Cryptology

An introduction

by
Ulf Lindqvist
translated and processed by Erland Jonsson

Contents

- Introduction
- Terminology
- History
- Symmetrical systems *(secret-key)*
- Asymmetrical systems *(public-key)*
- Key management
- Problems with cryptology
- The Future

What is Cryptography?

- Methods for the protection of information and communications

BQQ EBJIFDQ JBAABIÅKAB

CRR FCKJGER KCBBCJÄLBC

DSS GDLKHFS LDCCDKÖMCD

ETT HEMLIGT MEDDELANDE

(English: a secret message)

Who needs cryptography?

- Earlier: only diplomats and the military
- Now: ***everybody***!
- Reasons:
  - Electronic communication over insecure channels, e.g. the Internet
  - Traditional methods for authentication (signature, voice) does not work for data communications

Terminology (Swedish - English)

- Kryptologi, kryptoteknik
- Kryptografi
  - Kryptering
  - Dekryptering
  - Klartext
  - Kryptotext, kryptogram
  - Nyckel
- Forcering
  - Forcör
  - Forcera

- Cryptology
- Cryptography
  - Encryption
  - Decryption
  - Plaintext, Cleartext
  - Ciphertext, cryptogram
  - Key
- Cryptanalysis
  - Cryptanalyst
  - Break

History

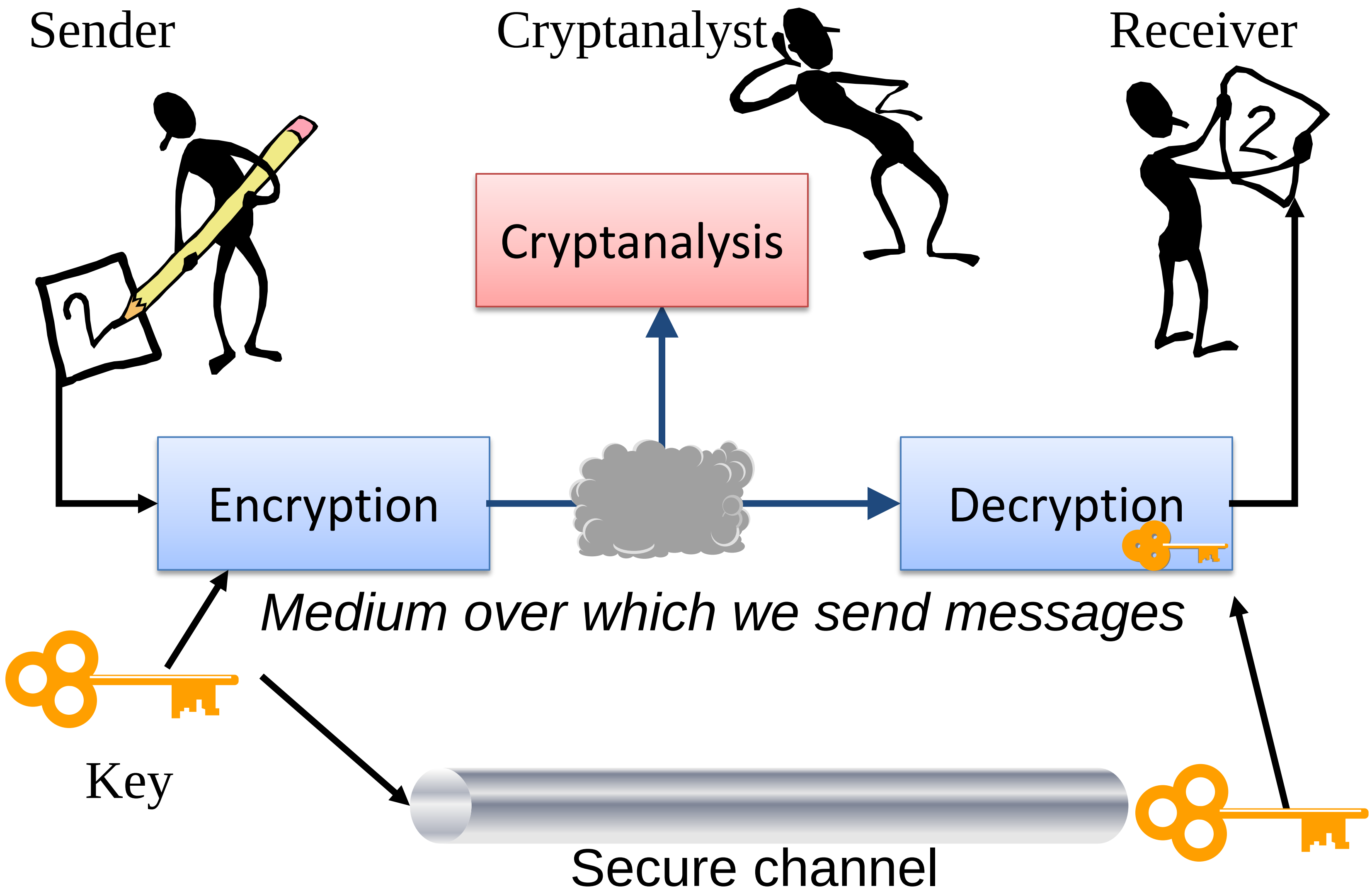
- The pre-scientific era
  - ca 400 BC — Sparta — (transposition)
  - ca 50 BC — Julius Caesar — (substitution)
  - 19th century — A Kerckhoff — (only the key secret)
  - 1920-ies — G S Vernam — (K as long as M)[1]
  - 1939-45 — 2nd Word War
- Scientific era
  - 1949 — C E Shannon
  - 1976 — W Diffie & M E Hellman

