

Lecture 4

Cryptography



Mobile Business II (SS 2023)

Prof. Dr. Kai Rannenberg

Chair of Mobile Business & Multilateral Security
Goethe University Frankfurt a. M.

mb

- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography

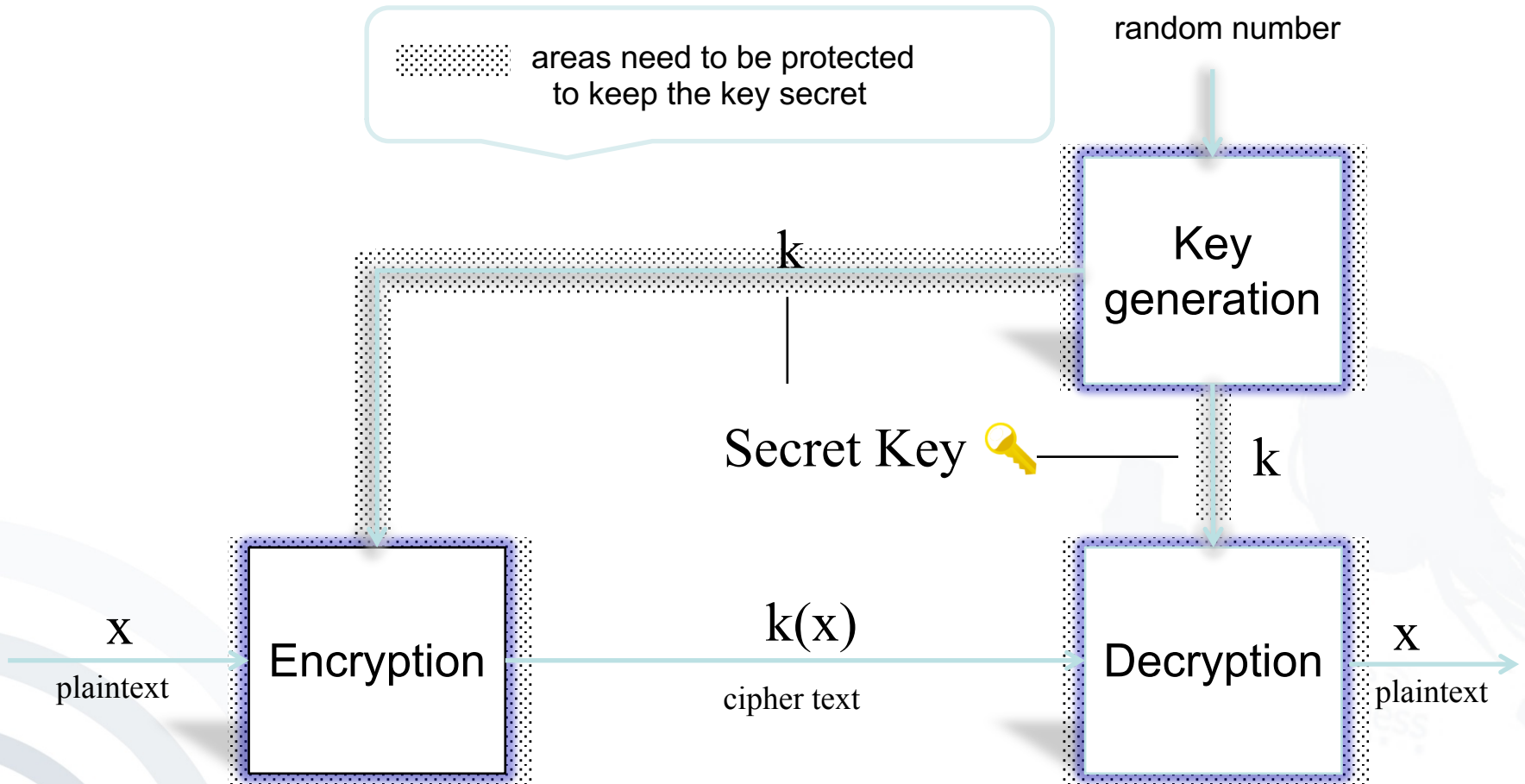
- Intention
 - Confidentiality (secrecy of messages):
encryption systems
 - Integrity (protection from undetected manipulation) and accountability:
authentication systems and **digital signature systems**
- Key distribution
 - **Symmetric:**
Both partners have the same key.
 - **Asymmetric:**
Different (but related) keys for encryption and decryption
- In practice mostly hybrid systems

- Introduction
- Symmetric Cryptosystems
 - General Concept
 - Caesar Cipher
 - AES
 - Advantages and Problems
- Public Key Cryptography

- Classical cryptosystems are usually based on symmetric encryption systems.
- Typical applications
 - confidential storage of user data
 - transfer of data between 2 users who negotiate a key via a secure channel
- Examples
 - Vernam-Code (one-time pad, Gilbert Vernam)
 - key length = length of the plaintext (information theoretically secure)
 - DES: Data Encryption Standard
 - key length 56 bit, so 2^{56} different keys
 - AES: Advanced Encryption Standard (Rijndael, [NIST])
 - 3 alternatives for key length: 128, 192 und 256 bit

- Introduction
- Symmetric Cryptosystems
 - General Concept
 - Caesar Cipher
 - AES
 - Advantages and Problems
- Public Key Cryptography

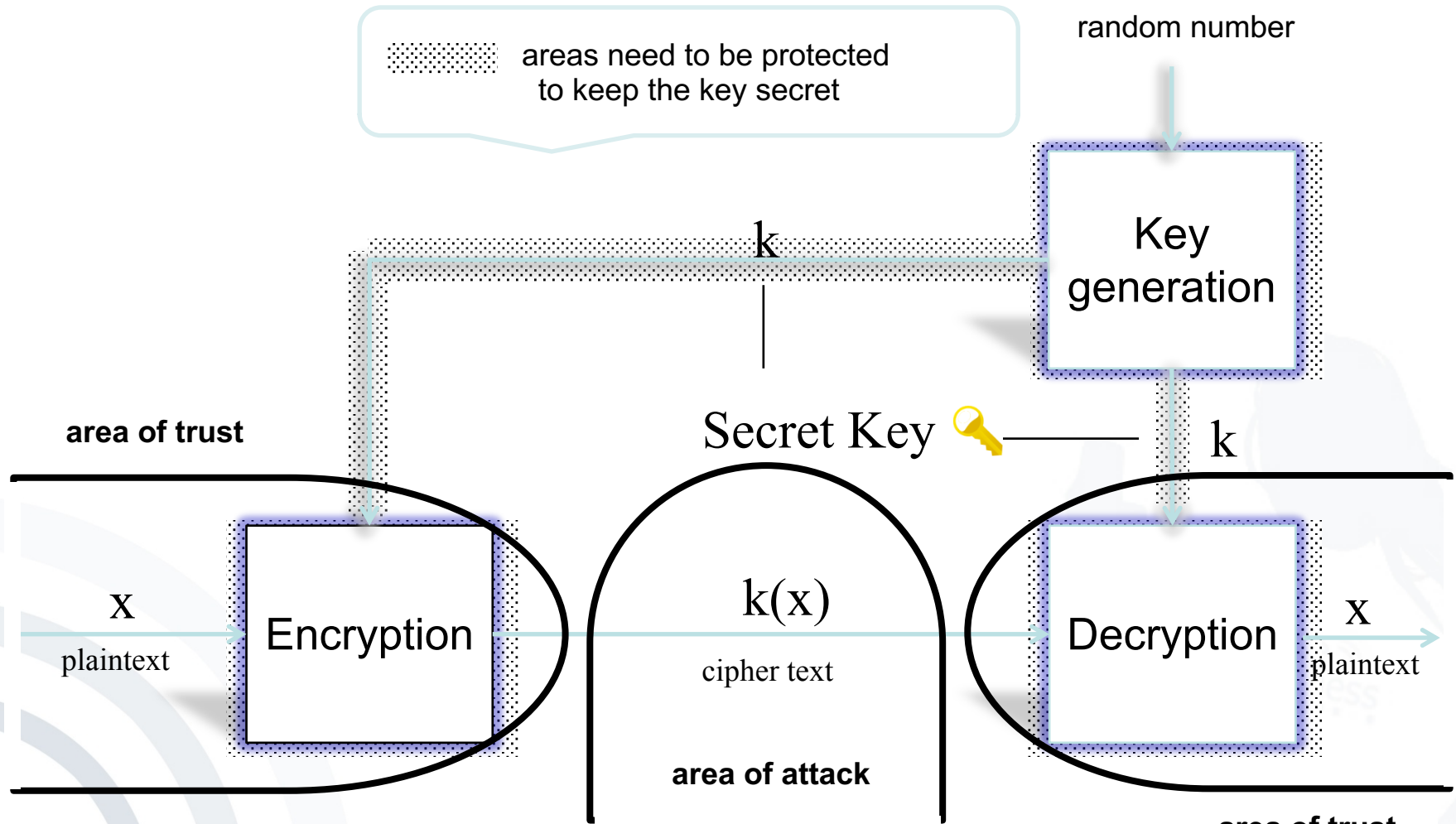
Symmetric Encryption Systems (1)



black box with lock, two equal keys

[based on Federrath and Pfitzmann 1997]

Symmetric Encryption Systems (2)



[based on Federrath and Pfitzmann 1997]

- **Keys have to be kept secret (*secret key* crypto system).**
- **It must not be possible to derive the plaintext or the used keys from the encrypted text (ideally encrypted text is not distinguishable from a numerical random sequence).**
- **Each key shall be equally probable.**
- **In principle each system with limited key length is breakable by testing all possible keys.**
- **Publication of encoding and decoding functions (algorithms) is considered as good style and is trust-building.**
- **Security of cryptosystems should base on the strength of chosen key lengths.**

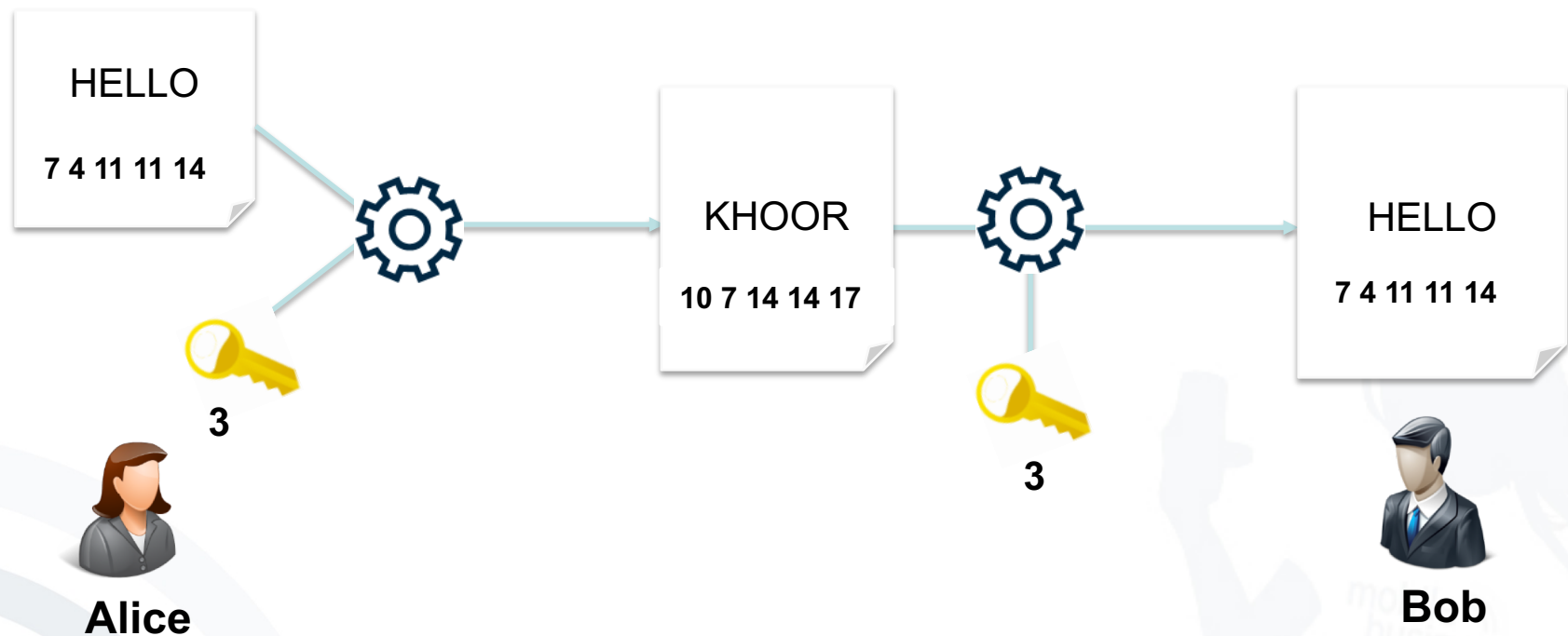
- Introduction
- Symmetric Cryptosystems
 - General Concept
 - Caesar Cipher
 - AES
 - Advantages and Problems
- Public Key Cryptography

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

- We assign a **number** for every **character**.
- This enables us to calculate with letters as if they were numbers.

Caesar Cipher: Example



- Very simple form of encryption.
- The encryption and decryption algorithms are very easy and fast to compute.
- It uses a very limited key space ($n=26$)
- Therefore, the encryption is very easy and fast to compromise.

