NeoPG

An Alternative to GnuPG

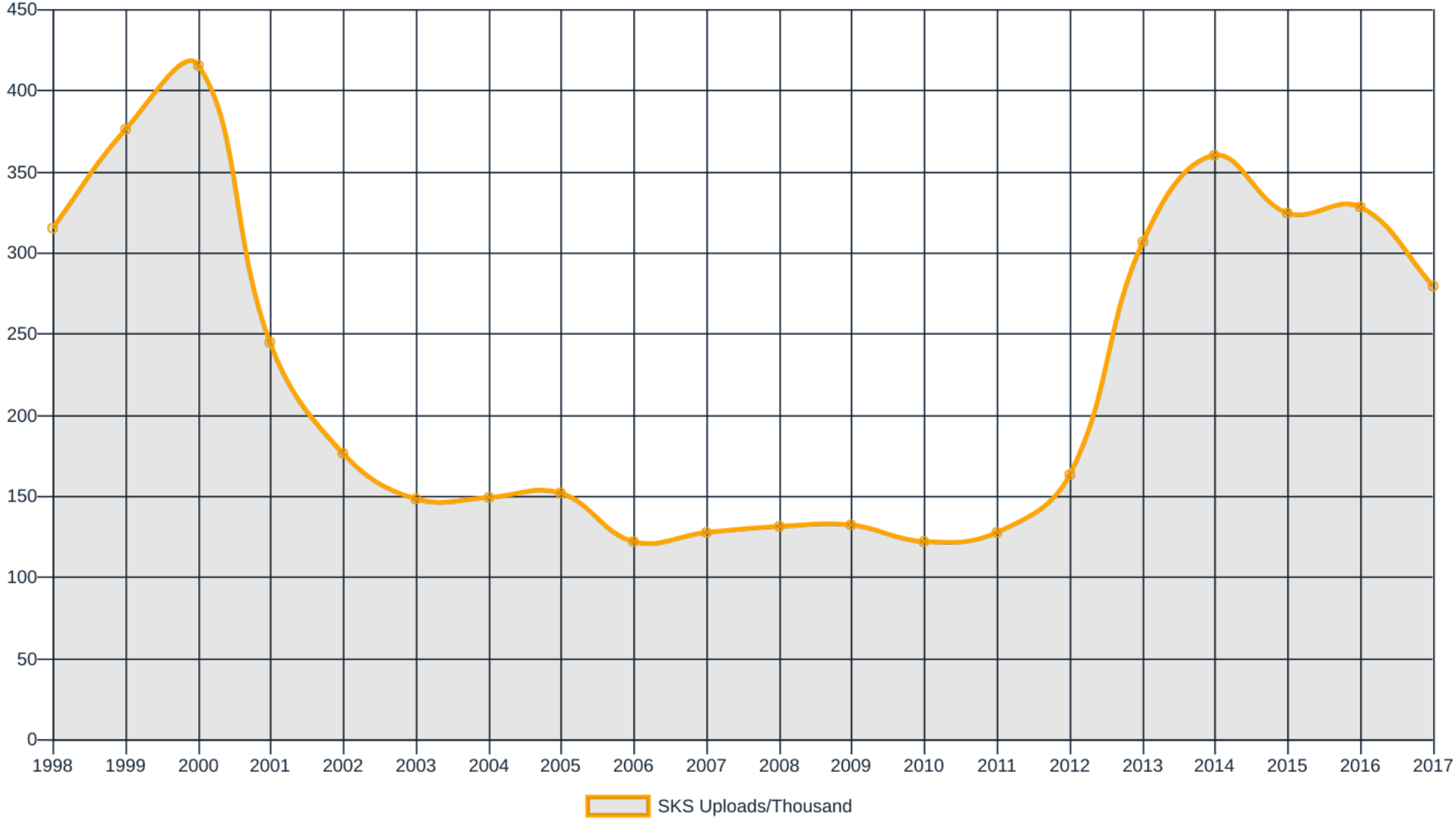
Marcus Brinkmann · BSI, Bonn (28. März 2018)

Overview

- OpenPGP
- Comparison of some implementations
- NeoPG

OpenPGP

Keys/1000 per year



Standardisation

- RFC2440 (1998)
- RFC3156 (2001): OpenPGP/MIME
- RFC4880 (2007) ← Current version
- RFC5581 (2009): Camellia block cipher
- RFC6637 (2012): ECDH+ECDSA (NIST)

Crypto-Museum (Examples)

