

ThingsBoard (edición IoT open Tech)

Juan Félix Mateos Barrado

jfmateos@educa.madrid.org

juanfelixmateos@gmail.com

The Things Network Madrid

Marzo 2020

ThingsBoard

Plataforma IoT open source con las siguientes características:

- Funciona en la nube o on-premises (local)
- Community Edition (CE) y Professional Edition (PE)
- Multi-tenant (instancias diferenciadas para cada tenant sobre una misma instalación)
- Aprovisionamiento de dispositivos, activos y clientes con relaciones entre ellos
- Almacenamiento de datos/telemetrías (SQL y NoSQL)
- Visualización mediante dashboards/widgets
- Motor de reglas tipo Node-RED

ThingsBoard PE

SUBSCRIPTION PLANS

All subscription plans include unlimited customers, dashboards, integrations, api calls, datapoints & messages

Plan	Description	Price	Support	White-labeling
Maker	To become familiar with ThingsBoard PE features	\$ 10 /MONTH	Community support	—
Prototype	The subscription is designed for PoCs and prototyping	\$ 99 /MONTH	Community support	White-labeling
Startup	For upcoming IoT Unicorns	\$ 199 /MONTH	Email support within 36 hours	White-labeling
Business	For established mid-level market players with mature IoT approach	\$ 299 /MONTH	Email support within 24 hours	White-labeling
Enterprise	Consider yourself a Fortune 500 company in the field? Subscribe this plan	\$ 500 /MONTH	Email support within 12 hours	White-labeling

[GET YOUR LICENSE >](#)

Community support ⓘ

White-labeling

ThingsBoard: Programa de la sesión (1/2)

1. Crear un dispositivo en ThingsBoard
 1. Probar con Postman (o curl) que el dispositivo recibe telemetrías correctamente
 2. Crear una regla en ThingsBoard que permita convertir las telemetrías enviadas por The Things Network al formato de ThingsBoard
2. Crear una aplicación y nodo en The Things Network
 1. Configurar el formato de carga de pago Cayenne LPP
 2. Configurar la integración con ThingsBoard
 3. Simular el envío de telemetrías desde The Things Network

ThingsBoard: Programa de la sesión (2/2)

1. Crear en ThingsBoard una integración con IFTTT que nos envíe una notificación al móvil cuando la temperatura supere el umbral de 25º
2. Crear en ThingsBoard una integración con Telegram que nos envíe un mensaje a un chat privado cuando la temperatura supere el umbral de 25º

Crear un dispositivo en ThingsBoard

The screenshot shows the ThingsBoard web application interface. On the left, there is a sidebar with the following menu items:

- PÁGINA PRINCIPAL
- CADENAS DE REGLAS
- CLIENTES
- ACTIVOS
- DISPOSITIVOS 1
- VISTAS DE ENTIDAD
- LIBRERÍA DE WIDGETS
- PANELES
- REGISTROS DE AUDITORÍA

The main area is titled "Dispositivos" and shows a list with one item: "NodoPrueba" (checkbox selected) and "NODOPRUEBA". Below the list are four icons: a share icon, a user icon, a shield icon, and a trash bin icon. To the right of the list are three orange circular buttons labeled 1, 2, and +, each with a different icon: an upward arrow, a file, and a plus sign.

A large pink arrow points from the "NodoPrueba" entry in the list to the "Agregar Dispositivo" (Add Device) dialog box on the right.

The "Agregar Dispositivo" dialog box contains the following fields:

- Nombre*: NodoTTNv2
- Tipo de dispositivo*: NodoTTNv2
- Etiqueta
- Es puerta de entrada
- Descripción

At the bottom right of the dialog box are two buttons: "AGREGAR" (ADD) and "CANCELAR" (CANCEL).

Probar con Postman que el dispositivo recibe telemetrías (1/3)

URL: https://tb.iotopentech.io/api/v1/TU_TOKEN/telemetry

The screenshot illustrates the process of testing device telemetry reception using Postman and ThingsBoard.

Postman Interface: On the left, the Postman interface shows a history of requests. A red arrow points to the "New Request" button (a plus sign icon) in the top bar, which is highlighted with a red circle. The main area shows an "Untitled Request" with a POST method and the URL https://tb.iotopentech.io/api/v1/TU_TOKEN/telemetry. A red callout box with text in Spanish points to the URL field: "Aquí hay que poner el token de acceso del dispositivo, que está disponible en la ficha del dispositivo en ThingsBoard".

ThingsBoard Device Detail View: On the right, a modal window titled "Nuevo Dispositivo TTNv2" (New Device TTNv2) displays device details. It includes tabs for "DETALLES", "ELEMENTOS", "ÚLTIMA TELEMETRÍA", "ALARMAS", and "EVENTOS". Below the tabs, there are buttons for "HACER PÚBLICO EL DISPOSITIVO", "ASIGNAR AL CLIENTE", and "GESTIONAR CREDENCIALES". At the bottom, there are two buttons: "COPIAR ID DEL DISPOSITIVO" and "COPIAR TOKEN DE ACCESO", both highlighted with a red oval. A red arrow points from the "COPIAR TOKEN DE ACCESO" button in the modal to the "TU_TOKEN" placeholder in the Postman URL field.

Probar con Postman que el dispositivo recibe telemetrías (2/3)

Encabezado: Content-Type: application/json

Cuerpo: {"temperature_3": 22.5}

The image displays two instances of the Postman application interface. The top instance shows an 'Untitled Request' with a POST method to 'https://tb.iotopentech.io/api/v1/.../telemetry'. The 'Headers' tab is selected, and a single header 'Content-Type' is defined with the value 'application/json'. A red circle labeled '1' points to the 'Headers' tab. The bottom instance also shows an 'Untitled Request' with a POST method to the same URL. The 'Body' tab is selected, and a JSON payload is defined: '1 {"temperature_3": 22.5}'. A red circle labeled '2' points to the 'Body' tab, and another red circle labeled '3' points to the JSON payload itself. Both instances have a red circle labeled '4' pointing to the blue 'Send' button in the top right corner. The Postman interface includes tabs for Params, Authorization, Body, Pre-request Script, Tests, and Settings, along with a 'Comments (0)' and 'Save' button.

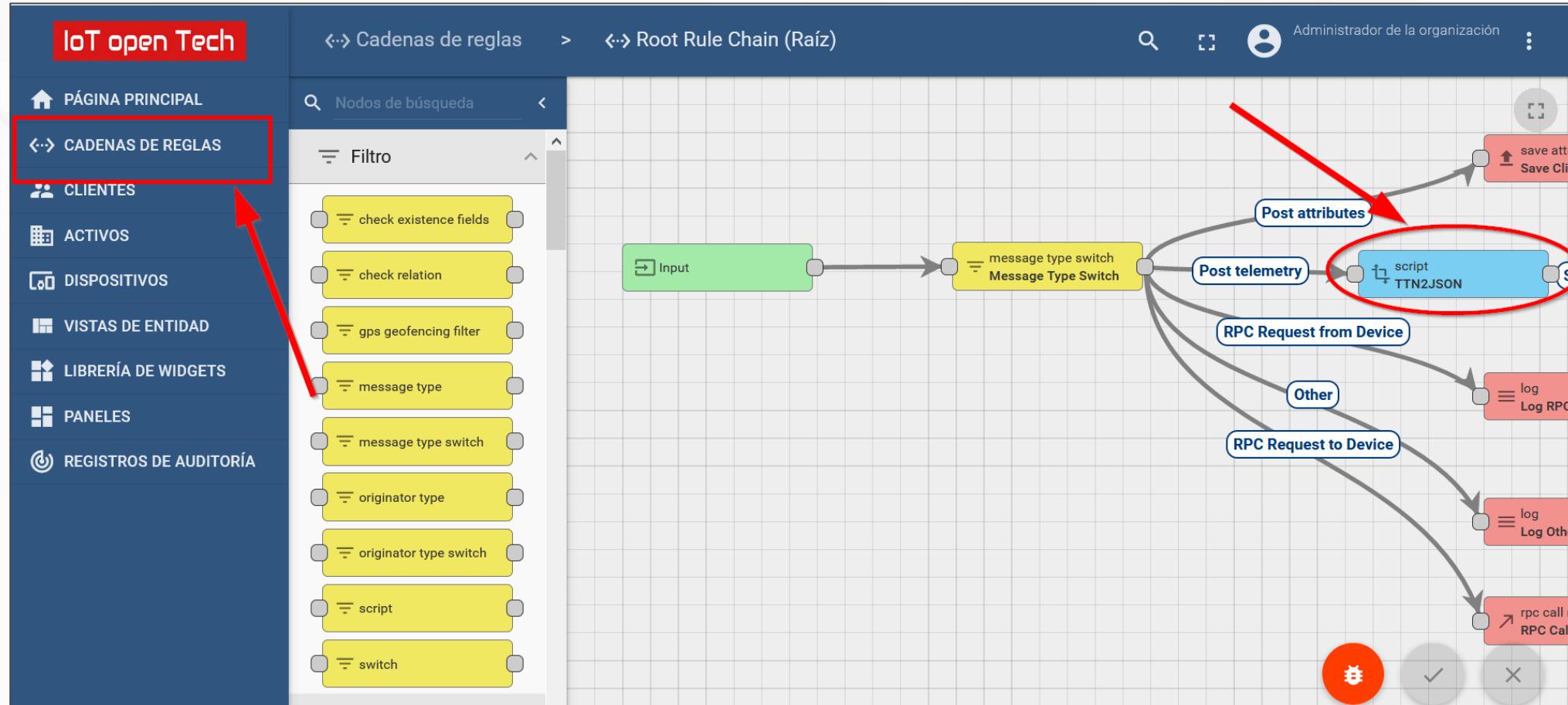
Probar con Postman que el dispositivo recibe telemetrías (3/3)



The screenshot shows a web-based interface for a device named "NODOTTNV2". The title bar says "Detalles del dispositivo". Below it, there are tabs: "DETALLES", "ATRIBUTOS", "ÚLTIMA TELEMETRÍA" (which is underlined in red), "ALARMAS", and "EVENTOS". A red circle with a white edit icon is positioned above the "EVENTOS" tab. The main content area is titled "Última telemetría". It contains a table with three columns: "Hora de la última actualización", "Clave", and "Valor". The first row shows the date and time "2020-03-06 10:29:10", the key "temperature_3", and the value "22.5". A red arrow points from the text "Probar con Postman que el dispositivo recibe telemetrías (3/3)" in the slide header down to the "Valor" column of this table row. The bottom of the table has pagination controls: "Page: 1", "Rows per page: 5", and "1 - 1 of 1".

