Qeedji

Legal notice

SBL10e generic_server 1.12.10 (001A_en)

© 2022 Qeedji

Rights and Responsibilities

Product information

Product design and specifications are subject to change at any time and `Qeedji` reserves the right to modify them without notice. This includes the hardware, the embedded software and this manual, which should be considered as a general guide to the product. The accessories supplied with the product may differ slightly from those described in this manual, depending on the developments of the various suppliers.

Precautions for use

Please read and heed the following warnings before turning on the power: - installation and maintenance must be carried out by professionals. - do not use the device near water. - do not place anything on top of the device, including liquids (beverages) or flammable materials (fabrics, paper). - do not expose the device to direct sunlight, near a heat source, or in a place susceptible to dust, vibration or shock.

Warranty clauses

The `Qeedji` device is guaranteed against material and manufacturing defects for a certain duration. Check the device warranty duration value at the end of the document. These warranty conditions do not apply if the failure is the result of improper use of the device, inappropriate maintenance, unauthorized modification, operation in an unspecified environment (see operating precautions at the beginning of the manual) or if the device has been damaged by shock or fall, incorrect operation, improper connection, lightning, insufficient protection against heat, humidity or frost.

WEEE Directive

This symbol means that your appliance at the end of its service life must not be disposed of with household waste, but must be taken to a collection point for waste electrical and electronic equipment or returned to your dealer. Your action will protect the environment. In this context, a collection and recycling system has been set up by the European Union.

Table of contents

Part I | Description and installation

1.1 Introduction

This manual explains how to install and configure your device SBL10e.

Recommendations and warnings

This device is designed for indoor use only.

To ensure better rendering of the SBL10e, the device should not be installed under direct sunlight.

The SBL10e device is designed to be illuminated 12 hours a day, 7 days a week.

Package Contents

Articles	Description
Device	SBL10e device with the default `regular` [1] application embedded.

[1] *It is possible to easily update the device with the* `generic_server` *application afterwards.*

> ☞ *In this documentation, the unit of measurement for dimensions is done in millimeters followed by its equivalent value in inches.*

1.1.1 Device dimensions

60,47 mm
(2,38")

18,20 mm
(0,70")

47,56 mm
(1.87")
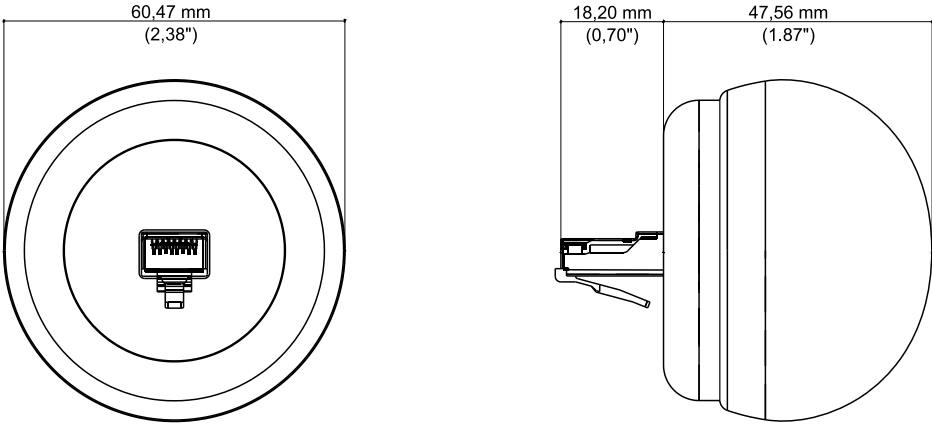
1.1.2 Labelling

Product label

The model of the device, the power supply characteristics, the serial number (PSN) and the MAC address are written on a label stuck on the case.
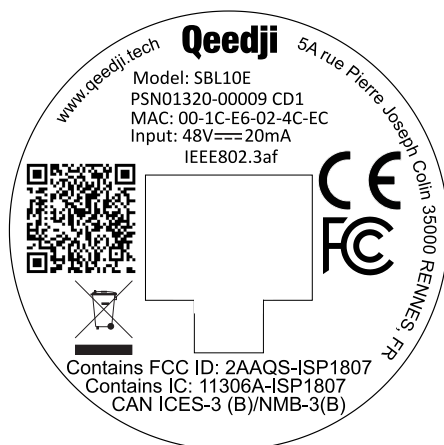
☞ *The QR code on the product label is corresponding to the product identification URL, for example:*
`i.qeedji.tech?model=SBL10e&sn=01320-00009&mac.lan1=00-1C-E6-02-4C-EC&mac.wpan1=DF-27-83-3C-8A-90` .

Packingbox label

This is the label stuck also on the packingbox. It is showing:

- the device model,
- the product serial number (PSN) (embedded also in the QR code),
- the manufacturer Website.
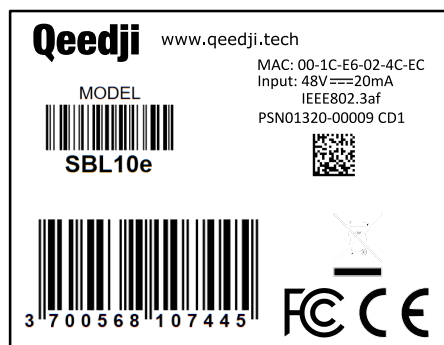
☞ *The QR code on the packingbox label is corresponding to the product PSN, for example:*
`PSN01320-00009 CD1` .

☞ *The serial number of the device could be requested in case of technical support.*

1.1.3 Installation

▌ ☞ *Install the SBL10e device on the Ethernet wall plugs of the buildings following the installation map given by your IT departement.*

The SBL10e device has to be plugged to an Ethernet wall plug supporting PoE IEEE802.3af.
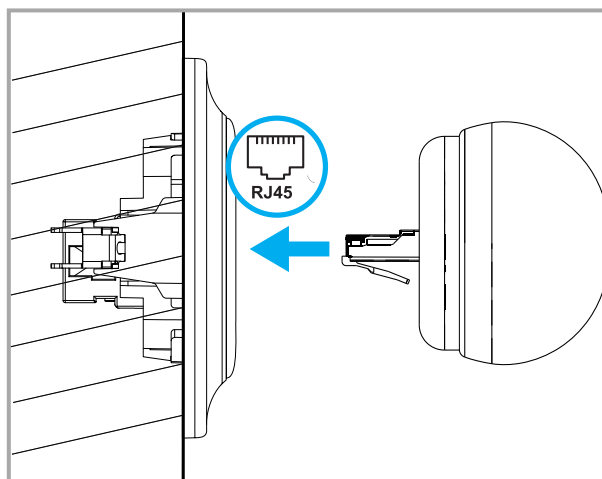
Given the device footprint, it is preconised to use Ethernet wall plug plastron with a right insertion.

Consequently, the Ethernet wall plug whose plastron is angled is not supported.

▌ ☞ *Thanks to the lock pin of its Ethernet connector, the SBL10e device can be installed on vertical surfaces, like walls as well as horizontal surfaces, like ceilings.*

1.1.4 Uninstallation

With a screw driver, hold down the pin of the Ethernet connector ① of the SBL10e device at the same time you are releasing with the hand ② the SBL10e device from the Ethernet wall plug.
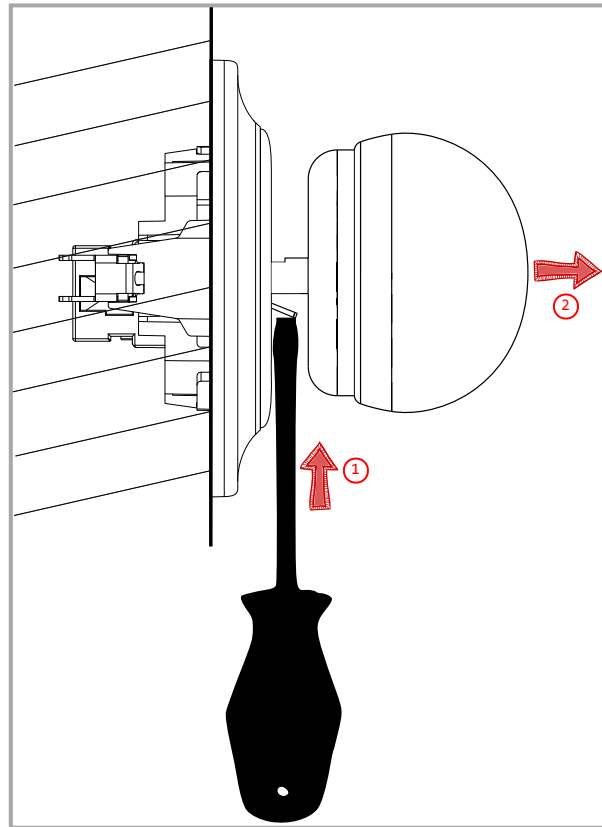
1.2 Smart Busy Light applications

To display the appropriate light state and color, the `generic_server` application periodically connects to a Web server supporting a database of the different state and color values for each device installed in the organization.

