Inbound Webhooks Trigger

Mavilyn Carlos - 2026-09-07 - Premium Workflow Actions (

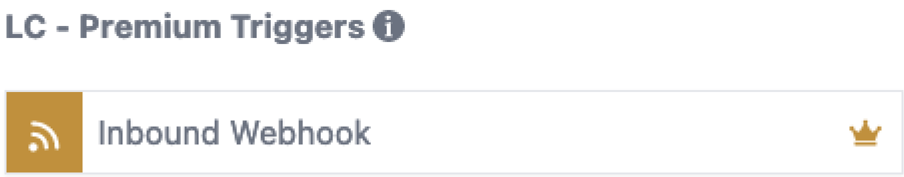
> If you've opted to have the Premium Feature Upgrade, you now have access to additional Premium and AI features within your account. If you do not have the Premium and AI features and would like to consider an upgrade, please email hello@usekilo.com

What is an Inbound Webhook?

An Inbound Webhook is a robust feature that facilitates the automatic data transfer from external systems to your GLM, improving integration and interoperability. When an event occurs in an external system, an HTTP request (POST, GET, or PUT) is sent to a specific URL linked to a trigger within your GLM account, initiating a workflow. This real-time data transfer capability enhances your GLM account's functionality and streamlines workflows across various platforms and tools.

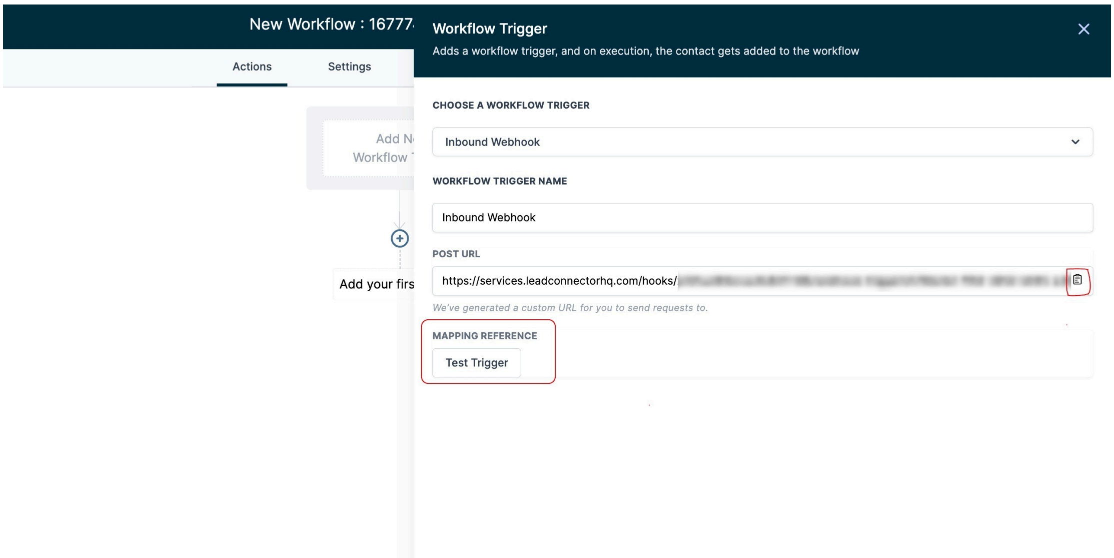
How to use the Inbound Webhook Workflow Trigger?

Briefly, bring your data to your GLM platform in 3 simple steps using the Inbound Webhook Trigger.



Step 1

Retrieve the webhook URL and use it in your application or system. Next, send the data to the webhook URL using the appropriate HTTP request method (POST, GET, or PUT) and click the "Test Trigger" button.



