Autocrypt automatisiert E-Mail-Kryptografie

# Dargereicht

**Jan Bundesmann**

Autocrypt will Anwendern die Verwaltung kryptografischer Schlüssel erleichtern. Mehrere Clients implementieren den Standard bereits.

Aggressiv Schlüssel verteilen, aber nur behutsam verschlüsseln – einer der Grundsätze aus der Autocrypt-Spezifikation. Das Dokument, Ende 2017 in Version 1.0 erschienen, definiert Regeln zur Verschlüsselung von Nachrichteninhalten und zum Schlüsseltausch. Es wurde speziell unter dem Gesichtspunkt Usability entwickelt. Denn daran mangelt es bei den derzeit geläufigen Verfahren zur E-Mail-Verschlüsselung, meinen unter anderem die Autocrypt-Entwickler. Sie beziehen sich dabei auf die zwei gängigen Verfahren OpenPGP und S/MIME.

Bisher liegt die Schlüsselverwaltung von OpenPGP vollständig in der Hand der Nutzer. Diese müssen für jeden Kontakt die Dateien von einem öffentlichen Schlüsselserver herunterladen. Da diese Server keine Mechanismen bieten, die hinterlegte Identität zu überprüfen, geschieht dies Out of Band, in einem separaten Kanal. Es ist zum Beispiel denkbar, dass jemand – absichtlich oder aus Versehen – einen öffentlichen Schlüssel unter einer falschen E-Mail-Adresse ablegt. Daher müssen Nutzer zusätzliche Arbeit ins Vertrauensmanagement stecken. Bei GnuPG gibt man Schlüsseln ein Vertrauenslevel. Die höchste Stufe sollte man dann wählen, wenn man von der Echtheit des Schlüssels überzeugt ist. Das gilt unter anderem, wenn man die Identität festgestellt hat, indem man etwa den Fingerprint des Schlüssels in Anwesenheit des Inhabers abgeglichen hat. Das lässt sich auch auf die Server hochladen. Dort ist hinterlegt, wer für die Echtheit des Eintrags bürgt. Ist eine vertrauenswürdige Stelle dabei, kann man auch ohne persönliche Prüfung einen Schlüssel verwenden – das ist die Idee hinter dem Web of Trust.

S/MIME hat das durch eine hierarchische Struktur gelöst: Anwender kaufen im Prinzip das Vertrauen bei einer dafür vorgesehenen Stelle, die den Schlüssel signiert und ein digitales Zertifikat ausgibt. Diese Stelle hat wiederum eine höhere Instanz beglaubigt, wodurch eine Zertifikatskette entsteht, an deren Spitze spätestens ein vertrauenswürdiges Gegenüber stehen sollte. Etliche Anbieter, auch kostenlose, stellen Zertifikate zur Verfügung.

Ein Client, der Autocrypt implementiert hat, verwendet wie PGP und S/MIME ein Public-Key-Verfahren – derzeit RSA mit 3072-Bit-Schlüsseln. Es ist keine zentrale Instanz involviert, die Software wickelt die Schlüsselverwaltung im Hintergrund ab, ohne dass Nutzer eingreifen müssen – so der Kerngedanke. Dass Nutzer Schlüssel direkt vom Inhaber erhalten, sehen die Entwickler als Grundlage für „Trust on first Use“ (TOFU). Im Idealfall tauschen die üblichen Verdächtigen Alice und Bob zwei Mails aus und sind ab dann im Besitz des öffentlichen Schlüssels ihres jeweiligen Kommunikationspartners. Alle künftigen Nachrichten sind verschlüsselt und signiert. Autocrypt vertraut der digitalen Unterschrift automatisch.

Anders als bei OpenPGP-Schlüsselservern sehen Nutzer andere Public Keys nur, wenn diese über eine E-Mail hereinkamen, die ihr Provider geprüft hat. Viele E-Mail-Provider fangen Phishing-E-Mails über spezielle Verfahren wie DKIM ab. Also auch solche E-Mails, die versuchen, mit einem vorgetäuschten Absender einen falschen Autocrypt Key unterzuschieben.

