Instituto Politécnico Nacional

Escuela Superior de Cómputo

Trabajo terminal

Dra. Sandra Díaz Santiago

Generación de tokens para protejer datos de tarjetas bancarias

Número 20180008

Daniel Ayala Zamorano
Laura Natalia Borbolla Palacios
Ricardo Quezada Figueroa

Enero de 2018

Contenido

Capítulo 1

Introducción

1.1. Justificación

1.2. Objetivos

Capítulo 2

Antecedentes

2.1. Introducción a la criptografía

La palabra criptografía proviene de las etimologías griegas *Kriptos* (ocultar) y *Graphos* (escritura), y es definida por la Real Academia Española como el arte de escribir con clave secreta o de un modo enigmático. De manera más formal se puede definir a la criptografía como la ciencia encargada de estudiar y diseñar por medio de técnicas matemáticas métodos y modelos capaces de resolver problemas en la seguridad de la información, como la confidencialidad de esta, su integridad y la autenticación de su origen.

2.1.1. Objetivos de la Criptografía

La criptografía tiene como finalidad cumplir los siguientes cuatro servicios.

1. **Confidencialidad**

   Es el servicio encargado de mantener legible la información solo a aquellos que estén autorizados a visualizarla.

2. **Integridad**

   Este servicio se encarga de evitar la alteración de la información de forma no autorizada, esto incluye la inserción, sustitución y eliminación de los datos.

3. **Autenticación**

   Este servicio se refiere a la identificación tanto de las personas que establece una comunicación, garantizando que cada una es quien dice ser; como del origen de la información que se maneja, garantizando la veracidad de la hora y fecha de origen, el contenido, tiempos de envió, entre otros.

4. **No repudio**

   Es el servicio que evita que el autor de la información o de alguna acción determinada, pueda negar su validez, ayudando así a prevenir situaciones de disputa.

2.1.2. Criptoanálisis y Ataques

La criptografía forma parte de una ciencia más general llamada criptología, la cual tiene otras ramas de estudio, como es el criptoanálisis, que es la ciencia encargada de estudiar los posibles ataques a sistemas criptográficos, que son capaces de contrariar sus servicios ofrecidos. Los ataques que realizan a sistemas criptográficos dependen de la cantidad de recursos o conocimientos con los que cuenta el adversario que realiza dicho ataque, dando así a la siguiente clasificación.

1. **Ciphertext-only attack**

En este ataque el adversario solo es capaz de obtener la información cifrado, y tratara de conocer su contenido en claro a partir de ella. Esta forma de atacar es la más básica, y todos los métodos criptográficos deben poder soportarla.

2. **Known-plaintext attack**

   Esta clase de ataques ocurren cuando el adversario puede obtener pares de información cifrada y su correspondiente información en claro, y por medio de su estudio, trata de descifrar otra información cifrada para la cual no conoce su contenido.

3. **Chosen-plaintext attack**

   Este ataque es muy parecido al anterior, con la diferencia de que en este el adversario es capaz de obtener los pares de información cifrada y en claro con el contenido que desee.

4. **Adaptively-chosen-plaintext attack**

   En este ataque el adversario es capaz de obtener los pares de información cifrada y en claro con el contenido que desee y además tiene amplio acceso o puede usar de forma repetitiva el mecanismo de cifrado.

5. **Chosen and adaptively-chosen-ciphertext attack** En este caso el adversario puede elegir información cifrada y conocer su contenido, dado que tiene acceso a los mecanismos de descifrado.

2.1.3. Clasificación de la criptografía

La criptografía puede clasificarse de forma histórica en dos categorías, la criptografía clásica y la criptografía moderna. La criptografía clásica es aquella que se utilizó desde la antigüedad, teniéndose registro de su usado desde hace más 4000 años por los egipcios, hasta la mitad del siglo XX. En esta los métodos utilizados para cifrar eran variados, pero en su mayoría usaban la transposición y la sustitución, además de que la mayoría se mantenían en secretos. Mientras que la criptografía moderna es la que se inició después la publicación de la *Teoría de la información* por Claude Elwood Shannon, dado que esta sentó las bases matemáticas para la criptología en general.

Una manera de clasificar es de acuerdo a las técnicas y métodos empleados para cifrar la información, esta clasificación se puede observar en la siguiente figura.

