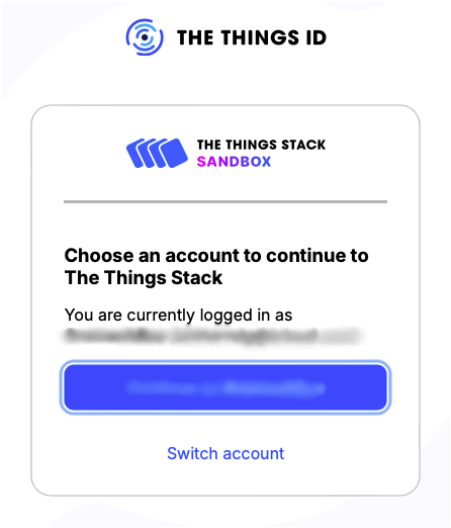
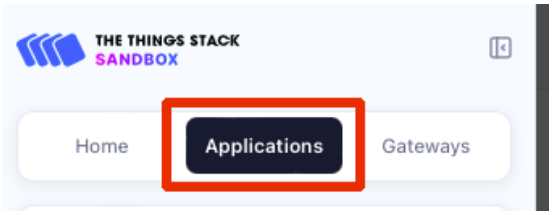
Konfiguration Lora TTN

Login bei:

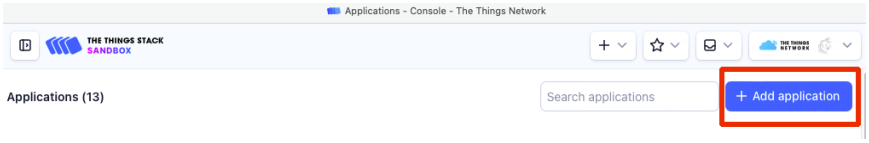
https://id.thethingsnetwork.org



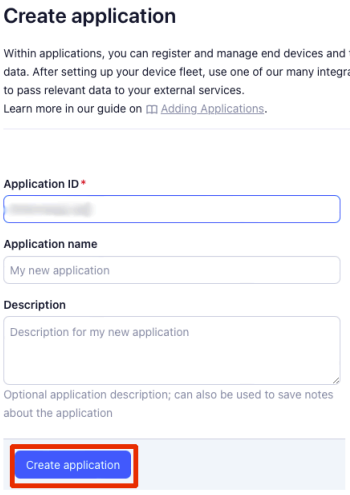
In Bereich Applications wechseln



Auswahl 'Add application'



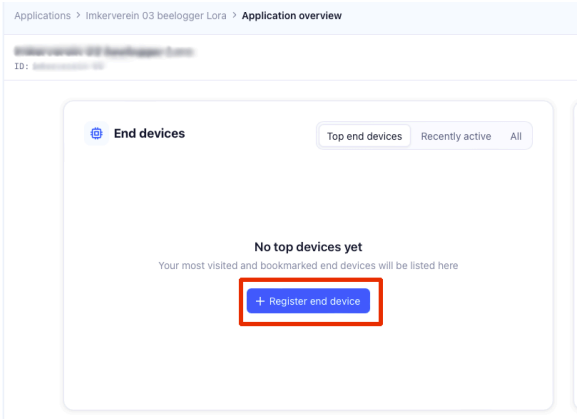
Felder ausfüllen



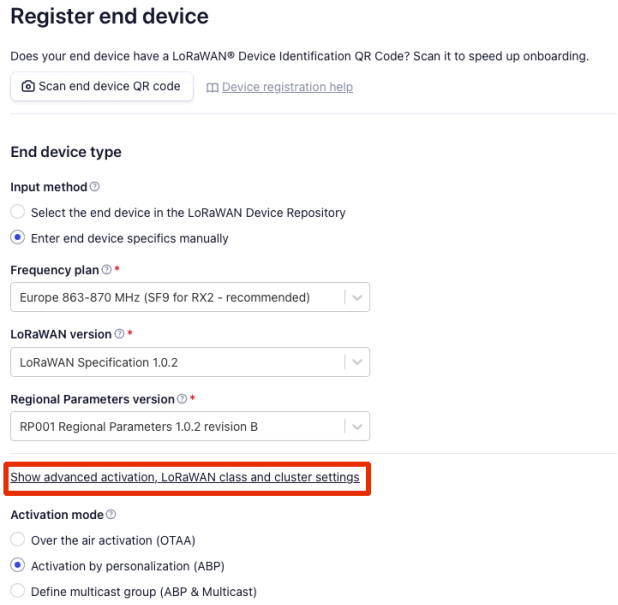
Application-ID wird von Things Stack vergeben und kann anschließend nicht geändert werden. Nur Kleinbuchstaben, ohne Leerschlag.

