Lucerne University of
Applied Sciences and Arts

HOCHSCHULE
LUZERN

Informatik

Information Security Fundamentals
04 Kryptologie 2
Asymmetrische Verfahren

Ausbildung

Prof. Konrad Marfurt
Studiengangleiter Wirtschaftsinformatik

T direkt +41 41 757 68 61
konrad.marfurt@hslu.ch

Rotkreuz 16.03.2017

Einige Folien stammen von Jörg Schwenk (Sicherheit und Kryptographie im Internet)

Aktueller Einstieg: «Hashfunktion SHA-1 gebrochen» (Golem.de)

Werfen Sie einen Blick auf den Artikel und kommentieren Sie seinen Titel bezüglich «Kollisionsresistenz»

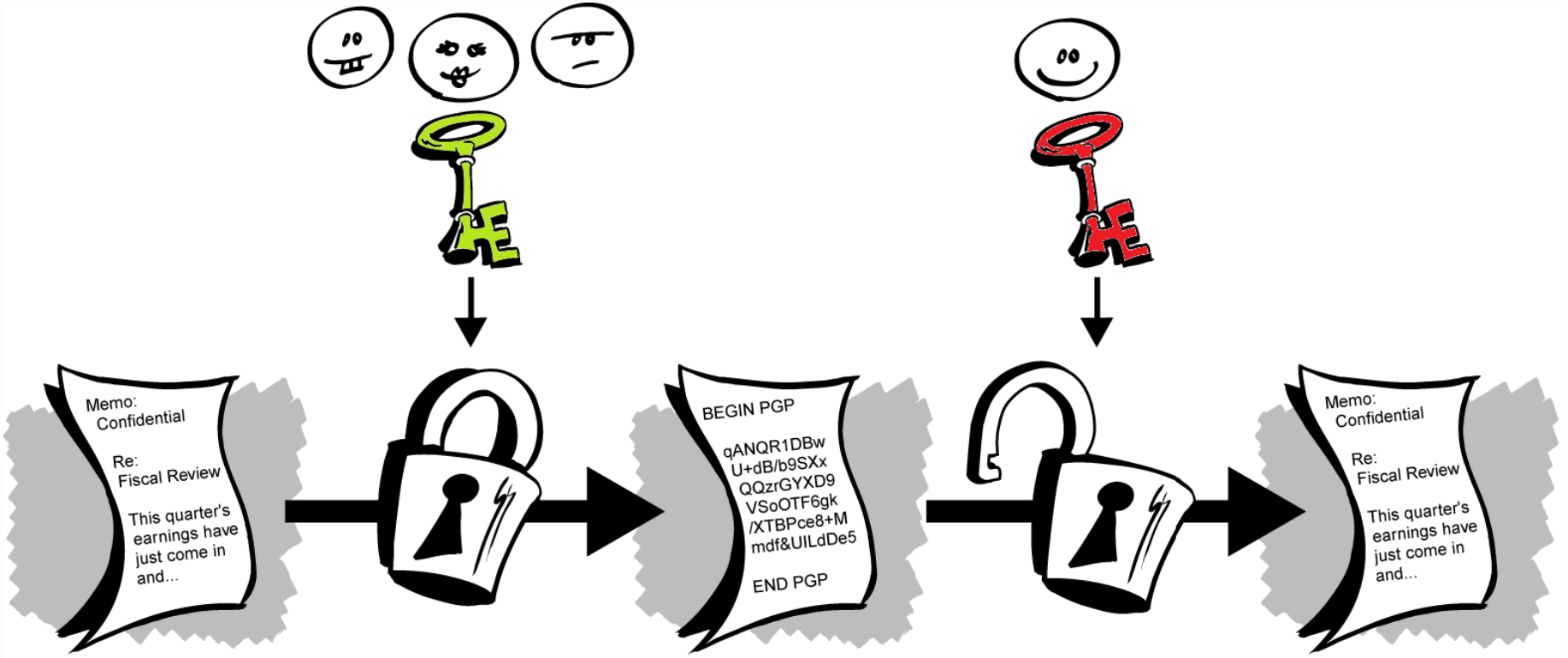
https://www.golem.de/news/kollissionsangriff-hashfunktion-sha-1-gebrochen-1702-126355.html

Es ist nur eine Einstiegsfrage zur Diskussion!

- Chrome stuft per 1.3.17 SHA-1 signierte Zertifikate als unsicher ein
- Bekannt waren bislang erst erfolgreiche Angriffe auf MD5
- CrypTool 2.0 enthält eine Demo für einen erfolgreichen Kollisionsangriff für MD5
- Das Tool Fastcollision.exe ermöglicht ebenfalls Kollisionsangriffe auf der Kommandozeile

Asymmetrische Verschlüsselung

- *Öffentlich* bekannt erst seit 1976! (Loyalität von MI5 et al ☺)
- **Öffentlicher Schlüssel** und **privater Schlüssel**
- SchlüsselPAAR wird erzeugt (komplexe mathematische Beziehung)
- Der eine Schlüssel kann **nicht** aus dem anderen berechnet werden



