The REST API is now versioned. For more information, see "About API versioning."

Rate limit

Use the REST API to check your current rate limit status.

About rate limits

You can check your current rate limit status at any time. For more information about rate limit rules, see "Resources in the REST API."

The REST API for searching items has a custom rate limit that is separate from the rate limit governing the other REST API endpoints. For more information, see "Search." The GraphQL API also has a custom rate limit that is separate from and calculated differently than rate limits in the REST API. For more information, see "Resource limitations." For these reasons, the API response categorizes your rate limit. Under `resources`, you'll see objects relating to different categories:

- The `core` object provides your rate limit status for all non-search-related resources in the REST API.
- The `search` object provides your rate limit status for the REST API for searching (excluding code searches). For more information, see "Search."
- The `code_search` object provides your rate limit status for the REST API for searching code. For more information, see "Search."
- The `graphql` object provides your rate limit status for the GraphQL API.
- The `integration_manifest` object provides your rate limit status for the `POST /app-manifests/{code}/conversions` operation. For more information, see "Registering a GitHub App from a manifest."
- The `dependency_snapshots` object provides your rate limit status for submitting snapshots to the dependency graph. For more information, see "Dependency Graph."
- The `code_scanning_upload` object provides your rate limit status for uploading SARIF results to code scanning. For more information, see "Uploading a SARIF file to GitHub."
- The `actions_runner_registration` object provides your rate limit status for registering self-hosted runners in GitHub Actions. For more information, see "Self-hosted runners."

For more information on the headers and values in the rate limit response, see "Resources in the REST API."

Get rate limit status for the authenticated user

✓ Works with GitHub Apps

Note: Accessing this endpoint does not count against your REST API rate limit.

Some categories of endpoints have custom rate limits that are separate from the rate limit governing the other REST API endpoints. For this reason, the API response categorizes your rate limit. Under `resources`, you'll see objects relating to different categories:

- The `core` object provides your rate limit status for all non-search-related resources in the REST API.

- The `search` object provides your rate limit status for the REST API for searching (excluding code searches). For more information, see "Search."
- The `code_search` object provides your rate limit status for the REST API for searching code. For more information, see "Search code."
- The `graphql` object provides your rate limit status for the GraphQL API. For more information, see "Resource limitations."
- The `integration_manifest` object provides your rate limit status for the `POST /app-manifests/{code}/conversions` operation. For more information, see "Creating a GitHub App from a manifest."
- The `dependency_snapshots` object provides your rate limit status for submitting snapshots to the dependency graph. For more information, see "Dependency graph."
- The `code_scanning_upload` object provides your rate limit status for uploading SARIF results to code scanning. For more information, see "Uploading a SARIF file to GitHub."
- The `actions_runner_registration` object provides your rate limit status for registering self-hosted runners in GitHub Actions. For more information, see "Self-hosted runners."
- The `source_import` object is no longer in use for any API endpoints, and it will be removed in the next API version. For more information about API versions, see "API Versions."

Note: The `rate` object is deprecated. If you're writing new API client code or updating existing code, you should use the `core` object instead of the `rate` object. The `core` object contains the same information that is present in the `rate` object.

HTTP response status codes for "Get rate limit status for the authenticated user"

Status code	Description
`200`	OK
`304`	Not modified
`404`	Resource not found

Code samples for "Get rate limit status for the authenticated user"

GET /rate_limit

cURL JavaScript GitHub CLI

```
curl -L \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version: 2022-11-28" \ http(s)://HOSTNAME/api/v3/rate_limit
```

Response

Example response Response schema

Status: 200

```
{ "resources": { "core": { "limit": 5000, "used": 1, "remaining": 4999, "reset": 1691591363 }, "search": { "limit": 30, "used": 12, "remaining": 18, "reset": 1691591091 }, "graphql": { "limit": 5000, "used": 7, "remaining": 4993, "reset": 1691593228 }, "integration_manifest": { "limit": 5000, "used": 1, "remaining": 4999, "reset": 1691594631 }, "source_import": { "limit": 100, "used": 1, "remaining": 99, "reset": 1691591091 }, "code_scanning_upload": { "limit": 500, "used": 1, "remaining": 499, "reset": 1691594631 }, "actions_runner_registration": { "limit": 10000, "used": 0, "remaining": 10000, "reset": 1691594631 }, "scim": { "limit": 15000, "used": 0, "remaining": 15000, "reset": 1691594631 }, "dependency_snapshots": { "limit": 100, "used": 0,
```