

# 1. PGP – warum, was und wie?

## Gliederung

1. Ein paar Worte zur Überwachung
2. Wo wird abgehört?
3. Verschlüsselung
4. Public Key-Verfahren
5. Signaturen
6. Das Web of Trust
7. Vorführung

Überwachung von E-Mail ist keine Theorie, sie ist auf allen Ebenen existierende Praxis. Wegen der relativ geringen Datenmengen und der digitalen Lesbarkeit ist flächendeckende Speicherung und z.B. Schlüsselwortsuche technisch trivial.

E-Mails sind wegen ihrer digitalen Natur und Offenheit aus Überwachungssicht noch um Längen einfacher zu handhaben als etwa Postkarten oder Telefongespräche.

E-Mails werden in der BRD nach TKÜV in der Regel nach Adresse überwacht, nur bei Auslandsverbindungen kommt globales Scannen in Betracht. Ob diese Regel der Realität entspricht, ist eine andere Frage.

Polizeiliche Anordnungen zur Überwachung von Mail sind noch selten, haben aber Wachstumsraten von weit über 1000% pro Jahr.

- ⊞ Und nur ein kleines mahnendes Wort: Verschlüsselung ist nicht nur für Mails relevant, die ihr als „sicherheitsrelevant“ einschätzt. Verschlüsselung sollte Routine sein. Wenn nicht aus Prinzip, dann doch zumindest aus der Überlegung, dass *eine* verschlüsselte Mail Verdacht erregt, eine von *vielen* verschlüsselten Mails nicht.
- ⊞ Also: Mails verschlüsseln ist angewandte Solidarität. . .

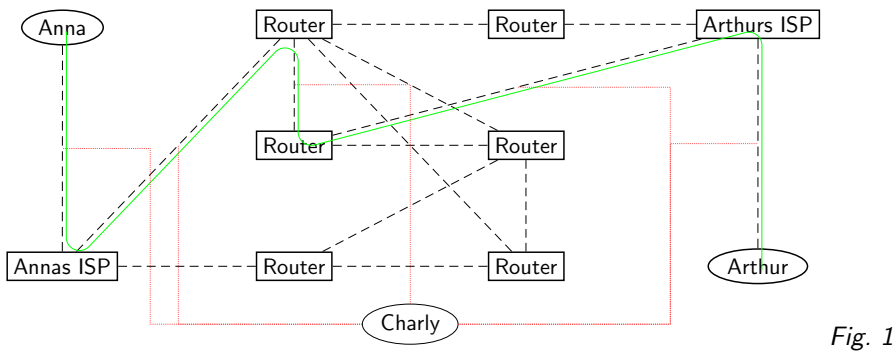


Fig. 1

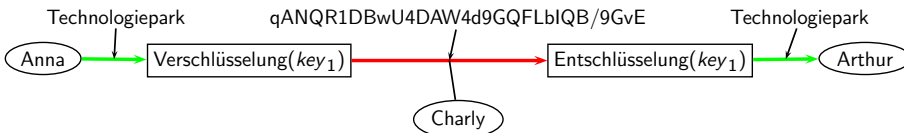


Fig. 2

## 2. Wo wird Netzverkehr abgehört?

(vgl. Fig. 1)

Der Weg von Paketen ist am Netz schwer vorherzusagen.

„Bedarfsträger“ haben an fast jeder Stelle Zugriff auf die transportierten Daten.

Das gilt natürlich nicht nur für Mails, sondern genauso für jede andere Kommunikation am Netz.

Das Problem der Kommunikationsprofile („wer mit wem?“) behandeln wir hier nicht (aber: Mixmaster).

## 3. Symmetrische Verschlüsselung

(vgl. Fig. 2)

Anna und Arthur vereinbaren eine Funktion, die aus normalem Text („Technologiepark“) scheinbar sinnlose Zeichen („qAN...“) macht. Die Funktion kann umgekehrt werden (d.h. aus den sinnlosen Zeichen wird wieder der Text) – aber zur Berechnung der Funktion braucht mensch den Schlüssel  $key_1$ .

