Modern Key Management with GPG

Werner Koch

Kernel-Recipes — Paris — September 28, 2017

Outline

Where we are

Modern algos

The Quick commands

Keyservers and such

Hints on integrating gpg

Wrapping Up

Versions

- GnuPG 2.2 released a few weeks ago.
- 2.1 has been around for nearly 3 years.
- New features
  - Easy key discovery for any mail address.
  - Full separation between private key and gpg
  - Curve25519 support
  - Better CLI support
  - ...
- End of life for 2.0 in 3 months.
- We keep 1.4 for its PGP-2 support and portability to pre-POSIX systems.

What's next

- RFC-4880bis work in 2.3
  - AEAD mode
  - SHA-256 fingerprint
  - New default algos
- "Moving up the stack":
  - Help integrating new features
  - Checking existing use
- Make Gnuk easier available
- Write more than reference manuals.

Outline

Where we are

Modern algos

The Quick commands

Keyservers and such

Hints on integrating gpg

Wrapping Up

Why ECC (1)

- ECC algorithms are very well researched.
- Instead of key sizes we speak of different curves
- For RSA et al. one implementation fits all sizes.
- For ECC each curve needs to be implemented separately.
  - A large class of curves can be implemented using a table of parameters.

Why ECC (2)

- Certain curves have a bad repudiation.
- In particular the NIST curves as required for Suite B.
- European Brainpool curves might be better . . .
  . . . still are too similar to the NIST curves.

So let's move on.
The new de-facto standard (RFC-7748) is:

- Curve25519
- Curve448-Goldilocks
- Variants for use with EdDSA

Why ECC (2)

- Certain curves have a bad repudiation.
- In particular the NIST curves as required for Suite B.
- European Brainpool curves might be better . . .
  . . . still are too similar to the NIST curves.

So let's move on.
The new de-facto standard (RFC-7748) is:

- Curve25519
- Curve448-Goldilocks
- Variants for use with EdDSA

Example rsa4096

```
commit 72339165aeedec035b821c89453236e2c6949bb6
tree 92c63895b041aa198518a25b87f8ebb727dc4743
parent 2b60d1fe650683ab4fa5690fa2f8c41605fb6e0e
author Werner Koch <wk@gnupg.org> 1505892912 +0200
committer Werner Koch <wk@gnupg.org> 1505892912 +0200
gpgsig -----BEGIN PGP SIGNATURE-----

 iQIzBAABCAAdFiEEssy2g4MyXWG6xQ+fzSGoCsjFJWUFAlnCGjAACgkQzSGoCsjF
 JWVm/g//cool4UycftJSh9Fuy9pmXjDxjudheeQ6UaaWYuMlBYZTVsyjdkknM4Iw
 f92HKm1ieJpXc1KS89nd/iJRXSYFl307hfFsBPuohGIgUaIFOoqyb8TOxXQ7INbg
 wTpDvbPMk0yZHNA8feHC1v+R2rRQbsUfQwmNtw9FpcvR0hZ7Lp+5jpLTU6th3zpI
 Dz3Rlo26kJ7aMxtH8xjlnXnevL/GPc4zFpNOWhjJhASeDjpEUid6WguaaWfJkLOo
 U0bM43yk1FXdr0KyoOdM0aqJNT49jlpND1xFtVB3/wivOFngwBgcrzLRHCcJFGS6
 HZJoIF0yQoVjmp9zSCrRwdQL6OybC2rWrlhIeEcy7XFwivtsVkr/H+t+Xty0AnFz
 vXi8deJa0E6L+k5E4CY3WvhDpV/CGWdd+owrr52nUZIIZGTgLv7QosOd3WCD6iya
 CqIBlEtEaVK7kX/2qhg4pn3/EQ6n2y+2fAcNGW6JAQK1Kui+BuheO9zSYhhUj1y1
 F72n0mM4Im7ndM+44Ctc+jTw/NbYDRGRhomGnMYYLLOKJ+RY1VLE+esFTVtfbTtm
 uiFOb427d5UPhNm/NY8hKAVcvbdlt335rQjR4+Wjo7suQAuP0zV182dHwXrCQ3Tk
 3hk60KOoiJj6nKhkOERaFkB/XhnUJGqNXPIrYtuoPwX2eQhQBvA=
 =Gvqf
 -----END PGP SIGNATURE-----
```

