Cryptography - A Crash Overview

Stanisław Radziszowski, Christopher A. Wood
Rochester Institute of Technology
{spr, caw4567}@cs.rit.edu

March 8, 2013

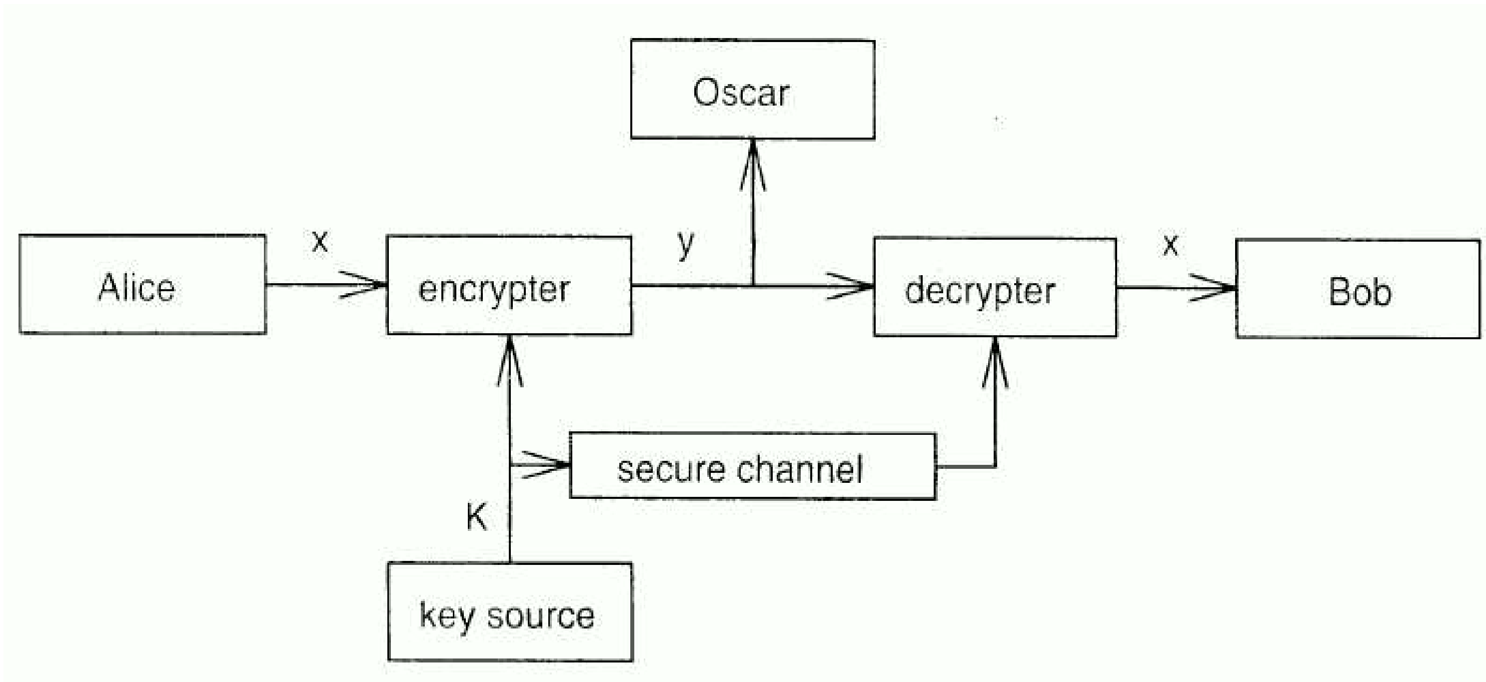
Cryptography
goals

Desired security properties in the digital world:

- confidentiality, secrecy
- data integrity
- authentication, of data origin and entity
- non-repudiation

Cryptography

basic scenario



[Stinson]

A Real-World Scenario

email security with PGP

- Alice wants to send Bob a secret email message
- The data must be encrypted by Alice and then decrypted by Bob.

How do we proceed?

R · I · T
Department of Computer Science

Cryptographic Ciphers

at the heart of everything

A *cipher* is an algorithm that converts plaintext into something that cannot be read by uninitiated persons and later allows retrieval of the plaintext.

