8.2 Algoritmos de Clave Simétrica

La criptografía moderna usa las mismas ideas básicas que la criptografía tradicional (la transposición y la sustitución), pero su orientación es distinta. La primera clase de algoritmos de encriptación que analizaremos en este capítulo se conocen como algoritmos de clave simétrica porque utilizan la misma clave para encriptar y desencriptar. En particular, nos enfocaremos en los cifrados en bloques, que toman un bloque de n bits de texto llano como entrada y lo transforman utilizando la clave en un bloque de n bits de texto cifrado.

8.2.1 DES—El Estándar de Encriptación de Datos

Este cifrado, el DES (Estándar de Encriptación de Datos), se adoptó ampliamente en la industria para usarse con productos de seguridad. Ya no es seguro en su forma original, pero aún es útil en una forma modificada. El algoritmo, que se parametriza mediante una clave de 56 bits, tiene 19 etapas diferentes. La primera etapa es una transposición, independiente de la clave, del texto llano de 64 bits. La última etapa es el inverso exacto de esta transposición. La etapa previa a la última intercambia los 32 bits de la izquierda y los 32 bits de la derecha. Las 16 etapas restantes son funcionalmente idénticas, pero se parametrizan mediante diferentes funciones de la clave. El algoritmo se ha diseñado para permitir que la desencriptación se haga con la misma clave que la encriptación. Los pasos simplemente se ejecutan en el orden inverso.

8.2.2 AES—El Estándar de Encriptación Avanzada

El NIST adoptó una estrategia sorprendentemente diferente para una burocracia gubernamental: promovió un concurso. En enero de 1997, los investigadores de todo el mundo fueron invitados a emitir propuestas para un nuevo estándar, que se llamaría AES (Estándar de Encriptación Avanzada). Las reglas fueron:

1. El algoritmo debe ser un cifrado de bloques simétricos.
2. Todo el diseño debe ser público.
3. Deben soportarse las longitudes de claves de 128, 192 y 256 bits.
4. Deben ser posibles las implementaciones tanto de software como de hardware.
5. El algoritmo debe ser público o con licencia en términos no discriminatorios. Se realizaron quince propuestas serias y se organizaron conferencias para presentarlas, en las cuales se alentó activamente a los asistentes a que encontraran errores en todas ellas. En noviembre de 2001 Rijndael se volvió un estándar del gobierno de Estados Unidos publicado como FIPS 197 (Estándar Federal para el Procesamiento de Información).

Rijndael

Desde una perspectiva matemática, Rijndael se basa en la teoría de campos de Galois, la cual da algunas propiedades verificables de seguridad. Sin embargo, también puede verse como código C, sin meterse a las matemáticas. Al igual que el DES, Rijndael utiliza sustitución y permutaciones, así como múltiples rondas. El número de rondas depende del tamaño de clave y del tamaño de bloque, y es de 10 para las claves de 128 bits con bloques de 128 bits y aumenta hasta 14 para la clave o el bloque más grande. Sin embargo, a diferencia del DES, todas las operaciones involucran bytes completos, para permitir implementaciones eficientes tanto en hardware como en software. La función rijndael tiene tres parámetros: plaintext, un arreglo de 16 bytes que contiene los datos de entrada; ciphertext, un arreglo de 16 bytes a donde se regresará la salida cifrada, y key, la clave de 16 bytes.

8.2.3 Modos de cifrado

Modo de libro de código electrónico

Para ver cómo puede utilizarse esta propiedad de cifrado de sustitución monoalfabética para vencer parcialmente el cifrado, utilizaremos el (triple) DES porque es más fácil diseñar bloques de 64 bits que de 128 bits, pero el AES tiene exactamente el mismo problema. La forma directa de utilizar el DES para cifrar una pieza grande de texto llano es dividirla en bloques consecutivos de 8 bytes (64 bits) y encriptarlos después uno tras otro con la misma clave. La última pieza de texto llano se rellena a 64 bits, en caso de ser necesario. Esta técnica se conoce como modo ECB (modo de Libro de Código Electrónico) en analogía con los libros de código pasados de moda en los que se listaba cada palabra de texto llano, seguida por su texto cifrado (por lo general, un número decimal de cinco dígitos).

Modo de encadenamiento de bloques de cifrado

Para frustrar este tipo de ataque, todos los cifrados en bloques pueden encadenarse de varias formas a fin de que el reemplazo de un bloque de la forma en que lo hizo Leslie cause que el texto llano se desencripte comenzando en el bloque reemplazado que se desechará. Una forma de encadenar es el encadenamiento de bloques de cifrado. En este método, a cada bloque de texto llano se le aplica un OR exclusivo con el bloque anterior de texto cifrado antes de ser encriptado. En consecuencia, el mismo bloque de texto llano ya no corresponde con el mismo bloque de texto cifrado, y la encriptación deja de ser un enorme cifrado de sustitución monoalfabética. Al primer bloque se le aplica un OR exclusivo con un IV (Vector de Inicialización) elegido de manera aleatoria, que se transmite (en texto llano) junto con el texto cifrado.

