Cifrado y descifrado con MPI

Walter Danilo Saldaña Salguero 19897

Jose Abraham Gutierrez Corado 19111

Javier Alejandro Cotto Argueta 19324

Programacion de Microprocesadores

Universidad del Valle de Guatemala

Facultad de Ingeniería

Octubre 2020

Contenido

Antecedentes de DES y descripción de su funcionamiento

A medida que la criptografía comenzó a avanzar, las primeras empresas informáticas comenzaron a investigar sus usos. IBM fue una de esas empresas que comenzó a invertir fuertemente en criptografía, al darse cuenta de que a medida que las computadoras evolucionan, las técnicas criptográficas se convertirían en activos valiosos para las empresas de todo el mundo. A través de la década de 1960, trabajaron con Lloyds Bank en proporcionar las técnicas subyacentes para las máquinas de cajero automático (ATM) que se utilizarán en Londres y sus alrededores.

DES fue el resultado de un proyecto de investigación establecido por IBM a fines de la década de 1960 que resultó en un cifrado conocido como LUCIFER. A principios de la década de 1970, se decidió comercializar LUCIFER y se introdujeron varios cambios importantes. IBM no fue el único que participó en estos cambios, ya que buscaron asesoramiento técnico de la Agencia de Seguridad Nacional (NSA) (otros consultores externos participaron, pero es probable que la NSA fuera el principal contribuyente desde un punto de vista técnico). La versión alterada de LUCIFER se presentó como una propuesta para el nuevo estándar de cifrado nacional solicitado por la Oficina Nacional de Estándares (NBS).

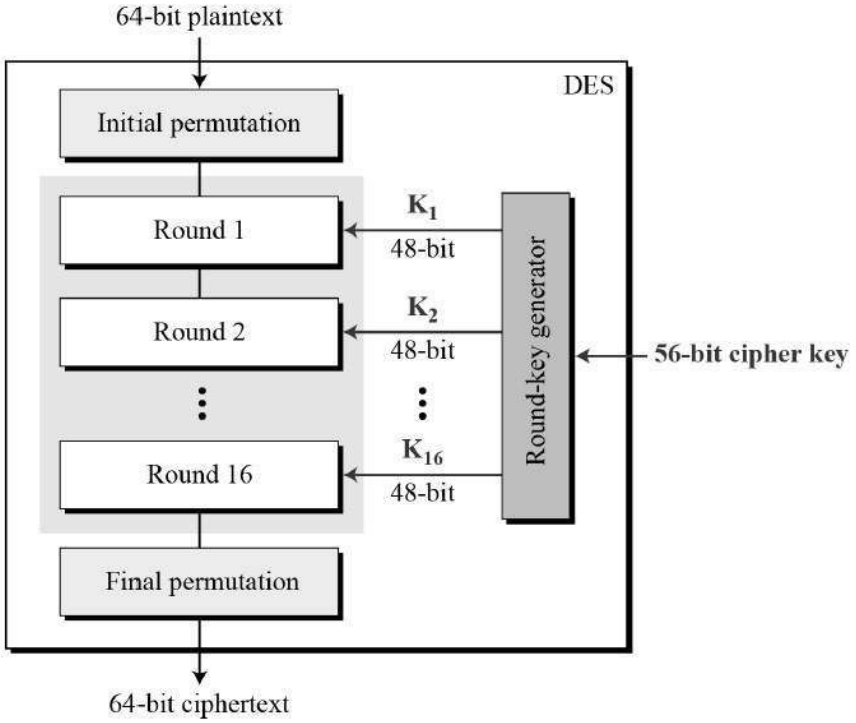
En 1968, la cabeza del Instituto de Ciencias de la Computación y Tecnología empezó a realizar un estudio acerca de la seguridad computacional que pudiera ser usado por el Gobierno. Sabiendo la experiencia y el conocimiento de la NSA en los campos de encriptación, la NBS contactó a la NSA por asistencia sobre la evaluación de la calidad de DES. NSA pidió una solicitud a través de un Registro Federal en mayo de 1973 para que los desarrolladores de DES pudieran *prestarles* su algoritmo. Tal solicitud fue rechazada y pidieron nuevamente una solicitud en agosto de 1974. NBS junto a la NSA, afirmó que el algoritmo hecho por IBM era el mejor algoritmo de encriptación que existía y que era el indicado para los dispositivos usados por el Gobierno (Bayh & Goldwater, 1978). Finalmente fue adoptado en 1977 como el Estándar de cifrado de datos - DES (FIPS PUB 46).