Damit verfolgen die Entwickler von Autocrypt ähnliche Ziele wie die von Pretty Easy Privacy – einfacheres Trust-Management. Letztere allerdings liefern eine Library, die sich in verschiedene Mail-Clients einbinden lässt. Autocrypt basiert hingegen auf einer Spezifikation, die Mail-Client-Entwickler selbst implementieren. Das Projekt beschreibt sich auf der Website als „social and technical effort“. Deshalb versuchen die Teilnehmer, insbesondere Entwickler ins Boot zu holen, damit frühzeitig Implementierungsbelange einfließen.

## Im Kopf

An die Stelle einer zentralen Schlüsselverwaltung tritt bei Autocrypt die aktive

Schlüsselverteilung im E-Mail-Header. Dort findet sich in einem eigenen Eintrag *Autocrypt* die E-Mail-Adresse als Identifikation für den Schlüssel, der ebenfalls dort platziert ist. Außerdem kann man seine Präferenz übermitteln, unter welchen Umständen kryptografische Operationen anzuwenden sind:

```
Autocrypt: addr=a@b.example.org; ↵
        [prefer-encrypt=mutual;] keydata=BASE64
```

*keydata* enthält einen Schlüssel im Format eines OpenPGP „Transferable Public Key“. Ein Autocrypt-Client liest diese Informationen aus eingehenden Mails. Zu jedem Kommunikationspartner speichert es, ob Verschlüsselung möglich und erwünscht ist. Damit entscheidet es autonom beim Verfassen einer Mail, ob Klartext verschickt wird oder chiffrierte Inhalte. Gemäß dem Grundsatz, nur behutsam zu verschlüsseln, empfiehlt Autocrypt dies nur in drei Fällen:

1. Der Absender wünscht dies explizit.
2. Man antwortet auf eine verschlüsselte Mail.
3. Alle Kommunikationsteilnehmer verwenden Autocrypt und haben in ihren Einstellungen hinterlegt – also per Mail-Header kundgetan –, dass sie verschlüsselte Mails bevorzugen.

## Voller Zugriff auf den Mail-Header

Da alle Autocrypt-Logik im Header geschieht, muss der Mail-Client diesen durchsuchen und individuell gestalten können. Für Webinterfaces müssen die Provider ihr Interface anpassen. Allerdings arbeiten beispielsweise auch die Entwickler der Browsererweiterung Mailvelope daran, Autocrypt zu integrieren, um damit Ende-zu-Ende-Verschlüsselung für Webmail-Clients zu implementieren.

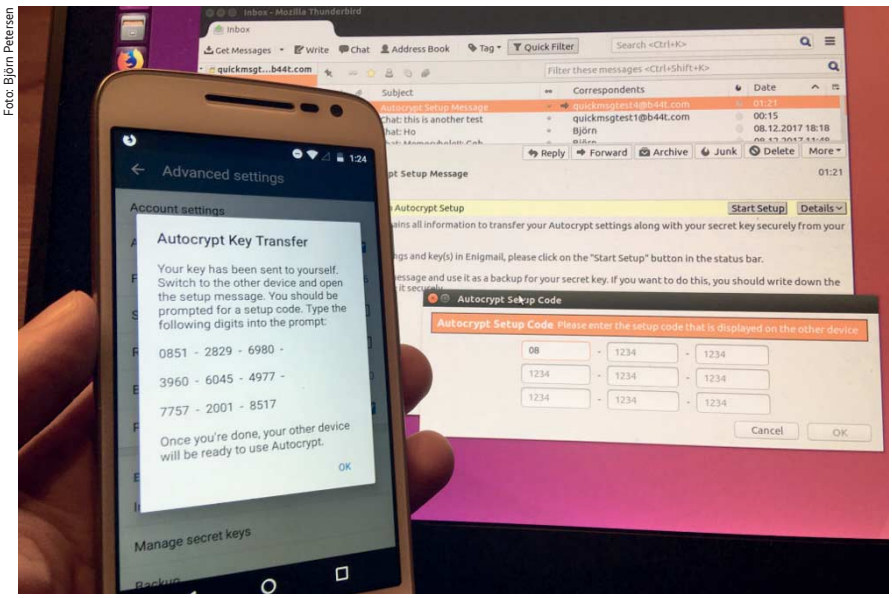
Tatsächlich muss für das Projekt erst noch die Zahl der Mail-Clients mit Autocrypt-Integration wachsen (siehe Kasten „Welche Software implementiert Autocrypt?“).

