Sławomir Trznadel
Anna Zugaj
Karol Górski
Andrzej Paszkiewicz
Instytut Telekomunikacji
Politechnika Warszawska

Zbigniew Kotulski
Janusz Szczepański
Instytut Podstawowych Problemów Techniki
Polska Akademia Nauk

PRZEGLĄD STANU WIEDZY NA TEMAT KRYPTOANALIZY LINIOWEJ ZE SZCZEGÓLNYM UWZGLĘDNIENIEM ALGORYTMU DES.

Przedmiotem artykułu są zagadnienia kryptoanalizy liniowej szyfrów blokowych ze szczególnym uwzględnieniem algorytmu DES. Obecnie jest to najbardziej efektywna metoda ataku na algorytm DES. Na przykładzie tego algorytmu przedstawione zostały zasady aproksymacji pojedynczej rundy i konstruowania wyrażenia liniowego dla całego szyfru. Podano także sposób wyznaczenia najbardziej efektywnego wyrażenia liniowego oraz szacowania liczby tekstów potrzebnych do przeprowadzenia skutecznej kryptoanalizy z tekstem jawnym. Przedstawiono szkic implementacji takiego ataku na szyfr DES. Ponadto omówiono możliwości rozszerzania kryptoanalizy liniowej dla wielu wyrażeń liniowych. Ostatnia część przedstawia wykorzystanie kryptoanalizy liniowej do weryfikacji bezpieczeństwa innych szyfrów blokowych (IDEA, RC5, LOKI).

1. WSTĘP

Symetryczne szyfry blokowe to jedno z podstawowych narzędzi współczesnej kryptografii. Używane są do zapewnienia poufności, integralności i uwierzytelnienia podmiotów lub danych. Ich szerokie zastosowanie wymaga dużego zaufania do ich bezpieczeństwa. Ze względu na to, że nie są znane konstrukcje szyfrów blokowych oferujące bezwarunkowe bezpieczeństwo ani praktyczne konstrukcje zapewniające udowadnialne bezpieczeństwo obliczeniowe, w praktyce konieczne jest heurystyczne podejście do oceny takich szyfrów. Szyfr uznawany jest więc za bezpieczny, jeżeli jego przestrzeń kluczy jest dostatecznie duża, by zapobiec atakowi metodą przeglądu wszystkich kluczy i jeżeli nie są znane inne, skuteczne ataki przeciwko niemu. Skuteczność ataku mierzy się poprzez porównanie jego złożoności (czasowej i pamięciowej) do złożoności ataku metodą przeglądu wszystkich kluczy. Obecnie zakłada się, że bezpieczeństwo szyfru nie zależy od klasy ataku (tylko z szyfrogramem, ze znanym tekstem jawnym, z wybranym tekstem jawnym, z wybranym szyfrogramem oraz ich adaptacyjne warianty) - pojawienie się skutecznego ataku, w którejkolwiek z tych klas (nawet jeżeli jest on określony tylko dla podzbioru kluczy) wskazuje na tzw. certyfikacyjną słabość szyfru. Oczywiście to podejście nie pozwala prognozować czy i jak długo szyfr pozostanie bezpieczny. W ocenie bezpieczeństwa brane są pod uwagę tylko te ataki, które są znane w chwili dokonywania oceny. Jednym z najważniejszych ataków rozważanych w ocenie bezpieczeństwa współczesnych szyfrów jest kryptoanaliza liniowa.

Kryptoanaliza liniowa została zastosowana po raz pierwszy przez Matsui w 1992 roku do analizy szyfru FEAL-8. Jej popularność wzrosła, gdy zastosowano ją do łamania najbardziej znanego z szyfrów iteracyjnych - DES, wobec którego okazała się jednym z najbardziej skutecznych ataków. Prezentujemy podstawy kryptoanalizy liniowej na przykładzie ataku na ten właśnie algorytm. Od chwili opublikowania pierwszej pracy Matsui zaproponowano szereg modyfikacji zwiększających skuteczność kryptoanalizy liniowej. Prace Matsui zapoczątkowały serię badań nad kryptoanalizą liniową innych szyfrów iteracyjnych.

2. KRYPTOANALIZA LINIOWA ALGORYTMU DES.

Kryptoanaliza liniowa DES'a dotyczy **ataku ze znanym tekstem jawnym** (known-plaintext attack), gdzie kryptoanalityk dysponuje parami: tekst jawny, szyfrogram. Zadanie polega na wywnioskowaniu klucza (bądź kluczy) zastosowanego do szyfrowania lub wydedukowania algorytmu do deszyfrowania kolejnych wiadomości zaszyfrowanych z tym samym kluczem.