'Application name' kann später noch geändert werden.

End Device registrieren



Erweiterte Registrierung - Teil 1

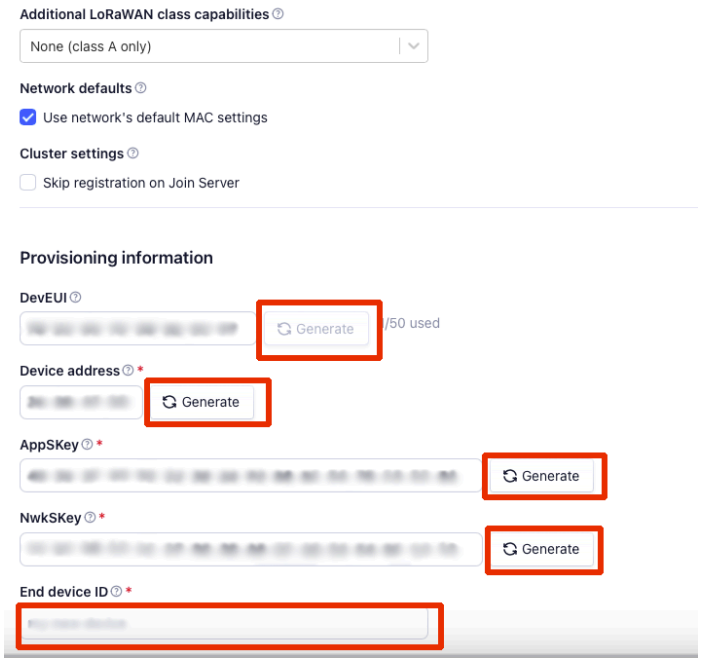


Wie oben ausfüllen

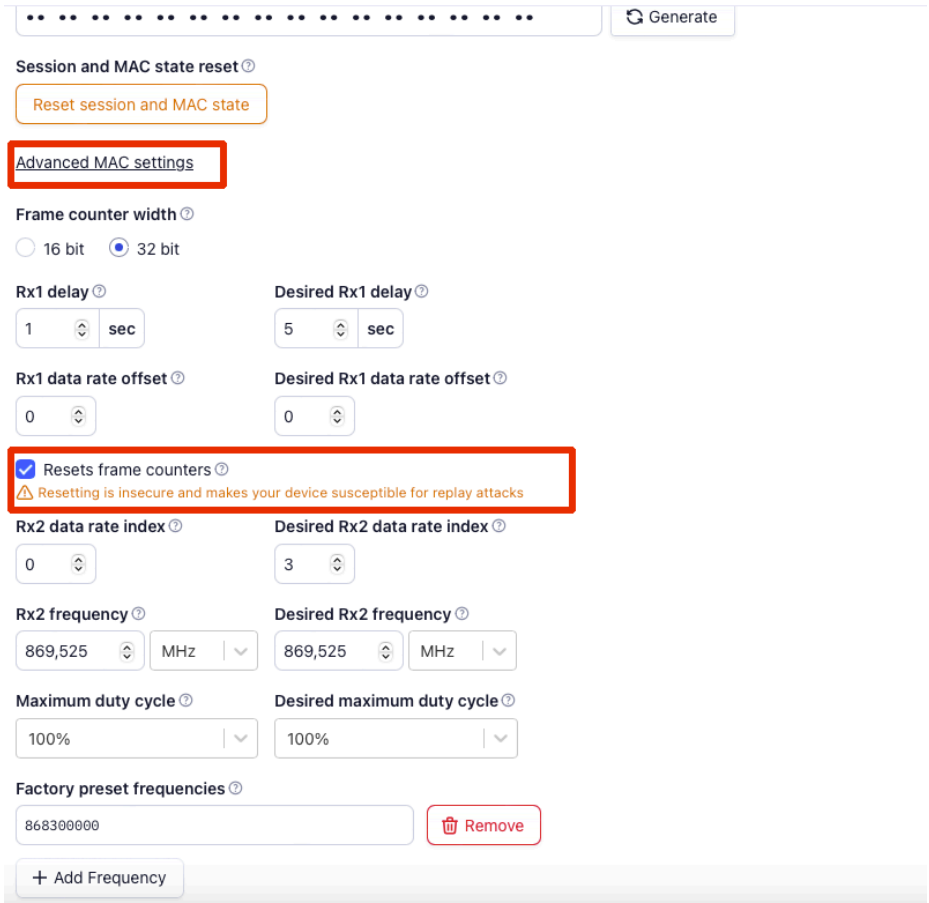
'Show advanced activation, LoRaWan class and cluster settings anwählen

