Maximilian Eibl, Martin Gaedke. (Hrsg.): INFORMATIK 2017,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 737

Verfahren zur vertrauenswürdigen Verteilung von Verschlüsselungsschlüsseln

Peter Fischer[1] Thomas Kunz[2] Katharina Lorenz[3] Ulrich Waldmann[4]

Abstract: Ein Grund für die geringe Verbreitung der E-Mail-Verschlüsselung liegt darin, dass viele Nutzer, die ihre E-Mails verschlüsseln möchten, nicht wissen, wie sie evtl. existierende Verschlüsselungsschlüssel der gewünschten Empfänger finden und einen gefundenen Schlüssel vor seiner Nutzung auf Herkunftsechtheit und Aktualität überprüfen können. Dieser Beitrag beschreibt ein Verfahren der Schlüsselverteilung auf Grundlage von „DNS Security Extensions" (DNSSEC). Das Verfahren ermöglicht Nutzern einen einfachen Zugang zu vertrauenswürdigen Schlüsseln. Dazu veröffentlichen die Nutzer ihre öffentlichen Schlüssel auf einem Schlüsselverzeichnis ihres E-Mail-Anbieters. Die Betreiber des zugehörigen DNS-Servers stellen gesicherte Informationen über das jeweilige Schlüsselverzeichnis zur Verfügung. Mittels einer Erweiterung der E-Mail-Anwendung wird der für eine E-Mail-Kommunikation benötigte öffentliche Schlüssel eines Kommunikationspartners automatisch ermittelt. Dazu ist ausschließlich die Angabe der E-Mail-Adresse des Empfängers notwendig.[5]

Keywords: E-Mail, Verschlüsselung, DNS, DNSSEC, OpenPGP, S/MIME, HKP, LDAP

1 Bedarf einer sicheren E-Mail-Kommunikation

Der Wunsch nach einer sicheren E-Mail-Kommunikation ist in der Bevölkerung nicht zuletzt aufgrund der Enthüllungen von Edward Snowden über die weltweite Massenüberwachung der Onlinekommunikation durch die Geheimdienste in den vergangenen Jahren stark gestiegen. Viele Bürger fragen sich, wie sie sich vor dieser Massenüberwachung schützen können und auf vertrauliche Weise über das Internet kommunizieren können [HSW16a, HSW16b].

Das Prinzip der Ende-zu-Ende-Verschlüsselung garantiert einen wirksamen Schutz gegen die Massenüberwachung. Bei der Ende-zu-Ende-Verschlüsselung wird eine Nachricht auf dem Rechner des Senders verschlüsselt und kann nur vom Empfänger auf dessen Rechner entschlüsselt werden. Insbesondere können keine weiteren Instanzen wie beispielsweise

[1] Heinlein Support GmbH, Schwedter Straße 8/9A, 10119 Berlin, p.fischer@heinlein-support.de
[2] Fraunhofer SIT, Rheinstr. 75, 64295 Darmstadt, thomas.kunz@sit.fraunhofer.de
[3] Design Research Lab, Universität der Künste, Einsteinufer 43, 10587 Berlin, k.lorenz@udk-berlin.de
[4] Fraunhofer SIT, Rheinstr. 75, 64295 Darmstadt, ulrich.waldmann@sit.fraunhofer.de
[5] Dieser Beitrag entstand im Rahmen des Projekts „Vertrauenswürdige Verteilung von Verschlüsselungsschlüsseln (VVV)" (`https://keys4all.de`), das vom Bundesministerium für Bildung und Forschung (BMBF) auf Grundlage des Forschungsrahmenprogramms der Bundesregierung zur IT-Sicherheit „Selbstbestimmt und sicher in der digitalen Welt" unter dem Förderkennzeichen 16KIS0354K gefördert wird.

doi:10.18420/in2017_73

E-Mail-Server die Nachrichten entschlüsseln. Die einer Ende-zu-Ende-Verschlüsselung zugrunde liegenden Verfahren sind seit vielen Jahren etabliert und in gängigen E-Mail-Anwendungen integriert. Sie beruhen auf asymmetrischen kryptografischen Verfahren wie RSA, bei denen jeder Kommunikationspartner über ein Schlüsselpaar, bestehend aus einem öffentlichen Schlüssel und einem privaten Schlüssel, verfügt. Der öffentliche Schlüssel darf veröffentlicht werden und dient der Verschlüsselung einer Nachricht. Die verschlüsselte Nachricht kann ausschließlich mit dem dazugehörigen privaten Schlüssel wieder entschlüsselt werden. Daher darf der private Schlüssel nur seinem Besitzer bekannt sein und Dritten nicht zugänglich sein.

Obwohl die Möglichkeit der Ende-zu-Ende-Verschlüsselung in vielen E-Mail-Anwendungen integriert ist, ist deren Ausgestaltung nur wenig benutzungsfreundlich. Viele Nutzer wissen nicht, wie sie das dazu notwendige Schlüsselpaar erzeugen und dieses in eine E-Mail-Anwendung einbinden können. Hinzu kommt das Problem der Schlüsselverteilung: Der Sender einer E-Mail benötigt den öffentlichen Schlüssel des Empfängers. Typischerweise werden die öffentlichen Schlüssel in sogenannten Verzeichnisdiensten veröffentlicht. Allerdings lässt sich in einigen E-Mail-Anwendungen nur ein einziger Verzeichnisdienst konfigurieren. Doch welcher ist der richtige Verzeichnisdienst, in dem der Schlüssel genau dieses Empfängers zu finden ist? Verzeichnisdienste stellen Insellösungen dar, die nicht miteinander verbunden sind. Für den Nutzer ist es schwer zu entscheiden, ob der Empfänger einen Schlüssel besitzt und von welchem Verzeichnisdienst der öffentliche Schlüssel ggf. heruntergeladen werden kann.