Shared-key block ciphers (a form of cryptographic primitive) are ciphers which use the same key for encryption and decryption.

Implication: The sender and receiver must have a way of obtaining the same key (more on this later).

Shared-Key Primitives and Algorithms

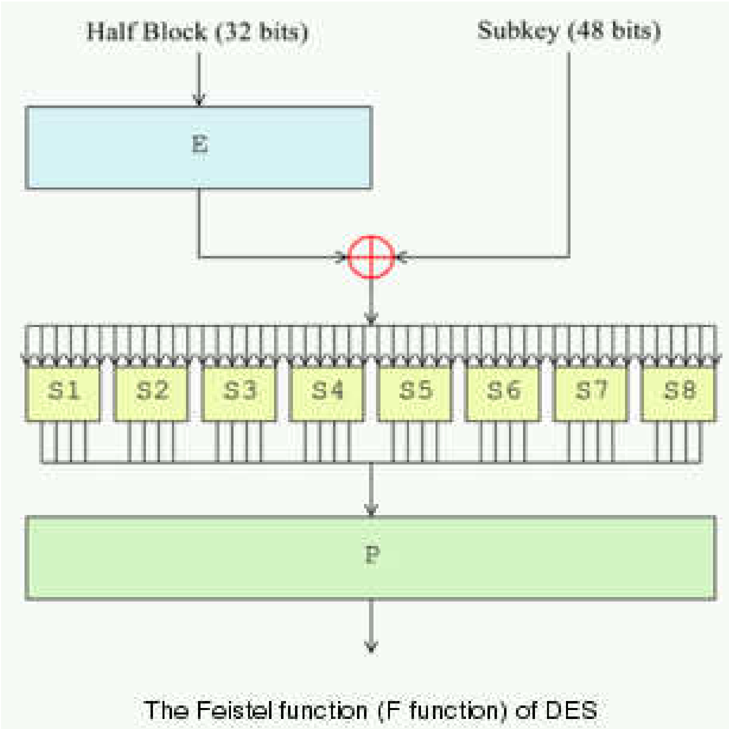
Some history of conventional cryptography

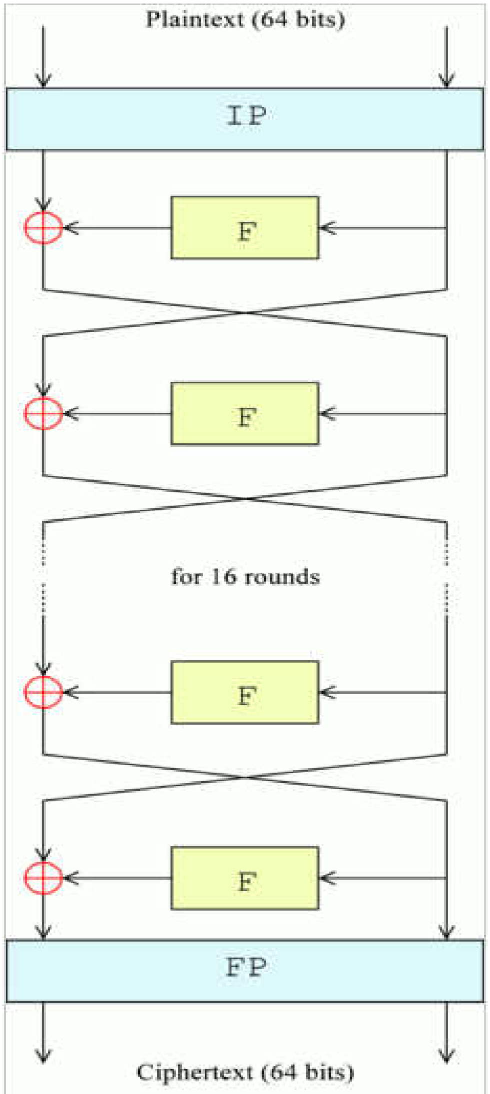
Block ciphers and other shared-key primitives and algorithms are far from new... but they are constantly changing.

