

3. Algoritmo DES (Data Encryption Standard)

3.1. Fundamentos

- Cifrado por bloques (**block cipher**)
 - Opera sobre un bloque de texto plano de n bits para producir un texto cifrado de n bits.
 - Típicamente, la longitud de un bloque es de 64 bits.
 - Pueden adaptarse para funcionar como cifradores de flujo (más generales y mayor aplicabilidad)
 - 2^n posibles bloques de texto plano diferentes
 - Para que sea reversible (descifrado), cada entrada debe producir un bloque de texto cifrado único
- Problema práctico: tamaño de bloque
 - tamaño de bloque pequeño ($n = 4$): el sistema es equivalente a un cifrado de sustitución clásico (vulnerable a un análisis estadístico del texto plano)
 - si n es grande y se permite cualquier sustitución arbitraria entre el texto plano y el texto cifrado, las características estadísticas del texto plano quedan enmascaradas (criptoanálisis estadístico sería imposible)
- PROBLEMA: Permitir cualquier sustitución arbitraria para un tamaño de bloque grande es inviable.
 - ¿cómo especificar esa sustitución?
 - el mapping subyacente (tabla especificando las transformaciones concretas) sería la propia clave
- Feistel propone una aproximación a este sistema ideal de cifrado cuando n es grande.
 - Simular sustitución completa con una secuencia de sustituciones parciales parametrizadas por una clave de k bits

3.2. El cifrado Feistel

- Esquema típico de cifrado por bloques en el que se basan la mayoría de los algoritmos de clave simétrica actuales
- Feistel propuso aproximar el cifrado de sustitución simple utilizando el concepto de **cifrado producto**
 - consiste en realizar dos o más operaciones básicas de cifrado en secuencia
 - el resultado final es criptográficamente más fuerte que los cifrados componentes.
- La propuesta de Feistel propuso alterna sustituciones y permutaciones
 - aplicación práctica de una propuesta de Claude Shannon(1945) para desarrollar un cifrado producto que alterna funciones de **confusión** y **difusión**.

(a) Difusión y confusión

- Bloques básicos de construcción de cualquier sistema criptográfico
- OBJETIVO: evitar el criptoanálisis basado en el análisis estadístico
 - *Cifrado ideal*: cualquier tipo de estadística sobre el texto cifrado es independiente de la clave utilizada y del texto plano involucrado.
- Shannon propone dos métodos para impedir el criptoanálisis estadístico

Difusion: Pretende disipar la estructura estadística del texto plano en el texto cifrado

- OBJETIVO: la relación estadística entre el cifrado y el texto plano lo más compleja posible
 - El cambio de un bit en el texto plano afecta al valor de muchos dígitos del texto cifrado,
 - Cada bit del texto cifrado se ve afectado por muchos dígitos del texto plano
- Se consigue aplicando repetidamente permutaciones antes de las funciones de sustitución empleadas
 - bits de diferentes posiciones en el texto plano contribuyen a la misma porción del cifrado.

Confusión: Pretende hacer la relación entre las estadísticas del texto cifrado y el valor de la clave lo más compleja posible

- OBJETIVO: evitar el descubrimiento de la clave
- En cifrado por bloques la transformación del bloque de texto plano a un bloque cifrado depende de la clave
- La manera en que se utiliza la clave para cifrar el texto debe ser tan compleja que sea difícil deducir la clave
- Se consigue mediante la utilización de un algoritmo de sustitución complejo.

