

Criptología

Dr. Aldo Juárez

Matemáticas

Teoría de la
Información

Teoría de
Códigos

Criptología

Criptología

kryptos = secreto + logos = ciencia

Ciencia que trata los problemas teóricos relacionados con la seguridad en el intercambio de mensajes en clave entre un emisor y un receptor a través de un canal de comunicaciones (en términos informáticos, ese canal suele ser una red de computadoras).

Criptología

Criptografía

Criptoanálisis

Criptografía

kriptos = secreto + *gráhos* = escritura

Es la disciplina que se encarga del estudio de códigos secretos o llamados también códigos cifrados.

Es decir, se encarga del cifrado de mensajes en clave y del diseño de criptosistemas*

(hablaremos de éstos más adelante)

Criptografía

En un sentido más amplio, la **criptografía** es la ciencia encargada de diseñar funciones o dispositivos, capaces de transformar **mensajes en claro o legibles** a **mensajes cifrados** de tal manera que esta transformación (**cifrar**) y su transformación inversa (**descifrar**) sólo pueden ser factibles con el conocimiento de **una o más llaves**.

Criptografía

La criptografía busca mantener la **confidencialidad** (el mensaje es accesible únicamente por destinatario autorizado) de los datos, además de garantizar la **autenticación** de los mismos (el emisor del mensaje es quien dice ser, y no otro), su **integridad** (el mensaje que leemos es el mismo que nos enviaron) y su **no repudio** (el emisor no puede negar el haber enviado el mensaje).

Cifrar o Encriptar

- En criptografía, **cifrar** o **encriptar** es el proceso de codificar un mensaje o información de modo tal que solo los individuos autorizados sean capaces de acceder a este, y aquellos que no estén autorizados no puedan hacerlo.
- Este proceso convierte la representación original de la información, conocida como **texto plano** (**mensajes en claro** o **legibles**), en una forma alternativa conocida como **texto cifrado**.

Criptología

Criptografía

Criptoanálisis

Criptoanálisis

Es la ciencia que estudia los métodos que se utilizan para, a partir de uno o varios **mensajes cifrados**, **recuperar** los **mensajes en claro** en ausencia de la(s) llave(s) y/o **encontrar** la llave o llaves con las que fueron cifrados dichos mensajes.

Es decir, trata de descifrar los mensajes en clave, rompiendo así el criptosistema.

Criptosistema

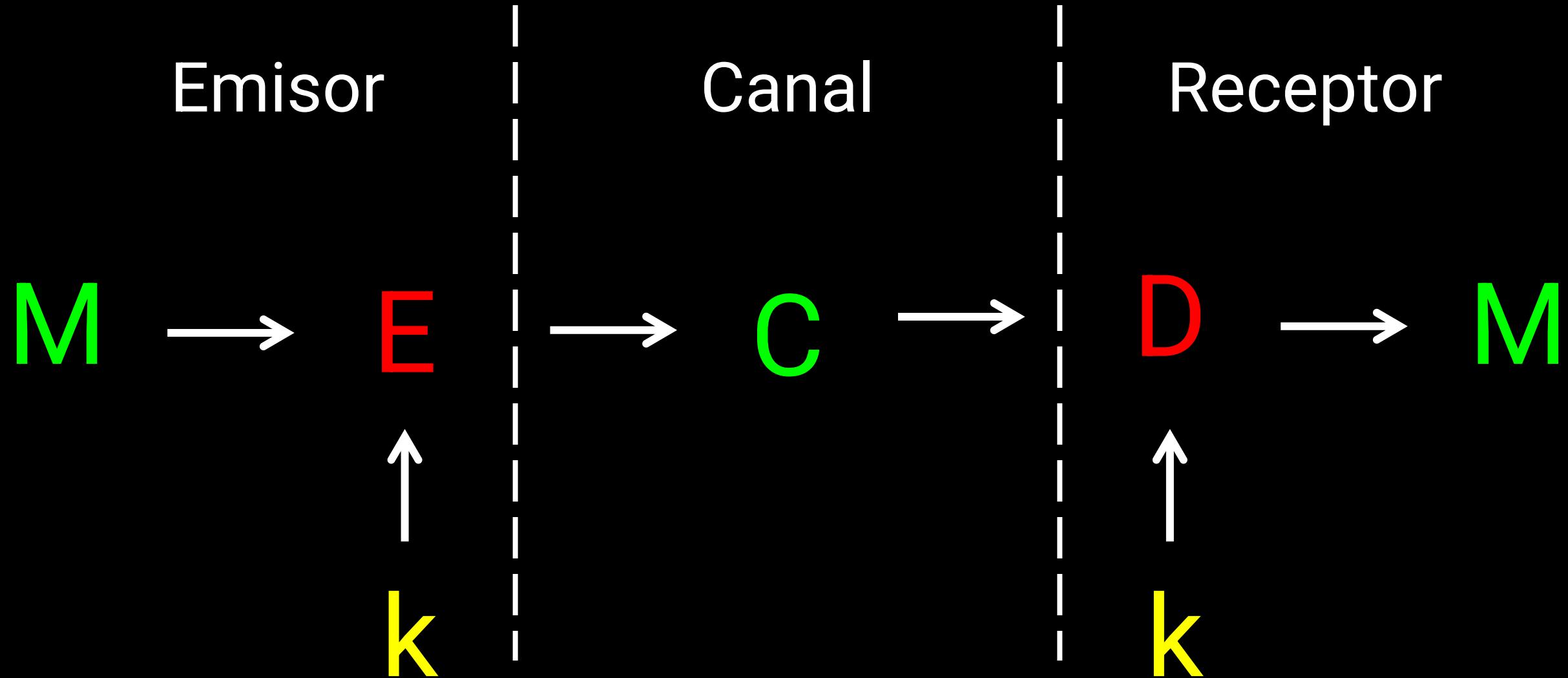
Es un sistema que toma información entendible y mediante un proceso definido lo convierte en un mensaje completamente distinto y en teoría incompresible para todos los que no conozcan como interpretarlo.

Criptosistema

Se define como una quíntupla (M, C, K, E, D) , donde:

- M conjunto de todos los mensajes sin cifrar (texto claro o plano *plain text*) que pueden ser enviados.
- C conjunto de todos los posibles mensajes cifrados o **criptogramas**.
- K conjunto finito de k claves (llaves) que se pueden emplear.
- E conjunto de transformaciones de **encriptado (cifrado)** o familia de funciones que se aplica a cada elemento de M para obtener un elemento de C . Para cada $k \in K$ existe una regla de cifrado $e_k \in E$.
- D es el conjunto de transformaciones de **descifrado**, análogo a E . Para cada $k \in K$ existe una regla de **descifrado** $d_k \in D$.

Criptosistema



Criptosistema

Si se tienen las funciones:

$$e_k : M \rightarrow C \text{ y } d_k : C \rightarrow M$$

Luego se cumple que:

$$d_k(e_k(M)) = M$$

Esto nos indica que si el texto original M es **encriptado** usando la función e_k ; y el texto **cifrado** C resultante es **descifrado** o **desencriptado** usando la función d_k , el resultado será el texto original M .

Esteganografía

- El mensaje o texto plano puede “ocultarse” de dos formas:

Criptografía vs. Esteganografía

- La esteganografía “**oculta**” la existencia del mensaje mientras que la criptografía lo vuelve ininteligible a ojos intrusos.
- Algunas técnicas:
 - Marcado de caracteres, pinturas, tinta invisible, fotos digitales, etc.

Esteganografía

- La esteganografía se puede definir como la ciencia de **ocultar los datos** como un archivo, imagen, video o cualquier mensaje en otro archivo, imagen, video o mensaje. En esteganografía, los **bits inútiles son reemplazados por bits útiles** para ocultar el archivo requerido en cualquiera de los archivos o datos mencionados anteriormente.

Ventajas/Desventajas de la Esteganografía

Desventajas

- Mucho overhead para ocultar pocos bits de info.
- Una vez que se “descubre” el sistema es totalmente inútil (igual que si obtienen la clave secreta bajo criptografía simétrica).

Se puede primero encriptar y luego “esteganografiar”.

Ventajas

- El mensaje oculto pasa desapercibido.

Historia de la criptografía

```
graph TD; Cripto[Criptografía] --> Clasica[Clásica]; Cripto --> Moderna[Moderna]
```

Criptografía

Clásica

Moderna

Criptografía Clásica

Transposición o
permutación

Grupos

Escítalá

Series

Columnas

ÜBCHI

Filas

Monoalfabético

Monográfica

Atbash, César,
Polybios ROT13, AFIN

Poligrámica

Playfair, Hill

Sustitución

Polialfabético

No periódica

Alberti, Vernam

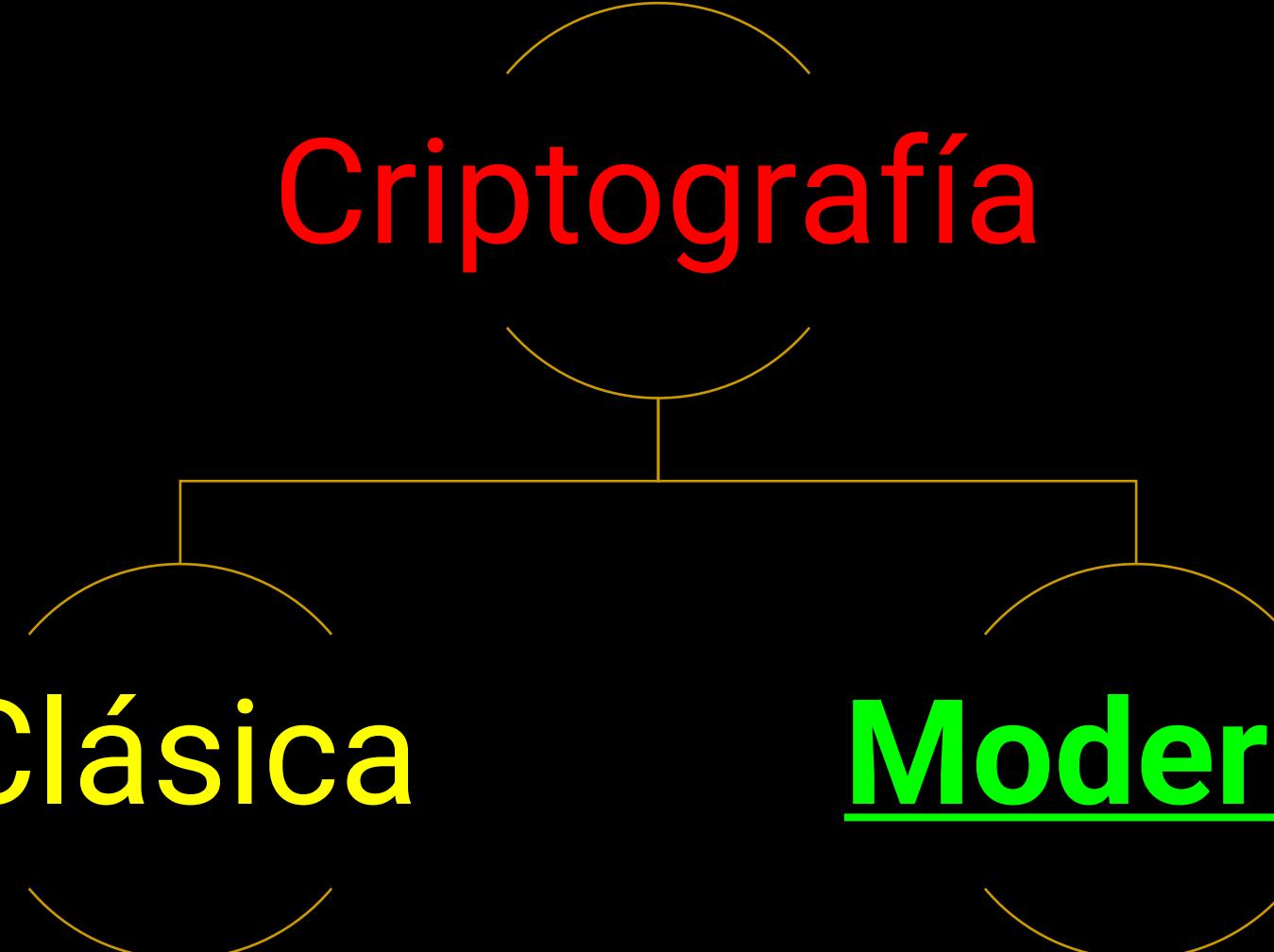
Periódica

Vigenère, Enigma

Mixto

ADFGVX

Historia de la criptografía



Criptografía

Clásica

Moderna

Criptografía moderna (Historia)

Dos hechos significativos marcan el comienzo en el mundo de la criptografía moderna. El primero de ellos, los estudios realizados por Claude Shannon (considerado el padre de la criptografía matemática) sobre la teoría de la información y criptología (1948).

Criptografía moderna (Historia)

- El mérito de Shannon fue el de desarrollar una teoría general de los criptosistemas basándose en modelos matemáticos, y sobre todo, el ser el primero en poder presentar estos trabajos a la luz pública. Sus trabajos solo se conocieron años después, una vez desclasificados.

Criptografía moderna (Historia)

El **segundo hecho** es la publicación de un artículo realizado por **Whitfield Diffie** y **Martin Hellman** (1976) en el que proponen un nuevo método de cifrado, creando criptosistemas de clave pública.

Criptografía moderna (Bases)

Los conceptos matemáticos básicos de las **adiciones**, las **multiplicaciones**, la **aritmética modular**, la **teoría de números** y las **operaciones XOR** son indispensables para entender el funcionamiento de los principales algoritmos de cifrado y descifrado.

Criptografía moderna (Reglas de Kerckhoff)

- No debe existir ninguna forma de recuperar mediante el criptograma el texto plano o la clave.
- Un criptosistema debe estar compuesto por dos tipos de información:
 - Pública: el algoritmo o familia de algoritmos que lo definen.
 - Privada: la clave utilizada para cada cifrado en particular.
- La forma de escoger la clave debe ser fácil de recordar y de modificar.
- Debe ser factible la comunicación del criptograma con los medios de transmisión habituales.
- La complejidad del proceso de recuperación del texto plano debe corresponderse con el beneficio obtenido.

