Una evidencia robusta de que el algoritmo DES fortalecido con una permutación inicial variable es eficiente

Rolando Flores Carapia, Víctor Manuel Silva García, Cornelio Yáñez Márquez y Oscar Camacho Nieto

Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional, México

Centro de Investigación en Computación, Instituto Politécnico Nacional, México

rfcarapia@yahoo.com, vsilvag@ipn.mx, www.cidetec.ipn.mx, {cyanez, oscarc}@cic.ipn.mx, www.cornelio.org.mx, www.cic.ipn.mx

Resumen. Utilizando el Teorema JV se puede asociar un numero de tamaño 1089 a una permutación de 64 posiciones en 63 pasos, este resultado se aplicó para reforzar DES mediante una permutación inicial variable, con lo cual se incrementa la complejidad computacional del algoritmo, ya que cada permutación trabaja como una llave. Sin embargo, aun queda pendiente saber cómo actúan la permutación inicial variable y la llave de 56 bits, con relación a la complejidad computacional del algoritmo. En esta investigación, se dará luz al problema anterior, en el sentido de que se propondrá un esquema de Monte Carlo usando el modelo del "Birthday" para el cálculo de probabilidades.

Palabras clave: Teorema JV, modelo *birthday*, DES, modelo de decisión de Monte Carlo, permutación variable.

A Robust Evidence of the DES Algorithm Strengthened by a Variable Initial Permutation being Efficient

Abstract. By using the JV Theorem, it is possible to associate a number of size 10^{89} to a 64 position permutation in 63 steps. This result has been applied to strengthening the DES algorithm by using a variable initial permutation, increasing the computational complexity of the algorithm, given that each permutation works as a key. However, it still remains to be seen how the variable initial permutation and the 56 bit key work regarding the computational complexity of the algorithm. In the current work, the former question is addressed by proposing a Monte Carlo scheme using the Birthday model for the probability calculation.

Keywords. JV theorem, birthday model, DES, Monte Carlo decision model, variable permutation.

1 Introducción

Antes de entrar en materia se harán algunas precisiones de notación. Por ejemplo, la cadena de 64 bits 0,0,...0 será representada como **0**. También, se hace notar que en el ciclo de encriptación DES [5] no se aplica la permutación inversa al final de éste, sino que únicamente se aplicará una permutación inicial variable; lo anterior es denotado como $e_{P,K}(X)$, donde *P* es una permutación sobre 64 posiciones, *K* es una llave de 56 bits y *X* es una cadena de 64 bits o texto plano.

Ahora bien, supongamos que a la cadena **0**, tomada como texto plano, se le aplican las 2^{56} llaves posibles de DES y que los textos cifrados para cada una de estas llaves son diferentes. Si esto es así, entonces es posible demostrar que el algoritmo DES, como se definió arriba, encripta de manera diferente [13] cuando se tiene cualquiera de estos dos casos:

a) Dados $P_1 \neq P_2$ y la misma llave.

b) Dados $P_1 \neq P_2$ y $K^1 \neq K^2$.

El primer caso está demostrado en [13].

Para el inciso b, bastará con mostrar que existe al menos un texto plano, X_0, para el cual $e_{P_1,K^1}(X) \neq e_{P_2,K^2}(X)$, con lo cual se muestra que

los arreglos de salida asociados a P_1, K^1 y P_2, K^2 son diferentes. Se sigue que es suficiente con tomar $X_0 = \mathbf{0}$, ya que para el texto plano $X_0 = \mathbf{0}$ se tiene que $e_{P_1,K^1}(X_0) \neq e_{P_2,K^2}(X_0)$ por la suposición hecha anteriormente.

Es claro que un caso particular del inciso b se da cuando $K^1 \neq K^2$ y la misma permutación.

