Migration from v3.2 to v3.5

GreenLight v3.5. comes with a new Platform Architecture.
Because of this massive change it is necessary that you migrate the data from the old GreenLight Appliance towards the new GreenLight Appliance (side by side migration) in case you want to keep the existing config and historical data

The following process is pretty straight forward and represents a step by step guide:

The following Data will be migrated:

- SQL Database (config + historical data)
- Notes Data (user.id and names.nsf)
- License file
- All Scripts you might have under

> /opt/panagenda/scripts/gl_sctions
> /opt/panagenda/scripts/gl_powershell
> /opt/panagenda/scripts/gl_sensor

Requirements

- **First of all , please make sure that you have enough disk space available on the new Appliance!**

  /opt/panagenda/pgdata (new appliance) should be as big as /var/lib/postgresql (old appliance)

- **Make sure that the old Appliance is running with the latest GreenLight code - v3.2.6 !**

- **Make sure that TCP Port 5432 is opened between both GreenLight Appliances!**
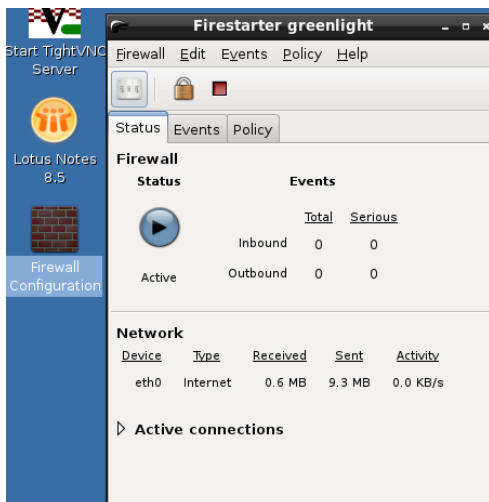
  How to verify?

  On **NEW** GreenLight Appliance:

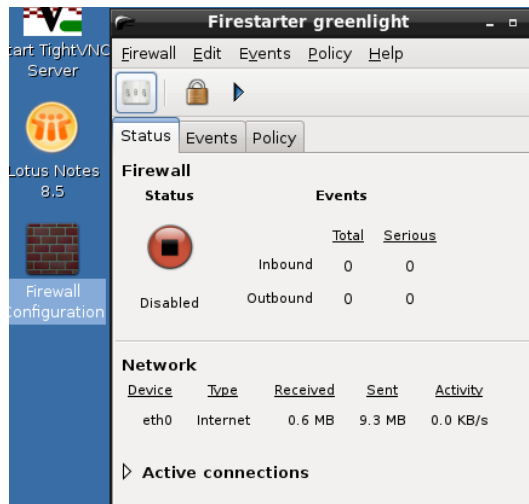 - open putty and enter the following command: telnet <old_greenlight_box> 5432

   If you get an error, please do the following:

  - open GUI on **OLD** greenLight Appliance
  - open firewall Settings
  - click on the STOP symbol (next to the lock)

 you should see now the stop button in the middle of this screen which indicates

that the FW has been stopped

How To MIGRATE the Data

- open putty and connect to **NEW** GreenLight Appliance
- Login as: root / config
- **enter:**

    ssh panagenda@<ip_OLD_GreenLight>

    enter "yes" to accept the fingerprint

    enter password: "panagenda"

    ...whenever you are connected, enter "exit"

- **open the folder (via putty)**

    /opt/panagenda/appdata/volumes/gl/scripts

```
[root@greenlight ~]# cd /opt/panagenda/appdata/volumes/gl/scripts/
```

- execute the following command (Note: if you have changed the password on the old GL appliance, please adjust the PWD in the migrate.sh script)

    ./migrate.sh <ip_OLD_GreenLight>

    example: ./migrate.sh 10.10.1.200

```
[root@greenlight scripts]# ./migrate.sh 192.168.81.151
```