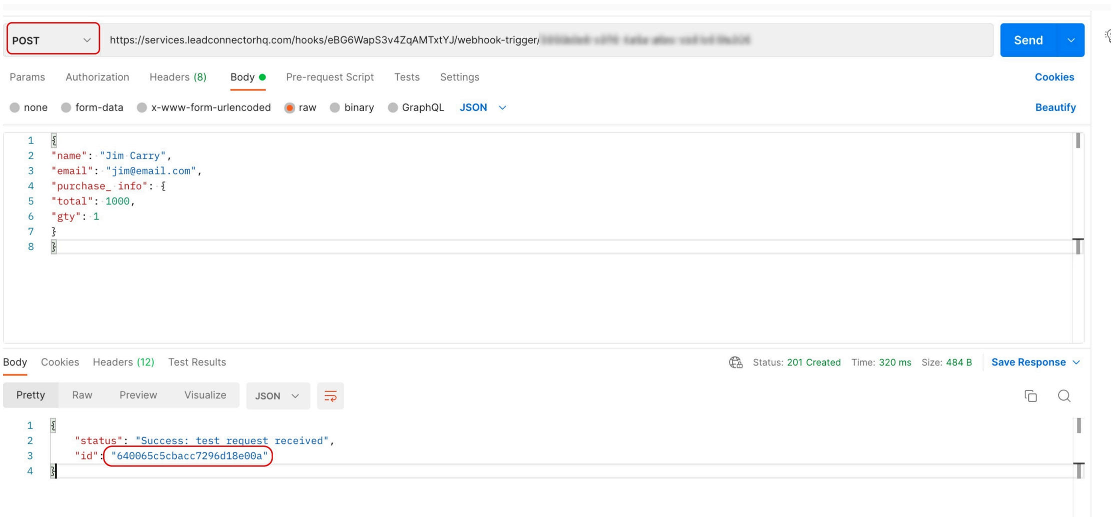
Step 2

Choose the data received from your application or system using the supported HTTP request method (POST, GET, or PUT) and save it as sample data for future reference. Click on "Load More" to view additional recent data. Once you've selected the desired data, save the trigger.

In this step, you can save the data reference for custom variables, which can be treated similarly to other contact custom fields or Location Custom Variables

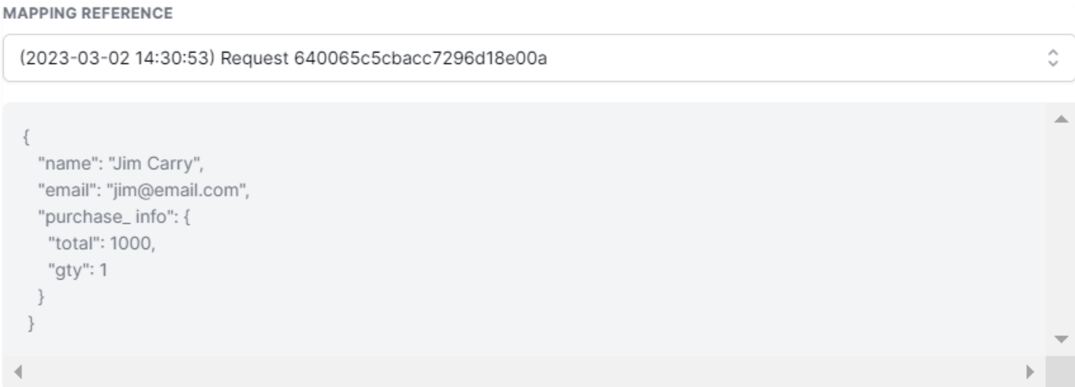
The next step and examples below provide more information on utilizing this reference.

Step 2.1: Send a supported HTTP request (POST, GET, or PUT) to the copied webhook URL and copy the ID in the response.

