Bernhard Esslinger, Ludger Schmidt, Benjamin Strenge, Arno Wacker

Unpopuläre E-Mail-Verschlüsselung – nicht nur ein Henne-Ei-Problem

Konzepte für anwenderfreundliche E-Mail-Sicherheit

E-Mail-Verschlüsselung könnte längst flächendeckend Verwendung finden. Dieser Artikel erläutert, warum dies nicht der Fall ist – und was dafür getan werden sollte, dies zu ändern. Voraussetzung für eine optimistische Vision ist, wie bei jeder Infrastruktur, das Vorhandensein geeigneter „Straßen“ und ihre langfristige Finanzierung. Dies ist wohl staatlicherseits (bisher) nicht gewünscht.

„People want safe communications, not usable cryptography. For encryption to be widely used, it must be built into attractive, easy-to-use apps like those people already rely on.“ (Justin Troutman, MIT Technology Review) [1]

E-Mail ist als Kommunikationsmedium bekanntermaßen allzu „offen“, und die viel benutzte Analogie zur Postkarte ist noch untertrieben: Es kann nicht nur jeder Transporteur einer normalen E-Mail deren Inhalt mitlesen, sondern Suchmaschinen können automatisiert Profile über viele Milliarden E-Mails täglich erstellen. Der E-Mail-Nutzer ist damit nicht nur zum Empfänger gezielter Werbung, sondern zum gläsernen Menschen geworden – und vermutlich viele Unternehmen bereits Opfer von Wirtschaftsspionage. Einer breiten Nutzung der vorhandenen Verschlüsselungstechnologie (s. Seite xxx in diesem Heft für einen Überblick), die dies verhindern könnte, stehen jedoch bis heute fundamentale Hindernisse im Weg:

- **Konkurrierende Standards**. Durch die gleichzeitige Einführung zweier nicht kompatibler, konkurrierender Standards im Jahr 1998 (S/MIME und OpenPGP [1]) konnte sich keine der beiden Lösungen durchsetzen.
- **Interoperabilitätsprobleme**. Die Sicherheitsstandards wurden nicht vollständig oder korrekt implementiert, und führen zu Kommunikationsproblemen. Besonders problematisch ist die Geräte-übergreifende, mobile Verwendung, die Nutzer heutzutage selbstverständlich erwarten.
- **Mangelhafte Usability**. Sicherheit ist nicht integriert. Initial muss jeder Nutzer für S/MIME ein Schlüsselpaar erzeugen, ein E-Mail-Zertifikat beantragen und installieren, für OpenPGP ist es ähnlich. Überfordert sind die Benutzer aber nicht nur bei der initialen Einrichtung, sondern häufig auch bei der alltäglichen Benutzung.

Bei einer Internetrecherche mit Suchbegriffen wie *„studie e-mail verschlüsselung“* oder *„why no one encrypts e-mail“*, finden sich interessante Zitate, die die Komplexität der Thematik veranschaulichen:

- „Mit 60 % ist eine deutliche Mehrheit der Unternehmen in Deutschland der Meinung, dass sie nicht ausreichend gegen Datendiebstahl, Wirtschaftsspionage oder Sabotageakte geschützt sind.“[2]
- „Es gibt starkes politisches Interesse, die Kommunikation aller Internet-Nutzer jederzeit ohne großen Aufwand mitlesen zu können. … Doch viel schwerer wiegt wohl Googles ureigenes, wirtschaftliches Interesse: Mit funktionierender Ende-zu-Ende-Verschlüsselung würde Google sein eigenes Geschäftsmodell torpedieren. … Richtige und vor allem einfache Ende-zu-Ende-Verschlüsselung für jedermann wäre machbar – wir müssen sie nur endlich einfordern.“[3]
- „Google stellte im Nov. 2015 die Ergebnisse einer Studie vor, nach der die Verschlüsselung von E-Mails große Fortschritte macht. Laut Studie stieg die Zahl an verschlüsselten E-Mails, die von Nicht-Gmail-Accounts stammten und die beim Google-Dienst Gmail eintrafen, von 33% auf 61% an (Zur Klarstellung: Betrachtet wird von Google die Verschlüsselung der Übertragung via TLS, nicht die Ende-zu-Ende-Verschlüsselung der E-Mail an sich, via S/MIME oder PGP).“[4]
- „In practice, using encrypted e-mail is awkward and annoying.“[5]

Darunter leiden keineswegs nur technische Laien, sondern auch IT-Experten, Behörden und Firmen. Bspw. trugen die vom BSI im Sphinx-Projekt geförderten Interoperabilitätstests erst nach etlichen Jahren zu interoperablen S/MIME-Mail-Gateways bei.

Nach einer von Bitkom im Januar 2016 veröffentlichten Studie[6] versenden die meisten Nutzer ihre E-Mails wegen zu hoher Anforderungen weiterhin unverschlüsselt. Ein

[1] http://www.technologyreview.com/view/533456 (12.12.2014)

[2] https://www.bitkom.org/Presse/Presseinformation/Studie-zu-Wirtschaftsschutz-und-Cybercrime.html Im Gegensatz zu Einzelpersonen glauben Firmen laut dieser Studie aus 2015 also sehr wohl, dass sie “etwas zu verbergen” haben.

[3] http://www.heise.de/security/artikel/Warum-Google-uns-echte-Verschluesselung-verweigert-2191797.html

[4] https://www.hornetsecurity.com/de/security-informationen/alles-geschutzt-oder-was-die-realitat-von-e-mail-verschlusselung

[5] http://arstechnica.com/security/2013/06/encrypted-e-mail-how-much-annoyance-will-you-tolerate-to-keep-the-nsa-away/

[6] https://www.bitkom.org/Presse/Presseinformation/Verschluesselung-von-E-Mails-kommt-nur-langsam-voran.html

Viertel der Befragten findet die Verfahren (Tools und Prozesse) grundsätzlich viel zu aufwändig. 64 % gaben als Grund für den Verzicht an, dass sie sich damit nicht auskennen würden. 59 % setzen sie nicht ein, weil ihre Kommunikationspartner keine Verschlüsselung nutzen. Lediglich 15 % hätten 2015 demnach (einen Teil ihrer) Mails verschlüsselt, ein Jahr zuvor waren es mit 14 % ähnlich wenige Nutzer. Einen deutlichen Anstieg gab es laut Bitkom nur kurz nachdem die NSA-Affäre im Juli 2013 ins Rollen kam. Danach sei die Verbreitung nicht mehr wesentlich gestiegen, obwohl sich die Benutzer im Klaren seien, dass sie sich mit verschlüsselter E-Mail vor Missbrauch ihrer persönlichen Daten schützen könnten.

