TEMA 9

Probleme de securitate ale algoritmilor criptografici (metode de criptanaliză)

Cuprins

- **Prezentare generală**
- Criptanaliza cifrurilor bloc:
  - Criptanaliza Liniară
  - Criptanaliza Diferențială
  - Alte tipuri de atac
  - Analiza Statistică
- Criptanaliza cifrurilor fluide
- Cazul general
  - Side Channel Attacks

Prezentare generală

Ce este criptanaliza?

- Teorie
  - distinsă de aleatoriu
- Mai puțin lucru decât analiza exhaustivă, chiar dacă nu este practic 2^127 vs 2^100
- Practic – recuperarea biților cheii, determinarea biților textului textului clar/ cifrat

Cuprins

- Prezentare generală
- **Criptanaliza cifrurilor bloc**
  - Criptanaliza Liniară
  - Criptanaliza Diferențială
  - Alte tipuri de atac
  - Analiza Statistică
- Criptanaliza cifrurilor fluide
- Cazul general
  - Side Channel Attacks

Originile Criptanalizei Diferențiale și Liniare

- Criptanaliza diferențială definită inițial pe DES
- Eli Biham și Adi Shamir, *Differential Cryptanalysis of the Data Encryption Standard*, Springer Verlag, 1993.
- Criptanaliza liniară definită inițial pe Feal de Matsui și Yamagishi, 1992.
- Matsui publică mai târziu un atac liniar pe DES.

Cereri de Text clar, Text cifrat

- Doar Text cifrat
- Text clar cunoscut: avem o mulțime de perechi de text calr și text cifrat (P_1,C_1), (P_2,C_2) … (P_i,C_i):

- Text clar ales:
  - Alegem P_i, obținem și C_i respectiv

- Text criptat ales:
  - Alegem C_i, obținem P_i respectiv

- Text clar ales - Text cifrat ales:
  - Alegem P_i și C_j, obținem C_i și C_j

Cereri de Text clar, Text cifrat

Cereri date (P_1,C_1), (P_2,C_2) … (P_i,C_i):

- Text clar adaptiv:
  - Introducem P_i, obținem C_i, alegem P_{i+1} …

- Text cifrat adaptiv :
  - Introducem C_i, obținem P_i, alegem C_{i+1} …

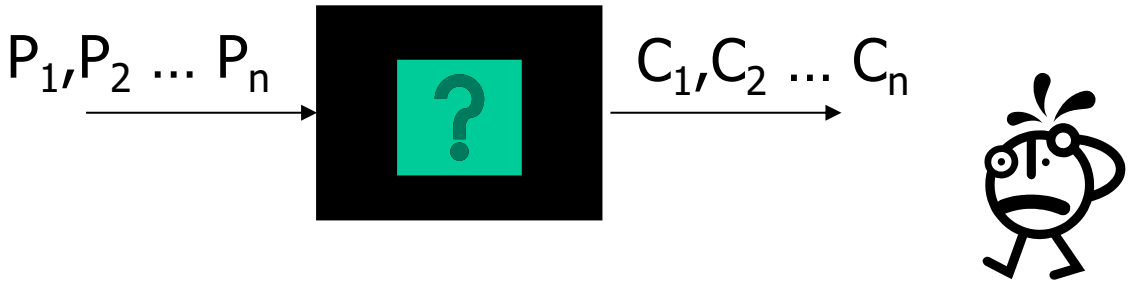
- Text clar adaptiv – Text cifrat adaptiv:
  - Introducem P_i și obținem C_i sau introducem C_i și obținem P_i apoi alegem următoarea cerere

Alte categorii de atac

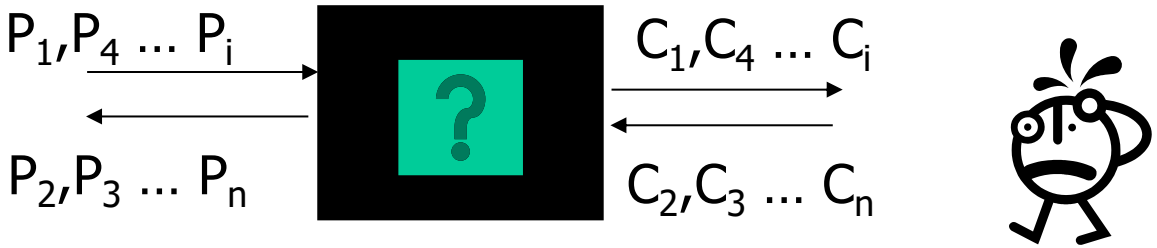
- cu chei relaționate - adversarul alege relația dintre chei, dar nu cheile în sine, și obține perechile de de text clar – text cifrat

PRP, SPRP

- Cutia conține fie cifrul bloc, fie o permutare aleatorie
- Permutarea pseudoaleatorie (PRP): Atacatorul nu poate crea polinominal mai multe cereri de text clar ales adaptiv sau text cifrat ales adaptiv (dar nu ambele) și determină conținutul cutiei cu probabilitatea ½ + *e* pentru un *e* > 0 non-neglijabil.



- Strong PRP (SPRP): aceeași idee ca la PRP, însă poate crea cereri în ambele direcții



Limitele atacului

- Dacă un atac are loc cu probabilitatea $\leq 2^{-x}$
- $x > 0$
- Mărimea blocului b
- Dacă $x \geq b$, atacul are nevoie de $\geq 2^{b}$ texte clare

Cuprins

- Prezentare generală
- **Criptanaliza cifrurilor bloc:**
  - **Criptanaliza Liniară**
  - Criptanaliza Diferențială
  - Alte tipuri de atac
  - Analiza Statistică
- Criptanaliza cifrurilor fluide
- Cazul general
  - Side Channel Attacks

Criptanaliza liniară

Notații

- P = text clar (plaintext)
- p_i = bitul *i* al lui P
- C = text cifrat
- c_i = bitul *i* al lui C
- K = Cheia (Key) (inițială sau expandată)
- k_i = bitul i al cheii K
- $\oplus_{i=1,n} p_i = p_1 \oplus p_2 \oplus \ldots \oplus p_n$
- X, Y, Z – submulțimi de biți

Criptanaliza liniară

Privire generală a atacului

- Se obține aproximarea (ările) lineară ale P,K,C relaționate cifrului

$$\oplus_{i \in X,}\ p_i \oplus_{j \in Y} c_j = \oplus_{g \in Z} k_g$$

care apar cu probabilitatea pr = ½ + e cu abaterea maximă $-½ \leq e_i \leq ½$.

- Se criptează niște P aleatori, se obțin C și se calculează respecivele k_g.
- Atacul cu text clar cunoscut
- Se ghicesc biții rămași ai cheii via căutare exhaustivă.

Exemplu – Single S-Box

P_2P_1 \ K_2K_1	00	01	10	11
00	10	11	00	01
01	11	00	01	10
10	00	01	10	11
11	01	10	11	00

perechi (P,C)

(a) $00 \rightarrow 00$

(b) $01 \rightarrow 01$

(c) $10 \rightarrow 10$

unde

$P_1 \oplus C_1 = 0$

$P_2 \oplus C_2 = 0$

Considerăm doar relațiile între 1 bit de intare 1 bite de ieșire și un bit al cheii:

(1) $Pr(P_1 \oplus C_1 = K_1) = 1$

(2) $Pr(P_2 \oplus C_2 = K_1) = 5/8$

(3) $Pr(P_2 \oplus C_2 = K_2) = 3/8$

Pentru oricare alte triplete P_i, C_i, K_i

$Pr(P_i \oplus C_i = K_i) = ½$

Utilizăm (1) și (3) pentru determinarea cheii.

