

1. Kryptografia



Kryptografia

Kryptografia je štúdium matematických techník na ochranu a utajenie informácií.

Niekedy sa používa aj termín Kryptológia, ktorá sa delí na

- Kryptografiu – vynachádzanie šifrovacích systémov a
- Kryptoanalýzu – študujúcu útoky voči šifrovacím systémom.

Úlohy kryptografie

- Utajenie informácie
- Zaistenie integrity údajov – zaistenie proti zmene správy
- Autentifikácia – zaistenie, že správa pochádza od určitého pôvodcu
- Identifikácia – zaistenie, že komunikujem stým s kým chcem
- Neodškripteľný digitálny podpis
- Steganografia – ukrytie správy v inom údajovom súbore

Kryptografické útoky



Kryptografické útoky

Útok na kryptografický systém je postup, ktorý odhalí priameho text (alebo aspoň jeho časť) alebo dokonca zistí šifrovací kľúč.

Typy kryptografických útokov

- Brute force attack
- Ciphertext only attack
- Known plaintext attack
- Chosen plaintext attack
- Chosen ciphertext attack
- Dictionary attack
- Rubber hose attack

Kryptosystém:

Kryptosystém je usporiadaná štvorica $(\mathcal{K}, \mathcal{M}, \mathcal{C}, \mathcal{T})$ kde

- \mathcal{K} je množina kľúčov
- \mathcal{M} je množina priamych textov
- \mathcal{C} je množina zašifrovaných textov
- \mathcal{T} je zobrazenie $\mathcal{T} : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$, ktoré každej dvojici $K \in \mathcal{K}$, $M \in \mathcal{M}$ priradí zašifrovanú správu $C \in \mathcal{C}$ a také, že

V tomto systéme $\mathcal{K} = \{A, B, \dots, Z\}$, kľúč $k = A \equiv 0$ je nepoužiteľný.

\mathcal{M} je množina všetkých zrozumiteľných slovenských textov.

2. Afinna sifra:

Základem affinní šifry je následující transformace:

$$C_i = a \cdot T_i + b \pmod{m}$$

C_i - i-té písmeno šifrovaného textu

T_i - i-té písmeno otevřeného textu

a - parametr a, $\gcd(a,m) = 1$

b - parametr b

m - modulo (jako modulo obvykle volíme prvočíslo, aby bylo předem jasné, že $\gcd(a, m) = 1$, a zároveň abychom útočníkovi nezjednodušovali práci (pokud modulo není prvočíslo, tak je méně možností, jak se text dá šifrovat - je tedy snazší šifru prolomit)).

Zašifrujte daný otevřený text pomocí affinní transformace.

Otevřený text: THEINITIAL

$a = 5$

$b = 9$

$m = 26$

$T \Rightarrow 19 \Rightarrow 5*19 + 9 \pmod{26} = 0 \Rightarrow A$
 $H \Rightarrow 7 \Rightarrow 5*7 + 9 \pmod{26} = 18 \Rightarrow S$
 $E \Rightarrow 4 \Rightarrow 5*4 + 9 \pmod{26} = 3 \Rightarrow D$
 $I \Rightarrow 8 \Rightarrow 5*8 + 9 \pmod{26} = 23 \Rightarrow X$
 $N \Rightarrow 13 \Rightarrow 5*13 + 9 \pmod{26} = 22 \Rightarrow W$
 $I \Rightarrow 8 \Rightarrow 5*8 + 9 \pmod{26} = 23 \Rightarrow X$
 $T \Rightarrow 19 \Rightarrow 5*19 + 9 \pmod{26} = 0 \Rightarrow A$
 $I \Rightarrow 8 \Rightarrow 5*8 + 9 \pmod{26} = 23 \Rightarrow X$
 $A \Rightarrow 0 \Rightarrow 5*0 + 9 \pmod{26} = 9 \Rightarrow J$
 $L \Rightarrow 11 \Rightarrow 5*11 + 9 \pmod{26} = 12 \Rightarrow M$

Dešifrování

Při dešifrování musíme eliminovat transformaci vzniklou šifrováním. Provedeme proto takovou transformaci:

$$T_i = (C_i - b) \cdot a^{-1} \pmod{m}$$

a^{-1} - multiplikativní inverze a v \mathbb{Z}_m

Příklad

Dešifrujte šifrovaný text pomocí affinní transformace.