Im Folgenden werden nicht nur die Ursachen näher beleuchtet, weshalb bisher nur wenige Menschen E-Mail-Verschlüsselung verwenden, sondern auch, wie eine Lösung aussehen kann: Die Autoren möchten sich bei den 6 Lehrstühlen, den 5 KMUs und den 6 Beratern bedanken, die für die Projektskizze "Sichere 1-Click-E-Mail" (1CMail) die konkrete Realisierung genau konzipiert und durchdacht haben[7].

1 Welche Anforderungen haben die Nutzer?

Obwohl die meisten verbreiteten Desktop-Mailclients prinzipiell sowohl S/MIME als auch OpenPGP unterstützen, stellt die **Usability** dieser Lösungen, und insbesondere das Zertifikats- und Schlüsselhandling, die Nutzer vor große Probleme. Besonders schlecht sieht es aus, wenn man beide Standards im selben Mail-Frontend nutzen will. Für mobile Mailclients stellt die Interaktionsgestaltung generell eine besondere Herausforderung dar – bedingt durch die stark eingeschränkte Displaygröße und meist nur virtuelle Tastaturen. Dabei unterstützen die vorhandenen Apps ohnehin meist nur *entweder* S/MIME *oder* OpenPGP, und das oft auch nur in eingeschränkter Form. Webmail-Interfaces unterstützen bestenfalls PGP-basierte Ansätze oder proprietäre Insellösungen, zumal eine S/MIME-Unterstützung technisch schwierig zu realisieren ist.

Doch selbst wenn die nötige Verschlüsselungstechnologie verfügbar ist, wird sie selten genutzt: Diverse Studien aus der Forschung im Bereich „Human-Computer Interaction and Security“ (HCISec) deuten auf mangelndes Verständnis der Sicherheitsfunktionen in Mailclients hin. So haben Garfinkel et al. [2] festgestellt, dass nur 34 % der befragten Amazon-Händler, die S/MIME-fähige Clients verwendeten (n=291), überhaupt wussten, dass ihr Client Verschlüsselung beherrscht, obwohl sich die meisten Befragten als „very sophisticated“ oder „comfortable“ bzgl. der Nutzung von Computern und dem Internet bezeichneten. Dies könnte u.a. darauf zurückzuführen sein, dass viele Nutzer die Verantwortung für Sicherheit nicht bei sich selbst, sondern beim IT-Personal sehen [3], obwohl die Bedeutung von „Security Awareness“ vielen IT-Managern bekannt und von Studien untermauert ist [4].

Seit 1999 diverse schwerwiegende Probleme bzgl. der Benutzbarkeit von PGP 5.0 aufgezeigt wurden [5][8], kamen andere zu ähnlich ernüchternden Resultaten bzgl. PGP 9 [6] und aktuellen S/MIME-fähigen Clients [7]. Auch Esslinger [8] berichtet diverse Probleme populärer Mailclients hinsichtlich der Benutzung von S/MIME- und PGP-Funktionen, die eklatante Mängel bzgl. der Erwartungskonformität, Konsistenz, Interoperabilität und Lernförderlichkeit belegen.[9] Neben mangelhaften Benutzungsschnittstellen und der Verwirrung von Nutzern durch inkonsistente Terminologie [7] sorgen auch unklare rechtliche Aspekte und Schwierigkeiten, ein Zertifikat („Digitale ID“) zu erhalten, dafür, dass eine größere Verbreitung von Mail-Verschlüsselung trotz vorhandener Technologie bis heute ausbleibt [9]. Diverse Autoren schlugen daher vor, auf Zertifizierungsstellen (CAs) zu verzichten und auf alternative Modelle wie bspw. selbst-signierte Zertifikate und Key Continuity Management (KCM) zurückzugreifen [3, 10, 11, 12]. Die Nutzer müssten sich bei einem solchen System jedoch selbst darum kümmern, die Authentizität der initial verwendeten Schlüssel zu verifizieren, bspw. durch manuellen Abgleich per Telefon [11], was zusätzlichen Aufwand und eingeschränkte Sicherheit bedeutet. Ein automatisierter Ansatz ohne diese Nachteile stammt von Google.[10]

Generell hat sich unter Experten noch kein klarer Konsens gebildet, ob Nutzer dabei unterstützt werden sollen, die Funktionsweise von Public-Key-Systemen sowie entsprechende Risiken zu verstehen [13, 14] bzw. hilfreiche mentale Modelle zu bilden [15], oder ob Benutzungsschnittstellen möglichst transparent gestaltet und die internen Prozesse vor den Nutzern verborgen werden sollten [4, 16]. Rückfragen insbesondere bei Schülern und auf Crypto-Parties ergaben, dass Benutzer bei E-Mail eigentlich nur Einfachheit und Zuverlässigkeit wollen – und Sicherheit als gegeben erwarten: Sie wollen ihre E-Mail schreiben, den oder die Empfänger eingeben, und dann auf Senden drücken. Dabei soll automatisch der sicherste Weg gewählt werden. Außerdem sollen die Mailclients ebenso wie Betriebssystem und Browser schon beim Kauf eines Computers dabei sein. Wenn Schlüsselmanagement nötig ist, wollen die Benutzer geführt und nicht mit „Fachchinesisch“ konfrontiert werden, andererseits aber auch keine unangenehmen Erfahrungen erleben (z. B. weil Mailclients sie nicht aufforderten, den Schlüsselspeicher per Passwort zu sichern, oder zuließen, den privaten Schlüssel zu löschen). Von Erklärungen zu Zertifikaten oder zu den Unterschieden zwischen Ende-zu-Ende (E2E)- und Transportverschlüsselung waren sie ebenso wenig zu begeistern wie von der Notwendigkeit, an jährliche Zertifikats-Erneuerungen zu denken. Ihre Erwartung war die Gleiche wie beim Telefonieren: Wenn man ein neues Gerät hat oder weitere Gesprächspartner hinzukommen, sollen keine Interoperabilitätsprobleme auftreten. Sichere E-Mail ist momentan jedoch nicht nur *initial* schwer einzurichten (Anfangshürde), sondern man stößt *andauernd* wieder auf