Putem determina K_1 din o (P,C) cu (1)

$P_1 \oplus C_1 = 0 = K_1$

Doar o $P_2 \oplus C_2 = 0$ nu este suficient pentru a deduce că K_2 e 1

E nevoie de o (P,C) adițională

(3) returnează 0, ce implică $K_2 = 1$.

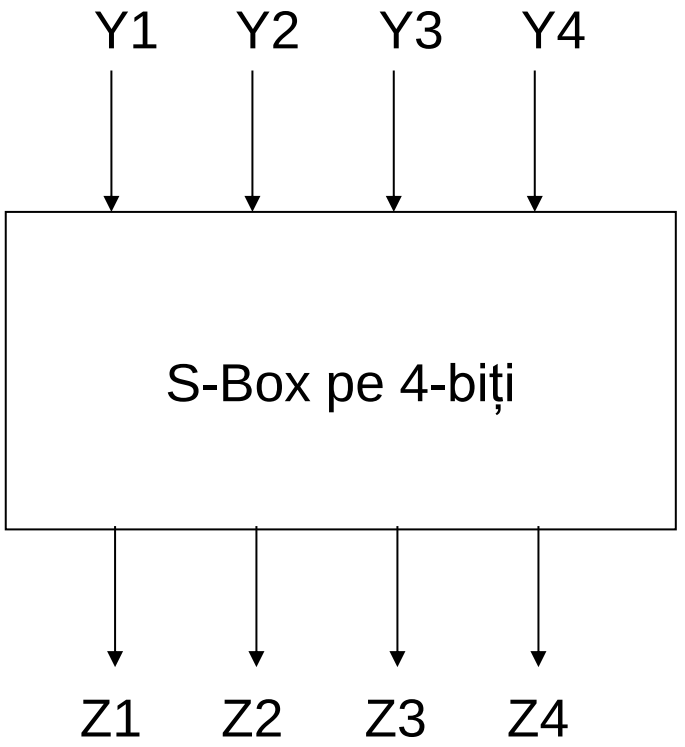
Ghicim că cheia = 10

Exemplu S-Box

Intrare: Ieșire (4 biți, hex)
0:E
1:4
2:D
3:1
4:2
5:F
6:B
7:8
8:3
9:A
A:6
B:C
C:5
D:9
E:0
F:7

Exemplul de S-Box e luat din *Tutorial on Linear and Differential Crypt*. H. Heys, Memorial U. of of Newfoundland

Exemplu S-Box



Y2 ⊕ Y3 = Z1 ⊕ Z3 ⊕ Z4 în 12 din cele 16 intrări
12/16 = ½ + ¼ abaterea este ¼
Y1 ⊕ Y4 = Z2 în ½ din perechi, deci nu avem abatere
Y3 ⊕ Y4 = Z1 ⊕ Z4 în 2 din cele16 perechi, deci abaterea este -3/8
2/16 = ½ -3/8

Găsirea relațiilor liniare

Forma generală a relației liniare:

$a_1Y_1 \oplus a_2Y_2 \oplus a_3Y_3 \oplus a_4Y_4$

$=$

$b_1Z_1 \oplus b_2Z_2 \oplus b_3Z_3 \oplus b_4 Z_4$

$a_i, b_i \in \{0,1\}$

Sumăm toate ecuațiile din tabel

(Se face doar o singură dată)

Găsirea relațiilor liniare

b1b2b3b4

a1a2a3a4

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	-2	-2	0	0	-2	6	2	2	0	0	2	2	0	0
2	0	0	-2	-2	0	0	-2	-2	0	0	2	2	0	0	-6	2
3	0	0	0	0	0	0	0	0	2	-6	-2	-2	2	2	-2	-2
4	0	2	0	-2	-2	-4	-2	0	0	-2	0	2	2	-4	2	0
5	0	-2	-2	0	-2	0	4	2	-2	0	4	-2	0	-2	-2	0
6	0	2	-2	4	2	0	0	2	0	-2	2	4	-2	0	0	-2
7	0	-2	0	2	2	-4	2	0	-2	0	2	0	4	2	0	2
8	0	0	0	0	0	0	0	0	-2	2	2	-2	2	-2	-2	6
9	0	0	-2	-2	0	0	-2	-2	-4	0	-2	2	0	4	2	-2
A	0	4	-2	2	-4	0	2	-2	2	2	0	0	2	2	0	0
B	0	4	0	-4	4	0	4	0	0	0	0	0	0	0	0	0
C	0	-2	4	-2	-2	0	2	0	2	0	2	4	0	2	0	2
D	0	2	2	0	-2	4	0	2	-4	-2	2	0	2	0	0	2
E	0	2	2	0	-2	-4	0	2	-2	0	0	-2	-4	2	-2	0
F	0	-2	4	-2	-2	0	2	0	0	-2	4	-2	-2	0	2	0

De un număr de ori are loc relația:

$$a_1Y_1 \oplus a_2Y_2 \oplus a_3Y_3 \oplus a_4Y_4 = b_1Z_1 \oplus b_2Z_2 \oplus b_3Z_3 \oplus b_4 Z_4$$

Găsirea relațiilor liniare

- Valoarea “a” a lui E: a1 =1, a2 = 1, a3 = 1, a4 = 0
- Valoarea “b” a lui 1: b1 = 0, b2 = 0, b3 = 0, b4 = 1
- Linia E, Coloana 1 dau valoarea 2
- Abaterea este 2/16 = 1/8
- Probabilitatea că X1⊕ X2 ⊕ X3 = Y4 este
  ½ + 1/8 = 5/8

Lema Piling-Up

Matsui

- Know $Pr(V_i = 0) = ½ + e_i$
- $Pr(V_1 \oplus V_2 \oplus \ldots \oplus V_n = 0) = ½ + 2^{n-1} \prod_{i=1}^{n} e_i$

- V_i sunt variabile aleatoare independente
- e_i este abaterea $-½ \leq e_i \leq ½$

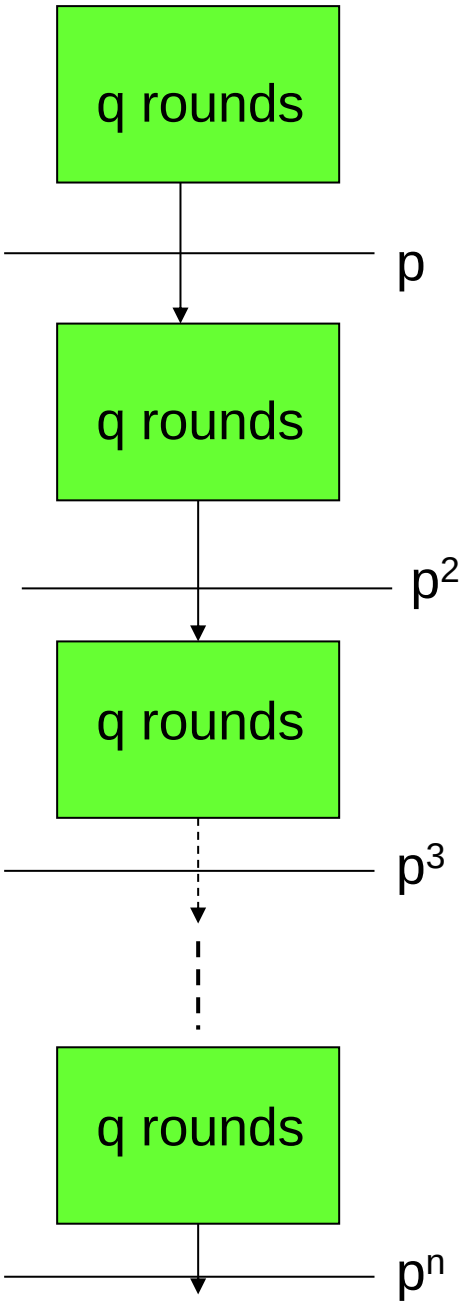
Se utilizează pentru combina ecuațiile liniare dacă le percpem ca pe niște variabile aleatoare independente

