Methods of Modern Symmetric Key Cryptography

Christina Boura

(christina.boura@uvsq.fr)

Université de Versailles, France

May 21, 2021

Cryptography

> *"Cryptography is the study and practice of methods for secure communications in the presence of adversaries."*

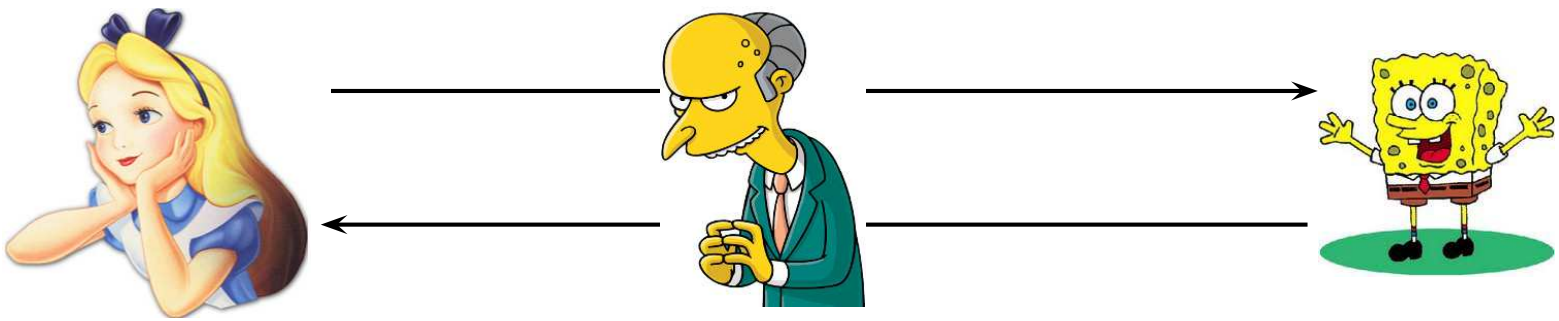
Ron Rivest

Ensure different security goals:

- Confidentiality: the message remains secret for unauthorized people.
- Authenticity: the sender is authentic.
- Integrity: the message was not modified during transition.

Confidentiality

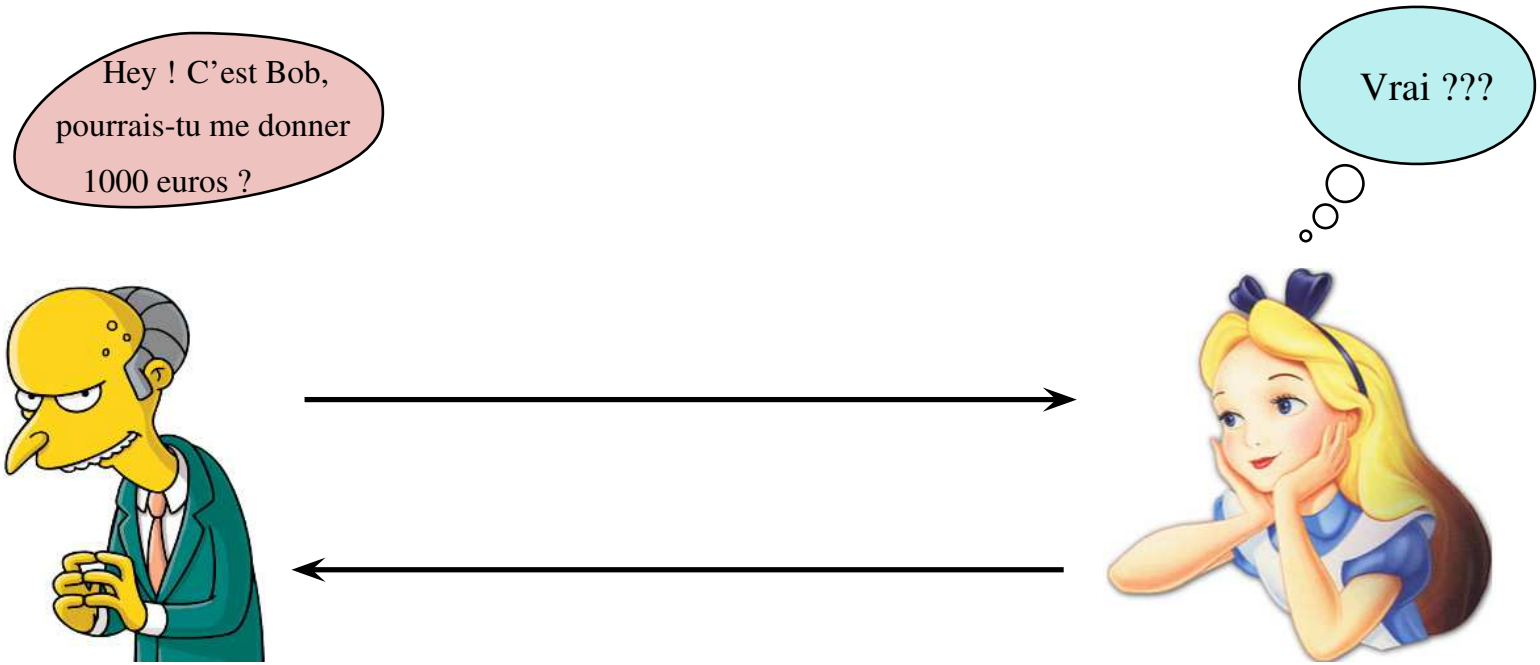
Protect the content of the information transmitted through a channel.



