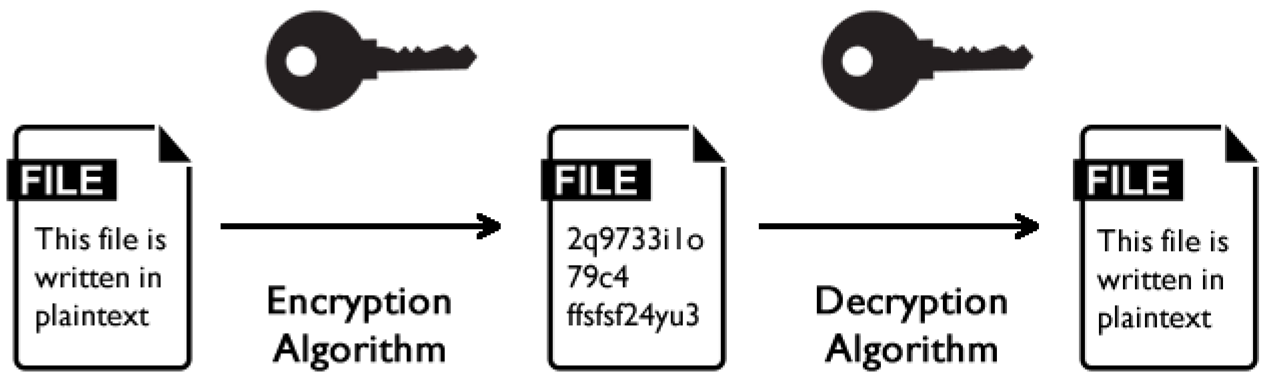
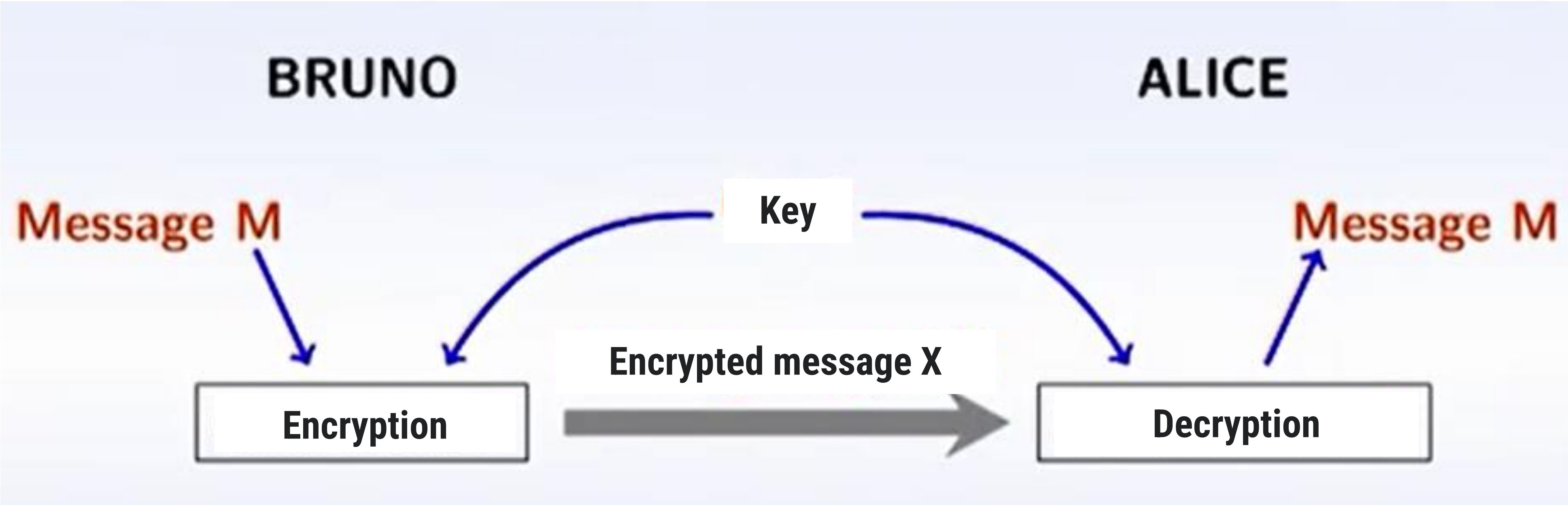
Symmetric Key Encryption

- Secret Key Encryption
- The same key is used for both encryption and decryption.
- Both parties agree on a private key beforehand.



Secret Key Encryption

- **Principle**



Secret Key Encryption

- **Principle**

Symmetric Key Algorithms

DES, AES, IDEA, 3DES, CAST, Skipjack, Serpent, Mars…

Advantages: Very fast
Disadvantage: Unsecured key transfer

To communicate securely

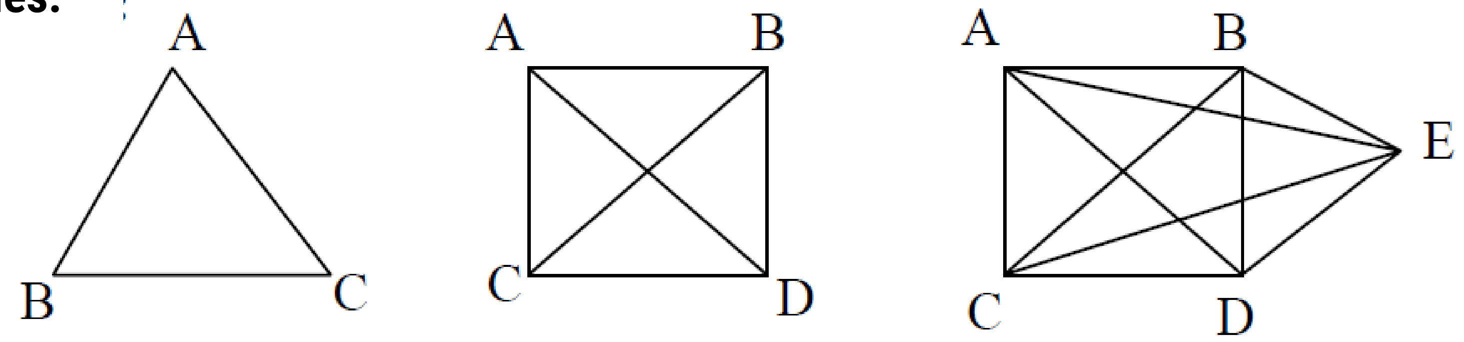
➔

Use a key for each pair

➔

$(n^2 - n) / 2$ **keys**

Examples:



