

# Changing a remote's URL

---

The `git remote set-url` command changes an existing remote repository URL.

Mac

**Windows**

Linux

## In this article

---

[Switching remote URLs from SSH to HTTPS](#)

[Switching remote URLs from HTTPS to SSH](#)

[Troubleshooting](#)

[Further reading](#)

**Tip:** For information on the difference between HTTPS and SSH URLs, see "[Which remote URL should I use?](#)"

The `git remote set-url` command takes two arguments:

- An existing remote name. For example, `origin` or `upstream` are two common choices.
- A new URL for the remote. For example:

- If you're updating to use HTTPS, your URL might look like:

```
https://github.com/USERNAME/REPOSITORY.git
```

- If you're updating to use SSH, your URL might look like:

```
git@github.com:USERNAME/REPOSITORY.git
```

## Switching remote URLs from SSH to HTTPS

Open Git Bash.

Change the current working directory to your local project.

List your existing remotes in order to get the name of the remote you want to change.

```
$ git remote -v
> origin  git@github.com:USERNAME/REPOSITORY.git (fetch)
> origin  git@github.com:USERNAME/REPOSITORY.git (push)
```

Change your remote's URL from SSH to HTTPS with the `git remote set-url` command.

```
$ git remote set-url origin https://github.com/USERNAME/REPOSITORY.git
```

Verify that the remote URL has changed.

```
$ git remote -v
# Verify new remote URL
> origin https://github.com/USERNAME/REPOSITORY.git (
> origin https://github.com/USERNAME/REPOSITORY.git (
```

The next time you `git fetch`, `git pull`, or `git push` to the remote repository, you'll be asked for your GitHub username and password. When Git prompts you for your password, enter your personal access token (PAT) instead. Password-based authentication for Git is deprecated, and using a PAT is more secure. For more information, see "[Creating a personal access token](#)."

You can [use a credential helper](#) so Git will remember your GitHub username and personal access token every time it talks to GitHub.

## Switching remote URLs from HTTPS to SSH

Open Git Bash.

Change the current working directory to your local project.

List your existing remotes in order to get the name of the remote you want to change.

```
$ git remote -v
> origin https://github.com/USERNAME/REPOSITORY.git (
```

```
> origin https://github.com/USERNAME/REPOSITORY.git (
```

Change your remote's URL from HTTPS to SSH with the `git remote set-url` command.

```
$ git remote set-url origin git@github.com:USERNAME/RE
```

Verify that the remote URL has changed.

```
$ git remote -v
# Verify new remote URL
> origin git@github.com:USERNAME/REPOSITORY.git (fetc
> origin git@github.com:USERNAME/REPOSITORY.git (push
```

## Troubleshooting

You may encounter these errors when trying to change a remote.

### No such remote '[name]'

This error means that the remote you tried to change doesn't exist:

```
$ git remote set-url sofake https://github.com/octocat/Spoon-
```

```
> fatal: No such remote 'sofake'
```

Check that you've correctly typed the remote name.