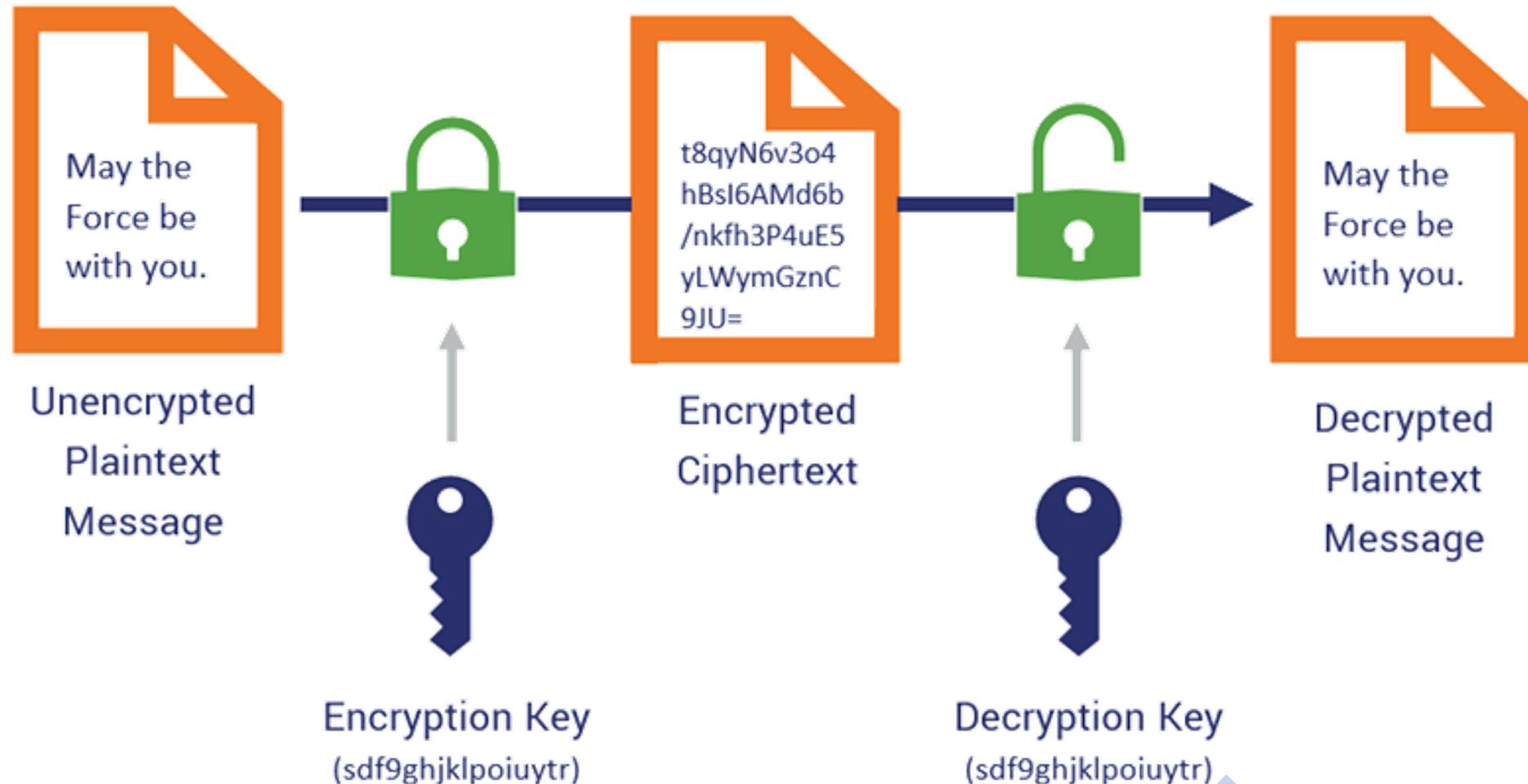
Criptografía moderna

**Cifrado
Simétrico**

**Cifrado
Asimétrico**

**Funciones
Hash**

How Symmetric Encryption Works



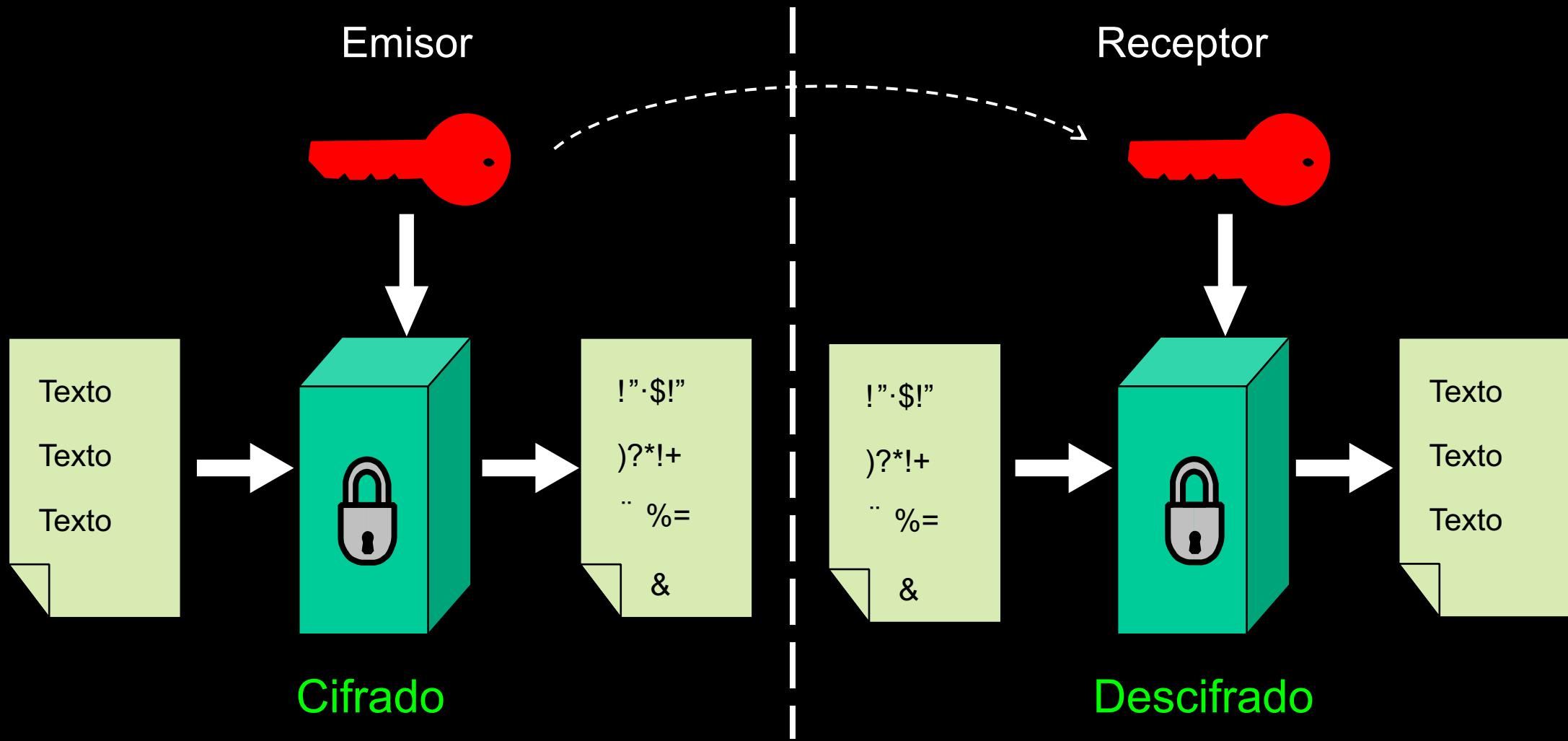
Cifrado Simétrico

Para poner esto en los términos más simples posibles, **cifrado simétrico** es un tipo de cifrado que usa **la misma clave** para cifrar y descifrar datos. Tanto el remitente como el destinatario tienen copias idénticas de la clave, que mantienen en secreto y no comparten con nadie.

Cifrado Simétrico

1. El remitente utiliza una clave de cifrado (generalmente una cadena de letras y números) para cifrar su mensaje.
2. El mensaje cifrado, llamado texto cifrado o criptograma, parece letras codificadas y nadie puede leerlo en el camino.
3. El destinatario utiliza una clave de descifrado para transformar el texto cifrado nuevamente en texto claro o plano.

Funcionamiento del cifrado simétrico



Cifrado Simétrico

Solo estas dos partes (remitente y destinatario) pueden leer y acceder a los datos. Es por eso que a veces también se llama **cifrado de clave secreta**, **criptografía de clave secreta**, **criptografía de clave privada**, **criptografía simétrica** y **cifrado de clave simétrica**.

Cifrado Simétrico

Tener solo una clave para servir tanto para las funciones de cifrado como de descifrado simplifica el proceso de encriptación.

Una de las **ventajas** de usar cifrado simétrico es que proporciona privacidad y confidencialidad de datos sin la complejidad adicional de múltiples claves.

Cifrado Simétrico

Cifrado por flujo

Cifrado por bloques

Cifrado por bloques vs por fujo

Una analogía sencilla para entender mejor el **cifrado por bloques** frente al **cifrado por flujos**, imagina que estás cifrando un libro. Puedes cifrar el contenido **página a página** (**cifrado por bloques**) o **letra a letra** (**cifrado por flujos**).

Cifrado por bloques vs por fujo

El **cifrado por bloques** y el **cifrado por flujos** son dos métodos distintos de cifrar datos con **algoritmos de cifrado simétricos**:

Cifrado de información en trozos. Un **cifrado por bloques** divide los mensajes de texto plano en bloques de tamaño fijo antes de convertirlos en texto cifrado mediante una clave.

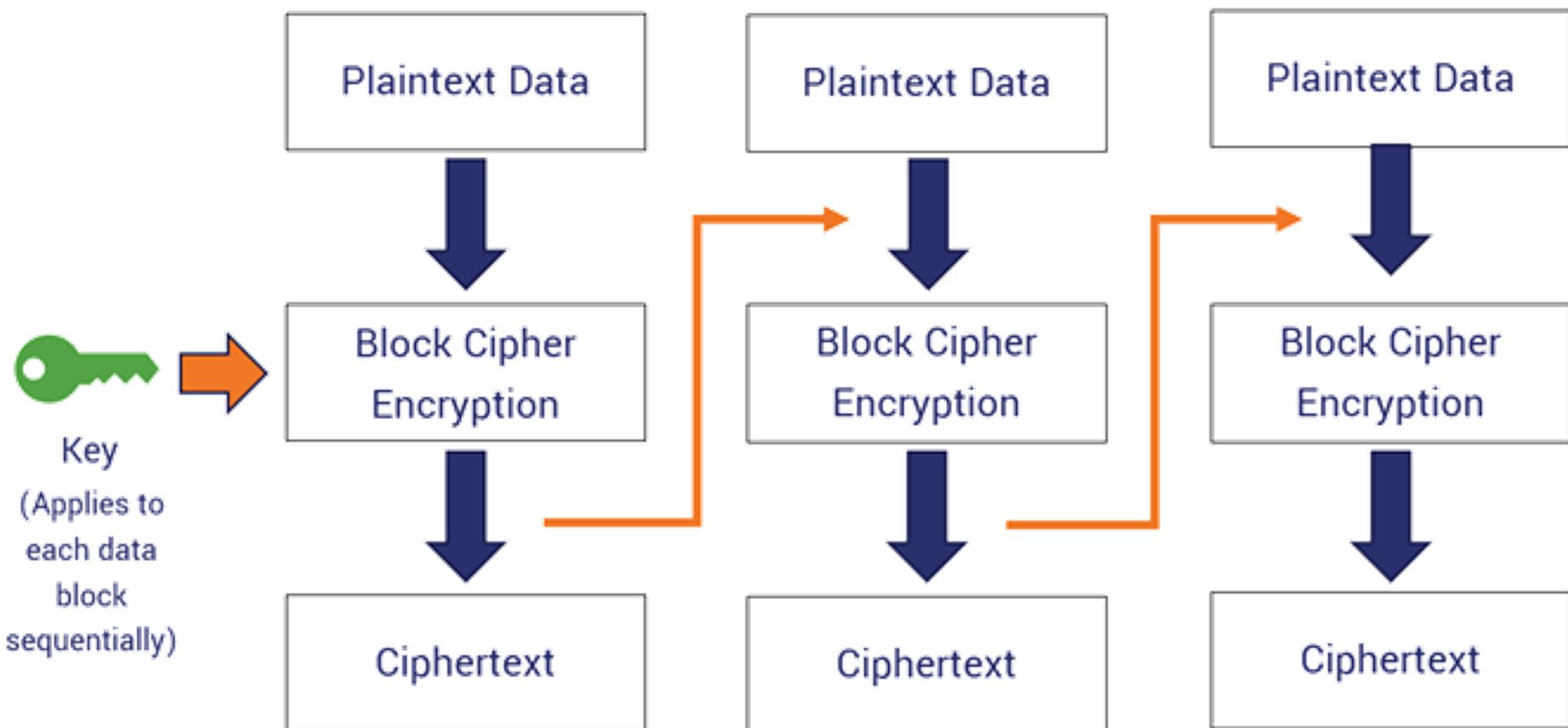
Cifrado de información bit a bit. Un **cifrado de flujo**, por el contrario, descompone un mensaje de texto plano en bits individuales, que luego se convierten individualmente en texto cifrado utilizando bits de clave.

Cifrado Simétrico

Cifrado por flujo

Cifrado por
bloques

How a Basic Block Cipher Chaining Operation Works



Cifrado por bloques

La idea básica de un **cifrado por bloque** es dividir el texto en bloques relativamente largos, normalmente de **64** o **128 bits**, y codificar cada bloque por separado. Se utiliza la **misma clave** de cifrado por cada bloque y es la clave de cifrado la que determina el orden en el que se llevan a cabo la sustitución, el transporte y otras funciones matemáticas en cada bloque.

Cifrado por bloques

El descifrado es similar, se ingresan bloques de texto cifrado y se producen bloques de texto plano.

Para cifrar mensajes más largos que el tamaño del bloque, se utiliza un modo de operación.

Cifrado por bloques

Cuando se utiliza una clave para el **cifrado simétrico**, la **longitud del texto plano** a cifrar suele ser **mucho mayor** que la **longitud de la clave**.

Así que para permitir que el algoritmo de cifrado se utilice en mensajes de diferentes longitudes el algoritmo de encriptación dividirá el texto plano o cifrado en bloques y los procesará secuencialmente.

Cifrado por bloques

El tamaño de los bloques será exactamente la longitud de la clave (menos los dígitos de control, etc.). El algoritmo de cifrado simétrico procesa estos bloques secuencialmente, y existen cinco modos de cifrado por bloques.

Modos cifrado
por bloques



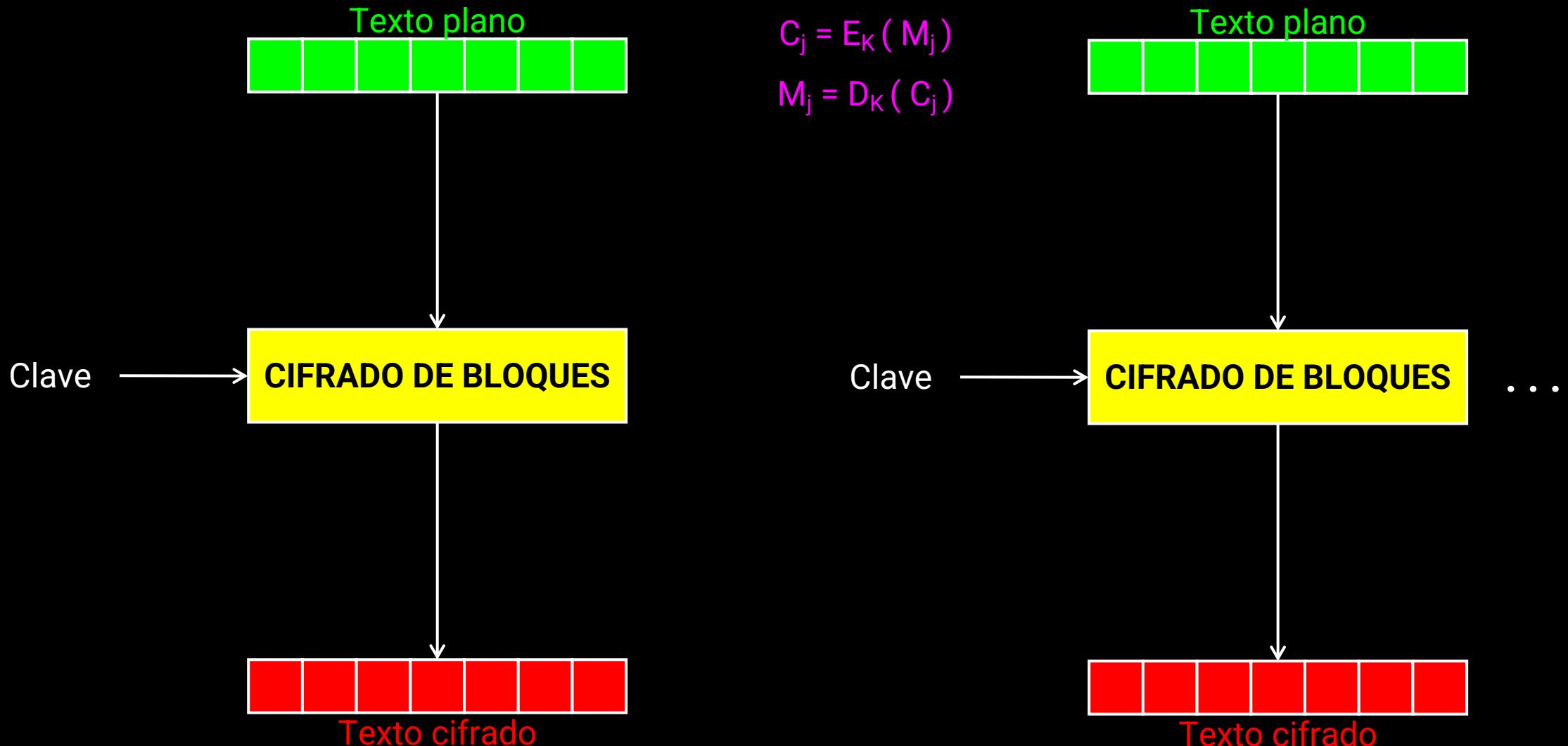
Modos de cifrado por bloques

Un modo de operación describe cómo aplicar repetidamente la operación de bloque único de un cifrado para transformar de forma segura cantidades de datos más grandes que un bloque.

Modos cifrado
por bloques



Modo ECB (Electronic Code Book)



Modo ECB (Electric CodeBook)

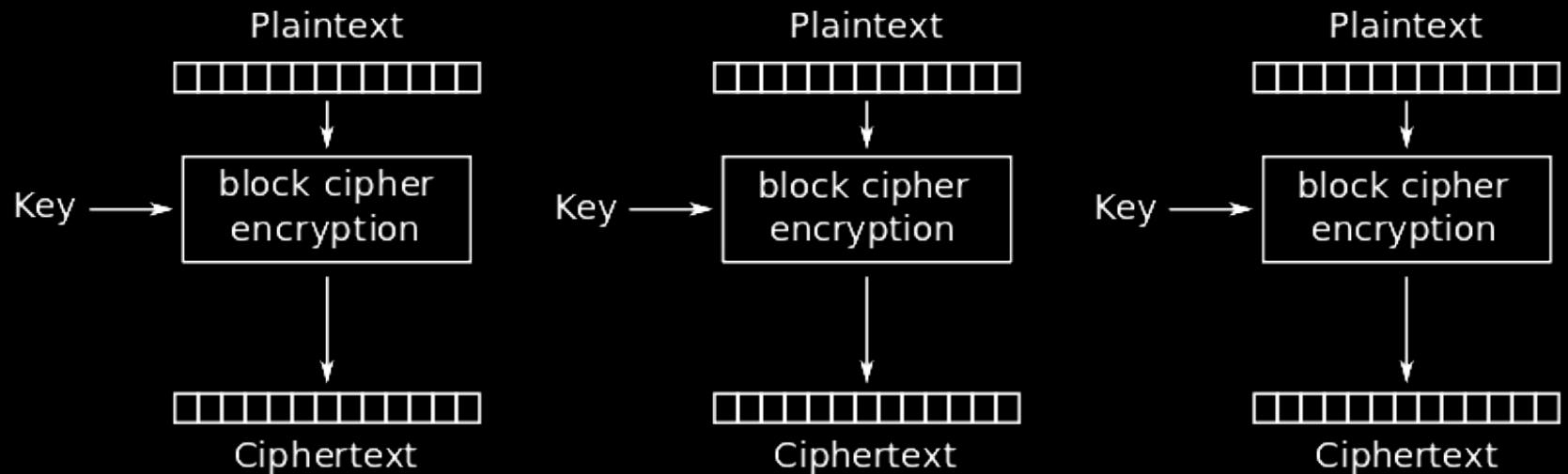
Se trata del modo más simple (y débil), que cifra los bloques por separado. No está diseñado para que los nuevos bloques dependan de la salida de los bloques anteriores. Esto significa que puedes cifrar o descifrar bloques de datos de forma independiente.

