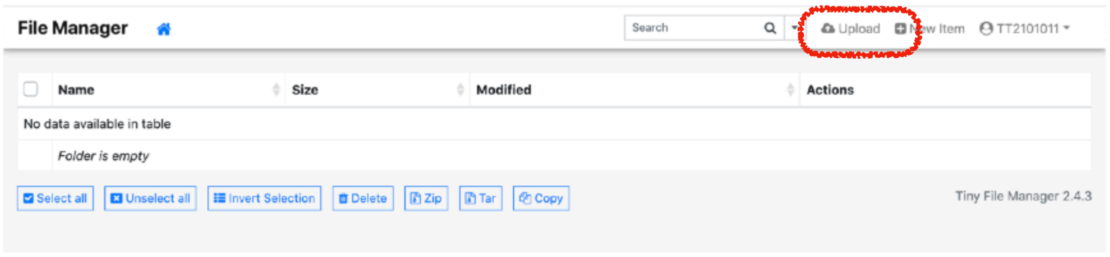
Webhook / HTTP Integration

1. Login to your test server space

See the resources doc for the web address. Your login ID & password will have been sent to you by email from someone at TTN

2. Upload the template

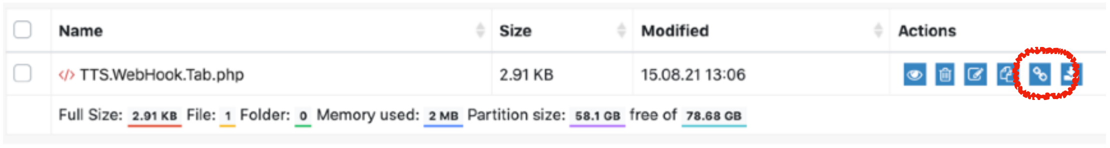
From the GitHub zip file, open the WebHook-to-Tab-PHP folder and locate the TTS.WebHook.Tab.php.



	Name	Size	Modified	Actions
No data available in table				
	Folder is empty			

Click the Upload button in the top right of the file manager browser window.

And either drag & drop the file or click in the box to use a file browser to choose the file. Click the Back or the Home (house) button to return to the list. The uploaded file will be listed.



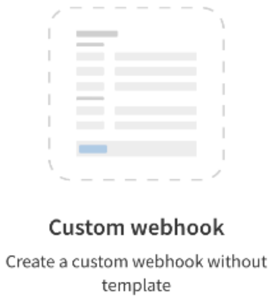
	Name	Size	Modified	Actions
	TTS.WebHook.Tab.php	2.91 KB	15.08.21 13:06	
	Full Size: 2.91 KB File: 1 Folder: 0 Memory used: 2 MB Partition size: 58.1 GB free of 78.68 GB			

To test it, click the Direct link button.

This will show you the version as plain text in your browser. Note the address of the script.

