Application Notes

Lightware Update REST API for UCX/MMX2 Devices

Update \{REST:API\}
Table of Contents

1. LIGHTWARE UPDATE REST API .....................................................3
  1.1. INTRODUCTION ......................................................................4
  1.2. DEVICE IDENTIFICATION – OPTIONAL ........................................4
    1.2.1. Querying the Product Name of the Device ..........................4
    1.2.2. Querying the Serial Number of the Device .........................4
    1.2.3. Querying the Device Label .................................................4
  1.3. THE UPDATING PROCESS .....................................................5
    1.3.1. Uploading the Package ......................................................5
    1.3.2. Configuration Parameters – Keep the Current Settings ........6
    1.3.3. Executing the Update .......................................................6
    1.3.4. Querying the State of the Updating Process .......................7
1. Lightware Update REST API

This document is about the updating process for UCX/MMX2 devices with Lightware Update REST API. This protocol allows integrating the update process into a third-party environment.
1.1. Introduction

The Update REST API (REpresentational State Transfer Application Public Interface) is developed to have a standardized update interface between the Lightware device and a third-party software tool (e.g. external controller). The update means - generally - to install a new Lightware Firmware Package (LFP2) to the Lightware device. The Update REST API is part of the entire Lightware REST API software architecture.

The Update REST API is available over HTTP/HTTPS and optionally with basic authentication (with setting a password for the user admin).

1.2. Device Identification – Optional

The following commands help get basic information from the desired device.

1.2.1. Querying the Product Name of the Device

This is a read-only parameter, which cannot be changed.

Request and Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
</tbody>
</table>

Example

- request: GET http://<ip/host>/api/ProductName
- response: <standard_response>
- body: <message>

1.2.2. Querying the Serial Number of the Device

This is a read-only parameter, which cannot be changed.

Request and Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
</tbody>
</table>

Example

- request: GET http://<ip/host>/api/SerialNumber
- response: 200 OK
- body: D1000001

1.2.3. Querying the Device Label

Request and Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
</tbody>
</table>

Example

- request: GET http://<ip/host>/api/V1/MANAGEMENT/LABEL/DeviceLabel
- response: <standard_response>
- body: <message>

1.2.4. Querying the Device Manufacturer

This is a read-only parameter, which cannot be changed.

Request and Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
</tbody>
</table>

Example

- request: GET http://<ip/host>/api/V1/MANAGEMENT/LABEL/DeviceManufacturer
- response: <standard_response>
- body: <message>

1.2.5. Querying the Device Model

This is a read-only parameter, which cannot be changed.

Request and Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
</tbody>
</table>

Example

- request: GET http://<ip/host>/api/V1/MANAGEMENT/LABEL/DeviceModel
- response: <standard_response>
- body: <message>

1.2.6. Querying the Device Type

This is a read-only parameter, which cannot be changed.

Request and Response

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
</tbody>
</table>

Example

- request: GET http://<ip/host>/api/V1/MANAGEMENT/LABEL/DeviceType
- response: <standard_response>
- body: <message>
1. The Updating Process

The API allows the device to update itself with an LFP2 FW package. This new approach means there is no need to run a software on the connected PC (in case of LDU2) for updating the Lightware device.

The Updating Steps

1. The user initiates the update process.
2. The device checks for authenticity and compatibility; extract the package.
3. The user configures the package (optional).
4. The user sends a POST to start the update.
5. The device performs the updating and extracting of the file system.
6. The device restores the settings or recalls the factory default values.
7. The device is ready to use.

1.3.1. Uploading the Package

This step may take up to one or two minutes to finish as the size of the LFP2 package is approx. 100-150 MB (depends on the device and FW version).

Request and Response

- request: POST http://<ip/host>/api/V1/MANAGEMENT/UPDATE/Package
- body: <LFP2 file>
- content-type: application/octet-stream
- response: <standard_response>
- body: <message>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
<tr>
<td>&lt;standard_response&gt;</td>
<td>Standard HTTP response</td>
<td>200 OK</td>
<td>&lt;message&gt;: OK The request has succeeded; the whole package is extracted and the basic checks are done.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>400 Bad request</td>
<td>&lt;message&gt;: Package incompatible: Incompatible partnumber: [...] The request cannot be processed; error during the uploading/checking. e.g. the package is not compatible with the device. The &lt;message&gt; is in text/plain format.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>500 Internal Server Error</td>
<td>&lt;message&gt;: Socket timeout The package uploading was not successful. Please try it again.</td>
</tr>
</tbody>
</table>

Example

- request: POST http://192.168.0.114/api/V1/MANAGEMENT/UPDATE/Package
- body: (the LFP2 file)
- response: 200 OK
- body: OK
1.3.2. Configuration Parameters – Keep the Current Settings

**DIFFERENCE:** This feature is available from FW package v2.1.0.

**ATTENTION!** Restoring the device settings is not allowed if the firmware is downgraded to a previous version or the existing (=running) FW package of the device is v1.x. In those cases, recalling the factory default settings is a must.