Modo ECB (Electric CodeBook)

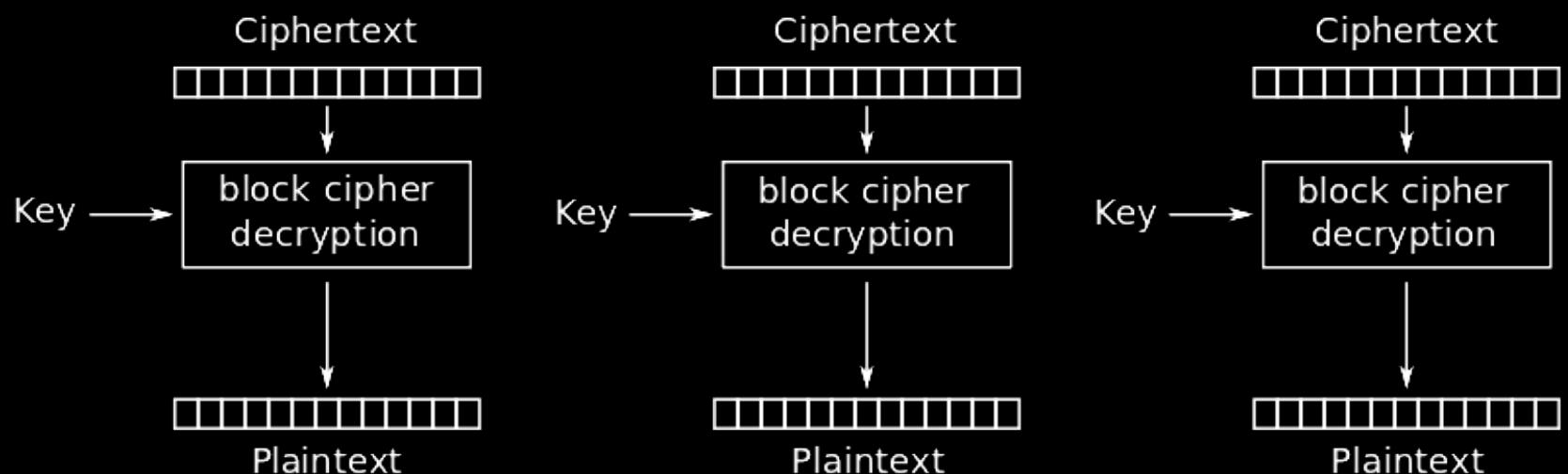
También es el más fácil de descifrar para los delincuentes porque todos los bloques de datos se cifran de la misma manera, lo que facilita la detección de patrones. Así que, si acabas enviando diferentes mensajes pero algunos de los mensajes son iguales, entonces vas a acabar con algunas instancias idénticas de texto cifrado.

Modo ECB (Electric CodeBook)

Básicamente, ECB tiende a dar demasiada información sobre el mensaje que intentas cifrar. Así que, la gran conclusión aquí es que no confiar en ECB para un cifrado seguro. El modo ECB no se recomienda a menos que sea el último recurso.



Electronic Codebook (ECB) mode encryption

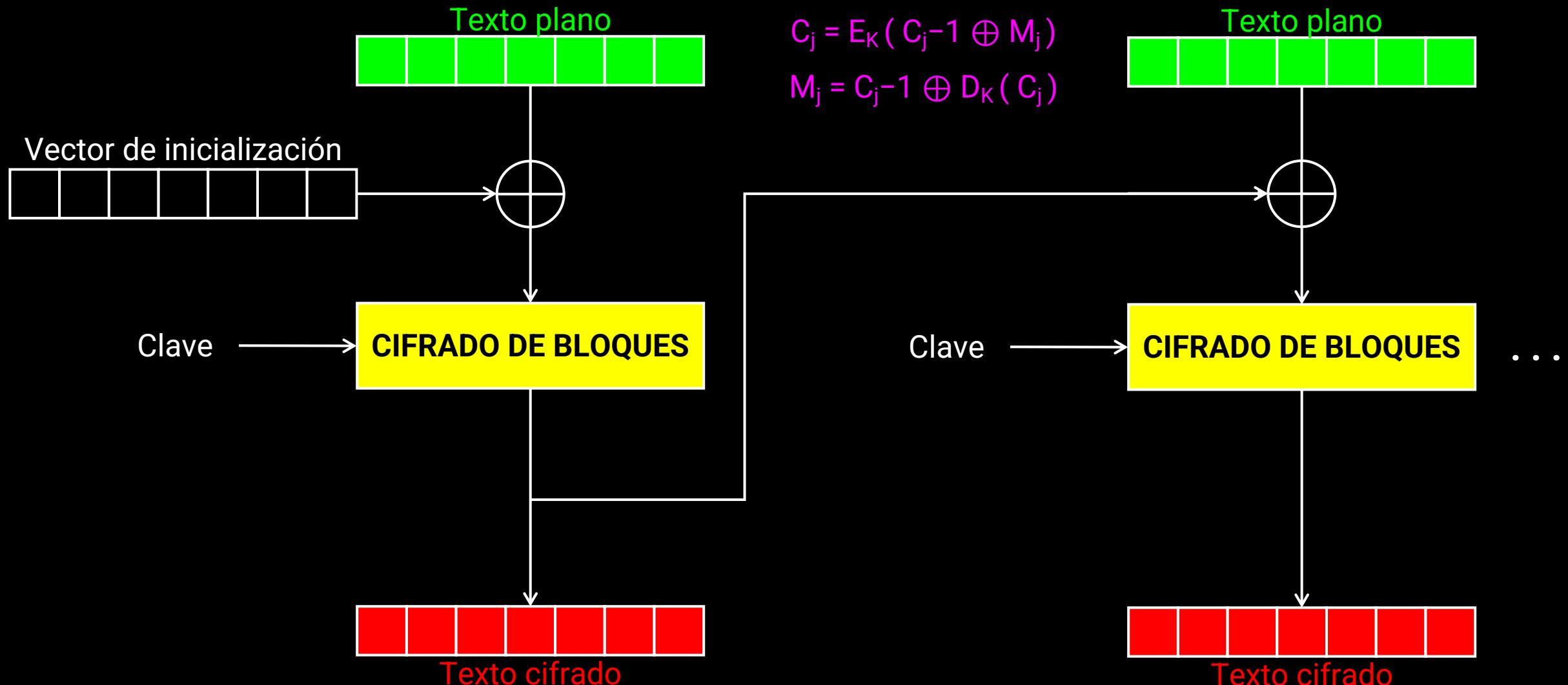


Electronic Codebook (ECB) mode decryption

Modos cifrado
por bloques



Modo CBC (Cipher Block Chaining)



Modo CBC (Cipher Block Chaining)

La esencia de este modo es que se trata de un proceso secuencial que se basa en bloques de datos anteriores. Así, los bloques de texto plano de datos de entrada se “encadenan” a bloques de salida anteriores de texto cifrado con el uso de un **vector de inicialización (IV)**. Y para que el **IV** sea más aleatorio, se le aplica un XOR antes del cifrado de cada entrada.

Modo CBC (Cipher Block Chaining)

Después de que los datos pasen por el cifrado, el resultado se encadena con el siguiente bloque de datos. Este proceso de encadenamiento de datos, conocido como encadenamiento explícito, ayuda a enmascarar aún más el mensaje. Lo hace añadiendo los datos del bloque anterior. Así, aunque envíes o recibas dos mensajes de texto plano exactamente iguales, el texto cifrado de cada uno será diferente debido a los datos encadenados.

Vector de inicialización (IV)

Un IV se usa como “estado inicial”. Al añadir el IV al cifrado se ocultan patrones en los datos cifrados que pueden permitir a un hacker descifrarlos por conjetura o prueba y error.

Vector de inicialización (IV)

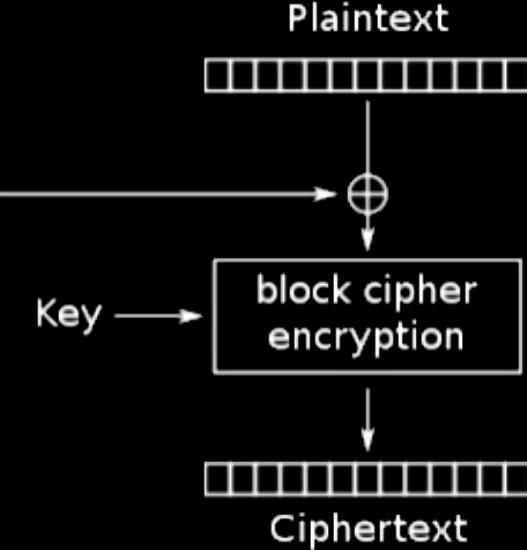
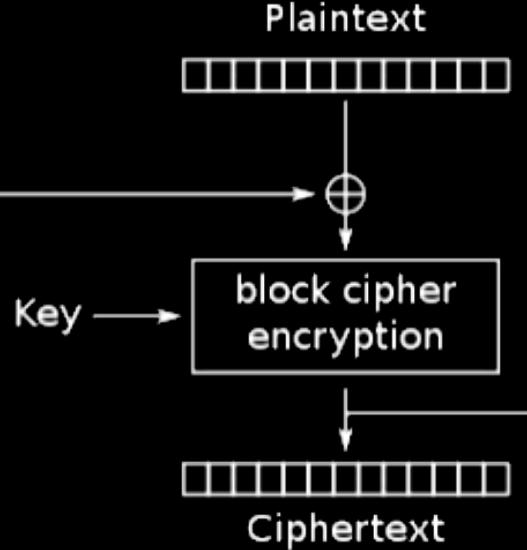
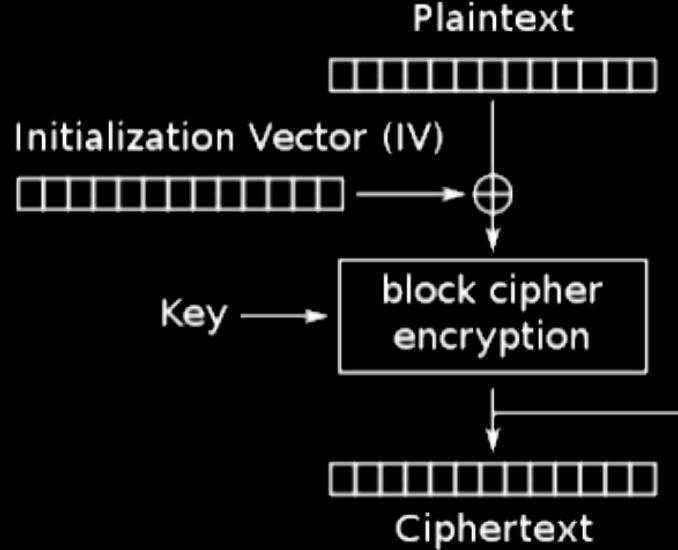
El IV ideal es un número aleatorio o pseudoaleatorio, también debe ser no repetitivo. Tanto la aleatoriedad como la no repetición son cruciales para evitar que los hackers encuentren patrones en partes similares del mensaje cifrado y luego utilicen esta información para descifrar el mensaje.

Vector de inicialización (IV)

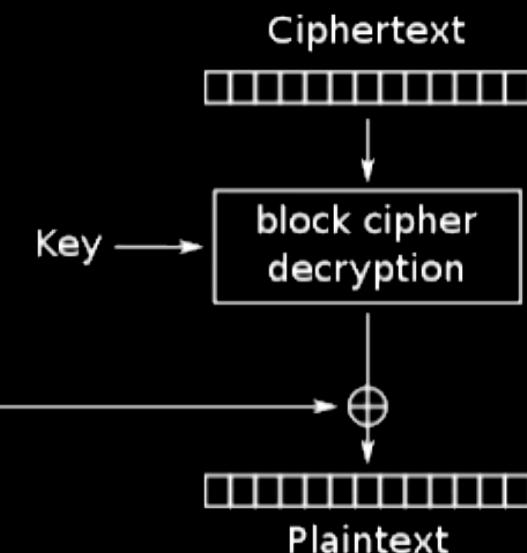
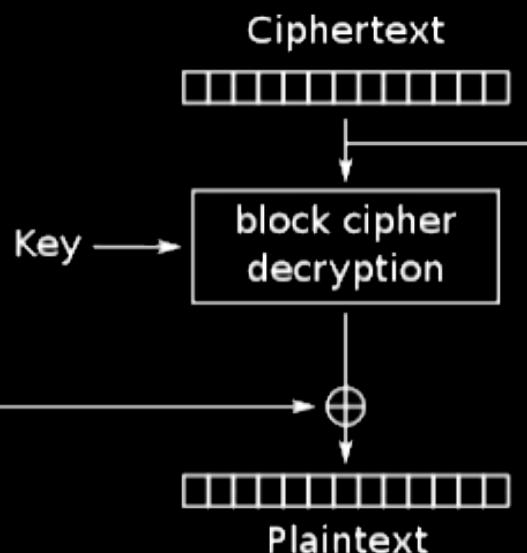
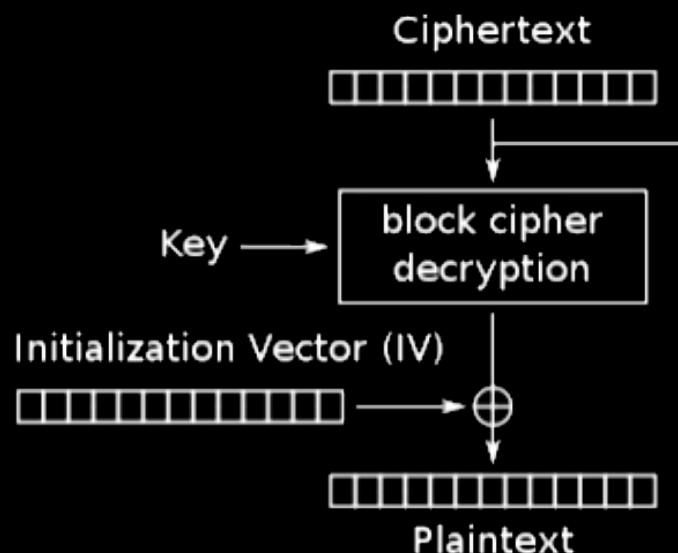
El IV no tiene por qué ser secreto, de hecho, el receptor suele conocer el IV para poder descifrar los datos cifrados cuando los recibe. Por lo tanto, tanto el remitente como el destinatario acordarían de antemano el IV. Además, el IV puede transmitirse de forma independiente o incluirse como parte de la configuración de la sesión antes del intercambio de mensajes.

Vector de inicialización (IV)

La longitud del IV en términos de número de bits o bytes depende del método de cifrado. En la mayoría de los casos, la longitud es comparable a la longitud de la clave de cifrado o del bloque del cifrado utilizado.