Šifrovaný text: ASDWXWAXJM

a = 5

b = 9

m = 26

Rozšířeným Euklidovým algoritmem zjistíme:

$$a^{-1} = 21$$

```
A => 0 => (0 - 9) * 21 mod(26) = 19 => T
S => 18 => (18 - 9) * 21 mod(26) = 7 => H
D => 3 => (3 - 9) * 21 mod(26) = 4 => E
X => 23 => (23 - 9) * 21 mod(26) = 8 => I
W => 22 => (22 - 9) * 21 mod(26) = 13 => N
X => 23 => (23 - 9) * 21 mod(26) = 8 => I
A => 0 => (0 - 9) * 21 mod(26) = 19 => T
X => 23 => (23 - 9) * 21 mod(26) = 8 => I
J => 9 => (23 - 9) * 21 mod(26) = 0 => A
M => 12 => (12 - 9) * 21 mod(26) = 11 => L
```

Útoky na affinní šifru:

Protože je zde již daleko více kombinací, tak by byl útok hrubou silou poměrně neefektivní (nic méně pořád proveditelný). Skutečnou slabinou affinní šifry je frekvenční analýza.

Klúč – dvojice prvkov k_1, k_2 okruhu \mathbb{Z}_{26} taká, že existuje prvek $k_1^{-1} \in \mathbb{Z}$ inverzný ku k_1 (t.j. $k_1 \otimes k_1^{-1} = 1 \equiv B$).

šifrovanie: $y = E_{k_1, k_2}(x) = (x \otimes k_1) \oplus k_2$

dešifrovanie: $x = D_{k_1, k_2}(y) = (y \ominus k_2) \otimes k_1^{-1}$

Množina klúčov \mathcal{M} – množina všetkých usporiadaných dvojíc (k_1, k_2) taká, že existuje $k_1^{-1} \in \mathbb{Z}$.

$k_1 = 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25$ – 12 možností

$k_2 = 0, 1, 2, \dots, 24, 25$ – 26 možností

Slabý klúč $(k_1, k_2) = (1, 0)$.

Množina použiteľných klúčov obsahuje $12 \cdot 26 - 1 = 311$ prvkov.

Kryptoanalýza afinnej šifry

„Ciphertext only attack“ hrubou silou vyžaduje vyskúšať 311 kľúčov.

Known plaintext attack:

Uhádнемe že $E_{k_1, k_2}(C) = P$, $E_{k_1, k_2}(F) = H$,
t.j. $E_{k_1, k_2}(2) = 15$, $E_{k_1, k_2}(5) = 7$

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	15	18	19	20	21	22	23	24	25

$$k_1 \otimes 2 \oplus k_2 = 15 \quad (2)$$

$$k_1 \otimes 5 \oplus k_2 = 7 \quad (3)$$

Odčítaním rovnice (2) od (3)

$$k_1 \otimes 3 = -8 \pmod{26} = 18 \quad /* 9 \equiv 3^{-1} \quad (4)$$

$$k_1 = 18 * 9 \pmod{26} = 162 \pmod{26} = 6 \quad (5)$$

Dosadením za k_1 do (2)

$$(6 \otimes 2) \oplus k_2 = 15 \quad (6)$$

$$k_2 = 15 \ominus 12 = 3 \quad (7)$$

3. Všeobecna monoalfabeticka sifra

Všeobecná monoalfabetická šifra

π – ľubovoľná permutácia abecedy \mathbb{Z}_{26}

π^{-1} – inverzná permutácia k permutácii π

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	P	Q	V	R	M	O	S	H	I	E	F	G	N	J	K	Y	Z	A	B	L	T	U	W	X	C

Šifrujeme znak po znaku predpisom $y = E_\pi(x) = \pi(x)$

Dešifrujeme znak po znaku predpisom $x = D_\pi(y) = \pi^{-1}(y)$

Priestor kľúčov \mathcal{K} je obrovský $|\mathcal{K}| = 26! \approx 10^{27}$

Pokus o matematickú formuláciu problému kryptoanalýzy

- p_{ij} pravdepodobnosť výskytu dvojice znakov $a_i a_j$ v jazyku.
- r_{pq} relatívna početnosť znakov $a_p a_q$ v zašifrovanom teste
- $x_{ip} = \begin{cases} 1 & \text{ak } a_i \text{ bolo zašifrované na } a_p \\ 0 & \text{inak} \end{cases}$