Store information securely.

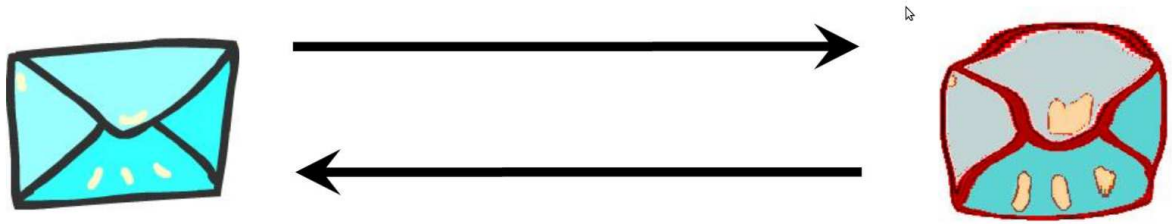
Authenticity

Be sure of the message origine and of the authenticity of its sender.



Integrity

Ensure that the message was not modified during transition.



Modern Cryptography

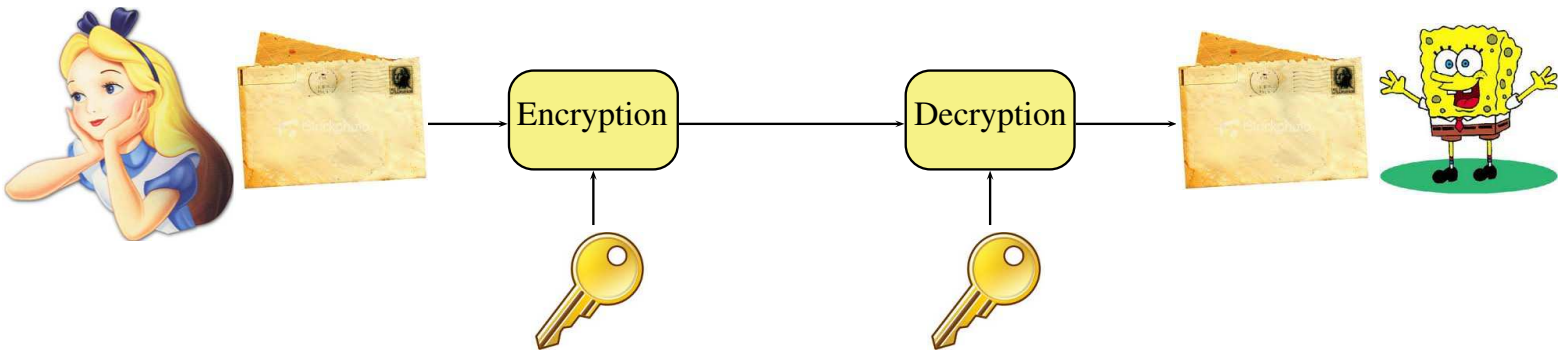
History of modern cryptography coincides with the history of computer development.