(b) Estructura del cifrado Feistel

- ENTRADA: bloque de texto plano de longitud $2w$ bits y la clave K
- Bloque de texto plano se divide en dos mitades: L_0 y R_0
- Las mitades pasan a través de n redondeos (fases)
- Finalmente se combinan para producir el bloque cifrado
- REDONDEOS:
 - Cada redondeo i tienen como entradas:
 - L_{i-1} y R_{i-1} del redondeo previo
 - La subclave K_i derivada de la clave K (las K_i son diferentes de K y entre sí)
 - Estructura de los redondeos (redondeo i):
 - Se realiza una sustitución sobre la mitad izquierda de los datos (L_{i-1})
 - ◇ se aplica la **función de redondeo** F a la mitad derecha (R_{i-1})
 - ◇ se hace XOR de la salida de esa función con la mitad izquierda (L_{i-1})
 - Después de esta sustitución, se realiza una permutación que intercambia las dos mitades (L_i y R_i).
 - La función de redondeo F tiene la misma estructura para cada redondeo
 - está parametrizada por la correspondiente K_i
- El último redondeo se sigue de un intercambio que deshace la permutación del último redondeo.

Esquema:

La implementación exacta de una red de Feistel depende los parámetros:

- **Tamaño de bloque:** Cuanto más grande es el tamaño de bloque, mayor seguridad, pero menor velocidad de cifrado/descifrado
 - Habitualmente: 64 bits
- **Tamaño de clave:** Cuanto más grande sea el tamaño de clave, mayor seguridad, pero menor velocidad de cifrado/descifrado
 - Claves mayores de 64 bits (habitual 128)
- **Número de redondeos:** Cuanto mayor es el número de redondeos, mayor seguridad
 - Habitual: 16 redondeos
- **Algoritmo de generación de subclaves:** Cuanto mayor sea la complejidad del algoritmo, más difícil será el criptoanálisis.
- **Función de redondeo:** Cuanto mayor es la complejidad de la función de redondeo, mayor resistencia al criptoanálisis

Otras consideraciones:

- **Cifrado/Descifrado rápido por software** Puede ser relevante si no es posible hacerlo por hardware
- **Facilidad de análisis:** Interesa que el algoritmo sea fácil de analizar permite detectar y subsanar vulnerabilidades subsanarlas.

Descifrado del algoritmo Feistel

- Esencialmente será el mismo que el proceso de encriptación
 - la entrada es el bloque cifrado
 - las claves se deben utilizar en orden inverso en cada uno de los pasos de redondeo
- **VENTAJA:** sólo es necesario implementar un algoritmo

3.3. Estándar de encriptación de datos (DES)

- Adoptado como estándar en 1977 por el Departamento Nacional de Estándares (NBS) ahora Instituto Nacional de Estándares (NIST)
 - Cifrador de bloques basado en una red Feistel con añadidos
 - bloques de 64 bits
 - clave de 56 bits.
 - Para descifrar se utiliza el mismo algoritmo con la misma clave.
- **Antecedente:** LUCIFER (IBM), cifrado de Feistel con bloques de 64 bits y clave de 128 bits.
 - producto de cifrado comercial destinado a implementarse en un chip
 - con participación de IBM y de la NSA (Agencia de Seguridad Nacional)
- DES: versión refinada de LUCIFER más resistente al criptoanálisis pero con un tamaño de clave de 56 bits
 - Puntos críticos:
 - reduce en 72 bits la longitud de clave usada en LUCIFER
 - clave de 56 bits se ha mostrado vulnerable a ataques de fuerza bruta
 - se desconocen las motivaciones del diseño de uno de sus componentes (cajas S)

(a) Cifrado DES

- Estructura basada en una red Feistel (añade permutaciones inicial y final)
- Entradas: bloque de texto plano (64 bits) + clave (56 bits)
- 3 fases genéricas:
 - permutación inicial
 - 16 redondeos (con funciones de sustitución y permutación)
 - un intercambio de bits junto con la permutación inicial inversa
- **ESQUEMA GENERAL**

- **Uso de la clave:**
 - inicialmente se pasa a una función de permutación
 - en cada redondeo se genera una subclave K_i : combinando de un desplazamiento circular y una permutación
 - misma función de permutación en cada redondeo, pero el resultado es diferente debido al desplazamiento circular.
- **Permutación inicial:**
 - Cada permutación se representa por un vector con la posición se mueven los diferentes bits correspondientes
 - La permutación final sería la inversa.