Găsirea relațiilor liniare

- Aplicăm pentru alți pași din interiorul funcției de rundă aceleași procese utilizate pentru S-Box
- Determinăm ecuațiile pentru întreaga rundă
- Incorporăm *whitening* (dacă există) în ecuații

Marginile liniare

- Mărginim ecuația liniară pe parcursul a *q* runde: $0 < p \leq 1$
- Cifrul are *nq* runde
- Estimăm marginea superioară $\leq p^n$
- 2^b texte clar posibile
- $\leq 2^b/p^n$ satisfac ecuațiile
- Biții cheii de rundă, ieșirea rundei/ intrarea rundei următoare nu sunt independenți
- Dacă $p^n \leq 2^{-b}$, nu are loc atacul



Aplicarea atacului

Când atacăm cifrul, încercăm să determinăm biții cheii primei sau ultimei runde, apoi repetăm atacul pe o versiune cu numărul de runde redus a cifrului

DES are 16 runde, căutăm cheia pentru prima sau ultima rundă, repetăm atacul pentru versiunea cu 15 runde …

Dacă aceiași biți ai chei expandate sunt utilizați în runde multiple, completăm biții cheii de rundă cum se fac cunpscuți

Criptanaliza liniara DES

- Aproximări liniare determinate prin căutare exhaustivă
  - Mai întâi pentru Boxle-S
  - Apoi extins pentru funcția de rundă și runde multiple.
- Aproximări
  - 5 aproximări bune pentru biții inițiali ai cheii cu abaterea *e* se plasează în diapazonul $\approx$ 0.031 to 0.218
  - Exemple,
    - Runda 1: $\oplus_{i \in X} fo_{i,1} \oplus p_{15} = k_{22}$ $X = \{7,18,24,29\}$ cuprobabiliattea 0.19
    - Ultima rundă : $\oplus_{i \in X} fo_{i,16} \oplus fin_{15,16} = k_{22}$ $X = \{7,18,24\}$ cu probabilitatea 0.66
  - 1 aproximare pentru biții cheii de rundă cu $e = O(2^{-3})$.
  - Restul cu $e = O(2^{-5})$ până la $O(2^{-30})$

fin_{ij} = bitul *i* al intrării funcției de rundă în runda *j*

fo_{ij} = bitul *i* al ieșirii funcției de rundă în runda *j*

Criptanaliza liniara DES

- Atac asupra textului clar
  - Găsim 14 biți ai cheii.
  - Ceilalți 42 biți îi găsim prin căutare exhaustivă.
  - 8 runde necesită 2^{21} P cu un succes de 96%.
  - 16 runde necesită 2^{47} P cu un succes de 96%
- Atac doar cu text cifrat
  - Găsim 7 biți ai cheii.
  - Presupunem că câțiva p_i sunt 0 pentru a avea ecuații doar cu C și K.
  - 8 runde necesită 2^{37} C cu 78% succes, presupunând că 1 p_i este 0
  - 16 runde necesită 1.82×2^{53} C cu 78% succes, presupunând că 5 p_i sunt 0.

Marginea liniară AES

- 4 runde $\leq 2^{-75}$
- 8 runde $\leq 2^{-150}$
  exponentul > 128, deci nu este necesar de estimat toate 10 runde

Cuprins

- Prezentare generală
- **Criptanaliza cifrurilor bloc:**
  - Criptanaliza Liniară
  - **Criptanaliza Diferențială**
  - Alte tipuri de atac
  - Analiza Statistică
- Criptanaliza cifrurilor fluide
- Cazul general
  - Side Channel Attacks

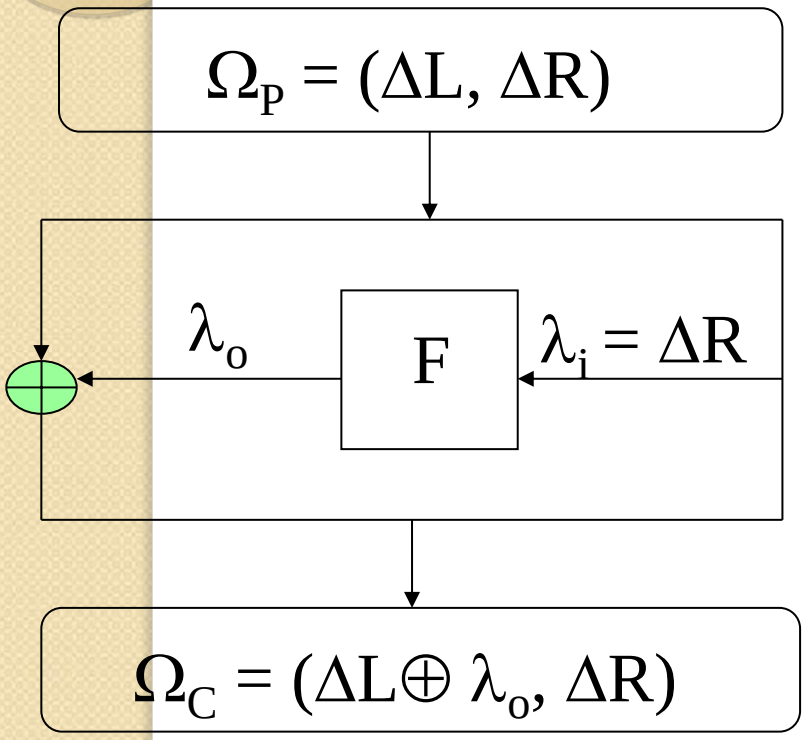
Criptanaliza Diferențială

Notații

- P = text clar
- C = text cifrat
- (P1,P2) = pereche text clar
- (C1,C2) = pereche text cifrat
- $\Delta P = P1 \oplus P2$
- $\Delta C = C1 \oplus C2$
- Caracteristica: $\Omega = (\lambda_{i1},\lambda_{o1},\lambda_{i2},\lambda_{o2},\ldots\lambda_{ir},\lambda_{or})$
  - λ_{ij}= $\oplus$ intrărilor la runda j
  - λ_{oj}= $\oplus$ a ieșirilor din runda j
  - dacă pr_j= probabilitatea λ_{oj} apare λ_{ij} dat atunci probabilitatea $\Omega = \Pi\ pr_j$ (marginea superioară)

Exemplu: Ω runda 1

Prima rundă a oricărei rețele Feistel
nu ajută la prevenirea criptanalizei diferențiale.



Dacă $\Delta R = 0$ atunci

$\lambda_o = 0$

$\Omega_C = (\Delta L, 0)$

cu probabilitatea 1.