Cipher Block Chaining (CBC) mode encryption

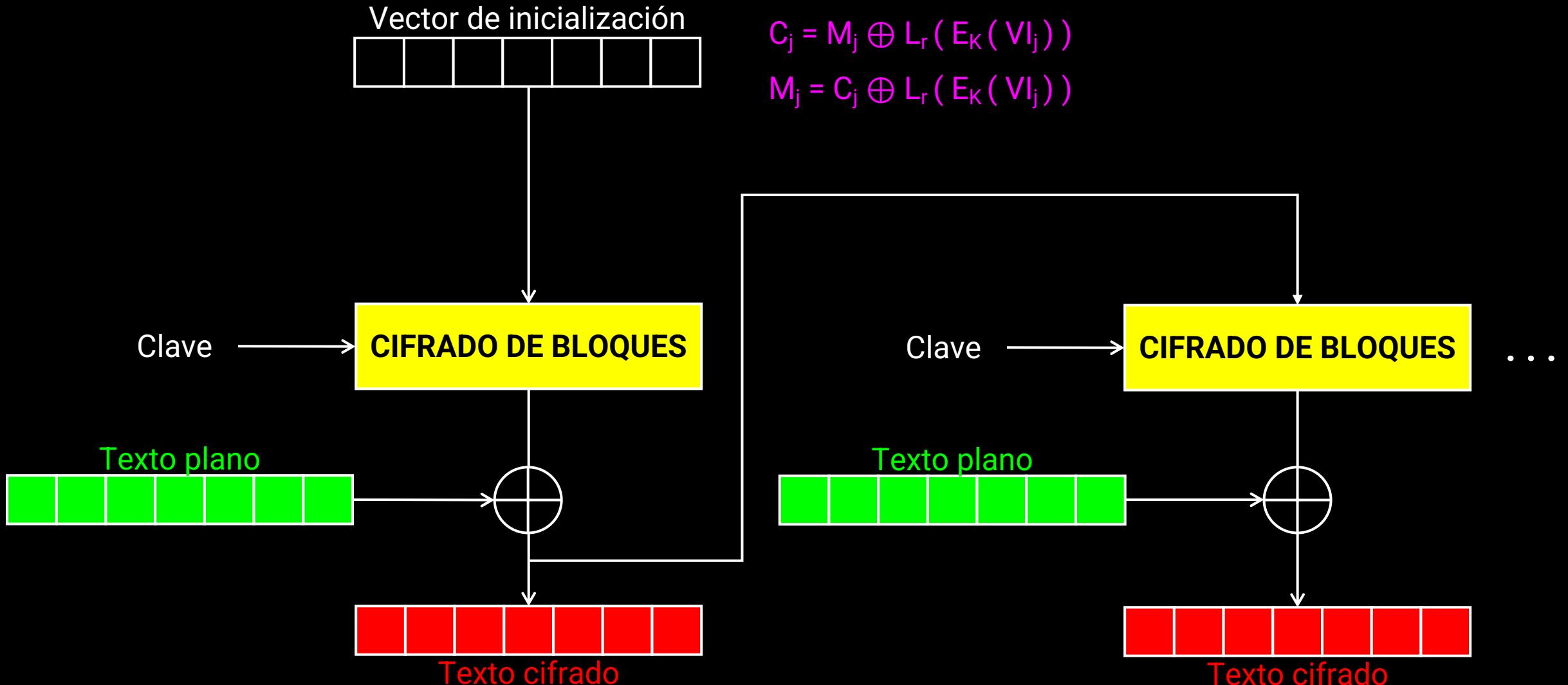


Cipher Block Chaining (CBC) mode decryption

Modos cifrado
por bloques



Modo CFB (Cipher Feedback)

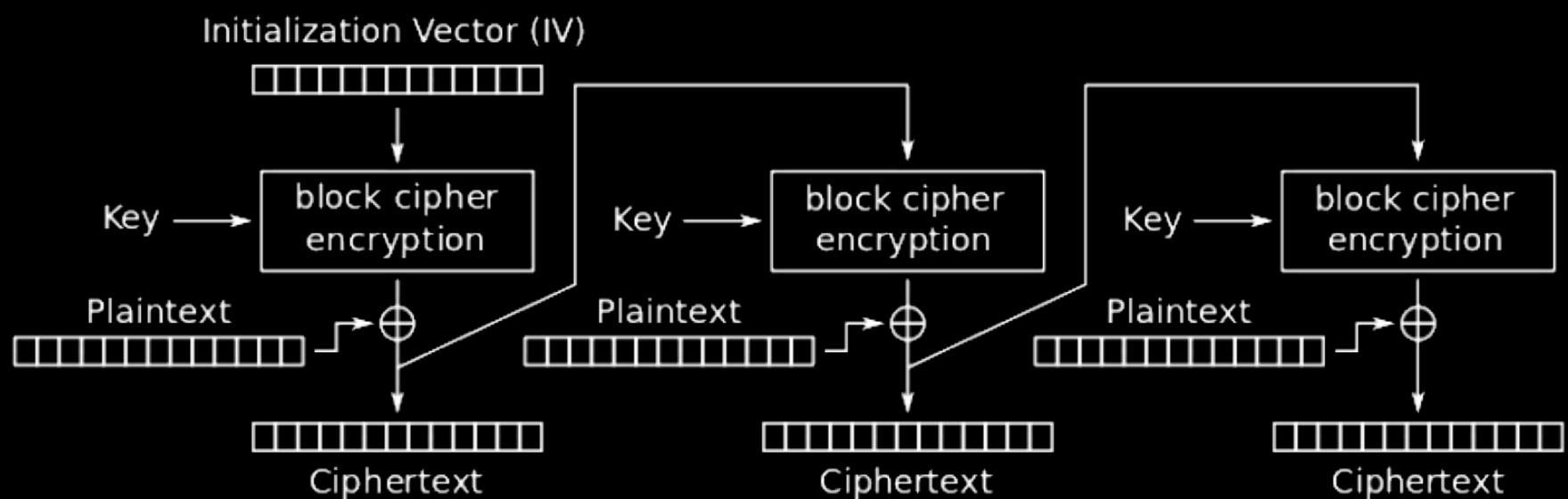


Modo CFB (Cipher Feedback)

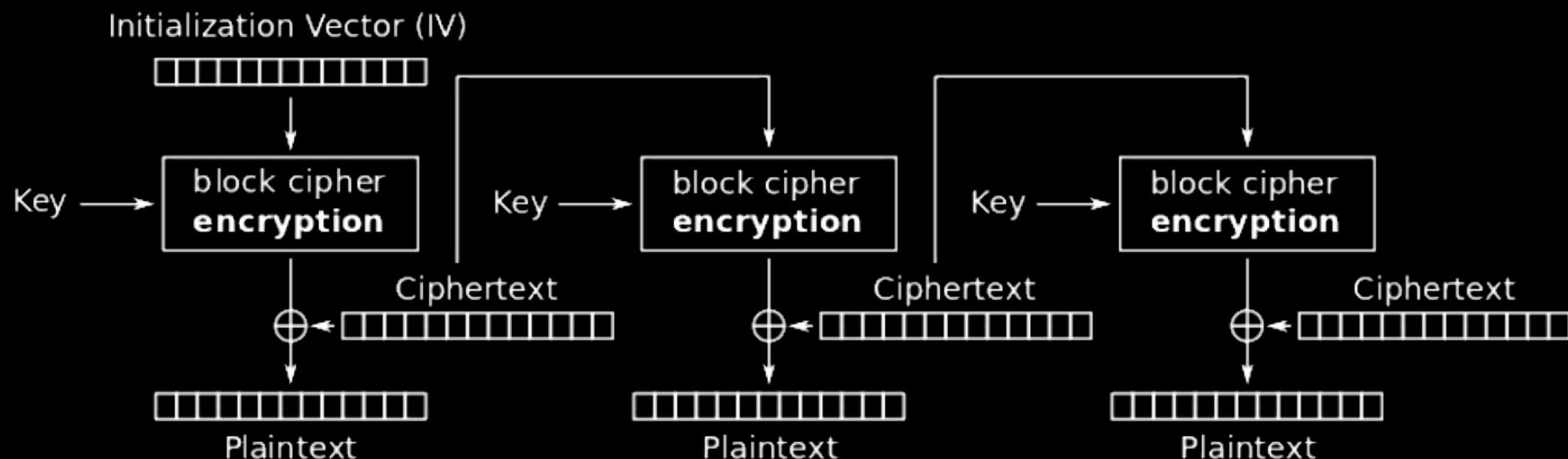
Este modo de funcionamiento genera bits pseudoaleatorios. Para ello, utiliza el texto cifrado del bloque anterior de la cadena (lo que se conoce como retroalimentación, de ahí su nombre) y una clave criptográfica.

Modo CFB (Cipher Feedback)

Un punto importante a tener en cuenta es que se utiliza una algoritmo de cifrado tanto en el proceso de cifrado como en el de descifrado, en lugar de utilizar una función de cifrado para el primero y una función de descifrado para el segundo. Además, la propagación de errores es un problema con el modo CFB porque un error en un bloque puede pasar al siguiente (aunque no afectaría al bloque posterior). Sin embargo, esto no es un problema en los modos OFB o CTR, a los que nos referiremos a continuación.



Cipher Feedback (CFB) mode encryption

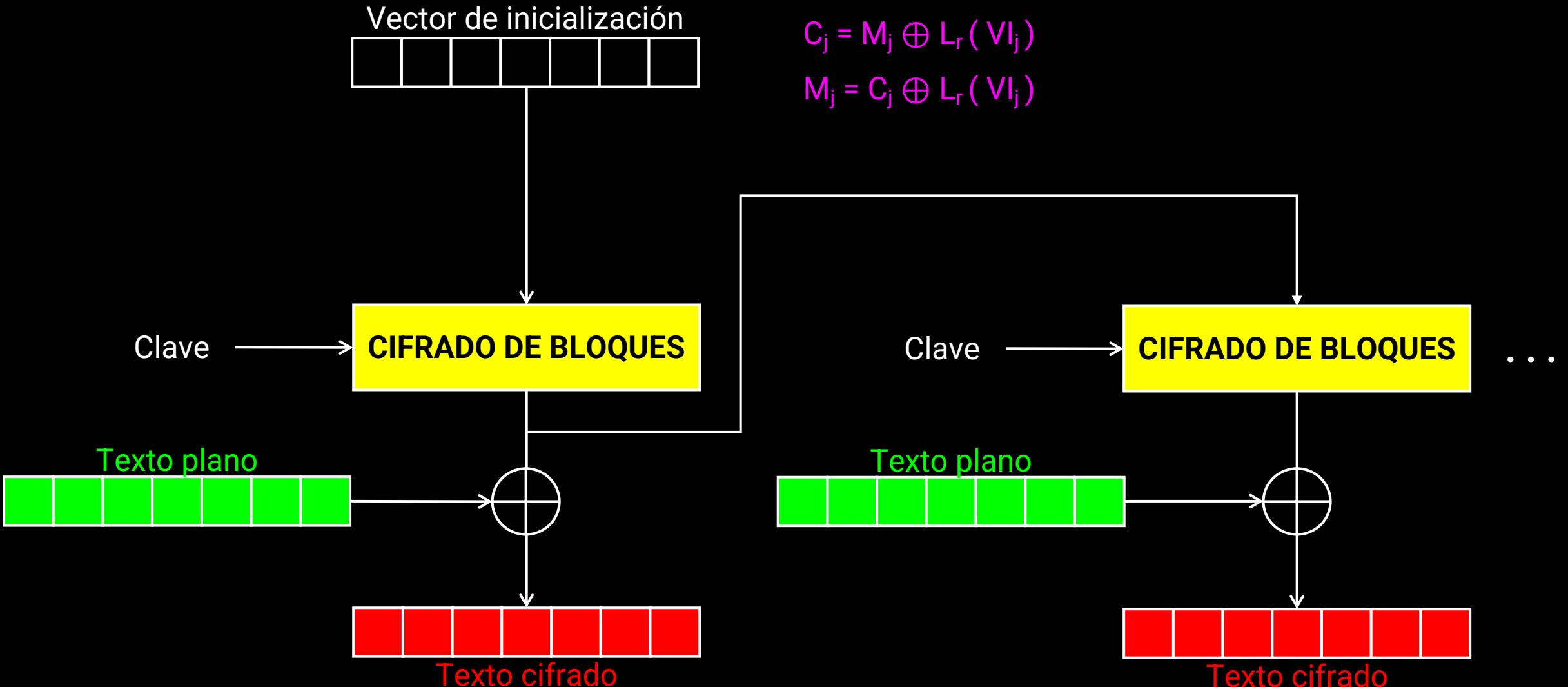


Cipher Feedback (CFB) mode decryption

Modos cifrado
por bloques



Modo OFB (Output Feedback)

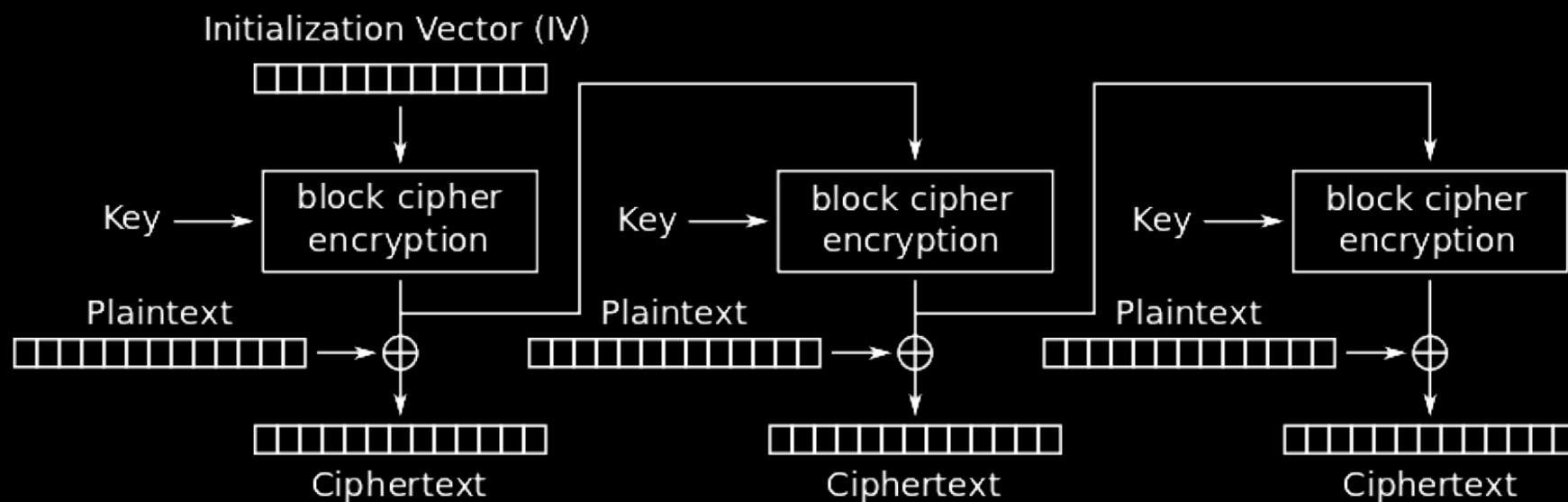


Modo OFB (Output Feedback)

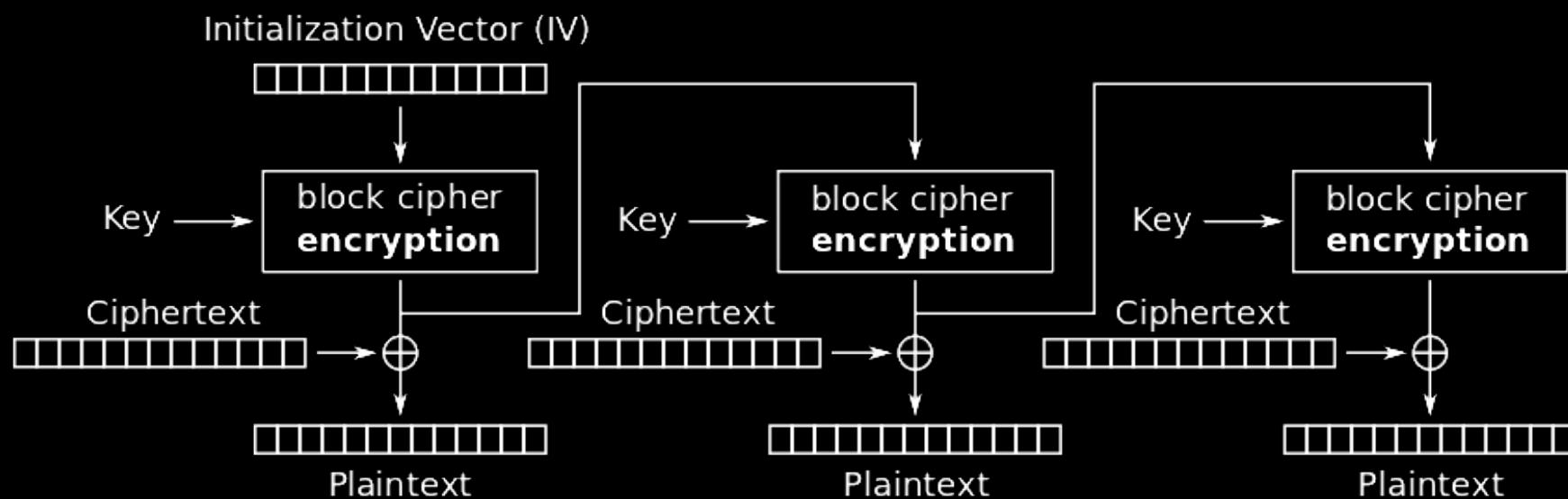
La idea de este modo es que es similar al CFB, pero la generación de números aleatorios se basa en la salida de la función de cifrado del bloque anterior como retroalimentación. El cifrado y descifrado OFB también utilizan el algoritmo de cifrado. Sin embargo, lo que OFB no hace, es utilizar el texto plano o el texto cifrado como parte del procesamiento de la función de cifrado.

Modo OFB (Output Feedback)

Básicamente, la salida de cada bloque se produce mediante la aplicación del cifrado a la salida del bloque anterior. Aplicando un XOR a las salidas con el texto plano, se obtiene el siguiente bloque de texto cifrado.



