GnuPG modern

Pavel Růžička <ruza@ruza.eu>

Lightning Talks, March 2015

GnuPG versions, required libraries/tools, optional software

Name	Version	Size	Tarball	Signature
GnuPG stable	2.0.26	4203k	download	download
GnuPG modern	2.1.2	4720k	download	download
GnuPG classic	1.4.18	3564k	download	download
Libgpg-error	1.18	702k	download	download
Libgcrypt	1.6.2	2418k	download	download
Libksba	1.3.2	587k	download	download
Libassuan	2.2.0	505k	download	download
Pinentry	0.9.0	453k	download	download
GPGME	1.5.3	946k	download	download
GPA	0.9.7	718k	download	download
Dirmngr	1.1.0	543k	download	download

GnuPG versions and affiliated sw

- **GnuPG stable (2.0)** - modularized version supporting OpenPGP, S/MIME, Secure Shell
- **GnuPG modern (2.1)** - brand new version with enhanced features like support for Elliptic Curve Cryptography. It will eventually replace the current stable (2.0)
- **GnuPG classic (1.4)** - old, single binary (even for ancient Unix platforms). It has no dependencies on listed libraries or Pinnetry and lacks many modern features
- Pinentry - collection of passphrase entry dialogs for GPG 2.x
- GPGME - standard library to access GPG functions from programming languages
- GPA - GUI to GPG
- Dirmngr - optional tool for GPG 2.x

GnuPG 2.1 - Modern

https://www.gnupg.org/faq/whats-new-in-2.1.html

GnuPG 2.1 - Modern

```
export INSTALL_PREFIX=/usr
make -f build-aux/speedo.mk native-gui
```

GnuPG 2.x version check and features

```
ruza@azur:~$ gpg2 --version
gpg (GnuPG) 2.1.2
libgcrypt 1.6.2

Home: ~/.gnupg
Supported algorithms:
Pubkey: RSA, ELG, DSA, ECDH, ECDSA, EDDSA
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
        CAMELLIA128, CAMELLIA192, CAMELLIA256
Hash: SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compression: Uncompressed, ZIP, ZLIB, BZIP2
```

GnuPG migration to Modern

```
...
gpg: starting migration from earlier GnuPG versions
gpg: porting secret keys from '~/.gnupg/secring.gpg' to gpg-agent
gpg: key 2AB83H3A: secret key imported
gpg: migration succeeded
```

~/.gnupg v1.x vs v2.x

```
brmlab@azur:~$ ls -ltr .gnupg/
-rw------- 1 brmlab brmlab 9398 Jan  4 20:01 gpg.conf
-rw------- 1 brmlab brmlab 1360 Jan  4 20:39 trustdb.gpg
-rw------- 1 brmlab brmlab  600 Jan  4 20:44 random_seed
-rw------- 1 brmlab brmlab 7792 Jan  4 20:50 secring.gpg
-rw------- 1 brmlab brmlab 5709 Jan  4 20:50 pubring.gpg
srwx------ 1 brmlab brmlab    0 Feb 19 00:02 S.gpg-agent
drwx------ 2 brmlab brmlab 4096 Feb 19 00:02 private-keys-v1.d

ruza@azur:~$ ls -latr .gnupg
-rw-------   1 ruza ruza       0 Jan  7 05:47 secring.gpg
-rw-------   1 ruza ruza       0 Jan  7 05:47 .gpg-v21-migrated
drwx------   2 ruza ruza    4096 Jan  8 15:05 crls.d/
-rw-------   1 ruza ruza    1984 Jan 10 10:56 sks-keyservers.netCA.pem
-rw-------   1 ruza ruza      54 Jan 26 13:20 dirmngr.conf
-rw-------   1 ruza ruza    8828 Jan 26 14:04 gpg.conf
drwx------   2 ruza ruza    4096 Jan 26 14:08 dirmngr-cache.d/
-rw-------   1 ruza ruza       0 Jan 26 14:10 dirmngr_ldapservers.conf
drwx------   2 ruza ruza    4096 Feb  9 15:11 private-keys-v1.d/
drwx------   2 ruza ruza    4096 Feb  9 15:11 openpgp-revocs.d/
-rw-------   1 ruza ruza 3618898 Feb 10 19:11 pubring.gpg
-rw-------   1 ruza ruza    6080 Feb 10 19:11 trustdb.gpg
-rw-------   1 ruza ruza     280 Feb 17 18:59 gpg-agent.conf
srwx------   1 ruza ruza       0 Feb 17 23:29 S.gpg-agent=
srwx------   1 ruza ruza       0 Feb 18 13:01 S.dirmngr=
srwx------   1 ruza ruza       0 Feb 18 19:56 S.uiserver=
-rw-------   1 ruza ruza       0 Feb 18 19:56 pubring.kbx
-rw-------   1 ruza ruza      87 Feb 18 19:57 gpa.conf
-rw-------   1 ruza ruza     600 Feb 19 00:03 random_seed
```

