

Kuando Busylight LoRa und The Things Network: Erste Schritte

Inhaltsverzeichnis

Erstellen Sie ein TTN-Konto und eine Anwendung	1
Registrieren Sie Ihr Gateway.....	3
Registrieren Sie Ihr Busylight	6
Testen des Busylight	9
TTN-Payload-Formatierer für das Busylight.....	10
Steuerung des Busylight mit HTTP-Anfragen	13
Busylight LoRa Hardware Nutzdatenformat	15

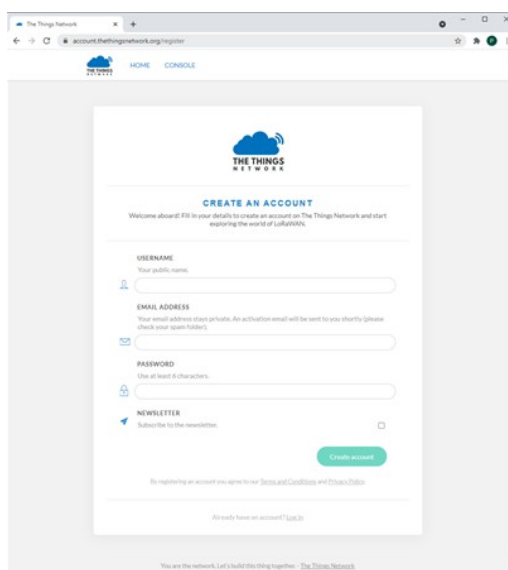
The Things Network ermöglicht die Adressierung der Busylight LoRa-Geräte mit https-Anfragen. Wir beschreiben, wie Sie auf verschiedene Weise Befehle an Ihr LoRa Busylight senden können.

Zur Demonstration verwenden wir „The Things Network Community Edition“. Bitte beachten Sie, dass die Community Edition Ihre Busylight-Steuerbefehle verzögern kann.

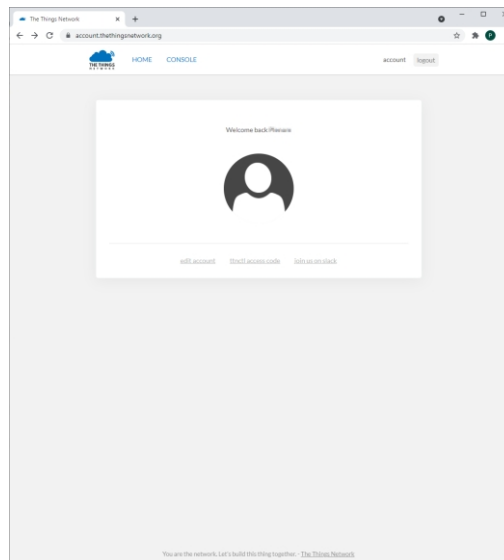
Erstellen Sie ein TTN-Konto und eine Anwendung

Der erste Schritt besteht darin, einen Benutzer für TTN zu erstellen: Öffnen Sie einen Browser und gehen Sie zu <https://www.thethingsnetwork.org/>

In der rechten oberen Ecke finden Sie die Schaltfläche „Anmelden“.

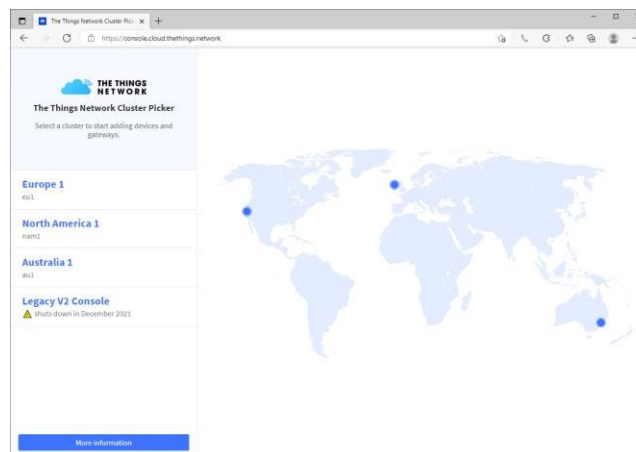


Nach dem Ausfüllen des Formulars erfolgt eine E-Mail-Bestätigung. Nach der Bestätigung sind Sie angemeldet.

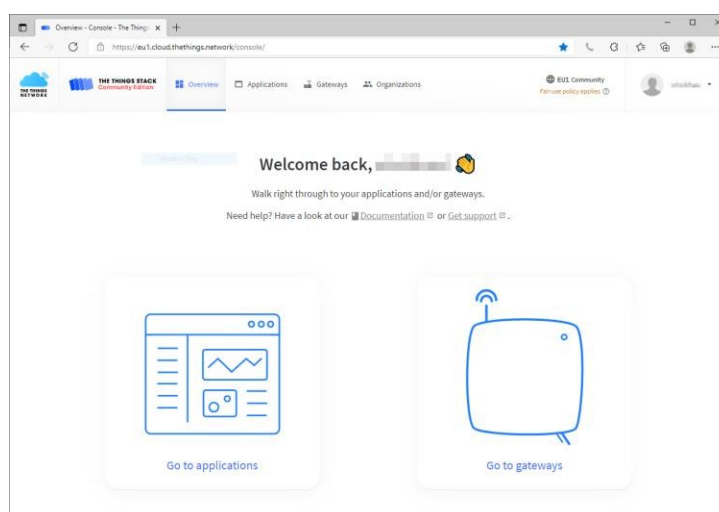


Bitte klicken Sie auf „Console“.