Asymmetrische Verfahren: Konzept

- Ein Schlüssel bleibt geheim, der andere wird veröffentlicht
- Was mit dem *einen* Schlüssel verschlüsselt wird, kann nur noch mit dem *anderen* Schlüssel des Paares entschlüsselt werden
- „Schlüssel“ ist eine vereinfachte Ausdrucksweise. Die Algorithmen sind nicht trivial und haben i.d.R. mehrere Parameter, von denen ein Teil öffentlich ist und der andere geheim bleiben muss. Diese Parameter-Sets sind die „Schlüssel“
- Rechenverfahren:
  - RSA
  - Diffie-Hellman
  - El Gamal
  - elliptische Kurven
- Alice verschlüsselt eine Nachricht mit dem öffentlichen Schlüssel von Bob → Nur Bob kann die Nachricht entschlüsseln, weil nur er seinen geheimen Schlüssel kennt

Asymmetrische Verfahren: Eigenschaften

- Der wesentliche Nachteil symmetrischer Verfahren ist die Schlüsselverteilung
- Kontrollfrage: wie viele Schlüssel brauchen n Personen, um untereinander ungestört sicher kommunizieren zu können?
  - A) symmetrische .....
  - B) asymmetrische Paare .....
- Geschwindigkeit: typische Implementationen symmetrischer Verfahren sind 2-3 Grössenordnungen schneller als asymmmetrische

- Sicherstellen der Authentizität (ist das wirklich Bob‘s Schlüssel?)
  - Hierarchische PKI: Z.B. Zertifikate nach dem X.509 Standard
  - Alternative: verteiltes Vertrauen (z.B. PGP)

Hybride Verfahren - Vorbemerkung (vgl. auch Lernzeug „CrypTool“)

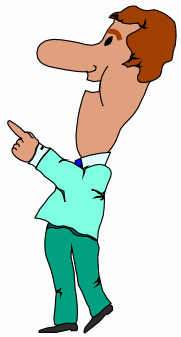
Beispiel: mit einem asymmetrischen Verfahren (hier Diffie-Hellman) einen zufällig gewählten Schlüssel für ein symmetrisches Verfahren austauschen (zum einmaligen Gebrauch)

- a ist allgemein bekannt
- Alice denkt sich a, Bob denkt sich b zufällig aus
- Alice übermittelt A (a hoch a), Bob übermittelt B (a hoch b)
- Alice berechnet B hoch a d.h. (a hoch b) hoch a, Bob berechnet A hoch b d.h. (a hoch a) hoch b
- Beide erhalten den gleichen Wert S, den sie als Schlüssel benutzen