Dieser Beitrag befasst sich mit dem Problem der Schlüsselverteilung und stellt dafür ein das neue Verfahren „Vertrauenswürdige Verteilung von Verschlüsselungsschlüsseln“ (VVV) vor, das im Folgenden VVV-Verfahren genannt wird. Dabei stehen zwei Ziele im Fokus: (1) Wodurch können die Speicherorte der Schlüssel eindeutig bestimmt werden? (2) Wie kann die Schlüsselverteilung benutzungsfreundlich und sicher umgesetzt werden?

In Abschnitt 2 wird dargelegt, wie die Verteilung von Schlüsseln derzeit bei den Verschlüsselungsverfahren S/MIME und OpenPGP erfolgt und warum diese ungenügend sind. Abschnitt 3 präsentiert das VVV-Verfahren. Abschnitt 4 zeigt eine prototypische Implementierung dieses Verfahrens. Anschließend werden in Abschnitt 5 alternative Verfahren zur Schlüsselverteilung vorgestellt. Der Beitrag endet mit einem Fazit.

2 Herkömmliche Verfahren der Schlüsselverteilung

Möchte ein Nutzer eine verschlüsselte E-Mail versenden, benötigt er den öffentlichen Schlüssel des Empfängers. Die beiden verbreitetsten Verfahren zur E-Mail-Verschlüsselung sind S/MIME [RT10] und OpenPGP [Ca07]. Diese Verfahren sind hinsichtlich der eingesetzten Schlüsselformate und Vertrauensmodelle nicht zueinander kompatibel. Beide Verfahren haben zudem das Problem, dass ihre Verfahren zur Schlüsselverteilung kompliziert und wenig benutzungsfreundlich sind. So kommt es beispielsweise zum Konflikt, wenn der Sender

und der Empfänger jeweils unterschiedliche Verfahren nutzen oder wenn der Sender eine E-Mail an mehrere Empfänger adressiert, von denen einige nur einen OpenPGP-Schlüssel und andere nur einen Schlüssel für S/MIME besitzen. Die E-Mail kann dann nicht versendet werden, weil jede E-Mail nur auf eine Weise (OpenPGP oder S/MIME) verschlüsselt werden kann.

2.1 Verteilung von S/MIME-Schlüsseln

Das S/MIME-Verfahren verwendet X.509-Zertifikate,[6] die typischerweise über LDAP-Verzeichnisdienste bereitgestellt werden. Hierzu muss der Sender einer E-Mail allerdings die Adresse desjenigen Verzeichnisdienstes kennen, in dem das Zertifikat des E-Mail-Empfängers zu finden ist, und diese in seiner E-Mail-Anwendung konfigurieren. Zudem muss der E-Mail-Sender der Zertifizierungsstelle, die den Schlüssel signiert hat, vertrauen, dass sie die Identität des Zertifikatsinhabers korrekt geprüft hat. Denn nur so kann der Nutzer sicher sein, dass das Zertifikat auch wirklich der Person gehört, der er eine verschlüsselte E-Mail senden möchte. Allerdings ist die Zertifizierungsstelle dem Sender der E-Mail nicht notwendigerweise bekannt und es ist unklar, weshalb er der digitalen Signatur einer ihm unbekannten Zertifizierungsstelle vertrauen sollte. Damit ist für den Sender auch fraglich, ob ein vorliegendes Zertifikat wirklich zum behaupteten Empfänger gehört.

In der Regel betreibt jede Zertifizierungsstelle ihren eigenen Verzeichnisdienst. Diese Verzeichnisdienste stellen „Insellösungen“ dar, d. h. sie sind in der Regel nicht untereinander vernetzt und somit nicht in der Lage, selbstständig in anderen Verzeichnisdiensten nach einem Zertifikat zu suchen, falls sie selbst eine Anfrage nach einem Zertifikat nicht beantworten können. Die E-Mail-Anwendung müsste also selbst ggf. in mehreren Verzeichnisdiensten nach dem Zertifikat suchen. Einige E-Mail-Anwendungen können dies jedoch nicht, da sie nur einen einzigen (aktiven) Verzeichnisdienst unterstützen. Nahezu unmöglich ist aus diesem Grund das Senden einer verschlüsselten E-Mail an mehrere Empfänger, wenn deren Zertifikate in jeweils unterschiedlichen Verzeichnisdiensten abgelegt sind.

Um die Probleme in Zusammenhang mit Verzeichnisdiensten zu umgehen, können die potentiellen Empfänger einer E-Mail in einem ersten Schritt zunächst dem Sender ihre Zertifikate übermitteln (z. B. persönliche Übergabe), allerdings ist dieser Weg ebenfalls nicht sehr benutzungsfreundlich und verhindert zudem das spontane Senden einer verschlüsselten E-Mail an einen Empfänger, dessen Zertifikat der Sender noch nicht hat.

[6] X.509-Zertifikate werden von einer Zertifizierungsstelle ausgestellt. Sie enthalten u. a. den öffentlichen Schlüssel des Nutzers und eine Signatur der Zertifizierungsstelle.

