

Sicher unterwegs oder Austausch digitaler Schlüssel und das Web-of-Trust

Das Internet ist ein weltweit offenes Netzwerk über das Daten ausgetauscht werden. Privatpersonen, Firmen, Ämter und Behörden oder sonstige Institutionen nutzen dieses Medium zur Kommunikation. So einfach und schnell es sein mag, sicher ist es nicht um vertrauliche Daten austauschen zu können. Gerade durch neuere „Abhörmaßnahmen“ (siehe <http://www.heise.de/newsticker/meldung/50937> bzw. „Überwachung der E-Mail Provider“ <http://www.heise.de/newsticker/meldung/52954>) wird dieses Thema auch immer interessanter.

Vom Absender bis hin zum Empfänger einer Nachricht bewegen sich die Daten auf nicht vorhersehbaren Wegen, die dann unter Umständen mitgelesen werden und somit auch verändert oder sogar missbraucht werden können. Der Austausch von digitalen Schlüsseln und der Einsatz von kryptografischen Verfahren dienen dem Schutz der Vertraulichkeit, Identität und Authentifizierung des Einzelnen.

Weitere Informationen findet man u.a. im Gnu-Handbuch zum Schutz der Privatsphäre (<http://www.gnupg.org/gph/de/manual/index.html>)

Warum GnuPG?

GnuPG (GNU Privacy Guard) ist ein Tool zur Verschlüsselung und Signierung digitaler Daten und Datenformat unabhängig, das heißt E-Mail, Texte, Bilder, Quellcode, Datenbanken, Festplatteninhalte, also alles kann entsprechend signiert und verschlüsselt werden. GnuPG entspricht der im RFC2440 festgelegten OpenPGP-Spezifikation und ist kompatibel zu PGP 5.x der Firma PGP, welches häufig im industriellen Bereich eingesetzt wird.

GnuPG verwendet dazu hauptsächlich ein hybrides Verfahren mit öffentlichen Schlüsseln. Zum Verschlüsseln kann GnuPG aber ebenso auch ein symmetrisches Verfahren einsetzen.

GnuPG wird derzeit als eines der sichersten Verfahren zum Verschlüsseln und Signieren von Daten angesehen. Auch nach heutigen Maßstäben ist bei sorgfältiger Anwendung eine Verschlüsselung mit GnuPG nicht kompromittierbar.

GnuPG ist im Gegensatz zu dem kommerziellen Pendant PGP von NAI freie Software und darüber hinaus nicht durch Ausfuhrbestimmungen anderer Länder (z.B. USA) im Funktionsumfang beschränkt.

Woher bekomme ich GnuPG?

GnuPG gibt es für fast alle gängigen Betriebssysteme.

GnuPG (Unix/Linux) : <http://www.gnupg.org>

MacGPG (MacOS) : <http://macgpg.sourceforge.net>

WinPT (MS-Windows): <http://winpt.sourceforge.net>

Was sind digitale Signaturen?

Eine digitale Signatur dient dazu sich seinem Gegenüber aus als der auszuweisen, der man vorgibt zu sein. Dies kann u.U. bei beglaubigten Dokumenten, elektronisches Rathaus, elektronische Steuererklärung und dergleichen von Bedeutung sein oder werden. Eine Signatur oder eben digitale Unterschrift, wird genauso geleistet wie eine handschriftliche, bei Bedarf und immer wieder neu. Um eine Unterschrift leisten zu können, brauche ich einen digitalen Schlüssel, oder ein Zertifikat. Die wohl bekannteste Lösung ist PGP, heute von der PGP Corporation (<http://www.pgp.com/>), oder deren freies Equivalent GnuPG (<http://www.gnupg.org/>). Es gibt auch noch weitere Lösungen, die technisch allerdings vollkommen anders und zu PGP/GnuPG inkompatibel sind. Hier wären auf der kommerziellen Seite z.B. **Thawte** (<http://www.thawte.com>) und auf der freien Seite **CACert.org** (<http://www.cacert.org/>) zu erwähnen, die sich dann später im Zertifikatsmanager von Browsern oder E-mail Clients einschreiben und hier verwaltet werden können.

Bei den beiden letztgenannten ist das Verfahren zur Erlangung eines Zertifikates relativ einfach:

Man holt sich ein kostenloses x.509 Zertifikat (<http://www.thawte.com/email/index.html>) bei Thawte. (:join:) oder CACert. Dabei gibt man die Nummer seines Personalausweises an.

An dieser Stelle hat man dann schon ein digitales Zertifikat und kann es natürlich schon verwenden, es enthält aber noch den Vermerk „freemail“ und ist noch nicht weiter beglaubigt durch einen entsprechenden Notar (Notary). Man kann es zwar schon verwenden, aber um sich eben verifiziert zu haben ist je nach Anbieter eine gewisse Anzahl von Notary Points notwendig, damit das Zertifikat dann auch mitsamt des eigenen Namens auftaucht. Von seinem Ausweis fertigt man hierzu einen Stapel Kopien an (wenigstens 10 Stück, besser mehr). Man füllt den ersten Kasten dieses Formulars

(<https://www.thawte.com/wot/tngerman.pdf>) aus und macht ebenfalls einen Stapel von Kopien (Sinnvollerweise genauso viele wie vom Ausweis).