- Introduction
- Symmetric Cryptosystems
 - General Concept
 - Caesar Cipher
 - AES
 - Advantages and Problems
- Public Key Cryptography

Advanced Encryption Standard

- The Data Encryption Standard (DES) was designed to encipher sensitive but not classified data.
- The standard has been issued in 1977.
- In 1998, a design for a computer system and software that could break any DES-enciphered message within a few days was published.
- By 1999, it was clear that the DES no longer provided the same level of security it had 10 years earlier, and the search was on for a new, stronger cipher.
- AES Rijndael was a winner of U.S. National Institute of Standards and Technology bid for advanced encryptions.
- AES has been approved for Secret or even Top Secret information by the NSA.

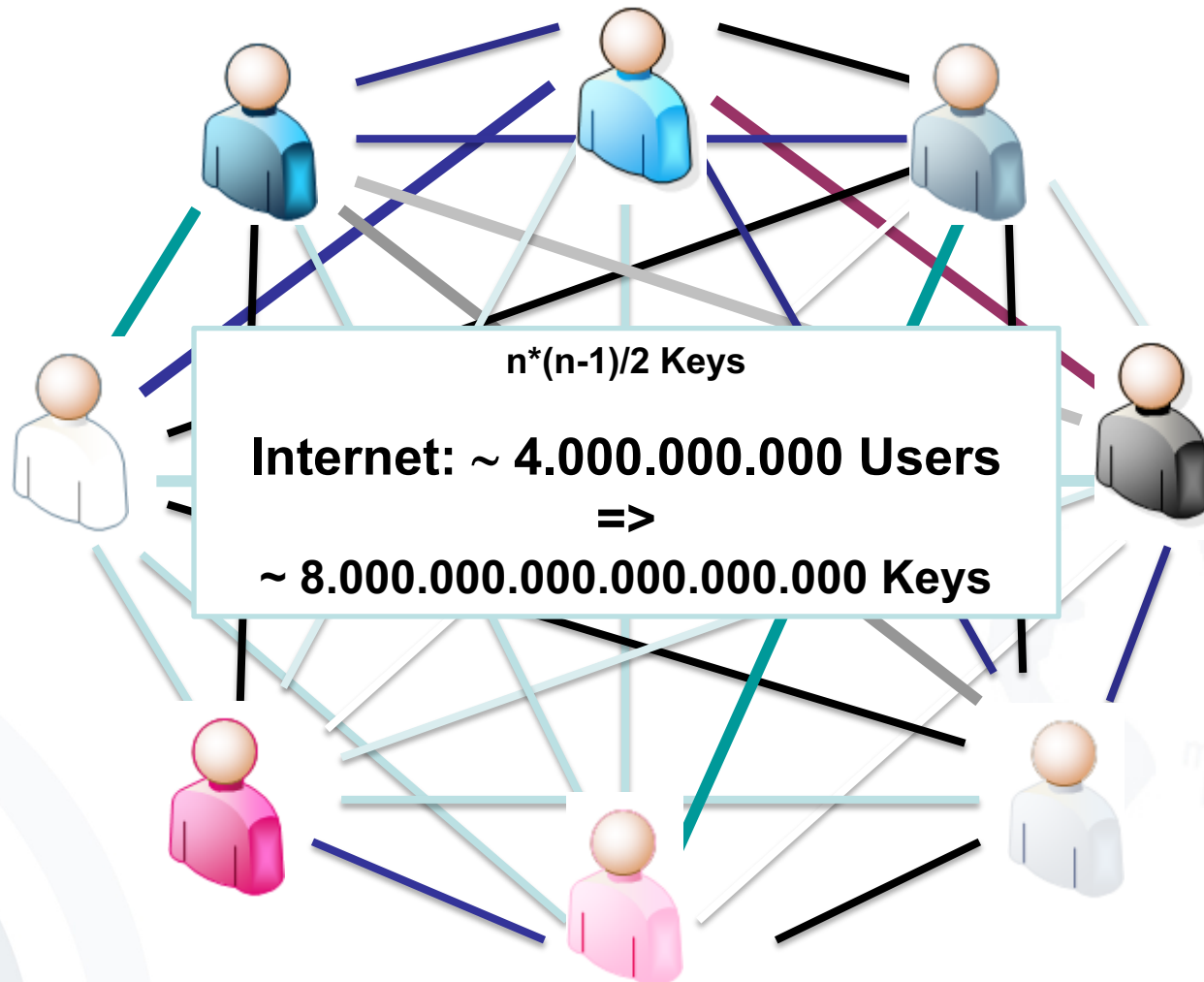
- Introduction
- Symmetric Cryptosystems
 - General Concept
 - Caesar Cipher
 - AES
 - Advantages and Problems
- Public Key Cryptography

Advantage: Algorithms are very fast

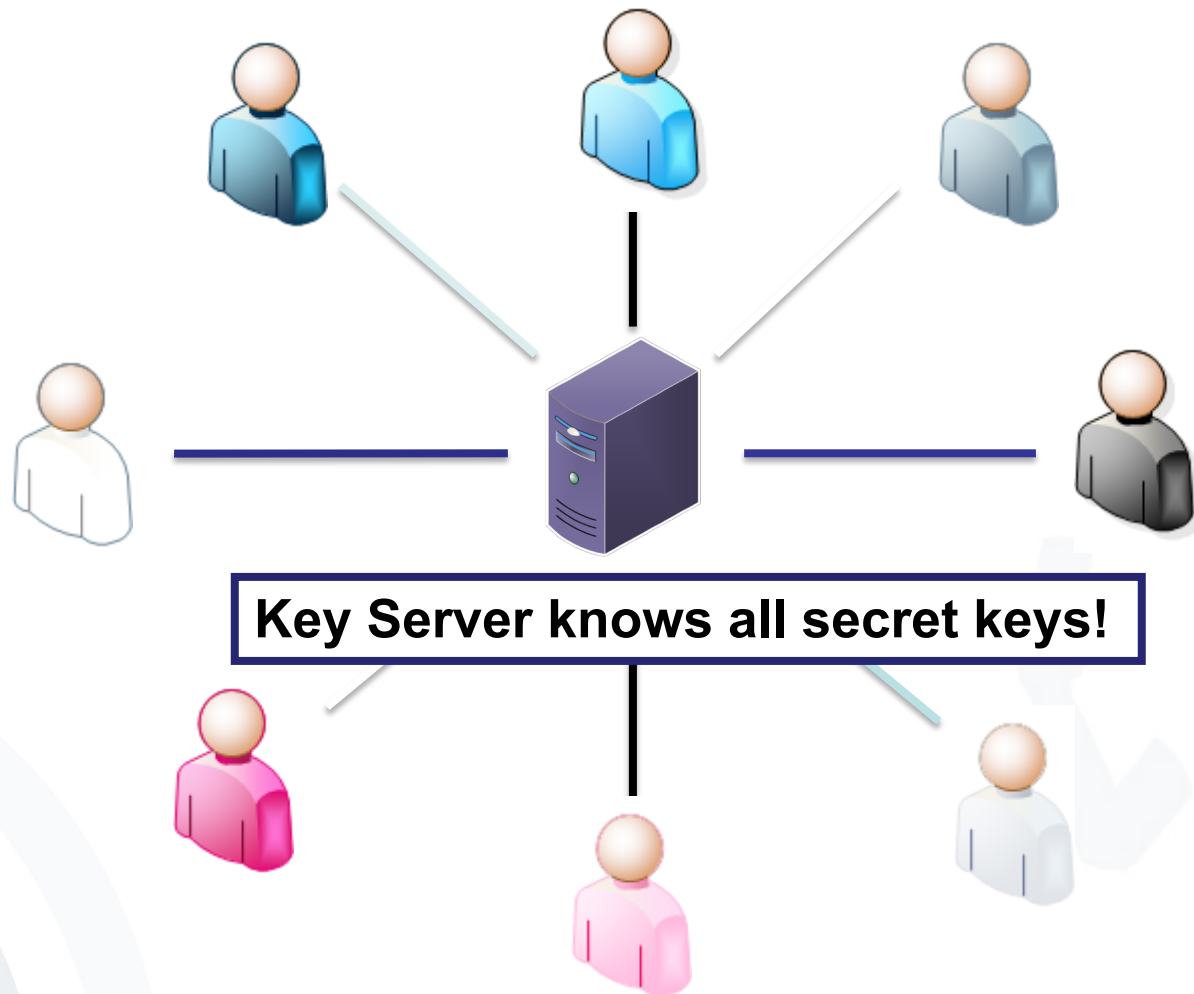
Algorithm	Performance*
RC6	78 ms
SERPENT	95 ms
IDEA	170 ms
MARS	80 ms
TWOFISH	100 ms
DES-edc	250 ms
RIJNDEAL (AES)	65 ms

* Encryption of 1 MB on a Pentium 2.8 GHz, using the FlexiProvider Java)

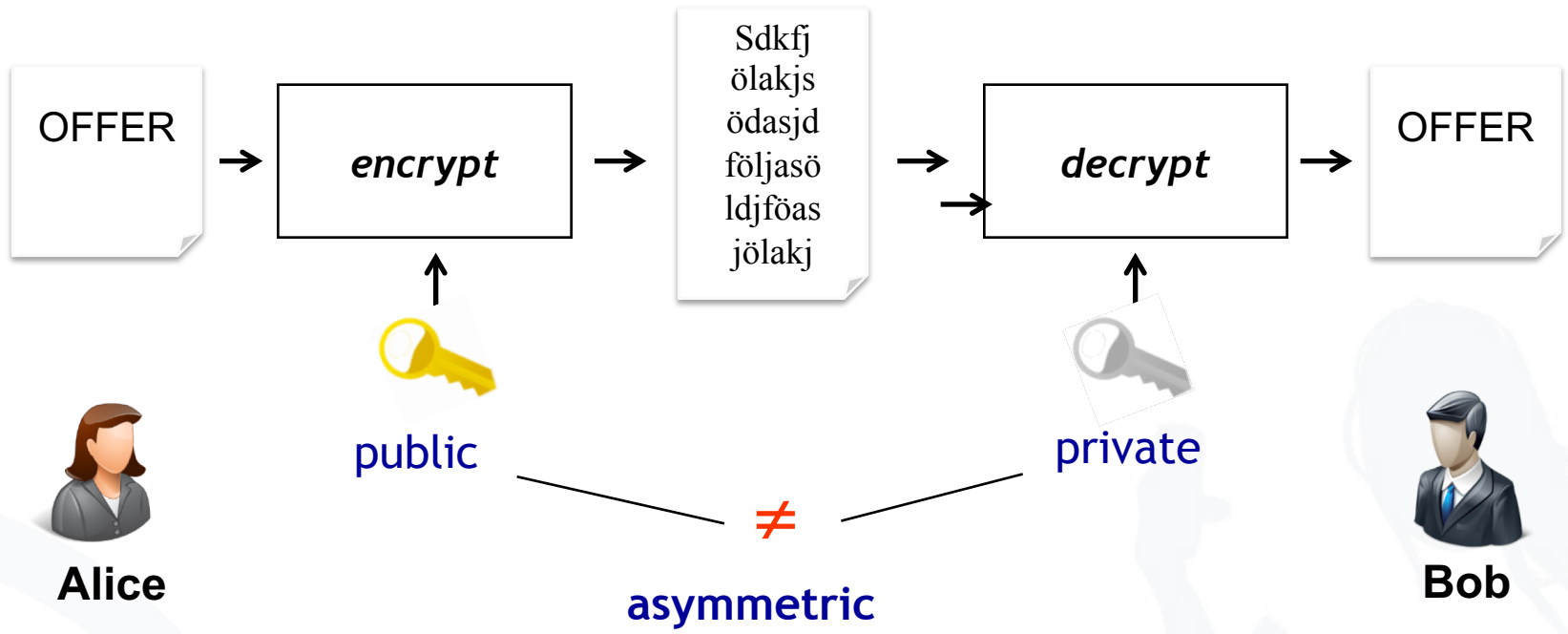
Problems of Symmetric Cryptosystems: Key Exchange



