Assignment 3 - Cryptography

Information & Communication Security
(SS 2016)

M.Sc. Welderufael B. Tesfay

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

Agenda

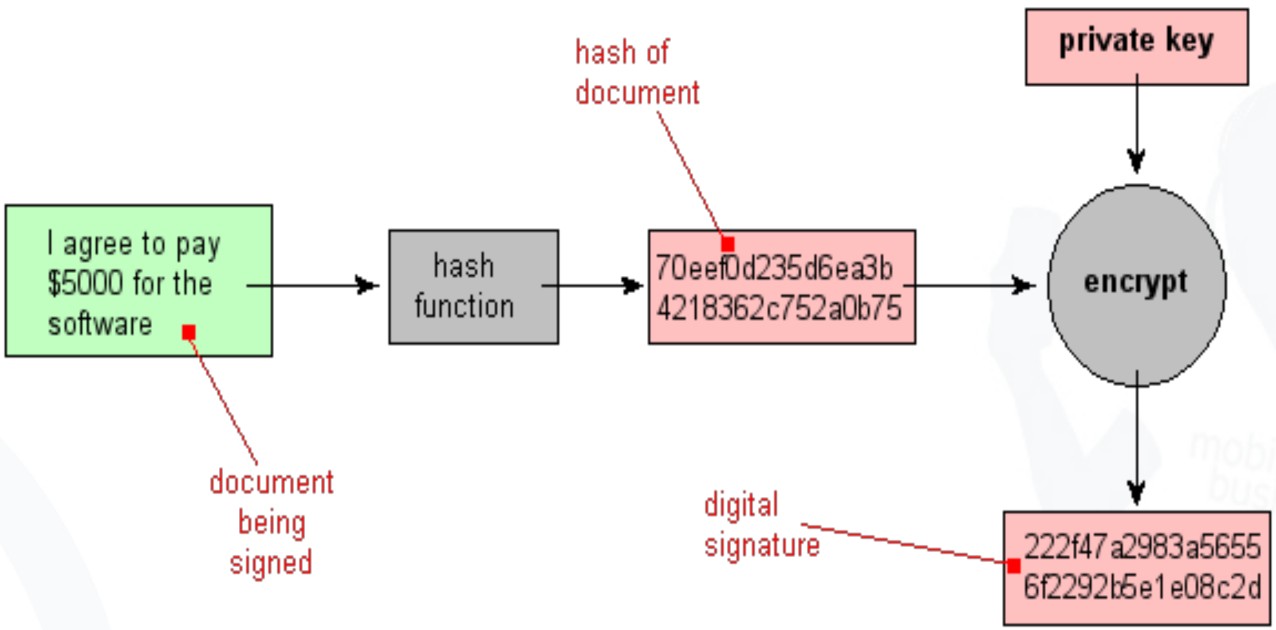
- Hash functions and digital signatures
- Caesar cipher
- Symmetric vs. asymmetric ciphers
- Stream ciphers (Vernam code)

Exercise 1: PGP

- Install PGP Email Desktop (trial version) or a similar software for mail encryption on your system. Create a new key pair, and send a signed and encrypted message to welderufael.tesfay@m-chair.de containing your newly created public key and a short summary of your experiences.
- PGP can be downloaded from http://www.symantec.com/business/desktop-email
  - Practical exercise, no solution required, check lecture notes for overview of PGP
  - Be careful to only send your public key
  - If you haven't done this yet, try it, sending encrypted mail is useful, and we want you to be able to do it.

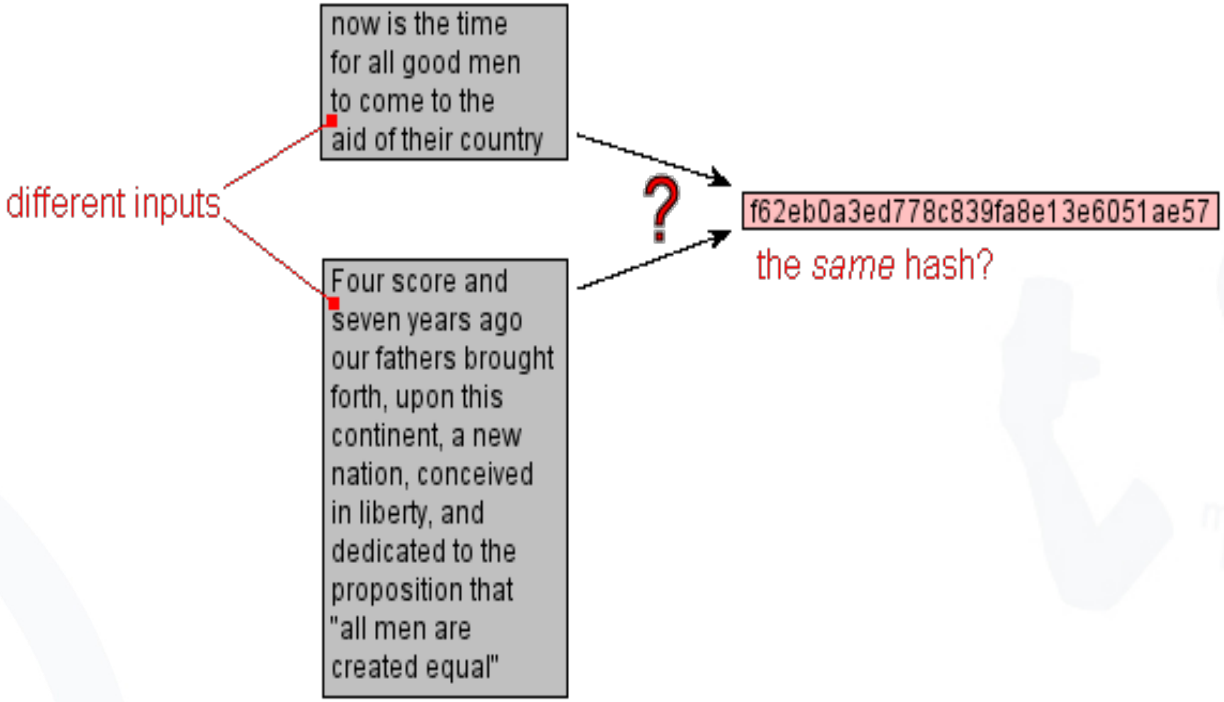
Exercise 2: Hash functions and signature systems

- The image below shows the steps of digitally signing a document. The sender receives the plain document and the digital signature.