Dane: $\mathbf{P_1, C_1 = E_k(P_1); P_2, C_2 = E_k(P_2); ... ; P_i, C_i = E_k(P_i)}$
Wnioskowanie: klucz **k** lub algorytm wnioskowania o $\mathbf{P_{i+1}}$ na podstawie $\mathbf{C_{i+1} = E_k(P_{i+1})}$.

Okazuje się, że można złamać 8-rundowy szyfr DES z $\mathbf{2^{18}}$ znanymi tekstami jawnymi oraz 16-rundowy DES z $\mathbf{2^{47}}$ znanymi tekstami jawnymi.

Prócz tego metoda ta może być zaadoptowana do **ataku z szyfrogramem**. Na przykład jeżeli tekst jawny zawiera zdania w języku angielskim reprezentowane przez kod ASCII, 8-rundowy DES jest możliwy do złamania z 2^{29} szyfrogramami.

Założeniem metody ataku ze znanym tekstem jawnym jest otrzymanie liniowej aproksymacji zadanego algorytmu szyfrowania. Wstępem jest skonstruowanie równania opisującego zależności pomiędzy wejściowymi i wyjściowymi bitami każdego z bloków **S**. Następnie należy rozszerzyć to odwzorowanie na cały algorytm co pozwala osiągnąć liniową aproksymację algorytmu bez wartości pośrednich.

Notacja:

P:	64-bitowy tekst jawny,
C:	odpowiadający 64-bitowy szyfrogram,
P_H:	lewa 32-bitowa część P,
P_L:	prawa 32-bitowa część P,
C_H:	lewa 32-bitowa część C,
C_L:	prawa 32-bitowa część C,
X_i:	32-bitowa pośrednia wartość w i-tej rundzie,
K_i:	48-bitowa wartość klucza w i-tej rundzie,
$F_i(X_i, K_i)$:	funkcja F w i-tej rundzie,
A[i]:	i-ty bit A,
A[i, j, ..., k]:	A[i] A[j] ... A[k].

Zasady analizy liniowej.

Celem kryptoanalizy liniowej jest znalezienie wyrażenia liniowego, które jest najbardziej efektywne dla danego algorytmu szyfrowania:

(1) $\mathbf{P[i_1, i_2, ..., i_a] \oplus C[j_1, j_2, ..., j_b] = K[k_1, k_2, ..., k_c],}$

gdzie $i_1, i_2, ..., i_a, j_1, j_2, ..., j_b, k_1, k_2, ..., k_c$ oznaczają stałe ulokowanie bitu, a równanie **(1)** daje prawdopodobieństwo $p \neq 1/2$ dla losowo wybranych odpowiadających sobie par P i C. Wielkość p - 1/2 jest miarą efektywności równania **(1)**.

Możliwe jest określenie jednego bitu $\mathbf{K[k_1, k_2, ... k_c]}$ poprzez zastosowanie następującego algorytmu bazującego na maksymalizacji prawdopodobieństwa:

Algorytm 1

Krok 1:
Niech T będzie liczbą jawnych tekstów, dla których lewa strona równania **(1)** równa jest 1.
Krok 2:
Jeżeli T > N/2 (N ozn. liczbę tekstów jawnych),
wtedy przyjmujemy $\mathbf{K[k_1, k_2, ... k_c]}$ = 0 (gdy p > 1/2) lub 1 (gdy p < 1/2),
jeżeli nie przyjmujemy $\mathbf{K[k_1, k_2, ... k_c]}$ = 1 (gdy p > 1/2) lub 0 (gdy p < 1/2).

Poprawność metody znacznie rośnie wraz ze wzrostem N czy p - 1/2. Poszukujemy najefektywniejszego przekształcenia liniowego(p - 1/2 jest maksymalne) i odpowiadającego mu najlepszego prawdopodobieństwa.

Aby osiągnąć te cele musimy dokonać liniowej aproksymacji S-boxów, a następnie rozszerzyć zagadnienie na cały algorytm DES.

W praktycznym ataku ze znanym tekstem jawnym na n-rundowy DES, musimy użyć najlepszego przekształcenia dla (n-1)-rundowego DES'a przyjmując, że ostatnia runda jest odszyfrowywany z użyciem Kn oraz że funkcja F zadana jest liniowym wyrażeniem. Otrzymujemy następujące wyrażenie, które daje najlepsze p-stwo dla (n-1)-rundowego DES'a:

(2) $\mathbf{P[i_1, i_2, ... i_a] \oplus C[j_1, j_2, ... j_b] \oplus Fn(C_L, K_n)[l_1, l_2, ... l_d] = K[k_1, k_2, ... k_c],}$

Do znalezienia $\mathbf{K_n}$ oraz $\mathbf{K[k_1, k_2, ... k_c]}$ może być zastosowana następująca metoda maksymalizacji p-stwa:

Algorytm 2

Krok 1:
Dla każdej kandydatury na $\mathbf{K_n}$, czyli: $\mathbf{K_n(i) (i = 1,2, ...)}$, niech $\mathbf{T_i}$ liczbą tekstów jawnych, dla których lewa strona równania (2) równa jest zero.