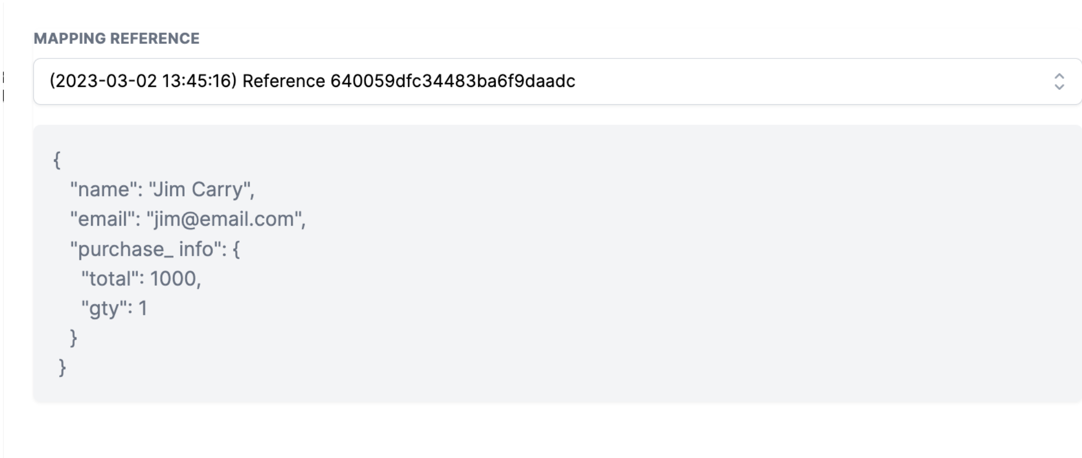


Step 2.2: Copy the unique ID obtained from the supported HTTP request (POST, GET, or PUT) and use it to search for the relevant mapping reference within your webhook trigger.

Please note that the most recent request will always be at the top of the list.

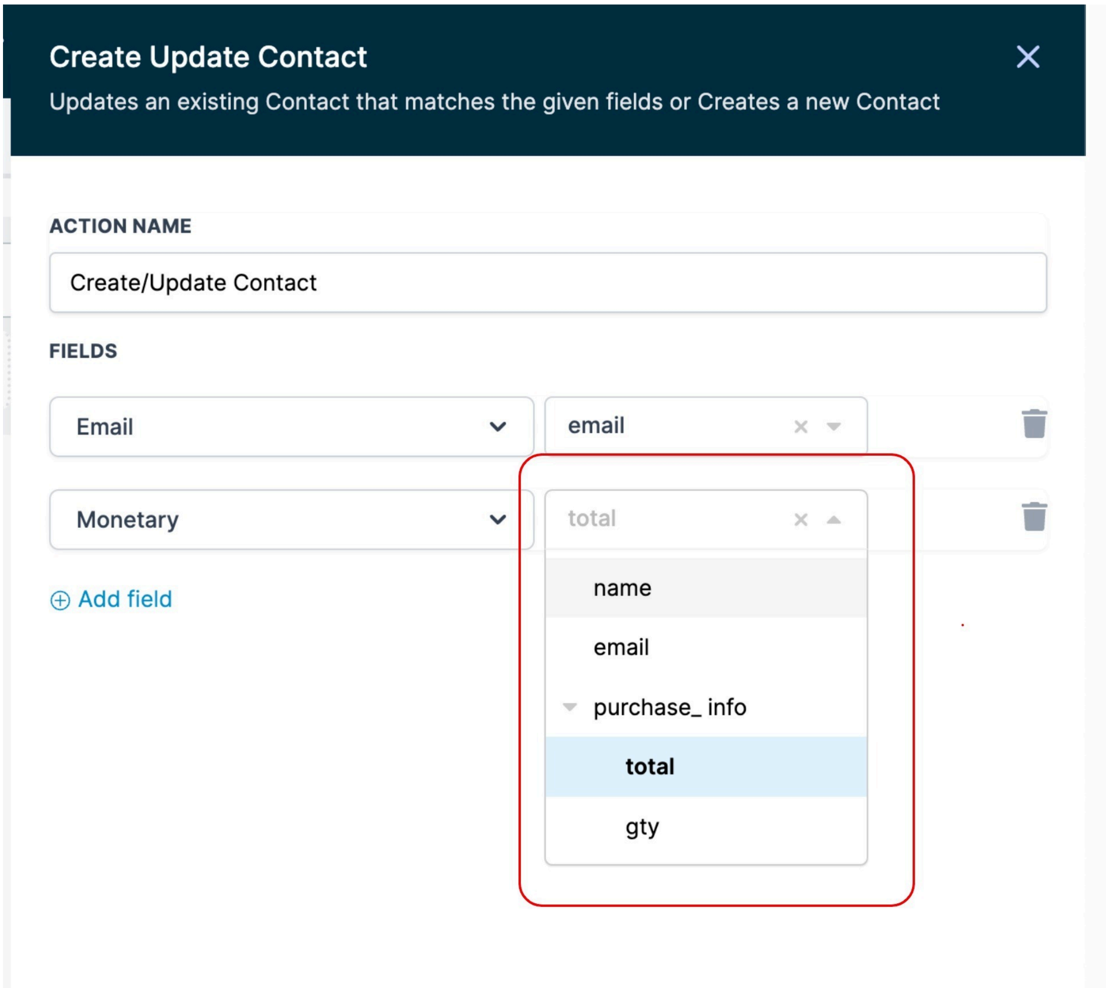


Step 2.3: Select the correct reference and verify the body.