Exercise 2: Hash functions and Signature Systems (2)

- When two different inputs produce the same hash value - collision



Exercise 2: Hash functions and Signature Systems (3)

- Given a fixed message m1, if we cannot find in a practical way a different message m2 such that `hash(m2) = hash(m1)`, then we say that this hash function is *collision-resistant*.
  a. In the digital signature scheme, why do we produce the signature on the hash of the document and not on the document directly?

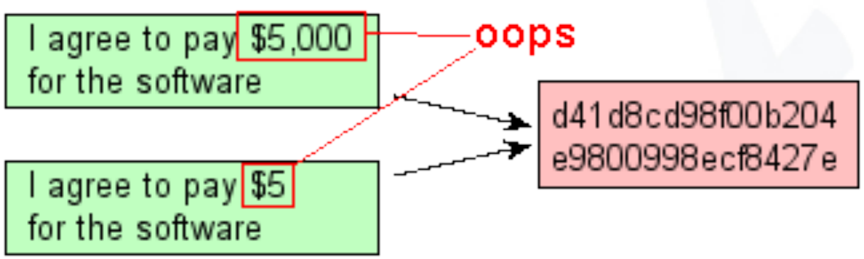
efficiency	integrity
The signature will be much shorter and thus save time since hashing is generally much faster than signing in practice.	Without the hash function, the text "to be signed" may have to be split (separated) in blocks small enough for the signature scheme to act on them directly. However, the receiver of the signed blocks is not able to recognize if all the blocks are present and in the appropriate order.

Exercise 2: Hash functions and Signature Systems (4)

- Given a fixed message `m1`, if we cannot find in a practical way a different message `m2` such that `hash(m2) = hash(m1)`, then we say that this hash function is *collision-resistant*.

b. Why is it important that hash functions are collision-resistant?

- In some digital signature systems, a party attests to a document by publishing a public key signature on a hash of the document.
  - If it is possible to produce two documents with the same hash, an attacker could get a party to attest to one, and then claim that the party had attested to the other.
- Software version comparison. An attacker who could produce two files with the same hash could trick users into believing they had the same version of a file when they in fact did not.



Exercise 3: Caesar Cipher

- Break the following ciphertext, given that the Caesar cipher was used to produce it is:

  NZIVSNCZB QA QV OMZUIVG

- (Hint: Start by a permutation of the alphabet by 1, then 2, … until the result makes sense in English)

Caesar Cipher

Ciphertext: **NZIVSNCZB QA QV OMZUIVG**

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

- For $k \in \{0..25\}$ we have:
  - An encryption function:
    - e: x -> (x+k) mod 26
  - A decryption function:
    - d: x -> (x-k) mod 26
  - In this case $k_e = k_d$

mobile business

Caesar Cipher

- Let‘s try:

Key	N	Z	I	V	S	N	C	Z	B		Q	A
1	M	Y	H	U	R	M	B	Y	A		P	Z
2	L	X	G	T	Q	L	A	X	Z		O	Y
3	K	W	F	S	P	K	Z	W	Y		N	X
4	J	V	E	R	O	J	Y	V	X		M	W
5	I	U	D	Q	N	I	X	U	W		L	V
6	H	T	C	P	M	H	W	T	V		K	U
7	G	S	B	O	L	G	V	S	U		J	T
8	**F**	**R**	**A**	**N**	**K**	**F**	**U**	**R**	**T**		**I**	**S**

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

The plaintext

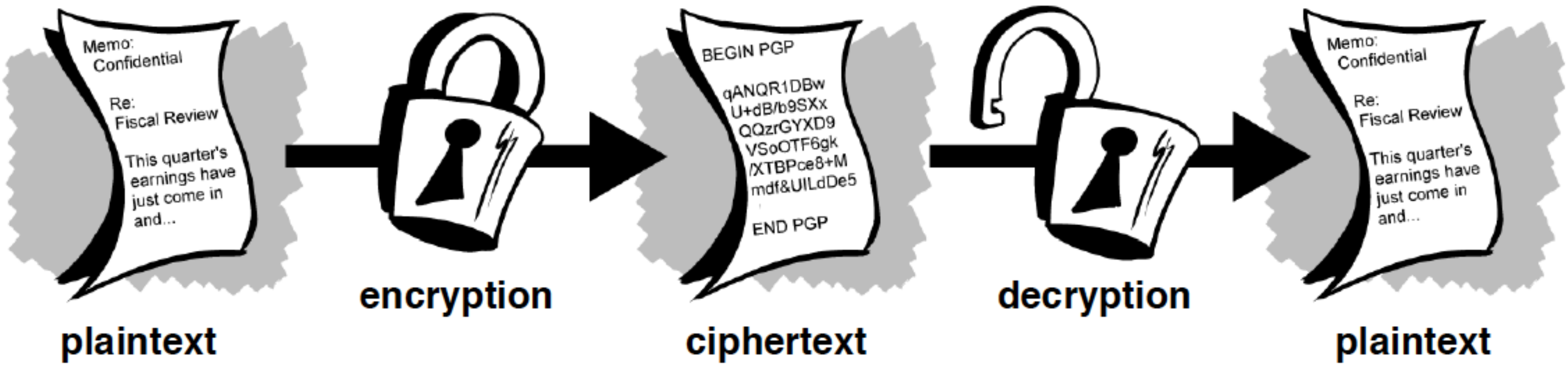
- The key is 8
- The plain text is:

FRANKFURT IS IN GERMANY

Assessment of Caesar Cipher

- 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.