Krok 2:
Niech $\mathbf{T_{max} = max(T_i)}$, a $\mathbf{T_{min} = min(T_i)}$.

- Jeżeli $\mathbf{T_{max}}$ - N/2 > $\mathbf{T_{min}}$ - N/2, wtedy przyjmujemy kandydaturę jako odpowiadającą $\mathbf{T_{max}}$ i przyjmujemy $\mathbf{K[k_1, k_2, ..., k_c]}$ = 0 (gdy p > 1/2) lub 1 (gdy p < 1/2),
- Jeżeli $\mathbf{T_{max}}$ - N/2 < $\mathbf{T_{min}}$ - N/2, wtedy przyjmujemy kandydaturę jako odpowiadającą $\mathbf{T_{min}}$ i przyjmujemy $\mathbf{K[k_1, k_2, ..., k_c]}$ = 1 (gdy p > 1/2) lub 0 (gdy p < 1/2).

Załóżmy, że p-stwo, iż $\mathbf{P[i_1, i_2, ..., i_a]}$ = 0 jest różne od 1/2. Wtedy nawet jeżeli wyeliminujemy ten czynnik z równania (2), nadal może ono pozostać efektywne. Prowadzi to do konkluzji, że algorytm 2 może być bezpośrednio zastosowany przy ataku ze znanym szyfrogramem.

Liniowa aproksymacja S-boxów.

Pierwszym zadanie jest zbadanie p-stwa, że pomiędzy bitami wejściowymi i wyjściowymi zachodzą koincydencje. Bardziej ogólnie, okazuje się to prawdą nie tylko dla pojedynczego bitu, ale także dla bitów na kilku pozycjach poddanych operacji XOR.

Definicja 1

Dla danego S-boxa Sa (a = 1, 2, ..., 8), $1 < \alpha < 63$ i $1 < \beta < 15$, zdefiniujmy wartość NSa (α,β) jako:

(3) $$NSa(\alpha, \beta) = \#\{x: 0 \le x < 64, (\oplus_{s=0}^{5} (x[s] \bullet \alpha[s])) = (\oplus_{t=0}^{3} (Sa(x)[t] \bullet \beta[t]))\},$$

gdzie • oznacza bitową operację AND, a # ilość.