[7] Beteiligt waren neben den Autoren u.a. die Lehrstühle von Prof. Schwenk, Prof Dürmuth, Prof. Kluge (Bochum), Prof. Borges (Saarbrücken), Prof. Dräxler (Kassel), von KMUs u.a. die Herren M. Schober, B. Oswald (novosec), M. Bartosch, Dr. Welter (White Rabbit), E.O. Wilhelm (gft), und als Berater u.a. W. Koch (GnuPG).

[8] http://www.gaudior.net/alma/johnny.pdf

[9] Ein 40-seitiges PDF von Esslinger (2016) mit einer erweiterten Auflistung von Verschlüsselungs-bezogenen „Bugs“ (mit Thunderbird) findet sich unter https://bugzilla.mozilla.org/show_bug.cgi?id=1243449.

[10] https://github.com/google/end-to-end/wiki/Key-Distribution

Probleme, die auch selbst dann mühselig sind, wenn man „aware“ ist und über Expertenwissen verfügt. Hierzu zwei Beispiele:

- Die Signatur einer Nachricht kann *heute* als gültig und *morgen* als ungültig angezeigt werden, nur weil das S/MIME-Zertifikat des Senders ausläuft. Thunderbird zeigt dann im Dialog "Nachrichten-Sicherheit" als Status an: "Kein Zertifikat vorhanden", statt dass es heißt „es ist abgelaufen“. Das widerspricht der Metapher eines Siegels.
- Aus dem gleichen Grund lehnt S/MIME es *heute* ab, an jemand verschlüsselt zu mailen, dem man *gestern* noch verschlüsselt mailen konnte: „Es sind nicht für alle Empfänger Zertifikate vorhanden. Anwendung konnte kein Verschlüsselungszertifikat für … finden.“ Statt den Sender entscheiden zu lassen, ob er nicht auch ein abgelaufenes Zertifikat verwenden will, wird ihm Angst eingejagt, als wäre etwas auf seinem Computer abhandengekommen.

Die Nutzer scheinen also insgesamt durchaus bereit, für die Sicherheit etwas zu tun – aber sie wollen dabei geführt werden und nicht permanent gegen Windmühlenflügel ankämpfen müssen.

2 Bausteine eines Lösungsansatzes

Um eine spürbare Verbesserung der Gesamtsituation zu erzielen und die Nutzung von E-Mail-Verschlüsselung signifikant zu erhöhen, muss ein kohärentes Gesamtkonzept verfolgt werden, das das komplexe Zusammenspiel sozialer, psychologischer, technischer und politischer Faktoren angemessen berücksichtigt. Hierzu sind diverse Detailprobleme zu lösen.

2.1 Rechtliche und psychologische Aspekte

Bis heute ist umstritten, ob und unter welchen Voraussetzungen E-Mail als Kommunikation bei vertraulichen Informationen, bspw. im Verkehr mit Behörden, Gerichten, Krankenkassen, Steuerberatern oder Personalabteilungen eingesetzt werden kann und darf. Eine gesetzlich verankerte Gewissheit, dass verschlüsselte und signierte Mails – nicht jedoch „normale“ E-Mails! – ein ebenso hohes Gewicht haben wie ein händisch unterschriebener Brief, würde zweifellos einen wirkungsvollen Motivator für die Nutzung der erforderlichen Krypto-Technologie darstellen.

Psychologische Erkenntnisse zur Technologieakzeptanz und Interaktionsgestaltung können darüber hinaus helfen, sichere E-Mail-Technologie so zu bauen, dass sie gut benutzbar ist und Menschen darin einen persönlichen Nutzen erkennen. So wurden bereits einige allgemeine Richtlinien für die Gestaltung der Mensch-Computer-Interaktion im Security-Bereich identifiziert [17, 18] und unterschiedliche Ansätze zur Identifikation geeigneter Metaphern für die Verschlüsselung und Signierung untersucht [10, 16, 17]. Soweit uns bekannt ist, wurden diese Ansätze jedoch bestenfalls prototypisch implementiert und sind bisher nicht in den realen Praxiseinsatz überführt worden.

2.2 Zwei-Faktor-Authentifizierung (2FA)

Der Zugriff auf private Schlüssel für E-Mail-Verschlüsselung ist üblicherweise durch „Wissen“ seitens des Benutzers in Form eines Passworts geschützt. Bis Cyborg-Technologie dem Menschen der Zukunft möglicherweise ein stabiles, präzises und unbeirrbares Gedächtnis mit der erforderlichen Kapazität beschert, bleibt die Verwendung einzigartiger, sicherer Passwörter in der Praxis jedoch ein ernstzunehmendes Problem. Hardware und Biometrie bieten die zusätzlichen Faktoren „Haben“ und „Sein“:

Hardware

Benutzer haben die letzten 15 Jahre im privaten Bereich kein zusätzliches Gerät zur Authentisierung akzeptiert. In manchen Firmen (bspw. Siemens, Boeing, DB) werden Smartcards genutzt, die auch E-Mail-Schlüssel enthalten – aber nur am PC. Ansatz für eine Nutzung auf unterschiedlichen Geräten sind virtuelle Smartcards. Windows 10 bringt eine eigene virtuelle Smartcard mit, letztlich eine Kapselung des Crypto-Layers, die nach oben so aussieht wie eine Smartcard. Das könnte interessant sein, da es ohne externe Infrastruktur oder weitere Software verfügbar ist.

