

INF3510 Information Security

University of Oslo

Spring 2012

Lecture 5

Cryptography



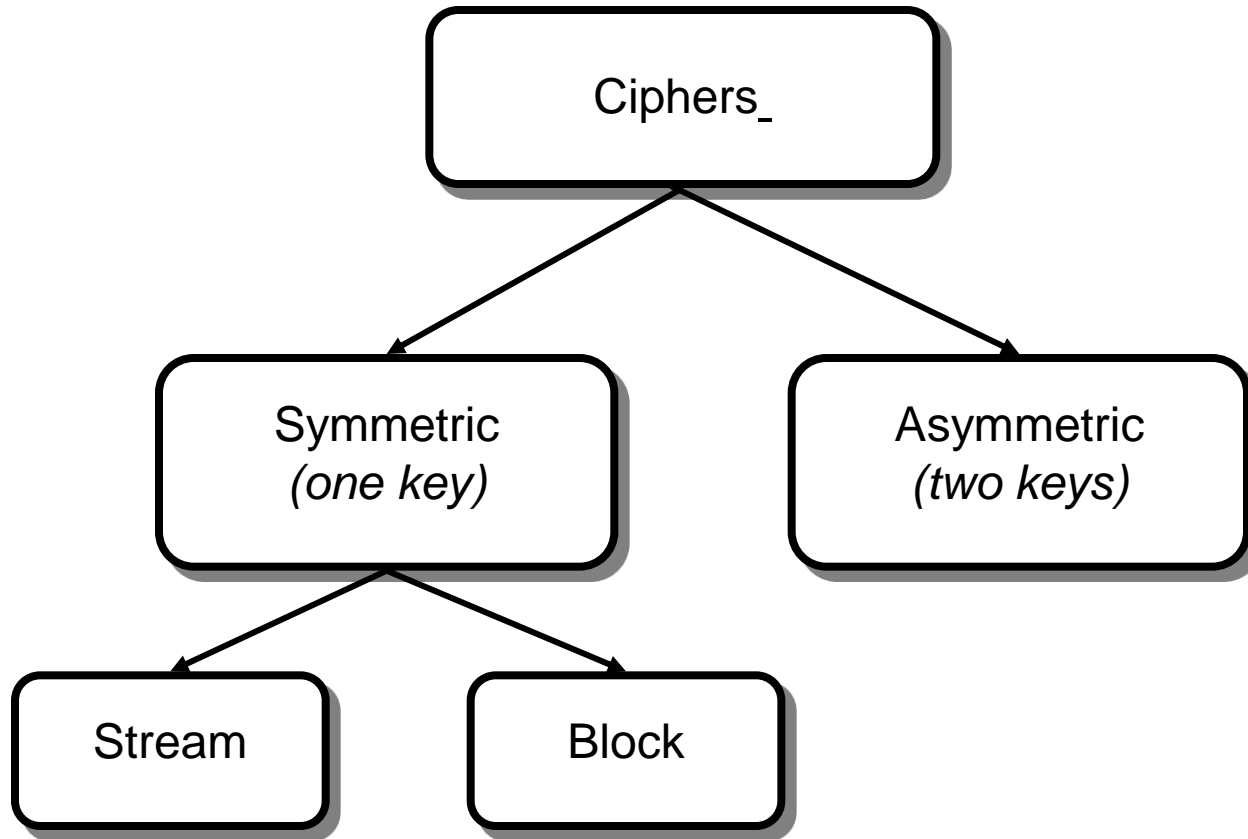
When is cryptography used?

- 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
 - **Message Authentication:**
 - So you can be sure that the message is from the place or sender it claims to be from

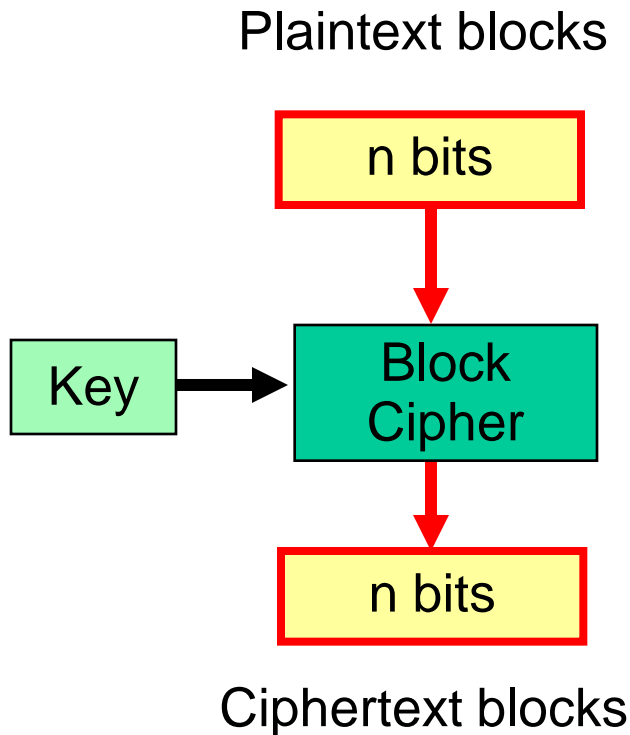
When is cryptography used?

- Some example situations:
 - **Historically**, the military and spy agencies were the main users of cryptology
 - Situation: transmitting messages over insecure channels
 - **Now**, it is used in many other areas, especially in electronic information processing and communications technologies:
 - **Banking**: your financial transactions, such as EFTPOS
 - **Communications**: your mobile phone conversations
 - **Info stored in databases**: hospitals, universities, etc.
- Cryptography can be used to protect information in storage or during transmission