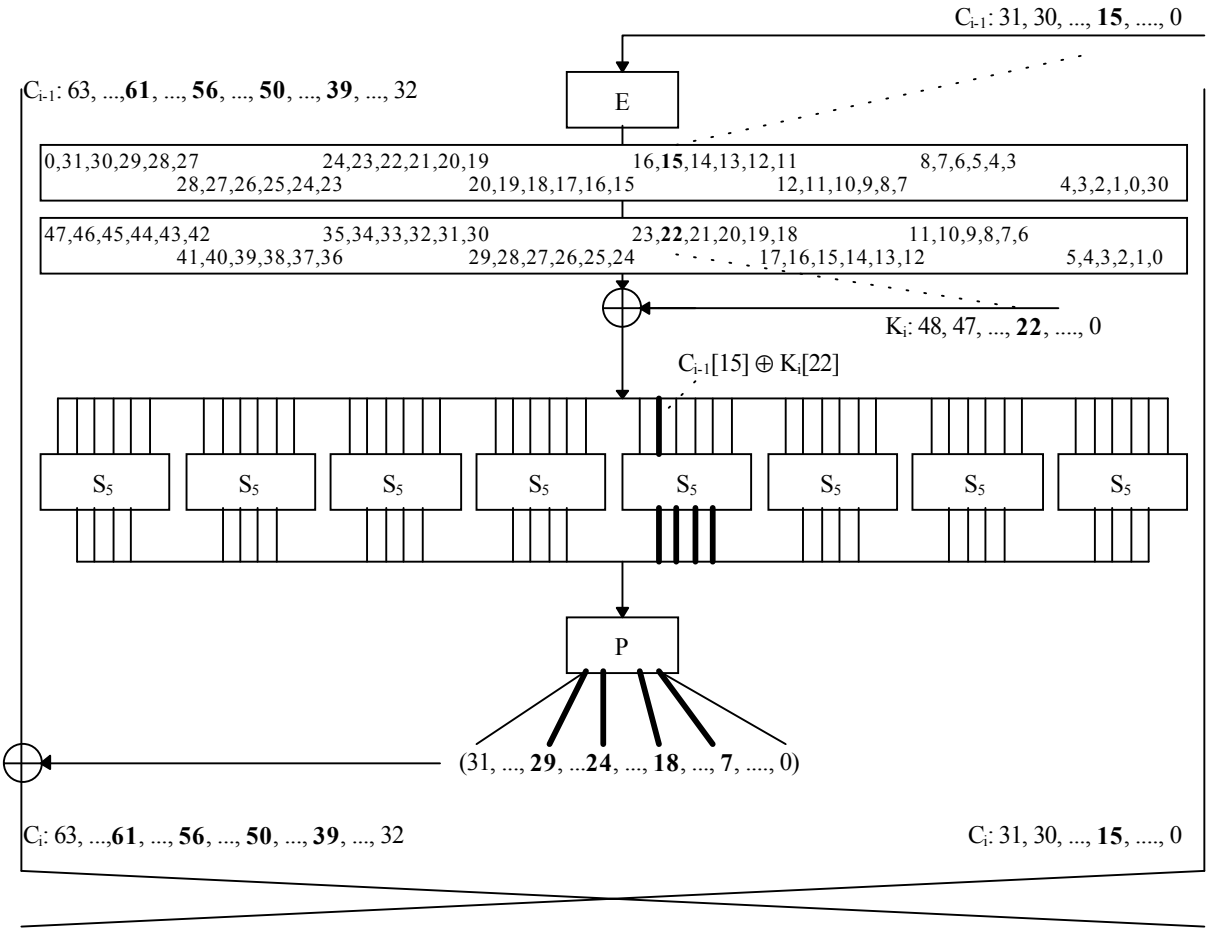
Np.
(4) NS5 (16, 15) = 12.

Jeżeli NSa (,) nie wynosi zero mówimy o korelacji pomiędzy bitami we i wy danego S-boxa. Równanie (4) mówi nam np, że czwarty bit wejściowy S5 koreluje z wartością wyjściową wszystkimi bitami wyjściowymi z prawdopodobieństwem 12/64 = 0.19.

Biorąc pod uwagę przekształcenia E i P w funkcji F widzimy następujące równanie zachodzące z p-stwem 0.19 dla ustalonego K i losowego X:

(5) $$X[15] \oplus F(X, K)[7, 18, 24, 29] = K[22].$$

Powyższe równanie otrzymywane jest na podstawie tablicy dystrybucji sporządzanych dla każdego z S-boxów. Jest ono najlepszą aproksymacją funkcji F.



Rys 1. Aproksymacja rundy skonstruowana na podstawie aproksymacji skrzynki S

Liniowa aproksymacja szyfru DES.

Następnym zadaniem jest rozszerzenie liniowej aproksymacji funkcji F na cały algorytm. Pierwszy przykład dotyczy 3-rundowego DES'a. Wykorzystując

równanie (5) otrzymujemy nową zależność dla pierwszej rundy zachodzącą z prawdopodobieństwem 12/64:

(6) X2[7, 18, 24, 29] ⊕ PH[7, 18, 24, 29] ⊕ PL[15] = K1[22].

Ta sama zależność jest prawdziwa dla ostatniej:

(7) X2[7, 18, 24, 29] ⊕ CH[7, 18, 24, 29] ⊕ CL[15] = K3[22].

Usuwając wspólne czynniki otrzymujemy liniową aproksymację dla 3-rundowego DES'a:

(8) PH[7, 18, 24, 29] ⊕ CH[7, 18, 24, 29] ⊕ PL[15] CL[15] = K1[22] ⊕ K3[22].

Prawdopodobieństwo, że równanie (8) zachodzi dla zadanych losowych P i odpowiadających im C wynosi: $(12/64)^2 + (1 - 12/64)^2 = 0.70$. Teraz możemy rozwiązać równanie (8) wyprowadzając **K1[22] ⊕ K3[22]** przy użyciu algorytmu 1.

Skuteczność algorytmu opisuje wzór:

$$p = \frac{1}{\sqrt{2p}} \int_{-2\sqrt{N}(p-1/2)}^{\infty} e^{-t^2/2} dt$$

N - liczba losowych tekstów jawnych,
p - prawdopodobieństwo, że (1) zachodzi i wartość p-1/2 jest dostatecznie mała.