Output Feedback (OFB) mode encryption

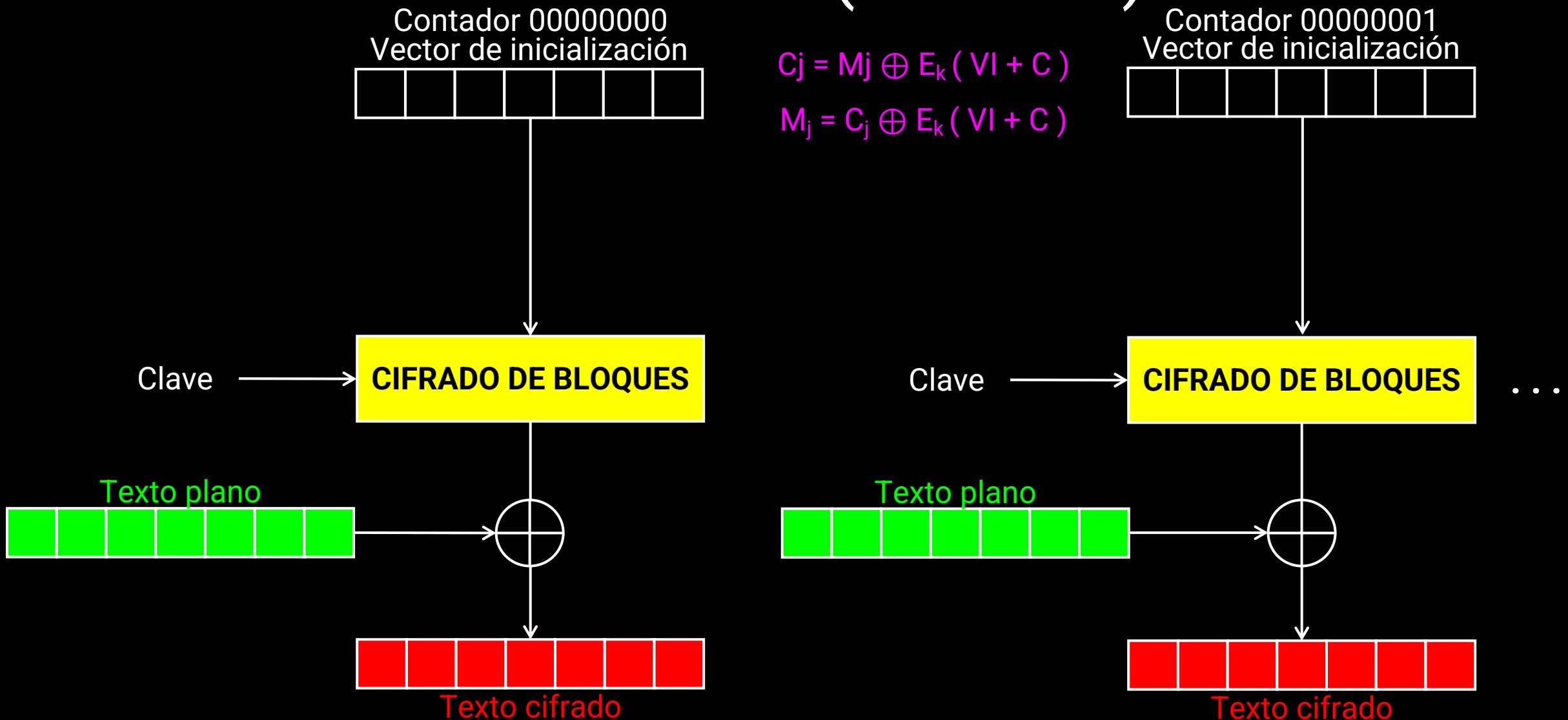


Output Feedback (OFB) mode decryption

Modos cifrado
por bloques

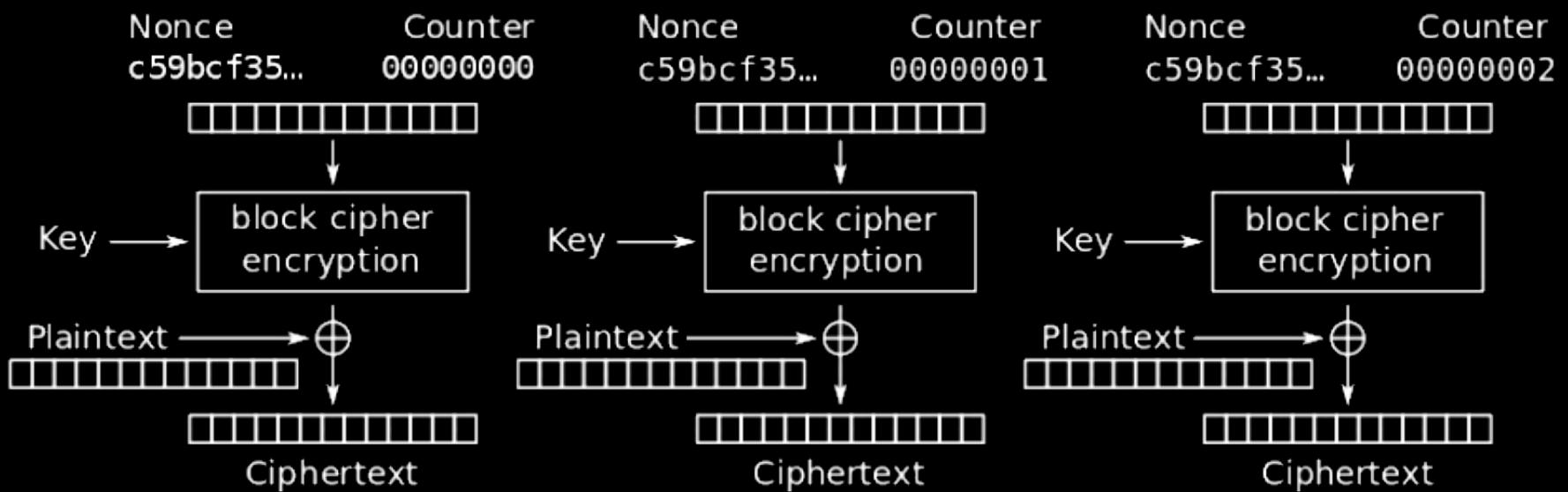


Modo CTR (Counter)

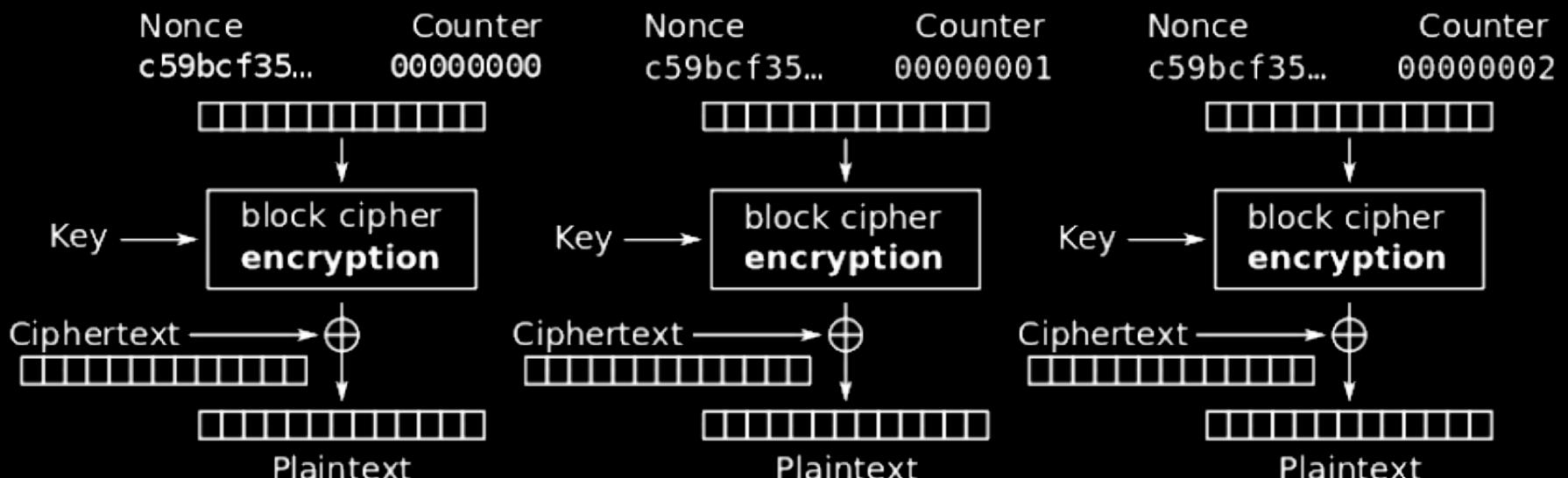


Modo CTR (Counter)

El modo CTR tiene similitudes con el modo OFB, pero sigue siendo diferente. Por ejemplo, a diferencia de los modos operativos anteriores, CTR no requiere encadenamiento explícito y es paralelizable. Esto significa que puede procesar y cifrar mensajes separados en paralelo (como los cifradores de flujo). Esto significa que, al no depender de la salida de un bloque anterior, puede descifrar dos bloques de forma independiente.



Counter (CTR) mode encryption



Counter (CTR) mode decryption

Redes Feistel

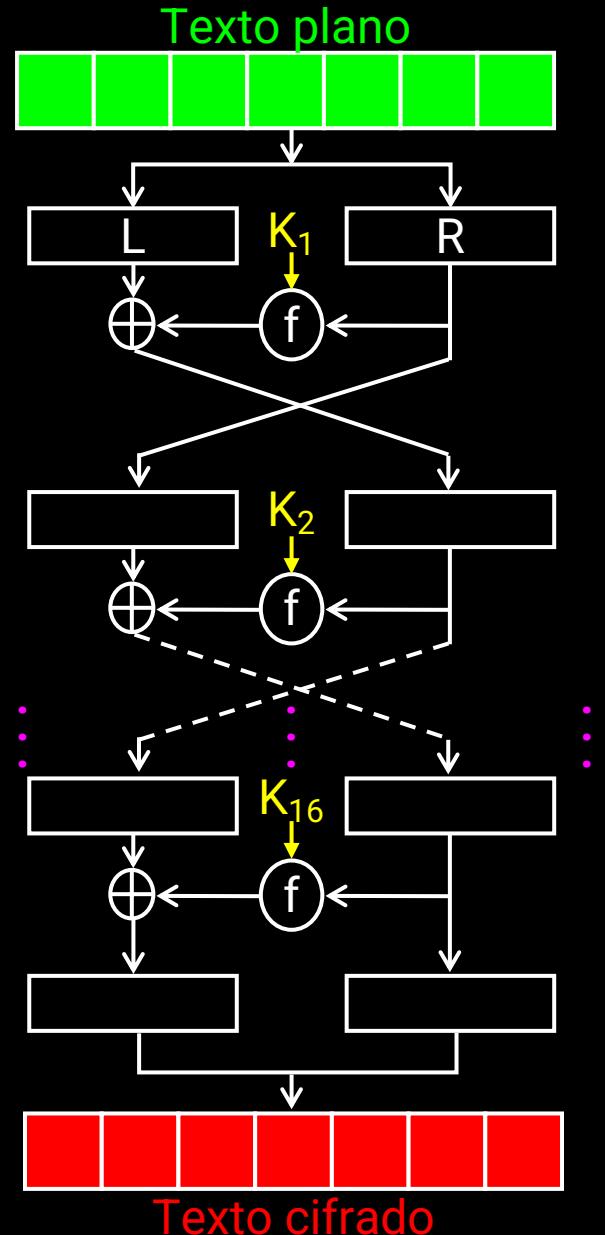
Una **red Feistel** es una técnica criptográfica utilizada en la construcción de algoritmos y mecanismos basados en cifrado de bloques. Diseñado por los empleados de IBM: Horst Feistel y Don Coppersmith, el primer uso de la **red Feistel** fue en el cifrado de bloques Lucifer. Una red Feistel también se conoce como **cifrado Feistel**.

Redes Feistel

Estas redes dividen el texto a cifrar, M , en dos mitades, L y R , y van cifrando iterativamente (repetitivamente) una de las mitades, luego las mitades se intercambian y el proceso se vuelve a repetir.

Redes Feistel

1. La entrada se divide en dos bloques de igual tamaño (64-256)
2. Claves de (56-256 bits)
3. Número de rondas 10-16
4. En cada ciclo se aplica una función hash (f) al bloque derecho
5. Con el resultado se realiza un XOR con el bloque izquierdo
6. Los bloques se intercambian



Algoritmos de cifrado simétrica por bloques

DES

3DES

Blowfish

Twofish

Mars

Serpent

RC6

Rijndael

Algoritmos de cifrado simétrica por bloques

La mayoría de los conjuntos de cifrado actuales se basan en cifrados por bloques. Existen varios algoritmos estándar de cifrado que utilizan cifradores de bloques.

Data Encryption Standard (DES)

Data Encryption Standard (DES) es el predecesor del algoritmo Rijndael, que no sólo era vulnerable a los ataques de fuerza bruta, sino también a los de criptoanálisis. Aunque sigue utilizándose, ha demostrado ser inseguro para grandes organizaciones y gobiernos.

Triple Data Encryption Standard (3DES)

3DES es un algoritmo de cifrado mejor que DES, ya que triplica el nivel de cifrado de DES y tiene un tiempo de seguridad medio. Sin embargo, es más lento que los demás métodos de cifrado por bloques y también está expuesto a ataques diferenciales y de claves relacionadas.

Blowfish

BLOWFISH es un algoritmo de cifrado simétrico y uno de los más comunes del dominio público que utiliza 16 rondas como DES. Cada ronda contiene una permutación dependiente y una sustitución dependiente de los datos para su funcionamiento. Se trata de una criptografía de cifrado por bloques sencilla, rápida y compacta que tiene una longitud variable que permite un compromiso entre seguridad y velocidad. Blowfish adolece de claves débiles, lo que a veces hace cuestionable su fiabilidad.

International Data Encryption (IDEA)

IDEA basa su concepto en la estructura de permutación por sustitución. Tiene una potente protección impenetrable contra el criptoanálisis diferencial y al mismo tiempo utiliza numerosas operaciones de grupo para mejorar su fuerza también utiliza múltiples operaciones de grupo para aumentar su fuerza contra ataques familiares y también es susceptible a diferentes clases de claves débiles.

Twofish

TWOFISH emplea redes Feistel de 16 rondas con blanqueamiento adicional de la entrada y la salida. Twofish es conocido por su flexibilidad, rapidez y, tiene una variedad de compensaciones de implementación. Se diseñó pensando en las tarjetas inteligentes y puede descifrar y cifrar a mayor velocidad. Funciona bien en una amplia gama de plataformas y aplicaciones, al tiempo que conserva la velocidad y la eficacia cuando se implementa en software y hardware.

MARS

MARS se basa en una estructura independiente. Utiliza texto plano de 128 bits con 32 rondas y la longitud de clave variable incluyendo la rotación dependiente de datos de multiplicación. Ofrece más seguridad y velocidad que 3DES y DES, pero es el más complicado de todos. La robustez, que era su principal objetivo de diseño, resultó ser su principal punto fuerte. Sin embargo, MARS contiene más mecanismos de parada por fallo que otros. Aparte del hecho de que no es adecuado para la implementación de tarjetas inteligentes y es relativamente complejo de analizar, MARS no tiene limitaciones significativas.

Serpent

SERPENT se basa en una estructura de red de sustitución-permutación. Tiene 128 bits de texto plano con 32 rondas y una longitud de clave variable separada de 128, 192 y 256. Utiliza dos modos diferentes de implementación que son el modo estándar y el modo bit-slice. La principal ventaja de la SERPENT es su naturaleza conservadora del número de rondas y, a diferencia de MARS, no tiene mecanismo de parada por fallo. No tiene ninguna limitación, salvo que las 32 rondas dificultan su ejecución y son más lentas en los minibloques.

RC6

RC6 es el más elegante, sencillo y fácil de entender de todos. Es la versión reforzada de RC5 y funciona con 20 rondas. La estructura Feistel funciona en 32 bits con multiplicación modular, XOR como suma. Su principal ventaja es su simplicidad y velocidad, que lo hacen adecuado para su diseño actual. El principal problema es su único punto de fallo y los problemas relacionados con el número de rondas que utiliza. La total arbitrariedad, las claves débiles y el hecho de que no llegue a 17 rondas del algoritmo son las otras limitaciones que posee.

Rijndael (AES Advanced Encryption Standard)

El algoritmo Rijndael conocido como AES fue adoptado para sustituir al DES tras descubrirse que es vulnerable y puede crackearse fácilmente. Rijndael es un algoritmo de cifrado simétrico por bloques que utiliza claves de 128, 192 y 256 tamaños. No utiliza la red Feistel, sino que su funcionamiento se basa en la red de sustitución y permutación. Está más protegido, es más rápido, difícil de descifrar y muy eficaz que DES.

Rijndael (AES)

Rijndael tiene la ventaja de no ser susceptible a ningún ataque, salvo los de fuerza bruta. Se comporta y funciona cómodamente en una amplia variedad de estructuras de hardware, desde tarjetas inteligentes de 8 bits hasta ordenadores de gran rendimiento. También funciona bien en software y el hecho de que pueda funcionar en dispositivos con poca memoria RAM hace que destaque sobre todos los demás algoritmos de cifrado.

Rijndael (AES)

Sin embargo, su limitación es que el cifrado inverso es menos adecuado para ser implementado en una tarjeta inteligente que el propio cifrado. Aunque, en comparación con otros, sigue siendo más rápido y mejor.

