Verschlüsseln mit Stil

PGP in der Praxis, Teil 1: Schlüssel und ihre Verwaltung, Signieren

> Erfreulicherweise wächst auch in der radikalen Linken das Bewusstsein, dass unverschlüsselte E-Mail nicht weit entfernt ist vom Plakatieren der enthaltenen Ausführungen. So kommen immer mehr Anfragen an die Datenschutzgruppe (wie auch an die Ortsgruppen) verschlüsselt an. Grund genug, unserem flammenden Aufruf in RHZ 2/04 ein paar tiefer schürfende Ausführungen folgen zu lassen.

Um kurz zu rekapitulieren: PGP funktioniert ein wenig wie eine große Wand von Briefkästen -- stellt euch ein gigantisches Hochhaus vor. Jeder Briefkasten hat eine Klappe, die in etwa dem öffentlichen Schlüssel entspricht, und einen Briefkastenschlüssel; das ist der geheime Schlüssel von PGP.

Wer die Klappe findet, kann Dinge in den Briefkasten werfen, und analog reicht der öffentliche Schlüssel, um Mail *an* eineN EmpfängerIn zu verschlüsseln. An etwas im Briefkasten kommt nur heran, wer den Briefkastenschüssel (also den geheimen Schlüssel von PGP) hat. Genauso wie ein Brief weg ist, wenn er hinter der Klappe verschwunden ist, könnt ihr eine nicht an euch verschlüsselte Mail auch dann nicht mehr entschlüsseln, wenn ihr sie vorher selbst verschlüsselt habt[1].

Die Briefkastenmetapher erklärt recht schön, warum eure MailpartnerInnen euren öffentlichen Schlüssel nur brauchen, wenn sie euch was schicken wollen, nicht aber, um eure Mails zu lesen. Da weiter nur der geheime Schlüssel durch die Passphrase geschützt ist, braucht ihr sie zum Verschlüsseln auch nicht einzugeben, wohl aber beim Entschlüsseln.

Viele Menschen verbreiten derzeit ihren Schlüssel per Mail oder auf Webseiten. Das ist nicht zwingend verkehrt, zumal Mailprogramme meist recht komfortable Funktionen zum Importieren von Schlüsseln aus Mails bieten. Thunderbird mit Enigmail (das nennen wir im folgenden TE; wir empfehlen TE für Menschen, die keine guten Gründe haben, was anderes zu verwenden und verlinken von unserer Webseite auf ein Archiv, mit dem man TE für Windows auf einen USB-Stick bekommt) macht das automatisch, wenn er in einer Mail Schlüssel sieht. Im Zweifel kann mensch Schlüssel in einem Anhang durch Rechtsklick und „Import PGP Key“[2] per Hand importieren, bei Schlüsseln im Mailtext gibts im OpenPGP-Menü unter „Sender's Key“ den Eintrag „Import Public Key“ (aber nur im „Expertenmodus“[3], weil es das wie gesagt in der Regel nicht braucht).

Bekommt ihr Schlüssel von Webseiten, speichert diese in eine Datei und benutzt „Import key from file“ im File-Menü der Schlüsselverwaltung (vgl. unten).

Die Keyserver

Auf Dauer ist es etwas lästig, immer zunächst Schlüssel rumschicken zu müssen, bevor mensch sicher kommunizieren kann. Deshalb gibt es die Keyserver. Das sind Rechner im Netz, die Schlüssel empfangen und verteilen. Schnittstellen dazu bieten praktisch alle Mailprogramme in ihrer Schlüsselverwaltung. Es gibt aber auch eigene Programme zur Schlüsselverwaltung, die unabhängig von Mailprogrammen funktionieren. Unser Tipp in dem Ressort ist gpa[4].

Um einen Schlüssel vom Keyserver zu holen, könnt ihr z.B. in TE im OpenPGP-Menü „Key Management“ wählen. In dem dann erscheinenden Fenster, der Schlüsselverwaltung, gibt es ein „Key Server“-Menü, in dem dann auch „Search for Key“ steht. Ihr könnt nach weitgehend beliebigen Teilen von Adressen und Namen suchen und werdet in der Regel eine Liste von Schlüsseln zurückbekommen. Wählt den oder die Schlüssel aus, die ihr haben wollt, fast fertig. Probiert es mal mit „datenschutzgruppe“ aus.

Es ist weitgehend egal, welchen Keyserver ihr verwendet; die verschiedenen Keyserver gleichen sich un-

tereinander ab. Wenn ihr gar keinen kennt und eure Schlüsselverwaltung keine mitbringt (unwahrscheinlich), tuts in der Regel pgp.mit.edu.

Weil die Keyserver die Verwendung von PGP deutlich erleichtern, empfehlen wir ihre Benutzung, also auch, dass ihr euren Schlüssel hochladet; das ist nicht schwer, in TEs Schlüsselverwaltung z.B. reicht es, den eigenen Schlüssel zu wählen und Keyserver/Upload public keys zu machen. Aber Vorsicht: Die Keyserver vergessen nichts, einmal hochgeladene Schlüssel können nicht wieder entfernt werden. Ladet euren Schlüssel also nur hoch, wenn ihr den zugehörigen geheimen Schlüssel nicht verschmeißt und die Passphrase nicht vergesst.