Encryption - Decryption

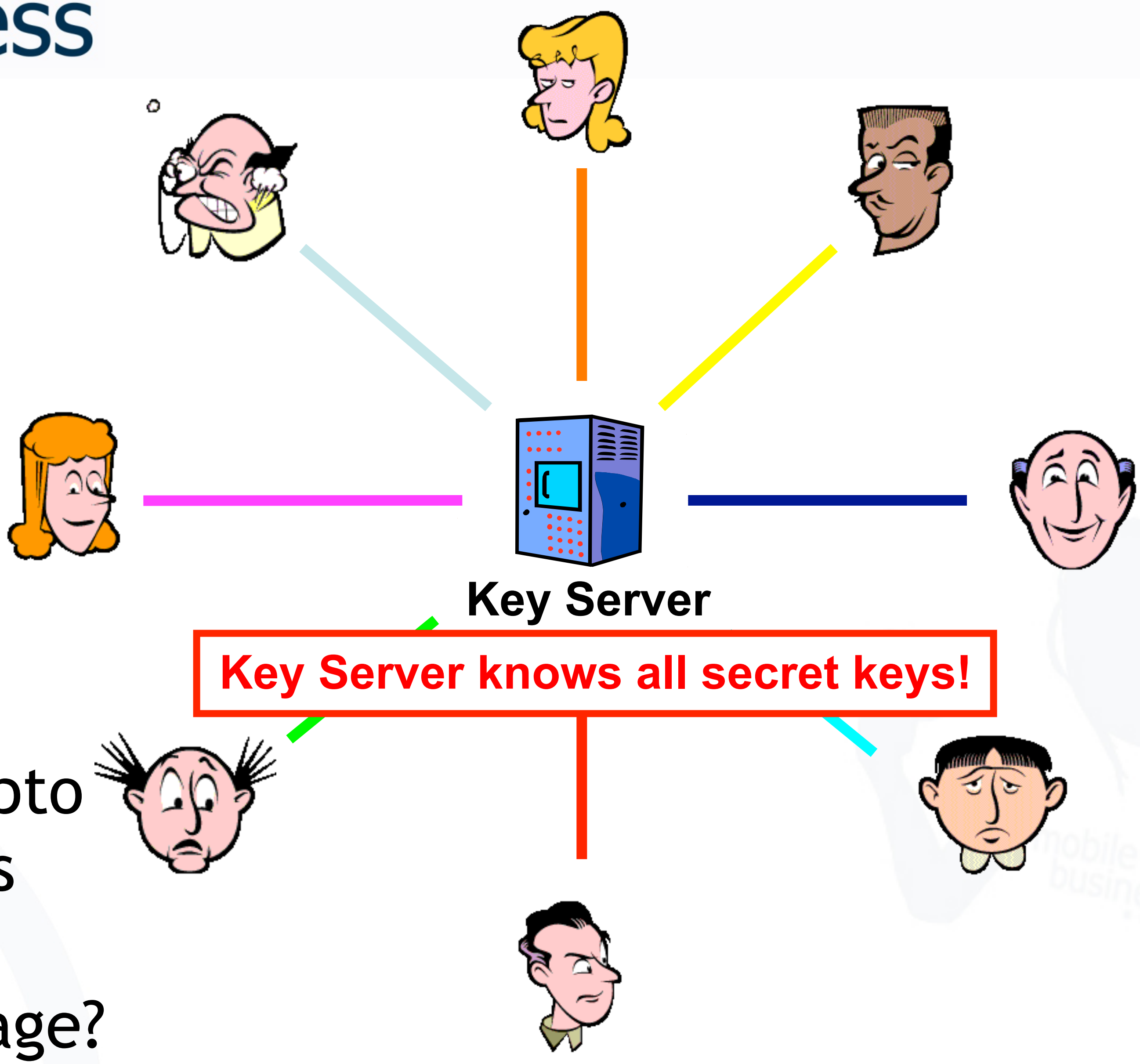


http://www.pgpi.org/doc/guide/6.5/en/intro/

Exercise 4: Misc

a. What is the difference between symmetric and asymmetric crypto systems?

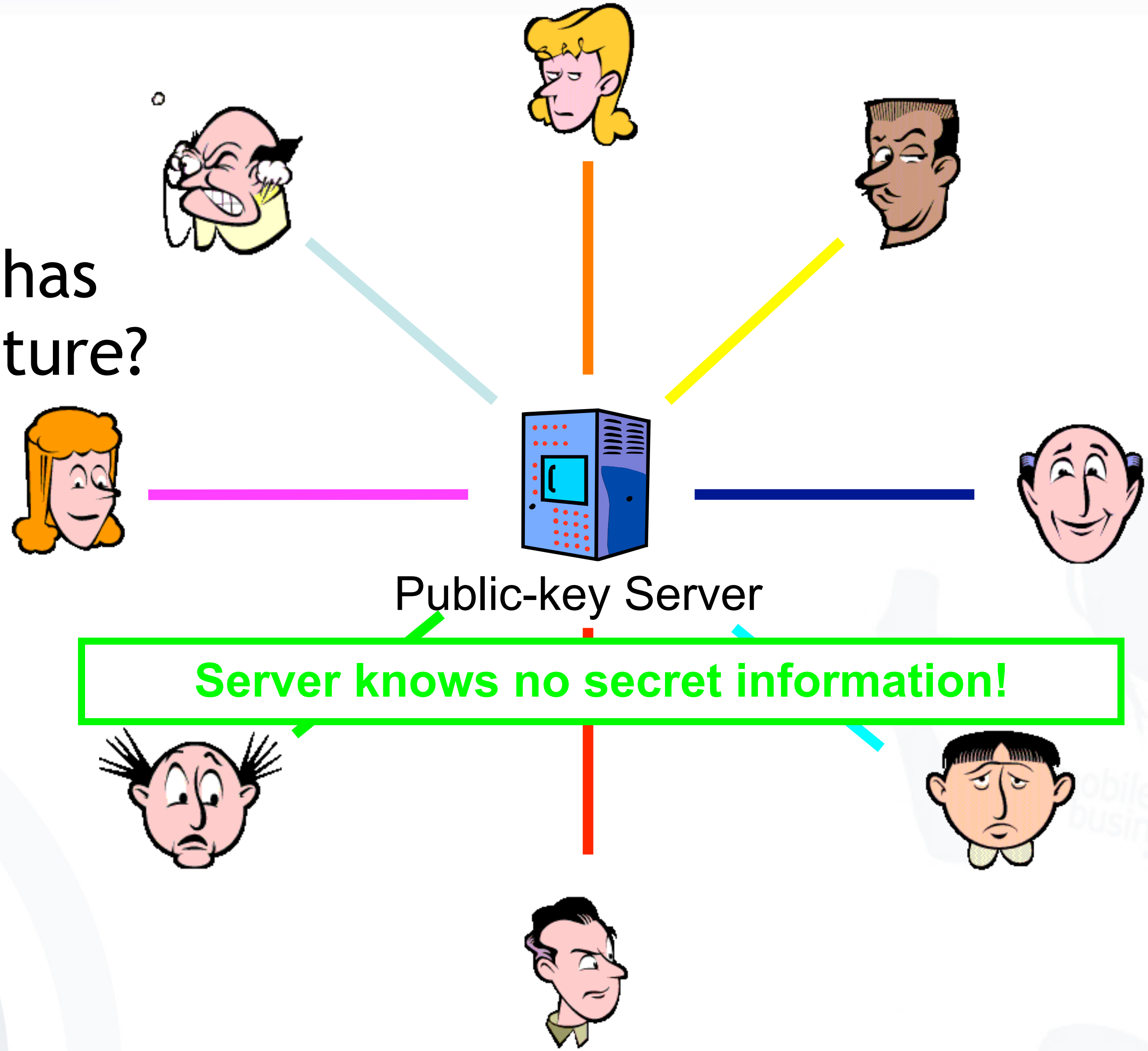
Exercise 4: Misc (a4)



