

title: GitHub Webhook Events Integration description: GitHub Webhooks allow you to build or set up integrations, such as GitHub Apps or OAuth Apps, which subscribe to certain events on GitHub.com.

By configuring the Webhook to point to Sematext, you can store all data about the triggered events in one place. Gain insight into the entire GitHub workflow in your Organization.

GitHub Webhook Events Quick Start

When an event is triggered, GitHub sends an HTTP POST payload to the Webhook's configured URL. Webhooks can be used to update an external issue tracker, trigger CI builds, update a backup mirror, or even deploy to your production server. You're only limited by your imagination.

Gain insights into your Organization's team pulse, patterns, metrics, development velocity, progress, trends, to spot opportunities for improvement in the daily workflow. Keep your team happy and enable them to be more productive and fulfilled.

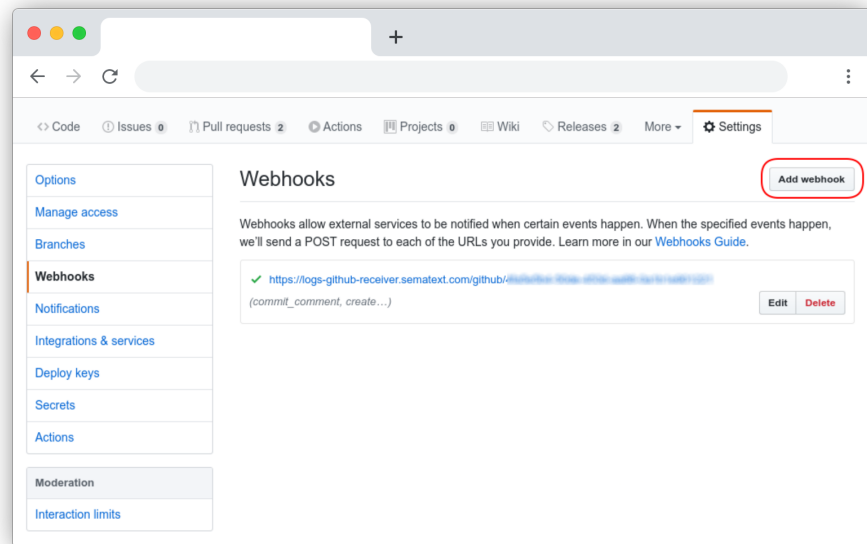
The Sematext GitHub Webhook Events Integration supports these events:

- Commit comments
- Branch or tag creation
- Branch or tag deletion
- Issues
- Issue comments
- Pull requests
- Pull request reviews
- Pull request review comments
- Pushes
- Releases

You can configure a Webhook to send events to Sematext either through the Github UI or with their API.

Set Up A GitHub Webhook In The GitHub UI

Open the Organization or Repository settings, navigate to Webhooks and click the Add webhook button.



Add the Payload URL.

`https://logs-github-receiver.semamtext.com/github/<LOGS_TOKEN>`

Or for EU

`https://logs-github-receiver.eu.semamtext.com/github/<LOGS_TOKEN>`

Webhooks / **Manage webhook**

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, `x-www-form-urlencoded`, etc). More information can be found in [our developer documentation](#).

Payload URL *

`https://logs-github-receiver.semamtext.com/github/<LOGS_TOKEN>`

Content type

`application/x-www-form-urlencoded`

Secret

SSL verification

By default, we verify SSL certificates when delivering payloads.

☒ **Enable SSL verification** ☐ **Disable** (not recommended)

Check the supported Events and Save the Webhook.

+

←

→

↺

Which events would you like to trigger this webhook?

☐ Just the push event.

☐ Send me everything.

☒ Let me select individual events.

☐ **Check runs**
 Check run is created, requested, rerequested, or completed.

☐ **Check suites**
 Check suite is requested, rerequested, or completed.

☒ **Commit comments**
 Commit or diff commented on.

☒ **Branch or tag creation**
 Branch or tag created.

☒ **Branch or tag deletion**
 Branch or tag deleted.

☐ **Deploy keys**
 A deploy key is created or deleted from a repository.

☐ **Deployments**
 Repository was deployed or a deployment was deleted.

☐ **Deployment statuses**
 Deployment status updated from the API.

☐ **Forks**
 Repository forked.

☐ **Wiki**
 Wiki page updated.

☒ **Issue comments**
 Issue comment created, edited, or deleted.

☒ **Issues**
 Issue opened, edited, deleted, transferred, pinned, unpinned, closed, reopened, assigned, unassigned, labeled, unlabeled, milestone, demilestoned, locked, or unlocked.