Minimalizovať

$$\sum_{i=1}^n \sum_{j=1}^n \sum_{p=1}^n \sum_{q=1}^n x_{ip} x_{jq} (p_{ij} - r_{pq})^2$$

za podmienok

$$\sum_{i=1}^n x_{ip} = 1 \quad \text{pre } p = 1, 2, \dots, n$$

$$\sum_{p=1}^n x_{ip} = 1 \quad \text{pre } i = 1, 2, \dots, n$$

$$x_{ip} \in \{0, 1\}$$

Slovenská Univerzita v Bratislavе / Department of Mathematics and Cryptology
Kryptologické laboratórium

Najčastejšie znaky slovenskej abecedy sú medzera a

A, O, E, I, N, T, S

Postup pri kryptoanalýze (Grošek, Porubský):

- Ak bola použitá taká permutácia, ktorý zachováva medzeru, treba analyzovať najskôr kratšie slová, ktoré poskytujú menší priestor pre kombinácie
- Hľadať charakteristické kombinácie znakov (trojice, štvorice). Tie sa najčastejšie vyskytujú na začiatkoch a na koncoch slov.
- Odhadnúť na základe „postranných informácií“, ktoré slová by sa mohli v teste vyskytnúť
- Odhadnúť, ktoré znaky sú samohlásky a ktoré spoluohlásky

Vytipovanie samohlások takto:

- samohlásky sú často obkolesené spoluohláskami
- spoluohláska sú často obkolesené samohláskami
- písmená s malým počtom rôznych susedov sú často spoluohláska a títo susedia sú často samohlásky
- ak sa dvojica XY vyskytuje často aj v opačnom poradí YX jedno z nich je samohláska
- skoro v každom normálnom slove je samohláska

Pri kryptoanalýze všeobecnej monoalfabetickej šifry využívame skutočnosť, že množina priamych textov \mathcal{M} je množinou výstupov z konkrétneho zdroja informácie.

Ten je charakterizovaný súborom pravdepodobností $P(x_1, x_2, \dots, x_n)$ vujadrujúcich že zdroj v okmihoch $1, 2, \dots, n$ vyšle postupne znaky x_1, x_2, \dots, x_n

$$P() = 1 \quad (6)$$

$$\sum_{x_1} \sum_{x_2} \cdots \sum_{x_n} P(x_1, x_2, \dots, x_n) = 1 \quad (7)$$

$$P(x_1, x_2, \dots, x_n) = \sum_{y_1} \sum_{y_2} \cdots \sum_{y_m} P(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m) \quad (8)$$

Pravdepodobnosť vyslania reťazca x_1, x_2, \dots, x_m od času n

$$P_n(x_1, x_2, \dots, x_m) = \sum_{y_1} \sum_{y_2} \cdots \sum_{y_{n-1}} P(y_1, y_2, \dots, y_{n-1}, x_1, x_2, \dots, x_m) \quad (9)$$

Stacionárny zdroj – $P_n(x_1, x_2, \dots, x_m)$ nezávisí na n

Nezávislý zdroj – vyslanie ľubovoľných dvoch slov v neprekryvajúcich sa časových intervaloch sú nezávislé javy.

Pre kryptoanalýzu všeobecnej monoalfabetickej šifry sa využívajú hlavne pravdepodobnosti $P(x_1)$, $P(x_1, x_2)$, $P(x_1, x_2, x_3)$.

Informácia jedného znaku x_i abecedy zdroja (SHANNON-HARTLEY)

$$I(x_i) = -\log_2 P(x_i) \quad (10)$$

Stredná informácia na jedno písmeno je

$$H_1 = \sum_{x_1} -P(x_1) \log_2 P(x_1) \quad (11)$$

Pri sledovaní dvojíc za sebou idúcich znakov je stredná informácia na dvojicu

$$H_2 = \sum_{x_1} \sum_{x_2} -P(x_1, x_2) \log_2 P(x_1, x_2) \quad (12)$$

Pri sledovaní n -tíc dvojíc za sebou idúcich znakov je stredná informácia na n -ticu

$$H_n = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_n} -P(x_1, x_2, \dots, x_n) \log_2 P(x_1, x_2, \dots, x_n) \quad (13)$$