1. One-time pad

History
Secret versus open research

- Cryptology used to be available only for government authorities (military, diplomats)
- Comprehensive open research started in the mid 70-ies
- The secret research is probably considerably bigger than open research
- ”NSA is believed to be 10-15 years ahead of the open research”

Traditional *symmetrical* model with secret key



Symmetrical systems

Theoretical versus practical security

- ***Theoretical*** security
  - No problems with key management
  - The cryptanalyst has unlimited resources
- ***Practical*** security
  - Many limitations
  - The cryptanalyst also has limitations
  - The security is a function of the estimated possibilities of the cryptanalyst

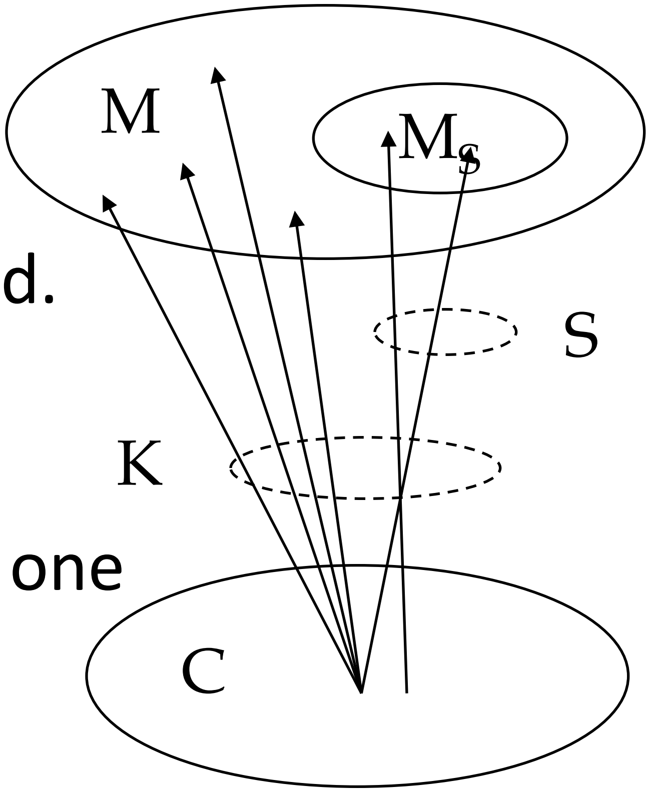
Symmetrical systems
Cryptanalysis

Assumption:
the cryptanalyst knows everything about the system except the key.

