

# Vertraulichkeit im Internet

Gefahren und Lösungsansätze

# Wer?

- Nils Faerber, Gründer von „kernel concepts“
- Typisches KMU – mehr K als M, zur Zeit 8 Beschäftigte
- Systemnahe Software-Entwicklung, vornehmlich „embedded“, bspw. portable Geräte, Steuerungen
- Elektronikentwicklung & Prototypenfertigung, insbesondere mit Microcontrollern



# Und warum?

- Bekommen oft und gerne sogenannte NDA – Non Disclosure Agreement
- Besagt: Als vertraulich gekennzeichnete Informationen, bspw. vom Auftraggeber, dürfen Dritten nicht zugänglich gemacht werden  
Das ist OK.
- Vertrauliche Informationen werden anschließend als einfache eMail zugesandt!  
Das ist nicht OK!

# Deshalb...

- Gespräche haben gezeigt, daß es oft nicht bekannt und/oder bewußt ist...
  - wie Kommunikation im Internet abläuft...
  - welche Angriffsmöglichkeiten bestehen...
  - wie dies mit vertretbaren Mitteln zu verhindern ist
- Es ist mir ein persönliches Bedürfnis, zu versuchen Bewußtsein zu schaffen und zu helfen aufzuklären

# Also los...

## Sicherheitsbedürfnisse

- Im Wesentlichen zwei:

### **Authentizität**

Ist mein Kommunikationspartner wirklich der, für den er sich ausgibt?

- Beispiel: Internet-Banking, PIN und TAN werden eingegeben – ist es wirklich meine Bank, der ich diese vertraulichen Informationen gerade gegeben habe?

# Sicherheitsbedürfnisse

- Beispiel: Ist die eMail des Chefs mit der Kündigung wirklich vom Chef?
- Beispiel-Fake: direkter SMTP Absender Fake

(SMTP – Simple Mail Transport Protocol, die Basis des gesamten Internet eMail Verkehrs)

# Absender Fälschung

```
nils@moi: /home/nils
nils@moi[~]telnet mail.kernelconcepts.de smtp
Trying 212.60.202.196...
Connected to mail.kernelconcepts.de.
Escape character is '^]'.
220 mail.kernelconcepts.de ESMTP Exim 4.72 Tue, 15 Nov 2011 13:18:12 +0100
HELO whitehouse.gov
250 mail.kernelconcepts.de Hello whitehouse.gov [192.168.2.30]
MAIL FROM: president@whitehouse.gov
250 OK
RCPT TO: nils.faerber@kernelconcepts.de
250 Accepted
DATA
354 Enter message, ending with "." on a line by itself
Subject: Dinner invitation

Dear Nils!
As president of the United States of America I would like to invite you to
join our yearly dinner party.
Kind regards
    the president
+
250 OK id=1RQHyI-0006nR-6V
quit
221 mail.kernelconcepts.de closing connection
Connection closed by foreign host.
nils@moi[~]
```

# Absender Fälschung, Ergebnis beim Empfänger



# Sicherheitsbedürfnisse

## Vertraulichkeit

Können unbefugte Dritte den Inhalt meiner Kommunikation abhören oder verfälschen?

- Beispiel: Bilanzveröffentlichung – können unbefugte Dritte bereits vor Veröffentlichung Kenntnis erlangen?
- Kann die Konkurrenz vertrauliche Informationen (bspw. technische Unterlagen) erhalten?

# Sicherheitsbedürfnisse

- Im geschäftlichen Umfeld werden oft Betriebsgeheimnisse per Internet kommuniziert, teils strafbewährt mit sog. Non Disclosure Agreement (NDA)
- Ist das Versenden solcher Informationen per eMail wirklich „sicher“?
- Entspricht dies den Anforderungen eines NDA?

# Kommunikation im Internet

- Die Stärke des Internet ist zugleich seine Schwäche
  - Alle Daten sind gleich, ungeachtet des Inhalts
    - eMail, WWW, Telefonie etc.
  - Datenübermittlung im Netz erfolgt „im Klartext“
    - So wie der Benutzer die Daten absendet, so werden sie auch weitergeleitet

# Kommunikation im Internet

- eMail ist vergleichbar mit einer Postkarte – nicht mit Brief!
- Jeder Teilnehmer auf dem Transportweg kann den Inhalt lesen
- Vertraulichkeit ist bei eMail *nicht* gewährleistet!