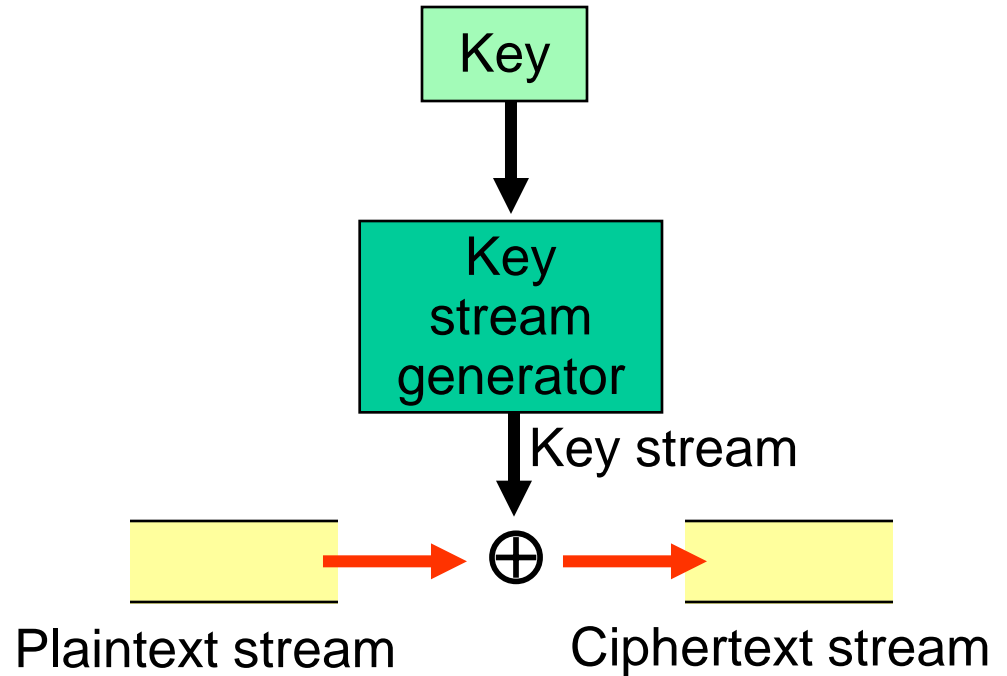
Taxonomy of modern ciphers



Block Cipher vs. Stream Cipher



Block cipher

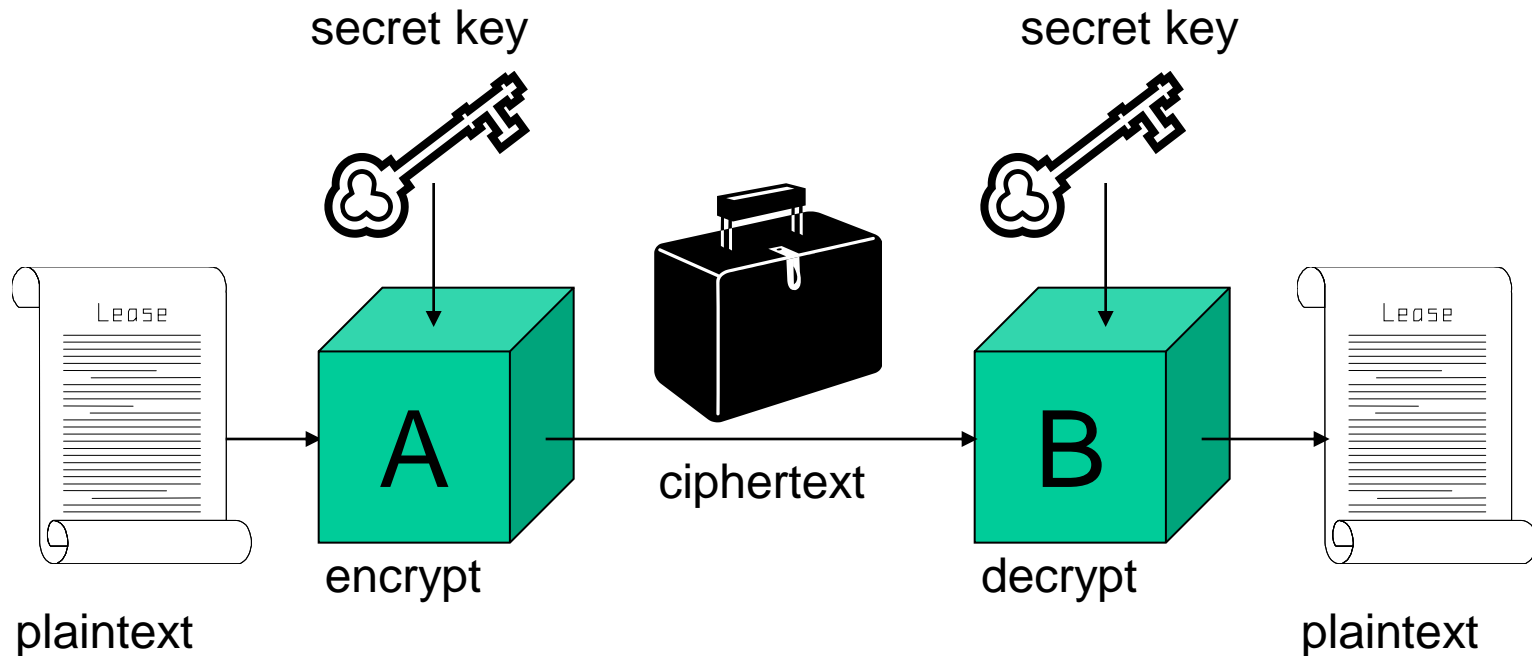


Stream cipher

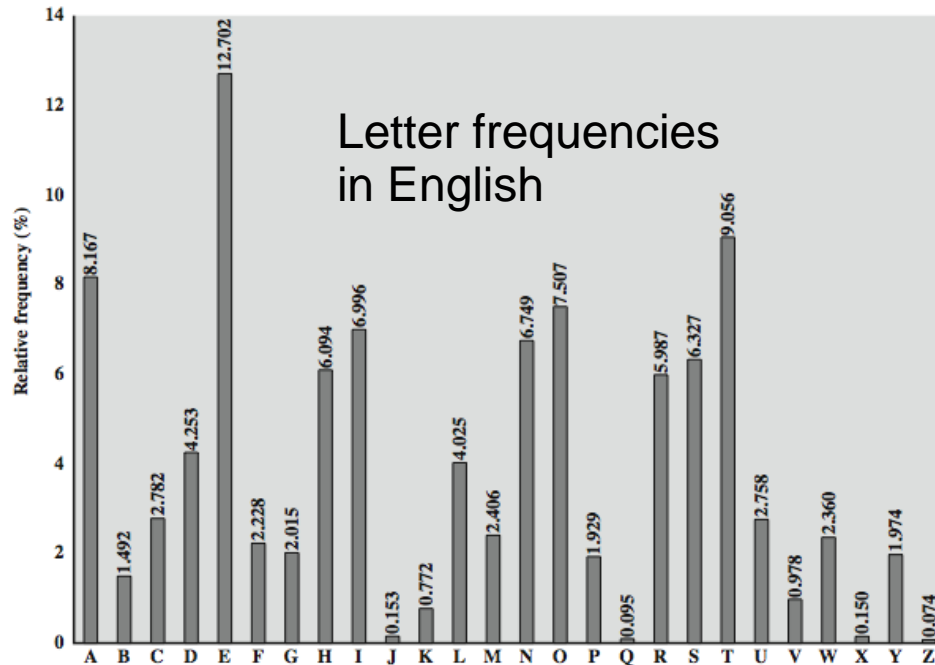
Terminology

- **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**: 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 Key Encryption



Letter Frequencies → statistical attacks

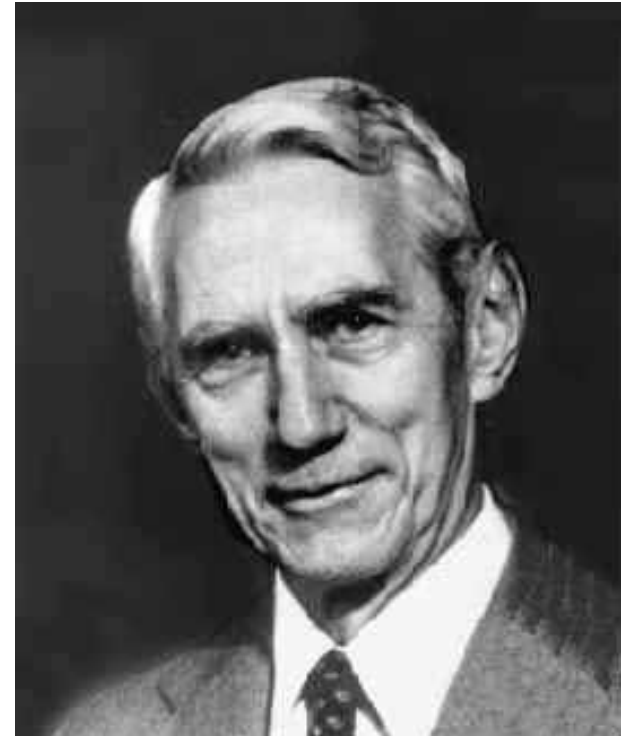


- Encryption must hide statistical patterns in data
- Achieved with a series of primitive functions