2.2 Verteilung von OpenPGP-Schlüsseln

Bei OpenPGP beglaubigen die Nutzer gegenseitig ihre öffentlichen Schlüssel und Identitäten, indem sie den Schlüsseln ihrer Kommunikationspartner ihre Benutzerkennung (User-ID)[7] und Signatur hinzufügen, und teilen sich ihre öffentlichen Schlüssel in der Regel auch gegenseitig mit. Dieses Verfahren ist für viele Nutzer jedoch nur schwer zu verstehen. Das gewonnene Vertrauen durch die gegenseitige Beglaubigung von Schlüsseln ist zudem begrenzt, da die User-IDs der Nutzer, die einen öffentlichen Schlüssel beglaubigt haben, oftmals unbekannt sind.

Als Alternative zu dem direkten Austausch von öffentlichen Schlüsseln zwischen den Nutzern können Nutzer ihre öffentlichen Schlüssel auf vorhandenen Schlüssel-Servern (sogenannten HKP-Servern)[8] veröffentlichen. Hierdurch entstehen jedoch einige Sicherheits- und Datenschutzprobleme. Im Gegensatz zu S/MIME-Zertifikaten, die von der Zertifizierungsstelle nach erfolgreicher Identitätsprüfung in dem Verzeichnisdienst veröffentlicht werden, veröffentlichen bei OpenPGP die Nutzer ihre öffentlichen Schlüssel selbst. Eine Identitätsprüfung findet nicht statt. Daher können auch Angreifer Schlüssel mit beliebigen E-Mail-Adressen erstellen und veröffentlichen oder Nutzer können irrtümlicherweise einen falschen Schlüssel auf dem Schlüssel-Server veröffentlichen. Für einen Nutzer, der einen Schlüssel über einen Schlüssel-Server bezieht, ist zunächst die authentische Herkunft nicht feststellbar. Der Nutzer muss eine Prüfsumme (Fingerprint) des Schlüssels mit dem Fingerprint vergleichen, den der wahre Eigentümer des Schlüssels ihm auf anderem Wege (z. B. persönlich oder telefonisch) mitgeteilt hat. Da diese Prüfung der Fingerprints nicht zwangsweise stattfindet, sind auch die mit diesen Schlüsseln erzeugten digitalen Signaturen nicht per se vertrauenswürdig.

Ein Datenschutzproblem besteht darin, dass anhand der Beglaubigungen eines öffentlichen Schlüssels die Kommunikationspartner des Schlüsselinhabers (der „soziale Graph“) rekonstruiert werden können. Zudem kann das Veröffentlichen des Schlüssels zu einem erhöhten Spam-Aufkommen führen, weil mit dem Schlüssel auch die E-Mail-Adresse veröffentlicht wird.

3 Verfahren für die vertrauenswürdige Verteilung von Schlüsseln

Das in diesem Abschnitt vorgestellte VVV-Verfahren ermöglicht es E-Mail-Anbietern, OpenPGP-Schlüssel und X.509-Zertifikate ihrer Kunden auf Anbieter-eigenen Schlüssel-Servern auf einheitliche Weise zu veröffentlichen und die Adressen der verwendeten Schlüssel-Server auf dem DNS-Server der E-Mail-Domain bekannt zu machen. Damit können beliebige Nutzer über die E-Mail-Adressen ihrer gewünschten Kommunikationspartner leicht deren Schlüssel für die E-Mail-Verschlüsselung finden.

[7] Eine User-ID besteht typischerweise aus Vorname, Nachname und E-Mail-Adresse.
[8] HKP steht für „OpenPGP HTTP Keyserver Protocol“ [Sh03].

3.1 Verteilung von Schlüsselinformationen über DNS

Weder bei S/MIME noch bei OpenPGP existiert eine direkte Verbindung zwischen der E-Mail-Adresse eines Nutzers oder der Domain des E-Mail-Anbieters und dem Speicherort des öffentlichen Schlüssels des Nutzers. Daher ist es nicht möglich, von einer E-Mail-Adresse oder Domain auf die Lokation eines dazugehörigen öffentlichen Schlüssels zu schließen.

Allerdings gibt es mit dem „Domain Name System“ (DNS) ein standardisiertes, etabliertes und auch stabiles System, welches neben dem Auflösen von Domains in IP-Adressen auch als Basis für die Verteilung von Schlüsselinformationen anhand von Domains dienen kann. Solche Domain-Informationen werden als DNS-Einträge in Form von so genannten „Resource Records“ auf dem für die Domain zuständigen DNS-Server abgelegt. Das DNS ermöglicht also eine direkte Verbindung zwischen der E-Mail-Adresse eines Nutzers und Informationen, die in den Resource Records der E-Mail-Domain liegen.

Zur Absicherung von DNS, welches selbst nur unzureichende Sicherheitsmechanismen bietet und daher als relativ leicht angreifbar und kompromittierbar gilt, wurden als Erweiterung die „DNS Security Extensions“ (DNSSEC) [Ar05a] entwickelt und stellen die notwendige Grundlage für die Verteilung von Schlüsselinformationen dar. Dieser Standard dient der Absicherung von DNS mittels kryptografisch signierter Ressource Records [Ar05b] und wird inzwischen von vielen Providern unterstützt. DNSSEC sieht vor, dass die Ressource Records mit dem privaten Schlüssel des jeweiligen Domain-Inhabers signiert werden. Die digitalen Signaturen werden in einem speziellen Ressource Record abgelegt. Der Hashwert des dazugehörenden öffentlichen Schlüssels wird mit dem privaten Schlüssel der übergeordneten Domain signiert, dieser Hashwert zusammen mit der Signatur sind in Ressource Records in der übergeordneten Domain zu finden. Dieses Prinzip setzt sich fort bis zu der sogenannten Root-Domain. Beispielsweise ist der öffentlichen Schlüssel der Domain *example.org* mit dem privaten Schlüssel der Domain *org* signiert und der öffentliche Schlüssel von *org* wiederum mit dem privaten Schlüssel der Root-Domain. Auf diese Weise entsteht eine eindeutige Vertrauenskette von signierten Schlüsseln der Domain-Inhaber, vergleichbar mit denjenigen Vertrauensketten bei hierarchischen Public-Key-Infrastrukturen (PKI). Im Gegensatz zu PKIs gibt es bei DNSSEC jedoch nur eine einzige Root-Domain.

