

Workshop – Secure Mail

shackaday

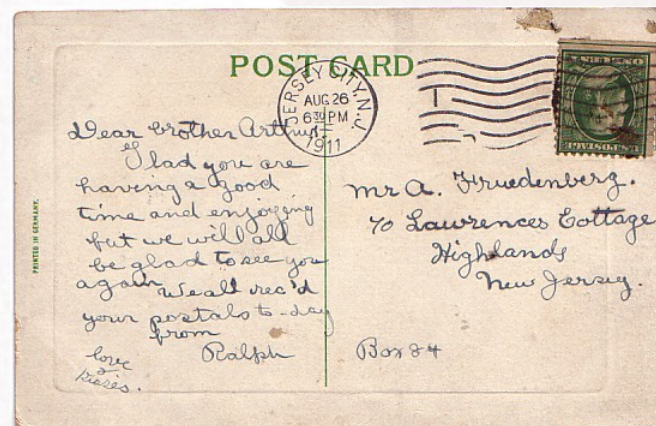
shack e.V.
Karlo Barta
2015-12-05

Version 1.2

Agenda

1. Motivation
2. Techniken
3. Schlüsselerwaltung
4. Web Of Trust
5. Clients einrichten
6. Keysigning-Party

Motivation



Motivation

- Neues kennenlernen
- Digitale Signaturen verwenden
- Informationen speichern/sendern
- Firmenintern nutzen
- u.v.m.