Dacă $\Delta R = 60\ 00\ 00\ 00$ atunci

$\lambda_o = 00\ 80\ 82\ 00$

$\Omega_C = (\Delta L \oplus 00\ 08\ 82\ 00,$
$60\ 00\ 00\ 00)$

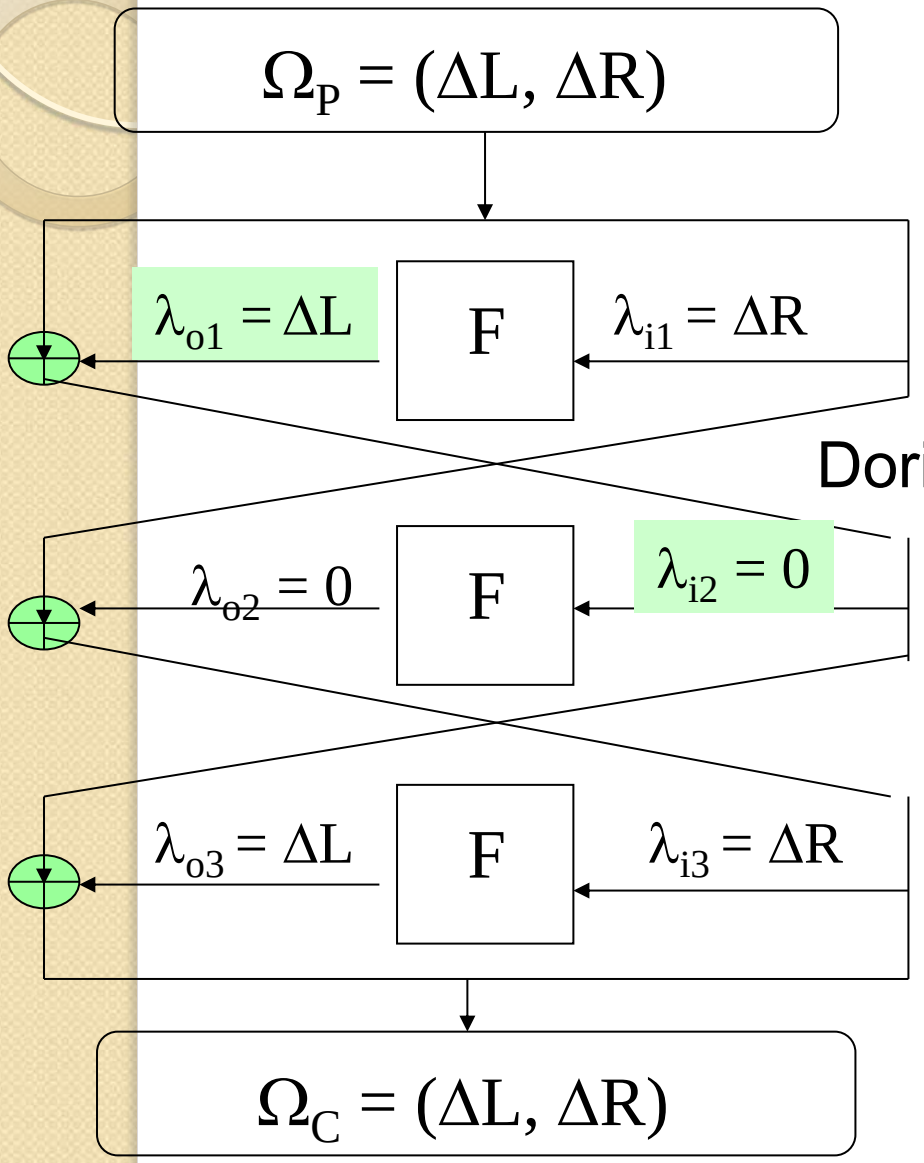
cu probabilitatea 14/64.

DES fără permutarea inițială și finală.

Găsirea caracteristicilor

- Proces similar cu cel folosit în exemplul criptării liniare
- Enumerarea tuturor cazurilor
- Se face doar o singură dată pentru tot procesul

Criptanaliza Diferențială - DES



Ω Cu 3 runde, unde $\Delta P = \Delta C$

Probabilitatea $(14/64)^2 \approx 0.048$

14/64

Dorim ieșirea primului F pentru a anula ΔL

1

14/64 Același Δ ca la intrarea primului F

Criptanaliza Diferențială

Privire generală a atacului

- Găsirea Ω cu o probabilitate non-neglijabilă.
  - Minimul de biți ai cheii de ghicit, însă permite ghicirea celor din ultima (sau prima) rundă.
  - Căutare exhaustivă pentru găsirea celor mai bune Ω.
- Determinarea biților cheii **ultimei runde**:
  - Alegem perechile (P1,P2) astfel încât ΔP furnizează λ_{i1} .
  - Decriptăm textul cifrat cu cheia ghicită pentru ultima rundă
  - Calculăm numărul de perechi (C1,C2) ca să se potrivească caracteristicii
  - Presupunem că biții cheii sunt ghiciți cu numărul maxim calculat.
  - Eliminam ultima rundă și atacăm cifrul redus.
- Putem de asemenea lucra cu **prima runda**:
  - Alegem perechile (C1,C2) astfel încât $\Delta C = \lambda_{or}$
  - Determinăm biții cheii în prima rundă.

Găsirea Ω

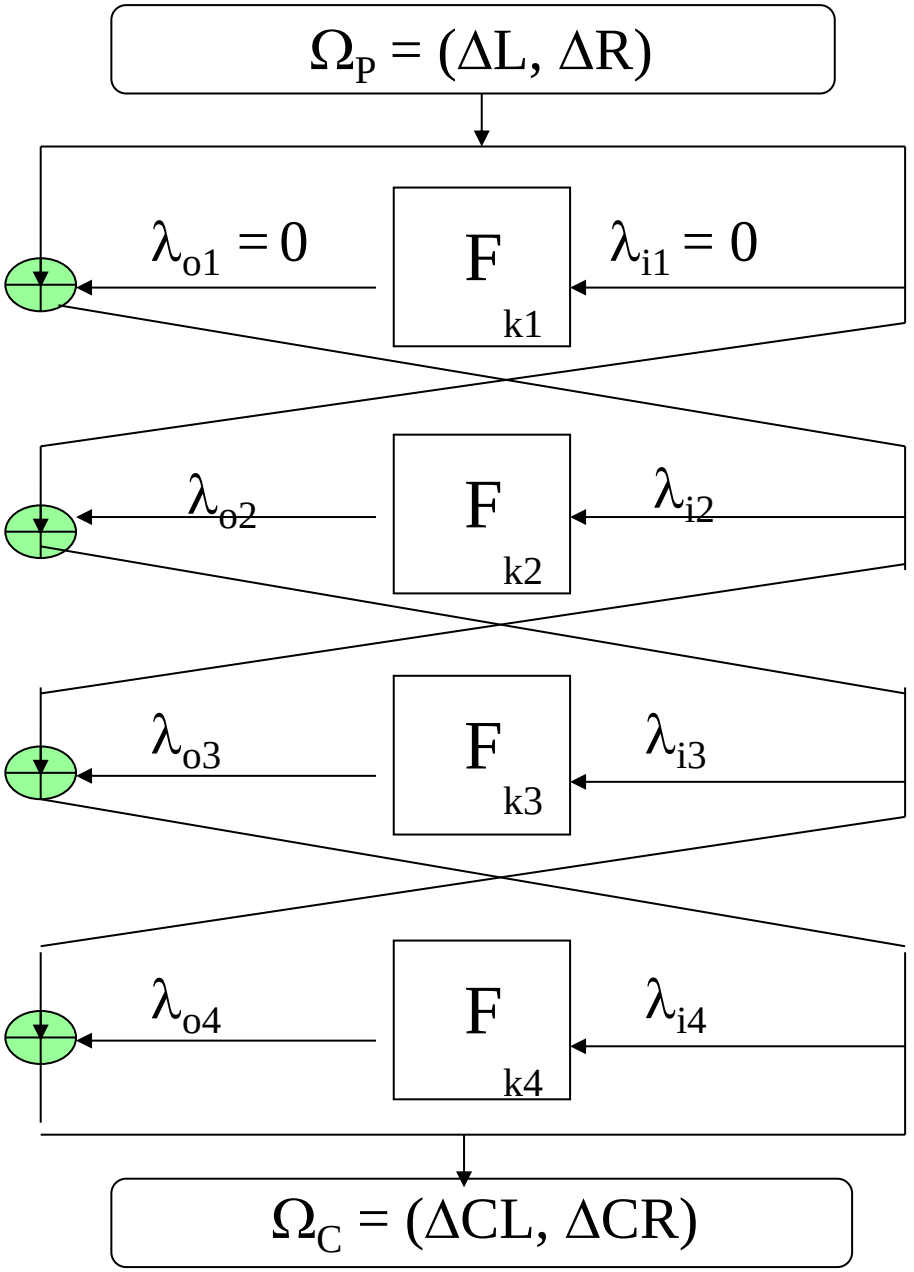
Tabelele de distribuire create manual pentru intrările ⊕ și ieșirile ⊕ pentru fiecare S-Box.