Ressource Records können mittels DNSSEC zwar digital signiert werden, eine Verschlüsselung von Ressource Records ist jedoch nicht vorgesehen. Daher erscheint DNSSEC ungeeignet, die Zertifikate von Nutzern direkt in Ressource Records abzulegen, da Zertifikate personenbezogene Daten wie Name und E-Mail-Adresse enthalten können. Über DNSSEC können allerdings stattdessen Informationen über den Verzeichnisdienst oder Schlüssel-Server übermittelt werden, von dem das Zertifikat eines Nutzers abgerufen werden kann.

3.2 Anwendungsfälle des VVV-Verfahrens

Die beiden wichtigsten Anwendungsfälle der Schlüsselverteilung sind das Veröffentlichen und das Abrufen von Schlüsseln. Nutzer veröffentlichen mit Hilfe des VVV-Verfahrens ihre eigenen öffentlichen Schlüssel. Sie werden mittels der VVV-Lösung für andere Nutzer automatisch abrufbar, so dass sich alle Nutzer die Schlüssel ihrer Kommunikationspartner leicht beschaffen können.

Möchte der Nutzer zu seiner E-Mail-Adresse einen Verschlüsselungsschlüssel veröffentlichen, so wählt er über die E-Mail-Anwendung seinen Schlüssel im Schlüsselspeicher bzw. Zertifikatsspeicher aus. Anhand der zugehörigen E-Mail-Domain prüft die Anwendung, ob der E-Mail-Anbieter eine DNSSEC-geschützte Veröffentlichung von Schlüsseln anbietet. Dies wird daran erkannt, dass Resource Records mit Informationen zu den entsprechenden Schlüssel-Servern auf dem DNS-Server der E-Mail-Domain vorliegen.

Zur Veröffentlichung des Schlüssels muss der Nutzer gegenüber dem E-Mail-Anbieter nachweisen, dass er auf das E-Mail-Postfach Zugriff hat. Die E-Mail-Anwendung vermittelt die Nutzer-Authentifizierung über HTTPS, wobei der Nutzer die E-Mail-Adresse (Benutzerkonto) und ein Passwort angeben muss, und sendet nach erfolgreicher Authentifizierung den ausgewählten Schlüssel zur Veröffentlichung an den E-Mail-Anbieter. Der Nutzer muss nun den Besitz des zugehörigen privaten Schlüssels nachweisen. Dazu verschlüsselt der E-Mail-Anbieter mit dem empfangenen öffentlichen Schlüssel einen Verifikationscode. Die E-Mail-Anwendung entschlüsselt den Verifikationscode mit dem zugehörigen privaten Schlüssel. Nach erfolgreicher Entschlüsselung sendet die Anwendung den Verifikationscode an den E-Mail-Anbieter zurück. War der Nachweis erfolgreich, so veröffentlicht der E-Mail-Anbieter den Schlüssel auf einem seiner Schlüssel-Server. Falls auf dem Schlüssel-Server bereits ein Schlüssel für das gleiche Verfahren vorliegt, wird der bisherige Schlüssel gelöscht und durch den neuen Schlüssel ersetzt. Der Nutzer kann aber auch einen veröffentlichten Schlüssel ersatzlos auf dem Schlüssel-Server löschen.[9]

Das Szenario der Schlüsselsuche für die E-Mail-Verschlüsselung zeigt Abbildung 1. Der Nutzer verfasst in seiner E-Mail-Anwendung eine E-Mail und hat die E-Mail-Verschlüsselung aktiviert. Sobald er eine vollständige E-Mail-Adresse in das Empfängerfeld eingetragen hat, sucht die Anwendung mit der E-Mail-Adresse nach dem entsprechenden Schlüssel des Empfängers. Dazu fragt die Anwendung das DNS nach der DNS-Server-Adresse der E-Mail-Domain des Empfängers und ruft mit dieser Information vom DNS-Server des E-Mail-Anbieters des Empfängers die entsprechenden Resource Records ab (Schritt 1). Die Anwendung überprüft die DNSSEC-Signaturen dieser Records (Schritt 2). Konnten die Server-Adressen auf diese Weise verifiziert werden, so sucht die Anwendung anhand der E-Mail-Adressen auf den Schlüssel-Servern und ruft die Schlüssel der Empfänger ab

[9] Das Löschen könnte aus verschiedenen Gründen vom Nutzer gewollt sein, beispielsweise weil der private Schlüssel kompromittiert wurde oder weil der Nutzer mit seiner E-Mail-Adresse nicht mehr in einem offen zugänglichen Verzeichnis stehen möchte. Das Löschen ist also nicht unbedingt mit einem Widerruf des Schlüssels gleichzusetzen.