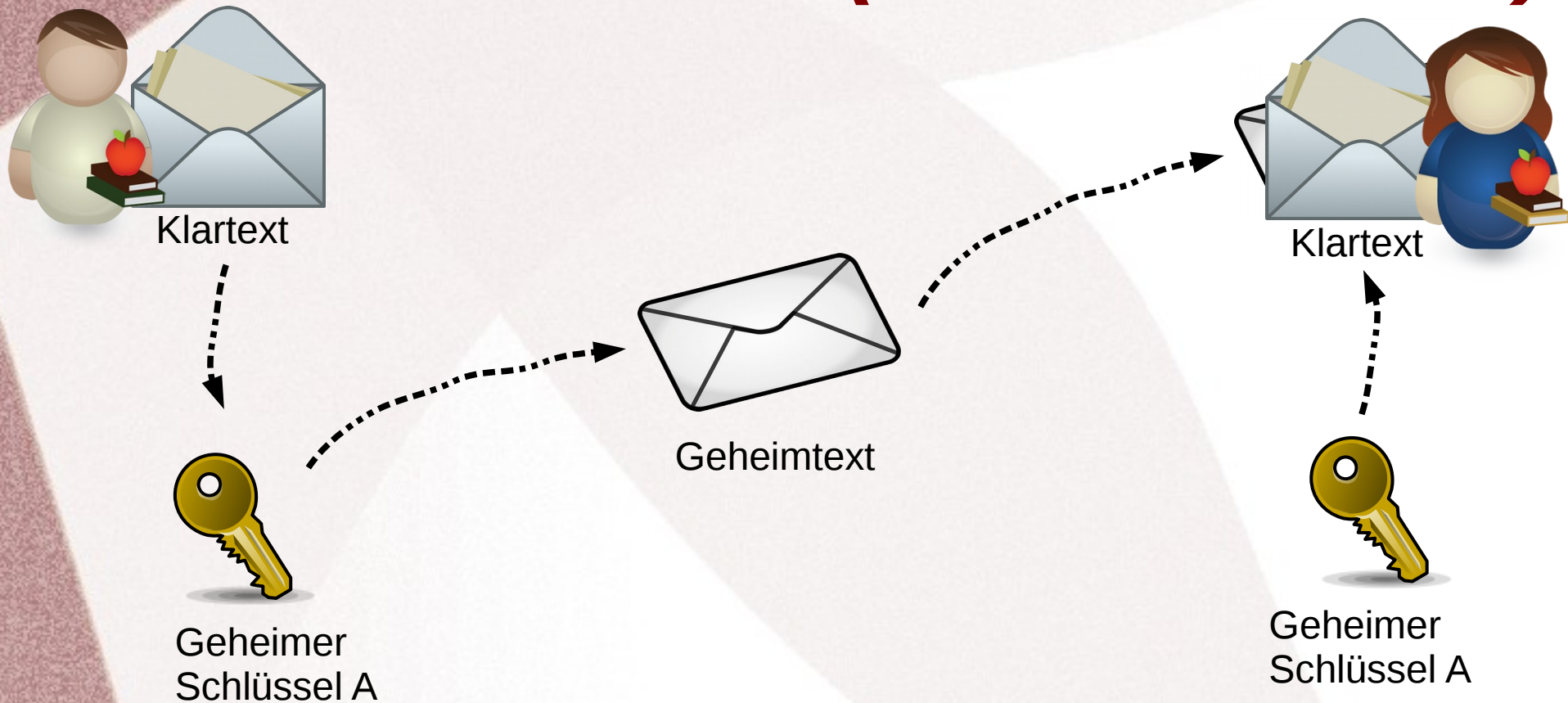
Techniken

- Symmetrisches Verfahren
 - AES, DES, Blowfish, ...
- Asymmetrisches Verfahren
 - RSA, ElGamal, DSA, ...
- Hash-Algorithmus
 - MD5, SHA1, RIPEMD160, ...

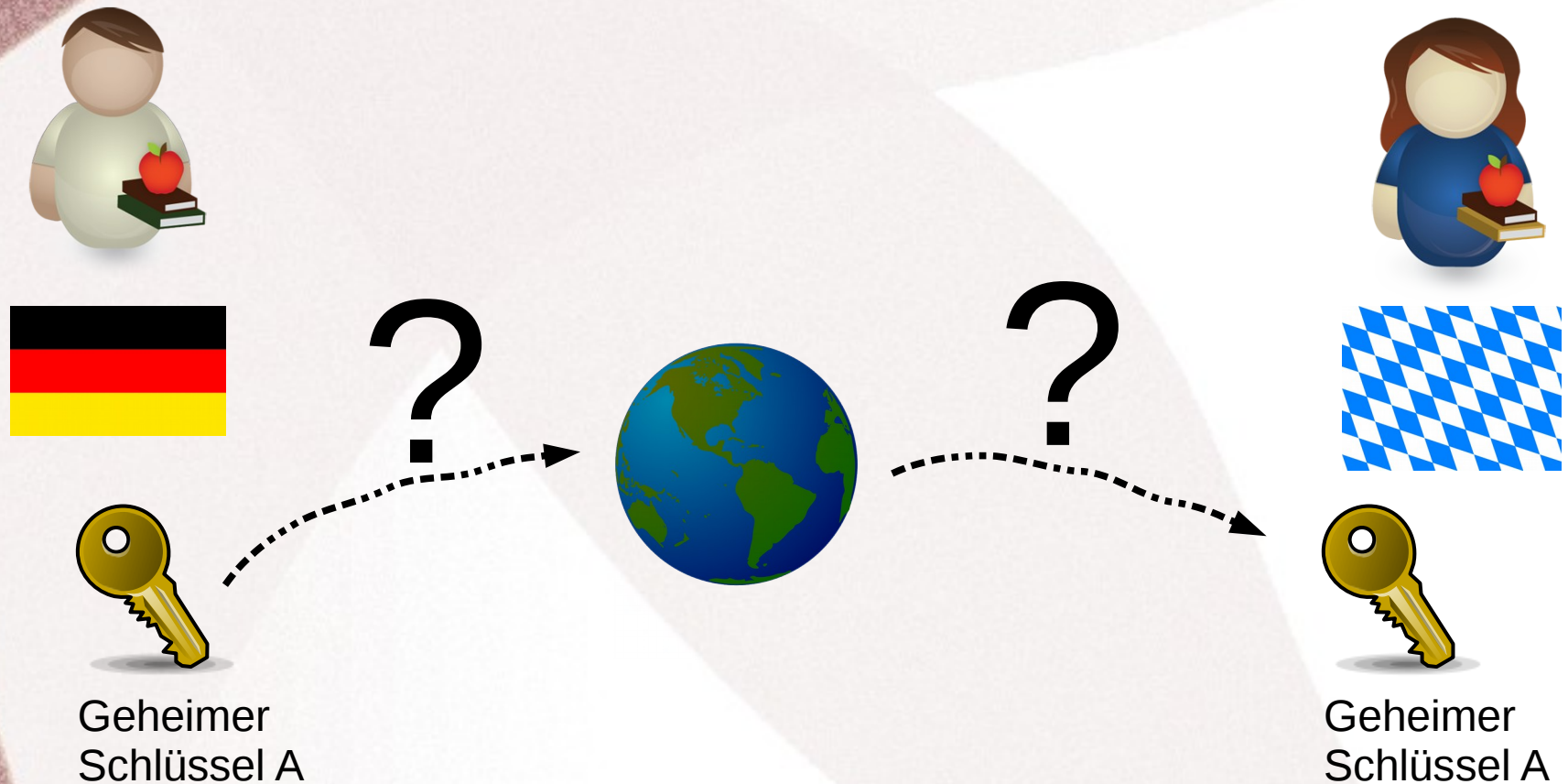
PGP und GnuPG

- PGP → proprietäre Software
- GnuPG → freie Software
- OpenPGP → Spezifikation (RFC 4880)

