

INF3510 Information Security

University of Oslo

Spring 2015

Lecture 5

Cryptography



University of Oslo, spring 2015
Audun Jøsang

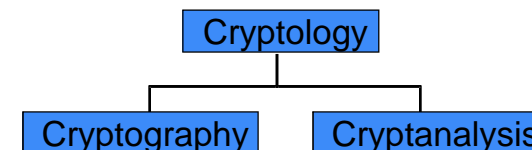
Agenda

- The concept of cryptography
- Symmetric-key ciphers
- Use of block ciphers
- Hash functions
- Public-key ciphers
- Digital signature

When is cryptography used?

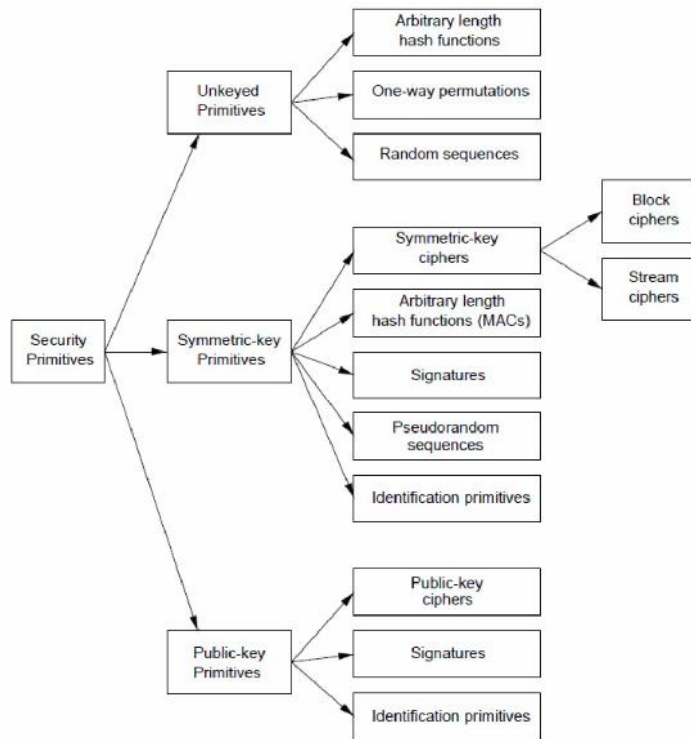
- If you require
 - **Confidentiality:**
 - So that your data is not made available to anyone who shouldn't have access.
 - That is, protection against snoops or eavesdroppers
 - **Data Integrity:**
 - So you know that the message content is correct, and has not been altered, either deliberately or accidentally
 - **Data Authentication:**
 - So you can be sure that the message is from the place or sender it claims to be from
- Data integrity and data authentication are equivalent.
 - Think about it !

Terminology

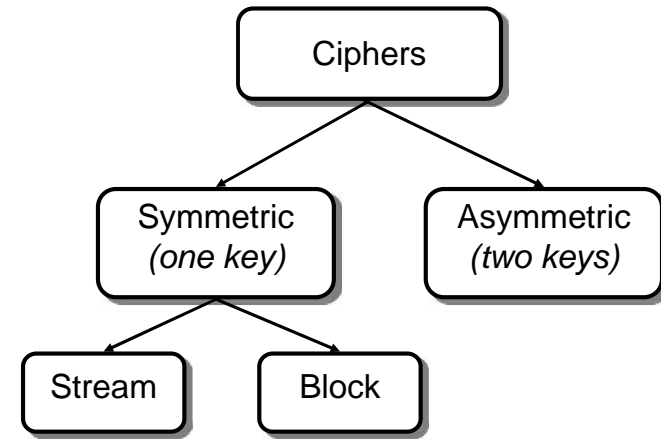


- **Cryptography** is traditionally the science of secret writing with the goal of hiding the meaning of a message (for confidentiality).
- **Cryptanalysis** is the science (and sometimes art) of *breaking* cryptosystems.
- Modern cryptography is used for many other things than just encryption for confidentiality.