Server datasource

The server datasource must fill these requirements in terms of Web service and data format.

Web service:

This is the format of the HTTP request sent by application running on device to the datasource server to get its appropriate light state and color:

GET / `<URL>` *HTTP/1.0*
Host: `<server_address>`

with this URL syntax:
`<URL> = https://<domain>:<port>/<path_to_json_file_server>`

or with this URL syntax using a `{$deviceInfo:psn}` variable allowing to work with the PSN value of the device:

with `<URL> = https://<domain>:<port>/<path_to_json_file_server>/{$deviceInfo:psn}`

or with this URL syntax using a `$deviceInfo:mac` variable allowing to work with the MAC address value of the device:

with `<URL> = https://<domain>:<port>/<path_to_json_file_server>/{$deviceInfo:mac}`
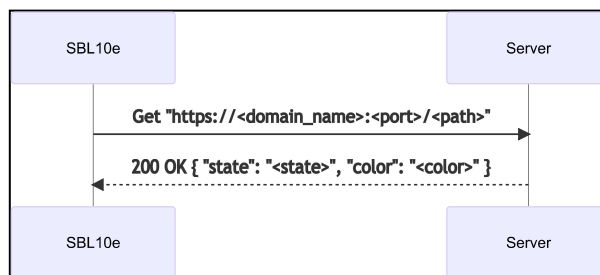
For example:
GET https://192.168.10.10:443/sbl10/00900-01504 *HTTP/1.0*
Host: `192.168.10.10`

Data format:

The body of the data returned for each device by the server datasource must be formatted in JSON format. The JSON body is built with the following way:

```
{ "state": "<state>", "color": "<color>" }
```

These are the supported values for each variable:

- `<state>` : *off*, *steady* or *flashing*,
- `<color>` : *red*, *orange*, *blue*, *yellow* or *green*

Example:

```
{ "state": "steady", "color": "red"}
```

Light states and colors

The device can support the states and colors values showed below.

Color	State
	`OFF`
`Red`	`ON` steady or `ON` flashing
`Green`	`ON` steady or `ON` flashing
`Blue`	`ON` steady or `ON` flashing
`Orange`	`ON` steady or `ON` flashing
`Yellow`	`ON` steady or `ON` flashing

> ☞ *The* `ON flashing` *state is flashing with this sequence:* `ON` *for 0,5 seconds then* `OFF` *for 0,5 seconds every one second.*

☞ *Depending on the application running on it, some color/state may be never used.*

☞ *The light color and state values are stored in the volatile memory (RAM). That means that in case the SBL10e device is unplugged from the Ethernet wall plug then plugged back again, the light comes back to its default state:* OFF *until its state is then modified by the App or by the user.*

Configuration

The `Smart Busy Light` application supports the configuration update:

- by connecting to the device configuration Web user interface `http://<device-ip-addr>/` and changing parameters,
- by pushing, from a WebDAV client or with the device Web user interface, a `prefs.json` configuration file on the device WebDAV directory `http://<device-ip-addr>/.conf/`,
- by pushing, from a WebDAV client or with the device Web user interface, a `.js` configuration script on the device WebDAV directory `http://<device-ip-addr>/.conf/`,
- by receiving a `configure` command with an appropriate `.js` configuration script from the `Qether` tool (Qether V1.12.10 or above).

Firmware upgrade

The `Smart Busy Light` application supports the firmware upgrade:

- by connecting to the device configuration Web user interface `http://<device-ip-addr>/` and loading an appropriate firmware file `bm0032_generic_server-sbl10e-xx.yy.zz.bin` [1] firmware file,
- by pushing a new `bm0032_generic_server-sbl10e-xx.yy.zz.bin` [1] firmware file at the root of the device WebDAV directory `http://<device-ip-addr>/` with a WebDAV client,
- by receiving an `install` command with an appropriate `bm0032_generic_server-sbl10e-xx.yy.zz.bin` [1] firmware file from the `Qether` tool.

[1] *Can work also with any other* `bm0032_<custom>-sbl10e-xx.yy.zz` *compatible firmware.*

> ☞ *After a firmware upgrade, the device is rebooting once.*

> ☞ *When the* `configuration` *command or the* `install` *command has been processed, the last* `Smart Busy Light` *state and color are restored.*

Preprogrammed flashing sequence

The SBL10e device has two modes:

- `Nominal mode` : the `Smart Busy Light` application runs properly and sets the light state and color as expected. When a configuration or a firmware upgrade is in progress, the light illumination can be temporarily inconsistent and follows the light flashing sequence shown in the table hereafter.
- `Recovery mode` : the `Smart Busy Light` application can not be executed. The light state or color can not be modified anymore. It is required to update the firmware to return to the `nominal mode`.

Depending on these modes, the `Smart Busy Light` applications can fall into one of these preprogrammed flashing sequences:

Mode	Smart Busy light behaviour	Information
Recovery	**2** very short and consecutive blue flashs (250 ms) with a 4,5 seconds periodicity	The Smart Busy Light application can not be executed. It should never happen. The device Web user interface is so not available. This sequence is displayed until a new firmware update is done with `Qether` tool. For further information, contact support@qeedji.tech.
Recovery	**3** very short and consecutive blue flashs (250 ms) with a 5 seconds periodicity	The software resource of the SBL10e device set at factory are not valid. It should never happen. For further information, contact support@qeedji.tech.
Nominal or recovery	**4** very short and consecutive blue flashs (250 ms) with a 5,5 seconds periodicity	A SBL10e device `Firmware update` is in progress. Please wait a couple of seconds.
Nominal	**5** very short and consecutive blue flashs (150 ms)	A SBL10e device `configuration` is in progress. Please wait a couple of seconds.
Nominal	**6** very short and consecutive blue flashs (150 ms)	The datasource is not consistent because some of the parameters are missing or have a wrong value. For further information about the datasource form, refer to the chapter § Configuration > Servers. For further information about the reporting of the problems faced with some datasource parameters, refer to the chapter § Maintenance > Logs.

Part II | **Applicative user interface**

2.1 Applicative user interface

The SBL10e device supports a Web user interface that can be accessed with a Web browser. The supported Web browsers are: `Google Chrome`, `Mozilla Firefox`, `MS-Edge (Chromium)`.

It is available from the URL: `http://<device_IP_addr>/`.

The URL falls into the `generic_server` applicative user interface: `http://<device_IP_addr>/webui/`. This pane allows to:

- watch the current `LED light color`,
- watch the current `LED light state`.

Part III

Administration console user interface
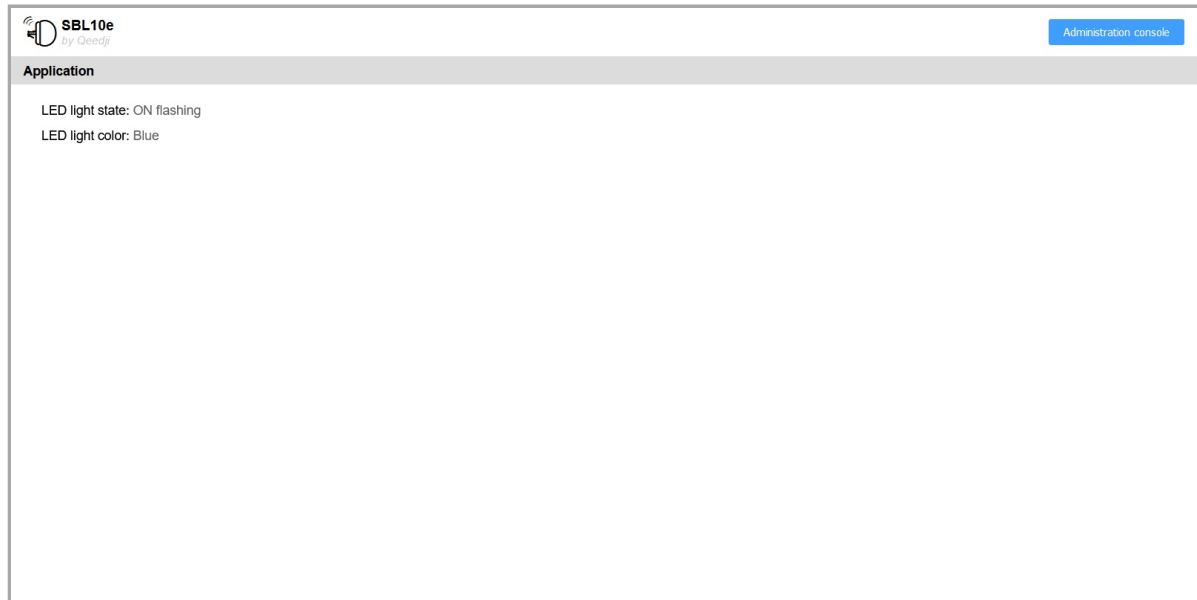
3.1 device configuration Web user interface

The SBL10e device supports a device configuration Web user interface that can be accessed with a Web browser. The supported Web browsers are: `Google Chrome` , `Mozilla Firefox` and `MS-Edge (Chromium)` .