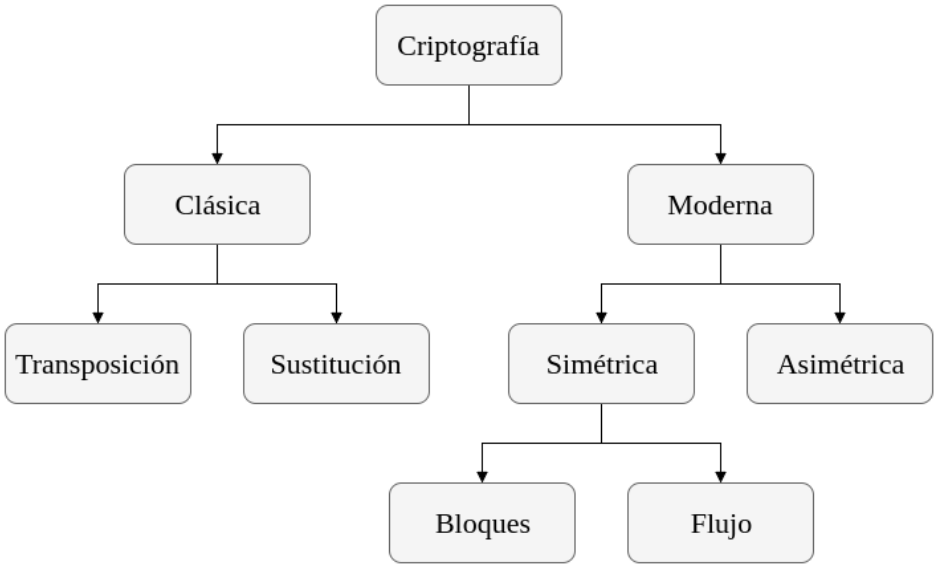


Figura 2.1: Clasificación de la criptografía.

Entrando dentro de la clasificación de la criptografía clásica, se tienen los cifrados por transposición, los cuales se basan en técnicas de permutación de forma que los caracteres de la información en claro se reordenen mediante algoritmos específicos, y los cifrados por sustitución, que utilizan técnicas de modificación de los caracteres por otros correspondiente a un alfabeto específico para el cifrado.

En cuanto a la criptografía moderna, esta tiene dos vertientes, la criptografía simétrica o de llave secreta y la asimétrica o de llave publica. Hablando de la primer vertiente, se puede decir que es aquella que utiliza un modelo matemático para cifrar y descifrar un mensaje utilizando únicamente una llave que permanece secreta.

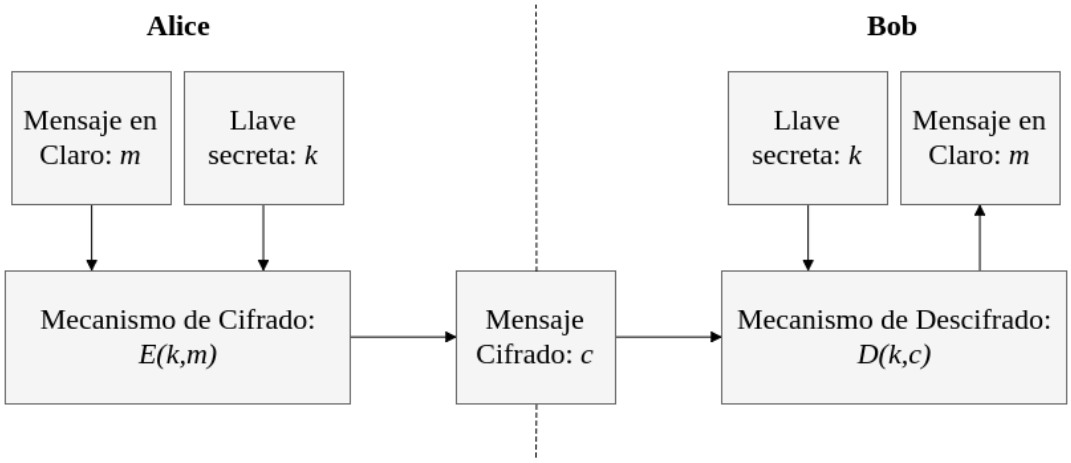


Figura 2.2: Canal de comunicación con criptografía simétrica.

En la figura anterior se puede observar el proceso para establecer una comunicación segura por medio de la criptografía simétrica. Primero, tanto Alice como Bob deben de establecer una llave única y compartida k,para que después, Alice, actuando como el emisor, cifra un mensaje m usando la llave k por medio del algoritmo de cifrado $E(k, m)$ para obtener el mensaje cifrado c y enviárselo a Bob. Posteriormente Bob, como receptor, se encarga de descifrar c con ayuda de la llave k por medio del algoritmo de descifrado $D(k, c)$ para obtener el mensaje original m.