Taxonomy of cryptographic primitives

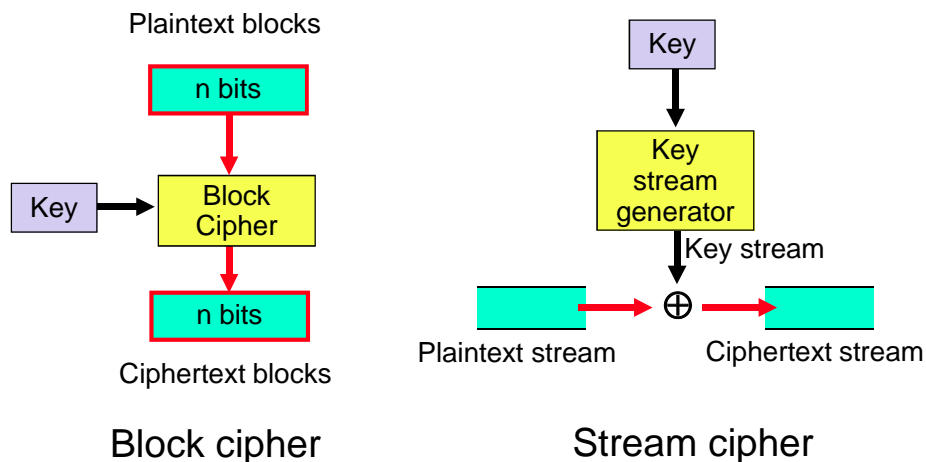


Taxonomy of modern ciphers (subset of previous diagram)



Ciphers are used to encrypt data for confidentiality

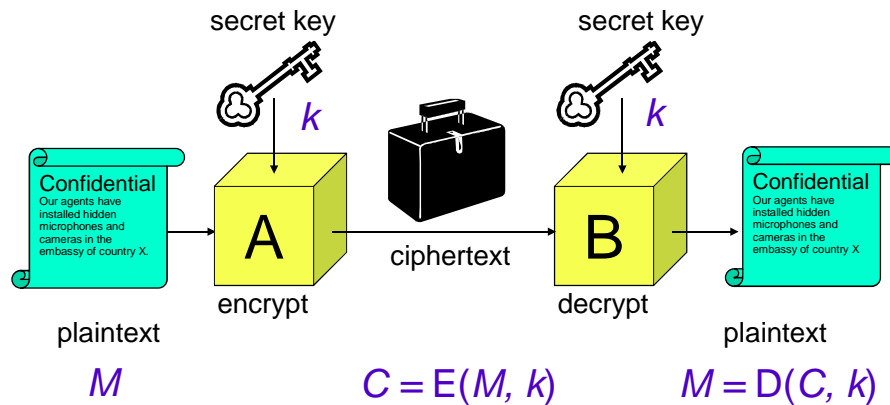
Block Cipher vs. Stream Cipher



Terminology

- **Encryption:** plaintext (clear text) 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:** 1 secret key used for both encryption and decryption.
- **Asymmetric ciphers:** Pair of private and public keys used. Computationally infeasible to derive the **private decryption key** from the corresponding **public encryption key**.

Symmetric Key Encryption



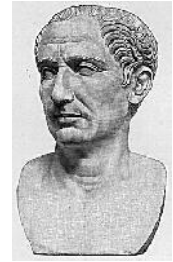
Caesar Cipher

Shifting letters in the alphabet

Example Caesar Cipher

$P = \{abcdefghijklmnopqrstuvwxyz\}$

$C = \{DEFGHIJKLMNOPQRSTUVWXYZABC\}$



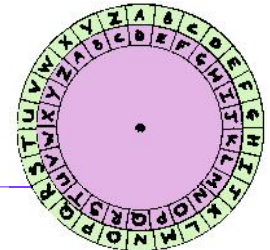
Plaintext: Security is very interesting

Chiphertext: VHFUXULWB LU YHRB LQWHUHVWLQJ

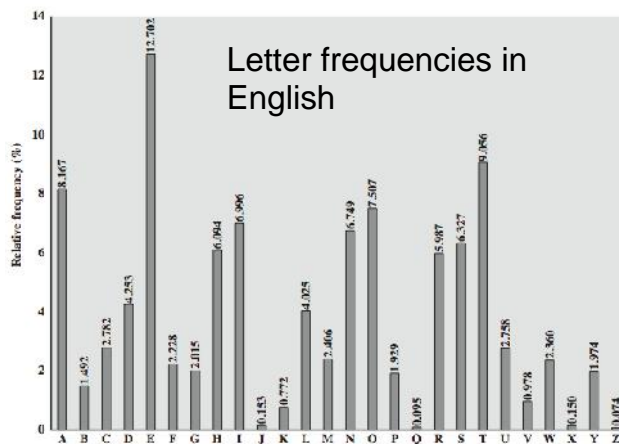
Note: Caesar Cipher in this form has fixed key $K = 3$.