# Kommunikation im Internet

- Der Weg der Daten durch das Netz ist nicht vorherbestimmt
  - Der Benutzer kann nicht sicherstellen, wer die Daten auf ihrem Weg transportiert und entsprechend Zugriff darauf hat
- Beispiel:  
`tracert www.google.com`

# Der Weg durch das Netz:

```
mrxvt
nils@moi: /home/nils
nils@moi[~]traceroute -q 1 www.google.com
traceroute to www.google.com (74.125.39.105), 30 hops max, 60 byte packets
 1 gateway.kc.loc (192.168.2.254)  0.327 ms
 2 bras2.dus.qsc.de (213.148.133.2)  48.109 ms
 3 core1.dus.qsc.de (87.234.13.61)  50.170 ms
 4 core1.fra.qsc.de (213.148.128.213)  55.687 ms
 5 peergw-google.fra.qsc.de (212.202.214.182)  59.116 ms
 6 72.14.238.44 (72.14.238.44)  65.781 ms
 7 72.14.239.60 (72.14.239.60)  96.979 ms
 8 209.85.254.116 (209.85.254.116)  86.014 ms
 9 209.85.254.134 (209.85.254.134)  98.919 ms
10 fx-in-f105.1e100.net (74.125.39.105)  97.655 ms
nils@moi[~]traceroute -q 1 www.google.com
traceroute to www.google.com (209.85.148.147), 30 hops max, 60 byte packets
 1 gateway.kc.loc (192.168.2.254)  0.332 ms
 2 bras2.dus.qsc.de (213.148.133.2)  59.296 ms
 3 core1.dus.qsc.de (87.234.13.61)  49.604 ms
 4 core1.fra.qsc.de (213.148.130.142)  61.075 ms
 5 peergw-google.fra.qsc.de (212.202.214.182)  72.476 ms
 6 72.14.238.46 (72.14.238.46)  82.775 ms
 7 209.85.254.41 (209.85.254.41)  92.572 ms
 8 fra07s07-in-f147.1e100.net (209.85.148.147)  96.015 ms
nils@moi[~]
```

# SMTP Verkehr – Offen lesbar:

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
22	34.117159	212.60.202.196	192.168.2.30	TCP	smtp > 51307 [ACK] Seq=219 Ack=269 Win=6912
23	35.929877	192.168.2.30	212.60.202.196	SMTP	C: DATA fragment, 17 bytes
24	35.930484	212.60.202.196	192.168.2.30	TCP	smtp > 51307 [ACK] Seq=219 Ack=286 Win=6912
25	37.654943	192.168.2.30	212.60.202.196	IMF	subject: Dinner invitation\r\n,
26	37.655554	212.60.202.196	192.168.2.30	TCP	smtp > 51307 [ACK] Seq=219 Ack=289 Win=6912
27	37.658528	212.60.202.196	192.168.2.30	SMTP	S: 250 OK id=1RQIH5-0006w0-Sk
28	37.658538	192.168.2.30	212.60.202.196	TCP	51307 > smtp [ACK] Seq=289 Ack=247 Win=1460

▶ Frame 25: 69 bytes on wire (552 bits), 69 bytes captured (552 bits)

- ▶ Ethernet II, Src: Wistron\_3e:4a:04 (00:16:d3:3e:4a:04), Dst: Xensourc\_4f:4f:d1 (00:16:3e:4f:4f:d1)
- ▶ Internet Protocol, Src: 192.168.2.30 (192.168.2.30), Dst: 212.60.202.196 (212.60.202.196)
- ▶ Transmission Control Protocol, Src Port: 51307 (51307), Dst Port: smtp (25), Seq: 286, Ack: 219,
- ▶ Simple Mail Transfer Protocol
- ▼ Internet Message Format
  - Subject: Dinner invitation\r\n
  - ▼ Message-Text
    - Dear Nils!
    - As president of the United States of America I would like to invite you to join our yearly dinner party.