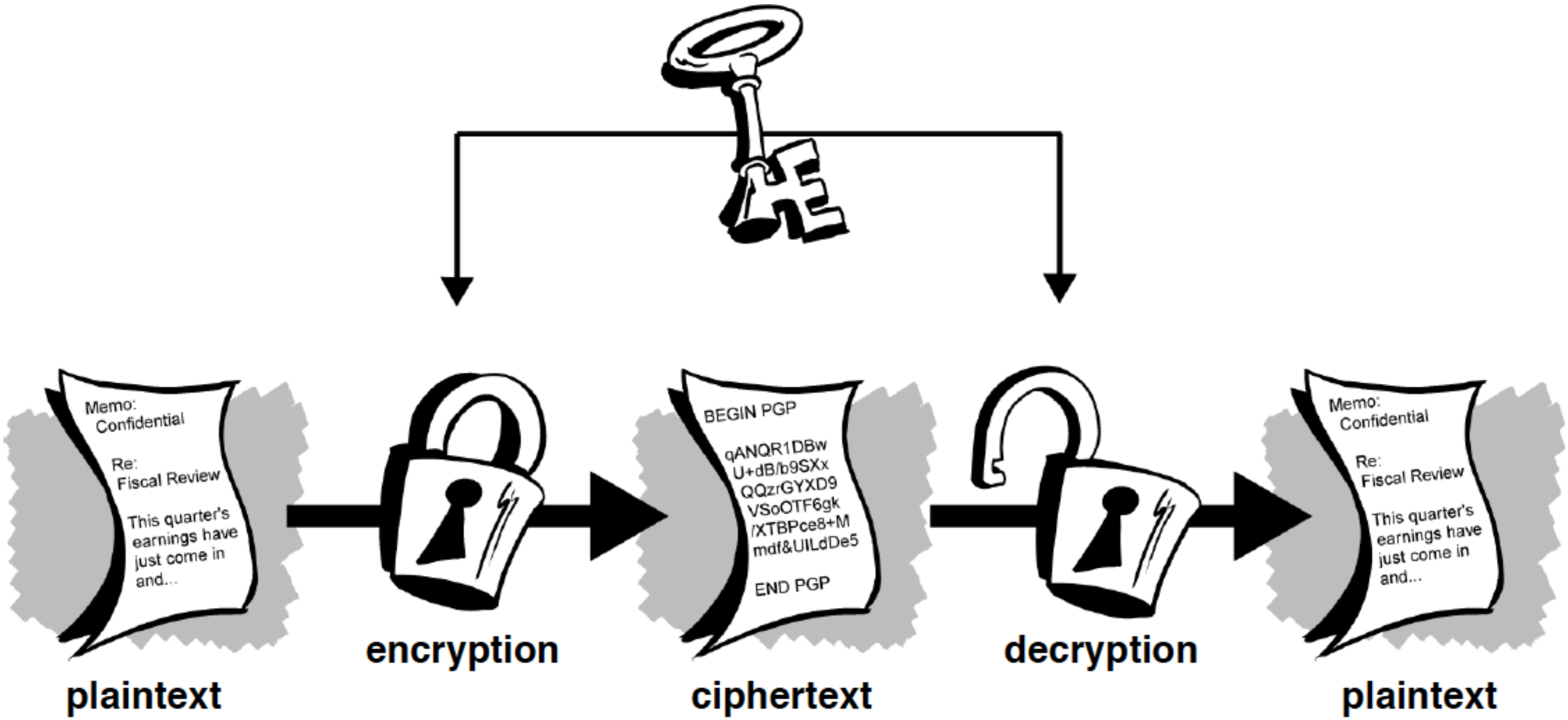
Which crypto system has this disadvantage?

Exercise 4: Misc (a3)

Which crypto system has this feature?