Unglücke dieser Art, aber bei Gruppenadressen vielleicht auch das plötzliche Verschwinden des/der Zuständigen, können erklären, warum für manche Leute mehrere Schlüssel in den Keyservern stehen -- meist ist der neueste Schlüssel dann die beste Wahl.

Es gibt aber auch gute Gründe für nicht mehr verwendete Schlüssel. Sie könnten etwa von vorneherein nur für eine begrenzte Zeit erzeugt (vgl. unten), im schlimmsten Fall gar „kompromittiert“ worden sein. Im Jargon der Kryptonerds bedeutet das: Schlüssel samt Passphrase sind in die falschen Hände gefallen. Um dann nicht unterzugehen, braucht ihr ein „Rückrufzertifikat“, das ihr bei der Erstellung eures Schlüssels miterzeugt haben solltet. Wenn ihr es verloren habt, könnt ihr z.B. in der Schlüsselverwaltung von TE ein neues erzeugen, indem ihr euren Schlüssel auswählt und unter „Generate“ „Revocation Certificate“ ausführt.

Hebt das Zertifikat gut auf, vielleicht bei vertrauenswürdigen GenossInnen. Wird es (von irgendwem) auf einen Keyserver geladen, wird euer Schlüssel ungültig, was bedeutet, dass sich gute Mailprogramme weigern, damit zu verschlüsseln. Ein Vorteil der Verwendung von Keyservern ist, dass sich solche Rückrufzertifikate leicht verbreiten lassen (TE hat dazu „Refresh all public keys“ im Keyserver-Menü seiner Schlüsselverwaltung). Ohne Keyserver müsste das Rückrufzertifikat ähnlich verbreitet werden wie vorher der Schlüssel, und das ist mal mindestens unbequem.

Nicht verschwiegen sei aber, dass Mailadressen auf Keyservern mehr oder weniger öffentlich sind (was ja Sinn der Sache ist). Es sollen schon Spammer versucht haben, sich aus diesem Adresspool zu bedienen. Im Vergleich z.B. zur Veröffentlichung auf Webseiten ziehen Adressen auf Keyservern aber nicht viel Spam oder vergleichbar lästige Aufmerksamkeit.

Digitale Signatur

Die Mathematik hinter „asymmetrischer“ Verschlüsselung, die hinter PGP steht, erlaubt etwas, das die Briefkastenmetapher nicht hergibt: die digitale Unterschrift. Die Idee ist, aus einem Dokument und eurem *geheimen* Schlüssel eine Zahl auszurechnen. Wer euren öffentlichen Schlüssel hat, kann nachrechnen, ob diese Zahl, die Unterschrift eben, stimmt, kann sie aber selbst nicht berechnen. In Summe ist das eine ideale Unterschrift: Ihre Gegenwart zeigt sicher, dass ein bestimmter geheimer Schlüssel am Werk war (was in der Realität natürlich verschiedene Dinge bedeuten kann), und jedeR kann das nachprüfen, ohne sie aber „durchpausen“ zu können.

Mailclients bieten euch in dem Menü, in dem ihr die Verschlüsselung auswählt, in der Regel auch an, die Mail zu „signieren“, was genau diese Unterschrift meint. Wenn ihr signiert, geht das nur mit eurem privaten Schlüssel, und deshalb fragt euch das Mailprogramm beim Verschicken signierter Mails auch beim Abschicken nach eurer Passphrase.

Nun wäre es sicher eine bessere Welt, wenn Banken PGP-Unterschriften für Überweisungen verlangen würden, und schon gar, wenn das haarsträubende „Bürgerportal“ der Regierung auf PGP beruhen würde. In unseren Kreisen gibt es für die Signatur aber normalerweise keinen zwingenden Grund, *Wenn* die Gegenseite mal lernt, dass Absender von Mails leichter zu fälschen sind als ein Lächeln *und* sie die Kreativität aufbringt, diesen Umstand zum Stiften von Konfusion auszunutzen („Was? Eine Stellungnahme des BuVo zu Gaza? Lass sehen!“), *dann* müsste mensch da nochmal anders drüber nachdenken.

Wirklich schaden dürfte die Signatur aber in aller Regel auch nicht, es sei denn, es ginge euch um das, was die Geheimdienste „Deniability“ nennen, der Möglichkeit nämlich, die AutorInnenschaft einer Mail abstreiten zu können. Durch die Vorratsdatenspeicherung (so sie denn durchgeführt wird) lassen sich allerdings ohne weitere Tricks verschickte Mails erschütternd einfach noch lange im Nachhinein zurückverfolgen, so dass mensch letztlich allenfalls MitbewohnerInnen belasten

könnte, und dann könnte euch die RH nicht mehr unterstützen.

Schließlich setzt die Prüfung der Unterschrift voraus, dass der Schlüssel auch wirklich verlässlich der unterschreibenden Person zugeordnet ist. Dieses Problem ist tatsächlich schwierig und wird Thema des zweiten Teils dieses Artikels sein. Dabei werden die Unterschriften dann ihren Starauftritt haben.