En este sentido, es muy importante el supuesto de que "a la cadena **0** tomada como texto plano se le aplican las 2^{56} llaves posibles de DES y que los textos cifrados para cada una de estas llaves es diferente". En este trabajo se dará una evidencia robusta de que esto es así, utilizando el método de Monte Carlo [7]; esto es, la posibilidad de que nos equivoquemos se puede reducir tanto como se quiera. Entonces, se puede considerar que el algoritmo es eficiente en el sentido de que cada bit que se agregue a la clave tiene como consecuencia que la complejidad computacional para resolverlo ante un ataque de fuerza bruta, se duplica. Es preciso notar que para el caso del DES que se propone, la clave se forma de 2 partes, a saber: la primera por la llave *K* de 56 bits con 2^{56} posibilidades, y la segunda por el número asociado a la permutación variable que puede ser hasta de 10^{89}. Estos números pueden ser representados por una cadena de 296 bits aproximadamente, por lo que hay 2^{296} números posibles. En este orden de ideas, se puede pensar que el algoritmo DES propuesto pudiera tener una complejidad computacional para resolverlo, con un ataque de fuerza bruta, de $2^{56} \times 2^{296} = 2^{352}$ aproximadamente. Esto es, se tendrían que probar para cada permutación todas las llaves *K* posibles, 2^{56}, y así encontrar el arreglo adecuado de salida [13].

2 Esquema de Monte Carlo

Se recurre al método de Monte Carlo [7] debido a que, por un lado, se requiere aplicar el algoritmo de encriptación 2^{56} veces a la cadena **0** (para cada una de las llaves de 56 bits); y por otro, se desean realizar las comparaciones de los elementos del conjunto de todas las cadenas de 64 bits, que son el resultado de las encriptaciones de **0** usando cada una de las llaves, y así averiguar si son diferentes. Esto implica que para esta tarea se requiere una enorme cantidad de trabajo de cómputo, por lo que se recurre al método de Monte Carlo. Denotemos al conjunto de todas las encriptaciones del texto plano **0**, utilizando las 2^{56} llaves posibles como *E*.

En este orden de ideas, para averiguar si es razonablemente cierto que los elementos del conjunto de encriptaciones *E* son diferentes, es necesario saber cuál es la probabilidad de que para una muestra *M* de tamaño *m* del conjunto *E*, al menos 2 de sus elementos son iguales. Para calcular la probabilidad anterior se puede hacer uso del modelo "*Birthday*" [12]. Nótese que cada uno de los elementos del conjunto *E* son cadenas de 64 bits.

Entonces, si se tiene una primera cadena como resultado de encriptar al texto plano **0** con una llave K^1 y, además, se considera la elección de K^2 de manera independiente a K^1; se sigue que la probabilidad de que la segunda cadena, resultado de aplicar K^2 a **0**, tenga un valor diferente a la primera, es que tome alguno de los 2^{56}–1 valores restantes, suponiendo que se tienen 2^{56} cadenas diferentes como posibles resultados. Como en este punto no se sabe cuántos elementos diferentes tiene el conjunto *E*, entonces la probabilidad de que la segunda cadena tome un valor diferente a la primera es a lo sumo $\frac{2^{56}-1}{2^{56}}$ [12]. Nótese que *E* puede tener como máximo 2^{56} valores diferentes.

No es complicado observar que para una tercera llave K^3, elegida de manera independiente a las otras 2, la probabilidad de que tome un valor restante diferente es como máximo:

$$\frac{2^{56}-1}{2^{56}} \times \frac{2^{56}-2}{2^{56}} \quad (1)$$

Siguiendo este procedimiento, la probabilidad de que *m* cadenas tomen valores diferentes es a lo sumo:

$$\frac{2^{56}-1}{2^{56}} \times \frac{2^{56}-2}{2^{56}} \times \ldots \times \frac{2^{56}+1-m}{2^{56}} \quad (2)$$

Por lo anterior, la probabilidad de que al menos 2 cadenas sean iguales, denotada como P_m, tiene como valor mínimo a:

$$P_m = 1 - \left[\frac{2^{56}-1}{2^{56}} \times \frac{2^{56}-2}{2^{56}} \times \ldots \times \frac{2^{56}+1-m}{2^{56}} \right] \qquad (3)$$

En este sentido, si se calcula a m tal que $\frac{2^{56}-1}{2^{56}} \times \frac{2^{56}-2}{2^{56}} \times \ldots \times \frac{2^{56}+1-m}{2^{56}} \cong \frac{1}{2}$, entonces se dirá que para este m particular l a probabilidad P_m es al menos de $\frac{1}{2}$. El cálculo de m da como resultado $m \cong 320000000$.

Ahora bien, en este punto se define el siguiente problema de decisión [4]: tomando como llaves cada una de las cadenas del conjunto $\mathbf{K} = \left\{ K \middle| K \in \{0,1\}^{56} \right\}$, ¿existen al menos 2 cadenas iguales cuando se encripta al texto plano **0**?

