

Secure Messaging for the Masses



Henrik Heigl
h@ ivamp. de

Key ID: 0x042B3EE5

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



Grundlagen



- **Sichere Daten-Kommunikation: Was gehört dazu?**
- **Authentizität**
Wie stelle ich sicher, daß die Nachricht, die ich erhalte, wirklich von der Person kommt, von der ich es glaube, das Sie es ist?
- **Integrität**
Wie stelle ich sicher, daß die Nachricht, nachdem sie abgesandt wurde, von niemandem verändert wurde?

Grundlagen /2



- **Verbindlichkeit**
sicherstellen, daß der Urheber und Absender einer Nachricht nachträglich nicht bestreiten kann, diese Nachricht verfasst zu haben
- **Vertraulichkeit**
Wie stelle ich sicher, dass die Nachricht nur vom Empfänger und keinem dritten gelesen wird?

Grundlagen /3

- **Lösungsansatz**
 - Verschlüsselung
 - Digitale Unterschrift (Signatur)
- **Voraussetzungen**
 - Jeder TeilnehmerIn hat ein Schlüsselpaar
 - einen öffentlichen Schlüssel
 - einen privaten Schlüssel

Mails verschlüsseln und signieren

- **Versenden einer verschlüsselten Mail**
 - Der Absender *verschlüsselt* die Mail mit dem öffentlichen Schlüssel des Empfängers
 - Der Absender versendet die Mail
 - Der Empfänger *entschlüsselt* die Mail mit seinem privaten Schlüssel

 - *Fazit: Die Bedingung der Vertraulichkeit ist erfüllt, alle anderen Bedingungen sind nicht erfüllt.*

Mails verschlüsseln und signieren /2

- **Versenden einer signierten Mail**
 - Der Absender signiert die Mail mit seinem privaten Schlüssel
 - Der Absender versendet die Mail
 - Der Empfänger prüft die Echtheit der Signatur mit dem öffentlichen Schlüssel des Absenders

 - *Fazit: Die Bedingungen Integrität, Authentizität und Verbindlichkeit sind erfüllt. Vertraulichkeit nicht!*

Mails verschlüsseln und signieren /3

Empfangen einer signierten Mail

- Der Empfänger prüft die Echtheit der Signatur mit dem öffentlichen Schlüssel des Absenders

- *Fazit: Integrität, Authentizität und Verbindlichkeit sind erfüllt. Vertraulichkeit nicht.*

Mails verschlüsseln und signieren /4

Versenden einer signierten und verschlüsselten Mail

- Der Absender
 - signiert die Mail mit seinem privaten Schlüssel
 - verschlüsselt die Mail mit dem öffentlichen Schlüssel des Empfängers
 - sendet die Mail
- Der Empfänger
 - entschlüsselt die Mail mit seinem privaten Schlüssel
 - prüft die Echtheit der Signatur mit dem öffentlichen Schlüssel des Absenders
 - *Fazit: Alle 4 Kriterien sind erfüllt.*

Schlüsselaustausch

Wie werden Schlüssel ausgetauscht?

- direkte persönliche Übergabe
- Brief, Fax oder Telefon
- Key-Server (Public Keyserver) WoT
- Email

Vertrauen

Vertrauen zu Schlüsseln

- Nur zu Schlüsseln
 - die man persönlich empfangen hat
 - die man per Email oder über einen Key-Server empfangen hat und deren Korrektheit man telefonisch, per Brief oder Fax überprüft hat
- Misstrauen bei Schlüsseln
 - die man nicht persönlich vom Besitzer empfangen hat
 - die man per Email oder über Key-Server erhält.