GnuPG ECC keys

```
brmlab@azur:~$ gpg2 --full-gen-key

Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
Your selection?

brmlab@azur:~$ gpg2 --expert --full-gen-key
Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
   (7) DSA (set your own capabilities)
   (8) RSA (set your own capabilities)
   (9) ECC and ECC
  (10) ECC (sign only)
  (11) ECC (set your own capabilities)
Your selection? 11
```

brmlab hackerspace prague

GnuPG ECC keys

```
Possible actions for a ECDSA key: Sign Certify Authenticate
Current allowed actions: Sign Certify

   (S) Toggle the sign capability
   (A) Toggle the authenticate capability
   (Q) Finished

Your selection? a

Possible actions for a ECDSA key: Sign Certify Authenticate
Current allowed actions: Sign Certify Authenticate

   (S) Toggle the sign capability
   (A) Toggle the authenticate capability
   (Q) Finished

   Your selection? q
```

brmlab hackerspace prague

GnuPG ECC keys

```
Please select which elliptic curve you want:
   (1) Curve 25519
   (2) NIST P-256
   (3) NIST P-384
   (4) NIST P-521
   (5) Brainpool P-256
   (6) Brainpool P-384
   (7) Brainpool P-512
Your selection? 1
```

GnuPG KeyServers

https://sks-keyservers.net/overview-of-pools.php

- pool.sks-keyservers.net - HKP (IPv4, IPv6) random
- eu.pool.sks-keyservers.net - EU
- ipv6.pool.sks-keyservers.net - IPv6
- ipv4.pool.sks-keyservers.net - IPv4
- subset.pool.sks-keyservers.net - support ECC keys (RFC6637)
- ha.pool.sks-keyservers.net - high-availibility
- p80.pool.sks-keyservers.net - port 80
- hkps.pool.sks-keyservers.net - HKP over SSL/TLS