(Schritt 3). Schließlich wird die E-Mail für die Empfänger verschlüsselt und abgesendet (Schritt 4).

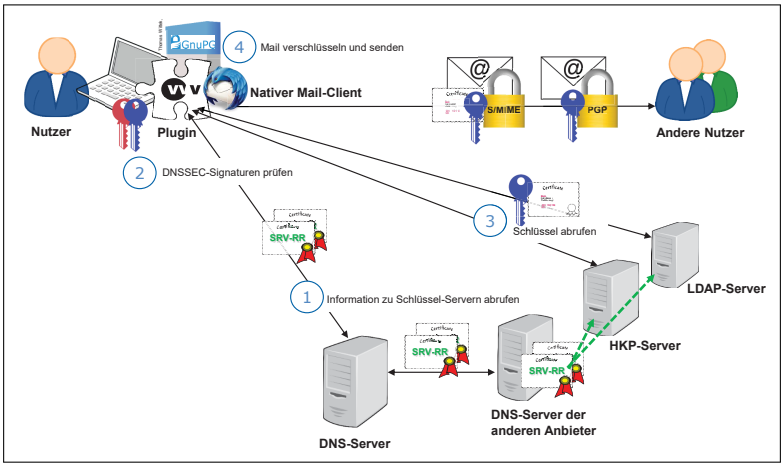

Abbildung 1: Gezielte Suche nach Verschlüsselungsschlüsseln

3.3 Erweitertes Vertrauensmodell des VVV-Verfahrens

Das VVV-Verfahren veröffentlicht die Information darüber, welcher Schlüssel zu welchem Nutzer gehört, indem der E-Mail-Anbieter so genannte „Service Resource Records“ gemäß RFC 6335 [Co11] auf dem DNS-Server der E-Mail-Domain bereitstellt. Die Records geben die authentischen Speicherorte der Zertifikate und Schlüssel an und sind vom E-Mail-Anbieter DNSSEC-signiert. Die E-Mail-Anwendung kann die Authentizität der Resource Records durch die Validierung der DNSSEC-Signaturen überprüfen. Die Informationsübertragung – einschließlich Veröffentlichung und Abruf der Schlüssel – wird durch kryptografisch gesicherte Dienste (DNSSEC, HTTPS, LDAPS) erbracht.

Das Verfahren gründet sich auf das Vertrauen, das Nutzer zur Internetfunktionalität (DNS), zu E-Mail-Adressen und deren Anbietern haben. Die Mechanismen des Web-of-Trust für OpenPGP bzw. der hierarchischen CA-Signaturen für S/MIME, die bei der herkömmlichen Veröffentlichung und Verifikation von Schlüsseln eine große Rolle spielen, sind bei diesem DNSSEC-basierten Verfahren von geringerer Bedeutung. Der E-Mail-Anbieter hat nun die entscheidende Rolle inne, da er für die Zuordnung von den Schlüsseln zu den E-Mail-Adressen der Nutzer bürgt. Damit brauchen die Nutzer nur solchen Instanzen zu vertrauen, denen sie beim Gebrauch von E-Mail-Adressen ohnehin vertrauen: Den Betreibern der grundlegenden Internetdienste (z. B. DNS) und den E-Mail-Anbietern. Nach diesem Vertrauensmodell ist es unerheblich, ob ein OpenPGP-Schlüssel die User-IDs anderer vertrauter Nutzer enthält oder ob ein X.509-Zertifikat von einer anerkannten Zertifizierungsstelle signiert wurde.

4 Prototypische Implementierung

Die E-Mail-Anwendung ist aus Nutzersicht die zentrale Komponente der E-Mail-Kommunikation. Das im vorigen Abschnitt beschriebene Verfahren wird beispielhaft in Form einer Erweiterung (Plugin) der E-Mail-Anwendung *Thunderbird*[10] implementiert. Teil der Lösung ist das frei erhältliche Programm *GnuPG*,[11] mit dem OpenPGP-Schlüssel erstellt und verwaltet werden. GnuPG ist die Verschlüsselungskomponente, welche im Hintergrund von E-Mail-Anwendungen das Ver- und Entschlüsseln mittels OpenPGP unterstützt.

4.1 Technische Umsetzung des Verfahrens

Im Folgenden werden die wesentlichen Komponenten vorgestellt, die in der prototypischen Implementierung zum Einsatz kommen.

Plugin. Zum Versenden einer verschlüsselten E-Mail muss der Nutzer in der E-Mail-Anwendung die Verschlüsselung aktivieren und die vollständige E-Mail-Adresse des gewünschten Kommunikationspartners in das Adressfeld eintragen. Das Plugin sorgt für die Beschaffung des für die Verschlüsselung notwendigen öffentlichen Schlüssels des Empfängers von einem Schlüssel-Server des betreffenden E-Mail-Anbieters. Das Plugin vermittelt zwischen der E-Mail-Anwendung mit der zugehörigen Kryptokomponente GnuPG und den Servern (DNS-Server, HKP-Server, LDAP-Server) der E-Mail-Anbieter. Es übergibt die ermittelten öffentlichen Schlüssel im Fall von OpenPGP an die GnuPG-Komponente und im Fall von S/MIME direkt an Thunderbird.