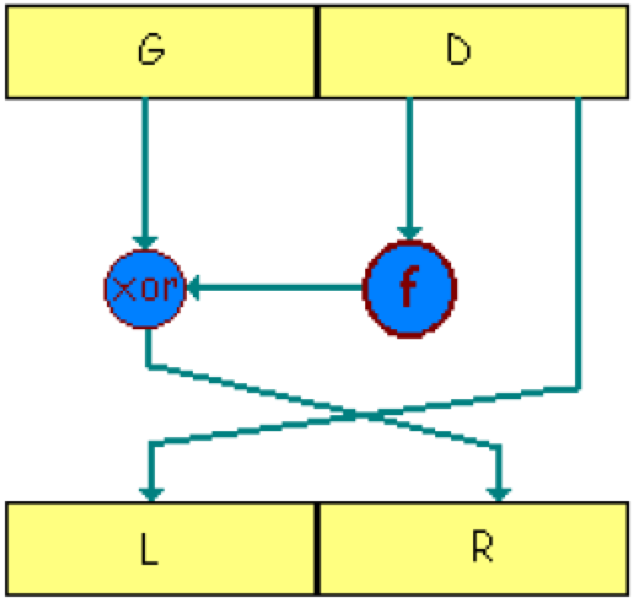
4 users	6 keys
5 users	10 keys
10 users	45 keys

Objective of secret key algorithms

✓ **Seeking perfection = Seeking randomness**

the encrypted message must appear as random as possible to limit the risk of attack

Random Feistel Bijection



- Choose a random function ***f*** having n bits as arguments
- Encrypt blocks divided into two parts Left and Right

➢ Encryption: L = D and R = G xor *f*(D)
➢ Decryption: D = L and G = R xor *f*(L)

We repeat the Feistel diagram a number of times (rounds)
(in DES, the number of rounds = 16)

DES (Data Encryption Standard)

- DES was the official encryption tool of the US government (until 2005), developed by IBM in the 1970s.
- 64-bit block and 64-bit secret key encryption system

DES:

- SYMETRIC
- REVERSIBLE
- BLOCK-BASED
- SECRET-KEY

DES KEY

The DES key is a 64-bit string: only 56 bits are actually used to define the key. The remaining 8 bits (8, 16, 24, 32, 40, 48, 56, 64) are parity bits

2^{56} possible keys (≈ 72 millions of billions possibilities)

DES: ENCRYPTION STEPS

Plaintext message = Series of 64-bit blocks

Steps:

DES uses a secret key of 56 bits, which it transforms into 16 "sub-keys" of 48 bits each (one for each iteration). The encryption process consists of 19 steps:

- **1st step**

The first step is a fixed (standard) transposition of the 64 bits to be encrypted.