- **Estructura de un redondeo simple** (redondeo i):
 - Mitades izquierda y derecha de los 64 bits se tratan separadamente: L y R
 - Fórmulas usadas:
 - $L_i = R_{i-1}$ (pasa R_{i-1} a la izquierda)
 - $R_i = L_{i-1} \text{ xor } F(R_{i-1}, K_i)$ (aplica F a R_{i-1} y lo combina con L_{i-1})
 - Función F :
 - La entrada R se expande a 48 bits utilizando una tabla
 - ◇ Esa tabla define una permutación más una expansión
 - Los 48 bits resultantes se combinan con los 58 bits de K_i mediante un XOR
 - El resultado se pasa a través de una función de sustitución (cajas S) que produce una salida de 32 bits que sufren una permutación.
 - **Cajas S** :
 - Cada caja S acepta 6 bits como entrada y genera 4 bits de salida.
 - Las cajas S son tablas de 16x4 elementos
 - ◇ cada una de las 4 filas representan una sustitución reversible de 16 bits.
 - ◇ el primer y el último bit de los 6, seleccionan una de las 4 sustituciones
 - ◇ los 4 bits centrales seleccionan una columna particular, con la que producirá la salida.
 - Finalmente, la salida es permutada para que el siguiente redondeo afecte al máximo número de bits posible.

- **Generación y uso de las subclaves (K_i):**
 - La clave de 56 bits se somete a una permutación
 - La salida (56 bits) se separa en dos mitades de 28 bits: C_0 y D_0
 - En cada redondeo, C_{i-1} y D_{i-1} se someten a un desplazamiento a la izquierda
 - Estos valores desplazados sirven de entrada para la siguiente fase de redondeo.
 - En cada fase de redondeo se unen estos valores y son nuevamente permutados, produciendo una salida de 48 bits (K_i) que sirve de entrada a la función F .

(b) Características

■ El efecto avalancha

- Propiedad deseable para este tipo de algoritmos (DES lo verifica)
- Un pequeño cambio en el texto plano produzca un gran cambio en el texto cifrado
- OBJETIVO: un cambio en un bit de texto plano o en un bit de la clave debería producir un cambio en muchos bits del texto cifrado
- Impide reducir el espacio de búsqueda de claves para un ataque de fuerza bruta

■ Criterios de diseño

- Principal problema del DES: no se conocen los criterios de diseño de todo el algoritmo
 - en particular de las cajas S
 - se desconoce si presentan alguna vulnerabilidad al criptoanálisis (puertas traseras)
 - la seguridad del algoritmo DES depende de la seguridad de las cajas S (única parte no lineal del algoritmo)

Mejoras al algoritmo DES

DEBILIDAD: Tamaño de clave reducido

APROXIMACIÓN: Cifrado múltiple on DES

Doble DES

Esquema más simple de encriptación múltiple

Tiene dos fases de encriptación y dos claves.

Dado un texto plano P y dos claves K_1 y K_2 , el texto cifrado C es:

$$C = E_{k_2}[E_{k_1}[P]]$$

El descifrado requiere que las claves se apliquen en orden inverso:

$$P = D_{k_1}[D_{k_2}[C]]$$

Este esquema presenta una clave de **112 bits**

Se incrementa enormemente la fortaleza criptográfica.

Triple DES con dos claves

Consiste en una triple encriptación con dos claves:

$$C = E_{k_1}[D_{k_2}[E_{k_1}[P]]]$$

El motivo para que el segundo paso sea una es que el algoritmo pueda descifrar datos encriptados por el algoritmo DES original

$$C = E_{k_1}[D_{k_1}[E_{k_1}[P]]]$$

Clave resultante: **112 bits**

Triple DES con tres claves

Se puede utilizar el triple DES con tres claves

$$C = E_{k_3}[D_{k_2}[E_{k_1}[P]]]$$

Se consigue mayor seguridad.

Tiene el inconveniente de que requiere una longitud de clave de 168 bits, que puede ser incómodo en algunos casos