DNS-Resolver. Das Verfahren setzt beim TLS-gesicherten Abruf eines Schlüssels von einem Schlüssel-Server eine Absicherung der TLS-Zertifikate auf Grundlage von DNSSEC / DANE gemäß RFC 6698 [HS12] voraus. Das Plugin besitzt die Funktionalität eines DNSSEC-konformen rekursiven DNS-Resolvers, kann also selbstständig die Resource Records der gesamten Vertrauenskette ermitteln und auswerten, um die DNSSEC-Signaturen zu validieren. Bei der Installation des Plugins werden die öffentlich verfügbaren Schlüssel der DNS-Root-Zone vorinstalliert, um dem Plugin in der Anwendung als Sicherheitsanker für die Verifikation der DNSSEC-Signaturen zur Verfügung zu stehen, vgl. Abschnitt 3.1.

DNS-Einträge. Die Service Resource Records mit den Informationen zu den HKP-Servern und LDAP-Servern werden auf dem DNS-Server unter der entsprechenden E-Mail-Domain bereitgestellt. Sie enthalten u. a. Angaben zu Netzwerkprotokoll, Adresse und Port, unter denen der Dienst angeboten wird, und schreiben die verschlüsselte Datenübertragung vor, wie es auch andere „private HKP-Dienste“ tun.[12] Der DNS-Eintrag für einen HKP-Server der E-Mail-Domain *mailbox.org* sieht beispielsweise so aus:

[10] `https://www.mozilla.org/de/thunderbird/`

[11] `https://www.gnupg.org/`

[12] Siehe Einstellungen des privaten HKP-Servers `https://sks.spodhuis.org`.

List. 1: Beispiel eines Service Resource Records für HKP

```
_hkps._tcp.mailbox.org 3600 IN SRV 0 0 443 pgp.mailbox.org
```

Schlüssel-Server. Die Veröffentlichung von OpenPGP-Schlüsseln erfolgt mit der TLS-gesicherten Variante des „HTTP Keyserver Protocols“ (HKP) [Sh03]. Im Gegensatz zu verbreiteten HKP-Implementierungen ist keine Synchronisation mit anderen HKP-Servern vorgesehen, da immer nur auf die Schlüssel-Server des jeweiligen E-Mail-Anbieters zugegriffen werden soll. Eine Synchronisation dieser Schlüssel-Server mit anderen HKP-Servern könnte beispielsweise dazu führen, dass die Schlüssel nicht exakt mit denen übereinstimmen, die der Nutzer ursprünglich veröffentlichen ließ. Zertifikate für S/MIME werden mittels des LDAP-Protokolls [Se15] veröffentlicht. Die TLS-Zertifikate zur Kommunikation mit HTTPS bzw. LDAPS sind auf Grundlage von DNSSEC / DANE abgesichert.

4.2 Benutzungsfreundliche Sicherheit

Um Nutzern einen intuitiven Umgang mit der Anwendung und somit einen unkomplizierten Zugang zu vertrauenswürdigen Verschlüsselungsschlüsseln zu ermöglichen, spielt die Gestaltung der Nutzeroberfläche eine bedeutende Rolle. Daher werden unterschiedliche Nutzergruppen in allen Phasen des Gestaltungs- und Entwicklungsprozesses mit einbezogen. Mithilfe dieser Nutzergruppen wurden grundlegende Anforderungen für das Bedienkonzept des Plugins ermittelt, welches im Wesentlichen auf drei Kernelementen basiert.

Startseite. Bei der Erstnutzung wird der Nutzer auf der Startseite des Plugins begrüßt. Er erhält hier einen kurzen Überblick über Zweck und Funktionalität des Plugins. Außerdem können weiterführende Links, Erklärungen oder Videos angeboten werden, um das Thema „E-Mail-Verschlüsselung“ anschaulicher aufzubereiten. Von der Startseite aus gelangt der Nutzer direkt zur Schlüsselverwaltung, um seine öffentlichen Schlüssel zu veröffentlichen.

Schlüsselverwaltung. In der Schlüsselverwaltung kann der Nutzer seine eigenen öffentlichen Schlüssel veröffentlichen, siehe Abbildung 2. Aus Nutzersicht gehören dazu folgende Schritte: (1) Über einen Einwilligungstext zum Datenschutz erfährt der Nutzer, was mit seinen Daten passiert, also welche Daten zu welchem Zweck, wo und wie lange gespeichert werden.[13] (2) Der Nutzer wählt aus einer Übersicht die Schlüssel für die Veröffentlichung aus. (3) Der Nutzer gibt das Passwort des jeweiligen E-Mail-Kontos ein und falls erforderlich das Passwort des zugehörigen privaten Schlüssels. (4) Durch visuelles Feedback wird dem Nutzer signalisiert, dass die Schlüssel erfolgreich veröffentlicht wurden.

[13] Siehe Kapitel 3.2.1 im Beitrag „Anforderungen des künftigen europäischen Datenschutzrechts an die vertrauenswürdige Verteilung von Verschlüsselungsschlüsseln“ von S. Blazy, S. Gonscherowski und A. Selzer in diesem Konferenzband.