It is available from the URL: `http://<device_IP_addr>/` .

The default credentials values, put at factory, to access to the device Web user interface are:

- login: `admin` ,

- password: `admin` .

The URL falls automatically into the applicative user interface¹. At the top right corner, click on the `Administration Console` button.

<div align="center">

Administration console

</div>

¹ *For further information, refer to the chapter §* *Applicative user interface*.

This is the device configuration Web user interface.
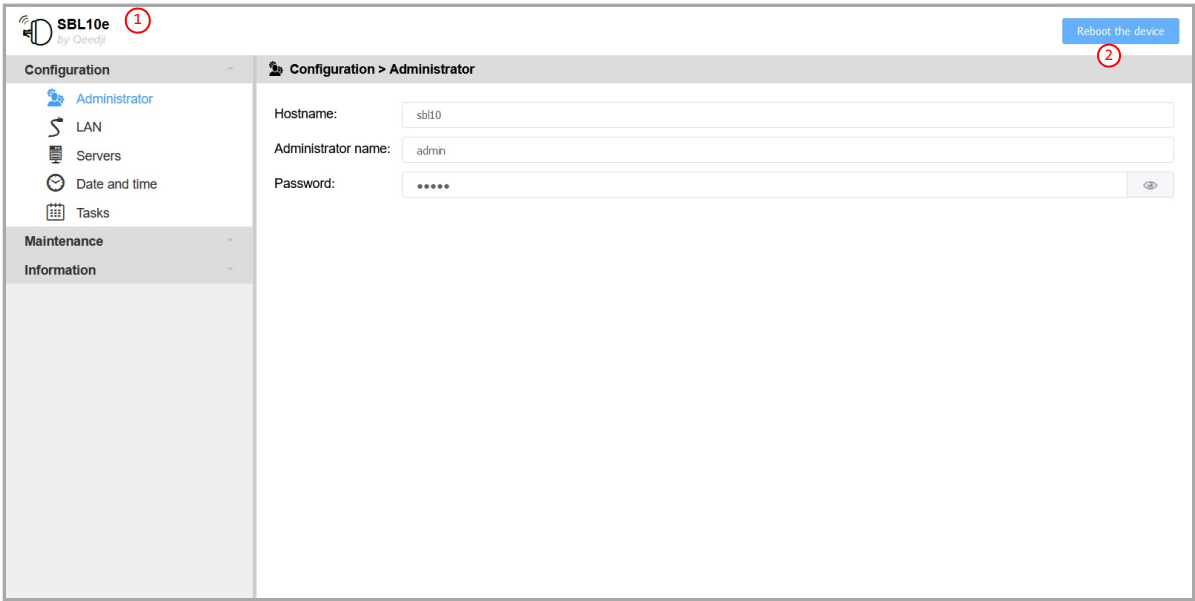
⚠ After you have changed and saved all your settings in the different panes, be sure to perform a device restart by clicking on the `Reboot the device` ② button so that your changes are fully reflected.

Click on the device logo ① at the left top corner to return to the applicative user interface.

3.1.1 Configuration > Administrator

In the `Configuration` tab, select the **Administrator** menu to change:

- the `Hostname` ,
- the login credentials:
  - `Administrator name` ,
  - `Password` .

> ☞ *It is recommended that you enter one unique `Hostname` value for each device. In case several SBL10e devices are located in different buildings or geographical locations, we recommend that you enter hostname values with information about the building and the location (e.g. `Hall-RD-Paris-1`).*

For security reasons, it may be useful to change the login credentials values. Please keep them in a safe place afterwards.

> ☞ *The same login credentials are used to access to the WebDAV server and to use Web services.*

3.1.2 Configuration > LAN

In the `Configuration` tab, select the **LAN** menu to set up the network configuration of the `LAN` interface of your device.

☞ *The device supports the UPnP and can be for example detected automatically in the local network environment of your computer.*

Enter a suitable LAN network configuration so that the device can access to the Web to get the local time with a NTP server.

☞ *By default, the device is configured with `Obtain an IP address automatically by DHCP` activated and `Time for attempts` deactivated. As soon as the DHCP server becomes available, the device ends by getting back a valid IP address given by the DHCP server within less than one minute.*

☞ *After a device reboot, when the device is configured with `Obtain an IP address automatically by DHCP` activated and `Time for attempts` is activated, in case the DHCP server is unavailable after the `Time for attempts` duration (ten minutes for the maximum and default value) has expired, the device ends up using the static IP address entered in the LAN configuration. The default static IP address is 192.168.0.2 when it has never been changed yet by the user. It is recommended to set an appropriate IP address, netmask and gateway if this case would happen. In case a daily reboot task is programmed, the device will restart this operation every days.*

☞ *When only the `Time for attempts` value is modified, press on TAB key of your keyboard to make appear the `Validate` button.*

3.1.3 Configuration > Servers

In the `Configuration` tab, select the **Servers** menu to define the configuration of the servers peripheral to your device.
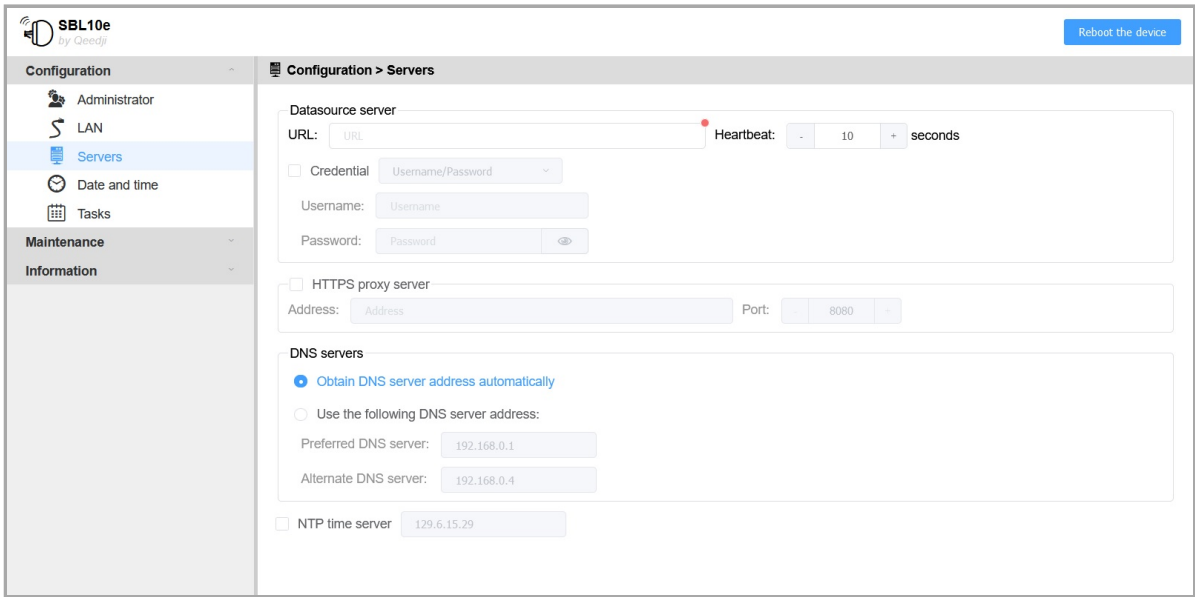
The `Datasource Server` is the server machine having the dynamic information of the current state and color to apply to your busylight:

- `Datasource Server` :
  - `URL` : enter the URL[2] including the `port` of your `Datasource Server` (only the *https* scheme is supported) ,
    - `Credential`
      - *checked*:
        - default type: *Username/Password*,
        - `username` : username to access to the server,
        - `password` : password to access to the server,
      - *unchecked* (default value): no username/password is required,
  - `Heartbeat` : periodicity of the connection to the `Datasource Server` , from 2 seconds (default value) to 120 seconds.
- `HTTPS proxy server` :
  - `Address` : enter the IPv4 address, or the domain name of your *proxy* server,
  - `Port` : enter the operating port of your *proxy* server,
- `DNS servers` ,
- `NTP time server` .