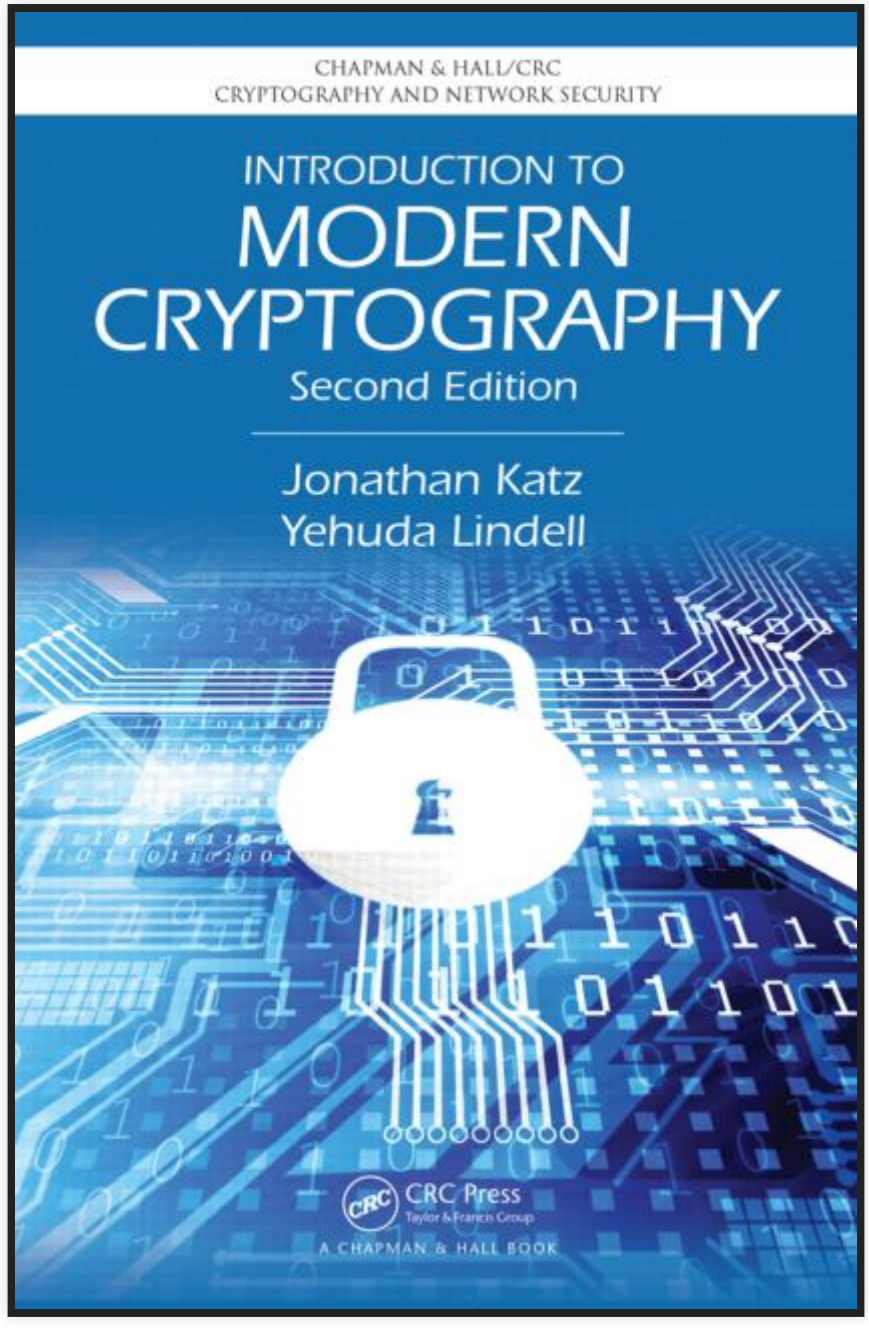
- SHA-1 (hardwired)
- 3DES, CAST5 (64-bit block cipher)

```
Werner Koch: AES256, AES192, AES, CAST5, 3DES
Triple Nerd: 3DES, CAST5, AES, AES192, AES256
GnuPG takes: 3DES
```

- CFB mode with integrity check
- No trust model

Shrinking Community

- "It's time for PGP to die." (Matthew Green, 2014)
- "I think of GPG as a glorious experiment that has run its course." (Moxie Marlinspike, 2015)
- "I'm giving up on PGP" (Filippo Valsorda, 2016)
- "Sorry, but I cannot decrypt this message. I don't have a version of PGP that runs on any of my devices." (Phil Zimmermann, 2015)


CHAPMAN & HALL/CRC
CRYPTOGRAPHY AND NETWORK SECURITY
INTRODUCTION TO
MODERN
CRYPTOGRAPHY
Second Edition
Jonathan Katz
Yehuda Lindell
CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

Serious Cryptography

A Practical Introduction to Modern Encryption

Jean-Philippe Aumasson

Foreword by Matthew D. Green

no starch press

OpenPGP Working Group (IETF)

June 2015: Beginning of RFC4880bis.

