IN2120 Information Security
University of Oslo
Autumn 2019

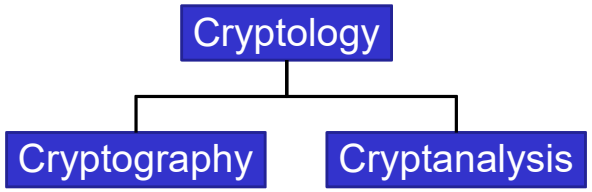
Lecture 2
Cryptography

University of Oslo, Autumn 2019
Audun Jøsang

Outline

- What is cryptography?
- Brief crypto history
- Symmetric cryptography
  - Stream ciphers
  - Block ciphers
  - Hash functions
- Asymmetric cryptography
  - Encryption
  - Diffie-Hellman key exchange
  - Digital signatures
  - Post-Quantum Crypto

Terminology

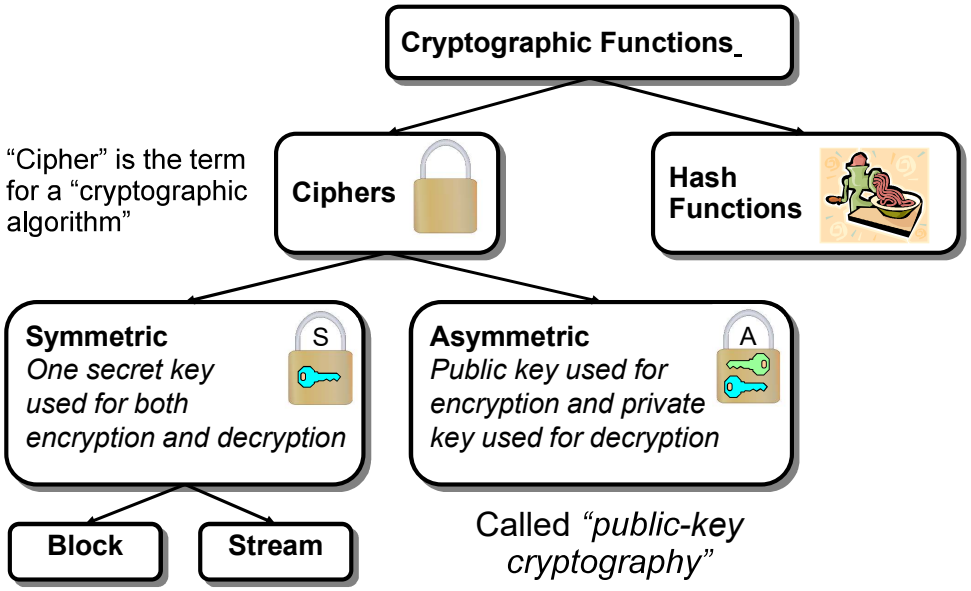


- **Cryptography** is the science of secret writing with the goal of hiding the meaning of a message.
- **Cryptanalysis** is the science of *breaking* cryptography.
- **Cryptology** covers both cryptography and cryptanalysis.

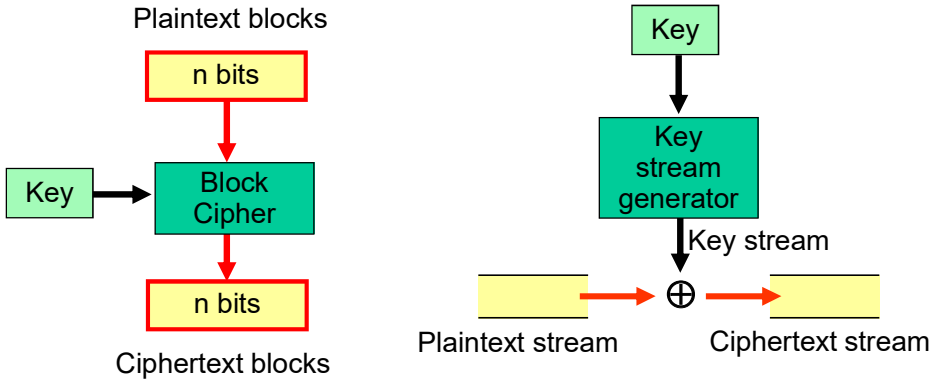
What can cryptography do?

- Crypto can provide the following security services:
  - **Confidentiality:**
    - Makes data unreadable to entities who do not have the appropriate cryptographic keys, even if they have the data.
  - **Data Integrity:**
    - Entities with the appropriate cryptographic keys can verify that data is correct and has not been altered, either deliberately or accidentally.
  - **Authentication:**
    - Entities who communicate can be assured that the other user/entity or the sender of a message is what it claims to be.
  - **Digital Signature and PKI (Public-Key Infrastructure):**
    - Strong proof of data origin which can be verified by 3rd parties.
    - Scalable (to the whole Internet) distribution of cryptographic keys.

Taxonomy of cryptographic functions

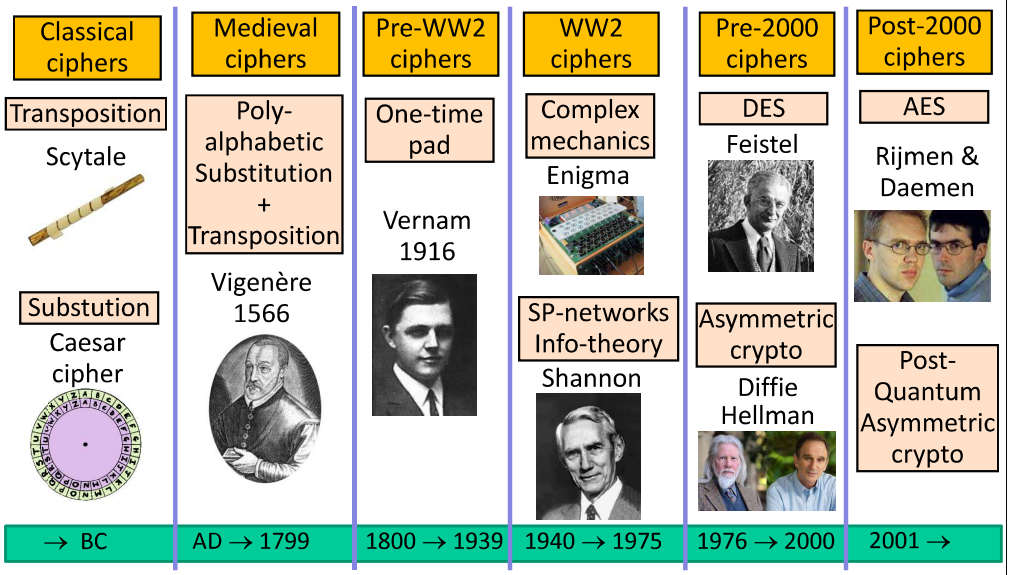


Block Cipher vs. Stream Cipher



Note that the key stream repeats itself and is not totally random, hence a stream cipher is not a One-Time-Pad.