> ☛ *The `generic_server` application can work even though the NTP time server is not set.*
>
> ☛ *The login credentials inside the URL are not supported, i.e.* `https://user1:password1@192.168.10.10:443/sbl10/01320-00550` .
>
> ☛ *If the server is not available after 20 (default value stored in the* `appli.network.datasource.nb_retries_before_cache_reset` *user preference) consecutive unsuccessful connection attempts, the light is switched Off until the next successful connection attempt.*
>
> ☛ *When only the* `Heartbeat` *value is modified, press on TAB key of your keyboard to make appear the* `Validate` *button.*
>
> ☛ *With the* `generic_server` *application, when a proxy server is used, the DNS server can not be used. To work around, the DNS feature must be implemented in the proxy server.*

[2] *The URL has to match the directory of your database. If you have more than one SBL10e device to control, it is advised to implement one directory per SBL10e with the directory name matching either the device PSN, or the device MAC address, i.e.:* `https://192.168.10.10:443/sbl10/01320-00550` *with 01320-00550, an example of PSN value from a SBL10e device, or* `https://192.168.10.10:443/sbl10/00-1c-e6-02-4c-ad` *with 00-1c-e6-02-4c-ad, an example of MAC address value from a SBL10e device.*

To put the same URL for all your SBL10e datasource inputs, you can use two kind of variable values intrinsic to the device:

Variable type	String to set in the input	Value example
deviceInfo:mac variable	`{$deviceInfo:mac}`	*00-1c-e6-02-4c-ad[3]*
deviceInfo:psn variable	`{$deviceInfo:psn}`	*01320-00550*

[3] *The format with double dot instead of dash is not supported when using* `{$deviceInfo:mac}` *variable (i.e.: 00:1c:e6:02:4c:ad).*

The URL would be for example:

- `https://192.168.10.10:443/sbl10/{$deviceInfo:mac}` or,
- `https://192.168.10.10:443/sbl10/{$deviceInfo:psn}` .

For further information about the device PSN value, refer to the chapter § Information > Devices. For further information about the device MAC value, refer to the chapter § Information > Network.

⚠ Upgrading the SBL10e device with another application type will clear the current datasource configuration data. When the SBL10e is properly configured, it is advised to build and save an appropriate SBL10e configuration script (`.js`) by using the configuration script template or save at least the `prefs.json` configuration file of your SBL10e device. For further information, refer to the chapter § Maintenance > Files.

⚠ Upgrading the SBL10e device with another application type will clear the current datasource configuration data. When the SBL10e is properly configured, it is advised to build and save an appropriate SBL10e configuration script (`.js`) by using the configuration script template or save at least the `prefs.json` configuration file of your SBL10e device. For further information, refer to the chapter § Maintenance > Files.

3.1.4 Configuration > Date and time

In the `Configuration` tab, select the **Date and Time** menu to check the time configuration:

- timezone,
- system date of your device (day and time).

☞ *The application can work even though the device is not on time. Anyway, in case the* `Reboot` *task is activated, it is advised to set an appropriate date and time, synchronized with a NTP server, to control exactly when the SBL10e must reboot. For further information, refer to the chapter §* *Configuration > Servers.*

3.1.5 Configuration > Tasks

In the `Configuration` tab, select the **Tasks** menu to activate a device `reboot manager` task and adjust the `reboot manager` task time.
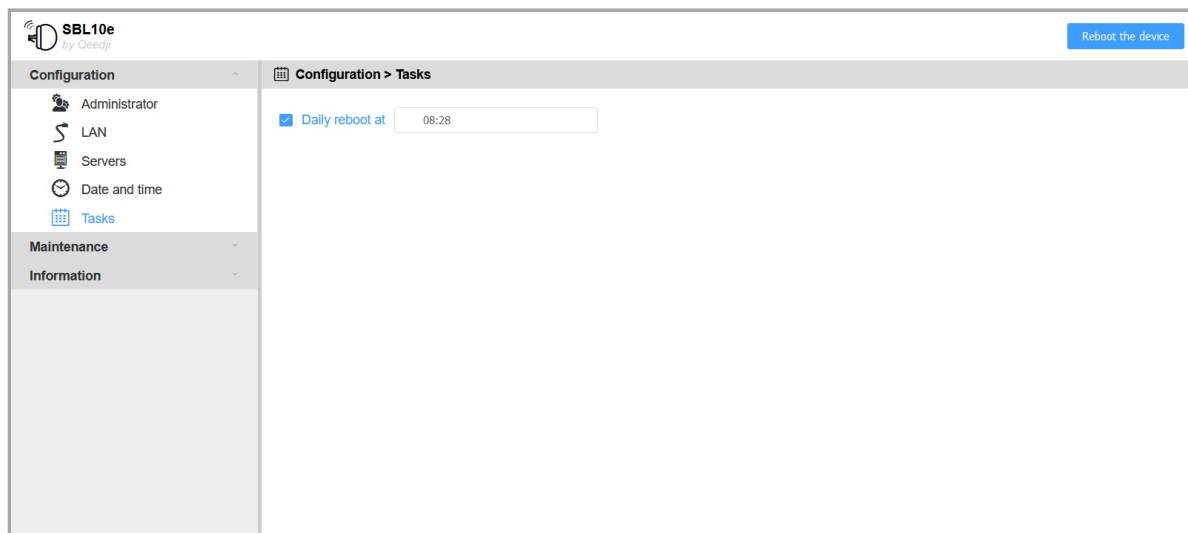
☞ *During the reboot task, the light state is* `off` *for a couple of seconds until the next data source server connection.*

☞ *If the NTP server set by the user is not available anymore and the* `system.task.reboot.enable` *user preference is true, the device is rebooting automatically every days, 24 hours after the last device reboot.*

3.1.6 Maintenance > Firmware

In the `Maintenance` tab, select the **Firmware** menu to view the version of the application installed on your device.
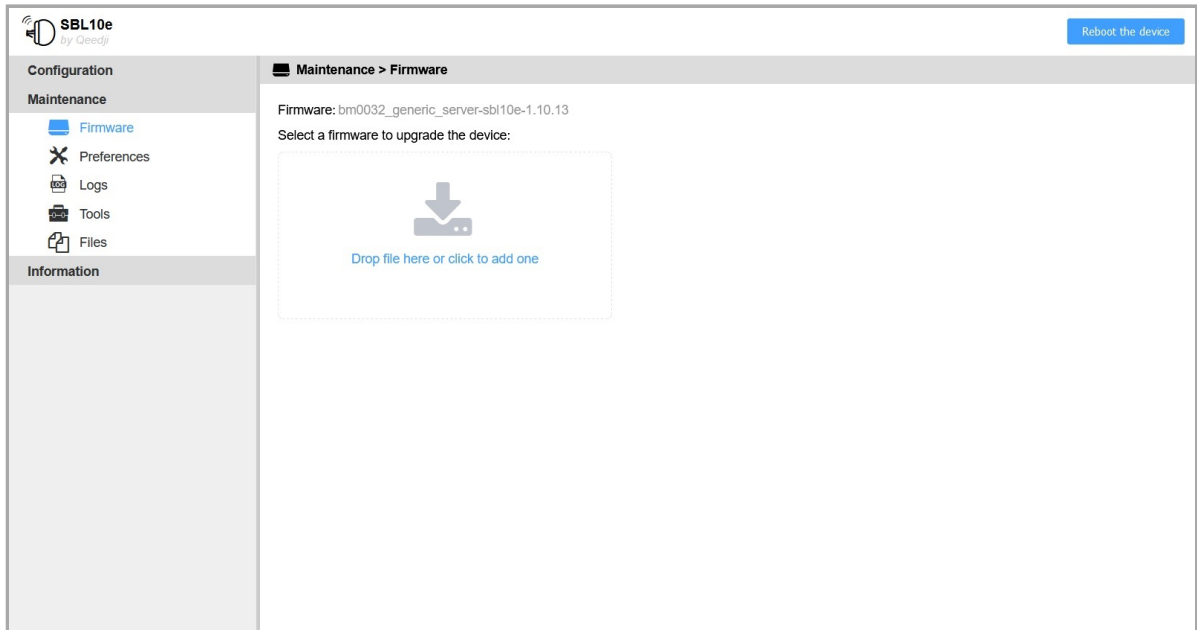
☞ *Corrective and evolutive maintenance software versions are regularly made available in the support tab of the official* `Qeedji` *website http://www.qeedji.tech/en/support/index.php?SBL10e/Generic_server. It is therefore advised to regularly update the device firmware version. From this website, download the appropriate latest firmware version available for your device model (* `.bin` *file). For further information, contact support@qeedji.tech.*

Drop your `.bin` file in the ⬇ `Drop file here or click to add one` location or click on it to add one, then click on the `Send` button to update the firmware version of your device. Wait a couple of seconds, the time to load and install the new firmware version. Connect again to the device configuration Web user interface and check the new firmware version.

The user preferences common to the applications are kept when upgrading from `regular` application to another application (and reciprocally) meaning:

- network configuration,
- file system,
- hostname,
- web user interface credentials.

⚠ Do not electrically disconnect the device during the firmware upgrade.