- block ciphers since the 1970s
  IBM's Lucifer,
  DES - Data Encryption Standard, 3DES
  IDEA - International Data Encryption Algorithm
  AES - Advanced Encryption Standard
- stream ciphers, RC4 - also can come from counter mode of block ciphers or hash functions
- MAC, HMAC - message authentication codes
- PRNG - pseudo-random number generators

Data Encryption Standard (1977-1998-...?)

Feistel cipher

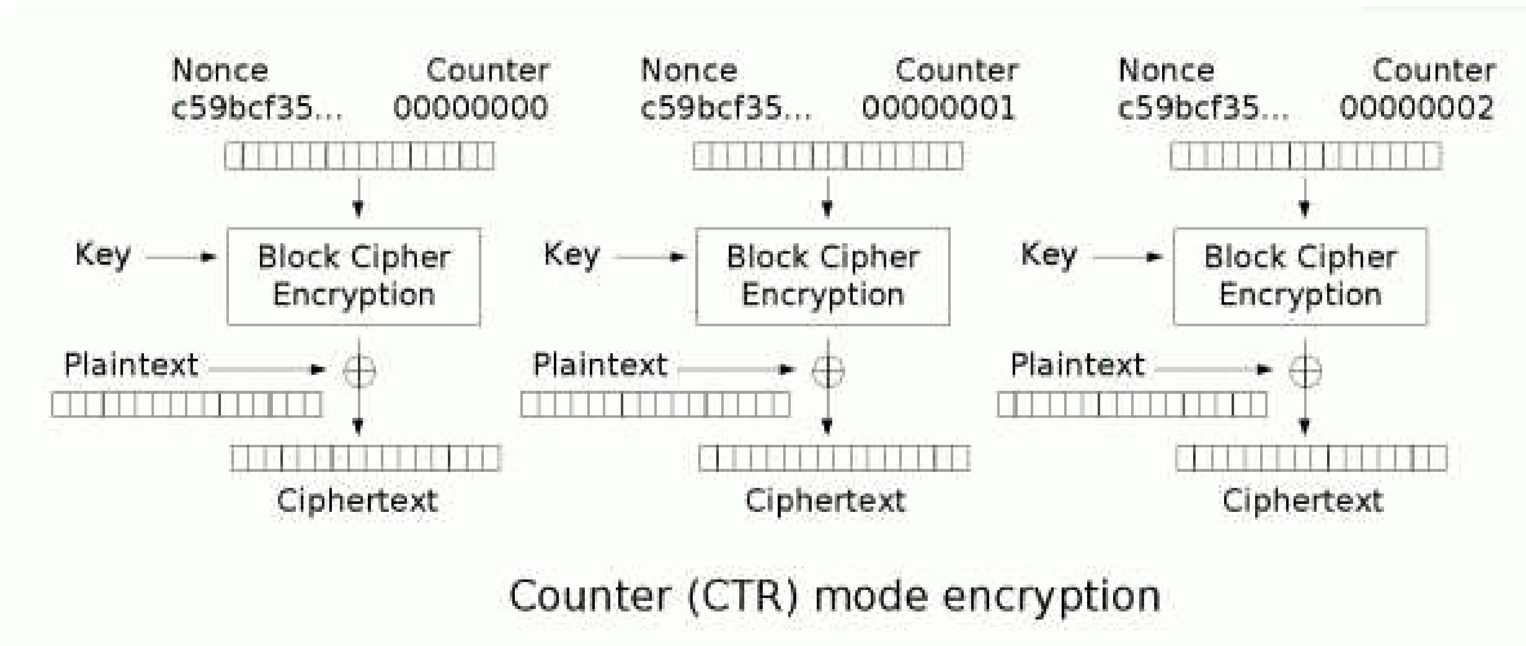




[Wikipedia]