HKPS keyserver

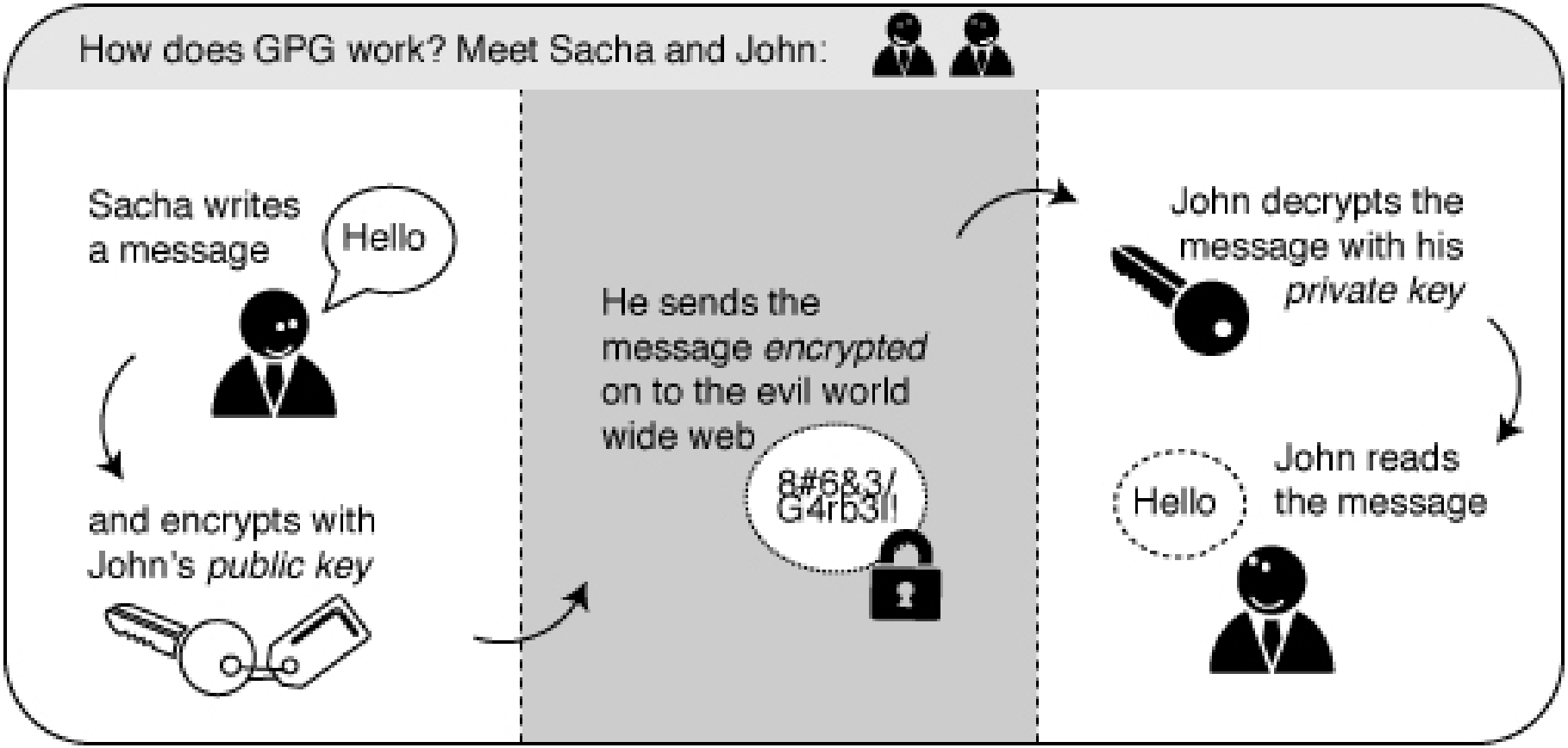
For GnuPG 1.4 and 2.0 installations this can be used by using the following parameters in gpg.conf:

```
~/.gnupg/gpg.conf:
  keyserver hkps://hkps.pool.sks-keyservers.net
  keyserver-options ca-cert-file=/path/sks-keyservers.CA.pem
```

GnuPG 2.1 users want to add the following in dirmngr.conf:

```
~/.gnupg/dirmngr.conf:
   hkp-cacert /path/sks-keyservers.CA.pem
```

Asymetric cipher encryption. How it works



If You insist on deniability, do not sign.