This is an optional step that allows keeping the current settings of the device. The default setting of an update is to recall the factory default values. If you want to keep the current settings of the device, delete the ‘dofactoryreset’ file from the uploaded package as follows:

**Request and Response**

- request: DELETE `http://<ip/host>/api/V1/MANAGEMENT/UPDATE/CONFIG/dofactoryreset`
- response: `<standard_response>`
- body: `<message>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
<tr>
<td>&lt;standard_response&gt;</td>
<td>Standard HTTP response</td>
<td>200 OK</td>
<td>&quot;Deleted&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>405 Method not allowed</td>
<td>The file does not exist in the uploaded package (e.g. it was deleted previously). Continue the update process.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>500 Internal Server Error</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

- request: `DELETE http://192.168.0.114/api/V1/MANAGEMENT/UPDATE/CONFIG/dofactoryreset`
- response: `200 OK`
- body: `Deleted`

1.3.3. Executing the Update

The command is for installing the uploaded package in place of the current system. This step may take up to 7 minutes.

**Request and Response**

- request: POST `http://<ip/host>/api/V1/MANAGEMENT/UPDATE/update`
- body: `<payload>`
- content-type: `text/plain`
- response: `<standard_response>`
- body: `<message>`

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
<tr>
<td>&lt;payload&gt;</td>
<td>optional</td>
<td>force</td>
<td>The update can be forced to run in Package incompatible error state, but the security check cannot be bypassed (Package untrusted state).</td>
</tr>
<tr>
<td>&lt;standard_response&gt;</td>
<td>Standard HTTP response</td>
<td>200 OK</td>
<td>The request has succeeded; the update process finished successfully.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>400 Bad request</td>
<td>The request cannot be processed. It may happen that the uploaded package is not compatible with the device (Package incompatible error state) and the force payload was not used.</td>
</tr>
</tbody>
</table>

**ATTENTION!** Please note that running a forced update can make the device inoperable.

**Example**

- request: `POST http://192.168.0.114/api/V1/MANAGEMENT/UPDATE/update`
- response: `200 OK`
- body: `OK`

When the ‘force’ payload is used:

- request: `POST http://192.168.0.114/api/V1/MANAGEMENT/UPDATE/update`
- body: `force`
- content-type: `text/plain`
- response: `200 OK`
- body: `OK`
1. Lightware Update REST API

1.3.4. Querying the State of the Updating Process

This command is for querying the current state of the updating process.

Request and Response

- request: GET http://<ip/host>/api/V1/MANAGEMENT/UPDATE/Status
- response: <standard_response>
- body: <message>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Parameter description</th>
<th>Values</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip/host&gt;</td>
<td>The IP address or the host name of the device.</td>
<td>e.g.: 192.168.0.110, myDevice</td>
<td></td>
</tr>
<tr>
<td>&lt;standard_response&gt;</td>
<td>Standard HTTP response</td>
<td>200 OK</td>
<td>The request has succeeded.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;message&gt; states:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Idle: not active, update has not started yet.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Package uploading: updating and checking is in progress.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Package ready: uploading is done, checking is successful.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>400 Bad Request</td>
<td>The request cannot be processed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;message&gt; states:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Package incompatible: the uploaded package is not compatible with the device.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Package untrusted: the uploaded package is not secure.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Failed: the update is not successful.</td>
</tr>
</tbody>
</table>

INFO: Please note that while the new firmware is being programmed in the device, it cannot send a response for a request. Thus the current state cannot be requested either.

After the firmware update is finished successfully, the device would send the 'Idle' response when requesting its status with this command.

Example

- request: GET http://192.168.0.114/api/V1/MANAGEMENT/UPDATE/Status
- response: 200 OK
- body: Package uploading