Webmailer bieten zumindest zum Teil Unterstützung für Tokens: Google unterstützt 2FA, indem der Benutzer seinen Google Authenticator (oder eine andere RFC 6238 kompatible App) zur Anmeldung konfigurieren kann, was den Google-Mail-Account einschließt[11]. Posteo.de (ein kleinerer deutscher Anbieter mit besonderem Fokus auf Sicherheit) bietet 2FA mit Yubikey[12]. Auch Mailbox.org (ein weiterer deutscher Anbieter) unterstützt Yubikey[13]. Das größte Problem physischer Tokens ist die Connectivity, also wie spricht das Token mit dem PC/Smartphone/etc. Yubikeys geben sich dem Host-Computer gegenüber als USB-Tastatur aus und funktionieren daher ohne jegliche Treiberinstallation. Da die meisten Smartphones keinen USB-Port haben, unterstützen neuere Yubikey-Versionen auch die Kommunikation via NFC.

Biometrie

Der Zugriff auf den Keystore kann rein biometrisch oder durch die Kombination aus Passwort und Biometrie gesichert werden. Letztendlich gibt eine Softwarekomponente nach der biometrischen Identifizierung „grünes Licht“ zum Entsperren des Keystores. Die Schlüssel darin liegen also nur mäßig geschützt im Keystore. Biometrie macht nur Sinn, wenn man ein vertrauenswürdiges Lesegerät hat.

Erschwert oder erleichtert 2FA also die Nutzung von verschlüsselter E-Mail? Alles, was Benutzern das Merken von langen Passworten vereinfacht oder abnimmt, ist prinzipiell ein Gewinn. Leisten könnten das am ehesten physikalische Smartcards oder Tokens. Aber damit hat man dann wieder andere Probleme wie Treiberproblematik, Backups, Verlust des Tokens, temporärer Wechsel von Arbeitsplätzen. Alternativ kann der zweite Faktor auch wie bei einer TAN dynamisch generiertes Wissen sein. Darauf

[11] https://www.google.com/landing/2step/ und https://developers.google.com/gmail/xoauth2_protocol
[12] https://posteo.de/site/leistungen#leistungendatenschutz
[13] support.mailbox.org/knowledge-base/article/webmail-mit-one-time-passwords

laufen im Moment die meisten Webanwendungen wie bspw. von Gmail, gmx.net, Mailbox.org und Posteo hinaus.[14]

2.3 Baustellen bzgl. Webmail-Interfaces

Mit dem Aufkommen von HTML5, AJAX und immer reichhaltigerer JavaScript (JS)-Bibliotheken wurde es möglich, viele Funktionen, die früher Desktop-Anwendungen vorbehalten waren, in moderne Webbrowser zu integrieren, bspw. von Tutao[15] und Adesso. **OpenPGP im Browser**: Ein aktives Open-Source-Projekt implementiert OpenPGP in JavaScript[16]. Die Sicherheit dieser Implementierung wurde bereits untersucht[17]. Die Bibliothek openpgp.js bildet den Ausgangspunkt für die Integration des OpenPGP-Standards in viele Webmail-Anwendungen.

Der OpenPGP-Standard ist seit Jahren unverändert. Inzwischen gibt es Bestrebungen, auch damit Metadaten (wie Subject, Name des Dateianhangs) zu verschlüsseln[18].

Leider gibt es bei allen untersuchten Browser-Implementierungen derzeit noch offene Probleme:

Speicherung der privaten Schlüssel eines Nutzers

Die Speicherung muss sowohl im Browser als auch auf dem externen privaten Active Wallet Service (siehe Kapitel 2.5 Erweiterung der PKI-Infrastruktur) möglich sein.

Zuverlässige Zufallszahlengeneratoren im Browser

Hier ist zu untersuchen, ob ggf. der <KeyGen>-Tag für eine Lösung herangezogen werden kann.

Erkennung von XSS und Schadsoftware

Zur Erkennung von Schadsoftware gibt es das GitHub-Projekt DOMPurify. Der Autor von DOMPurify hat diese Problematik schon in den Projekten MAKE (NRW Ziel 2) und JSAgents (BMBF, Schwerpunkt XSS) untersucht.

S/MIME in JavaScript

Für S/MIME gibt es momentan keine komplette JavaScript-Implementierung; die Kryptobibliothek Cryptojs[19] unterstützt zumindest PKCS#7.

Ähnlich wie bei mobilen und Desktop-Mailclients kann auch bei Webmail ein **Offline-Betrieb** ermöglicht werden. Nach Möglichkeit soll eine **Suchfunktion auf verschlüsselten Mails** eingeführt werden, z.B. durch Erstellung eines (verschlüsselten) Index nach erstmaliger Entschlüsselung. Da eine Webanwendung von ihrer Natur her schnell aktualisiert werden kann, kann auch die Einbindung **neuer Verschlüsselungsalgorithmen** erprobt werden, insbesondere Post-Quanten-Kryptographie.

2.4 Mobile Mailclients

Der unaufhaltsame Trend zur mobilen Nutzung von E-Mail-Diensten macht es unumgänglich, dass Mail-Verschlüsselung nicht nur am Desktop-Rechner, sondern auch auf Smartphones, Tablets und in absehbarer Zukunft wohl auch auf intelligenten Brillen verwendbar sein muss. Anwendungen auf mobilen Geräten haben aufgrund der eingeschränkten Benutzeroberfläche besondere Anforderungen an die Benutzerführung und die Usability des Systems, beispielsweise eine stark verkleinerte Ansicht und in aller Regel nur eine emulierte Tastatur. Gleichzeitig ist auch das typische Nutzungsverhalten verschieden: Häufig werden E-Mails auf mobilen Geräten lediglich gelesen, aber nicht geschrieben. Dies vereinfacht das parallele Implementieren von S/MIME und PGP.