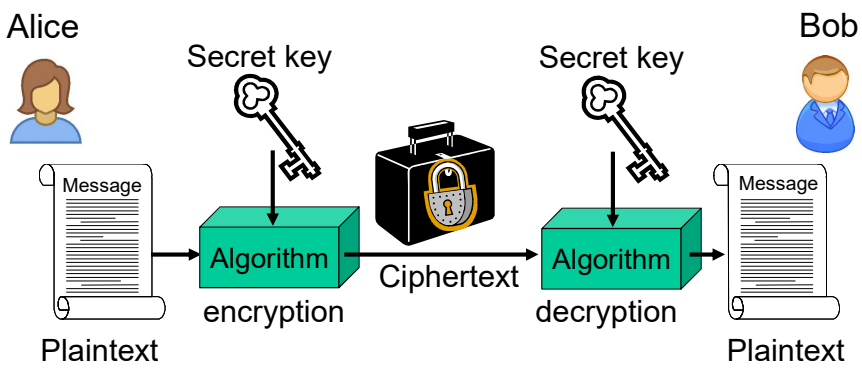
Evolution of Ciphers



Terminology

- Encryption: plaintext (cleartext) M is converted into a ciphertext C under the control of a key k.
  - We write $C = E(M, k)$.
- Decryption with key k recovers the plaintext M from the ciphertext C.
  - We write $M = D(C, k)$.
- Symmetric ciphers: the secret key is used for both encryption and decryption.
- Asymmetric ciphers: Pair of private and public keys where it is computationally infeasible to derive the private decryption key from the corresponding public encryption key.

Symmetric cryptography (secret key)



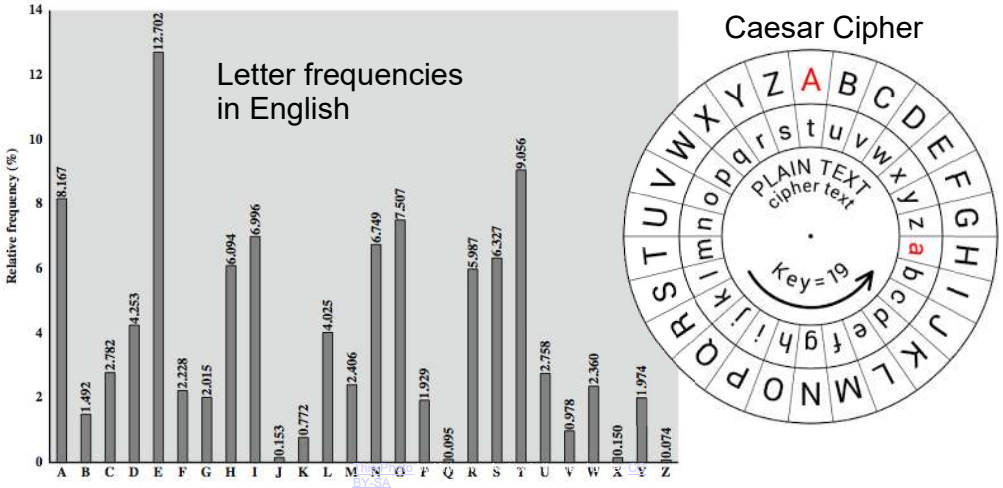
- "Secret key" means that the key is shared "in secret" between entities who are authorized to encrypt and decrypt

Strength of Ciphers

Factors for cryptographic strength:

- **Key size**.
  - Exhaustive key-search time depends on the key size.
  - Typical key size for a symmetric cipher is 256 bit.
  - Attacker must try $2^{256}/2$ keys on average to find the key, which would take millions of years, which is not practical.
  - With N different keys, the key size is $\log_2(N)$.
- **Algorithm strength**.
  - Key discovery by cryptanalysis can exploit statistical regularities in the ciphertext.
  - To prevent cryptanalysis, the bit-patterns / characters in the ciphertext should have a uniform distribution, i.e. all bit-patterns / characters should be equally probable.

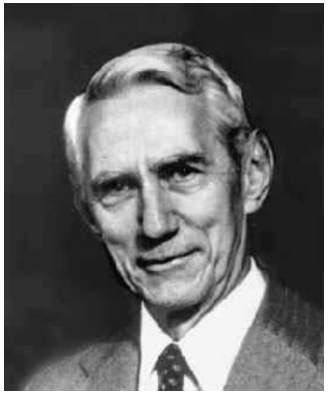
Letter Frequencies → Statistical cryptanalysis



Historic ciphers, like the Caesar Cipher, are weak because they fail to hide statistical regularities in the ciphertext.

Claude Shannon (1916 – 2001)

The Father of Information Theory – MIT / Bell Labs

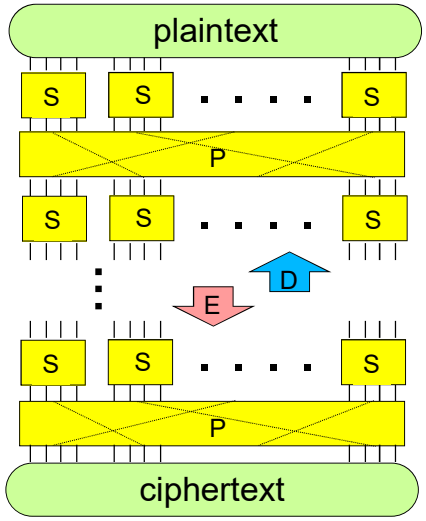


- Information Theory
  - Defined the „binary digit" (bit) as information unit
  - Defined information „entropy" to measure amount of information
- Cryptography
  - Model of secrecy systems
  - Defined perfect secrecy
  - Principle of S-P encryption (substitution & permutation) to hide statistical regularities