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

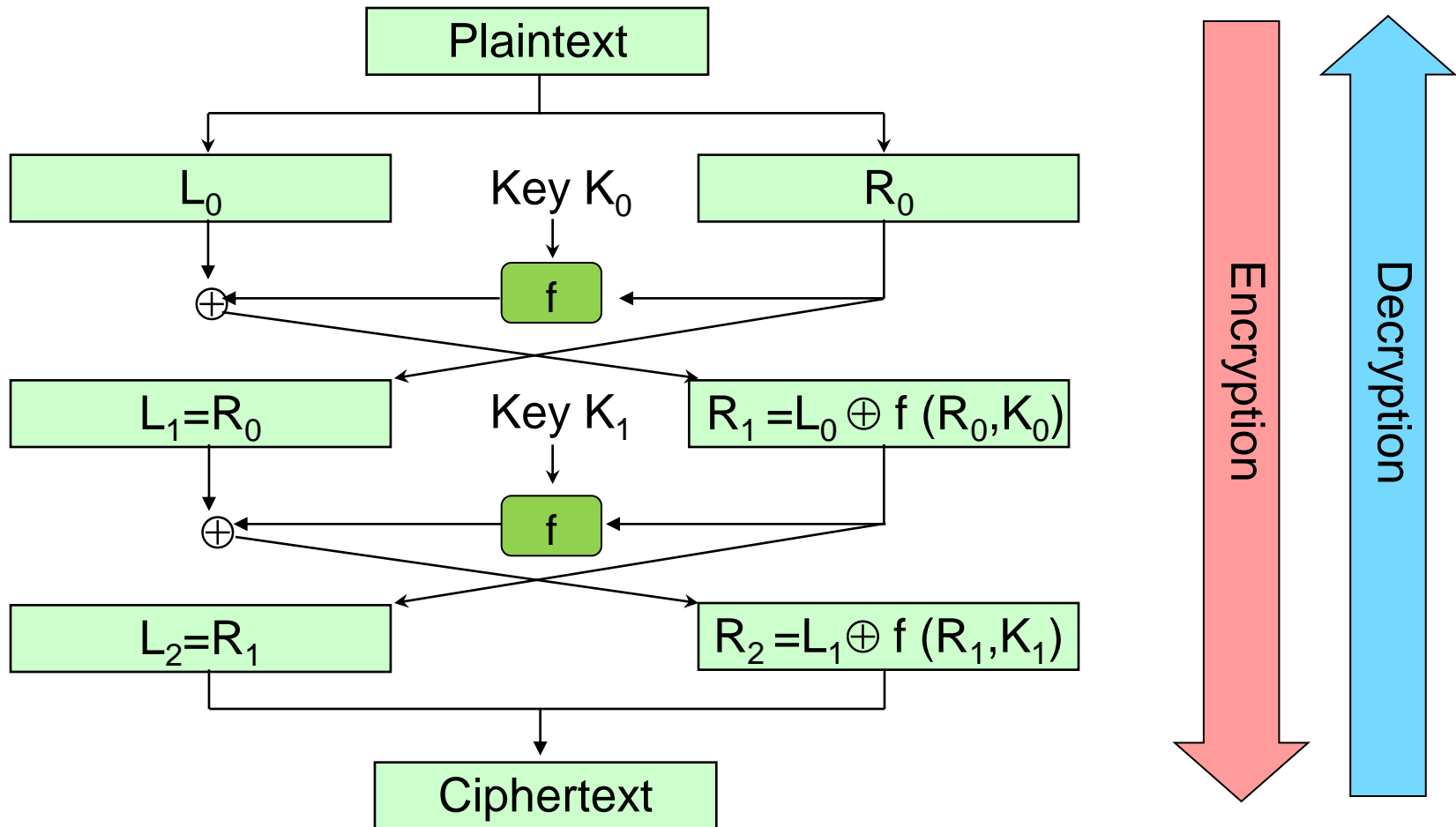


Horst Feistel's (1915 – 1990) and his revolutionary cipher design

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



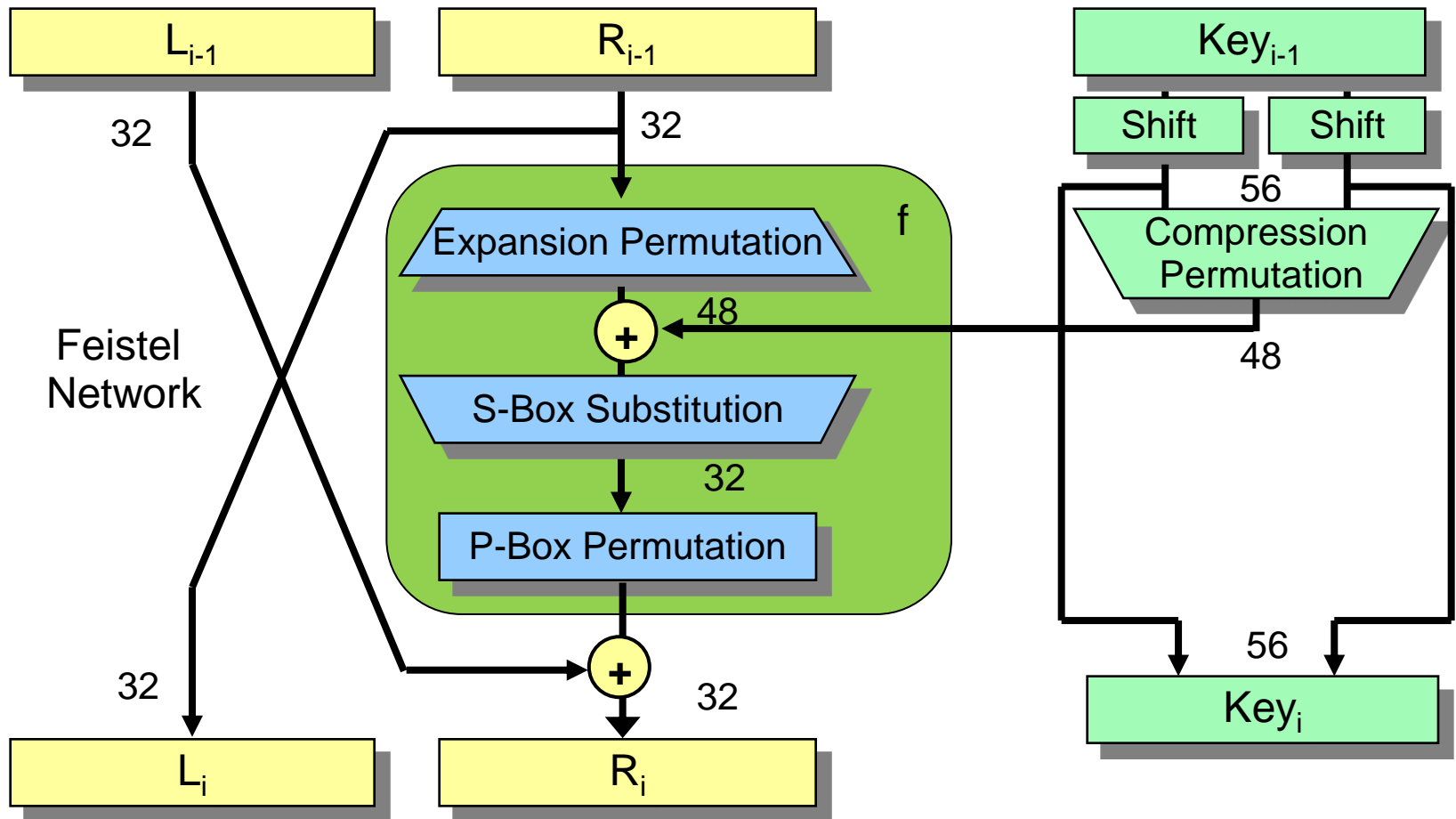
2-round Feistel Network



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 were controversial in 1977; today, exhaustive search on 56-bit keys is very feasible.
- Controversial because of classified design criteria, however no loop hole was ever found.

One Round of DES



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
- 128-bit, 196-bit, and 256-bit key sizes.
- Rijndael is not a Feistel cipher.

Advanced Encryption Standard (AES) Contest (1997-2001)

January 1997

Call for cipher proposals

Key sizes: 128, 192 or 256 bit, **block size:** 128 bits

June 1998

15 Candidates

from USA, Canada, Belgium,
France, Germany, Norway, UK, Israel,
Korea, Japan, Australia, Costa Rica

Stage 1

**Assess security,
software efficiency,
& flexibility**

August 1999

5 final candidates

Mars, RC6, Rijndael, Serpent, Twofish

Stage 2

**Assess security,
& hardware efficiency**

October 2000

1 winner: Rijndael Belgium

November 2001

AES FIPS PUB 197 standard

Rijndael, the selected AES cipher

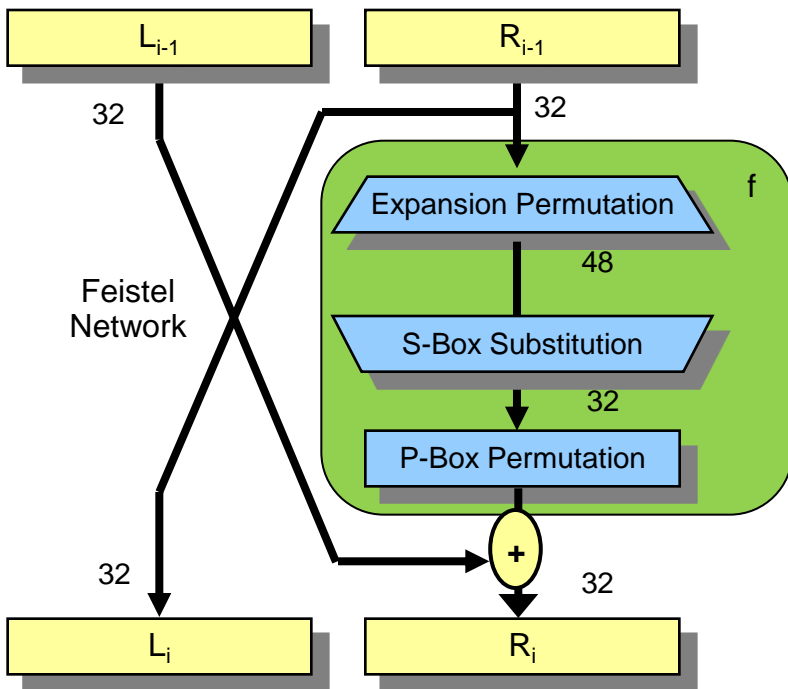
Designed by Vincent Rijmen and Joan Daemen from Belgium



Comparison DES – AES single round

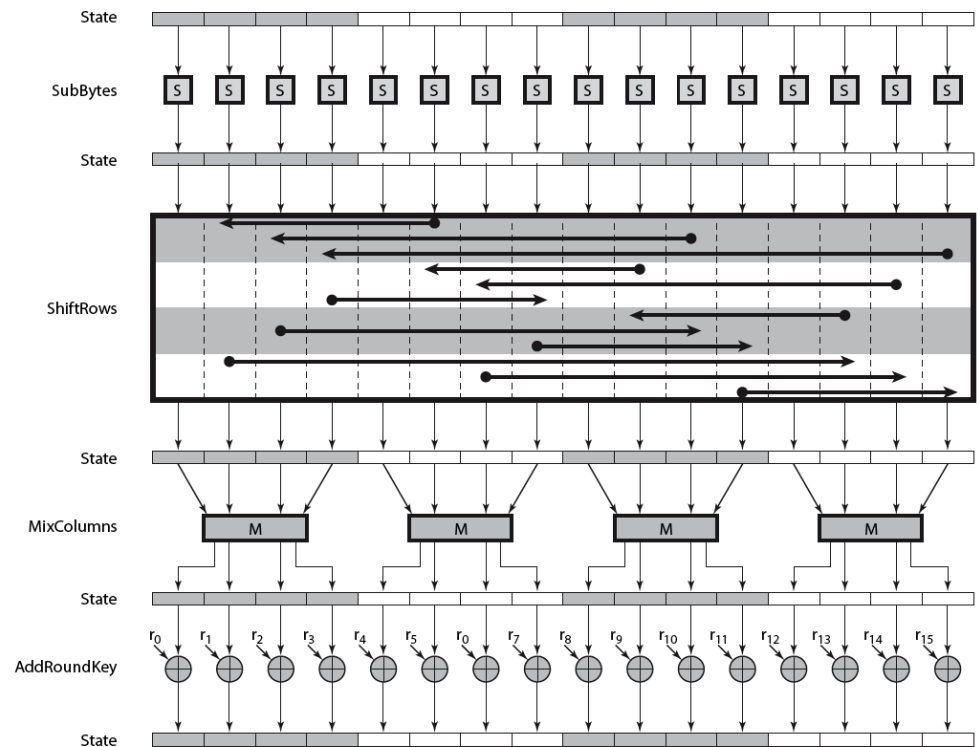
DES

DES Round



AES

AES Round



Using encryption for real

- With a block cipher, encrypting a n -bit block M with a key k gives a ciphertext block $C = E(M, k)$.
- Given a well designed block cipher, observing C would tell an adversary nothing about M or k .
- What happens if the adversary observes traffic over a longer period of time?
 - The adversary can detect if the same message had been sent before; if there are only two likely messages “buy” and “sell” it may be possible to guess the plaintext without breaking the cipher.