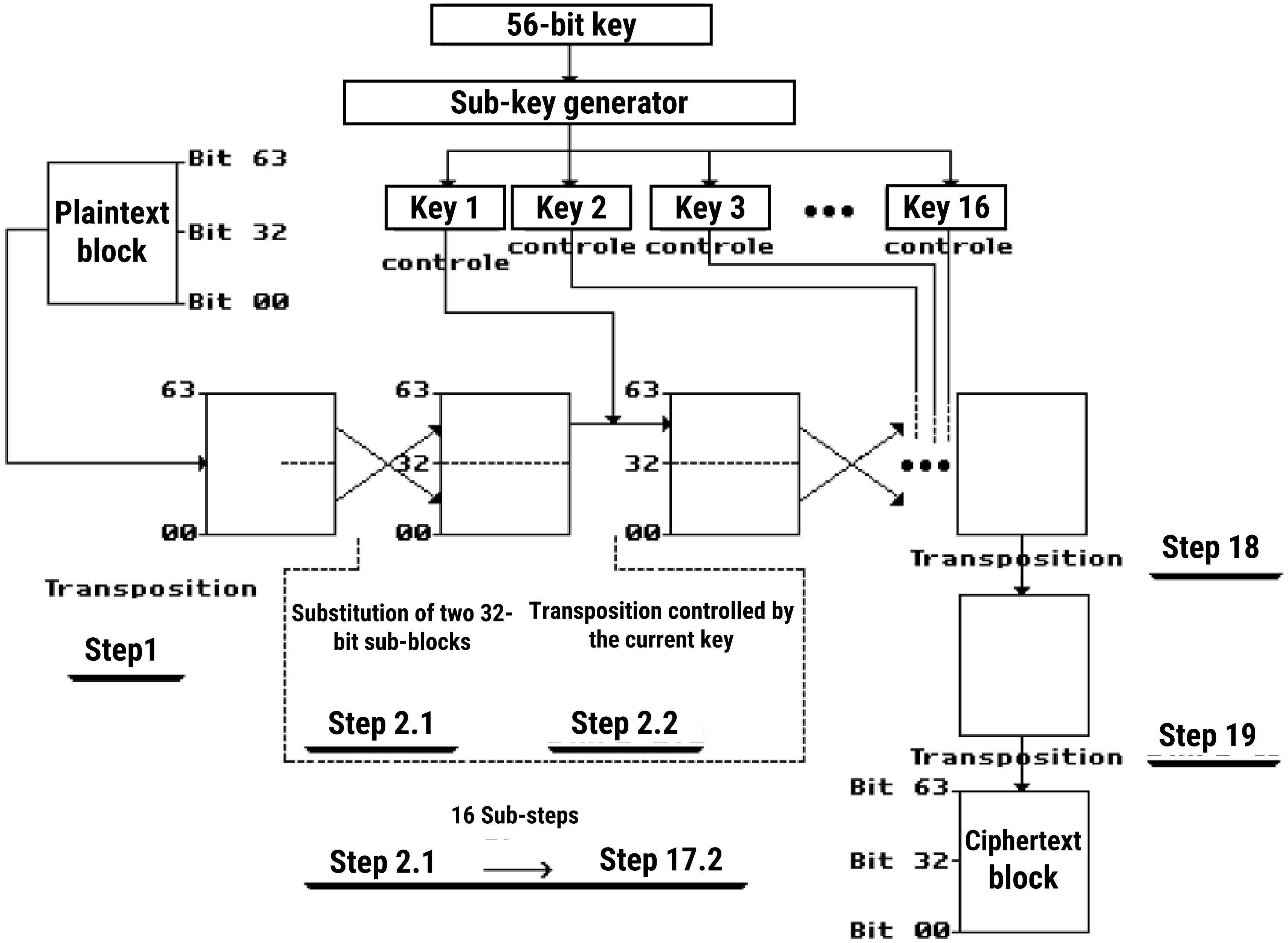
- **Following 16 steps**

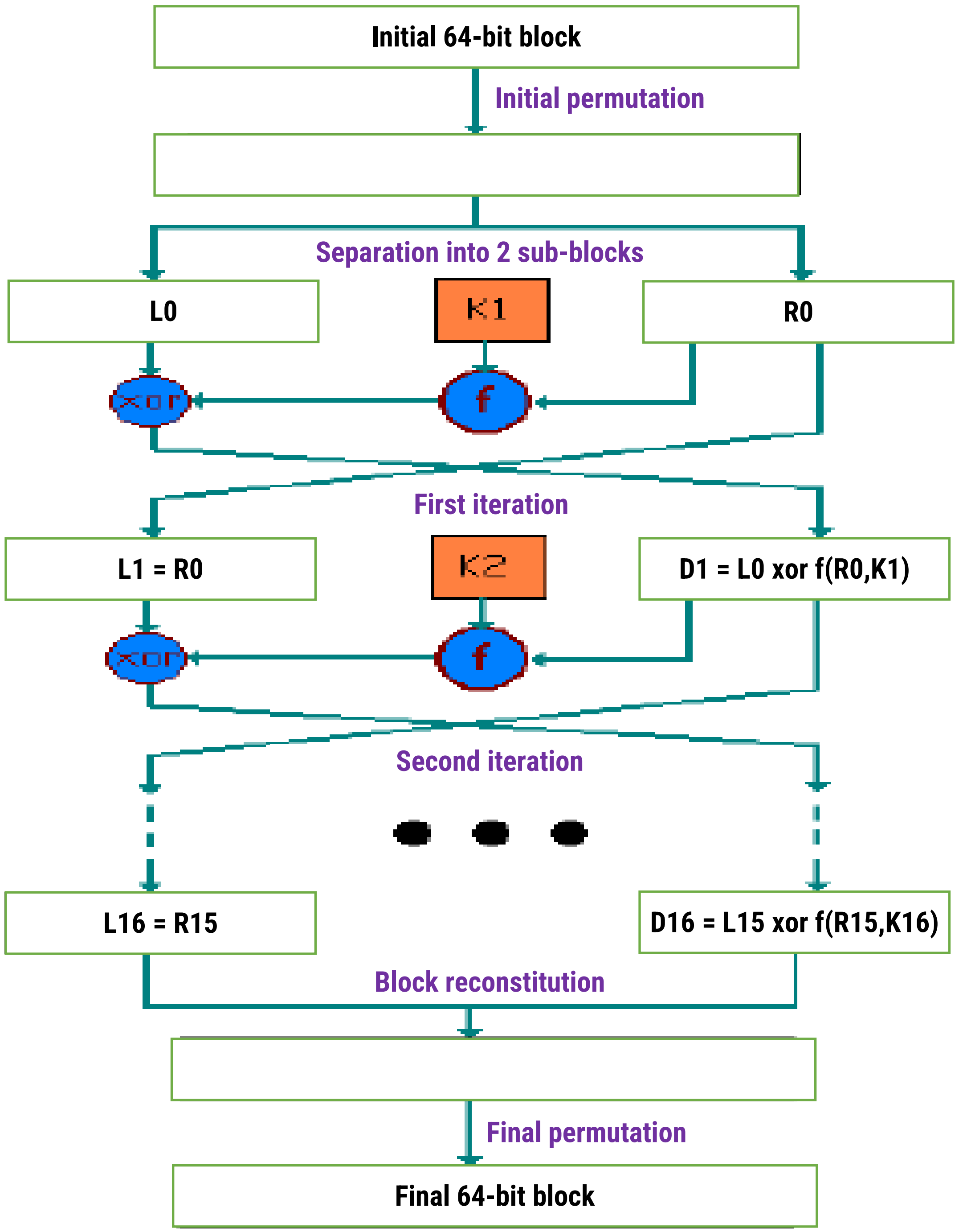
The following **16 steps** can be divided into **2 "sub-steps"** each. Firstly, the **64-bit block** is **split** into **2x32 bits**, and a substitution is performed between these two blocks; in fact, these two blocks will simply be exchanged with each other. Secondly, the **32-bit block** with the highest weight (the block ranging from **bit #32 to bit #63**) undergoes a transposition controlled by the **sub-key** corresponding to the current step.

- **Steps 18 and 19**

The last two steps are two transpositions.

