Introduction
Old encryption schemes

Luis J. Dominguez Perez

CRYPTO-CO, Junio 29 de 2016

Agenda 1

Historical encryption
- Attacks
- Enigma

Contemporary Encryption
- Introduction
- Cifradores de Flujo
- Block Encryption
- Random Numbers
- DES
- Modes of operation
- AES

Substitution

Also known as cypher by replacement, this is one of the simplest methods to encrypt plaintext.

The general idea is to substitute each letter f rom the alphabet by another (or in some cases, the same one), this way, the text cannot be understood on plain sight:

Ejemplo:

$$A \to L$$
$$B \to C$$
$$C \to J$$
$$\vdots$$

BABA = CLCL

Atacks to the substitution cypher

- Brute force, exhaustive search.

The attacker has the cyphertext since it may have listened to the conversation; additionally, she has part of the original text, for example: the header of the message (i.e. *%PDF-1.4, PK, GIF87a, 0xFFD8*)

Now, she only needs to try with all of the possible keys on the start of the text until she is successful.

Brute force attacks

Formally,

Basic exhaustive key search, or brute force attack

Given the pair (x, y), the plaintext, and the cyphertext, and let $K = \{k_0, \ldots, k_{n-1}\}$ the space of all of the possible keys. A brute force attack verifies on each $k_i \in K$ if:

$$d_{k_i}(y) \stackrel{?}{=} x,$$

If this logic relation holds true, then we have found the key, and the process stops, otherwise it continues.

$d(.)$ is the decryption function. In practice, this depends on the protocol.

Brute force attacks

In the principle, every *symmetric* encryption algorithm is susceptible to brute force attacks. The feasibility of this attacks depends on the key space (the number of possible keys).

For example, the PIN of the bank cards is of 4 digits, there are 10^4 possible PIN. In this ase, taking money from an ATM wouldn't take long if it weren't by the banks, who block a card after a small number of failed attempts.

Brute force attacks

Now, if launching an attack using a modern computers takes too much time (i.e., tenths of years), we can say that en cipher is *computationally safe* against brute force attacks.

In the caseof the substitution cipher, the letter A was substituted by the letter L, but we had 26 choices. The letter B was substituted by the letter C, from the 25 remaining options, and so on.

The number of possible substitutions in a brute force attack is:

Brute force attacks

Now, if launching an attack using a modern computers takes too much time (i.e., tenths of years), we can say that en cipher is *computationally safe* against brute force attacks.

In the caseof the substitution cipher, the letter A was substituted by the letter L, but we had 26 choices. The letter B was substituted by the letter C, from the 25 remaining options, and so on.

The number of possible substitutions in a brute force attack is:

$$26 \cdot 25 \cdots 1 = 26! \approx 2^{88}.$$

How much is 2^{88}?

- An Intel Core i7 @3.4 GHz CPU processor have around 2^{31} CPU cycles per second, on each of its 4 CPU cores...

CPU cycles

- To do 2^{32} CPU cycles, we need to double the number of CPU cycles than the previous levelr (2^{31}), this is, thise processor can do 2^{33} CPU cycles in total.
- we have $88 - 33 = 55$, then we need to duplicate the number of CPUs 55 veces.
- That's a geometric growth, such is the case of the Ambalappuzha Paal Payasam legend (chees and rice grains)

CPU cycles

- To do 2^{32} CPU cycles, we need to double the number of CPU cycles than the previous levelr (2^{31}), this is, thise processor can do 2^{33} CPU cycles in total.
- we have $88 - 33 = 55$, then we need to duplicate the number of CPUs 55 veces.
- That's a geometric growth, such is the case of the Ambalappuzha Paal Payasam legend (chees and rice grains)

- However, that's only 1 second. . .

- However, that's only 1 second. . . perhaps we need to analyse the information

For benchmarking purpose, some people disable the TurboBoost fcility of their CPUs to avoid mixed results due to thermal radiation, and other factors.

Then, we can safely say that this schemes is safe against brute force attacks...

- However, that's only 1 second. . . perhaps we need to analyse the information

For benchmarking purpose, some people disable the TurboBoost fcility of their CPUs to avoid mixed results due to thermal radiation, and other factors.

Then, we can safely say that this schemes is safe against brute force attacks... (unless the *brutus* was the one who have chosen the key)

A *dfferent* attack

In the brute force attack we take the cipher as a black box, without analyzing its internals.

The substitution cipher could be broken using an analytic attack.

The main weakness of the cipher is that each symbol in the plaintext has a unique representation in the ciphertext. This is, the statistics properties of the plaintext are preserved in the ciphertext.

Letters in the language

The most used letter in the English language is the letter “e” (around the 13% of all of the texts), then the letter “t” with a 9%, and the letter “a” with 8%.

In Spanish Castillian, the frequency is similar (the letter “e” is also the most used). One can construct a table for the Spanish language taking any book, and counting the number of times each letter appears in the text.

Letters in the language

The most used letter in the English language is the letter “e” (around the 13% of all of the texts), then the letter “t” with a 9%, and the letter “a” with 8%.

In Spanish Castillian, the frequency is similar (the letter “e” is also the most used). One can construct a table for the Spanish language taking any book, and counting the number of times each letter appears in the text.

Here we have them from the most frequent to the least frequent:
E A O S R N I D L C T U M P B G V Y Q H F Z J Ñ X W K

Notes about counting letters

- In a dictionary, the most frequent letter tends to be the letter "a"
- The previous order is preserved in a book, for example, el Quijote.
- There are exemptions
- There are mane phrases in Spanih that has the letter "e": qué, le, sé, etc.

Finally, with statistics it is easy to decrypt a ciphertext encrypted by substitution.

Cæsar encryption

The Cæsar encryption is a special type of cipher by substitution in which the values from the alphabet are rotated a fixed number of positions.

For example, if the key was 13, then, the substitution table is something like:

$$\begin{aligned} A &\to N \\ B &\to \tilde{N} \\ C &\to O \\ &\vdots \end{aligned}$$

For the Spanish language.

Cæsar encryption

The Cæsar encryption is a special type of cipher by substitution in which the values from the alphabet are rotated a fixed number of positions.

For example, if the key was 13, then, the substitution table is something like:

$$
\begin{aligned}
A &\to N \\
B &\to \tilde{N} \\
C &\to O \\
&\vdots
\end{aligned}
$$

For the Spanish language.

What is the key space?