Hora de la última actualización	Clave	Valor
2020-03-06 10:29:10	temperature_3	22.5

Crear una regla en ThingsBoard para convertir el formato de TTN al formato de ThingsBoard (1/2)



Crear una regla en ThingsBoard para convertir el formato de TTN al formato de ThingsBoard (2/2)

TTN2JSON

Transformación - script

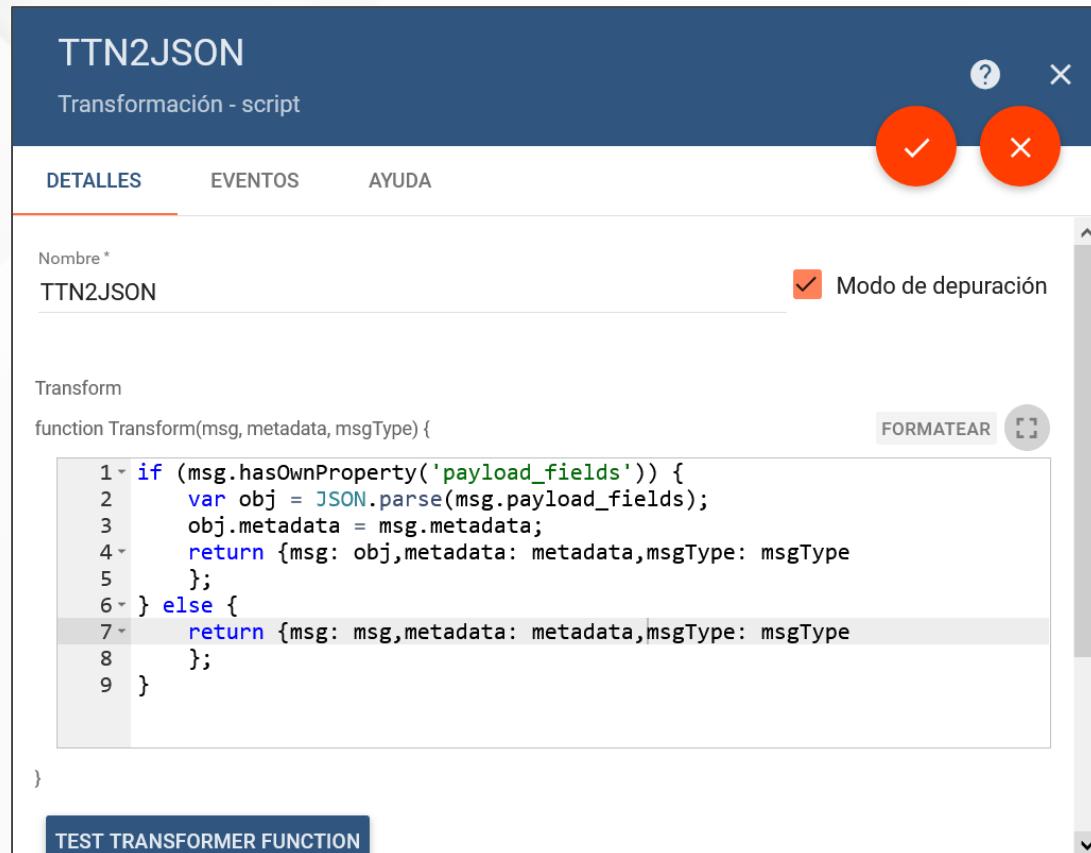
DETALLES EVENTOS AYUDA

Nombre * TTN2JSON Modo de depuración

Transform

```
function Transform(msg, metadata, msgType) {  
    if (msg.hasOwnProperty('payload_fields')) {  
        var obj = JSON.parse(msg.payload_fields);  
        obj.metadata = msg.metadata;  
        return {msg: obj,metadata: metadata,msgType: msgType};  
    } else {  
        return {msg: msg,metadata: metadata,msgType: msgType};  
    }  
}
```

TEST TRANSFORMER FUNCTION



```
if (msg.hasOwnProperty('payload_fields')) {  
    var obj = JSON.parse(msg.payload_fields);  
    obj.metadata = msg.metadata;  
    return {msg: obj,metadata: metadata,msgType: msgType};  
};  
} else {  
    return {msg: msg,metadata: metadata,msgType: msgType};  
};  
}
```

Crear una aplicación y nodo en The Things Network

The screenshot shows the The Things Network Console interface. At the top, there's a navigation bar with the logo 'THE THINGS NETWORK CONSOLE COMMUNITY EDITION', 'Applications', 'Gateways', 'Support', and a user profile 'jfmateos'. Below the navigation, the breadcrumb navigation shows 'Applications > jfmateos_nodo_v2_2 > Devices > jfmateos_nodo_v2_2_02'. On the right, there are three tabs: 'Overview' (which is selected), 'Data', and 'Settings'. The main content area is titled 'DEVICE OVERVIEW' and contains the following information:

- Application ID:** jfmateos_nodo_v2_2 (highlighted with a yellow box)
- Device ID:** jfmateos_nodo_v2_2_02
- Description:** ABP
- Activation Method:** ABP
- Device EUI:** A field with a copy icon and a hex value.
- Application EUI:** A field with a copy icon and a hex value.

Configurar el formato de carga de pago Cayenne LPP

The screenshot shows the The Things Network Console interface. At the top, there is a navigation bar with the logo "THE THINGS NETWORK CONSOLE COMMUNITY EDITION", a user profile icon "jfmateo", and links for "Applications", "Gateways", and "Support". Below the navigation bar, the breadcrumb navigation shows "Applications > jfmateos_nodo_v2_2 > Payload Formats". A red oval highlights this breadcrumb path. In the center, there is a tab bar with "Overview", "Devices", "Payload Formats" (which is highlighted with a red oval), "Integrations", "Data", and "Settings". Below the tab bar, the section title "PAYLOAD FORMATS" is displayed in blue. Underneath, the "Payload Format" section is shown with the subtitle "The payload format sent by your devices". A red oval highlights the "Cayenne LPP" option in the dropdown menu. At the bottom right, there are buttons for "Cancel" and "no changes to save".

Configurar la integración con ThingsBoard (1/2)

The screenshot shows the The Things Network Console interface. At the top, there's a navigation bar with 'THE THINGS NETWORK' logo, 'CONSOLE COMMUNITY EDITION', 'Applications', 'Gateways', 'Support', and a user profile 'jfmateo'. Below the navigation, the breadcrumb path 'Applications > jfmateos_nodo_v2_2 > Integrations' is highlighted with a red oval. The main content area displays three integration cards:

- EVRYTHNG** v2.6.0 by EVRYTHNG: An icon featuring a car, a smartphone, and various objects connected to a central cloud.
- HTTP Integration** v2.6.0 by The Things Industries B.V.: An icon with blue arrows pointing in opposite directions between two clouds.
- IFTTT Maker** v2.6.0 by The Things Industries B.V.: An icon with the IFTTT logo and a cloud symbol.

A large red circle highlights the 'HTTP Integration' card.

Configurar la integración con ThingsBoard (2/2)

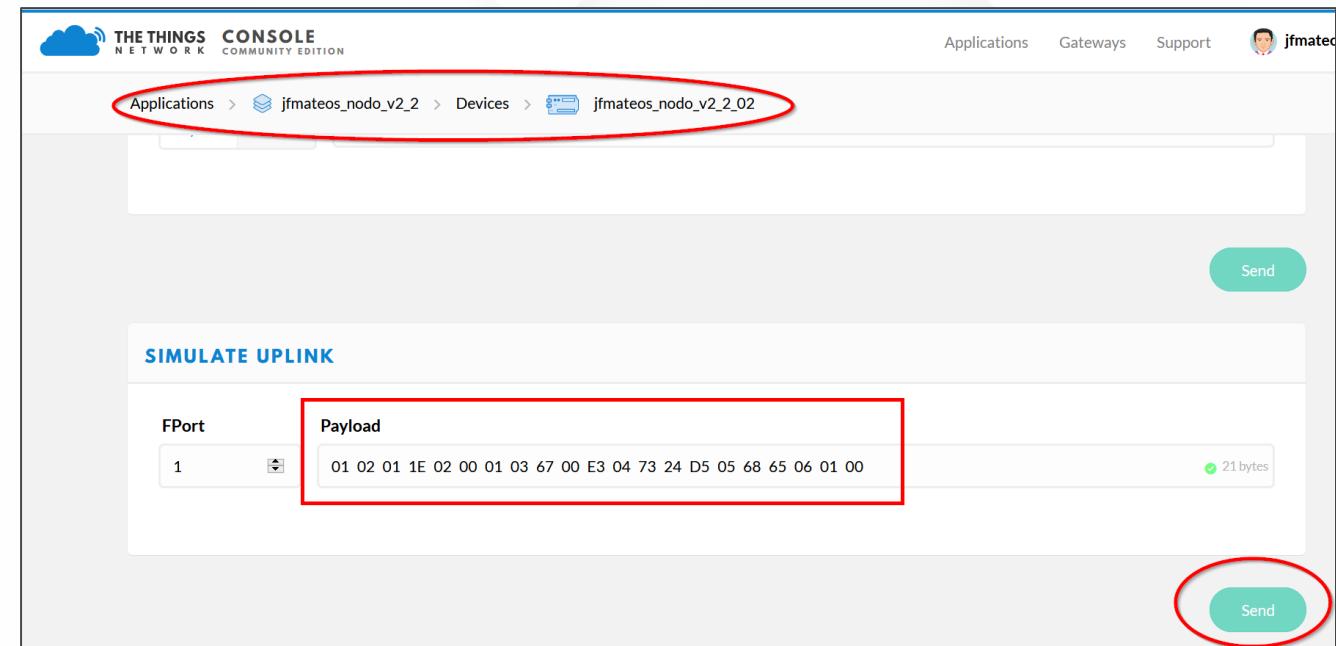
The screenshot shows the 'SETTINGS' page for an integration named 'ttn2tb' within an application. The configuration includes:

- Access Key**: A dropdown menu showing 'default key' with options for 'devices' and 'messages'. The 'devices' option is highlighted.
- URL**: The URL is set to `https://tb.iotopentech.io/api/v1/devices/12345678901234567890/telemetry`.
- Method**: The HTTP method is set to 'POST'.
- Authorization**: An empty input field for the Authorization header value.
- Custom Header Name**: The name of a custom header is set to 'Content-Type'.
- Custom Header Value**: The value of the custom header is set to 'application/json'.

Simular el envío de telemetrías desde The Things Network

0102011E020001036700E3047324D5056865060100

```
{ "analog_in_1": 2.86,  
  "barometric_pressure_4": 942.9,  
  "digital_in_2": 1,  
  "digital_out_6": 0,  
  "relative_humidity_5": 50.5,  
  "temperature_3": 22.7}
```



Integrar ThingsBoard con IFTTT

Para integrar ThingsBoard con IFTTT vamos a necesitar el **token del servicio Webhook de IFTTT**.