Wählen Sie nun den für Sie geeigneten Netzwerkcluster aus, in der Regel den nächstgelegenen.



Nach der Auswahl gelangen Sie zur TTN-Konsole:



Klicken Sie nun auf „Anwendungen“ – „Anwendung hinzufügen“.

The screenshot shows the 'Add application' page in the The Things Network console. The form is titled 'Add application' and contains the following fields:

- Owner:** A dropdown menu with 'personal' selected.
- Application ID:** A text input field containing 'testapplication'.
- Application name:** A text input field containing 'Busylight test application'.
- Description:** A text area with the placeholder text 'Description for my new application'.

Below the description field, there is a note: 'Optional application descriptions can also be used to save notes about the application.' At the bottom of the form is a blue button labeled 'Create application'.

Bitte notieren Sie sich die Anwendungs-ID, Sie werden sie später benötigen.

Registrieren Sie Ihr Gateway

In der Regel ist das Verfahren zum Hinzufügen eines LoRa-Gateways zu TTN in der Gateway-Dokumentation sehr gut beschrieben, bei den gängigsten Gateways auch in der TTN-Dokumentation.

Die TTN-Dokumentation finden Sie hier:

<https://www.thethingsnetwork.org/docs/devices-and-gateways/adding-gateways/>

Hier zeigen wir ein einfaches Gateway mit dem Semtech UDP Packet Forwarder.

Um das Gateway zu registrieren, müssen Sie die EUID des Gateways kennen.

Klicken Sie in der TTN-Konsole auf „Gateways“ und dann auf „Gateway hinzufügen“.

Add gateway

General settings

Owner:

Gateway ID:

Gateway EUI:

Gateway name:

Gateway description:

Optional gateway description, can also be used to save notes about the gateway

Gateway server address:

The address of the Gateway Server to connect to

Require authenticated connection: ☐ Enabled

Controls whether this gateway may only connect if it uses an authenticated Basic Station or MQTT connection

Gateway status: ☐ Public

The status of this gateway may be visible to other users

Gateway location: ☐ Public

The location of this gateway may be visible to other users and on public gateway maps

Attributes: [+ Add attributes](#)

Attributes can be used to set arbitrary information about the entity, to be used by scripts, or simply for your own organisation

LoRaWAN options

Frequency plan:

Schedule downlink rate: ☐ Enabled

Enable server-side buffer of downlink messages

Enforce duty cycle: ☒ Enabled

Recommended for all gateways in order to respect spectrum regulations

Schedule any time delay: milliseconds

Configure gateway delay (minimum: 10ms, default: 500ms)

Gateway updates

Automatic updates: ☐ Enabled

Gateway can be updated automatically

Channel:

Channel for gateway automatic updates

[Create gateway](#)

© 2023 The Things Stack by The Things Network and The Things Industries

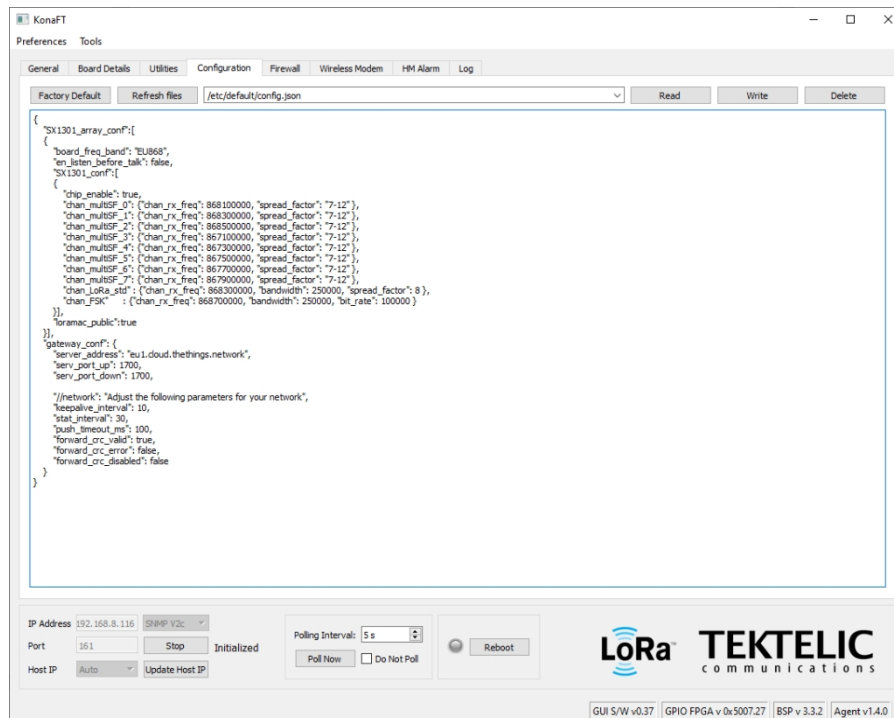
Bitte wählen Sie die für Ihr Land geeigneten Werte für Frequencies (Frequenzen) aus.