Zentrales Problem ist aber die Verteilung und Synchronisierung von Zertifikaten und Schlüsseln über verschiedene Geräte hinweg (üblicherweise zwischen einem primär genutzten Desktop- oder Webmail-Client und dem sekundär genutzten mobilen Gerät). Besonders mobile Geräte profitieren daher von der im folgenden Absatz erläuterten, verbesserten und automatisierten Infrastruktur.

2.5 Erweiterung der PKI-Infrastruktur

Aktuelle PKIs sind auf die Beantragung eines Zertifikats für ein einzelnes Gerät ausgelegt, heutige Nutzer besitzen aber eine Vielzahl von Geräten. Durch einen neu zu entwickelnden Active Wallet Service (**AWS**) könnten Nutzer bei der Installation von Zertifikaten auf mehreren Geräten unterstützt werden. Außerdem kann so eine dauerhafte Ablage für Schlüsselpaare geschaffen werden, um auch archivierte verschlüsselte Mails jederzeit lesen zu können. Dies zu implementieren ist kein Neuland, denn in größeren Unternehmen werden PKI-Prozesse so hochgradig automatisiert, abgesichert und in alle Mailclients eingebunden.

Nötig ist dazu die Bereitstellung des PKI-Backend-Systems inkl. der Distribution der Zertifikate. Außerdem werden die Benutzer-Prozesse (Workflows) für Beantragung, Erneuerung und Sicherung/Wiederherstellung von Zertifikaten und Schlüsseln modelliert und das Management der CA-Schlüssel bzw. die Anbindung an externe CAs durchgeführt. Würde dies gefördert, stehen sowohl deutsche PKIs (bspw. Universitäts-Rechenzentren) zur Verfügung als auch die Anbindung an weitere Trustcenter (wie D-Trust, Startcom).

Hier einige Details zu den technischen Schnittstellen aus der Architektur-Konzeption: Das PKI-Backend von OpenXPKI[20] unterstützt nicht nur Hardware-Security-Module (HSMs), sondern hat vor allem eine Anbindung an externe CA-Dienste (und deren öffentliche Zertifikate). Für mobile und Desktop-Mailclients können die Workflows mit Standard-Protokollen wie SCEP umgesetzt werden. Zertifikate könnten über ein öffentliches Web-Interface (WebUI und REST API) sowie in einem LDAP-Directory bereitgestellt werden. Mit einem benutzerspezifischen AWS können private Schlüssel zwischen verschiedenen Geräten des Benutzers sicher verteilt werden. Zusätzlich kann der AWS auch nur den Sessionkey einer einzigen Mail auf Anforderung an das Mail-Frontend senden, so dass auf dem Frontend nie die privaten Schlüssel liegen müssen. Die notwendigen Funktionen können sowohl in Software als auch in Hardware (ggf. auf einem Embedded Device wie dem RaspberryPi) bereitgestellt werden.

[14] www.zeit.de/digital/datenschutz/2015-07/urlaub-internetcafe-mail-facebook
[15] www.bestvpn.com/blog/16671/tutanota-private-email-review-vs-protonmail/
[16] http://openpgpjs.org/, https://github.com/openpgpjs/openpgpjs Die kommende Version 2.0 wird auf bessere Performance und native crypto der darunter liegenden Plattform setzen.
[17] https://cure53.de/pentest-report_openpgpjs.pdf
[18] https://www.gnupg.org/blog/20150426-openpgp-summit.html
[19] https://github.com/gwjjeff/cryptojs
[20] http://www.openxpki.org/

2.6 Politik und staatliche Förderung

Im Rahmen des nationalen IT-Gipfels[21] wurde Ende 2015 in Berlin die "Charta zur Stärkung der vertrauenswürdigen Kommunikation" vorgestellt und unter anderem von Bundesinnenminister Thomas De Maizière unterzeichnet. Wie wird diese Charta gelebt?

Widersprüchliches vom Staat

Etliche Betrachter merken an, dass sich unser Staat in dieser Sache sehr widersprüchlich verhält. Selbst Mitarbeiter des BMI sprachen davon, dass ihr Amt zwei Seelen in seiner Brust habe:

- Einerseits werden Bürger (die sich über die „Datensammel-Industrie“ empörten) aufgefordert zu verschlüsseln, und das BSI finanziert dazu auch die Erstellung hervorragender Tools wie Gpg4win.
- Andererseits will das Innenministerium ein Recht auf die Schlüssel[22], aber neuerdings ohne Backdoors[23].
- Konkrete Vorschläge und Anforderungen kamen aber auch von „allgemein anerkannten“ Organisationen (NGOs) wie der Gesellschaft für Informatik[24]. Die von der GI unter dem Titel „Informatiker fordern erneut sichere und einfach anwendbare Verschlüsselungsverfahren für E-Mails“ zusammengefassten Anforderungen entsprechen großenteils den Evaluierungen im Projekt 1CMail.

Die o.g. Charta formuliert: "Wir stärken vertrauenswürdige Kommunikation insbesondere durch Ende-zu-Ende-Verschlüsselung". Ziel sei es, "Verschlüsselungs-Standort Nr. 1 auf der Welt" zu werden. Die Unterzeichner bekennen sich in der Charta unter anderem zur Förderung der Nutzerfreundlichkeit von Verschlüsselung, zur Technologieneutralität, zur Transparenz bezüglich der eingesetzten Verfahren sowie zur Weiterentwicklung von Verfahren zur Ende-zu-Ende-Verschlüsselung. Wie die Teilnehmer der entsprechenden Arbeitsgruppe ausgesucht wurden ist nicht offengelegt. Den Beteiligten war es nicht möglich, sich auf technische Prozeduren und klare Finanzierungen festzulegen. Stattdessen wurde „als erster gemeinsamer Schritt“ wieder einmal Awareness vorgeschlagen. Material zur Awareness und zur Benutzung im Internet gibt es aber schon reichlich.[25]

Das Wirtschaftsministerium warnt vor Wirtschaftsspionage und erklärt: „IT-Sicherheit ist zentraler Wirtschaftsfaktor“[26]. Andererseits findet sich das Wort „Verschlüsselung“ nicht auf dieser Seite – nur im Kampagnenmotiv „Architekt. Chef. Datenverschlüssler.“ und auf der Seite http://www.it-sicherheit-in-der-wirtschaft.de/.