Entre los beneficios de este tipo de criptografía está su utilidad para cifrar archivos personales, su relativa facilidad de uso y para garantizar la confidencialidad e integridad debido a el uso de una llave, y su rapidez, pero en contraparte, su uso genera problemas para organizar y compartir las llaves secretas de una forma segura y eficiente.

Ahora, adentrándose en la criptografía asimétrica, se tiene que su idea principal es el uso de 2 llaves distintas para cada persona, una llave pública para cifrar que este disponible para cualquier otra persona, y una llave privada para descifrar, que se mantiene disponible solo para su propietario.

El proceso para establecer una comunicación segura por medio de este tipo de criptografía es el siguiente, primero, Alice nuevamente como el emisor, cifra un mensaje m con la llave publica de de Bob pk usa el algoritmo de cifrado $E(pk, m)$ para obtener c y enviarlo. Después Bob como receptor, se encarga de descifrar c por medio del algoritmo de descifrado $D(sk, c)$ haciendo uso de su llave privada sk. Este proceso se refleja gráficamente el la siguiente figura.

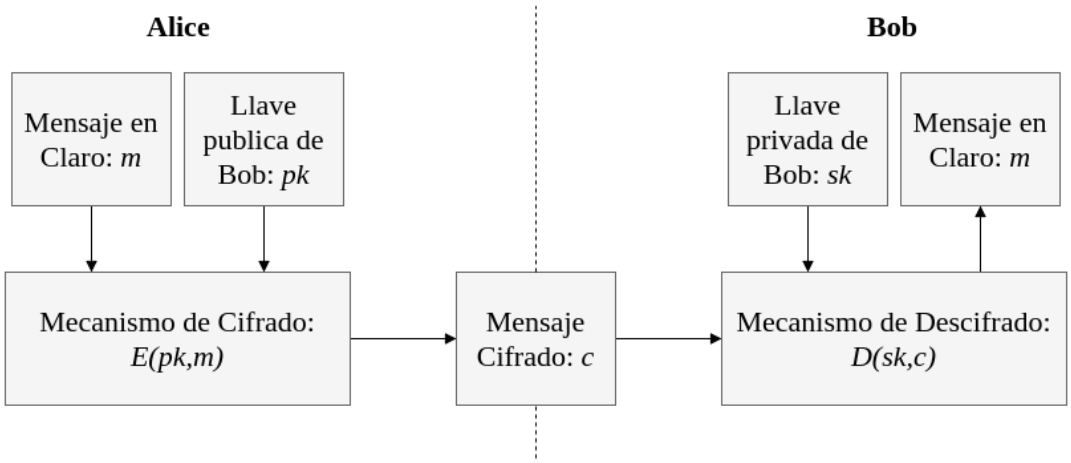


Figura 2.3: Canal de comunicación con criptografía asimétrica.

Entre los uso que se le da a esta criptografía está el mantener la distribución de llaves privada segura, y establecer métodos que garantizan la autenticación y el no repudio, como por ejemplo en las firmas y certificados digitales.

2.2. Cifrados por bloques

2.3. Modos de operación

Por sí solos, los cifrados por bloques solamente permiten el cifrado y descifrado de bloques de información de tamaño fijo. Para la mayoría de los casos, menos de 256 bits[1], lo cual es equivalente a alrededor de 8 caracteres. Es fácil darse cuenta de que esta restricción no es ningún tema menor: en la gran mayoría de las aplicaciones, la longitud de lo que se quiere ocultar es arbitraria.

Los modos de operación permiten extender la funcionalidad de los cifrados por bloques para poder aplicarlos a información de tamaño irrestricto. Formalizamos este concepto definiendo a un cifrado por bloques como una función C (ecuación 2.1) y a un modo de operación como una función M (ecuación 2.2).

$$C(L, B) \rightarrow Bc \tag{2.1}$$

En donde L es la llave y B es el bloque a cifrar; ambos con un tamaño definido: $L \in \{0,1\}^k$ (k es el tamaño de la llave) y $B \in \{0,1\}^n$ (n es el tamaño de bloque). Bc representa al bloque cifrado, el cuál también tiene longitud n.

