

# GitHub

## Introduction

The GitHub integration collects events from the [GitHub API](#).

- <https://docs.github.com/en/rest?apiVersion=2022-11-28>

Logs

## Audit

The GitHub audit log records all events related to the GitHub organization.

To use this integration, you must be an organization owner, and you must use an Personal Access Token with the admin:org scope.

This integration is not compatible with GitHub Enterprise server.

## Code Scanning

The Code Scanning lets you retrieve all security vulnerabilities and coding errors from a repository setup using Github Advanced Security Code Scanning feature.

To use this integration, GitHub Apps must have the security\_events read permission. Or use a personal access token with the security\_events scope for private repos or public\_repo scope for public repos.

## Secret Scanning

The Github Secret Scanning lets you retrieve secret scanning for advanced security alerts from a repository setup using Github Advanced Security Secret Scanning feature.

To use this integration, GitHub Apps must have the secret\_scanning\_alerts read permission. Or you must be an administrator for the repository or for the organization that owns the repository, and you must use a personal access token with the repo scope or security\_events scope. For public repositories, you may instead use the public\_repo scope.

## Dependabot

The Github Dependabot lets you retrieve known vulnerabilities in dependencies from a repository setup using Github Advanced Security Dependabot feature.

To use this integration, you must be an administrator for the repository or for the organization that owns the repository, and you must use a personal access token with the repo scope or security\_events scope. For public repositories, you may instead use the public\_repo scope.

## Issues

The Github Issues datastream lets you retrieve github issues, including pull requests, issue assignees, comments, labels, and milestones. See About Issues for more details. You can retrieve issues for specific repository or for entire organization. Since Github API considers pull requests as issues, users can use github.issues.is\_pr field to filter for only pull requests.

All issues including closed are retrieved by default. If users want to retrieve only open requests, you need to change State parameter to open.

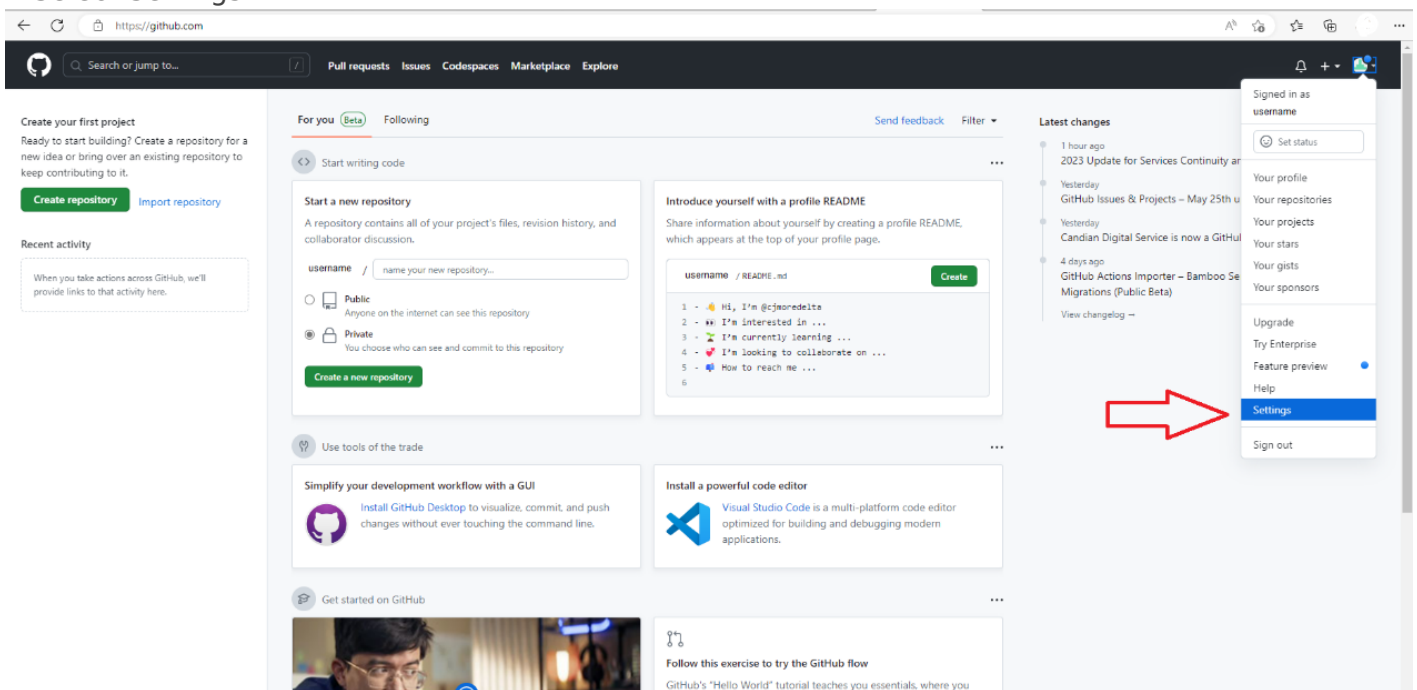
To use this integration, users must use Github Apps or Personal Access Token with read permission to repositories or organization. Please refer to Github Apps Permissions Required and Personal Access Token Permissions Required for more details.

