

# Setting your commit email address

## In this article

About commit email addresses

Setting your commit email address on GitHub

Setting your commit email address in Git

You can set the email address that is used to author commits on GitHub.com and on your computer.

Mac Windows Linux

## About commit email addresses [↗](#)

GitHub uses your commit email address to associate commits with your account on GitHub.com. You can choose the email address that will be associated with the commits you push from the command line as well as web-based Git operations you make.

For web-based Git operations, you can set your commit email address on GitHub.com. For commits you push from the command line, you can set your commit email address in Git.

Any commits you made prior to changing your commit email address are still associated with your previous email address.

**Note:** You cannot verify email addresses from disposable email address services (services that allow you to receive email at a temporary address that expires after a certain time). If you'd like to keep your email address private, you can use a GitHub-provided `noreply` email address. For more information, see "[Setting your commit email address](#)."

If you'd like to keep your personal email address private, you can use a `noreply` email address from GitHub Enterprise Cloud as your commit email address. To use your `noreply` email address for commits you push from the command line, use that email address when you set your commit email address in Git. To use your `noreply` address for web-based Git operations, set your commit email address on GitHub and choose to **Keep my email address private**.

You can also choose to block commits you push from the command line that expose your personal email address. For more information, see "[Blocking command line pushes that expose your personal email address](#)."

To ensure that commits are attributed to you and appear in your contributions graph, use an email address that is connected to your account on GitHub.com, or the `noreply` email address provided to you in your email settings. For more information, see "[Adding an email address to your GitHub account](#)."

**Note:** If you created your account on GitHub.com *after* July 18, 2017, your `noreply` email address for GitHub Enterprise Cloud is an ID number and your username in the form of `ID+USERNAME@users.noreply.github.com`. If you created your account on GitHub.com *prior to* July 18, 2017, and enabled **Keep my email address private** prior to that date, your `noreply` email

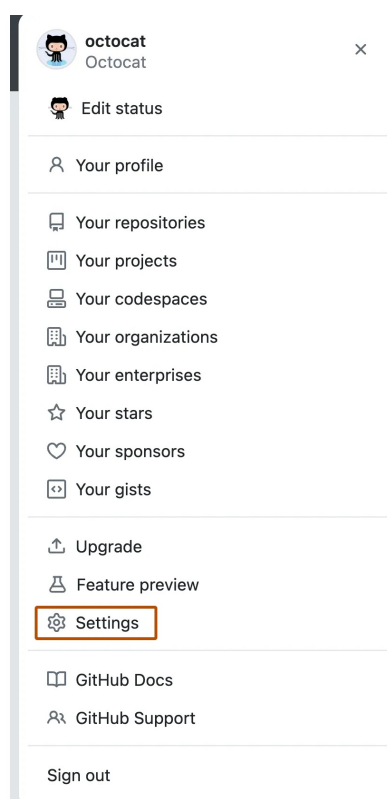
address from GitHub Enterprise Cloud is [USERNAME@users.noreply.github.com](mailto:USERNAME@users.noreply.github.com). You can get an ID-based `noreply` email address for GitHub Enterprise Cloud by selecting (or deselecting and reselecting) **Keep my email address private** in your email settings.


If you use your `noreply` email address for GitHub Enterprise Cloud to make commits and then [change your username](#), those commits will not be associated with your account on GitHub.com. This does not apply if you're using the ID-based `noreply` address from GitHub Enterprise Cloud. For more information, see "[Changing your GitHub username](#)."

## Setting your commit email address on GitHub [↗](#)

If you haven't enabled email address privacy, you can choose which verified email address to author changes with when you edit, delete, or create files or merge a pull request on GitHub Enterprise Cloud. If you enabled email address privacy, then the commit author email address cannot be changed and will be a no-reply by default. For more information about the exact form the no-reply email address can take, see "[Setting your commit email address](#)."

- 1 In the upper-right corner of any page, click your profile photo, then click **Settings**.



- 2 In the "Access" section of the sidebar, click  **Emails**.
- 3 In "Add email address", type your email address and click **Add**.
- 4 [Verify your email address](#).
- 5 In the "Primary email address" dropdown menu, select the email address you'd like to associate with your web-based Git operations.

#### Add email address

#### Primary email address

Because you have email privacy enabled, octocat@github.com will be used for account-related notifications as well as password resets. 101839405+octocat@users.noreply.github.com will be used for web-based Git operations, e.g., edits and merges.

- 6 To keep your email address private when performing web-based Git operations, select **Keep my email addresses private**.

## Setting your commit email address in Git [🔗](#)

You can use the `git config` command to change the email address you associate with your Git commits. The new email address you set will be visible in any future commits you push to GitHub.com from the command line. Any commits you made prior to changing your commit email address are still associated with your previous email address.

## Setting your email address for every repository on your computer [🔗](#)

- 1 Open TerminalTerminalGit Bash.
- 2 Set an email address in Git. You can use your [GitHub Enterprise Cloud-provided noreply\\_email address](#) or any email address.

```
git config --global user.email "YOUR_EMAIL"
```

- 3 Confirm that you have set the email address correctly in Git:

```
$ git config --global user.email  
email@example.com
```

- 4

Add the email address to your account on GitHub Enterprise Cloud, so that your commits are attributed to you and appear in your contributions graph. For more information, see "[Adding an email address to your GitHub account](#)."

## Setting your email address for a single repository [🔗](#)

GitHub Enterprise Cloud uses the email address set in your local Git configuration to associate commits pushed from the command line with your account on GitHub.com.

You can change the email address associated with commits you make in a single repository. This will override your global Git configuration settings in this one repository, but will not affect any other repositories.

- 1 Open TerminalTerminalGit Bash.
- 2 Change the current working directory to the local repository where you want to

configure the email address that you associate with your Git commits.

- 3 Set an email address in Git. You can use your [GitHub Enterprise Cloud-provided noreply\\_email address](#) or any email address.

```
git config user.email "YOUR_EMAIL"
```

- 4 Confirm that you have set the email address correctly in Git:

```
$ git config user.email  
email@example.com
```

- 5

Add the email address to your account on GitHub Enterprise Cloud, so that your commits are attributed to you and appear in your contributions graph. For more information, see "[Adding an email address to your GitHub account](#)."

## Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)