Managing GitHub Actions settings for a repository

In this article

You can disable or configure GitHub Actions for a specific repository.

Note: GitHub-hosted runners are not currently supported on GitHub Enterprise Server. You can see more information about planned future support on the GitHub public roadmap.

About GitHub Actions permissions for your repository

By default, after GitHub Actions is enabled on your GitHub Enterprise Server instance, it is enabled on all repositories and organizations. You can choose to disable GitHub Actions or limit it to actions in your enterprise. For more information about GitHub Actions, see "Learn GitHub Actions."

You can enable GitHub Actions for your repository. When you enable GitHub Actions, workflows are able to run actions located within your repository and any other public or internal repository. You can disable GitHub Actions for your repository altogether. When you disable GitHub Actions, no workflows run in your repository.

Alternatively, you can enable GitHub Actions in your repository but limit the actions a workflow can run.

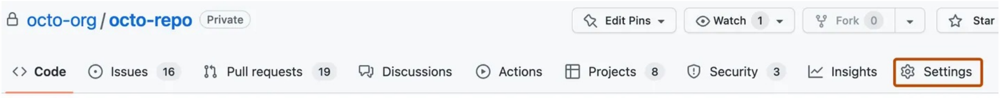
Managing GitHub Actions permissions for your repository

You can disable GitHub Actions for a repository, or set a policy that configures which actions can be used in the repository.

Note: You might not be able to manage these settings if your organization has an overriding policy or is managed by an enterprise that has overriding policy. For more information, see "Disabling or limiting GitHub Actions for your organization" or "Enforcing policies for GitHub Actions in your enterprise."

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.

2. Under your repository name, click ⚙ **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click ▶ **Actions**, then click **General**.

4. Under "Actions permissions", select an option.

   If you choose **Allow select actions**, actions within your enterprise are allowed, and there are additional options for allowing other specific actions. For more information, see "Allowing select actions to run."

5. Click **Save**.

Allowing select actions to run

When you choose **Allow select actions**, local actions are allowed, and there are additional options for allowing other specific actions:

- **Allow actions created by GitHub:** You can allow all actions created by GitHub to be used by workflows. Actions created by GitHub are located in the `actions` and `github` organizations. For more information, see the `actions` and `github` organizations.

- **Allow Marketplace actions by verified creators:** This option is available if you have GitHub Connect enabled and configured with GitHub Actions. For more information, see "Enabling automatic access to GitHub.com actions using GitHub Connect." You can allow all GitHub Marketplace actions created by verified creators to be used by workflows. When GitHub has verified the creator of the action as a partner organization, the ✓ badge is displayed next to the action in GitHub Marketplace.

- **Allow specified actions:** You can restrict workflows to use actions in specific organizations and repositories. Specified actions cannot be set to more than 1000.

  To restrict access to specific tags or commit SHAs of an action, use the same syntax used in the workflow to select the action.

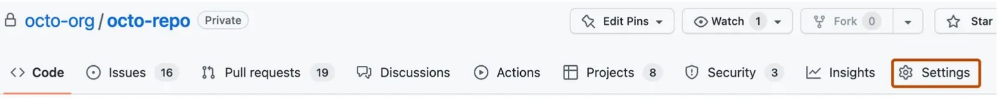
  - For an action, the syntax is `OWNER/REPOSITORY@TAG-OR-SHA`. For example, use `actions/javascript-action@v1.0.1` to select a tag or `actions/javascript-action@a824008085750b8e136effc585c3cd6082bd575f` to select a SHA. For more information, see "Finding and customizing actions."

  You can use the `*` wildcard character to match patterns. For example, to allow all actions in organizations that start with `space-org`, you can specify `space-org*/*`. To allow all actions in repositories that start with octocat, you can use `*/octocat**@*`. For more information about using the `*` wildcard, see "Workflow syntax for GitHub Actions."

This procedure demonstrates how to add specific actions to the allow list.

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.

2. Under your repository name, click **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click **Actions**, then click **General**.

4. Under "Actions permissions", select **Allow select actions** and add your required actions to the list.

5. Click **Save**.

Enabling workflows for forks of private repositories

If you rely on using forks of your private repositories, you can configure policies that control how users can run workflows on `pull_request` events. Available to private and internal repositories only, you can configure these policy settings for your enterprise, organizations, or repositories.