Vertrauen /2

- Beispiel :
 - Eva sendet ihren öffentlichen Schlüssel an Adam.
 - Karl fängt die Nachricht ab. Er erzeugt ein neues Schlüsselpaar mit dem Namen und der Email-Adresse von Eva. Den öffentlichen Schlüssel sendet er an Adam mit dem Absender von Eva.
 - Adam nimmt den öffentlichen Schlüssel von Eva in sein Schlüsselbund auf und verschlüsselt von nun an Email an Eva mit diesem Schlüssel.
 - Karl fängt die Email ab, entschlüsselt sie mit dem von ihm angefertigten privaten Schlüssel, liest sie, verändert sie gegebenenfalls, verschlüsselt sie mit dem echten öffentlichen Schlüssel von Eva und sendet sie weiter an die Empfängerin
 - Eva empfängt die Mail und ist der Ansicht, dass sie von Adam kommt und niemand die Möglichkeit hatte sie zu lesen und verändern.

Auf diese Weise kann Karl jede Mail, die zwischen Adam und Eva ausgetauscht wird lesen und verändern. Eva und Adam merken davon nichts.

Genauso verfährt Karl wenn Adam seinen öffentlichen Schlüssel an Eva übermittelt.

Überprüfung

Wie kann die Echtheit von Schlüsseln geprüft werden

- Fingerprint
- Signatur von Schlüsseln
 - Adam kennt Eva und hat den Schlüssel von ihr persönlich erhalten, deswegen signiert Adam den Schlüssel von Eva und stuft ihn als voll vertrauenswürdig ein.
 - Karl kennt Adam aber nicht Eva. Weil Adam Eva's Schlüssel signiert hat, akzeptiert er den Schlüssel von Eva, stuft ihn aber nur zu 50% als vertrauenswürdig ein, da er den Schlüssel nicht von Eva persönlich erhalten hat.
 - Keysigning Partys
 - CAs

Durch die Signatur der Schlüssel für Personen, von denen man den Schlüssel persönlich erhalten hat, entsteht das *Web of Trust*.

Schlüssel von Personen, die man nicht persönlich kennt und deren Schlüssel man lediglich akzeptiert, weil sie von anderen signiert wurden, sollten nicht als **voll** vertrauenswürdig signiert werden.

Öffentlichen Schlüssel signieren

Wie wird der öffentliche Schlüssel einer anderen Person signiert ?

- Der fremde öffentliche Schlüssel wird mit dem eigenen privaten Schlüssel signiert
- Der signierte Schlüssel wird an den Besitzer zurückgesandt
- Der Besitzer fügt diese Signatur seinem öffentlichen Schlüssel hinzu
- Weitere Infosz.B. : <http://www.ivamp.de/sig.pdf>

Weitere Tipps und Regeln

Einige Regeln zum Umgang mit dem Schlüsselpaar

- Privaten Schlüssel nicht auf der eigenen Festplatte speichern, sondern auf einer Diskette
- Als Passphrase sollten sichere Passworte gewählt werden (strong)
- Wird die Passphrase vergessen, kann verschlüsselte Email nicht mehr gelesen werden. Der öffentliche Schlüssel muss also für ungültig erklärt werden (revoke). Um ihn zu widerrufen wird ein Schlüssel-Rückruf-Zertifikat benötigt. Dies kann nur mit Kenntnis der Passphrase erstellt werden, deswegen ist es sinnvoll ein solches Zertifikat zu erstellen, bevor der eigene öffentliche Schlüssel bekannt gemacht wird.
- Den eigenen öffentlichen Schlüssel signieren bevor er weitergegeben wird (selfsig).