You should see this kind of process: (this can take minutes/hours depending on the amount of data you have)

```
[root@greenlight scripts]# ./migrate.sh 192.168.81.151
info: Pulling files/data from 192.168.81.151
info: Checking installed GreenLight version
info: Currently installed version is greenlight-2:3.2.2.12374
info: Stopping GreenLight Docker containers before data integration
info: Stopping the Ubuntu Tomcat
 * Stopping Tomcat servlet engine tomcat7
   ...done.
info: Ubuntu Tomcat stopped
info: Copying files from 192.168.81.151
Connected to 192.168.81.151.
sftp>   get /opt/notes/notes-data/notes/data/user.id                    /opt/panagenda/appdata/v
Fetching /opt/notes/notes-data/notes/data/user.id to /opt/panagenda/appdata/volumes/notes/user.id
sftp>   get /usr/share/ibm/notes/data/names.nsf                    /opt/panagenda/appdata/volumes/n
Fetching /usr/share/ibm/notes/data/names.nsf to /opt/panagenda/appdata/volumes/notes/names.nsf
sftp>   get /opt/panagenda/greenLight2.lic                    /opt/panagenda/appdata/volumes/g
Fetching /opt/panagenda/greenLight2.lic to /opt/panagenda/appdata/volumes/gl/greenLight2.lic
sftp>   get -r /opt/panagenda/scripts/gl_action              /opt/panagenda/appdata/volumes/gl/scripts
Fetching /opt/panagenda/scripts/gl_action/ to /opt/panagenda/appdata/volumes/gl/scripts/gl_action
Retrieving /opt/panagenda/scripts/gl_action
sftp>   get -r /opt/panagenda/scripts/gl_sensor              /opt/panagenda/appdata/volumes/gl/scripts
Fetching /opt/panagenda/scripts/gl_sensor/ to /opt/panagenda/appdata/volumes/gl/scripts/gl_sensor
Retrieving /opt/panagenda/scripts/gl_sensor
sftp>   get -r /opt/panagenda/scripts/gl_powershell     /opt/panagenda/appdata/volumes/gl/scripts
Fetching /opt/panagenda/scripts/gl_powershell/ to /opt/panagenda/appdata/volumes/gl/scripts/gl_powershell
Retrieving /opt/panagenda/scripts/gl_powershell
info: Dropping current database
info: Dropped database successfully
info: Creating new database
info: Created new database successfully
info: Create .pgpass file
info: .pgpass file created
info: Creating database dump from Ubuntu appliance [192.168.81.151]
```

```
ALTER TABLE
ALTER TABLE
ALTER TABLE
ALTER TABLE
DELETE 2
UPDATE 2
DELETE 2
vacuum analyze
VACUUM
info: Starting the Ubuntu Tomcat
 * Starting Tomcat servlet engine tomcat7
   ...done.
info: Ubuntu Tomcat started
info: Migration done. Rebooting machine in 3 seconds...
```

Whenever this is done, the new GreenLight Appliance restarts automatically. **So the Core Migration is done and you can use the new Appliance.**

The OLD greenLight appliance is off (http process) and should be completely switched off.

OPTIONAL: Manual Tasks (after migration)

Likely that you have modified some items in the past. These elements needs to be manually touched.

for example:

- **modify/adjust ramkin-config.properties** (this file has NOT been migrated automatically!!!)

  /opt/panagenda/appdata/volumes/gl/ramkin-config.properties   (e.g. notes id password, timezones...)

- **Adjust your self develop scripts**

  If you have added Linux Shell Script Actions or Linux Shell Script Sensors or Custom PowerShell Scripts ...or even custom SQL code , please check and adjust them if necessary

  /opt/panagenda/appdata/volumes/gl/scripts/gl_action

  /opt/panagenda/appdata/volumes/gl/scripts/gl_sensor

  /opt/panagenda/appdata/volumes/gl/scripts/gl_powershell

  /opt/panagenda/appdata/volumes/gl/scripts/sql

- **Check script dependencies**

  put missing dependencies into /opt/panagenda/appdata/volumes/gl/ramkin/binaries

  put debian package files into /opt/panagenda/appdata/volumes/gl/ramkin/install

  example: certain DEB installations you need

  NOTE: nagios plugins , ldap utilities are part of the GL Image already. !

- Add custom scripts /commands to /opt/panagenda/appdata/volumes/gl/cron/crontab*