Stredná informácia na jeden znak je

$$H = \frac{1}{n} H_n \quad (14)$$

Limita tejto hodnoty pre $n \rightarrow \infty$ je entropia zdroja

$$\text{Entropia zdroja } \mathcal{H} = \lim_{n \rightarrow \infty} \frac{1}{n} H_n \quad (15)$$

Náš odhad: $\mathcal{H}(\text{slovenského jazyka}) = 1,57 \text{[bit/znak]}$, $\kappa = 0,0553$.

4. Vigenerovska



Polyalfabetické šifry.

Nevýhoda monoalfabetických šifier – relatívna početnosť zašifrovaného písmena v zašifrovanom teste závisí na pravdepodobnosti výskytu tohto písmena v použitom jazyku.

Nová myšlienka – síce šifrovať znak po znaku, ale každý znak priameho textu inak.

Teda zašifrovaný text $y_1y_2 \dots y_n$ dostaneme z priameho textu $x_1x_2 \dots x_n$ takto:

$$\begin{aligned} y_1 &= E_{K_1}(x_1) \\ y_2 &= E_{K_2}(x_2) \\ &\vdots \\ y_n &= E_{K_n}(x_n) \end{aligned}$$



Vigenèrovské šifry

Najjednoduchší spôsob je nasledovný: zvolí sa kľúč – napr. „HESLO“ a potom zašifrovaný text $y_1y_2 \dots y_n$ dostaneme z priameho textu $x_1x_2 \dots x_n$ takto:

$$\begin{aligned}
 y_1 &= x_1 \oplus H \\
 y_2 &= x_2 \oplus E \\
 y_3 &= x_3 \oplus S \\
 y_4 &= x_4 \oplus L \\
 y_5 &= x_1 \oplus O \\
 y_6 &= x_6 \oplus H \\
 y_7 &= x_7 \oplus E \\
 y_8 &= x_8 \oplus S \\
 y_9 &= x_9 \oplus L
 \end{aligned}$$



Kasiského test na zistenie dĺžky kľúča

Odevad	Vložit	Rozložené stránky	Význam	Družstvo	Poslat	Zobrazit	Výměna	PDF
YZ	▼	1 / 25						

Hľadajú sa opakované výskyty toho istého reťazca v zašifrovanom teste.
Je nádej, že vzdialenosť výskytov je násobkom dĺžky kľúča.

Dĺžka kľúča je pravdepodobne najväčším spoločným deliteľom vzdialenosí rovnakých výskytov

5. Index koincidencie

Index koincidencie

Ak by všetky znaky abecedy $A = \{a_1, a_2, \dots, a_q\}$ s q znakmi mali rovnakú pravdepodobnosť, potom $p(a_i) = \frac{1}{q}$.

Hľadáme spôsob, ako kvantifikovať mieru nerovnomernosti pravdepodobností.

$$\sum_{i=1}^q (p(a_i) - \frac{1}{q})^2$$

Index koincidencie (2)

Definícia

Číslo $\sum_{i=1}^q p(a_i)^2$ sa nazýva **index koincidencie**.

Čím je index koincidencii väčší než $\frac{1}{q}$, tým viac sa rozdelenie pravdepodobnosti viac líši od rovnomerného rozdelenia.

Pre slovenskú telegrafnú abecedu bez medzery je index koincidencie asi $0,06027$, pričom $\frac{1}{q} = 0,03846$.

Pre slovenskú abecedu s diakritikou, číslami a interpunkčnými znakmi v kódovaní používanom v počítačoch sme odhadli index koincidencie na $0,0553$.

Ďalší význam indexu koincidencie:

Pravdepodobnosť, že dva náhodne vybrané znaky z jazyka (resp. zo zdroja informácie) sa budú oba rovnať a_i je $p(a_i)$.