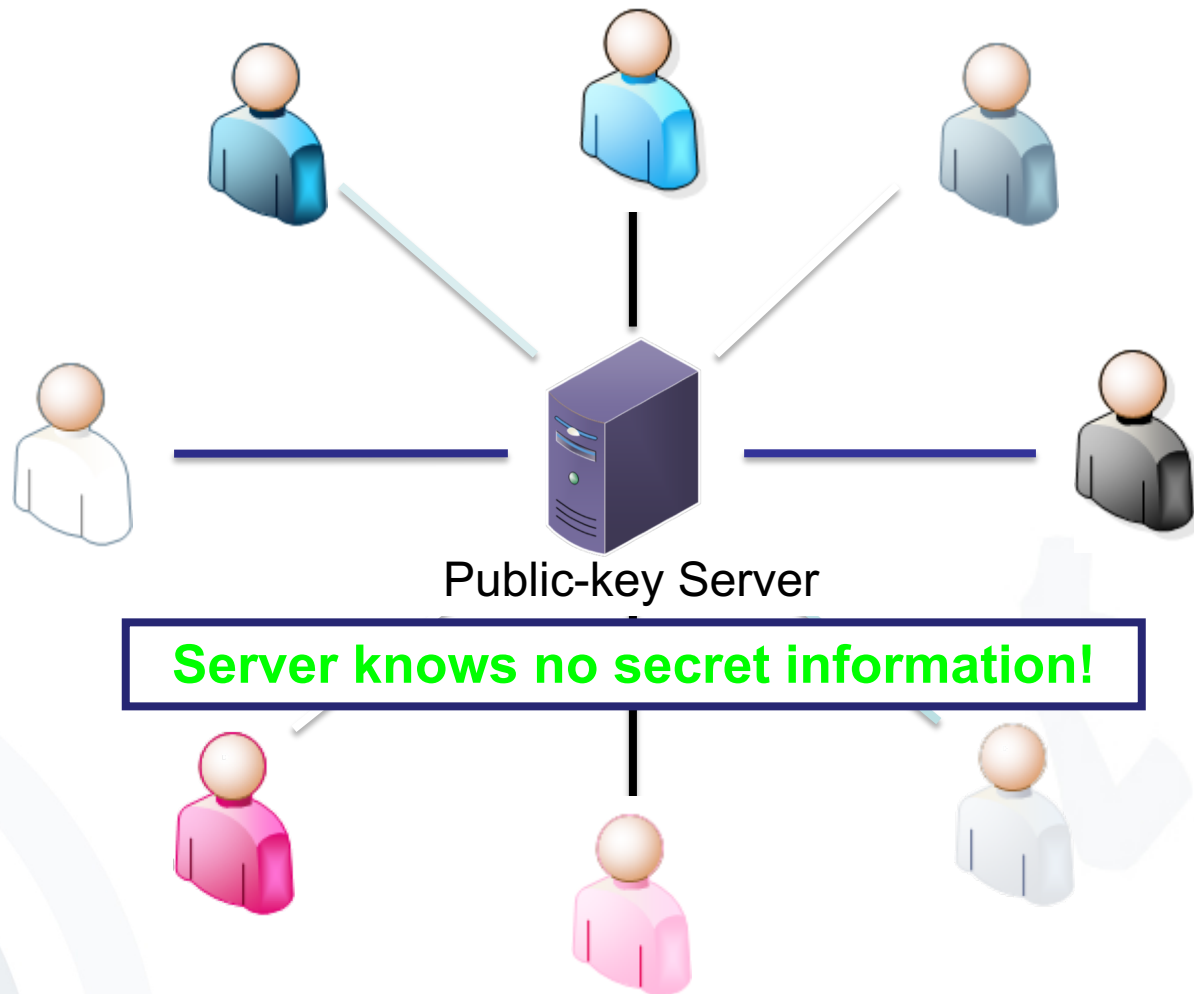
Symmetric Encryption: A Possible Solution



- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP



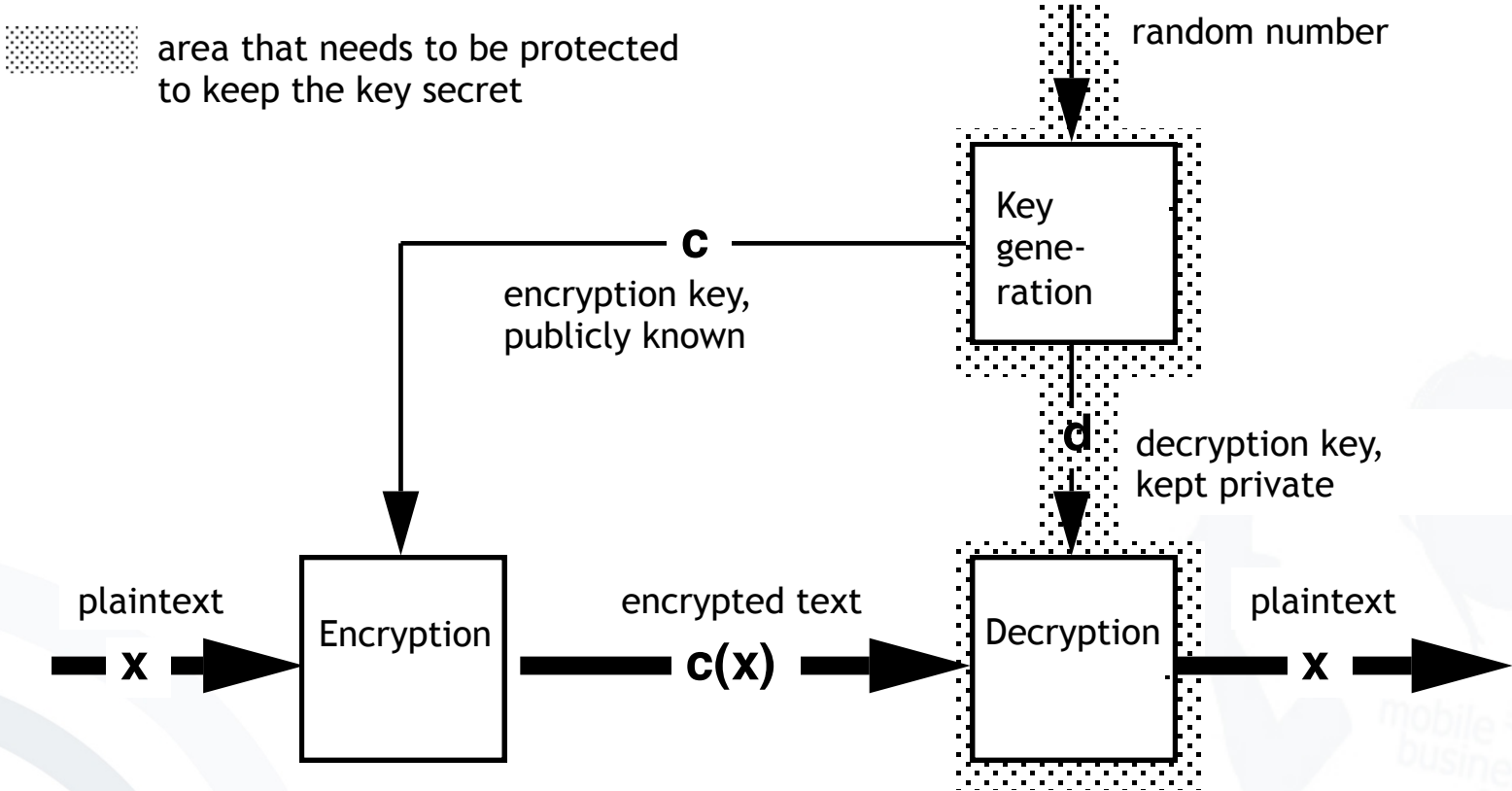
Key Exchange Problem Solved!



- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP

- Public key systems are based on asymmetric encryption.
- Use of ‘corresponding’ key pairs instead of one key:
 - Public key is solely for encryption.
 - Encrypted text can only be decrypted with the corresponding private (undisclosed) key.
- Deriving the private key from the public key is hard (practically impossible).
- The public key can be distributed freely, even via insecure ways (e.g. directory (*public key* crypto system)).
- Messages are encrypted via the public key of the addressee.
- Only the addressee possesses the private key for decoding (and has to manage the relation between the private and the public key).

Asymmetric Encryption Systems



box with slot, access to messages only with a key

[based on Federrath and Pfitzmann 1997]

- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP

- RSA
 - Rivest, Shamir, Adleman, 1978
 - is based on the assumption that the factorization of the product of two (big) prime numbers ($p \cdot q$) is “difficult” (product is basis for the keys)
 - key lengths typically 1024 bit, today rather 2048 [Rivest et al., 1978]
- Diffie-Hellman
 - Diffie, Hellman, 1976, first patented algorithm with public keys
 - allows the exchange of a secret key
 - is based on the “difficulty” of calculating discrete logarithms in a finite field [Diffie, Hellman, 1976]

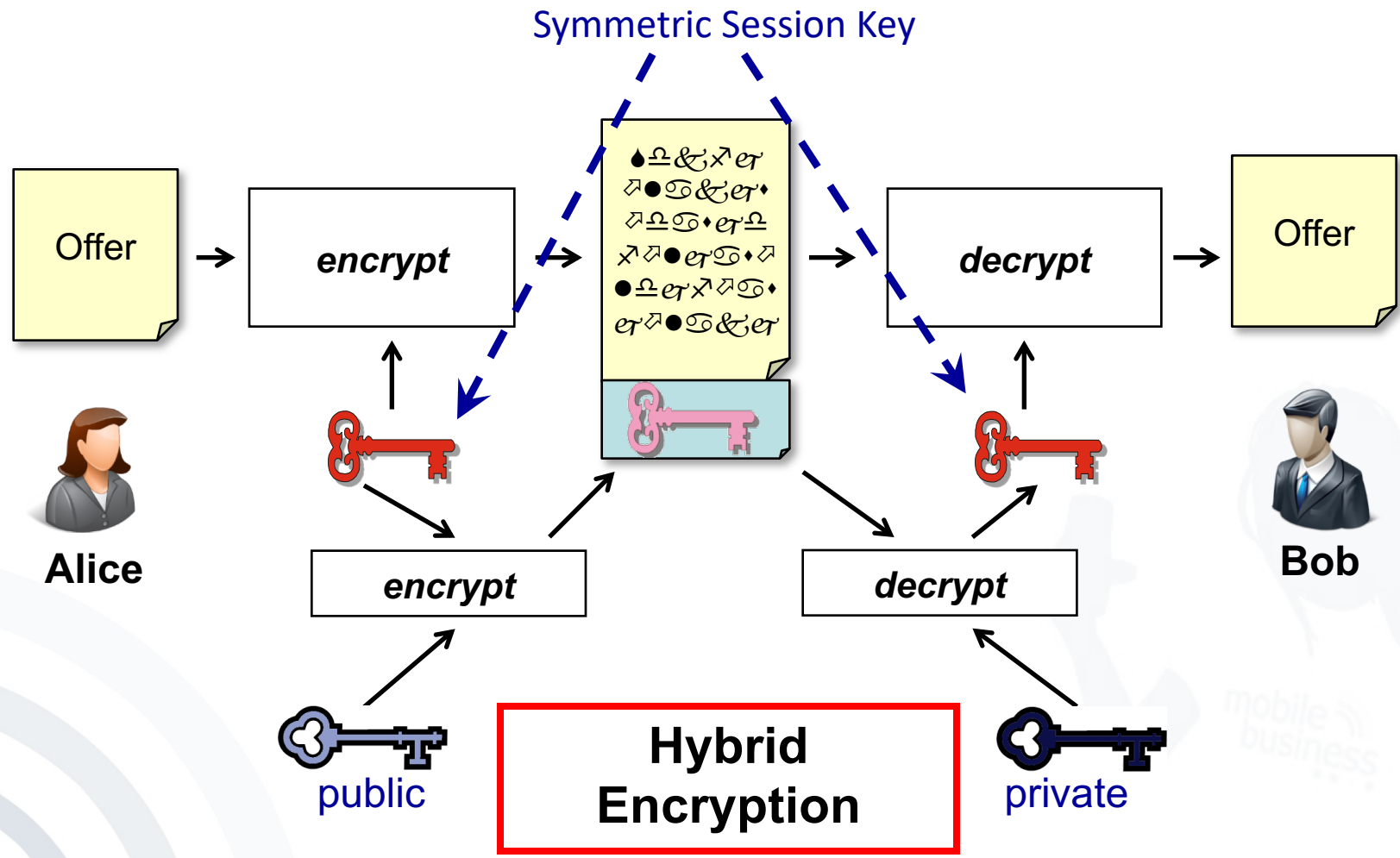
- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP

Algorithm	Performance*	Performance compared to Symmetric encryption (AES)
RSA (1024 bits)	6.6 s	Factor 100 slower
RSA (2048 bits)	11.8 s	Factor 180 slower

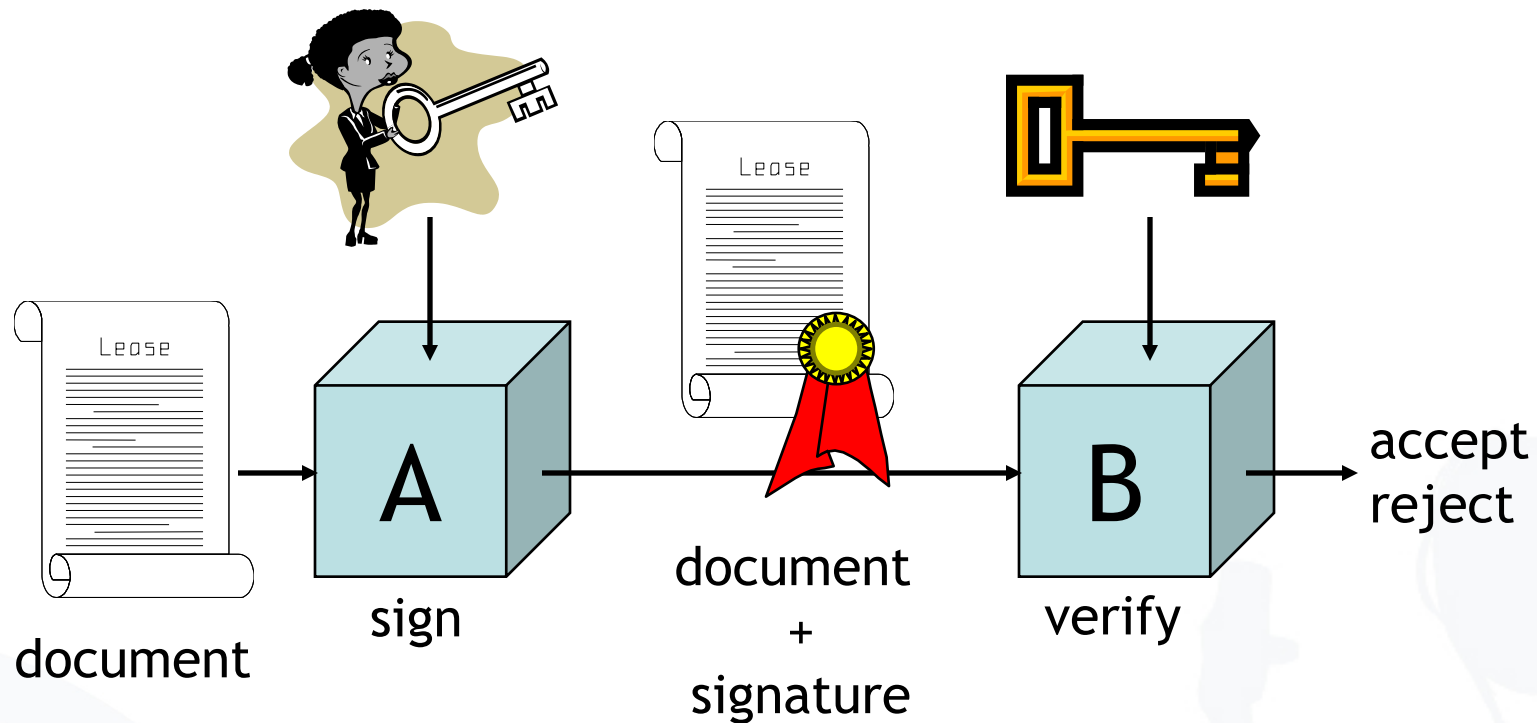
Disadvantage: Complex operations with very big numbers

⇒ Algorithms are very slow.