Zestawienie wyników:

liczba par tekstów (N)	$1/4 \mid p -1/2 \mid^{-2}$	$1/2 \mid p -1/2 \mid^{-2}$	$\mid p -1/2 \mid^{-2}$	$2 \mid p -1/2 \mid^{-2}$
Prawdopodobieństwo sukcesu	84.1%	92.1%	97.7%	99.8%

Powtórzmy rozważania dla 5-rundowego DES'a. Równanie (5) zaadoptujemy do 2-giej i 4-tej rundy, a z obliczonej zależności zależności NS1 (27, 4) = 22 otrzymujemy równanie, które zastosujemy do 1-giej i 5-tej rundy.

(10) X[27, 28, 30, 31] ⊕ F(X, K)[15] = K[42, 43, 45, 46].

Zatem liniowa aproksymacja 5-rundowego DES'a:

(11) PH[15] ⊕ PL[7, 18, 24, 27, 28, 29, 30, 31] CH[15] CL[7, 18, 24, 27, 28, 29, 30, 31] = = K1[42, 43, 45, 46] ⊕ K2[22] ⊕ K4[22] ⊕ K5[42, 43, 45, 46].

Poniżej przedstawiono najkorzystniejsze wyrażenie aproksymujące 16-rundowego DES'a:

(13) PH[7, 18, 24] ⊕ PL[12, 16] ⊕ CH[15] ⊕ CL[7, 18, 24, 27, 28, 29, 30, 31] = = K1[19, 23] ⊕ L3 ⊕ L7 ⊕ L11 ⊕ K15[22] ⊕ K16[42, 43, 45, 46],

najkorzystniejsze prawdopodobieństwo:

(14) $p = 1/2 - 1.49 * 2^{-24}$

oraz liniowe aproksymacje funkcji F używane w każdej z rund:

E- DCA-ACD-DCA-A

gdzie:

znak ' - ' oznacza, że żadna aproksymacja nie jest potrzebna,

A: X[15] ⊕ F(X, K)[7, 18, 24,29] = K[22]
C: X[29] ⊕ F(X, K)[15] = K[44]
D: X[15] ⊕ F(X, K)[7, 18, 24] = K[22]
E: X[12, 16] ⊕ F(X, K)[7, 18, 24] = K[19, 23]

Możliwe jest złamanie 16-rundowego DES'a z użyciem 2^{47} znanych tekstów jawnych poprzez rozwiązanie następującego równania:

(15) PH[7, 18, 24] ⊕ PL[12, 16] ⊕ CH[15] ⊕ CL[7, 18, 24, 29] ⊕ F16(CL, K16) [15] = = K1[19, 23] ⊕ K3[22] ⊕ K4[44] ⊕ K5[22] ⊕ K7[22] ⊕ K8[44]

W ten sposób znajdujemy 14 bitów klucza, a pozostałe 42 powinny być odnalezione poprzez atak wyczerpujący. Tak więc można złamać DES'a z niewielką zajętością pamięci szybciej aniżeli poprzez wyczerpujące poszukiwanie 56-bitowego klucza.

3. OPIS I IMPLEMENTACJA ATAKU 2R NA ALGORYTM DES.

Zasady ataku 2R.

W poprzednim rozdziale opisana została metoda wymagająca 2^{47} znanych tekstów jawnych, wykorzystująca 2 równania opisujące 16-rundowego DES'a z wykorzystaniem aproksymacji 15-rundowej, która zachodzi z prawdopodobieństwem $1/2 - 1.19 * 2^{-22}$. Inna metoda ataku wykorzystuje aproksymację 14-

rundową, która daje najlepsze prawdopodobieństwo wynoszące $1/2 - 1.19 * 2^{-21}$:

(16) **PL[7, 18, 24] ⊕ CL[15] ⊕ CH[7, 18, 24, 29] = K2[22] ⊕ K3[44] ⊕ K4[22] ⊕ K6[22] K7[44] ⊕ K8[22] ⊕ K10[22] ⊕ K11[44] ⊕ K12[22] ⊕ K14[22],**

(17) **CL[7, 18, 24] ⊕ PL[15] ⊕ PH[7, 18, 24, 29] = K13[22] ⊕ K12[44] ⊕ K11[22] ⊕ K9[22] ⊕ K8[44] ⊕ K7[22] ⊕ K5[22] ⊕ K4[44] ⊕ K3[22] ⊕ K1[22],**

gdzie P, C i K oznaczają odpowiednio tekst jawny, szyfrogram i klucz DES'a zredukowanego do 14-tu rund.