Problems with Caesar Cipher:

- Short key
- Does not hide statistical patterns



Letter Frequencies → statistical attacks



- Encryption must hide statistical patterns in data
- Achieved with a series of primitive functions

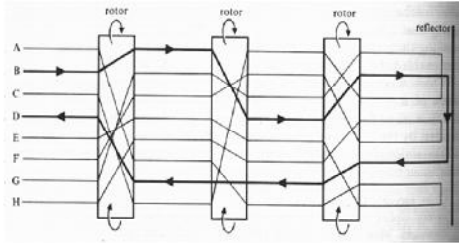
Kerckhoffs' principles (1883)



- The system should be, if not theoretically unbreakable, unbreakable in practice.
- The system design should not require secrecy, so compromise of the design should not damage the security of the system. This is commonly known as "don't do security by obscurity" (Kerckhoffs' principle).
- The key should be easily memorable without notes and should be easily changeable
- The cryptograms should be transmittable by telegraph
- The apparatus or documents should be portable and operable by a single person
- The system should be easy to operate, neither requiring a long list of rules nor involving mental strain

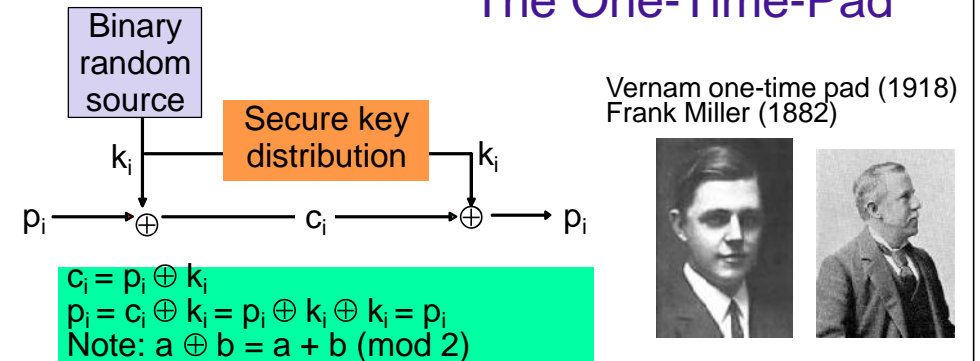
Enigma

- German WW II crypto machine
- Many different variants
- Follows Kerckhoffs' principles
- Analysed by Polish and English mathematicians



Broken by Alan Turing's «bombe» cryptanalysis machine during WW II.

A crypto system with perfect security: The One-Time-Pad



- Offers perfect security assuming the key is perfectly random, of same length as the message; and only used once. Proved by Claude E. Shannon in 1949.
- Problem: Very large keys required

The perfect cipher: One-Time-Pad

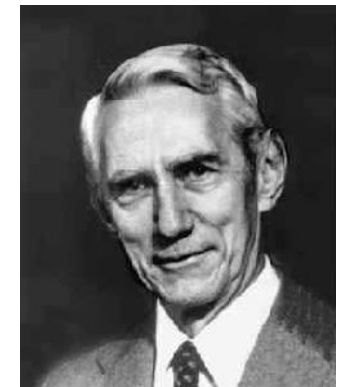


- Old versions used paper punch tape of random data
- Modern versions can use DVDs with Gbytes of random data

Claude Shannon (1916 – 2001)

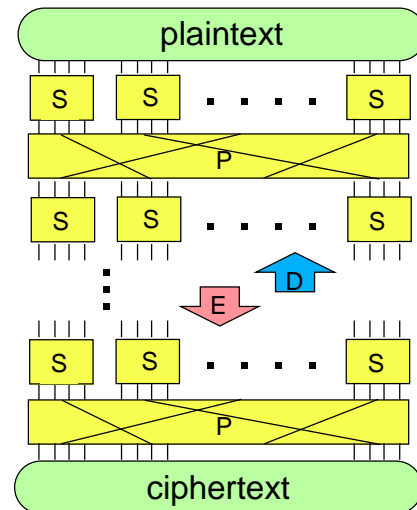
The Father of Information Theory – MIT / Bell Labs

- Information Theory
 - Defined the „binary digit“ (bit) as information unit
 - Definition of „entropy“ as a measure of information amount
- Cryptography
 - Model of a secrecy system
 - Definition of perfect secrecy
 - Designed S-P networks, i.e. a series of substitution & permutation functions



