About locked repositories

In this article

Repositories can be locked to prevent changes, often for migrations.

About locked repositories 🔗

When you migrate repositories to or from GitHub products, your origin and destination repositories may be "locked" for migration. While a repository is locked, you cannot make any changes to the repository, such as pushing commits, creating issues, or commenting on pull requests.

Whether your repositories will be locked during migration depends on the tooling you use and the options you choose when you run the migration. When a repository is locked, a banner with the following text is displayed on the repository's page on GitHub:

> This repository is currently being migrated. It's locked while the migration is in progress.

Often, repositories are unlocked automatically when the migration is complete. In other cases, unlocking a repository is a manual step, and the process required to unlock a repository depends on the migration tool you used.

Repositories locked by GitHub Enterprise Importer 🔗

While a migration is in progress, access to the destination repository is locked by GitHub Enterprise Importer. If the migration completes successfully, the repository will unlock automatically. However, if there's a problem with the migration, including a migration failure, the repository may remain locked.

GitHub Enterprise Importer does not lock source repositories by default. Source repositories will only be locked if you specify the `--lock-source-repo` option in the GitHub CLI, or the `lockSource` attribute in the `startRepositoryMigration` GraphQL mutation.

> **Note:** We do not recommend locking source repositories unless you are certain you will not want to unlock them later. Consider archiving the repositories instead. For more information, see "[Archiving repositories](#)."

For information about how to unlock repositories that were locked by GitHub Enterprise Importer, see "[Troubleshooting your migration with GitHub Enterprise Importer](#)."

Repositories locked by the "Organization migrations" REST API 🔗

When you call the "Start an organization migration" endpoint to generate a migration archive for a source repository, the repository is not locked by default. The repository is only locked if you set the `lock_repositories` parameter to `true`.

If you lock a repository via this endpoint, you can unlock the repository using the "Unlock an organization repository" endpoint.

If the repository is stored on GitHub Enterprise Server, a site administrator can also unlock the repository using the site admin dashboard. For more information, see "Locking a repository" in the GitHub Enterprise Server documentation.

Repositories locked by `ghe-migrator` 🔗

When you use `ghe-migrator`, the destination repository on GitHub Enterprise Server is locked by default and is not automatically unlocked.

If the import succeeded, you can unlock the repository with the `ghe-migrator unlock` command. For more information, see "Migrating data to GitHub Enterprise Server."

If the import failed, not all of your data has been migrated, and we recommend deleting the repository and retrying the migration, to prevent data loss.

If you're sure you want to use the repository, a site administrator can unlock the repository using the site admin dashboard. For more information, see "Locking a repository" in the GitHub Enterprise Server documentation.

The source repository is not locked by default, only if the `--lock` argument is specified when preparing the repository for export with the `ghe-migrator add` command. To unlock the repository, use the `ghe-migrator unlock` command. For more information, see "Migrating data to GitHub Enterprise Server."

Repositories locked by Enterprise Cloud Importer 🔗

When you use Enterprise Cloud Importer, the destination repository is locked by default and is not automatically unlocked.

If the import succeeded, you can unlock the repository by clicking the **Unlock** button in Enterprise Cloud Importer.

If the import failed, you cannot unlock the repository yourself. Because a failed migration means that not all of your data has been migrated, we recommend deleting the repository and retrying the migration, to prevent data loss.

If you're sure you want to unlock the repository, contact us through the GitHub Support portal.

Repositories locked by the `startImport` GraphQL mutation 🔗

When you use the `startImport` GraphQL mutation, the destination repository is locked by default and is not automatically unlocked.

If the import succeeded, you can unlock the repository with the `unlockImportedRepositories` GraphQL mutation. For documentation, contact your Expert Services or GitHub Partner representative.

If the import failed, you cannot unlock the repository yourself. Because a failed migration means that not all of your data has been migrated, we recommend deleting the repository and retrying the migration, to prevent data loss.

If you're sure you want to unlock the repository, contact us through the [GitHub Support portal](#).

Legal

[Terms](#)   [Privacy](#)   [Status](#)   [Pricing](#)   [Expert services](#)   [Blog](#)