* Encryption of 1 MB on a Pentium 2.8 GHz, using the FlexiProvider (Java)

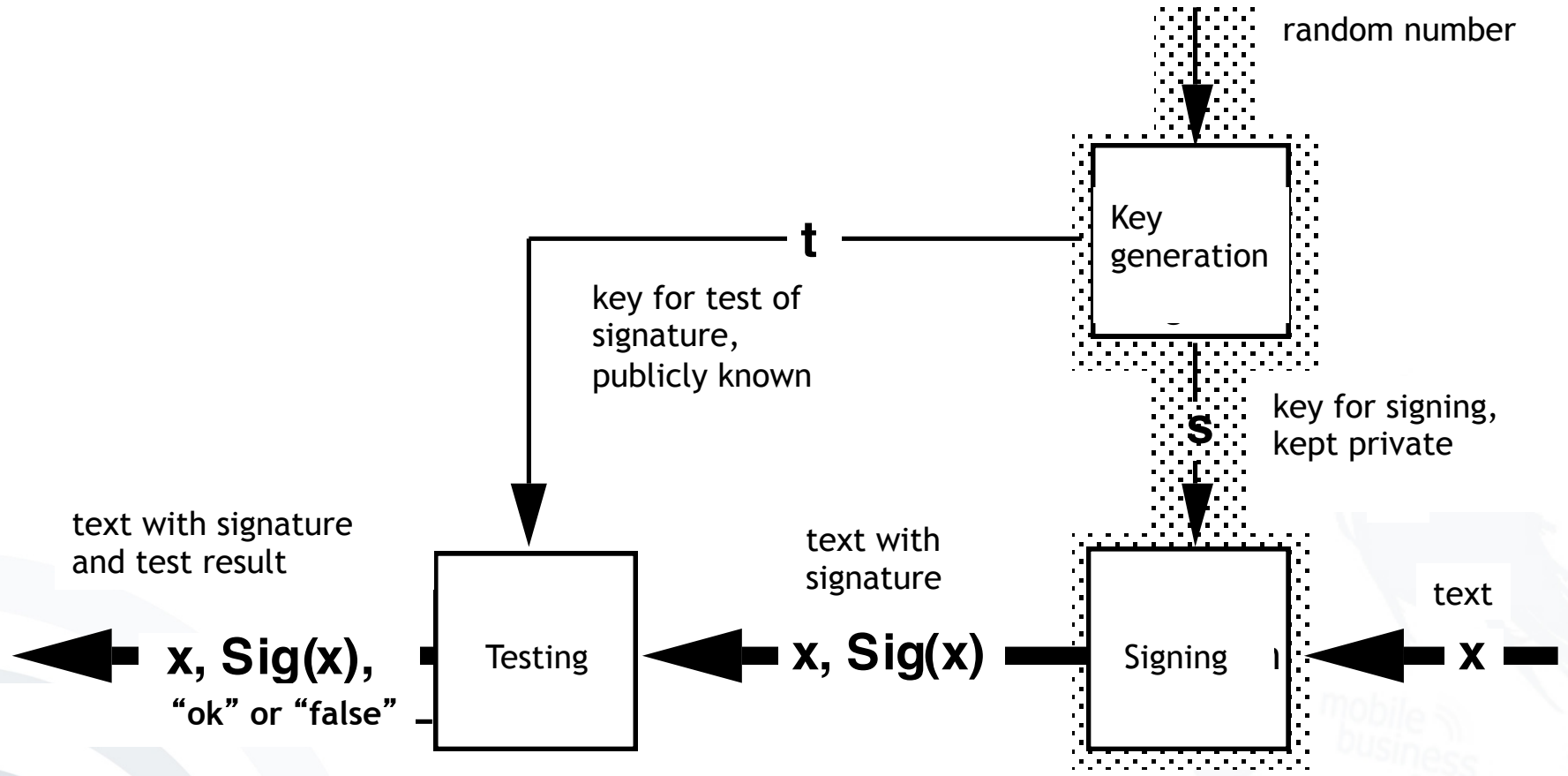


- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP



- ➔ Protect the authenticity and integrity of documents signed by **A**
- ➔ **B** has to get an authentic copy of **A**'s public key.

Asymmetric Signature System



➔ locked glass show-case; just one key to put something in

Von: Heiko Rossnagel
Betreff: Klausur MC1

An: Jan Muntermann
Cc:

-----BEGIN PGP MESSAGE-----
Version: PGP 8.0 - not licensed for commercial use: www.pgp.com

hQCMA5/VPPIP3satAQP+LqxvxFsk4G/TaexpMLX436biwBp6xP8pa89R7ro...
uHEs07/tFrJFQJpPBcUWouy47p4sR2FO+IXqJuJyHp5ExMGIdmQcGXEOs2...
B5TXKtUB8YJ|dpPncK61as78RBP1sq8VDrAlYopEAeqMMw2pkBuoxyo3KCiR...
Ag4DIYlowhVX62wQCAD2L9WAA97xEUBWMET6kR9n5+oafTBF+ROlv6UOz2T...
Alkh23iQ0LI9Drye/uygpcQpT2HhTtZY1AjjudLvi+GsegOlWmBjY8q8G1Y...
kDP3GEanyDiDU6R9F1XF0vxPNMk6Ek8hH6qZ37hhDNDCKxkSjM3nJ2VuuL...
u0uXNA9iAC96dhg7NpvzCJI2J7xRMtuBc9BUI8LXODrvGLwnLtaD5+EvgL1...
dfvQ3NiGrUEQsOHVxwjQdMtr8C09kREYLuAdD7j/05WtsAdbAVMn72PYFOI...
i77MitBfAbxXF0gFS7/b2LccbaK8fx6e1VNFnVO7B/9qpdOGg5WZVP2eQA5...
h2oTOSjWCRp/v5s9Og1aUtcAxdlRAjQPhVsFS2eXXMn9ZzvNIFMh6Ktqpm...
m39jRjPE9Ob/HLjMwPAXUHyneh9QrCX1X5qHORNcjIYVrnQyZGIk8t39059...
crlrhf6ht7SwGgfgGW2aL8HyiFFVDC6e1JfEwzdf6e1JfEwzdf6e1JfEwz...
E1IJGt9QLiwMmXormxcOg+WR2I...
Njwtr+1SkqMCXs+PzcAHDsiuGz...
pE3huhK5cfvulUg7+Oa9SUay4J...
NZncI3vJgkZeZr1bh+pi4dRjsO...
=hC09

-----END PGP MESSAGE-----

heiko rossnagel
frankfurt direkt:
-25306 D-60054 frankfurt

PGPTray - Enter Passphrase

Message was encrypted to the following public key(s):

Heiko Rossnagel <heiko.rossnagel@m-lehrstuhl.de> (DH/2048)
Jan Muntermann <munterma@wiwi.uni-frankfurt.de> (RSA/1024)

Enter passphrase for your private key: Hide Typing

OK Cancel

Text Viewer

*** PGP SIGNATURE VERIFICATION ***

*** Status: Good Signature from Valid Key

*** Signer: Heiko Rossnagel <heiko.rossnagel@m-lehrstuhl.de>
(0x85964FC9)

*** Signed: 26.02.2004 11:40:49

*** Verified: 26.02.2004 11:45:25

*** BEGIN PGP DECRYPTED/VERIFIED MESSAGE ***

Hallo Jan.
My exercises for the "MC1" test are enclosed:

*** END PGP DECRYPTED/VERIFIED MESSAGE ***

Copy to Clipboard OK

- **RSA: Rivest, Shamir, Adleman**

- Asymmetric encryption system which also can be used as a signature system via “inverted use”,
- Message encrypted with the private key (= signing key) gives the signature,
- Decoding with the public key (=testing key) has to produce the message.

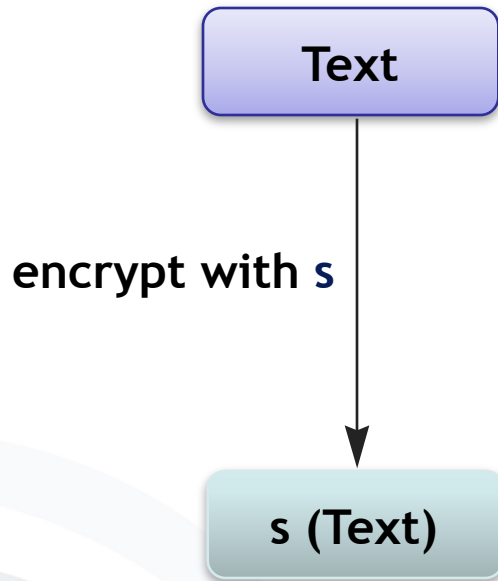
[Rivest et al. 1978]

- **DSA: Digital Signature Algorithm**

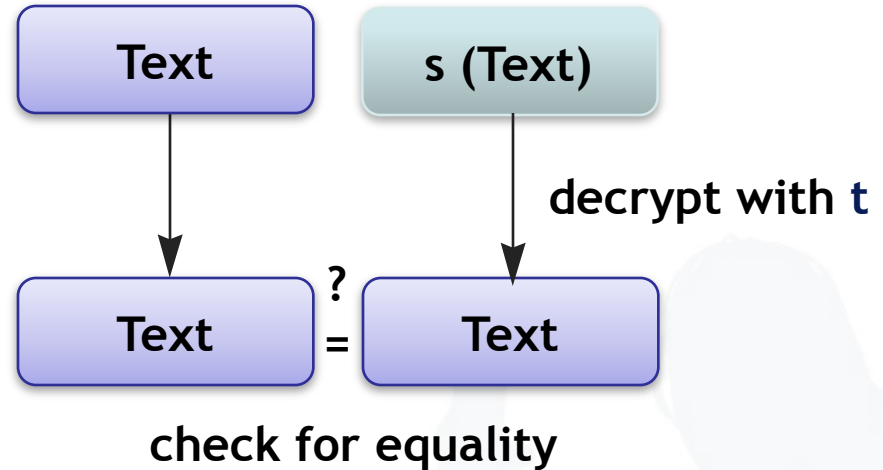
- Determined in the Digital Signature Standard of the NIST (USA),
- Based on discrete logarithms (Schnorr, ElGamal),
- Key length is set to 1024 bit.

Asymmetric Signature System (Simplified Example RSA)

Sender / Signer

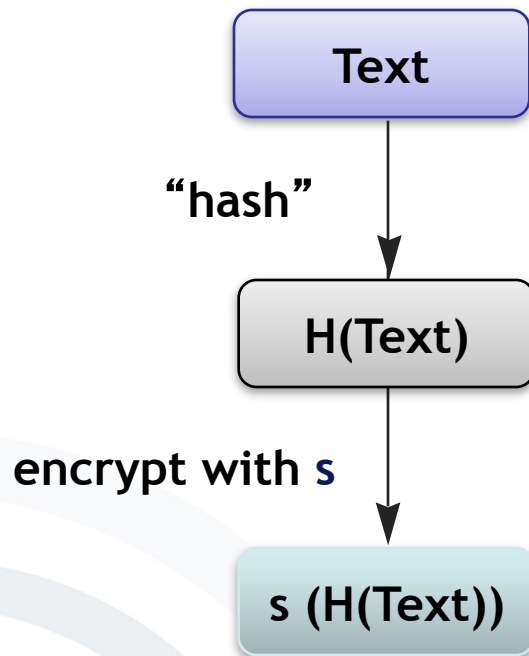


Addressee / Verifier

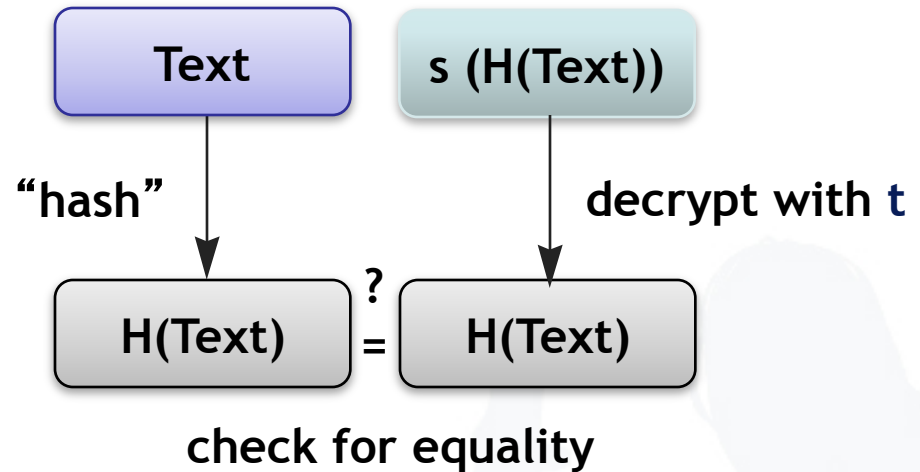


- ⇒ Signing key s only with the sender, test key t public
- ⇒ Example is often mistakenly generalized.