- Different types of cryptanalysis attacks:
  - Ciphertext-only
    - The cryptanalyst only has encrypted text
  - Known-plaintext
    - The cryptanalyst has some plaintext-ciphertext pairs
  - Chosen-plaintext/Chosen-ciphertext
    - The cryptanalyst can make tests with selected plaintext and get the corresponding encrypted text

Symmetrical systems
Cryptanalysis cont'd

- Theoretical cryptanalysis
  - All possible keys are tested (unlimited time)
    and the cipher text is decrypted.
  - Successful cryptanalysis is possible if this leads to exactly one interpretable message.



Symmetrical systems

Cryptanalysis cont'd

Ciphertext: `DJFKRIOPEPWPEPS`

Key 1: `ajfkjrisdfkjsdd`

Key 2: `jfjreieifkdjdjf`

PT 1: **`attack at eight`**

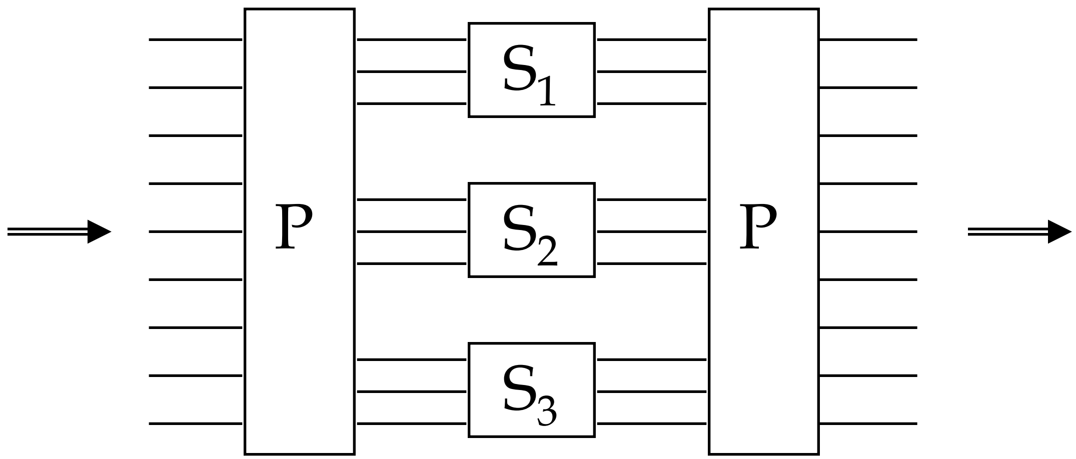
PT 2: **`no attack today`**

Symmetrical systems
Diffusion — Confusion

- Diffusion
  - Changing one symbol in the plaintext affects many symbols in the ciphertext
  - Changing one symbol in the key affects many symbols in the ciphertext
- Confusion
  - The ciphertext must depend on the plaintext and the key in a complicated way so that the derivation of statistical relations is hard to do

Symmetrical systems
Product ciphers

- Product ciphers gives good diffusion and confusion



P is a permutation, possibly the same in every step.
S_i are general, different, non-linear substitutions.

