

Upgrading GreenLight - only for >=3.5.x

Starting with v3.5.3 we have changed the upgrade procedure in order to have the same process in place across all panagenda products. From now on you have to perform the update on the appliance/console level.



This new upgrade procedure is only valid for 3.5.x (NOT for 3.2.x)

Two Types

There are two types of upgrade files available:

- "gl-upgrade.zip" file (~ 200 MB) (GreenLight appliance need to access docker.panagenda.com)
- "gl-upgrade-offline.zip" (~ 1200 MB), (GreenLight appliance does NOT need access to docker.panagenda.com).

Procedure

- Log off from the UI of GreenLight
- Upload the GreenLight upgrade package (gl-upgrade.zip or gl-upgrade-offline.zip) to the appliances **/tmp/** directory (using WinSCP for example)
- Use ssh or putty to connect to the appliance and execute the following commands:

ONLINE Upgrade

```
cd /tmp
unzip gl-upgrade.zip
./gl-upgrade.sh
```

or

OFFLINE Upgrade

```
cd /tmp
unzip gl-upgrade-offline.zip
./gl-upgrade-offline.sh
```

- Whenever done, it takes 2-5 minutes until all Docker images are up and running again
- Reload the GL UI (Browser) and Login



If the upgrade process sticks for several minutes while the system tries to stop all containers, ...

```
gathering host information
Installer is running on a panagenda appliance
Upgrading GreenLight version from 3.5.3.12739 to 4.0.0.13248
Creating Metabase / cqx user roles if not exists
Shutting down current Docker containers
/usr/lib/python2.7/site-packages/requests/__init__.py:80: Request
  RequestsDependencyWarning)
Stopping gl_nginx    ... done
Stopping gl_node     ... done
Stopping gl_cron     ... done
Stopping gl_tomcat   ...
Stopping gl_postgres ...
```

...make sure you have closed all webbrowser connections to GL

Example Output

The following screenshot shows the output during a gl-upgrade-offline update.

```

[root@ tmp]# ./gl-upgrade-offline.sh
Verifying archive integrity... 100% MD5 checksums are OK. All good.
Uncompressing GreenLight by panagenda 100%

The log file can be found here: /opt/panagenda/logs/greenlight-upgrade_2019-02-01_122739.log

Gathering host information
Installer is running on a panagenda appliance
The file /opt/panagenda/appdata/VERSION does not exist. Defaulting to version 0.0.0
Installing GreenLight version 3.5.3.12739
Shutting down current Docker containers
Stopping gl_nginx ... done
Stopping gl_node ... done
Stopping gl_tomcat ... done
Stopping gl_cron ... done
Stopping gl_postgres ... done
Removing gl_nginx ... done
Removing gl_node ... done
Removing gl_tomcat ... done
Removing gl_cron ... done
Removing gl_postgres ... done
Removing network gl_panagenda_network
Installing latest Docker images
Extracting Docker images from the images.tar.gz file
images.tar
Loading Docker images from the images.tar file
Loaded image: docker.panagenda.com/ramkin:1.1.0
Removing existing docker.panagenda.com/ramkin images
Untagged: docker.panagenda.com/ramkin:1.0.2
Untagged: docker.panagenda.com/ramkin@sha256:ad04bf767198f9d7516d7031d318c6aa28ddaf087aacc7dcf25c3ff43eb2286
Deleted: sha256:b9f02cdef0f52690f3da814a8ff07a8f576cd161bb633860fae304766931dc06
Deleted: sha256:cddeb536e0b60091a867d14ee86e833e15a93f82feb743289e3d98fld3b33598
Deleted: sha256:f4a954545e3dd0cb7a169167858ec51a6de576fe3220720eb9b0cbcc8bdb2d64
Deleted: sha256:bccced05447cf2aeee712874bafa4202ccfclb0962e0497f993a38bd0bc13632
Deleted: sha256:865b4fedc73da8729a0af8d63ca82f891f7b5a625414dbf44fec4687182086b0
Deleted: sha256:e112a2b84ce8197a3831a7cf3f5a0ecec19e4a0b107f7348bfeb36a0cbaebe7ee
Deleted: sha256:8a478341b277d0abdc50314836fd9ccc7d68ed9f54621988361fle19b02e9965
Updating version file
Updating GreenLight directory
Updating Tomcat webapp
Creating new Docker containers
Stopping current GreenLight containers
Removing current GreenLight containers
Removing network gl_panagenda_network
Network gl_panagenda_network not found.
Creating gl_postgres ... done
Restarting Docker Daemon
Creating gl_tomcat ... done
Creating gl_node ... done
Creating gl_nginx ... done
Creating gl_cron ...
Creating gl_tomcat ...
Creating gl_node ...
Creating gl_nginx ...
Done

```