pass Unix password manager

CONCEPTS

Each password is one GPG-encrypted file under ~/.password-store; the folder tree is your organisation.

Plain text typed or generated
↓ `pass insert`
GPG file name.gpg in the store
↓ `pass git push`
Git remote optional backup

The first line of a file is the password; put extra data (user, URL) on later lines.

Back up your GPG secret key: without it the store cannot be decrypted.

SETUP

`pass init <gpg-id>`
Create the store for a GPG key

`pass git init`
Turn the store into a git repo

`pass git remote add origin <url>`
Add a remote to the store

`PASSWORD_STORE_DIR=<dir> pass`
Use another store location

BROWSE

`pass`
List all entries as a tree

`pass ls <folder>`
List one folder

`pass find <name>`
List entries whose name matches

`pass grep <text>`
Search inside decrypted entries

READ

`pass show <name>`
Print an entry

`pass <name>`
Print an entry (shortcut)

`pass -c <name>`
Copy the first line to the clipboard

`pass -c<N> <name>`
Copy line N instead

READ (cont.)

`pass -q <name>`
Show as a QR code

ADD AND EDIT

`pass insert <name>`
Add an entry, typed twice hidden

`pass insert -m <name>`
Add a multi-line entry

`pass edit <name>`
Edit an entry in $EDITOR

`pass generate <name> <length>`
Generate a random password

`pass generate -n <name> <length>`
Generate without symbols

`pass generate -c <name> <length>`
Generate and copy, do not print

`pass generate -i <name> <length>`
Replace only the first line

ORGANISE

`pass mv <old> <new>`
Rename or move an entry

`pass cp <old> <new>`
Copy an entry

`pass rm <name>`
Delete an entry

`pass rm -r <folder>`
Delete a folder

`pass mv <name> <folder>/`
Move into a folder

GIT SYNC

`pass git log`
Show store history

`pass git push`
Push the store

`pass git pull`
Pull changes from the remote

`pass git clone <url> ~/.password-store`
Restore on a new machine

TEAMS AND SECURITY

`pass init <id1> <id2>`
Encrypt for several recipients