0000 00 16 3e 4f 4f d1 00 16 d3 3e 4a 04 08 00 45 10 ..>00... .>J...E.  
0010 00 37 d4 ff 40 00 40 06 03 ea c0 a8 02 1e d4 3c .7..@.@. ....<

Frame (69 bytes) Reassembled DATA (180 bytes)

Frame (frame), 69 bytes Packets: 28 Displayed: 28 Marked: 0 Load time: 0:00.000 Profile: Default

# SMTP – Store & Forward

- Ausgelegt dafür, daß eMails indirekt versendet werden können
- Weitere Server können auf dem Weg zum eigentlichen Empfänger dazwischen liegen:

```
# host kmsi.de
kmsi.de has address 212.185.18.70
kmsi.de mail is handled by 20 mail6.kdz-ws.net.
kmsi.de mail is handled by 30 mail4.kdz-ws.net.
kmsi.de mail is handled by 10 mail5.kdz-ws.net.
```

# SMTP – Store & Forward

- Früher sehr üblich, wie bei Mailboxnetzen
- Erste eMails wurden bereits vor „dem Internet“ über UUCP verteilt  
(UUCP – Unix to Unix CoPy)

*Also:* Es ist für den Benutzer praktisch nicht erkennbar, wer die eMail Daten auf ihrem Weg einsehen könnte oder kann!

# Firewall und Virens Scanner, ist das nicht sicher genug?

- Nein!

Sie dienen nur der Verhinderung direkter Angriffe auf den eigenen Rechner bzw. das eigene Netzwerk.

- Ein Virens Scanner könnte Spionage-Viren und/oder Trojaner (Stichwort „Bundes-Trojaner“) verhindern, die Firewall die Ausspähung im lokalen Netz.

# Wirksame Methoden

- Einzig Kryptografische Verfahren können Authentizität und Vertraulichkeit herstellen und gewährleisten.
- Kryptografie ist kein Hexenwerk!
- Kryptografie braucht nicht notwendigerweise komplizierte Infrastruktur.
- Kryptografie kann von jedermann effektiv und effizient angewendet werden!

# Kryptografie

- Es gibt zwei grundlegende Arten,

Symmetrische Verfahren:

Ein Schlüssel dient zur Ver- *und* Entschlüsselung

# Kryptografie

Asymmetrische, sog. „Public Key“, Verfahren  
Verfahren:

Basiert auf einem Schlüsselpaar (bspw. Schlüssel A & B), jeder Schlüssel des Paares dient zur jeweils inversen Operation des anderen:

Verschlüsselung mit A, Entschlüsselung mit B,  
Verschlüsselung mit B, Entschlüsselung mit A

# Kryptografie

- Symmetrische Verfahren
  - Schnell
  - Verwendung zur Herstellung von Vertraulichkeit, ohne den Schlüssel kann der Inhalt nicht gelesen oder verändert werden
- Asymmetrische Verfahren
  - Langsam
  - Verwendung zur Herstellung von Authentizität

# Kryptografie

## Herstellung von Authentizität

- Basiert auf asymmetrischen Verfahren
- Benutzer erzeugt ein Schlüsselpaar ( bspw.  $A+B$ )
- Einer der Schlüssel wird zum „privaten“ und der andere zum „öffentlichen“ Schlüssel
- Welcher der beiden  $A+B$  für welchen Zweck verwendet wird ist egal, sie dürfen nur nach der Definition *nie* wieder umdefiniert werden

# Kryptografie

## Herstellung von Authentizität

- Der öffentliche Schlüssel kann und sollte öffentlich „verteilt“ werden
  - Je breiter er gestreut wird, desto höher das mögliche Vertrauen Dritter darin
- Über Öffentliche Schlüssel-Server:
  - Freie Schlüssel-Server erhalten Vertrauen durch Redundanz – je breiter die Streuung, desto schwerer zu (ver-)fälschen
  - Kommerzielle betreiben sog. Trust-Center

