, 2:46 pm | ➤ |
| Admin Dashboard for Android | <div><div></div></div> 33.33% | 🕒 Nov 21st, 5:07 pm | ➤ |
| Inventory Agent for Android | <div><div></div></div> 33.2% | 🕒 Mar 26th, 2:55 pm | ➤ |
| Admin Dashboard for IOS | <div><div></div></div> 0% | 🕒 Oct 18th, 3:16 pm | ➤ |
| Admin Dashboard for Web | <div><div></div></div> 0% | 🕒 Feb 18th, 7:10 pm | ➤ |
| Inventory Agent for IOS | <div><div></div></div> 0% | 🕒 Feb 7th, 7:57 pm | ➤ |
| MDM Agent for IOS | <div><div></div></div> 0% | 🕒 Oct 18th, 3:11 pm | ➤ |
| Test Flyve Dashboard | <div><div></div></div> 0% | 🕒 Feb 7th, 8:22 pm | ➤ |

A list with the files requiring translations will be displayed, click on one of them.

| Name | Category | Progress | Last Update |
|--|---------------|-------------------------------|---|
|  strings_pref_accueil.xml | Uncategorized | <div><div></div></div> 100% |  Oct 17th, 5:35 pm |
|  strings_pref_global.xml | Uncategorized | <div><div></div></div> 100% |  Oct 17th, 5:37 pm |
|  strings_pref_inventory.xml | Uncategorized | <div><div></div></div> 100% |  Oct 17th, 5:38 pm |
|  value_time.xml | Uncategorized | <div><div></div></div> 100% |  Oct 17th, 5:39 pm |
|  strings.xml | Uncategorized | <div><div></div></div> 97.14% |  Oct 17th, 5:38 pm |
|  stringsxml-android | Uncategorized | <div><div></div></div> 2.05% |  Mar 26th, 2:55 pm |
|  strings_pref_advanced.xml | Uncategorized | <div><div></div></div> 0% |  Sep 5th, 3:15 pm |
|  strings_pref_auth.xml | Uncategorized | <div><div></div></div> 0% |  Sep 5th, 3:15 pm |

A modal window will appear, click on Translate.

Resource: [stringsxml-android](#) > (Spanish (Mexico))

437

total words

146

total strings

Translate

0 words reviewed

3 words translated

434 words untranslated

Download file to translate

View strings online


Download for use ?

View language stats

Download only reviewed translations ?

Delete translations

Upload file

 Translation last updated by [Rafa Hernández](#) 1 day, 5 hours ago

You'll see all the words in the file, you can filter them by clicking on Untranslated.

| 146 All | | 141 Untranslated | 5 Unreviewed |
|---|--------------------------------|---------------------|--------------------------|
| <input type="text"/> <input type="button" value="x"/> <input type="button" value=""/> | | | |
| 1 | Inventory | Inventario | <input type="checkbox"/> |
| 2 | Inventory Agent is started | | <input type="checkbox"/> |
| 3 | Inventory Agent is stopped | | <input type="checkbox"/> |
| 4 | Inventory Agent is restarted | | <input type="checkbox"/> |
| 5 | Agent is running in background | | <input type="checkbox"/> |
| 6 | Start Inventory | | <input type="checkbox"/> |

Select the word to translate, in the box next to it, where indicated, add your translation, save the translation and you're done.

Thank you for your contribution!

For more information about Transifex checkout their [Help Center Documentation](#).

1.13 Amazon Web Services

Amazon Web Services is a cloud services platform that offers a wide range of functionality to business, providing computing resources to host from simple web pages to more elaborated applications, data storage and more.

To start using GLPI Network from the AWS, follow these steps to create your account and launch the instance.

1.13.1 1. Create an account

1. Go to [Amazon Web Services](#).
2. Click on create account, it will redirect you to another page, click again on create a new account.
3. Fill your information
 - 3.1 Email, password

Create an AWS account

Email address

Password

Confirm password

AWS account name ⓘ

[Continue](#)

[Sign in to an existing AWS account](#)

© 2018 Amazon Web Services, Inc. or its affiliates.
All rights reserved.
[Privacy Policy](#) | [Terms of Use](#)

3.2 Contact information

Contact Information

All fields are required.

Please select the account type and complete the fields below with your contact details.

Account type ⓘ

☒ Professional ☐ Personal

Full name

Company name

Phone number

Country/Region

United States ▾

Address

Street, P.O. Box, Company Name, c/o

Apartment, suite, unit, building, floor, etc.