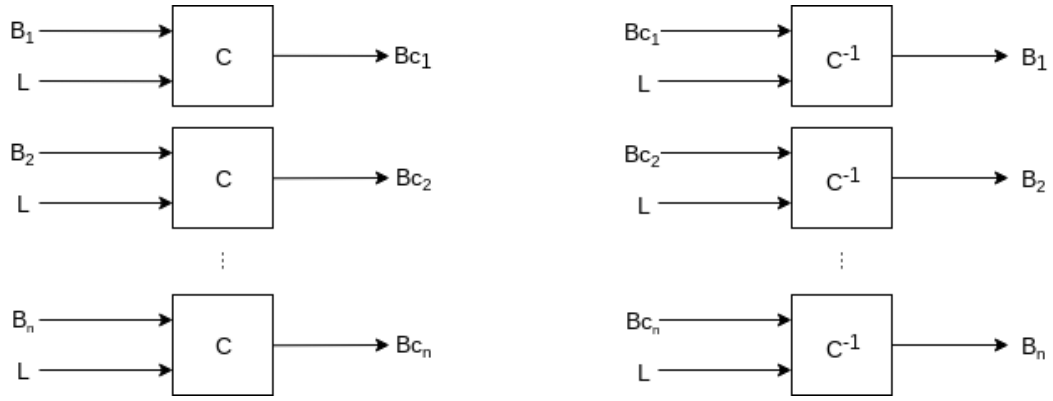
$$M(L, T) \rightarrow Tc \tag{2.2}$$

En este caso L es la misma que en 2.1, T y Tc son el texto original y el texto cifrado, respectivamente, y ambos son de longitud arbitraria: $T, Tc \in \{0,1\}^*$.

Un primer enfoque (y quizás el más intuitivo) es partir el mensaje original en bloques del tamaño requerido y después aplicar el algoritmo a cada bloque por separado; en caso de que la longitud del mensaje no sea múltiplo del tamaño de bloque, se puede agregar información extra al último bloque para completar el tamaño requerido. Este es, de hecho, el primero de los modos que presentamos a continuación (*Electronic Codebook*, ECB); su uso no es recomendado, pues es muy inseguro cuando el mensaje original es simétrico a nivel de bloque [1]. También presentamos otros tres modos, los cuales junto con ECB, son los más comunes.

2.3.1. *Electronic Codebook* (ECB)

La figura 2.4 muestra un diagrama esquemático de este modo de operación. Según la ecuación 2.2, el algoritmo recibe a la entrada una llave y un mensaje de longitud arbitraria: la llave se pasa sin ninguna modificación a cada función del cifrado por bloques; el mensaje se debe de partir en bloques ($T = B_1||B_2||\ldots||B_n$).



(a) Cifrado. (b) Descifrado.

Figura 2.4: Modo de operación ECB.

```
1  entrada: llave L; bloques de mensaje B_1, B_2 ... B_n.
2   salida: bloques de mensaje cifrado Bc_1, Bc_2 ... Bc_n.
3  inicio
4    para_todo B
5      Bc_i ← C(L, B_i)
6    fin
7    regresar Bc
8  fin
```

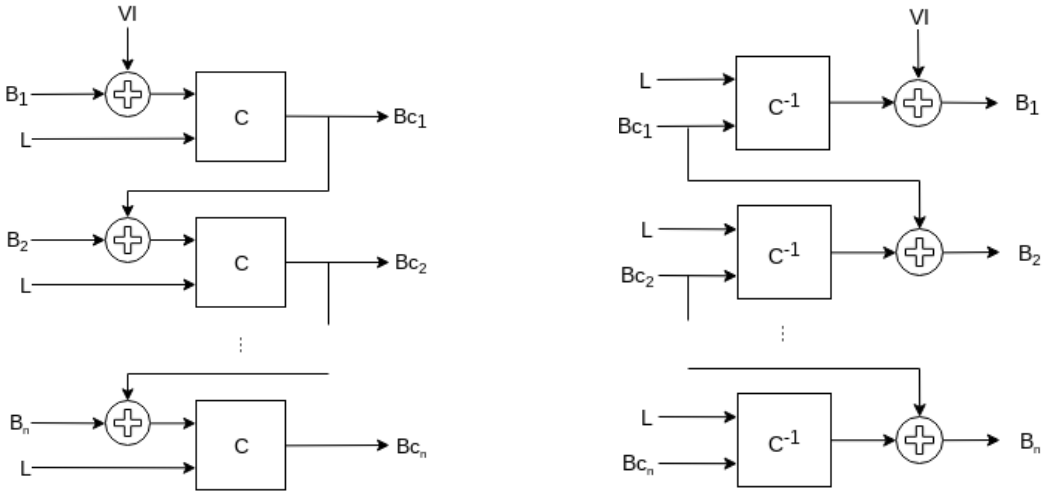
Pseudocódigo 2.1: Modo de operación ECB, cifrado.

```
1  entrada: llave L; bloques de mensaje cifrado Bc_1, Bc_2 ... Bc_n.
2   salida: bloques de mensaje original B_1, B_2 ... B_n.
3  inicio
4    para_todo Bc
5      B_i ← C^-1(L, Bc_i)
6    fin
7    regresar B
8  fin
```

