Repository roles for an organization

In this article

You can customize access to each repository in your organization by assigning granular roles, giving people access to the features and tasks they need.

Repository roles for organizations

You can give organization members, outside collaborators, and teams of people different levels of access to repositories owned by an organization by assigning them to roles. Choose the role that best fits each person or team's function in your project without giving people more access to the project than they need.

From least access to most access, the roles for an organization repository are:

- **Read**: Recommended for non-code contributors who want to view or discuss your project
- **Triage**: Recommended for contributors who need to proactively manage issues and pull requests without write access
- **Write**: Recommended for contributors who actively push to your project
- **Maintain**: Recommended for project managers who need to manage the repository without access to sensitive or destructive actions
- **Admin**: Recommended for people who need full access to the project, including sensitive and destructive actions like managing security or deleting a repository

You can create custom repository roles. For more information, see "Managing custom repository roles for an organization."

Organization owners can set base permissions that apply to all members of an organization when accessing any of the organization's repositories. For more information, see "Setting base permissions for an organization."

Organization owners can also choose to further limit access to certain settings and actions across the organization. For more information on options for specific settings, see "Managing organization settings."

In addition to managing organization-level settings, organization owners have admin access to every repository owned by the organization. For more information, see "Roles in an organization."

Warning: When someone adds a deploy key to a repository, any user who has the private key can read from or write to the repository (depending on the key settings), even if they're later removed from the organization.

Permissions for each role

Note: The roles required to use security features are listed in "Access requirements for security features" below.

Repository action	Read	Triage	Write	Maintain	Admin
Manage individual, team, and outside collaborator access to the repository	×	×	×	×	✓
Pull from the person or team's assigned repositories	✓	✓	✓	✓	✓
Fork the person or team's assigned repositories	✓	✓	✓	✓	✓
Edit and delete their own comments	✓	✓	✓	✓	✓
Open issues	✓	✓	✓	✓	✓
Close issues they opened themselves	✓	✓	✓	✓	✓
Reopen issues they closed themselves	✓	✓	✓	✓	✓
Have an issue assigned to them	✓	✓	✓	✓	✓
Send pull requests from forks of the team's assigned repositories	✓	✓	✓	✓	✓
Submit reviews on pull requests	✓	✓	✓	✓	✓
Approve or request changes to a	×	×	✓	✓	✓

changes to a pull request with required reviews					
Apply suggested changes to pull requests	✕	✕	✓	✓	✓
View published releases	✓	✓	✓	✓	✓
Edit wikis in public repositories	✓	✓	✓	✓	✓
Edit wikis in private repositories	✕	✕	✓	✓	✓
Apply/dismiss labels	✕	✓	✓	✓	✓
Create, edit, delete labels	✕	✕	✓	✓	✓
Close, reopen, and assign all issues and pull requests	✕	✓	✓	✓	✓
Enable and disable auto-merge on a pull request	✕	✕	✓	✓	✓
Apply milestones	✕	✓	✓	✓	✓
Mark duplicate issues and pull requests	✕	✓	✓	✓	✓
Request pull request reviews	✕	✓	✓	✓	✓
Merge a pull request	✕	✕	✓	✓	✓
Push to (write) the person or team's assigned repositories	✕	✕	✓	✓	✓
Edit and delete anyone's comments on commits, pull	✕	✕	✓	✓	✓

commits, pull requests, and issues					
Hide anyone's comments	×		✓	✓	✓
Transfer issues (see "Transferring an issue to another repository" for details)	×	×	✓	✓	✓
Act as a designated code owner for a repository	×	×	✓	✓	✓
Mark a draft pull request as ready for review	×	×	✓	✓	✓
Convert a pull request to a draft	×	×	✓	✓	✓
Create status checks	×	×	✓	✓	✓
Create and edit releases	×	×	✓	✓	✓
View draft releases	×	×	✓	✓	✓
Edit a repository's description	×	×	×	✓	✓
Manage topics	×	×	×	✓	✓
Enable wikis and restrict wiki editors	×	×	×	✓	✓
Enable project boards	×	×	×	✓	✓
Configure pull request merges	×	×	×	✓	✓
Configure a publishing source for GitHub Pages	×	×	×	✓	✓
Manage branch protection	×	×	×	×	✓