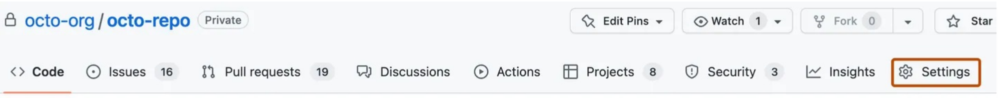
If a policy is disabled for an enterprise or organization, it cannot be enabled for a repository.

- **Run workflows from fork pull requests** - Allows users to run workflows from fork pull requests, using a `GITHUB_TOKEN` with read-only permission, and with no access to secrets.
- **Send write tokens to workflows from pull requests** - Allows pull requests from forks to use a `GITHUB_TOKEN` with write permission.
- **Send secrets to workflows from pull requests** - Makes all secrets available to the pull request.
- **Require approval for fork pull request workflows** - Workflow runs on pull requests from collaborators without write permission will require approval from someone with write permission before they will run.

Configuring the fork policy for a private repository

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.

2. Under your repository name, click **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click **Actions**, then click **General**.

4. Under **Fork pull request workflows**, select your options.

5. Click **Save** to apply the settings.

Setting the permissions of the `GITHUB_TOKEN` for your repository

You can set the default permissions granted to the `GITHUB_TOKEN`. For more information about the `GITHUB_TOKEN`, see "Automatic token authentication." You can choose a restricted set of permissions as the default, or apply permissive settings.

The default permissions can also be configured in the organization settings. If your repository belongs to an organization and a more restrictive default has been selected in the organization settings, the same option is selected in your repository settings and the permissive option is disabled.

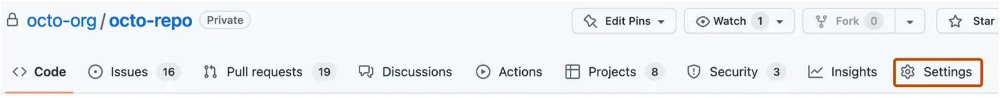
Anyone with write access to a repository can modify the permissions granted to the `GITHUB_TOKEN`, adding or removing access as required, by editing the `permissions` key in the workflow file. For more information, see `permissions`.

Configuring the default `GITHUB_TOKEN` permissions

By default, when you create a new repository in your personal account, `GITHUB_TOKEN` only has read access for the `contents` and `packages` scopes. If you create a new repository in an organization, the setting is inherited from what is configured in the organization settings.

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.

2. Under your repository name, click ⚙ **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click ▶ **Actions**, then click **General**.

4. Under "Workflow permissions", choose whether you want the `GITHUB_TOKEN` to have read and write access for all scopes (the permissive setting), or just read access for the `contents` and `packages` scopes (the restricted setting).

5. Click **Save** to apply the settings.

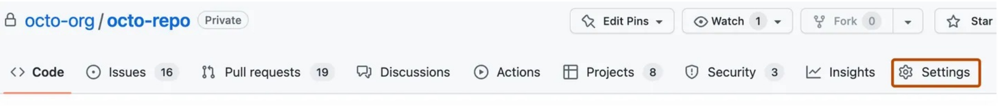
Preventing GitHub Actions from creating or approving pull requests

You can choose to allow or prevent GitHub Actions workflows from creating or approving pull requests.

By default, when you create a new repository in your personal account, workflows are not allowed to create or approve pull requests. If you create a new repository in an organization, the setting is inherited from what is configured in the organization settings.

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.

2. Under your repository name, click ⚙ **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click **Actions**, then click **General**.

4. Under "Workflow permissions", use the **Allow GitHub Actions to create and approve pull requests** setting to configure whether `GITHUB_TOKEN` can create and approve pull requests.

5. Click **Save** to apply the settings.

Allowing access to components in an internal repository

Actions and reusable workflows in your internal repositories can be shared with internal and private repositories in the same organization or enterprise. For information about internal repositories, see "About repositories."

You can use the steps below to configure whether actions and reusable workflows in an internal repository can be accessed from outside the repository. For more information, see "Sharing actions and workflows with your enterprise." Alternatively, you can use the REST API to set, or get details of the level of access. For more information, see "GitHub Actions Permissions" and "GitHub Actions Permissions."

