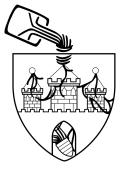
# "Meine Mails gehören mir!" Workshop zur E-Mail-Verschlüsselung

Peter Gewald & Manuel Groß

2015-11-28

#### Wer sind wir?



https://ccc-ol.de/



https://mainframe.io/

# Was ist eigentlich unser Problem?



CC-BY-SA 2.0 Markus Winkler

- Massenhafte, anlasslose Überwachung
- False Positives
- Chilling Effect

#### Sicherheitsziele

- Verschlüsselung
  - ► Text kann nicht von anderen gelesen werden
- Authentizität
  - ► Identität des Absenders sichergestellt
- Integrität
  - ► Nachricht nicht von dritten verändert

### Lösungsvorschlag: Verschlüsselung

----BEGIN PGP MESSAGE-----Version: GnuPG v2

hOTMA9vhUHWYLEX1A0/+07en6i6B00h91Ev02Zu/kEKXHW09GGNeXiPVtdGKMiUv FJwfZb7lL6hrpwNZJM8+iYWwKD7nv+nP4CHV5UlG0BLhMLMqPVM440TmR9APaiBf qUrUs8UzQ2syQCC52EUYGmwkRkjQofIVSsJj99TOwXt0M20zAdv1MQUH54HqFRRB Ou0g90437kPW8nBX226vWl./Sc6O0m2nvHmmkNrn+0B+sNimsVaWm+Cp5Y0p8k7vv aAdbuvRDddVVUK0zhNN+0PXiNolAzXU0eaBGnVhWL50aYZiCvat51HIRu+lURU0z nzdliZyssDis2rN/iVj8RcDWrpTmiKuKQAtKXIstLziYrQh6+2Nf37IztLjbIRkJ GkTuZSMAwJurlpDzWrik3t+sxXqF9drOxH1KSE9sM3EPZCYomdTdBVTep67iO0mI 5K8Txd04Tack8VHbVRd5IBJy8lVVnS8UyCdV5yjjx+KMxfZxUPlX9Ym0mB/jh06p d/eLU9TwomWNZtk7I8hVpVpmm8BvT+fbwTi33aUCcCDa8ic7vC6FsA3kiig7aDDL r3n0CR19LPibdvAMliHDddxOae4wbU2hN5aBFEiHBCOJPceI9euPrZmfzIMO3ou3 IP6cZBGLTawpHpz7NSzYBCAgKAXlQqXG2ptvVVL1Z96KhSxhZZ/LQc0MdySoYU0F AgwDvvGd00Bv5bYBD/0b5R7JBgt6aFN6MvnPr0XtlgEU4n9n+FRCvXVp03elCwlv ZGJ80iAa0SP00m0ECq+05Cs0JMPzmXq08Km7aKizY0nDv6ZAAD2VEbSfZkue0k03 bIbfbyeANOAlG80JTWsJSA2mPLXvD+E856jb6xVZpoAmrJjFfiSu2/awzw8giygH mVB+2vEUxcl8ollUdaPFketiEfWE00nAmYo2hT8WEqvviNNYT7pWll0NHU8kXZF3 j8ZyD377MNO4sR0il3f557L+e+zDJw6tCuDVQZjmeVihUijBlNNbwkZAC/+Dv8nS mJ9u9ZA9i2P0635SmZiseYWsphGhsI+2ngswtmvv2lYr9raX+PUYvG5nIAtD7CYr ge40nvHDFmFbIPrid6pzi+g6Gde0KCf0YJv+MlLtCi0INAsWFlvudT0crnR0ok0Z MXHval/Ez8XG0Mat4baZasVaOCL9CXanrxQWMG0h/dZaBZB0hkVGSXi/Fe5Bvlkz ggYCd5n5gJtT9e5RRELDGIBwaIJWV8TUXfhlOKBOsf5FT5zUr0ELSAzz+crkJkkh U+sloRzI471uNQoCbCehuv16sZ9niNlvrggtlcXvGVamZSMgH0lUxN6dNh+ASu9w E982fPxAWcMSJyUIkiHx1NTYZdDXRóNqVQ14JA6YFmjf/vl/4dQGbNOvkhIyNqsB 8RgAgArhykdvd/+aD9Xdu0zGZ0d2tUg7p40mIi/vPi3/m5VaWiuUAUcdG0008bkW

- Klartext → Geheimtext
- mathematisches Verfahren
- geheime Zahl als "Passwort"
- symmetrisch
  - selbes Verfahren
  - selber Schlüssel
- asymmetrisch
  - selbes Verfahren
  - unterschiedliche Schlüssel