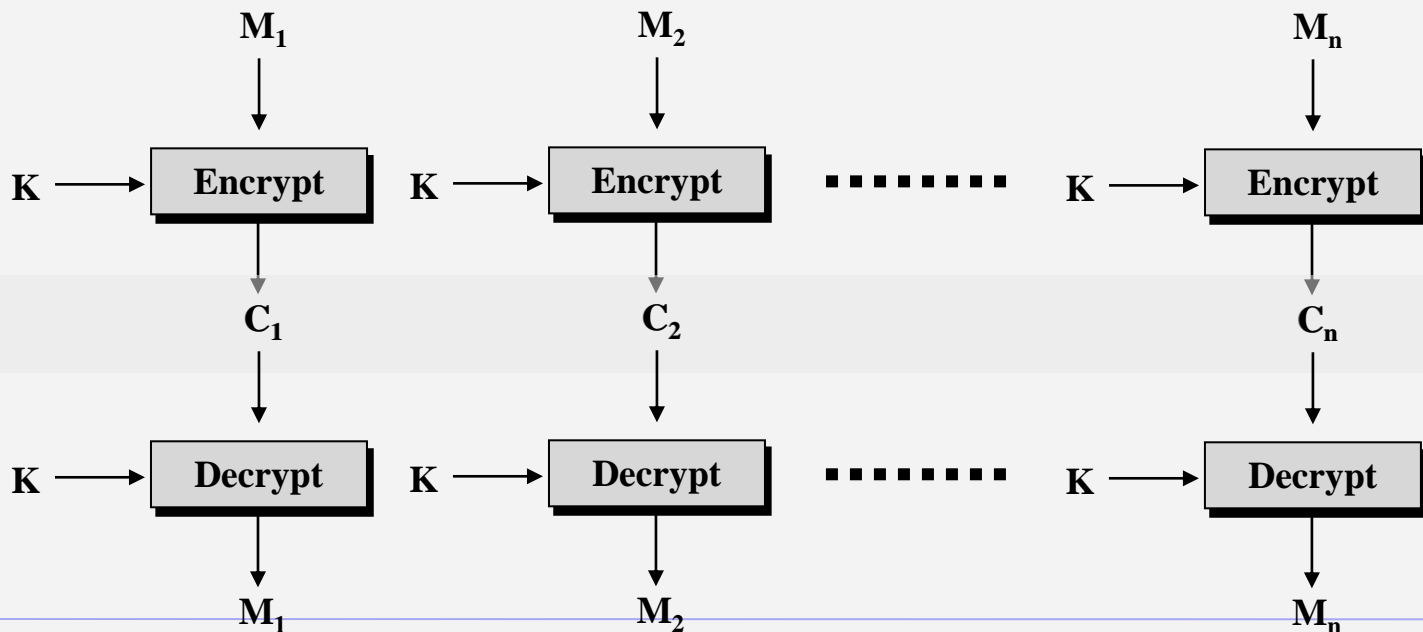
Block Ciphers: Modes of Operation

- Block ciphers can be used in different modes in order to provide different security services.
- Common modes include:
 - **E**lectronic **C**ode **B**ook (ECB)
 - **C**ipher **B**lock **C**haining (CBC)
 - **O**utput **F**eedback (OFB)
 - **C**ipher **F**eedback (CFB)
 - **C**ounter Mode (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

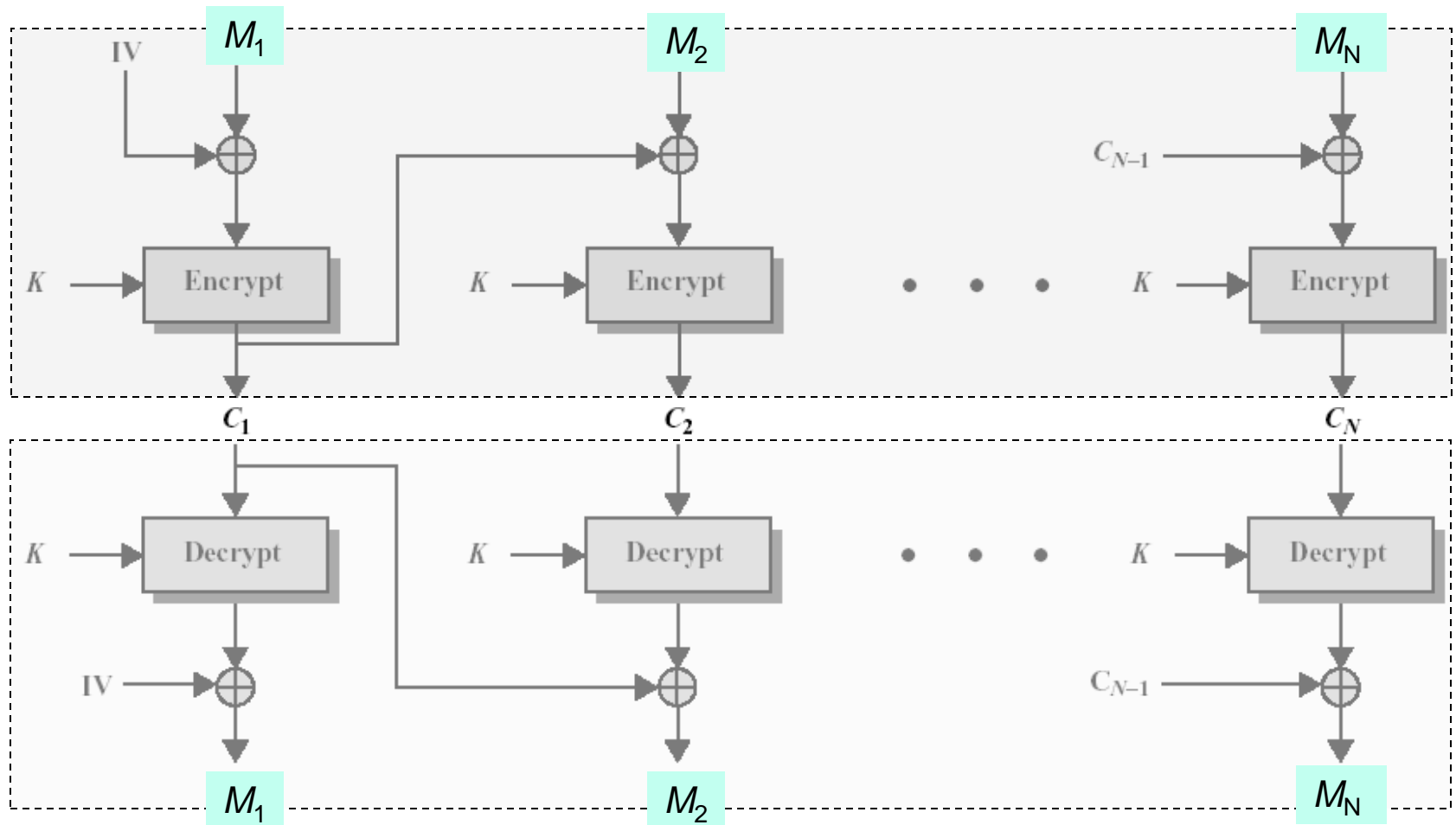


ECB Mode

- **ECB Mode Issues**

- Problem: For a given key, the same plaintext block always encrypts to the same ciphertext block.
 - This may allow an attacker to construct a code book of known plaintext/ciphertext blocks.
 - The attacker could use this codebook to insert, delete, reorder or replay data blocks within the data stream without detection
- Other modes of operation can prevent this, by not encrypting blocks independently
 - For example, using the output of one block encryption as input to the next (chaining)