1. On GitHub, navigate to the main page of the internal repository.

2. Under your repository name, click **Settings**.

3. In the left sidebar, click **Actions**, then click **General**.

4. Under **Access**, choose one of the access settings:

   - **Not accessible** - Workflows in other repositories cannot access this repository.
   - **Accessible from repositories in the 'ORGANIZATION NAME' organization** - Workflows in other repositories that are part of the 'ORGANIZATION NAME' organization can access the actions and reusable workflows in this repository. Access is allowed only from private or internal repositories.
   - **Accessible from repositories in the 'ENTERPRISE NAME' enterprise** - Workflows in other repositories that are part of the 'ENTERPRISE NAME' enterprise can access the actions and reusable workflows in this repository. Access is allowed only from private or internal repositories.

5. Click **Save** to apply the settings.

Allowing access to components in a private repository

Actions and reusable workflows in your private repositories can be shared with other private repositories in the same organization or enterprise. For information about private repositories, see "About repositories."

You can use the steps below to configure whether actions and reusable workflows in a private repository can be accessed from outside the repository. For more information, see "Sharing actions and workflows with your enterprise." Alternatively, you can use the REST API to set

REST API to set, or get details of the level of access. For more information, see "GitHub Actions Permissions" and "GitHub Actions Permissions."

1. On GitHub, navigate to the main page of the private repository.

2. Under your repository name, click **Settings**.

3. In the left sidebar, click **Actions**, then click **General**.

4. Under **Access**, choose one of the access settings:

   - **Not accessible** - Workflows in other repositories cannot access this repository.
   - **Accessible from repositories in the 'ORGANIZATION NAME' organization** - Workflows in other repositories that are part of the 'ORGANIZATION NAME' organization can access the actions and reusable workflows in this repository. Access is allowed only from private repositories.
   - **Accessible from repositories in the 'ENTERPRISE NAME' enterprise** - Workflows in other repositories that are part of the 'ENTERPRISE NAME' enterprise can access the actions and reusable workflows in this repository. Access is allowed only from private repositories.

5. Click **Save** to apply the settings.

Configuring the retention period for GitHub Actions artifacts and logs in your repository

You can configure the retention period for GitHub Actions artifacts and logs in your repository.

By default, the artifacts and log files generated by workflows are retained for 90 days before they are automatically deleted. You can change this retention period to anywhere between 1 day or 400 days.

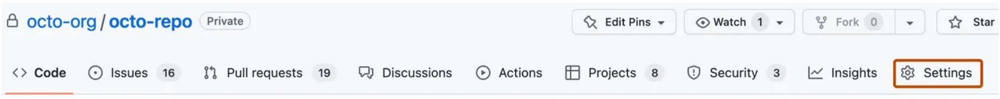
When you customize the retention period, it only applies to new artifacts and log files, and does not retroactively apply to existing objects. For managed repositories and organizations, the maximum retention period cannot exceed the limit set by the managing organization or enterprise.

You can also define a custom retention period for a specific artifact created by a workflow. For more information, see "Removing workflow artifacts."

Setting the retention period for a repository

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.

2. Under your repository name, click **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click **Actions**, then click **General**.

4. In the "Artifact, log, and cache settings" section, under **Artifact and log**

retention, enter a new value.

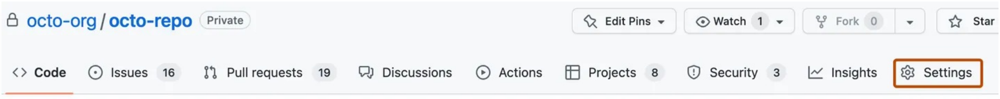
5. Click **Save** to apply the change.

Configuring cache storage for a repository

By default, the total cache storage that GitHub Actions uses on the external storage for your GitHub Enterprise Server instance is limited to a maximum of 10 GB per repository, and the maximum allowed size that can be set for a repository is 25 GB. However, these default sizes might be different if an enterprise owner has changed them. If you exceed the limit, GitHub will save the new cache but will begin evicting caches until the total size is less than the repository limit.

You can set a total cache storage size for your repository up to the maximum size allowed by the organization or enterprise policy settings.

1. On your GitHub Enterprise Server instance, navigate to the main page of the repository.
2. Under your repository name, click ⚙ **Settings**. If you cannot see the "Settings" tab, select the ··· dropdown menu, then click **Settings**.



3. In the left sidebar, click ⊙ **Actions**, then click **General**.
4. Under **Cache size limit**, enter a new value.
5. Click **Save** to apply the change.

Legal

Terms Privacy Status Pricing Expert services Blog