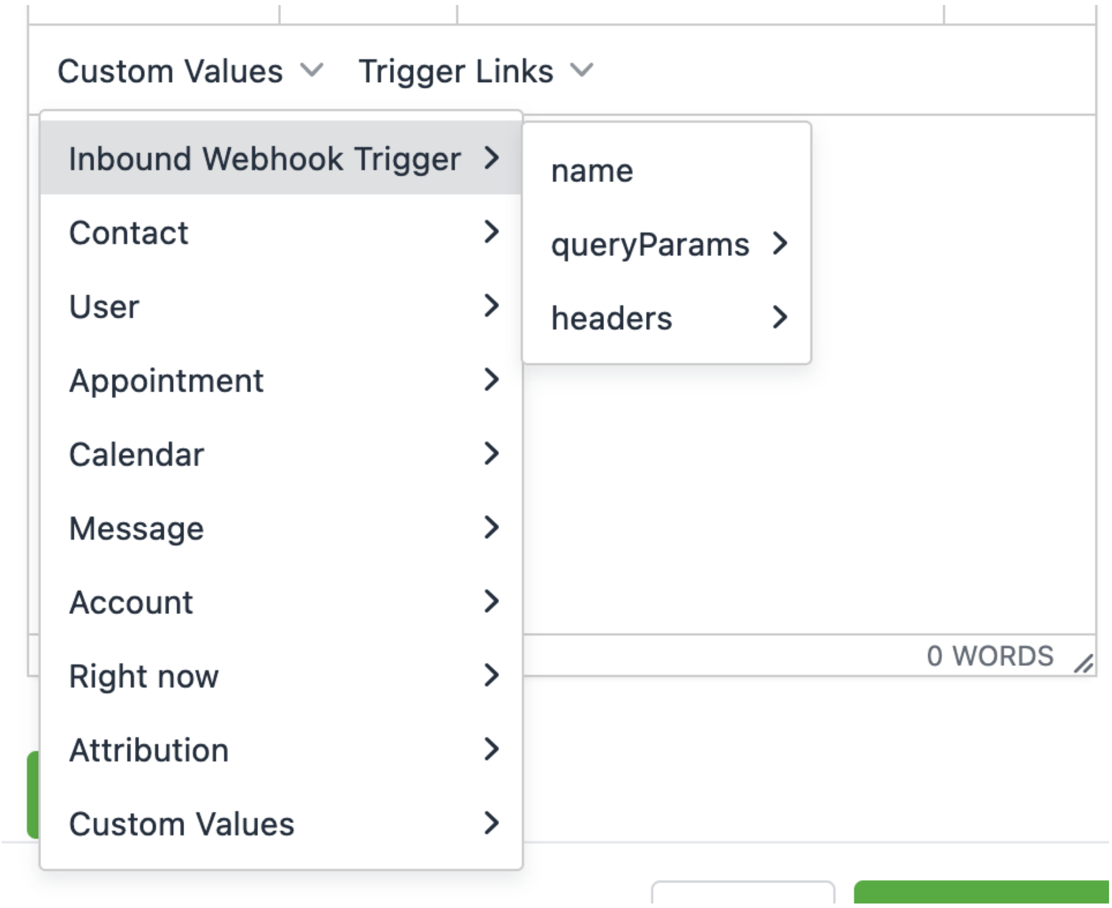


Step 3

You will be directed to the "Create Update Contact" action after saving the trigger. "Create Update contact From there, you can select the necessary fields and map the incoming data you received from the trigger. Please note that to create a new contact or map data to an existing one, you must provide either an email or phone field. If your trigger data is not carrying email or phone you can remove this action from workflow and workflow will continue to execute contactless.