Inicialmente, DES se implementó con una clave de 128 bits. En ese momento, esto habría sido prácticamente, técnica y financieramente intensivo para romper. Ahora se sabe que gracias a la interferencia de la NSA, el tamaño de clave finalmente elegido fue una clave de

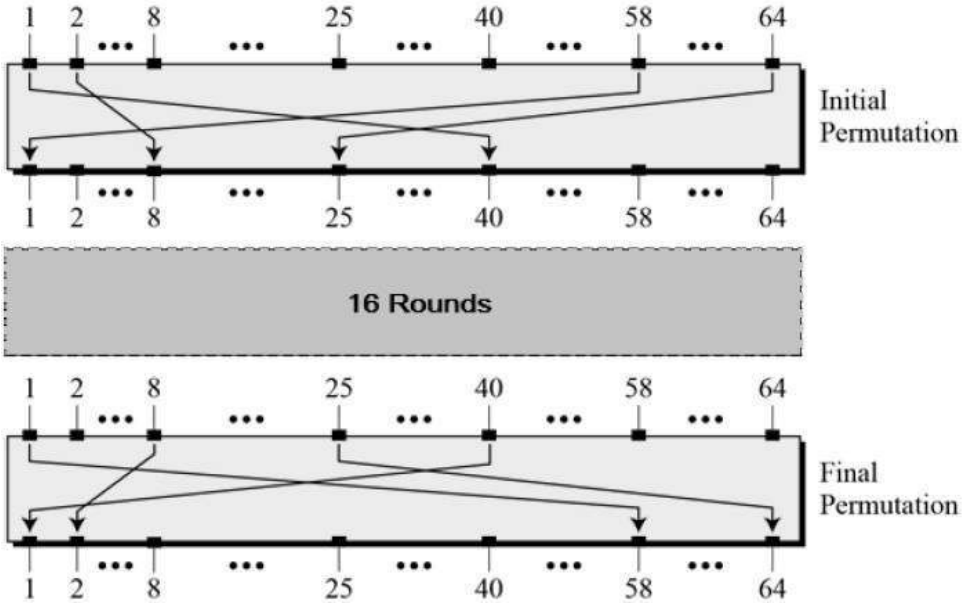
56 bits. Esto hizo que redujera intrínsecamente la seguridad de DES ya que cuanto mayor es la clave, más difícil es agrietarse (Chalmers, 2019).

En 2001 DES fue reemplazado por el Estándar de cifrado avanzado (AES, por las iniciales en inglés de Advanced Encryption Standard), el cual es un estándar de cifrado adoptado por el gobierno de los Estados Unidos. Comprende tres cifrados por bloques: AES-128, AES-192 y AES-256, adoptados entre una extensa colección originalmente publicada como Rijndael. Cada cifrado AES tiene un bloque de 128 bits de tamaño, con llaves de 128, 192 y 256 bits respectivamente (*Apéndice A. Estándares de cifrado*, n.d.).