# Kryptografie

## Herstellung von Authentizität

- Kommunikationspartner K1 schickt eine Nachricht N an K2
- K1 verschlüsselt N mit seinem privaten Schlüssel und sendet Klartext von N sowie Schlüsseltext von N an K2
- K2 entschlüsselt Schlüsseltext mit dem öffentlichen Schlüssel von K1 und kann nun den Klartext mit dem Entschlüsselungsergebnis vergleichen

# Authentizität In der Praxis

- Von der Nachricht  $N$  wird die Prüfsumme (sog. Hash oder CRC) gebildet
- Es wird nur die Prüfsumme verschlüsselt und mit dem Klartext übertragen
- Empfänger kann Prüfsumme entschlüsseln und mit selbst erzeugter Prüfsumme der Nachricht vergleichen
  - Prüfsumme muß möglichst kollisionsfrei sein!

# Kryptografie

## Herstellung von Vertraulichkeit

- Größtes Problem: Schlüsselaustausch
- Wie kann ich meinem Kommunikationspartner den Schlüssel zur Entschlüsselung mitteilen, ohne das Dritte diesen auch erhalten können?
- Wie verhindere ich sog. „man in the middle“ Angriffe?
  - Ein unbefugter Dritter auf dem Übermittlungsweg fängt die Kommunikation ab und fügt seinen Schlüssel anstatt des Originals ein!

# Schlüsselaustausch - Symmetrisch

- Bei symmetrischen Verfahren muß der Schlüssel über einen sicheren Kanal dem Partner übermittelt werden  
(versiegelter Brief, Telefondiktat, persönliche Übergabe etc.)
- Sehr unpraktisch und unpraktikabel
- Lösung: Asymmetrische Verfahren

# Vertrauliche Kommunikation - Asymmetrisch

- Kommunikationspartner K1 will an K2 eine vertrauliche Nachricht N versenden.
- K1 verschlüsselt N mit dem öffentlichen Schlüssel von K2
- Der private Schlüssel von K2 ist ausschließlich K2 bekannt → nur K2 kann die Nachricht erfolgreich entschlüsseln

# Vertraulichkeit

## In der Praxis

- Es wird ein zufälliger symmetrischer Schlüssel erzeugt
- Der symmetrische Schlüssel wird mit dem öffentlichen Schlüssel von K2 verschlüsselt
- Die Nachricht wird mit dem symmetrischen Schlüssel verschlüsselt
- Beides wird an K2 übermittelt
- Nur K2 kann den sym. Schlüssel entschlüsseln

# Stand heute

- Praktisch alle Verfahren zur Authentikation und Vertraulichkeit basieren auf Public Key Verfahren
- Größte Herausforderung ist Herstellung des „Vertrauens“ in die Quelle des öffentlichen Schlüssels
- Betrieb eines Trust-Center ist *sehr* aufwändig und teuer
- Freie Lösungen basieren auf „Web of Trust“

# Authentikation

- Basiert auf Ausstellung sog. „Zertifikate“
- Werden ausgestellt durch eine „Certification Authority“ (CA)
- Zertifikat ist ein durch die CA signierter „elektronischer Ausweis“
- Prüfung der Echtheit über öffentlichen Schlüssel der CA

# Problem - Kompromittierung der CA?

- Jüngster Fall in Niederlande, Firma „Digi Notar“
- Wurden gehackt und falsche Zertifikate ausgestellt
  - Digi Notar galt als voll vertrauenswürdig
  - Jeder *mußte* den falschen Zertifikaten vertrauen
- Angeblich sind fast alle großen CAs bereits „infiltriert“ und damit kompromittiert
- Eine CA ist ein „single point of failure“

# Freie Lösungen

- CaCert, kostenlose Zertifikate für Personen und Server
- Vertrauen durch „Assurer“, Personen, die vertrauenswürdig die Identität prüfen
- Primär genutzt für Server-SSL Zertifikate
- Interesse?  
*Ich* bin Assurer!  
Benötigt: Lichtbild-Ausweis und ein Formular.

# Freie Lösungen

- GnuPG – freien PGP Implementation
- Primär genutzt zur Herstellung von Vertraulichkeit und Authentizität persönlicher Kommunikation (eMail)
- Web-of-Trust durch gegenseitige Signatur des Schlüssels, sog. „Key signing party“
- Das können wir *gleich* machen!