## GitHub Integration Procedures

This integration is not compatible with GitHub Enterprise server.

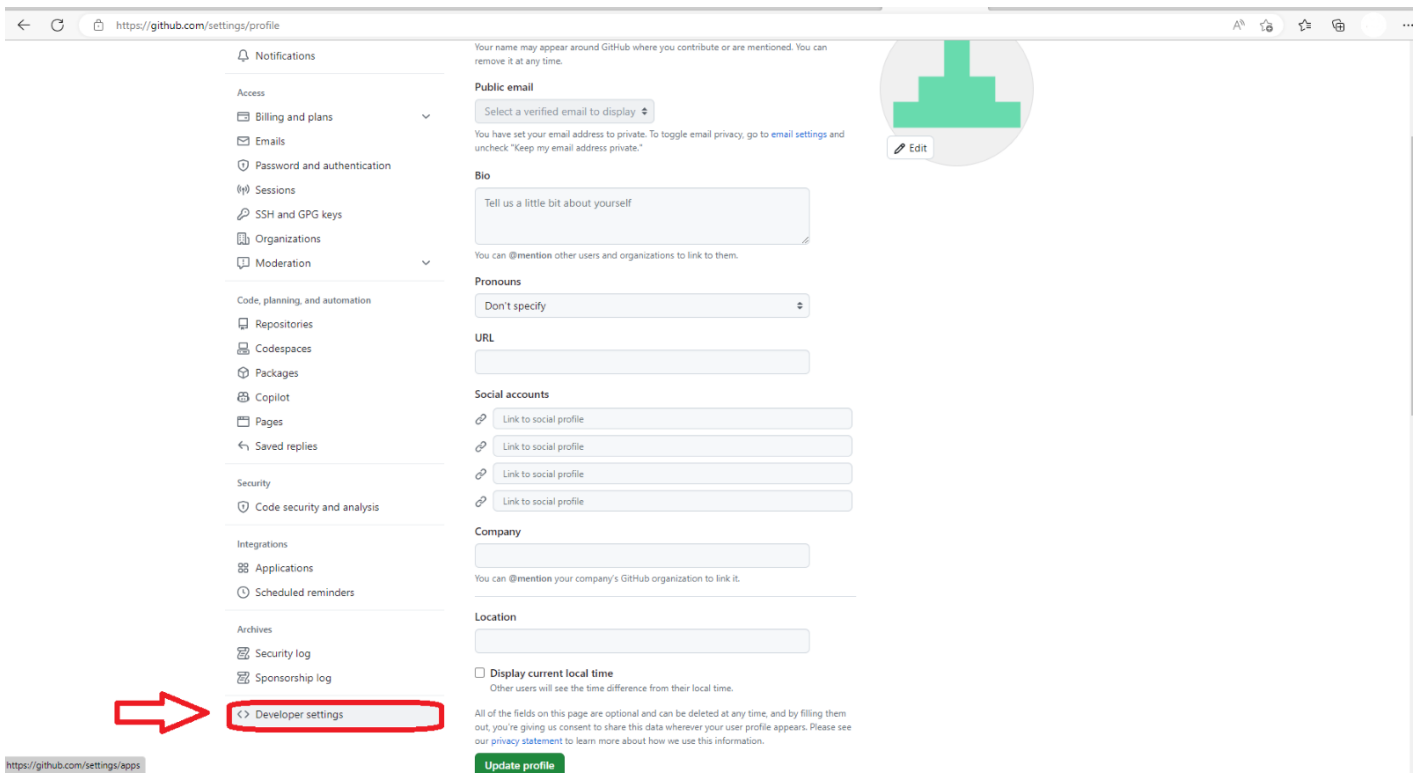
Please provide the following information to CyTech:

### 1. Select Settings



The screenshot shows the GitHub website interface. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Codespaces, Marketplace, and Explore. Below the navigation bar, the main content area is divided into several sections. On the left, there is a 'Create your first project' section with 'Create repository' and 'Import repository' buttons. Below that is a 'Recent activity' section. The main content area is titled 'For you (Beta)' and 'Following'. It features a 'Start writing code' section with a 'Start a new repository' card and an 'Introduce yourself with a profile README' card. The 'Start a new repository' card has a form with a 'username' field and radio buttons for 'Public' and 'Private' visibility. The 'Introduce yourself with a profile README' card has a 'Create' button. On the right side, there is a 'Latest changes' section with a list of recent updates. A user profile menu is open on the right side, showing options like 'Your profile', 'Your repositories', 'Your projects', 'Your stars', 'Your gists', 'Your sponsors', 'Upgrade', 'Try Enterprise', 'Feature preview', 'Help', 'Settings', and 'Sign out'. A red arrow points to the 'Settings' option in the menu.

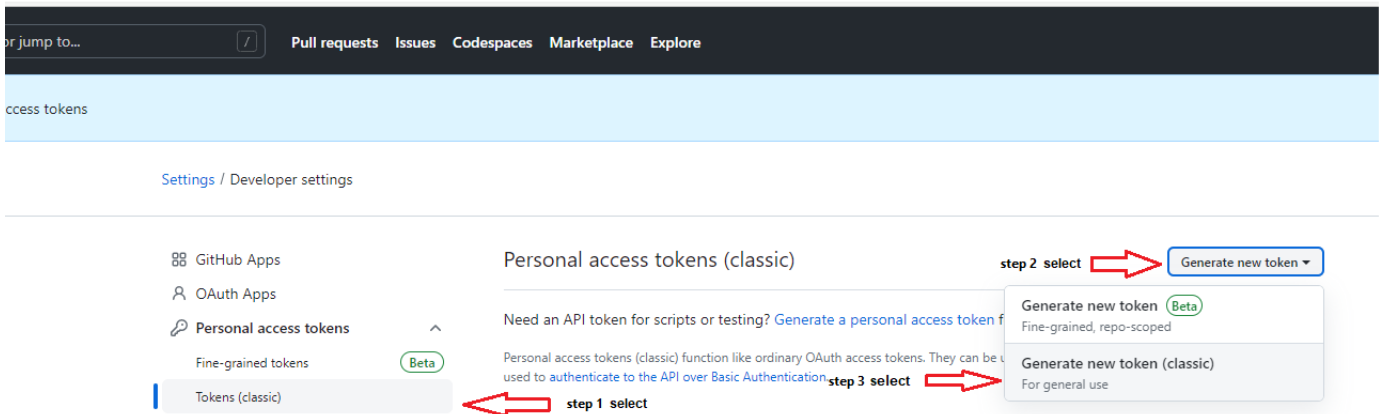
## 2. Select Developer Settings



The screenshot shows the GitHub profile settings page at <https://github.com/settings/profile>. The left sidebar contains various settings categories. The 'Developer settings' option is highlighted with a red box and a red arrow pointing to it. The main content area shows profile information fields such as Public email, Bio, Pronouns, URL, Social accounts, Company, and Location. A green 'Update profile' button is visible at the bottom.