3.1.7 Maintenance > Preferences

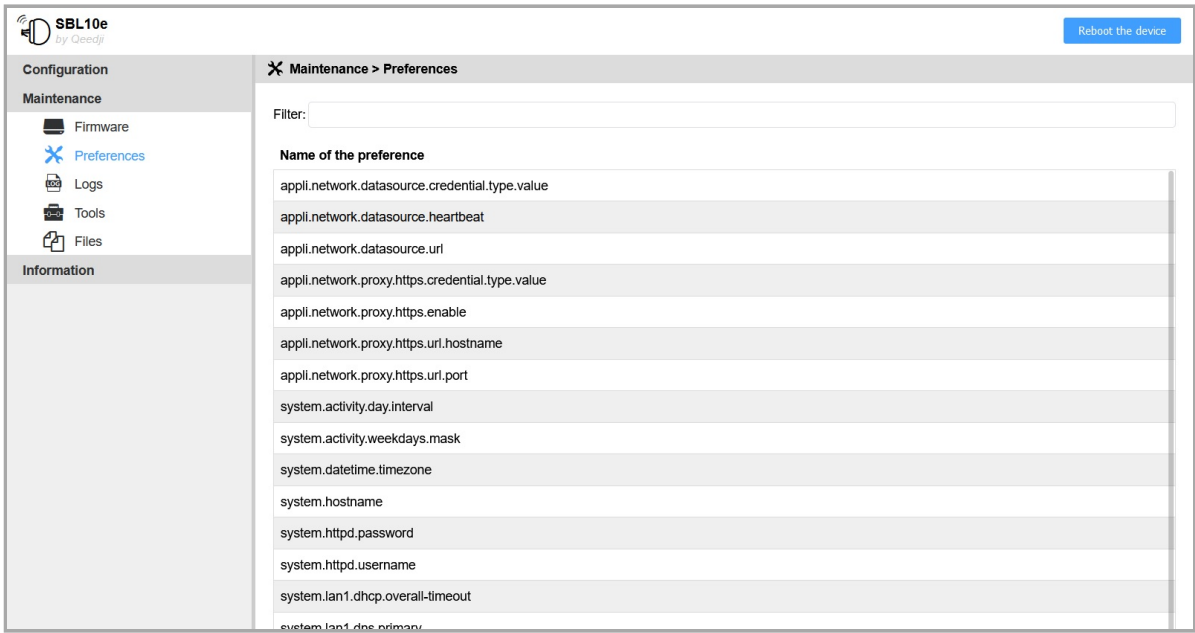In the `Maintenance` tab, select the **Preferences** menu to view all the preferences.

The filter allows to display only the preferences whose name contains the string entered in the filter. All the preferences have optimal default values.

Double click on a preference to change its value.

At the bottom right of the page, the `Restore factory preferences` button allows to reset a subset of preferences allowing the device to reprogram its factory preferences. In this case, the LAN network configuration returns to DHCP.

Click on the `Reboot the device` button so that the modifications are taken into account.

⚠ After a user preference restoration, in case a `.js` configuration script, suitable for the application of this SBL10e device, is available on the `TFTP` server, the user preference `system.tftp.enable` is set to *true*. Consequently, the SBL10e device is rebooting once again to take into account the `.js` configuration script available on the `TFTP` server.

Timezone

| Continent	*Country/Town* pair values supported for the `system.datetime.timezone` preference
Africa	Africa/Brazzaville, Africa/Cairo, Africa/Casablanca, Africa/Harare, Africa/Lagos, Africa/Nairobi, Africa/Onitsha, Africa/Windhoek
America	America/Anchorage, America/Argentina/Buenos_Aires, America/Barbados, America/Bogota, America/Boston, America/Caracas, America/Chicago, America/Chihuahua, America/Costa_Rica, America/Dallas, America/Denver, America/Godthab, America/Halifax, America/Houston, America/Los_Angeles, America/Manaus, America/Mexico_City, America/Montevideo, America/New_York, America/Phoenix, America/Recife, America/Regina, America/Rio_de_Janeiro, America/San_Francisco, America/Santiago, America/Sao_Paulo, America/St_Johns, America/Tijuana, America/Washington,_D.C.
Asia	Asia/Ahmedabad, Asia/Almaty, Asia/Amman, Asia/Baghdad, Asia/Baku, Asia/Bangalore, Asia/Bangkok, Asia/Beijing, Asia/Beirut, Asia/Chengdu, Asia/Chennai, Asia/Chongqing, Asia/Colombo, Asia/Delhi, Asia/Dongguan, Asia/Dubai, Asia/Guangzhou, Asia/Hangzhou, Asia/Hanoi, Asia/Ho_Chi_Minh, Asia/Hong_Kong, Asia/Hyderabad, Asia/Irkutsk, Asia/Jakarta, Asia/Jerusalem, Asia/Kabul, Asia/Karachi, Asia/Kathmandu, Asia/Kolkata, Asia/Krasnoyarsk, Asia/Kuala_Lumpur, Asia/Kuwait, Asia/Lahore, Asia/Magadan, Asia/Mumbai, Asia/Nagoya, Asia/Nanjing, Asia/Oral, Asia/Osaka, Asia/Pune, Asia/Quanzhou, Asia/Seoul, Asia/Shanghai, Asia/Shenyang, Asia/Shenzhen, Asia/Surat, Asia/Taipei, Asia/Tbilisi, Asia/Tehran, Asia/Tianjin, Asia/Tokyo, Asia/Vladivostok, Asia/Wuhan, Asia/Xi'an, Asia/Yakutsk, Asia/Yangon, Asia/Yekaterinburg, Asia/Yerevan, Asia/Zhengzhou
Atlantic	Atlantic/Azores, Atlantic/Cape_Verde, Atlantic/South_Georgia
Australia	Australia/Adelaide, Australia/Brisbane, Australia/Darwin, Australia/Hobart, Australia/Perth, Australia/Sydney
Europe	Europe/Amsterdam, Europe/Athens, Europe/Belgrade, Europe/Berlin, Europe/Brussels, Europe/Dusseldorf, Europe/Helsinki, Europe/Istanbul, Europe/London, Europe/Madrid, Europe/Minsk, Europe/Moscow, Europe/Paris, Europe/Sarajevo, Europe/Warsaw
Pacific	Pacific/Auckland, Pacific/Fiji, Pacific/Guam, Pacific/Honolulu, Pacific/Majuro, Pacific/Midway, Pacific/Noumea, Pacific/Tongatapu
UTC	Etc/UTC

3.1.8 Maintenance > Logs

In the `Maintenance` tab, select the **Logs** menu to view the logs.

In the example, there is no error raised in the logs except the warning for the time which is not set.

When the `system.tftp.server` user preference is *true*:

- in case there is some available `.js` configuration script on the `TFTP` server with the appropriate file name pattern, this message is printed: `TFTP client enable: matching configuration script available on TFTP server (<IP address>)`.
- in case there is no `.js` configuration script on the `TFTP` server with the appropriate file name pattern, this message is printed: `TFTP client enable: no configuration script available on TFTP server for this device (<IP address>)`.
- in case the `TFTP` server is not available, this message is printed: `TFTP client enable: error server did not respond (<IP address>)`.

⚠ To be successfully taken into account, the content of the `.js` configuration script available on the `TFTP` server must also be suitable for the SBL10e application.
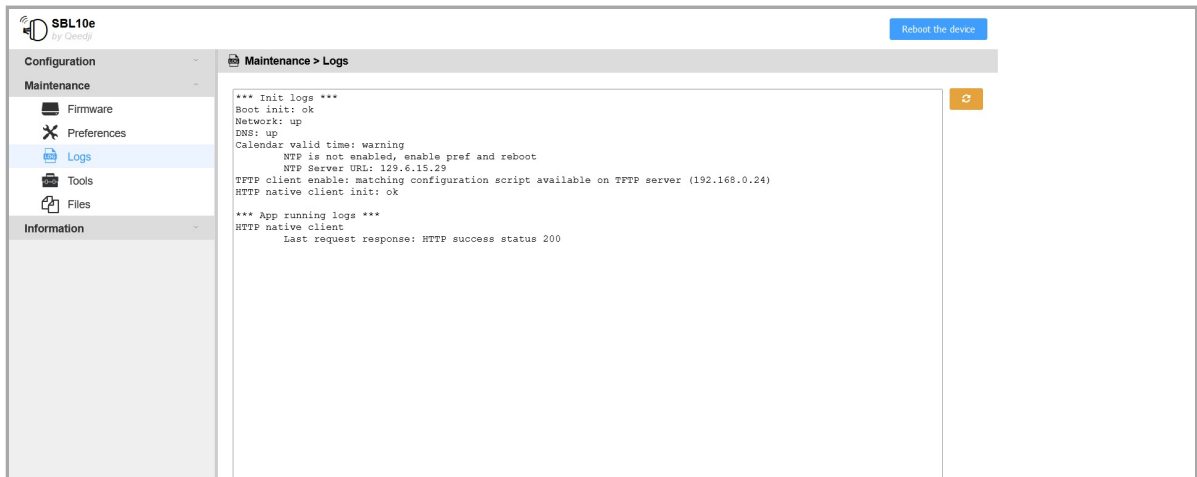
In the example, when the NTP is set and the NTP server available, there is no error or warning at all raised in the logs.
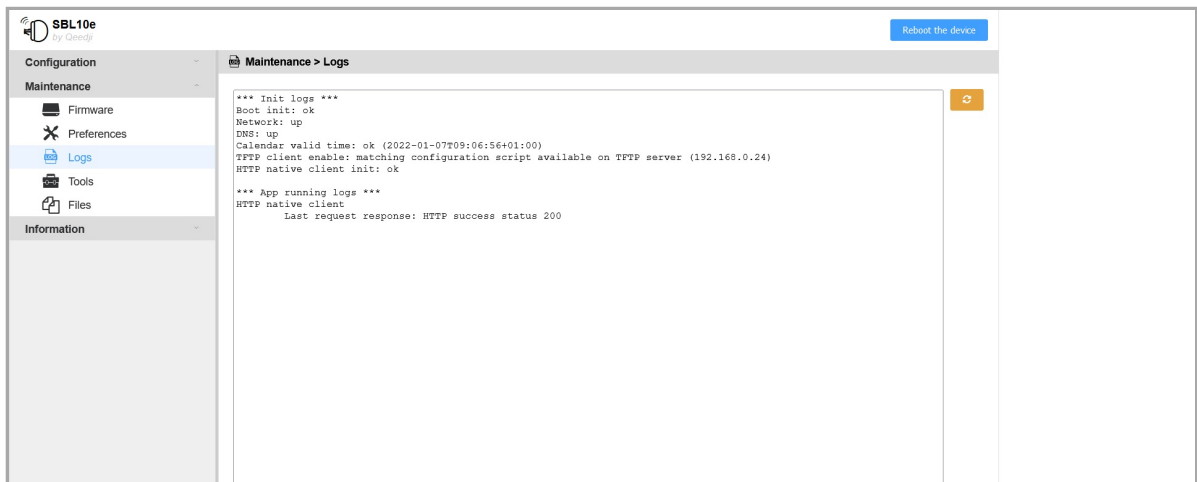
In case your SBL10e device is flashing 6 times in the blue color every 4 seconds meaning that the SBL10e configuration is probably not correct, you are invited to check the logs in this window to try to fix the trouble. It is assumed that the warning linked to the date does not prevent the application to work properly.