a

$A = \alpha^{a}$

$B = \alpha^{b}$

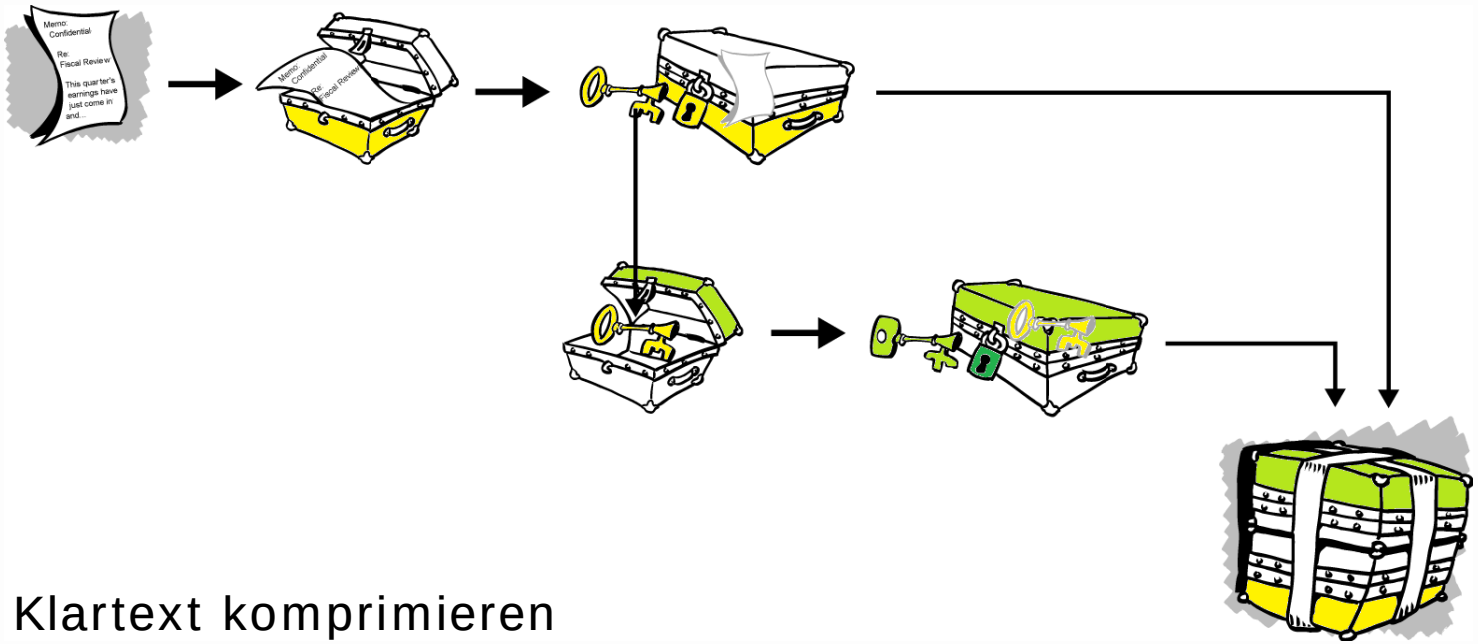


b

$S = (\alpha^{a})^{b}$

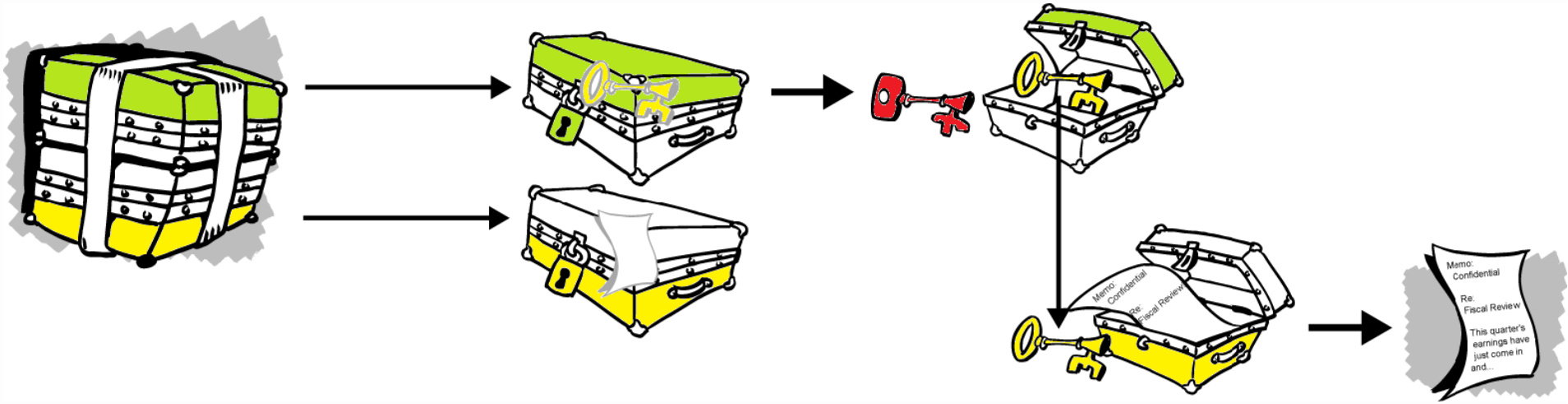
$S = (\alpha^{b})^{a}$

Nachricht mit hybrider Verschlüsselung versenden (Beispiel von PGP – Phil Zimmermann)



1. Klartext komprimieren
2. Zufälligen Sitzungsschlüssel erzeugen (1 x Gebrauch)
3. Klartext mit Sitzungsschlüssel symmetrisch verschlüsseln (z.B. IDEA)
4. Sitzungsschlüssel mit öffentlichem Schlüssel des Empfängers verschlüsseln
5. Gesamtes Paket (Textchiffrat und „Schlüsselbox“) übermitteln