City

State / Province or region

Postal code

☐ Check here to indicate that you have read and agree to the terms of the [AWS Customer Agreement](#)

3.3 Payment information

Payment Information

Please type your payment information so we can verify your identity. We will not charge you unless your usage exceeds the [AWS Free Tier Limits](#). Review [frequently asked questions](#) for more information.

Credit/Debit card number

Expiration date

10 2018

Cardholder's name

Billing address

☒ Use my contact address

My Address

☐ Use a new address

Secure Submit

Note: If you create an account for the first time, you'll have 12 months of Free Tier Access. No charges will be made to your credit or debit cards as long as you don't pass the limits of the [AWS free tier](#).

4. Account verification, you'll receive a phone call asking you to introduce a code through your phone's keypad.
5. Choose a support plan. For more information, see the [Compare AWS Plan Support](#).
6. Once your account has been activated you'll receive a confirmation email, after this you'll have full access to the Amazon Web Services.

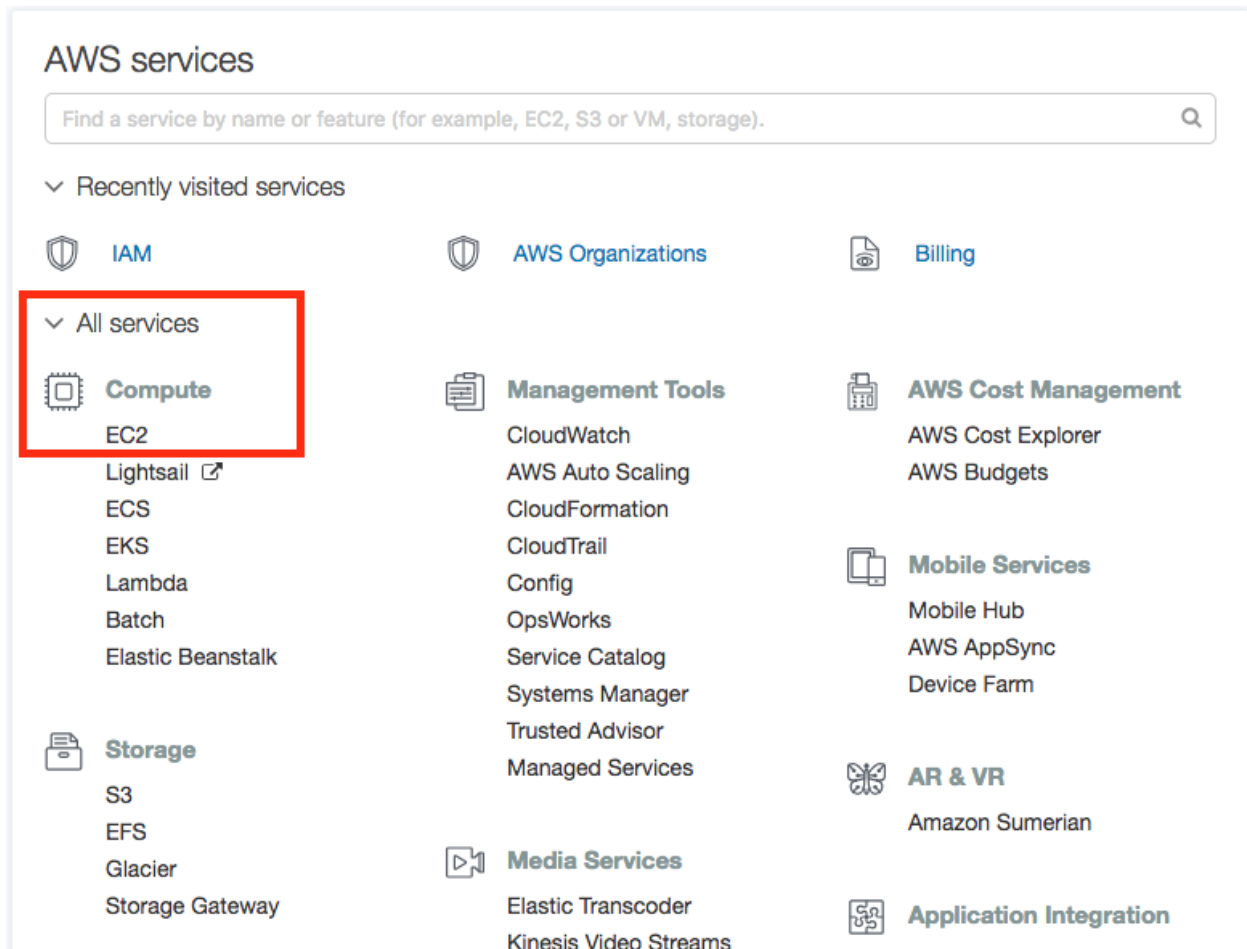
1.13.2 2. Create a Key pair

To access and launch EC2 instances, you'll need a SSH key.