Enigma machine

The enigma machine is a look-a-like typewriter machine with a fixed number of rotors. These rotor connected serially turn themselves in a different way to each typing on the keyboard, so that, the output of a given rotor is the input of the next one, at the end, a light displaying the final position is shown.

These rotors could change their initial position (which takes the roles of the key), then, each time a message is delivered, a new configuration is used.

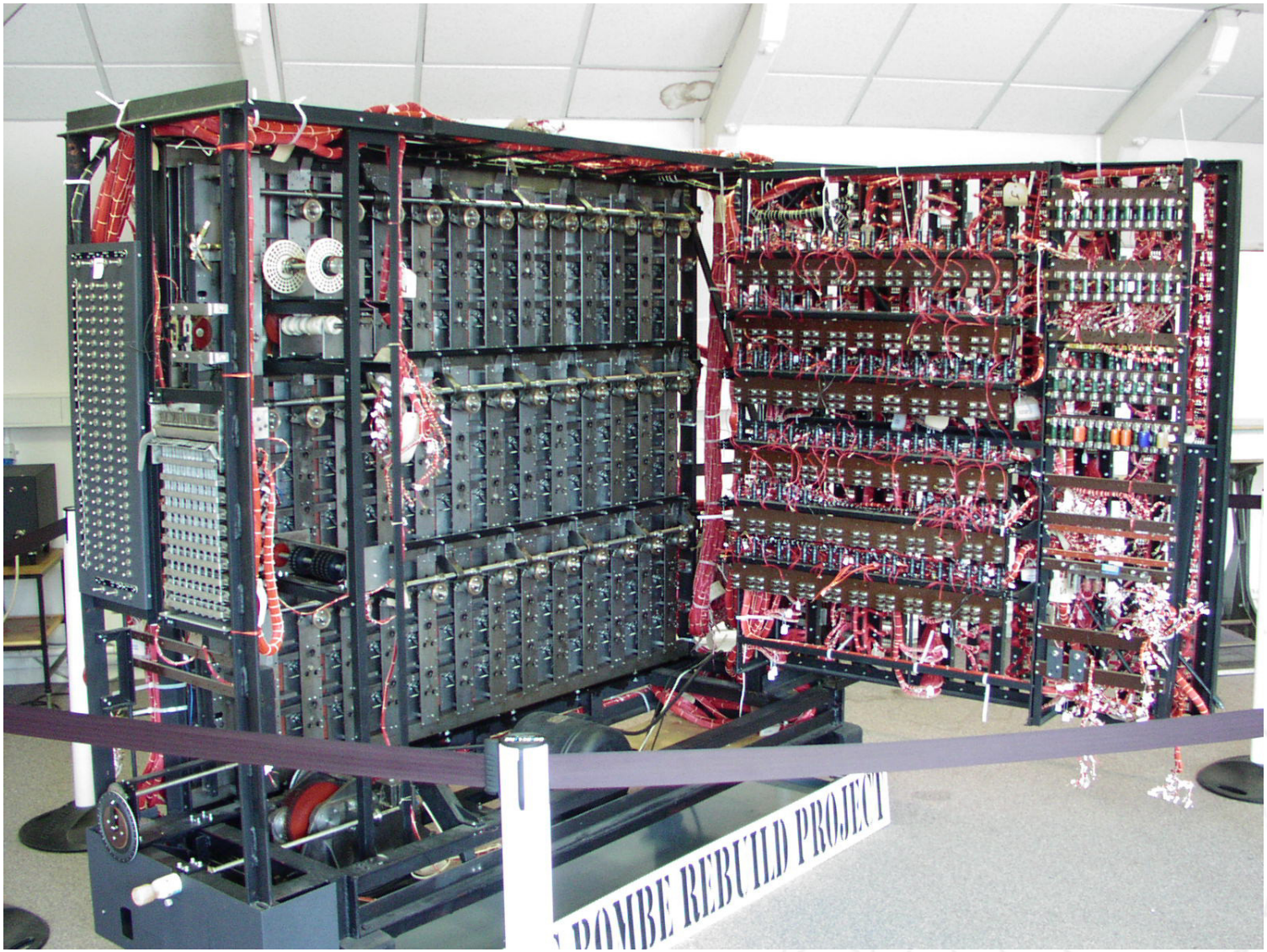
World War II

During the Francoist Spain, nazis provided a mini-Enigma machines to Francisco Franco in order to test them.

Once probed, they were extensively used during the World War II for order delivery to the front lines.

All of the German attacks were surprise attacks, and 100% effective. Some time latter, the British (and Polish) in the Bletchley Park (directed by Alan Turing), and the polish machines (the *bombes*, he allied could decrypt the messags, and saved millions of lives.

Polish Bombes



Enigma simulator

Online

Agenda 2

Historical encryption
- Attacks
- Enigma

Contemporary Encryption
- Introduction
- Cifradores de Flujo
- Block Encryption
- Random Numbers
- DES
- Modes of operation
- AES

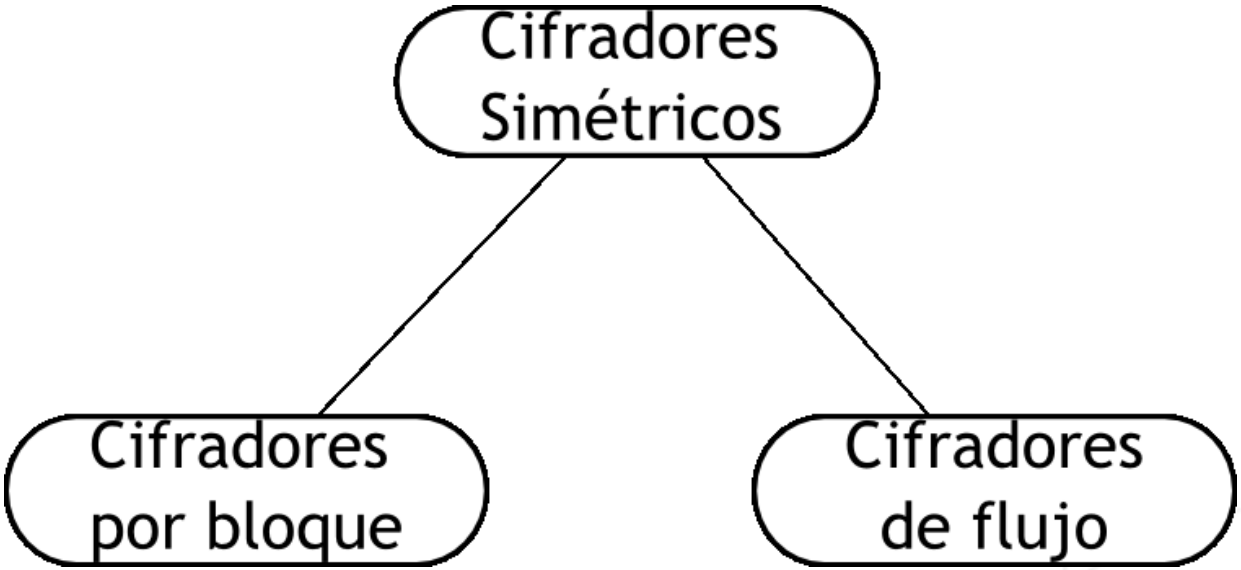
Cryptosystem

A *cryptosystem* is a 5-tupla $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, with the following restrictions:

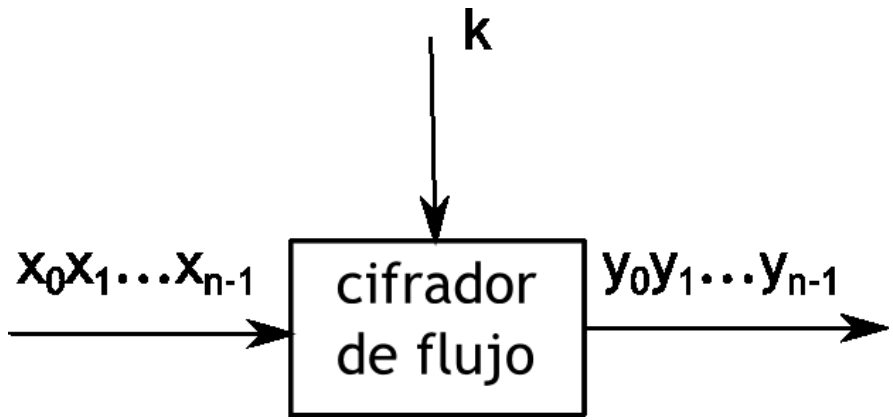
- $\mathcal{P}$ is the finite set of all of the possible plaintexts
- $\mathcal{C}$ is the finite set of all of the possible ciphertexts
- $\mathcal{K}$, the *key space*, is the finite set of al of the possible keys
- $\forall K \in \mathcal{K}$, $\exists E_K \in \mathcal{E}$ (encryption rule), $\exists D_K \in \mathcal{D}$ (decryption rule)

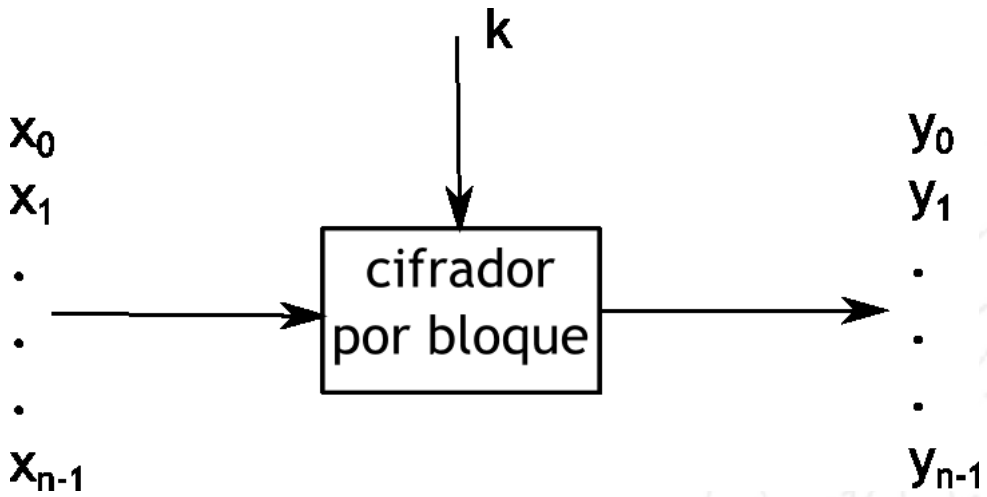
Each $E_K : \mathcal{P} \to \mathcal{C}$, $D_K : \mathcal{C} \to \mathcal{P}$, are functions such that $\forall x \in \mathcal{P}$, $D_K(E_K(x)) = x$.

Symmetric encryptors

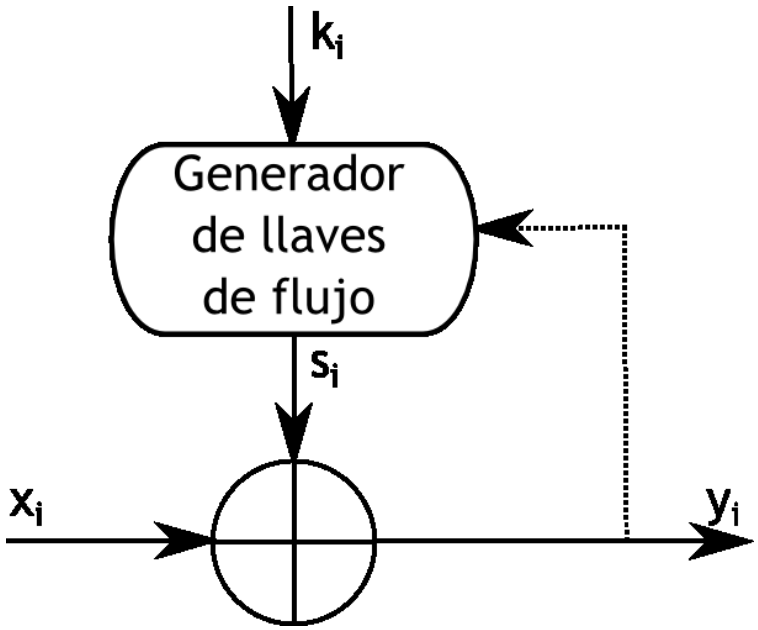


Types of symmetric encryptors





Diagram



Stream encryption.
Encrypt individual bits. This is done when adding a bit of the plaintext data stream to the key stream.

The *synchronous* schemes are the ones without the dotted line on the diagram (the encryption depends exclusively on the key). They are *asynchronous*, whern there is dependency of the previously encrypted bit (the dotted line is active).

