

Guide to Git - Host Git Repositories on a cPanel Account

Guide to Git

[Git Terms](#)

[Git Commands](#)

[Deployment](#)

[Set Up
Deployment](#)

[Deployment Cron
Jobs](#)

[Host Repositories on the
Command Line](#)

[Access Private
Repositories](#)

[For System
Administrators](#)

In This Document

Create or clone a repository.
Update the repository's Git configuration.
Clone the repository locally.
Push local changes to the hosted repository.

Overview

Note:

In cPanel & WHM version 72 and later, you can easily perform these tasks in cPanel's [Git Version Control](#) interface (*cPanel >> Home >> Files >> Git Version Control*). While many Git tasks require command-line access, this interface automates some parts of the process and allows you to view historical information for your repositories in [Gitweb](#).

With the appropriate permissions, cPanel accounts can host Git repositories. Git's version control software tracks changes in a system of files that multiple users can manage simultaneously. This tutorial uses the command line to create or clone a new Git repository, update the repository's configuration, and clone the repository locally for updates.

- For a list of common Git commands and their options, read our [Guide to Git - Common Git Commands](#) documentation.
- For more information about Git, read [Git's documentation](#).

Important:

- To perform the steps in this tutorial, you **must** ensure that the following statements are true:
 - You must possess an active cPanel account with available disk space.
 - Your system administrator must [enable the Shell Access setting](#) for your cPanel account.
 - You must register your public key in cPanel's [SSH Access](#) interface (*cPanel >> Home >> Security >> SSH Access*) and authorize it for SSH access.
- Throughout this tutorial, `example` represents the repository name and `Project` represents the directory that contains the repository.
- You can use cPanel's [Terminal](#) interface (*cPanel >> Home >> Advanced >> Terminal*) to access the command line from within the cPanel interface.

Host a Git repository on your cPanel account

1

Create or clone a repository.

You can create a Git repository for any existing directory, or you can create a new, empty directory for your repository. If a Git repository already exists for your project, you can clone it to your cPanel account instead.

Note:

- cPanel's *Git Version Control* interface (*cPanel >> Home >> Files >> Git Version Control*) enforces several restrictions on repository paths:
 - You **cannot** include whitespace or the following characters in repository paths:

```
\ * | " ' < > & @ ` $ { } [ ] ( ) ; ? : = % #
```

- You **cannot** use this feature with repositories in the following cPanel-controlled directories:

▼ [Click to view...](#)

- .cpanel
- .trash
- etc
- mail
- ssl
- tmp
- logs
- .cphorde
- spamassassin
- .htpasswd
- var
- cgi-bin
- .ssh
- perl5
- access-logs

If you use the command line to create a repository in a restricted path, the interface will **not** display the repository.

- To manage an existing repository in cPanel's *Git Version Control* interface (*cPanel >> Home >> Files >> Git Version Control*), create a repository in that repository's directory. The system will automatically update the repository's configuration and add it to the list of cPanel-managed repositories.

Create a new repository Clone an existing repository

To create a new repository, perform the following steps:

- Use SSH to log in to your cPanel account on the command line.
- To navigate to the directory that will contain your repository, run the following command:

```
cd ~/Project/example
```

Note:

To create a new directory to store your repository, run the following command and then navigate to that directory:

```
mkdir -p ~/Project/example
```

- To initialize the directory as a Git repository, run the following command:

```
git init
```

Important:

To clone a private repository, you **must** perform additional steps. For more information, read our [Guide to Git - Set Up Access to Private Repositories](#) documentation.

To clone an existing repository, perform the following steps:

1. Use SSH to log in to your cPanel account on the command line.
2. To navigate to the directory that will contain your repository, run the following command:

```
cd ~/Project
```

Note:

To create a new directory to store your repository, run the following command and then navigate to that directory:

```
mkdir -p ~/Project
```

3. To clone the repository, run the following command :

```
git clone https://domain.com/Account/example.git example.git
```

In this example, `https://domain.com/Account/example.git` represents the repository's clone URL.

Notes:

- Many developers host their code repositories on [GitHub](#). GitHub repository URLs generally resemble the following example, where `Account` represents the GitHub account name and `example` represents the repository name:

```
https://github.com/Account/example.git
```

- The system may require a large amount of time to clone larger repositories. Until this process finishes, HEAD information will be unavailable in cPanel's *Git Version Control* interface (*cPanel >> Home >> Files >> Git Version Control*).
- cPanel's *Git Version Control* interface (*cPanel >> Home >> Files >> Git Version Control*) does **not** allow username-and-password pairs in remote repository URLs.

2

Update the repository's Git configuration.

This optional step configures the Git repository to remain up to date as you push changes from the local branch. For more information about Git's configuration file and its options, read [Git's git-config documentation](#).

To update the configuration, run the following command from within the repository directory:

```
git config receive.denyCurrentBranch updateInstead
```

3

Clone the repository locally.

Important:

- This feature enforces several restrictions on clone URLs. For more information, read our [Git Version Control](#) documentation.
- You can use cPanel's [SSH Access](#) interface (*cPanel >> Home >> Security >> SSH Access*) to add and manage SSH keys, which you can use to access the cPanel-hosted repository. Because SSH keys allow access to the entire cPanel account, and not just a single repository, exercise caution when you perform this action.

To clone the cPanel-account-hosted repository, access your local computer via the command line and run the following command:

```
git clone ssh://username@hostname/home/username/Project/example.git
```

- `username` represents the cPanel account username.
- `hostname` represents the hostname for the server that hosts your cPanel account.

4

Push local changes to the hosted repository.

After you finish this tutorial, you can make changes to the repository's files on your local computer. You **must** run the following command in order to push changes that you make on your local computer to the hosted repository:

```
git push origin master -u
```

This command pushes your revisions to the copy of the repository that exists on your cPanel account.

Note:

cPanel & WHM version 72 updated cPanel's Git installation to automatically configure some settings. When you push changes to cPanel & WHM version 70 and earlier, run the following command instead:

```
git push origin master -u  
--exec=/usr/local/cpanel/3rdparty/bin/git-receive-pack
```

Additional documentation

Suggested documentation [For cPanel users](#) [For WHM users](#) [For developers](#)

- [Guide to Git - Deployment](#)
- [Guide to Git](#)
- [Guide to Git - Common Git Commands](#)
- [Guide to Git - Host Git Repositories on a cPanel Account](#)
- [Guide to Git - Git Terms](#)

- [Guide to Git - Deployment](#)
- [Git Version Control](#)
- [Gitweb](#)
- [Gitweb](#)
- [Git Version Control](#)

- [Guide to Git](#)
- [Guide to Git - For System Administrators](#)

- [UAPI Functions - VersionControl::create](#)
- [UAPI Modules - VersionControlDeployment](#)
- [UAPI Functions - VersionControlDeployment::create](#)
- [UAPI Functions - VersionControlDeployment::retrieve](#)
- [UAPI Functions - VersionControlDeployment::delete](#)