Shannon's S-P Network

- "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**

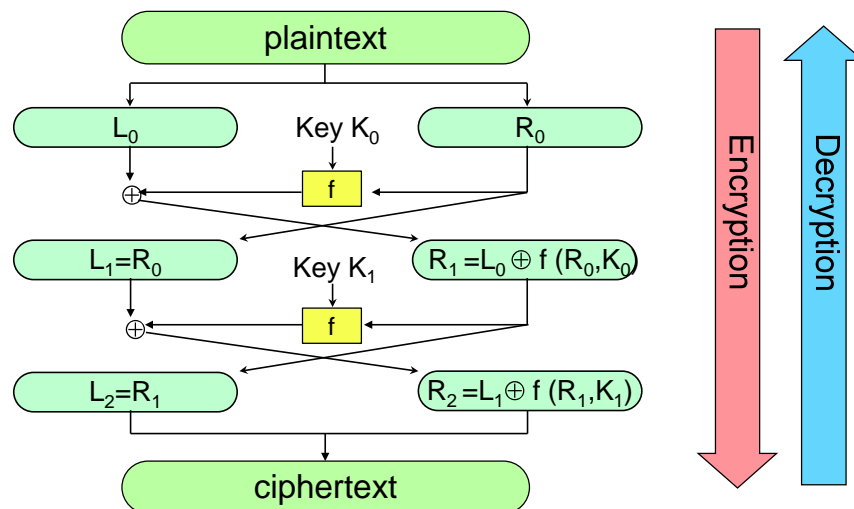


Horst Feistel's (1915 – 1990) and his revolutionary cipher design

- The **Feistel cipher** is a general and elegant architecture for designing ciphers according to S-P networks
- Split input text block in two halves
 - Perform S-P transformation on one half
 - XOR output with other half
 - Swop Halves
 - Repeat for multiple rounds
- Advantage: The S-P functions do **not** have to be invertible !!!



2-round Feistel Network (DES)



DES - Data Encryption Standard

- Published in 1977 by the US National Bureau of Standards for use in unclassified government applications with a 15 year life time.
- 16 round Feistel cipher with 64-bit data blocks, 56-bit keys.
- 56-bit keys sufficient in 1977; today exhaustive search on 56-bit keys only takes hours.
- DES was controversial because of classified design criteria, however no loop hole found.

AES - Advanced Encryption Standard

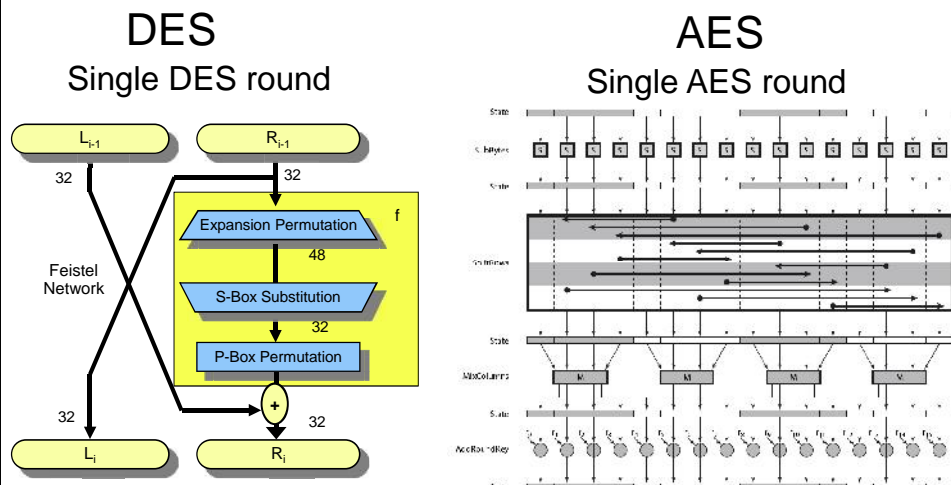
- Public competition to replace DES: because 56-bit keys and 64-bit data blocks no longer adequate.
- Rijndael nominated as the new Advanced Encryption Standard (AES) in 2001 [FIPS-197].
- Rijndael (pronounce as “Rhine-doll”) designed by Vincent Rijmen and Joan Daemen.
- 128-bit block size (Note error in Harris p. 809)
- 128-bit, 192-bit, and 256-bit key sizes.
- Rijndael is not based on a Feistel network.