Attention: Save it in a safe place! This key will be displayed to you only once, and it's required to connect to AWS servers and EC2 Linux instances, which means your GLPI.

1. Sign in to your AWS account, go to [Amazon Web Services](#)
2. After signing in, you'll be redirected to your [AWS console](#).

3. Display all services and select the EC2 service.



4. Under Network & Security, select Key pairs
5. Click on Create Key Pair
6. You'll be prompted for a name
7. A public and private key will be generated, download the private key, remember to **save it in a safe place**.

1.13.3 3. Deploy GLPI

1. Sign in to your AWS Marketplace account (the same credentials of your AWS account), go to [AWS Marketplace](#)
2. Search GLPI Network, provided by Teclib', there you'll be able to view the product overview, pricing, usage, support and reviews.
3. Click on Continue to Subscribe.

GLPI Network Basic
By: **TECLIB** Latest Version: 9.3.1-GNTK02-BASIC

GLPI Network is a subscription based on GLPI project with support from the editor and a network of local partners helping you to get the better of GLPI.

Linux/Unix ☆☆☆☆☆ (0) [Free Trial](#)

[Continue to Subscribe](#)

[Save to List](#)

Typical Total Price
\$0.297/hr
Total pricing per instance for services hosted on m3.medium in US East (N. Virginia). [View Details](#)

[Overview](#) [Pricing](#) [Usage](#) [Support](#) [Reviews](#)

4. You'll be able to review and accept the Terms and conditions along with the pricing information.
5. After accepting the Terms and subscribing, click on Continue to configuration.
6. Review the configuration of the instance, make sure the SSH key pair is the same as the one you created previously, click on Launch.
7. After your GLPI instance has been launched on the server, it'll have a public DNS. You can retrieve it from the EC2 Dashboard.

aws Services Resource Groups

EC2 Dashboard Events Tags Reports Limits

INSTANCES **Instances** Launch Templates Spot Requests Reserved Instances Dedicated Hosts Scheduled Instances

IMAGES AMIs Bundle Tasks

ELASTIC BLOCK STORE Volumes Snapshots Lifecycle Manager

NETWORK & SECURITY Security Groups Elastic IPs

[Launch Instance](#) [Connect](#) [Actions](#)

Filter by tags and attributes or search by keyword

| Name | Instance ID | Instance Type | Availability Zone | Instance State | Status Checks | Alarm Status |
|--------------------|-------------|---------------|-------------------|----------------|----------------|--------------|
| GLPI Network 9.3.0 | | t2.micro | us-east-1a | running | 2/2 checks ... | None |

Public DNS informations

Instance: **GLPI Network 9.3.0** **Public DNS: ec2-XXX-XX-XXX-X.compute-1.amazonaws.com**

Description **Status Checks** **Monitoring** **Tags**

| | | |
|-------------------|-----------------------|--|
| Instance ID | Public DNS (IPv4) | ec2-XXX-XX-XXX-X.compute-1.amazonaws.com |
| Instance state | IPv4 Public IP | |
| Instance type | IPv6 IPs | - |
| Elastic IPs | Private DNS | ip-172-31-94-121.ec2.internal |
| Availability zone | Private IPs | 172.31.94.121 |
| Security groups | Secondary private IPs | |
| Scheduled events | VPC ID | vpc- |
| AMI ID | Subnet ID | subnet- |
| Platform | Network interfaces | eth0 |

8. Connect through SSH, for this you'll need the username "ubuntu" and the public DNS:

```
ssh ubuntu@ec2-XXX-XX-XXX-X.compute-1.amazonaws.com
```

Type yes when asked to continue connecting.