brmlab hackerspace prague

Email source

```
From: Nekdo Nekde <nekdo@nekde.eu>
To: Pavel Ruzicka <ruza@ruza.eu>
Subject: Re: o necem
Content-Type: multipart/encrypted; boundary="opT7a7wXMBdUmrNcaP1d7nBMu5XOArcnV"; micalg=
X-GPGit: applied

This is an OpenPGP/MIME signed message (RFC 4880 and 3156)
--opT7a7wXMBdUmrNcaP1d7nBMu5XOArcnV
Content-Type: application/pgp-encrypted; name="msg.asc"
Content-Disposition: inline; filename="msg.asc"
Content-Transfer-Encoding: 7bit

Version: 1
--opT7a7wXMBdUmrNcaP1d7nBMu5XOArcnV
Content-Type: application/octet-stream
Content-Disposition: inline
Content-Transfer-Encoding: 7bit

-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.10 (GNU/Linux)

hQIMA+CODVguDvvdARAA1AhQyS021YSUa3muU+zD7LF4EKfOY5ZVGMa4x5DkeCeZ
OousSalNl9+XRbgZfhNQJUbteGgF+0KI+WI3eGBwuB6dQoNuRcObkRIw3JFNMb1k
```

Sent mail (outbox problem)

```
ruza@azur:~$ gpg -d /tmp/sent-email.pgp
gpg: encrypted with 3072-bit RSA key, ID 0xA3947D27E9442, created 2014-12-18
      "Na Piste Mi <napiste@gmail.com>"
gpg: encrypted with 4096-bit ELG key, ID 0xF7EEAF5998070C1D, created 2015-01-01
      "Pavel Ruzicka (ruza.eu) <ruza@ruza.eu>"
Content-Type: multipart/mixed; boundary="H2uIiESbpAEk7ic6SodKtWdQOu1jAMDuX"

--H2uIiESbpAEk7ic6SodKtWdQOu1jAMDuX
Content-Type: text/plain; charset=iso-8859-2
Content-Transfer-Encoding: quoted-printable

Diky za pozvanku.....
-- 8< -- SNIP -- >8 --
```

S/MIME aka "those WWW certificates"

S/MIME builds over MIME and CMS (better for top-down hierarchy, i.e. enterprise)

MIME (Multi-Purpose Internet Mail Extensions) is a standard way of putting arbitrary data into emails, with a "type" (an explicit indication of what the data is supposed to mean) and gazillions of encoding rules and other interoperability details.
CMS (Cryptographic Message Syntax) is a binary format for encrypting and signing data. CMS relies on X.509 certificates.
Public key distribution: top-down hierarchy, a small number of "root CAs" signs certificates for many users

PGP family

Pretty Good Privacy (PGP) is an original program that had an patent and cipher export issues. (www.pgp.com)

OpenPGP is an Internet standard (RFCs) with own encryption methods (similar to CMS) and encoding formats (i.e. "ASCII Armor").
Public key distribution: Web of Trust (WoT), decentralized PKI, everybody is a CA. (the idea being that if an attacker "cannot fool everybody for a long time").

GnuPG/GPG GPL Licensed alternative to the PGP suite, compliant with OpenPGP standart.

Encrypting/signing email content

PGP/inline

```
Date: Sat, 28 Feb 2015 01:51:21 +0100
From: Pavel Ruzicka <ruza@ruza.eu>
User-Agent: Mozilla/5.0 (X11; Linux x86_64) Gecko Thunderbird
MIME-Version: 1.0
To: = <pruzicka@catta.cz>
Subject: PGP/inline
OpenPGP: id=943E7ECC;
url=http://ruza.eu/ruzagpg.txt
Content-Type: text/plain; charset=utf-8
Content-Transfer-Encoding: quoted-printable


-----BEGIN PGP MESSAGE-----
Charset: utf-8
Version: GnuPG v2
Comment: Using GnuPG with Thunderbird - http://www.enigmail.net/

hQQOA/fur1mYBwwdEA/+M/ehR/7j+96nQ+RZ86Tb+4NnOH+ce2hL1sjTPMOy/cbv
...=3Dx7sm
```

```
Subject:  PGP/MIME
OpenPGP: id=943E7ECC;
url=http://ruza.eu/ruzagpg.txt
Content-Type: multipart/encrypted;
 protocol="application/pgp-encrypte
 boundary="ueAP6jg0sa2iFFIt67Uib4GP


This is an OpenPGP/MIME encrypted m
--ueAP6jg0sa2iFFIt67Uib4GPcqrhxMSqS
Content-Type: application/pgp-encry
Content-Description: PGP/MIME versi

Version: 1


--ueAP6jg0sa2iFFIt67Uib4GPcqrhxMSqS
Content-Type: application/octet-str
Content-Description: OpenPGP encryp
Content-Disposition: inline; filena

-----BEGIN PGP MESSAGE-----
Version: GnuPG v2
```