	Output ⊕					
Input ⊕	…	2	3	4	5	…
2	…	0	8	0	4	…
3	…	2	2	10	6	…

Segmentul tabelului de distribuire pentru S-Box 0 din DES

Dacă intrarea ⊕ este 2, ieșirea ⊕ este 5 - 4 chei posibile.

Criptanaliza Diferențială - DES



Ω - 4 runde

Ω_P cu

$\Delta L = 20\ 00\ 00\ 00$

$\Delta R = 00\ 00\ 00\ 00$

Atunci

$\lambda_{o1} = 00\ 00\ 00\ 00$

$\lambda_{i2} = \Delta L = 20\ 00\ 00\ 00$

λ_{i2} afectează doar prima S-Box, deci 28 biți ai λ_{o2} sunt 0.

$\lambda_{o4} = \lambda_{i3} \oplus \Delta CL$

$= \lambda_{i1} \oplus \lambda_{o2} \oplus \Delta CL$

$= \lambda_{o2} \oplus \Delta CL$

cunoaștem toți biții în afară de 4 ai λ_{o2}

Cunoaștem jumătățile de dreapta ale textului cifrat $\Rightarrow$ cunoaștem intrarea în runda 4. λ_{i4} : cel mult 11 biți nenuli ai ΔCR variază între perechi.

Criptanaliza Diferențială

Numărul de texte în clar

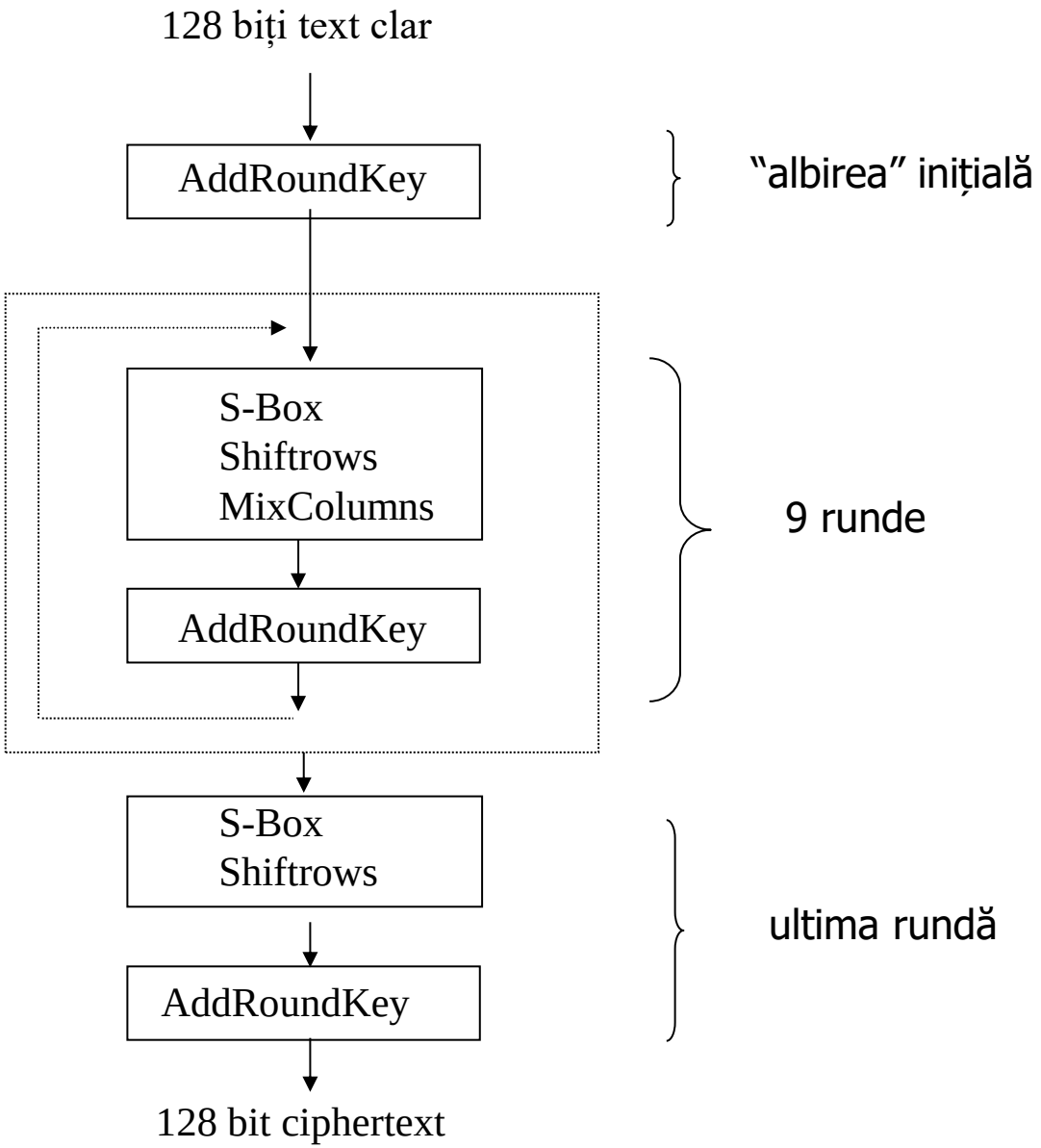
- Folosim m = c/pr(Ω) perechi de text în clar, pentru unele c > 0 (c - mici).
- Text clar ales: Selectăm m perechi care satisfac ΔP.
- Text clar cunoscut: avem mulțimea de P-uri, însă nu le alegem, deci este nevoie de găsit perechile care satisfac ΔP.
  - $2^{|P|/2}(2m)^{½}$ texte în clar neceasre
    - Putem forma $½ (2^{|P|/2}(2m)^{½})^2 = 2^{|P|}m$ perechi.
    - $2^{|P|}$ de ΔP posibile.
    - $2^{|P|}m/ 2^{|P|} = m$ perechi în mediu creează ΔP.
  - Dacă > numărul de P avute, atacul nu e posibil.

Criptanaliza Diferențială- DES

- Orice versiune cu runde reduse ale lui DES se poate sparge prin text clar cunoscut mai rapid decât cu căutarea exhaustivă a cheii.

Nr de runde	Nr de texte în clar alese	Nr de texte în clar cunoscute
4	2^{3}	2^{33}
6	2^{8}	2^{36}
8	2^{14}	2^{38}
9	2^{24}	2^{44}
11	2^{31}	2^{47}
13	2^{39}	2^{52}
16	2^{47}	2^{55}

AES – 128 biți


128 biți text clar
AddRoundKey
"albirea" inițială
S-Box
Shiftrows
MixColumns
AddRoundKey
9 runde
S-Box
Shiftrows
AddRoundKey
ultima rundă
128 bit ciphertext

AES Differentials

- AES: fiecare bit nenul în delta de intrare în rundă contribuie cu o probabilitate de 2^{-6} sau 2^{-7} la diferența de ieșire.
  - Dacă diferența de intrare în rundă este 0 cu excepția unui bit, diferența probabilă specifică de apariție la ieșirea rundei este $\leq 2^{-6}$
  - Dacă diferența de intrare în rundă este 0 cu excepția a 2 biți, diferența probabilă specifică de apariție la ieșirea rundei este $\leq 2^{-12}$
- Doar datorită S-Box – ceilalți pași în rundă nu influențează probabilitatea diferențială

Diferențialele AES

- Marginea pentru 2 runde: $\leq 2^{-24}$
- Marginea pentru 4 runde : $\leq 2^{-96}$

suficient de mică pentru eliminarea atacului diferențial asupra 10 runde!!!

