kuando HTTP Manual (MacOS)

Version: 1.0(4)

Content

1. Introduction

The kuando HTTP offers you a new application scenario to your kuando Busylight. Applications can now control your Busylight with http requests.

Kuando HTTP runs as a tray icon app in the notification area.

If you need support for kuando HTTP, please contact support@busylight.com.

For more information or tools to help with the implementation go to https://www.plenom.com.

Kuando HTTP for MacOS is 100% compatible to the Windows versions and kuandoHUB.

Kuando HTTP for MacOS is tested with these MacOS Releases:

- MacOS Big Sur (11.0.1)
- MacOS Catalina (10.15.7)
- MacOS Mojave (10.14)
- MacOS High Sierra (10.13)
- MacOS Sierra (10.12)

2. HTTP API interface

The HTTP interface allow you to control the Busylight from your own software.

The HTTP Server access token is only required for remote access from other devices.

The HTTP server URL can only be set from the Registry. If the value cannot be processed, the http server will be disabled. The default value, http://localhost:8989/ will allow connections from the same machine. If you want to allow remote access, change the configuration value to http://+:8989/.

For details about the HTTP server URL, see here: https://docs.microsoft.com/en-us/dotnet/api/system.net.httplistener

The http server access token is only used if a request is coming from a remote server.

See Appendix A – Configuration settings for configuration details.

HTTP GET API interface

The HTTP GET interface is a very simple interface to control the Busylight. A http get command can be sent from an internet browser or a webhook as an example.

Below are the GET Requests that can be made: The examples show local requests which do not need a security token.

If a security token is needed, it is added as an additional parameter (http_token=tokendata)

If you want to use the default value, then you do not need to provide a parameter.

Light command:
```
http://localhost:8989?action=light&red=100&green=100&blue=100
```

Parameters:

Parameter Name	Values
Red	0..100 (Default 0)
Green	0..100 (Default 0)
Blue	0..100 (Default 0)

Alert Command:
```
http://localhost:8989?action=alert&red=100&sound=5&volume=25
```

Parameters:

Parameter Name	Values
Red	0..100 (Default 0)
Green	0..100 (Default 0)
Blue	0..100 (Default 0)
Sound	0..8 (See list)
Volume	0..100 (See list)

Sound list:

Sound	Sound number
(No Sound)	0
Fairy Tale	1
Funky	2
Kuando Train	3 (Default)
Open Office	4
Quiet	5
Telephone Nordic	6
Telephone original	7
Telephone Pick Me Up	8

Volume can be set to these values:

Volume	Volume value
100%	100
75%	75 (Default)
50%	50
25%	25
Mute	0

Blink command

```
http://localhost:8989?action=blink&blue=100
```

Parameters:

Parameter Name	Values
Red	0..100 (Default 0)
Green	0..100 (Default 0)
Blue	0..100 (Default 0)
Ontime (0.1 seconds steps light on)	Default: 5 (0.5 Seconds)
Offtime (0.1 seconds steps light off)	Default: 5 (0.5 Seconds)

Jingle command

```
http://localhost:8989?action=jingle&red=100&sound=3&volume=100
```

Parameters:

Parameter Name	Values
Red	0..100 (Default 0)
Green	0..100 (Default 0)
Blue	0..100 (Default 0)

Sound	0..8 (See list)
Volume	0..100 (See list)

For Sound and Volume values, see the Alert command.

Pulse command

```
http://localhost:8989?action=pulse&blue=100&red=100
```

Parameters:

Parameter Name	Values
Red	0..100 (Default 0)
Green	0..100 (Default 0)
Blue	0..100 (Default 0)

ColorWithFlash command

```
http://localhost:8989?action=colorwithflash&green=100&red=100&flashblue=100
```

Parameters:

Parameter Name	Values
Red	0..100 (Default 0)
Green	0..100 (Default 0)
Blue	0..100 (Default 0)
flashred	0..100 (Default 0)
flashgreen	0..100 (Default 0)
flashblue	0..100 (Default 0)

Off command

```
http://localhost:8989?action=off
```

The Off-Command has no parameters associated with it.

currentpresence command

```
http://localhost:8989?action= currentpresence
```

This command will return the current presence as json string. It contains the running priority information and the priority present in my own priority, regardless if it is running or not.

PowerShell Sample

```
Invoke-WebRequest -URI "http://localhost:8989?action=light&red=100&green=100&blue=100"
```

JavaScript Sample

```
const Http = new XMLHttpRequest();
    const url='http://localhost:8989?action=light&red=0&green=100&blue=0';
    Http.open("GET", url);
    Http.send();

    Http.onreadystatechange = (e) => {
      console.log(Http.responseText)
      }
```

HTTP POST API interface

The HTTP POST interface can be used by any software which is able to do a HTTP POST request, e.g. PowerShell, JavaScript inside a Browser or a desktop app.