S/MIME vs PGP coexistence

If you want to use S/MIME you should not enable the Enigmail option "encrypt if possible" (nor the one from S/MIME).

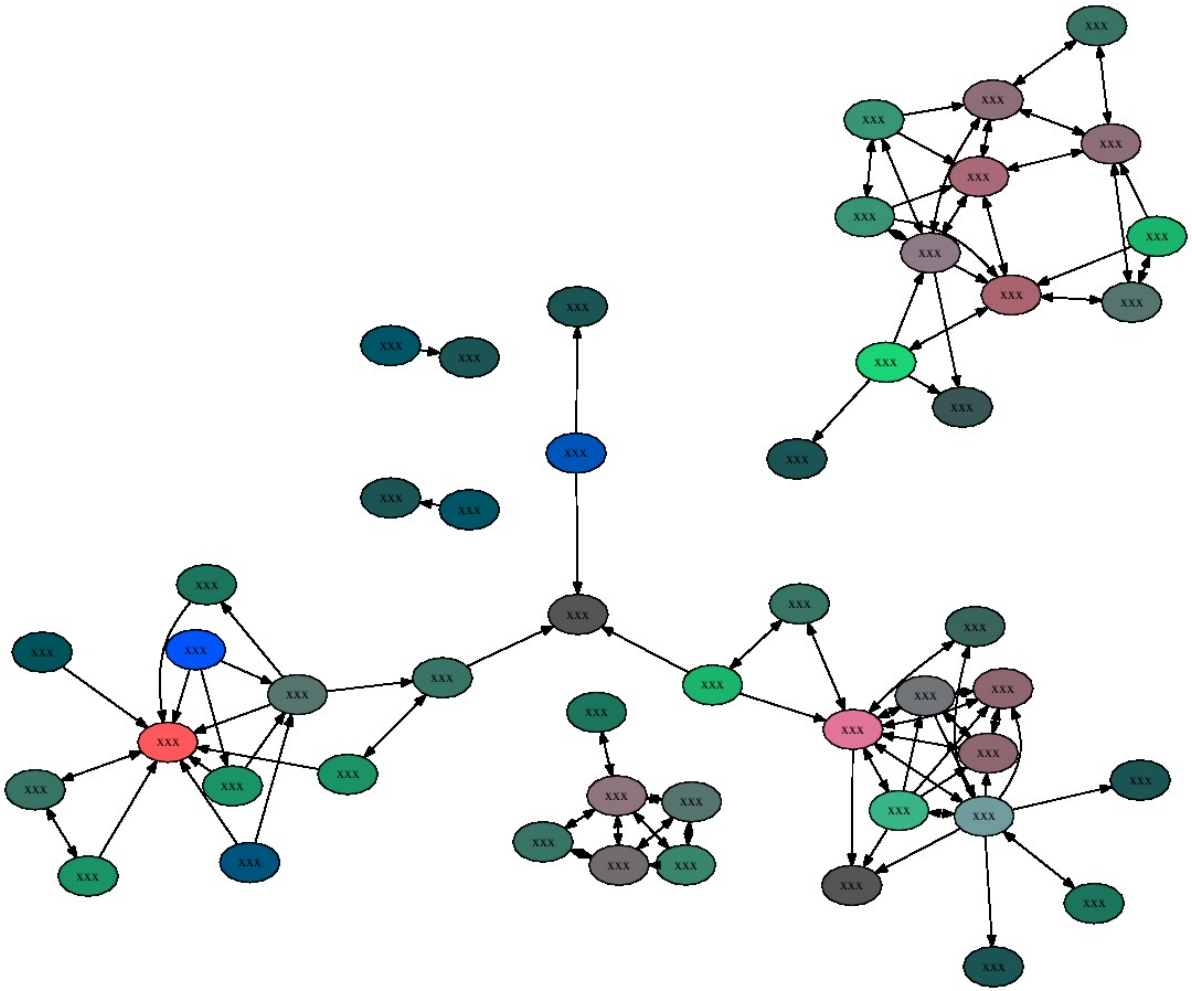
Web of (shy) Trust

```
$ gpg2 --keyserver hkp://pool.sks-keyservers.net --search-keys president@whitehouse.gov
gpg: data source: http://itunix.eu:11371
(1)     Barack Hussein Obama <president@whitehouse.gov>
          2048 bit RSA key AF19CFE9, created: 2014-05-25, expires: 2018-05-25
(2)     Obama <president@whitehouse.gov>
          2048 bit RSA key 3B9C907F, created: 2013-09-26 (revoked)
(3)     tESTER <PRESIDENT@whitehouse.gov>
          2048 bit RSA key 6B49DA35, created: 2013-05-24
(4)     Barak Obama (I'm the president) <obama@whitehouse.gov>
          2048 bit RSA key B110EE8F, created: 2010-12-09
(5)     Barack Hussein Obama (DOD) <president@whitehouse.gov>
          1024 bit DSA key 0B72EB0F, created: 2009-04-27, expires: 2012-01-20 (expired)
(6)     BUsh the past coming... <president@whitehouse.gov>
          1024 bit DSA key 6909AF98, created: 2008-10-27
(7)     clinton_lewinsky <president@whitehouse.gov>
          1024 bit DSA key AD3EE118, created: 2008-10-27, expires: 2013-10-26 (expired)
(8)     ElPresi! (the president of the white house...) <president@whitehouse.g
          2048 bit RSA key 0BCC736D, created: 2008-10-26
(9)     bushbushbushbushbush <president@whitehouse.gov>
          1024 bit DSA key E3F0063A, created: 2008-02-10
(10)    George Bush (I am a fag. I support the NWO.) <president@whitehouse.gov
          512 bit DSA key DE415F3C, created: 2008-01-26, expires: 2008-01-27 (revoked) (
(11)    abc <president@whitehouse.gov>
          1024 bit DSA key CEBBC2C4, created: 2007-10-27
Keys 1-11 of 27 for "president@whitehouse.gov".  Enter number(s), N)ext, or Q)uit >
```