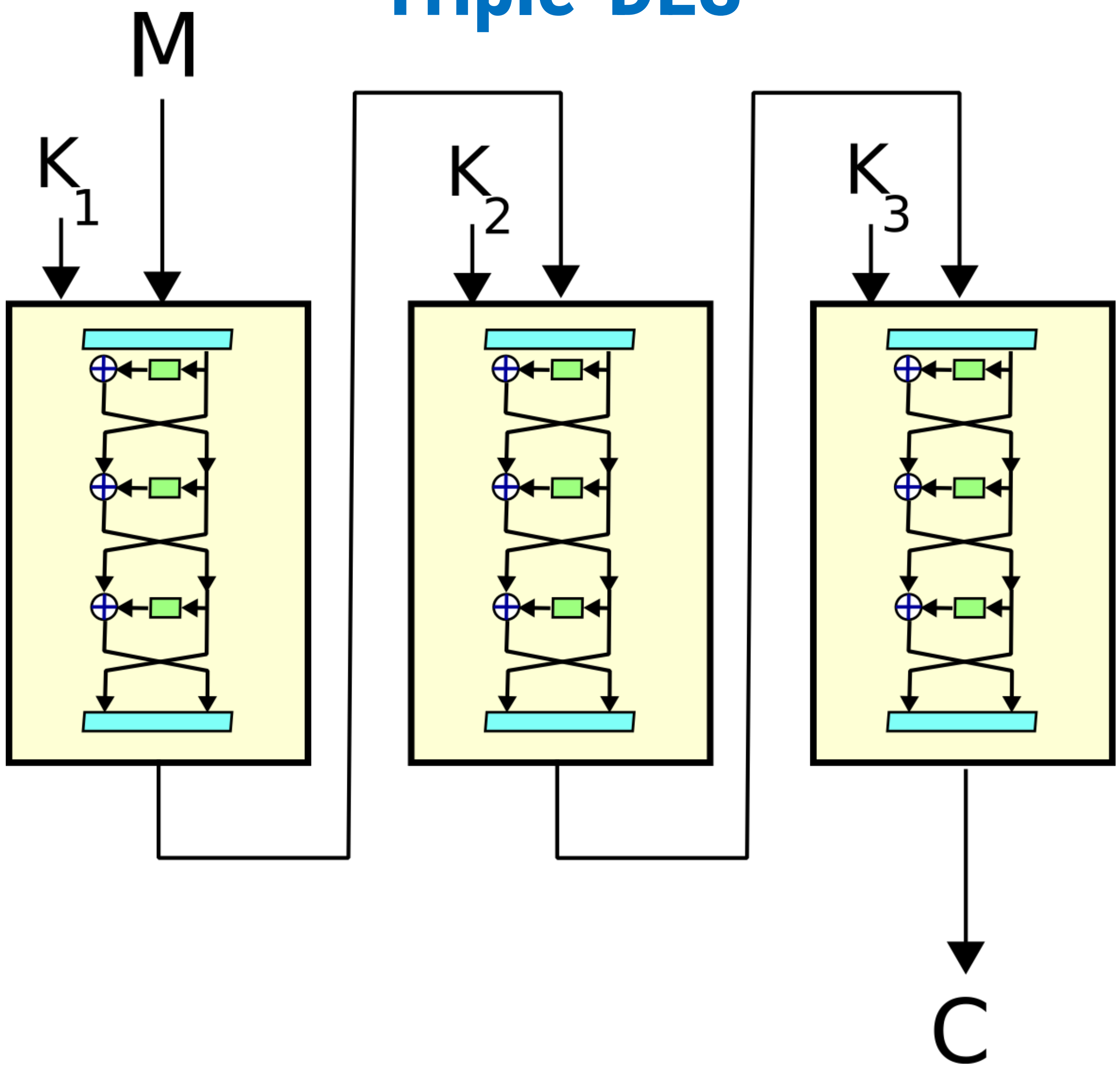
DES ENCRYPTION DIAGRAM





Initial 64-bit block
Initial permutation
Separation into 2 sub-blocks
L0
K1
R0
xor
f
First iteration
L1 = R0
K2
D1 = L0 xor f(R0,K1)
xor
f
Second iteration
L16 = R15
D16 = L15 xor f(R15,K16)
Block reconstitution
Final permutation
Final 64-bit block

Triple-DES



AES (Advanced Encryption Standard)

- The algorithm proceeds in **blocks** of **128 bits**, with a **key** of **128 bits** as well.

- Each block undergoes a sequence of **5 transformations** repeated **10 times**.

AES: ENCRYPTION STEPS

1. **Addition of the secret key (by a XOR).**

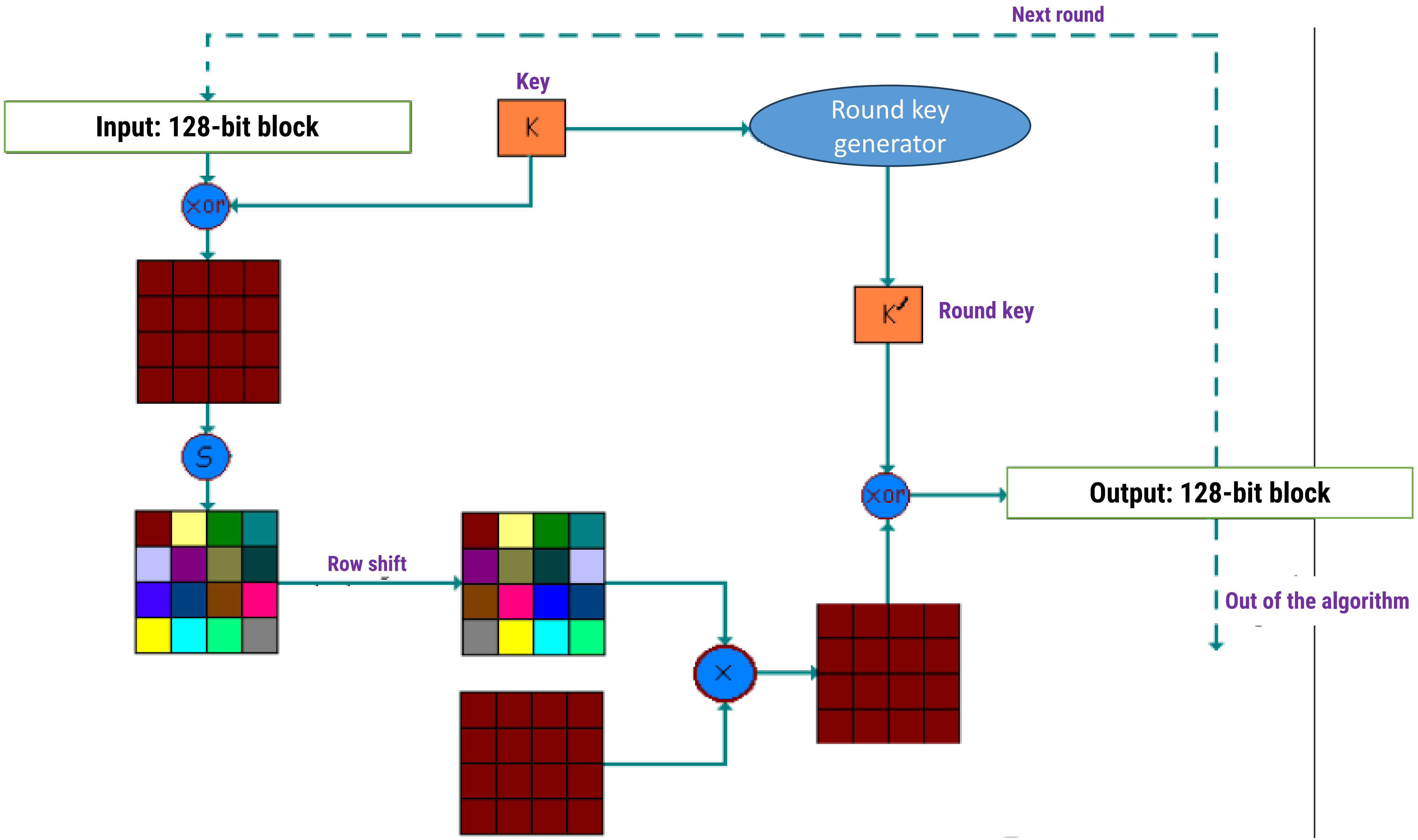
2. **Nonlinear byte transformation:** the 128 bits are divided into 16 blocks of 8 bits, themselves distributed in a 4×4 table. Each byte is transformed by a nonlinear function S.