Shannon's S-P Network

Removes statistical regularities in ciphertext

- "S-P Networks" (1949)
  - **Substitutions** & **Permutations**
  - Substitute bits e.g. 0001 with 0110
  - Permute parts e.g. part-1 to part-2
  - Substitution provides "confusion" i.e. complex relationship between input and output
  - Permutation provide "diffusion", i.e. a single input bit influences many output bits
  - Iterated S-P functions a specific number of times
  - Functions **must be invertible**



AES - Advanced Encryption Standard

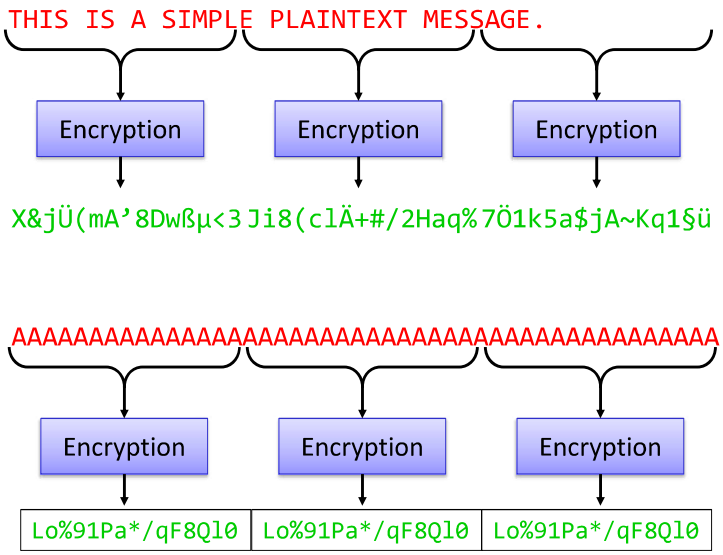
- DES (Data Encryption Standard) from 1977 had a 56-bit key and a 64-bit block. In the mid-1990s DES could be cracked with exhaustive key search.
- In 1997, NIST announced an open competition for a new block cipher to replace DES.
- The best proposal called "Rijndael" was nominated as AES (Advanced Encryption Standard) in 2001.
- AES has key sizes of 128, 192 or 256 bit and block size of 128 bit.

AES is designed by Vincent Rijmen and Joan Daemen from Belgium

Block Ciphers: Modes of Operation

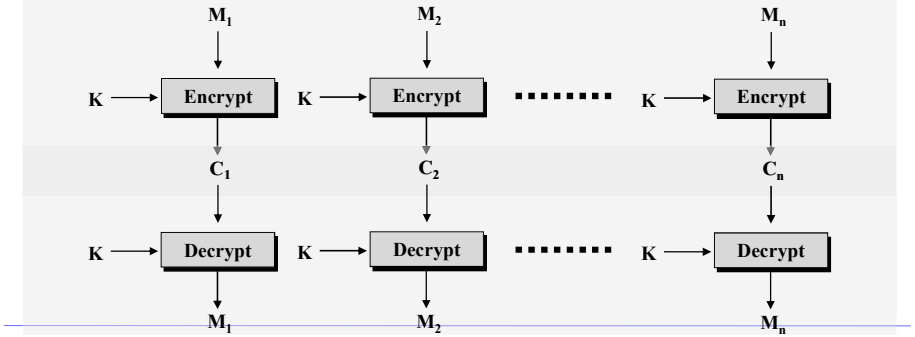
- Block ciphers can be used in different modes in order to provide specific security protection.
- Common modes include:
  - **E**lectronic **C**ode **B**ook (ECB) — *Insecure*
  - **C**ipher **B**lock **C**haining (CBC)
  - **O**utput **F**eed**B**ack (OFB)
  - **C**ipher **F**eed**B**ack (CFB)
  - **C**oun**T**e**R** Mode (CTR) — *Secure*

Electronic Code Book (ECB-mode)

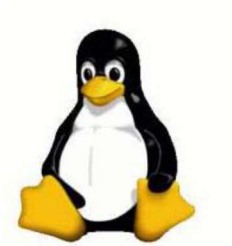


Electronic Code Book

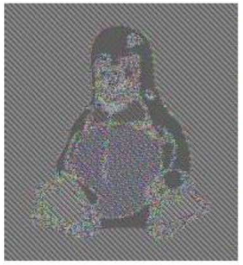
- ECB Mode encryption
  - Simplest mode of operation
  - Plaintext data is divided into blocks $M_1, M_2, \ldots, M_n$
  - Each block is then processed separately
    - Plaintext block and key used as inputs to the encryption algorithm