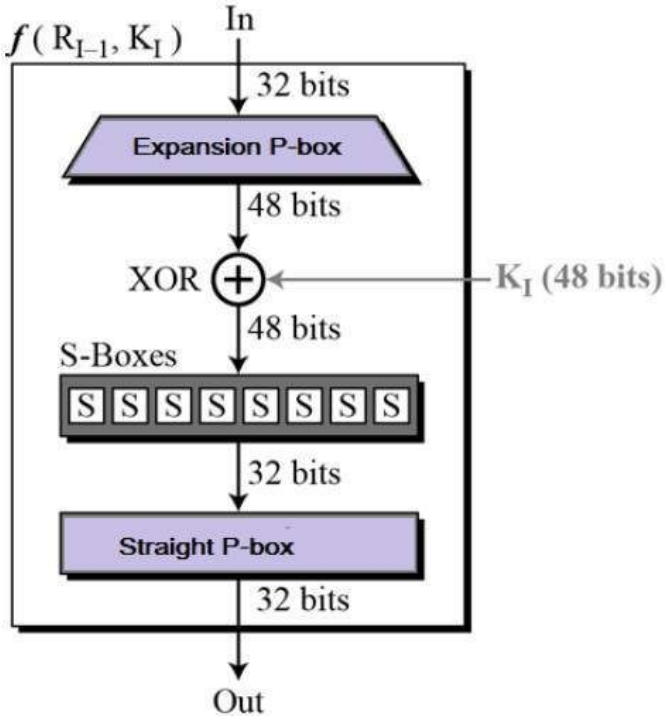
DES es un cifrado de bloques, lo que significa que opera en bloques de texto sin formato de un tamaño determinado (64 bits) y devuelve bloques de texto cifrado del mismo tamaño. Por lo tanto, DES da como resultado una permutación entre los 2 ^ 64 (léalo como: "2 elevado a la 64 potencia") posibles arreglos de 64 bits, cada uno de los cuales puede ser 0 o 1. Cada bloque de 64 bits se divide en dos bloques de 32 bits cada uno, medio bloque izquierdo L y medio derecho R. (Esta división solo se usa en ciertas operaciones).



Estructura general del algoritmo DES, donde en las 16 rondas usa una estructura Feistal con una llave de encriptación de 56 bits.



Permutaciones iniciales y finales del programa, los cuales hacen los cambios de bits



La función *f, el corazón del programa.* En esta parte, la función DES aplica una clave de 48 bits a los 32 bits más a la derecha para producir una salida de 32 bits. Imagenes provenientes: de https://www.tutorialspoint.com/cryptography/data_encryption_standard.htm

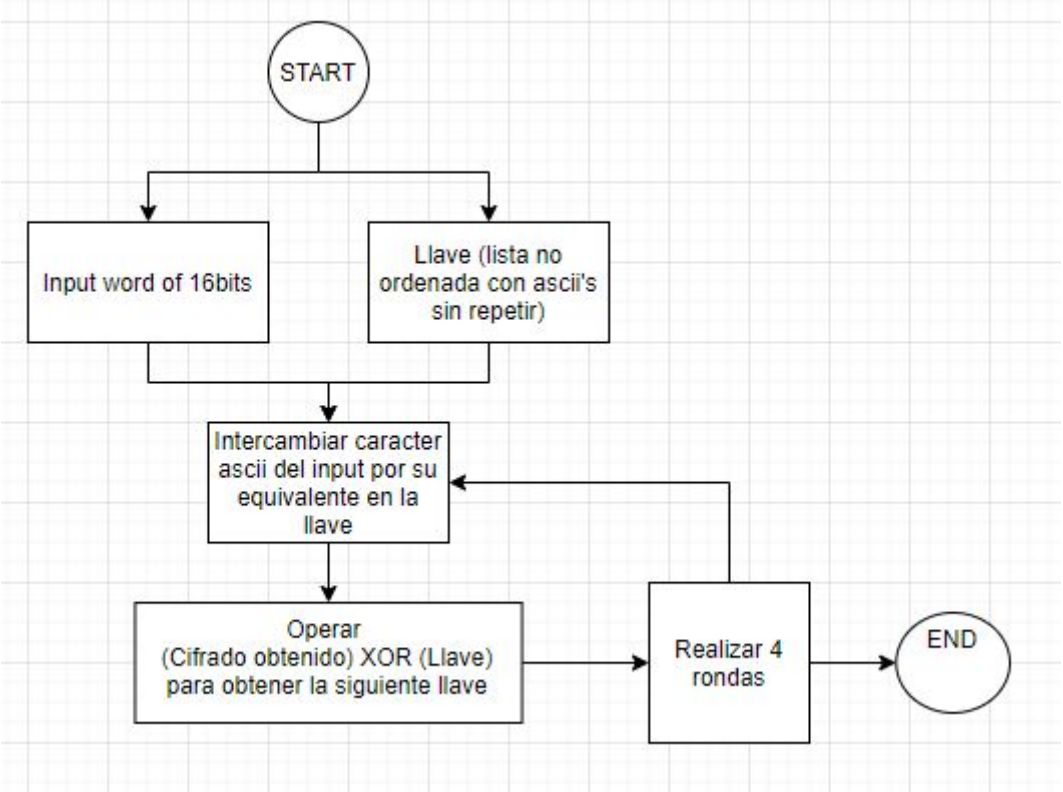
Comparación entre DES y el algoritmo propuesto