Block Cipher in Use

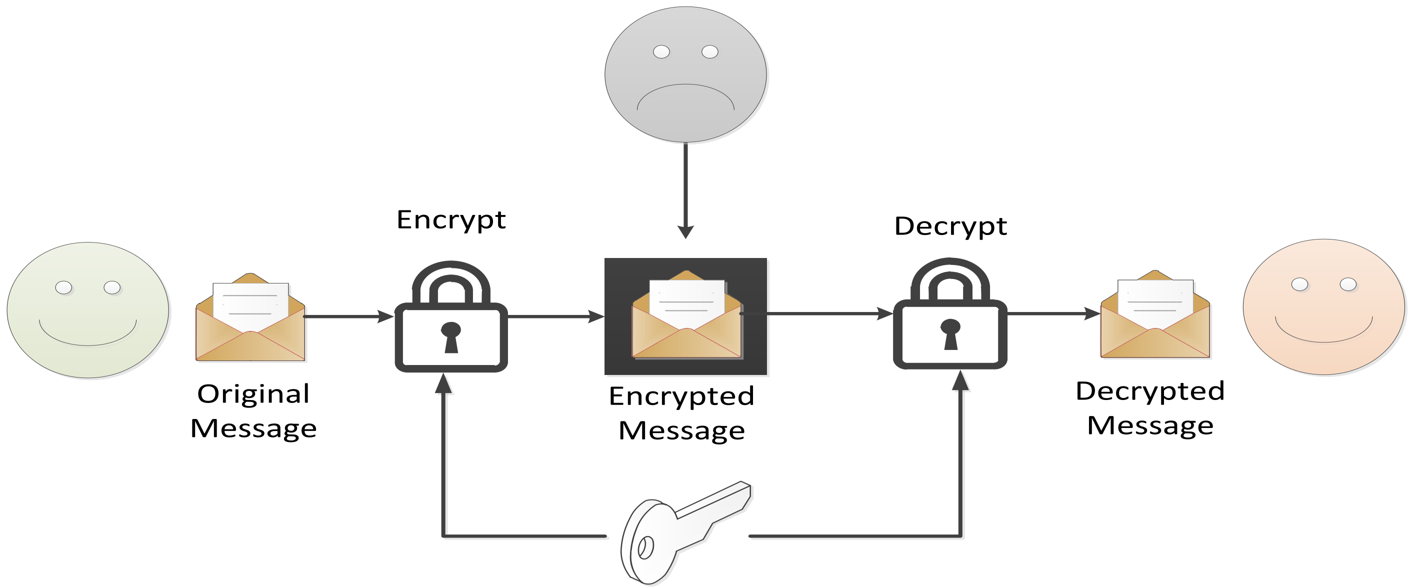
shared key



[Wikipedia]

Cryptographic Ciphers

at the heart of everything



But how does Bob know which key Alice used to encrypt the data? If AES-256 is used, certainly he's not going to try all 2^{256} possibilities!

Public-Key Cryptosystems

public-key primitives and algorithms

Public keys:

- Public-key cryptosystems
  RSA - Rivest, Shamir, Adleman
  ElGamal, McEliece cryptosystems
  ECC - elliptic curve cryptosystems
- Signatures
  DSS/DSA - Digital Signature Standard/Algorithm
  ECDSA - Elliptic Curve Digital Signature Algorithm
- PKI - public-key infrastructure, only if we had it right :-(
  DH - Diffie-Hellman key agreement
  key management, distribution and X.509
- Homomorphic cryptography - Paillier, Gentry

Public-Key Cryptosystems

RSA and ECC

RSA by Rivest-Shamir-Adleman, 1977
has an edge over ECC, because

- it is simple and well understood
- links nicely to basic number theory
- deployed earlier on many systems