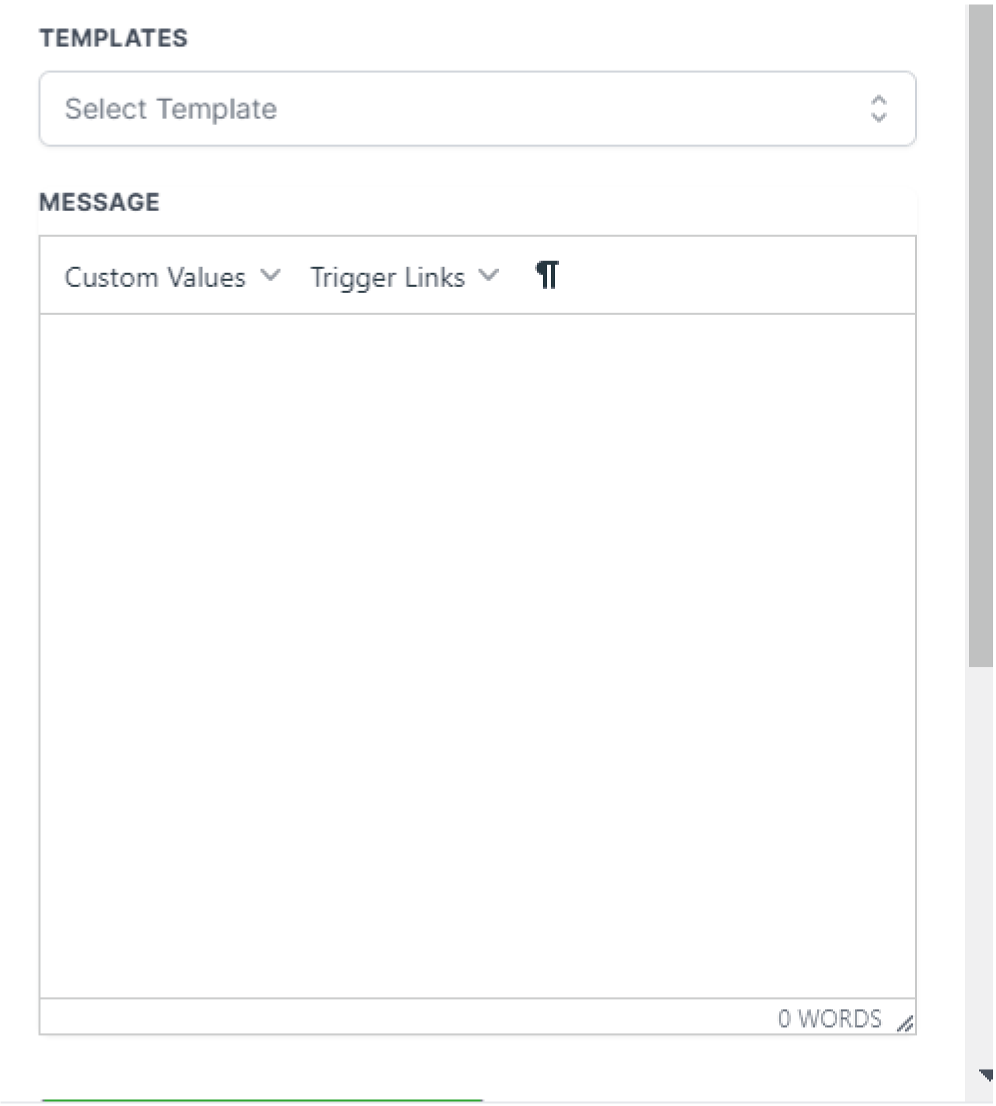
By completing the above steps, you can now use the data you received within the workflow as custom values in email, SMS, if/else, etc.