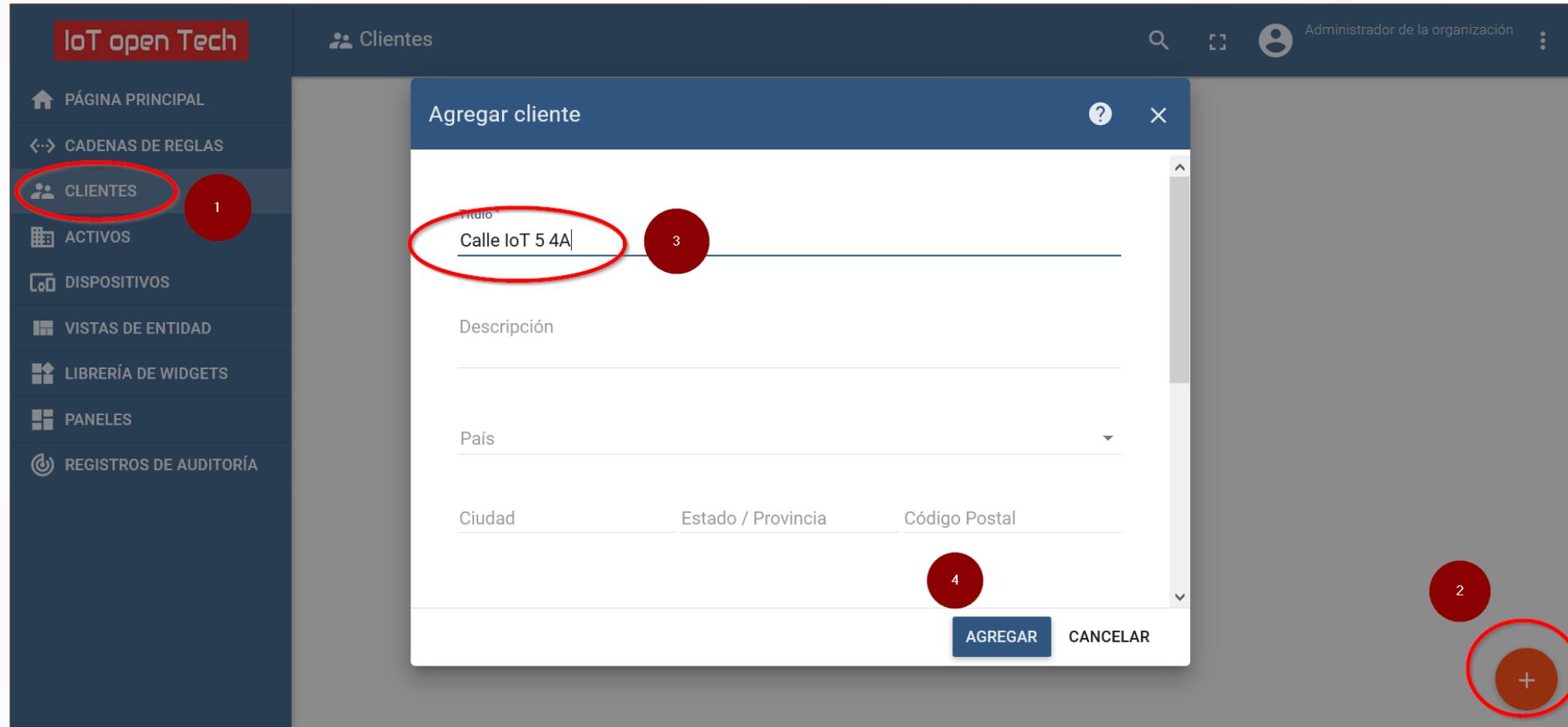
Este token lo almacenaremos como un **atributo**; todas las entidades de ThingsBoard (dispositivos, activos, clientes...) pueden tener atributos.

Como es probable que un mismo cliente tenga varios dispositivos y una sola cuenta de IFTTT, en lugar de asignar el atributo al dispositivo:

1. Crearemos un cliente en ThingsBoard
2. Asignaremos el dispositivo al cliente
3. Crearemos el atributo en el cliente

Integrar ThingsBoard con IFTTT

Crear un cliente en ThingsBoard



Integrar ThingsBoard con IFTTT

Asignar el dispositivo al cliente

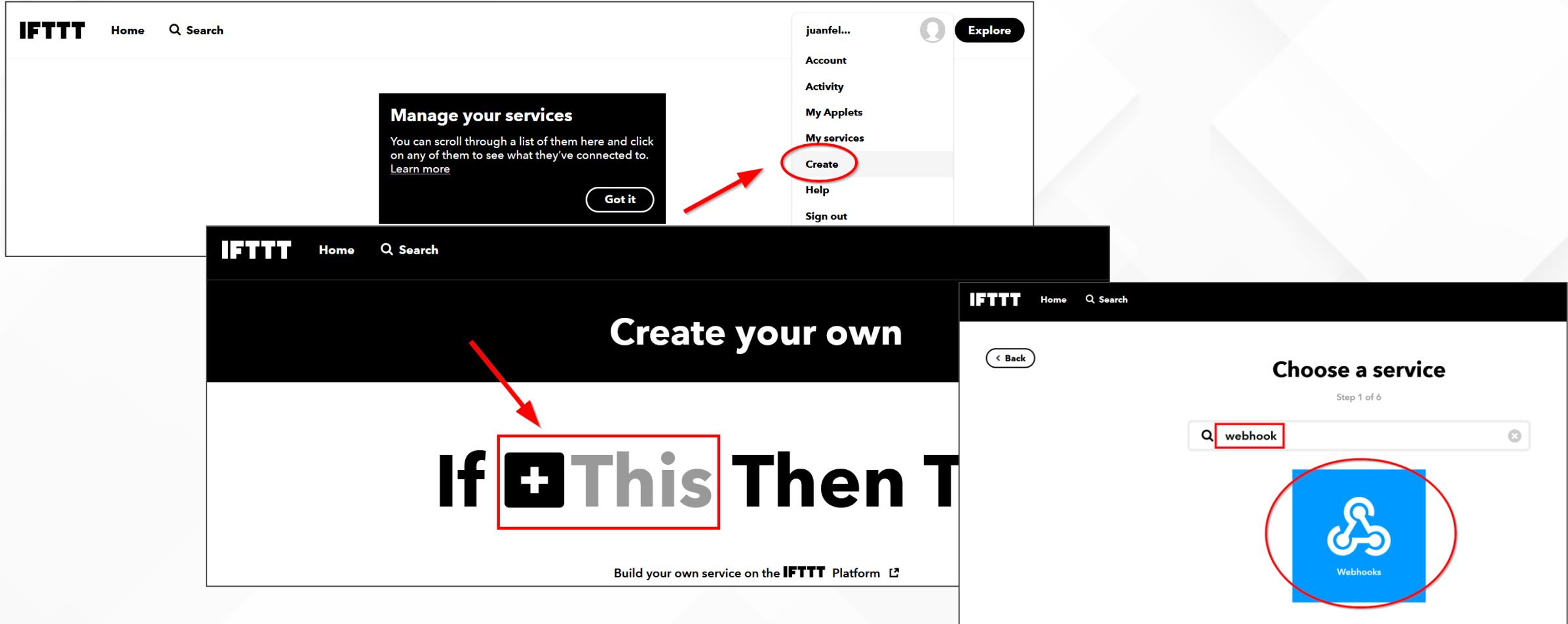
The screenshot displays two main interface sections of the IoT open Tech platform.

Left Panel: Shows a list of clients. One client entry is visible: "Calle IoT 5 4A" (Address) and "SIN DIRECCIÓN" (No address). Below the entry are several icons, with the fourth icon from the left (a gear-like symbol) highlighted with a red box.

Right Panel: Shows a modal dialog titled "Asignar Dispositivo(s) Al Cliente". The dialog contains a search bar and a list of devices. The device "NodoTTNv2" is selected, indicated by a checked checkbox and a red box around it. The "ASIGNAR" button at the bottom right of the dialog is also highlighted with a red box.

Integrar ThingsBoard con IFTTT

Crear el Applet en IFTTT (1/4)



Integrar ThingsBoard con IFTTT

Crear el Applet en IFTTT (2/4)

The image consists of three screenshots from the IFTTT web interface, illustrating the process of creating an applet to integrate ThingsBoard.

- Screenshot 1: Choose trigger**
The title "Choose trigger" is at the top. A "Back" button is in the top-left corner. A blue card titled "Receive a web request" is highlighted with a red circle. The text below it describes the trigger: "This trigger fires every time the Maker service receives a web request to notify it of an event. For information on triggering events, go to your Maker service settings and then the listed URL (web) or tap your username (mobile)".
- Screenshot 2: Complete trigger fields**
The title "Complete trigger fields" is at the top. It shows "Step 2 of 6". A blue card titled "Event Name" has the value "jfmateos_alertaTemperatura" highlighted with a red oval. Below it, a note says: "The name of the event will be "button_pressed" or "front_door_opened". A "Create trigger" button is at the bottom.
- Screenshot 3: Main IFTTT interface**
The IFTTT logo is at the top left. The navigation bar includes "Home" and "Search". A "Back" button is visible. The main area displays the "If This Then That" template, with the "That" part highlighted by a red box and a red arrow pointing to it.

Integrar ThingsBoard con IFTTT

Crear el Applet en IFTTT (3/4)

The image consists of two side-by-side screenshots from the IFTTT mobile application.

Left Screenshot: Choose action service

- Title:** Choose action service
- Step:** Step 3 of 6
- Search Bar:** A search bar with the text "Notifications" is highlighted with a red rectangle.
- Action Card:** A card for "Notifications" is shown, featuring a blue background with a white bell icon and the word "Notifications". This card is also circled with a red circle.

Right Screenshot: Choose action

- Title:** Choose action
- Step:** Step 4 of 6
- Back Button:** A "Back" button is visible at the top left.
- Action Options:** Two options are listed:
 - Send a notification from the IFTTT app:** Described as sending a notification to devices from the IFTTT app.
 - Send a rich notification from the IFTTT app:** Described as sending a rich notification to devices from the IFTTT app, including a title, image, and link.