3. **Row shift:** the last 3 rows are shifted cyclically to the left: the 2nd row is shifted by one column, the 3rd row by 2 columns, and the 4th row by 3 columns.

4. **Column scrambling:** Each column is transformed by linear combinations of the different elements of the column (i.e: multiplying the 4×4 matrix by another 4×4 matrix).

5. **Addition of the turn key:** At each round, a round key is generated from the secret key by a sub-algorithm. This round key is added by a XOR to the last block obtained.

AES: ENCRYPTION DIAGRAM



5 transformations repeated 10 times

Public Key Encryption

- Asymetric encryption
- 2 keys: public and private
- A message encrypted with one of the two keys can only be decrypted with the other key

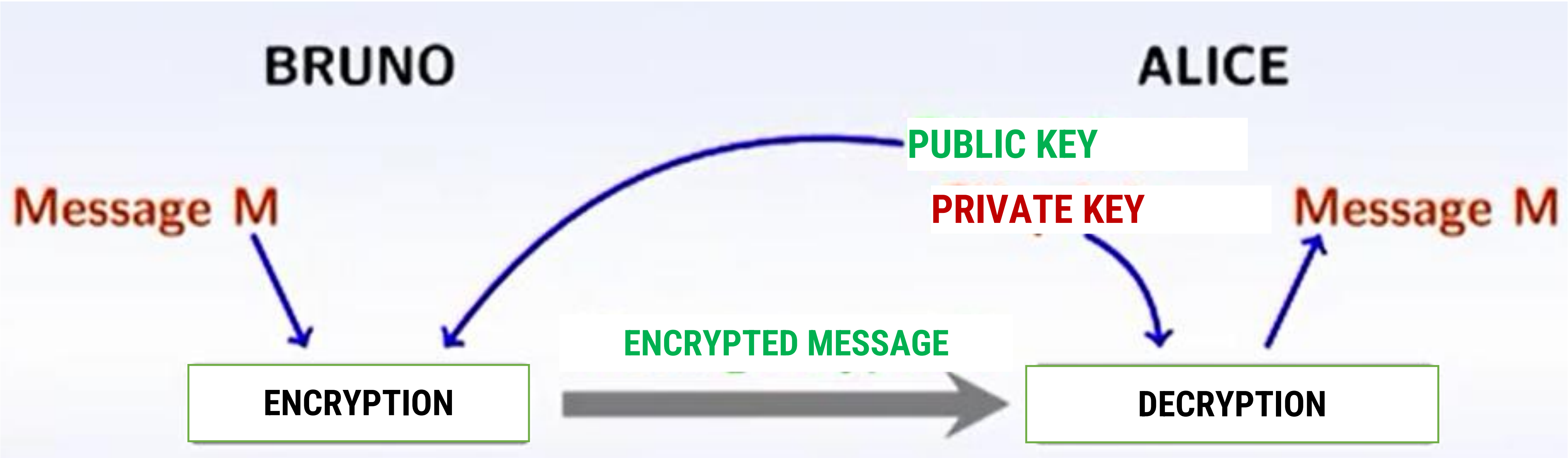
Algorithms: RSA, Diffie-Hellman, ElGamal, DSA…

Advantages:
100 users, we use 100 pairs of keys (4950 keys for a symetric encryption).

Disadvantages:
- Public key algorithms are complex and are 100 to 1000 times slower than secret key algorithms.
- Public key cryptosystems are vulnerable to certain attacks