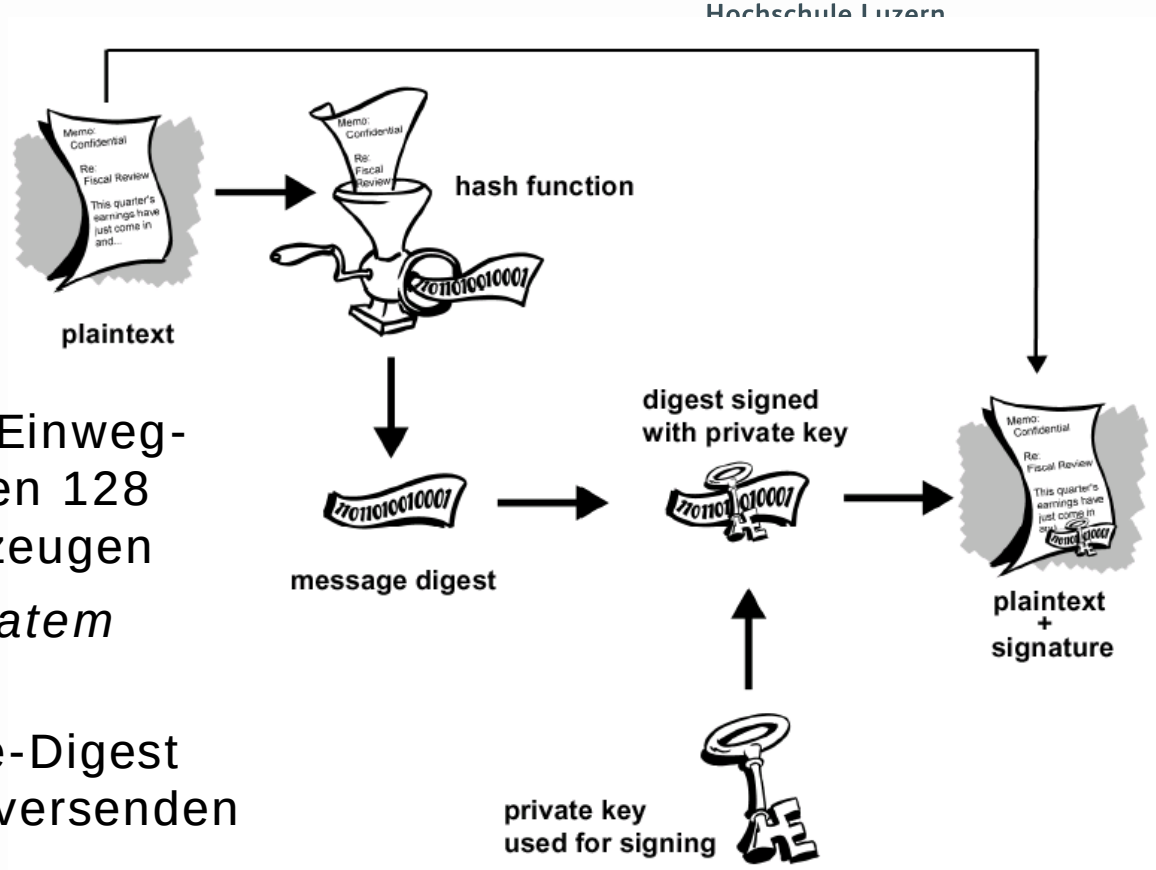
Entschlüsseln beim Empfänger



6. Der Empfänger öffnet die „Schlüsselbox“ mit Hilfe seines privaten Schlüssels
7. Er erhält so den Sitzungsschlüssel für das IDEA-Verfahren
8. Mit dem Sitzungsschlüssel erhält er den Klartext aus dem Chiffrat zurück

Kontrollfrage:
Kann der Absender das Chiffrat entschlüsseln?

Digitale Unterschrift



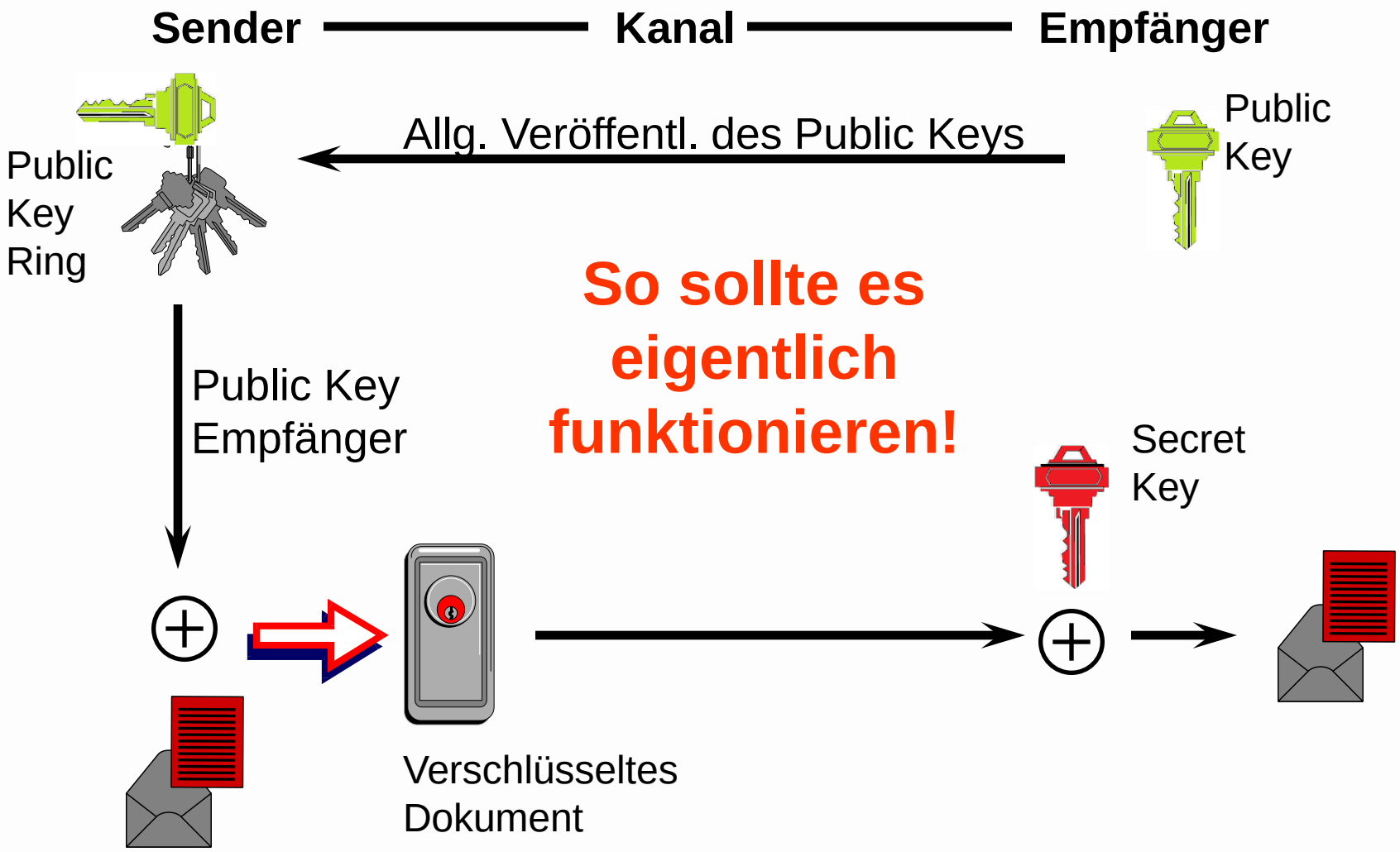
1. Aus Klartext (x MB) mit Einweg-Funktion (z.B. MD5) einen 128 Bit „Message-Digest“ erzeugen
2. Message-Digest mit *privatem* Schlüssel verschlüsseln
3. Verschlüsselten Message-Digest zusammen mit Klartext versenden