DES es un cifrado de bloques de una cadena sin formato de 64 bits, y devuelve una cadena cifrada del mismo tamaño. Utiliza el principio de general de Feistel para encriptar las 64 permutaciones anteriores y posteriores, para dar como resultado la cadena cifrada.

Nuestra propuesta consiste en implementar el cifrado de César modificado dentro del cifrado DES sustituyendo por completo el algoritmo de cifrados en base a Feistel. Cambiar la cantidad de los bloques, pues estos serán de límite de 16 bits, por lo que aceptará también de menor cantidad. Incluso, nuestro factor de conversión será la llave previamente configurada. De esta forma, integramos las permutaciones de DES en el cifrado, pues en lugar de permutar a partir de una matriz constante, permutamos intercambiando ASCII's a partir de una regla de equivalencia que será definida por la llave, y esta permutación es entonces un cifrado de César.

Bloques funcionales, con descripción de la función, diagramas explicativos y características del nuevo algoritmo

Diagrama de Flujo del algoritmo propuesto



Características:

- Cifrado de César, el cual reemplaza la estructura de Feistel.
- Cadena de 16 bits.
- Operación XOR de 4 rondas entre la cadena y la llave para la obtención de nueva llave.
- Utilización de hilos para las tareas de cifrar y descifrar.

Catálogo de subrutinas y funciones usadas en los programas, indicando parámetros de entrada, salidas y una breve descripción de la función

- Escribir(char): agrega un carácter a un documento binario
    - Entrada un carácter
    - No hay salidas
- GenerateKey: Crea una llave aleatoria de 16 bits
    - No hay salidas
- Encriptar: Distribuye las tareas de cifrado entre los hilos cada 16 caracteres
    - No hay entrada
    - No hay salidas
- Desencriptar: Distribuye las tareas de descifrado entre los hilos cada 16 caracteres
    - No hay entrada
    - No hay salida
- Cifrar: subrutina paralela para aplicar el algoritmo de encriptación
    - Rango del hilo
    - No hay salida
- Descifrar: subrutina paralela para aplicar el algoritmo de desencriptación
    - Rango del hilo
    - No hay salida

Nivel de paralelismo utilizado a nivel del procesamiento del texto (datos) y de las funciones de cifrado y descifrado (tareas)

Los datos utilizan el nivel de paralelismo a nivel de bits, pues las cadenas ingresadas pueden ser de menor cantidad de bits, por lo que las operaciones trabajan con la cantidad que se tenga dentro de la cadena de bits, y no con el tamaño que debe de ser. Los procesos se dividen por la cantidad de bits de las palabras, por lo que las cadenas menores a 16 bits, necesitaran la misma cantidad de procesos que una de 16 bits.

