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

Users

Use the REST API to get public and private information about authenticated users.

Get the authenticated user 🔗

If the authenticated user is authenticated with an OAuth token with the `user` scope, then the response lists public and private profile information.

If the authenticated user is authenticated through OAuth without the `user` scope, then the response lists only public profile information.

HTTP response status codes for "Get the authenticated user"

Status code	Description
`200`	OK
`304`	Not modified
`401`	Requires authentication
`403`	Forbidden

Code samples for "Get the authenticated user"

Example 1: Status Code 200 ⇕

GET `/user`

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/user
```

Response with public and private profile information

Example response   Response schema

Status: 200

```
{ "login": "octocat", "id": 1, "node_id": "MDQ6VXNlcjE=", "avatar_url": "https://github.com/images/error/octocat_happy.gif",
"gravatar_id": "", "url": "https://HOSTNAME/users/octocat", "html_url": "https://github.com/octocat", "followers_url":
"https://HOSTNAME/users/octocat/followers", "following_url": "https://HOSTNAME/users/octocat/following{/other_user}", "gists_url":
```

https://HOSTNAME/users/octocat/followers", "following_url": "https://HOSTNAME/users/octocat/following{/other_user}", "gists_url":
"https://HOSTNAME/users/octocat/gists{/gist_id}", "starred_url": "https://HOSTNAME/users/octocat/starred{/owner}{/repo}",
"subscriptions_url": "https://HOSTNAME/users/octocat/subscriptions", "organizations_url": "https://HOSTNAME/users/octocat/orgs",
"repos_url": "https://HOSTNAME/users/octocat/repos", "events_url": "https://HOSTNAME/users/octocat/events{/privacy}",

Update the authenticated user 🔗

Note: If your email is set to private and you send an `email` parameter as part of this request to update your profile, your privacy settings are still enforced: the email address will not be displayed on your public profile or via the API.

Parameters for "Update the authenticated user"

Headers

`accept`   string

Setting to `application/vnd.github+json` is recommended.

Body parameters

`name`   string

The new name of the user.

`email`   string

The publicly visible email address of the user.

`blog`   string

The new blog URL of the user.

`twitter_username`   string or null

The new Twitter username of the user.

`company`   string

The new company of the user.

`location`   string

The new location of the user.

`hireable`   boolean

The new hiring availability of the user.

`bio`   string

The new short biography of the user.

HTTP response status codes for "Update the authenticated user"

Status code	Description
200	OK
304	Not modified

401	Requires authentication
403	Forbidden
404	Resource not found
422	Validation failed, or the endpoint has been spammed.

Code samples for "Update the authenticated user"

`PATCH` `/user`

cURL   JavaScript   GitHub CLI

```
curl -L \ -X PATCH \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version:
2022-11-28" \ http(s)://HOSTNAME/api/v3/user \ -d '{"blog":"https://github.com/blog","name":"monalisa octocat"}'
```

Response

Example response   Response schema

Status: 200

```
{ "login": "octocat", "id": 1, "node_id": "MDQ6VXNlcjE=", "avatar_url": "https://github.com/images/error/octocat_happy.gif",
"gravatar_id": "", "url": "https://HOSTNAME/users/octocat", "html_url": "https://github.com/octocat", "followers_url":
"https://HOSTNAME/users/octocat/followers", "following_url": "https://HOSTNAME/users/octocat/following{/other_user}", "gists_url":
"https://HOSTNAME/users/octocat/gists{/gist_id}", "starred_url": "https://HOSTNAME/users/octocat/starred{/owner}{/repo}",
"subscriptions_url": "https://HOSTNAME/users/octocat/subscriptions", "organizations_url": "https://HOSTNAME/users/octocat/orgs",
"repos_url": "https://HOSTNAME/users/octocat/repos", "events_url": "https://HOSTNAME/users/octocat/events{/privacy}",
```

List users 🔗

✔ Works with [GitHub Apps](#)

Lists all users, in the order that they signed up on GitHub Enterprise Server. This list includes personal user accounts and organization accounts.

Note: Pagination is powered exclusively by the `since` parameter. Use the [Link header](#) to get the URL for the next page of users.

Parameters for "List users"

Headers

`accept` string

Setting to `application/vnd.github+json` is recommended.

Query parameters

`since` integer

A user ID. Only return users with an ID greater than this ID.

---

per_page   integer

The number of results per page (max 100).

Default: `30`

HTTP response status codes for "List users"

Status code	Description
`200`	OK
`304`	Not modified

Code samples for "List users"

> **GET** `/users`

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/users
```

Response

Example response   Response schema

Status: 200

```
[ { "login": "octocat", "id": 1, "node_id": "MDQ6VXNlcjE=", "avatar_url": "https://github.com/images/error/octocat_happy.gif",
"gravatar_id": "", "url": "https://HOSTNAME/users/octocat", "html_url": "https://github.com/octocat", "followers_url":
"https://HOSTNAME/users/octocat/followers", "following_url": "https://HOSTNAME/users/octocat/following{/other_user}", "gists_url":
"https://HOSTNAME/users/octocat/gists{/gist_id}", "starred_url": "https://HOSTNAME/users/octocat/starred{/owner}{/repo}",
"subscriptions_url": "https://HOSTNAME/users/octocat/subscriptions", "organizations_url": "https://HOSTNAME/users/octocat/orgs",
"repos_url": "https://HOSTNAME/users/octocat/repos", "events_url": "https://HOSTNAME/users/octocat/events{/privacy}",
```

Get a user 🔗

✓ Works with [GitHub Apps](#)

Provides publicly available information about someone with a GitHub account.