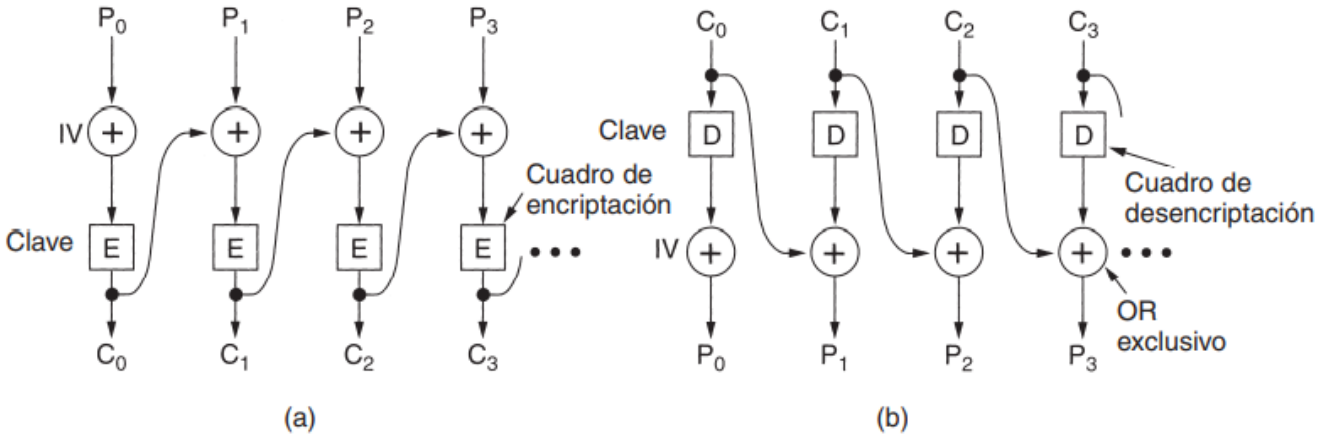

Figura 8-12. Encadenamiento de bloques de cifrado. (a) Encriptación. (b) Desencriptación.

Modo de retroalimentación de cifrado

El encadenamiento de bloques tiene la desventaja de que requiere la llegada de un bloque completo de 64 bits antes de que pueda comenzar la desencriptación. Este modo no es adecuado para terminales interactivas, en las que se pueden escribir líneas máximo de ocho caracteres y es necesario detenerse a esperar una respuesta. Para la encriptación byte por byte, modo de retroalimentación de cifrado, se utiliza (triple) DES. Para el AES la idea es exactamente la misma; sólo se utiliza un registro de desplazamiento de 128 bits. En esta figura se muestra el estado de la máquina de encriptación después de que se han encriptado y enviado los bytes 0 a 9. Cuando llega el byte 10 de texto llano, como se ilustra en la figura 8-13(a), el algoritmo DES opera en el registro de desplazamiento de 64 bits para generar texto cifrado de 64 bits. El byte más a la izquierda de ese texto cifrado se extrae y se le aplica un OR exclusivo con P10. Ese byte se transmite en la línea de

transmisión. La desencriptación con el modo de retroalimentación de cifrado hace lo mismo que la encriptación.