Po zastosowaniu powyższych równań od 2-giej do 15-tej rundy w algorytmie 16-rundowym otrzymujemy następujące 2 równania:

(18) **PH[7, 18, 24] ⊕ F1(PL, K1) [7, 18, 24] ⊕ CH[15] ⊕ CL[7, 18, 24, 29] ⊕ F16(CL, K16) [15] =**
= K3[22] ⊕ K4[44] ⊕ K5[22] ⊕ K7[22] ⊕ K8[44] ⊕ K9[22] ⊕ K11[22] ⊕ K12[44] ⊕ K13[22] ⊕ K15[22],

(19) **CH[7, 18, 24] ⊕ F16(CL, K16) [7, 18, 24] ⊕ PH[15] ⊕ PL[7, 18, 24, 29] ⊕ F1(PL, K1) [15] =**
= K14[22] ⊕ K13[44] ⊕ K12[22] ⊕ K10[22] ⊕ K9[44] ⊕ K8[22] ⊕ K6[22] ⊕ K5[44] ⊕ K4[22] ⊕ K2[22].

Naszym zadaniem jest rozwiązanie tych równań w celu wyznaczenia występujących w nich bitów klucza.
W tym celu zdefiniujmy „efektywne bity tekstu" i „efektywne bity klucza" dla równania (18) i (19), jako bity które wpływają na lewe strony tych równań.

W ten sposób wyróżniliśmy 4 klasy bitów:

- efektywne bity tekstu równania (18) (13 bitów):
PL[11], PL[12], PL[13], PL[14], PL[15], PL[15], CL[0], CL[27], CL[28], CL[29], CL[30], CL[31], PH[7, 18, 24] ⊕ CH[15] ⊕ CL[7, 18, 24, 29],

- efektywne bity klucza równania (18) (12 bitów):
K1[18], K1[19], K1[20], K1[21], K1[22], K1[23], K16[42], K16[43], K16[44], K16[45], K16[46], K16[47],

- efektywne bity tekstu równania (19) (13 bitów):
CL[11], CL[12], CL[13], CL[14], CL[15], CL[15], PL[0], PL[27], PL[28], PL[29], PL[30], PL[31], CH[7, 18, 24] ⊕ PH[15] ⊕ PL[7, 18, 24, 29],

- efektywne bity klucza równania (19) (12 bitów):
K16[18], K16[19], K16[20], K16[21], K16[22], K16[23], K1[42], K1[43], K1[44], K1[45], K1[46], K1[47].

Jak widać z jednego równania możemy wyznaczyć 13 bitów klucza - 12 efektywnych i 1 bit prawej strony równania. Stąd poprzez rozwiązanie równań (18) i (19) wyznaczamy 26 bitów klucza przy pomocy 26 efektywnych bitów tekstu.

Szkic implementacji ataku

Faza zliczania danych.

Krok 1.
Przygotować 213 liczników TAtA (0tA<213) i inicjujemy je wartością zero, gdzie tA odpowiada wartościom 13 efektywnych bitów tekstu równania (18).

Krok 2.
Dla każdego tekstu jawnego P i odpowiadającego mu szyfrogramu C należy obliczyć wartość tA z kroku 1 i zwiększyć licznik TAtA o jeden.

Faza zliczania klucza.

Krok 3.
Przygotować 212 liczników KAkA (0kA<212) i inicjujemy je wartością zero, gdzie kA odpowiada wartościom 12 efektywnych bitów klucza równania (18).

Krok 4.
Dla każdego kA z kroku 3 niech KAkA będzie sumą takich TAtA , dla których lewa strona równania (18), która jest ściśle wyznaczona przez kA i tA , wynosi 0.

Krok 5.
Należy posortować KAkA w zależności od wielkości | KAkA - N/2 | i nazwać je KAlA (0lA<212). Wtedy, dla każdego lA :
Jeżeli (KAlA - N/2)0, odgadujemy lewą stronę równania (18) równą 0,
Jeżeli (KAlA - N/2)>0, odgadujemy lewą stronę równania (18) równą 1.

Kandydat na klucz odpowiadający KAlA reprezentuje lA-te pod względem prawdopodobieństwa 13 bitów klucza.

Równanie (19) powinno być rozwiązane analogicznie z zastosowaniem notacji TBtB , KBkB , KBlB, zamiast TAtA, KAkA , KAlA. Po wykonaniu tego zadania odkryliśmy 26 bitów klucza o następującym położeniu (po PC-1):

K[0], K[1], K[3], K[4], K[8], K[9], K[14], K[15], K[18], K[19], K[24], K[25], K[31], K[32], K[38],

K[39], K[41], K[42], K[44], K[45], K[50], K[51], K[54],
K[55], K[5] ⊕ K[13] ⊕ K[17] ⊕ K[20] ⊕ K[46], K[2] ⊕ K[7] ⊕ K[11] ⊕ K[22] ⊕ K[26] ⊕ K[37] ⊕ K[52].