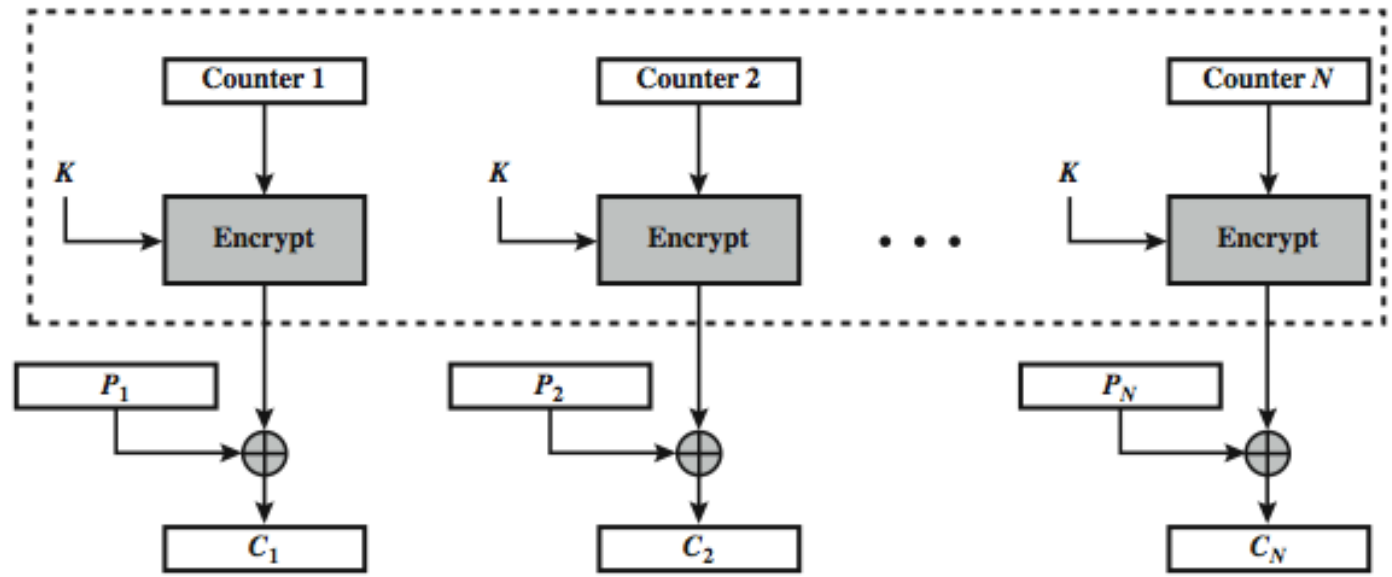
Cipher Block Chaining Mode



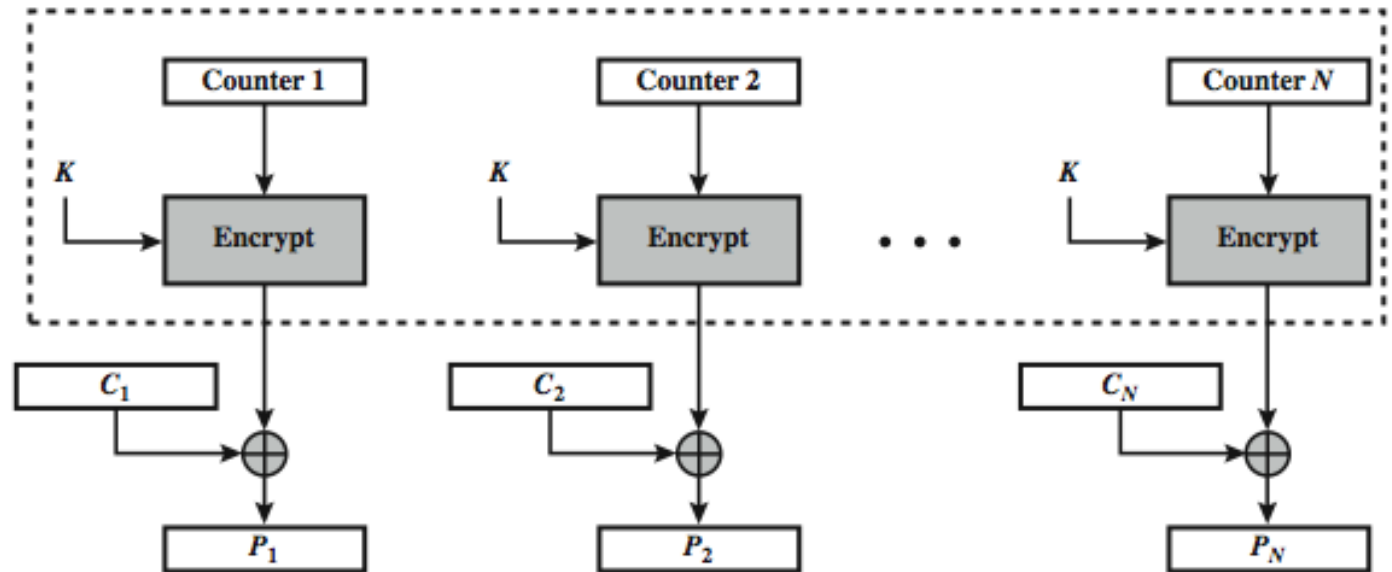
CBC Mode

- **CBC Mode Issues**
 - Chaining guards against the construction of a 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



(a) Encryption



(b) Decryption

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

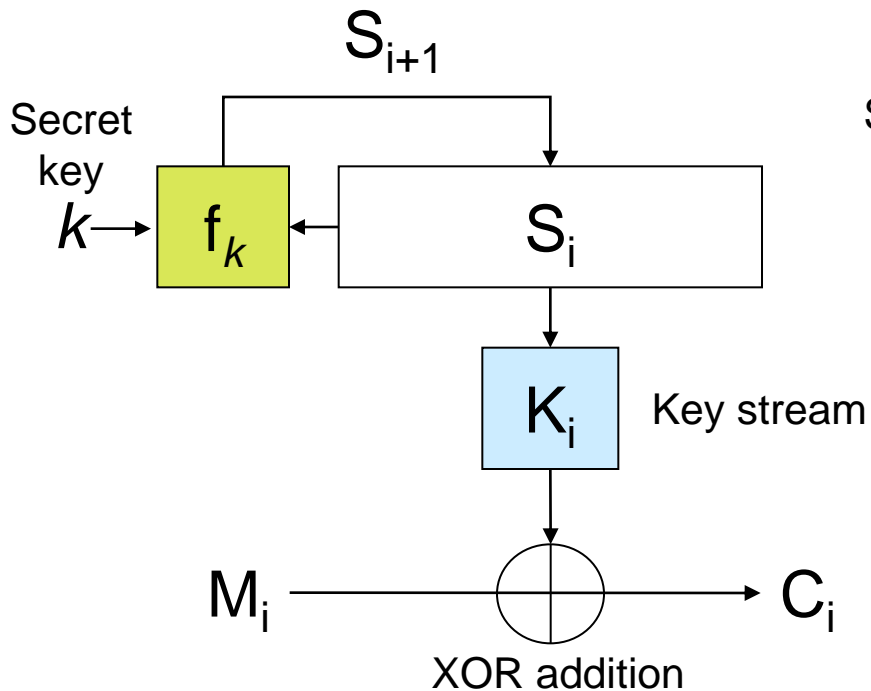
- Block ciphers are often used for providing **confidentiality services**
- They are used for applications involving processing large volumes of data, where time delays are not critical.
 - Examples:
 - Computer files
 - Databases
 - Email messages
- Block ciphers can also be used to provide **integrity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

Stream Ciphers

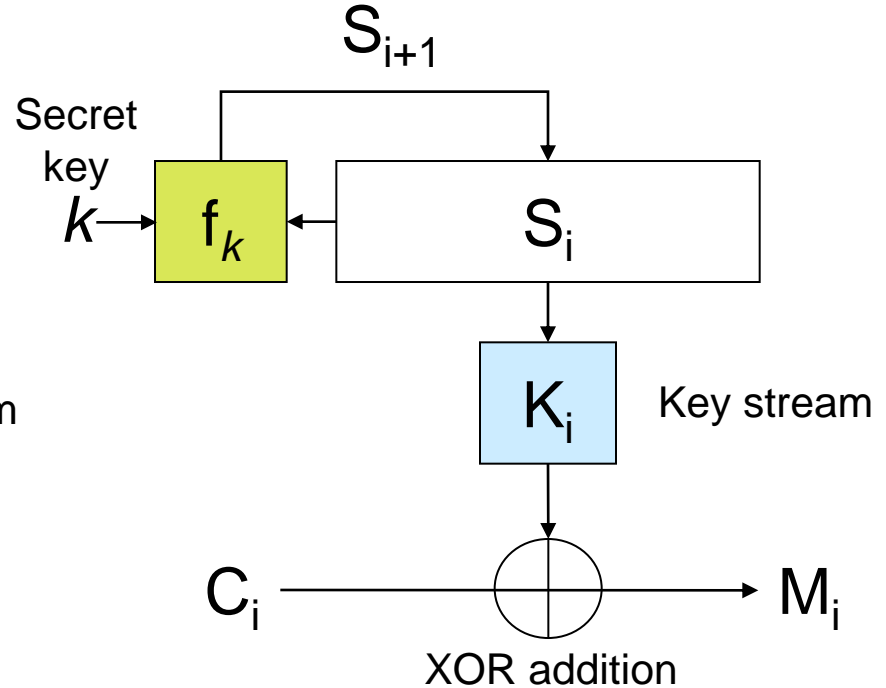
- Consist of a key stream generator and a function for combining key stream and data.
- The combining function tends to be simple, XOR (exclusive-OR) is a typical example.
- The key stream generator takes as its input a key k seed S_0 and updates its state with a state transition function f_k , $S_{i+1} = f_k(S_i)$.
- The output at step i is the bitstream key K_i derived from S_i

Stream Ciphers

Encryption



Decryption



Encryption and decryption are usually identical operations.

Stream Ciphers

- In such a cipher, a bit error in ciphertext bit i causes a single bit error in plaintext bit i .
- Wireless networks use stream ciphers to protect **data confidentiality**.
- An adversary can make precise relative changes to the plaintext by modifying the corresponding ciphertext bits.
- Stream ciphers therefore cannot be used for **integrity** protection.

Cryptanalysis, by what the attacker knows

The attacker's goal is to discover the secret key.

- *Ciphertext-only*: attacker has access only to a collection of ciphertexts.
- *Known-plaintext*: attacker has a set of ciphertexts to which he knows the corresponding plaintexts.
- *Chosen-plaintext (chosen-ciphertext)*: attacker can obtain the ciphertexts (plaintexts) corresponding to an arbitrary set of plaintexts (ciphertexts) of his own choosing.
- *Related-key attack*: attacker has ciphertexts encrypted under different unknown keys with known relationship, e.g, keys that differ by 1 bit.

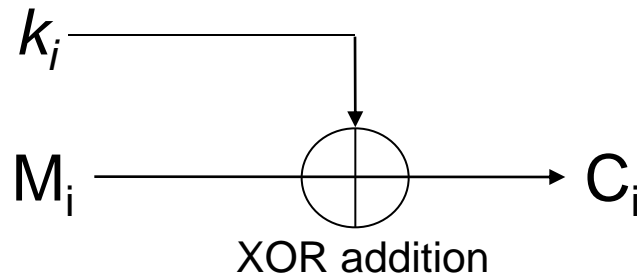
Is there a 'perfect' cipher?

