Seguridad de DES

Hay en esencia dos tipos de ataques:

1) Búsqueda exhaustiva de llaves (Fuerza bruta)
2) Ataques analíticos

Como bien lo describe el nombre, para (1) la idea es aplicar fuerza bruta para encontrar la llave.
Por otro lado, para (2) es necesario entender el funcionamiento del criptosistema para poder aplicar ataques especialmente diseñados.

Tras la propuesta de DES, inmediatamente saltaron a la vista dos problemas:

1) El espacio de llaves es muy pequeño, entonces es vulnerable a fuerza bruta
2) Las S-cajas se mantuvieron en secreto, llevando a la idea de que pudieron haber ataques/propiedades matemáticas efectivas para romper DES que solo conocen

los diseñadores de DES

Hasta la fecha del libro, es más fácil y eficiente aplicar fuerza bruta y romper DES en cuestión de horas/días

Búsqueda exhaustiva de llaves

Algo a destacar es que IBM propuso una longitud de llaves de 128 bits, lo cual fue sospechosamente reducido a 56bits. La motivación oficial es sospechosa.

Ahora, ¿a qué nos referimos con búsqueda exhaustiva de llaves?

Definición (3.5.1) Búsqueda exhaustiva de llaves:

Input: Una pareja (x,y) de texto plano x y texto cifrado y.

Output: Llave k tq $DES_k(x) = y$

Ataque: Probar las 2^{56} posibles llaves.

Observación: ¡Se puede encontrar una llave incorrecta!

(Probabilidad de $1/2^{16}$)

Una computadora regular (2010) no es realmente apta para la búsqueda exhaustiva, pero se pueden hacer máquinas especializadas (a un precio).

1977 Whitfield Diffie. Martin Hellman

Propuesta no construida

20 millones de 1977-USD ↝ ~104.2 millones de USD

Tiempo no especificado

1993 Michael Wiener

Propuesta no construida

1 millón de 1993-USD ↝ ~2.16 millones de USD

Tiempo de ~1.5 días.

1998 Electronic Frontier Foundation

Máquina Deep Crack (~1800 c.i. de 24 test c/u)

< 250,000 1998-USD ↝ ~485,000 USD

Tiempo récord de 56h pero promedio de 15 días

2006 Universidades de Bochum y Kiel (Alemania)

COPACOBANA (Cost-Optimized PArallel COde-Breaker)

~ 10.000 2006USD → ~ 15,700 USD

Tiempo promedio 7 días

⇒ 56 bits es una longitud de llave muy chica.

⇒ DES nada más es "efectivo" para usos cortos (horas) o cuando los datos encriptados son de bajo valor.

Ataques analíticos

En 1990 Eli Biham y Adi Shamir descubren el criptoanálisis diferencial (DC)

En 1993 Mitsuru Matsui publica el ataque llamado criptoanálisis lineal (LC)

En ambos casos, estos son ataques en general aplicables (y eficientes) para cifrados de bloque.

Sin embargo, estos no son eficientes para DES. En el caso de DC, al parecer por diseño.

DC → 2^{47} parejas conocidas escogidas o 2^{55} aleatorias

LC → 2^{43} parejas escogidas

¿Por qué no es práctico?

1) Números muy grandes
2) Toma mucho tiempo recolectar estas parejas
3) Tras todo esto se obtiene una sola llave.

Resumen:

1990	Biham-Shamir	DC	$\sim 2^{47}$ escogidas
1993	Matsui	LC	$\sim 2^{43}$ escogidas

Implementaciones en Software y Hardware

Software: Implementaciones en computadoras de escritorio, celulares, etc.

Hardware: Implementaciones en circuitos integrados e.g. ASIC o FPGA.

Software

Una implementación directa es poco práctica por las permutaciones involucradas, las cuales son lentas en software

Además, las S-cajas pequeñas si bien son prácticas en hard

ware, sólo son moderadamente eficientes CPU's (2010).
Para "darle la vuelta" a esto se suelen usar tablas de las operaciones DES
Una implementación optimizada requiere aprox. 240 ciclos por bloque en un CPU de 32-bits. Tomando un CPU de 2GHz ~> 533 Mbits/s
Sin optimizar nos deja alrededor de 100Mbits/s.
Eli Biham desarrolló bit slicing, lo cual puede aplicarse para mejorar considerablemente el desempeño. Sin embargo, tiene desventajas dependiendo de la aplicación.

Hardware

Uno de los criterios de DES era la eficiencia de implementación por hardware. Las permutaciones implementadas son fáciles en hardware a través de cableado sin necesidad de funciones lógicas. Luego, las S-cajas pequeñas son fáciles de implementar con lógica Booleana. Una implementación eficiente por área se puede hacer con menos de 30000 puertas. Caben en chips de identifica-

cón de frecuencias de radio).

En ASIC y FPGA modernos (2010) se puede realizar con eficiencia de 100 Gbits/s