Integrar ThingsBoard con IFTTT

Crear el Applet en IFTTT (4/4)

Complete action fields
Step 5 of 6

Message

`{{OccurredAt}}: Alarma temperatura {{Value1}} en {{Value2}}`

Add ingredient

Create action

Review and finish
Step 6 of 6

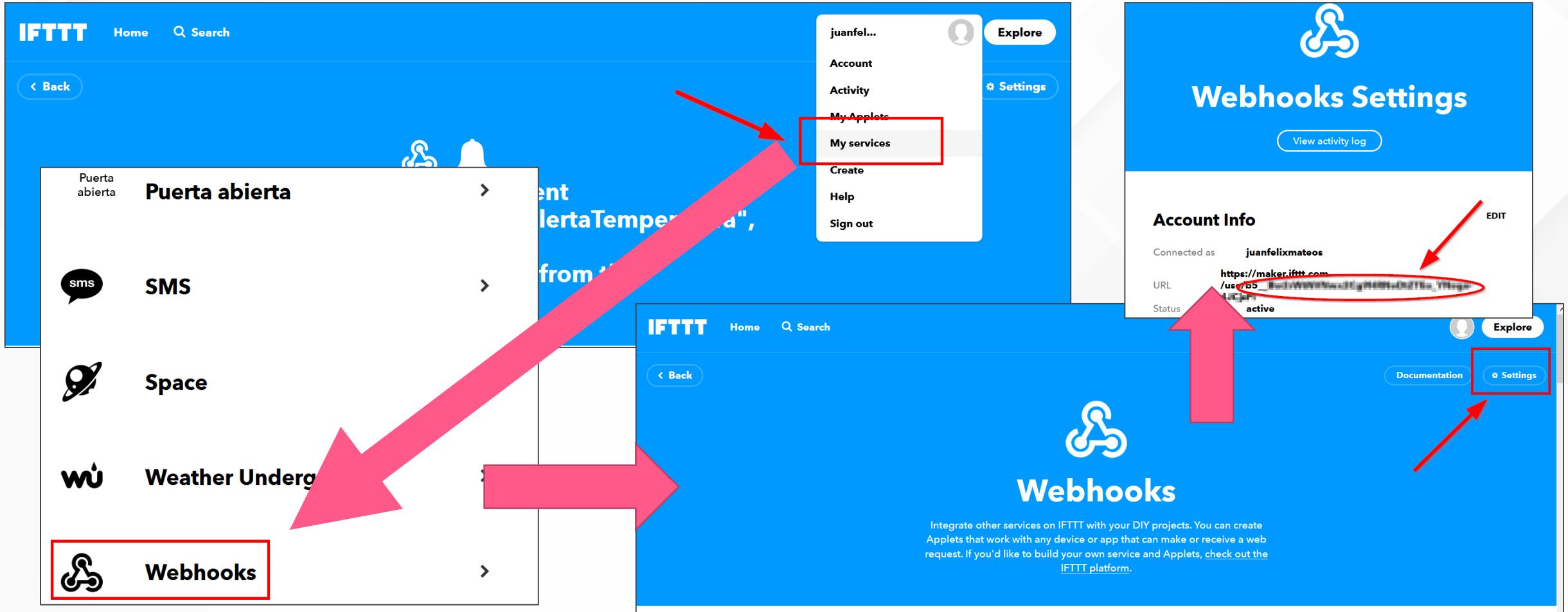
If Maker Event "jfmateos_alertaTemperatura", then Send a notification from the IFTTT app

By juanfelixmateos 88/140

Finish

Integrar ThingsBoard con IFTTT

Obtener el token del servicio Webhook



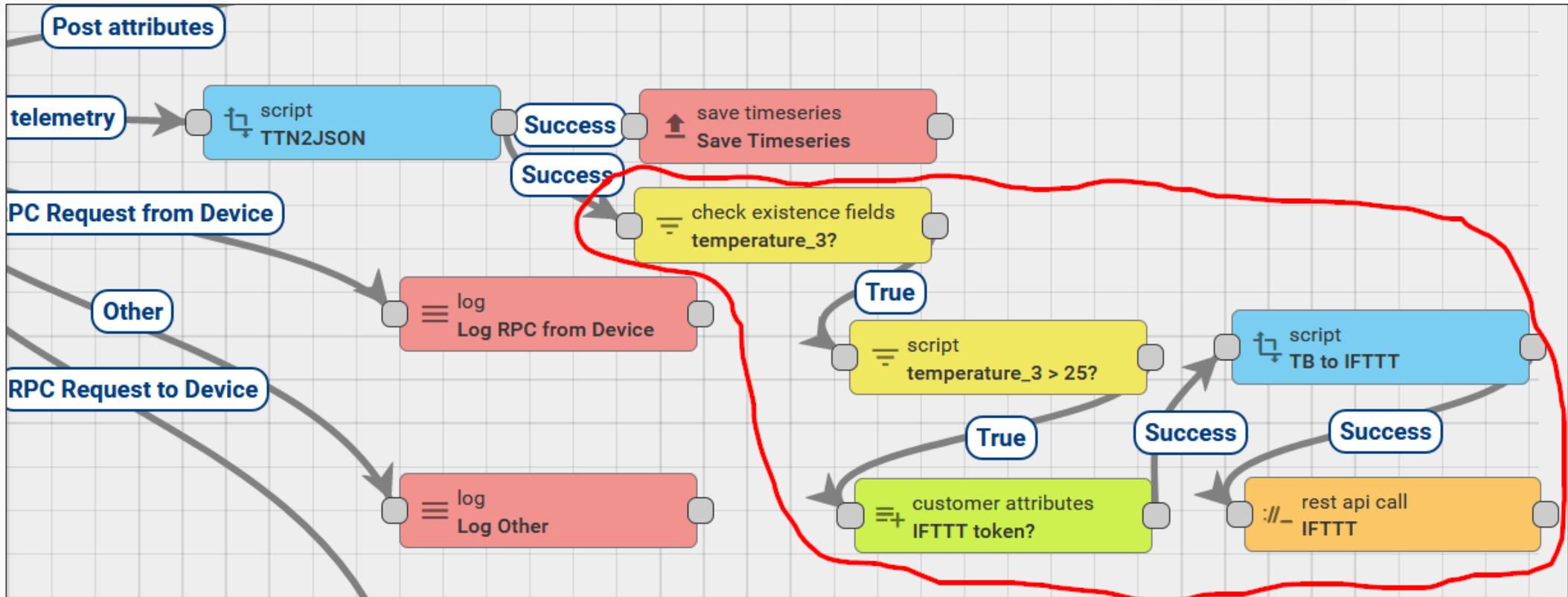
Integrar ThingsBoard con IFTTT

Crear el atributo en el cliente

The screenshot shows the ThingsBoard web application interface. On the left, the navigation bar includes links for IoT open Tech, PÁGINA PRINCIPAL, CADENAS DE REGLAS, CLIENTES (circled in red), ACTIVOS, DISPOSITIVOS, VISTAS DE ENTIDAD, LIBRERÍA DE WIDGETS, and PANELES. The main area displays a list of clients, with one client named "Calle IoT 5 4A" highlighted. This client card shows a red circle around its name and another around the "Atributos" tab. A red arrow points from this tab to a modal window titled "Agregar atributos". The modal contains fields for "Clave*" (with "token_webhook_ifttt" circled in red), "Tipo de valor" (set to "Cadena de c...", circled in red), and "Valor de la cadena de caracteres" (containing the value "GJHMMwQZTbuJWeg-AKqjP", circled in red). At the bottom of the modal are "AGREGAR" and "CANCELAR" buttons.

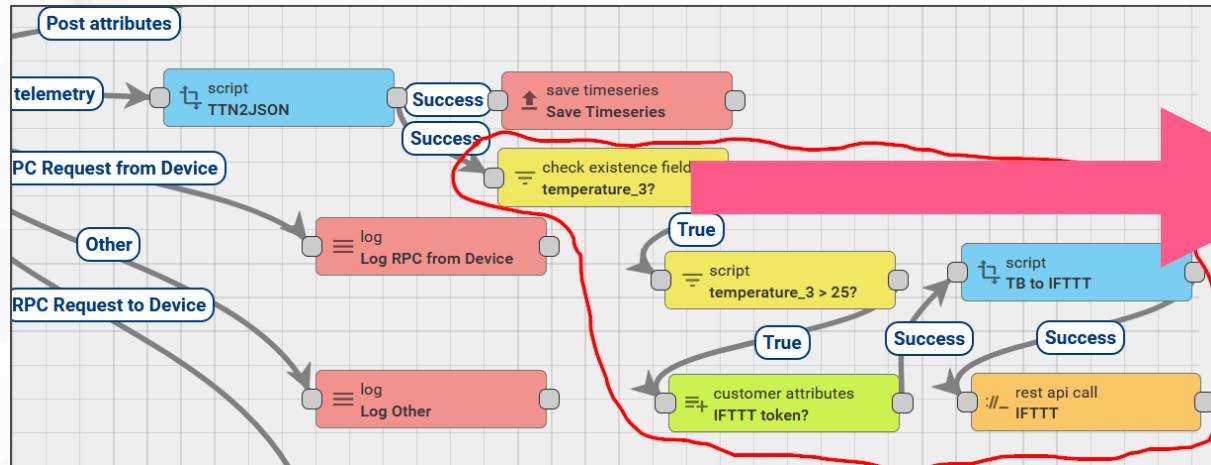
Integrar ThingsBoard con IFTTT

Nodos de la cadena de reglas



Integrar ThingsBoard con IFTTT

Nodos de la cadena de reglas



TEMPERATURE_3?
Filtro - check existence fields

Modo de depuración

DETALLES EVENTOS AYUDA

Nombre *
temperature_3?

Message data *
temperature_3 Message data
You should press "enter" to complete field input.

Message metadata *

Message metadata
You should press "enter" to complete field input.