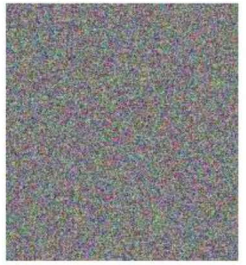
Vulnerability of ECB-mode



Plaintext

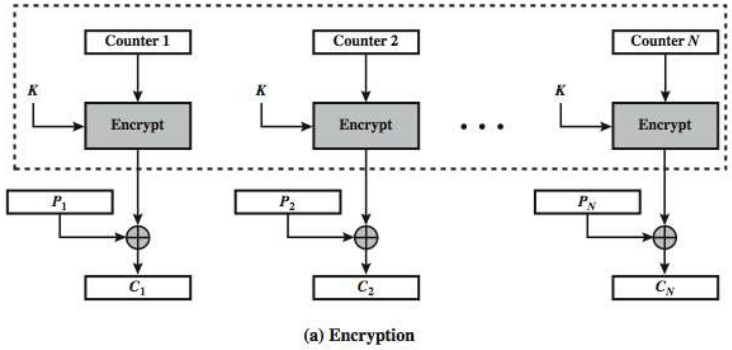


Ciphertext using ECB mode

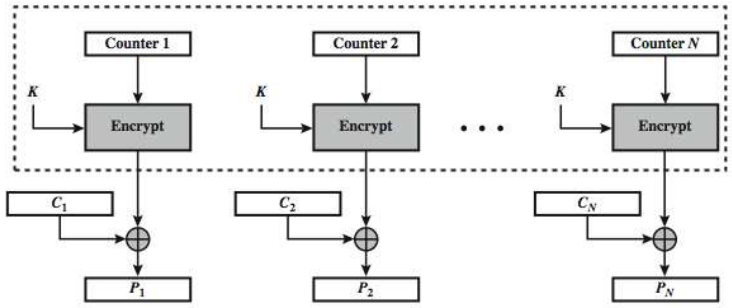


Ciphertext using secure mode

CTR Counter Mode

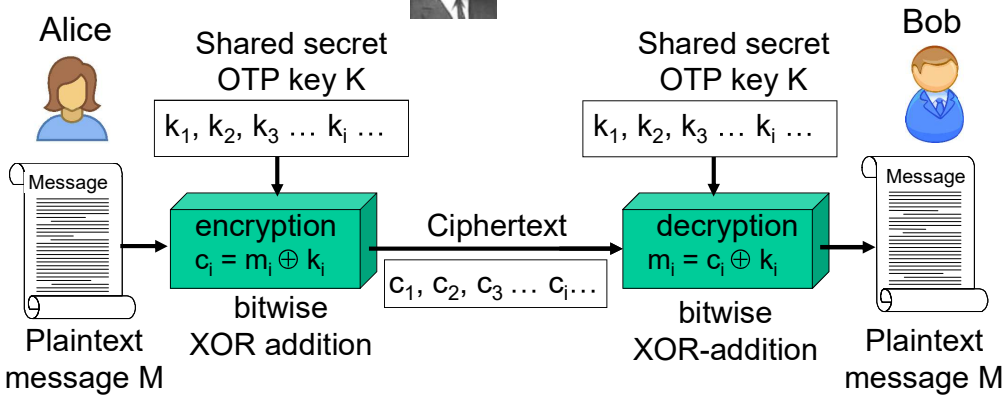


(a) Encryption



(b) Decryption

One-Time-Pad:  Gilbert Vernam, 1917



- Property of bitwise XOR addition:
  $k_i \oplus k_i = 0$ and $m_i = c_i \oplus k_i = m_i \oplus k_i \oplus k_i$
- OTP offers perfect security assuming the OTP key is perfectly random, of same length as the message, and only used once

The perfect cipher: One-Time-Pad

- Old version used a paper tape of random data
- Modern versions can use DVDs with Gbytes of random data

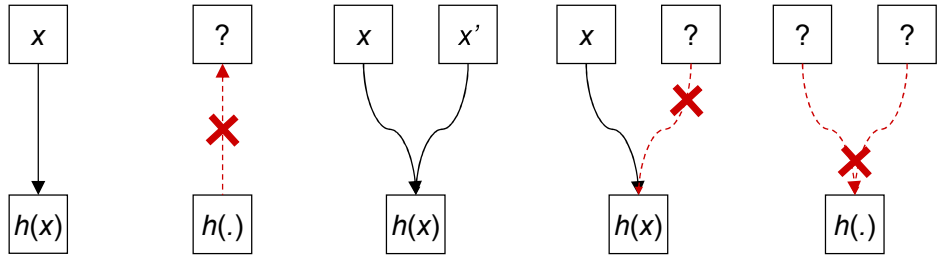
Integrity Check Functions
- Hash functions
- MAC functions

Hash functions (message digest functions)

Requirements for a one-way hash function *h*:

1. Ease of computation: given *x*, it is easy to compute *h*(*x*).
2. Compression: *h* maps inputs *x* of arbitrary bitlength to outputs *h*(*x*) of a fixed bitlength *n*.
3. One-way: given a value *y*, it is computationally infeasible to find an input *x* so that *h*(*x*)=*y*.
4. Collision resistance: it is computationally infeasible to find *x* and *x'*, where $x \neq x'$, with *h*(*x*)=*h*(*x'*) (note: two variants of this property).

Properties of hash functions



Ease of computation | Pre-image resistance | Collisions exist but are hard to find | Weak collision resistance (2nd pre-image resistance) | Strong collision resistance

Applications of hash functions

- Comparing files
- Protection of password
- Authentication of SW distributions
- Bitcoin
- Generation of Message Authentication Codes (MAC)
- Digital signatures
- Pseudo number generation/Mask generation functions
- Key derivation

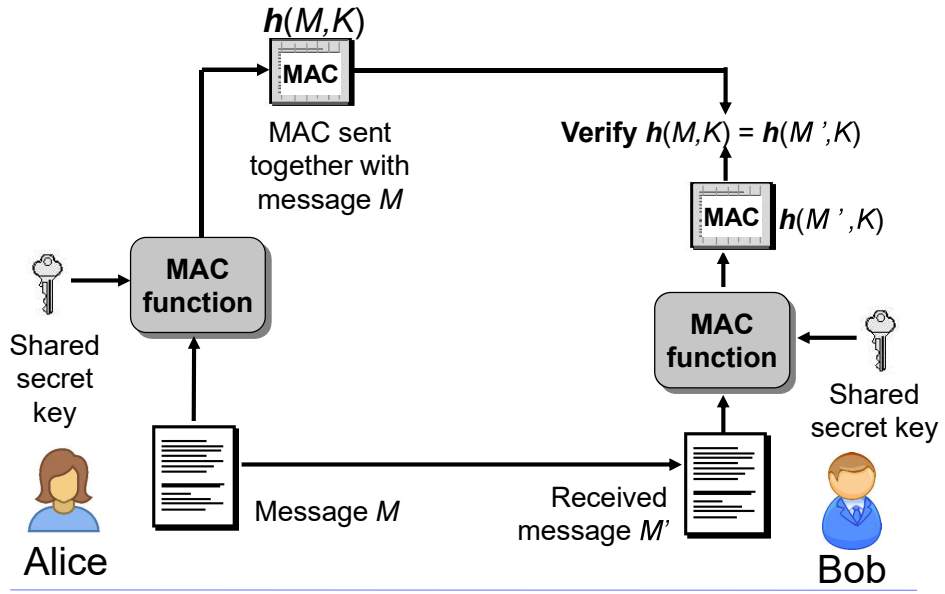
Well-known hash functions

- **MD5** (1991): 128 bit digest. Relatively easy to break by finding collisions, due to short digest and poor design. Not to be used in new applications, but may be used in legacy applications.
- **SHA-1** (Secure Hash Algorithm):160 bit digest. Designed by NSA in 1995 to operate with DSA (Digital Signature Standard). Attacks exist. Not recommended, but sometimes still in use.
- **SHA-2** designed by NSA in 2001 provides 224, 256, 384, and 512 bit digest. Considered secure. Replacement for SHA-1.
- **SHA-3:** designed by Joan Daemen + others in 2010. Standardized in 2015. Digest of: 224, 256, 384, and 512 bit. SHA-3 has little use, because SHA-2 is considered strong.