Example ed25519

```
commit 2b60d1fe650683ab4fa5690fa2f8c41605fb6e0e
tree 7494139e7560bf6f6a0b9e8ebee74dbbb01b6bcb
parent 4ee52a72377b4279ba81a3a1c2324a66cfd2c619
author Werner Koch <wk@gnupg.org> 1505892819 +0200
committer Werner Koch <wk@gnupg.org> 1505892819 +0200
gpgsig -----BEGIN PGP SIGNATURE-----

 iHUEABYIAB0WIQTB0OtpIZ5K7sC6HCHj/f8hjkW3KwUCWcIZ1AAKCRDj/f8hjkW3
 K6PzAPOT/keoxJGIWRGiXpiKQQbX2utH/cnR+sM/YO7q4bL1LgEAktfdJ2Z1ZxJm
 4K/rozUhx8OrvIuw5YP0QcJAem83dgA=
 =XNb3
 -----END PGP SIGNATURE-----
```

Performance

Zeitcontrol and Gnuk tokens:

(milliseconds measured inside gpg on an X220)

cpu	algo	sign	(verify)
nxp	rsa2048	470	0.1
nxp	rsa4096	2800	0.9
stm32	ed25519	45	6.0

- RSA is 60 times slower than Ed25519 for signing.
- RSA is always fast as lightning for verification.
- Our Ed25519 verification code is a bit slow.

Performance

Zeitcontrol and Gnuk tokens:

(milliseconds measured inside gpg on an X220)

cpu	algo	sign	(verify)
nxp	rsa2048	470	0.1
nxp	rsa4096	2800	0.9
stm32	ed25519	45	6.0

- RSA is 60 times slower than Ed25519 for signing.
- RSA is always fast as lightning for verification.

- Our Ed25519 verification code is a bit slow.

Outline

Where we are

Modern algos

The Quick commands

Keyservers and such

Hints on integrating gpg

Wrapping Up

Gpg and its prompts

- Written as replacement for PGP-2.
- Direct the user into the right direction
- LibGPGME for common tasks
- Hard to automate (requires FSM)

Better API?

- Too many options and uncertainty which are really needed.
- Meanwhile we know the common use patterns . . .

Let's welcome the –quick-foo commands.

Gpg and its prompts

- Written as replacement for PGP-2.
- Direct the user into the right direction
- LibGPGME for common tasks
- Hard to automate (requires FSM)

Better API?

- Too many options and uncertainty which are really needed.
- Meanwhile we know the common use patterns . . .

Let's welcome the –quick-foo commands.

Gpg and its prompts

- Written as replacement for PGP-2.
- Direct the user into the right direction
- LibGPGME for common tasks
- Hard to automate (requires FSM)

Better API?

- Too many options and uncertainty which are really needed.
- Meanwhile we know the common use patterns . . .

Let's welcome the –quick-foo commands.

Key generation

```
$ gpg --quick-generate-key USER_ID [ALGO [USAGE [EXPIRE]]]
```

Try "future-default" for `ALGO`.
If you don't want a passphrase, do this

```
$ gpg --passphrase '' --batch --quick-generate-key USER_ID
```

Changing the expiration date

- The default is to create keys which expire in two years.
- OpenPGP allows to prolong the expiration date.

To set the expiration to 2 years from now:

```
$ gpg --quick-set-expire FINGERPRINT -
```

Adding a subkey

Subkeys are very useful for key management. Adding more subkeys is easy:

```
$ gpg --quick-add-key FINGERPRINT [ALGO [USAGE [EXPIRE]]]
```

Adding/Revoking a user id

Got a new mail address?

```
$ gpg --quick-add-uid FINGERPRINT NEW_USER_ID
```

Lost that address?

```
$ gpg --quick-revoke-uid FINGERPRINT USER_ID
```

Tell others which user id to see:

```
$ gpg --quick-set-primary-uid FINGERPRINT USER_ID
```

Key signing

Key signing party:

```
$ gpg --quick-sign-key FINGERPRINT [NAMES]
```

Mark a key locally as verified:

```
$ gpg --quick-lsign-key FINGERPRINT [NAMES]
```

Encryption w/o a keyring

Instead of importing a key and using its fingerprint, the -f option can be used:

```
$ gpg -f FILE_WITH_KEY -e DATA
```

The new export filters can be used to create a key file.