Two main branches since the 70s :

- Secret key (or symmetric) cryptography
- Public key (or asymmetric) cryptography

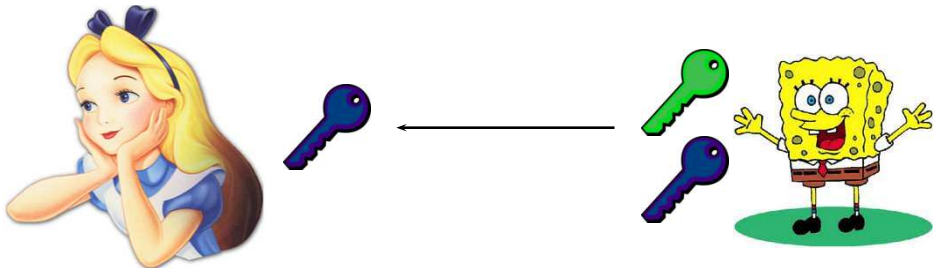
Symmetric key encryption

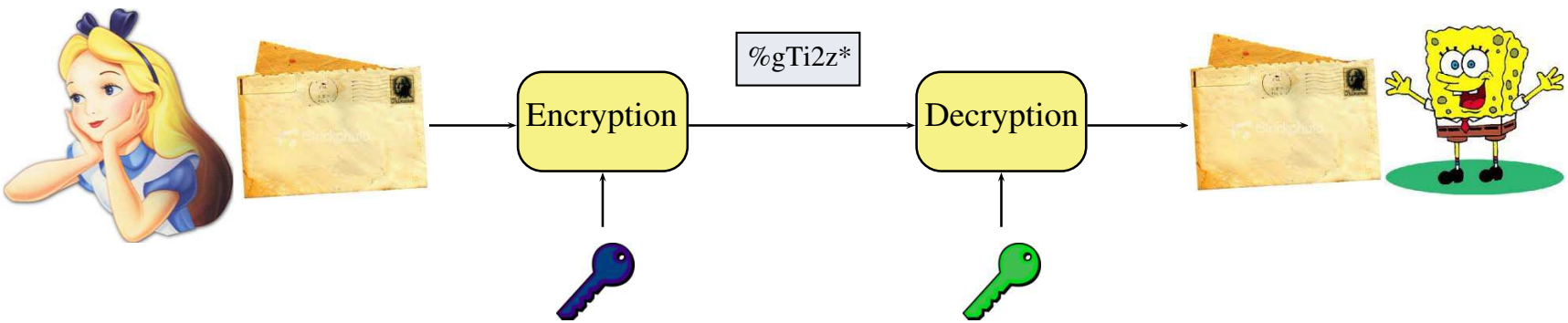
Alice and Bob exchange the secret key through a **secure channel**.



Key-exchange problem $\Rightarrow$ birth of the public key cryptography.