- Brainpool, SECG, Curve25519 for ECDSA+ECDH
- Integration of Ed25519
- SHA2-256 fingerprint
- AEAD mode (EAX)

Unfortunately missing:

- KDF Argon2 (to replace S2K)

OpenPGP Working Group (IETF)

"The Chairs and the AD have concluded that there is not sufficient interest to successfully complete the work of the [OpenPGP] working group."
– IESG Secretary, 11 Nov 2017

RFC4880bis

- Implemented in GnuPG (g10 Code GmbH)
- Implemented in RNP (Ribose Inc.)
- Curve25519+ECDH instead of X25519
- Informal development now

OpenPGP-Profile

Documenting:

- Scope of implementation of RFC4880(bis)
- Additional restrictions (input, output)
- Intentional differences

At least: One profile per implementation

Better: Shared profiles

Interoperability

Currently: Star-topology (test with standard and GnuPG)

- Shared test vectors
- Interoperability tests
- Better visibility of implementations
- Strenghten the ecosystem outside of GnuPG

Implementations

Overview

Project	Responsible	Language, Crypolib.	API	Platforms	License
GnuPG	g10 Code, GnuPG e.V.	C, libgcrypt	CLI, GPGME	Linux, Windows, macOS	GPLv3
Sequoia-PGP	PEP Foundation	Rust, libnettle	Rust, C	Linux, ?	tbd (GPLv3, LGPLv3)
OpenKeychain	Community	Java, BouncyCastle	Android Service	Android	GPLv3 (API: Apache)
RNP	Ribose, Inc.	C, Botan	C	Linux, macOS	BSD 2-clause
NeoPG	Community	C++, Botan	C++, C	Linux, macOS (Windows)	BSD 2-clause / GPLv3 (legacy)

API-Requirements (examples)

- Signature-verification *without* key management: `verify(data, signature, public_key)`
- Dito, for a specific subkey
- Public-Key-Informationen *without* import
- Packet-stream-analysis for research
- Stability *and* progress

Is there a GnuPG library?

2010

This has been frequently requested. However, the current viewpoint of the GnuPG maintainers is that this would lead to several security issues and will therefore not be implemented in the foreseeable future. However, for some areas of application gpgme could do the trick.

2013

No, nor will there be.

Tapping potential

- Applications know requirements better
- Allow unforeseen potential
- Exchange in both directions
- Better software quality in depth

NeoPG

NeoPG as an Alternative

- Fork of GnuPG
- Cleanup code, easier development
- Stable, extensible API
- Find new applications
- Easier to use

A Community

- Developed on GitHub
- Discussion over social media
- Requirements through community dialogue
- Change-documentation in issues, pull-requests
- Background information in blogs
- Code Review (4-Eyes)

Software-Engineering

High quality despite large changes.

- Coding Standards (`clang-format`, SonarQube)
- Continuous Integration Tests
- Test-Coverage
- Static Analysis
- Fuzzing

Architecture

- One repository
- One library
- One CLI-tool
- lightweight process isolation (Heartbleed)

Delegate Responsibility

- C++, STL, some Boost
- libcurl for network access
- SQLite for persistent database
- Botan as crypto-library

Command Line

- Git-style subcommands (`neopg encrypt ...`)
- Compatibility layer (`neopg gpg2 ...`)
- Colored terminal output
- Better diagnostics

Status

- Infrastructure established
- Minus 250.000 LoC (~50%)
- Minus 120 CLI options (~30%)
- First API parts (OpenPGP parser)

Example: Parser-Design

- Ad-Hoc Parser are common cause of errors
- Formal language for all untrusted input
- Consistent: even for trivial input (1 Byte!)
- C++ allows typesafe formal grammar …
- … parser generator without loss of performance
- Some mistakes can be avoided systematically

https://neopg.io/blog/no-ad-hoc-parser/

neopg.io

@neopg_