Cifrado Simétrico

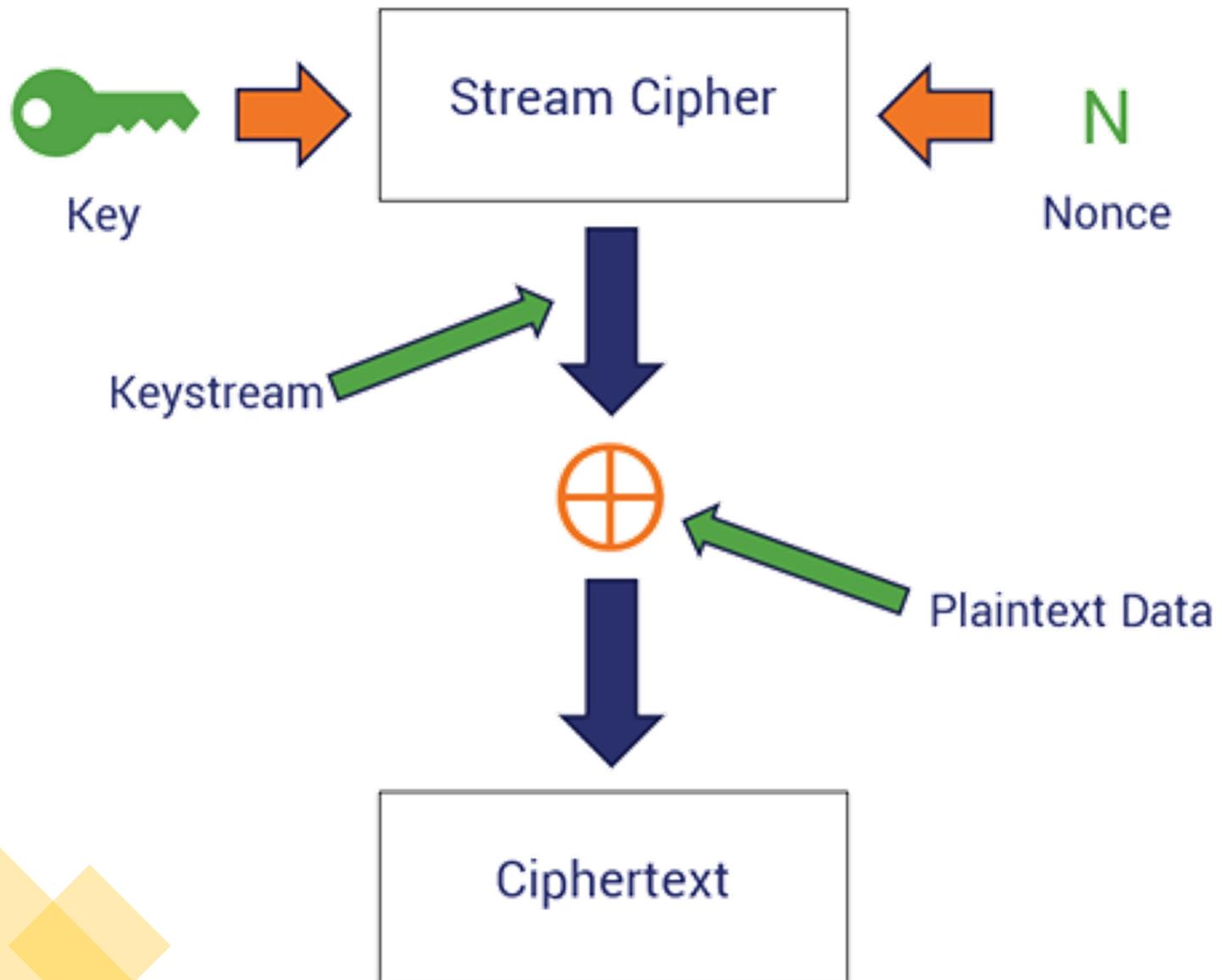
Cifrado por flujo

Cifrado por
bloques

Cifrado por flujo

Un **cifrado de flujo** es un cifrado que encripta (y desencripta) con el flujo de datos. A diferencia de los cifradores de bloques, que requieren la formación de bloques antes del cifrado, estos cifran los datos en largos flujos pseudoaleatorios. Básicamente, esto significa que **puedes procesar un bit de datos cada vez** en lugar de esperar a que se forme un bloque de datos.

How a Basic Stream Cipher Works



Cifrado por flujo

Un cifrado de flujo utiliza cambios temporales en los datos de texto plano para cifrar una cadena continua de números binarios. Como resultado, este método de cifrado funciona bit a bit, utilizando secuencias de claves para generar texto cifrado para mensajes de texto plano de longitudes arbitrarias.

El cifrado combina una clave (128/256 bits) y un dígito nonce (64-128 bits) para generar el flujo de claves (**keystream**), que es un número pseudoaleatorio XOR con el texto en claro para generar el texto cifrado.

Cifrado por flujo

El **flujo de claves** (**keystream**) debe ser diferente en cada iteración de cifrado, aunque la clave y el nonce puedan reutilizarse para mantener la seguridad. Para construir el flujo de claves, los cifradores de flujo generan un nonce único (un número que sólo se utiliza una vez utilizando registros de desplazamiento de realimentación).

Cifrado por flujo

Síncrono

Asíncrono

Cifrados de flujo síncronos

Cifrados de flujo síncronos (también conocidos como **cifrados de clave automática** o **KAK**): estos tipos de cifrados generan flujos de claves independientemente de cualquier texto plano o texto cifrado anterior.

Cifrado por flujo

Síncrono

Asíncrono

Cifrados de flujo asíncronos

cifrados de flujo asíncronos (también conocidos como **cifrados de flujo autosincronizados, autoclave de texto cifrado** o **CTAK**): estos cifrados, por otro lado, se basan en bits de texto cifrado anteriores para generar secuencias de claves.

Algoritmos
de clave
simétrico
por flujo

Salsa20

ChaCha20

RC4

A5

Salsa20

Fue diseñado por Daniel J. Bernstein es un cifrado de flujo síncrono que genera su salida como bloques de 64 bytes derivados de la clave, el nonce y el número de bloque; así, permite obtener un bloque de salida para cualquier posición independientemente de cualquier bloque generado previamente. Se diseñó para utilizarse con una clave de 32 bytes (256 bits) y 20 rondas. Pero puede utilizarse con sus versiones de 8 y 12 rondas y con una clave más pequeña. Se basa en operaciones de suma, XOR y rotaciones de distancia constante, todas ellas sobre una matriz 4x4 de elementos de palabra de 32 bits (clave, nonce, contador de bloque y palabras constantes). La salida de esta matriz tras aplicar estas operaciones durante n rondas es un flujo de 16 palabras (64 bytes) que se utiliza para cifrar (XOR) el texto plano.

ChaCha20

ChaCha20 es un algoritmo que soporta claves de 128 y 256 bits y de alta velocidad, a diferencia de AES que es un cifrado por bloques, ChaCha20 es un cifrado de flujo. Tiene características similares a su predecesor Salsa20, pero con una función primitiva de 12 o 20 rondas distintas. En implementaciones de software, es mucho más eficiente y rápido que AES, por lo que rápidamente se ha hecho un hueco dentro de los algoritmos más usados en la actualidad.

Rivest Cipher 4 (RC4)

Fue diseñado por Ron Rivest. Tiene un parámetro de tamaño de 8 bits. Genera datos pseudoaleatorios de byte en byte y mantiene un estado interno de 256 bytes. La operación de combinación es XOR. La clave puede ser de cualquier tamaño hasta el tamaño del estado, de 256 bytes o 2048 bits para la versión de 8 bits. RC4 es bastante sencillo de implementar en software y su uso está muy extendido.

A5

GSM fue el primer sistema de comunicación móvil digital implantado en todo el mundo. Permitía comunicarse mediante llamadas de voz y enviar SMS con buena calidad. Para garantizar la confidencialidad de las llamadas, utiliza el algoritmo de cifrado de flujo A5.

A5 tiene algunas variantes: A5/0, A5/1 y A5/2. El A5/1 es el más potente y se utiliza en Europa y América. Debido a las restricciones a la exportación de criptografía, el A5/0 (el más débil) se utiliza en los países del tercer mundo y en los sancionados por la ONU. El A5/2 se utiliza en Asia.

A5

A5/0 no utiliza ningún generador de números pseudoaleatorios. Utiliza como flujo de salida, la misma entrada, negada. Por lo tanto, este algoritmo no proporciona confidencialidad.

A5/1 utiliza 3 LFSR para producir una clave de 64 bits (LFSR de 19, 22 y 23 bits cada una) y su salida se aplica un XOR con el texto plano. Este algoritmo está implementado por hardware en la mayoría de los teléfonos móviles para permitir el cifrado y descifrado en tiempo real.

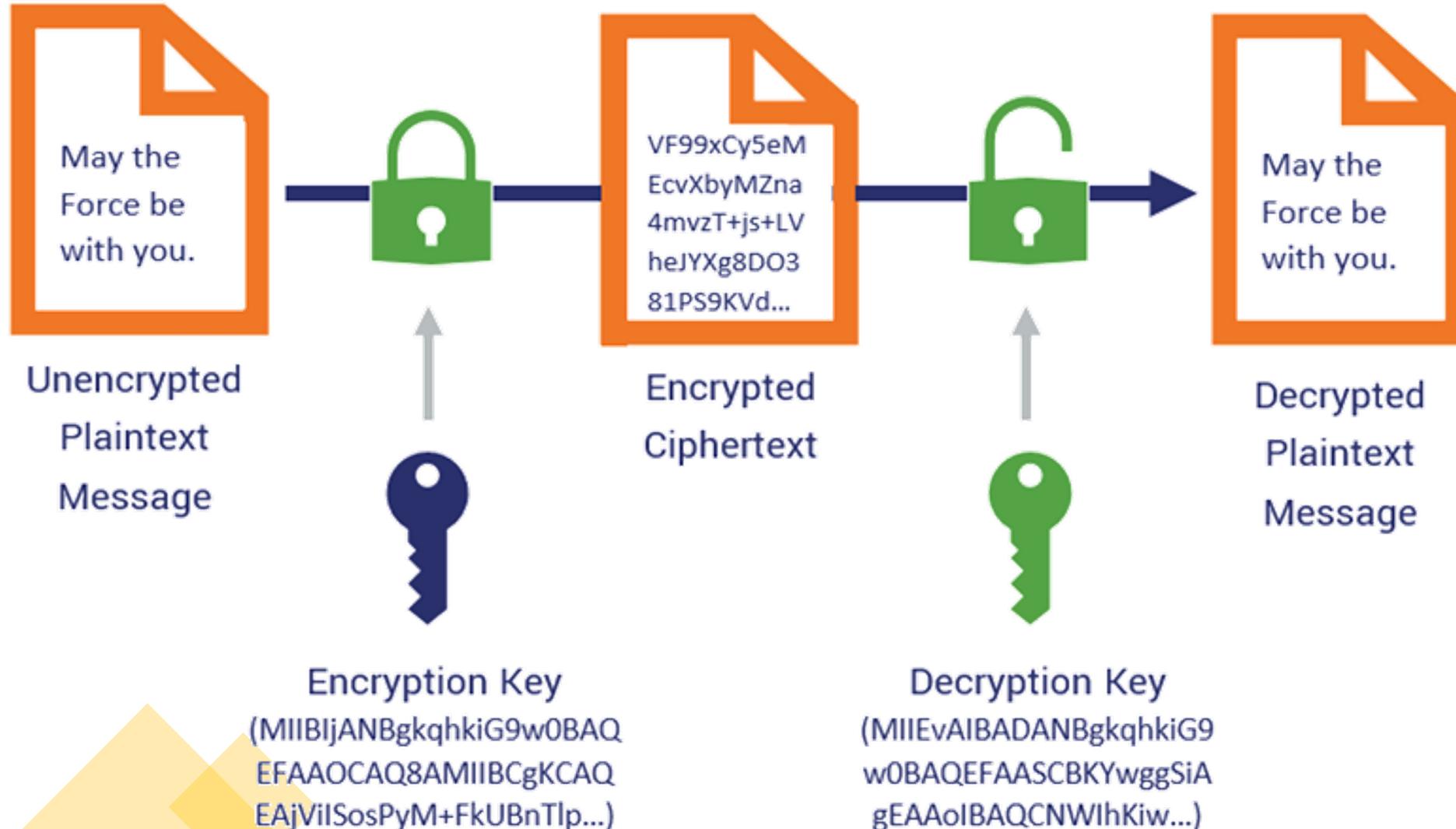
Criptografía moderna

Cifrado
Simétrico

Cifrado
Asimétrico

Funciones
Hash

How Asymmetric Encryption Works



Cifrado asimétrico

El **cifrado asimétrico** o **de clave pública** utiliza dos claves distintas: una de ellas puede ser **pública**, la otra es **privada**. La posesión de la clave pública no proporciona suficiente información para determinar cuál es la clave privada.

Cifrado asimétrico

Una **clave pública** es una clave criptográfica que puede ser utilizada por cualquier persona para cifrar un mensaje de manera que sólo pueda ser descifrado por el destinatario con su **clave privada**. Una **clave privada** (también conocida como **clave secreta**) sólo se comparte con el iniciador de la clave.

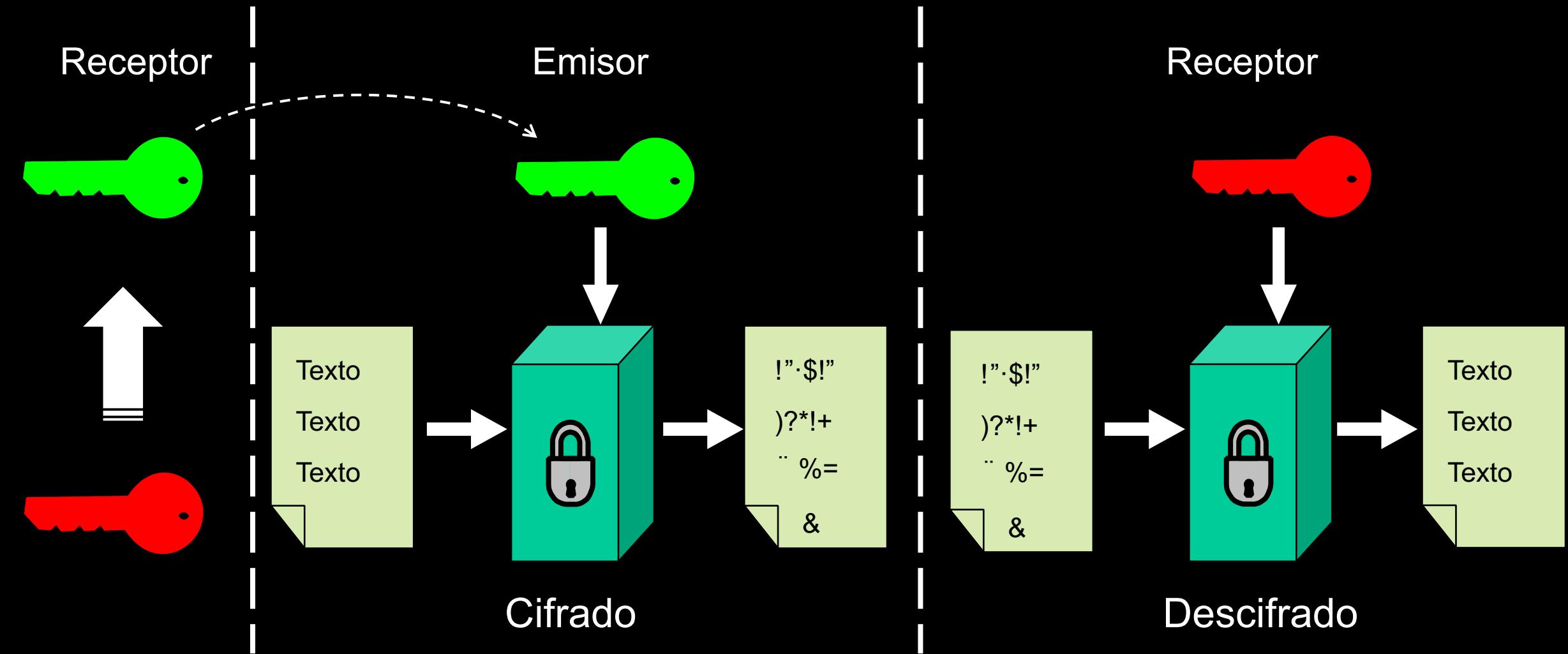
Cuando alguien quiere enviar un mensaje cifrado, puede obtener la **clave pública** del destinatario de un directorio público y utilizarla para cifrar el mensaje antes de enviarlo. El destinatario del mensaje puede entonces descifrarlo utilizando su **clave privada** correspondiente.

Cifrado asimétrico

Si el remitente encripta el mensaje con su **clave privada**, sólo podrá desencriptarlo con la **clave pública** del remitente, lo que permitirá autenticarlo. Estos procesos de cifrado y descifrado se ^{Cifrado Asimétrico}producen automáticamente pues los usuarios no necesitan bloquear y desbloquear físicamente el mensaje.

El principal beneficio de la **criptografía asimétrica** es el **aumento de la seguridad de los datos**. Es el proceso de cifrado más seguro porque los usuarios nunca tienen que revelar o compartir sus **claves privadas**, lo que disminuye las posibilidades de que un hacker descubra la **clave privada** de un usuario durante la transmisión.

