Rate limits for OAuth apps

In this article

About rate limits for OAuth apps

Determining rate limits for an OAuth app

Further reading

Rate limits restrict the rate of traffic to GitHub.com, to help ensure consistent access for all users.

> **Note**: Consider building a GitHub App instead of an OAuth app. The rate limit for GitHub Apps using an installation access token scales with the number of repositories and number of organization users. Conversely, OAuth apps have lower rate limits and do not scale. For more information, see "Differences between GitHub Apps and OAuth apps" and "About creating GitHub Apps."

About rate limits for OAuth apps 🔗

GitHub sets a limit on the number of requests an OAuth app can send to the server within a specific time period. This limit helps to prevent abuse and denial-of-service attacks, and ensures that the system remains available for all users.

GitHub may apply additional secondary rate limits to some actions, to ensure API availability. You can avoid secondary rate limits by following best practices and staying within the rate limit guidelines listed below. For more information about secondary rate limits, see "Best practices for using the REST API" and "Resources in the REST API."

OAuth apps act on behalf of a user, by making requests with a user access token after the user authorizes the app. User access token requests from OAuth apps are authenticated with an OAuth token. For more information, see "Authorizing OAuth apps."

Determining rate limits for an OAuth app 🔗

> You can confirm your current rate limit status at any time using the REST API. For more information, see "Resources in the REST API."

OAuth apps can encounter rate limits during the following two actions:

1. When signing in users.

2. When making API calls.

OAuth apps should always cache their tokens, and only rarely need to sign in a user. Repeatedly signing in a user can be a sign of a bug, most frequently seen as an infinite loop between the app and GitHub. If an app signs the user in ten times within one hour, the next sign in within the same hour will require re-authorization of the application. This ensures the user is aware that the app is minting so many tokens, and provides a break

in what may be an infinite loop otherwise. This ten *sign in* rate limit is distinct from the ten *token* limit also enforced for OAuth apps. For information about the ten token limit, see "[Authorizing OAuth apps](#)."

OAuth apps are also limited to 5,000 requests per hour and per authenticated user. All requests from OAuth apps that are authorized by a user or a personal access token owned by the user, and requests authenticated with any of the user's authentication credentials, share the same quota of 5,000 requests per hour for that user.

OAuth apps are subject to a higher limit of 15,000 requests per hour and per authenticated user when both of the following are true:

- The request is from an OAuth app that's owned or approved by a GitHub Enterprise Cloud organization.
- The authenticated user is a member of the GitHub Enterprise Cloud organization.

For more information about rate limits, see "[Resources in the REST API](#)" and "[Rate limit](#)" in the REST API documentation.

Further reading 🔗

- "[Resource limitations](#)" in the GraphQL API documentation