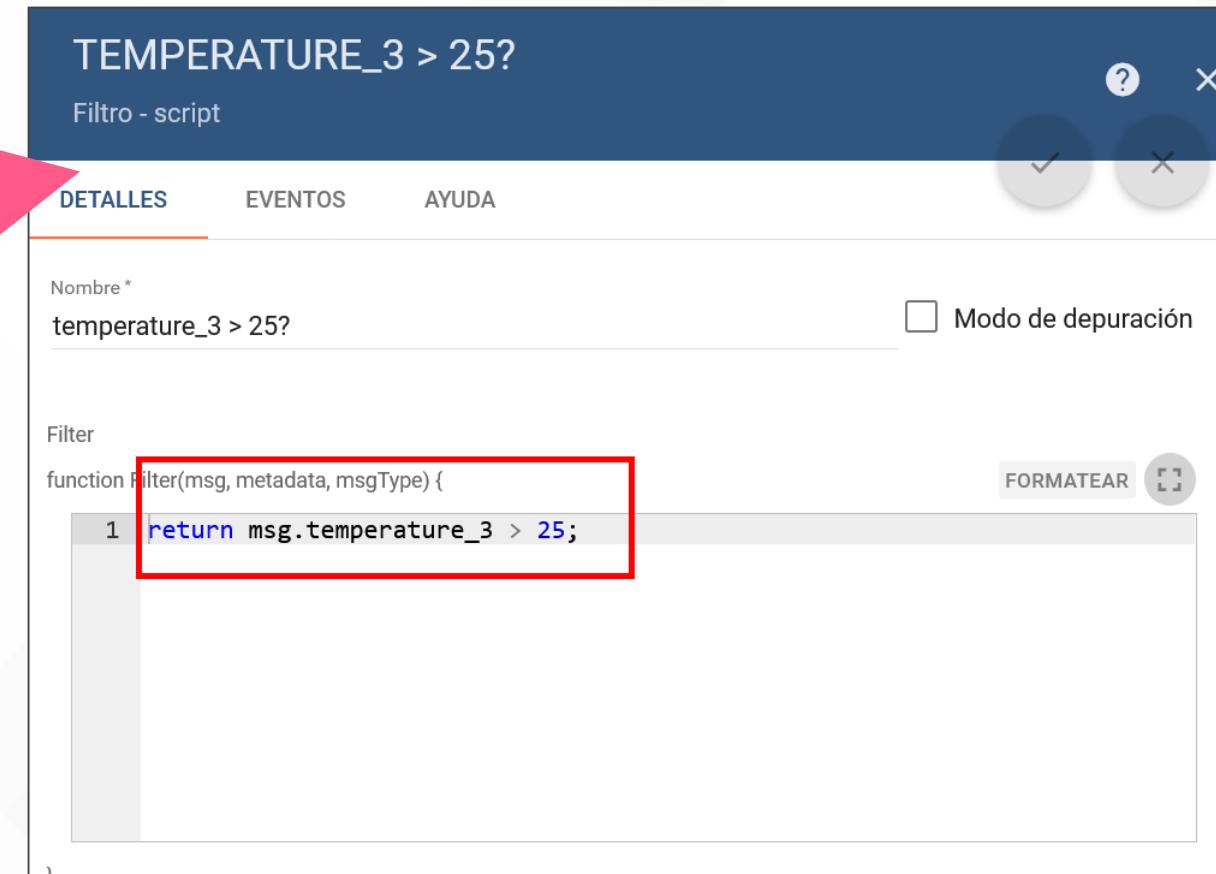
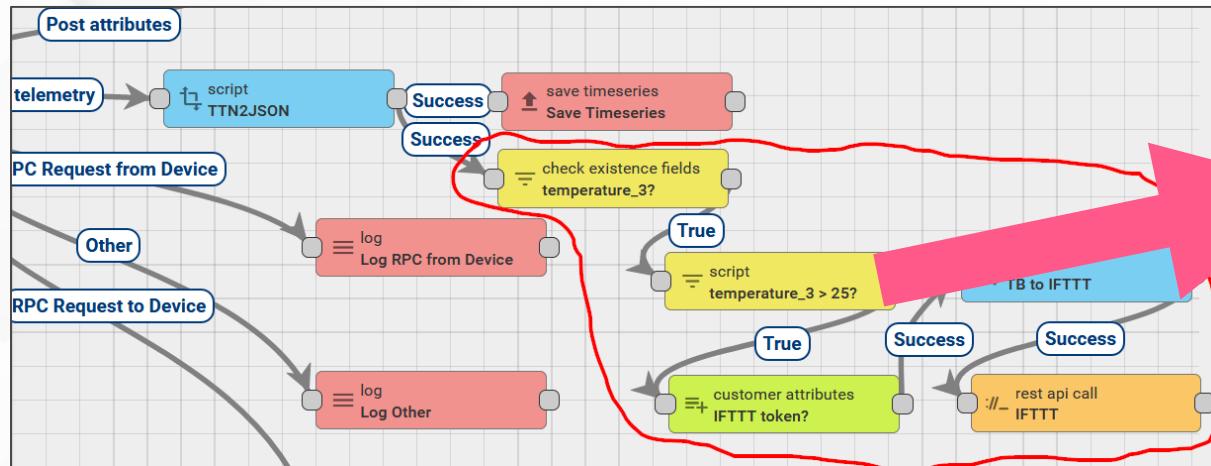
Check that all selected keys are present
If selected, checks that all specified keys are present in the message data and metadata.

Descripción

This screenshot shows the configuration of a "check existence fields" filter named "TEMPERATURE_3?". It includes fields for "Nombre" (set to "temperature_3?") and "Message data" (set to "temperature_3"). A red box highlights the "Message data" input field. Below the configuration, there is explanatory text and a checkbox for "Check that all selected keys are present".

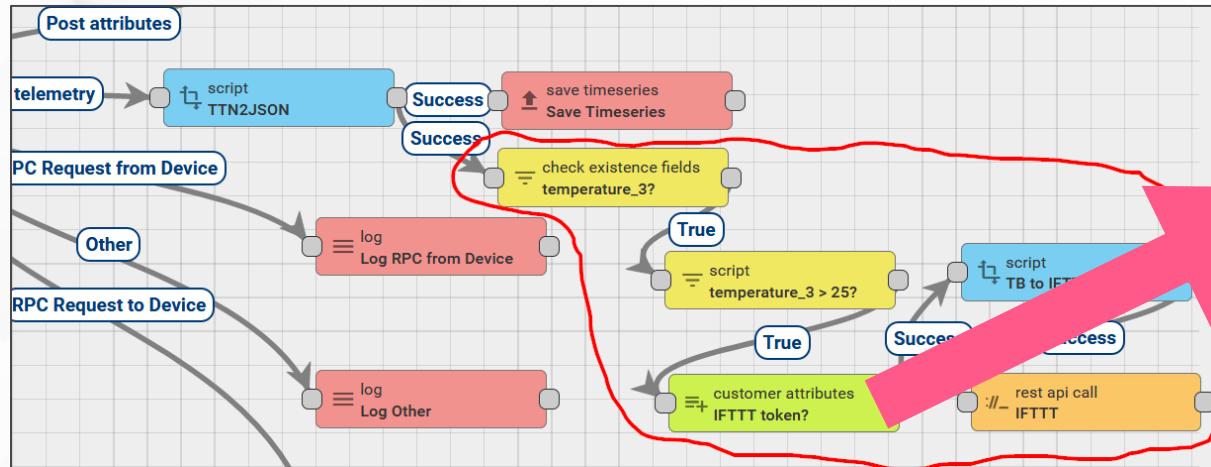
Integrar ThingsBoard con IFTTT

Nodos de la cadena de reglas



Integrar ThingsBoard con IFTTT

Nodos de la cadena de reglas



IFTTT TOKEN?

Enriquecimiento - customer attributes

DETALLES EVENTOS AYUDA

Nombre * IFTTT token?

Modo de depuración

Attributes mapping *

Latest telemetry

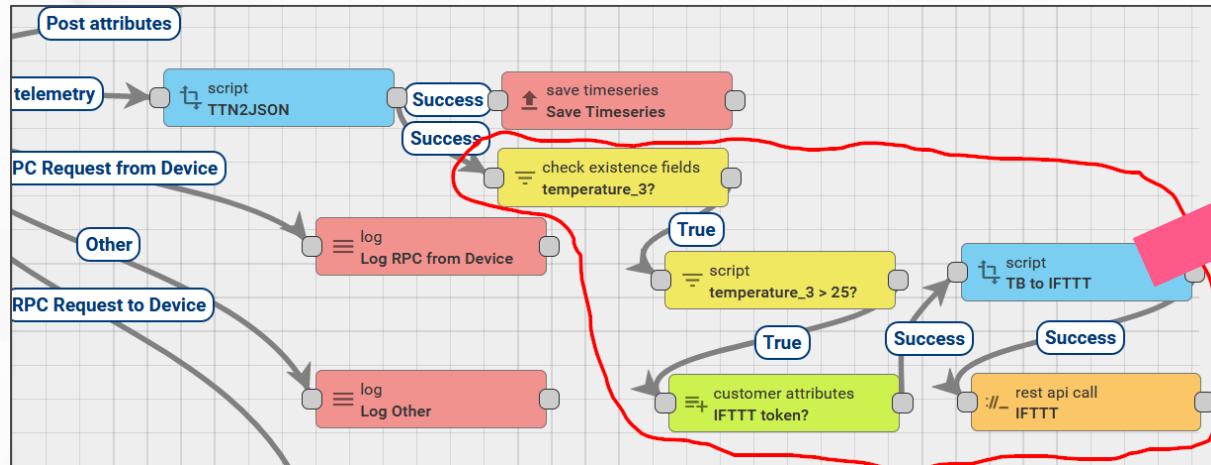
Source attribute	Target attribute
token_webhook_ifttt	token_webhook_ifttt

+ AGREGAR

Descripción

Integrar ThingsBoard con IFTTT

Nodos de la cadena de reglas



TB TO IFTTT
Transformación - script

Nombre *
TB to IFTTT

Modo de depuración

Transform

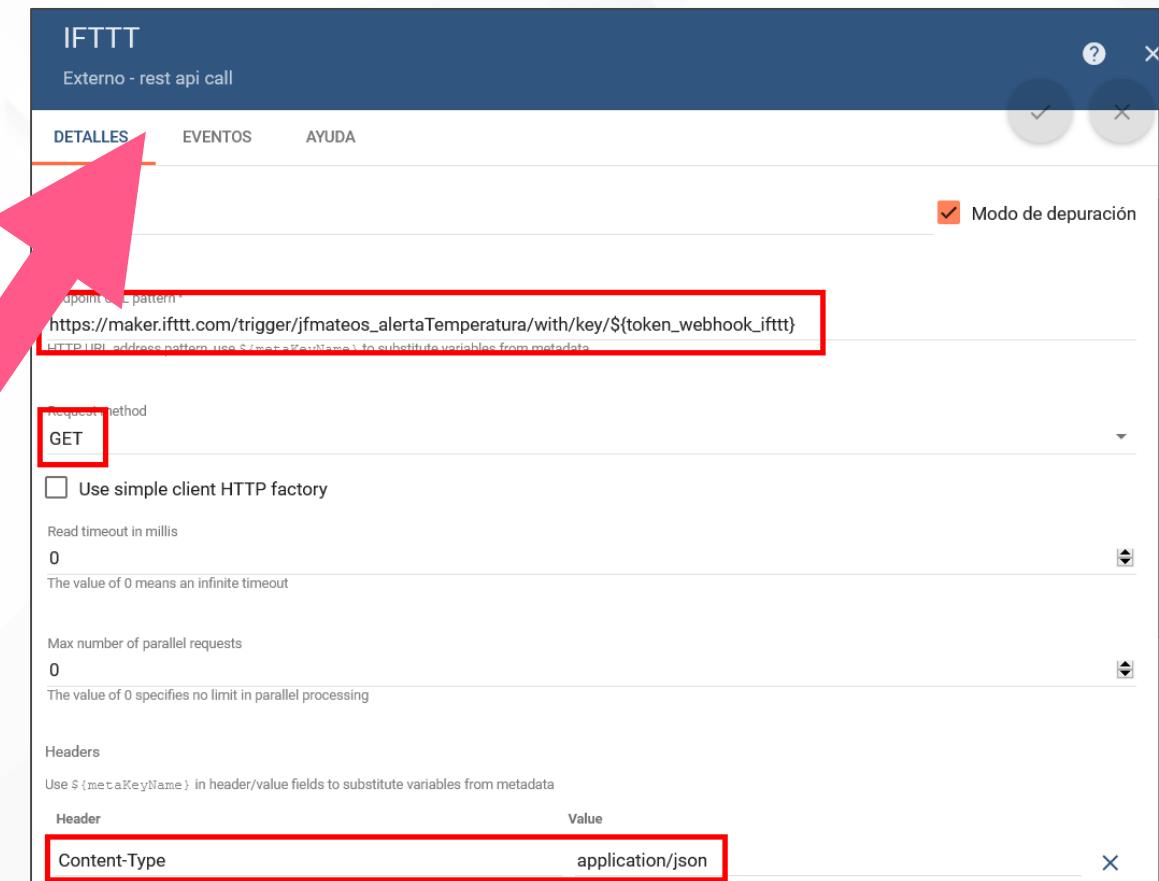
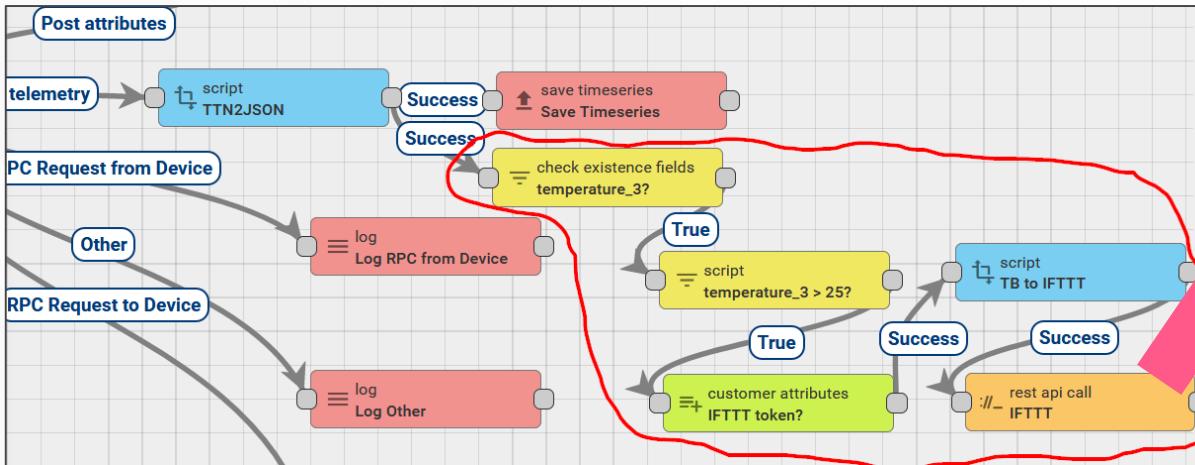
```
function Transform(msg, metadata, msgType) {
    return {
        msg: {
            "value1": msg.temperature_3,
            "value2": metadata.deviceName
        },
        metadata: metadata,
        msgType: msgType
    };
}
```