Jav, že dva náhodne vybrané znaky budú rovnaké je zjednotením nasledujúcich disjunktívnych javov

- že oba znaky sa budú rovnať a_1 – pravdepodobnosť $p(a_1)^2$
- že oba znaky sa budú rovnať a_2 – pravdepodobnosť $p(a_2)^2$
-
- že oba znaky sa budú rovnať a_q – pravdepodobnosť $p(a_q)^2$

Pravdepodobnosť javu, že dva náhodne vybrané znaky budú rovnaké, je súčet pravdepodobností týchto javov, a teda $\sum_{i=1}^q p(a_i)^2$

Zistovanie dĺžky kľúča metódou koincidencie

Majme dva priame texty

$$\mathbf{r} = r_1 r_2 \dots r_n, \mathbf{s} = s_1 s_2 \dots s_n$$

Pravdepodobnosť, že $r_i = s_i$ je index koincidencie slov. jazyka κ .

Nech teto texty sú zašifrované znak po znaku rovnakým kľúčom.

Príslušné zašifrované texty sú "

$$\bar{\mathbf{r}} = T_1(r_1) T_2(r_2) \dots T_n(r_n),$$

$$\bar{\mathbf{s}} = T_1(s_1) T_2(s_2) \dots T_n(s_n).$$

Pravdepodobnosť javu, že $T_i(r_i) = T_i(s_i)$, je rovnaká ako pravdepodobnosť javu, že $r_i = s_i$, lebo $T_i(r_i) = T_i(s_i)$ práve vtedy, keď $r_i = s_i$. Teda

$$P(T_i(r_i) = T_i(s_i)) = P(r_i = s_i) = \kappa$$

Ak sledujeme počet zhôd na rovnakých miestach zašifrovaného a posunutého zašifrovaného textu, počet zhôd by mal nápadne stúpnúť, ak je posun o násobok dĺžky kľúča.

6. Hillovska sifra

Majme priamy text v q -znakovnej abecede $A = \{a_0, a_1 \dots, a_{q-1}\}$.

Prvky abecedy A stotožníme s prvkami okruhu \mathbb{Z}_q .

Na abecede A tak máme operácie \oplus a \otimes .

Ak je q prvočíslo, je \mathbb{Z}_q poľom a ku každému $a \in A$ $a \neq 0$ existuje $a^{-1} \in A$ také, že $a \otimes a^{-1} = 1$.

Ak q nie je prvočíslo, potom inverzné prvky existujú len k tým znakom, ktoré nie sú súdeliteľné s q .

Preferujeme teda q prvočíslo.

Existujú konečné telesá s $q = p^n$ prvkami, kde p je prvočíslo.

Sú to tzv. Galoisove polia, značia sa $GF(p^n)$.

Na abecedách, ktoré nemajú prvočíselný počet prvkov alebo počet prvkov rovnajúci sa prirodzenej mocnine prvočísla, nemožno zaviesť operácie \oplus a \otimes tak, aby štruktúra (A, \oplus, \otimes) bola poľom.

 Hillovská šifra je polyalfabetická šifra šifrujúca naraz celý blok priameho textu dĺžky n .

$$\underbrace{x_1 x_2 \dots x_n}_{x_1} \underbrace{x_{21} x_{22} \dots x_{2n}}_{x_2} \dots \dots \dots \underbrace{x_{m1} x_{m2} \dots x_{mn}}_{x_m} \quad (1)$$

Kľúčom je štvorcová matica typu $n \times n$ taká že k nej existuje inverzná matica \mathbf{K}^{-1} .

$$\mathbf{K} = \begin{pmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{pmatrix} \quad (2)$$

 Sifrovanie

$$\mathbf{y} = \mathbf{K}\mathbf{x} \quad \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} = \begin{pmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} \quad (3)$$

$$\begin{aligned} y_1 &= k_{11}x_1 + k_{12}x_2 + \dots + k_{1n}x_n \\ y_2 &= k_{21}x_1 + k_{22}x_2 + \dots + k_{2n}x_n \\ &\dots \\ y_n &= k_{n1}x_1 + k_{n2}x_2 + \dots + k_{nn}x_n \end{aligned}$$

Dešifrovanie

$$\mathbf{x} = \mathbf{K}^{-1}\mathbf{y}$$

Dešifrovanie je korektné, lebo

$$\mathbf{K}^{-1}\mathbf{y} = \mathbf{K}^{-1} \cdot (\mathbf{K} \cdot \mathbf{x}) = (\mathbf{K}^{-1} \cdot \mathbf{K}) \cdot \mathbf{x} = \mathbf{I} \cdot \mathbf{x} = \mathbf{x} \quad (4)$$

Príklad: Abeceda

A, B, C, D, E, F, G, H, I, J, K, L, M, N,
O, P, Q, R, S, T, U, V, W, X, Y, Z} $\equiv \mathbb{Z}_{26}$.