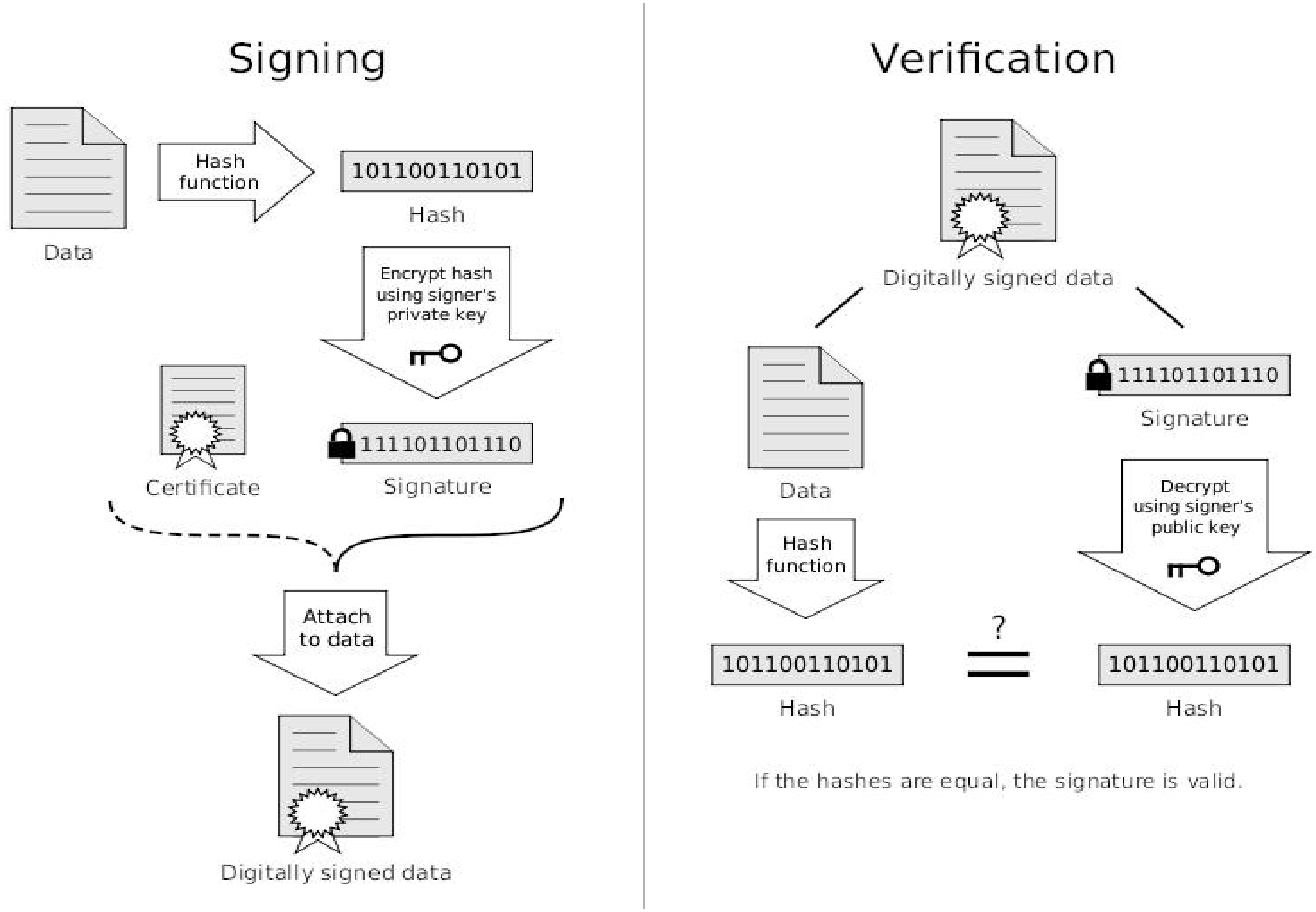
ECC by Koblitz-Miller, 1985
has an edge over RSA, because

- it uses short keys (163+ bits ECC vs. 1024+ bits RSA)
- delivers much better performance
- ECC uses great theory of elliptic curves on top of classical number theory used by RSA