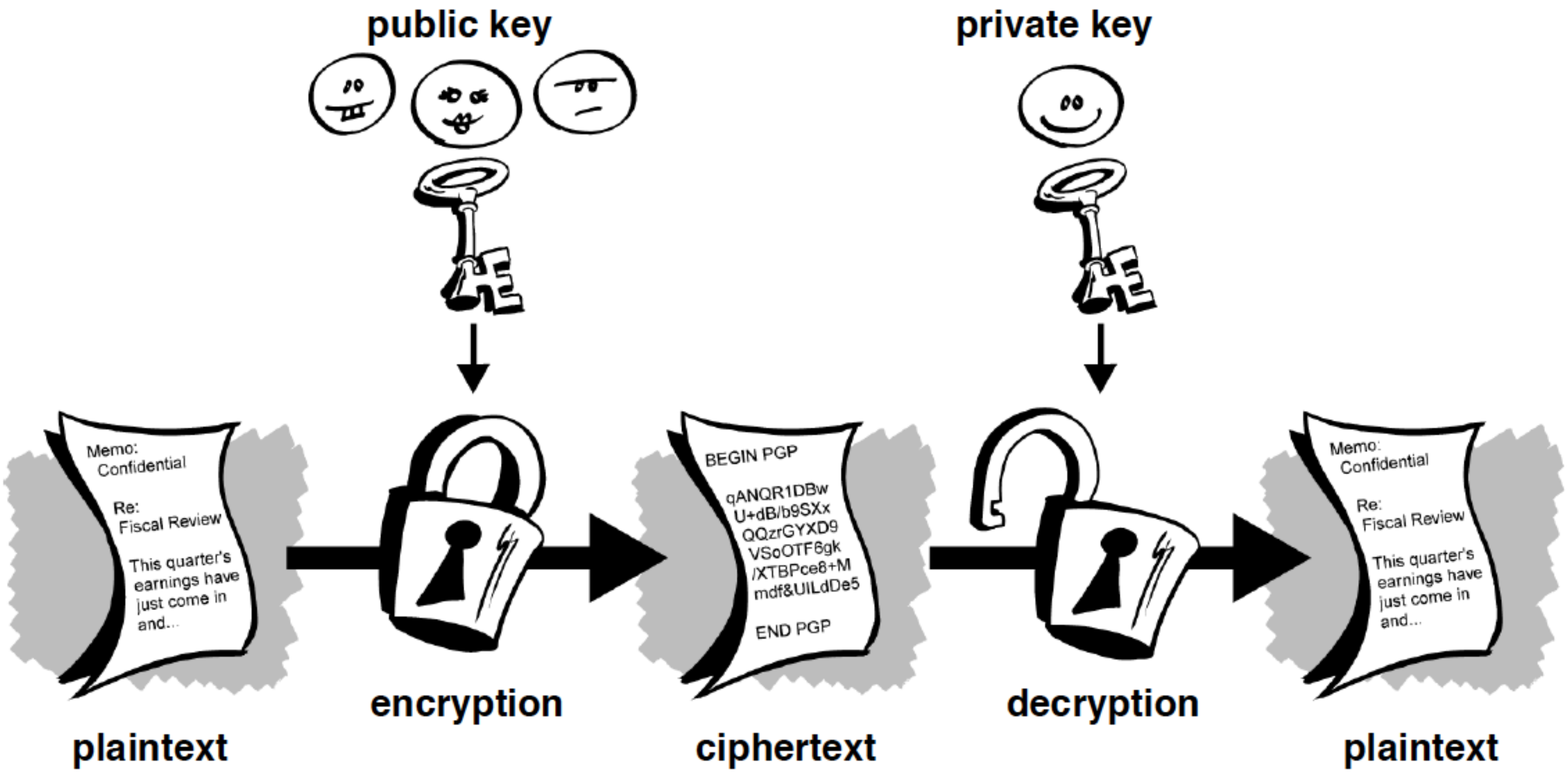


Guess which crypto system this is



Symmetric or Asymmetric?

This crypto system is...?



Symmetric or Asymmetric?

Symmetric Encryption

Advantage: Algorithms are very fast

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

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

[J. Buchmann: Lecture Public Key Infrastrukturen, FG Theoretische Informatik, TU-Darmstadt]

Performance of Public Key Algorithms

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)**

[J. Buchmann: Lecture Public Key Infrastrukturen, FG Theoretische Informatik, TU-Darmstadt]

Summary 4: Misc (a)

a. Differences between symmetric and asymmetric cryptosystems.

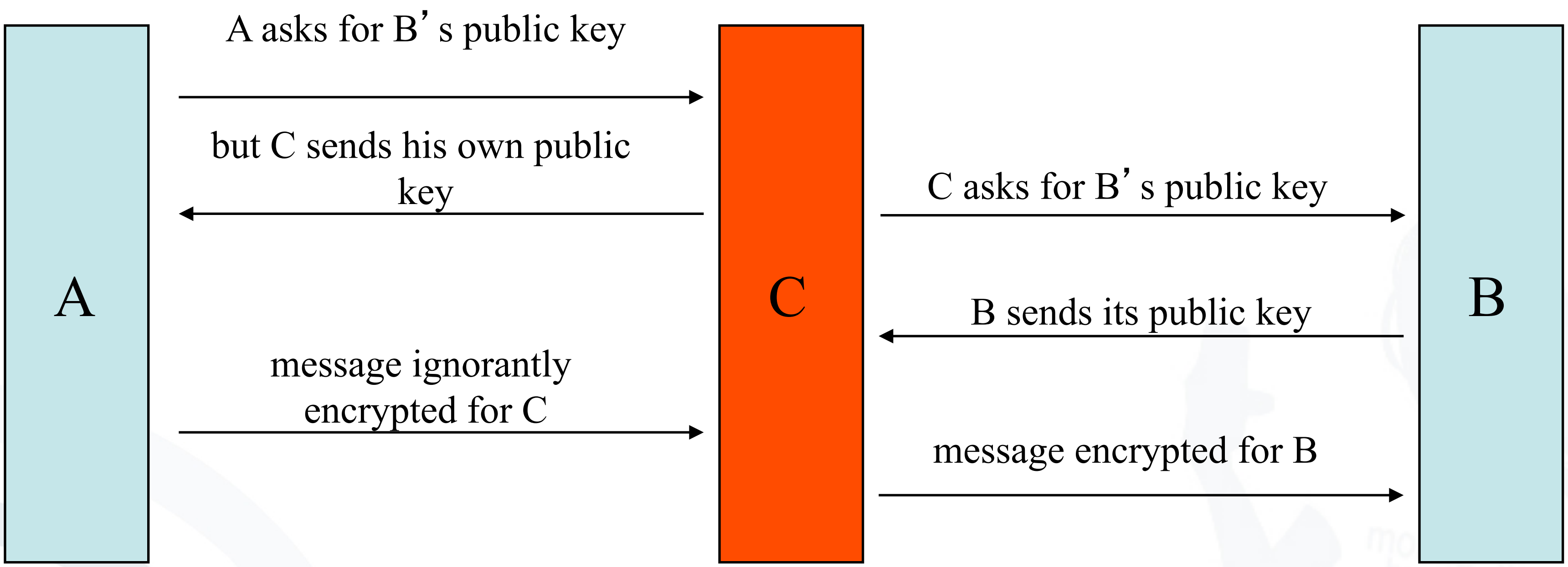
Symmetric	Asymmetric
Both encryption and decryption is done with the same key.	Encryption with public key, decryption with private key.
One key per communication pair is necessary.	Does not require a secure communication channel. Public key can be freely distributed.
Efficient in terms of performance	Less efficient
Keys have to be kept secret	Only keep own private key secret
Secure agreement and transfer are necessary.	Does not require agreement on a shared key.
A center for key distribution is possible but this party then knows all secret keys!	A center for key distribution is possible and this party does not know the secret keys.