Este problema de decisión puede tener únicamente 2 respuestas, a saber: **sí** o **no**. Sin embargo, si se toma una muestra de tamaño m del conjunto de encriptaciones del texto plano **0**, se sigue que para este caso particular de decisión se tienen los siguientes escenarios:

a) Si la respuesta es **sí**, ésta es correcta.
b) Si la respuesta es **no**, se corre el riesgo de que sea falsa con una probabilidad de a lo sumo $\frac{1}{2}$.

Entonces, si se repite este experimento, digamos unas 40 veces, donde cada uno de estos experimentos se realizan de manera independiente, y para cada uno de estos casos la respuesta es **no**, se sigue que la probabilidad de que esta respuesta sea falsa es a lo sumo $\frac{1}{2^{40}}$. Si esto último es así, se puede considerar "razonable", **sobre todo para situaciones prácticas**, la propuesta de que el resultado de aplicar todas las llaves del conjunto **K** al texto plano a **0**, son diferentes.

En esta parte se menciona someramente la forma en que se tomarán las muestras pseudoaleatorias. La parte fraccionaria de los números irracionales trascendentes [14], como π, son de gran utilidad ya que se considera cumplen con la propiedad de "normalidad" [7], aunque hasta ahora no se ha dado una prueba matemática de que esto es así. Por otro lado, aún queda pendiente mostrar cómo se obtendrán 39 irracionales trascendentes más, los cuales se usarán para obtener cada una de las muestras.

Como se sabe, un número irracional se dice que es trascendente si no es solución de una expresión de la forma $a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0 = 0$, donde los a_i son enteros para $i = 0 \ldots n$ [14]. En este sentido, no es complicado demostrar que cualquier número de la forma $b\pi$, donde b es un entero diferente a cero, es a su vez un irracional trascendente; de hecho, la prueba puede realizarse por reducción al absurdo. Como una situación particular podemos escoger a b primo impar mayor que 1, esto es, $b \geq 3$. Se sigue que se pueden elegir pseudoaleatoriamente 39 números primos y multiplicarlos por π para obtener los irracionales trascendentes que nos hacen falta. Por último, se aclara que la fracción decimal de π se obtuvo de [1]; y además, en la referencia [2] se muestra una manera de cómo obtener los decimales que se deseen de la fracción de π.

3 Resultados de la aplicación del esquema de Monte Carlo

Inicialmente se deben obtener 320000000 cadenas de 56 bits cada una, las cuales deben ser todas diferentes, sin embargo, en la práctica esto no sucede así. Cuando se multiplica a π por un primo, que puede ser 3 o hasta de 10 dígitos y se toma la parte fraccionaria del producto para seleccionar todas las cadenas solicitadas, pueden existir algunas repeticiones, por lo que habrá que tomar algunos dígitos más con el objeto de que todas ellas sean diferentes. La cantidad de dígitos que se tomaron en este trabajo de la fracción decimal del producto $p \times \pi$, fue de un poco mas que 6400000000.

Cada una de las 40 aplicaciones de 320000000 de cadenas de 56 bits encriptando al texto plano **0**, arrojó para cada situación particular que todas las cadenas encriptadas fueron

diferentes. Con lo cual se puede decir que la **negación** de "Al menos hay 2 textos cifrados iguales cuando se aplican las 2^{56} llaves posibles de 56 bits al texto plano **0**", puede ser falsa con una probabilidad de error, de a lo sumo $\frac{1}{2^{40}}$; lo cual nos conduce a pensar que es altamente probable que todas las cadenas cifradas del texto plano **0** son diferentes.

A continuación se mostrará un resultado particular del producto entre el primo 2135944739 y π. Además, se toman algunas cadenas de 56 bits del producto y se encripta a la cadena **0**.

Resultados del producto.

A partir de la posición cero: 83 21 9A 96 46 19 FE 09 D1 DA 83 5E A2 16 C2 92 43 A0 3F 33 43 53 0A DC FC D0 40 A3 CD 49 B3 42 6F 3A 6B 28 9E E9 96 5C 00 4E A1 97 AC 8D AA D0 70 FC.

A partir de la posición 1000000: 2B 30 F7 66 A8 32 BA 04 4D 09 30 34 BD BC BA 7D D8 A1 E0 22 F7 AD 73 81 F0 D8 FE 46 87 F8 06 C6 06 DD 3E 4B DA 6F 93 B9 E8 BA 5F E7 E6 40 82 F4 A3 A2.