Message Authentication Codes

- A message *M* with a simple message hash *h*(*M*) can be changed by attacker.
- In communications, we need to verify the origin of data, i.e. we need message authentication.
- MAC (message authentication code) can use hash function as *h*(*M, k*) i.e. with message *M* and a secret key *k* as input.
- To validate and authenticate a message, the receiver has to share the same secret key used to compute the MAC with the sender.
- A third party who does not know the key cannot validate the MAC.

Practical message integrity with MAC



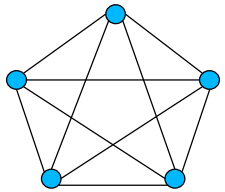
MAC and MAC functions

- Terminology
  - MAC is the computed message authentication code $h(M, k)$
  - MAC function is the algorithm used to compute a MAC
- Different types of MAC functions are e.g.
  - HMAC (Hash-based MAC algorithm))
  - CBC-MAC (CBC based MAC algorithm)
  - CMAC (Cipher-based MAC algorithm)
- MAC functions, a.k.a. keyed hash functions, support data origin authentication services.

Public-Key Cryptography

Problem of symmetric key distribution

- Shared key between each pair
- In network of *n* users, each participant needs *n*-1 keys.
- Number of exchanged secret keys:
  = $n(n-1)/2$
  = number of glasses touching at cocktail party
- Grows exponentially, which is a major problem.
- Is there a better way?
  - Public-key cryptography



Network of 5 nodes

Cocktail party

James H. Ellis (1924 – 1997)

Inventor of pub-key crypto, but received little recognition

- British engineer and mathematician
- Worked at GCHQ (Government Communications Headquarters)
- Idea of non-secret encryption to solve key distribution problem
- Encrypt with non-secret information in a way which makes it impossible to decrypt without related secret information
- Never found a practical method

Clifford Cocks (1950 –)

Inventor of RSA algorithm in 1973, recognized in 1998

- British mathematician and cryptographer
- Silver medal at the International Mathematical Olympiad, 1968
- Worked at GCHQ (equivalent to NSA)
- Heard from James Ellis the idea of non-secret encryption in 1973
- Spent 30 minutes in 1973 to invent a practical method
- Equivalent to the RSA algorithm
- Was classified TOP SECRET
- Result revealed in 1998
- Fellow of the British Royal Society in 2015.

Malcolm J. Williamson (1950 – 2015)

Inventor of key exchange but received little recognition

- British mathematician and cryptographer
- Gold medal at the International Mathematical Olympiad, 1968
- Worked at GCHQ until 1982
- Heard from James Ellis the idea of non-secret encryption, and from Clifford Cocks the practical method.
- Intrigued, spent 1 day in 1974 to invent a method for secret key exchange without secret channel
- Equivalent to the Diffie-Hellmann key exchange algorithm

Ralph Merkle, Martin Hellman and Whitfield Diffie

- Merkle invented (1979) the Merkle Hash Tree and the Merkle Digital Signature Scheme, used e.g. in Bitcoin. Resistant to quantum computers.
- Diffie & Hellman(1976) invented a practical key exchange algorithm with discrete exponentiation.
- D&H defined public-key encryption (equiv. to non-secret encryption) (1976)
- Defined digital signature
- *"New directions in cryptography" (1976)*

Diffie-Hellman key agreement (key exchange)

(provides no authentication)

Alice picks private random integer a

Bob picks private random integer b

$g^a \bmod p$

$g^b \bmod p$

Alice computes the shared secret $(g^b)^a = g^{ab} \bmod p$

Bob computes the same shared secret $(g^a)^b = g^{ab} \bmod p$.

Attackers can not recover the integers a or b because discrete logarithm of large integers is computationally difficult. Hence, attackers are unable to compute the secret key = $g^{ab} \bmod p$.

Applications of Diffie-Hellman Key Exchange

- IPSec (IP Security)
  - IKE (Internet Key Exchange) is part of the IPSec protocol suite
  - IKE is based on Diffie-Hellman Key Agreement
- SSL/TLS
  - Several variations of SSL/TLS protocol including
    - Fixed Diffie-Hellman
    - Ephemeral Diffie-Hellman
    - Anonymous Diffie-Hellman

Ron Rivest, Adi Shamir and Len Adleman

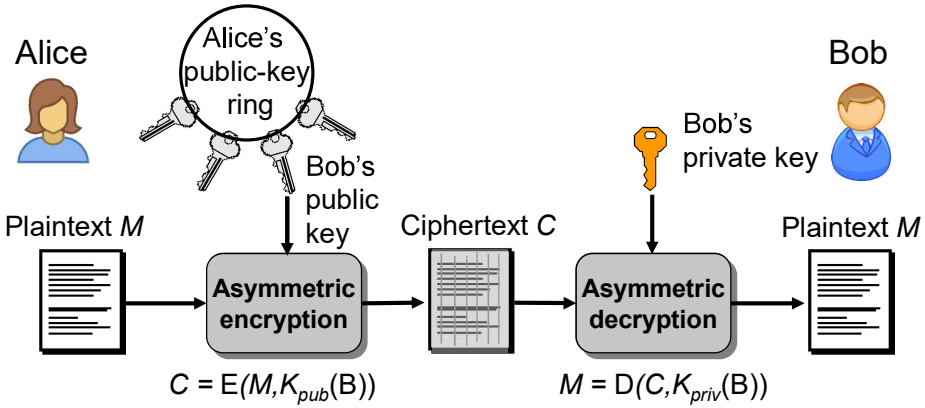


- Read about public-key cryptography in 1976 article by Diffie & Hellman: *"New directions in cryptography"*
- Intrigued, they worked on finding a practical algorithm
- Spent several months in 1976 to re-invent the method for non-secret/public-key encryption discovered by Clifford Cocks 3 years earlier
- Named RSA algorithm
- Uses a pair of keys: public key and private key

Asymmetric Ciphers: Examples of Cryptosystems

- RSA: best known asymmetric algorithm.
  - RSA = Rivest, Shamir, and Adleman (published 1977)
  - Historical Note: U.K. cryptographer Clifford Cocks invented the same algorithm in 1973, but didn't publish.
- ElGamal Cryptosystem
  - Based on the difficulty of solving the discrete log problem.
- Elliptic Curve Cryptography
  - Based on the difficulty of solving the EC discrete log problem.
  - Provides same level of security with smaller key sizes.