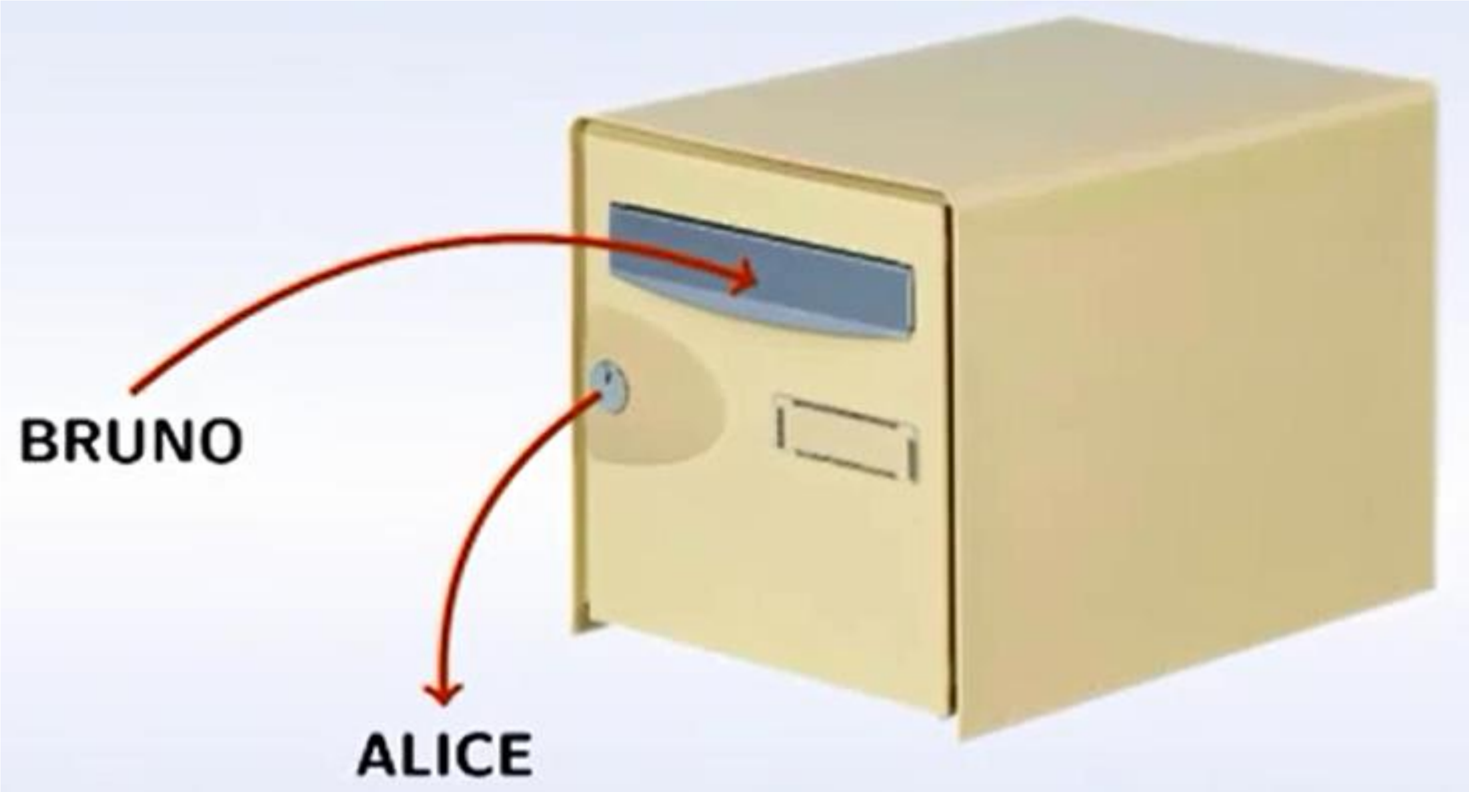
Public Key Encryption

- **Principle**



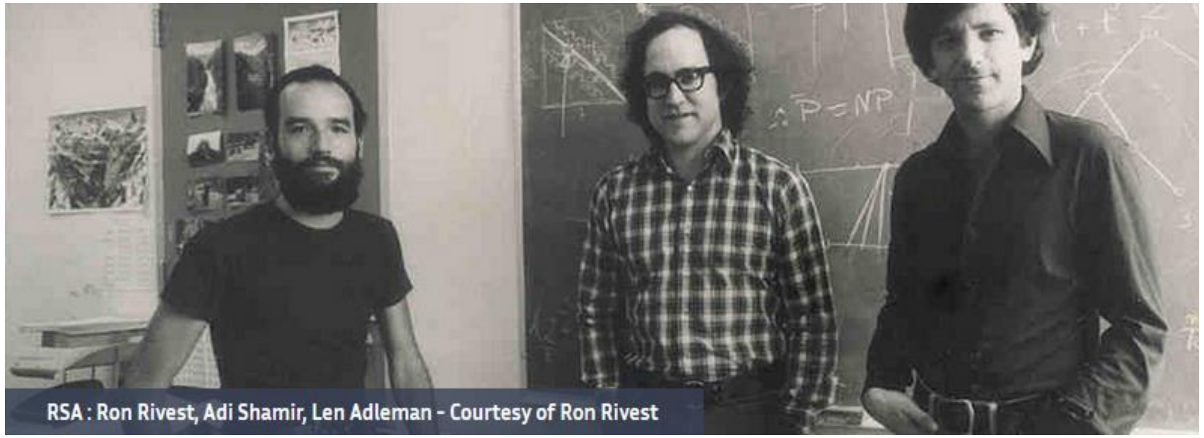
Public Key Encryption

- **Principle**



RSA (Rivest, Shamir, Adelman)

- Developed in 1978 by Ronald Rivest, Adi Shamir and Leonard Adelman.
- Most public key cryptosystems are based on this algorithm.
- Based on Factorization



RSA : Ron Rivest, Adi Shamir, Len Adleman - Courtesy of Ron Rivest

Mathematical principles of RSA

- ## Congruence

Consider **n** an integer such as: **n ≥ 2**

We say that a is congruent to b modulo n, if (a-b) is divisible by n

We note: **$a \equiv b \pmod{n}$**

Example

- $28 \equiv 2 \pmod{26}$, **Because 28 – 2 is divisible by 26**
- $85 = 26 + 59$ donc $85 \equiv 59 \pmod{26}$
- $85 = 3 \times 26 + 7$ donc $85 \equiv 7 \pmod{26}$

Mathematical principles of RSA

▪ Modular addition

Consider **a,b** and **n** integers :

$$a + b \pmod{n} = a \pmod{n} + b \pmod{n}$$

Example

Calculate : $133 + 64 \pmod{26}$

- $133 + 64 = 197 = 7 \times 26 + 15 \equiv 15 \pmod{26}$
-
  - $133 = 5 \times 26 + 3 \equiv 3 \pmod{26}$
  - $64 = 2 \times 26 + 12 \equiv 12 \pmod{26}$
  - $133 + 64 \equiv 3 + 12 \equiv 15 \pmod{26}$

Mathematical principles of RSA

▪ Modular multiplication

Consider **a,b** and **n** integers:

$$a \times b \pmod{n} = a \pmod{n} \times b \pmod{n}$$

Example

Calculate: $3 \times 27 \pmod{26}$

- $3 \times 27 = 81 = 3 \times 26 + 3 \equiv 3 \pmod{26}$
- $27 \equiv 1 \pmod{26}$ **then** $3 \times 27 \equiv 3 \times 1 \equiv 3 \pmod{26}$

Mathematical principles of RSA

Factorization complexity

- $5 \times 7 = ?$
- $35 = ?$
- Factorize 1591?
- Calculate 37×43
- Calculate $p \times q$ is more easier than factorize $n = pq$

The complexity estimates the calculation time (or the number of elementary operations) necessary to perform an operation

Mathematical principles of RSA

Factorization complexity

- **Addition**
  - The sum of two digits (eg. 6+8) is of complexity 1
  - The sum of two integers of n digits is of complexity n
  - Example: 1234+2323: 4 additions
- **Multiplication**
  - The multiplication of two integers of n digits is of complexity n^2
  - Example: 1234×2323: 16 multiplications
- **Factorisation** : $\exp(4n^{\frac{1}{3}})$

Mathematical principles of RSA

- **Complexity of multiplying and factorizing numbers of n digits**

n	multiplication	factorisation
3	9	320
4	16	572
5	25	934
10	100	5 528
50	2 500	2 510 835
100	10 000	115 681 968
200	40 000	14 423 748 780

Mathematical principles of RSA

- **Modular exponentiaition**

Find out an efficient method to caclulate $a^k \pmod{n}$

Example: Let's calculate $5^{11} \pmod{14}$

We notice that 11 in base 2 = (1,0,1,1) then 11 = 8 + 2 + 1

$5^{11} = 5^8 \times 5^2 \times 5^1$

Let's calculate $5^{2^i} \pmod{14}$:

$$5 \equiv 5 \pmod{14}$$
$$5^2 \equiv 25 \equiv 11 \pmod{14}$$
$$5^4 \equiv 5^2 \times 5^2 \equiv 11 \times 11 \equiv 121 \equiv 9 \pmod{14}$$
$$5^8 \equiv 5^4 \times 5^4 \equiv 9 \times 9 \equiv 81 \equiv 11 \pmod{14}$$

Consequence:

$$\begin{aligned} 5^{11} &\equiv 5^8 \times 5^2 \times 5^1 \equiv 11 \times 11 \times 5 \\ &\equiv 11 \times 55 \equiv 11 \times 13 \equiv 143 \equiv 3 \pmod{14} \end{aligned}$$