9. Use the following command to retrieve the login credentials.

```
sudo cat /opt/teclib/glpi-network-scripts/.glpi-network-credentials
```

```
$ ssh ubuntu@██████████.compute-1.amazonaws.com
The authenticity of host '██████████.compute-1.amazonaws.com (██████████)' can't be established.
ECDSA key fingerprint is SHA256:██████████.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '██████████.compute-1.amazonaws.com,██████████' (ECDSA) to the list of known hosts.
-----
  Ubuntu 16.04.5 LTS                built 2018-08-22
-----

  -----
  |--  --|  ---/  ---|  |  |  --  |  _ \ ( )
  | |  |  |--  |  |  |  |  |  |  |_) | /
  | |  |  --  |  |  |  |  |  |  |  <
  | |  |  ---|  ---|  |---|  |  |  |_) |
  |--  |  ---\---|---|---|---/

  GLPI Network Distribution
  (based on GLPI Project)

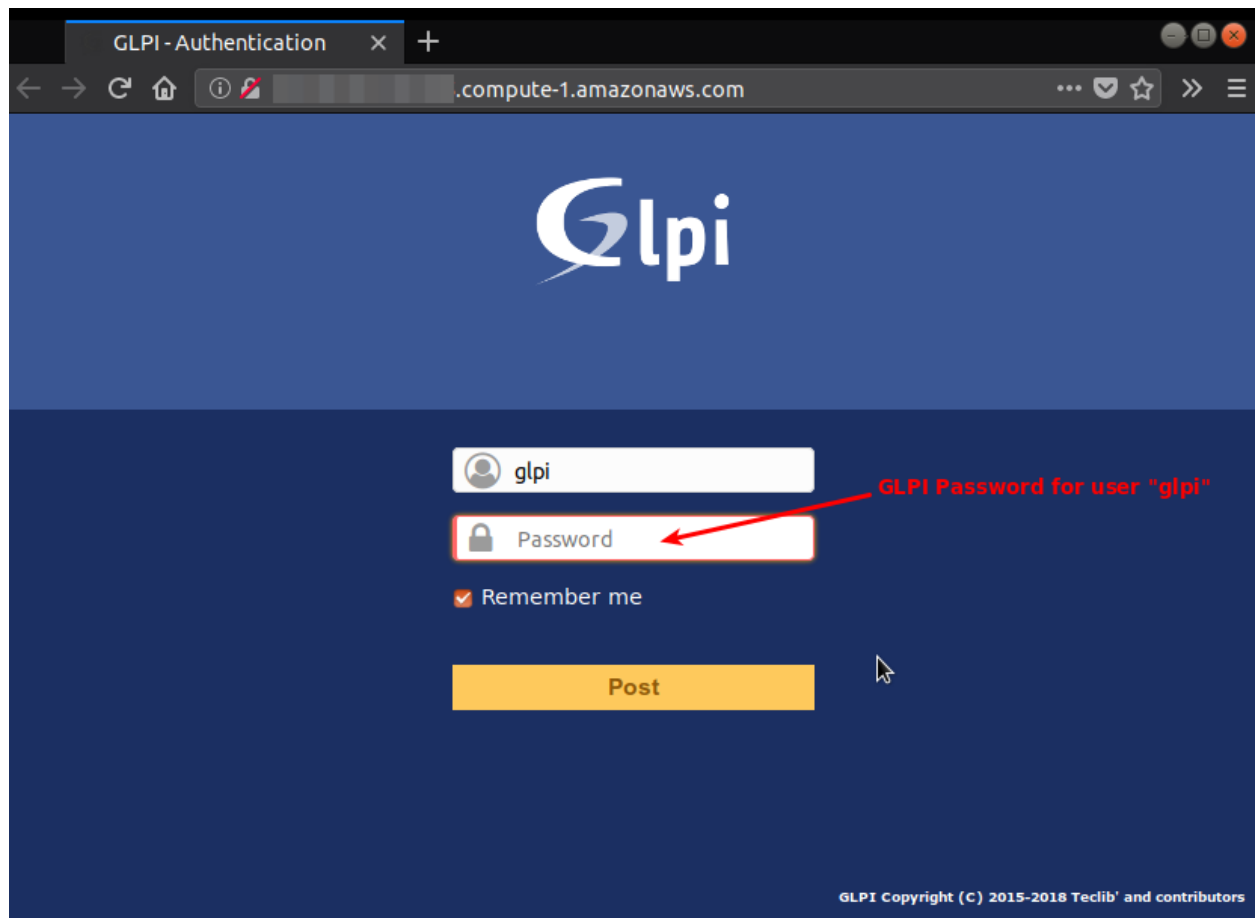
-----
* Documentation:  https://help.glpi-network.com
* Support:       https://services.glpi-network.com
* Community:     https://glpi-project.org/community/
-----

0 packages can be updated.
0 updates are security updates.

ubuntu@ip-██████████:~$ sudo cat /opt/teclib/glpi-network-scripts/.glpi-network-credentials
MYSQL_ROOT_PASS=YzZmNzBmZTYzZTU1
MYSQL_GLPI_PASS=ZWlYNDdiYTRhMTgz
GLPI_PASSWORD=MDE5ODhj
ubuntu@ip-██████████:~$
```

Attention: We strongly recommend that you retrieve and record this information outside the deployed instance and then delete the file.