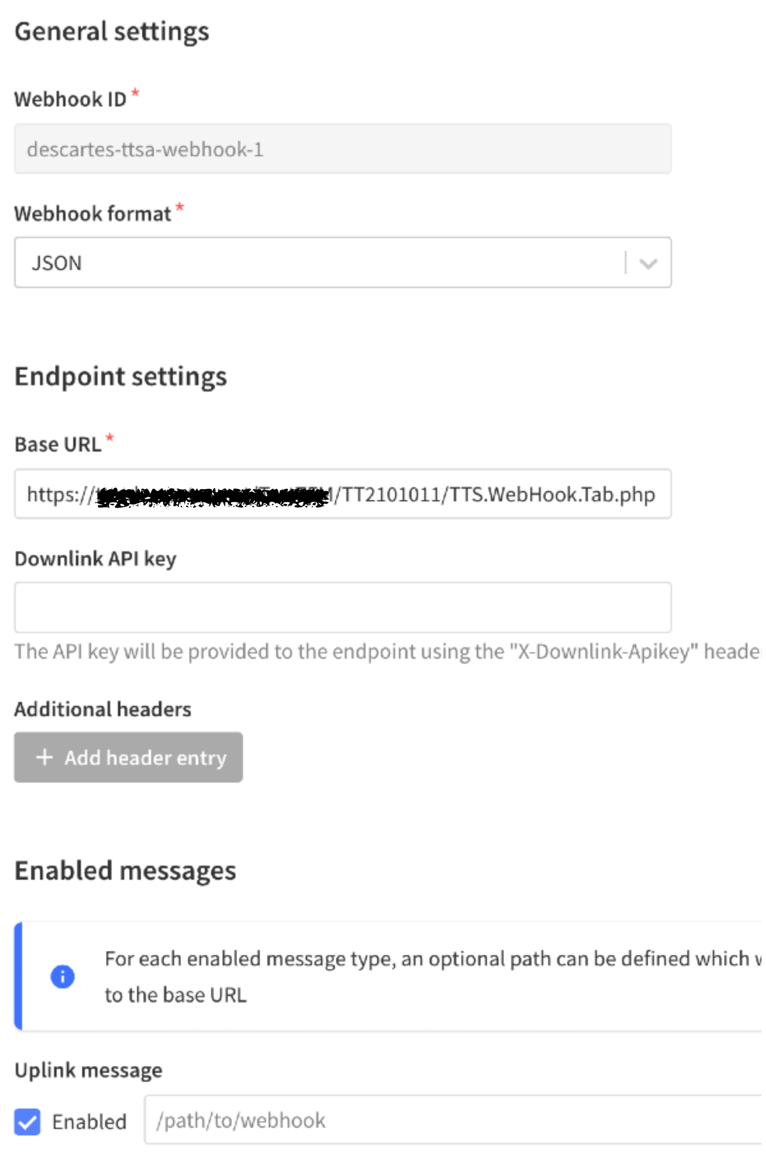
3. Setup the webhook integration

Go to the MQTT integration in the TTS console and click the + Add webhook button. Scroll down to the bottom of the different options - we are setting up a Custom webhook.



- You will need to create a Webhook ID.
- Choose JSON for the Webhook format.
- Copy the web address of the script from the file manager direct link button above. Paste this in to the Base URL.
- Click the Uplink message Enabled check box but do not enter anything in the box next to it.

Your setup should look like the screen shot on the next page.


General settings

Webhook ID *

descartes-ttsa-webhook-1

Webhook format *

JSON

Endpoint settings

Base URL *

https://[redacted]/TT2101011/TTS.WebHook.Tab.php

Downlink API key

The API key will be provided to the endpoint using the "X-Downlink-Apikey" heade

Additional headers

Enabled messages

For each enabled message type, an optional path can be defined which v
to the base URL

Uplink message

☑ Enabled /path/to/webhook

Scroll down to the Add webhook button to save the settings.

You can now trigger an uplink on your device or from the console.

In the browser file manager, if you refresh or click on the home/house symbol it will show you the data files that have been generated.

- The one with the date in reverse will hold all uplinks for that date.
- The one with just the application name will hold all uplinks for that app.
- The one with a combination of app name & device name holds all uplinks for that device.

You can view the contents of the files by clicking on the eye icon or clicking on the file name.

You can copy the file contents from the browser & paste straight in to a spreadsheet.

