age als Alternative zu GPG ?

Rainer Peipp

02. Februar 2022

GPG / PGP: Vorbemerkung

- PGP (Pretty Good Privacy) ist ein von Phil Zimmermann 1991 entwickeltes Programm zum Verschlüsseln und Signieren von Dateien.
- Exportbeschränkungen behinderten den internationalen Einsatz
- 1995 wurde der Sourcecode als Buch (PGP Source Code and Internals) veröffentlicht. Daraus entstand PGPi.
- Die Rechte am Sourcecode wanderten durch verschiedene Hände (u. a. McAfee).
- Die daraus entstandenen Probleme führten zur Entwicklung des OpenPGP-Standards.
- GnuPG war die erste Implementierung dieses Standard als freie Alternative zu PGP.

GPG /PGP: Eigenschaften

- PGP verwendet ein Public–Key–Verfahren (asymmetrische Verschlüsselung) mit den Nutzern eindeutig zugeordneten Schlüsselpaaren.
- Die eigentliche Nachricht wird wird mit einem symmetrischen Verfahren und einem zufällig erzeugten Session–Key verschlüsselt.
- Für jeden Empfänger wird der Session–Key mit seinem zugehörigen Public–Key verschlüsselt.
- Eine Nachricht kann damit an mehrere Empfänger verschickt werden, ohne für jeden einzeln verschlüsselt werden zu müssen.
- Für die Sicherheit des Ansatzes ist die Schlüsselverwaltung elementar. Der Private–Key darf nie aus den Händen des Besitzers geraten.
- Es gibt keine zentrale Zertifizierungsinstanz für die Schlüssel. Das Vertrauen wird von den Benutzern mit einem Web–of–Trust–Ansatz selbst verwaltet.

PGP / GPG: Funktionsweise

- Eine Nachricht kann entweder
  - signiert (unterschrieben),
  - verschlüsselt oder
  - signiert und verschlüsselt

  werden.
- Die Signatur sichert die
  - Authentizität (von wem stammt die Nachricht) und
  - Integrität (die Nachricht ist unverfälscht).
- Weitere notwendige Arbeitsschritte:
  - Schlüsselerstellung (für den symmterischen Schlüssel)
  - Hashbildung
  - Komprimierung
  - Codierung

PGP /GPG: Status

- Sehr weit verbreitet
  - Unterstützung auch bei GMX und Web.de (Mailvelop)
  - Pretty Easy Privacy als Versuch, die Komplexität zu reduzieren
  - Integration in viele Mail–Programme (Thunderbird)
- Auf allen relevanten Plattformen verfügbar
- Schleppt viel historischen Ballast mit:
  - Viele Algorithmen: RSA, DSA, AES, IDEA, Elgamal, ...
  - Verschiedene Schlüsselvarianten
  - Verschiedene Hash-Algorithmen
  - Verschiedene Encodings
  - Etc.
- Schlüsselaustausch komplex

PGP /GPG: Kritik

- Schlüsselverwaltung mit systemischen Mängeln
  - Der Keyserver-Ansatz gilt als gescheitert
- Viele Implementierungsfehler
  - Mail-Verschlüsselung: Thunderbird schlampte mit PGP-Schlüsseln (https://www.heise.de/news/Thunderbird-schlampte-mit-PGP-\Schluesseln-6051767.html)
  - S/MIME und PGP: E-Mail-Signaturprüfung lässt sich austricksen (https://www.heise.de/security/meldung/S-MIME-und-PGP-E-Mail-Signaturpruefung-laesst-sich-austricksen-4411230.html)
  - PGP und S/MIME: E-Mail-Verschlüsselung akut angreifbar (Efail-Lücke) (https://www.heise.de/security/meldung/PGP-E-Mail-Verschluesselung-akut-angreifbar-4048489.html)
  - pEp-Foundation hat Sicherheitslücke in Enigmail/pEp geschlossen (https://www.heise.de/security/meldung/pEp-Foundation-hat-\Sicherheitsluecke-in-Enigmail-pEp-geschlossen-4191426.html)
- Selbst der Erfinder (Phil Zimmermann) benutzt PGP nicht (https://www.vice.com/en/article/vvbw9a/even-the-inventor-of-\pgp-doesnt-use-pgp)

PGP /GPG: Links / Literatur

- c't special Daten Schützen 2021, S. 26 ff.: Sylvester Tremmel: E-Mails bestmöglich absichern
- c't 19/2020, S. 154 ff.: Sylvester Tremmel: Eingebaute Verschlüsselung, OpenPGP–UnterStützung in Thunderbird 78
- Sylvester Tremmel: FAQ: PGP–Verschlüsselung mit Thunderbird `https://heise.de/-5074725`
- Linux Magazin 01/2022, S. 46 ff.: Peer Heinlein: Private Post, Praxiswissen: Grundlagen der E–Mail–Verschlüsselung

age: Actually Good Encryption

(*nicht ganz ernst gemeintes Akronym*)

Was ist age?