The HTTP POST interface enables all SDK functionalities including registering Priority configuration and Data Sources.

The HTTP POST interface accepts json data.

Common Parameters

The json structure has four common parameters, which control the kuandoHUB operation.

Parameter Name	Values
action	Command for kuandoHUB
sender	Datasource for kuandoHUB priority operation
eventtype	Event Type for kuandoHUB priority operation
eventname	The name of the specific event for kuandoHUB priority operatrion
parameter	This property contains an embedded json structure with the parameters for the specific action. The structure is different and specific for each action. (See examples below.)

Light command:

```
{
            "action":"Light",
            "sender":"SDK",
            "eventtype":"Light",
            "eventname":"Color",
            "parameter":"{
                        \"RedRgbValue\":0,
                        \"GreenRgbValue\":100,
                        \"BlueRgbValue\":0
                        }"
}
```

Parameters:

Parameter Name	Values
RedRgbValue	0..100 (Default 0)
GreenRgbValue	0..100 (Default 0)
BlueRgbValue	0..100 (Default 0)

Alert Command:

```
{
            "action":"Alert",
            "sender":"SDK",
            "eventtype":"Alert",
            "eventname":"Alert",
            "parameter":"{
                        \"color\":{
                                    \"RedRgbValue\":255,
                                    \"GreenRgbValue\":0,
                                    \"BlueRgbValue\":0
                                    },
                        \"clip\":5,
                        \"volume\":50
                        }"
}
```

Parameters:

Parameter Name	Values
RedRgbValue	0..100 (Default 0)
GreenRgbValue	0..100 (Default 0)
BlueRgbValue	0..100 (Default 0)
Clip	0..8 (See list)
Volume	0..100 (See list)

Sound list:

Clip	Sound number
(No Sound)	0
Fairy Tale	1
Funky	2
Kuando Train	3 (Default)
Open Office	4
Quiet	5
Telephone Nordic	6
Telephone original	7
Telephone Pick Me Up	8

Volume can be set to these values:

Volume	Volume value
100%	100
75%	75 (Default)
50%	50
25%	25
Mute	0

Blink command

```
{
```

```
            "action":"Blink",
            "sender":"SDK",
            "eventtype":"Blink",
            "eventname":"Blink",
            "parameter":"{
                        \"color\":{
                                    \"RedRgbValue\":255,
                                    \"GreenRgbValue\":0,
                                    \"BlueRgbValue\":0
                                    },
                        \"ontime\":5,
                        \"offtime\":5
                        }"
}
```

Parameters:

Parameter Name	Values
RedRgbValue	0..100 (Default 0)
GreenRgbValue	0..100 (Default 0)
BlueRgbValue	0..100 (Default 0)
Ontime (0.1 seconds steps light on)	Default: 5 (0.5 Seconds)
Offtime (0.1 seconds steps light off)	Default: 5 (0.5 Seconds)

Jingle command

```
{
            "action":"Jingle",
            "sender":"SDK",
            "eventtype":"Jingle",
            "eventname":"Jingle",
            "parameter":"{
                        \"color\":{
                                    \"RedRgbValue\":255,
                                    \"GreenRgbValue\":255,
                                    \"BlueRgbValue\":0
                                    },
                        \"clip\":9,
                        \"volume\":50
                        }"
}
```

Parameters:

Parameter Name	Values
RedRgbValue	0..100 (Default 0)
GreenRgbValue	0..100 (Default 0)
BlueRgbValue	0..100 (Default 0)
Clip	0..8 (See list)
Volume	0..100 (See list)

For Sound and Volume values, see Alert command.

Pulse command

```
{
            "action":"Pulse",
            "sender":"SDK",
            "eventtype":"Pulse",
            "eventname":"Pulse",
            "parameter":"{
                        \"color\":{
                                    \"RedRgbValue\":255,
                                    \"GreenRgbValue\":255,
                                    \"BlueRgbValue\":0
                                    },
                        \"pulsesequence\":{
                                    \"Color\":{
                                                \"RedRgbValue\":255,
                                                \"GreenRgbValue\":255,
                                                \"BlueRgbValue\":0
                                                },
                                    \"Step1\":3,
                                    \"Step2\":21,
                                    \"Step3\":36,
                                    \"Step4\":50,
                                    \"Step5\":36,
                                    \"Step6\":21,
                                    \"Step7\":10
                                    }
                        }"
}
```

Parameters:

Parameter Name	Values
RedRgbValue	0..100 (Default 0)
GreenRgbValue	0..100 (Default 0)
BlueRgbValue	0..100 (Default 0)
Step1 .. Step7	Intensity of the step (0..100)