Asymmetric Encryption: Basic encryption operation

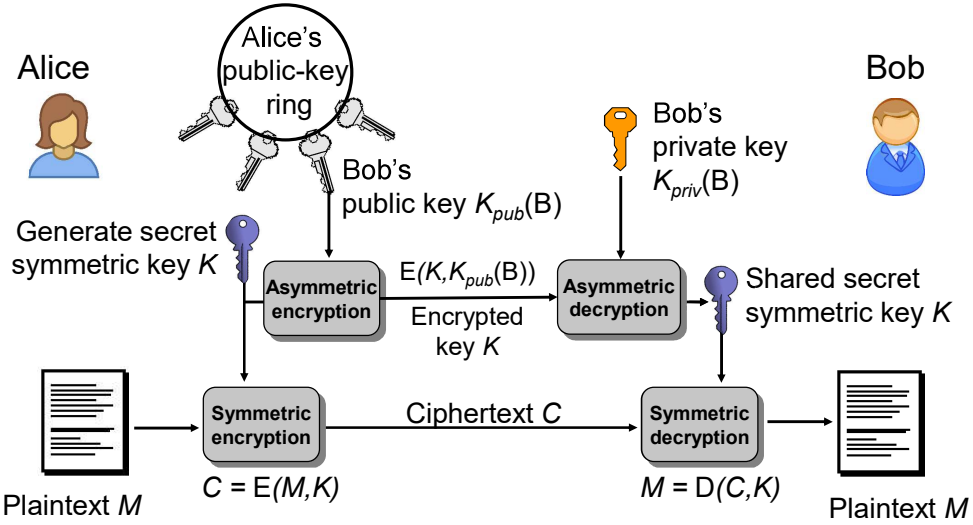


- In practical applications, large messages are not encrypted directly with asymmetric algorithms. Hybrid systems are used.

Hybrid Cryptosystems

- Symmetric ciphers are faster than asymmetric ciphers (because they are less computationally expensive), but ...
- Asymmetric ciphers simplify key distribution, therefore ...
- a combination of both symmetric and asymmetric ciphers can be used – a hybrid system:
  - The asymmetric cipher is used to distribute a randomly chosen symmetric key.
  - The symmetric cipher is used for encrypting bulk data.

Confidentiality Services: Hybrid Cryptosystems

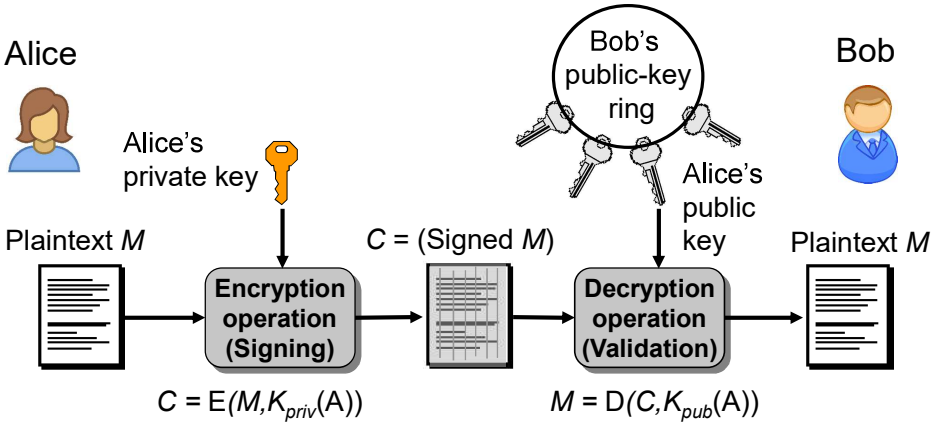


Digital Signatures

Digital Signature Mechanisms

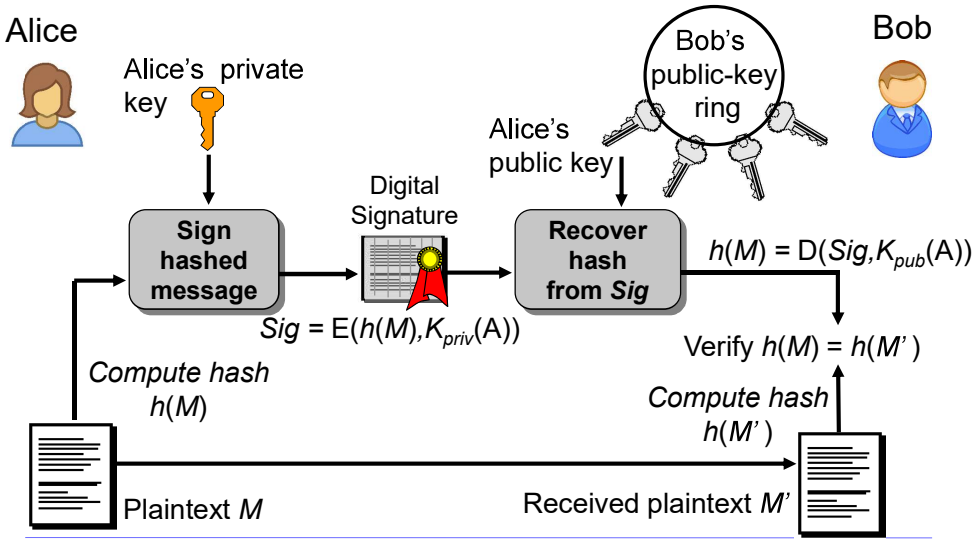
- A MAC cannot be used as evidence to be verified by a 3rd party.
- Digital signatures can be verified by 3rd party.
  - Used for non-repudiation,
  - data origin authentication and
  - data integrity
- Digital signature mechanisms have three components:
  - key generation
  - signing procedure (private)
  - verification procedure (public)