Prediction: finally ECC will take over due to smaller key size

Public-key System in Use

signature by hash and public-key encryption

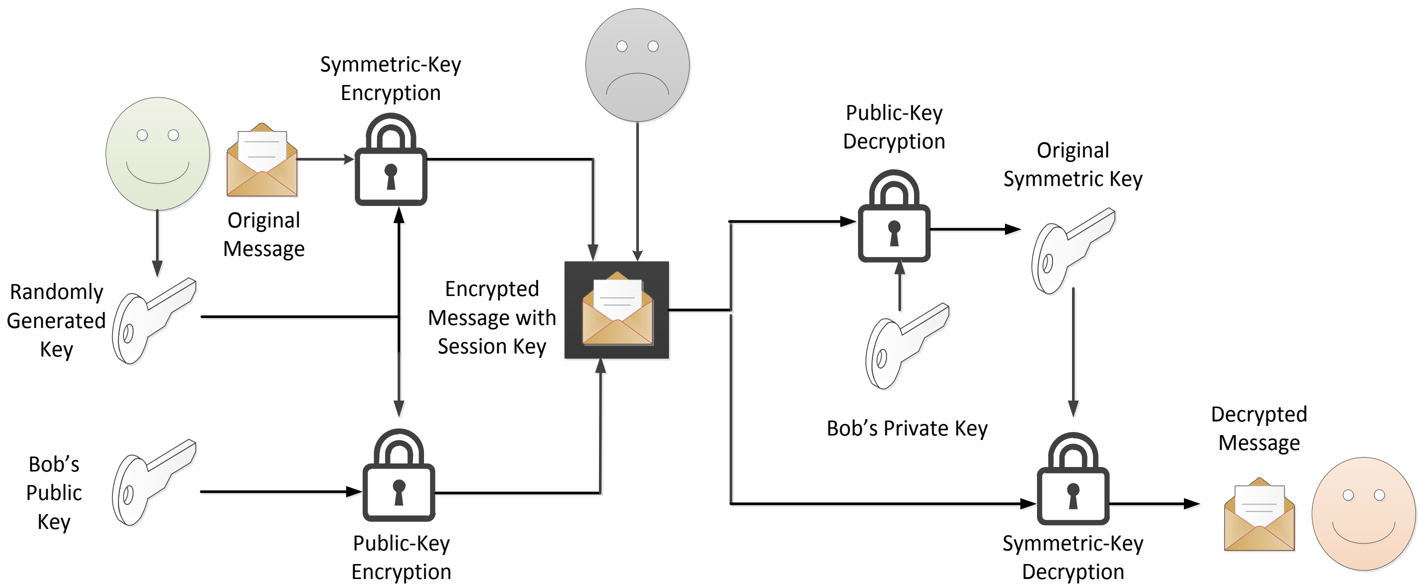


[Wikipedia]

Public-Key Cryptographic in Use

establishing symmetric keys

Symmetric keys are generated randomly and transferred using public-key cryptography.



Since anyone can retrieve Bob's public key, how can he be sure that Alice is the person who sent him the email message?