Cuprins

- Prezentare generală
- **Criptanaliza cifrurilor bloc**:
  - Criptanaliza Liniară
  - Criptanaliza Diferențială
  - **Alte tipuri de atac**
  - Analiza Statistică
- Criptanaliza cifrurilor fluide
- Cazul general
  - Side Channel Attacks

Atacul Boomerang

- P,P'.Q,Q' – texte în c lar
- C,C',D,D' texte cifrate respective
- Cifrul e un șir de runde
- E = funcția de criptare
- Privim E ca compunerea a două funcții E_0,E_1
  - De ex., dacă E constă din *n* runde, E_0 reprezintă primele n_0 runde, E_1 – restul n-n_0 runde
  - $E(P) = E_1(E_0(P))$

Atacul Boomerang

- Caracteristica pentru E_0 : $\Delta \rightarrow \Delta^*$
- Caracteristica pentru E_1^{-1}: $\nabla \rightarrow \nabla^*$

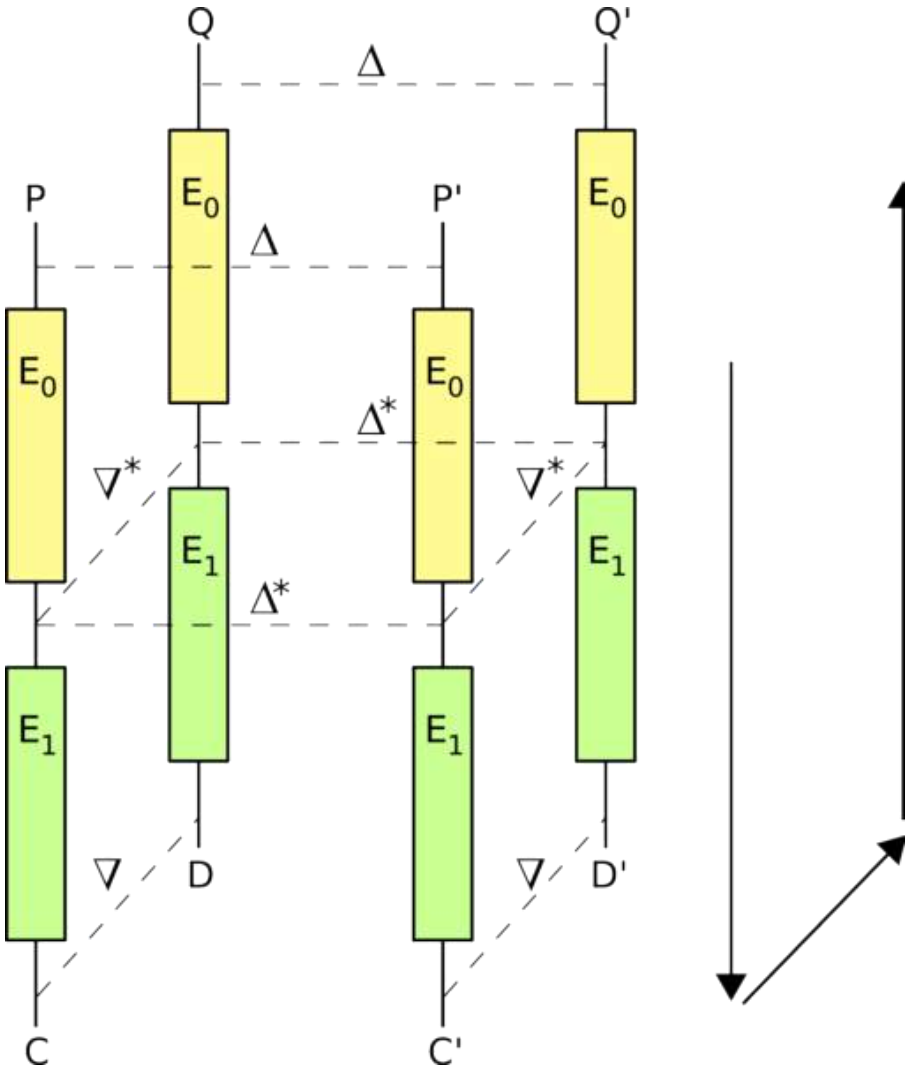
Alegem texte în clar astfel încât

- $P \oplus P'$ produce $\Delta \rightarrow \Delta^*$
- $P \oplus Q$ produce $\nabla \rightarrow \nabla^*$
- $P' \oplus Q'$ produce $\nabla \rightarrow \nabla^*$

Apoi arătăm că

- $D \oplus D'$, $Q \oplus Q'$ corespund la $\Delta^* \rightarrow \Delta$ pentru E_0^{-1}

Atacul Boomerang



Atacul Boomerang

$E_0(Q) \oplus E_0(Q') =$

$= E_0(Q) \oplus E_0(Q') \oplus E_0(P) \oplus E_0(P) \oplus E_0(P') \oplus E_0(P') =$

$= [E_0(P) \oplus E_0(P')] \oplus [E_0(P) \oplus E_0(Q)] \oplus [E_0(P') \oplus E_0(Q')] =$

$= [E_0(P) \oplus E_0(P')] \oplus [E_1^{-1}(C) \oplus E_1^{-1}(D)] \oplus$

$\oplus [E_1^{-1}(C') \oplus E_1^{-1}(D')] = \Delta^* \oplus \nabla^* \oplus \nabla^* = \Delta^*$

Atacul Boomerang

Găsim caracteristica valabilă pentru E_0 și cea pentru E_1

Generăm perechi folosind cereri text în clar ales – text cifrat ales:

- $P' = P \oplus \Delta$
- Solicităm P,P' care să fie cifrate în C, C'
- $D = C \oplus \nabla$
- $D' = C' \oplus \nabla$
- Solicităm D, D' care să fie decriptate în Q,Q'

Diversificarea cheilor

- Conceput pentru a fi eficient
- Rekeying (exemplu de aplicații de rețea care gestionează mai multe fluxuri de date)
  - Cheia (nu cea expandată) poate fi stocată de către aplicație sau introdusă de fiecare dată când se aplică cifrul – costul de extindere a cheii
- Tradeoff – lipsa totală de aleatorie în biții cheii extinse

Diversificarea cheilor

- Utilă în cazul încercărilor de ghicirea a biților cheii în orice atac

  - AES: biții cheii extinse sunt XOR ai altor doi biți
  - MISTY1, Camellia: aceiași biți ai cheii extinse sunt utilizați în locuri multiple
  - RC6 : mai dificil – nu sunt expresii evidente relaționate biților cheii extinse

Chei relaționate (Related Keys)

- Atacatorul specifică relația dintre două chei, dar nu între cheile actuale
- Obține perechea de text clar – text cifrat pentru fiecare cheie
- Încearcă să determine cheile de rundă
- Exemplu: *Atacul slide*
- AES poate avea două chei K1, K2 astfel încât K2 is K1 “alunecată” într-o singură rundă. De ex. biții cheii extinse din runda 1 cu utilizarea K1 = biții din runda 2 ai chei extinse din K2
  - Boxele S și XOR cu pas constant previn “sliding”-ul pentru mai mult de o rundă

Alte atacuri

- Atacul adaptabil în blocuri
- Cryptanaliza non-liniară (algebrică)
- Square Attack - numit pentru atacul la cifrul bloc Square - un predecesor al lui Rijndael

Cuprins

- Prezentare generală
- **Criptanaliza cifrurilor bloc:**
  - Criptanaliza Liniară
  - Criptanaliza Diferențială
  - Alte tipuri de atac
  - **Analiza Statistică**
- Criptanaliza cifrurilor fluide
- Cazul general
  - Side Channel Attacks

Teste Statistice

- Au fost efectuate 16 teste pe 8 seturi de date pentru fiecare cifru.
  - **Nu dovedește că cifrul e securizat**
  - **Eșecul testului indică o slăbiciune**
  - NIST AES finaliștii competiției: > 96.33% de cazuri au trecut testele
- Ce înseamnă dacă cifrul ratează testul?
  - Există o relație între P,C,K – însă nu se cunoaște exact care e ea
  - De exemplu, cheia cu 1 în bitul j poate fi predispusă să producă text cifrat cu mai multe 0 decât 1.