Public key encryption



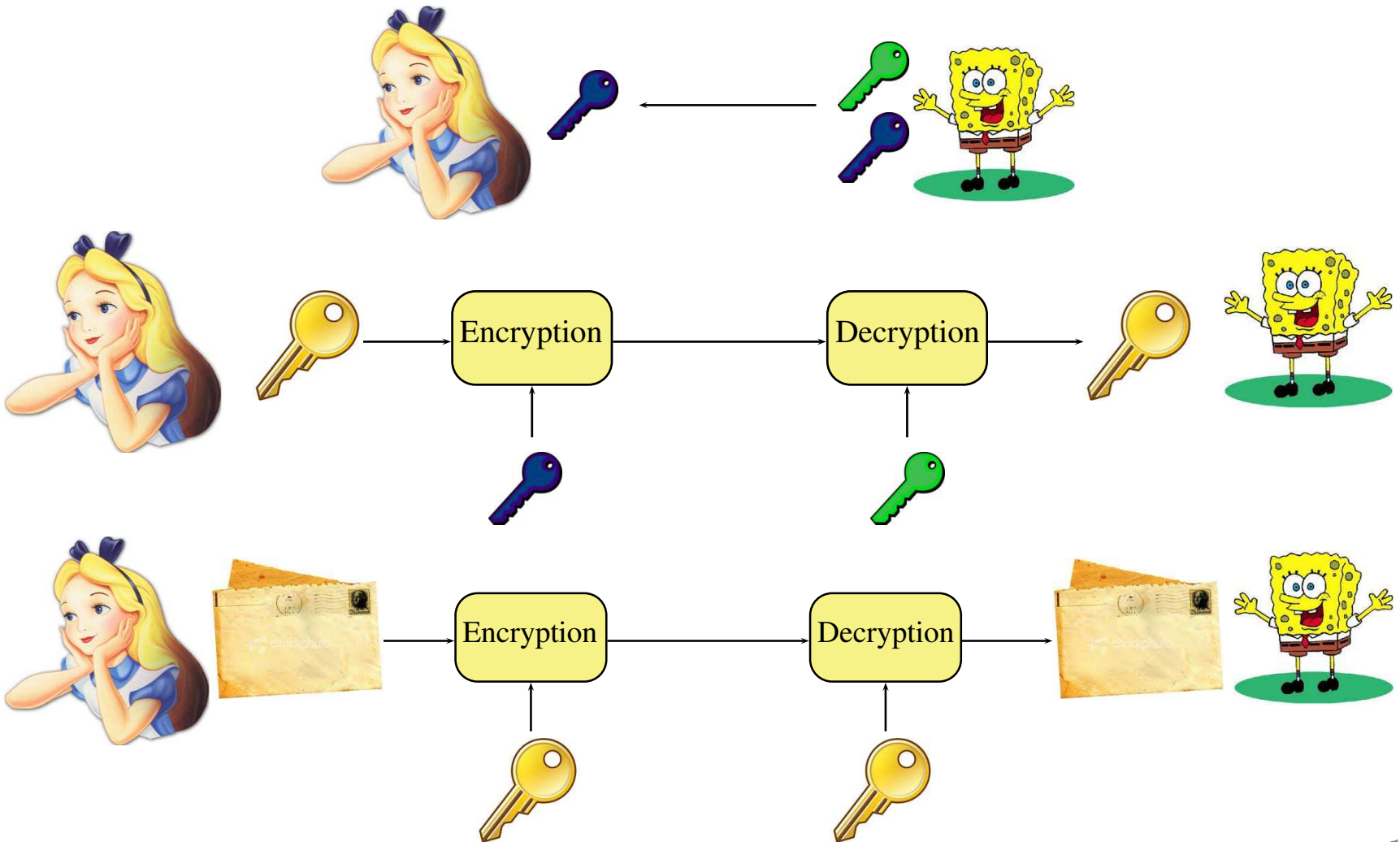


Advantages and disadvantages of each system

	Advantages	**Disadvantages**
Symmetric key	Fast systems Relatively short-keys	Need secure key-exchange n users: $\frac{n(n-1)}{2}$ keys
Public key	No key-exchange needed n users: $2n$ keys	Slow systems Relatively long-keys

Hybrid encryption

Idea: Use a combination of asymmetric and symmetric encryption to benefit from the strengths of every system.



Hybrid encryption

- Use a public-key cryptosystem to exchange a key (session key).
- Use the exchanged key to encrypt data by using a symmetric-key cryptosystem.

Advantages:

- Slow public-key cryptosystem is used to encrypt a short string only.
- Fast symmetric-key cryptosystem is used to encrypt the longer communication session.

Symmetric encryption schemes

Stream ciphers

- Combine (XOR) plaintext bits with a keystream generated by a pseudo-number generator.
- Keystream should have good statistical properties.
- Advantages: Performance and low hardware complexity.

Block ciphers

- Operate on blocks of data.
- Probably the best understood symmetric primitives.
- Can be used to build hash functions, stream ciphers, MACs, authenticated encryption algorithms, PRNGs...

Permutation-based schemes

Symmetric encryption schemes

Stream ciphers

- Combine (XOR) plaintext bits with a keystream generated by a pseudo-number generator.
- Keystream should have good statistical properties.
- Performance and low hardware complexity.