Outline

Where we are

Modern algos

The Quick commands

Keyservers and such

Hints on integrating gpg

Wrapping Up

Key discovery

- Keyservers can't map an address to a key.
- Only the mail provider can do that.
- Mail addresses are not under the user's authority like their keys are.
- Mail provider provides the key (web key directory).
- Keyservers are decentralized; this is a Good Thing™.

- Verifying keyservers harm the PGP ecosystem.
  - They need to be under a single authority.
  - The return of the X.500 dilemma.

Key discovery

- Keyservers can't map an address to a key.
- Only the mail provider can do that.
- Mail addresses are not under the user's authority like their keys are.
- Mail provider provides the key (web key directory).
- Keyservers are decentralized; this is a Good Thing™.
- Verifying keyservers harm the PGP ecosystem.
  - They need to be under a single authority.
  - The return of the X.500 dilemma.

Key Validation

- The Web-of-Trust is a geek's instrument.
  - Hard to explain.
  - Global social graph.
  - It does not scale.
- The Trust On First Use (TOFU) paradigm is better.
  - Easy to explain. ✓
  - Local. ✓
  - Keeps the PGP properties. ✓

Key Validation

- The Web-of-Trust is a geek's instrument.
  - Hard to explain.
  - Global social graph.
  - It does not scale.
- The Trust On First Use (TOFU) paradigm is better.
  - Easy to explain. ✓
  - Local. ✓
  - Keeps the PGP properties. ✓

Outline

Where we are

Modern algos

The Quick commands

Keyservers and such

Hints on integrating gpg

Wrapping Up

The two interfaces — human

- This is plainly for human comsumption
- Translated.
- Uses the native charset
- Strings may change with each release

Never use it for scripting!

The two interfaces — machine

- This is mainly for scripting
- Fixed strings
- Always UTF-8
- Only compatible changes since 1.0

Enable this interface using

```
--batch --with-colons --status-fd=2
```

When using the interactor (`--command-fd`) leave out `--batch`. "awk -F:" is your friend. See doc/DETAILS for a full description.

Import and export filter

Remove funny signatures. My gpg.conf:

```
import-filter drop-sig=   sig_created_d=2015-12-24
import-filter drop-sig=|| sig_created_d=2016-03-16
```

Show keys in a file

```
$ gpg --import-options show-only --import FILE
```

Export only the userids with a given mail address

```
$ gpg -a --export-options=export-minimal \
  --export-filter keep-uid=mbox=wk@gnupg.org \
  --export FINGERPRINT
```

Ssh-agent

It is more than 10 years old:

```
$ ssh-add
```

transfers existing keys into GnuPG's key store and makes them permanent.

- Works nicely with smartcards
- Use a subkey for ssh
- ssh-add still works

- You can't live without it.

Ssh-agent

It is more than 10 years old:

```
$ ssh-add
```

transfers existing keys into GnuPG's key store and makes them permanent.

- Works nicely with smartcards
- Use a subkey for ssh
- ssh-add still works

- You can't live without it.

Outline

Where we are

Modern algos

The Quick commands

Keyservers and such

Hints on integrating gpg

Wrapping Up

GnuPG 2.2

- Modern algos
- Better scriptability
- Auto key discovery when a mail address is given.
  - We need to talk to providers.
- Take care:
  - Debian has 2.1.18 plus some changes.
  - Ubuntu has a partly broken 2.1.11

Thanks for listening. Questions?

https://gnupg.org/ftp/blurbs/kernel-recipes-2017-modern-key-management.pdf

GnuPG 2.2

- Modern algos
- Better scriptability
- Auto key discovery when a mail address is given.
  - We need to talk to providers.
- Take care:
  - Debian has 2.1.18 plus some changes.
  - Ubuntu has a partly broken 2.1.11

Thanks for listening. Questions?

https://gnupg.org/ftp/blurbs/kernel-recipes-2017-modern-key-management.pdf

GnuPG 2.2

- Modern algos
- Better scriptability
- Auto key discovery when a mail address is given.
  - We need to talk to providers.
- Take care:
  - Debian has 2.1.18 plus some changes.
  - Ubuntu has a partly broken 2.1.11

Thanks for listening. Questions?

https://gnupg.org/ftp/blurbs/kernel-recipes-2017-modern-key-management.pdf