Operation

Stream encryption, and decryption

The plaintext, the cophertext, and the key stream consist inindividual bits: $x_i, y_i, s_i \in \{0, 1\}$

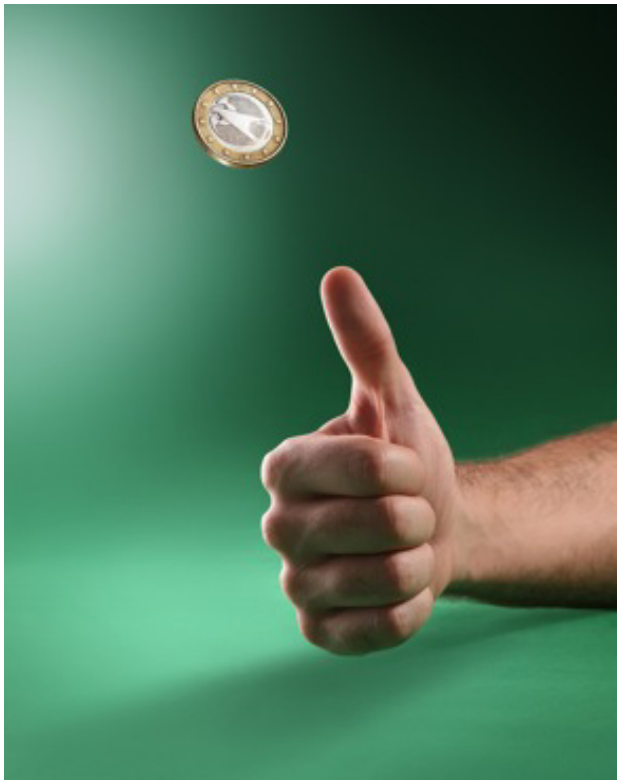
- Encryption: $y_i = e_{s_i}(x_i) \equiv x_i + s_i \bmod 2$
- Decryption: $x_i = d_{s_i}(y_i) \equiv y_i + s_i \bmod 2$

Note: Addition modulo 2 equals to the XOR operation.

Block encryptors

Block encryptors.
They encrypt a full block of bits from the plaintext in one operation with the key This means that the encryption of anybit of the plaintext inside the block depends on the other bits within the block. In practice, most of the block encryptors expect blocks of 128 bits (AES), or 64 bits(DES).

Random Numbers



Truly random number generators.

- The truly random number generators (TRNGs) have the property that their output cannot be reproduced.; for example, if we toss 100 coins, and keep a record of the results as a bit sequence, such a sequence cannot virtually be reproduced again (the probability of doing it again is $1/2^{100}$).
- The TRNGs are based on physical processes.

Pseudo-random numbers

Pseudo-random number generators.

- The Pseudo-random number generators (PRNGs) generate bit sequences that can be computed from an initial "seed" value.

For example, the ASCI C rand(.) function is something like:

$$s_0 = 12345$$
$$s_{i+1} \equiv 1103515245 \cdot s_i + 12345 \bmod 2^{31}, i = 0, 1, \ldots$$

Cryptographically secure pseudo-random number generator

A **cryptographically secure pseudo-random number generator** (CSPRNGs) is a special type of generator that is unpredictable. Given a bit sequence, there is no algorithm that determines the next bit with a probability above of 50%. Similarly, given a bit sequence, it is impossible to determine the previous bit.

This unpredictabilidad of the CSPRNGs is unique in cryptography, hence, we cannot take a random number generator not designed specifically for cryptography, since it may be of no commercial use in a final product.

Unconditionally secure

Unconditionally secure.
A cryptosystem is unconditionally secure (or secure in terms of the information security) if it cannot be broken regardless of the amount of computing resources.

Let suppose there is a symmetric cryptosystem with a key of 10000 bits that can only be broken using brute force. You may need 2^{10000} computers. This encryption scheme is not unconditionally secure, but it is computationally secure (we estimate there are between 2^{239}, and 2^{289} atoms in the visible universe)

One-time pad

Here we have an unconditionally secure cryptosystem:

One-time pad

This is a stream encryption scheme with the following properties:

- the key stream $s_0, s_1, \ldots$ is generated by a TRNGs
- the key stream is only known by the peers of the communication
- each bit in the key stream s_i is used only once, forever.

this is known as one-time pad. The one-time pad in unconditionally secure.

More about the one-time pad.

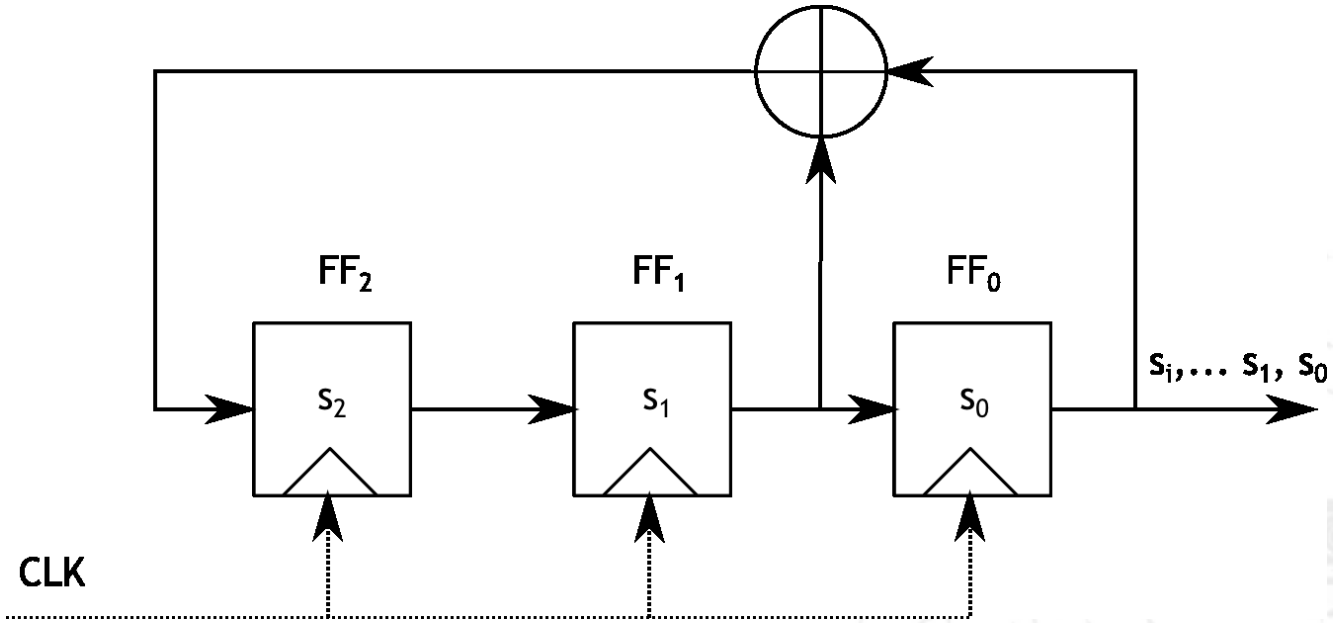
Each bit of the ciphertext is computed in the following way:

$$\begin{aligned} y_0 &\equiv x_0 + s_0 \bmod 2 \\ y_1 &\equiv x_1 + s_1 \bmod 2 \\ &\vdots \\ y_{n-1} &\equiv x_{n-1} + s_{n-1} \bmod 2 \end{aligned}$$