Celem następnej fazy ataku jest znalezienie 56 - 26 = 30 bitów klucza.

Faza wyczerpującego przeglądania bitów klucza.

Krok 6.
Niech Wm (m = 0,1,2,.....) będzie ciągiem kandydatów (26 bitów) na klucz uporządkowanym w kierunku malejącego prawdopodobieństwa. Wm zależy od lA i lB.

Krok 7.
Dla kolejnych Wm poszukujemy pozostałych 30 bitów klucza dopóki nie zostanie znaleziona prawidłowa wartość całego klucza.

Szkic kodu programu w języku C.

```
for ( i=0; i<243; i++)
{
        P = Genetate_Random_Plaintext ( );
        C = Encipher_Plaintext ( P );

        TA [ 13bit_address_pointed_by_P_and_C ]++;
        TB [ 13bit_address_pointed_by_P_and_C ]++;
}

for ( k=0; k<212; k++)
        for ( t=0; t<243; t++)
        {
        if ( Left_Side_of_Equation_18 ( t, k ) == 0 )
                        KA [ k ] += TA [ t ];
        if ( Left_Side_of_Equation_19 ( t, k ) == 0 )
                        KB [ k ] += TB [ t ];
        }

Rearrange_Counters ( KA , KAlA );
Rearrange_Counters ( KB , KBlB );
for ( m=0; m<224; m++)
{
K26 = Derive_m_th_Likely_26bits ( m, KAlA, KBlB );
Return_Value = Search_Remaining_30bits ( K26 );
if ( Return_Value == FOUND )
        exit ( SUCCES );
}

exit ( FAILURE );
```

4. MOŻLIWOŚCI ROZSZERZANIA KRYPTOANALIZY LINIOWEJ

Atak tylko z szyfrogramem.

Powyższa metoda ataku może być zaadoptowana do **ataku ze znanym szyfrogramem** poprzez wykorzystanie informacji o tekście jawnym np.:

- faktu, że tekst jawny składa się ze znaków będących kodami ASCII,
- faktu, że tekst jawny został zapisany w języku naturalnym np. angielskim,
- wyznaczenie takiej aproksymacji, niekoniecznie najlepszej jeśli chodzi o prawdopodobieństwo, która wykorzystuje wyłącznie te bity tekstu jawnego, które są zerami np. co siódmy bit wejściowy w przypadku kodów ASCII.

Kierunki rozwoju kryptoanalizy liniowej.

- aproksymacja więcej niż jednego S-boxa w jednej rundzie,
- wykorzystanie więcej niż jednego równania do wyznaczania tych samych bitów klucza (w przypadku DESa dało zmniejszenie liczby wymaganych tektów jawnych o czynnik 38),
- wykorzystanie elementów nieliniowych,
- uogólnienie analizy liniowej,
- wykorzystanie łańcuchów Markowa do modelowania prawdopodobieństwa,
- analiza liniowo-różnicowa.

Kryptoanaliza liniowa innych szyfrów blokowych.

- FEAL
Analiza liniowa jest skutecznym atakiem na FEAL.

- IDEA
Próbę kryptoanalizy liniowej algorytmu IDEA podjęli Hawkes i O'Connor. Okazało się, że przeprowadzenie ataku jest znacznie bardziej skomplikowane niż w przypadku algorytmu DES. Wynika z trzech powodów. Po pierwsze operacje nieliniowe w IDEAi są zależne od klucza, a ponadto operują na dużych blokach danych. Po drugie podklucze są połączone operacją nieliniową. Po trzecie wyjście aproksymowanych operacji jest często wejściem do innej operacji aproksymowanej w tej samej rundzie, z tego powodu nie zawsze można zakładać niezależność aproksymacji przy założeniu niezależności kluczy - powstaje więc problem, czy stosować lemat Piling-Up, a jeśli tak to w jaki sposób.

Najbardziej nieliniową operacją w IDEAi jest mnożenie, toteż aproksymacji mnożenia poświęcimy najwięcej uwagi. Propozycja Hawkesa i O'Connora to aproksymacja mnożenia za pomocą tzw. aproksymacji najmłodszych bitów (LSB). Uważają oni, że jest to najlepsza aproksymacja dla IDEAi, ponieważ przybliża mnożenie z dużym prawdopodobieństwem, a dodawanie z prawdopodobieństwem 1.

Podsumowując IDEA okazała się odporna na kryptoanalizę liniową.