Symmetrical systems

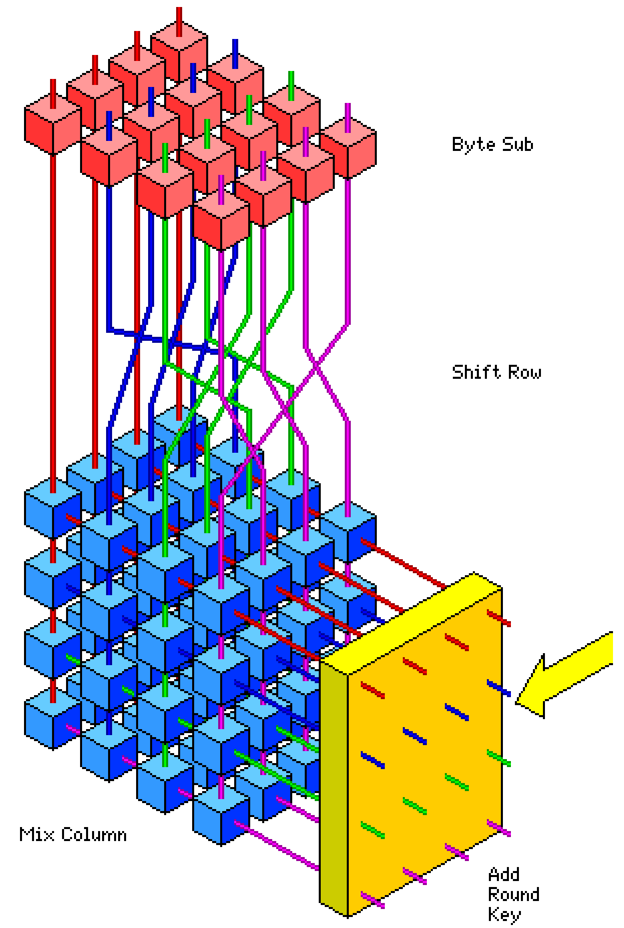
Example of system: DES

- Data Encryption Standard
- Encrypts blocks of 64 bits, 56-bit key
- Developed by IBM and NSA, American standard since 1977
- NSAs involvement is discussed
  - Suspicion of back-doors has not been confirmed
  - Rather, NSA seems to have strengthened the algorithm
- Status: Algorithm is considered very strong, but the key is too short
- Development: Triple DES (3DES), 112 bits
- Since 2000: Advanced Encryption Standard (AES)

Symmetrical systems

Example of system: AES

- **A**dvanced **E**ncryption **S**tandard
  - International competition (NIST)
  - Call for candidates (1997)
  - Winners from Belgium (Rijndael)
  - Standard made official 2001
- Based on *Permutations* & *Substitutions*
  - Not completely symmetric (encryption versus decryption)
  - 128 bit data blocks
  - Supports 128, 192 and 256 bit key length (with 10, 12, 14 rounds, respectively)
  - Has become the “standard”!
    - 10G Ethernet, ZigBee, SSL, IPSec, WiMAX …



Symmetrical systems
Key lengths

- Based on *computational complexity*
- Comparison for *exhaustive search*:

Key length	Asymm. RSA	Possible keys	Time a	Time b
40 bits		$2^{40} \approx 1.1 \cdot 10^{12}$	1 week	(9 s)
56 bits		$2^{56} \approx 7.2 \cdot 10^{16}$	1200 yrs	1 week
100 bits	2048 bits	$2^{100} \approx 1.3 \cdot 10^{30}$	$2 \cdot 10^{16}$ yrs	$3 \cdot 10^{11}$ yrs
128 bits	3072 bits	$2^{128} \approx 3.4 \cdot 10^{38}$	$6 \cdot 10^{24}$ yrs	$9 \cdot 10^{19}$ yrs