Mathematical principles of RSA

- **Modular exponentiaition**

Example: calculate $17^{154} \pmod{100}$

We notice that 154 in base 2 = (1,0,0,1,1,0,1,0) then 154 = 128 + 16 + 8 + 2

$17^{154} = 17^{128} \times 17^{16} \times 17^{8} \times 17^{2}$

Let's calculate 17, 17^2, 17^4, 17^8,…, $17^{128} \pmod{100}$:

$$\begin{aligned}
17 &\equiv 17 \pmod{100} \\
17^2 &\equiv 17 \times 17 \equiv 289 \equiv 89 \pmod{100} \\
17^4 &\equiv 17^2 \times 17^2 \equiv 89 \times 89 \equiv 7921 \equiv 21 \pmod{100} \\
17^8 &\equiv 17^4 \times 17^4 \equiv 21 \times 21 \equiv 441 \equiv 41 \pmod{100} \\
17^{16} &\equiv 17^8 \times 17^8 \equiv 41 \times 41 \equiv 1681 \equiv 81 \pmod{100} \\
17^{32} &\equiv 17^{16} \times 17^{16} \equiv 81 \times 81 \equiv 6561 \equiv 61 \pmod{100} \\
17^{64} &\equiv 17^{32} \times 17^{32} \equiv 61 \times 61 \equiv 3721 \equiv 21 \pmod{100} \\
17^{128} &\equiv 17^{64} \times 17^{64} \equiv 21 \times 21 \equiv 441 \equiv 41 \pmod{100}
\end{aligned}$$

$$\begin{aligned}
17^{154} &\equiv 17^{128} \times 17^{16} \times 17^{8} \times 17^{2} \equiv 41 \times 81 \times 41 \times 89 \\
&\equiv 3321 \times 3649 \equiv 21 \times 49 \equiv 1029 \equiv 29 \pmod{100}
\end{aligned}$$

Mathematical principles of RSA

- **Prime number**

Each positive integer **a (a > 1)** is said to be prime number if its only divisors are **1** and **itself**

- **Coprime numbers**

Two integers **a** and **b** are coprime numbers if **gcd(a,b)=1**

Mathematical principles of RSA

- ***Fermat's* Little Theorem**

  If p is a prime number and a is an integer then:

$$a^p \equiv a \pmod p$$

- **Corollary**

  if p does not divide a then:

$$a^{p-1} \equiv 1 \pmod p$$

- Example: $p = 3$, $a = 2$
  - $2^3 \equiv 2 \pmod 3$
  - $2^2 \equiv 1 \pmod 3$

Mathematical principles of RSA

▪ Improved *Fermat's* Little Theorem

Consider p and q two distinct prime nulbers and let $n = pq$

For each integer a such that $\gcd(a,n)=1$ we have:

$$a^{(p-1)(q-1)} \equiv 1 \pmod{n}$$

- Example : $p = 5$, $q = 7$
  - $n = p \times q = 35$
  - $(p - 1) \times (q - 1) = 4 \times 6 = 24$
  - For $a = 1, 2, 3, 4, 6, 8, 9, 11, 12, 13,..$ $a^{24} \equiv 1 \pmod{35}$

Mathematical principles of RSA

- **Principle of the Euclidean Algorithm**

pgcd(a,b) = pgcd(b, a mod(b))

- **Extended Euclidean Algorithm**

Calculate the Bézout coefficients *u* and *v* such that:

a*u*+b*v* = gcd(a,b)

Mathematical principles of RSA

- **The inverse modulo n**

Let **a** and **x** two integers, we say that **x** is an inverse of **a** modulo **n** if:

$$ax \equiv 1 \pmod{n}$$

Example :

$3 \times 9 \equiv 1 \pmod{26}$

9 is an inverse of 3 modulo 26

- **a** has an inverse modulo **n** if and only if: **gcd(a,n)=1**
- If $au + nv = 1$ then **u** is an inverse of **a** modulo **n**

RSA ENCRYPTION

▪ Encryption parameters

- **Look for a difficult problem:**

Factorizing an integer that is the product of two distinct prime numbers."

- **Calculation of the two keys, public and private**:

Using the Euclidean algorithm and Bézout's coefficients.

$au+bv = pgcd(a,b)$

- **Environment:**

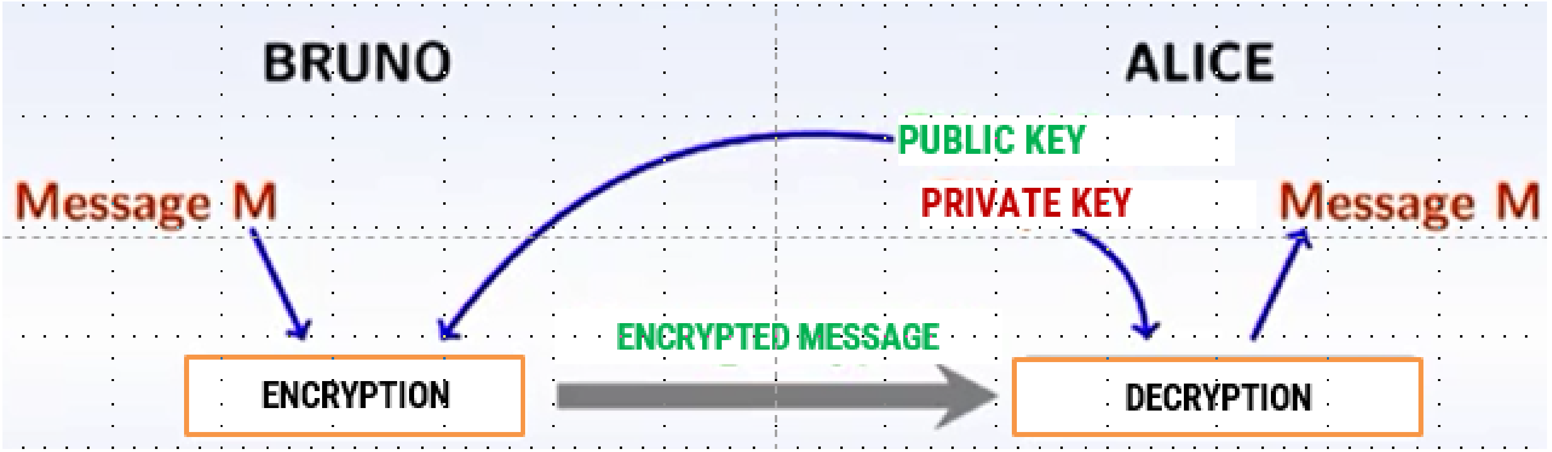
Calculations are done modulo an integer.