# Software

- Freie PGP software: GnuPG zur Signatur und Verschlüsselung
- Für Linux:  

```
apt-get install gnupg  
apt-get install seahorse
```
- Für Windows:  
<http://www.gpg4win.org/>

# Für eMail Clients

- Outlook
  - Gpg4Win OL Plugin
- Mozilla Thunderbird
  - Enigmail

# Warum „Freie Software“?

- Unfreie Software kann nicht von unabhängigen Dritten überprüft werden!
- Man muß dem Anbieter *vertrauen*
  
- Glauben Sie mir – Sie wollen *keinem* kommerziellem Anbieter von Kryptografielösungen vertrauen!

# Schritte...

- Nach der Installation
  1. Schlüssel erzeugen
  2. Öffentlichen Schlüssel zum Schlüsselservers hochladen, bspw.:  
`x-hkp://gpg-keyserver.de`
  3. Schlüssel „Fingerprint“ aufschreiben oder ausdrucken
  4. Mit Fingerprint und Lichtbildausweis von Dritten beglaubigen lassen

# Key Signing Party!

- Let's party!

Mein Fingerprint ist:

```
pub      1024D/11B5A878 2000-02-28
```

```
Schl.-Fingerabdruck =
```

```
        57CD 8B0F E1F2 561A 0BD8  2761 2577 8851 11B5 A878
```

```
uid          Nils Faerber <nils.faerber@kernelconcepts.de>
```

```
uid          Nils Faerber <nils@unix-ag.org>
```

```
uid          Nils Faerber <nils@kernelconcepts.de>
```

```
sub      2048g/6E4970BF 2000-02-28
```

# Erreichbare Eigenschaften

- Software ist einfach zu installieren
- Schlüsselverteilung über öffentliche Server ist relativ komfortabel möglich
- Verifikation eines fremden Schlüssels kann im Zweifelsfall einmal telefonisch anhand des Fingerprint erfolgen (gegenseitiges Diktat)

# Erreichbare Eigenschaften

- Verbleibendes Hauptrisiko:  
Unautorisierter Zugriff auf den privaten Schlüssel
- Ist auf Festplatte durch „Passphrase“ geschützt
- Könnte theoretisch geknackt werden
- Abhilfe: Smartcards,  
Privater Schlüssel ist sicher in Hardware  
gespeichert und ist nicht auslesbar  
(Beispiel: OpenPGP Smartcard – 16,40 EUR)

# Abgrenzung zu „offiziellen“

- Anerkannte Trust-Center CAs können Zertifikate für sog. „qualifizierte digitale Signatur“ ausstellen
- Diese ist äquivalent einer rechtsverbindlichen eigenhändigen Unterschrift
- Die GnuPG PGP Signatur ist dies nicht...  
Who cares?  
Sie wissen aber jetzt, mit wem sie kommunizieren und wer mitlesen kann!

# Kommentar

## ePostbrief, de-Mail, ePA

- Der ePostbrief bietet **keine** Ende-zu-Ende Verschlüsselung und ist damit praktisch unbrauchbar – schon gar nicht für Rechtsverbindlichkeit (siehe AGBs der Post)
- Bei de-Mail und dem elektronischen Personalausweis hat die Bundesdruckerei aus Kostengründen den Betrieb eines qualifizierten Trust-Centers abgelehnt → auch der ePA liefert keine qualifizierte digitale Signatur!

# Kommentar

## CAs und Trust-Center

- Trotz der hohen Anforderungen an CAs sind diese kompromittierbar
- Der Betrieb ist so aufwändig und teuer, daß immer weniger Betreiber Zertifikate für qualifizierte digitale Signaturen anbieten

*Warum also nicht gleich freie Lösungen fördern?*

The end...

Vielen Dank für Ihre Aufmerksamkeit!

Fragen?  
Auch gerne an:

[nils.faeber@kernelconcepts.de](mailto:nils.faeber@kernelconcepts.de)

(Die Vortragsfolien versende ich auf Anfrage auch gerne per digital signierter eMail :)