10. Access the application via web browser at `http://<EC2_Instance_Public_DNS>`
11. Log in with the username “glpi” account and the password you retrieved in step 9



Note: For more information you can also check the [AWS Documentation](#) regarding EC2

1.14 Cloud Services

1.14.1 GLPI Network

GLPI Network is a subscription service for companies and organizations, it includes support from the developers and access to the local partners in your own language.

You'll have access to exclusive benefits such as new features request, additional services and plugin support according to your subscription.

1.14.2 1. Register

1. Go to [GLPI Network Register form](#)
2. Fill your information:
 - 2.1 Name, Email, Password

Register

Language

English

Name

E-Mail

Password

Confirm Password

The password must have a minimum length of 8 and follow at least 3 of these rules:

- *uppercase characters (A – Z)*
- *lowercase characters (a – z)*
- *digits (0 – 9)*
- *non-alphanumeric (!, \$, #, or %)*

Register

3. Click on My subscription, choose the Standard plan which brings you Flyve MDM

The screenshot shows the 'My subscription' page in the GLPI Network interface. On the left is a dark blue sidebar with links for 'My subscription' and 'Contact us'. The main content area is titled 'My subscription' and displays two subscription plans side-by-side: 'Standard' and 'Advanced'.

| Standard | Advanced |
|----------------------------|----------------------------|
| Unlimited Tickets ? | Unlimited Tickets ? |
| 12 month commitment | 12 month commitment |
| GLPI last major version | GLPI last major version |
| 2 supported instances ? | 4 supported instances ? |
| Additional services ? | Additional services ? |
| plugins ? | plugins ? |
| Support standard plugins ? | Support standard plugins ? |
| | Support advanced plugins ? |
| € 300.00/month(s) | € 1 000.00/month(s) |
| CHOOSE | CHOOSE |

4. Fill your billing information, then click on Continue

Gipi Network

My subscription

Contact us

My subscription

Standard

Billing information

| | |
|------------------|-----------|
| First name | Last name |
| Company | E-mail |
| Address (Line 1) | |
| Address (line 2) | |
| Zip code | City |
| Country | |
| France | |
| Tax/VAT number | |
| Phone number | |

CONTINUE

Any questions, issues or suggestions ?
Please contact Customer Care:
contact@novafirstcloud.com
+34 616 214 848
<https://services.gipi-network.com/>

5. Fill your payment information, then click on Validate

The screenshot shows the 'My subscription' page on the GLPI Network website. The page has a dark blue sidebar with 'My subscription' and 'Contact us' links. The main content area is titled 'My subscription' and features a 'Standard' subscription plan. A 'Payment information' section is highlighted, showing a 'Subscription cost' table with a 'Total' of € 300.00. Below this, there is a 'Fill-in your payment information' section with fields for E-mail, Card holder name, Card number, and Expiration date. A 'VALIDATE' button is at the bottom of the form. The page also includes logos for Stripe, Mastercard, and Visa, and a 'SSL CERTIFIED' badge. At the bottom, there is contact information for Customer Care: contact@novafirstcloud.com, +34 616 214 848, and https://services.glpi-network.com/.

Welcome to Flyve MDM, you're now subscribed!

Now you can create tickets to get support.