protection rules					
Merge pull requests on protected branches, even if there are no approving reviews	×	×	×	×	✓
Create tags that match a tag protection rule	×	×	×	✓	✓
Delete tags that match a tag protection rule	×	×	×	×	✓
Create and edit repository social cards	×	×	×	✓	✓
Delete an issue (see "Deleting an issue")	×	×	×	×	✓
Define code owners for a repository	×	×	✓	✓	✓
Add a repository to a team (see "Managing team access to an organization repository" for details)	×	×	×	×	✓
Manage outside collaborator access to a repository	×	×	×	×	✓
Change a repository's visibility	×	×	×	×	✓
Make a repository a template (see "Creating a template repository")	×	×	×	×	✓
Change a	×	×	×	×	✓

Change a repository's settings	✗	✗	✗	✗	✓
Manage team and collaborator access to the repository	✗	✗	✗	✗	✓
Edit the repository's default branch	✗	✗	✗	✗	✓
Rename the repository's default branch (see "Renaming a branch")	✗	✗	✗	✗	✓
Rename a branch other than the repository's default branch (see "Renaming a branch")	✗	✗	✓	✓	✓
Manage webhooks and deploy keys	✗	✗	✗	✗	✓
Manage the forking policy for a repository	✗	✗	✗	✗	✓
Transfer repositories into the organization	✗	✗	✗	✗	✓
Delete or transfer repositories out of the organization	✗	✗	✗	✗	✓
Archive repositories	✗	✗	✗	✗	✓
Create autolink references to external resources, like Jira or Zendesk (see "Configuring autolinks to reference external	✗	✗	✗	✗	✓

[external resources](#)")					
[Enable GitHub Discussions](#) in a repository	×	×	×	✓	✓
[Create and edit categories](#) for GitHub Discussions	×	×	✓	✓	✓
[Move a discussion to a different category](#)	×		✓	✓	✓
[Manage pinned discussions](#)	×	×	✓	✓	✓
[Convert issues to discussions in bulk](#)	×	×	✓	✓	✓
[Lock and unlock discussions](#)	×	✓	✓	✓	✓
[Individually convert issues to discussions](#)	×	✓	✓	✓	✓
[Create new discussions and comment on existing discussions](#)	✓	✓	✓	✓	✓
[Delete a discussion](#)	×	✓	✓	✓	✓

Access requirements for security features

In this section, you can find the access required for security features, such as Advanced Security features.

Repository action	Read	Triage	Write	Maintain	Admin
Receive [Dependabot alerts for insecure dependencies](#) in a repository	×	×	✓	✓	✓
[Dismiss Dependabot](#)	×	×	✓	✓	✓

alerts					
Designate additional people or teams to receive security alerts	✗	✗	✗	✗	✓
Manage access to GitHub Advanced Security features (see "Managing security and analysis settings for your organization")	✗	✗	✗	✗	✓
View dependency reviews	✓	✓	✓	✓	✓
View code scanning alerts on pull requests	✓	✓	✓	✓	✓
List, dismiss, and delete code scanning alerts	✗	✗	✓	✓	✓
View and dismiss secret scanning alerts in a repository	✗	✗	✓	✓	✓
Resolve, revoke, or re-open secret scanning alerts	✗	✗	✓	✓	✓
Designate additional people or teams to receive secret scanning alerts **in repositories**	✗	✗	✗	✗	✓

Note: Repository writers and maintainers can only see secret scanning alert information for their own commits.

Further reading

- "Managing user access to your organization's repositories"
- "Adding outside collaborators to repositories in your organization"
- "Project (classic) permissions for an organization"