TEST TRANSFORMER FUNCTION

Integrar ThingsBoard con IFTTT

Nodos de la cadena de reglas

[https://maker.ifttt.com/trigger/jfmateos_alertaTemperatura/with/key/\\${token_webhook_ifttt}](https://maker.ifttt.com/trigger/jfmateos_alertaTemperatura/with/key/${token_webhook_ifttt})

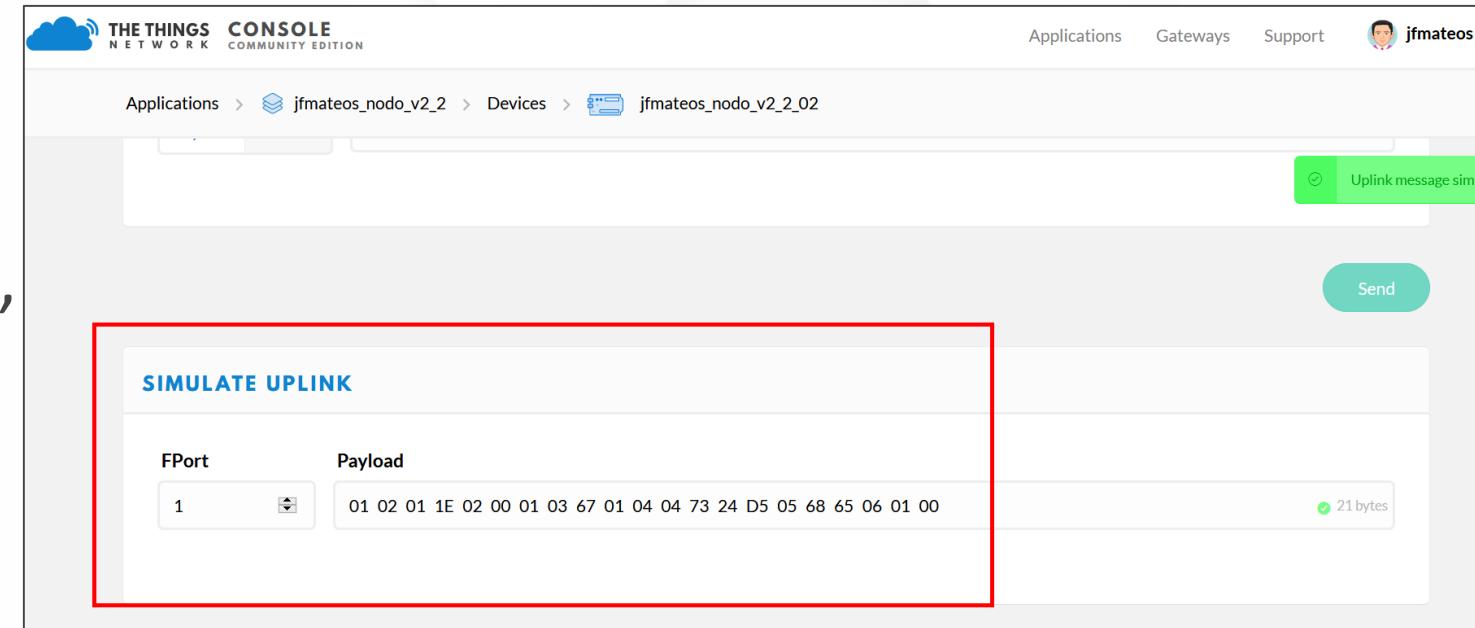


Integrar ThingsBoard con IFTTT

Probar la integración

01 02 01 1E 02 00 01 03 67 01 04 04 73 24 D5 05 68 65 06 01 00

```
{ "analog_in_1": 2.86,  
"barometric_pressure_4":  
942.9,  
"digital_in_2": 1,  
"digital_out_6": 0,  
"relative_humidity_5": 50.5,  
"temperature_3": 26}
```



Integración con Telegram

1. Crear un bot de Telegram y obtener su token
2. Escribir un mensaje al bot y obtener el id del Chat mediante la API de Telegram
3. Utilizar nodos de reglas para componer el mensaje que requiere la API de Telegram y enviarlo mediante un nodo Rest API Call.

Integración con Telegram

Crear un bot de Telegram

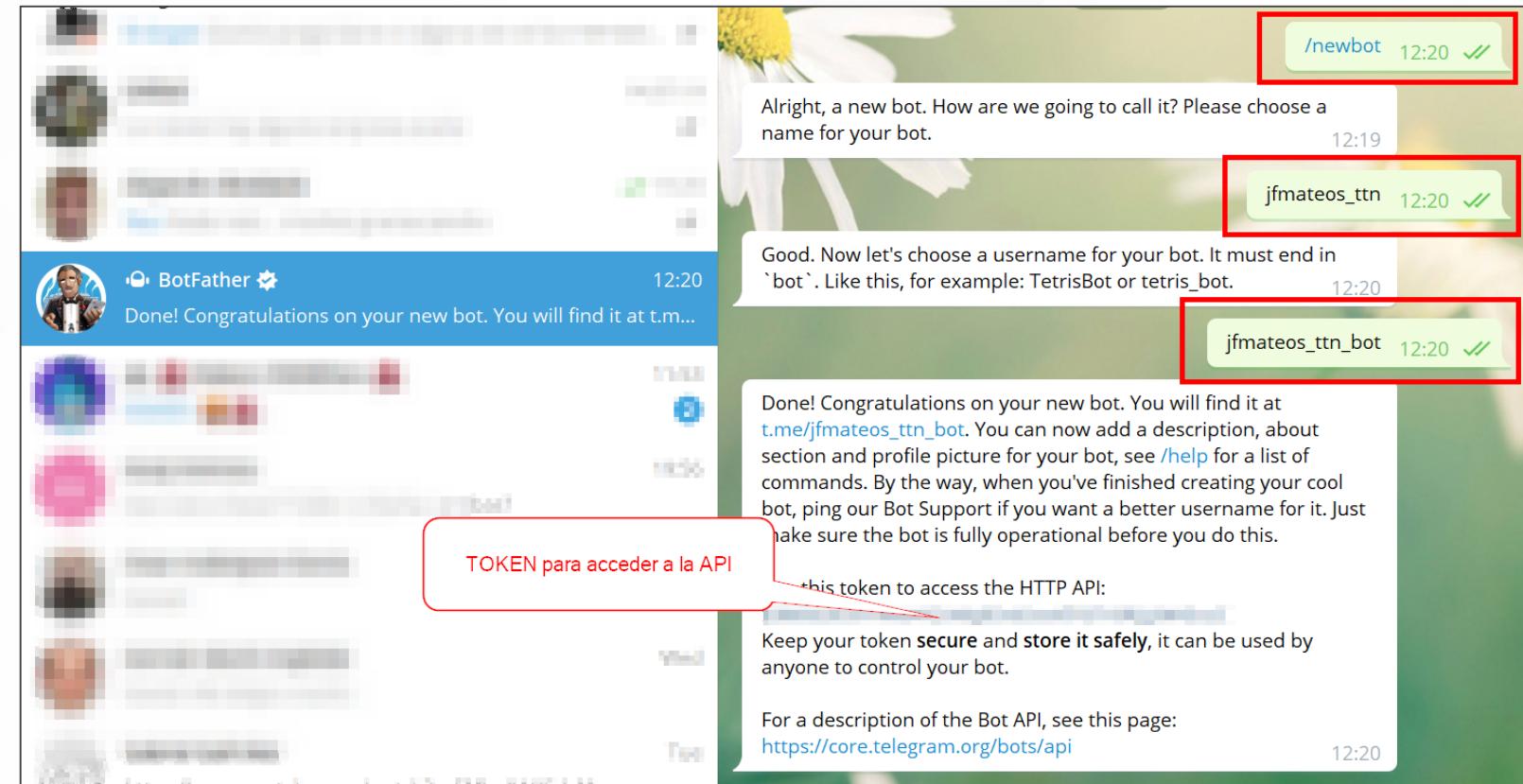
Entablamos la siguiente conversación con BotFather