Sender / Signer



Addressee / Verifier

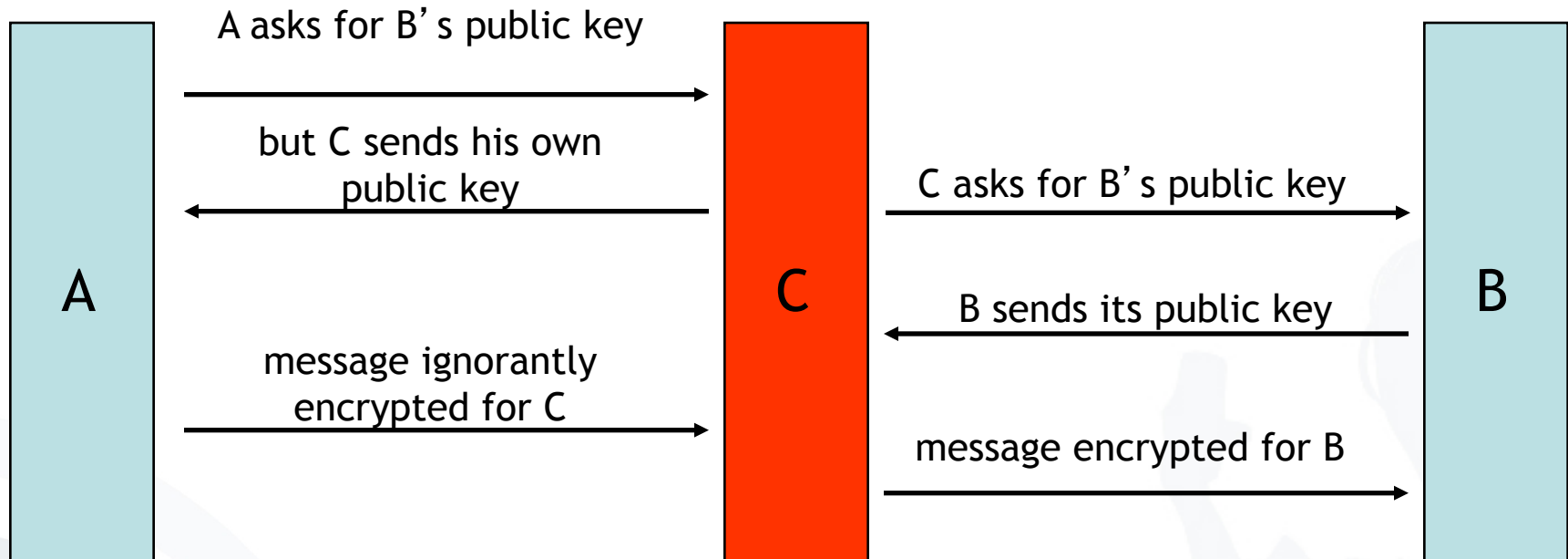


- ⇒ Signing key s only with the sender, test key t public
- ⇒ Example is often mistakenly generalized.

- **General hash functions ($H(s)$)**
 - Transformation of an **input string s** into an **output string h of fixed length** which is called hash value.
 - Example: mod 10 in the decimal system
- **Cryptographic hash functions**
 - Generally require further characteristics
 - $H(s)$ is easily to compute for each s .
 - $H(s)$ must be difficult to invert: In terms of figures it is difficult to compute s from h .
 - Virtual collision freedom: In terms of figures it is difficult to create collisions $H(s_1) = H(s_2)$.
 - Examples: SHA-1, MD5, MD4

- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP

“Man in the middle attack”



- ⇒ Keys are certified: a 3rd person/institution confirms (with its digital signature) the affiliation of the public key to a person.

Certification of Public Keys (1)

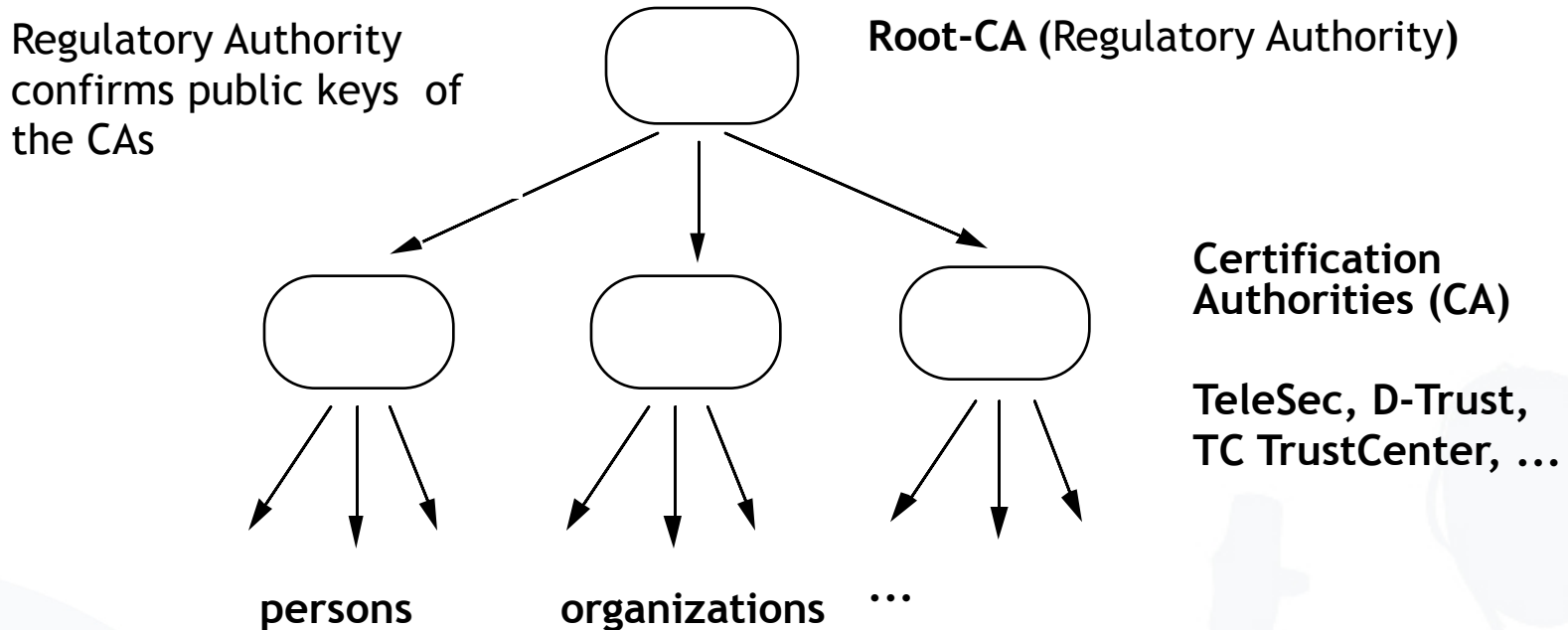
- **B** can freely distribute his own public key.
- But: Everybody (e.g. **C**) could distribute a public key and claim that this one belongs to **B**.
- If **A** uses this key to send a message to **B**, **C** will be able to read this message!
- Thus:
How can **A** decide if a public key was really created and distributed by **B** without asking **B** directly?
 - ➔ Keys get **certified**, i.e. a third person/institution confirms with its (digital) signature the **affiliation of a public key to entity B**.
 - ➔ Public Key Infrastructures (PKIs)

Three types of organization for certification systems (PKIs?):

- Central certification authority (CA)
 - A single CA, keys often integrated in checking software
 - Example: older versions of Netscape (CA = Verisign)
- Hierarchical certification system
 - CAs which in turn are certified by “higher” CA
 - Examples: PEM, Teletrust, infrastructure according to Signature Law
- Web of Trust
 - Each owner of a key may serve as a CA
 - Users have to assess certificates on their own
 - Example: PGP (but with hierarchical overlay system)

Hierarchical Certification of Public Keys

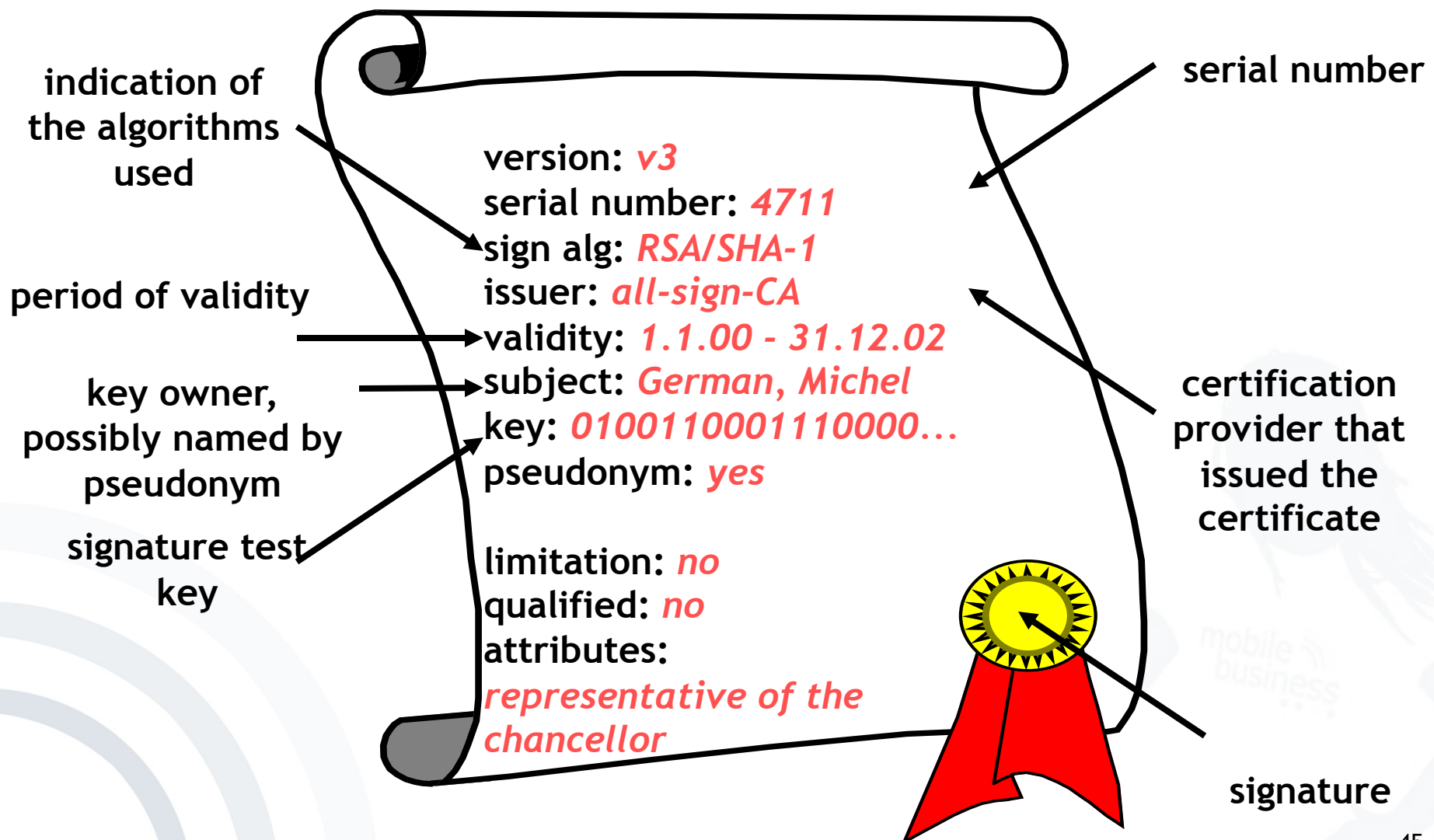
(Example: German Signature Law)



- The actual checking of the identity of the key owner takes place at so called Registration Authorities (e.g. notaries, bank branches, T-Points, ...)
- Security of the infrastructure depends on the reliability of the CAs.