- Yes - if you require **confidentiality**, the **One Time Pad** is provably secure.
- But we don't use it much due to its disadvantages.
- Disadvantages of the one-time pad cipher are:
 - each key can only be used once
 - each key is typically very large (lots of data)
 - requires secure distribution of large key (lots of data)
- Basically, key management of OTP is difficult

One-Time Pad Cipher

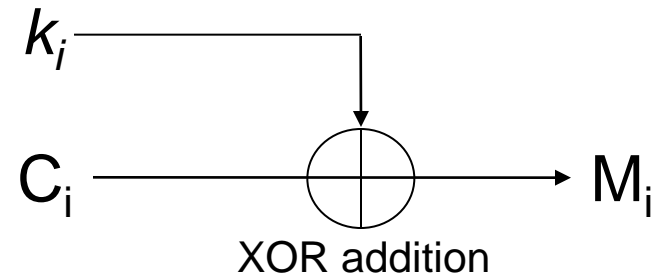
Encryption

One-time pad
key bit stream



Decryption

One-time pad
key bit stream



- Encryption and decryption are identical operations.
- $|k| = |M|$
- Each key k must never be reused

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

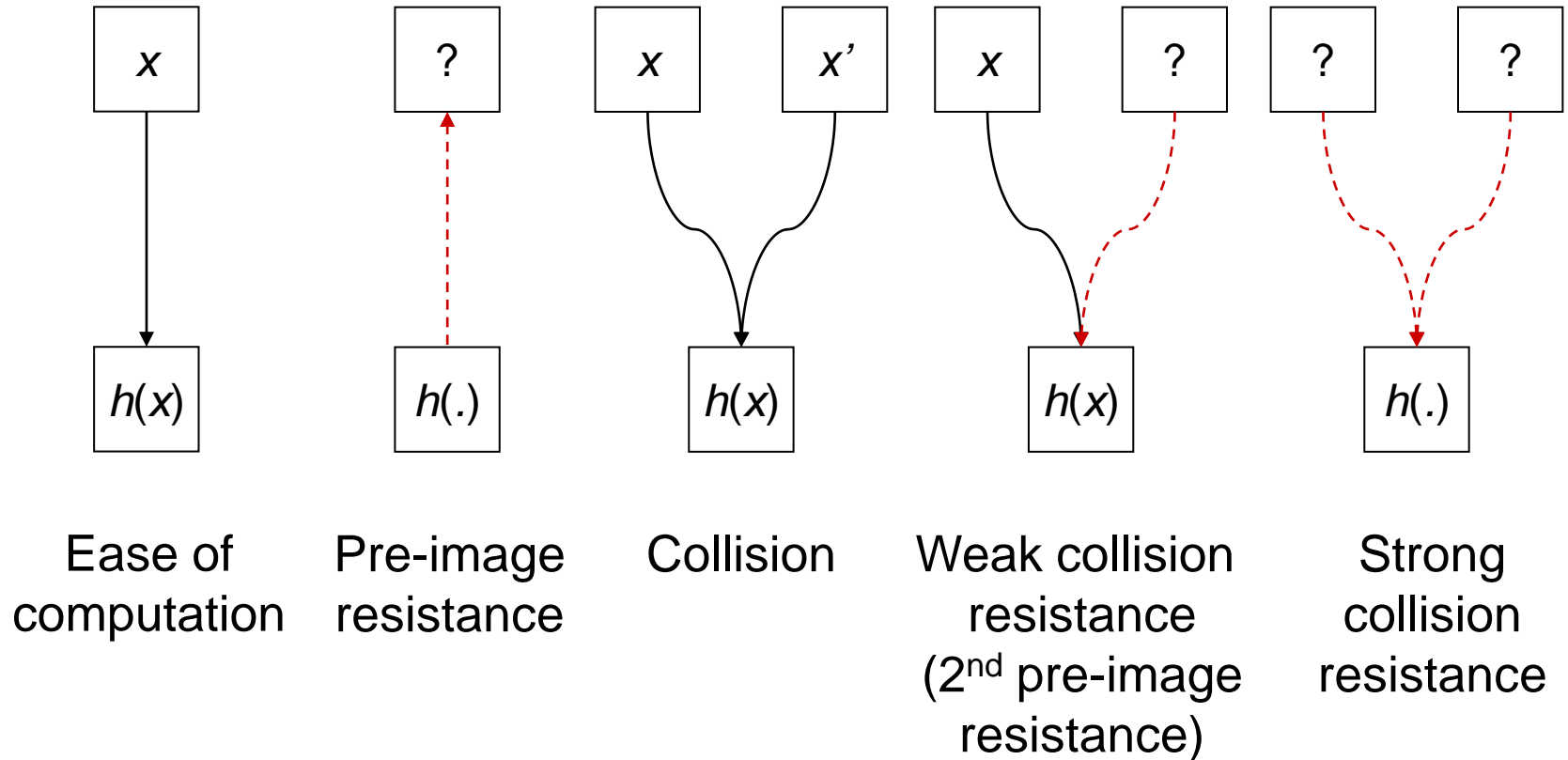
Integrity Check Functions

Hash functions (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



Frequently used hash functions

- MD5: 128 bit digest. Broken. Often used in Internet protocols but no longer recommended.
- SHA-1 (Secure Hash Algorithm): 160 bit digest. Potential attacks exist. Designed to operate with the US Digital Signature Standard (DSA);
- SHA-256, 384, 512 bit digest. Still secure. Replacement for SHA-1
- RIPEMD-160: 160 bit digest. Still secure. Hash function frequently used by European cryptographic service providers.
- NIST competition for new secure hash algorithm, announcement of winner expected in 2012.

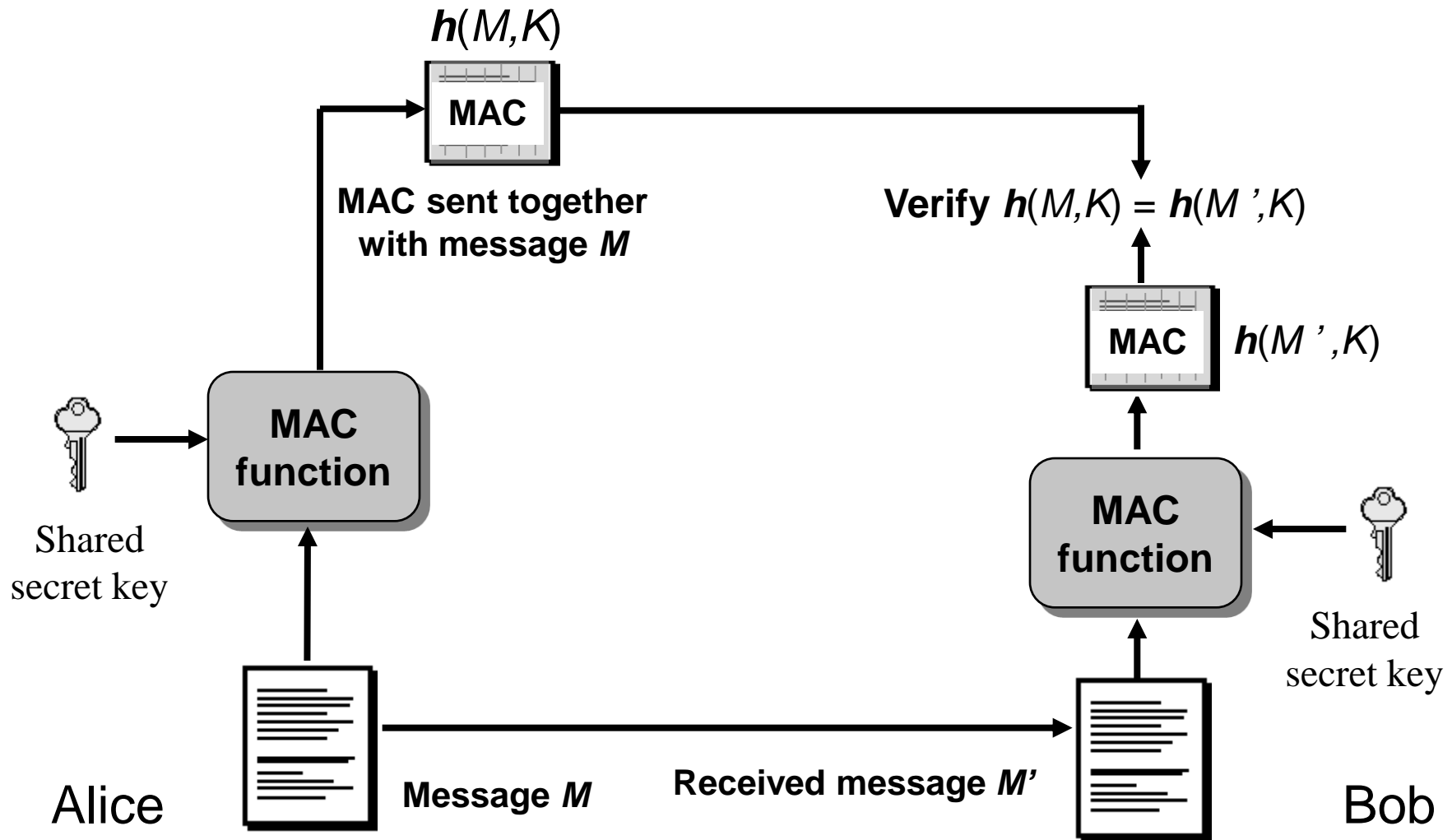
Message Authentication Codes

- 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 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.

