

Access Token Scope



- [Repository and organization access](#)
 - [All \(public, private, and limited\)](#)
 - [Public only](#)
 - [Specific repositories](#)
 - [Administrator capabilities](#)

Forgejo supports scoped access tokens, which allow users to restrict tokens to operate only on selected URL routes. Scopes are grouped by high-level API routes, and further refined as follows:

- `read`: `GET` routes
- `write`: `POST`, `PUT`, `PATCH`, and `DELETE` routes (in addition to `GET`)

Forgejo token scopes are as follows:

Name	Description
(no scope)	Not supported. A scope is required even for public repositories.
activitypub	<code>activitypub</code> API routes: ActivityPub related operations.
read:activitypub	Grants read access for ActivityPub operations.
write:activitypub	Grants read/write/delete access for ActivityPub operations.
admin	<code>/admin/*</code> API routes: Site-wide administrative operations (hidden for non-admin accounts).
read:admin	Grants read access for admin operations, such as getting cron jobs or registered user emails.
write:admin	Grants read/write/delete access for admin operations, such as running cron jobs or updating user accounts.
issue	<code>issues/*</code> , <code>labels/*</code> , <code>milestones/*</code> API routes: Issue-related operations.
read:issue	Grants read access for issue operations, such as getting issue comments, issue attachments, and milestones.

Name	Description
write:issue	Grants read/write/delete access for issue operations, such as posting or editing an issue comment or attachment, and updating milestones.
misc	Miscellaneous and settings top-level API routes.
read:misc	Grants read access to miscellaneous operations, such as getting label and gitignore templates.
write:misc	Grants read/write/delete access to miscellaneous operations, such as markup utility operations.
notification	<code>notification/*</code> API routes: User notification operations.
read:notification	Grants read access to user notifications, such as which notifications users are subscribed to and reading new notifications.
write:notification	Grants read/write/delete access to user notifications, such as marking notifications as read.
organization	<code>orgs/*</code> and <code>teams/*</code> API routes: Organization and team management operations.
read:organization	Grants read access to org and team status, such as listing all organizations a user has visibility to, teams, and team members.
write:organization	Grants read/write/delete access to org and team status, such as creating and updating teams and updating org settings.
package	<code>/packages/*</code> API routes: Packages operations.
read:package	Grants read access to package operations, such as reading and downloading available packages.
write:package	Grants read/write/delete access to package operations. Currently the same as <code>read:package</code> .
repository	<code>/repos/*</code> API routes except <code>/repos/issues/*</code> : Repository file, pull-request, and release operations.
read:repository	Grants read access to repository operations, such as getting repository files, releases, and collaborators.
write:repository	Grants read/write/delete access to repository operations, such as getting and updating repository files, creating pull requests, and updating collaborators.
user	<code>/user/*</code> and <code>/users/*</code> API routes: User-related operations.
read:user	Grants read access to user operations, such as getting user repository subscriptions and user settings.
write:user	Grants read/write/delete access to user operations, such as updating user repository subscriptions, followed users, and user settings.

Repository and organization access

When creating a Forgejo access token, three options are available for which repositories and organizations the access tokens can access: [All \(public, private, and limited\)](#), [Public only](#), and [Specific repositories](#).

It is recommended to create the most restrictive access token possible that meets a user's functional needs. If an access token is compromised in the future, a more restrictive access token will have a smaller impact on a user's operations.

All (public, private, and limited)

Access tokens created with the "All (public, private, and limited)" option will have no restrictions placed upon them. Any permissions can be selected.

Public only

Access tokens created with the "Public only" option will be limited, preventing access to private resources that the user would normally have access to. Any permissions can be selected, but APIs affecting these objects will be limited:

- Repository access will be limited to public repositories.
- Issue access will be limited to issues in public repositories.
- Organization access will be limited to public organizations.
- User access will be limited to public users.
- Notification access will be limited to notifications on public repositories.
- Package access will be limited to packages owned by a public user or organization.

Additional capabilities which are usually [granted to Forgejo site administrators and repository administrators](#) are restricted when using a public only token.

Specific repositories

Access tokens created with the "Specific repositories" option will be limited to the repositories selected.

All public repositories are accessible, but access is limited to read-only APIs if the public repository is not included in the specified repositories.

Only the `read:repository`, `write:repository`, `read:issue`, and `write:issue` permissions can be used for a specific repository access token. Other scopes are not permitted because the API would not have a related repository to check against the selected list.

Specific repository access tokens cannot be used to perform administrative operations within a repository, even if the token owner could usually perform these actions. For example, the token cannot be used to transfer the repository to a new owner, cannot be used to add a collaborator to the repository, cannot change the repository from private to public.

Additional capabilities which are usually [granted to Forgejo site administrators and repository administrators](#) are restricted when using a specific repository token.

Administrator capabilities

Forgejo site administrators and repository administrators have elevated capabilities, unless a public only or specific repository access token is used.

For example, site administrators can perform actions such as, but not limited to:

- impersonate other users by adding `?sudo={username}` to the API endpoints,
- have `write:repository` access to all repositories,
- access `GET /users/{username}/activities/feeds` for any user,
- transfer repository ownership immediately without confirmation,
- bypass quota restrictions,
- and create repositories for other users.

Repository administrators can perform actions such as, but not limited to:

- view tracked time entered by other users, and add tracked time for other users, on an issue in the repository,
- query the permissions of other users against the repository,
- add and remove collaborators,
- and convert a mirror repository into a normal repository.

🔒 #2

🔒 🔒 🔒 2026-10-31 13:36:40 UTC

🔒 🔒 🔒 2026-10-31 13:36:41 UTC