Erweiterte Registrierung - Teil 2

DevEUI, Device address, AppSKey und NwkSKey generieren
End device ID benennen **(kann später nicht geändert werden)**.



In der Zeile rechts von ’Device overview’ ’Settings’ danach ’Advanced MAC Settings’ anwählen



Bei 'Factory preset frequencies' die Frequenzen des Gateways eintragen, mit dem sich die End Device verbinden soll => siehe Liste von 'Multi_LORA.ino'

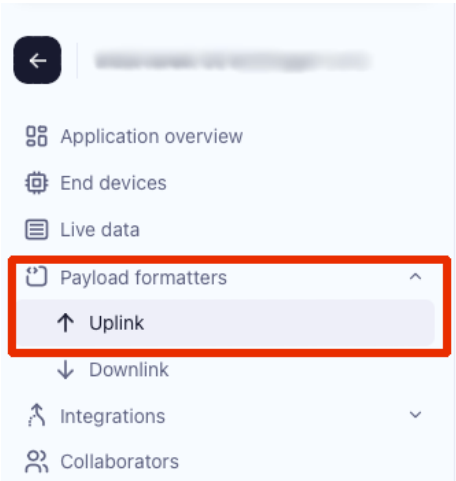
```
#if LORA_CHANNELS == 1
  LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(1, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(2, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band

#elif LORA_CHANNELS == 3
  LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band

#else
  LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENTI);      //
g-band
  LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7),  BAND_CENTI);      //
g-band
  //  LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK,  DR_FSK),
BAND_MILLI);      // g2-band
```

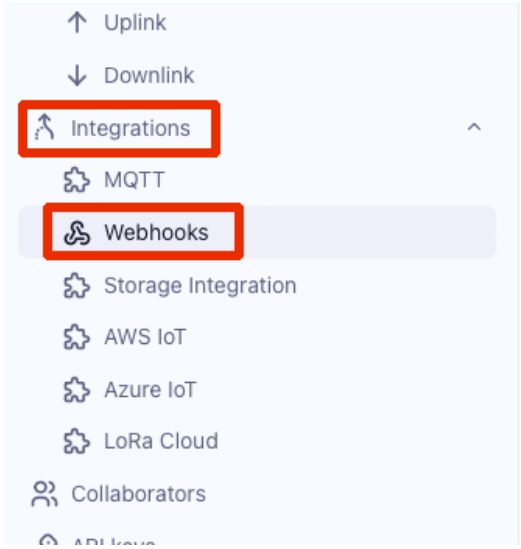
In der Seitenleiste Payload formatiert -> Uplink anwählen,