## Welche Software implementiert Autocrypt?

Da Autocrypt lediglich eine Spezifikation ist, müssen die Entwickler der Mail-Clients selbst für eine Implementierung sorgen, während beispielsweise Pretty Easy Privacy (pEp) Plug-ins für eine Vielzahl von Clients zur Verfügung stellt. In der Folge ist die Menge der Autocrypt-Clients noch relativ überschaubar. Dazu gehört das Thunderbird-Plug-in Enigmail. Für PCs gibt es auch einige Tools, die sich in weitere Software einbetten lassen, wie *muacrypt*, *pyac* und *gmime*.

Auch Android-Nutzer können Autocrypt verwenden. Das erlaubt derzeit der Mail-Client K-9 Mail. Die Schlüsselverwaltung übernimmt in diesem Fall OpenKeychain. *delta.chat* ist ein Messenger, den man über den alternativen App-Store F-Droid beziehen kann. Er verwendet die Telegram-UI, aber nicht deren Backend. Stattdessen basiert *delta.chat* auf E-Mail-Infrastruktur: Per IMAP-Push erhält er eingehende Nachrichten, der Versand läuft über SMTP.

Foto: Björn Petersen




**Möchte man mehrere Clients mit Autocrypt betreiben, lassen sich die Einstellungen per Setup Message übertragen. Dabei handelt es sich um eine AES-verschlüsselte E-Mail. Den Schlüssel müssen Anwender von Hand auf dem zweiten Gerät eingeben.**

Um die Usability zu erhöhen, fiel die Entscheidung der Entwickler auf den Header als Transportmedium anstelle eines Anhangs. Damit wollte man unter anderem vermeiden, dass Nutzer ohne das nötige Wissen abgeschreckt werden – etwa durch einen Anhang, den sie für Spam oder Schadsoftware halten. Denn: Wichtiger als die Verschlüsselung seien letztlich die kommunizierten Inhalte, weswegen die Schwelle, auf Kryptografie zu verzichten, relativ niedrig gesetzt ist.

Damit nicht zuvor geheime Inhalte plötzlich offengelegt werden, sollten Autocrypt-Clients keine Passagen beim Antworten auf verschlüsselte Mails standardmäßig einbinden – etwa als Zitat –, wenn die Antwort im Klartext verfasst wird.

Derzeit ist Verschlüsselung mit 3072-Bit-RSA-Schlüsseln vorgesehen. Der Client soll davon zwei Paare anlegen: eines zum Signieren und Selbstzertifizieren, eines zum Verschlüsseln. Eine verschlüsselte Mail im Sinne von Autocrypt ist immer sowohl verschlüsselt als auch signiert.

Für den Betrieb mit mehreren Clients oder auf mehreren Geräten sieht die Spezifikation Setup Messages vor. Das sind E-Mails mit allen Informationen, um Autocrypt einzurichten, die AES-128-verschlüsselt sind. Der hierfür nötige Schlüssel besteht aus 36 Ziffern. Anwen-

### iX-TRACT

- Da Ende-zu-Ende-Verschlüsselung bei E-Mails nach wie vor die Ausnahme bildet, arbeiten mehrere Initiativen daran, diese zu vereinfachen.
- Die Spezifikation Autocrypt ist im Dezember 2017 in Version 1.0 erschienen. Sie soll das Schlüsselmanagement erleichtern und automatisch verschlüsseln, wenn das die Kommunikationsteilnehmer wünschen.
- Erste Clients haben Autocrypt integriert.