In der Gateway-Konfiguration müssen Sie die Serveradresse eingeben, die hier im Registrierungsformular angezeigt wird.

Beispiel: Für ein Tectelic-Gateway müssen Sie den FQDN des Servers in der Datei `/etc/default/config.json` eingeben.

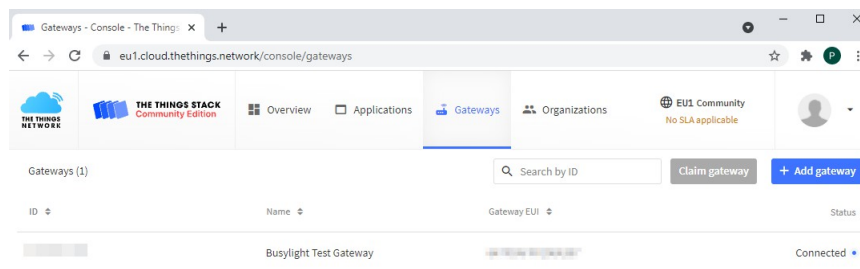
Die Serveradresse muss angepasst werden.

Hier sehen Sie ein Beispiel für die europäische TTN-Community-Cloud:



Nachdem Sie den Inhalt der Datei geändert und gespeichert haben, müssen Sie den Paketweiterleiter (oder das Gateway) neu starten.

Wenn alles korrekt durchgeführt wurde, wird der Gateway im Status „verbunden“ angezeigt.



Registrierung Ihres Busylight

Für die Registrierung Ihres Busylight-Geräts benötigen Sie folgende

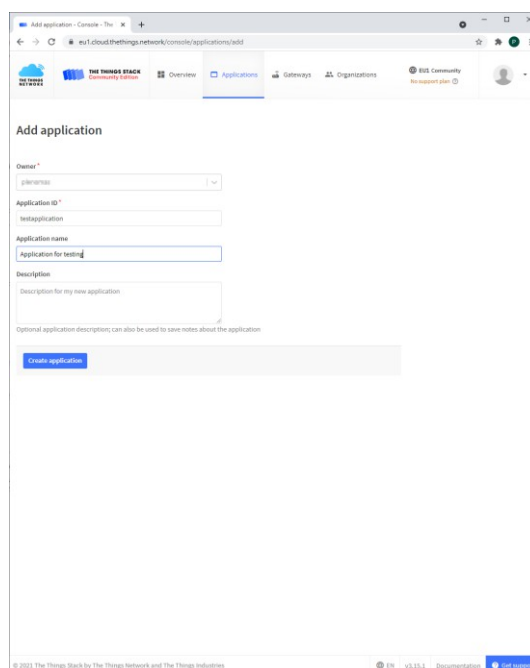
Informationen: DeviceEUI (8 Byte Hex)

AppEUI (8 Byte Hex)

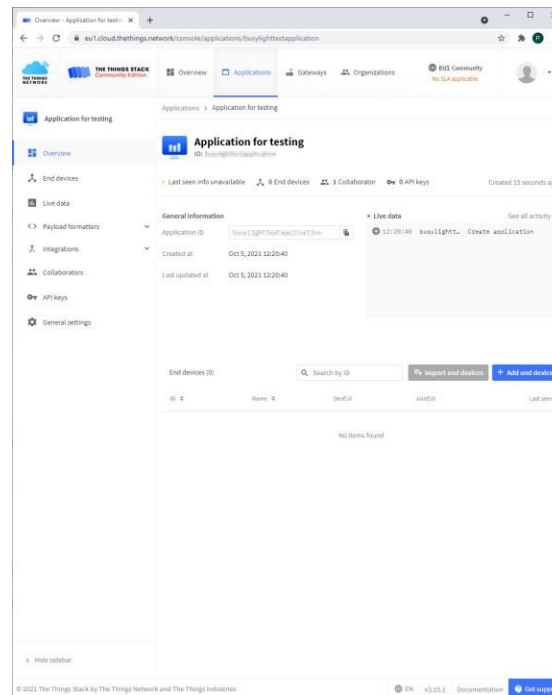
AppKey (16 Byte Hex)

Wenn Sie bereits eine Anwendung in TTN erstellt haben, können Sie den nächsten Schritt überspringen.