GitHub Apps with the `Plan` user permission can use this endpoint to retrieve information about a user's GitHub Enterprise Server plan. The GitHub App must be authenticated as a user. See "[Identifying and authorizing users for GitHub Apps](#)" for details about authentication. For an example response, see 'Response with GitHub Enterprise Server plan information' below"

The `email` key in the following response is the publicly visible email address from your GitHub Enterprise Server [profile page](#). When setting up your profile, you can select a primary email address to be "public" which provides an email entry for this endpoint. If you do not set a public email address for `email`, then it will have a value of `null`. You only see publicly visible email addresses when authenticated with GitHub Enterprise Server. For more information, see

[Authentication](#).

The Emails API enables you to list all of your email addresses, and toggle a primary email to be visible publicly. For more information, see "[Emails API](#)".

Parameters for "Get a user"

Headers

accept string

Setting to `application/vnd.github+json` is recommended.

Path parameters

username string Required

The handle for the GitHub user account.

HTTP response status codes for "Get a user"

Status code	Description
200	OK
404	Resource not found

Code samples for "Get a user"

Example 1: Status Code 200 ⇅

> GET /users/{username}

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/users/USERNAME
```

Default response

Example response   Response schema

Status: 200

```
{ "login": "octocat", "id": 1, "node_id": "MDQ6VXNlcjE=", "avatar_url": "https://github.com/images/error/octocat_happy.gif",
"gravatar_id": "", "url": "https://HOSTNAME/users/octocat", "html_url": "https://github.com/octocat", "followers_url":
"https://HOSTNAME/users/octocat/followers", "following_url": "https://HOSTNAME/users/octocat/following{/other_user}", "gists_url":
"https://HOSTNAME/users/octocat/gists{/gist_id}", "starred_url": "https://HOSTNAME/users/octocat/starred{/owner}{/repo}",
"subscriptions_url": "https://HOSTNAME/users/octocat/subscriptions", "organizations_url": "https://HOSTNAME/users/octocat/orgs",
"repos_url": "https://HOSTNAME/users/octocat/repos", "events_url": "https://HOSTNAME/users/octocat/events{/privacy}",
```

Get contextual information for a user 🔗

Provides hovercard information when authenticated through basic auth or OAuth with the `repo` scope. You can find out more about someone in relation to their pull requests, issues, repositories, and organizations.

The `subject_type` and `subject_id` parameters provide context for the person's hovercard, which returns more information than without the parameters. For example, if you wanted to find out more about `octocat` who owns the `Spoon-Knife` repository via cURL, it would look like this:

```
curl -u username:token
  https://api.github.com/users/octocat/hovercard?subject_type=repository&subject_id=1300192
```

Parameters for "Get contextual information for a user"

Headers

accept string

Setting to `application/vnd.github+json` is recommended.

Path parameters

username string Required

The handle for the GitHub user account.

Query parameters

subject_type string

Identifies which additional information you'd like to receive about the person's hovercard. Can be `organization`, `repository`, `issue`, `pull_request`. **Required** when using `subject_id`.

Can be one of: `organization`, `repository`, `issue`, `pull_request`

subject_id string

Uses the ID for the `subject_type` you specified. **Required** when using `subject_type`.

HTTP response status codes for "Get contextual information for a user"

Status code	Description
200	OK
404	Resource not found
422	Validation failed, or the endpoint has been spammed.

Code samples for "Get contextual information for a user"
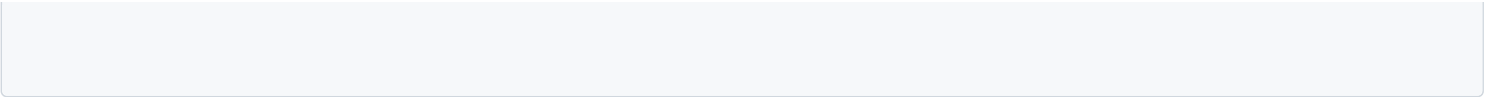
GET /users/{username}/hovercard

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/users/USERNAME/hovercard
```

Response

Example response | Response schema

Status: 200

```
{ "contexts": [ { "message": "Owns this repository", "octicon": "repo" } ] }
```