Teste Statistice

- **Frecvența (Monobit):** Propoziții alcătuite din 0 și 1 în secvența de biți foarte aproape de ½ .
- **Frecvența într-un bloc:** Test de frecvență aplicat la blocuri de lungime fixă într-o secvență de biți.
- **Runs:** Numărul de secvențe formate doar din 0 sau 1 într-o succesiune de biți este determinat.
- **Longest Run of Ones within a Block:** Cea mai mare secvență de 1 într-un bloc e cunoscută.
- **Binary Matrix Rank:** matricele 32x32 sunt alcătuite din secvența de biți și sunt calculate rangurile lor. Determină dacă există o dependență liniară în segmentele de biți de lungime fixată într-o succesiune de biți.
- **Transformarea discretă Fourier :** determină dacă există șabloane repetitive într-o succesiune de biți.
- **Etc.**

Cuprins

- Prezentare generală
- Criptanaliza cifrurilor bloc:
  - Criptanaliza Liniară
  - Criptanaliza Diferențială
  - Alte tipuri de atac
  - Analiza Statistică
- **Criptanaliza cifrurilor fluide**
- Cazul general
  - Side Channel Attacks

Criptanaliza cifrurilor fluide

- Un singur LSRF poate fi spart ușor: algoritmul **Berlekamp-Massey**
- **Atacul de corelație**
  - Generatorul G al fluxului de chei constă dintr-o mulțime de LFSR-uri și o funcție non-liniară
  - Adversarul cunoaște G și unele segmente ale fluxului
  - Încearcă să relaționeze ieșirea cheii cu ieșirea unui sau mai mulți LFSR
  - Căutarea exhaustivă printre stările posibile ale LFSR în G
    - *n* LFSR-uri,
    - 2^i -1 stări inițiale posibile pentru LFSR-ul *i*
    - Π (2^i -1), *i* = 1 to *n*
  - Dacă fiecare LSFR este corelat cu fluxul cheii:
    - Σ (2^i -1) i = 1 to n (ghicește primul LSRF și îl păstrează constant, ghicește al doilea LSFR …)

Criptanaliza cifrurilor fluide

- Informația disponibilă atacatorului – aceleași idei ca la cifrurile bloc:
  - Doar text cifrat
  - Perechi de text clar - text cifrat
    - text clar cunoscut din informația standard a antetelor protocoalelor de rețea, formatele fișierelor…
  - Versiuni alese, adaptate

Criptanaliza cifrurilor fluide

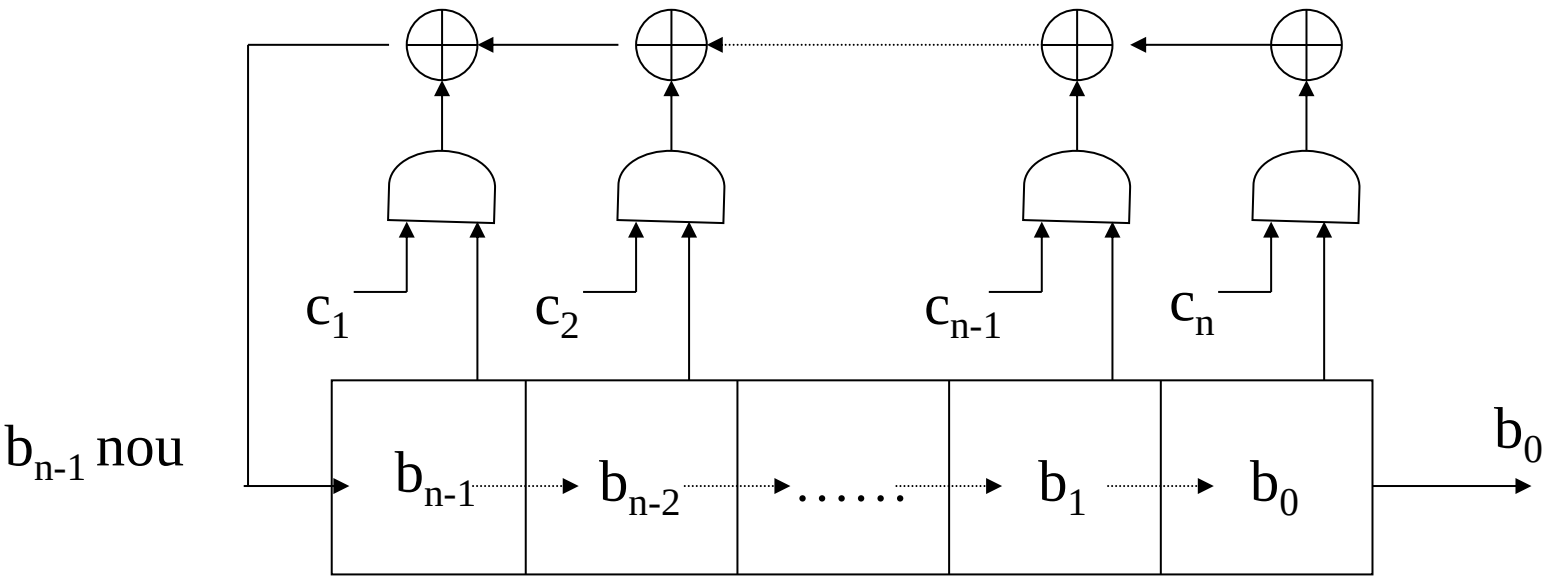
- Observarea atacurilor
  - Distingerea fluxului cheii de biții aleatori
  - Teste statistice
  - Nu se presupune că cifrul poate fi spart în practică
- Analiza canalului lateral
  - Analiza temporală
  - Defecțiune diferențială
  - Memorie – sunt disponibili biți ai cheii și starea internă

Algoritmul Berlekamp-Massey

De ce nu e suficient doar LFSR

- Având o secvență de biți $s^n = s_0 s_1 s_2 \ldots s_{n-1,}$, găsește LFSR-ul corespunzător.
- Inițializează ghicirea LFSR.
- Parcurge s^n, compară cu ieșirea următoare din LFSR.
  - Dacă al (N+1) termen al LFSR = s_N,LFSR generează s_N
  - Altfel modifică LFSR
  - $O(n^2)$

Reprezentarea polinomială LFRS



$$1 + c_1x + c_2x^2 + \ldots\ldots\ldots\ldots\ldots\ldots\ c_{n-1}x^{n-1} + c_nx^n$$

folosește b_{n-j} ca valoarea x^j

$$b_{n-1} = c_1b_{n-1} + c_2b_{n-2} + \ldots\ldots\ c_{n-1}b_1 + c_nb_0 \text{ nou}$$

Algoritmul Berlekamp-Massey

Input: $s^n = s_0 s_1 s_2 \ldots s_{n-1}$

```
C(x) = 1; L = 0; m = -1; B(x) = 1; N = 0; // initialize
while (N < n) {
        d = (s_N + ∑_{i=1,Li} c_i s_{N-i}) mod 2;          // next discrepancy

        if (d == 1) {                                      // update LFSR
             T(x) = C(x); C(x) = C(x) + B(x)*x^{N-m};
              if L ≤ N/2 {
                   L = N+1-L; m = N; B(x) = T(x);
              }
        }
        ++N;
}
return(L,C);
```

C(X) = reprezentarea polinomială a LFSR

C_i = coeficienții lui C.