Mit den Kopien und dem Ausweis begibt man sich zu der Veranstaltung (Keysigning-Party) und macht mit jedem der anwesenden Notare das gleiche Prozedere:

1. Original Ausweis vorzeigen
2. Ausweiskopie abgeben
3. Thawte bzw. CACert Formular vor Ort unterschreiben und abgeben

Hiernach ist das eigene Zertifikat dann durch mindestens 2 Notare beglaubigt (um über die entsprechenden 50 Points zu kommen), die bestätigen, das man selbst auch der ist, den man vorgibt zu sein. Danach ist es nur noch notwendig z.B. im E-mail Programm (Mozilla Thunderbird, MS-Outlook, Kmail, etc.) das entsprechende Zertifikat einzubinden und man ist seinem gegenüber authentifiziert und kann hiernach seine E-mails signieren. Dies ist also vom Ablauf her analog zu einem beglaubigten Dokument (Zeugnis, etc.) welches man bei einem Notar beglaubigen lassen kann.

Key-Signing-Partys

Ziel einer solchen Veranstaltung ist es einer möglichst großen Zahl von Personen die Möglichkeit zu geben Ihre Public-Keys (und natürlich auch die oben angegebenen Zertifikate) entsprechend auszutauschen und sich gegenseitig am besten anhand eines Personalausweises zu *authentifizieren* und die Daten zu *verifizieren*. Mit jeder Veranstaltung dieser Art wächst das sog. Datennetz des Vertrauens (Web-of-Trust).

Planung und Durchführung von Keysigning-Partys:

- Len Sassaman: <http://sion.quickie.net/keysigning.txt>
- Peter Palfrader: <http://www.palfrader.org/ksp-It2k4.html>
- Alex Brennen: <http://alfie.ist.org/projects/gpg-party/gpg-party.de.html>

Was ist bei einer solchen Keysigning-Party zu beachten?!

Eigentlich nicht viel. Lediglich der vorher erstellte GPG Key bzw. die ausgefüllten und Unterschriebenen Formulare anderer Anbieter (thawte, CACert, etc.) sind nebst einem Stapel seines kopierten Ausweises mitzubringen. Der Ausweis selber ist natürlich auch mitzuführen und bei Verlangen vorzuzeigen.

Noch mal die **Checkliste für eine Keysigning Party:**

1.	GPG / PGP-PublicKey in ausgedruckter Form (Papierstreifen) z.B.: Henrik Heigl < h@ivamp.de > Key ID: 0x042B3EE5 Key fingerprint = 1847 C70A 88ED 8C73 E866 F607 33B5 B05C 042B 3EE5	<u>Check</u>
2.	Ggf. Ausgefülltes und unterschriebenes Thawte Formular	
3.	Ggf. Ausgefülltes und unterschriebenes CACert Formular	
4.	Ausweiskopien in ausreichender Menge (mindestens soviele wie Keys und Zertifikate da sind)	
5.	Personalausweis zum vorzeigen	

Keyserver

Um Schlüssel auch Abseits einer Keysigning Party auszutauschen bzw. einen entsprechenden Public Key eines anderen Benutzers abzufragen sind im Internet sog. Keyserver, die sich gegenseitig abgleichen, damit ein in sich konsistenter Datenbestand vorherrscht.

Zur Abfrage und Ablage von öffentlichen Schlüsseln (public-key) empfiehlt sich einen der folgenden Keyserver zu verwenden

- subkeys.pgp.net (Port 11370)
- andom.sks.keyserver.penguin.de (Port 11370)
- <http://keyserver.noreply.org> (Port 80)
- <http://keyserver.mine.ru> (Port 80)

Secure Messaging

Auch Messenger bzw. Messaging Systeme wie ICQ, Yahoo Messenger, Microsoft Messenger, AIM, IRC, etc. haben diverse Verfahren zur verschlüsselten Kommunikation.

Wie man z.B. Freies & sicheres IM mit Jabber & OpenPGP betreibt ist unter <http://kai.iks-jena.de/jabber/index.html> beschrieben. Unter <http://securspot.com/howto/encryption/chat/psi/> wird beschrieben wie man Jabber mit dem Jabber-Client Psi (<http://psi.affinix.com/>) mit GnuPG „verheiratet“.

Eine etwas detailliertere Aufstellung findet man z.B. unter <http://web.swissjabber.ch/docs/psi/index.shtml> und unter <http://www.jabber.org/user/publicservers.php> sind noch entsprechende Server angegeben, die auch Gateways zu ICQ, AIM und anderen bereitstellen.

GnuPG mit Jabber

Jabber (<http://www.jabber.org/>) ist ebenso wie gaim (<http://gaim.sf.net/>) für IM-Clients sowas wie Firefox für Browser. Als XML-basierender Service ist Jabber ein offenes Open-Source-Instant-Messaging-System. Die Clients sind klein und schnell, man hat eine große Auswahl an Clients und Servern. Und außerdem, womit wir wieder beim Thema wären: Jabber unterstützt SSL- und GnuPG-Verschlüsselung.