Funcionamiento del cifrado asimétrico



Algoritmos
de cifrado
asimétrico

Diffie-Hellman

RSA

DSA

ElGamal

ECC

Diffie-Hellman

Este algoritmo fue uno de los primeros de clave pública desarrollado por Whitfield Diffie y Martin Hellman en 1976, de ahí el nombre. También es conocido como el algoritmo de Intercambio Exponencial Diffie-Hellman, el cual basa su seguridad en la dificultad de calcular logaritmos discretos en un campo finito y se emplea para la distribución de claves, pero no para cifrar y descifrar.

Rivest-Shamir-Adleman (RSA)

El algoritmo más conocido y representativo de la criptografía de clave pública es precisamente RSA desarrollado en el Instituto Tecnológico de Massachusetts (MIT) en 1977 por Ron Rivest, Adi Shamir y Leonard Adleman, quienes bautizaron a su esquema de cifrado con las iniciales de sus apellidos.

Digital Signature Algorithm (DSA)

Es un algoritmo que tiene como finalidad firmar documentos electrónicos, de ninguna manera cifra información. Fue desarrollado por el NIST que es una agencia de la Administración de Tecnología del Departamento de Comercio de los Estados Unidos.

ElGamal

Taher ElGamal nacido en Egipto en 1995, desarrollo un algoritmo de cifrado de datos y firma digital en 1985, basado en el trabajo realizado por Diffie-Hellman, esto es, se basa en el problema del logaritmo discreto.

Criptografía de curva elíptica (ECC)

Muchos criptosistemas requieren el uso de grupos algebraicos, de manera que los sistemas de cifrado basados en curvas elípticas pueden utilizar los que provienen de los grupos de curvas elípticas, donde un grupo es un conjunto de elementos con operaciones aritméticas definidas en estos elementos , y para los grupos de curvas elípticas, estas operaciones específicas están definidas geométricamente.

Criptografía moderna

**Cifrado
Simétrico**

**Cifrado
Asimétrico**

**Funciones
Hash**

Funciones HASH

Es una función que proporciona un pequeño conjunto de bits a partir del mensaje, mucho menor que éste, y que es casi exclusivo (es muy difícil encontrar otro mensaje para el que la función resumen dé el mismo resultado).

Funciones Hash o de Resumen

Dado un mensaje original calculan un resumen del mensaje, sea cual sea la longitud del mensaje y siempre que utilicemos el mismo algoritmo, la longitud del resumen es siempre la misma.

Propiedades de una función hash segura:

- Unidireccionalidad
- Compresión
- Facilidad de cálculo
- Difusión:

Algoritmos
de
funciones
Hash o de
resumen

MD5 (128 bits)

SHA-1 (160 bits)

SHA-2 (256 o 512 bits)

MD5

MD5 es uno de los algoritmos de resumen criptográficos diseñados por el profesor Ronald Rivest del MIT (Massachusetts Institute of Technology). Cuando un análisis indicó que el algoritmo MD4 era inseguro, se decidió a programar el MD5 para sustituirlo en 1991. Las debilidades en MD4 fueron descubiertas por Hans Dobbertin.

Message Digest 5 (MD5)

- Algoritmo básico de Message Digest 5
 - a) Un mensaje M se convierte en un bloque múltiplo de 512 bits, añadiendo bits si es necesario al final del mismo.
 - b) Con los 128 bits de cuatro vectores iniciales ABCD de 32 bits cada uno y el primer bloque del mensaje de 512 bits, se realizan diversas operaciones lógicas entre ambos bloques.
 - c) La salida de esta operación (128 bits) se convierte en el nuevo conjunto de 4 vectores A'B'C'D' y se realiza la misma función con el segundo bloque de 512 bits del mensaje, y así hasta el último bloque del mensaje. Al terminar, el algoritmo entrega un resumen que corresponde a los últimos 128 bits de estas operaciones.

SHA-1

El algoritmo SHA-1 fue desarrollado por la NSA (National Security Agency de USA). Produce firmas de 160 bits, a partir de bloques de 512 bits del mensaje original.

SHA-1

El algoritmo es similar a MD5, y se inicializa igual que éste, añadiendo al final del mensaje un uno seguido de tantos ceros como sea necesario hasta completar 448 bits en el último bloque, para luego juxtaponer la longitud en bytes del propio mensaje. A diferencia de MD5, SHA-1 emplea cinco registros de 32 bits en lugar de cuatro.

SHA-2

Es una familia de dos funciones hash similares, con diferentes tamaños de bloque, conocidas como SHA-256 y SHA-512. Se diferencian en el tamaño de palabra. Se diferencian en el tamaño de las palabras: SHA-256 utiliza palabras de 32 bits, mientras que SHA-512 utiliza palabras de 64 bits. También existen versiones truncadas de cada estándar, conocidas como SHA-224, SHA-384, SHA-512/224 y SHA-512/256. Estas versiones también fueron diseñadas por la NSA.

Firma Digital

Una firma digital es una técnica matemática utilizada para validar la autenticidad e integridad de un mensaje, software o documento digital.

La firma digital, a diferencia de una firma tradicional, no es un nombre sino que consta de dos "claves" o secuencias de caracteres separadas. Consiste en aplicar mecanismos criptográficos al contenido de un mensaje o documento con el objetivo de demostrar al receptor del mensaje lo siguiente:

- que el emisor del mensaje es real (**autenticación**);
- que éste no puede negar que envió el mensaje (**no repudio**);
- que el mensaje no ha sido alterado desde su emisión (**integridad**).

Firma Digital

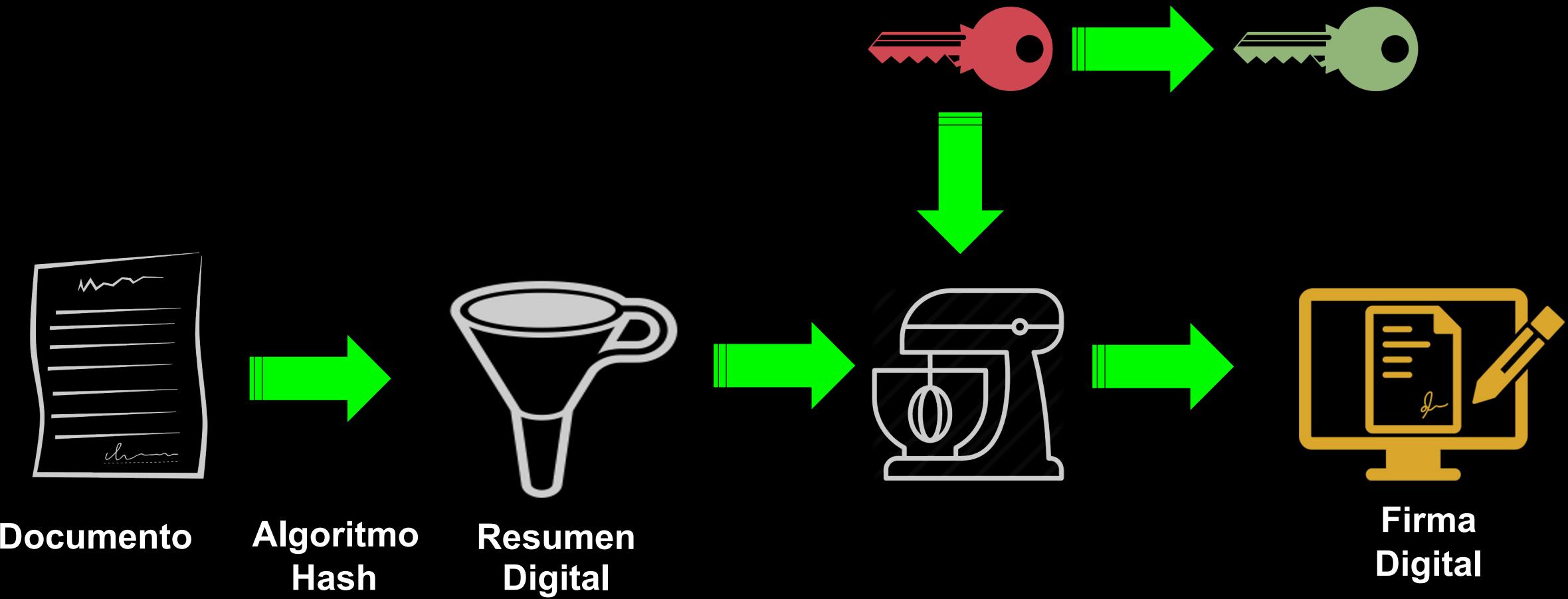
- El valor hash de un mensaje, cifrado con la clave privada de una persona es su firma digital sobre ese mensaje
- La firma digital de una persona varía de un documento a otro asegurando así la autenticidad de cada palabra del documento
- Como la clave pública del firmante es conocida, cualquiera puede verificar el mensaje y la firma digital

Funcionamiento

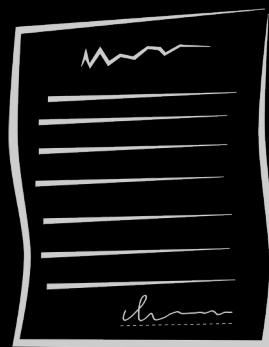
Las firmas digitales se basan en la criptografía de clave pública, también conocida como criptografía asimétrica. Normalmente hay tres algoritmos involucrados con el proceso de firma digital:

- **Generación de dos claves que están matemáticamente vinculadas:** un algoritmo proporciona una clave privada junto con su clave pública correspondiente.
- **Firma:** este algoritmo produce una firma al recibir una clave privada y el mensaje que se está firmando.
- **Verificación:** este algoritmo comprueba la autenticidad del mensaje al verificarlo junto con la firma y la clave pública.

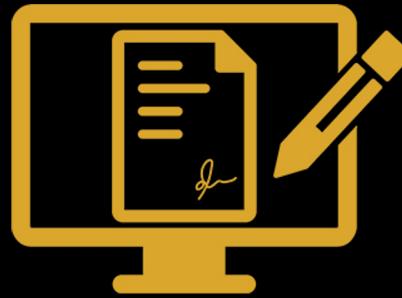
Creación Firma Digital



Envío Firma Digital

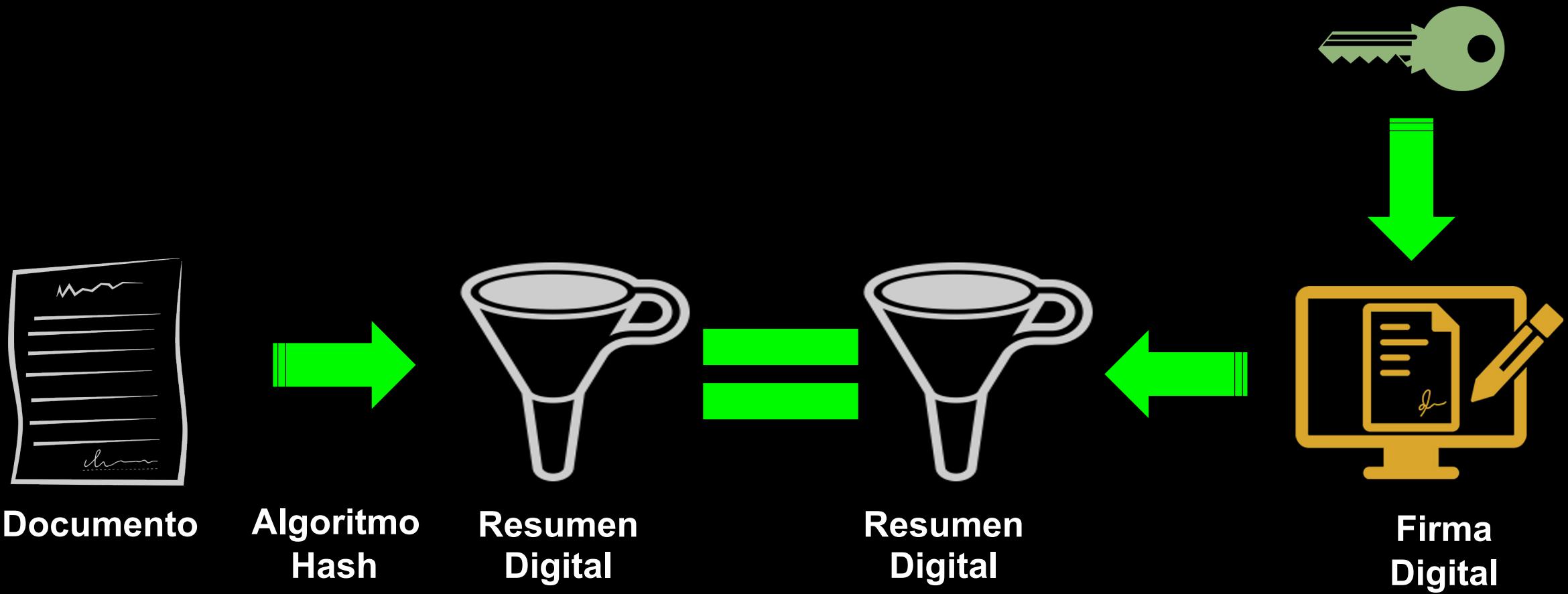


Documento



Firma
Digital

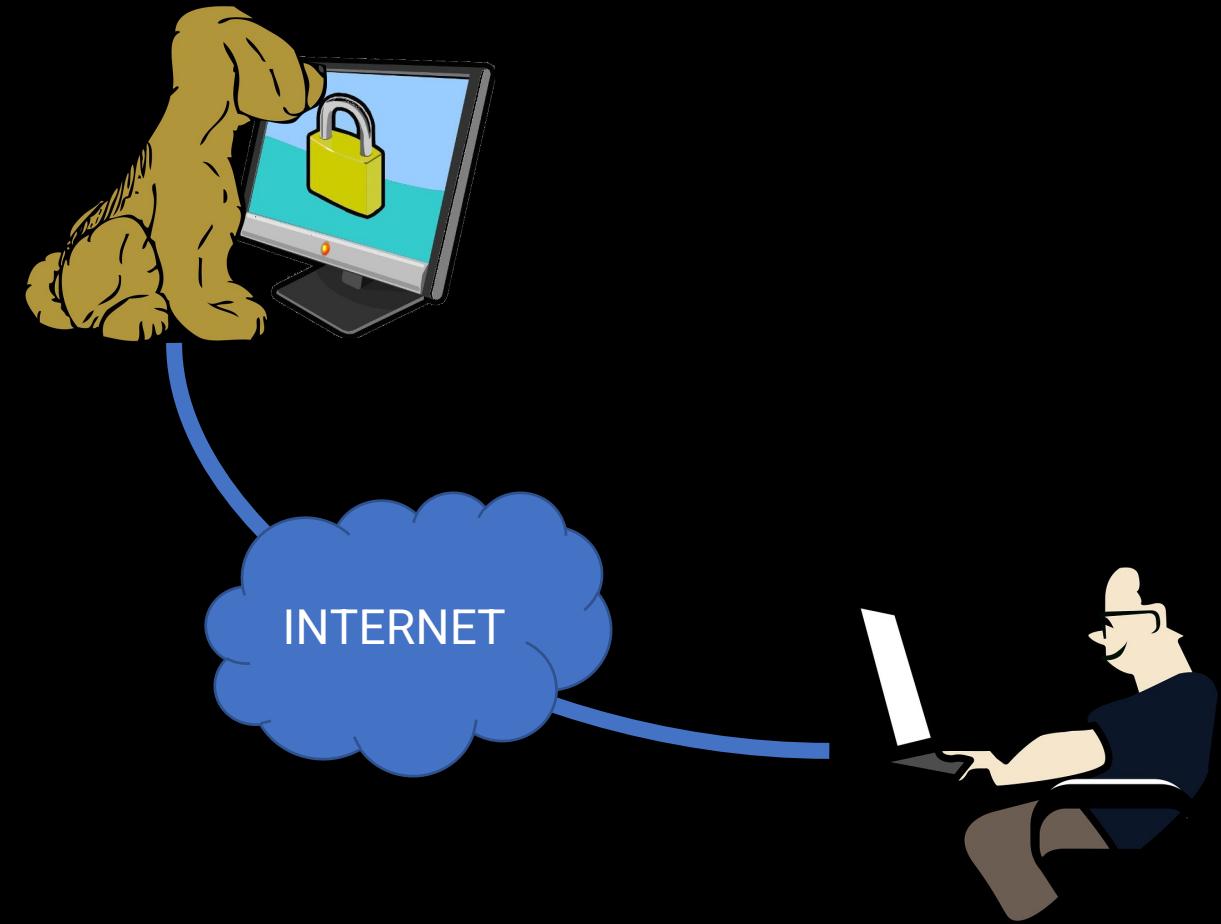
Verificación Firma Digital



Usos

- Autenticación de los mensajes
- Firma de pagos, autorizaciones, documentos, contratos, etc.
- Sellado de documentos
- Servicios de mensajería certificados
- Creación de Certificados Digitales

Certificado Digital



Certificado Digital

- Documento electrónico que asocia una identidad a una clave
- Certificados ≠ Firma
 - Se implementa usando la firma
- Certificados ≠ Autenticación
 - La autenticación puede ser implementada usando certificados digitales
 - Lo mismo para otros servicios como autorización, etc
- Los certificados son estáticos
 - Cambios => Re-emisión