Exercise 4: Misc (b)

b. Why is certification of public key necessary? Name an attack that is possible if keys are not certified.

Exercise 4: Misc (b1)

What is the name of this attack?



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

Exercise 4: Misc

What are advantages and disadvantages of asymmetric crypto systems?

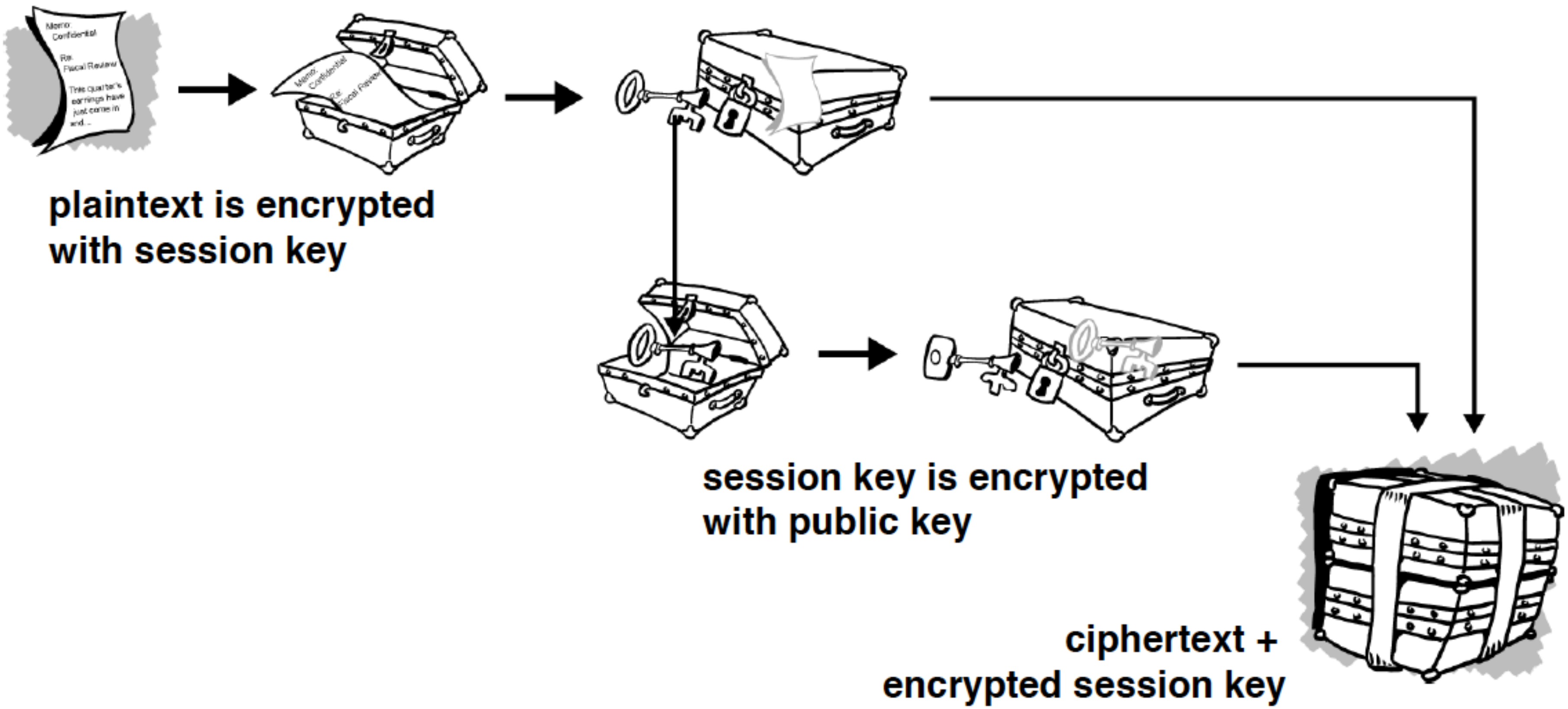
Advantages:

- No secret must be shared
- Only one key per endpoint

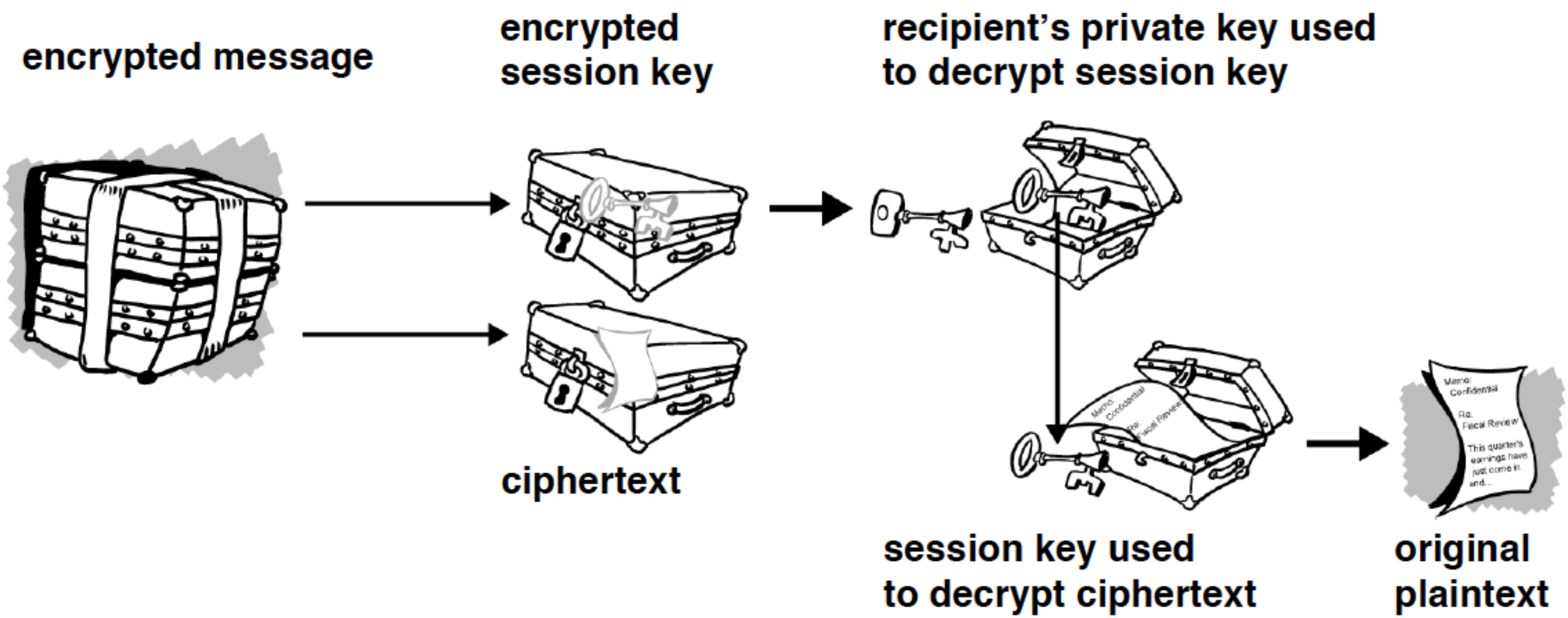
Disadvantages:

- Algorithms are very slow
- Man-in-the-middle-attack

Hybrid encryption: PGP Example Encryption



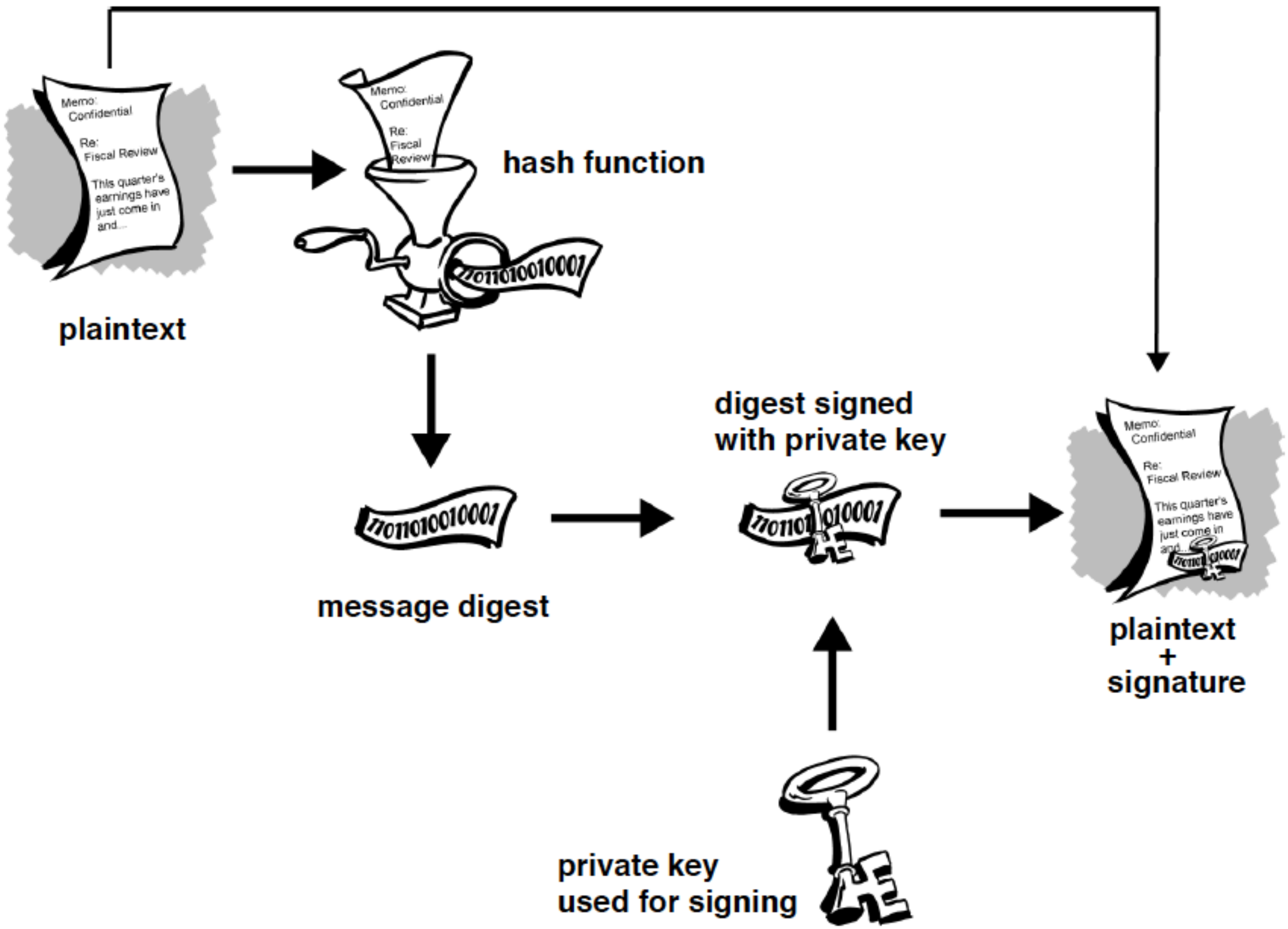
PGP Decryption



PGP example continued…

- Encryption offers
  - Confidentiality
- Digital Signatures offer
  - Authentication
  - Integrity