Digital signature: Basic operation

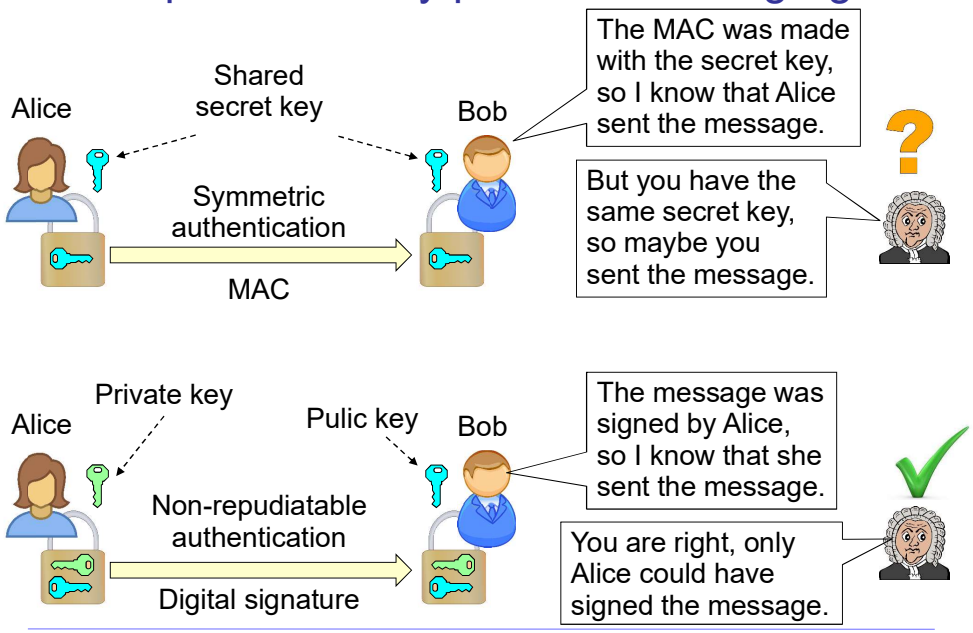


- In practical applications, message *M* is not signed directly, only a hash value *h*(*M*) is signed.

Practical digital signature based on hash value



Non-repudiation only possible with DigSig




SINET ≡ DN.no Logg inn Kunde

Norge kan snart ikke vokte statshemmeligheter lenger

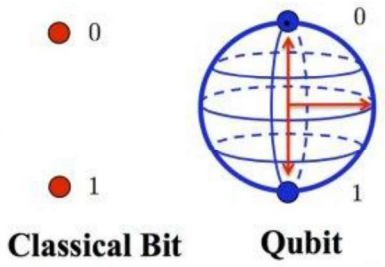
TEKNOLOGI

TEKST Osman Kibar FØLG MEG
FOTO Åge Peterson, Gorm K. Gaare & Helge Skodvin
01 DESEMBER 2017 KOLSÅS/OSLO

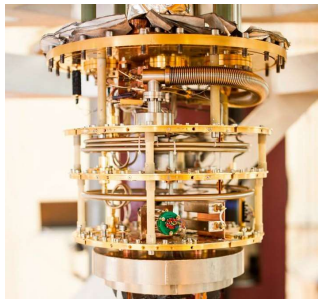
DN.no, 1 December 2017

Principle for Quantum Computing

- Quantum Computing (QC) uses quantum superpositions instead of binary bits to perform computations.



Experimental Quantum Computer



- Quantum algorithms, i.e. algorithms for quantum computers, can solve certain problems much faster than classical computer algorithms.

QC Threat to Traditional Cryptography

- Shor's Quantum Algorithm (1994) can factor integers and compute discrete logarithms efficiently. With a powerful quantum computer (at least 1 million qubits), Shor's algorithm would be devastating to traditional public key crypto algorithms.

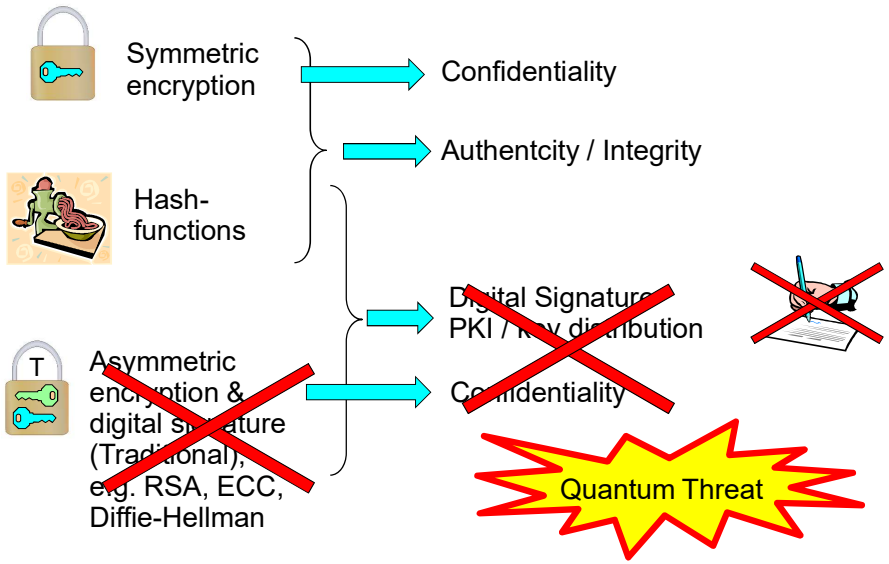
- Grover's Quantum Search Algorithm (1996) can be used to brute-force search for a *k*-bit secret key with an effort of only