Content of a Key Certificate

(according to German Signature Law and Regulation)

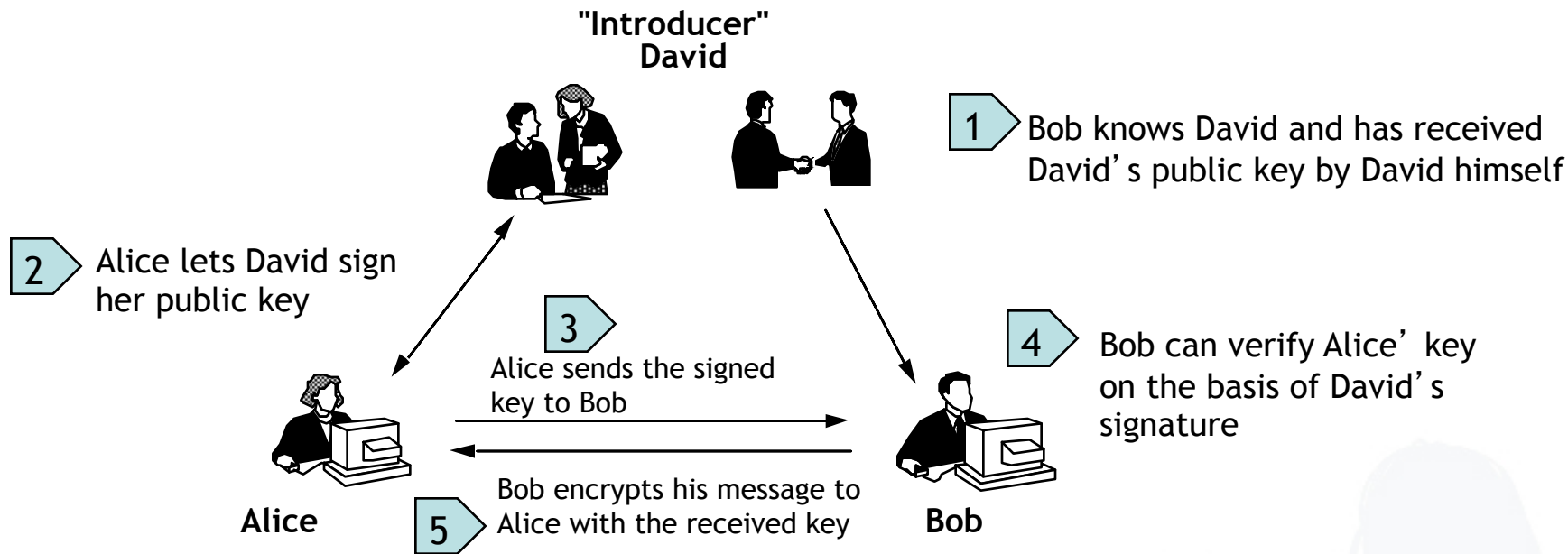


Tasks of a Certification Authority

(according to German Signature Law and Regulation)

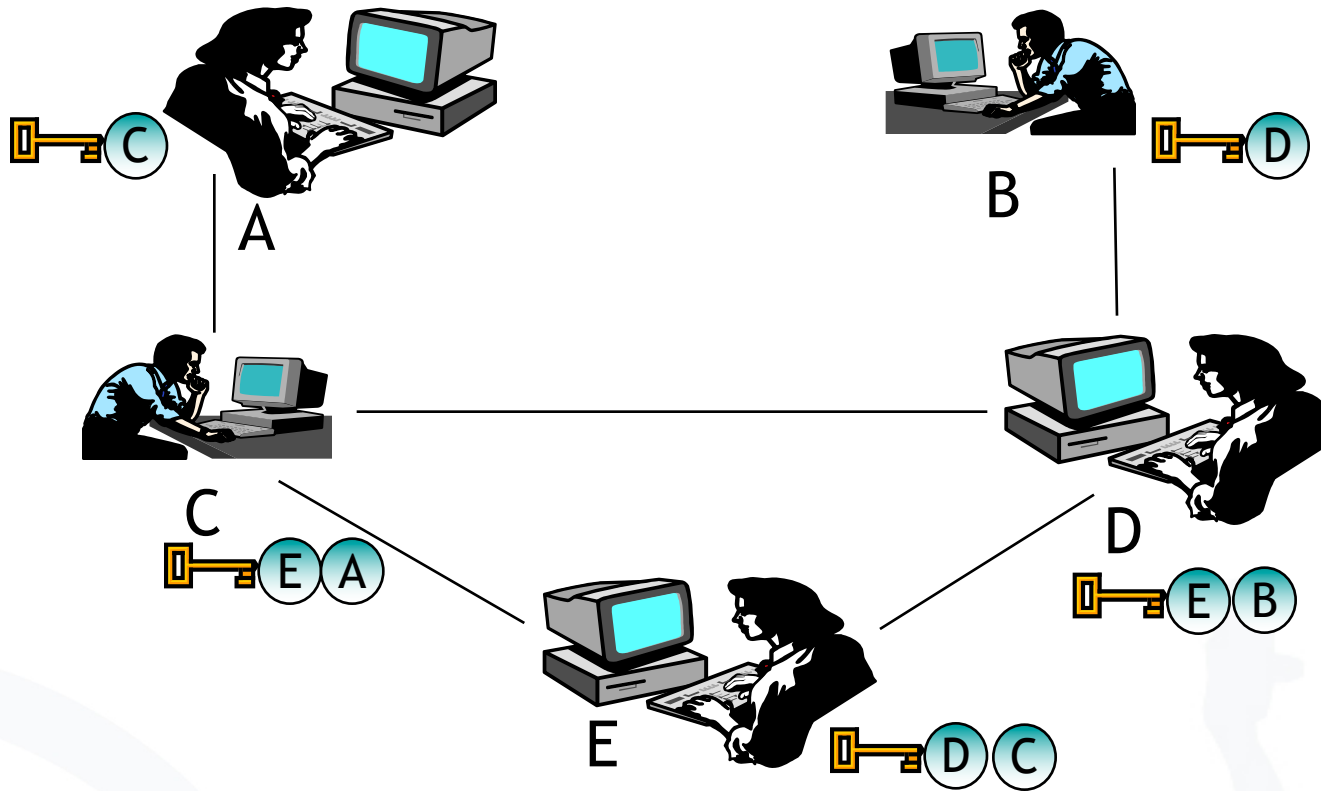
- Reliable identification of persons who apply for a certificate
- Information on necessary methods for fraud resistant creation of a signature
- Provision for secure storage of the private key
 - at least Smartcard (protected by PIN)
- Publication of the certificate (if wanted)
- Barring of certificates
- If necessary issuing of time stamps
 - for a fraud resistant proof that an electronic document has been at hand at a specific time

- Checking of the following items by certain confirmation centers (BSI, TÜVIT, ...)
 - Concept of operational security
 - Reliability of the executives and of the employees as well as of their know-how
 - Financial power for continuous operation
 - Exclusive usage of licensed technical components according to SigG and SigV
 - Security requirements as to operating premises and their access controls
- Possibly license of the regulation authority



- Each user can act as a “CA”.
- Mapping of the social process of creation of trust.
- Keys are “certified” through several signatures.
- Expansion is possible by public key servers and (hierarchical) CAs.

Web of Trust Example



Web of Trust:

- Certification of the public keys mutually by users
- Level of the mutual trust is adjustable.

- Introduction
- Symmetric Cryptosystems
- Public Key Cryptography
 - General Concept
 - Algorithms
 - Hybrid Systems
 - Digital Signature
 - Key Management
 - Example: PGP

- PGP = Pretty Good Privacy
 - De facto-Standard for freely accessible e-mail encryption systems on the Internet
 - First implementation by Phil Zimmermann
 - Long trial against Phil Zimmermann because of suspicion of violation of export clauses
 - In U.S., free version in cooperation with MIT (agreement with RSA because of the patent)
 - Meanwhile commercialized: www.pgp.com
 - Gnu Privacy Guard (GPG): non-commercial Open Source variant (OpenPGP, RFC2440)

OpenPGP: Encrypt Message

The screenshot shows an email client interface with the following elements:

- Main Window:** 'Verfassen: MB II Slides'. Recipient: 'Christian Kahl <christian.kahl@m-le...>'. Subject: 'MB II Slides'. Body text: 'Hi Christian, please find attached the MB II slides for lect...'
- OpenPGP-Schlüssel auswählen Dialog:** A list of keys for encryption.

Empfänger für Verschlüsselung wählen	Vertrauen	Ablauf...	Schlüssel-ID
<input checked="" type="checkbox"/> Benutzer-ID			
<input checked="" type="checkbox"/> Christian Kahl <christian.kahl@m-lehrstuhl.de>	absolutes Ver...		14E21EDA
<input type="checkbox"/> Alexander Boettcher ("Nur wenige wissen, wie viel man wissen muss, um zu...)	abgelaufen	02.09.2006	8D539C6E
<input type="checkbox"/> Alexander Boettcher <ab764283@inf.tu-dresden.de>	-		A63325B3
<input type="checkbox"/> Alexander Boettcher <ab764283@os.inf.tu-dresden.de>	abgelaufen	11.10.2005	F26EE0CD
<input type="checkbox"/> Andre Meixner <s4538672@inf.tu-dresden.de>	-		7C433232
<input type="checkbox"/> ...	-		7E39E652
<input type="checkbox"/> ...	-		52B1B05D
<input type="checkbox"/> ...	-		A0D40924
<input type="checkbox"/> ...	-		79B42C58
<input type="checkbox"/> ...	-		B06F3816
<input type="checkbox"/> ...	-		0789B57F
<input type="checkbox"/> ...	-	11.04.2011	165A5F90
<input type="checkbox"/> ...	-		9347DB3C
<input type="checkbox"/> ...	-	20.02.2009	48CC64C2
<input type="checkbox"/> ...	-		8EF041F1
<input type="checkbox"/> ...	-		289E7DB2
<input type="checkbox"/> Katja Liesebach <katja.liesebach@m-chair.net>	absolutes Ver...		C4495AF0
<input type="checkbox"/> Katrin Borcea <kati@inf.tu-dresden.de>	-		F7C207CE
- OpenPGP-Bestätigung Dialog:** A confirmation box with a question mark icon. Text: 'VERSCHLÜSSELTE Nachricht an folgende Empfänger senden: christian.kahl@m-lehrstuhl.de'. Note: 'Hinweis: Die Nachricht wurde mit folgenden Benutzer-IDs / Schlüsseln verschlüsselt: 0x42B8B29914E21EDA, 0x23EE4D96C4495AF0'. Buttons: 'Ja', 'Nein'.

OpenPGP: Decrypt Message

Betreff: MB II Slides
Von: [Katja Liesebach <katja.liesebach@m-chair.net>](mailto:katja.liesebach@m-chair.net)
Datum: 19:18
An: [Christian Kahl <christian.kahl@m-chair.net>](mailto:christian.kahl@m-chair.net)

```
-----BEGIN PGP MESSAGE-----
Charset: ISO-8859-15
Version: GnuPG v1.4.7 (MingW32)
Comment: Using GnuPG with Mozilla

hQEOAzxc3rSs71RREAQAoa4NK8beVOV
iEsWpmlxA11HIpTZtIKd9ecdjVlOFOJ
6xxXLtS6PkSb0k5nKkMZ1147F80IrvW
/0md5jClR8N/NJeuSfsW6w1LUpTVHQQ
zQAvcf2AvjqHHw4UldKW8ewB3GG4zqD
XxkOviAC+ADTcPgF5FvYpPbEiKS9D8d
gZrBd07YIfdH0oMBgga9K
JMwn2/s+Mn6AqNVhdPJuh8VaFvLW+up
3GZ+msGd3v4P80Z1VBS4sc
jOkaydJkxKqriLNqgiY39ltyZUtowl
Jaa+uPK2pq1A311DHEoqm8y
cFJW5KxpgNFGyixn7wU6I+e7d6Df8Q
==eEkh
-----END PGP MESSAGE-----
```

OpenPGP-Eingabe

Bitte geben Sie Ihre OpenPGP-Passphrase oder SmartCard-PIN ein

Erst nach 5 Minuten

OK

Betreff: MB II Slides
Von: [Katja Liesebach <katja.liesebach@m-chair.net>](mailto:katja.liesebach@m-chair.net)
Datum: 19:18
An: [Christian Kahl <christian.kahl@m-chair.net>](mailto:christian.kahl@m-chair.net)

Hi Christian,

please find attached the MB II slides for lecture 7.

--

Dipl.-Medien-inf. Katja Liesebach

Johann Wolfgang Goethe University Frankfurt a. M.
 Institute of Business Informatics
 Chair of Mobile Business and Multilateral Security
 Graefstr. 78, D-60486 Frankfurt a. M., Germany

Internet: <http://m-chair.net>
 Fon: +49 (69) 798-25313
 Fax: +49 (69) 798-25306

- Certification of public keys by users: “Web of Trust”
- Differentiation between ‘validity’ and ‘trust’
 - ‘Trust’ :
trust that a person / an institution signs keys only if their authenticity has really been checked
 - ‘Validity’ :
A key is valid for me if it has been signed by a person / an institution I trust (ideally by myself).
- Support through key-servers:
 - Collection of keys
 - Allocation of ‘validity’ and ‘trust’ remains task of the users
- Path server:
Finding certification paths between keys

OpenPGP-Schlüssel verwalten

Datei Bearbeiten Anzeigen Schlüssel-Server Erzeugen

Zeige Schlüssel, deren Benutzer-ID oder Schlüssel-ID folgendes enthalten:

Benutzer-ID	Vertrauen	Ablauf-D...	Typ
Alexander Boettcher ("Nur wenige wissen, wie viel man wissen muss, um z...	abgelaufen	02.09.2006	öffentlich
⊕ Alexander Boettcher <ab764283@inf.tu-dresden.de>	absolutes Vertrauen		öffentlich
⊕ Alexander Boettcher <ab764283@os.inf.tu-dresden.de>	abgelaufen	11.10.2005	öffentlich
Andre Meixner <s4538672@inf.tu-dresden.de>	-		öffentlich
Andreas Albers <andreas.albers@m-lehrstuhl.de>	absolutes Vertrauen		öffentlich
Andreas Pfitzmann <pfitza@inf.tu-dresden.de> NO LEGAL RELEVANCE	absolutes Vertrauen		öffentlich
André Deuker <andre.deuker@m-lehrstuhl.de>	absolutes Ve		
Birgit Pretscheck <birgit.pretscheck@gmx.net>	-		
Christian Kahl <christian.kahl@m-lehrstuhl.de>	absolutes Ve		
⊕ Denis Royer <me@myasterisk.de>	absolutes Ve		
Elvira Koch <Elvira.Koch@m-lehrstuhl.de>	volles Vertra		
Felix Göpfert (keine Passphrase) <fg798936@inf.tu-dresden.de>	-		
⊕ Hagen Wahrig <wahrig@web.de>	-		
⊕ Jan Zibuschka <zibuschka@m-lehrstuhl.de>	absolutes Ve		
⊕ Kai Rannenber <Kai.Rannenber@m-lehrstuhl.de>	absolutes Ve		
Katja Liesebach <katja.liesebach@inf.tu-dresden.de>	-		
Katja Liesebach <katja.liesebach@m-chair>	absolutes V		
⊕ Katrin Borcea <kati@inf.tu-dresden.de>	-		
Marco Lehmann <m99@gmx.li>	-		
⊕ Mathias Staab <mathias.staab@arcor.de>	-		
Mike Beremann (dienstlich, TU Dresden, unbeschrnkt altia) <mb41@inf.t...	-		