Rijndael, the selected AES cipher

Designed by Vincent Rijmen and Joan Daemen from Belgium



Comparison DES – AES single round



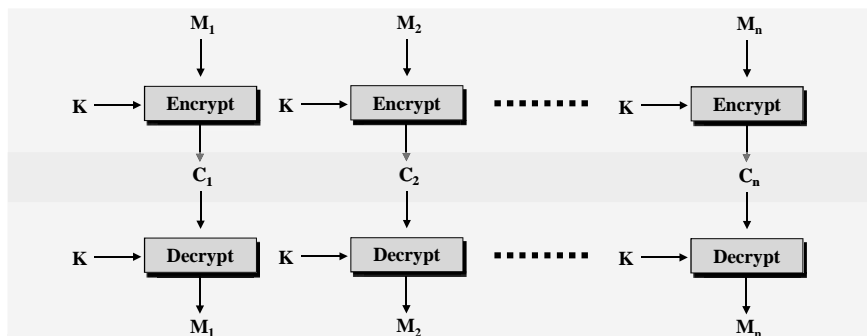
Block Ciphers: Modes of Operation

- 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**eedback (OFB)
 - **C**ipher **F**eedback (CFB)
 - **C**ounter **M**ode (CTR)

Electronic Code Book

• ECB Mode encryption

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



Vulnerability of ECB mode



Plaintext



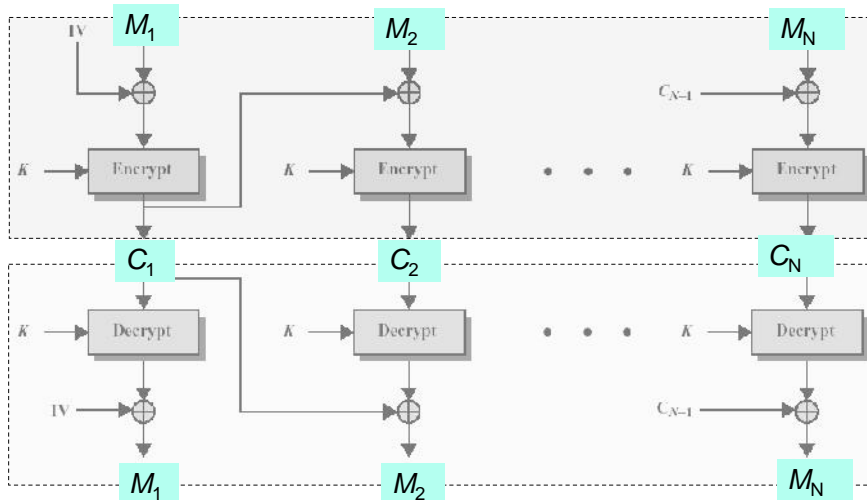
Ciphertext using
ECB mode



Ciphertext using
CBC mode

- ECB mode not normally used because equal plaintext blocks give equal ciphertext blocks, which is bad.
- CBC and CTR modes are often used instead because more secure and simple to use.

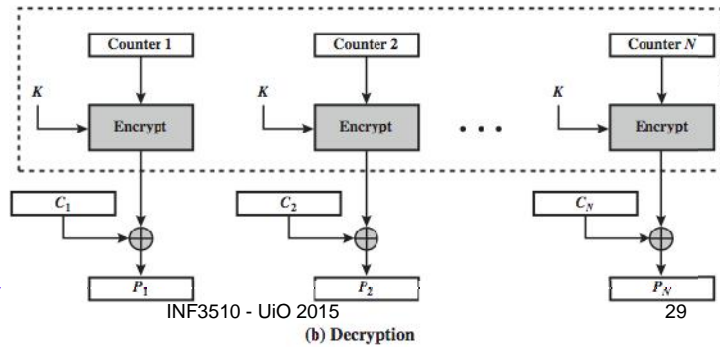
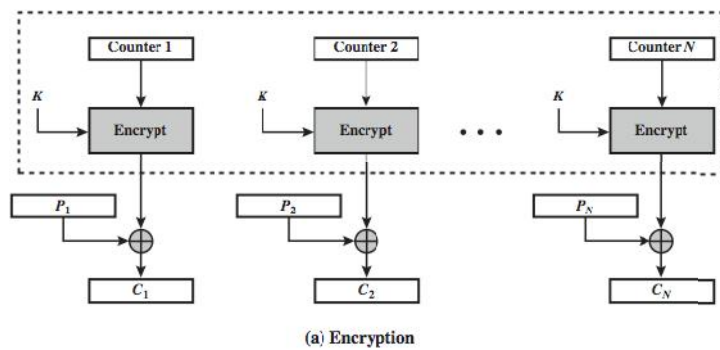
Cipher Block Chaining Mode



CBC Mode Issues

- Chaining guards against pre-fabricated code book
 - The same plaintext block encrypts to different ciphertext blocks each time.
- May assist in detecting integrity breaches
 - Such as the insertion, deletion or reordering of data blocks into the ciphertext.
- What happens when there is an error?
 - If there is a bitflip error (0 to 1 or vice versa) that block and the following block will be decrypted incorrectly
 - If a ciphertext bit, or even a character is inserted or deleted this will be detected because of the incorrect ciphertext length
 - Not multiples of block size
 - Inserting or deleting a block will cause incorrect decryption