Pseudocódigo 2.2: Modo de operación ECB, descifrado.

2.3.2. *Cipher-block Chaining* (CBC)

En CBC la salida del bloque cifrador uno se introduce (junto con el siguiente bloque del mensaje) en el bloque cifrador dos, y así en sucesivo. Para poder replicar este comportamiento en todos los bloque cifradores, este modo de operación necesita un argumento extra a la entrada: un vector de inicialización. De esta manera la salida del bloque i depende de todos los bloques anteriores; esto incrementa la seguridad con respecto a ECB.



(a) Cifrado. (b) Descifrado.

Figura 2.5: Modo de operación CBC.

En la figura 2.5 se muestran los diagramas esquemáticos para cifrar y descifrar; en los pseudocódigos 2.3 y 2.4 se muestran unos de los posibles algoritmos a seguir. Es importante notar que mientras que el proceso de cifrado debe ser forzosamente secuencial (por la dependencias entre salidas), el proceso de descifrado puede ser ejecutado en paralelo.

```
1   entrada: llave L; vector de inicialización VI;
2            bloques de mensaje B_1, B_2 ... B_n.
3    salida: bloques de mensaje cifrado Bc_1, Bc_2 ... Bc_n.
4   inicio
5     Bc_0 ← VI                  // El vector de inicialización
6     para_todo B                // entra al primer bloque.
7       Bc_i ← C(L, B_i ⊕ Bc_{i-1})
8     fin
9     regresar Bc
10  fin
```

Pseudocódigo 2.3: Modo de operación CBC, cifrado.

```
1  entrada: llave L; vector de inicialización VI;
2           bloques de mensaje cifrado Bc_1, Bc_2 ... Bc_n.
3   salida: bloques de mensaje original B_1, B_2 ... B_n.
4  inicio
5    Bc_0 ← VI
6    para_todo Bc
7      B_i ← C^{-1}(L, Bc_i) ⊕ Bc_{i-1}
8    fin
9    regresar B
10 fin
```

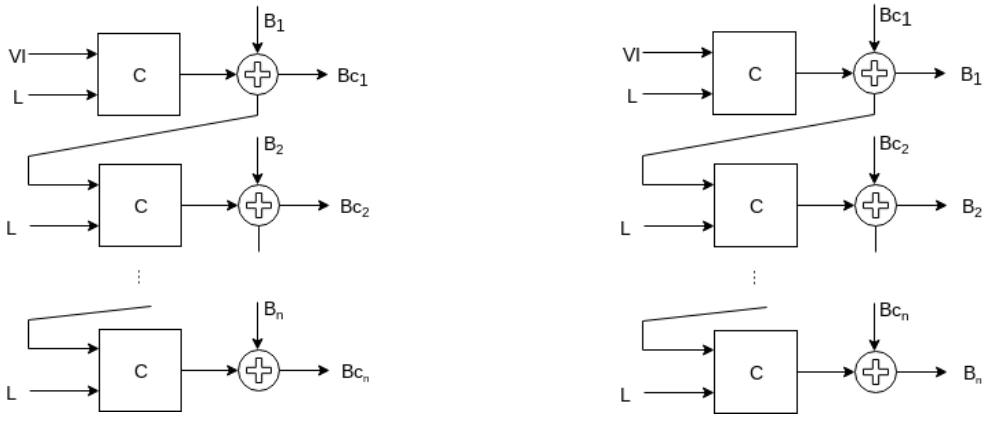
Pseudocódigo 2.4: Modo de operación CBC, descifrado.