Ist die Funktion gut gewählt, kann Charly ohne Kenntnis von  $key_1$  den Klartext nicht (mit vernünftigem Aufwand) rückgewinnen.

- ⊃ Ein simples Beispiel für die Funktion könnte sein: „Wandele jedes Zeichen in die Position seines ersten Vorkommens auf einer zufällig gewählten Seite eines Buchs.“ Der Schlüssel ist dann der Titel des Buchs.

Probleme:

1. Übermittlung des Schlüssels
2. Wer verschlüsseln kann, kann auch entschlüsseln
3. Gute Wahl der Verschlüsselungsfunktion
4. Sicherheit der Leitungen zwischen Anna und der Verschlüsselung bzw. zwischen der Entschlüsselung und Arthur

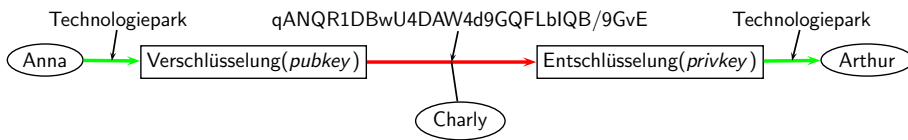


Fig. 3

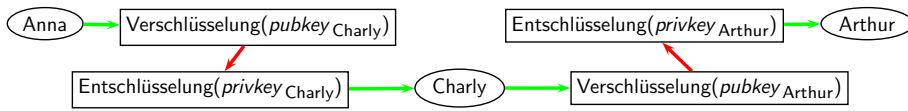


Fig. 4

## 4. Asymmetrische Verschlüsselung

- Mindestens die ersten beiden Probleme lösen asymmetrische (public-key) Verschlüsselungsverfahren.

Idee: Verschlüsselungsfunktion braucht nur einen Teil des Schlüssels, (den *öffentlichen Schlüssel*) zum Entschlüsseln braucht man den vollen Schlüssel (den *privaten Schlüssel*).

- (Gute) Metapher: Briefkasten – jeder kann einwerfen, nur wer einen Schlüssel hat, kommt an den Inhalt ran. Die Gefahr, dass Leute mit Nachschlüsseln oder roher Gewalt an den Inhalt herankommen, ist bei aktuellen Verfahren und verantwortungsvollem Umgang mit den Schlüsseln praktisch vernachlässigbar.

Probleme:

1. Sicherheit des Schlüssels
2. Sicherheit bis zum Verschlüsseln
3. „Vertauschen der Namensschilder“

(vgl. Fig. 3)

## 5. Digitale Signatur

Wenn Charly Anna vortäuscht, Arthur zu sein und umgekehrt Arthur, er, Charly, sei Anna, kann folgendes passieren:

(vgl. Fig. 4)

Unter anderem um so etwas (man-in-the-middle attack) zu verhindern, gibt es die Digitale Signatur. Sie ist etwas, das man nur berechnen kann, wenn man den privaten Schlüssel hat, dessen Richtigkeit man aber schon mit der Kenntnis der öffentlichen Schlüssel prüfen kann.

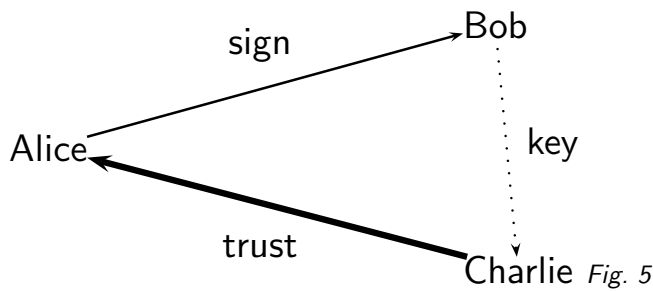
- Dass es so eine Funktion gibt, scheint zunächst unwahrscheinlich, sie ist aber ein relativ unproblematisches Abfallprodukt üblicher asymmetrischer Verschlüsselungsverfahren.
- Mit einer digitalen Signatur kann man Schlüssel „unterschreiben“.

Problem: Anna bräuchte Arthurs Schlüssel bereits, um die Echtheit von Arthurs Schlüssel zu untersuchen.

