

Pass-Cli and GPG

Setting up Pass & GPG

- [🔗 Setting up pass on git with a gpg key](#)
- [🔗 vim-gnupg + Neovim + MacOS + pinentry](#)
- [🔗 Set up a GPG key for signing Git commits on MacOS](#)

PassForlos (pass-client for iOS)

- [🔗 mssun/passforios](#)

GPG Key Export / Import

```
# identify the key Bash
gpg --list-secret-keys grieser.chris@gmail.com

# example output
pub 4096R/ABC12345 2020-01-01 [expires: 2025-12-31]
uid Your Name <user@example.com>
sub 4096R/DEF67890 2020-01-01 [expires: 2025-12-31]
```

Remember the ID of your key (second column, after the slash, e.g. "ABC12345"). If you have a "sub" entry, you can ignore it.

```
# export key Bash
gpg --export-secret-keys YOUR_ID_HERE > private.key
```

Copy the key file to the other machine using a secure transport.

```
# import Bash
gpg --import private.key
```

Source: [🔗 GPG: Extract private key and import on different machine - makandra.org](#)

Troubleshooting GPG

Importing Key

```
# "unsafe permissions on homedir" Bash
find ~/.gnupg -type f -exec chmod 600 {} \;
```

```
find ~/.gnupg -type d -exec chmod 700 {} \;
```

```
# "key already known": delete existing keys first      Bash
gpg --delete-keys
gpg --delete-secret-keys
```

```
# "error sending to agent: No pinentry"                Bash
https://superuser.com/a/1135950
gpg --batch --import private.key
```

Config & Pinentry

```
# GPG config change not recognized:                    Bash
# → restart gpgconfig
gpgconf --kill gpg-agent
```

```
# "gpg: decryption failed: No pinentry"                Bash
```

- path for pinentry-program in `gpg-agent.conf` incorrect?
- program not installed?

```
# create symlink to the location required there      Bash
if [[ $(uname -p) == "i386" ]]; then
    mkdir -p /opt/homebrew/bin/
    ln -sf /usr/local/bin/pinentry-mac
/opt/homebrew/bin/pinentry-mac
fi
```

Trust Level

```
# There is no assurance this key belongs to the named user  Bash
# https://stackoverflow.com/a/34132924

# get key ID via
gpg --list-secret-keys grieser.chris@gmail.com

gpg --edit-key <KEY_ID>
gpg> trust

# Your decision? 5
# Do you really want to set this key to ultimate trust? (y/N) y

gpg> quit
```