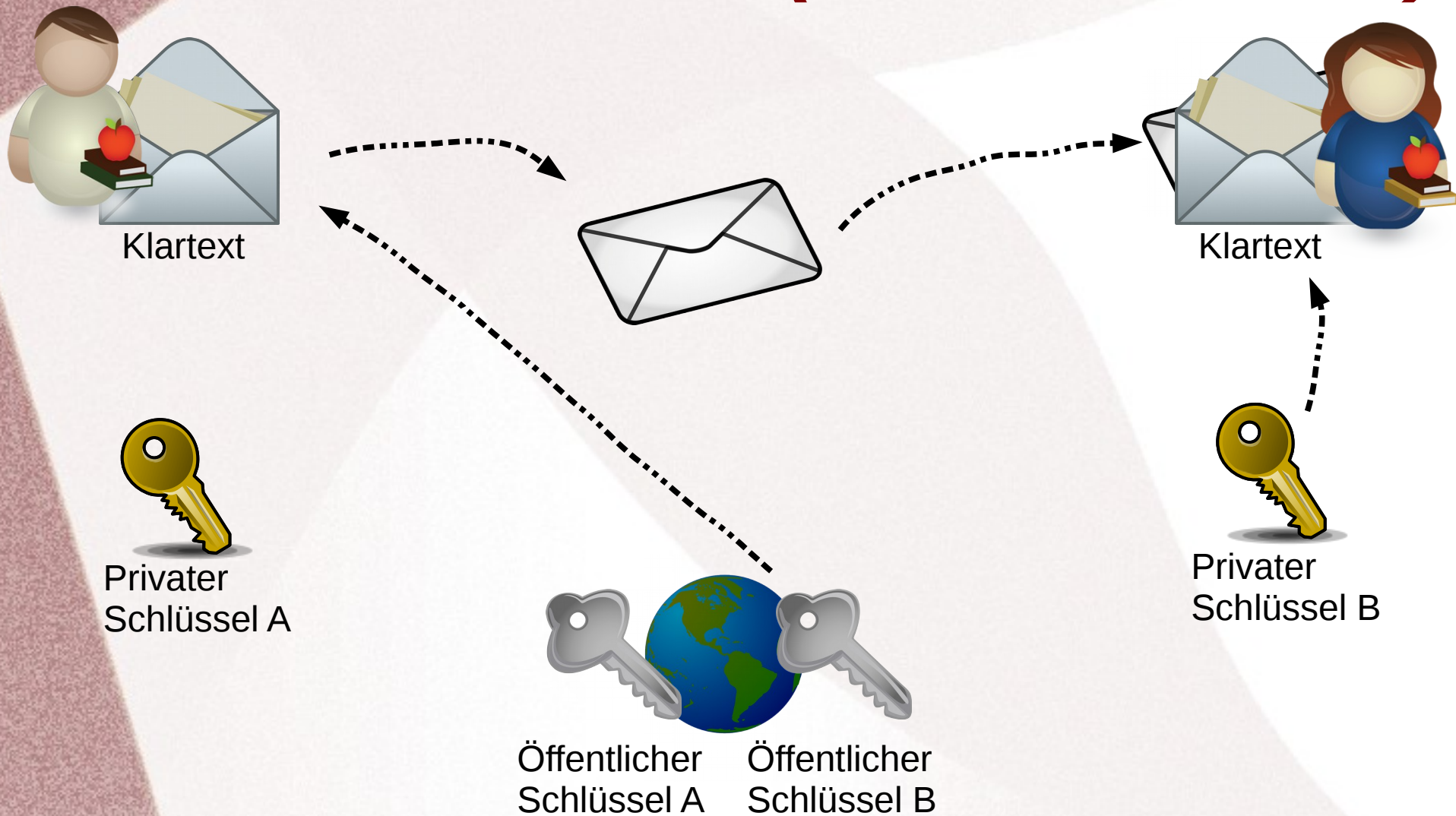
Symmetrisches Verfahren (verschlüsseln)