# Ziel 1: Verschlüsselung (Abgrenzung)

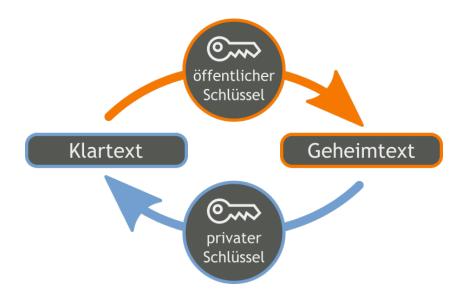
Was ist damit nicht gemeint?

- Webseiten (http(s))
  - → Ende-zu-Ende Verschlüsselung (Transportverschlüsselung)
- Messenger
  - $\rightarrow$  Verschieden: Transportverschlüsselung, manchmal Ende-zu-Ende Verschlüsselung
- Mail
  - → Per default unverschlüsselt
  - → Typisch zum Server, nicht Ende-zu-Ende!
- DE-Mail
  - per default nur abschnittweise verschlüsselt
- ⇒ Nachrichten am Speicherort ungeschützt

## Asymmetrische Verschlüsselung



## Asymmetrische Verschlüsselung



# Ziel 2: Authentizität (Vertrauen)

#### Problem:

Wer garantiert uns, dass Bob wirklich der Absender ist?

#### Ansätze:

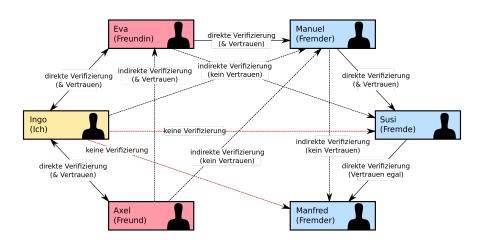
- a) Zentral
- b) Dezentral

# Ansatz a) Zentral (Server)

- Idee: zentraler Ansatz
  - ► Indirekter Austausch über Dritte (z.B. Browser oder Mailprogramm)
  - ► Hierarchie/Zertifikatsliste (schonmal reingeschaut?)



# Ansatz b) Dezentral: Vertrauensnetzwerk (Web of trust)



CC-BY-SA 3.0 Hauke Laging

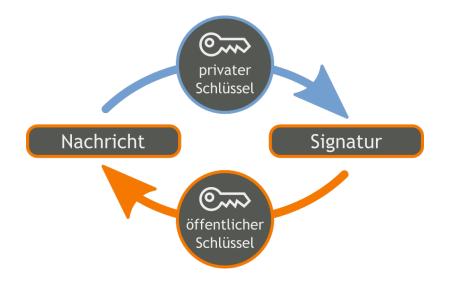
### Cryptoparties



CC-BY 2.0 wbritzl

- Neue Leute kennenlernen
- Spaß an lustigen, alten, Ausweisbilder anderer haben
- Selbst Cryptoparties veranstalten ⇒ Web of trust stärken

Ziel 2 und 3: Authentizität und Intigrität mittels Signatur



#### **GPG**

Was ist jetzt eigentlich GPG?



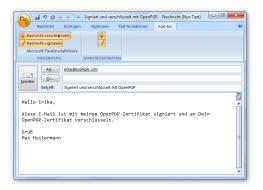
- "Gnu Privacy Guard", Open Source-Implementierung von OpenPGP
- ,,PGP" (,,Pretty Good Privacy"), alternative, nun proprietäre Implementierung
- Sehr verbreitet im Open Source Umfeld
- Eher ungebräuchlich im geschäftlichen Umfeld
- Web of Trust Prinzip
- Plugins für viele Clients
- Metadaten nicht verschlüsselt!

**GPG** 

• Demo: Mail (Source, Header)

### Beispiel: GPG in MS Outlook

- Gpg4win: OpenPGP und S/MIME
- MS Outlook 2003/2007/2010 und 2013 (32 Bit)
- Windows XP, Vista, 7 and 8
- https://www.gpg4win.de/doc/de/gpg4win-compendium\_11.html



http://www.gpg4win.org/img/sc-gpgol-sendSignEncryptedMail\_de.png

### Beispiel: GPG in Thunderbird

- Ebenfalls Gpg4win installlieren
- Zusätzliches Plugin für Thunderbird nötig: Enigmail
- Anleitung http://blog.ch-becker.de/2011/04/27/ pgpgpg-unter-windows-mit-thunderbird-emails-verschlusseln/