PGP / GnuPG

- PGP bietet die folgenden Funktionen:
 - Verschlüsseln von Email und Dateien
 - Digitale Signatur
 - Sicheres Löschen von Dateien
 - Erzeugen von Schlüsselpaaren
 - Management eigener und fremder Schlüssel
- PGP verwendet folgende Algorithmen
 - RSA, El Gamal oder DSS - ein Public-Key-Verfahren oder IDEA, 3DES oder CAST-128 - symmetrische Verschlüsselungsverfahren mit einem 128 Bit Schlüssel
 - MD5 - zur Erzeugung eines - möglichst - eindeutigen 128 Bit langen Hash-Wertes
- Deutsches online-PGP-Handbuch von Christopher Creutzig, Andreas Buhl, Phil Zimmermann: <http://www.foebud.org/pgp/>
- Freies Equivalent ist GnuPG (Gnu privacy Guard) oder kurz GPG
 - Unter GPL und nach OpenPGP Standards veröffentlicht
 - Ist voll PGP kompatibel
- Viele freie Implementierungen für GnuPG
 - Enigmail (Mozilla Thunderbird Plugin); Plattformübergreifend
 - WinPT – Windows GUI



X.509 E-Mail Zertifikate

- **Thawte** – www.thawte.com
 - Kommerzieller Anbieter, aber freie Mail Zertifikate
 - Web-of-Trust
 - Freies Mailzertifikat wird per Weboberfläche angefordert, in den Mailclient eingebaut und kann von dort aus in eine pkcs Datei exportiert werden (Einbau in andere Programme)
 - Notarys beglaubigen die Authentizität (geringe Leveltiefe)
 - Weitere Infos: <http://www.ivamp.de/thawte/>



- **CACert** – www.cacert.com
 - Ähnlich wie Thawte, rein privates Projekt
 - Abruf und Einbindung etwas gewöhnungsbedürftiger, aber vom Prinzip her dasselbe
 - Notare werden ähnlich wie bei Thawte ernannt



X.509 E-Mail Zertifikate /2



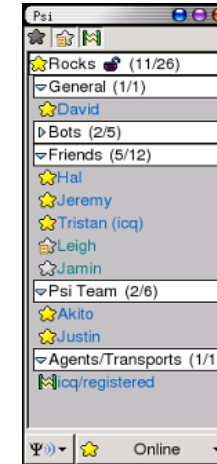
- **Verisign** – www.verisign.com
 - Plattformen: Alle, teils Plattformunabhängig (Token)
 - Kommerzielle Zertifikate mit einer zentralen Vergabe und Kontrollstelle
 - Diverse Mehrdienste (Tokens, mobile Security, etc.)



- **Ciphire** – <https://www.ciphirebeta.com/cm.html>
 - Plattformen: Windows, Linux, Mac
 - Programm wird mailclientunabhängig installiert (kein X.509 Zertifikat)
 - Ein Ciphre-Zertifikat wird per Programm auf dem Ciphre Server angefordert und dort verwaltet.
 - Da z.Zt. Noch Beta gibt es auch noch keine genaueren Angaben

Messenger

- **PSI** - <http://psi.affinix.com/>
 - Plattformen: alle
 - Multimessenger mit GNUPG Unterstützung und Jabber Client



- **Jabber** - <http://www.jabber.org/>
 - Plattformen: Windows, Linux, Mac; diverse Client und Serverimplementierungen
 - Secure Messaging basierent auf XMPP Standard
 - Jabber kann mehr: <http://jabber-fanatix.org>



- **Trillian** - <http://www.ceruleanstudios.com>
 - Plattform: Windows
 - Secure mit eigenem Algorithmus (wenn Gegenstelle auch Trillian hat)



Ausblick und Diskussion

- Nur wenn man mitmacht verbreitet es sich
 - PGP-Schlüssel
 - Mail Signaturen und Zertifikate
- Jeder ist für seine Sicherheit selbst verantwortlich
- Der gläserne User ?!

Weiterführendes ...

- **Signaturgesetz:**
http://bundesrecht.juris.de/bundesrecht/sigg_2001/
<http://www.netlaw.de/gesetze/sigg.htm>
- **Kleine Secure-Mail und –Messaging FAQ**
<http://www.ivamp.de/sig.pdf>
- **Weitere Projekte:**
 - NASLite Terabyteserver – <http://www.terabyteserver.de.ms>
 - WLAN Gruppe Rhein-Main – <http://www.hellfish-rm.de>