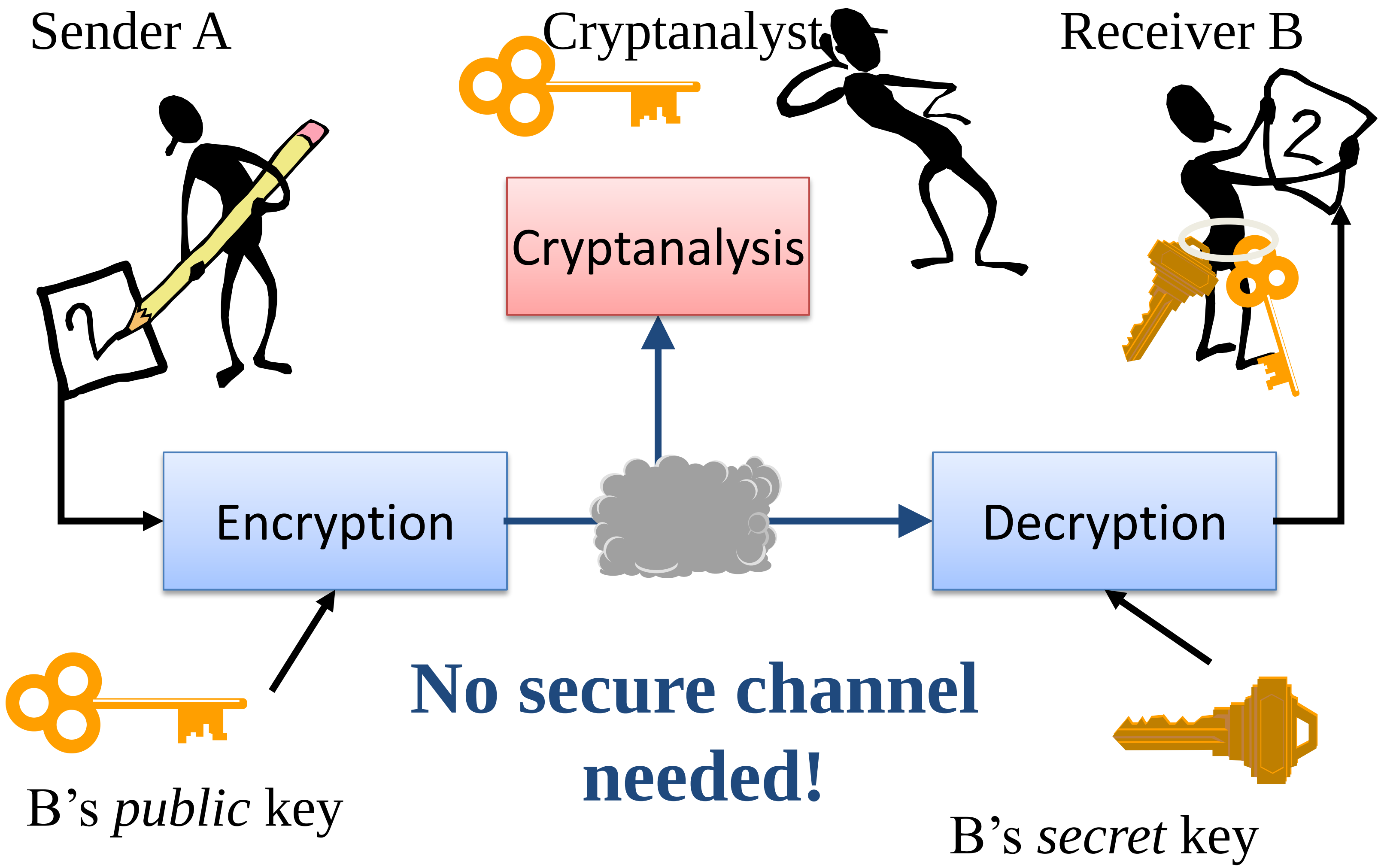
(100 million combinations correspond to 27 bits)

See Table 2.1 and 21.2

Asymmetrical systems

- Principle was discovered in 1976
- Two keys,
  - one public key, k_{PUB} for encryption and
  - one secret key k_{PRIV} for decryption (thus: public-key-systems)
- The public key, k_{PUB}, distributed openly to everybody ***(as a matter of fact: this is the basic idea)***
- Slow encryption, primarily used for signing and distribution of symmetrical keys

Asymmetrical systems Model



Asymmetrical systems
Basic Principles

- One-way functions
  - y = f(x) easy to calculate, but finding x for a given y, so that y = f(x), i.e. $x = f^{-1}(y)$ is very hard (impossible in practice)
  - Example: Multiplication of two 100-character primes is easy, but finding the two primes for a given product is hard
- Trap-door one-way functions
  - $y = f_P(x)$ easy, but $x = f_P^{-1}(y)$ hard to calculate, unless you know the parameter P

Asymmetrical systems Characteristics

- No secure channel needed
- Communication between n units, requires 2n keys (cp symmetrical system: about ~n^2)
- Authentication important
  - If a false public key is used, the intruder can read the message
  - The sender is unknown to me
    (since the public key is available to everybody)
    - Solution: signatures

Asymmetrical systems

Signatures

- In principle the system goes "backwards"
  - Encryption with k_{PRIV}
  - everybody can decrypt with k_{PUB}