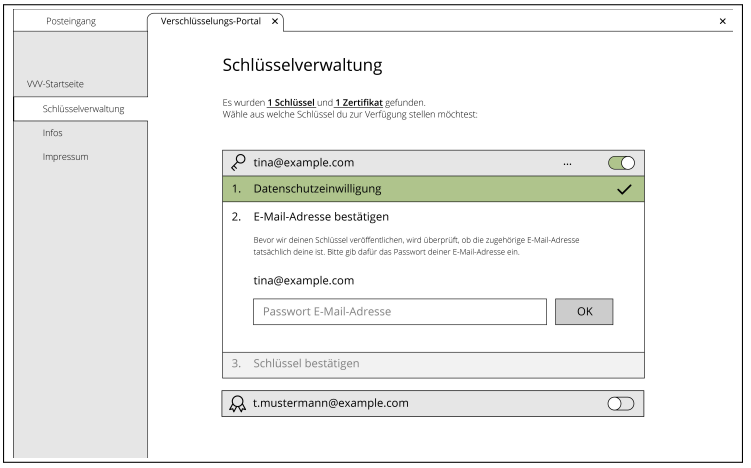

Abbildung 2: Schlüsselverwaltung

Schlüsselabruf. Das Abrufen der öffentlichen Schlüssel der Kommunikationspartner verläuft automatisiert im Hintergrund, sobald der Nutzer die vollständige E-Mail-Adresse des Empfängers in das Adressfeld eingegeben hat (vgl. Abschnitt 4.1). Das Plugin signalisiert dem Nutzer lediglich durch einen visuellen Indikator (z. B. ein Symbol im Adressfeld der eingegebenen E-Mail-Adresse), ob ein Schlüssel oder Zertifikat gefunden wurde. Wenn der Schlüssel des Empfängers nicht ermittelt werden konnte, bekommt der Nutzer durch einen entsprechenden Dialog Hilfestellung und Handlungsoptionen angeboten.

5 Alternative Verfahren zur Schlüsselverteilung

In den letzten Jahren wurden bereits verschiedene Ansätze verfolgt, um die Schlüsselverteilung, insbesondere von OpenPGP-Schlüsseln, zu vereinfachen. In diesem Abschnitt werden drei solcher Verfahren vorgestellt: „Public Key Association“ (PKA), „DANE Bindings for OpenPGP“ und „OpenPGP Web Key Service“. Alle drei Verfahren haben gewisse Schwächen, die bei der Entwicklung der VVV-Lösung aufgegriffen wurden.

5.1 Schlüssel-Informationen auf dem DNS-Server

Im Verfahren „Public Key Association“ (PKA) [Ko06] zur Veröffentlichung von OpenPGP-Schlüsseln wird für jede E-Mail-Adresse ein eigener so genannter „Public Key Association Resource Record“ im DNS angelegt. Der Record ist der Form nach ein „Text String Resource Record“ mit frei definierbarem Text und kann neben einer Versionsangabe des Protokolls, der E-Mail-Adresse und dem Fingerprint des OpenPGP-Schlüssels auch die Web-Adresse

einer Bezugsquelle des Schlüssels enthalten. Das Verfahren ist kein RFC-Standard. Es wird in der Praxis zwar eingesetzt, hat sich aber nicht weiter durchgesetzt. Das Verfahren kann zudem einige Sicherheits- und Datenschutzanforderungen nicht erfüllen:

- Das Verfahren ist zeitlich vor der Verbreitung von DNSSEC definiert worden. Es propagiert die Veröffentlichung der Fingerprints von OpenPGP-Schlüsseln, so dass das herkömmliche OpenPGP-Vertrauensmodell auf Basis des Vergleichs von Fingerprints weiterhin notwendig ist.
- Das Verfahren skaliert schlecht und verringert die Performance des DNS, da jede E-Mail-Adresse einen eigenen Resource Record erfordert. Zudem enthalten die Einträge mit der E-Mail-Adresse des Nutzers und dem Fingerprint des OpenPGP-Schlüssels personenbezogene Daten, die ungeschützt ausgelesen werden können.

5.2 Schlüssel auf dem DNS-Server

RFC 7929 [Wo16] spezifiziert das Verfahren „DANE Bindings for OpenPGP“, einen experimentellen Ansatz für die Verbreitung und das Auffinden von OpenPGP-Schlüsseln auf Basis von DNSSEC und DANE. Im Gegensatz zu dem im obigen Abschnitt 5.1 beschriebenen Verfahren und dem VVV-Verfahren, vgl. Abschnitt 3, welche ebenfalls auf DNSSEC basieren, werden hier die öffentlichen OpenPGP-Schlüssel direkt in DNS Resource Records abgelegt.[14] Die DNS-Anfragen werden nicht für die Domain einer E-Mail-Adresse gestellt, sondern für die komplette E-Mail-Adresse. Hieraus ergeben sich jedoch einige Probleme, die gegen diesen Ansatz sprechen:

- Eine Verschlüsselung ist mittels DNSSEC nicht möglich. Die OpenPGP-Schlüssel bzw. Zertifikate der Nutzer enthalten jedoch personenbezogene Daten und sollten daher nicht unverschlüsselt übertragen werden.
- Wie das Verfahren in Abschnitt 5.1 skaliert auch dieses Verfahren schlecht und verringert die DNS-Performance. Zudem wäre die maximal zulässige Größe von DNS Records schnell überschritten, wenn öffentliche OpenPGP-Schlüssel umfangreiche Beglaubigungen anderer Nutzer enthalten.

5.3 Schlüssel auf einem Webserver

Der IETF-Draft „OpenPGP Web Key Service“ [Ko17] beschreibt ein Verfahren, mit dem öffentliche OpenPGP-Schlüssel veröffentlicht und abgerufen werden können. Zum Veröffentlichen des Schlüssels wird im ersten Schritt via HTTP eine E-Mail-Adresse des

[14] Ein ähnlicher Entwurf zur Veröffentlichung von S/MIME-Zertifikaten ist in [HS17] beschrieben.