Wenn Sie keine Anwendung haben, erstellen Sie bitte eine mit der Schaltfläche „+ Anwendung hinzufügen“.

The screenshot shows a web browser window with the URL 'eu1.cloud.thethings.network/console/applications/add'. The page title is 'Add application'. It contains several input fields: 'Device' (a dropdown menu with 'plenum' selected), 'Application ID' (a text field with 'testapplication' entered), 'Application name' (a text field with 'Application for testing' entered), and 'Description' (a text area with the placeholder 'Description for my new application'). Below these fields is a blue button labeled 'Create application'. The footer of the page includes copyright information '© 2022 The Things Stack by The Things Network and The Things Industries', version 'v3.5.1', and a 'Get support' link.

Bitte geben Sie die Anwendung ein, indem Sie darauf klicken.



Bitte klicken Sie jetzt auf „+ Endgerät hinzufügen“ und dann auf „Manuell“.

Bitte füllen Sie das Formular wie folgt aus und verwenden Sie dabei den für Ihren Standort geeigneten Frequenzplan:

The screenshot shows the 'Register end device' page in the The Things Network console. The browser address bar shows the URL: `eu1.cloud.thethings.network/console/applications/busylighttestapplication/devices/add/manual`. The page has a sidebar on the left with navigation links: Overview, End devices (selected), Live data, Payload formatters, Integrations, Collaborators, API keys, and General settings. The main content area is titled 'Register end device' and has two tabs: 'From The LoRaWAN Device Repository' and 'Manually' (selected). The form contains several sections: 'LoRaWAN version' (MAC V1.0.3), 'Regional Parameters version' (PHY V1.0.3 REV A), 'Frequency plan' (Europe 863-870 MHz (SF12 for RX2)), 'Show advanced activation, LoRaWAN class and cluster settings', 'Activation mode' (Over the air activation (OTAA) selected), 'Additional LoRaWAN class capabilities' (Class C (Continuous)), 'Network defaults' (Use network's default MAC settings checked), 'Cluster settings' (Use external LoRaWAN backend servers unchecked), 'DevEUI' (0/50 used), 'AppEUI' (Fill with zeros), 'AppKey' (Generate), 'End device ID' (eui-20...), 'After registration' (View registered end device selected), and a 'Register end device' button at the bottom.

Register end device - Application x +

eu1.cloud.thethings.network/console/applications/busylighttestapplication/devices/add/manual

THE THINGS NETWORK THE THINGS STACK Community Edition

Overview Applications Gateways Organizations EU1 Community No support plan

Application for testing

Overview End devices Live data Payload formatters Integrations Collaborators API keys General settings

Applications > Application for testing > End devices > Register manually

Register end device

From The LoRaWAN Device Repository Manually

LoRaWAN version * MAC V1.0.3

Regional Parameters version * PHY V1.0.3 REV A

Frequency plan * Europe 863-870 MHz (SF12 for RX2)

Show advanced activation, LoRaWAN class and cluster settings ^

Activation mode *
☒ Over the air activation (OTAA)
☐ Activation by personalization (ABP)
☐ Define multicast group (ABP & Multicast)

Additional LoRaWAN class capabilities *
Class C (Continuous)

Network defaults *
☒ Use network's default MAC settings

Cluster settings *
☐ Use external LoRaWAN backend servers

DevEUI *
0/50 used

AppEUI *
Fill with zeros

AppKey *
Generate

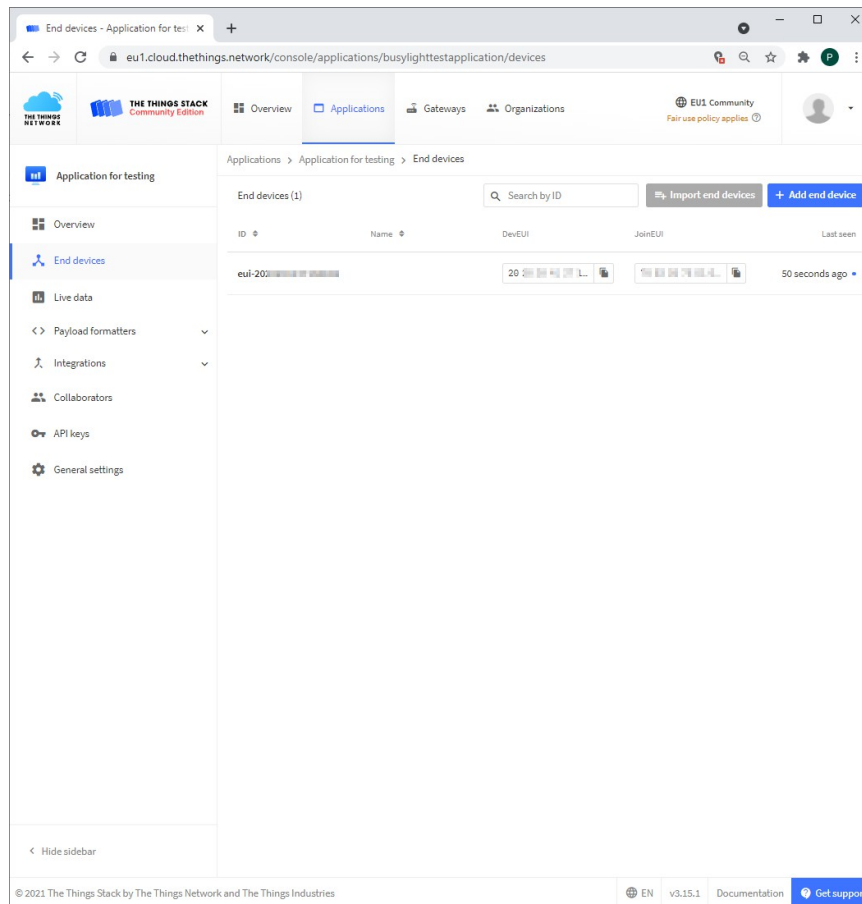
End device ID *
eui-20...
This value is automatically prefilled using the DevEUI