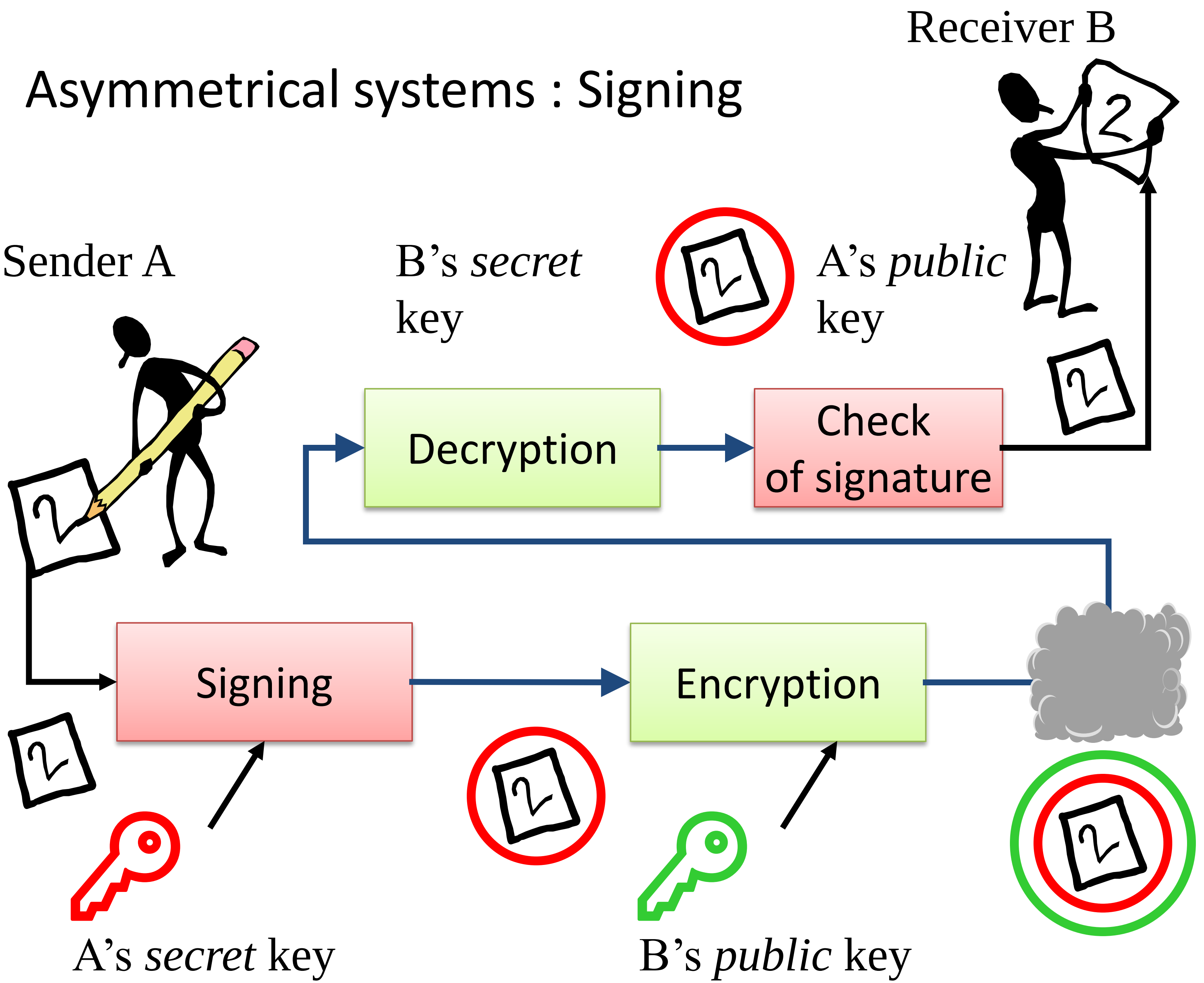
Note! No confidentiality!

- Signing *and* encryption:

Send A: calculates $E(k_{PUB\text{-}B}, E(k_{PRIV\text{-}A}, M)) = C$

Rec B: calculates $D(k_{PUB\text{-}A}, D(k_{PRIV\text{-}B}, C)) = M$

Asymmetrical systems : Signing



Asymmetrical systems
Example of system: Diffie-Hellman

- A way to distribute the key to a symmetrical system without a secure channel
- One-way function: $f(x) = a^x$ (modulo p)
- Basic principle:
  A and B choose a number b (not secret)

A: Chooses x_1 and calculates $y_1 = b^{x_1}$
A sends y_1 to B.
A calculates $y_2^{x_1} = \underline{b^{x_1 x_2}}$

B: Chooses x_2 and calculates $y_2 = b^{x_2}$
B sends y_2 to A.
B calculates $y_1^{x_2} = \underline{b^{x_1 x_2}}$