2.3.3. *Cipher Feedback* (CFB)

Al igual que la operación de cifrado de CBC, ambas operaciones de CFB (cifrado y descifrado) están encadenadas bloque a bloque, por lo que son de naturaleza secuencial. En este caso, lo que se cifra en el primer paso es el vector de inicialización; la salida de esto se opera con un `xor` sobre el primer bloque de texto en claro, para obtener el primer bloque cifrado (figura 2.6).

Esta distribución presenta varias ventajas con respecto a CBC: las operaciones de cifrado y descifrado son sumamente similares, lo que permite ser implementadas por un solo algoritmo (pseudocódigo 2.5); tanto para cifrar como para descifrar solamente se ocupa la operación de cifrado del algoritmo a bloques subyacente. Estas ventajas se deben principalmente a las propiedades de la operación `xor` (ecuación 2.3).

$$A \oplus B = C \quad \Rightarrow \quad A = B \oplus C \tag{2.3}$$



(a) Cifrado. (b) Descifrado.

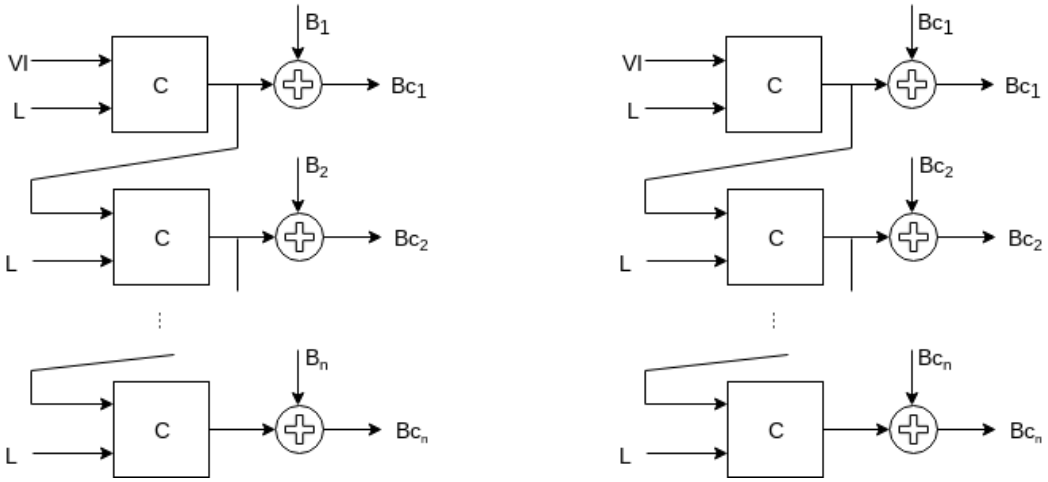
Figura 2.6: Modo de operación CFB.

```
1  entrada: llave L; vector de inicialización VI;
2           bloques de mensaje (cifrado o descifrado) B_1, B_2 ... B_n.
3   salida: bloques de mensaje (cifrado o descifrado) Bc_1, Bc_2 ... Bc_n.
4  inicio
5    Bc_0 ← VI
6    para_todo B
7      Bc_i ← C(L, Bc_{i-1}) ⊕ B_i
8    fin
9    regresar Bc
10 fin
```

Pseudocódigo 2.5: Modo de operación CFB (cifrado y descifrado).

2.3.4. *Output Feedback* (OFB)

Este es muy similar al anterior (CFB), salvo porque la retroalimentación va directamente de la salida del cifrador a bloques. De esta forma, nada que tenga que ver con el texto en claro, llega al cifrado a bloques; este solamente se la pasa cifrando una y otra vez el vector de inicialización.



(a) Cifrado. (b) Descifrado.

Figura 2.7: Modo de operación OFB.

```
1  entrada: llave L; vector de inicialización VI;
2           bloques de mensaje (cifrado o descifrado) B_1, B_2 ... B_n.
3   salida: bloques de mensaje (cifrado o descifrado) Bc_1, Bc_2 ... Bc_n.
4  inicio
5    auxiliar ← VI
6    para_todo B
7      auxiliar ← C(L, auxiliar)
8      Bc_i ← auxiliar ⊕ B_i
9    fin
10   regresar Bc
11 fin
```

Pseudocódigo 2.6: Modo de operación OFB (cifrado y descifrado).

2.4. Cifrados de flujo

A diferencia de los cifrados de bloque, que trabajan sobre grupos enteros de bits a la vez, los cifrados de flujo trabajan sobre bits individuales, cifrándolos uno por uno. Una manera de verlos es como cifrados por bloques con un tamaño de bloque igual a 1.