Kontrollfrage: Wieso ist der Absender und wieso die Unversehrtheit der Nachricht damit überprüfbar?

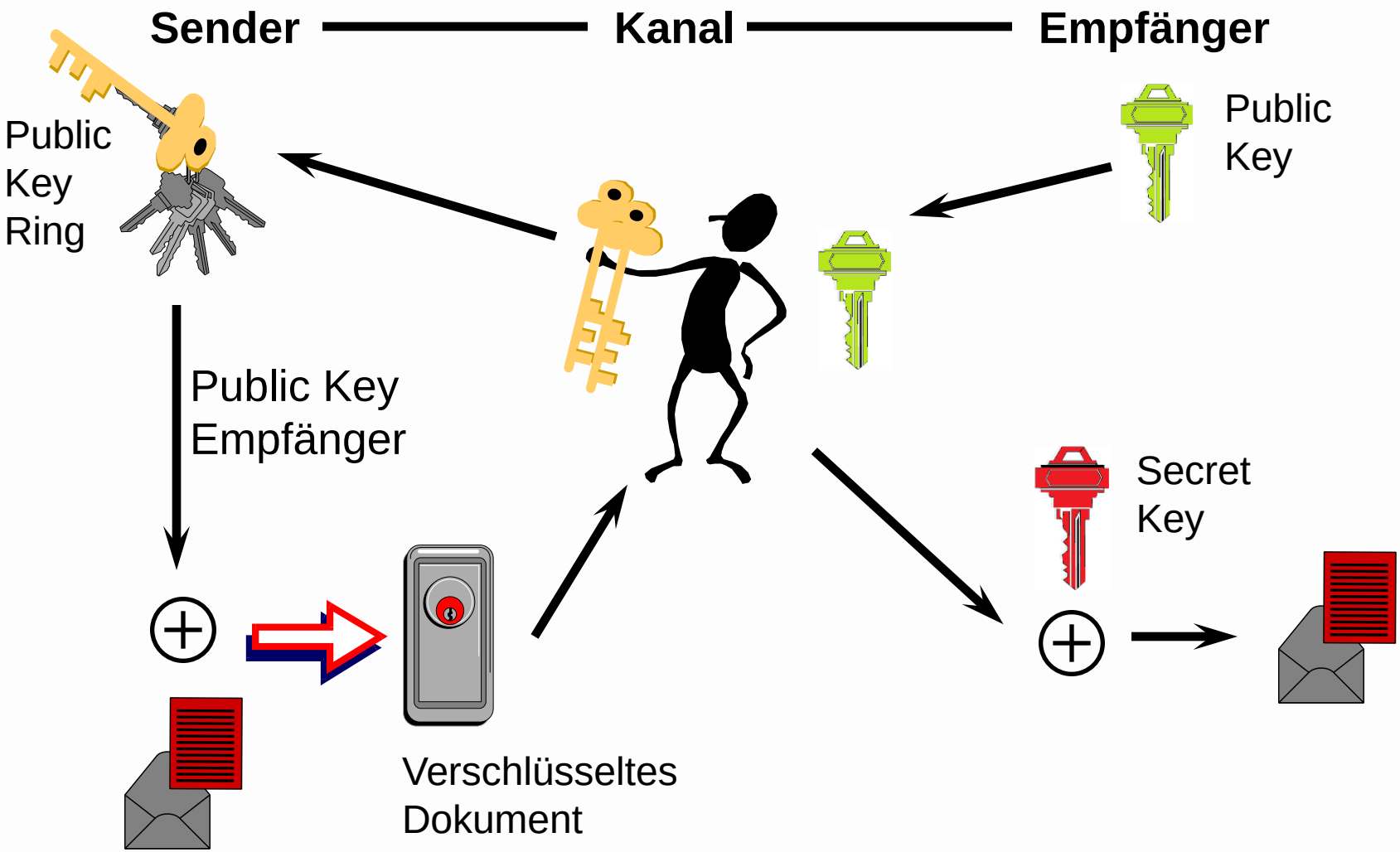
.....

.....