$$\sqrt{2^k} = 2^{k/2}$$

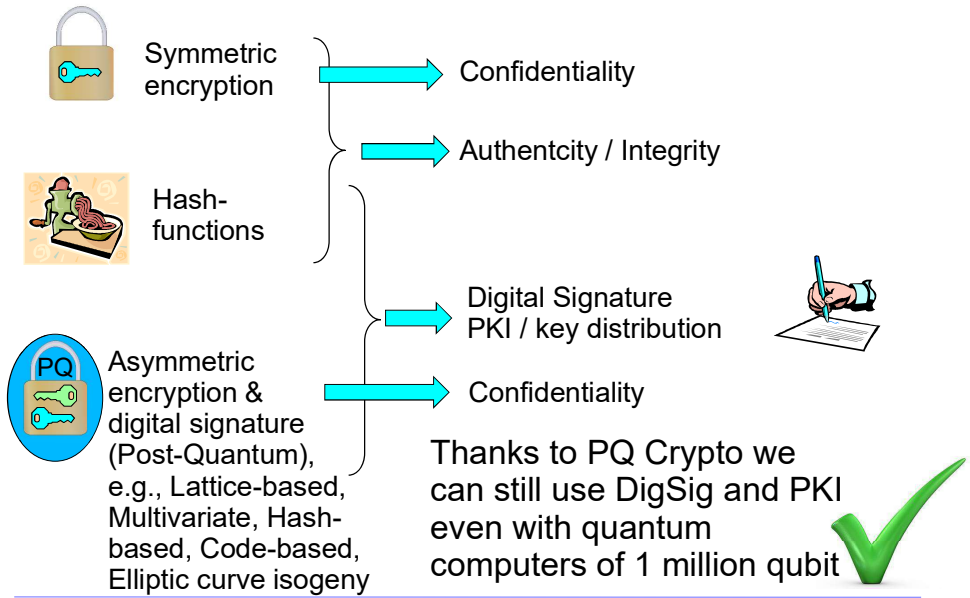
which effectively doubles the required key sizes for ciphers.

- QC has been dismissed by most cryptographers until recent years. General purpose quantum computers do not currently exist, but are predicted to be built in foreseeable future.

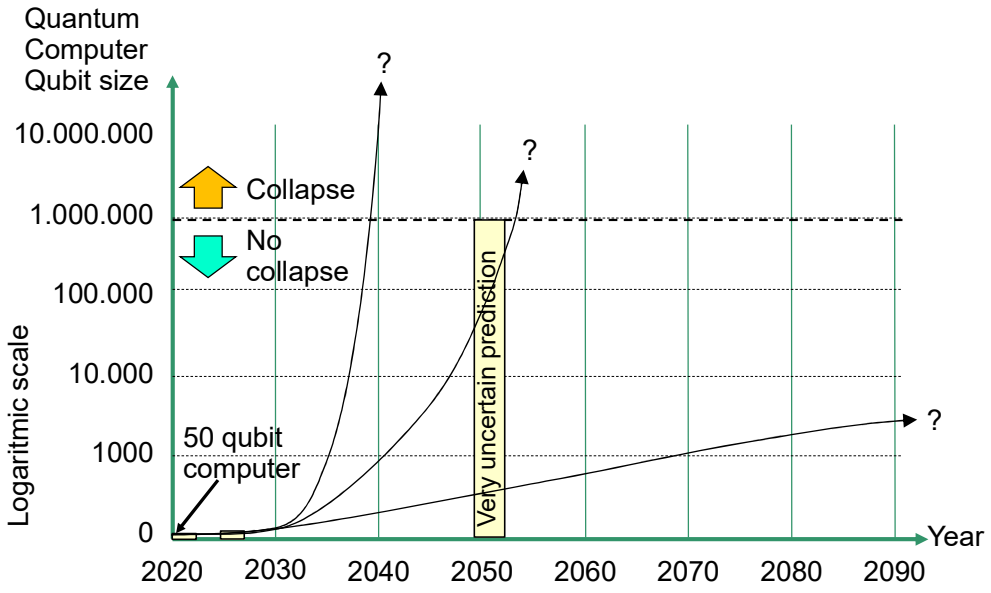
Cryptographic Functions and Services



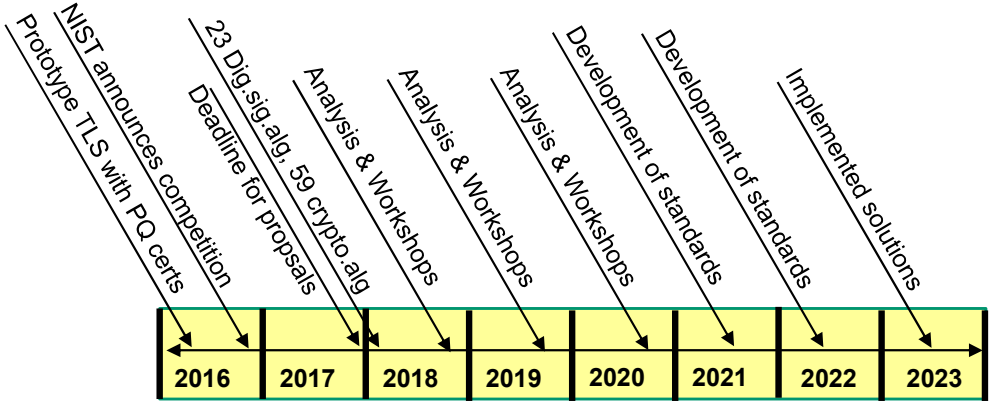
Cryptographic Functions and Services



Collapse of traditional asymmetric crypto?

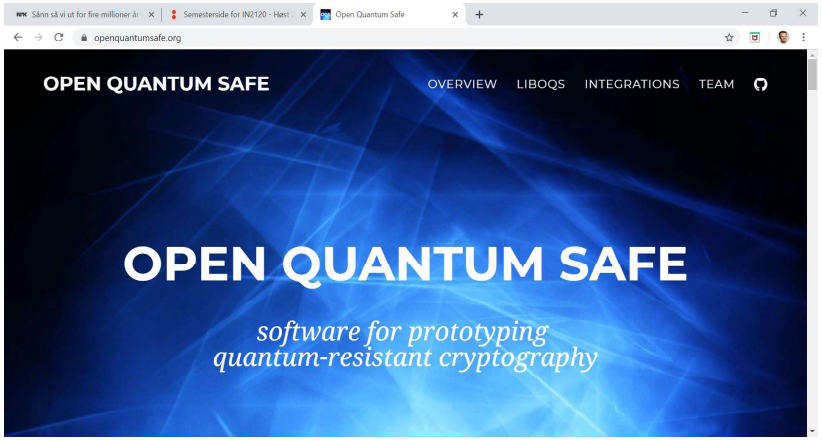


Towards Standardized PQC



- The term “Post-Quantum Crypto” means crypto which is resistant to powerful quantum computers.
- Many organizations plan to start using PQC just to be on the safe side, and not risk bad publicity.

PQC already works



- Many initiatives for prototyping PQC in real applications
- Version of Chrome Browser with PQC TLS
- Disadvantage of PQC is high complexity and computation load

End of lecture