Asymmetrical systems

Example of system: RSA

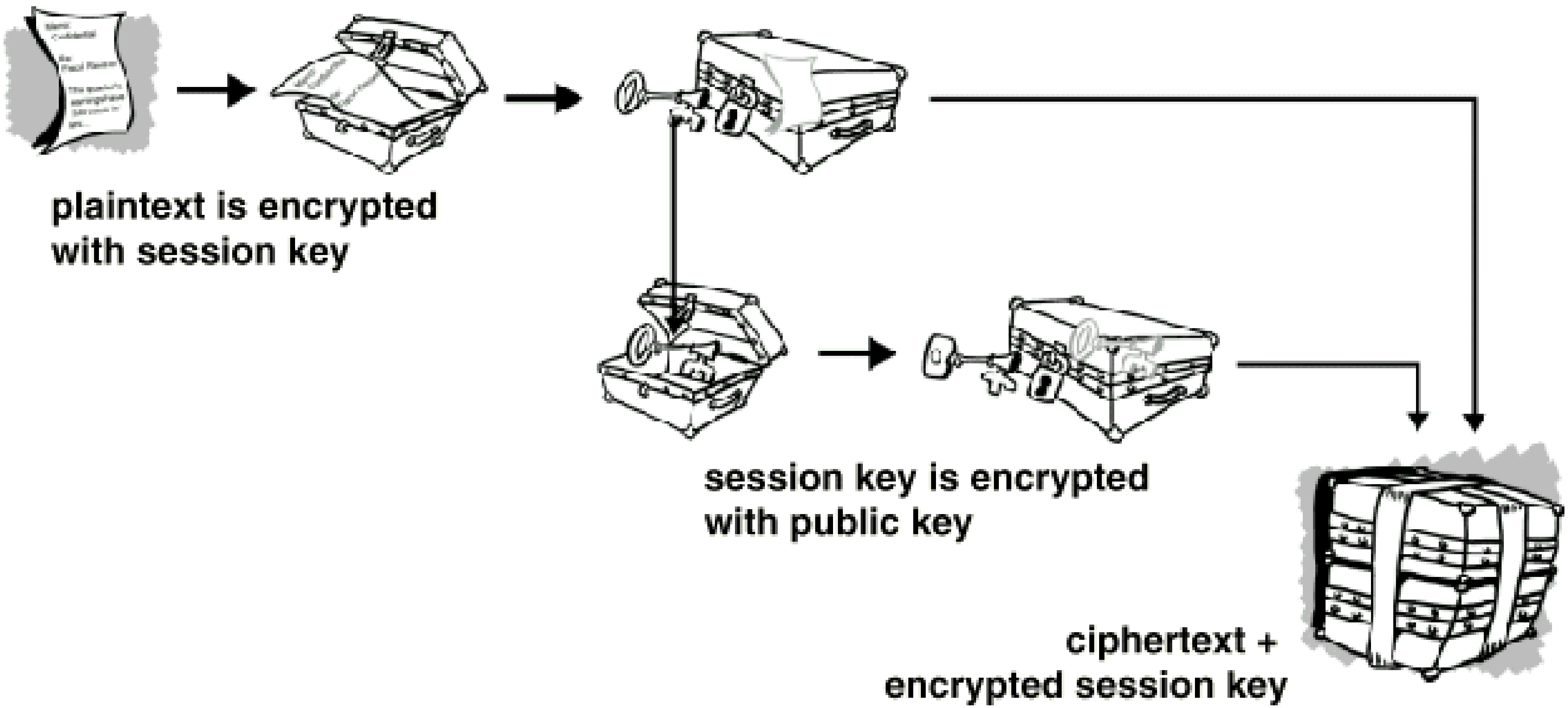
- RSA (Rivest, Shamir, Adleman) 1978
  - Uses a trap-door function that is based on the multiplication of big primes
  - Security depends on the development of
    1) Algorithms for factoring
    2) Computers with high computational power
  - Most well-known and most commonly-used asymmetrical system

