

Accessing your migration logs for GitHub Enterprise Importer

In this article

About migration logs

Viewing a repository migration log on GitHub

Downloading a repository migration log with the GitHub CLI

Accessing an organization migration log

To review a record of events and errors that occurred during a repository migration, you can access the migration log on GitHub or using the GitHub CLI.

About migration logs

Each time you run a migration with GitHub Enterprise Importer, a migration log is created. The migration log lists the steps that were completed as part of the migration and includes additional information.

- Who ran the migration
- The source of the migration
- How long the migration took
- Any resources that didn't migrate as expected

You can access the migration log for a repository migration in multiple ways.


- On GitHub, by viewing the "Migration Log" issue in the migrated repository. You can use this issue to discuss any warnings with your team and record any decisions.
- By downloading a log file using the GitHub CLI.

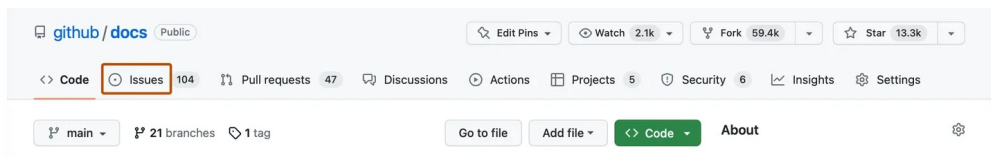
When you run an organization migration, GitHub Enterprise Importer additionally creates a repository named `gei-migration-results` in the destination organization. This repository contains information about the migration of organization-level data and duplicates the information in the "Migration Log" issues for each migrated repository.

For more information about interpreting error messages in your migration log, see "[Troubleshooting your migration with GitHub Enterprise Importer](#)."

Viewing a repository migration log on GitHub

People with read access to a repository can access the migration log for the repository on GitHub.

- 1 Navigate to the migrated repository in your destination organization.
- 2 Under your repository name, click  **Issues**.



- Click the issue with the title "Migration Log."

Downloading a repository migration log with the GitHub CLI [↗](#)

Organization owners and people with the migrator role can download migration logs using the GitHub CLI. For more information about the migrator role, see "[Granting the migrator role for GitHub Enterprise Importer](#)."

You can download the latest migration log for an individual repository with the `download-logs` command. The exact command depends on your migration source.

- [Downloading a repository migration log with the ADO2GH extension](#)
- [Downloading a repository migration log with the BBS2GH extension](#)
- [Downloading a repository migration log with the GEI extension](#)

Migration logs are available to download for 24 hours after the migration is completed.

Downloading a repository migration log with the ADO2GH extension [↗](#)

If your migration source is Azure DevOps, you can download the latest migration log for an individual repository with the `gh ado2gh download-logs` command. Replace `DESTINATION` with the destination organization, `REPOSITORY` with the repository name, and `FILENAME` with a file name for the downloaded file.

```
Shell

gh ado2gh download-logs --github-target-org DESTINATION --target-repo REPOSITORY
--migration-log-file FILENAME
```

If you don't already have a `GH_PAT` environment variable set for a personal access token with access to the destination organization, add `--github-target-pat TOKEN`, replacing `TOKEN` with the personal access token. For personal access token requirements, see "[Managing access for GitHub Enterprise Importer](#)."

Downloading a repository migration log with the BBS2GH extension [↗](#)

If your migration source is Bitbucket Server, you can download the latest migration log for an individual repository with the `gh bbs2gh download-logs` command. Replace `DESTINATION` with the destination organization, `REPOSITORY` with the repository name, and `FILENAME` with a file name for the downloaded file.


```
Shell

gh bbs2gh download-logs --github-target-org DESTINATION --target-repo REPOSITORY
--migration-log-file FILENAME
```

If you don't already have a `GH_PAT` environment variable set for a personal access token with access to the destination organization, add `--github-target-pat TOKEN`, replacing `TOKEN` with the personal access token. For personal access token requirements, see "[Managing access for GitHub Enterprise Importer](#)."

Downloading a repository migration log with the GEI extension

If your migration source is a GitHub product, you can download the latest migration log for an individual repository with the `gh gei download-logs` command. Replace `DESTINATION` with the destination organization, `REPOSITORY` with the repository name, and `FILENAME` with a file name for the downloaded file.

```
Shell 

gh gei download-logs --github-target-org DESTINATION --target-repo REPOSITORY --
migration-log-file FILENAME
```

If you don't already have a `GH_PAT` environment variable set for a personal access token with access to the destination organization, add `--github-target-pat TOKEN`, replacing `TOKEN` with the personal access token. For personal access token requirements, see "[Managing access for GitHub Enterprise Importer](#)."

Downloading all the repository migration logs for a migration script

To ensure you have access to migration logs for all your migrated repositories, you can use the `--download-migration-logs` flag when generating a migration script for repository migrations. When you use this flag, the script will include the `download-logs` command for each repository migrated in the script. For more information, see "[Migrating repositories with GitHub Enterprise Importer](#)."

Note: You can only use the `--download-migration-logs` flag with repository migrations, not with organization migrations.

Accessing an organization migration log

Owners of the destination organization can access the migration log for an organization migration on GitHub.

To access the migration log for an organization migration, navigate to the repository named `gei-migration-results` in your destination organization.

The `README.md` file in the root of the repository includes the following information about the organization migration:

- Any warnings or errors related to the migration of organization-level data, such as settings and teams
- The number of repositories that were successfully migrated and the number of repositories that failed to migrate

The `/success` and `/failure` directories contain one file for each repository that was successfully migrated or that failed to migrate, respectively. These files follow the naming convention `REPO_NAME.md`.

Note: The `gei-migration-results` repository is created at the beginning of the migration process but is only updated with your migration logs after the migration finishes.

Legal