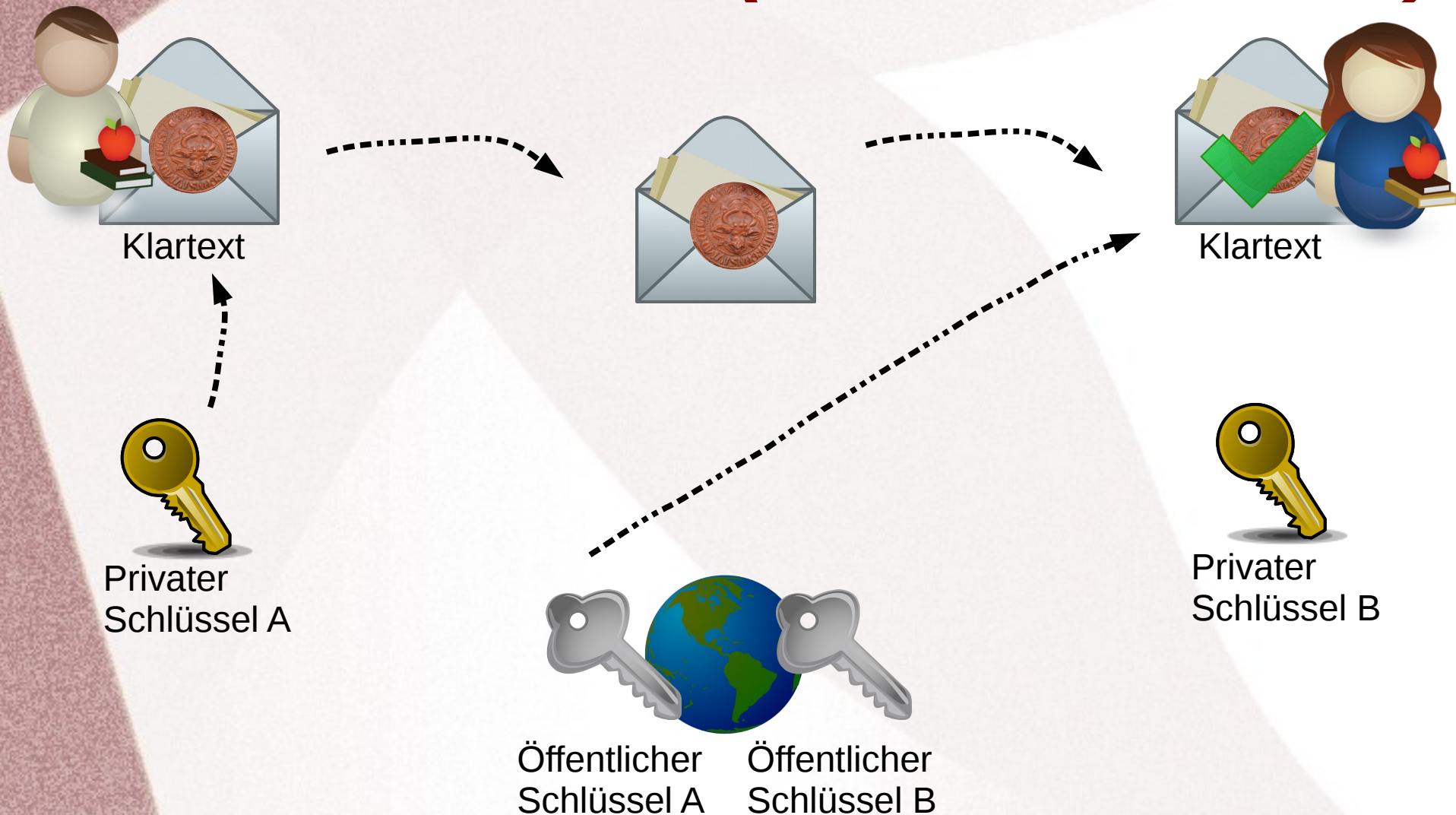
Asymmetrisches Verfahren



Asymmetrisches Verfahren (verschlüsseln)



Asymmetrisches Verfahren (unterschreiben)



Schlüsselverwaltung

Meine Schlüssel:



Schlüsselverwaltung

Meine Schlüssel:



Privater Schlüssel



- Schlüsselstärke
- Backup erstellen
- Nie verlieren
- Optional: Passphrase schützt

Schlüsselverwaltung

Meine Schlüssel:



Annulierer Schlüssel

- Backup erstellen
- Nie verlieren
- Annuliert öffent. Schlüssel
 - Auf Keyserver hochladen

Schlüsselverwaltung

Meine Schlüssel:



Öffentlicher Schlüssel

- Auf Keyserver hochladen
- Backup erstellen
- Vertrauensstufe
- Unterschriften
- Optional: Ablaufdatum festlegen