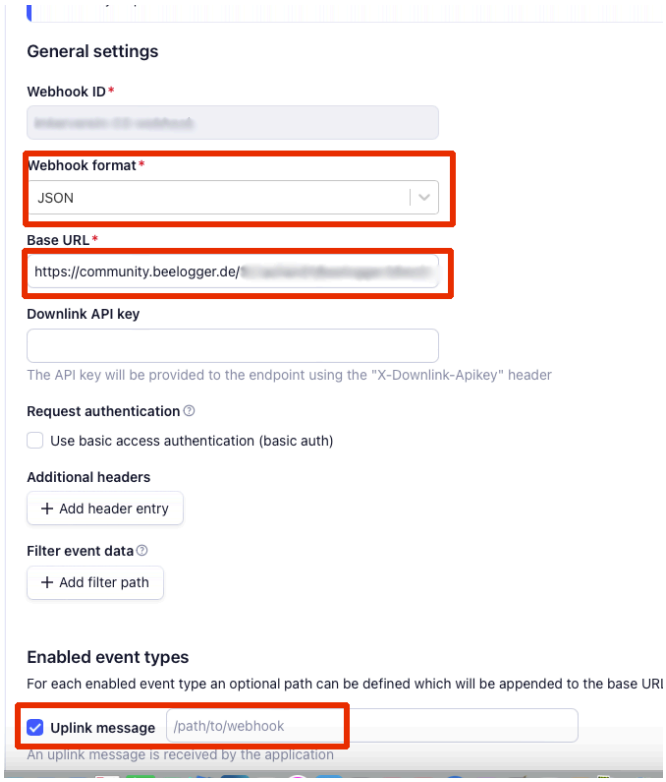
Im Menu '**Formatter type**' '**Custom Javascript formatter**' anwählen
Den Inhalt von '**Formatter code'** ersetzen mit der Datei '**payloadformatter_220227.js**' aus dem Sketchordner



Save changes nicht vergessen…

Um die Daten an den beelogger communityServer zu übertragen ist ein ’Webhook’ einzurichten. Hierzu ist in der linken Menüleiste unter ’Integrations’ der Unterpunkt ’Webhooks’ auszuwählen, anschließend ’Add webhook’ und ’Cutrom webhook’





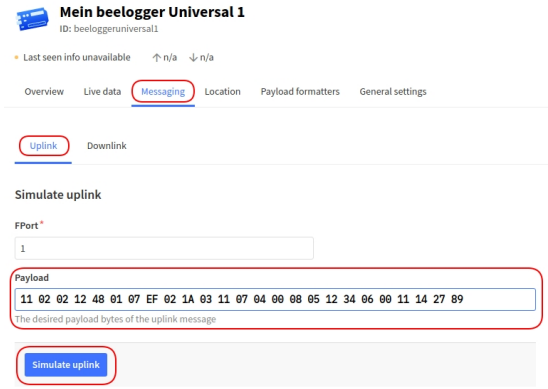
Als Webhook ID bietet es sich an, das beelogger Verzeichnis aus der URL zu wählen z.B. beelogger 1, beelogger 2, beelogger 3, … oder Duo1, Duo2, Duo3, usw..

Für das “Webhook format” ist “JSON” einzustellen.
Als “Base URL” ist der Link zur jeweiligen beelogger_log.php anzugeben.
Beispiele:

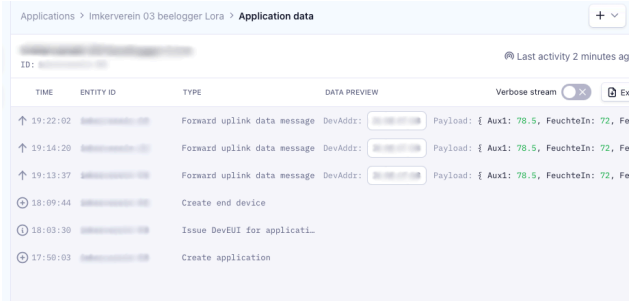
System mit einer Waage:
https://community.beelogger.de/UserName/beelogger1/beelogger_log.php?Passwort=deinpasswort&LORA=1

Zurück zu ’Device overview’ = > ’End device info’ => ’Messaging’ => ’Simulate uplink’ kann unter Payload die Musterdatei aus der beelogger-Website eingegeben werden.

11 02 02 12 48 01 07 EF 02 1A 03 11 07 04 00 08 05 12 34 06 00 11 14 27 89



Sind alle Angaben korrekt gemacht, zeigt sich der uplink unter ’Live Data’



Damit ist die Konfiguration abgeschlossen.