E-Mail-Anbieters abgerufen, an die im zweiten Schritt der eigene öffentliche Schlüssel via SMTP gesendet wird. Der E-Mail-Anbieter veröffentlicht den empfangenen Schlüssel auf seinem Webserver. Bezüglich der verwendeten unterschiedlichen Protokolle HTTP und SMTP erscheint dieses Verfahren inkonsistent.

Zum Abruf des Schlüssels eines gewünschten Kommunikationspartners anhand dessen E-Mail-Adresse dient ein HTTPS-basiertes Verfahren. Hierbei wird eine HTTP-GET-Anfrage an einen Webserver gesendet, welcher über die Domain der E-Mail-Adresse, zu der der dazugehörige Schlüssel gesucht wird, erreichbar ist. Der lokale Teil der E-Mail-Adresse wird in den Pfad der URL kodiert. Dieses proprietäre Verfahren verzichtet auf HKP- und LDAP-Server, obwohl diese bei vielen Anbietern bereits etabliert sind.

6 Fazit

Das in diesem Beitrag vorgestellte VVV-Verfahren sorgt dafür, dass E-Mail-Anwendungen die authentischen OpenPGP-Schlüssel und X.509-Zertifikate gewünschter Kommunikationspartner automatisch abrufen und damit E-Mails problemlos verschlüsseln können. Die Nutzer können auf die manuelle Verteilung und Überprüfung von Schlüsseln verzichten und brauchen sich nicht mehr um die Adressen von Schlüssel-Servern zu kümmern.

Der Nutzer autorisiert bei diesem Verfahren die Veröffentlichung seines öffentlichen Schlüssels und kann jederzeit die Veröffentlichung zurückziehen. Das Plugin führt den Nutzer auf benutzungsfreundliche Weise durch den Veröffentlichungsprozess von der Auswahl des lokalen Schlüssels bis hin zur Bestätigung der Veröffentlichung durch den E-Mail-Anbieter. Für das Hochladen des Schlüssels an den E-Mail-Anbieter existiert bisher keine standardisierte Schnittstelle, so dass die prototypische Implementierung in dieser Hinsicht bislang nur eine proprietäre Schnittstelle zu einem ausgewählten E-Mail-Anbieter besitzt. Für eine Verbreitung des Verfahrens ist die Spezifikation einer einheitlichen von den E-Mail-Anbietern akzeptierten Schnittstelle erforderlich.

Die veröffentlichten Schlüssel sind eindeutig auffindbar, weil die Speicherorte über das DNS standardkonform, eindeutig und sicher mittels Resource Records kommuniziert werden. Das Abrufen erfolgt automatisch im Hintergrund durch das Plugin, sobald der Nutzer eine vollständige E-Mail-Adresse im Adressfeld einer E-Mail eingegeben hat, und ist somit ebenfalls sehr benutzungsfreundlich. Damit ist es möglich, eine Mail für mehrere Empfänger zu verschlüsseln, auch wenn deren Schlüssel auf unterschiedlichen Schlüssel-Servern veröffentlicht sind. E-Mail-Anbieter können dieses Verfahren leicht realisieren, weil dazu auf dem DNS-Server nur weitere Resource Records angelegt werden müssen. Bestehende Schlüssel-Server können daher bei diesem Verfahren weiter genutzt werden.

Literatur

[Ar05a] Arends, R.; Austein, R.; Larson, M.; Massey, D.; Rose, S.: RFC 4033 - DNS Security Introduction and Requirements. 2005.

[Ar05b] Arends, R.; Austein, R.; Larson, M.; Massey, D.; Rose, S.: RFC 4034 - Resource Records for the DNS Security Extensions. 2005.

[Ca07] Callas, J.; Donnerhacke, L.; Finney, H.; Shaw, D.; Thayer, R.: RFC 4880 - OpenPGP Message Format. 2007.

[Co11] Cotton, M. S.; Eggert, L.; Touch, Dr. J. D.; Westerlund, M.; Cheshire, S.: Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry. (6335), 2011.

[HS12] Hoffman, P.; Schlyter, J.: RFC 6698 - The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA. 2012.

[HS17] Hoffman, P.; Schlyter, J.: Using Secure DNS to Associate Certificates with Domain Names For S/MIME. (draft-ietf-dane-smime-16), 2017.

[HSW16a] Herfert, M.; Selzer, A.; Waldmann, U.: Laientaugliche Schlüsselgenerierung für die Ende-zu-Ende-Verschlüsselung – Schlüssel für alle durch die Volksverschlüsselung. Datenschutz und Datensicherheit – DuD, 05:290–294, 2016.

[HSW16b] Herfert, M.; Selzer, A.; Waldmann, U.: Selbstdatenschutz in Zeiten massenhafter E-Mail-Überwachungen. BvD-News, 01:57–59, 2016.

[Ko06] Koch, W.: Public Key Association, http://www.g10code.de/docs/pka-intro.de.pdf. 2006.

[Ko17] Koch, W.: OpenPGP Web Key Service - draft-koch-openpgp-webkey-service-03. 2017.

[RT10] Ramsdell, B.; Turner, S.: RFC 5751 - Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2: Message Specification. 2010.

[Se15] Sermersheim, J.: RFC 4511 - Lightweight Directory Access Protocol (LDAP): The Protocol. (4511), 2015.

[Sh03] Shaw, D.: The OpenPGP HTTP Keyserver Protocol (HKP). (draft-shaw-openpgp-hkp-00), 2003.

[Wo16] Wouters, P.: RFC 7929 - DNS-Based Authentication of Named Entities (DANE) Bindings for OpenPGP. 2016.