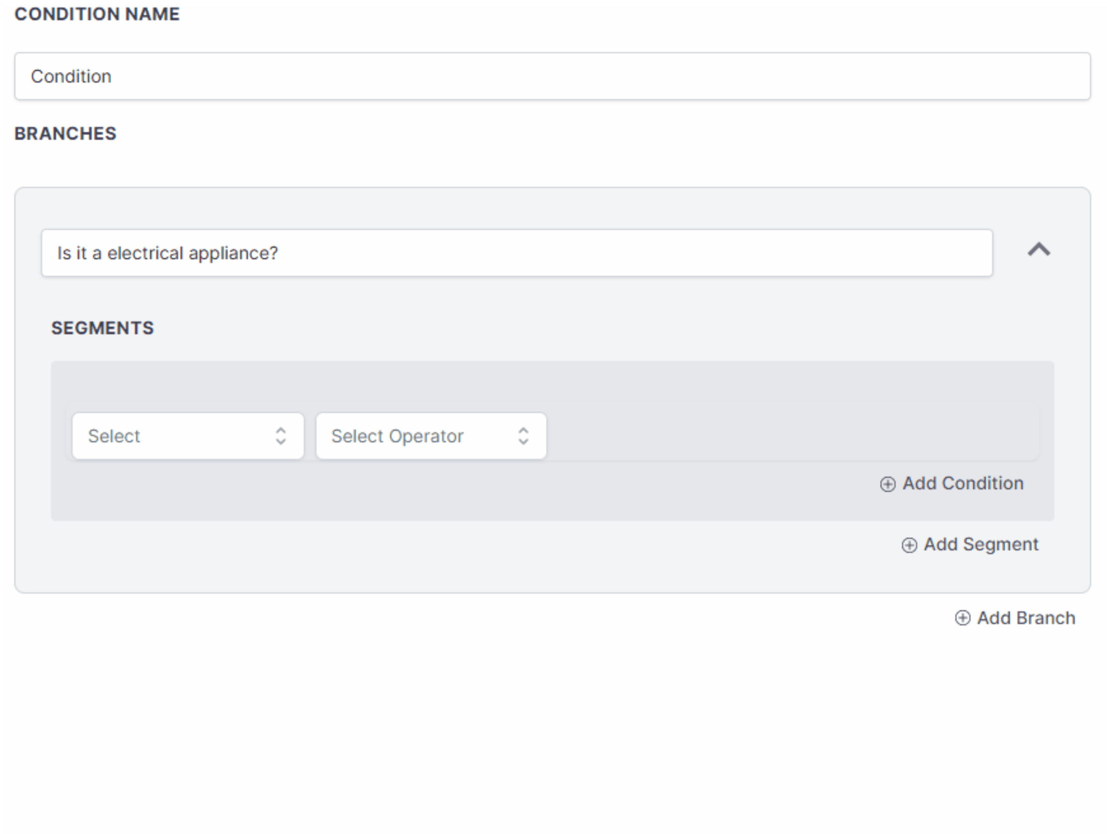
Example:

The custom values of Inbound Webhook can be utilized in all of the actions within your workflow. Some examples of how to do this are provided below.

In your communication Actions:



In If/Else Actions:



As shown above, you can use the values from the Inbound Webhook inside all other actions as you see fit.

Can workflow execute without contact?

- Workflow can run contactless without any Contact data dependency so you can send any payload data via Inbound Webhook Triggers and use it in workflow.
- You can proceed without contact and use actions that are not dependent on contact information. Custom Webhooks, Google Sheets, Slack, ChatGPT, and all Internal Tools can be executed without contact.
- If necessary, you can use the Create/Update or Find Contact actions to retrieve the contact data to the workflow.

  Example:

- Send order data via inbound webhook and add the order information to Google Sheets, use if/else to categorize based on order value, and send a Slack notification.
- Retrieve the contact with Contact ID using the Find contact action

Important points to be considered:

- Always send the request using a supported method (POST, GET, or PUT) when interacting with the webhook.
- Ensure the data is sent as a JSON object, the only supported data format.
- To compile correctly, keys must be a single string without space separations; consider using CamelCase or snake_case instead of separating key names with spaces.
- Providing an Email or Phone number in the payload is mandatory, as the workflow requires contact information. An Email or Phone is required to Find or Create the Contact.
- Arrays are not supported in custom values. You can send them in the request but cannot use them inside actions.
- If your data structure changes, re-select the Mapping Reference inside the Inbound Webhook Trigger setup to address those fields in other actions correctly.
- If your Inbound Webhook Trigger URL gets compromised or leaked and you want to prevent unwanted requests, Delete the existing Inbound Webhook Trigger and Add a New Inbound Webhook Trigger. A new URL with a different ending ID will be generated; update your integrations to match the new URL. Once saved, incoming requests from the old URL won't enter your workflow; only requests from the new URL will.