Übungen zu Systemsicherheit

Jürgen Kleinöder,
Michael Gernoth, Reinhard Tartler,

SYSSEC

A.1 Übersicht

- Termine
  - Mi. 12:30 - 14:00 (01.153)
  - (Fr. 12:30 - 14:00 (01.153))
  - Anmeldung über Waffel
    - https://waffel.informatik.uni-erlangen.de/

Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.

A.2 Überblick über 1. Übung

- ,,Sichere'' Kommunikation
  - Was bedeutet ,,sicher''
  - Vertraulichkeit
  - Authentizität
- PGP/GnuPG
- Web of trust
- ,,Keysigning-Party''

A.3 Sichere Kommunikation

- Emails werden (normalerweise) unverschlüsselt übertragen

```
220 faui45.informatik.uni-erlangen.de ESMTP spoken here
MAIL FROM:<gernoth@cs.fau.de>
250 2.1.0 Ok
RCPT TO:<tartler@cs.fau.de>
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
Subject: Geheim

Das geheime Codewort lautet: 62yxY^
.
```

- Sicherheit ähnlich einer Postkarte
- Ziel: End-zu-end Verschlüsselung (digitaler Briefumschlag)
- Mittel: Asymmetrische Verschlüsselung

Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.

A.4 "Sichere"- Email

- Heute genutzte Standards:
  - **OpenPGP**
  - S/MIME
- Integration in gängige Emailprogramme
  - Thunderbird (Enigmail)
  - Outlook (GPGol, siehe http://gpg4win.de)
  - KMail (nativ)
  - mutt (nativ)
  - Evolution (nativ)

A.5 GnuPG - Einführung

- PGP wurde 1991 von Phil Zimmermann veröffentlicht
  - Probleme mit US-Exportbeschränkung führten zu einscannen der in Buchform aus den USA exportierten Sourcen (HIP 1997)
  - 1997 wurde PGP von McAffe aufgekauft und kommerzialisiert
- GnuPG ist eine freie Alternative zu PGP
- Kommandozeilenprogramm zur:
  - Schlüsselerstellung
  - Schlüsserverwaltung
  - Verschlüsseln/Entschlüsseln
  - Signieren/Verifizieren

Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.

A.6 Schlüsselerstellung

- Schlüsselpaar auf der Kommandozeile erzeugen:

```
$ gpg --gen-key
Please select what kind of key you want:
Your selection? 1 (DSA and Elgamal)

DSA keypair will have 1024 bits.
What keysize do you want? (2048)

Please specify how long the key should be valid.
Key is valid for? (0) 2y
Key expires at Do 05 Nov 2009 12:45:35 CET

Is this correct? (y/N) y

Real name: Mickey Maus
Email address: simimaus@cip.informatik.uni-erlangen.de
Comment:
You selected this USER-ID:
    "Mickey Maus <simimaus@cip.informatik.uni-erlangen.de>"
[...]
pub   1024D/096DE85A 2007-11-06 [expires: 2009-11-05]
Key fingerprint = CBEA 5B96 069A 3E7B 5E43  71CA 148B 51DE 096D E85A
uid Mickey Maus <simimaus@cip.informatik.uni-erlangen.de>
sub   2048g/14237160 2007-11-06 [expires: 2009-11-05]
```


A.7 Verschlüsselung/Entschlüsselung

- Dateien können auf der Kommandozeile ver- und entschlüsselt werden
- Bei der Verschlüsselung wird nur der öffentliche Schlüssel benötigt
- Bei der Entschlüsselung wird der private Schlüssel benötigt
- Verschlüsseln einer geheimen Datei an M. Maus (Key-ID 0x096de85a)

```
$ gpg --recipient 096de85a --encrypt geheim.txt
$ ls geheim.txt.*
geheim.txt geheim.txt.gpg
```

- Entschlüsseln der Datei, wenn passender privater Schlüssel vorhanden

```
$ gpg --decrypt geheim.txt.gpg > out.txt
Enter passphrase:
```


Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.