CTR Counter Mode



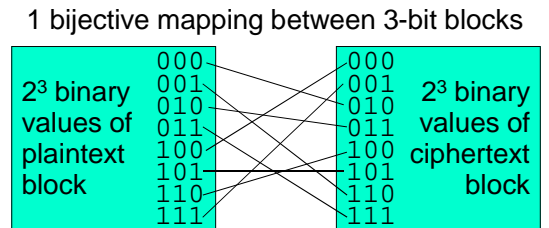
Advantages and Limitations of CTR

- Efficiency
 - can do parallel encryptions in h/w or s/w
 - can preprocess in advance of need
 - good for bursty high speed links
 - good for HD encryption
- Random access to encrypted data blocks
- Provable security (good as other modes)
- But must ensure never reuse key/counter values, otherwise could break

Block cipher: Applications

- Block ciphers are often used for providing **confidentiality services**
- They are used for applications involving processing large volumes of data, where long time delays can not be tolerated.
 - Examples:
 - Computer files
 - Databases
 - Email messages
- Block ciphers can also be used to provide **integrity services**, i.e. for message authentication

Block ciphers give bijective mappings between binary values



- Given block size $m = 64$ bit and key length $l = 64$ bit
- Number of different cipher bijective mappings determined by key is: $2^{64} = 18446744073709551616$
- Number of possible block bijective mappings of 2^{64} binary values is: $2^{64}! = ??$ (more than 2^{71} decimal digits)

Set of all possible bijective mappings from cleartext block to ciphertext block

Set of bijective mappings determined by key

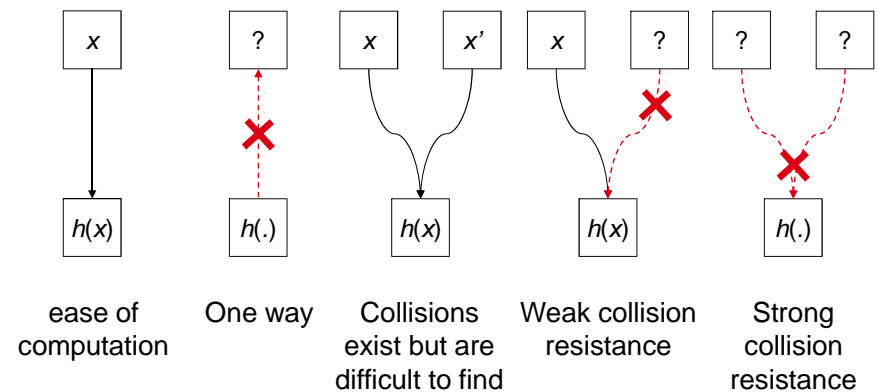
- Block ciphers only use a tiny fraction of possible mappings
- One-Time-Pad ciphers can potentially use all mappings

Integrity Check Functions

- Hash functions
- MAC functions



Properties of hash functions



Current hash functions

- **MD-2 (1989) and MD5 (1991)**: 128 bit digest. Broken, too short digest. Not recommended, but used in legacy applications.
- **SHA-1 (Secure Hash Algorithm)**: 160 bit digest. Potential attacks exist. Designed by NSA in 1995 to operate with DSA (Digital Signature Standard). Not recommended, but 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**: NIST announced competition for new algorithm (2007)
- 2012: Keccak selected as official SHA-3 algorithm in Oct. 2012
- SHA-3 has same designer as for AES: Joan Daemen + others.
- 2014 draft standard SHA 3 published, still in draft in 2015.
 - no need/hurry to replace SHA-2.

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.

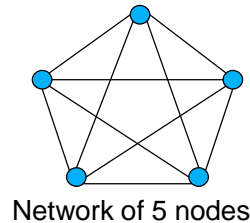
MAC and MAC algorithms

- MAC means (at least) two things:
 1. The computed message authentication code $h(M, k)$
 2. General name for algorithms used to compute a MAC
- In practice, the MAC algorithm is e.g.
 - HMAC (Hash-based MAC algorithm)
 - CBC-MAC (CBC based MAC algorithm)
 - CMAC (Cipher-based MAC algorithm)
- MAC algorithms, 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 big problem.
- Is there a better way?



James H. Ellis (1924 – 1997)

- 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 –)

- British mathematician and cryptographer
- Silver medal at the International Mathematical Olympiad, 1968
- Works at GCHQ
- 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



Malcolm J. Williamson

- 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-Hellman key exchange algorithm



Public Key Encryption

- Proposed in the open literature by Diffie & Hellman in 1976.
- Each party has a **public encryption key** and a **private decryption key**.
- Reduces total number of exchanged keys to n
- Computing the private key from the public key should be computationally infeasible.
- The public key need not be kept secret but it is not necessarily known to everyone.
- There can be applications where even access to public keys is restricted.