Practical message integrity with MAC



Security of hash functions



- Large block size necessary to resist birthday
- Birthday paradox:
 - A group of 253 persons is needed to have $p = 0.5$ that any person has birthday on a specific date. Seems reasonable.
 - A group of only 23 persons is needed to have $p = 0.5$ that any two persons have birthday on the same date. Seems strange.
- Finding any two hashes that are equal (collision) in a large table of hash values is therefore relatively easy.
- A block size of n bits is considered to provide only $n/2$ bit complexity.
- To provide strong collision resistance, large blocks are needed. 160 bit hash block is currently a minimum.

Hash functions and Message Authentication

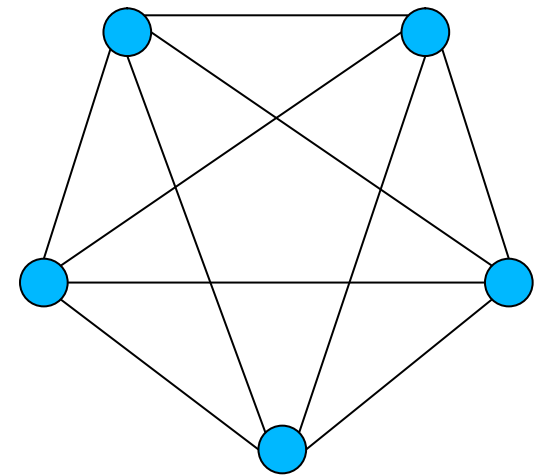
- Shared secret key is used with a MAC
- When used during message transmission, this provides **Message Authentication**:
 - A correct MAC value confirms the sender of the message is in possession of the shared secret key
 - Hence, much like a password, it confirms the authenticity of the message sender to the receiver.
- Indeed, message integrity is meaningless without knowing who sent the message.

Public-Key Cryptography



Symmetric key distribution

- Shared key between each pair
- In network of n users, each participant needs $n-1$ keys.
- Total number of exchanged keys:
 $= (n-1) + (n-2) + \dots + 2 + 1$
 $= n(n-1)/2$
- Grows exponentially, which is problematic.
- Is there a better way?



Network of 5 nodes

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-Hellmann 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 exponentiation.



Merkle, Hellman and Diffie

- D&H defined public-key encryption (equiv. to non-secret encryption)
- Defined digital signature
- Published 1976 in "*New directions in cryptography*"

Diffie-Hellman key agreement (key exchange)

(provides no authentication)

Alice picks random integer a



$$g^a \bmod p$$



Bob picks random integer b

$$g^b \bmod p$$



Computationally impossible to compute discrete logarithm



Alice computes the shared secret

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

Bob computes the same secret

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

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

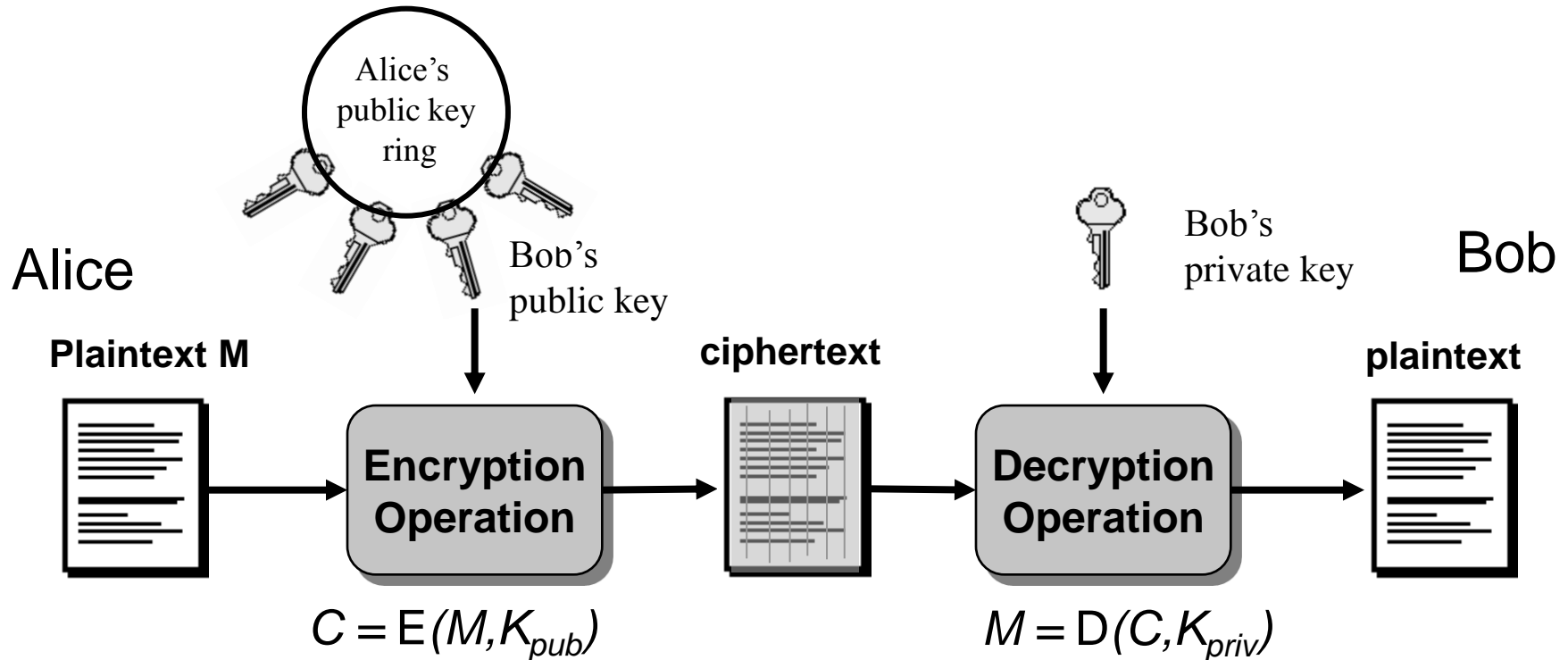
RSA Algorithm

- $n = pq$ 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 $ed = 1 \pmod{z}$
- Encryption of message m where $(1 < m < n)$.
 - Compute: $c = m^e \pmod{n}$
- Decryption of ciphertext c
 - Compute: $m = c^d \pmod{n}$
- 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

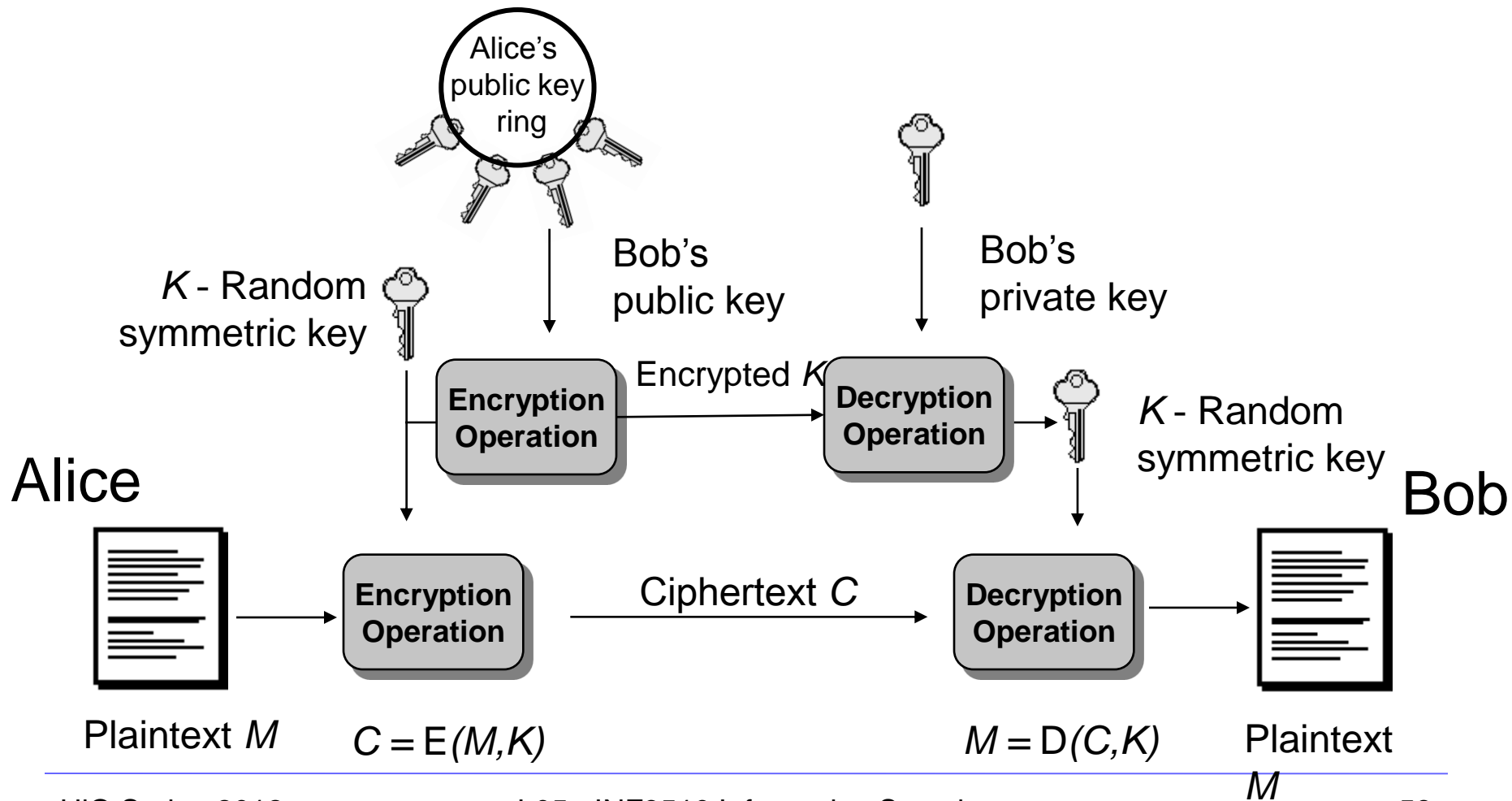


- In practice, large messages are not encrypted directly with asymmetric algorithms. Hybrid systems are used, where only symmetric session key is encrypted with asymmetric alg.