Cryptography

unkeyed primitives and algorithms

Primitives, algorithms and protocols can be
unkeyed, symmetric-key or public-key

Unkeyed

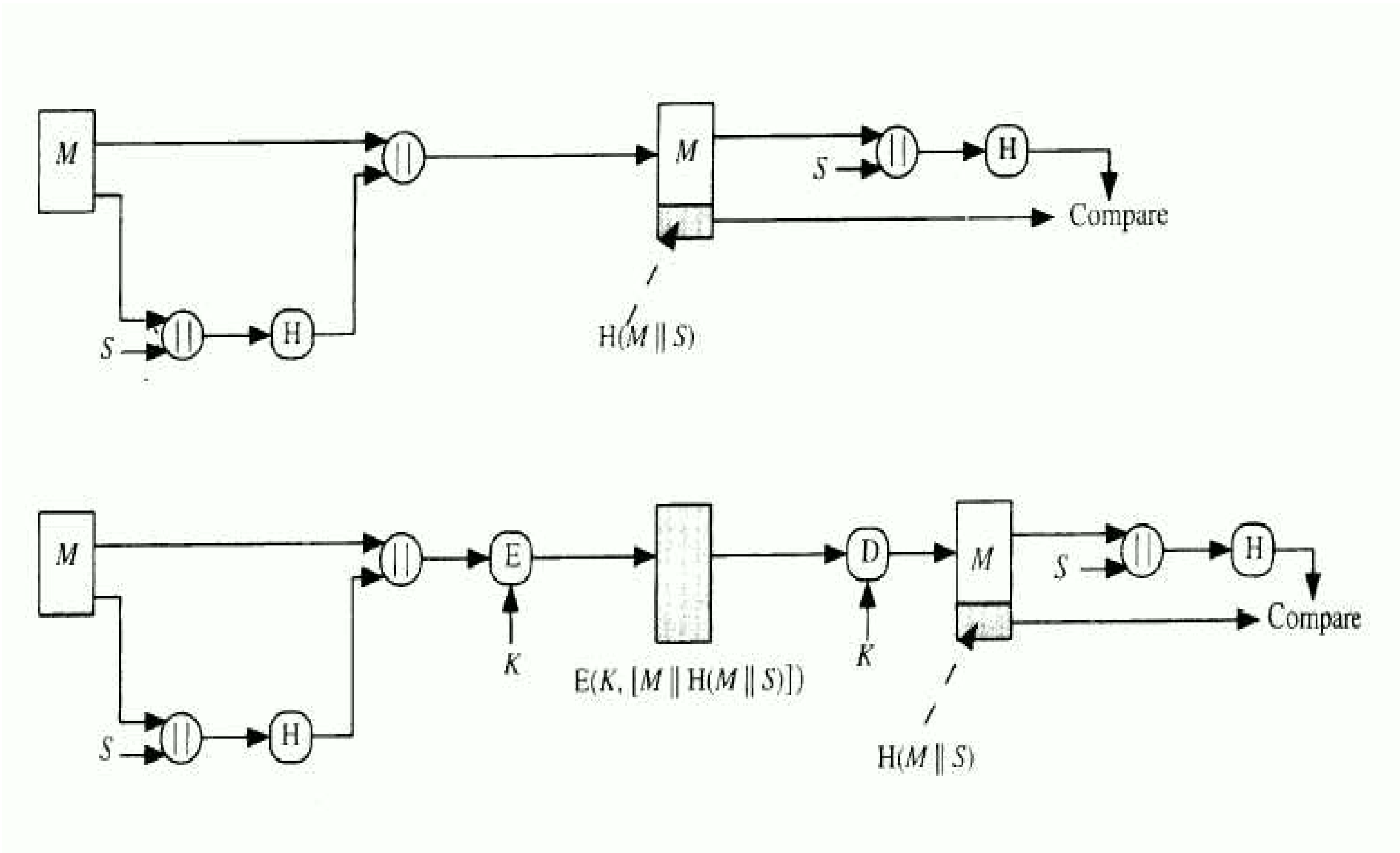
- hashing, SHA-family (large part of these lectures)
- one-way permutations exist, or **NP** is not that much ...

Use

- hash and sign
- random sequences - Blum-Blum-Shub BBS generator, stream cipher outputs, $H(n), H(n+1), H(n+2), \ldots$
- many other ...

Hash in Use

message authentication - clear and encrypted



[Stallings]