Ralph Merkle, Martin Hellman and Whitfield Diffie

- Merkle invented (1974) and published (1978) Merkle's puzzle, a key exchange protocol which was unpractical
- Diffie & Hellman invented (influenced by Merkle) a practical key exchange algorithm using discrete logarithm.
- D&H defined public-key encryption (equiv. to non-secret encryption)
- Defined digital signature
- "New directions in cryptography" (1976)



Diffie-Hellman key agreement (key exchange) (provides no authentication)

Alice picks random integer a



Alice computes the shared secret

$$(g^b)^a = g^{ab} \text{ mod } p$$

$$g^a \text{ mod } p$$

$$g^b \text{ mod } p$$

Bob picks random integer b



Bob computes the same secret

$$(g^a)^b = g^{ab} \text{ mod } p.$$

Diffie-Hellman Applications

- **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**

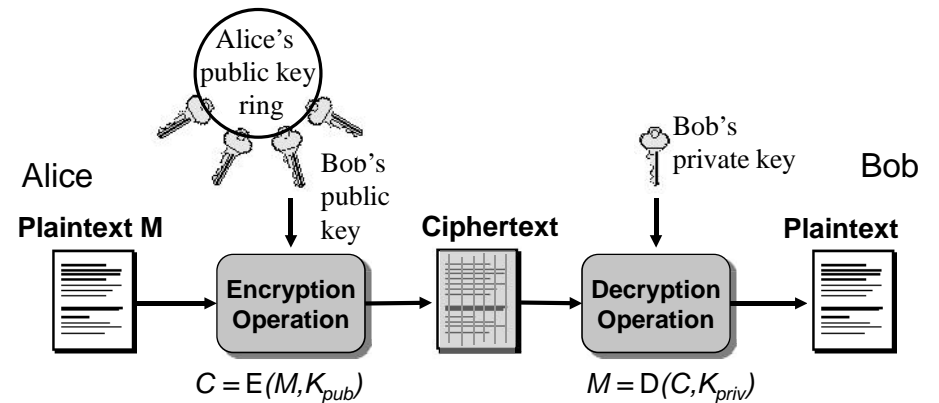
RSA Algorithm

- $n = pq$ which is made public (but not p and q)
- Calculate secret: $z = (p-1)(q-1)$
- Choose a public key e
- Compute private key d such that $ed = 1 \text{ mod}(z)$
- Encryption of message m where $(1 < m < n)$.
 - Compute: $c = m^e \text{ mod } n$
- Decryption of ciphertext c
 - Compute: $m = c^d \text{ mod } n$
- Security depends on the difficulty of factorizing n
 - so the prime factors p and q must be LARGE

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

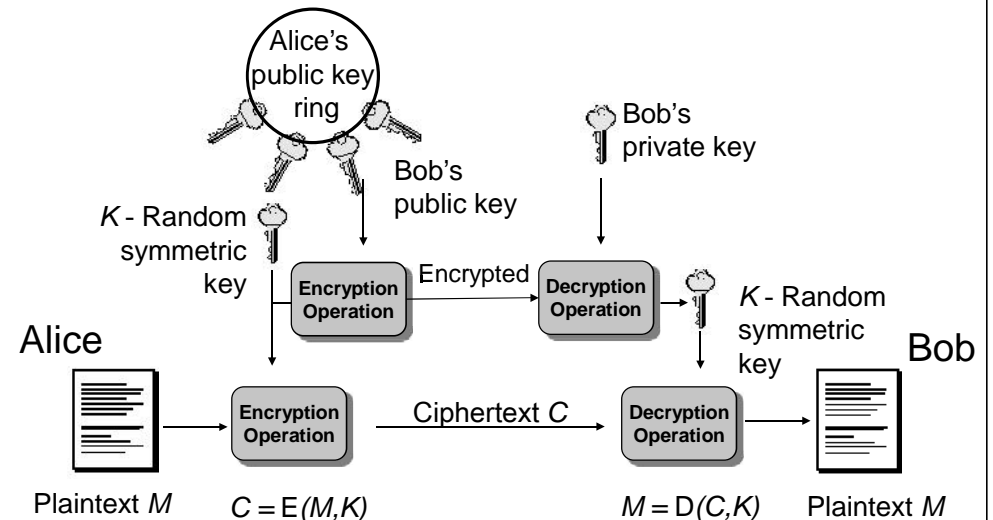


- In practical application, large messages are not encrypted directly with asymmetric algorithms. Hybrid systems are used.