Block ciphers

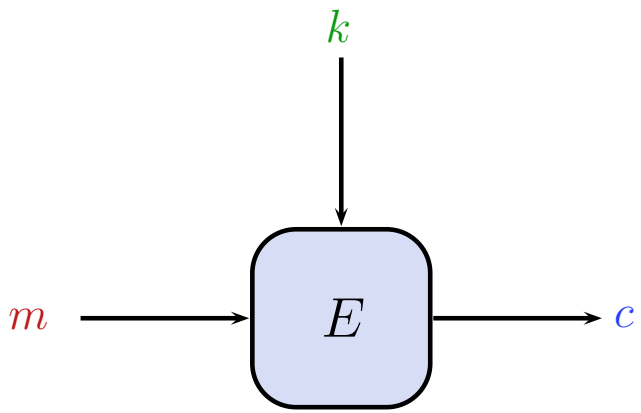
- Operate on blocks of data.
- Probably the best understood symmetric primitives.
- Can be used to build hash functions, stream ciphers, MACs, authenticated encryption algorithms, PRNGs...

Permutation-based schemes

Block ciphers

Encrypt a block of **message** m into a block of **ciphertext** c under the action of the **key** k.

$$\begin{aligned} E : \{0,1\}^n \times \{0,1\}^\kappa &\rightarrow \{0,1\}^n \\ (m,k) &\mapsto E(m,k) = c \end{aligned}$$

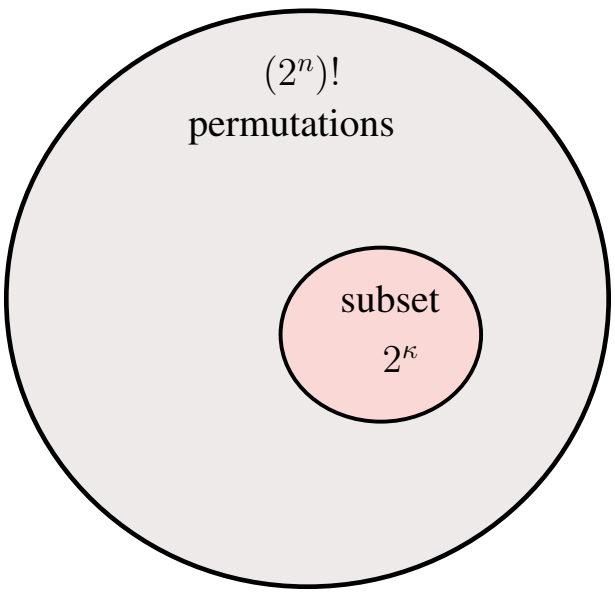


- Given k, it must be easy to compute c from m.
- Given m, c it must be hard to compute k such that $E(m,k) = c$.

Two important parameters:

- block size, n ($64 - 256$ bits)
- key size, κ ($80 - 256$ bits)

A block cipher generates a family of permutations indexed by a key k.



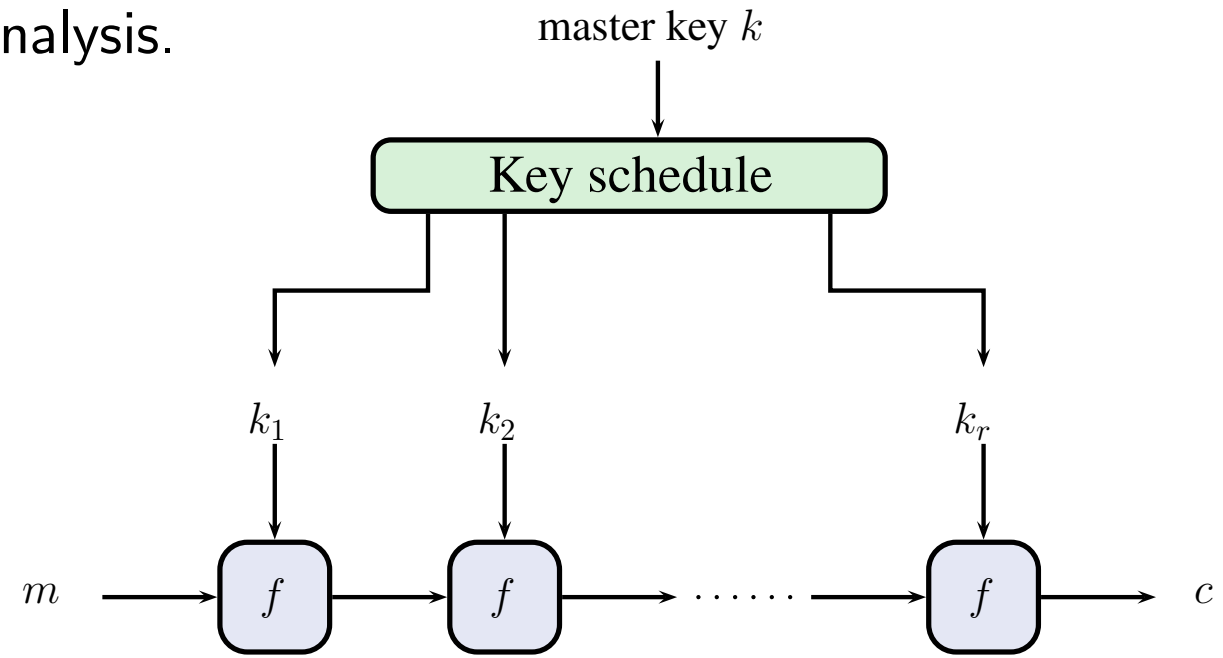
Ideal design: 2^κ permutations chosen uniformly at random from all $2^n! \approx 2^{(n-1)2^n}$ permutations.