Digital Signatures



Exercise 5: Stream ciphers

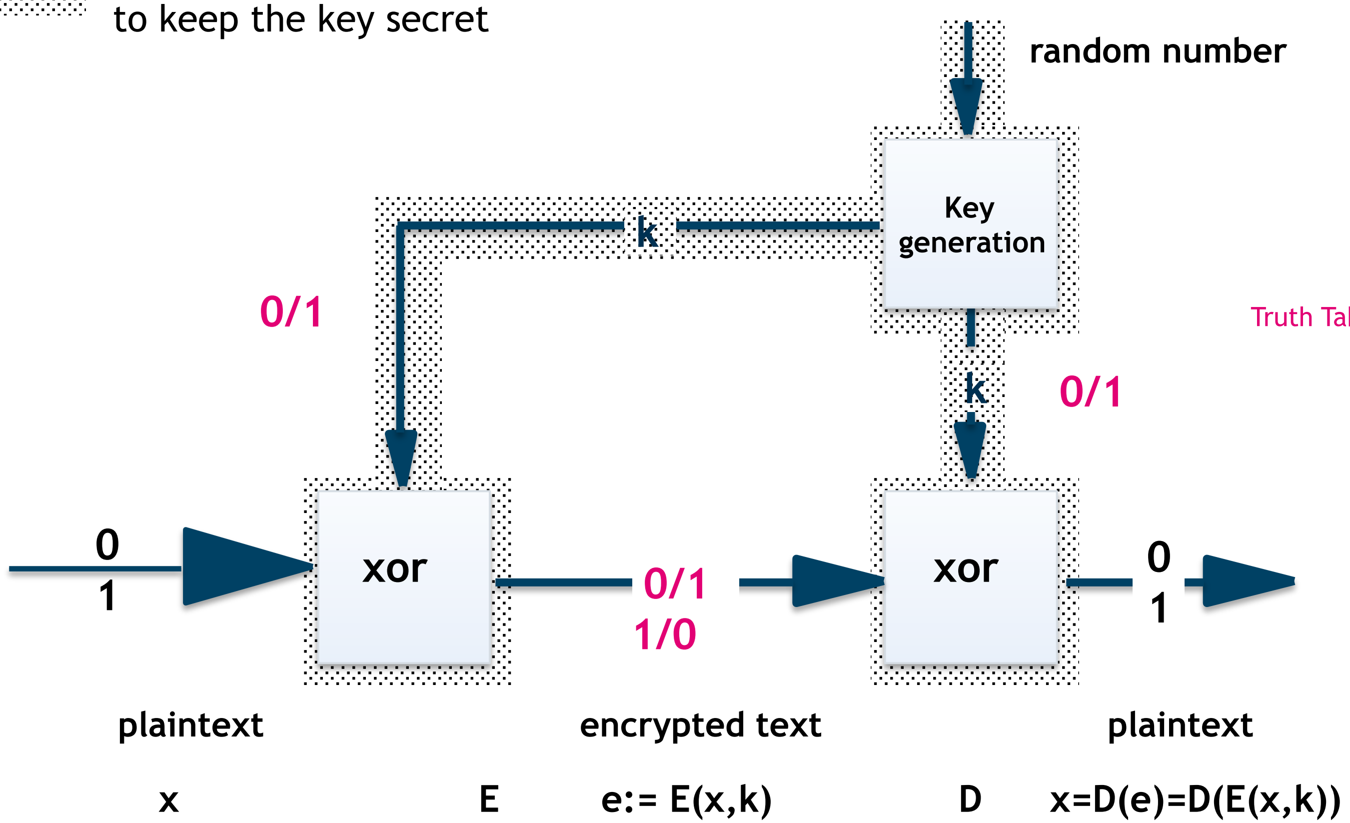
a) What is a one-time pad (Vernam-code)?

One Time Pad

- Invented by Gilbert Vernam
- The length of the key is as long as the length of the plaintext.
- The key is randomly chosen and only used once.
- Every key has the same probability.

Example One Time Pad

area that needs to be protected to keep the key secret



X_i	S_i	Y_i
0	0	0
0	1	1
1	0	1
1	1	0

Truth Table of the XOR operation

[based on Federrath and Pfitzmann 1997]

Exercise 5: Stream ciphers

- b) Alice wants to encrypt the letter A, where the letter is given in ASCII code. The ASCII value for A is $65_{10} = 1000001_2$. Using Vernam-code, which of the following keys are suitable to encrypt this plaintext:
  - b1) 10100110
  - b2) 0011111
  - b3) 101010

X_i	S_i	Y_i
0	0	0
0	1	1
1	0	1
1	1	0

Truth Table of the XOR operation

Exercise 5: Stream ciphers

- c) Encrypt the message using Vernam code and using XOR as an encryption function and the key in b).

Plaintext (A)	1000001
Key (B)	0011111
Ciphertext (A xor B)	1011110

X_i	S_i	Y_i
0	0	0
0	1	1
1	0	1
1	1	0

Truth Table of the XOR operation

Thank you!

- Questions: sec@m-chair.de

References

- [Federrath Pfitzmann 1997] Hannes Federrath, Andreas Pfitzmann: Bausteine zur Realisierung mehrseitiger Sicherheit. in: Günter Müller, Andreas Pfitzmann (Hrsg.): Mehrseitige Sicherheit in der Kommunikationstechnik, Addison-Wesley-

  Longman1997, 83-104.