Ein grundsätzliches Dilemma

Wie bei allen Infrastrukturen hat man asymmetrische Verhältnisse und die Frage, was man zuerst braucht („Henne-Ei“-Problematik): Wer die erste öffentliche Straße baut oder das erste Telefon kauft, hat wenig davon. Je später jemand einsteigt, desto geringer sind sein Risiko und sein Aufwand. Aber ohne die Erstinvestition in die Straße (PKI, Schlüsselmanagement, gute Client-SW) wird es auch keine guten Autos geben und keine Menschen, die mit den vorhandenen Tools ihre Privatsphäre schützen können.

Staatliche Förderung

In der Ausschreibung[27] „zur Förderung von Forschungsinitiativen auf dem Gebiet des Selbstdatenschutzes“ steht unter anderem: "Vertraulichkeit unterstützen: Um im Internet vertraulich kommunizieren zu können und sensible Informationen zu schützen, können Verschlüsselungsverfahren eingesetzt werden. Deren wirksame Anwendung erfordert u.a. die Aushandlung kryptographischer Schlüssel sowie die Anbindung an Vertrauensinfrastrukturen. Es sollen daher alltagstaugliche und breit einsetzbare Verfahren zur Unterstützung einer vertraulichen Kommunikation und Datenhaltung entwickelt werden, welche auch für Laien nutzbar sind."

Will man etwas Nachhaltiges und wirklich breit Verfügbares, das interoperabel, nahezu selbsterklärend und mit einem Klick zu installieren ist, dann muss staatliche Förderung hier weg von kleinen Projekten, Verteilungsschlüsseln und Prototypen (die oft gerade so lange laufen, bis die Papers geschrieben und die Drittmittel aufgebraucht sind), und hin zu einer nachhaltigen Förderung für eine Infrastruktur und eine Produktsuite (BMBF gemeinsam mit BMWi und BMI), die bis zu ihrer Fertigstellung in zwei Jahren ca. 10 Millionen Euro kostet, und anschließend jährlich für Wartung und Support jeweils weitere 1-2 Millionen[28]. Sind das dem Staat die Sicherheit vor massenhaftem, anlasslosem Lauschen und der Schutz der Privatsphäre wert? Ein BSI ohne politische Vorgaben und eine Nachverfolgung der bisher geförderten Projekte könnte hier zu besseren staatlichen Investitionen führen.

Ohne diesen großen Wurf wird es bei vielen kleinen, gutgemeinten Anstrengungen bleiben.

3 Wir schaffen das – es ist wirklich umsetzbar

Ein übergreifender Lösungsansatz betrachtet alle gängigen Mailclients und Verschlüsselungsstandards. Eine konsistente und leicht nutzbare Lösung wäre Open-Source-basiert und für Endanwender kostenlos. Nur so ergäbe sich eine spürbare Auswirkung auf die Sicherheit aller Bürger. Abb. 1 zeigt die Architektur eines ausgearbeiteten, aber staatlich (noch) nicht geförderten Gesamtansatzes[29].

<Bildtext_Abb-1: Architektur des Gesamtansatzes >

---

[21] http://www.bmwi.de/DE/Themen/Digitale-Welt/nationaler-it-gipfel.html

[22] http://www.golem.de/news/crypto-wars-2-0-de-maizi-re-und-eu-wollen-recht-auf-entschluesselung-1501-111841.html (21.1.2015)

[23] http://www.golem.de/news/nach-terroranschlaegen-de-maizi-re-will-keine-backdoors-in-kryptographie-1511-117530.html (19.11.2015)

[24] http://www.gi.de/index.php?id=3740&tx_ttnews[tt_news]=1821 (1.4.2015)

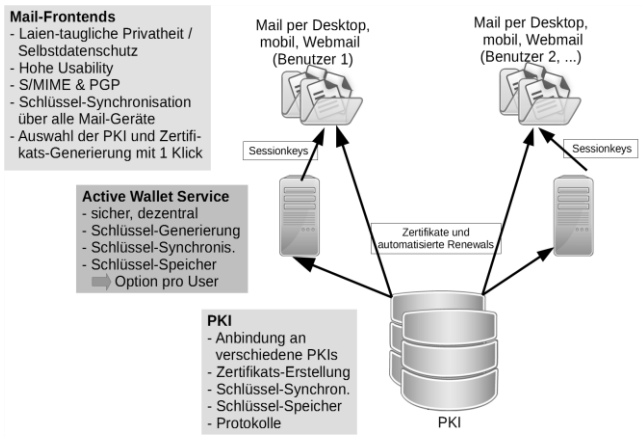
[25] https://www.verbraucher-sicher-online.de/thema/e-mail-verschluesselung, http://www.selbstdatenschutz.info/e-mail_verschluesseln, https://E-Mailselfdefense.fsf.org/de/index.html, http://www.german-privacy-fund.de/e-mails-verschlusseln-leicht-gemacht/, http://futurezone.at/digital-life/wie-man-e-mails-verschluesselt/24.598.005, https://support.mozilla.org/de/kb/nachrichten-digital-signieren-und-verschlusseln sind nur wenige Beispiele. Ein sehr verständlicher Vortrag von Prof. Wacker samt Folien und Zuhörerfragen finden sich auf https://www.uni-kassel.de/eecs/fachgebiete/ais/wissenstransfer/sichere-e-mail.html.

[26] http://www.bmwi.de/DE/Themen/Digitale-Welt/it-sicherheit.html

[27] http://www.bmbf.de/foerderungen/25038.php

[28] Dies entspricht ca. 0,0005 Promille des Bruttoinlandsprodukts.

[29] Insbesondere im PGP-Umfeld gibt es eine steigende Anzahl von Projekten, die moderne Software-Architekturen vergleichend untersuchen und auch probieren, an der nächsten Generation sicherer E-Mail oder E-Mail-ähnlicher Kommunikation zu arbeiten. Siehe bspw. die Übersicht: https://leap.se/en/docs/tech/secure-E-Mail.