1.15 Demo Trial

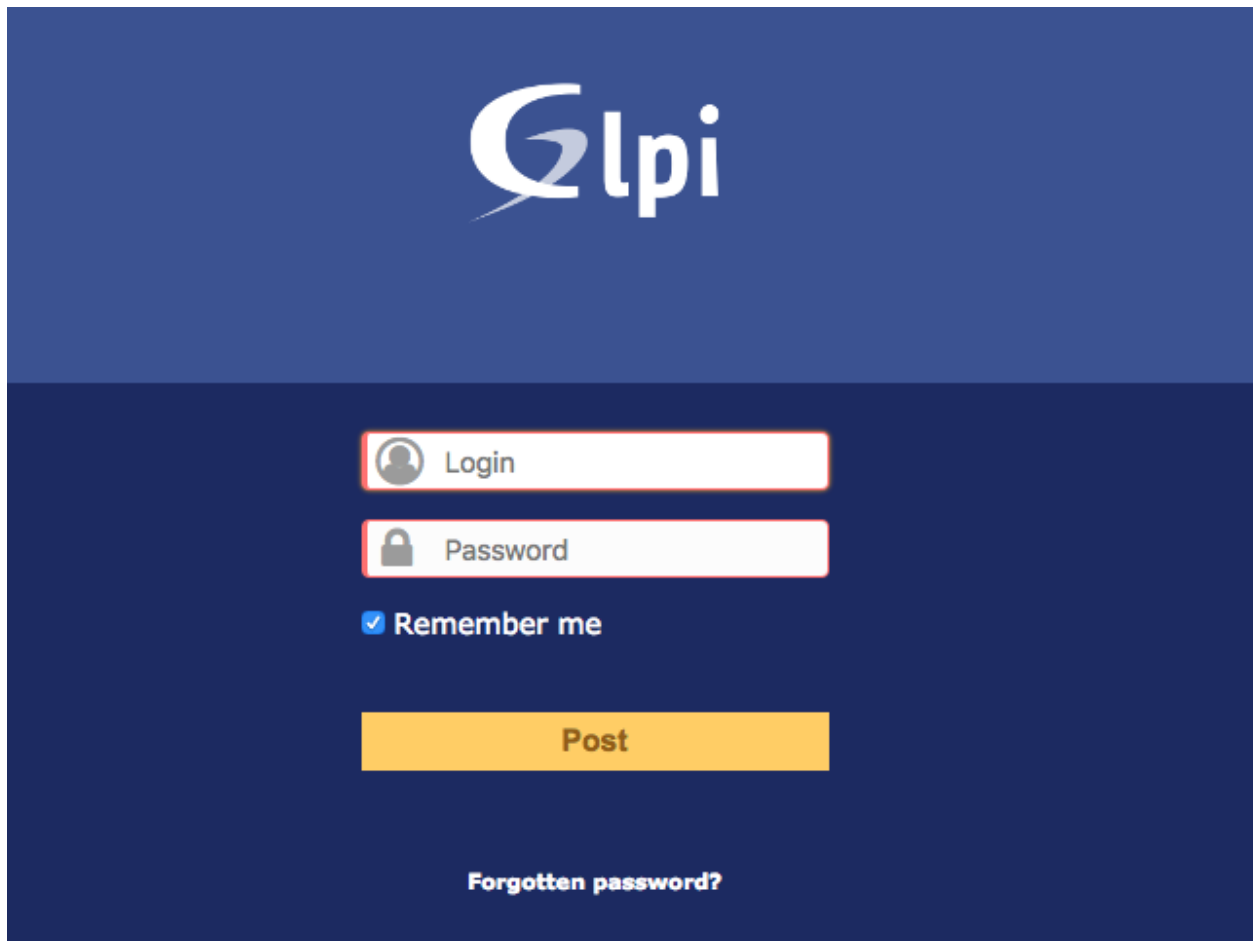
Flyve MDM offers a website for demo purposes for both the GLPI plugin and the Web MDM Dashboard.

To access any of them:

1. Request your credentials through our [Website Flyve MDM](#)
2. Click on **Ask for Demo**
3. Fill the formulaire
4. You'll receive your credentials in your email.

You will now be able to enter our Demo sites and start using Flyve MDM

- To try Flyve MDM plugin for GLPI, go to this link <https://demo-api.flyve.org/>



The image shows a login interface for Glpfi. It features a dark blue background with the Glpfi logo at the top. Below the logo, there are two input fields: one for 'Login' with a user icon and one for 'Password' with a lock icon. A 'Remember me' checkbox is located below the password field. A yellow 'Post' button is positioned below the checkbox. At the bottom, there is a link for 'Forgotten password?'.

Glpfi

Login

Password

☒ Remember me

Post

Forgotten password?

- To try Flyve MDM Web Dashboard, go to this link <https://demo-dashboard.flyve.org/>



Sign in

Use your Flyve MDM account
[What's this?](#)

Next

[Terms and Conditions](#)

© 2017 - 2018 Teclib'.

English -Britain



Note: You can access both demos with the same credentials.