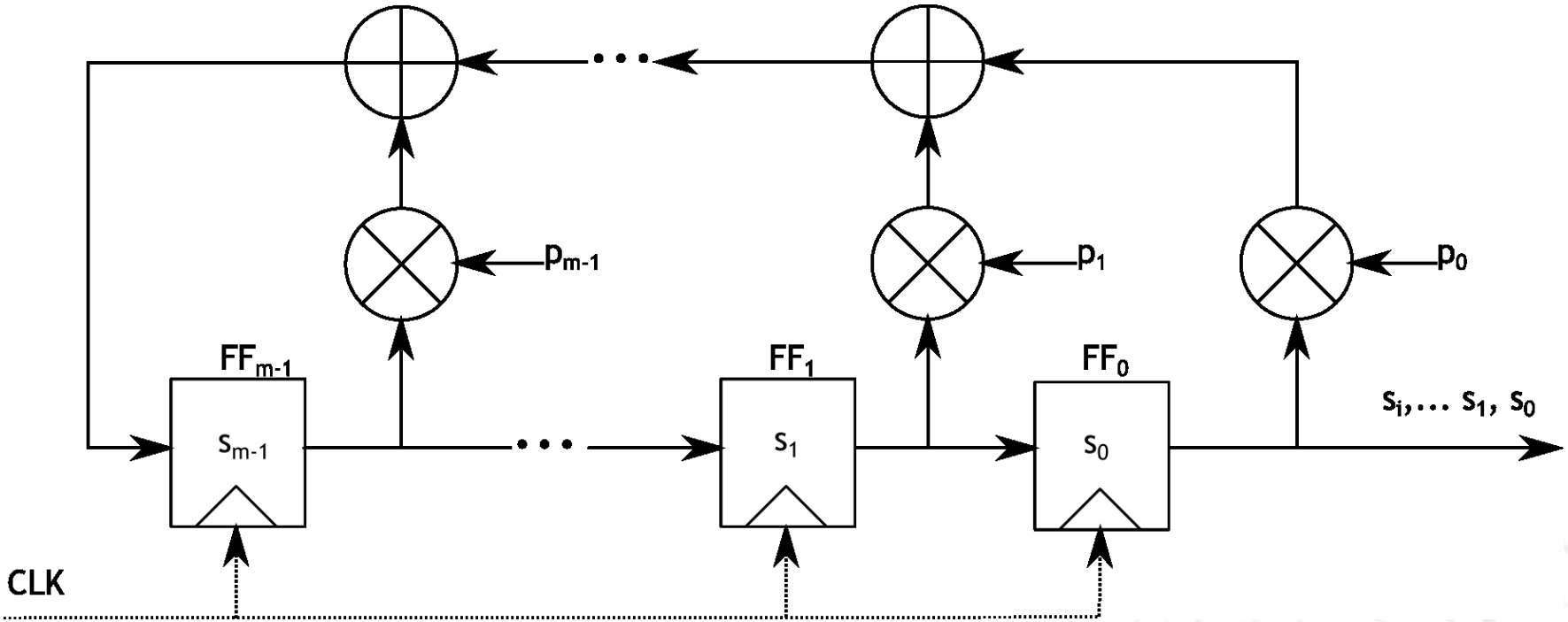
this is an equation with 2 unknown variables per bit. Even if we known y_i, the $x_i \in \{0, 1\}$ values have exactly the same probability if we used a TRNG; however, since the random bits cannot be reused, we have now the key distribution problem.

Linear Feedback Shift Register

A **LFSR** consist on synchronize storogin elements (flip-flops), and a feedback route. The number of storing elements define the degree of the LFSR. The feedback network computes the input of the last flip-flop as an addition modulo 2 (XOR) from certain flip-flops in the diagram.