A partir de posicion 1000000000: D3 68 9C 47 82 05 8E D3 53 74 18 9E 9C CF 74 A5 BE 72 DA 18 6B C2 70 35 C2 5A 65 83 90 8C AB 92 6B EF 41 03 02 1B 86 40 D6 88 F7 90 E3 69 B7 F5 E3 05.

Llave o entrada y cadena cifrada o salida:

Entrada 0: 83 21 9A 96 46 19 FE y salida 83 4C 79 23 5E EC 10 B3. Entrada 143: DA 46 15 B5 88 52 FE y salida AB BB B8 C4 5B D7 6C AD. Por ultimo, entrada 152858: BA 04 4D 09 30 34 BD y salida 0A F2 9A 1E 89 A9 26 0D.

Debido a los comentarios de uno de los árbitros, los autores hacemos el siguiente análisis: cuando se menciona que si se logra conocer la llave K, "entonces el oponente procede a cifrar los siguientes mensajes M 1= {0…001}, M 2= {0…010}, …, M 63 = {1…000}, produciendo las consiguientes cifras, C_1, C_2, C_63". El razonamiento anterior es incorrecto, al menos como está escrito, debido que aún con el conocimiento de la llave K no se pueden cifrar a M 1, M 63, ya que no se conoce la permutación inicial, recuérdese que es variable, por lo que la deducción que sigue es falsa. Por otro lado, cuando los autores utilizaron como texto claro a la cadena **0**, lo hicieron con el objeto de realizar una prueba robusta, no porque sea un "esquema reducido de textos claros", de hecho, en la práctica no es recomendable usar la cadena **0**. En conclusión, el ataque planteado por el árbitro no se puede llevar a cabo, al menos como está escrito. También se aclara que, en este trabajo el objetivo más importante es mostrar cuál es la relación entre las permutaciones y la llave.

Por último, cuando se expresa que el criptosistema de cifrado propuesto está fortalecido con relación a DES, los autores se refieren a un ataque de fuerza bruta, ya que es el ataque que puede ser aplicado a cualquier criptosistema, nunca podemos hablar de un ataque que puede ser ideado en el futuro debido a que éste se desconoce.

4 Conclusiones

En este trabajo se da una evidencia robusta de que el algoritmo fortalecido de DES, con una permutación variable al inicio del proceso de encriptación, da como resultado que este procedimiento es eficiente. Esto es, cada bit que se le agrega a la clave del algoritmo incrementa en un factor de 2 la complejidad computacional para resolverlo, cuando se realiza un ataque de fuerza bruta. Para este caso particular de DES la clave consta de 2 partes, a saber: una de 296 bits aproximadamente y otra de 56 bits. Lo anterior significa que la complejidad computacional para resolverlo es de $2^{296} \times 2^{56} = 2^{352}$ aproximadamente.

En este orden de ideas, es probable que se puede llegar al mismo resultado cuando se trata de Triple-DES [6] con una permutación variable al inicio, y también, de manera similar cuando se trate del algoritmo *Advanced Encryption Standard*-AES [9], con una permutación variable al inicio. Además, con una permutación inicial variable, para el caso de DES, se pueden evitar ataques como: Diferencial [3] y Lineal [10], al menos como se llevan a cabo en este momento.

Por último, cuando se utiliza un sistema de encriptación simétrico como: DES, Triple-DES o AES, en un esquema de comunicación con llave pública y firma electrónica [8], el número asociado a la permutación variable puede ser enviado mediante cualquiera de los

criptosistemas asimétricos: RSA [11], o ElGamal [15].

Agradecimientos

Los autores agradecen el apoyo de las siguientes instituciones para la realización de esta obra: Instituto Politécnico Nacional (Secretaria Académica, COFAA, SIP, CIDETEC, CIC), CONACyT y Sistema Nacional de Investigadores (SNI).