Iterated block ciphers

Idea: Iterate a round function f several times. The function f^r is waited to be strong for large r.

Advantages:

- Compact implementation.
- Easier analysis.



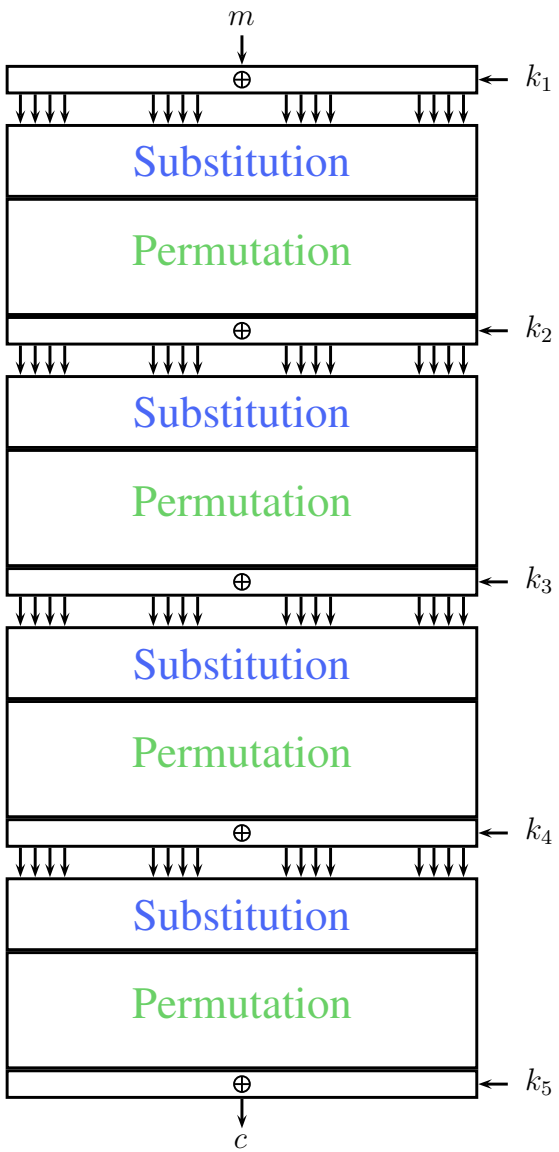
Use a key schedule to extend the user-supplied (or master) key to a sequence of r subkeys.

How to build the round function?