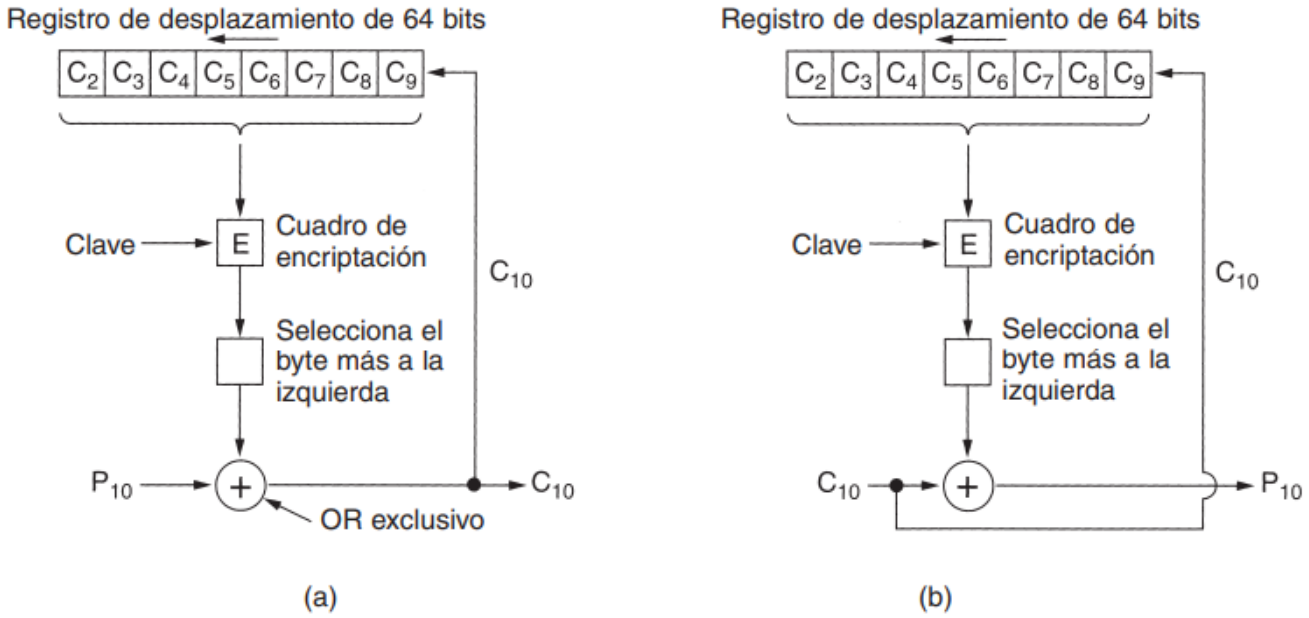

Figura 8-13. Modo de retroalimentación de cifrado. (a) Encriptación. (b) Desencriptación.

Un problema con el modo de retroalimentación de cifrado es que si un bit del texto cifrado se invierte de manera accidental durante la transmisión, se dañarán los 8 bytes que se desencriptan mientras el byte incorrecto se encuentra en el registro de desplazamiento.

Modo de cifrado de flujo

Existen aplicaciones en las que un error de transmisión de 1 bit que arruina 64 bits de texto llano es demasiado. Para estas aplicaciones, existe una cuarta opción, el modo de cifrado de flujo. Funciona encriptando un vector de inicialización y usando una clave para obtener un bloque de salida. A continuación se encripta este bloque usando la clave para obtener un segundo bloque de salida. A continuación este bloque se encripta para obtener un tercer bloque, y así sucesivamente. La secuencia (arbitrariamente grande) de bloques de salida, llamada flujo de claves, se trata como un relleno de una sola vez y se le aplica OR exclusivo con el texto llano para obtener el texto cifrado.

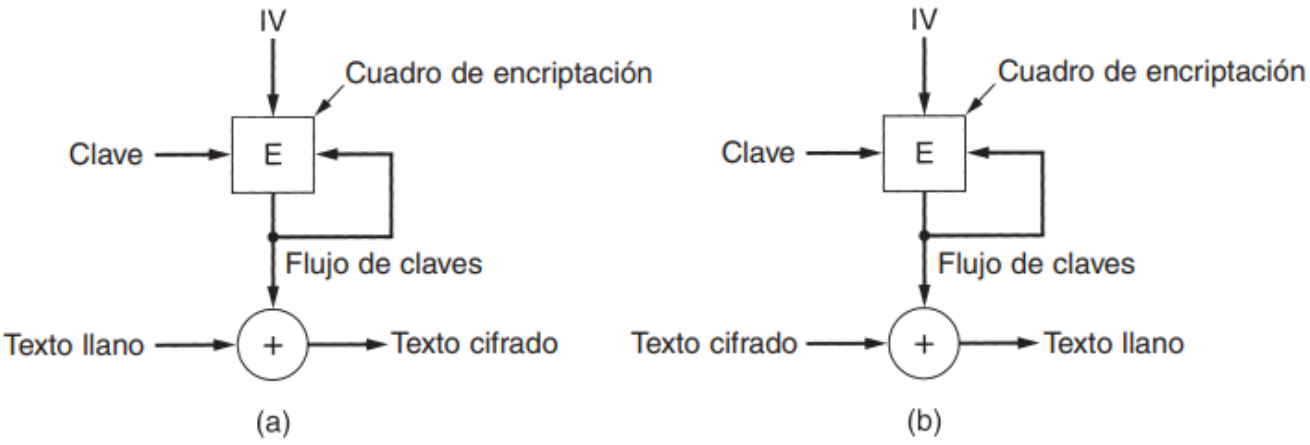

Figura 8-14. Un cifrado de flujo. (a) Encriptación. (b) Desencriptación.