A.8 Signaturen

- Signaturen dienen dem Nachweis der Urheberschaft
- Für die Generierung wird der private Schlüssel benötigt
- Bei der Überprüfung muss nur der öffentliche Schlüssel vorliegen
- Signieren einer Datei mit dem eigenen privaten Schlüssel

```
$ gpg --sign geheim.txt
Enter passphrase:
```

- Überprüfen der Signatur einer Datei

```
$ gpg --verify geheim.txt.gpg
gpg: Signature made Di 06 Nov 2007 13:14:00 CET using DSA key ID
096DE85A
gpg: Good signature from "Mickey Maus <simimaus@cip.informatik.uni-
erlangen.de>"
```


A.9 Schlüsselserver

- Öffentliches Verzeichnis zur Ablage von PGP Schlüsseln
  - ➤ Anonym
  - ➤ keine Verifikation
  - ➤ bequem

```
$ gpg --search-keys siretart@debian.org
gpg: searching for "siretart@debian.org" from hkp server ...
(1)     Reinhard Tartler <siretart@debian.org>
        Reinhard Tartler <siretart@tauware.de>
        Reinhard Tartler <siretart@ubuntu.com>
          1024 bit DSA key 945348A4, created: 2005-02-12
Keys 1-1 of 1 for "siretart@debian.org".  Enter number(s), N)ext,
or Q)uit > 1
gpg: requesting key 945348A4 from hkp server keyserver.ubuntu.com
gpg: key 945348A4: "Reinhard Tartler <siretart@debian.org>" 129 new
signatures
gpg: 3 marginal(s) needed, 1 complete(s) needed, classic trust
model
gpg: depth: 0  valid:   2  signed:  65  trust: 0-, 0q, 0n, 0m, 0f, 2u
gpg: depth: 1  valid:  65  signed:  54  trust: 63-, 0q, 0n, 1m, 1f, 0u
gpg: next trustdb check due at 2007-12-31
gpg: Total number processed: 1
gpg:         new signatures: 129
```


Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.

A.9 Schlüsselserver

- Schlüssel an den Schlüsselserver senden

```
$ gpg --keyserver blackhole.pca.dfn.de --send-keys
gpg: sending key 945348A4 to hkp server blackhole.pca.dfn.de
```

- Schlüssel mit bekannter ID von Schlüsselserver abrufen

```
$ gpg --recv-key A86B35C5
gpg: requesting key A86B35C5 from hkp server subkeys.pgp.net
gpg: key A86B35C5: public key "Linus Torvalds
<Linus.Torvalds@Helsinki.FI>" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```


A.10 Schlüssel signieren

- Liste der Unterschriften eines Schlüssels anzeigen

```
$ gpg --list-sigs 945348a4
pub   1024D/945348A4 2005-02-12 [expires: 2011-02-15]
uid Reinhard Tartler <siretart@debian.org>
sig 3 945348A4 2007-01-03  Reinhard Tartler <siretart@debian.org>
sig 3 EAB4BA1F 2007-01-23  Rainer Sennwitz <Rainer.Sennwitz@...>
sig 2 DFA06867 2005-11-26  Michael Gernoth <michael@zerfleddert.de>
[...]
```

- Fremden Schlüssel signieren, dabei Identität des Inhabers überprüfen

```
$ gpg --edit-key 945348A4
Command> lsign

pub  1024D/945348A4  created: 2005-02-12  expires: 2011-02-15
usage: SC
                     trust: unknown       validity: unknown
 Primary key fingerprint: 9300 5DC2 7E87 6C37 ED7B  CA9A 9808 3544
9453 48A4
[...]
Really sign? (y/N) y
Command> save
```


Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.

A.11 Web of trust

- Transitives Vertrauen? (Pathfinder: http://pgp.cs.uu.nl/)
- ,,Strong Set'' definiert eine grosse Menge an Schlüsseln, zwischen denen Signaturpfade existieren. (derzeit ca. 36000 Schlüssel im Strong Set)

© Universität Erlangen-Nürnberg - Informatik 4, 2009

Reproduktion jeder Art oder Verwendung dieser Unterlage, außer zu Lehrzwecken an der Universität Erlangen-Nürnberg, bedarf der Zustimmung des Autors.