ColorWithFlash command

```
{
            "action":"ColorWithFlash",
            "sender":"SDK",
            "eventtype":"Light",
            "eventname":"ColorWithFlash",
            "parameter":"{
                        \"color\":{
                                    \"RedRgbValue\":0,
                                    \"GreenRgbValue\":255,
                                    \"BlueRgbValue\":0
                                    },
                        \"flash\":{
                                    \"RedRgbValue\":0,
                                    \"GreenRgbValue\":0,
                                    \"BlueRgbValue\":255
                                    }
                        }"
}
```

Parameters:

The color parameter determines the solid color, the color of the short flashes. Both colors are defined as designated in the parameter table:

Parameter Name	Values
RedRgbValue	0..100 (Default 0)

GreenRgbValue	0..100 (Default 0)
BlueRgbValue	0..100 (Default 0)

Off command

```
{
            "action":"Off",
            "sender":"SDK",
            "eventtype":"Light",
            "eventname":"Off",
            "parameter":""
}
```

The Off-Command has no specific parameters.

RegisterDataSource command

This command allows you to register a custom data source in kuandoHUB for priority operation. It is recommended to send this command on every connect to kuandoHUB to make sure that the data source exists and has the most recent version. See kuandoHUB operation chapter for details.

```
{
            "action":"RegisterDataSource",
            "sender":null,
            "eventtype":null,
            "eventname":null,
            "parameter":"{
                        \"DataSourceName\":\"SDK\",
                        \"eventnames\":{
                                    \"Light\":{
                                                \"EventNames\":[
                                                            \"Light\",
                                                            \"Green\",
                                                            \"Red\",
                                                            \"Yellow\",
                                                            \"Off\"],
                                                \"IsAlertPriority\":false
                                                },
                                    \"Alert\":{
                                                \"EventNames\":[
                                                            \"Alert\",
                                                            \"Other Notification\"],
                                                \"IsAlertPriority\":true
                                                }
                                    }
                        }"
}
```

CreateInitialPriority command

The CreateInitialPriority command allows to create Priority entries in the kuandoHUB priority operation.

```
{
            "action":"CreateInitialPriority",
            "sender":null,
            "eventtype":null,
            "eventname":null,
            "parameter":"{
                        \"enabled\":true,
                        \"sender\":\"SDK\",
                        \"eventtype\":null,
                        \"eventnames\":[],
                        \"IsAlertPriority\":false
                        }"
}
```

This command creates a line in kuandoHUB priorities with the data Source name "SDK" containing all eventtypes and event names.

The command will have no effect if a line containing the data source name is already in the kuandoHUB priorities list.

currentpresence command

```
{
            "action":"currentpresence",
            "sender":"SDK",
            "eventtype":"",
            "eventname":"",
            "parameter":""
}
```

The currentpresence-Command has no specific parameters.

This command will return the current presence as json string. It contains the running priority information and the priority present in my own priority, regardless if it is running or not.

Powershell sample

This command switches the Busylight to green light. Please make sure you have registered the SDK DataSource and have an enabled priority line entry in kuandoHUB.

```
$lightcmd =
'{"action":"Light","sender":"SDK","eventtype":"Light","eventname":"Color","parameter":"{
\"RedRgbValue\":0,\"GreenRgbValue\":100,\"BlueRgbValue\":0}"}'

Invoke-WebRequest -Body $lightcmd -Method 'POST' -Uri 'http://localhost:8989'
```

JavaScript sample

This commands switches the Busylight to green light. Please make sure you have registered the SDK DataSource and have an enabled priority line entry in kuandoHUB.

```
    const Http = new XMLHttpRequest();
    const url='http://localhost:8989';
    Http.open("POST", url);

Http.send('{"action":"Light","sender":"SDK","eventtype":"Light","eventname":"Color","par
ameter":"{\\"RedRgbValue\\":0,\\"GreenRgbValue\\":100,\\"BlueRgbValue\\":0}"}');
    Http.onreadystatechange = (e) => {
      console.log(Http.responseText)
      }
```

Please note the double-escaped quotes in the parameter section.

3. HTTP Responses

These response codes are defined in the Busylight HTTP implementations:

200 (OK)	Process request successful
404 (no found)	Busylight device is not connected
401 (unauthorized)	http token invalid

Appendix A – Configuration settings

The configuration is stored in this configuration file:

~/Library/Preferences/com.plenom.busylight.http.settings.plist

These settings can be customized:

Setting	Description	Default Value
http_uri	Listening URL	http://localhost:8989/
http_token	Access token for remote access	

You can use xcode or a text editor to edit the values.