Asymmetrische Verschlüsselung (1/2)
Animation zum Ablauf



Asymmetrische Verschlüsselung (2/2)
„man in the middle“-Angriff



Sicherheit von asymmetrischen Verfahren

- Geheimhaltung des privaten Schlüssels!!!
  - Auswendiglernen, Aufschreiben, Disk(ette), USB-Stick, …?
  - SIM-Karte?
  - Smartcard?
  - HSM (Hardware Security Module) / «Krypto-Box»?

- Authentizität der öffentlichen Schlüssel!!!
  - Zertifikate, Beglaubigungen
  - Persönliche Übergabe («key signing party», …)

… welche Anwendungen kennen / benutzen Sie?

… diskutieren Sie die jeweiligen Vor- und Nachteile

… wo sind die jeweiligen Schlüssel hinterlegt und welche Regeln / formalen Prozesse gelten für den Umgang damit?

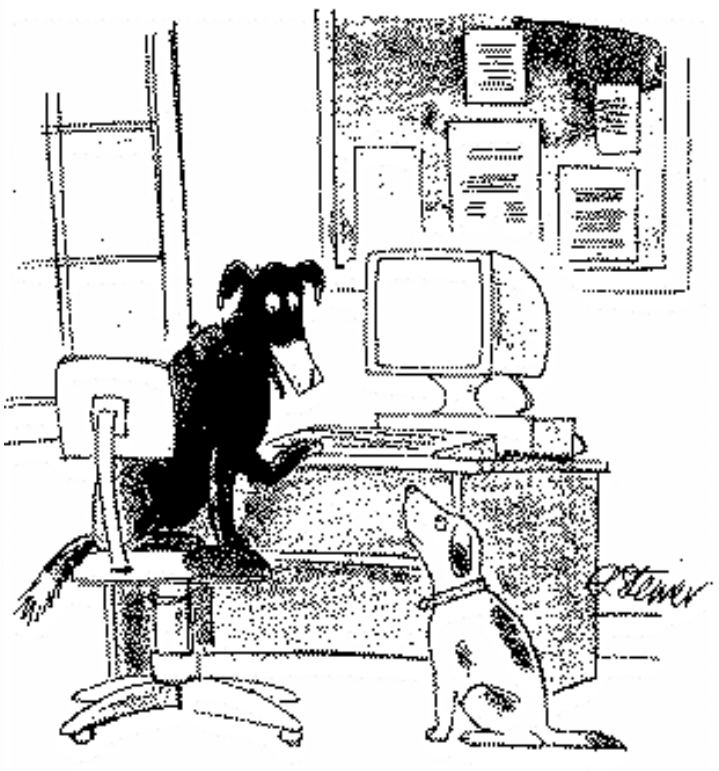
Das Problem der Authentizität

Problem:

Gehört der Public Key demjenigen, der ihn vorzeigt?

Lösung:

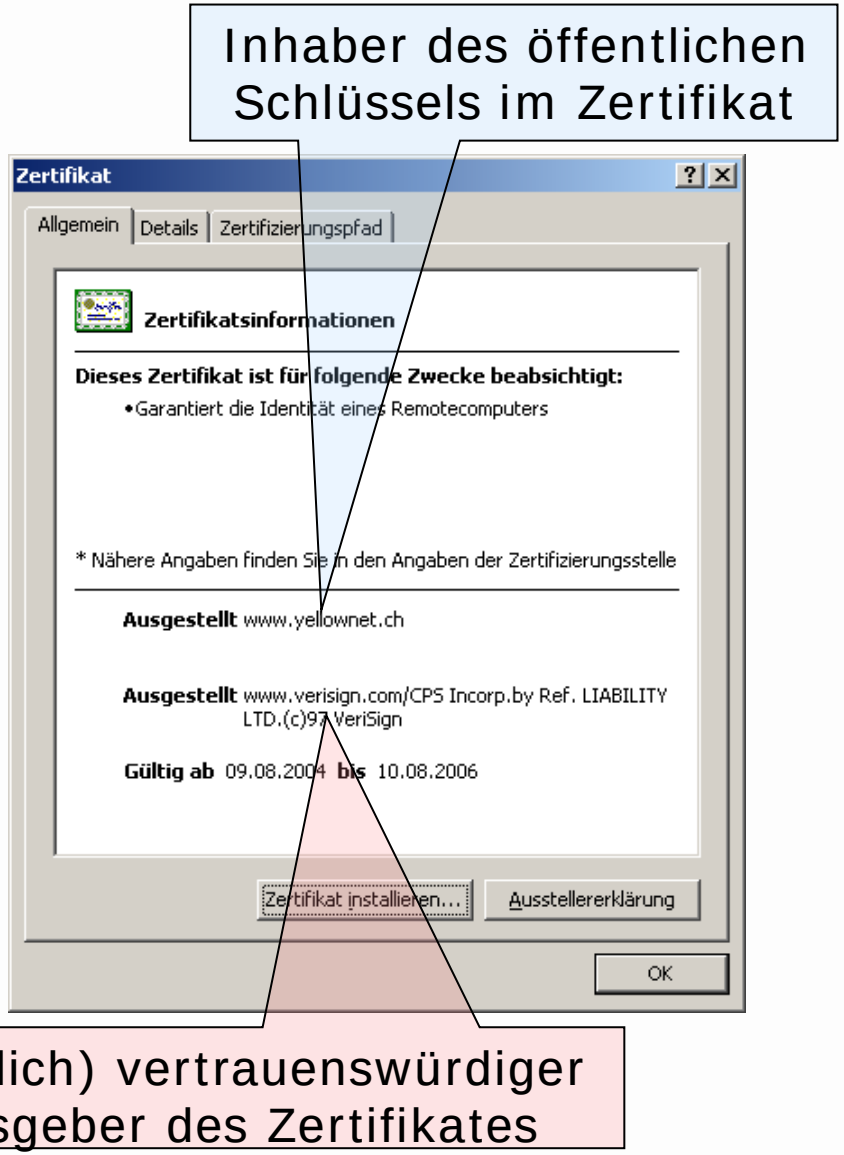
Beglaubigung der Zugehörigkeit des Public Key zur behaupteten Identität durch eine/n vertrauenswürdigen Dritte/n («trusted third party»)