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

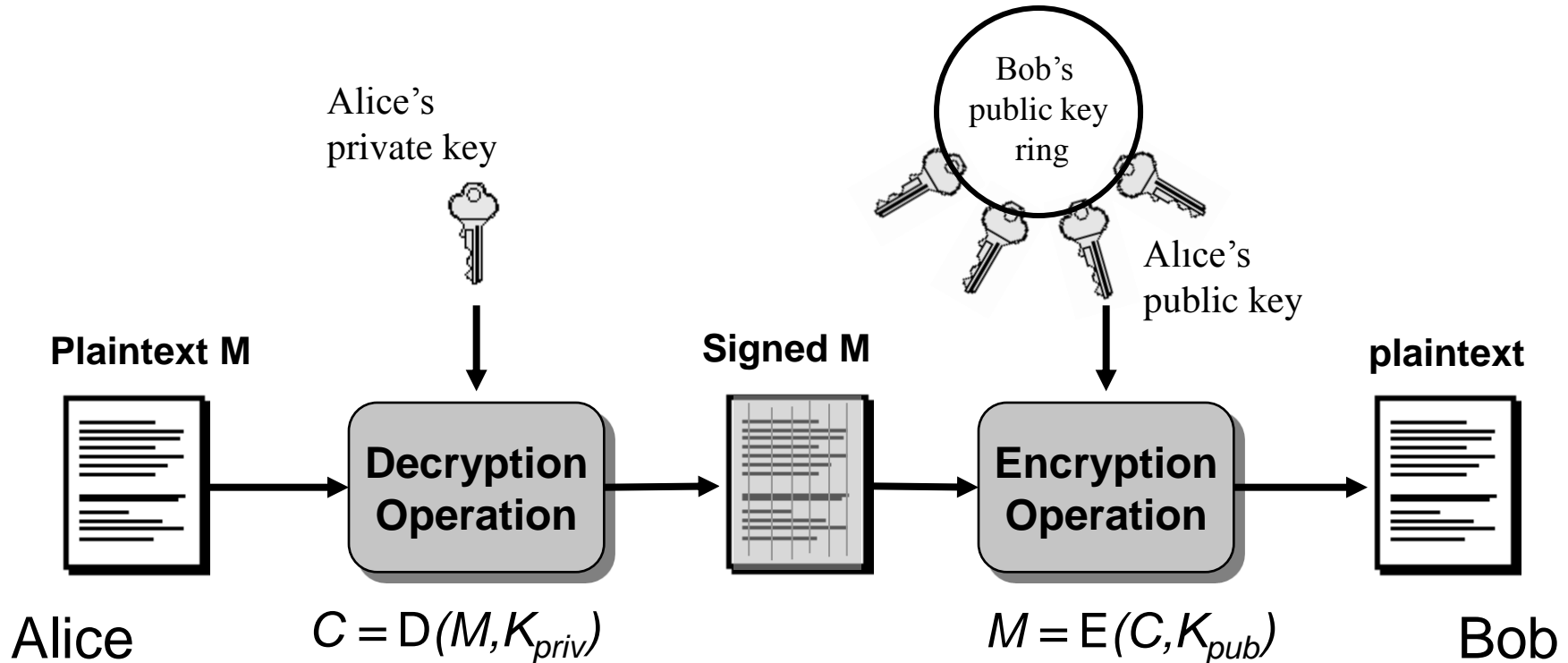


Digital Signatures

Digital Signature Mechanisms

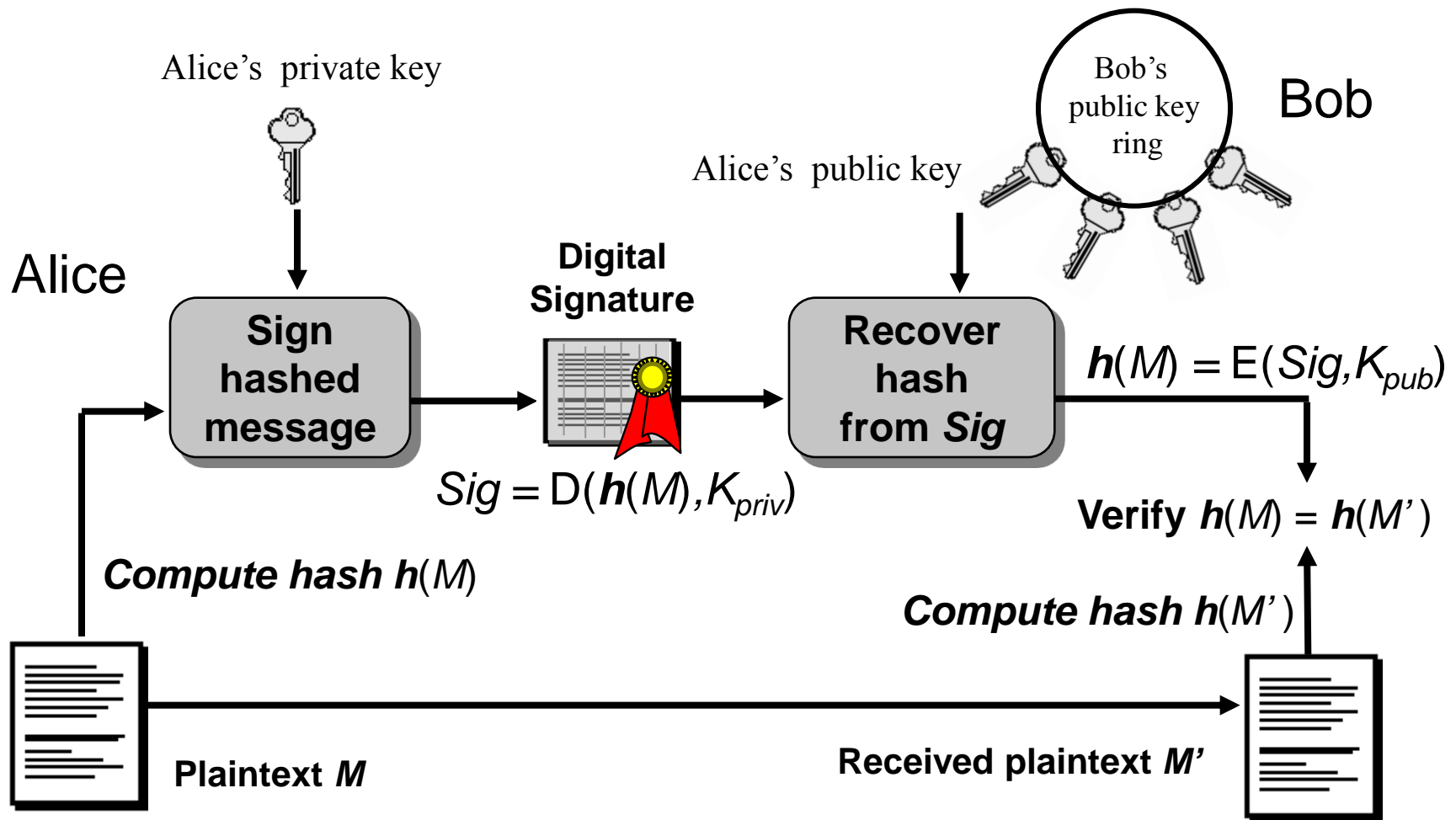
- A MAC cannot be used as evidence that should be verified by a third party.
- Digital signatures used for non-repudiation, data origin authentication and data integrity services, and in some authentication exchange mechanisms.
- Digital signature mechanisms have three components:
 - key generation
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



Digital Signatures

- To get an authentication service that links a document to A 's name (identity) and not just a verification key, we require a procedure for B to get an authentic copy of A 's public key.
- Only then do we have a service that proves the authenticity of documents 'signed by A '.
- This can be provided by a PKI (Public Key Infrastructure)
- Yet even such a service does not provide **non-repudiation** at the level of persons.

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

Ciphers and security

- A cipher must
 - be hard to cryptanalyse
 - use a sufficiently large key
 - Algorithm secrecy makes cryptanalysis harder, but
 - can give false assurance, i.e. “security by obscurity”
 - challenging to keep cipher design confidential
 - safest to assume that attacker knows cipher
- 
- Auguste Kerckhoffs proposed in 1883 that communication security should only be based on the secrecy of the key
 - Still, many organisations use secret algorithms.

End of lecture