Schlüsseleigenschaften

Primäre Benutzer-ID:

Schlüssel-ID:

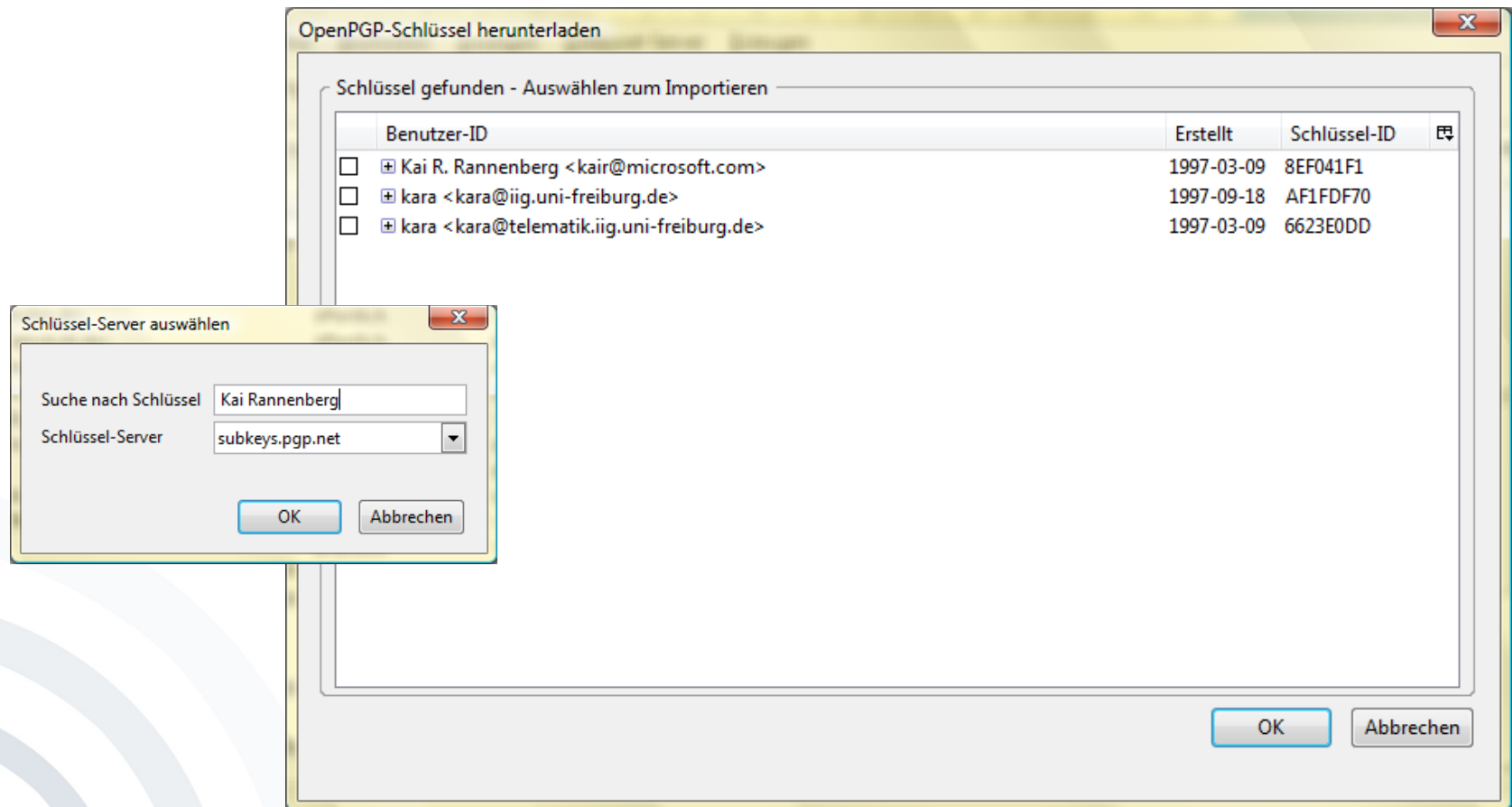
Typ:

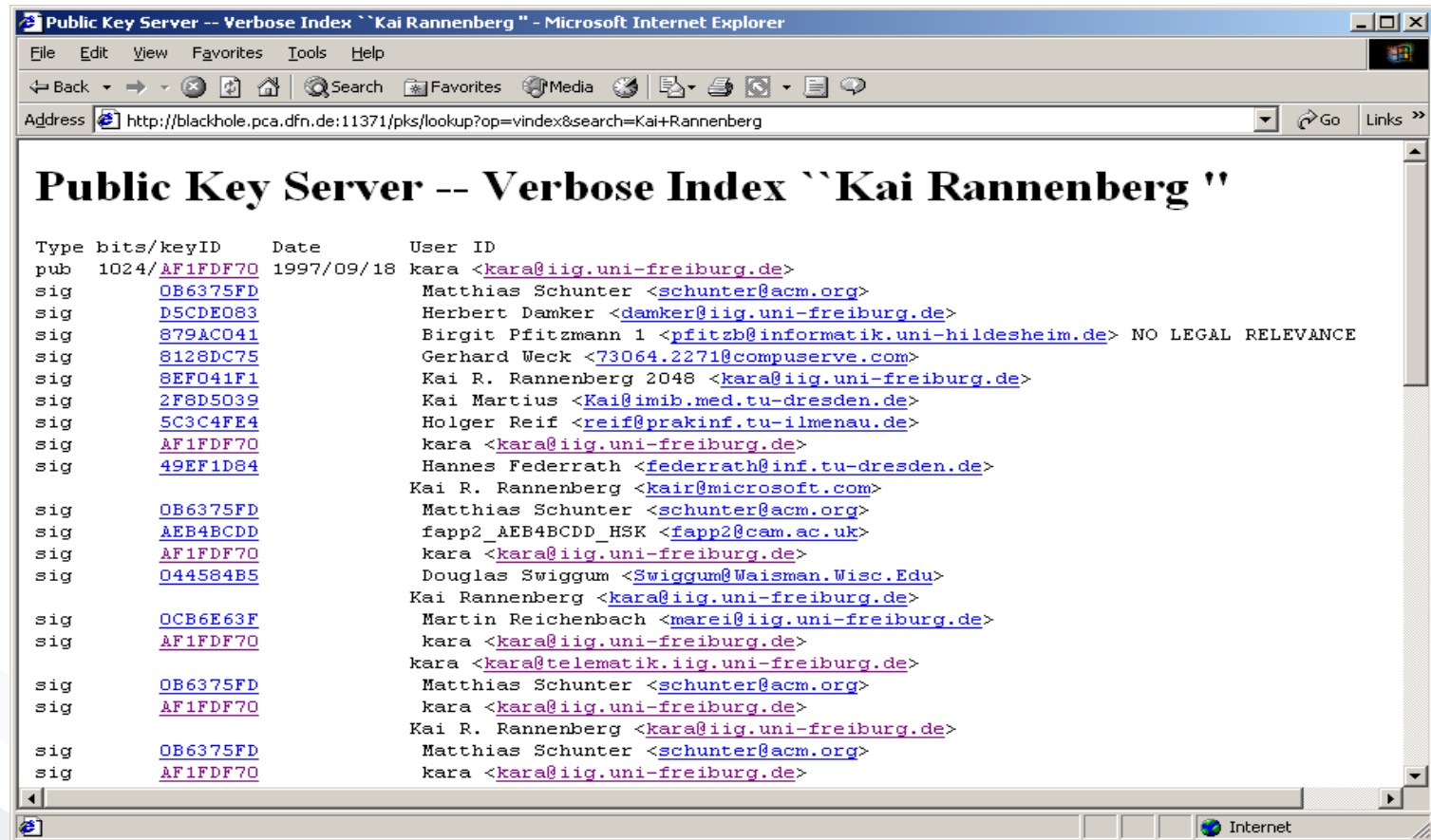
Vertrauen:

Besitzer-Vertrauen:

Fingerabdruck:

Typ	ID	Algo...	Stär...	Erzeugt	Ablauf-Datum
Unterschlüssel	0x98F0...	ELG	2048	07.09.2007	nie





- Network of public-key servers:
 - www.cam.ac.uk/pgpnet/email-key-server-info.html
 - <http://pgp.mit.edu/>

- Brute-Force-Attacks on the pass phrase
 - PGPCrack for conventionally encrypted files
- Trojan horses, changed PGP-Code
 - e.g. predictable random numbers, encryption with an additional key
- Attacks on the computer of the user
 - Not physically deleted files
 - Paged memory
 - Keyboard monitoring
- Analysis of electromagnetic radiation
- Non-technical attacks
- Confusion of users [Whitten, Tygar 1999]

“Anybody who asserts that a problem is readily solved by encryption, understands neither encryption nor the problem.”

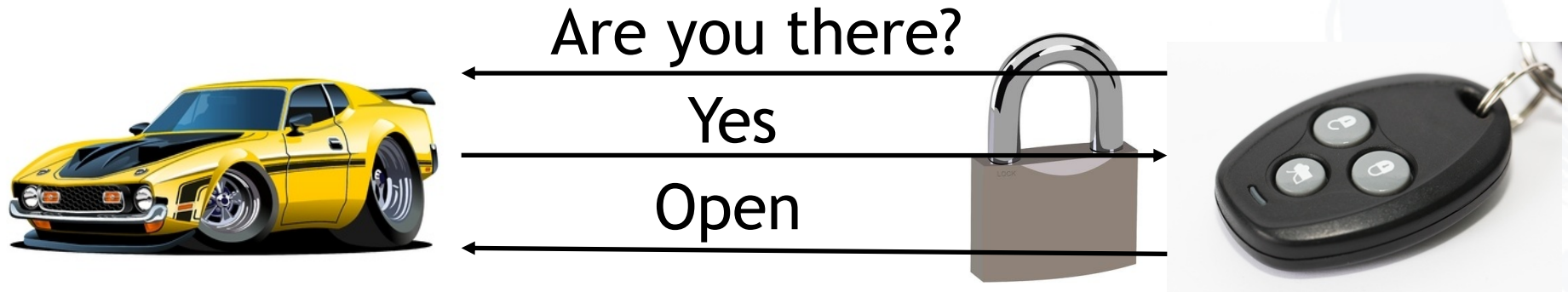
(Roger Needham /
Butler Lampson)



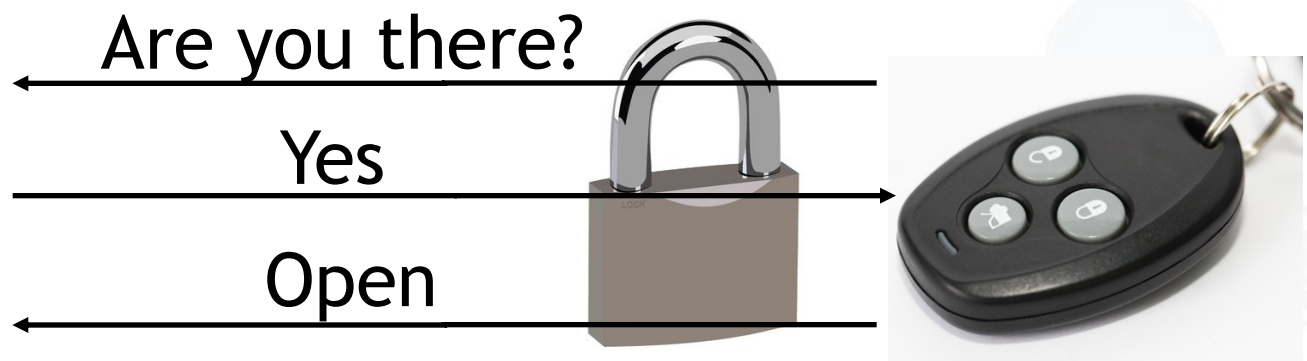
[Marshall Symposium 1998] [Randell 2004]