- **Decryption:**

Thanks to Fermat's Little Theorem.

RSA ENCRYPTION

- Encryption steps
  - Calculation of the public and private keys
  - Message encryption
  - Message decryption



RSA ENCRYPTION

Step 1: Keys preparation

Step 1.1: Choice of two prime numbers

Alice performs the following operations:

Choice of two distinct prime numbers ***p*** and ***q***

Calculation of $n = p \times q$

Calculation of $\varphi(n) = (p - 1) \times (q - 1)$

Example

- $p = 5$ et $q = 17$
- $n = p \times q = 85$
- $\varphi(n) = (p - 1) \times (q - 1) = \varphi(n) = 64$

RSA ENCRYPTION

Step 1: Keys preparation

Step 1.2: Choice of an exponent and calculate its inverse

Alice chooses an exponent e such that $gcd(e, \varphi(n)) = 1$

Alice calculates the inverse d of e modulo $\varphi(n)$ using the Extended Euclidean Algorithm: $d \times e \equiv 1 \pmod{\varphi(n)}$

Example

- $e = 5$ *and we have* $gcd(e, \varphi(n)) = gcd(5,64) = 1$
  - $5 \times 13 + 64 \times (-1) = 1$
  - *then* $5 \times 13 \equiv 1 \pmod{64}$
  - *the inverse of* e *modulo* $\varphi(n)$ *is* $d = 13$

RSA ENCRYPTION

Step 1: Keys preparation

Step 1.3: Public key

The public key of Alice is composed of two numbers: ***n*** and ***e***

Step 1.4: Private key

Alice keeps secret her private key: ***d***

According to the previous example:

$n = 85$ and $e = 5$

$d = 13$

RSA ENCRYPTION

Step 2: Message encryption

Step 2.1: Message

- Bruno wants to sent a secret message to Alice
- He transforms his message into one (or many) integers ***m***
- The integer m verifies $0 \leq m < n$

Example

$m = 10$

RSA ENCRYPTION

Step 2: Message encryption

Step 2.2: Encrypted message

- Bruno procures the public key of Alice: n and e
- He calculates the encrypted message $x \equiv m^e \pmod{n}$
- He transmits the message x to Alice

Example

- $m = 10$, $n = 85$ and $e = 5$
- $x \equiv m^e \pmod{n} \equiv 10^5 \pmod{85}$
  - $10^2 = 100 \equiv 15 \pmod{85}$
  - $10^4 = (10^2)^2 \equiv 15^2 \equiv 225 \equiv 55 \pmod{85}$
  - $10^5 = 10^4 \times 10 \equiv 55 \times 10 \equiv 550 \equiv 40 \pmod{85}$
- Then, The encrypted message is $x = 40$

RSA ENCRYPTION

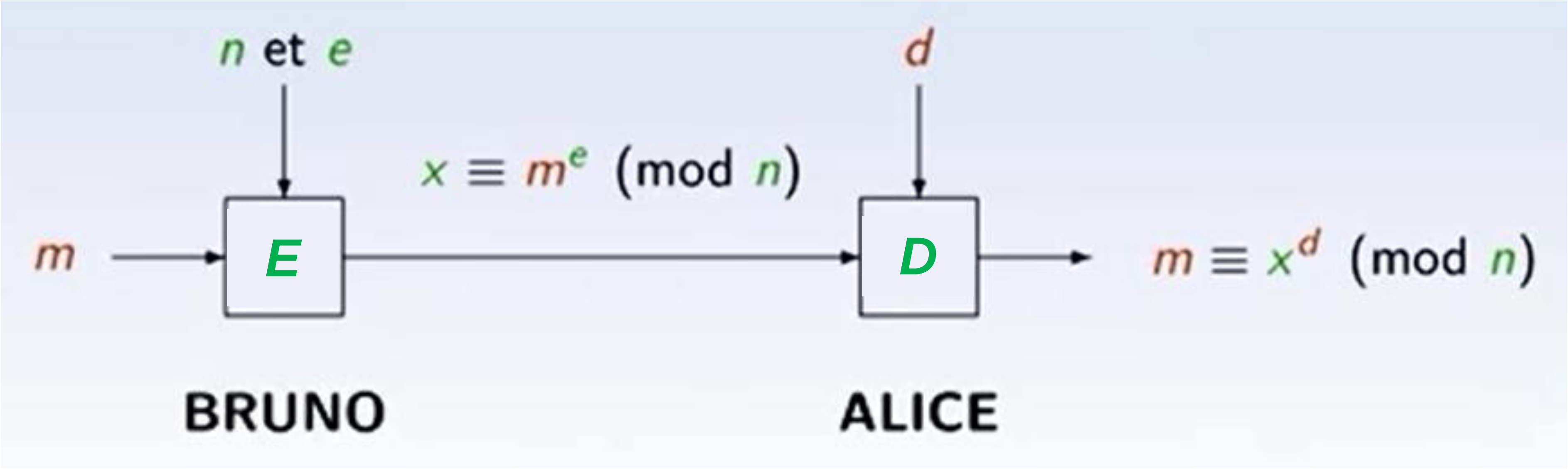
Step 3: Message decryption

- Alice receives the message n encrypted by Bruno
- Alice decrypts it using her private key d
- $m \equiv x^d \pmod{n}$

Example

- $x = 40,\ d = 13,\ n = 85$ $\quad 40^{13} \pmod{85}$
  - $40^2 = 1\,600 \equiv 70 \pmod{85}$
  - $40^4 = (40^2)^2 \equiv 70^2 \equiv 4\,900 \equiv 55 \pmod{85}$
  - $40^8 = (40^4)^2 \equiv 55^2 \equiv 3\,025 \equiv 50 \pmod{85}$
- $40^{13} \equiv 40^{8+4+1} \equiv 40^8 \times 40^4 \times 40 \equiv 50 \times 55 \times 40 \equiv 10 \pmod{85}$
- We find again the message $m = 10$ encrypted by Bruno

RSA ENCRYPTION



Security of RSA

- It is presumed difficult to deduce the private key (d) from the public key (n, e). If one could factorize n to find p and q, it would be possible to obtain the key d by using e, the public exponent. Thus, the security of RSA is dependent on the difficulty of the factorization problem.

- Since n is a very large number, it is very difficult to calculate its decomposition into prime factors.

- In practice, n is a number whose binary representation is on the order of 350 to 400 bits. Indeed, it is important to choose p and q carefully.