### iX-Wertung

⊕ Verteilung kryptografischer Schlüssel ohne zentrale Architektur

⊕ Synchronisation mehrerer Clients über Setup Messages

⊖ keine Authentifizierung der Zuordnung von Schlüsseln zu Nutzern

der müssen ihn zum Aktivieren einer neuen Anwendung von Hand übertragen. Die Clients bieten hierfür ein eigenes Eingabefenster.

Der Schlüssel lässt sich aber auch in eine beliebige PGP-Installation importieren. Dafür muss der Software der Inhalt der Setup-Mail zum Entschlüsseln übergeben werden, darin verpackt befindet sich der geheime Schlüssel, den Autocrypt erstellt hat. Nutzer benötigen die erwähnten 36 Ziffern zum Entschlüsseln – inklusive der Bindestriche. So lassen sich die Autocrypt-verschlüsselten Nachrichten auch auf anderen Clients lesen, die lediglich OpenPGP beherrschen. Diese nehmen natürlich keine Anpassungen im Mail-Header vor, sodass nicht sichergestellt ist, ob die Verschlüsselungspräferenzen erhalten bleiben.

Allerdings ist Autocrypt nicht unumstritten. Hauptkritik ist die zugrunde liegende opportunistische Sicherheit aus dem RFC 7435. Diese sieht „die meiste Zeit etwas Schutz“ (some protection most of the time) vor. Klartextkommunikation ist die Basis, erst nach mindestens zwei Mails aktivieren die Teilnehmer Verschlüsselung – wie übrigens auch bei pEp. Die Entwickler von Autocrypt sehen darin insgesamt eine Stärke ihres Ansatzes, denn in ihren Augen steigt die Zahl der verschlüsselten Mails und das führt letztlich zu einer größeren Verbreitung von Ende-zu-Ende-Verschlüsselung. Damit schützt der Ansatz aber primär vor passiven Angriffen, also dem groß angelegten Mitschneiden des Mailverkehrs.

## Den Schlüssel auf dem Mailserver tauschen

Spezifikation 1.0 sieht keine Authentifizierung der Schlüssel vor, wodurch sich für aktive Angreifer ein Hebelpunkt ergibt. Haben diese etwa Einfluss auf den Betreiber des Mailservers, können sie die Header der versandten E-Mails austauschen. Mangels kryptografischer Validierung der Schlüssel und TOFU ist das für Empfänger nicht ersichtlich. Die Autoren des Privacy-Handbuchs empfehlen deshalb sogar, Autocrypt in Enigmail 2.0 komplett zu deaktivieren. Die Betreiber des Mail-Providers mailbox.org warnen vor der „trügerischen Sicherheit“, die Autocrypt bietet. Sie setzen selbst auf das Projekt Vertrauenswürdige Verteilung von Verschlüsselungsschlüsseln (VVV), an dem auch mailbox.org arbeitet. VVV soll auf Grundlage der DNS-Sicherheitserweiterungen DNSSEC die Echtheit von kryptografischen Schlüsseln sicherstellen und ist eine Alternative zu Autocrypt (siehe Kasten „Wer möchte noch die Schlüsselverteilung vereinfachen?“).

Eine Lösung plant die Autocrypt-Community für eine der kommenden Releases. Bereits jetzt erschweren in ihren Augen Man-in-the-Middle-Angriffe, dass DKIM-Signaturen auf die Autocrypt-Header es erfordern, an vielen Stellen anzugreifen. Zusätzlich empfehlen sie Out-of-Band-Verifikation für alle, die sich vor aktiven Angriffen schützen wollen. Sie weisen darauf hin, dass etwa auch der Messenger Signal ohne Schlüsselverifikation ebenso anfällig gegen einen aktiven Angriff ist wie Autocrypt.

