

NAME

git-credential-store - Helper to store credentials on disk

SYNOPSIS

```
git config credential.helper store [options]
```

DESCRIPTION**Note**

Using this helper will store your passwords unencrypted on disk, protected only by filesystem permissions. If this is not an acceptable security tradeoff, try [git-credential-cache\(1\)](#), or find a helper that integrates with secure storage provided by your operating system.

This command stores credentials indefinitely on disk for use by future Git programs.

You probably don't want to invoke this command directly; it is meant to be used as a credential helper by other parts of git. See [gitcredentials\(7\)](#) or EXAMPLES below.

OPTIONS

--file=<path>

Use <path> to store credentials. The file will have its filesystem permissions set to prevent other users on the system from reading it, but will not be encrypted or otherwise protected. Defaults to `~/.git-credentials`.

EXAMPLES

The point of this helper is to reduce the number of times you must type your username or password. For example:

```
$ git config credential.helper store
```

```
$ git push http://example.com/repo.git
```

```
Username: <type your username>
```

```
Password: <type your password>
```

```
[several days later]
```

```
$ git push http://example.com/repo.git
```

```
[your credentials are used automatically]
```

STORAGE FORMAT

The `.git-credentials` file is stored in plaintext. Each credential is stored on its own line as a URL like:

```
https://user:pass@example.com
```

When Git needs authentication for a particular URL context, credential-store will consider that context a pattern to match against each entry in the credentials file. If the protocol, hostname, and username (if we already have one) match, then the password is returned to Git. See the discussion of configuration in [gitcredentials\(7\)](#) for more information.

GIT

Part of the [git\(1\)](#) suite