Un cifrado de flujo aplica transformaciones de acuerdo a un flujo de llave: una secuencia de símbolos pertenecientes al espacio de llaves. El flujo de llave puede ser generado de manera aleatoria, o por un algoritmo pseudoaleatorio que toma una semilla a la entrada, o por una semilla y símbolos cifrados anteriormente.

Entre las ventajas de los cifrados de flujo sobre los cifrados de bloque se encuentran el hecho de que son más rápidos en hardware, son más útiles cuando el buffer es limitado o se necesita procesar la información al momento de llegada. La propagación de los errores es limitada o nula, por lo que también son más útiles en casos en los que hay probabilidades altas de errores en la transmisión [2].

Los cifrados de bloques funcionan sin ninguna clase de memoria (por sí solos); en contraste, la función de cifrado de un cifrado de flujo puede variar mientras se procesa el texto en claro, por lo cuál tienen un mecanismo de memoria asociado. Otra denominación para estos cifrados es *de estado*, por que la salida no depende solamente del texto en claro y de la llave, sino que también depende del estado actual.

Una clasificación común es en *síncronos* y en *autosincronizables*. A continuación describimos a grandes rasgos ambos modelos.

2.4.1. Síncronos

Un cifrado de flujo síncrono es aquel en el que el flujo de la llave es generado de manera independiente del texto en claro y del texto cifrado. Se puede definir un modelo general con las siguientes tres ecuaciones.

$$e_{i+1} = f(e_i, L) \tag{2.4}$$

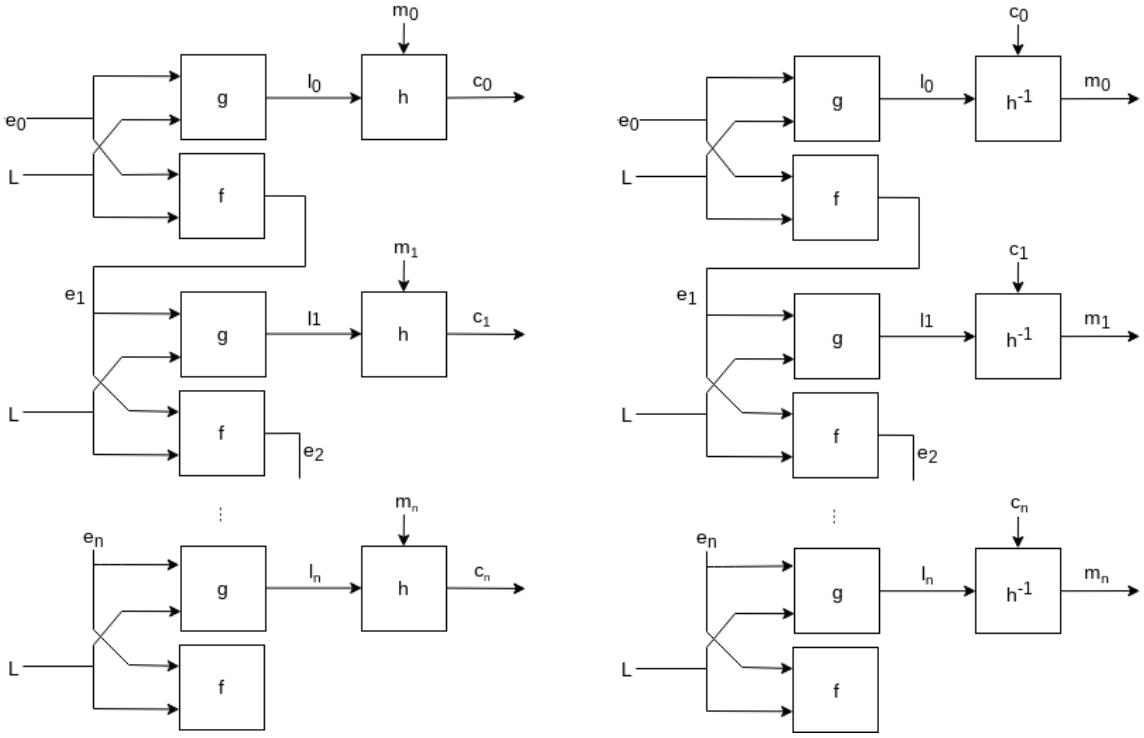
$$l_i = g(e_i, L) \tag{2.5}$$

$$c_i = h(l_i, m_i) \tag{2.6}$$

La letra e representa el estado del cifrado,L es la llave, l es la salida del flujo de llave, c es el texto cifrado y m es el texto en claro. La función de la ecuación 2.4 (f) es la que describe el cambio de estado; este se determina a

partir del estado actual y de la llave. En la ecuación 2.5 se describe la acción del flujo de llave (g): para determinar el próximo símbolo se emplea solamente el estado actual y la llave. La tercera ecuación (2.6, h) describe la acción de combinar el flujo de la llave con el mensaje, y así obtener el texto cifrado.

