Universidad Nacional Autónoma de México

Facultad de Ciencias

Criptografía y Seguridad

Tarea Moral 5

García Ponce José Camilo - 319210536

1. Código para DES
   En el archivo `des.py` se encuentra el código, la función `cifrar` recibe la llave y el mensaje en binario y le aplica el cifrado.

2. Prueba de Llaves
   Las pruebas de las llaves débiles se encuentran en el archivo `llaves_debiles`, podemos notar que las dos primeras no son buenas debido a que generar subllaves todas iguales (puros 1s y puros 0s), lo cual no es muy bueno ya que los procesos van a ser muy similares al ser las mismas subllaves, y las ultimas dos llaves tiene un gran cantidad de 0s (o 1s) seguidos luego de la mitad de subllave, lo cual tampoco es tan bueno ya que hace que las subllaves demasiado similares.
   Las pruebas de las llaves semi débiles se encuentran en el archivo `llaves_semidebiles`, podemos notar que varias de las llaves siguen generando subllaves con una gran cantidad de 0s (o 1s) seguidos y también algunas subllaves que son las mismas (por ejemplo con $fe01fe01fe01fe01$, tenemos que las subllaves 2 a 8 y las 16 son la misma) y en otras llaves como $1fe01fe01fe01fe0$ podemos observar que la estructura de las subllaves generadas son muy similares entre ellas, solo con pequeños cambios en algunos dígitos.
   Las pruebas de las llaves posiblemente débiles se encuentran en los archivos `llaves_posiblemente_debiles_1`, `llaves_posiblemente_debiles_2`, `llaves_posiblemente_debiles_3` y `llaves_posiblemente_debiles_4`, podemos notar que en `llaves_posiblemente_debiles_1` las llaves tienen algunos problemas debido a que las subllaves que generan tienen una gran cantidad de 0s seguidos (antes o después de la mitad, por lo general) y también que algunas subllaves iguales, en `llaves_posiblemente_debiles_2` las llaves tiene algunos problemas ya que las subllaves que generan llaves con una gran cantidad de 1s (o 0s) seguidos (antes o despues de la mitad), también tiene algunas subllaves iguales y las llaves $1fffee0010ef1fe01$ y $1fe0e0f10ef1f10e$ generan subllaves con formatos muy similares (o al menos eso es lo que se noto), en `llaves_posiblemente_debiles_3` las llaves tiene algunos problemas debido a que las subllaves que generan tienen subllaves que son iguales y otras tienen una gran cantidad de 0s (1s) seguidos (antes o después de la mitad) y por ultimo en en `llaves_posiblemente_debiles_4` las llaves tiene algunos problemas ya que las subllaves que generan tienen subllaves iguales, algunas subllaves con una gran cantidad de 1s (o 0s) consecutivos (luego o antes de la mitad), pero en la llave $1f01fee00e0ef1f1$ no se noto algo tan problemático.

3. Diseño de DES y estas llaves
   Estas llaves no son tan buenas por algunos problemas que causan, por ejemplo si a un mensaje le aplicamos una llave débil y luego al mensaje cifrado le aplicamos la misma llave débil obtenemos el mensaje original, esto debido a que las subllaves que se generan son iguales por lo tanto aplicar dos veces el algoritmo seria como descifrarlo (creo), lo que hace que estas llaves débiles no tengan mucha seguridad.
   Las llaves semidebiles tampoco son tan buenas ya que podemos encontrar pares de estas llaves que aplicadas dos veces (como lo visto arriba, primero una al mensaje y luego la otra llave al mensaje cifrado) obtenemos el mensaje original, esto pasa ya que as subllaves que generan ambas llaves son las mismas y pasa algo similar a lo de arriba, donde al aplicar el proceso de la misma forma con la misma llave se descifra.
   Y con las llaves posiblemente débiles solo se generan subllaves que son similares, por lo cual no van a ser tan seguras como las demás llaves, pero al menos son más seguras que las llaves débiles o semidebiles.
   En resumen estas llaves no son las mejores ya que generan subllaves con similares o con patrones similares, lo cual no es buena, ya que en el proceso de DES se hacen varias rondas con el mismo proceso solo usando diferentes subllaves, entonces si las subllaves son muy similares los resultados también lo van ser (creo), lo cual nos genera que la seguridad de este proceso no sea lo mejor. Y las mejores llaves serian las que no generan subllaves iguales o similares.

4. Fuentes

DATA ENCRYPTION ALGORITHM. (2024). *Umsl.edu*.
https://www.umsl.edu/~siegelj/information_theory/projects/des.netau.net/improvements%20in%20des.html
Weak key. (2024, January 15). *Wikipedia*.
https://en.wikipedia.org/wiki/Weak_key#Weak_keys_in_DES