## Fazit

Autocrypt hat ein nutzbares Stadium erreicht, leidet aber derzeit wie Pretty Easy Privacy unter einem Mangel an fertigen Clients. Um für Nutzer interessant zu werden, sind Software und eine ausreichende Verbreitung allerdings notwendige Voraussetzungen.

In Anbetracht der Alternativen, die gleichzeitig entstehen, stellt sich auch die Frage nach der Kompatibilität mit den anderen Ansätzen. Zusätzlich sollten vorhandene Schlüsselsammlungen nicht einfach ihren Wert verlieren.

Die Kritik an Autocrypt bezieht sich nicht auf dessen Implementierung, sondern auf grundsätzliche Designentscheidungen wie den Verzicht auf eine kryptografische, automatisierte Verifizierung der Schlüssel. Damit lasse sich lediglich kryptografisches Rauschen erzeugen, das vor passiven Gefahren schütze. Aus der Sicht der Entwickler sind allerdings Szenarien für aktive Angriffe auf Autocrypt trotzdem nicht einfach. Und wer sich vor aktiven Angreifern schützen will, sollte auch weiterhin auf ein explizites Verifizieren der Daten setzen. (jab@ix.de)

## Wer möchte noch die Schlüsselverteilung vereinfachen?

Nach Jahrzehnten reiner Lehre und Lamentieren über komplizierte zentrale Schlüsselinfrastukturen gibt es inzwischen vier Projekte, die sich diesem Thema widmen. Auch die drei Projekte neben Autocrypt kommen ursprünglich aus dem deutschsprachigen Raum und werden maßgeblich dort entwickelt.

GnuPG hat in den letzten zwei Jahren das Konzept der Web Key Directories entwickelt. Das erfordert weiterhin eine Serverinfrastruktur. Mail-Clients sollen automatisch beim Verfassen einer Nachricht von den Servern öffentliche Schlüssel holen – geht es nach den GnuPG-Entwicklern, übernehmen die Mail-Provider den Betrieb der dafür nötigen Infrastruktur und implementieren es zudem in ihre Web-Clients.

Pretty Easy Privacy (pEp) implementiert ein Peer-to-Peer-Netz zum Schlüsselaustausch. Codierte Nachrichten werden gemäß dem OpenPGP-Standard verschlüsselt verschickt, wenn keine andere Methode verfügbar ist. Gibt es einen direkten Onlinekanal zwischen den Kommunikationspartnern, läuft der Versand über das P2P-Netz oder über eine Anonymisierungsplattform wie Qabel. Im Gegensatz zu Autocrypt versucht pEp, E-Mails immer zu verschlüsseln. Klartextnachrichten sind lediglich als Notlösung vorgesehen.

Ein vom Bundesministerium für Bildung und Forschung gefördertes Projekt heißt „Vertrauenswürdige Verteilung von Verschlüsselungsschlüsseln“ (Projekt VVV). Daran beteiligt sind Universitätsinstitute und mit mailbox.org ein Mail-Provider. Die Leitung liegt beim Fraunhofer-Institut für Sichere Informationstechnologie. Nach dem VVV-Konzept liegen öffentliche Schlüssel an einer definierten Stelle, die Clients aus dem DNS-Eintrag der E-Mail-Domain in Form eines „Service Resource Record“ abfragen können. Dieser ist mittels DNS Security Extensions (DNSSEC) signiert. Durch eine vollständige Kette aus Signaturschlüsseln bis zur Root-Domain des DNS sollen Nutzer so bezogenen Schlüsseln vertrauen können.

Alle vier Ansätze sollen im Hintergrund ablaufen. Mail-Clients beziehen oder sammeln automatisch Schlüssel. Das manuelle Beschaffen fällt weg, ebenso das händische Setzen des Vertrauenslevels – weil es entweder gegeben ist oder die Spezifikation darauf verzichtet.

*Alle Links: ix.de/ix1805064*