L = complexitatea liniară a LFSR

Berlekamp-Massey Exemplu

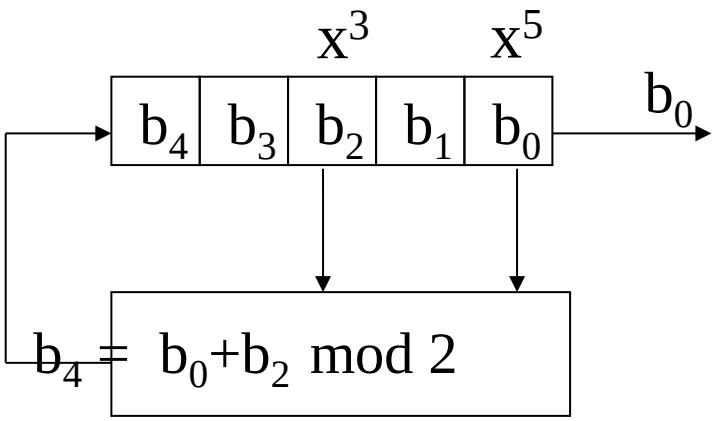
E dat:

$s^n = 001101110$, n = 9

Ieșire:

Polynomial: $1+x^3+x^5$

Poate determina:



Initial state 00110
01100: 0 ⊕ 1, move in 1, output 0
10110: 0 ⊕ 1, move in 1, output 0
11011: 1 ⊕ 0, move in 1, output 1
11101: 1 ⊕ 1, move in 0, output 1
01110: 0 ⊕ 1, move in 1, output 0
10111: etc …

Valori la sfârșitul fiecărei iterații din while

s_n	d	T(x)	C(x)	L	m	B(x)	N
-	-	-	1	0	-1	1	0
0	0	-	1	0	-1	1	1
0	0	-	1	0	-1	1	2
1	1	1	$1+x^3$	3	2	1	3
1	1	$1+x^3$	$1+x+x^3$	3	2	1	4
0	1	$1+x+x^3$	$1+x+x^2+x^3$	3	2	1	5
1	1	$1+x+x^2+x^3$	$1+x+x^2$	3	2	1	6
1	0	$1+x+x^2+x^3$	$1+x+x^2$	3	2	1	7
1	1	$1+x+x^2$	$1+x+x^2+x^5$	5	7	$1+x+x^2$	8
0	1	$1+x+x^2+x^5$	$\mathbf{1+x^3+x^5}$	5	7	$1+x+x^2$	9

N = 4, L = 3: $C(x) = c_1s_2 + c_2s_1 + c_3s_0 = 1*1+0*0+1*0 = 1$

$s_4 = 0 \neq C(x)$, so set d = 1

Criptanaliza cifrurilor fluide

- Atacuri asupra cifrurilor non-FSR
  - Depinde de componente
  - Analiza funcțiilor pentru relațiile dintre
    - Fluxul cheilor și cheie sau starea inițială
    - Biții fluxului de chei
- Ghicirea submulțimii de biți necunoscuți utilizați intern pentru determinarea stării

Cuprins

- **Prezentare generală**
- Criptanaliza cifrurilor bloc:
  - Criptanaliza Liniară
  - Criptanaliza Diferențială
  - Alte tipuri de atac
  - Analiza Statistică
- Criptanaliza cifrurilor fluide
- **Cazul general**
  - **Side Channel Attacks**

Analiza canalelor laterale

- **Timp**
  - Depinde oare numărul de cicluri CPU de valorile exacte utilizate în operație? ex. exponentul RSA
  - Memory access – do exact values impact tables used, time to read from a table and/or number of memory accesses? ex. AES using tables of 32-bit values
  - Accesul la memorie – influențează oare valorile exacte tabelele utilizate timpul de citire din tabele și/sau numărul de accesări de memorie? ex. utilizarea de către AES a tabelelor cu valori de 32 biți
- **Acustica**
  - Impactată de operațiuni sau valorile exacte folosite?
- **Memoria**
  - Pot fi oare citite valorile intermediare din memorie printr-un alt proces?

Timing – Exemplu

k: listă din *n* biți ai cheii

d: date de 16 octeți

Fie că criptarea e din *n* runde

n = 16;

d = text clar;

for (i=0; i < n; ++i) {

  d = f(d,k[i]); // acționează cu k asupra datelor, însă

                  // timpul să nu depindă de *k*

    d[i] = d[i] $^{int(k[i])}$ mod 256;

                  // modifică un octet, timpul depinde de k

}

Timing – Exemplu

Dar dacă utilizăm căutarea cu ajutorul unui tabel?

table(a,b): funcția returnează tabelul a, intrarea b

```
d = text clar;
x = 0;
for (i=0; i < n; ++i) {
                                    // acționează cu k asupra datelor, însă
                                    // timpul să nu depindă de k
      d = f(d,k[i]);
   // căutarea memoriei - a fost deja stocată în memoria cache?
   // (k[i] același ca octetul cheii precedente)
      x= table(k[i], d[i]);
}
```

Timing and Power Analysis

- P. Kocher, Timing Attacks on Implementations of RSA, DH, DSS and Other Systems, Crypto 1996.
- A. Shamir and E. Tromer, Acoustic Cryptanalysis on Nosy People and Noisy Machines, 2004 presentation
- J. Kelsey, B. Schneier, D. Wagner and C. Hall, Side Channel Cryptanalysis of Product Ciphers. Journal of Computer Science, 8(2-3),pages 141-158, 2000. (DES, IDEA, RC5 used in examples)
- Companies, ex. Riscure, sell software for performing timing analysis on smart cards.
- etc.

Defecțiune diferențială

- Inducere de defecțiuni în dispozitiv
- Observarea ieșirilor fără defecțiune și cu defecțiune
  - Exemplu: radiații
- Defecțiunea exactă introdusă poate fi necunoscută
- Se presupune că dispozitivul poate fi manipulat – chips-urile pot fi proiectate astfel încât să nu mai funcționeze dacă sunt manipulate
- Mai puțin practice decât alte atacuri
- Public Key Ciphers: Boneh, Denillo and Lipton, On the Importance of Checking Cryptographic Protocols for Faults. Eurocrypt 1997.
- Private Key Ciphers: Biham and Shamir, Differential Fault Analysis of Secret Key Cryptosystems, Technion CS Technical Report 1997.

Memorie

- Procesul care accesează aceeași memorie (memoria cache) folosită și de cifru poate da informații
- Folosit pentru a ataca AES (OS specific, implementare)
  Dacă atacatorul poate efectua atacul, există probleme mai mari de securitate cu privire la sistem.

- Osvik, Shamir, Tromer, Cache Attacks and Countermeasures, the Case of AES. CT-RSA 2006.

Concluzii

- Există numeroase tehnici de analiză, **analiza diferențială** și **liniară** sunt cele mai puternice
- Cele mai multe dintre primitivele criptografice au fost sparte, inclusiv **2 cifruri bloc standard** și **1 funcție hash standard**

  => criptanaliștii și-au făcut treaba bine
- **Este mai dificil** să proiectezi o primitivă criptografică care să fie atât eficace cât și sigură decât să spargi una

Întrebări???