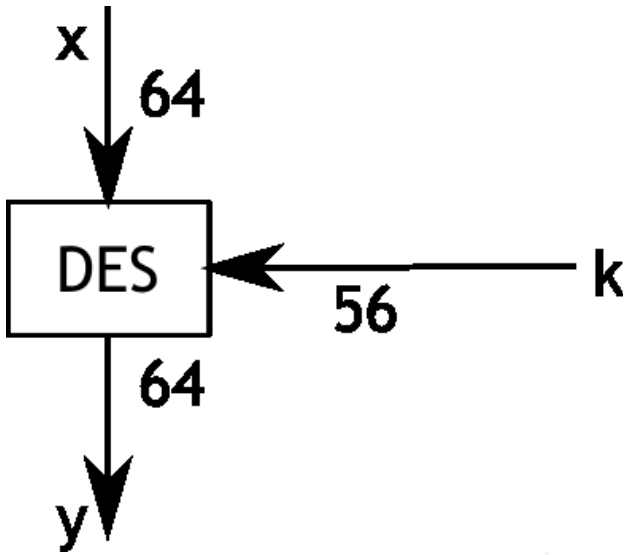


LFSR

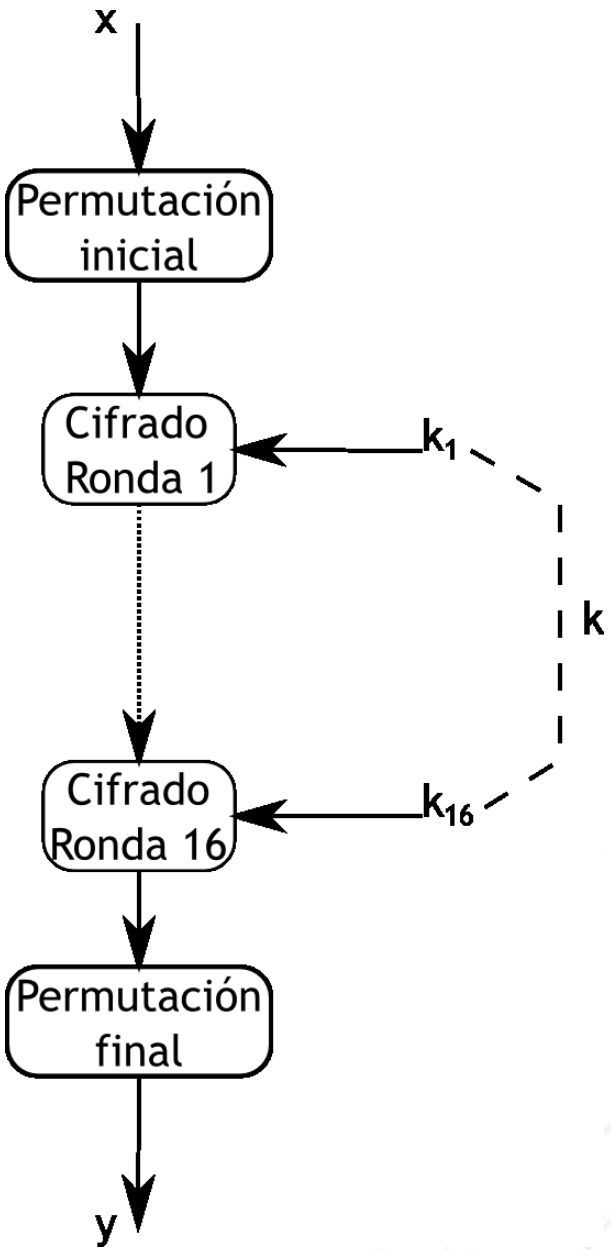


DES black box

The DES is an encryption method that takes 64-bits length blocks, and a key of 56-bits.



DES general diagram



Attacks

- 1977 - Diffie, and Hellman suggested a VLSI chip design that could theoretically test 10^6 keys/sec. An equipment with 10^6 of these circuits could break the key in about 10 hours. Cost: USD $20'000,000.00
- 1990 - Eli Biham, and Adi Shamir suggested a differential cryptoanálisis
- 1993 - Mitsuru Masui suggested a linear cryptoanálisis

Modes of operation

We already know DES works with 64-bits blocks, what if we need to encrypt more than 64 bits? Can We take blocks of 64 bits?

Modes of operation

We already know DES works with 64-bits blocks, what if we need to encrypt more than 64 bits? Can We take blocks of 64 bits? (no)

Modes of operation in block

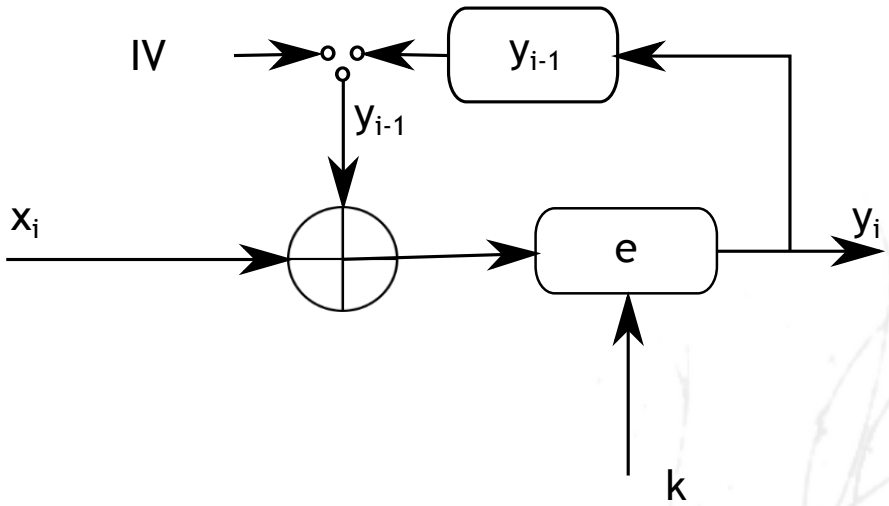
- ECB - Electronic Codebook Block
- CBC - Cipher Block Chaining

Modes of operation in stream

- CFB - Cipher Feedback
- OFB - Output Feedback

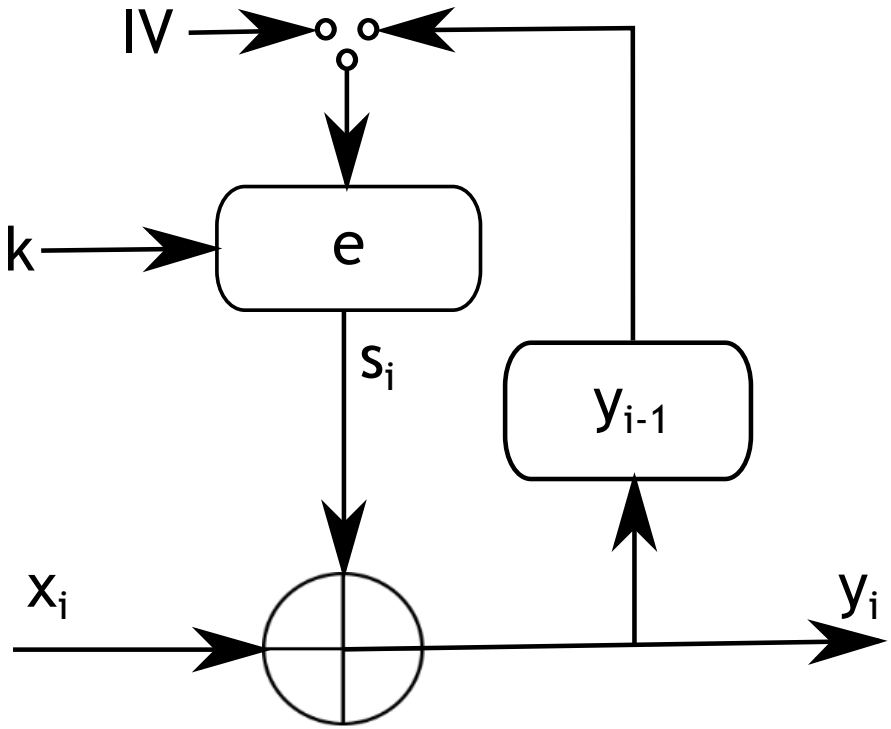
Blocks

- ECB - The message is broken in 64 bits pieces (we complete with zeros).
- CBC - Computes an XOR with the previous block output to encrypt (requires an initialization vector).