- Beschrieben in "A simple file encryption tool & format" (`https://docs.google.com/document/d/11yHom20CrsuX8KQJXBBw04s80Unjv8zCg_A7sPAX_9Y/preview`)
- Lizenz BSD 3-Clause
- Geschrieben in Go (`https://github.com/FiloSottile/age`)
- Portierung in Rust verfügbar (`https://github.com/str4d/rage`)
- Besteht nur aus zwei Programmen
  - age
  - age-keygen

age: Entwicklungsziele

- Sehr einfache Kommandozeilen-Syntax
- Gute Eignung für die Nutzung in Pipes
- Gut als Backend für andere Programme zu nutzen
- Kleine Public/Private–Schlüsselpaare
- Optional können (vorhandene) ssh-keys genutzt werden
- Kaum Konfigurationsmöglichkeiten
- Moderner (streambarer) Verschlüsselungsalgorithmus aus der Klasse der AEADs (Authenticated Encryption with Associated Data)

age: Geplant für später

- Passwort–Backend über `https://www.passwordstore.org/`
- YubiKey PIV support via PKCS#11
- Passwort–geschützte Schlüssel
- age–mount für verschlüsselte Dateisysteme

age: Nicht vorgesehen

- Archivierung
- Jede Art von Signieren (Empfehlung, signify/minisign zu nutzen)
- Signieren von git commits oder Packages
- Irgend etwas mit Email
- Web–of–Trust bzw. Schlüsselverteilung

age: Beispiele 1

Schlüssel generieren:

```
$: age-keygen
# created: 2022-02-02T16:10:37+01:00
# public key: age1v72ztgs8lulymg32dvaku7p77uw8kge52um498r5cp8psc95qawsdq6qpd
AGE-SECRET-KEY-1KWH9KYDFDDXP37NS906Q4Y9T5GR6SNLSMFMXT7ECTP2MLLMXFALQKDPW6U

$: age-keygen >> ~/.config/age/keys.txt
Public key: age1h2rfgjalmgsay6mt3ljqt946ye09yawe23gn6zdrquk8hafl2dcqmnsatd
```

Verschlüsselung mit Public–Key:

```
echo "hi" | age -r age1h2rfgjalmgsay6mt3ljqt946ye09yawe23gn6zdrquk8hafl2dcqmnsatd -o hello.age
```

Verschlüsselung mit mehreren Public–Keys:

```
echo "hi" | age -r age1h2rfgjalmgsay6mt3ljqt946ye09yawe23gn6zdrquk8hafl2dcqmnsatd \
 -r age1sg4km0h54lefxq93skhqdm93k0clvzm9cr52jkmc6tfks9fh45dsqwvegr -o hello2.age
```

Verschlüsselung mit Passwort

```
echo "hi" | age -p -o hello_pw.age
```

age: Beispiele 2

Verschlüsselung an mehrere Empfänger über Dateiliste:

```
$: echo age1sg4km0h54lefxq93skhqdm93k0clvzm9cr52jkmc6tfks9fh45dsqwvegr > empfaenger.txt
$: echo age166nrwszfttwa5qxz79khus47qsw5aw4xgvxpwuja84zujgedh5mqav3zgn >> empfaenger.txt
$: echo "hi" | age -R empfaenger.txt -o hello4.age
```

Verschlüsselung mit (vorhandenen) SSH–Keys

```
$: echo "hi" | age -R ~/.ssh/id_rsa.pub -o hello_ssh.age
$:
$: cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAuKfXubPaneFBCsX6BsTaKJ2ZhOCoU3szUeSxE+Xob74\
ZFQ6UjOWxoW0F+lqIxP/iR85zexvfVkD8ngSSddqqr+KqCBWTRlmKu2mg5YpYqIJSAveaYi3KbMB6yk\
ur1zXdkAsHQw9k6XYW6p2K3YpnVmEK6MRvGxihKdN7Uwcn0uNQiQZMRyoOZ/7sMYe353kXRcuHbZrpT\
Os9OVvh1LcynGoLCnKQsshUKYv4M9aF8MAp76vXOWNRavnrWWmq0ruM42tRAM2TbN0ToHUdxIXMW1G0\
zaSe10Qc9tUmn+1ombqZDTrJQGR3tZB8pKwGfjjDQoHwT8nc2vnwP8eddUrLZw== peipp@rpc001
$:
$: echo "hi" |age -r "ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAuKfXubPaneFBCsX6BsTaK\
J2ZhOCoU3szUeSxE+Xob74ZFQ6UjOWxoW0F+lqIxP/iR85zexvfVkD8ngSSddqqr+KqCBWTRlmKu2mg\
5YpYqIJSAveaYi3KbMB6ykur1zXdkAsHQw9k6XYW6p2K3YpnVmEK6MRvGxihKdN7Uwcn0uNQiQZMRyo\
OZ/7sMYe353kXRcuHbZrpTOs9OVvh1LcynGoLCnKQsshUKYv4M9aF8MAp76vXOWNRavnrWWmq0ruM42\
tRAM2TbN0ToHUdxIXMW1G0zaSe10Qc9tUmn+1ombqZDTrJQGR3tZB8pKwGfjjDQoHwT8nc2vnwP8edd\
UrLZw==" -o hello_ssh.age
$:
$: age -d -i ~/.ssh/id_rsa hello_ssh.age
hi
```

Verschlüsselung an eine Liste von Empfängern über https–URL

```
$: curl https://github.com/benjojo.keys|age -R - -o hello_ssh.age hello.txt
```

Fazit

- age ist kein Ersatz für gpg, aber eine Alternative für bestimmte Einsatzzwecke
- age ist gedacht als (möglicher) Ersatz für gpg zur Dateiverschlüsselung, Backup, Streams, ...
- age wird sich nicht um Signaturen, Schlüsselverwaltung und Email-Integration kümmern

Links

Heise: age 1.0.0: Neue und simplere GPG-Alternative
`https://www.heise.de/news/age-1-0-0-Neue-und-simplere-GPG-Alternative-6185439.html`

Neil Madden: A few comments on ‘age’
`https://neilmadden.blog/2019/12/30/a-few-comments-on-age/`