Web Of Trust

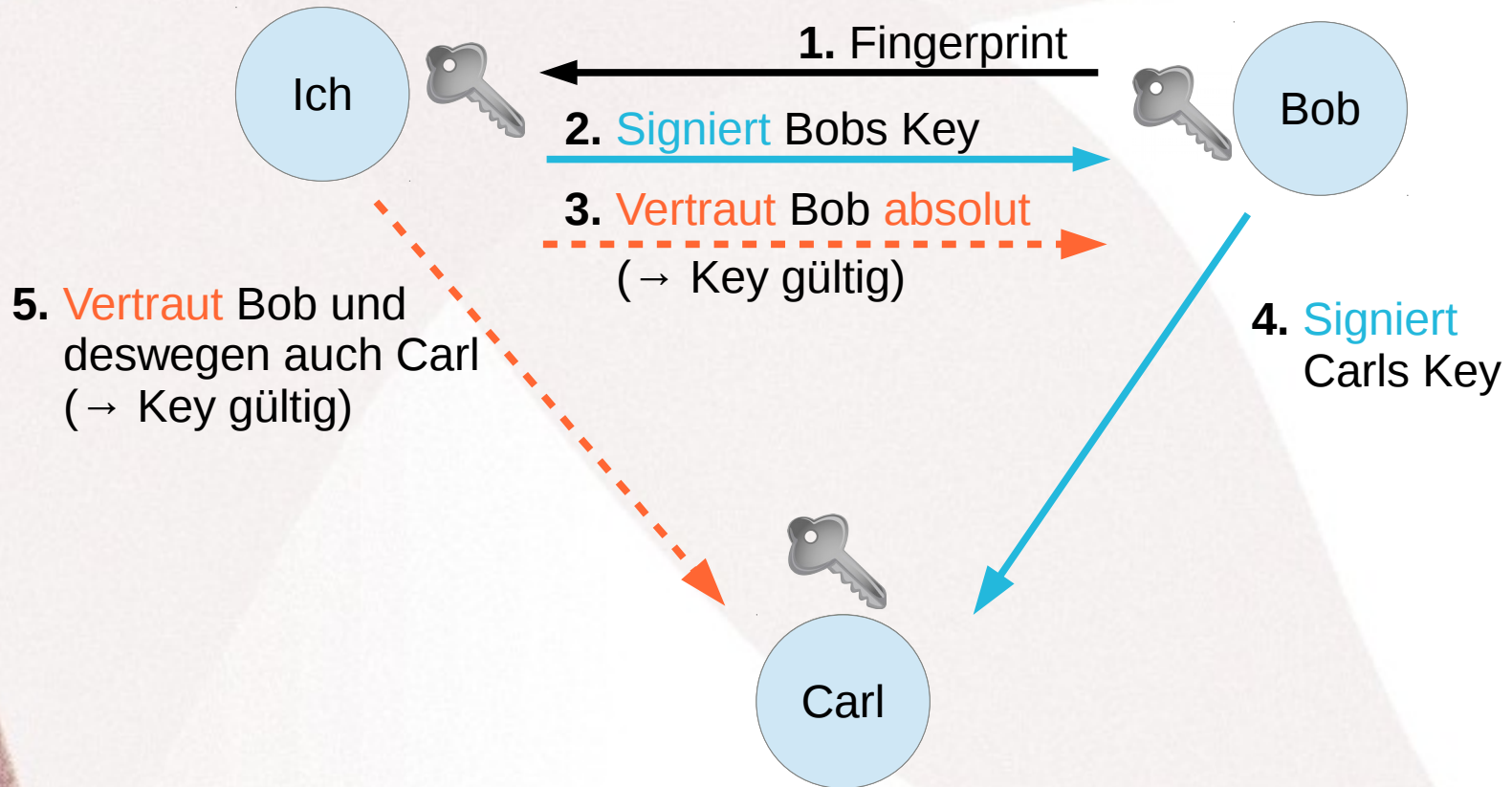
- **Signiere** öffentlichen Schlüssel von Person X
 - Gebe bekannt, dass ich den Schlüssel zur Person mit **Prüfgrad p** überprüfte
- **Beurteile** Schlüssel von Person X
 - Gebe bekannt, dass ich X bezüglich GnuPG-Erfahrungen o. -Kenntnisse vertraue (*Owner Trust*)

Web Of Trust – Grundlage

Disclaimer

- Der **Prüfgrad p** wird aus Gründen der Übersicht in den folgenden Beispielen weggelassen
- Ein Fremdschlüssel wird entweder signiert (= 1) oder nicht (= 0)