/newbot

jfmateos_ttn

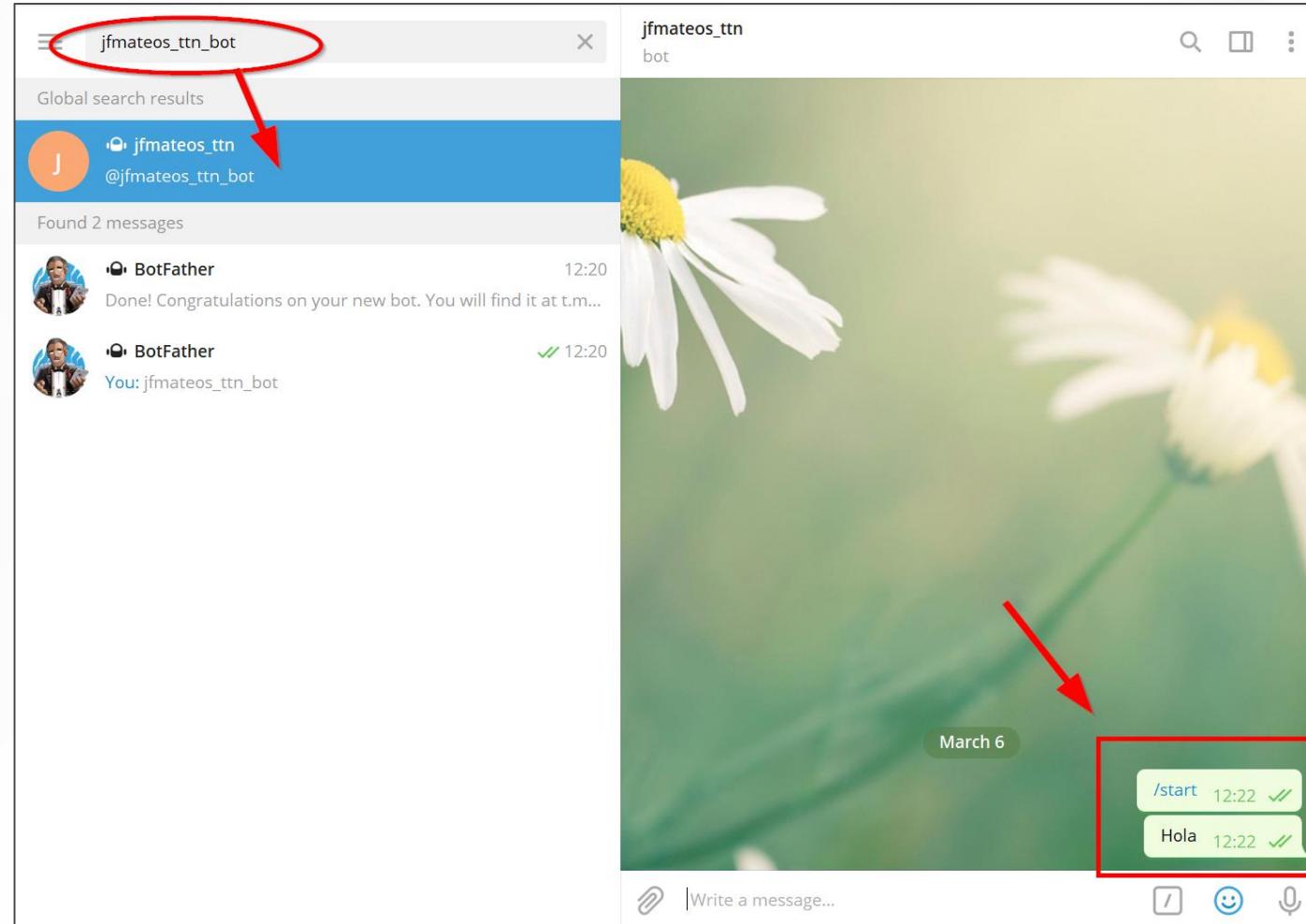
jfmateos_ttn_bot

Copiar el token



Integración con Telegram

Iniciar un chat privado con el bot

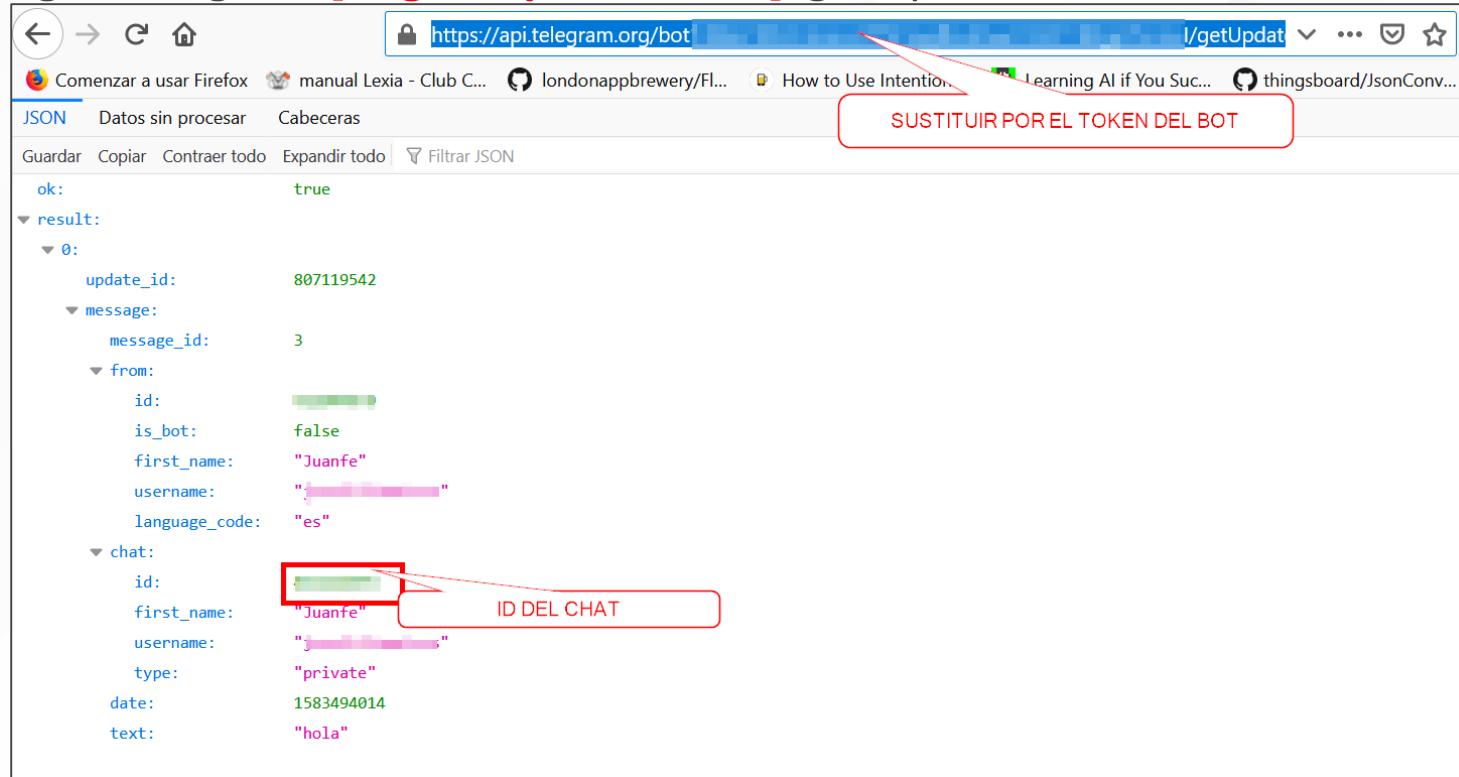


Integración con Telegram

Obtener el ID del chat

Acceder con el navegador web a esta dirección:

- [https://api.telegram.org/bot\[Pegar aquí el token\]/getUpdates](https://api.telegram.org/bot[Pegar aquí el token]/getUpdates)



Integración con Telegram

La API de Telegram

Para realizar peticiones a la API de Telegram se utiliza el siguiente formato:

- <https://api.telegram.org/bot<token>/<método>>

El método para enviar mensajes es **sendMessage**

sendMessage			
Use this method to send text messages. On success, the sent Message is returned.			
Parameter	Type	Required	Description
chat_id	Integer or String	Yes	Unique identifier for the target chat or username of the target channel (in the format <code>@channelusername</code>)
text	String	Yes	Text of the message to be sent, 1–4096 characters after entities parsing
parse_mode	String	Optional	Send Markdown or HTML , if you want Telegram apps to show bold , <i>italic</i> , fixed-width text or inline URLs in your bot's message.
disable_web_page_preview	Boolean	Optional	Disables link previews for links in this message
disable_notification	Boolean	Optional	Sends the message silently . Users will receive a notification with no sound.

Integración con Telegram

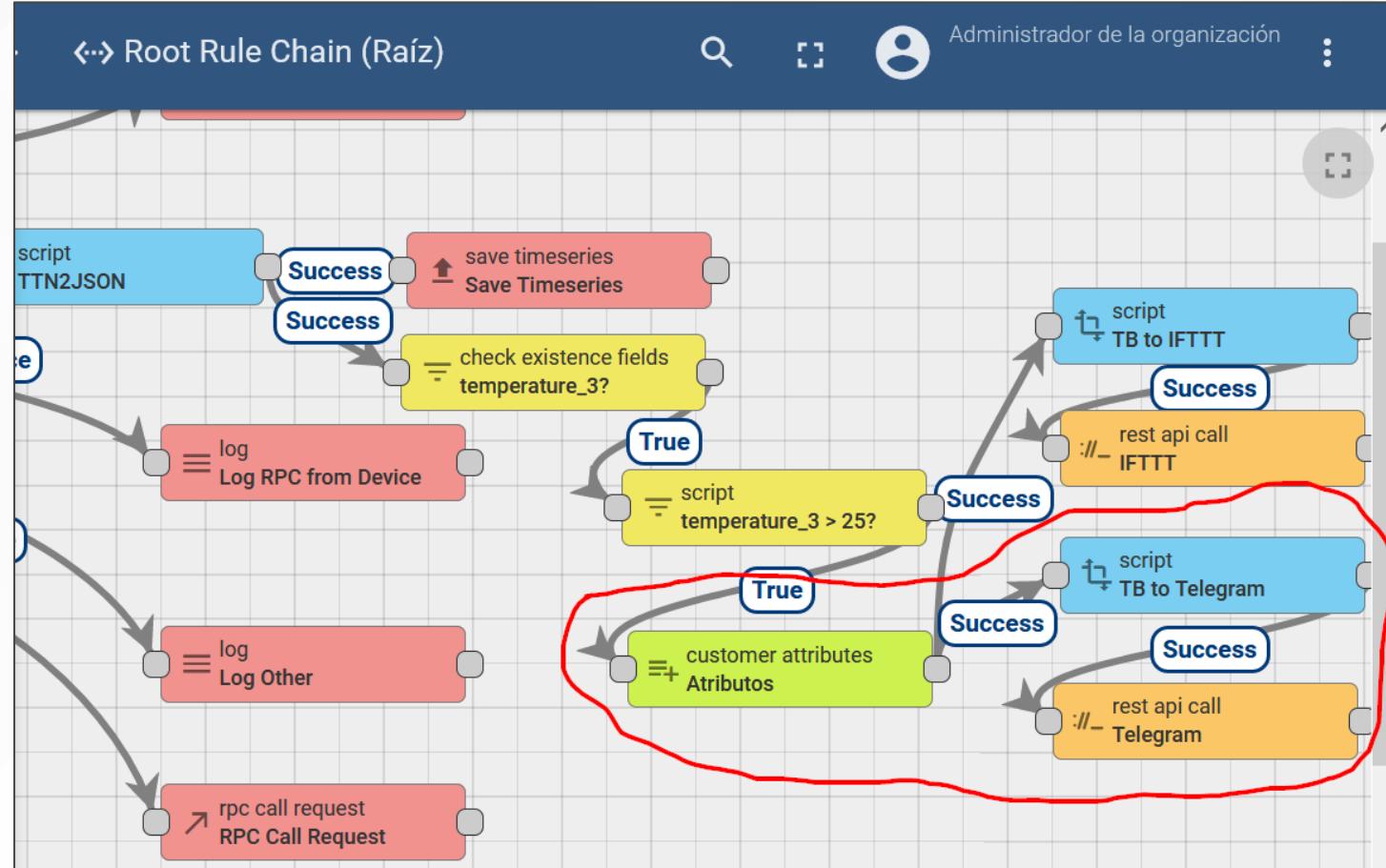
Atributos de cliente para el token y chat id