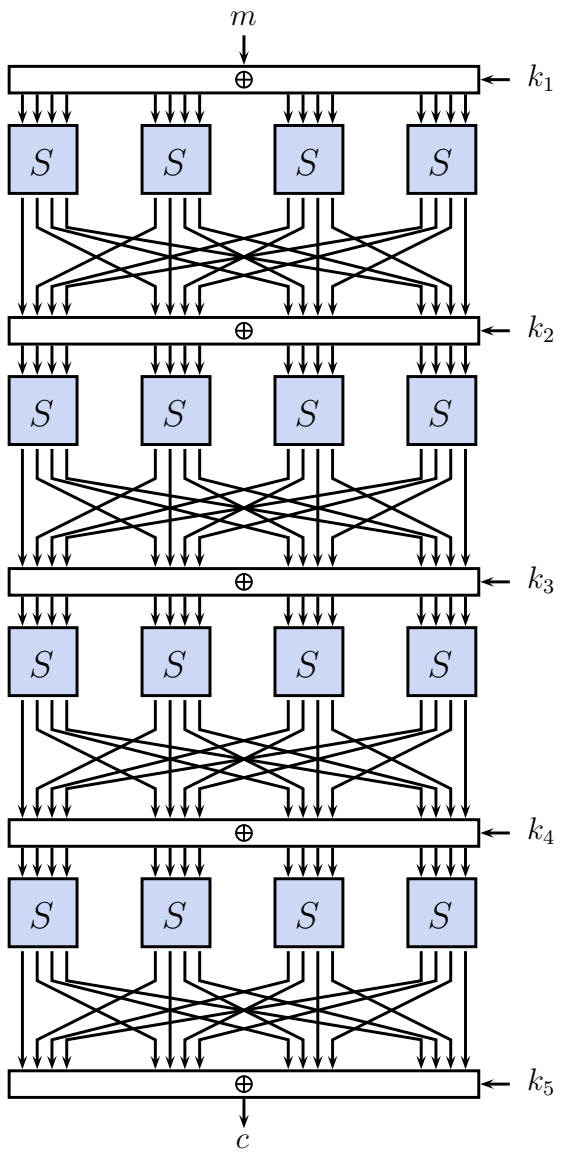
Two major approaches:

- Feistel network
- Substitution-Permutation Network (SPN)

Substitution Permutation Network (SPN)



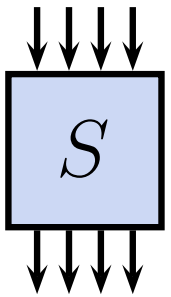
Substitution Permutation Network (SPN)



Cryptographic Sboxes

An Sbox can be seen as a vectorial Boolean function

$$S : \mathbb{F}_2^n \to \mathbb{F}_2^m$$



- Typically $n = m$ and $n \in \{3, 4, 5, 6, 7, 8\}$

Example (Sbox of PRESENT)

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$S(x)$	12	5	6	11	9	0	10	13	3	14	15	8	4	7	1	2

Algebraic Normal Form of the Sbox

$$
\begin{aligned}
S_1 &= x_1 + x_3 + x_4 + x_2x_3 \\
S_2 &= x_2 + x_4 + x_2x_4 + x_3x_4 + x_1x_2x_3 + x_1x_2x_4 + x_1x_3x_4 \\
S_3 &= 1 + x_3 + x_4 + x_1x_2 + x_1x_4 + x_2x_4 + x_1x_2x_4 + x_1x_3x_4 \\
S_4 &= 1 + x_1 + x_2 + x_4 + x_2x_3 + x_1x_2x_3 + x_1x_2x_4 + x_1x_3x_4
\end{aligned}
$$

- An Sbox is usually the only **nonlinear** component of the cipher.
- Security arguments for the cipher heavily depend on the properties of the Sbox.

Security of a block cipher

Symmetric-key approach:

Try to make the system **secure against all known attacks**.

- No attack should be faster than exhaustive search on the key.
- The more an algorithm was analyzed without any attack found, the more we can trust it.

Exhaustive search

Expected time to recover a κ-bit key: $2^{\kappa-1}$ operations.

κ (bits)	Time complexity (operations)	Security
40	2^{40}	easy to break
64	2^{64}	practical to break
80	2^{80}	not currently feasible
128	2^{128}	very strong
256	2^{256}	exceptionally strong

Table from [Knudsen, Robshaw, "*The Block Cipher Companion*", 2011.]

- The universe is less than 2^{80} microseconds old!
- The number of the protons in the universe is $\approx 2^{265}$.

Statistical attacks

Statistical attacks exploit relations that hold with a certain probability only.

- Rely on the existence of a distinguisher.

A distinguisher $\mathcal{D}$ for a block cipher $(E_k)_k$ is an algorithm taking N pairs (x_i, y_i), $1 \leq i \leq N$ and returning 0 or 1.