Die Schlüssel

Wenn ihr das hier lest, habt ihr vermutlich längst euren Schlüssel erzeugt. Für den Fall, dass ihr irgendwann mal einen neuen Schlüssel braucht (oder einen weiteren) oder anderen Leuten bei der Erzeugung helft, sind zwei Fragen zu beantworten.

Erstens ist das die Frage nach der Schlüssellänge. Dabei gilt, dass längere Schlüssel (also mit „mehr Bits“) schwerer zu knacken sind. Für das, was in PGP eingebaut ist, könnten 1024 Bits für Organisationen mit viel Zeit und Geld allmählich in den Bereich des Knackbaren kommen. Andererseits brauchen längere Schlüssel mehr Rechenzeit im Alltagsbetrieb. Auf Rechnern aus dem dritten Jahrtausend ist das aber nicht mehr wirklich zu bemerken. Wenn ihr also euren Schlüssel nicht gerade auf eurem Uralt-Mobiltelefon verwenden wollt, nehmt einfach 4096 Bit-Schlüssel und vergesst die ganze Sache.

Zweitens ist da noch die Frage, ob der Schlüssel irgendwann „ablaufen“, er also nach einer bestimmten Zeit automatisch ungültig werden soll. In der reinen Lehre ist es fast immer eine gute Idee, Schlüssel dann und wann zu wechseln, vor allem, wenn sie zur Verschlüsselung großer Datenmengen verwendet werden. PGP-Schlüssel verschlüsseln aber normalerweise nur recht überschaubare Datenmengen (nämlich im Groben ein paar hundert Bits pro verschickter Mail). Dazu kommt, dass ein „reibungsloser“ Übergang von einem ablaufenden auf einen neuen Schlüssel nicht ganz einfach ist, insbesondere, wenn ihr im Web of Trust (vgl. Teil 2 in der nächsten Ausgabe) seid. In dem Sinn ist unsere Empfehlung: Lasst eure Schlüssel für immer gültig sein. Mit einem Rückrufzertifikat könnt ihr den Schlüssel immer noch ablaufen lassen, aber dann zu einem Zeitpunkt, an dem es für euch bequem ist.

Ausklang

PGP wurde in der Steinzeit entwickelt (also den achtziger Jahren), noch bevor Mails Anhänge haben konnten. Damals hat der PGP-Autor Phil Zimmermann ein System erdacht, mit dem die konfusen Bytes, in die ein normaler Text verschlüsselt wird, in Mails, die konfuse Bytes nicht leiden können, transportiert werden können. Dieses „alte“ System, manchmal unter „inline“ firmierend, hat etliche Nachteile, unter anderem, dass es schwer ist, Umlaute darin ohne Verwürfelungen zu transportieren, und auch, dass es nicht gut mit wirklichen Anhängen zusammengeht.

Deshalb wurde auch schon vor vielen Jahren etwas ersonnen, das sich PGP/Mime nennt. Wenn es irgendwie geht, solltet ihr das verwenden, wenn ihr eure Mails verschickt (TE fragt euch beim Absenden, und im Expertenmodus könnt ihr auch Regeln einrichten, die das für bestimmte EmpfängerInnen automatisch auswählen). Im Groben gibt es überhaupt nur einen Grund, PGP/Mime *nicht* zu verwenden: Webmail, die nichts von PGP versteht. Und wenn ihr so einen Webmailer verwendet, solltet ihr schnell wechseln. Aus unserem Spektrum gibts PGP-Webmail beispielsweise bei riseup.net und immerda.ch.

Beim nächsten Mal beantworten wir dann die große Frage nach dem Vertrauen unter GenossInnen.

Datenschutzgruppe der Roten Hilfe Heidelberg

datenschutzgruppe@rotehilfe.de

PGP Fingerprint: a3d8 4454 2e04 6860 0a38 a35e d1ea ecce f2bd 132a

http://www.datenschmutz.de

[1] In den meisten Mailprogrammen merkt ihr das nicht unbedingt, weil diese ungefragt jede Mail, die ihr verschlüsselt, auch noch gleich für euch verschlüsseln, also quasi eine Kopie in euren eigenen Briefkasten werfen. Es gibt Menschen, die dieses Verhalten blöd finden, weshalb es in der Regel abschaltbar ist, bei Thunderbird/Enigmail etwa im Expertenmodus in den Einstellungen zu „Sending“ als „add my own key to the recipients list“.

[2] Die Bedienungen sind hier nur als Beispiel zu verstehen -- auf dem Mac ist das kein Rechtsklick, und wenn euer Enigmail deutsch spricht, ist die Bezeichnung natürlich Deutsch.

[3] Der „Expertenmodus“ ist nicht nur für ExpertInnen interessant. Um ihn anzuschalten, wählt im OpenPGP-Menü „Preferences“ und kreuzt „Display expert settings“ an; damit bekommt ihr tolle Reiterlein in den Einstellungen und mehr Einträge im OpenPGP-Menü.

[4] http://www.gnupg.org/gpa.html