„On the Internet nobody knows you‘re a dog“
Quelle: New York Times (1991 – sic!)

Public Keys beglaubigen durch Zertifikate

Zum Ausprobieren:
- Inspizieren Sie das Zertifikat einer https Seite
- Was ist der Zertifizierungspfad?
- Woher kommt das Vertrauen Ihres Browsers?
- Noch genauer: wo speichert Ihr Browser die dazu nötigen Informationen und wer kann sie wann und wie verändern?
- Wo finden Sie und wie gross ist die Certificate Revocation List?



«Anatomie» eines Zertifikats

Name des Inhabers: Alice Onliner
CA: Zertifizierungsinstanz des Staates Kryptoland
Öffentlicher Schlüssel der CA: FA 4E 9E F0 DD 9E ED F2 DD 23 A2 8A 4E 23 E3 D4
Öffentlicher Schlüssel des Inhabers: F2 D2 0E ED FA 4E 9E 0A F2 DD 23 8A 32 44 F3 E9
Seriennummer der Zertifikats: 8549285792859
Gültigkeitszeitraum: 2007-07-01 – 2011-06-30
Signatur der CA: DD 9E EF 2D D2 38 A3 2D FA 4E 9E DA 49 47 F0 22

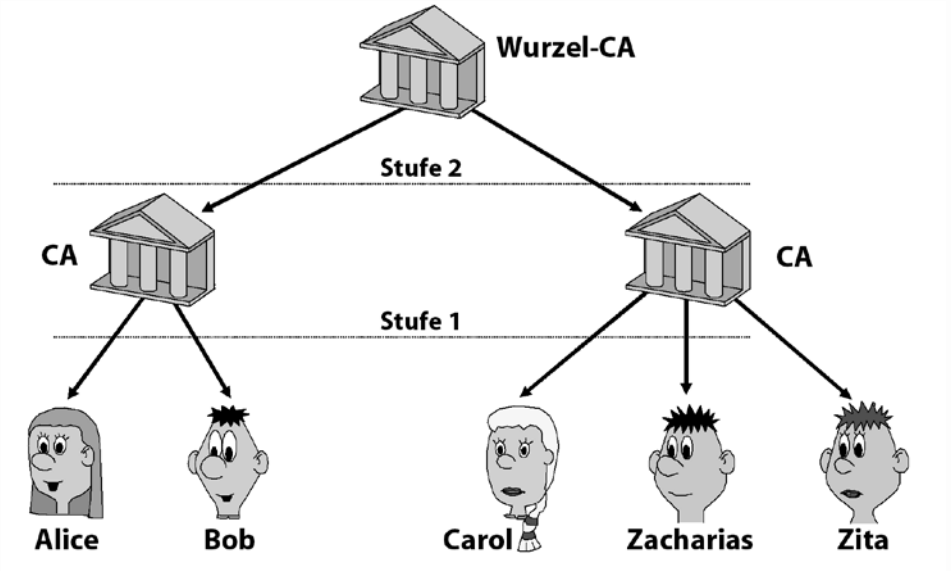
Kontrollfragen:

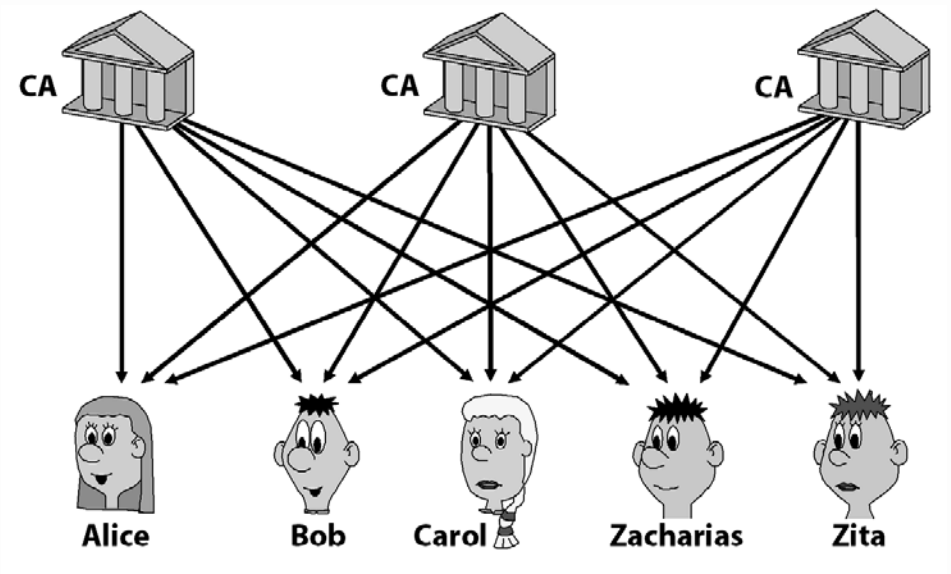
- Was alles überprüfen Sie in welcher Reihenfolge, bevor Sie dem öffentlichen Schlüssel des Inhabers vertrauen?
- Wie können Sie dem öffentlichen Schlüssel der CA vertrauen?
- Der private(!) Schlüssel der CA ist als Stempel dargestellt; warum?