Hybrid systems

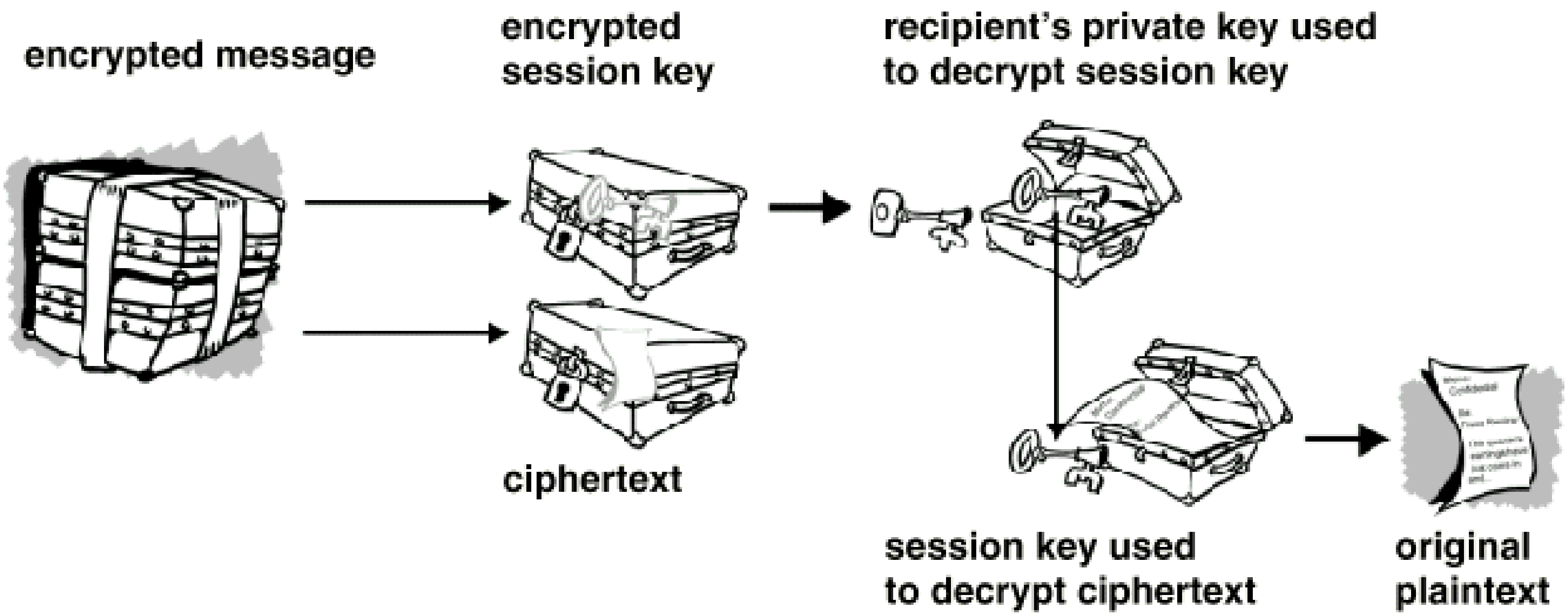
Example of system: PGP

- PGP (Pretty Good Privacy) GnuPG
  - Application for encryption and signing of emails
  - Uses RSA and IDEA (originally)
  - Large-scale test of the management of asymmetrical keys (signing, publication etc)
  - Disadvantages:
    - Certain lack of compatibility (different versions)
    - Hard to protect the secret key

PGP Encryption



PGP Decryption



Key Management

- Protocols
  - Combinations of symmetrical and asymmetrical systems are used for key distribution
  - Hard to design good protocols
- Key escrow
  - Users are forced to deposit “main keys” so that authorities (police, customs etc) can interpret secret communications
  - Fiercely debated in the USA, to some extent also within the EU
  - Organisationally questionnable

Problems with cryptography

- Based on computational complexity, which is theoretically hard. Therefore it is easy to find a bad system that looks good. It is considerably harder to show that it really *is bad*.
- Requires profound knowledge to make good designs.
- Electronic commerce is based on the difficulty to factorize big numbers…
- Non-cryptological issues is the real problem!

Problems with Cryptography
Block Cipher Modes

- In Electronic Codebook (ECB) mode, a message is split into blocks and each is encrypted separately.

(from Wikipedia)

The Future

- The Advanced Encryption Standard (AES) is replacing DES.
- New methods required to build robust applications
- The need for commercial IT-services is a driving force, e.g. Internet B2B and B2C applications
- Political decisions slow down progress(?)
- Quantum Cryptography