
pypass

Release

November 08, 2014

1	Manpage	1
1.1	Synopsis	1
1.2	Description	1
1.3	Commands	2
1.4	Simple Examples	2
1.5	Extended Git Example	2
1.6	Files	2
1.7	Environement Variables	2
1.8	See Also	2
1.9	Author	2
1.10	Copying	2
1.11	See Also	2

1.1 Synopsis

`pypass [COMMAND] [OPTIONS] [ARGS]`

1.2 Description

`pass` is a very simple password store that keeps passwords inside `gpg2(1)` encrypted files inside a simple directory tree residing at `~/.password-store`. The `pass` utility provides a series of commands for manipulating the password store, allowing the user to add, remove, edit, synchronize, generate, and manipulate passwords.

If no `COMMAND` is specified, `COMMAND` defaults to either `show` or `ls`, depending on the type of specifier in `ARGS`. Otherwise `COMMAND` must be one of the valid commands listed below.

Several of the commands below rely on or provide additional functionality if the password store directory is also a git repository. If the password store directory is a git repository, all password store modification commands will cause a corresponding git commit. See the `EXTENDED GIT EXAMPLE` section for a detailed description using `init` and `git(1)`.

The `init` command must be run before other commands in order to initialize the password store with the correct gpg key id. Passwords are encrypting using the gpg key set with `init`.

1.3 Commands

1.4 Simple Examples

1.5 Extended Git Example

1.6 Files

1.7 Environement Variables

1.8 See Also

1.9 Author

1.10 Copying

1.11 See Also

gpg2(1), pwgen(1), git(1), xclip(1).