The screenshot shows a client details interface for "CALLE IOT 5 4A". The "ATRIBUTOS" tab is selected. A red circle highlights the section titled "Atributos del servidor". Below it, a table lists attributes:

Clave ↑	Valor
chatid_telegram	ENTERO
token_telegram	TEXTO
token_webhook_ifttt	
Hora de la última actualización	2020-03-06 12:40:48
	2020-03-06 12:39:56
	2020-03-06 11:23:28

Integración con Telegram

Cadena de reglas



Integración con Telegram

Cadena de reglas: Añadir los atributos

Ampliamos el nodo de enriquecimiento que teníamos anteriormente

The screenshot shows the 'Root Rule Chain (Raiz)' interface on the left and the 'Atributos' (Attributes) configuration dialog on the right.

Root Rule Chain (Raiz) Overview:

- The flowchart starts with a 'script TTN2JSON' node.
- It branches into two parallel paths:
 - The top path leads to 'Success' nodes, then 'save timeseries Save Timeseries' and 'check existence fields temperature_3?' nodes.
 - The bottom path leads to 'log Log RPC from Device' and 'log Log Other' nodes.
- Both paths converge at a 'True' condition node, which then triggers a 'script temperature_3 > 25?' node.
- If the condition is 'True', it leads to another 'True' condition node, which then triggers a 'script TB to IFTTT' node.
- The 'script TB to IFTTT' node leads to a 'rest api call Telegram' node.
- The entire process ends with a 'Success' node.

Atributos Configuration Dialog:

- Nombre ***: customer attributes / Atributos
- Modo de depuración**: checked
- Attributes mapping ***:
 - Source attribute**: token_webhook_ifttt
 - Target attribute**: token_webhook_ifttt
 - Mapping entries** (highlighted with a red border):
 - chatid_telegram
 - token_telegram

Integración con Telegram

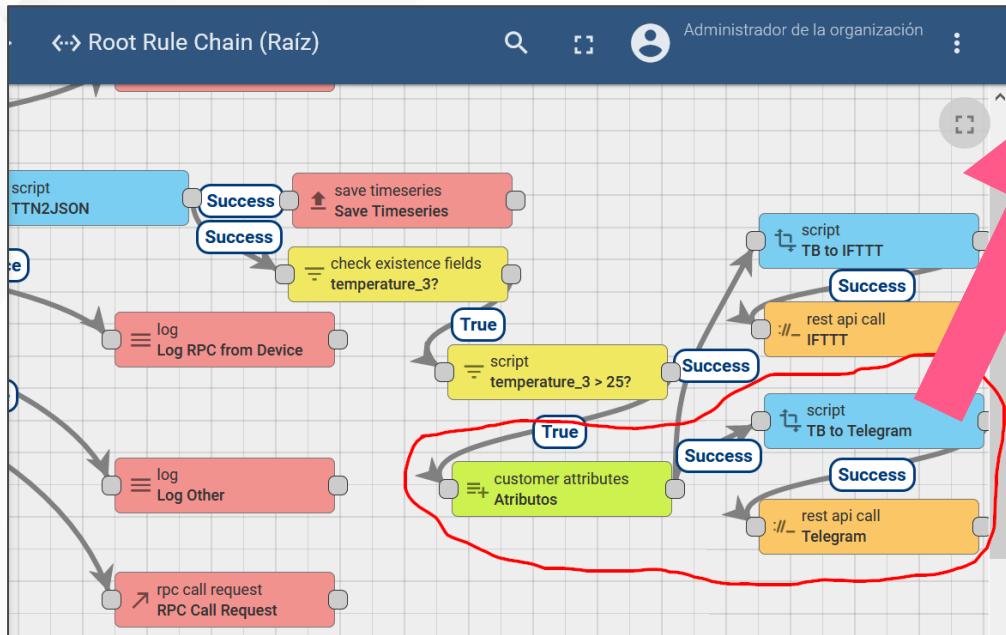
Cadena de reglas: Construir el cuerpo

The image shows a screenshot of a rule chain editor interface. On the left, the 'Root Rule Chain (Raíz)' is displayed with a complex flow of nodes. A red arrow points from the 'TB TO TELEGRAM' transformation dialog on the right to the 'script TB to Telegram' node in the rule chain. The transformation dialog has a title 'TB TO TELEGRAM' and a subtitle 'Transformación - script'. It includes tabs for 'DETALLES', 'EVENTOS', and 'AYUDA'. The 'Nombre *' field is filled with 'TB to Telegram'. A checkbox for 'Modo de depuración' (Debug mode) is checked. The 'Transform' section contains the following JavaScript code:

```
function Transform(msg, metadata, msgType) {  
    1 var mensaje = {};  
    2 mensaje.text = "El dispositivo " + metadata.deviceName;  
    3 mensaje.text += " ha alcanzado " + msg.temperature_3 + " grados";  
    4 mensaje.chat_id = metadata.chatid_telegram;  
    5 return {  
    6     msg: mensaje,  
    7     metadata: metadata,  
    8     msgType: msgType  
    9 }  
}
```

Integración con Telegram

Cadena de reglas: Realizar la petición



The screenshot shows the 'TELEGRAM' configuration dialog. It includes the following fields:

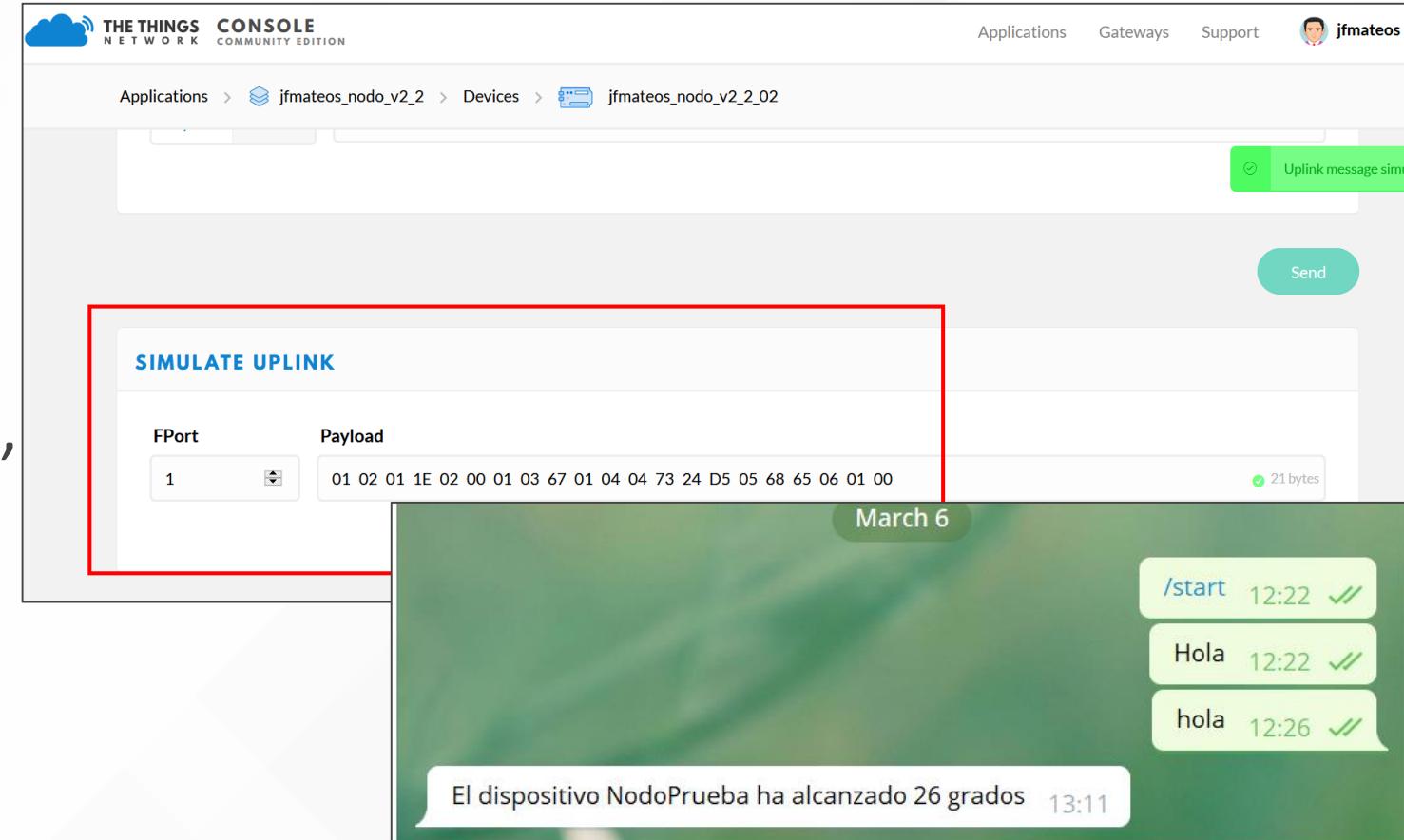
- Nombre ***: Telegram
- Endpoint URL pattern**: `https://api.telegram.org/bot${token_telegram}/sendMessage`
- Método**: POST
- Use simple client HTTP factory**: Unchecked
- Read timeout in millis**: 0
- Max number of parallel requests**: 0
- Headers**:
 - Content-Type: application/json

Integración con Telegram

Probar la integración

01 02 01 1E 02 00 01 03 67 01 04 04 73 24 D5 05 68 65 06 01 00

```
{ "analog_in_1": 2.86,  
"barometric_pressure_4":  
942.9,  
"digital_in_2": 1,  
"digital_out_6": 0,  
"relative_humidity_5": 50.5,  
"temperature_3": 26}
```



jfmateos@educa.madrid.org

juanfelixmateos@gmail.com

GRACIAS