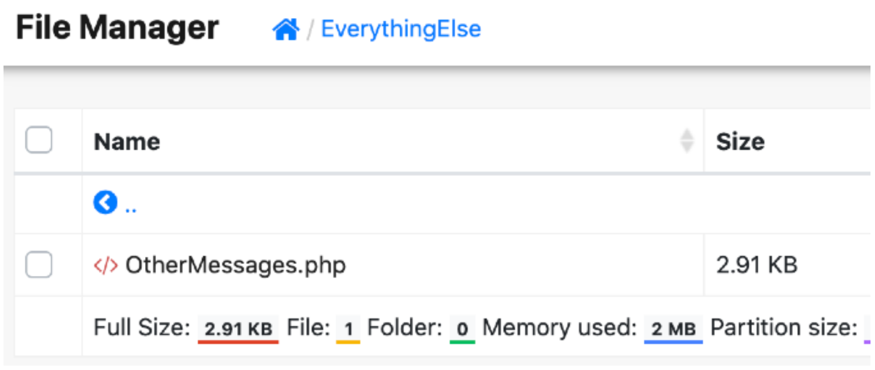
TTS Documentation

https://www.thethingsindustries.com/docs/integrations/webhooks/

Things to try

Filtering the messages

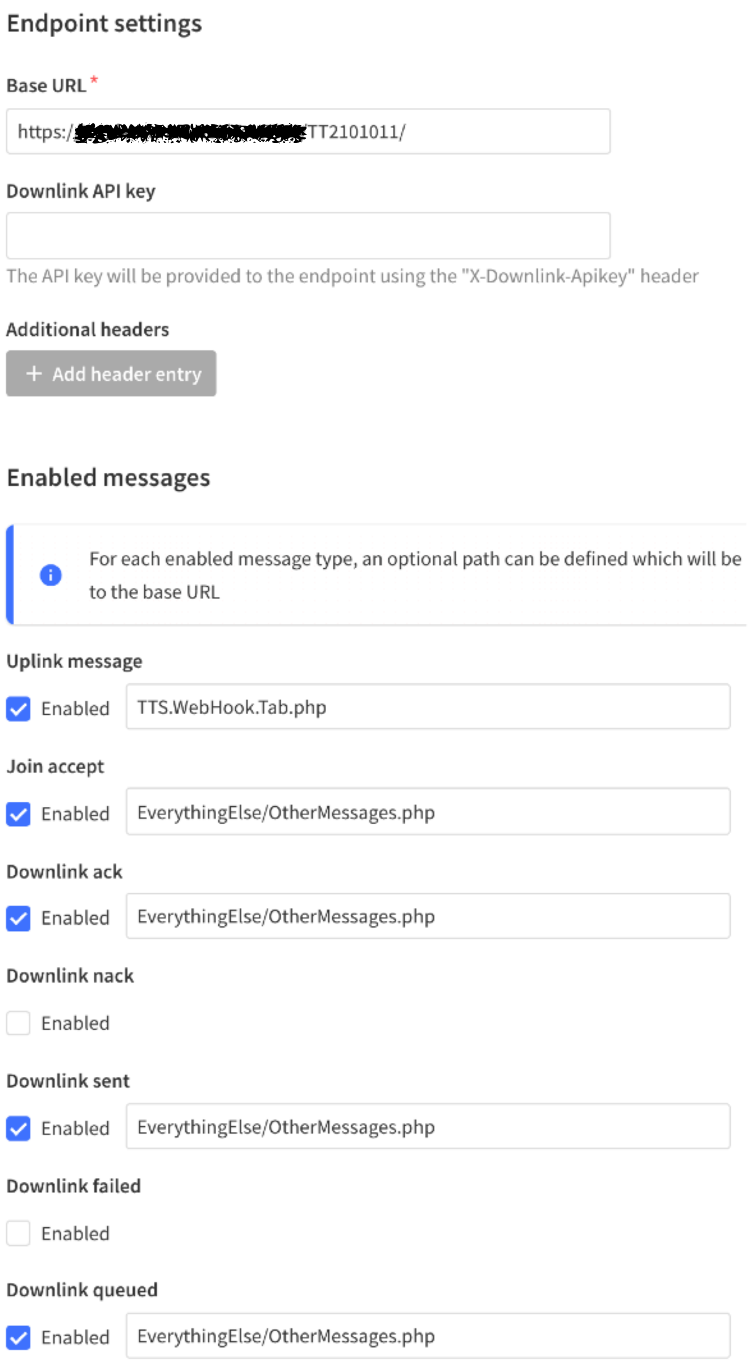
The principal focus is on uplinks. But it can be useful to log the other activity for a device without cluttering up the uplink data. One simple way to do this is to direct the webhook to a copy of the script in a sub-folder - this can be a catch all for all the other message types.


File Manager / EverythingElse

	Name	Size
	..	
☐	OtherMessages.php	2.91 KB

Full Size: 2.91 KB File: 1 Folder: 0 Memory used: 2 MB Partition size:

Create a directory/folder in the file manager and upload a new copy of the TTS.WebHook.Tab.php. Save confusion by renaming it, either before you upload or after. Change the Base URL to point to the server and then set the locations for each message you want to capture.


Endpoint settings

Base URL *

https:/[redacted]TT2101011/

Downlink API key

The API key will be provided to the endpoint using the "X-Downlink-Apikey" header

Additional headers

Enabled messages

For each enabled message type, an optional path can be defined which will be to the base URL

Uplink message
✓ Enabled TTS.WebHook.Tab.php

Join accept
✓ Enabled EverythingElse/OtherMessages.php

Downlink ack
✓ Enabled EverythingElse/OtherMessages.php

Downlink nack
☐ Enabled

Downlink sent
✓ Enabled EverythingElse/OtherMessages.php

Downlink failed
☐ Enabled

Downlink queued
✓ Enabled EverythingElse/OtherMessages.php

You create a new item using the button top right called New Item.

You can then upload as you did before.

The third icon on the listing with a pencil will allow you to change the name.

The Base URL has been altered to point to the web server directory.

Note the trailing / in the Base URL.

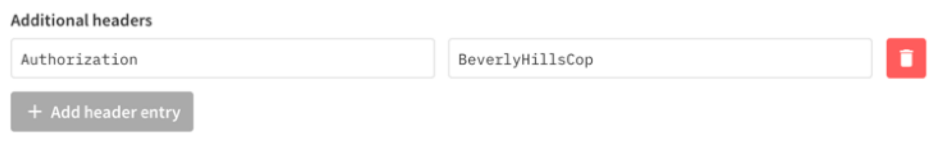
The Uplink message has the name of the original script in it.

The other messages that have been chosen to log have the directory name & script name entered for each of them.

You can have a script for each message if you wish.

Use your own authorisation key

People put web addresses everywhere, Google gets everywhere - by adding your own header key which you can check for, you can prevent web crawlers or naughty people from trying to post information to your web hook.



Authorization: BeverlyHillsCop

Add a new entry in to your web hook entry and then you will be able to check for it in the PHP code.

To get the all the headers from the HTTP submission and retrieve the header setup above we use:

```
$allHeaders = getallheaders();
$Authorization = $allHeaders[‘Authorization'];
```

You can then put in an if statement to check the value and respond accordingly.