INFORMATION SECURITY

How do I delete secret subkeys correctly?

Ask Question

1

I tried to use `gpg --delete-secret-keys` to delete some revoked subkeys but ended up accidentally deleting my primary key instead.

I was able to reproduce my mistake with the following commands:

```
$ gpg --batch --passphrase '' --quick-generate-key 'test key' rsa4096 cert 0
gpg: key 0xA52099E0E7EB77A5 marked as ultimately trusted
gpg: revocation certificate stored as '~/.gnupg/openpgp-
revocs.d/D7D79C32883EA862C586881DA52099E0E7EB77A5.rev'
$ gpg --batch --passphrase '' --quick-add-key
D7D79C32883EA862C586881DA52099E0E7EB77A5 rsa4096 sign 0
$ gpg --list-keys
pub   rsa4096/0xA52099E0E7EB77A5 2019-04-10 [C]
      Key fingerprint = D7D7 9C32 883E A862 C586  881D A520 99E0 E7EB 77A5
uid                  [ultimate] test key
sub   rsa4096/0x20AA2F4F7A28CD01 2019-04-10 [S]
      Key fingerprint = 9CAE 802D A78E 4624 BD8F  88FE 20AA 2F4F 7A28 CD01
$ gpg --delete-secret-keys 9CAE802DA78E4624BD8F88FE20AA2F4F7A28CD01

sec  rsa4096/0xA52099E0E7EB77A5 2019-04-10 test key

Delete this key from the keyring? (y/N)
```

Even though I specified the *subkey* by fingerprint, `gpg` asks me to confirm the deletion of the *primary key*.

The manual states:

> `--delete-secret-keys name`
>
> Remove `key` from the secret keyring. In batch mode the `key` must be specified by fingerprint

> caution is done because gpg can’t be sure that the secret key (as controlled by gpg-agent) is only used for the given OpenPGP public key.

I tried using batch mode as well:

```
$ gpg --batch --yes --delete-secret-keys 9CAE802DA78E4624BD8F88FE20AA2F4F7A28CD01
$ gpg --list-secret-keys
# Empty output.
# Primary key has been deleted.
```

I specified the *subkey* by fingerprint but `gpg` interpreted the command as if I had specified my *primary key* instead.

What is the correct way to do this? Did I understand it wrong?

pgp gnupg openpgp

edited Apr 10 at 8:45

asked Apr 10 at 7:57

Matheus Moreira
156 1 12

1 Answer

0

✓

As explained by Peter Lebbing and Daniel Kahn Gillmor on the mailing list, the answer for `gpg 2.2.15` is to ask `gpg-agent` to delete the secret subkey.

```
# Obtain the keygrip of the subkey
$ gpg --with-keygrip --list-secret

# Ask gpg-agent to delete the key
# There should be a graphical conf
$ gpg-connect-agent "delete_key $H
```

`gpg-agent` is the program that actually manages the secret keys. Each secret key corresponds to a file named `"$GNUPGHOME"/private-keys-v1.d/"$KEYGRIP".key` . In order to delete a secret subkey, the user must obtain its keygrip and then ask `gpg-agent` to delete it.

Documentation for the `delete_key` command:

```
# and a loopback pinentry is allo
# the user for confirmation.  If
# only be deleted if it is a refe
OK
```

Apparently, there is no way to tell `gpg` to ask `gpg-agent` to delete a secret subkey on the user's behalf. An issue has been opened about this.

answered Apr 12 at 4:03

Matheus Moreira
156 1 12

1 Note that this doesn't necessarily ensure that the key cannot be forensically recovered. – forest Apr 12 at 6:34