After registration
☒ View registered end device
☐ Register another end device of this type

Register end device

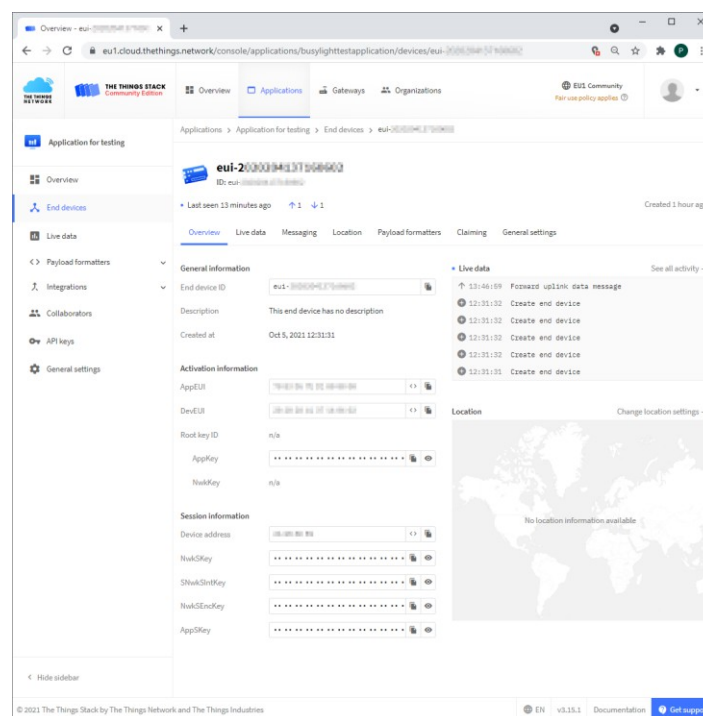
© 2021 The Things Stack by The Things Network and The Things Industries EN v3.15.1 Documentation Get support

Nach der Registrierung und wenn alles in Ordnung ist, wird das Gerät als verbunden angezeigt:

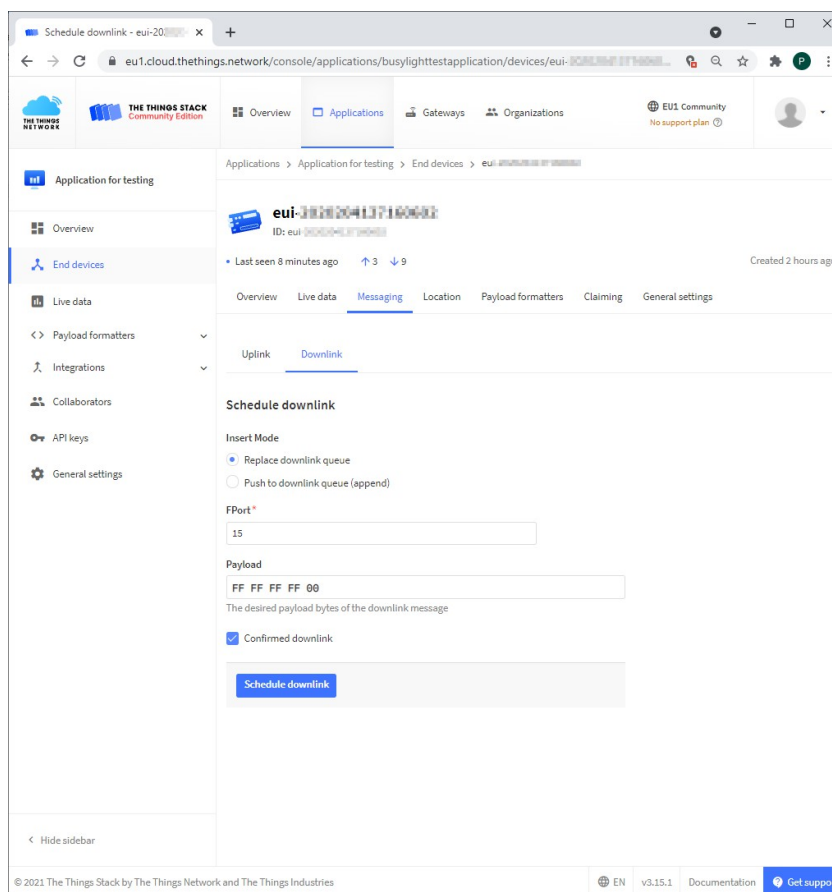


Testen des Busylight

Sie können Ihr verbundenes Busylight testen, indem Sie die Downlink-Nutzlast von der TTN-Konsole senden. Öffnen Sie dazu bitte die Seite mit den Gerätedetails.



Klicken Sie hier bitte auf „Messaging – Downlink“:



Der Fport muss auf 15 eingestellt werden. Die Nutzlast muss in Hexadezimalschreibweise angegeben werden. Bitte lesen Sie dazu das Kapitel über das Hardware-Nutzlastformat.

In diesem Fall leuchtet die Busylight-Anzeige dauerhaft weiß.

TTN-Nutzdatenformatierer für die Busylight

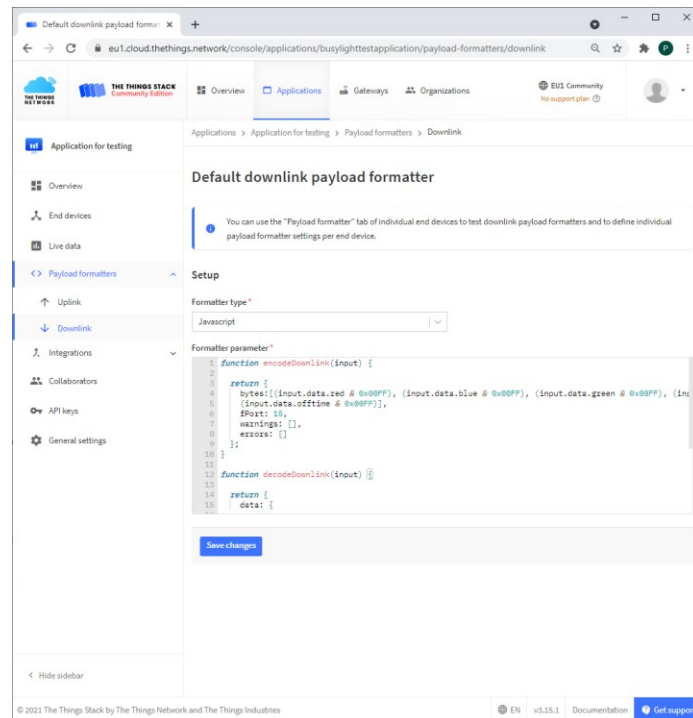
Ein Payload-Formatierer ermöglicht es Ihnen, lesbare JSON-Zeichenfolgen zur Steuerung der Busylight zu senden, anstatt die rohen Hardware-Bytes, wie im Kapitel „Busylight LoRa Hardware Payload Format“ beschrieben.

Um den Payload-Formatierer einzufügen, öffnen Sie die Anwendung und klicken Sie auf das Menü „Payload-Formatierer – Downlink“.

.

Bitte wählen Sie „Javascript“ als Formatierungstyp und geben Sie die Quelle in das Formatierungsparameterfeld ein

.



Hier ist der vollständige Formatierer:

```
function encodeDownlink(input) {
    return {
        bytes: [(input.data.red & 0x00FF), (input.data.blue & 0x00FF), (input.data.green &
0x00FF), (input.data.ontime & 0x00FF),
        (input.data.offtime & 0x00FF)],
        fPort: 15,
        warnings: [],
        fehler: []
    };
}

function decodeDownlink(input) {
    return {
        Daten: {
            rot: input.bytes[0], grün:
            input.bytes[2], blau:
            input.bytes[1], ontime:
            input.bytes[3], offtime:
            input.bytes[4]
        },
        warnungen: [],
        fehler: []
    };
}
```

Zum Dekodieren der Uplink-Nachrichten können Sie diesen Uplink-Dekoder verwenden:

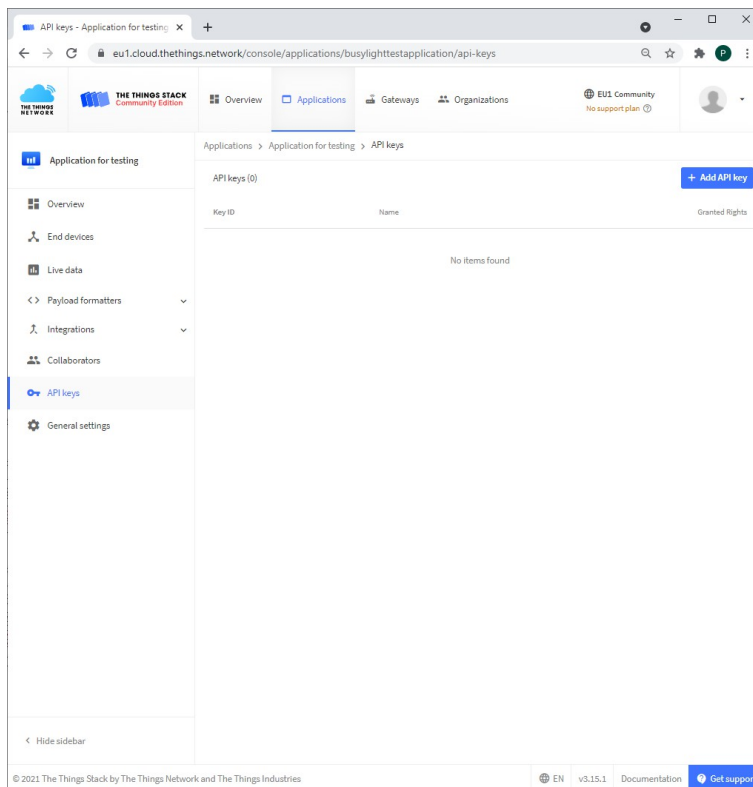
```
function decodeUplink(input) { if
(input.bytes.length == 24)
{
  return {
    data: {
      RSSI: byteArrayToLong(input.bytes, 0),
      SNR: byteArrayToLong(input.bytes, 4),
      messages_received: byteArrayToLong(input.bytes, 8),
      messages_send: byteArrayToLong(input.bytes, 12), lastcolor_red:
input.bytes[16],
      lastcolor_blue: input.bytes[17],
      lastcolor_green: input.bytes[18],
      lastcolor_ontime: input.bytes[19],
      lastcolor_offtime: input.bytes[20], sw_rev:
input.bytes[21],
      hw_rev: input.bytes[22], adr_state:
input.bytes[23]
    },
    warnungen: [],
    Fehler: []
  };
}
else
{
  return {data: { bytes:
input.bytes,
  },
  warnungen: [],
  fehler: []
}
}
}

byteArrayToLong = function(/*byte[]*/byteArray, /*int*/from) {
  return byteArray[from] | (byteArray[from+1] << 8) | (byteArray[from+2] << 16) |
(byteArray[from+3] << 24);
};
```

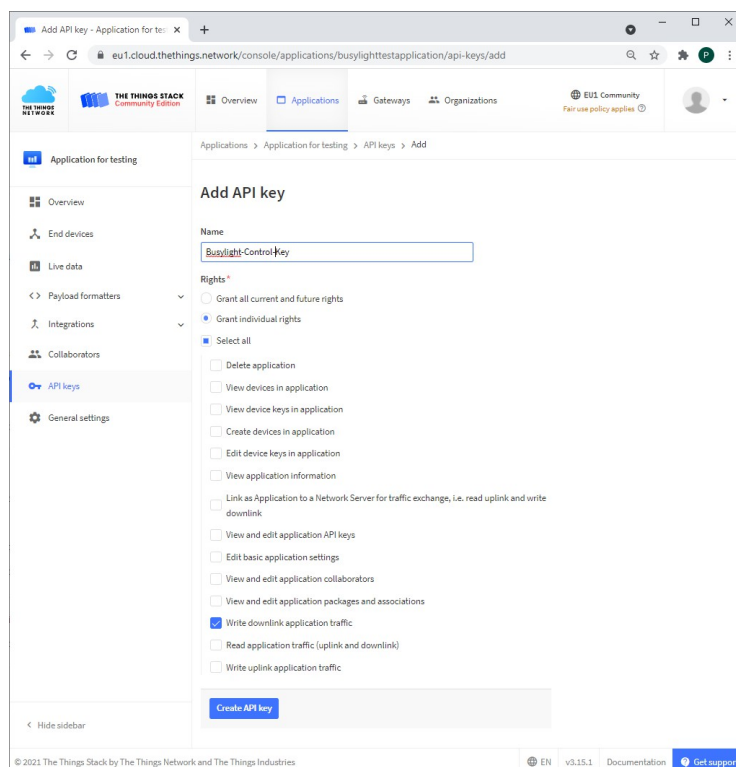
Steuerung der Busylight mit HTTP-Anfragen

Wenn Sie die Busylight mit HTTP-/HTTPS-Anfragen steuern möchten, müssen Sie einen API-Schlüssel erstellen.

Um einen Schlüssel zu erstellen, öffnen Sie die Anwendung und klicken Sie auf „API-Schlüssel“.