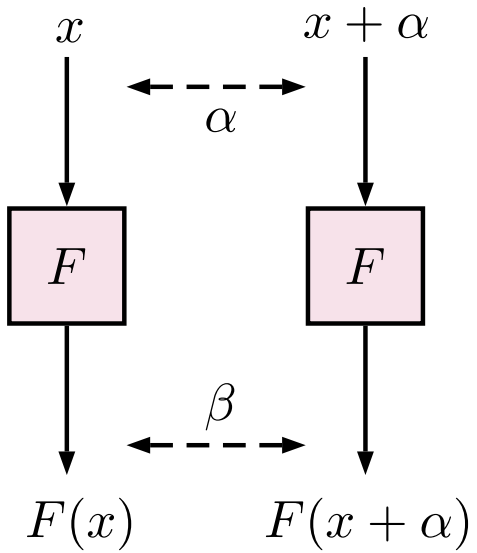
Goal: Decide if the N pairs are input-output pairs of the target block cipher or not:

- 1: If the (x_i, y_i) are input-output pairs of E_k for some key k.
- 0: If the (x_i, y_i) are input-output pairs of a random permutation.

Differential cryptanalysis

Differential cryptanalysis: one of the most prominent attacks against block ciphers [Biham - Shamir '90].

For an SPN cipher, the security against differential cryptanalysis **reduces** on the differential properties of the Sbox.



Difference distribution table (DDT)

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	4	.	2	.	.	2	.	.	2	.	.	2	.	.	4
2	.	4	2	.	.	.	.	2	.	2	.	4	2	.	.	.
3	.	.	2	2	.	.	2	2	4	.	.	.	.	4	.	.
4	.	.	2	.	2	2	2	.	.	2	2	2	.	2	.	.
5	.	.	2	2	4	4	2	2	.	.	.	.	.	.	.	.
6	.	.	2	2	.	.	2	2	.	.	2	2	.	.	2	2
7	.	.	2	.	2	2	2	.	.	2	.	.	.	2	2	2
8	.	.	.	.	.	.	.	.	4	.	2	2	4	.	2	2
9	.	.	.	2	2	2	.	2	.	2	.	.	.	2	2	2
a	.	.	.	2	2	2	.	2	.	2	2	2	.	2	.	.
b	.	.	.	.	4	4	.	.	4	.	.	.	4	.	.	.
c	.	4	.	2	.	.	2	.	.	2	4	.	2	.	.	.
d	.	.	.	.	.	.	.	.	.	.	4	4	.	.	4	4
e	.	.	2	2	.	.	2	2	4	.	.	.	.	4	.	.
f	.	4	2	.	.	.	.	2	.	2	.	.	2	.	4	.

$$\delta(\alpha, \beta) = \#\{x : S(x) + S(x + \alpha) = \beta\}$$

Differential Uniformity

All entries of the DDT must be small.

Differential uniformity ([Nyberg 93]

$$\delta(S) = \max_{\alpha,\beta \neq 0} \#\{x \in \mathbb{F}_2^n : S(x) + S(x+a) = b\}$$

- $\delta(S)$ is always even and $\delta(S) \geq 2$.

Functions S for which this bound is achieved are called Almost Perfect Nonlinear (APN) functions.

Permutations of $\mathbb{F}_2^n$, n even

Big APN Problem Do these exist APN permutations for n even?

Dillon permutation ($n = 6$) [Dillon 09]

$$
\begin{aligned}
S = \{&0, 54, 48, 13, 15, 18, 53, 35, 25, 63, 45, 52, 3, 20, 41, 33, 59,\\
&36, 2, 34, 10, 8, 57, 37, 60, 19, 42, 14, 50, 26, 58, 24, 39, 27, 21,\\
&17, 16, 29, 1, 62, 47, 40, 51, 56, 7, 43, 44, 38, 31, 11, 4, 28, 61,\\
&46, 5, 49, 9, 6, 23, 32, 30, 12, 55, 22\}
\end{aligned}
$$

- $\delta(S) = 2$
- $\mathcal{L}(S) = 16$
- $\deg(S) = 4$

Dillon's permutation is the only known APN permutation for an even number of variables.

Other criteria

- Differential uniformity is only one of the many criteria that an Sbox should satisfy.
- Take into account resistance against other attacks (linear, boomerang, algebraic . . . cryptanalysis)
- Implementation properties are also important.
- Resistance against side-channel attacks.