The logs are allowing to know whether:

- the datasource server is available,
- some state and color data are existing for this specific device.

3.1.9 Maintenance > Tools

In the `Maintenance` tab, select the **Tools** menu to:

- view the available space on the flash memory storage[1] (max 7916 KiB),
- format the flash memory storage[1],
- add the `Trusted certificate` (.crt) for the `Datasource server`.
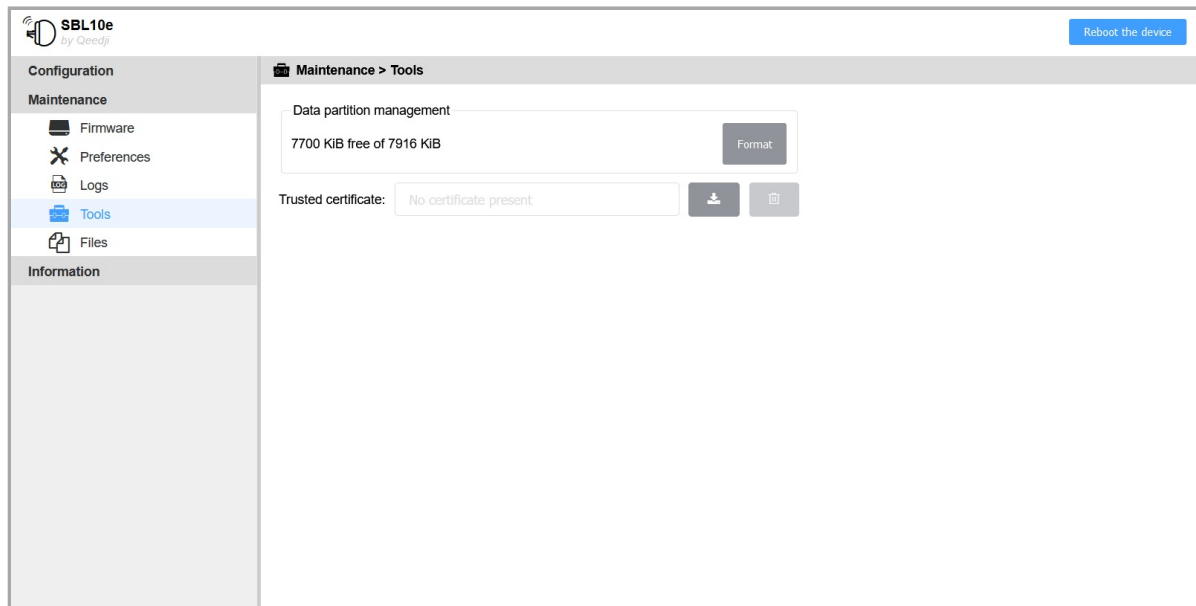
> ☞ *The trusted certificate is not checked in this version.*

[1] *The flash memory storage is used to store all the directories and files hosted at the root of the WebDAV directory, and the user preferences as well. In case a flash formatting is done, the device returns to the default factory settings. In this case, the trusted certificate is erased and the datasource server data are cleared.*

3.1.10 Maintenance > Files

In the `Maintenance` tab, select the **Files** menu to see the directories and files hosted at the root directory of the WebDAV server.

As soon as a modification is done through the device configuration Web user interface, a `prefs.json` file, corresponding to the new device configuration, is created in the `.conf` folder.

When the user preference `system.tftp.enable` is *true*, a `tftp_crc` file, containing the CRC of the `.js` configuration script downloaded from the `TFTP` server is written in the `.conf` folder. To be downloaded again from the `TFTP` server, either the suitable configuration script must be modified on the `TFTP` server, or the `tftp_crc` file must be removed.

⚠ After having pressed on the `Restore factory preferences` button, the `prefs.json` file and the `tftp_crc` file are erased.

⚠ The content of the `prefs.json` ③ configuration file can be different for `generic_server` application and other applications.

Save or restore the device configuration

⚠ It is recommended to save the `configuration.js` preciously to configure your SBL10e device in order to be able to restore its configuration afterwards.

⚠ The content of the `configuration.js` is depending on the used application. Do use the suitable `configuration.js` for the `generic-server` application.
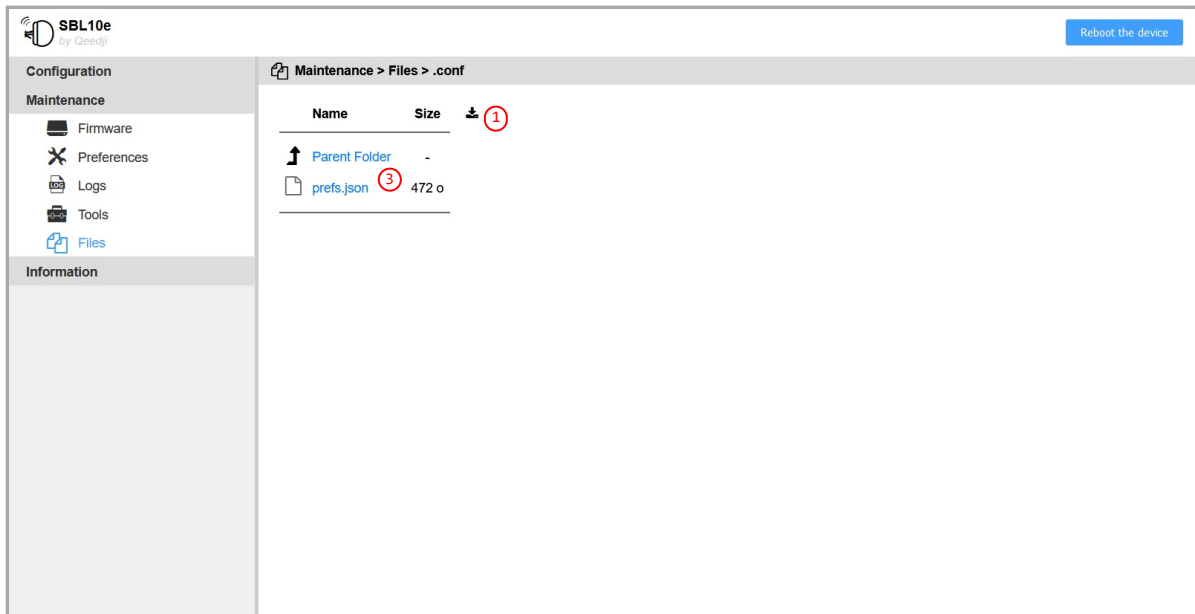
Date and time

The system date and time can only be set and updated thanks to a NTP server. In this case, you have to:

- First:
    - define the timezone value,
    - define the NTP server IP address,
    - check that the NTP time server is activated (activated by default).
- Secondly:
    - define the gateway URL to access to internet,
    - define valid primary and secondary DNS servers.