Web Of Trust – Grundlage



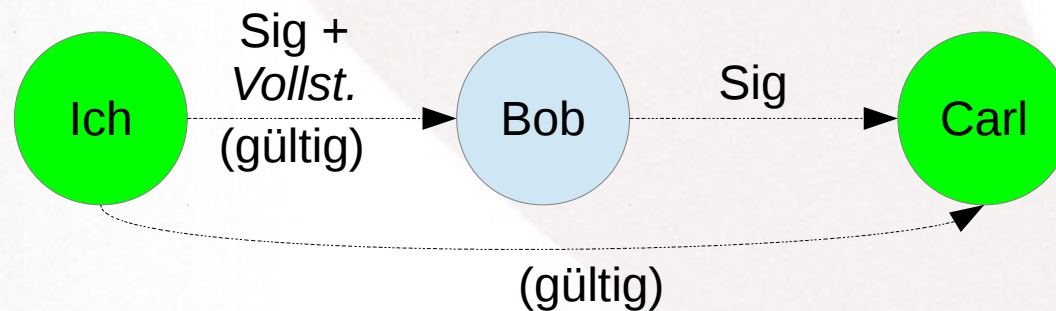
Web Of Trust – Vertrauen

Vertrauensstufen:

- Absolut (Ultimate)
 - Vollständig (Fully)
 - Marginal/Teilweise (Marginally)
 - Gar nicht (NOT trust)
 - Keine Ahnung (I don't know)
- **Vertrauen das Person X Keys korrekt und gewissenhaft überprüfen kann**

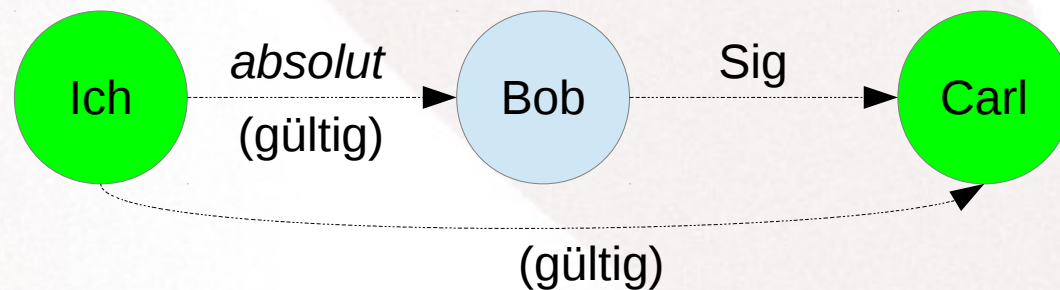
Web Of Trust – Policy

- Je nach *Owner Trust* ist das vertrauen vollständig, wenn...
 - ein Key 1x eigensigniert und Status vollständiges Vertrauen gesetzt ist
 - ein Key 1x fremdsigniert vom Vollständig-User



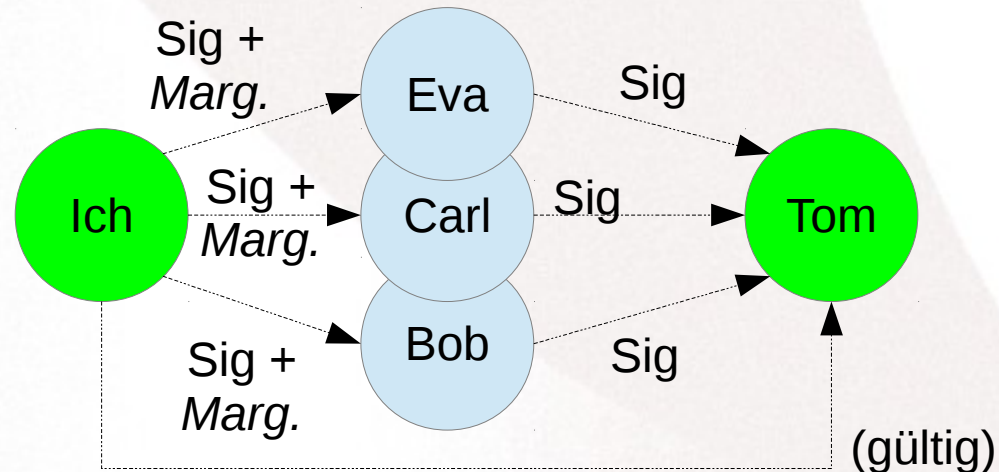
Web Of Trust – Policy

- Je nach *Owner Trust* ist das vertrauen vollständig, wenn...
 - der Status absolut ein Key erhält
 - ein Key 1x fremdsigniert vom Absolut-User