Referencias

1. **Bailey, D.H. (2011).** *A Compendium of BBP-Type Formulas for Mathematical Constants*. Retrieved from http://crd.lbl.gov/~dhbailey/dhbpapers/bbp-formulas.pdf.
2. **Bailey, D.H., Plouffe, S.M., Borwein, P.B., & Borwein, J.M. (1997).** The Quest for Pi. *The Mathematical Intelligencer*, 19(1), 50–56.
3. **Biham, E. & Shamir, A. (1993).** Differential cryptanalysis of the full 16-round DES. *Advances in Cryptology-CRYPTO' 92. Lecture Notes in Computer Science*, 740, 487–496.
4. **Daemen, J. & Rijmen, V. (2001).** *Advances Encryption Standard (AES),* FIPS 197.
5. *Data Encryption Standard (DES),* FIPS PUB 46-2, 1993.
6. *Data Encryption Standard (DES)*, FIPS PUB 46-3, 1999.
7. **Elgamal, T. (1985).** A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4), 469–472.
8. **Gentle, J.E. (2003).** *Random Number Generation and Monte Carlo Methods (2nd ed*.). New York: Springer.
9. **Gómez, A. (2006).** *Enciclopedia de la Seguridad Informatica*. Alfaomega: Mexico.
10. **Matsui, M. (1994).** Linear Cryptanalysis Method for DES cipher. *Advances in Cryptology-EUROCRYPT'93. Lecture Notes in Computer Science*, 765, 386–397.
11. **Rivest, R.L., Shamir, A., & Adleman, L. (1978).** A method for obtaining digital signatures and Public-key cryptosystems. *Communications of the ACM*, 21(2), 120–126.
12. **Rosen, K.H. (2003).** *Discrete Mathematics and its Applications (5th ed.)*. Boston: McGraw Hill.
13. **Silva, V.M., Flores, G.R., López, I., & Rentería, C. (2010).** Algorithm for Strengthening Some Cryptographic Systems. *Journal of Applied Mathematical Sciences*, 4(20), 967–976.
14. **Spivak, M. (1997).** Calculus: Cálculo Infinitesimal (2nd ed.). Barcelona: Reverté.
15. **Stinson, D.R. (2002).** *Cryptography: Theory and practice (2nd ed.)*. Boca Raton: Chapman & Hall/ CRC.

Rolando Flores Carapia Ingeniero en Comunicaciones y Electrónica por la Escuela Superior de Ingeniería Mecánica y Eléctrica (ESIME) del Instituto Politécnico Nacional en la Ciudad de México, en el año de 1996. Maestro en Ciencias de la Computación Por el Centro de Investigación en Computación (CIC) del Instituto Politécnico Nacional en 2006. Actualmente estudiante de Doctorado en Ciencias de la Computación en el mismo centro. Profesor en el Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC) del Instituto Politécnico Nacional, desde 2006.

Víctor Manuel Silva García Licenciado en Física y Matemáticas (1972) por la Escuela Superior de Física y Matemáticas del IPN. Grado de Maestro en Ciencias en Estadística Aplicada (1980) por el Colegio de Posgraduados de Chapingo, México, y Grado de Doctor en Ciencias de la Computación (2007) por el CIC-IPN. Actualmente es Director del Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional. Miembro del Sistema Nacional de Investigadores. Áreas de Interés: Criptografía y Criptoanálisis, Desarrollo Tecnológico en software, hardware y sistemas híbridos, Matemáticas Aplicadas, Memorias Asociativas, Redes Neuronales y Teoría de Códigos.

Cornelio Yáñez Márquez Licenciado en Física y Matemáticas (1989) por la ESFM-IPN. Grados de maestro en ciencias (1995) en ingeniería de cómputo y de doctor en ciencias de la computación (2002), obtenidos en el CIC-IPN. Profesor investigador titular C del CIC-IPN. Presea Lázaro Cárdenas 2002, recibida de manos del C. Presidente de la República. Miembro del Sistema Nacional de Investigadores. Areas de Interés: Memorias Asociativas, Redes Neuronales, Morfología Matemática e Ingeniería de Software.

Oscar Camacho Nieto Ingeniero en Comunicaciones y Electrónica (1989) por la ESIME ZAC–IPN. Grados de maestro en ciencias (1995) en ingeniería de cómputo y de doctor en ciencias de la computación (2003), obtenidos en el CIC-IPN. Suficiencia de investigador en el doctorado en arquitectura de computadores (1999) por la Universidad Politécnica de Cataluña en Barcelona, España. Profesor investigador titular C del CIC-IPN. Miembro del Sistema Nacional de Investigadores. Áreas de Interés: Arquitectura de Computadoras, Memorias Asociativas, Microprocesadores, Sistemas Digitales y Redes Neuronales.

Artículo recibido el 15/12/2010; aceptado el 24/10/2011.