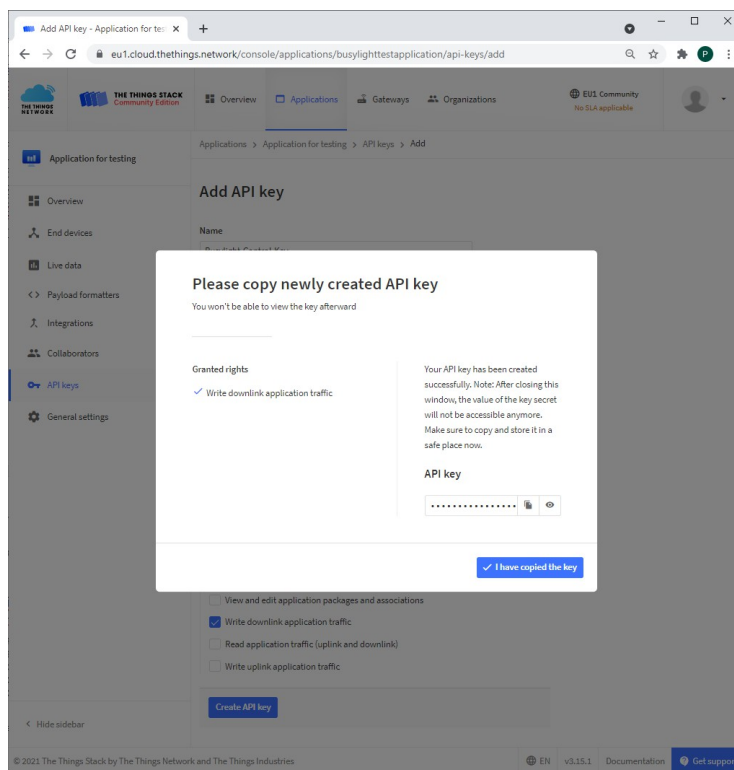


Klicken Sie dann auf „+ API-Schlüssel hinzufügen“.



Geben Sie einen aussagekräftigen Namen ein und erteilen Sie die Berechtigung zum Schreiben von Downlink-Datenverkehr.

Nach dem Speichern müssen Sie den Schlüssel für Ihre Anwendung kopieren. Bitte beachten Sie, dass Sie nicht ein zweites Mal darauf zugreifen können!



Sie können nun Ihr Busylight mit einer HTML-Anfrage steuern. Bitte ändern Sie die gelben Textteile entsprechend Ihren Anforderungen.

Sie müssen eine POST-Anfrage an diese URI senden:

`https://your_ttn_server/api/v3/as/applications/your_application_id/devices/your_device_id/down/push`

Sie senden diesen Body:

```
{
  „downlinks”: [{ „decoded_payload”:
    {
      „rot”: 0,
      „grün”: 0,
      „blau”: 255,
      „ontime”: 255,
      „offtime”: 0
    },
    „f_port”: 15
  }]
}
```

Und Sie müssen folgende Header senden:

Authorization: Bearer `your_api_key`
Content-Type: application/json
User-Agent: `busylight/v1`

Hier ist ein PowerShell-Beispiel, das Busylight auf durchgehend blau umschaltet:

```
$cmd2 = '{
  "downlinks": [{
    "decoded_payload": {
      "red": 0,
      "green": 0,
      "blue": 255,
      "ontime": 255,
      "offtime": 0
    },
    "f_port": 15
  }
}]
}',
```

Free-form Snip

```
Invoke-WebRequest -Method 'POST' `
-Headers @{"Authorization" = "Bearer NNSXS.EC:PH08HPLAKH9C:0PS:PH0TQWAPQ00T8X8BLLTY,ACNAG8888X147:EMTPF08:8RABL:3J0K:IN0P:NAG0P4:0VLC8:EX00XA";
           "Content-Type" = "application/json";
           "User-Agent" = "busylight/v1"}
-Body $cmd2
-Uri https://eul.cloud.thethings.network/api/v3/as/applications/xxxxxxxxxxxx/devices/eui-2000000010000000/down/push
```

Busylight LoRa Hardware-Nutzlastformat

Das Busylight erwartet eine 5-Byte-Binär-Nutzlast zum Umschalten der Farben.

Byte 0: Intensität der roten Farbe (0..255)

Byte 1: Blau Farbintensität (0..255)

Byte 2: Grün Farbintensität (0..255)

Byte 3: Schritte ein (0..255)

Byte 4: Aus-Schritte (0..255)

Beispiel für blaues statisches Licht:

```
Byte[0]=0
Byte[1]=255
Byte[2]=0
Byte[3]=255
Byte[4]=0
```

Die Hexadezimalform lautet: 00FF00FF00

Für den Betrieb von TTN [https](https://www.thingsnetwork.io/) muss bei Verwendung von `frm_payload` zur Angabe der Nutzlast für das Endgerät das Byte-Array als Base64-kodierte Zeichenfolge gesendet werden.

Bei Verwendung des Nutzlastformatierers können Sie die Werte mit einer JSON-Zeichenkette angeben:

```
{
  „downlinks”: [ [ „decoded_payload”:
    {
      „red”: 0,
      „green”: 0,
      „blue”: 255,
      „ontime”: 255,
      „offtime”: 0
    },
    „f_port”: 15
  ]
}
```