Debido a la utilización de hilos para la realización de tareas, se habla de un Paralelismo a nivel de Tareas, debido a que cada hilo se ejecuta según el proceso asignado, por lo tanto los procesos de cifrado y descifrado, se dividen en hilos para tener secuencia entre ellos. Debido a las modificaciones que se realizaron dentro del código y la cantidad de rondas que se hacen por cadena de 16 bits(4 rondas), se implementó un Paralelismo a nivel de instrucciones, ya que las “instrucciones” dentro de las tareas, se hacen simultáneamente.

Discusión

Dentro del cifrado de DES, se pueden presentar varios retos, como la multiplicación entre matrices que son las permutaciones, la asignación de valores de una matriz según la matriz de permutación, incluso la implementación del algoritmo de Feistel, que consiste en dividir la cadena de texto, en dos cadenas para hacerles un intercambio de bit según una tabla predeterminada.

En la propuesta ya mencionada, se redujo el tamaño de la cadena de texto pues se divide en bloques de 16 bits para una mejor administración del conjunto de bits, lo cual es una mejora al programa de DES; también se reduce seguramente el tiempo de ejecución considerablemente, debido al intercambio de Feistel con César, eh incluso las permutaciones son más cortas por lo mismo del tamaño de la cadena de bits.

La implementación de pthreads hace mucho más eficiente al momento de trabajar tareas secuenciales, porque se reduce el tiempo de ejecución del programa dado a que trabaja con diferentes funciones y datos a la vez por su nivel complejo de paralelismo, logrando un mejor el trabajo a la hora de cifrar descifrar y no depende del lenguaje para que funcione correctamente. El nivel de paralelismo a nivel de bits permite que los datos sean mayores a 16 bits, y que esto permita realizar operaciones de cifrado y descifrado con estas cadenas de “mayor cantidad de bit”. El programa esta realizado de forma que divida en bloques de 16 bits la cadena total.

Conclusión

- DES fue un momento fundamental para la criptografía. La introducción del sistema permitió seguir investigando académicamente, así como cuestionar el papel del gobierno en el cifrado.
- Pthreads son un componente fundamental al momento de encriptar y desencriptar dado a su nivel de paralelismo, el cual permite el manejo de distintos datos y funciones trabajando a la vez, disminuyendo el tiempo de respuesta.
- Con la unión de distintos métodos de cifrados, ya sean algoritmos existentes o no, se pueden crear nuevos algoritmos de encriptación que pueden llegar a ser más complejos y eficientes que cuando estaban estos separados, llevando a una desencriptación igual de compleja.

Bibliografía

- GeeksForGeeks. "Caesar Cipher in Cryptography - GeeksforGeeks." GeeksforGeeks, 2 June 2016, www.geeksforgeeks.org/caesar-cipher-in-cryptography/.
- Rodriguez, Daniel. "Comparación Entre Los Cifrados DES y AES." *Analytics Lane*, 8 July 2019, www.analyticslane.com/2019/07/08/comparacion-entre-los-cifrados-des-y-aes/. Accessed 1 Oct. 2020.
- Gonzales, Victor. *Patrones de Paralelismo: Una Aproximación Basada En Bibliotecas Génericas*. core.ac.uk/download/pdf/288499156.pdf. Accessed 2020 Oct. 2020.
- Chalmers, R. (2019, March 7). DES: The story of the Data Encryption Standard. Coin Rivet. https://coinrivet.com/des-the-story-of-the-data-encryption-standard/
- Bayh, B., & Goldwater, B. (1978). SENATE SELECT COMMITTEE ON INTELLIGENCE UNITED STATES SENATE (pp. 1–2). https://cryptome.org/2012/05/nsa-crypto-des.pdf
- Apéndice A. Estándares de cifrado. (n.d.). Docs.Fedoraproject.Org. Retrieved October 20, 2020, from https://docs.fedoraproject.org/es-ES/Fedora/13/html/Security_Guide/chap-Security_Guide-Encryption_Standards.html
- Data Encryption Standard - Tutorialspoint. (2019). Tutorialspoint.Com. https://www.tutorialspoint.com/cryptography/data_encryption_standard.htm