Exemplarisch wird hier erläutert, wie die konkrete Umsetzung erfolgen könnte: Im Gegensatz zu derzeit existierenden oder mit „Volksverschlüsselung“[30] und Mailvelope[31, 32] in der Einführung begriffenen Lösungen wird eine Anwender-freundliche Lösung beide Verschlüsselungsstandards S/MIME und OpenPGP gemeinsam unterstützen, diese automatisch erkennen, die Benutzung unterschiedlicher Geräte unterstützen, und den Benutzer nicht mit technischen Fragen überfordern. Das Problem besteht hier nicht in der grundlegenden Realisierbarkeit, sondern in der nahtlosen Integration beider Standards und der Schnittstellen zur PKI:

Verbesserung vorhandener Lösungen: Zunächst wird für den Anwender innerhalb der Anwendung Transparenz geschaffen, bspw. in Form einer „Ampel“-Anzeige ähnlich der im „Calomel“-SSL-Plugin für Firefox[33]. Außerdem werden Fehlermeldungen im Mail-Frontend für den Benutzer verständlicher gestaltet. Ziel ist es, dass der Benutzer nicht nur einen Hinweis erhält, dass ein Fehler aufgetreten ist, sondern auch erkennt, welche Art von Fehler und warum dieser aufgetreten ist, und dass er einen konkreten Vorschlag zur Behebung erhält. So können beispielsweise die eingesetzten Kryptoalgorithmen einer E-Mail angezeigt werden. Für Experten wird zusätzlich genau dargestellt, warum bspw. die Signatur einer E-Mail als „ungültig“ gilt.

Implementierung neuer Sicherheits-Funktionen: Bisher konnten E-Mail-Zertifikate für S/MIME nur „extern“, also z.B. mit einem Web-Browser, erstellt/beantragt werden. Die Zertifikatserstellung kann aber genauso gut direkt im Mailclient erfolgt, der sich auch um eine automatische Verlängerung ablaufender Zertifikate kümmert. Dies stellt eine deutliche Verbesserung der Usability dar und beseitigt existierende hohe Einstiegshürden. Verbesserungen hier können bspw. in die offizielle Thunderbird-Version einfließen, um eine flächendeckende Verbreitung sicherzustellen.

Die konzipierten Features könnten bestehende Standards nutzen und mit den bisher verbreiteten Mail-Frontends interoperabel gemacht werden. Nur die Forderung nach einer Token-basierten 2FA lässt sich nicht Geräteübergreifend realisieren.

Etliche Ansätze wurden angekündigt oder sind auch schon wieder am Verschwinden (DarkMail, DIME, LEAP, PIXELATED, DENAME), andere beginnen gerade mit einer breiten, über E-Mail und die oben genannten Anforderungen hinausgehenden Unterstützung der Privatsphäre (insbesondere das PEP-Projekt[34]).

Juristisch interessant ist die Frage, ob Firmen und Behörden überhaupt unverschlüsselte E-Mails versenden dürfen. Für bestimmte Berufsgruppen wie Steuerberater ist dies theoretisch heute schon untersagt. Eine leicht nutzbare Lösung könnte hier helfen, noch bestehende praktische Hürden zu überwinden.

Aus einer psychologischen Perspektive ist es relevant, eine niedrigschwellige Möglichkeit zu schaffen, die das Verschlüsseln so einfach wie das Mailen selbst macht.

Wie Marktstudien wiederholt gezeigt haben (u.a. in den BMBF-Projekten Sec2 und dem BMWi-Projekt SkIdentity) ist Sicherheit als nichtfunktionale Eigenschaft eines Systems bei Privatkunden nicht vermarktbar: Die Nutzer erwarten Sicherheit, sind aber nicht bereit, dafür zu zahlen.[35] Daher besteht bei einem Projekt, das dazu dienen soll, allen Bürgerinnen und Bürgern die Nutzung verschlüsselter E-Mail zu ermöglichen, ein Zuwendungsbedarf.

Die bisherigen Untersuchungen zeigten, dass die komplette Abdeckung aller relevanten Clients und Sicherheitsstandards möglich ist, und wie Usability, **Privacy-by-Design** und moderne Schlüsselverteilung lösbar sind.

Es gibt schon viele bewährte Einzelkomponenten, mit denen etwas Ganzheitliches gebaut werden kann – kostenlos für die Benutzer und frei von Sonderlocken, deren Hauptzweck es ist, Konkurrenten vom Markt fernzuhalten. Aber bisher scheint dies politisch nicht gewollt.

---

[30] „Volksverschlüsselung“ (VVS) ist ein gemeinsames Projekt von Fraunhofer und Telekom, das ab Mitte 2016 kostenlose S/MIME-Zertifikate zur Verfügung stellt. Die Rollenaufteilung ist aktuell so, dass Fraunhofer die Nutzer-Software erstellt und die Telekom Hardware und Betrieb der PKI übernimmt. Die PKI hat keine Abhängigkeit zur Fraunhofer-PKI und ihre CA self-signed. Die Nutzer-Software ist erstmal nur ein Outlook-Plugin für S/MIME. Ein automatisiertes Renewal der Email-Zertifikate ist ebenso geplant wie ein Konfigurationstool, das die Mailclients Outlook und Thunderbird, aber nur für Windows unterstützt, und Zertifikate in die Browser Firefox, Internet Explorer und Chrome integrieren kann, um dort eine Client-seitige TLS-Authentifizierung vorzunehmen.

[31] https://www.mailvelope.com PGP- und Browser-basierte Lösung, die u.a. bei GMX und web.de von knapp 200.000 1&1-Kunden aktiviert wurde. Dabei werden PGP-Mails per Default auch signiert. Mailvelope generiert die PGP-Schlüssel im Browser und sichert sie symmetrisch verschlüsselt in dem eigenen lokalen Repository (und auf Wunsch auch in der 1&1-Cloud). Beim Cloud-Backup wird ein generiertes 26-stelliges Passwort verwendet und damit ein deutlich höheres Sicherheitsniveau erreicht im Vergleich zu Benutzerpasswörtern, die angreifbar sind. Mit der kommenden API und einer tieferen (Editor-)Integration wird die Nutzbarkeit entscheidend verbessert. Aktuell hat Mailvelope je ca. 150.000 Weekly Users in Chrome und in Firefox.