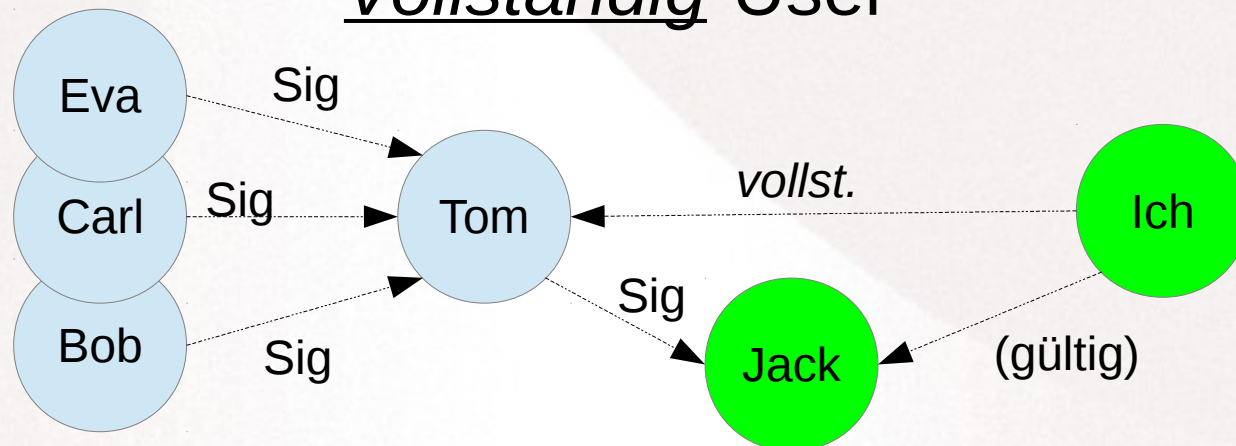
Web Of Trust – Policy

- Je nach *Owner Trust* ist das vertrauen vollständig, wenn...
 - unterschiedliche Keys 3x eigensigniert und marginales Vertrauen gesetzt ist
 - die drei Marginal-User gemeinsam einen Key fremdsignieren



Web Of Trust – Policy

- Je nach *Owner Trust* ist das vertrauen vollständig, wenn...
 - der Status vollständig ein Key erhält
 - drei Keys 1x fremdsigniert vom Vollständig-User
 - ein Key 1x fremdsigniert vom Vollständig-User



Web Of Trust – Policy

- Trust-Modell nachlesen (pgp, classic, direct, always, auto)
- Key Policy festgelegt
 - ggf. Policy im Web veröffentlichen
 - Beispiel
- Default-max-cert-depth 3 (5)
- Default-cert-level 0
 - Grad der Echtheitsprüfung bei der Signierung eines Keys

Revoked / Untrusted / Expired Keys

- Öffentliche Schlüssel kann vom Original-Eigentümer zurückgezogen werden
- Öffentliche Schlüssel kann noch nicht vom *Web Of Trust* bestätigt werden
- Öffentliche Schlüssel kann ablaufen

Mail-Clients einrichten



Keysigning Party

*„Trusting a key is not the same as trusting the key's owner“
(Phil Zimmermann)*

Schritte:

1. Infos vom Key vergleichen
2. *Owner Trust* / Nutzererfahrung setzen
3. *Signing Level* setzen
4. Öffentlichen Schlüssel hochladen

Keysigning Party

„Trusting a key is not the same as trusting the key's owner“
(Phil Zimmermann)

Schritte:

1. Infos vom Key vergleichen

Schlüsselmerkmal	vorliegender Schlüssel <i>(Infos vom Schlüssel)</i>	Angaben Kontakt <i>(Infos vom Owner selbst)</i>
Eigenzertifikat		
Benutzer-ID(s)		
Fingerprint		
Schlüssellänge		
Erstellungsdatum		
Ablaufdatum		

Keysigning Party

„Trusting a key is not the same as trusting the key's owner“
(Phil Zimmermann)

Schritte:

1. Infos vom Key vergleichen ✓
2. *Owner Trust* / Nutzererfahrung setzen ✓
3. *Signing Level* setzen ✓
4. Öffentlichen Schlüssel hochladen
(siehe nächste Folie)

Keyserver

- Liste:
 - wwwkeys.de.pgp.net
 - wwwkeys.eu.pgp.net
 - pgp-keyserver.de
 - pgpkeys.eu
 - keyserver.secretresearchfacility.com
- i.d.R. synchronisieren sich viele Keyserver untereinander

Keysigning Party

„Trusting a key is not the same as trusting the key's owner“
(Phil Zimmermann)

1. Infos vom Key vergleichen ✓
2. *Owner Trust* / Nutzererfahrung setzen ✓
3. *Signing Level* setzen ✓
4. Öffentlichen Schlüssel hochladen ✓

Agenda II (mehr)