For further information, refer to the chapter:

- § Configuration > Servers,
- § Configuration > LAN,
- § Configuration > Date and time,

☞ *In case the SBL10e device can not fetch a valid date and time through NTP at device boot-up, the clock does not progress and stays with the value `01/01/2020 00:00`. The date and time metadata of the files added after this date on the file system is also `01/01/2020 00:00`. The last modification date for a file can be only be seen with a WebDAV client.*

☞ *When the server NTP is activated and the device is properly configured, the date and time for the SBL10e device is updated automatically by NTP at the device boot-up. Then it is progressing every seconds.*

☞ *The support for date and time file metadata display in this pane will be available in a next version.*

Application upgrade

The current application can be upgraded by pushing a new firmware file `bm0032_generic_server-sbl10e-xx.yy.zz.bin` at the root of the device WebDAV directory `http://<device-ip-addr>/` with a WebDAV client.

After the firmware file pushing, a device reboot is required so that the new firmware file is taken into account.

Configuration update

The configuration of the application can be updated also by pushing an appropriate `.js` configuration script (or a suitable `prefs.json` file) suitable for your application in the `.conf` WebDAV directory (`http://<device-ip-addr>/.conf`) with the Web user interface or with a WebDAV client.

> ⚠ Loading a wrong `prefs.json` would lead to some loss of data like the datasource server configuration. So check the consistency of the `prefs.json` file before uploading it in the device. To avoid any error on the configuration of the application and the configuration of the SBL10e device, it is advised to use a `.js` configuration script (and not with a `prefs.json` file) which is testing before executing anything that it is suitable for the SBL10e device and suitable for the application running on the device. Qeedji provides configuration script template. It is then highly recommended for the user to save an appropriate `.js` configuration script for each SBL10e device installed in his building.

A `000000000000.js` template is available for download here.

In this case, the file pattern must be either:

- `configuration.js` : suitable for any device whatever its MAC address,
- `000000000000.js` : suitable for any device whatever its MAC address,
- `<device_LAN1_MAC_address>.js` (with the format `ABCDEFABCDEF.js`): suitable for device whose MAC address is matching.

After having downloaded the configuration script template:

- edit the `000000000000.js` configuration script and uncomment/modify the appropriate lines according to your needs,
- rename the configuration script if required,
- once saved, drop it in the `.conf` WebDAV directory like explained above,
- when the .js configuration script is satisfying, save it preciously to be able to restore its configuration after wards.

After a `.js` configuration script uploading in the device, the device is rebooting automatically once to take the new configuration into account.

> ☞ The `prefs.json` file is available in the `.conf` WebDAV directory of the devive as soon as the SBL10e device configuration is modified at least once by the user. After a device configuration updating with a `prefs.json` file, a device reboot is required so that the new configuration is taken into account.

> ☞ Pushing a `.js` configuration script in the `.conf` WebDAV directory (`http://<device-ip-addr>/.conf`) with a WebDAV client could raise a warning at the WebDAV client end, after the `.js` file transferring is completed because the device is automatically rebooting once when it is received. For example, after the `.js` file sending with BitKinex WebDAV is done, another network request is done by the WebDAV client while the device is currently rebooting. So a WebDAV error at the WebDAV client end leads to an automatic file resending which is causing another device reboot and so on, and this, until the WebDAV client application is closed. For example, after the `.js` file sending with CarotDAV WebDAV client, the error leads only to the displaying of a warning message. The user has just to ignore the error at the WebDAV client end.

3.1.11 Information > Device

In the `Information` tab, select the **Device** menu to view system information about the device.
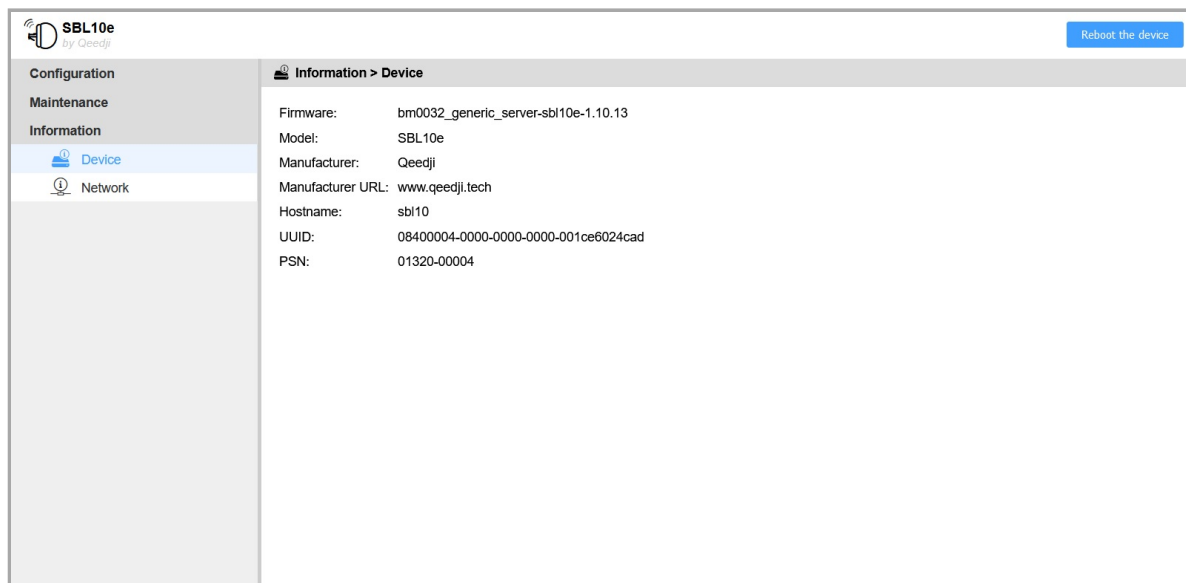
- `Firmware` : label and version of the firmware embedded in the device,

- `Model` : model of the `Qeedji` device,

- `Manufacturer` : product manufacturer name,

- `Manufacturer URL` : manufacturer Website,

- `Hostname` : name of the device on the network,

- `UUID` : Universal Unique IDentifier,

- `PSN` : Product Serial Number.

3.1.12 Information > Network

In the `Information` tab, select the **Network** menu to view a summary of the device's network configuration.

☞ *The displaying of the IP V6 address value starting with the prefix* `fe80::` *is not supported in this pane. For further information, contact your IT department so that your network is advertising the IP V6 address with another prefix (ex:* `fc00::` *).*

Part IV | Technical information

4.1 Technical specifications

Model	Manufacturer
SBL10e	Qeedji

Power supply	Information
PoE IEEE802.3af	POE power supply input: ES1 / PS2 (48 V – 100 VA)

Processors	
CPU	Nordic Semiconductor nRF52
Security processor	ARM CryptoCell 310

Storage	
Flash Memory for file system	8 MBytes

Network	Other information
1x Ethernet	10/100 Base T, male connector

WPAN
Bluetooth Low Energy 5
Frequency band: 2.402 to 2.480 GHz
Tx Power: +8 dBm

Operating temperature	Storage temperature
+0 °C to +40 °C	-20 °C to +60 °C
+32 °F to +104 °F	-4 °F to +140 °F

Operating humidity	Storage humidity
< 80 %	< 85 %

Weight	Dimensions (W x H x D) (RJ45 male connector included)
35 g	60,5 mm x 60,5 mm x 67 mm
0,077 lb	2,36" x 2,36" x 2,63"

Plastic enclosure flame rating
Base: PVC UL 94-5VA, bulb: Polycarbonate UL 94 V-2

Warranty
1 year

4.2 Conformities

EUROPE

In conformity with the following European directives:

- `LVD 2014/35/EU` ,
- `EMC 2014/30/EU` ,
- `RED 2014/53/EU` .

USA

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference's by one or more of the following measures:

- reorient or relocate the receiving antenna,
- increase the separation between the equipment and the receiver,
- connect the equipment into an outlet on a circuit different from that to which the receiver is connected,
- consult the dealer or an experienced radio/TV technician for help.

This equipment complies with FCC's radiation exposure limits set forth for an uncontrolled environment under the following conditions:

- this equipment should be installed and operated such that a minimum separation distance of 20 cm is maintained between the radiator (antenna) and user's/nearby person's body at all times,
- this transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

- this device may not cause harmful interference,
- this device must accept any interference received, including interference that may cause undesired operation.

Qeedji is not responsible for any changes or modifications not expressly approved by the party responsible for compliance. such modifications could void the user's authority to operate the equipment.

CANADA

This device contains licence-exempt transmitter(s)/receiver(s) that comply with Innovation, Science and Economic Development Canada's licence-exempt RSS(s). Operation is subject to the following two conditions:

- this device may not cause interference,
- this device must accept any interference, including interference that may cause undesired operation of the device.

Part V | Contacts

5.1 Contacts

For further information, please contact us:

- **Technical support**: support@qeedji.tech,
- **Sales department**: sales@qeedji.tech.