Example: Keyless Entry System

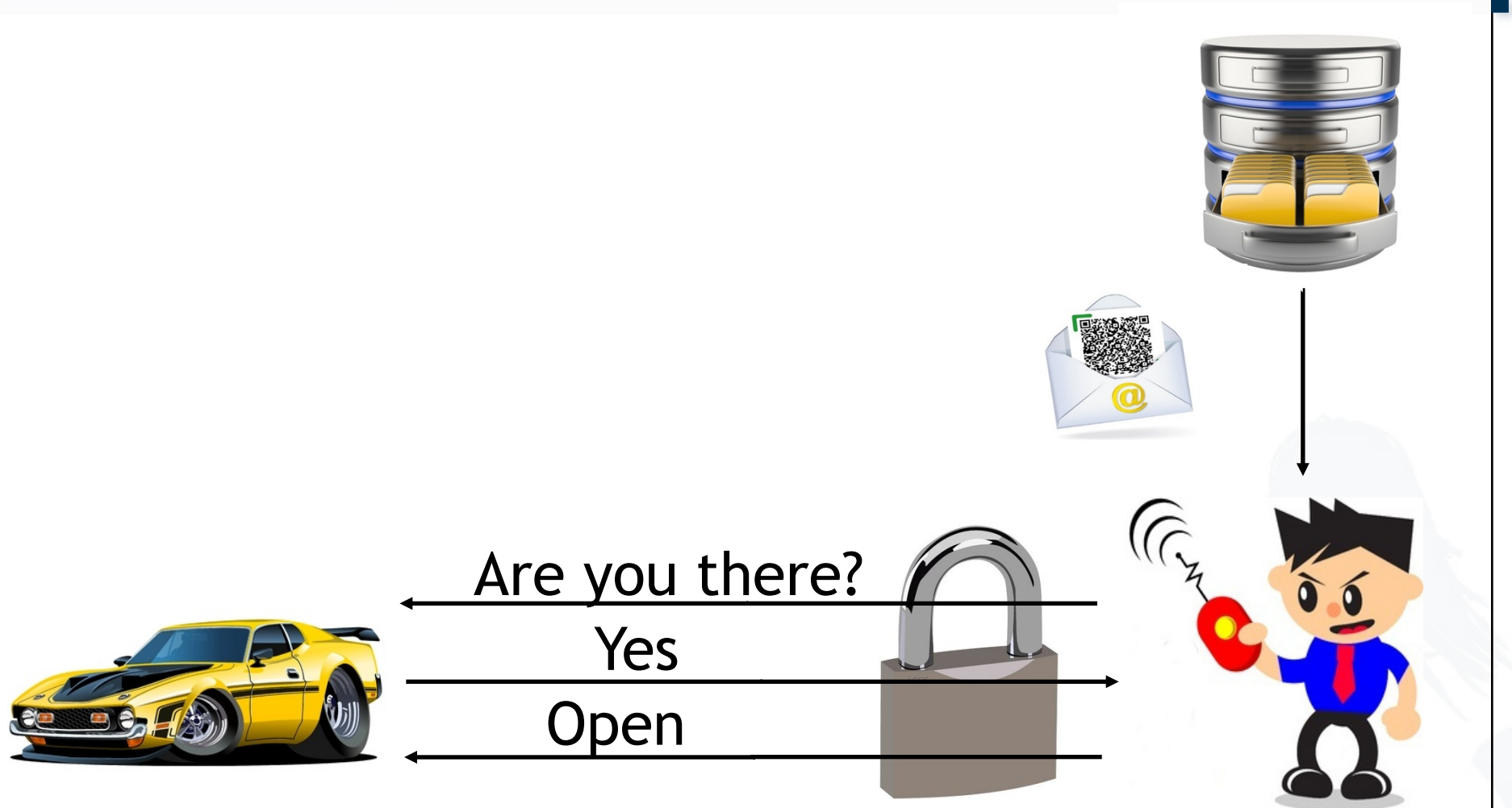
- Solution: Protect communication with crypto?
- e.g. symmetric cryptography + hash/signature



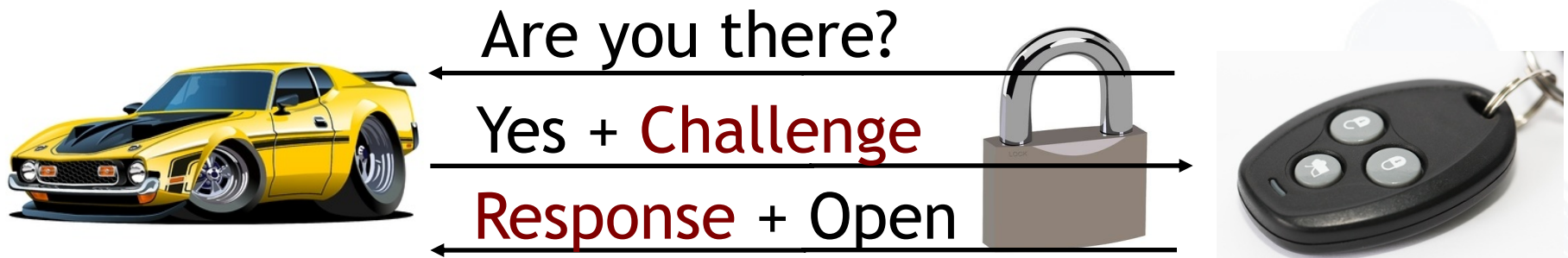
Replay Attack: Eavesdrop



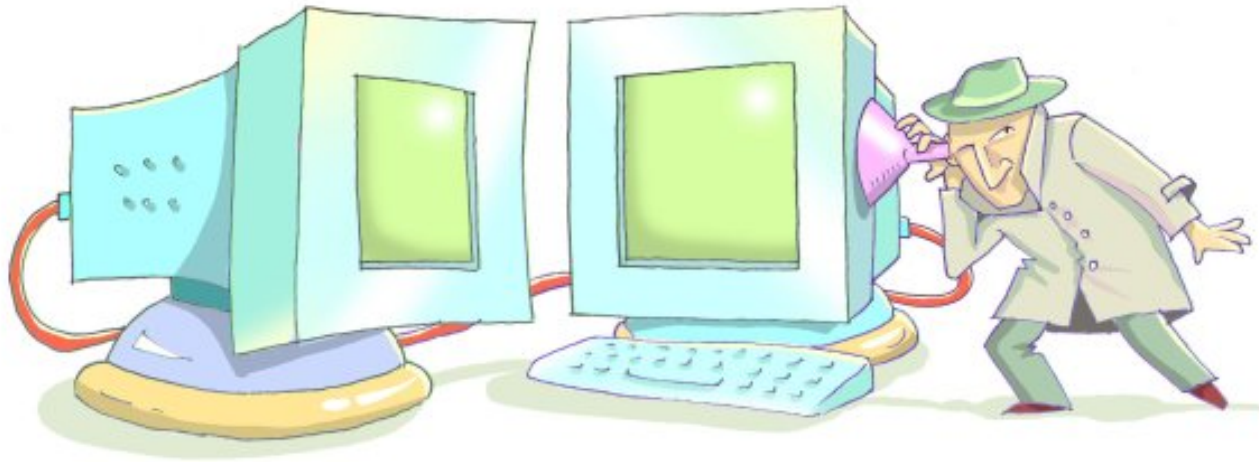
Replay Attack: Replay



- e.g. Challenge-Response helps

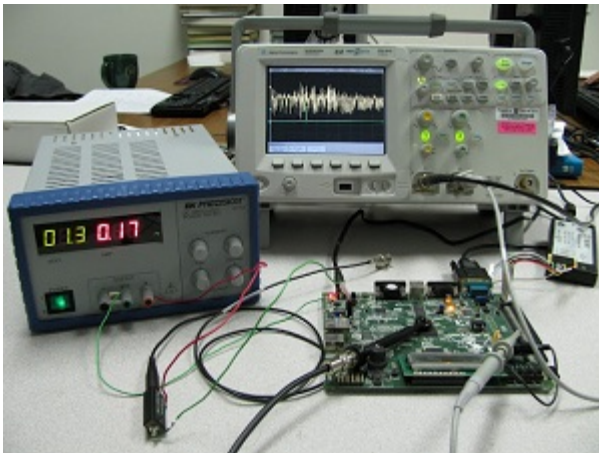


- A secure cryptoalgorithm does not imply that the implementation is also secure

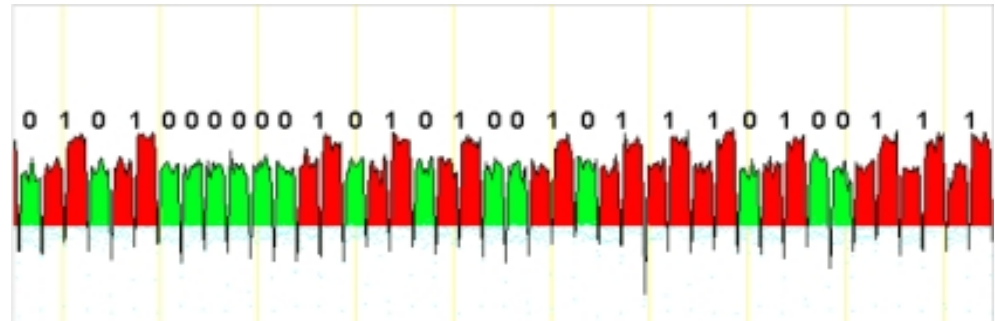


Source: Eran Tromer

- Side-Channels: Time, Power, Noise, Radiation, ...

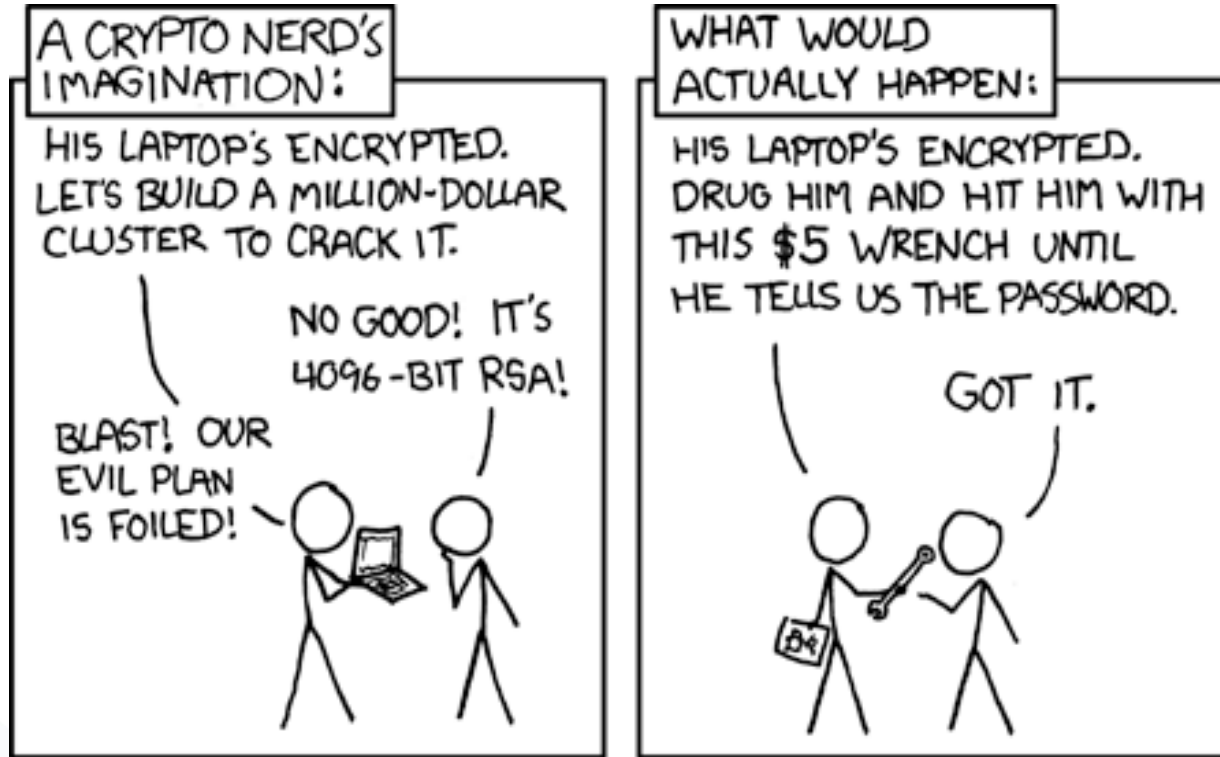


Source: CESCA



Source: Gilbert Goodwill

- Other data (side-channel) leaks information
- Conclusion on processed bits possible



Source: <https://xkcd.com/538/>

1. Florencio, D. & Herley, C., 2007. A large-scale study of web password habits. *Proceedings of the 16th international conference on World Wide Web - WWW '07*, p.657. Available at:
<http://portal.acm.org/citation.cfm?doid=1242572.1242661>.
2. Florêncio, D., Herley, C. & Coskun, B., 2007. Do strong web passwords accomplish anything? *Proceedings of the 2nd USENIX workshop on Hot topics in security (HOTSEC'07)*, p.10. Available at:
<http://portal.acm.org/citation.cfm?id=1361419.1361429>.
3. Norberg, P.A., Horne, D.R. & Horne, D.A., 2007. The Privacy Paradox: Personal Information Disclosure Intentions versus Behaviors. *Journal of Consumer Affairs*, 41(1), pp.100-126.

- Bishop, M. (2005) Introduction to Computer Security, Addison Wesley, Boston, pp. 97-116.
- Cremers, Cas, et al. "Distance hijacking attacks on distance bounding protocols." 2012 IEEE Symposium on Security and Privacy. IEEE, 2012.
- Diffie, W. and Hellman, M. E. (1976) New Directions in Cryptography, *IEEE Transactions on Information Theory* (22:6), pp. 644-654.
- Federrath, H. and Pfitzmann, A. (1997) Bausteine zur Realisierung mehrseitiger Sicherheit, in: G. Müller and A. Pfitzmann (Eds.): *Mehrseitige Sicherheit in der Kommunikationstechnik*, Boston, Addison Wesley, pp. 83-104.
- The Marshall Symposium: Address Roger Needham, May 29, 1998, Rackham School of Graduate Studies, University of Michigan web.archive.org/web/20081201182254/http://www.si.umich.edu/marshall/docs/p201.htm, accessed 2015-04-15.
- Randell, B. (2004) *Brief Encounters*; Pp. 229-235 in: Andrew Herbert, Karen Spärck Jones: *Computer Systems: Theory, Technology, and Applications*; New York, Springer 2004
- Rivest, R. L.; Shamir, A. and Adleman, L. (1978) A Method for Obtaining Digital Signatures and Public Key Cryptosystems, *Communications of the ACM* (21:2), pp. 120-126.
- Whitten, A. and Tygar, J. (1999) *Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0*. In: Proceedings of the 9th USENIX Security Symposium, August 1999, www.gaudior.net/alma/johnny.pdf