1. Verschlüsselte Daten
2. Manipulation und Fälschung
3. Krypto-Präferenz
4. Offline-Key

Verschlüsselte Daten

- .asc → Transport armor file (GnuPG)
- .key → Public Key
- .gpg → Verschlüsselte Datei
- .sig → Signierte Datei

Daten – Verschlüsseln

CLI:

```
gpg --armor --recipient "Karlo Barta"  
--output "test.txt.asc" --encrypt "test.txt"
```

```
gpg --recipient "Karlo Barta" --encrypt  
test.txt
```

(Quelle: <http://wiki.kairaven.de/open/krypto/gpg/p/gpg9>, 2015-12-04)

Daten – Signieren

CLI:

`gpg --sign test.txt (binär)`

`gpg --clearsign test.txt (Text+Signatur)`

`gpg --armor --detach-sign test.txt (separat)`

(Quelle: <http://wiki.kairaven.de/open/krypto/gpg/p/gpg9>, 2015-12-04)

Daten – Überprüfen

CLI:

`gpg --verify test.txt.gpg` (binär)

`gpg test.txt.asc` (Text+Signatur)

`gpg --verify test.txt.asc test.txt` (separat)

(Quelle: <http://wiki.kairaven.de/open/krypto/gpg/p/gpg9>, 2015-12-04)

Manipulation und Fälschung

Manipulationen:

- Schlüssel mit falschen Benutzer-Ids und E-Mail-Adresse auf Key Server
- Auf Key Server den Originalschlüssel revoked durch gefälschen Schlüsselupload

→ **Eigenzertifikat vom Key prüfen**

(Quelle: <http://wiki.kairaven.de/open/krypto/gpg/p/gpg3>, 2015-12-04)

Manipulation und Fälschung

Fälschungen und Imitate:

- Schlüssel und Fingerprint fälschen
→ **Schlüssellänge prüfen**
- Angreiferschlüssel mit Unterschriften von imitierten Angreiferschlüssel anbringen, um Vertrauen zu erzielen
→ **?**

(Quelle: <http://wiki.kairaven.de/open/krypto/gpg/p/gpg3>, 2015-12-04)

Manipulation und Fälschung

MITM Angriff:

- Schlüsselaustausch von zwei Personen mit Angreifer öffentl. Schlüssel austauschen
- **zweiten sicheren Kanal für Schlüsselaustausch, bzw. Fingerprint nutzen (Telefon, Treffen, ...)**

(Quelle: <http://wiki.kairaven.de/open/krypto/gpg/p/gpg3>, 2015-12-04)

Krypto-Präferenz

Die Bevorzugung von Algorithmen durch die Reihenfolge in verschiedenen Listen:

- Verschlüsselung
- Hash
- Komprimierung

Krypto-Präferenz

Verschlüsselung:

- 3DES
- CAST5 (CAST-128)
- Blowfish
- Twofish
- AES/192/256
- Camellia128/192/256

Krypto-Präferenz

Hash:

- MD5: 128 bits (Rainbow-Tables)
- SHA1: 160 bits (NSA-Entwicklung)
- SHA2: 224/256/384/512 bits (NSA-Entwicklung)
- RIPEMD-160: 128/256/320/512 bits

Krypto-Präferenz

Komprimierung:

- nicht komprimiert
- ZIP
- ZLIB
- BZIP2

Krypto-Präferenz

- Bei lokaler Verschlüsselung mit eigenen Schlüsseln:
 - erste Stelle der Algorithmen-Liste
- Bei Verschlüsselung von Daten oder Kommunikationsinhalten mit fremden Schlüssel:
 - Abgleich der Algorithmen-Liste vom Schlüsselanwender und Schlüsseleigentümer

Krypto-Präferenz

Anzeigen:

- `gpg --version` (allgemein)
- `gpg --edit-key Schlüssel-ID`
 - `showpref` (eigene private Schlüsselreihenfolge)

Setzen:

- `gpg --edit-key Schlüssel-ID`
 - `setpref AES256 AES192 AES CAST5 RIPEMD160 SHA256 BZIP2 ZIP MDC`

Offline-Key

- Welche Sicherheitsmaßnahmen sind bei Offline-Hauptschlüssel zu achten?
 - Live-boot von CD
 - Offline-Key außerhalb von PC (USB-Stick)
 - ... (eigene Ideen)

Offline-Key – Erstellung

Quelle:

<http://www.openpgp-schulungen.de/inhalte/einrichtung/materialien/keygen-anleitung.html>
, 2015-12-04

Quelle:

<https://wiki.debian.org/Subkeys?action=fullsearch&context=180&value=master+keys&titlesearch=Titles>
, 2015-12-04