Telegram Trigger node common issues

# Telegram Trigger node common issues#

Here are some common errors and issues with the Telegram Trigger node and steps to resolve or troubleshoot them.

## Stuck waiting for trigger event#

When testing the Telegram Trigger node with the Test step or or Test workflow buttons, the execution may appear stuck and unable to stop listening for events. If this occurs, you may need to exit the workflow and open it again to reset the canvas.

Stuck listening events often occur due to issues with your network configuration outside of n8n. Specifically, this behavior often occurs when you run n8n behind a reverse proxy without configuring websocket proxying.

To resolve this issue, check your reverse proxy configuration (Nginx, Caddy, Apache HTTP Server, Traefik, etc.) to enable websocket support.

## Bad request: bad webhook: An HTTPS URL must be provided for webhook#

This error occurs when you run n8n behind a reverse proxy and there is a problem with your instance's webhook URL.

When running n8n behind a reverse proxy, you must configure the WEBHOOK\_URL environment variable with the public URL where your n8n instance is running. For Telegram, this URL must use HTTPS.

WEBHOOK\_URL

To fix this issue, configure TLS/SSL termination in your reverse proxy. Afterward, update your WEBHOOK\_URL environment variable to use the HTTPS address.

WEBHOOK\_URL

## Workflow only works in testing or production#

Telegram only allows you to register a single webhook per app. This means that every time you switch from using the testing URL to the production URL (and vice versa), Telegram overwrites the registered webhook URL.

You may have trouble with this if you try to test a workflow that's also active in production. The Telegram bot will only send events to one of the two webhook URLs, so the other will never receive event notifications.

To work around this, you can either disable your workflow when testing or create separate Telegram bots for testing and production.

To create a separate telegram bot for testing, repeat the process you completed to create your first bot. Reference Telegram's bot documentation and the Telegram bot API reference for more information.

To disable your workflow when testing, try the following:

Halts production traffic

This workaround temporarily disables your production workflow for testing. Your workflow will no longer receive production traffic while it's deactivated.

• Go to your workflow page.

• Toggle the Active switch in the top panel to disable the workflow temporarily.

• Test your workflow using the test webhook URL.

• When you finish testing, toggle the Inactive toggle to enable the workflow again. The production webhook URL should resume working.