About authentication to GitHub

In this article

You can securely access your account's resources by authenticating to GitHub Enterprise Server, using different credentials depending on where you authenticate.

About authentication to GitHub

To keep your account secure, you must authenticate before you can access certain resources on GitHub Enterprise Server. When you authenticate to GitHub Enterprise Server, you supply or confirm credentials that are unique to you to prove that you are exactly who you declare to be.

You can access your resources in GitHub Enterprise Server in a variety of ways: in the browser, via GitHub Desktop or another desktop application, with the API, or via the command line. Each way of accessing GitHub Enterprise Server supports different modes of authentication.

- Your identity provider (IdP)
- Username and password with two-factor authentication
- Personal access token
- SSH key

Authenticating in your browser

You can authenticate to GitHub Enterprise Server in your browser in a number of ways.

- **Username and password only**
  - You'll create a password when you create your account on GitHub Enterprise Server. We recommend that you use a password manager to generate a random and unique password. For more information, see "Creating a strong password."
- **Two-factor authentication (2FA)** (recommended)
  - If you enable 2FA, after you successfully enter your username and password, we'll also prompt you to provide a code that's generated by a time-based one time password (TOTP) application on your mobile device. For more information, see "Accessing GitHub using two-factor authentication."
  - In addition to authentication with a TOTP application, you can optionally add an

alternative method of authentication with a security key using WebAuthn. For more information, see "Configuring two-factor authentication."

- **External authentication**
  - Your site administrator may configure your GitHub Enterprise Server instance to use external authentication instead of a username and password. For more information, see "About authentication for your enterprise."

Authenticating with GitHub Desktop

You can authenticate with GitHub Desktop using your browser. For more information, see "Authenticating to GitHub in GitHub Desktop."

Authenticating with the API

You can authenticate with the API in different ways. For more information, see "Authenticating to the REST API."

Authenticating to the API with a personal access token

If you want to use the GitHub REST API for personal use, you can create a personal access token. If possible, GitHub recommends that you use a fine-grained personal access token instead of a personal access token (classic). For more information about creating a personal access token, see "Managing your personal access tokens."

Authenticating to the API with an app

If you want to use the API on behalf of an organization or another user, GitHub recommends that you use a GitHub App. For more information, see "About authentication with a GitHub App."

You can also create an OAuth token with an OAuth app to access the REST API. However, GitHub recommends that you use a GitHub App instead. GitHub Apps allow more control over the access and permission that the app has.

Authenticating to the API in a GitHub Actions workflow

If you want to use the API in a GitHub Actions workflow, GitHub recommends that you authenticate with the built-in `GITHUB_TOKEN` instead of creating a token. You can grant permissions to the `GITHUB_TOKEN` with the `permissions` key.

Note that `GITHUB_TOKEN` can only access resources within the repository that contains the workflow. If you need to make changes to resources outside of the workflow repository, you will need to use a personal access token or GitHub App.

For more information, see "Automatic token authentication."

Authenticating with the command line

You can access repositories on GitHub Enterprise Server from the command line in two ways, HTTPS and SSH, and both have a different way of authenticating. The method of authenticating is determined based on whether you choose an HTTPS or SSH remote URL when you clone the repository. For more information about which way to access, see "About remote repositories."

HTTPS

You can work with all repositories on GitHub Enterprise Server over HTTPS, even if you are behind a firewall or proxy.

If you authenticate with GitHub CLI, you can either authenticate with a personal access token or via the web browser. For more information about authenticating with GitHub CLI, see `gh auth login`.

If you authenticate without GitHub CLI, you must authenticate with a personal access token. When Git prompts you for your password, enter your personal access token. Alternatively, you can use a credential helper like Git Credential Manager. Password-based authentication for Git has been removed in favor of more secure authentication methods. For more information, see "Managing your personal access tokens." Every time you use Git to authenticate with GitHub Enterprise Server, you'll be prompted to enter your credentials to authenticate with GitHub Enterprise Server, unless you cache them with a credential helper.

SSH

You can work with all repositories on GitHub Enterprise Server over SSH, although firewalls and proxies might refuse to allow SSH connections.

If you authenticate with GitHub CLI, the CLI will find SSH public keys on your machine and will prompt you to select one for upload. If GitHub CLI does not find a SSH public key for upload, it can generate a new SSH public/private keypair and upload the public key to your account on your GitHub Enterprise Server instance. Then, you can either authenticate with a personal access token or via the web browser. For more information about authenticating with GitHub CLI, see `gh auth login`.

If you authenticate without GitHub CLI, you will need to generate an SSH public/private keypair on your local machine and add the public key to your account on your GitHub Enterprise Server instance. For more information, see "Generating a new SSH key and adding it to the ssh-agent." Every time you use Git to authenticate with GitHub Enterprise Server, you'll be prompted to enter your SSH key passphrase, unless you've stored the key.

GitHub's token formats

GitHub issues tokens that begin with a prefix to indicate the token's type.

Token type	Prefix	More information
Personal access token (classic)	`ghp_`	"Managing your personal access tokens"
Fine-grained personal access token	`github_pat_`	"Managing your personal access tokens"
OAuth access token	`gho_`	"Authorizing OAuth apps"
User access token for a GitHub App	`ghu_`	"Authenticating with a GitHub App on behalf of a user"
Installation access token for a GitHub App	`ghs_`	"Authenticating as a GitHub App installation"
Refresh token for a GitHub App	`ghr_`	"Refreshing user access tokens"