- RC5

Przeprowadzone dotychczas badania nad RC5 wykazały, że algorytm ten z zalecanymi parametrami (RC5-32/12/16) jest odporny na kryptoanalizę liniową. Kryptoanaliza 5-rundowego RC5 wymaga analizy 2^{47} par tekstów, czyli 8 razy tyle co w przypadku 16-rundowego algorytmu DES.

- LOKI89

Badania skrzynek S przeprowadzone przez Tokita i in. wykazały, że dla LOKI89 istnieje tylko jedna aproksymacja rundy, która daje najlepsze prawdopodobieństwo równe $1/2 + 1.25 * 2^{-5}$.

Z tego powodu można się spodziewać, że LOKI będzie odporny na kryptoanalizę liniową. Poza badaniem własności skrzynek S, Tokita i in. zastosowali algorytm wyznaczania najlepszego wyrażenia liniowego Matsui i wyznaczyli najlepsze prawdopodobieństwa dla 16-rundowego LOKI89 $p = 1.37 * 2^{-49}$.

Okazało się więc zgodnie z przypuszczeniami, że LOKI89 jest odporny na kryptoanalizę liniową.

SPIS LITERATURY:

[1] E. Biham, A. Shamir, „Differential Cryptanalysis of the Data Encryption Standard", Springer Verlag 1993, ISBN 3-540-97930-1

[2] E. Biham, „On Matsui's Linear Cryptanalysis", Advances in Cryptology Eurocrypt'94, Springer Verlag 1994, ISBN 3-540-60176-7

[3] U. Blöcher, M. Dichtl, „Problems with the Linear Cryptanalysis of DES Using more than one Active S-Box per Round", Fast Software Encryption, Springer Verlag 1994, ISBN 3-540-60590-8

[4] C. Harpes, G.G. Kramer, J. L. Massey, „A Generalization of Linear Cryptanalysis and Applicability of Matsui's Piling-Up Lemma", Advances in Cryptology Eurocrypt'95, Springer Verlag 1995, ISBN 3-540-59409-4

[5] Ph. Hawkes, L. O'Connor, „On Applying Linear Cryptanalysis to IDEA", Advances in Cryptology Asiacrypt'96, ISBN 3-540-61872-4

[6] FIPS PUB 46-2 Data Encryption Standard, NIST ...

[7] B. S. Kaliski Jr., M.J.B Robshaw, „Linear Cryptanalysis Using Multiple Approximations", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5

[8] B. S. Kaliski, Y. L. Yin, „ On differential and Linear Cryptanalysis of the RC5 Encryption Algorithm", Advances in Cryptology Crypto'95, Springer Verlag 1995, ISBN 3-540-60221-6

[9] L.R. Knudsen, M.J.B. Robshaw, „Non-Linear Approximations in Linear Cryptanalysis, Advances in Cryptology Eurocrypt'96, Springer Verlag 1996, ISBN 3-540-61186-X

[10] X. Lai, „On the Design and Security of Block Ciphers", ETH Series in Information Processing, vol. 1, Zurich 1992.

[11] S. Langford, M.E. Hellman, „Differential-linear Cryptanalysis", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5

[12] M. Matsui, „ Linear Cryptanalysis Method for DES Cipher", Advances in Cryptology Eurocrypt'93

[13] M. Matsui, „On Correlation Between the Order of S-boxes and the Strength of DES", Advances in Cryptology Eurocrypt'94, Springer Verlag 1994, ISBN 3-540-60176-7

[14] M. Matsui, „The First Experimental cryptanalysis of Data Encryption Standard", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5

[15] A. J. Menezes, P. C. van Oorschot, S. A. Vanstone, „Handbook of Applied Cryptography", CRC Press 1997, ISBN 0-8493-8523-7

[16] K. Ohta, K. Aoki, „Linear Cryptanalysis of the Fast Data Encipherment Algorithm", Advances in Cryptology Crypto'94, Springer Verlag 1994, ISBN 3-540-58333-5

[17] K. Ohta, S. Morai, K. Aoki, „Improving the Search Algorithm for Best Linear Expression", Advances in Cryptology Crypto'95, Springer Verlag 1995, ISBN 3-540-60221-6

[18] R. L. Rivest, „The RC5 Encryption Algorithm", Fast Software Encryption, Springer Verlag 1995.

[19] B. Schneier, „Kryptografia dla praktyków", WNT

[20] A. A. Selçuk, „New Results in Linear Cryptanalysis of RC5", Fast Software Encryption 1998, Springer Verlag 1998, ISBN 3-540-64265-X

[21] T. Tokita, T. Sorimachi, M. Matsui, „Linear Cryptanalysis of LOKI and S^2DES", Advances in Cryptology Asiacrypt'94, Springer Verlag 1994, ISBN 3-540-59339-X