[32] Mailvelope basiert ebenfalls auf der oben erwähnten openpgp.js.

[33] https://calomel.org/firefox_ssl_validation.html

[34] Am viel Versprechendsten erscheint das PEP-Projekt (Pretty Easy Privacy), das eng mit Ggp4win und Enigmail zusammen arbeitet und Open-Source ist, aber auch kommerziellen Support bietet. Es nutzt neben OpenPGP auch OTR, und zum Keymanagement statt einer zentralen Infrastruktur ein Peer-to-Peer-Design. Besonders hervorzuheben sind die automatisierte, abgestufte Vorgehensweise, die für jede Kommunikation immer die maximal mögliche Sicherheit zu erreichen versucht, und die folgenden vier weiteren Design-Prinzipien für Privacy-by-Default: Ease-of-use bei Installation, Konfiguration und Integration; Verständlichkeit aller Status; Interoperabilität verschiedener Messaging-Systeme; Plattform-Unabhängigkeit. Ergänzt wird diese Transparenz für User um Parametrisierungsmöglichkeiten für Experten und das Bekenntnis zur Unterstützung aller Anwendungen, die die Nutzer nutzen wollen Siehe https://pep.foundation/docs/pEp-whitepaper.pdf
Siehe auch https://netzpolitik.org/2014/pretty-easy-privacy-whatsapp-verschluesseln-und-facebook-nachrichten-auch-mit-outlook/

[35] Vergleiche auch den sehr interessanten Blogeintrag vom 1.12.2015 zum Ende von whiteout.io nach drei Jahren, das kostenpflichtig sichere E-Mail für alle Geräte über die Cloud anbot: https://tankredhase.com/2015/12/01/whiteout-post-mortem/

Literatur

[1] S/MIME und OpenPGP: RFC 3851 (Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification), RFC 5751 (Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification); RFC 3156 (MIME Security with OpenPGP)

[2] Garfinkel, Simson L.; Margrave, David; Schiller, Jeffrey I.; Nordlander, Erik; Miller, Robert C. (2005): How to make secure E-Mail easier to use. In: Gerrit van der Veer und Carolyn Gale (Hg.): CHI 2005. Portland, Oregon, USA, S. 701–710

[3] Gross, Joshua B.; Rosson, Mary Beth (2007): Looking for Trouble: Understanding End-user Security Management. In: Proceedings of the 2007 Symposium on Computer Human Interaction for the Management of Information Technology. New York, NY, USA: ACM (CHIMIT '07). Online verfügbar unter http://doi.acm.org/10.1145/1234772.1234786

[4] Straub, Tobias (2006): Usability Challenges of PKI. Dissertation. TU Darmstadt

[5] Whitten, Alma; Tygar, J. Doug (1999): Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. In: Proceedings of the 8th USENIX Security Symposium

[6] Sheng, Steve; Broderick, Levi; Koranda, Colleen Alison; Hyland, Jeremy J. (2006): Why Johnny still can't encrypt: evaluating the usability of E-Mail encryption software. In: Symposium On Usable Privacy and Security

[7] Fry, Ann; Chiasson, Sonia; Somayaji, Anil (2012): Not sealed but delivered: The (un) usability of s/mime today. In: Annual Symposium on Information Assurance and Secure Knowledge Management (ASIA'12), Albany, NY

[8] Esslinger, Bernhard (2014): Sichere E-Mail mit S/MIME. In: Datenschutz und Datensicherheit - DuD 38 (5), S. 305–313. DOI: 10.1007/s11623-014-0116-7

[9] DeWitt, Alexander John Anthony George (2007): Usability Issues with Security of Electronic Mail. Dissertation. Brunel University

[10] Roth, Volker; Straub, Tobias; Richter, Kai (2005): Security and usability engineering with particular attention to electronic mail. In: International Journal of Human-Computer Studies 63 (1-2), S. 51–73. DOI: 10.1016/j.ijhcs.2005.04.015

[11] Kapadia, Apu (2007): A Case (Study) For Usability in Secure E-Mail Communication. In: IEEE Security & Privacy Mag. 5 (2), S. 80–84. DOI: 10.1109/MSP.2007.25

[12] Farrell, Stephen (2009): Why Don't We Encrypt Our E-Mail? In: IEEE Internet Computing 13 (1), S. 82–85. DOI: 10.1109/MIC.2009.25

[13] Whitten, Alma (2004): Making security usable. Carnegie Mellon University

[14] West, Ryan (2008): The Psychology of Security. In: Commun. ACM 51 (4), S. 34–40. DOI: 10.1145/1330311.1330320

[15] Wash, Rick; Rader, Emilee (2011): Influencing Mental Models of Security: A Research Agenda. In: Proceedings of the 2011 Workshop on New Security Paradigms Workshop. New York, NY, USA: ACM (NSPW '11), S. 57–66. Online verfügbar unter http://doi.acm.org/10.1145/2073276.2073283

[16] Balfanz, Dirk; Durfee, Glenn; Grinter, Rebecca E.; Smetters, D. K. (2004): In Search of Usable Security: Five Lessons from the Field. In: IEEE Security and Privacy 2 (5), S. 19–24. DOI: 10.1109/MSP.2004.71

[17] Johnston, J.; Eloff, J. H. P.; Labuschagne, L. (2003): Features: Security and Human Computer Interfaces. In: Computers and Security 22 (8), S. 675–684. DOI: 10.1016/S0167-4048(03)00006-3

[18] Nurse, Jason R. C.; Creese, Sadie; Goldsmith, Michael; Lamberts, Koen: Guidelines for usable cybersecurity: Past and present. In: 2011 International Workshop on Cyberspace Safety and Security (CSS). Milan, Italy, S. 21–26