En la figura 2.8 se describe de manera gráfica las operaciones de cifrado y descifrado; estas guardan muchas similitudes con el modo de operación OFB, con la única excepción de que este trabaja con bloques del tamaño del cifrado subyacente. En otras palabras, si definiéramos el tamaño del bloque (y en consecuencia el tamaño del vector de inicialización) como 1, entonces OFB sería un cifrado de flujo síncrono.



(a) Cifrado.

(b) Descifrado.

Figura 2.8: Esquema general de un cifrado de flujo síncrono.

El nombre de esta categoría proviene del hecho de que ambos entes del proceso comunicativo (emisor y receptor) deben encontrarse sincronizados (usar la misma llave y encontrarse en la misma posición) para que la comunicación tenga éxito: si se insertan dígitos extras al mensaje cifrado, la sincronización se pierde. Los cifrados de flujo síncronos no tienen propagación de error: aunque ciertos bits sean modificados (pero no borrados) durante su transmisión, el resto del mensaje sigue siendo descifrable.

2.4.2. Autosincronizables

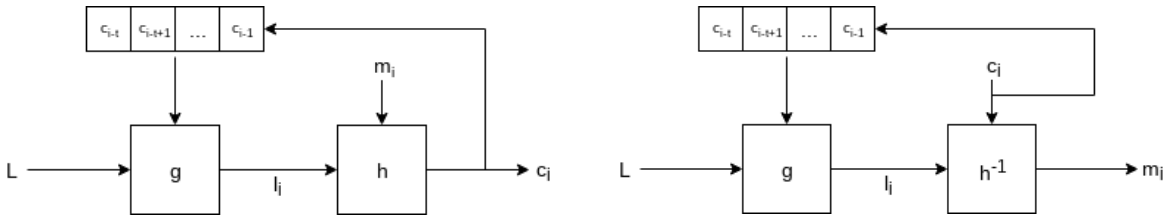
En esta clasificación se engloban a aquellos cifrados cuyo flujo de llave es resultado de la propia llave original y de cierto número previo de dígitos cifrados. Las ecuaciones que describen su comportamiento son las siguientes.

$$e_{i+1} = (c_{i-t}, c_{i-t+1}, \ldots, c_{i-1}) \tag{2.7}$$

$$l_i = g(e_i, L) \tag{2.8}$$

$$c_i = h(l_i, m_i) \tag{2.9}$$

La notación es la misma que en las ecuaciones 2.4, 2.5 y 2.6. En este caso, el próximo estado depende de t (el tamaño de la ventana) dígitos cifrados anteriormente. En la figura 2.9 se describe de manera gráfica el proceso de cifrado y descifrado.



(a) Cifrado. (b) Descifrado.

Figura 2.9: Esquema general de un cifrado de flujo autosincronizable.

En una antítesis de la categoría anterior, el nombre de esta indica que no es necesario que el emisor y el receptor estén sincronizados: si se llegan a perder bits en la transmisión, el esquema es capaz de autosincronizarse, pues el flujo de la llave depende de cierto número de bits anteriores. A esta categoría también se le conoce como «asíncrona».

La propagación de los errores depende del tamaño de ventana (el número t de bits previos utilizados para calcular la próxima llave), si se modifica un bit, entonces los próximos t serán incorrectos.

2.5. Funciones hash

Bibliografía

[1] Debrup Chakraborty y Francisco Rodríguez-Henríquez. "Block Cipher Modes of Operation from a Hardware Implementation Perspective". En: *Cryptographic Engineering*. Ed. por Çetin Kaya Koç. Springer, 2009, págs. 321-363. ISBN: 978-0-387-71816-3. DOI: 10.1007/978-0-387-71817-0_12. URL: https://doi.org/10.1007/978-0-387-71817-0_12.

[2] Alfred Menezes, Paul C. van Oorschot y Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996. ISBN: 0-8493-8523-7.

Lista de figuras

Lista de tablas

Lista de pseudocódigos