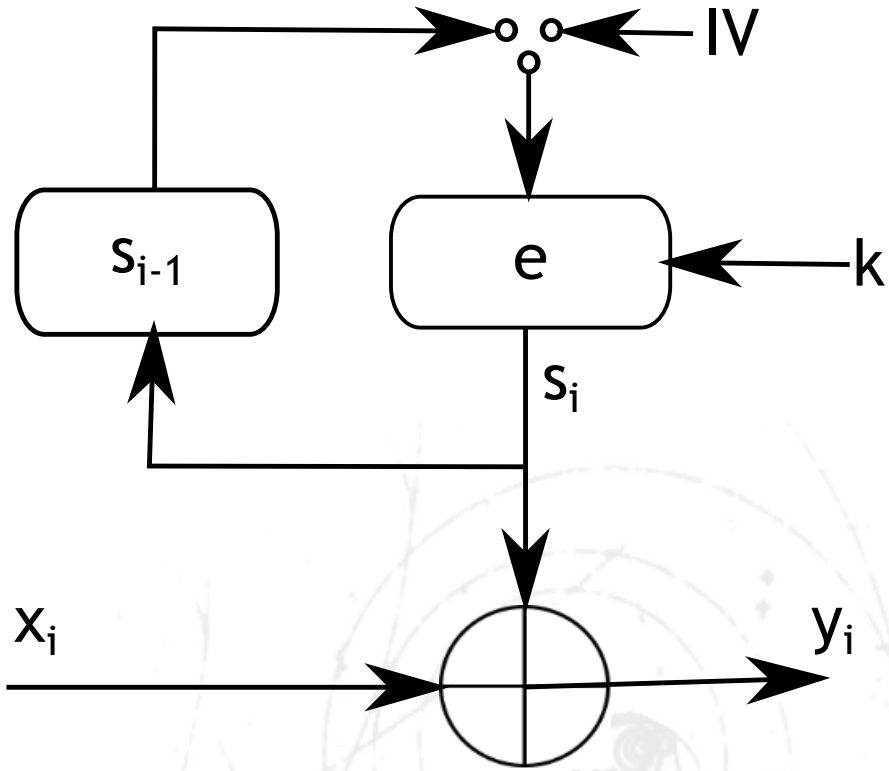


Stream

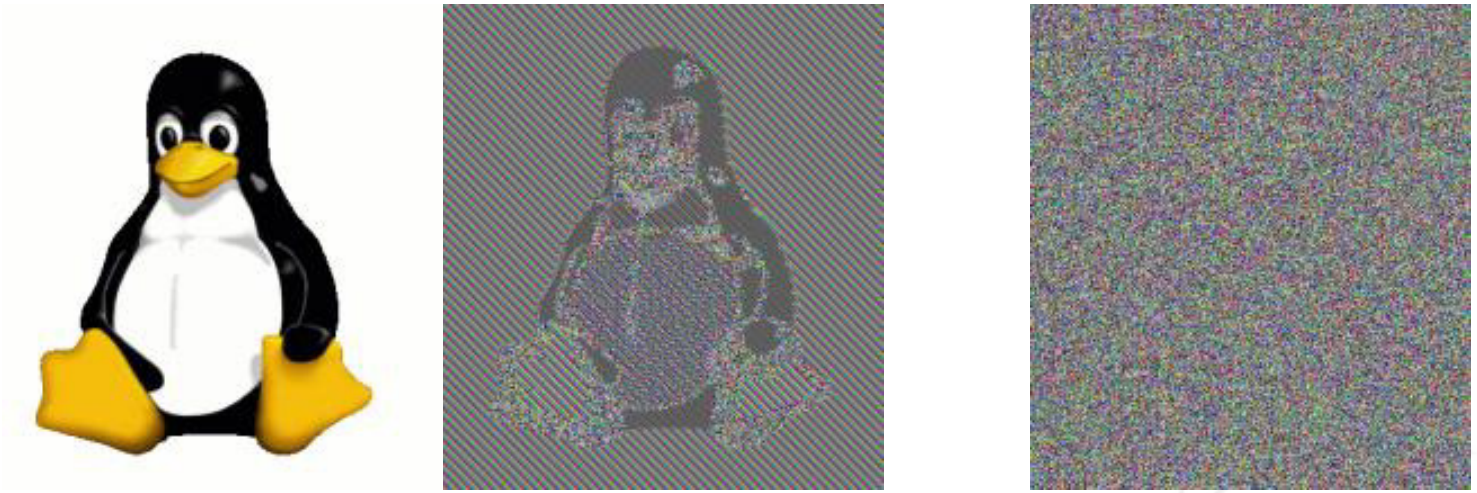
CFB - Computes an XOR with the previous output to encrypt



