

# Using the GitHub CLI to forward webhooks for testing

## In this article

About receiving webhooks with GitHub CLI

Receiving webhooks with GitHub CLI

You can use the GitHub CLI to test webhooks in your development environment without the complexity of port forwarding or third-party tools.

## About receiving webhooks with GitHub CLI

When you make changes to your integration code, running the code in a local environment lets you test and iterate quickly without deploying the code. You can use GitHub CLI to forward webhooks to your local environment.

Webhook forwarding in the GitHub CLI only works with repository and organization webhooks. If you want to test other types of webhooks locally, you'll need to do this manually. For more information, see "[Testing webhooks](#)."

**Warning:** Webhook forwarding is only designed for use during testing and development. It is not supported for use in production environments for handling live webhooks.

## Receiving webhooks with GitHub CLI

To learn more about GitHub CLI, see "[About GitHub CLI](#)."

- 1 To install the GitHub CLI extension to enable webhook forwarding, use the `extension install` subcommand.

```
gh extension install cli/gh-webhook
```

- 2 If you want to forward organization webhooks, add the `admin:org_hook` scope to your GitHub CLI login so you have the required permissions.

```
gh auth refresh --scopes admin:org_hook
```

- 3 Start your application locally, and take a note of the URL where it's expecting to receive webhooks. This guide assumes that your application is listening for webhook events at `http://localhost:3000/webhook`.

- 4 To set up webhooks to be delivered to your application, run the `webhook forward`

subcommand. Replace `REPOSITORY` with the name of your repository. For example, `monalisa/octocat`. Replace `EVENTS` with a comma-separated list of the events that you want to receive. For example, `issues,pull_request`. Replace `URL` with the local URL where your application expects to receive webhooks. For example, `"http://localhost:3000/webhook"`. To listen for organization webhooks instead of repository webhooks, replace the `--repo` flag with the `--org` flag. For example `--org="octo-org"`.

```
gh webhook forward --repo=REPOSITORY --events=EVENTS --url=URL
```

Leave the command running in the background. It will receive all of the specified events for the specified repository and forward them to your webhook handler running at the specified URL.

**Note:** Only one person can use webhook forwarding at a time for each repository and organization. If you try to set up webhook forwarding and someone else is already working with that organization or repository, you'll receive a `Hook already exists` error.

## Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)