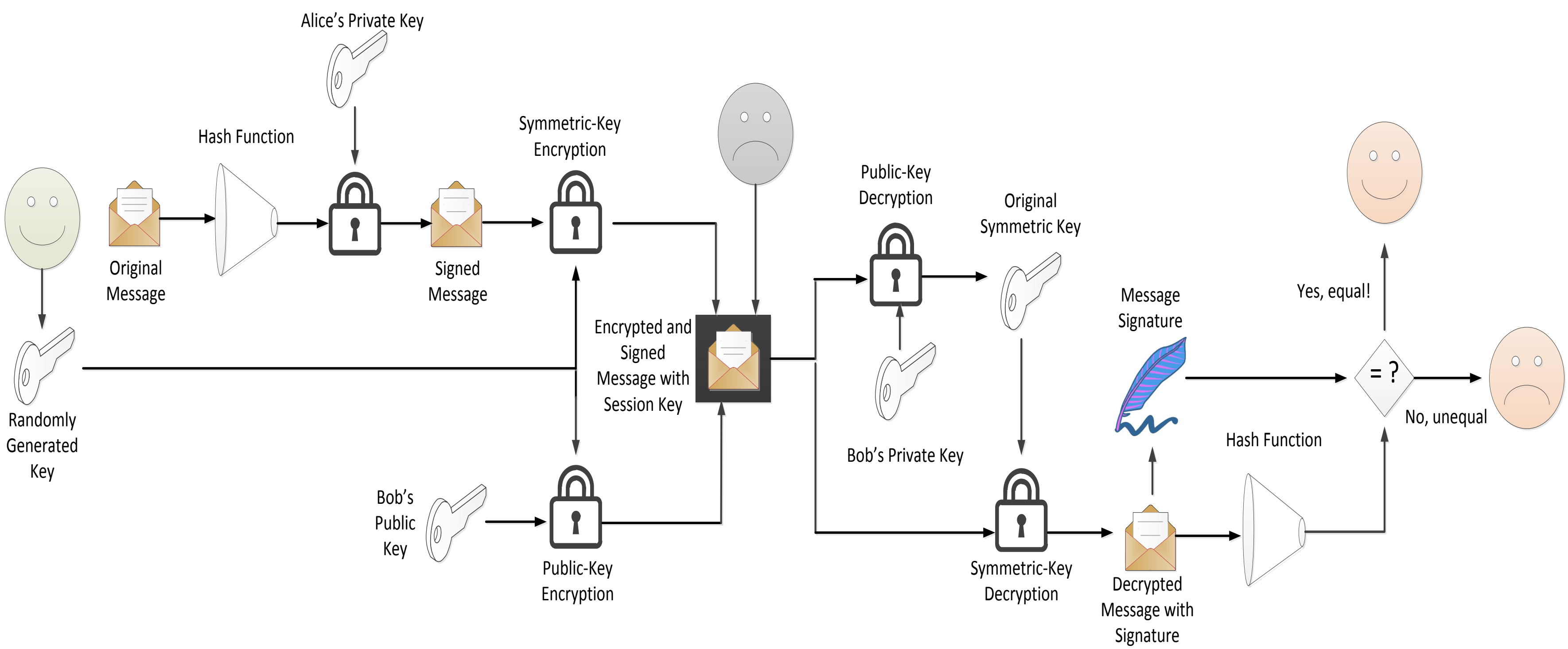
Digital Signatures

combining asymmetric encryption with hash algorithms

- Treat the hash of an email message as the “signature”
- Encrypt the hash digest using the sender’s private key
- The receiver can decrypt the message (using the previous approach) and hash digest, compute the hash of the message, and compare against the received digest.
- If they’re equal, great! If not, something bad happened...

PGP Message Transmission

confidentiality and integrity for email messages



Mathematics in Cryptography

Math in primitives

- Keyless: so far mostly bit juggling,
  we will see soon what kind of math is in SHA-3
- Shared-key: much more since AES ’2001,
  mostly around binary Galois fields $GF(2^k)$
- Public-key: heavy use of number theory,
  now essentially in all PKC, including ECC

Math in cryptanalysis

- Linear and differential cryptanalysis
- Probability and statistics, random oracle models
- Number theoretical algorithms: primality, factoring
- Discrete logarithms: cyclic group discovery, index calculus,
  counting points on elliptic curves, theory of elliptic curves

Cryptography Engineering

evaluation criteria

Security engineer must consider:

- **Level of security.** Or, how many security bits you need.
- **Functionality.** Or, how primitive are the primitives.
- **Performance.** Or, how fast is fast enough.
- **Simplicity.** Is there still anybody who can understand it?

Each party stresses a different measure:

- risk (politicians)
- cost (managers)
- use (most of us)

Can security/software engineer satisfy all of them?

Cryptography

limits

Cryptography is an important, but only a relatively small part of security:

- right choice of tools is hard
- implementation errors are common
- variety of side-channel attacks can bypass best crypto
- social attacks

References

- Niels Ferguson, Bruce Schneier and Tadayoshi Kohno, *Cryptography Engineering*, John Wiley & Sons, 2010.
- Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, *CRC Handbook of Applied Cryptography*, CRC Press 1996

  `http://www.cacr.math.uwaterloo.ca/hac`
- Bruce Schneier, *Applied Cryptography*, second edition, John Wiley & Sons, 1996.
- William Stallings, *Cryptography and Network Security. Principles and Practice*, fifth edition, Prentice Hall, 2011.
- Douglas R. Stinson, *Cryptography: Theory and Practice*, third edition, CRC Press 2006.

R·I·T
Department of Computer Science