Refer to the `Qeedji` Website for FAQ, application notes, and software downloads: https://www.qeedji.tech/

```
Qeedji FRANCE
INNES SA
5A rue Pierre Joseph Colin
35700 RENNES

Tel: +33 (0)2 23 20 01 62
Fax: +33 (0)2 23 20 22 59
```

Part VI | Appendix

6.1 Appendix: Web services

These are the supported Web services for the `generic_server` application to command and control the SBL10e:

Webservice path	HTTP method				Query string parameters	Body	Function	From the generic_server application version
	GET	POST	PUT	DELETE				
api/v1/sys/power			yes		<state>=rebooting	""	Reboot the device	1.10.13
api/v1/software/version	yes				None	i.e.: {"value":"1.10.13"}	Get the device delivery software version	1.10.13
api/v1/software/label	yes				None	{"value":"bn0032_generic_server"}	Get the device delivery software label	1.10.13
api/v1/sys/sn	yes				None	PSN Short representation: i.e.: {"value":"01320-00004"}	Get the device SN	1.10.13
api/v1/sys/model-name	yes				None	{"value":"SBL10e"}	Get the device model name	1.10.13
api/v1/sys/manufacturer	yes				None	{"value":"Qeedji"}	Get the manufacturer	1.10.13
api/v1/sys/manufacturer-url	yes				None	{"value":"www.qeedji.tech"}	Get Web Site of manufacturer	1.10.13
api/v1/sys/uuid	yes				None	Uuid string value: <uuid> = <psn>-<48x0>-<mac-48> i.e.: {"value":"08400004-0000-0000-0000-001ce6024cad"}	Get the device UUID	1.10.13
api/v1/wpan1/mac	yes				None	Bluetooth MAC address value user formatted: i.e.: {"value":"db:f0:8c:72:64:a3"}	Get the device Bluetooth MAC address	1.10.13
api/v1/leds/light	yes				None	Get (plain text, separator CR): i.e.: {"state":"steady","color":"blue"} <state>= off | steady | flashing <color>=red | orange | blue | yellow | green	Get busylight led color and state	1.10.13
api/v1/sys/datetime	yes				None	{"value":"2021-08-11T06:09:58-02:30"}	Get busylight date and time	1.10.13

Examples syntax with CURL tool:

- reboot the device:

```
curl --user "<USERNAME>:<PASSWORD>" -i -X PUT "http://<DEVICE_IP_ADDR>/api/v1/sys/power?state=rebooting"
```

- get device state & color:

```
curl --user "<USERNAME>:<PASSWORD>" -X GET "http://<DEVICE_IP_ADDR>/api/v1/leds/light"
```

- get device firmware version:

```
curl --user "<USERNAME>:<PASSWORD>" -X GET "http://<DEVICE_IP_ADDR>/api/v1/software/version"
```

6.2 Appendix: Qether

In case an application can not be executed, the SBL10e returns to a `Recovery` mode, waiting for firmare update.

The provided `Qether` tool allows to make some remote operations on the SBL10e, based on its device MAC address like:

- SBL10e device firmware upgrade,
- SBL10e device configuration update,
- SBL10e device reboot.

The `<product_type>` is an extract of the device PSN value. For example, the *0132x-xxxx* `PSN` value leads to the *0132* `<product_type>`.

The `<SBL10e_device_MAC_address>` is the MAC address of the device with the format `00:1C:E6:AB:CD:EF`.

> ⚠ The MAC address of the device is written on the label stuck at the back of the SBL10e device with the format `00-1C-E6-AB-CD-EF`.

Discover command example

This command allows to find out the SBL10e devices available on the local network:

```
qether.exe FF 0132 discover
```

Configuration command syntax

Send a `.js` configuration script and apply it (default parameters):

```
qether.exe <SBL10e_device_MAC_address> <product_type> configure -f configuration.js
```

> ☞ When using `Qether`, no specific filename pattern is required for the `.js` configuration script, except the `` `.js `` file extension.

> ☞ The `system.httpd.username` preference value is limited to 15 characters max. The `system.httpd.password` preference value is limited to 100 characters max. The alphanumeric characters and the following characters `{}|~[]!#_$&()/;<=@|^%?+~((),'` are supported for the `system.httpd.username` and `system.httpd.password` preference values.

> ☞ The `system.hostname` preference value is limited to 15 characters max. The alphanumeric characters, the character `-` and the character `.` are supported for the `system.hostname` preference value.

> ☞ To get an IP address with the DHCP server, set `system.lan1.ipv4.static-addr` with the value `false`. Else to work with a static IP address, set `system.lan1.ipv4.static-addr` with the value `true`.

Reboot command syntax

Reboot the target device:

```
qether.exe <SBL10e_device_MAC_address> <product_type> reboot
```

Firmware upgrade command syntax

Send a firmware file, with default transfer parameters, and install it. For example:

```
qether.exe <SBL10e_device_MAC_address> <product_type> install -f bm0032_generic_server-sbl10e-setup-1.11.12.bin
```

> ☞ Qether needs first to be installed first on your MS-Windows computer. For further information, refer to the *Qether user manual*.

6.3 Appendix: Device configuration with TFTP server (+ DHCP server code 66)

The SBL10e device can be configured thanks to a `.js` configuration script (Javascript) hosted on a `TFTP` [1] server associated to a `DHCP` server (code *66* option) properly configured and available on the local network.

[1] Trivial File Transfer Protocol

The `.js` configuration script downloading can be done as soon as a `DHCP` server is available, even whether the device is configured with a static IP address. Once connected to the `DHCP` server, the device can get:

- the IP address value of its network interface, when the option `Obtain IP address automatically by DHCP` is activated then,
- the primary DNS value when the `system.lan1.dns.static` user preference is *false* then,
- the `.js` configuration script from the `TFTP` server when the `system.tftp.enable` user preference is *true*.

Prerequisites:

- the approriate `.js` configuration script must be is available in the exported directory of the `TFTP server`. It must:
  - be suitable for the device, its firmware type and its firmware version,
  - match an appropriate filename pattern:
    - `000000000000.js` or,
    - `<device_LAN1_MAC_address>.js` (with the format `ABCDEFABCDEF.js`).

When a `.js` configuration script is modified on the `TFTP` server, the device must be restarted once so that the new configuration script is taken into account by the device.

> ☞ *When using a `TFTP` server, the `configuration.js` filename pattern is not supported.*

> ⚠ The downloading of a `.js` configuration script from a `TFTP` server can be done only at the device booting-up and when the device has never downloaded it before or when the script content has been modified since the last download (CRC check).

DHCP server configuration

The `DHCP` server must be configured to be associated to a `TFTP` server. For that, you need to use code 66 option (TFTP Server), using the IPv4 address value of the `TFTP` server.

For example, for a `Microsoft DHCP server`, you need to define the option *Boot Server Host Name* and give the IPv4 address of the `TFTP` server. It can be in *Extended option* and/or *Server Options*.

> ☞ *The service must be restarted so that the modifications are fully reflected.*
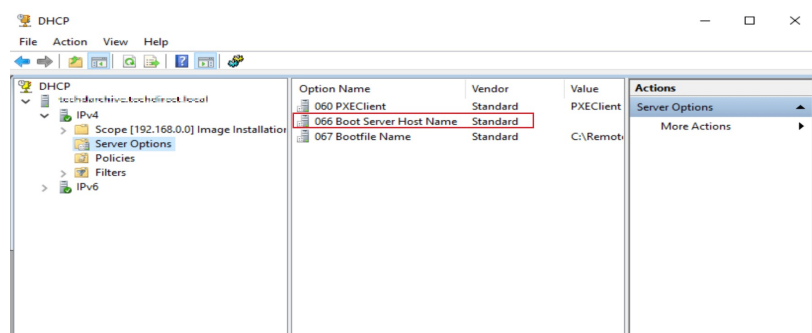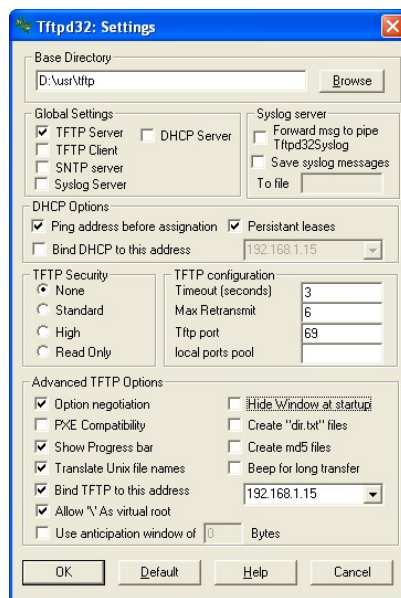

TFTP server configuration

The configuration is depending on the used software client. In all cases, you need to:

- get the directory URL that can be seen by `TFTP` clients,
- choose a `TFTP` security *None*,
- keep the default port (69).

Here is an example of the `tftpd32` software with MS-Windows.

In this example, the `TFTP` server address is `192.168.1.15` and the exported directory is `D:/usr/tftp` .

☛ *In this pane, enter the IP address of the* `TFTP` *server. Indeed entering the* `TFTP` *server domain name may prevent the feature to work properly.*

Copy the `.js` configuration script in the exported directory of the `TFTP server` .

☛ *It is recommended to have one* `.js` *configuration script per device by following the pattern* `<MAC>` *.js.*