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

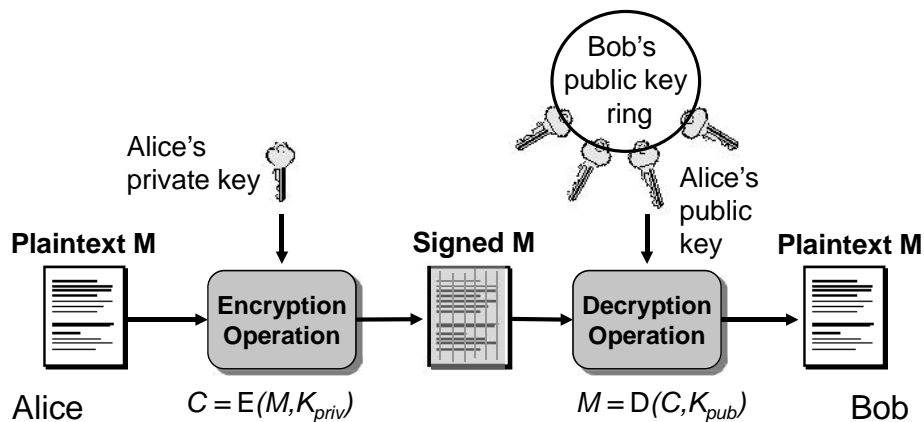


Digital Signatures

Digital Signature Mechanisms

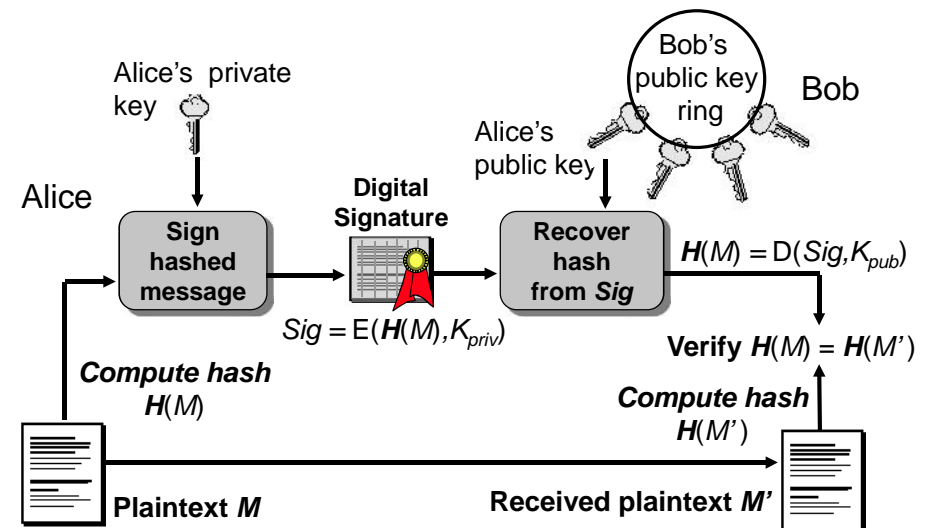
- A MAC cannot be used as evidence that should be verified by a third party.
- Digital signatures used for non-repudiation, data origin authentication and data integrity services, and in some authentication exchange mechanisms.
- Digital signature mechanisms have three components:
 - key generation
 - 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



Problems for digital signatures

- Digital signatures depend totally on PKIs.
 - Reliable PKIs are hard to set up and operate.
- WYSIWYS (*What You See Is What You Sign*) means that the semantic content of signed messages can not be changed by accident or intent.
 - WYSIWYS is essential but very difficult to guarantee.
- Revoking certificates invalidates digital signatures.
 - Repudiate a signature by claiming theft of private key
- Key decay and algorithm erosion limits life time of digital signatures.
 - Future computers can falsify old signatures

Difference between MACs & Dig. Sig.

- MACs and digital signatures are both authentication mechanisms.
-  MAC: the verifier needs the secret that was used to compute the MAC; thus a MAC is unsuitable as evidence with a third party.
 - The third party does not have the secret.
 - The third party cannot distinguish between the parties knowing the secret.
-  Digital signatures can be validated by third parties, and can in theory thereby support both non-repudiation and authentication.

Key length comparison:

Symmetric and Asymmetric ciphers offering comparable security

AES Key Size	RSA Key Size	Elliptic curve Key Size
-	1024	163
128	3072	256
192	7680	384
256	15360	512

End of lecture