Certificado Digital

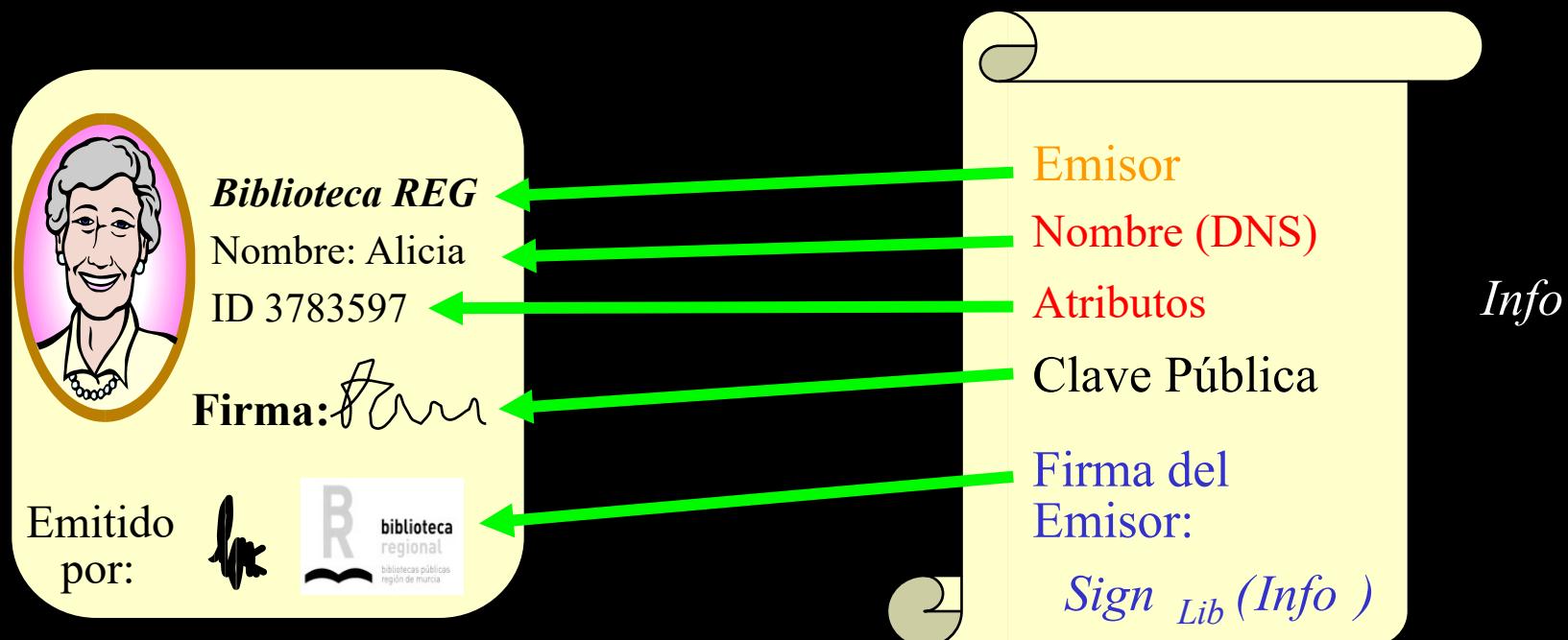
Un certificado digital, es un documento electrónico que contiene la firma digital de una **autoridad de confianza (CA)** del certificado, vincula una clave pública con una identidad y se puede usar para verificar que una clave pública pertenece a una persona o entidad en particular.

Certificado Digital

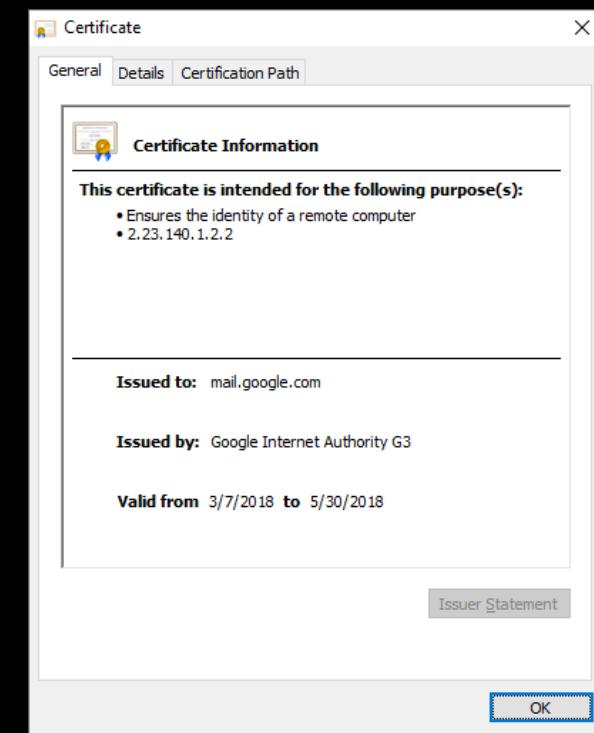
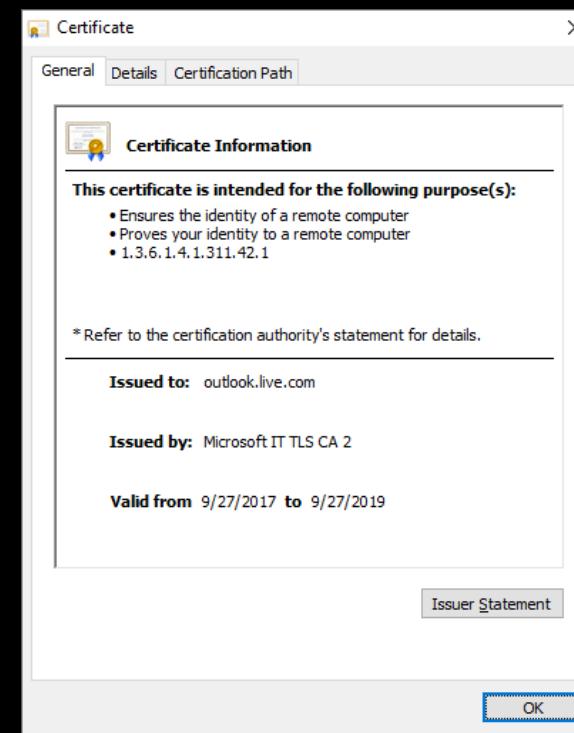
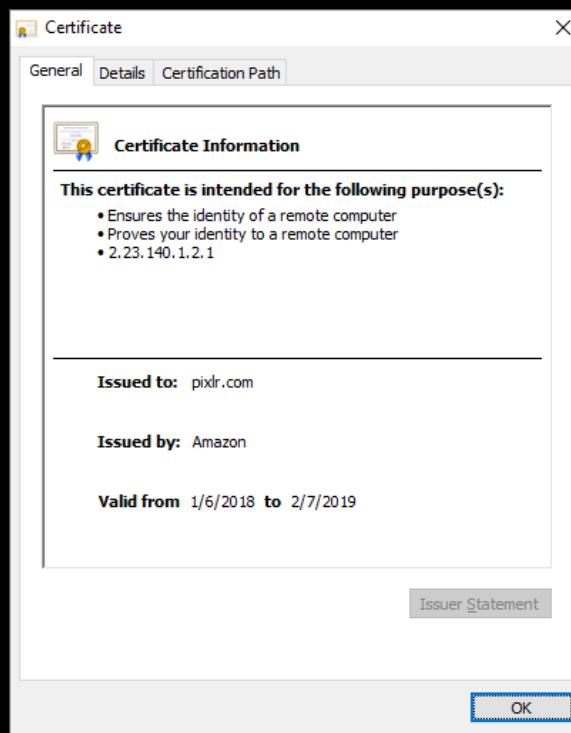
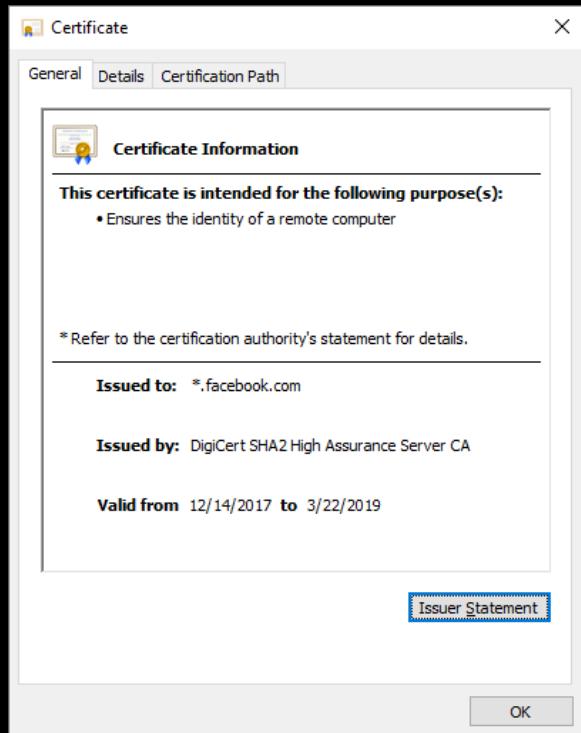
Los certificados sirven para establecer lazos de confianza entre sistemas o personas, ya que si se confía en la autoridad de confianza entonces se puede confiar en la llave pública del dueño del certificado. Tratando así de resolver el problema de relacionar las identidades con las llaves públicas.

Tipos de certificados

X.509, PGP, SPKI/SDSI



Ejemplos Certificados



Infraestructuras de Clave Pública (PKI)

Sistema de publicación de los valores de clave pública utilizados en criptografía asimétrica.

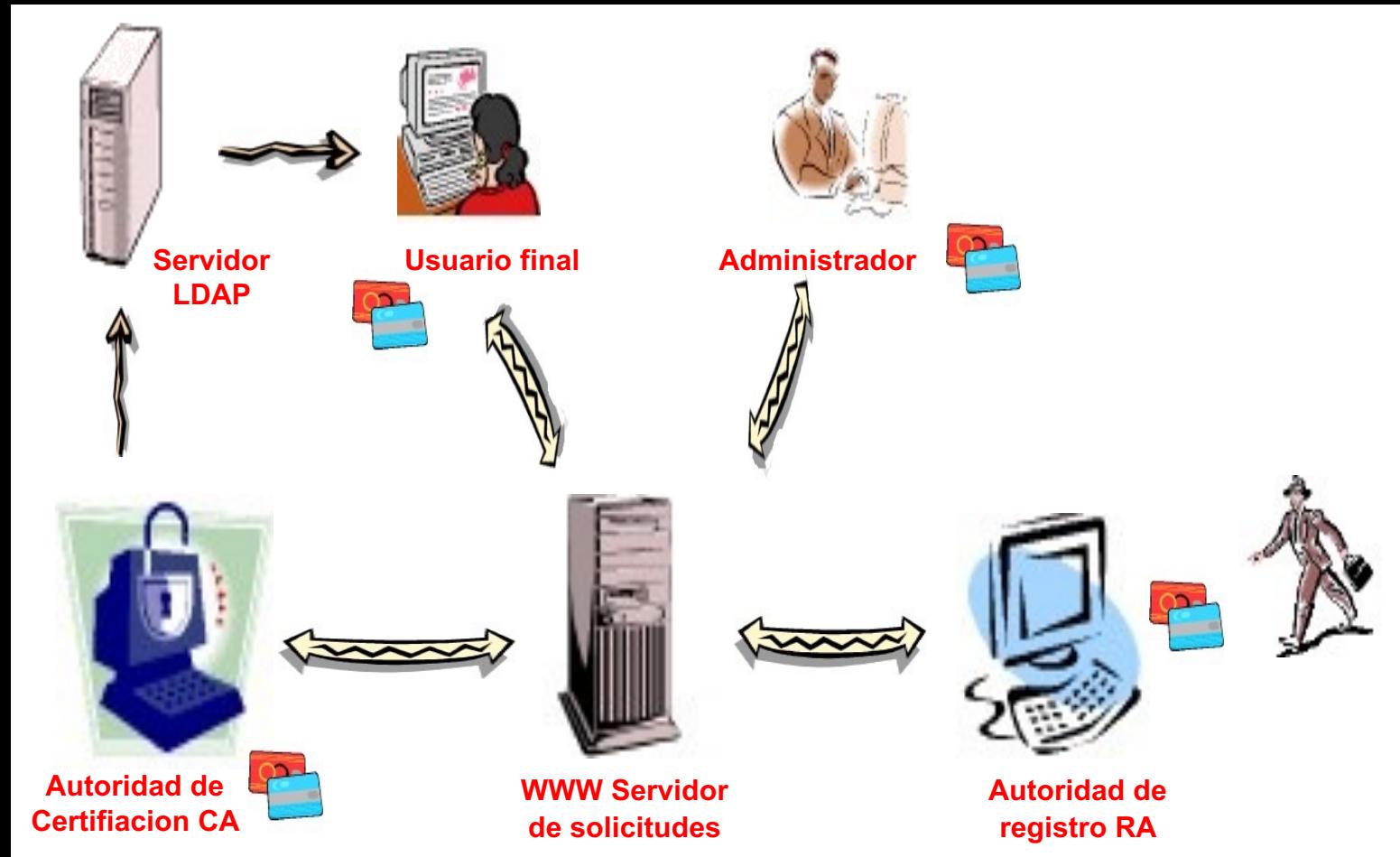
Principales elementos:

- Autoridad de Certificación (CA)
- Autoridad de Registro (RA)
- Repositorio de Certificados
- Entidades Finales

Operaciones básicas:

- Certificación
- Validación
- Revocación
- Publicación y distribución de certificados y de notificaciones de revocación

Infraestructuras de Clave Pública (PKI)



Autoridad de Certificación (CA)

Encargada de dar validez a la información contenida en los certificados

Funciones principales:

- Certificación
- Publicación
- Revocación

La CA nunca puede estar en línea. Su clave privada es
TOP SECRET

Autoridad de Certificación (CA)

- Las CAs se organizan en jerarquías de certificación, estableciendo así cadenas de confianza.
- CAs “globales” o “universales”
- Verisign, Entrust, Identrust, FNMT, ...
- <https://scotthelme.co.uk/static/Feb-2018/caList.txt>
- <https://www.pkicloud.com/ocsp-stats.html>

Autoridades de Registro (RA)

Son un elemento de confianza muy importante

Entidades encargadas de:

- Verificar la información que aparecerá en el certificado.
- Enviar las solicitudes de certificación a la CA
- Gestionar las solicitudes de revocación o de recuperación de claves

Repositorios de Certificados

- Se suele utilizar Servidores de directorios basados en X.500 y LDAP

Entidades a Certificar

Son las entidades que obtienen un certificado firmado por la CA

Los certificados no se limitan sólo a personas (procesos...)

- Los servidores web seguros son procesos certificados.

Dependiendo del tipo de sujeto:

- Certificados de cliente.
- Certificados de proveedor.

Revocación

Una CRL (*Certification Revocation Lists*) es una lista de los certificados revocados, publicada y firmada periódicamente por la CA

El usuario utiliza esta lista para comprobar la validez de los certificados.

Revocación

El usuario utiliza esta lista para comprobar la validez de los certificados.

Signed fields	Version of CRL format
	Signature Algorithm Object Identifier (OID)
	CRL Issuer Distinguished Name (DN)
	This update (date/time)
	Next update (date/time) - optional
	Subject (user) Distinguished Name (DN)
	CRL Entry Certificate Revocation CRL entry Entry Serial Number Date extensions
	CRL Entry... Serial... Date... extensions

	CRL Extensions
	Signature on the above fields

Firma Electrónica Avanzada (FEA)

La Firma Electrónica Avanzada (FEA) permite cifrar/firmar, documentos/mensajes electrónicos, así como enviarlos de forma integra y segura a través de medios electrónicos, con la validez de la firma autógrafa. Conocida algunas veces como e-Firma

FEA - Propósito

- Identificar al emisor del mensaje como autor legítimo del documento/mensaje electrónico.
- Sólo el receptor del documento/mensaje electrónico los pueda descifrar.
- La integridad y autenticidad pueden ser verificadas por cualquier interesado.

FEA en México

La **FIEL** es un ejemplo de Firma Electrónica Avanzada, la cual se expide en las oficinas del SAT en México de manera gratuita tanto para personas físicas como para personas morales. Para su obtención se recaban datos biométricos como los son el scan del iris, la huella digital, la firma autógrafa y una fotografía del interesado.

Composición de la FIEL

La FIEL se compone de dos archivos:

- El Certificado (.CER) forma una tarjeta de identidad y contiene tu nombre, RFC y CURP. También se le llama **Llave pública**, ya que cualquier persona puede tener acceso a ella.
- La Llave privada (.KEY) se encuentra protegida con tu contraseña de acceso y asegura que sólo tú puedes usar tu FIEL para firmar.

Características

- **Integridad.** Protege al identificar cuando el documento/mensaje electrónico original es modificado.
- **No repudio.** Vincula la identidad del autor con el documento/mensaje electrónico.
- **Autenticidad.** Acredita al emisor del documento/mensaje electrónico con la misma validez de una firma autógrafa.
- **Confidencialidad.** Cifra o codifica el envío del documento/mensaje electrónico sólo el receptor designado puede descifrar o decodificar el mensaje.

- TLS/SSL
- SSH
- IPSEC
- Kerberos
- Entre otras aplicaciones, se utiliza (al menos como una opción) en protocolos de Internet como TLS (RFC 2246) para la navegación web segura y SSH (RFC 4251) para el inicio de sesión remoto seguro. También se utiliza en redes inalámbricas WEP, en PPTP (RFC 2637) y en muchas otras aplicaciones.