Es gibt diverse Clients für Windows (ALP - <http://uckan.info/text/verschluesst-jabbern-mit-psi.php>) oder mac OS (Adium X - <http://www.adiumx.com/>)

Weitere infos auch unter http://www.web-blog.net/comments/P169_0_1_0/

Verschlüsselt Jabbern mit Psi.- <http://uckan.info/text/verschluesst-jabbern-mit-psi.php>
Gnupg unter Windowseinfach - <http://weblog.plasticthinking.org/item/2004/11/10/gnupg-unter-windows-verschlusssung-fast-ganz-einfach>

Kryptographische Verfahren

- Symmetrische Verschlüsselung (derselbe Schlüssel; Algorithmus z.B. AES, IDEA, DES)
- Asymmetrische Verschlüsselung (Public / Private Key Verschlüsselung; Algorithmen: RSA, DSA)
- Hybride Verfahren (Symmetrischer Key wird berechnet und mit asymmetrischen Verfahren verschlüsselt übertragen)

mehr dazu z.B. unter

<http://www.informatik.tu-darmstadt.de/TI/Lehre/SS04/Seminar/PKI.html>

Kurzanleitung GPG:

Key erstellen:

```
gpg --gen-key
```

Signierten Key auf File importieren:

```
gpg --import [Datei]
```

z.B. `gpg --import 00112233.asc`

Alle Keys im Schlüsselring anzeigen:

```
gpg --list-keys
```

Alle Keys anzeigen, die z.B. foo in der Adresse oder im Kommentar enthalten:

```
gpg --list-keys foo
```

Eine speziellen Key anzeigen lassen:

```
gpg --list-keys -v 00112233
```

Auf dem Keyserver einen Schlüssel in Volltextsuche suchen:

```
gpg --search-keys foo
```

Key vom Server holen bzw. updaten:

```
gpg --recv-key 00112233
```

alle Fingerprint aus dem Schlüsselring anzeigen:

```
gpg --fingerprint
```

Einen Fingerprint anzeigen lassen:

```
gpg --fingerprint 00112233
```

Key signieren:

```
gpg --sign-key 00112233
```

exportieren:

```
gpg --export -a 00112233 > 00112233.asc
```

Key Hochladen:

```
gpg --send-key 00112233
```

Key bearbeiten (z.B. Trustlevel einstellen):

```
gpg --edit-key 00112233
```

Alle Keys Updaten:

```
gpg --refresh-keys
```

verschlüsseln:

```
gpg --encrypt Empfänger [Datei]
```

weiterführende Dokumente

- Gnupg FAQ : [http://www.gnupg.org/\(de\)/documentation/faqs.html](http://www.gnupg.org/(de)/documentation/faqs.html)
- Gnupg-MiniHowTo: http://www.dewinter.com/gnupg_howto/german/GPGMiniHowto.html
- Kryptographiewörterbuch: <http://www.cryptnet.net/fdp/crypto/crypto-dict.html>
- Handbook of Applied Cryptography: <http://cacr.math.uwaterloo.ca/hac/>
- BSI für Bürger
Bundesamt für Sicherheit in der Informationstechnik
(Datensicherung, Viren, Dialer, Verschlüsselung, etc. erklärt für Laien)
<http://www.bsi-fuer-buerger.de/schuetzen/>
- Deutsche GnuPG- und PGP-Anleitung: <http://kai.iks-jena.de/pgp/>
- Einrichtung von GnuPG und Enigmail mit Thunderbird: <http://php.ch-becker.de/how2/winpt/>
- öffentlicher Keyserver mit Interface: <http://pgp.mit.edu/> oder <http://www.keyserver.net/en/>

Bücher:

Applied Cryptography
Schneier, Bruce
Verlag Wiley & Sons
ISBN 0471128457

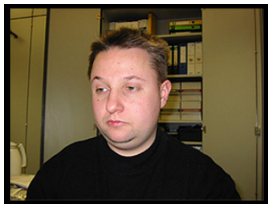
Kryptografie und Public-Key-Infrastrukturen im Internet
Schmeh, Klaus
Dpunkt Verlag
ISBN 3932588908

IT-Sicherheit, Studienausgabe
Eckert, Claudia
Verlag Oldenbourg
ISBN: 3486576763

Internet-Sicherheit - Internet, LAN und WLAN schützen, Hacker abwehren und sicher surfen, mit DVD
Otto, Alexander
Alileo Press
ISBN: 3898425541

Info und Kontakt:

Danke zuerst einmal an die LUG Frankfurt, LUG-Darmstadt, Christoph Rummel, Karlheinz Geyer, Volker Weber und natürlich auch allen anderen, die mir geholfen haben dieses Dokument (wissentlich und unwissentlich ;-)) zu erstellen. Dieses Dokument ist u.a. unter Zuhilfenahme der hier angegebenen Quellen erstellt worden.



Henrik Heigl <h@ivamp.de>

Key ID: 0x042B3EE5

Key fingerprint = 1847 C70A 88ED 8C73 E866 F607 33B5 B05C 042B 3EE5