Abhilfe: Eine dritte Stelle (deren öffentlichen Schlüssel Anna bereits kennt) muss den öffentlichen Schlüssel von Arthur gegenüber Anna signieren.

Beispiele dafür:

- In euren Browsern sind die öffentlichen Schlüssel von Firmen (z.B. verisign) enthalten, mit denen nachgeprüft wird, ob z.B. die Webseite von Microsoft auch wirklich von Microsoft kommt.
- Im Zuge der Signaturinitiative des Innenministeriums ist eine staatliche Stelle zur Ausgabe signierter Schlüssel geplant.



- Natürlich wollen wir uns in unserer Kryptografie nicht auf das Innenministerium (das mit Sicherheit Nachschlüssel anfertigen wird) oder auf große Firmen (die viel Geld für ihre Dienste verlangen und nicht zwingend unabhängig von allerlei Diensten sind) verlassen. PGP verwendet daher eine andere, soziale Möglichkeit zur Verifizierung von Schlüsseln.

## 6. Das Web of Trust

Das Web of Trust basiert auf zwei Mechanismen:

- Leute signieren die öffentlichen Schlüssel anderer Leute und bestätigen damit ihre Authentizität
- Leute vertrauen anderen Leuten, dass diese wirklich nur das signieren, was sie geprüft haben

Es ergibt sich ein Netz des „Kennens“, über das sich im Idealfall die Identitäten klären lassen. Im einfachsten Fall sind daran drei Leute beteiligt:

(vgl. Fig. 5)

Dabei vertraut Charlie Alice (voll), d.h., er akzeptiert einen von Alice unterschriebenen Schlüssel, als hätte er ihn selbst geprüft. Außerdem hat Alice Bobs Schlüssel unterschrieben und hat ihn Bob zurückgegeben und auf einen Keyserver hochgeladen.

Wenn nun Charlie den Schlüssel von Bob bekommt (oder von einem Keyserver holt), sieht Charly die Unterschrift von Alice und weiß, dass Alice glaubt, der Schlüssel gehöre Bob. Weil Charlie Alice vertraut, wird sie den Schlüssel als authentisch akzeptieren. In Wirklichkeit ist das ganze etwas subtiler. Ihr könnt in der Regel den „Grad“ des Vertrauens in Personen angeben und auch die Sicherheit, mit der ihr den Schlüssel überprüft habt. Die PGP-Programme berechnen daraus dann einen „Grad“ des Vertrauens, das sie in einen Schlüssel haben.

Praktisch ist z.B. wenn es in einer Gruppe einen Menschen gibt, der/die alle Schlüssel in der Gruppe signiert hat und umgekehrt seinen Schlüssel von so vielen Menschen aus anderen Gruppen wie möglich signieren lässt. Das Ergebnis ist, dass, wer diesem „Schlüsselhüter“ vertraut, automatisch auch die Schlüssel aller Gruppenmitglieder überprüfen kann.

So schön diese Idee – eine soziale Lösung für ein technisches Problem, ganz im Gegensatz zu den versuchten Lösungen sozialer Probleme durch technische Maßnahmen, denen wir in dieser Gesellschaft so oft ausgesetzt sind – auch ist, das weite Web of Trust ist vorerst noch Utopie, insbesondere, weil die dahinterstehenden Konzepte doch etwas diffizil sind. Eine populäre Alternative sind „grass-roots“-Signierkampagnen (der Heiseverlag macht sowas), die aber leider an Unsinn wie Personalausweisen hängen, und Keysigning-Partys.

Eine Frage ist noch: Wie kriegt mensch raus, ob der Schlüssel für eineN BekannteN authentisch ist, ob wir ihn also signieren und damit auch allen, die uns vertrauen als echt anempfehlen? Populäre Methoden:

- Schlüssel z.B. auf Floppy persönlich übergeben
- Fingerprints auf Zettel persönlich übergeben

- Den Fingerprint telefonisch vorlesen lassen und darauf achten, dass die Stimme stimmt.

## 7. Zum Ende

<http://www.datenschmutz.de>