http://blog.ch-becker.de/wp-content/uploads/2011/04/enigmail\_suche.png

### Beispiel: GPG mit MAC OS X

• https://gpgtools.org/



https://gpgtools.org/images/screenshots/gka-create-key.1375965203.png

#### PGP funktioniert!

#### TOP SECRET//COMINT//REL TO USA, AUS

#### TOP SECRET//COMINT//REL TO USA, AUS//20320108

\*\*\*\*\*\*\*\*\*\*\*\*\*

THIS INFORMATION IS DERIVED FROM FAA
COLLECTION UNDER FAA COUNTERTERRORISM CERT

THIS INFORMATION IS PROVIDED FOR INTELLIGENCE PURPOSES IN AN EFFORT TO DEVELOP POTENTIAL LEADS. IT CANNOT BE USED IN AFFIDAVITS, COURT PROCEEDINGS OR SUBPOENAS, OR FOR OTHER LEGAL OR JUDICIAL PURPOSES.

### Vorgehen

#### Erstellen (einmalig):

- Schlüsselpaar erstellen
- Optional: Auf Schlüsselserver (keyserver) laden

#### Signieren:

- 3 Fremden Schlüssel (public key) laden
- Fingerprint prüfen/vergleichen
- $\bullet$  Identität der Person prüfen (Ausweis)  $\leftarrow$
- Schlüssel signieren
- Signieren Schlüssel auf Schlüsselserver laden oder per Mail schicken

### GPG unter Linux - Installation

#### **Debian Pakete:**

- gnupg2
- signing-party (für caff)
- msmtp

## GPG unter Linux - Konfiguration

- Config: ∼/.gnupg/gpg.conf
- keyserver pgp.mit.edu
- Vorteil: Kein manuelles --keyserver <keyserver>

GPG - Live

• Demo: Schlüssel erstellen, Signieren

## 1: GPG unter Linux - Schlüsselpaar erstellen

- \$ gpg --gen-key
  - ► RSA oder DSA?
  - ► Schlüssellänge?
  - ► Gültigkeit?
  - Name (kein Pseudonym)
  - Mailadresse
  - ▶ Passphrase (Langes\_Passwort > S4I4t)
- Upload (sofern öffentlich):
  - \$ gpg --send-keys <key-id>

# GPG unter Linux - Signieren

8

- Laden des fremden Keys:
  - \$ gpg --recv-keys <key-id>
- Prüfung des Fingerprints:
  - \$ gpg --fingerprint <key-id>
- Identitätsprüfung (Personalausweis, Führerschein)
- Signieren:
  - \$ gpg --sign-key <key-id>
- Passphrase eingeben
- Upload-Varianten:
  - Unsicher Direkter Upload auf Keyserver Sicher Verschicken des signierten Keys per Mail an die signierten Mailadressen

# GPG unter Linux - Signieren mit caff

- Automatisch:
  - \$ caff <key-id>
- Benötigt konfigurierten SMTP-client
- Beispiel msmtp
- Config: ∼/.msmtprc
  - ► host, port, user, passwort

# GPG unter Linux - Empfängerseite

- Datei signature.asc per Mail bekommen
- Signatur importieren:

```
gpg --import signatur.asc
```

• Signatur Uploaden:

```
gpg --send-keys <meine key-id>
```

Done! Ready for GPG-Mails!

### GPG Keys Revoken

- Entweder automatisch (Gültigkeit) oder manuell
- Wichtig: Vorher Cross-signieren
- Revoken ≠ Löschen
- Widerrufszertifikat:

- Importieren, Uploaden
- ggf. direkt nach Generierung erzeugen
- Besser: Backup private key!

#### **Fazit**

- Ende-zu-Ende Verschlüsselung als sicherer Kommunikationsweg
- Metadaten nicht verschlüsselt!
- Je mehr Nutzer, desto funktionaler!

# Hands-on - Let's go!

- Schlüsselpaar erstellen
  - ▶ gpg --gen-key
  - ► (keyserver definieren)
  - ▶ gpg --send-keys <key-id>
- 2 Signieren mit Identitätsprüfung (z.B. ohne caff)
  - ▶ gpg --recv-keys <key-id>
  - ▶ gpg --fingerprint <key-id>
  - ▶ gpg --sign-key <key-id>
- Signaturen per Mail verschicken
  - ▶ caff <key-id>
- Uploaden auf den Keyserver (Empfänger)
  - ▶ gpg --import signatur.asc
  - ▶ gpg --send-keys <meine key-id>