Modo de contador

Un problema que todos los modos tienen, excepto el modo de libro de código electrónico, es que el acceso aleatorio a datos encriptados es imposible. Con un archivo encriptado mediante encadenamiento de bloques de cifrado, el acceso a un bloque aleatorio requiere que primero se desencripten todos los bloques que estén delante de él, lo cual es muy costoso. Por esta razón, se ha inventado otro modo, el modo de contador, como se ilustra en la figura 8-15. Aquí el texto llano no se encripta en forma directa. En su lugar, se encripta el vector de inicialización más una constante, y al texto cifrado resultante se le aplica un OR exclusivo con el texto llano. Al incrementar en 1 el vector de inicialización por cada nuevo bloque, es más fácil desencriptar un bloque en cualquier parte del archivo sin tener que desencriptar primero todos sus predecesores.

8.2.4 Otros cifrados

DES y Rijndael son los algoritmos criptográficos de clave simétrica más conocidos. Sin embargo, vale la pena mencionar que se han diseñado otros cifrados de clave simétrica. Algunos de ellos están incluidos en varios productos.

8.2.5 Criptoanálisis

El primero es el criptoanálisis diferencial (Biham y Shamir, 1993). Esta técnica puede utilizarse para atacar cualquier cifrado en bloques. El segundo avance que vale la pena mencionar es el criptoanálisis lineal (Matsui, 1994). Éste puede descifrar el DES con sólo 243 textos llanos conocidos. Funciona aplicando un OR exclusivo a ciertos bits del texto llano y el texto cifrado en conjunto y buscando patrones en el resultado. El tercer avance es el análisis del consumo de energía eléctrica para averiguar las claves secretas. Las computadoras por lo general utilizan 3 voltios para representar un bit 1 y 0 voltios para representar un bit 0. Por lo tanto, procesar un 1 gasta más energía eléctrica que procesar un 0. El cuarto avance es el análisis de temporización. Los algoritmos criptográficos están llenos de instrucciones if que prueban bits en las claves de ronda.

8.3 Algoritmos de Clave Pública

Históricamente el problema de distribución de claves siempre ha sido la parte débil de la mayoría de los criptosistemas.Los criptologos siempre daban por hecho que las claves de encriptación y desencriptación eran la misma. Pero la clave tenía que distribuirse a todos los usuarios del sistema. Por lo tanto, parecía haber un problema inherente: las claves se tenían que proteger contra robo y se tenían que distribuir. En 1976,propusieron una clase nueva de criptosistema, en el que las claves de encriptación y desencriptación eran diferentes y la clave de desencriptación no podía derivarse de la clave de encriptación.En su propuesta, el algoritmo de encriptación (con clave), E, y el algoritmo de desencriptación (con clave), D, tenían que cumplir con los tres requisitos siguientes. Estos requisitos pueden expresarse sencillamente como sigue:

1. D(E(P)) = P.
2. Es excesivamente difícil deducir D de E.
3. E no puede descifrarse mediante un ataque de texto llano seleccionado. La criptografía de clave pública requiere que cada usuario tenga dos claves: una clave pública, usada por todo el mundo para encriptar mensajes a enviar a ese usuario, y una clave privada, que necesita el usuario para desencriptar los mensajes.

8.3.1 El algoritmo RSA

RSA. Ha sobrevivido a todos los intentos para romperlo por más de un cuarto de siglo y se le considera muy robusto. Mucha de la seguridad práctica se basa en él. Su mayor desventaja es que requiere claves de por lo menos 1024 bits para una buena seguridad (en comparación con los 128 bits de los algoritmos de clave simétrica), por lo cual es muy lento. Su método se basa en ciertos principios de la teoría de los números.

1. Seleccionar dos números primos grandes, p y q (generalmente de 1024 bits).
2. Calcular $n = p \times q$ y $z = (p - 1) \times (q - 1)$.
3. Seleccionar un número primo con respecto a z, llamándolo d.
4. Encontrar e tal que $e \times d = 1 \ mod \ z$.

8.3.2 Otros algoritmos de clave pública

El primer algoritmo de clave pública fue el de la mochila (Merkle y Hellman, 1978). La idea aquí es que alguien es dueño de una gran cantidad de objetos, todos con pesos diferentes. El dueño cifra el mensaje seleccionando secretamente un subgrupo de los objetos y colocándolos en la mochila. El peso total de los objetos contenidos en la mochila se hace público, así como la lista de todos los objetos posibles. La lista de los objetos que se metieron en la mochila se mantiene en secreto. Con ciertas restricciones adicionales, el problema de determinar una lista posible de los objetos a partir del peso dado se consideró no computable, y formó la base del algoritmo de clave pública. Aunque se ha modificado nuevamente, el algoritmo de la mochila no se considera seguro y pocas veces se usa.