OFB - The feedback is independent from the actual message



Comparison

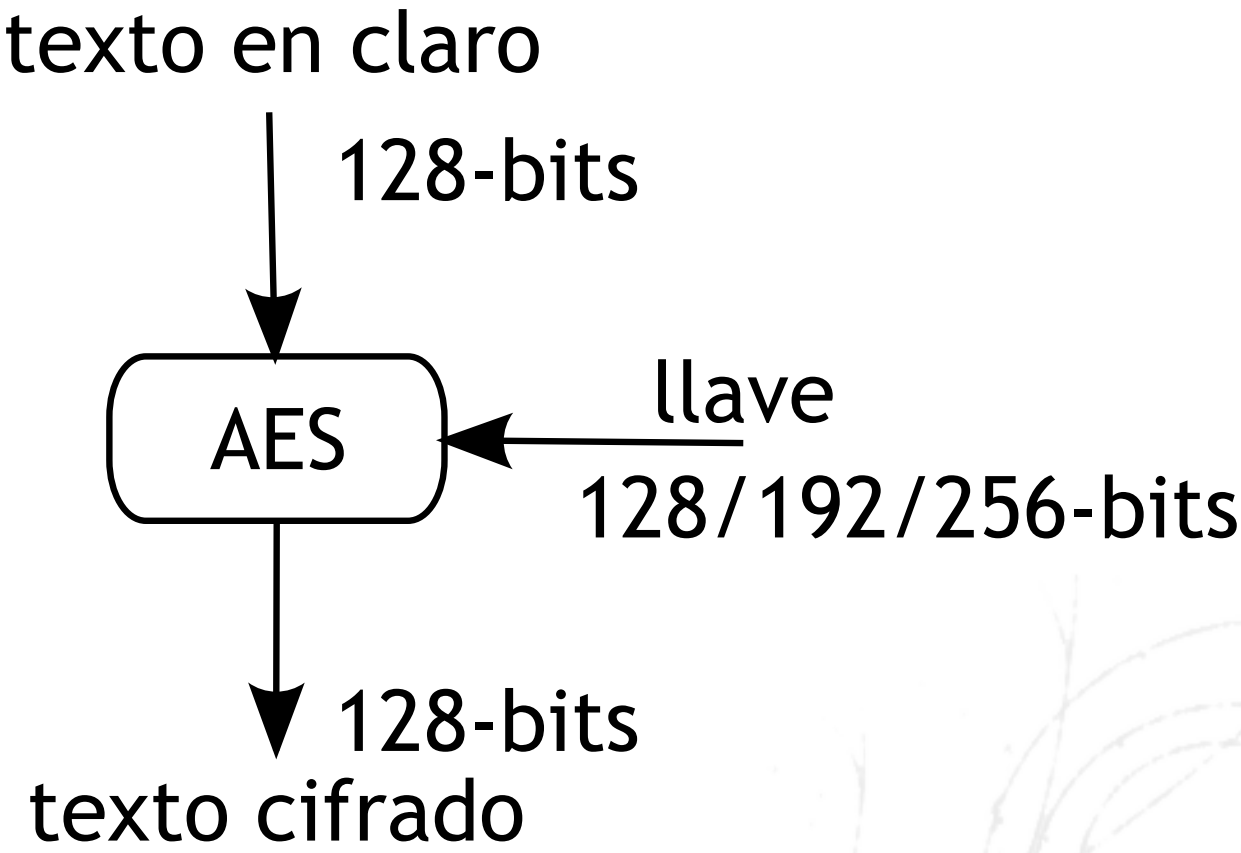


AES - Advanced Encryption Standard

Why a new standard?

- Brute force attacks
- Previous enhancement (Triple DES) is three times slower
- DES is only efficient in hardware
- There are new types of attacks
- Using 64-bit blocks is no longer useful in most situations

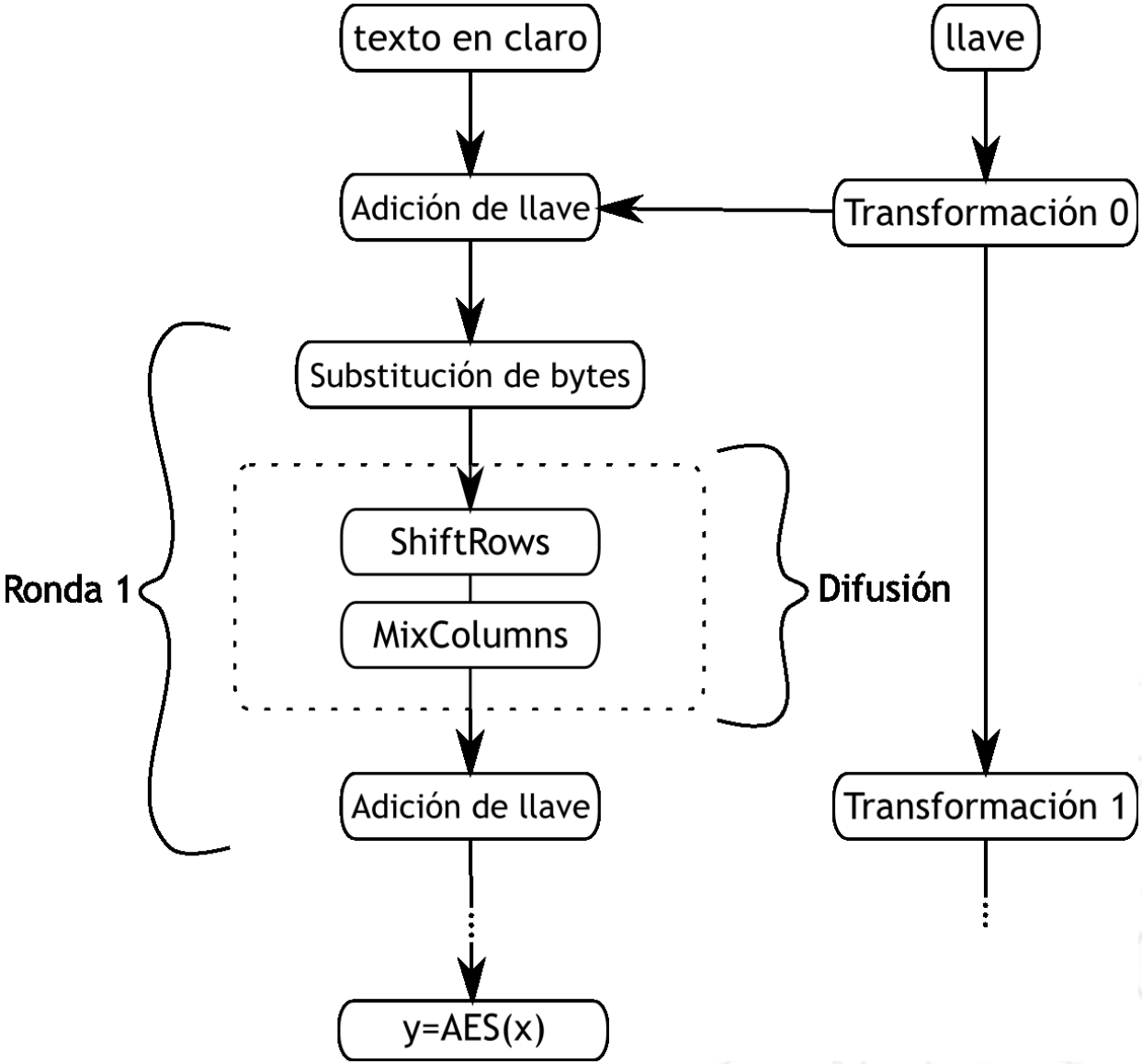
AES - Black box



AES - description

- AES does not uses a Feistel function, we encryt a full block per round
- We need 10, 12 o 14 rounds to encrypt data using keys of 128, 192 or 256 bits
- Each round has 3 layers: Key addition (inclusion), Bytes substitution, and Diffusion
- The difussion layers is ubdivided in: ShiftRow, that permutes byte-level data; and MixColumn, that mix 4-bytes long blocks inside a matrix

AES - rounds


texto en claro
llave
Adición de llave
Transformación 0
Substitución de bytes
ShiftRows
MixColumns
Ronda 1
Difusión
Adición de llave
Transformación 1
y=AES(x)