☐ **Labels**
 Label created, edited or deleted.

☐ **Collaborator add, remove, or changed**
 Collaborator added to, removed from, or has changed permissions for a repository.

☐ **Meta**
 This particular hook is deleted.

☐ **Milestones**
 Milestone created, closed, opened, edited, or deleted.

☐ **Packages**
 GitHub Packages published or updated in a repository.

☐ **Page builds**
 Pages site built.

☐ **Projects**
 Project created, updated, or deleted.

☐ **Project cards**
 Project card created, updated, or deleted.

☐ **Project columns**
 Project column created, updated, moved or deleted.

☐ **Visibility changes**
 Repository changes from private to public.

☒ **Pull requests**
 Pull request opened, closed, reopened, edited, assigned, unassigned, review requested, review request removed, labelled, unlabelled, synchronized, ready for review, locked, or unlocked.

☒ **Pull request reviews**
 Pull request review submitted, edited, or dismissed.

☒ **Pull request review comments**
 Pull request diff comment created, edited, or deleted.

☒ **Pushes**
 Git push to a repository.

☐ **Registry packages**
 Registry package published or updated in a repository.

☒ **Releases**
 Release created, edited, published, unpublished, or deleted.

☐ **Repositories**
 Repository created, deleted, archived, unarchived, publicized, privatized, edited, renamed, or transferred.

☐ **Repository imports**
 Repository import succeeded, failed, or cancelled.

☐ **Repository vulnerability alerts**
 Security alert created, resolved, or dismissed on a repository.

☐ **Stars**
 A star is created or deleted from a repository.

☐ **Statuses**
 Commit status updated from the API.

☐ **Team adds**
 Team added or modified on a repository.

☐ **Watches**
 User stars a repository.

☒ **Active**
 We will deliver event details when this hook is triggered.

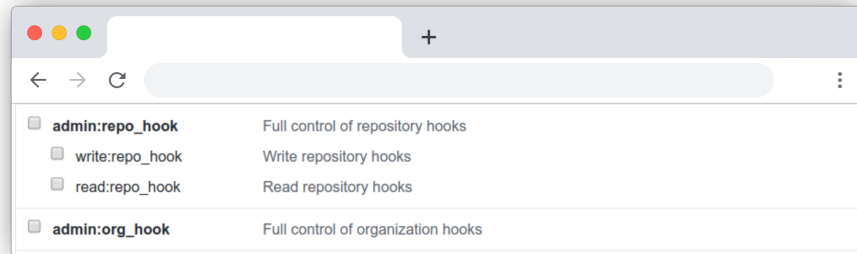
Update webhook

Delete webhook

Save the Webhook.

Set Up A GitHub Webhook Using The GitHub API

First create a new API token. Open the Personal Access tokens settings in your GitHub account. It should be here. Click **Generate new token**. The only permissions you need are related to hooks.



Finish creating the token and save it somewhere safe. You will need it in the next step.

Now you need to create a webhook. Using cURL you can now create a new Webhook in any Organization or Repository you have access to.

Organization

```
curl -X POST \  
  https://api.github.com/orgs/<ORGANIZATION_NAME>/hooks \  
  -H 'authorization: token <YOUR_API_TOKEN>' \  
  -H 'cache-control: no-cache' \  
  -d '{  
    "config": {  
      "url": "https://logs-github-receiver.sematest.com/github/<LOGS_TOKEN>"  
      # for EU "url": "https://logs-github-receiver.eu.sematest.com/github/<LOGS_TOKEN>"  
    },  
    "events": [  
      "issues",  
      "issue_comment",  
      "pull_request",  
      "pull_request_review",  
      "pull_request_review_comment",  
      "commit_comment",  
      "push",  
      "release",  
      "create",  
      "delete"  
    ]  
  }'
```

```
]
}'
```

Repository

```
curl -X POST \
  https://api.github.com/repos/<GITHUB_USERNAME>/<REPOSITORY_NAME>/hooks \
  -H 'authorization: token <YOUR_API_TOKEN>' \
  -H 'cache-control: no-cache' \
  -d '{
    "config": {
      "url": "https://logs-github-receiver.sematext.com/github/<LOGS_TOKEN>"
      # for EU "url": "https://logs-github-receiver.eu.sematext.com/github/<LOGS_TOKEN>"
    },
    "events": [
      "issues",
      "issue_comment",
      "pull_request",
      "pull_request_review",
      "pull_request_review_comment",
      "commit_comment",
      "push",
      "release",
      "create",
      "delete"
    ]
  }'
```