http://pool.sks-keyservers.net:11371/pks/lookup?op=vindex&search=president%40whitehouse.gov

brmlab hackerspace prague

WoT metadata



gpgit.pl

https://github.com/mikecardwell/gpgit
https://grepular.com/

gpgit.pl - enable procmail filtering

System-wide (postfix)

```
mailbox_command = /path/to/procmail
```

per user

```
ruza@mail:~$ cat ~/.forward
|/usr/bin/procmail
```

gpgit.pl - .procmailrc

```
:0 cfw
| ~/gpgit.pl ruza@ruza.eu ruza+gsm@ruza.eu

:0 cfw
| /usr/bin/formail -A "X-GPGit: applied"
```

Android PGP/inline

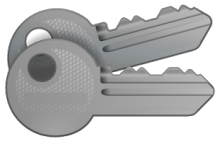
Email client:

K-9 mail

Kaiten mail

Crypto provider:

APG



OpenKeychain

http://en.flossmanuals.net/k9/encryption-and-security/

Android PGP/MIME+S/MIME

Email client:

Maildroid (FlipdogSolutions)

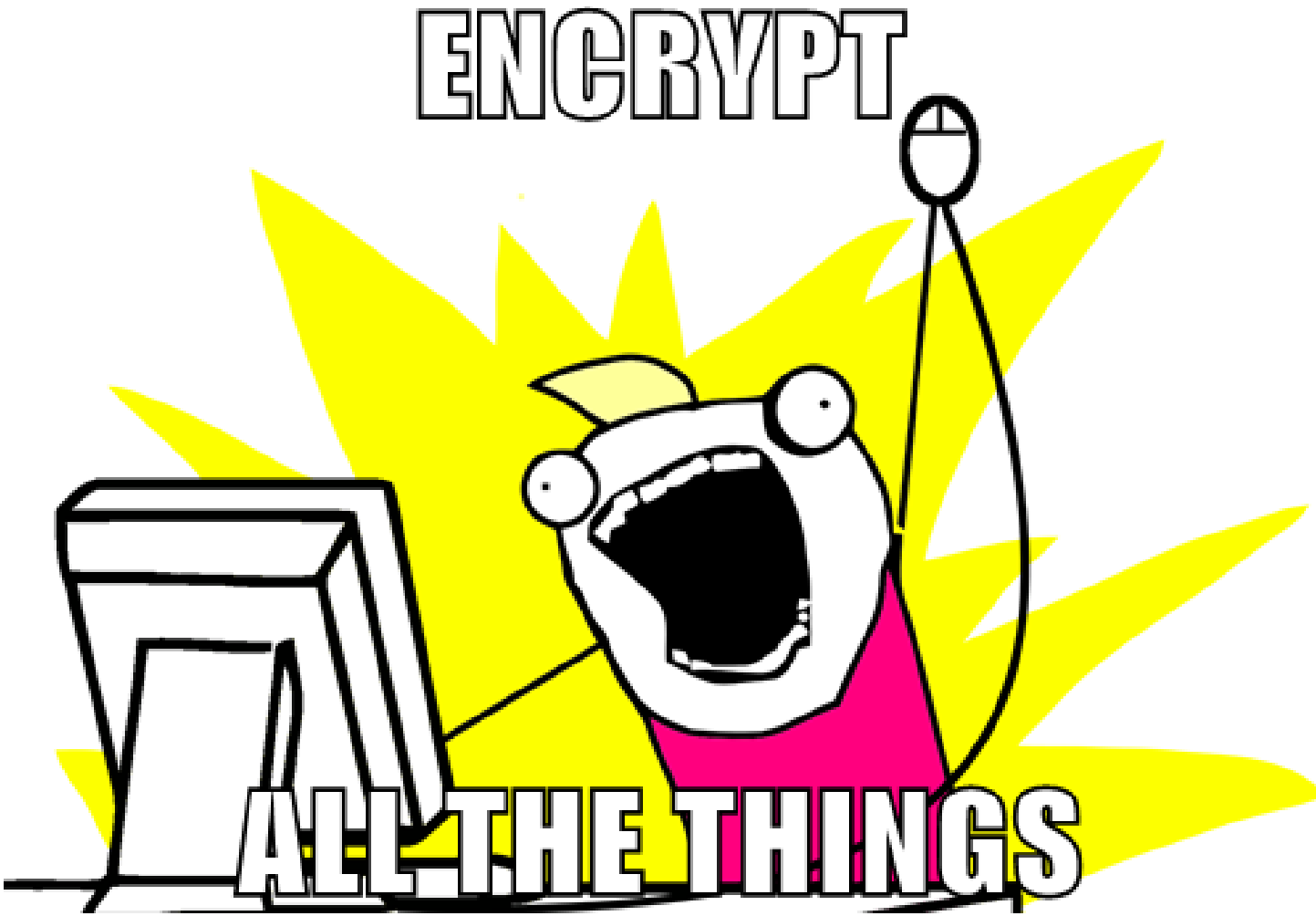
Crypto provider:

FlipdogSolutions Crypto Plugin

pass(1)

http://www.passwordstore.org/

F*ck Five Eyes, Nine Eyes, Fourteen Eyes and others too



brmlab hackerspace prague