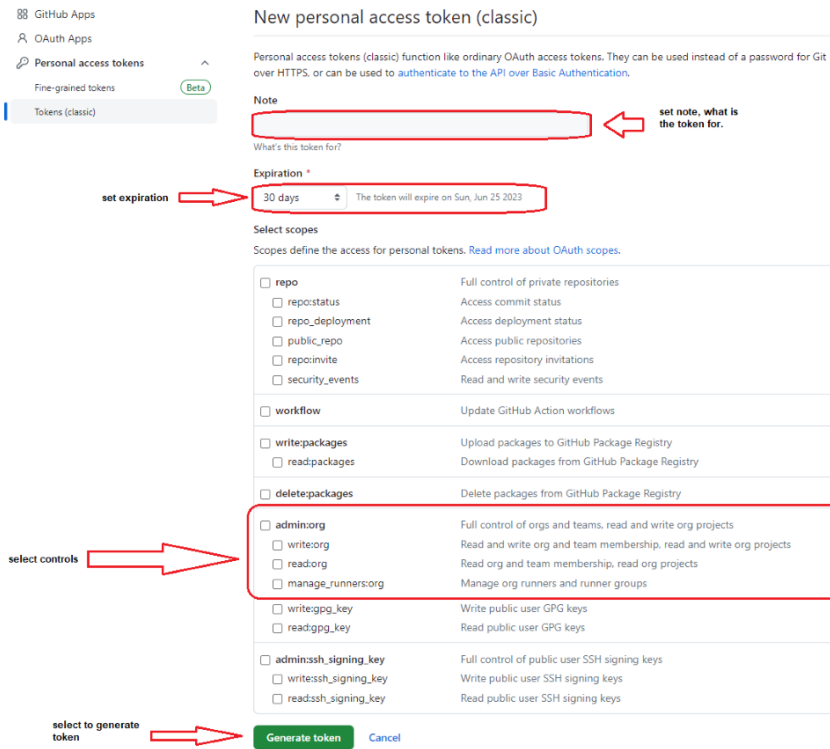
## 3. Select token (classic)

<https://github.com/settings/tokens>



The screenshot shows the GitHub 'Personal access tokens (classic)' page. The left sidebar shows the navigation path: 'Personal access tokens' > 'Tokens (classic)'. A red arrow labeled 'step 1 select' points to 'Tokens (classic)'. The main content area has a 'Generate new token' button, with a red arrow labeled 'step 2 select' pointing to it. A dropdown menu is open, showing two options: 'Generate new token (Beta)' and 'Generate new token (classic)'. A red arrow labeled 'step 3 select' points to 'Generate new token (classic)'. The page also includes a description of personal access tokens and a link to generate a new token.

## 4. Select scope admin:scope



## Collect GitHub logs via API

1. Personal Access Token - the GitHub Personal Access Token. Requires the 'admin:org' scope
2. Organization Name - The GitHub organization name/ID

## GHAS Code Scanning

1. Personal Access Token - the GitHub Personal Access Token. Requires the 'public\_repo' scope for public repositories and 'security\_events' scope for private repositories. \nSee List code scanning alerts for a repository
2. Repository owner - The owner of GitHub Repository. If repository belongs to an organization, owner is name of the organization

## GHAS Dependabot

1. Personal Access Token - The GitHub Personal Access Token. \nSee Authenticating with GraphQL
2. Repository owner - The owner of GitHub Repository

## Github Issues

1. Personal Access Token - the GitHub Personal Access Token.
2. Repository owner - The owner of GitHub Repository. If repository belongs to an organization, owner is name of the organization.

## GHAS Secret Scanning

1. Personal Access Token - the GitHub Personal Access Token. Requires admin access to the repository or organization owning the repository along with a personal access token with 'public\_repo' scope for public repositories and repo or security\_events scope for private repositories.  
See List secret scanning alerts for a repository
2. Repository owner - The owner of GitHub Repository

---

Revision #3

Created 23 April 2024 09:36:31

Updated 19 June 2024 06:54:01