Hierarchisches Konzept vs. Web of Trust

Es gibt weitere Versionen hierarchischer Konzepte, bei denen sich z.B. CAs verschiedener Firmen gegenseitig vertrauen usw.

Ein Web of Trust kann auch ohne CAs entstehen, indem sich z.B. PGP Nutzer ihre Zertifikate gegenseitig bestätigen (signieren) und sich das Vertrauen so aufbauen kann: Bob kennt Zita und vertraut Carol's Zertifikat, weil es von Carol signiert wurde





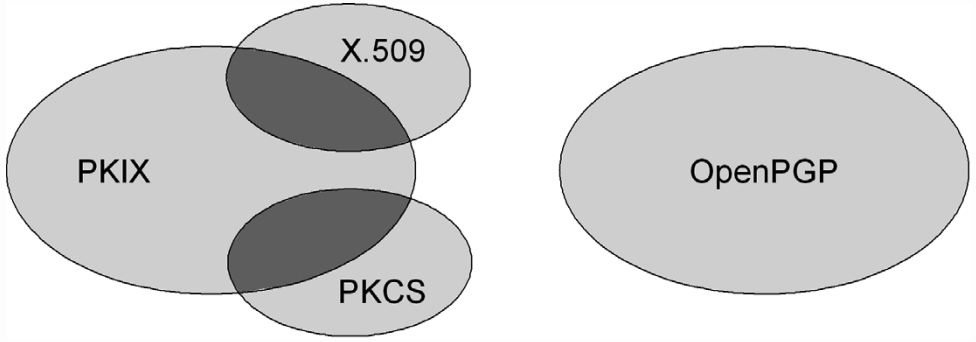
PKIX und OpenPGP
zwei nicht kompatible Standards

PKIX:

- Die Abbildung rechts zeigt die wesentlichen Informationen eines PKIX Zertifikates
- PKIX und X.509 überlappen bezüglich des Formats der Zertifikate; beide Standards sind sehr umfangreich
- Beides sind hierarchische PKIs

OpenPGP:

- Entstanden aus PGP Software
- Unterstützt das Web of Trust Konzept
- Auch hierarchisch nutzbar



Was gehört zu einer PKI? (Details im Buch von Schmeh, Kap. 33 bzw. 25)

Komponenten:

- Zertifizierungsstelle (CA) mit hoch gesichertem private key (HSM)
- Zertifikatsserver ggf. mit Verzeichnisdienst (z.B. LDAP)
- Zeitstempeldienst (**T**ime **S**tamp **A**uthority)
- Registrierungsstelle, Sperrstelle mit entsprechenden Interfaces
- **P**ersonal **S**ecurity **E**nvironment: device oder Konzept für die Speicherung des/der private keys

Rollen:

- Leiter, Operator, Administrator, Registrator, Sperragent, Anwender

Prozesse (Zertifikatsmanagement):

- Initialisierung
- Erneuerung
- Key Recovery
- Sperrung (bei hierarchischen CAs einfacher als im Web of Trust)

Hausaufgabe (Lektüre ca. 15-20 Minuten) Bundesgesetz über die elektronische Signatur, ZertES

Werfen Sie einen Blick ins «Bundesgesetz über die elektronische Signatur, ZertES» und lokalisieren Sie einige Elemente einer PKI, welche der Gesetzgeber in der Schweiz geregelt hat.

Das Gesetz finden Sie in der systematischen Rechtssammlung unter der Nummer 943.03 (https://www.admin.ch/opc/de/classified-compilation/20131913/201701010000/943.03.pdf)

Hinweise:

- In der Intermediate-Stufe wird eine Vorlesung IT-Recht angeboten
- Die Bundesverwaltung pflegt auch eine Verschlüsselungs-Software für klassifizierte Dokumente (www.securecenter.ch) von Schweizer Behörden – inkl. Lernprogramm (Gastzugang)

Nachbereitung: Schmeh Kapitel 25 bzw. 33 (online) und Schwenk Kapitel 1 (ILIAS). Dazu RSA Mini-Simulation und Fraunhofer Animation auf http://hslu-i.org/ISF/04_krypto_asym/