VSTUPNA MATICA

$$\mathbf{K} = \begin{pmatrix} 17 & 4 & 3 & 9 \\ 1 & 13 & 21 & 16 \\ 10 & 12 & 5 & 9 \\ 13 & 6 & 3 & 12 \end{pmatrix}$$

Či je regulárna, zistíme tak, že v niektorom tabuľkovom procesore vypočítame jej determinant. Tu je $\det \mathbf{K} = -11305$ a $\text{mod}(-11305, 26) = 5$ je číslo, ku ktorému existuje v \mathbb{Z}_{26} inverzny prvok – totož 21.

Mýpočet inverznej matice. Ekvivalentnými úpravami matíc upravime maticu $(\mathbf{K}|\mathbf{I})$, kde \mathbf{I} je jednotková štvorcová matica, na tvar $(\mathbf{I}|\mathbf{K}^{-1})$.

$$\left(\begin{array}{cccc|cccc} 17 & 4 & 3 & 9 & 1 & 0 & 0 & 0 \\ 1 & 13 & 21 & 16 & 0 & 1 & 0 & 0 \\ 10 & 12 & 5 & 9 & 0 & 0 & 1 & 0 \\ 13 & 6 & 3 & 12 & 0 & 0 & 0 & 1 \end{array} \right)$$
$$\left(\begin{array}{cccc|cccc} 17 & 4 & 3 & 9 & 1 & 0 & 0 & 0 \\ 0 & 25 & 4 & 17 & 3 & 1 & 0 & 0 \\ 0 & 2 & 17 & 19 & 4 & 0 & 1 & 0 \\ 0 & 6 & 16 & 25 & 13 & 0 & 0 & 1 \end{array} \right)$$

Known plaintext attack proti hillevskej šifre. Predpokladajme, že poznáme n dvojíc priameho textu a príslušného ciphertextu.

$$\mathbf{y}_1 = \mathbf{Kx}_1, \mathbf{y}_2 = \mathbf{Kx}_2, \dots, \mathbf{y}_n = \mathbf{Kx}_n \quad (5)$$

Zostrojme štvorcové matice typu $n \times n$ \mathbf{X}, \mathbf{Y} , ktorých stĺpce budú tvorené stĺpcovými vektormi $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$, resp. $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n$, t.j.:

$$\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n), \quad \mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n).$$

Potom vzťahy (5) možno zapísať v maticovom tvare

$$\mathbf{Y} = \mathbf{K} \cdot \mathbf{X} \quad (6)$$

Vynásobením rovnice (6) maticou \mathbf{X}^{-1} sprava (za predpokladu, že \mathbf{X}^{-1} existuje) dostávame:

$$\mathbf{Y} \cdot \mathbf{X}^{-1} = (\mathbf{K} \cdot \mathbf{X}) \cdot \mathbf{X}^{-1} = \mathbf{K} \cdot (\mathbf{X} \cdot \mathbf{X}^{-1}) = \mathbf{K} \cdot \mathbf{I} = \mathbf{K}$$

7. Transpozicna – permutacna sifra

Principem transpoziční šifry je změna pořadí jednotlivých znaků textu (permutace) na základě předem domluveného systému. Výhodou tohoto postupu je jeho jednoduchost – může ho použít klidně dítě, protože není obvykle třeba jakákoli znalost matematiky. Nevýhodou je jeho více či méně snadná analýza (dle pravidla transformace), další významnou nevýhodou je snadné odhalení jazyka otevřeného textu pomocí frekvenční analýzy (znaky zůstavají totožné, mění se jen jejich pořadí).

Použitím transpoziční šifry dochází k difúzi – rozprostření redundance jazyka napříč zprávou.

Příklad

OT: SKAKALPESPRESOVESPRESZELENOULOUKU
Pravidlo: Zapišme text do tabulky s pěti sloupcí (padding: „X“), tabulku překlopme podle diagonály a čtěme po řádcích.

ŠT: SLRESNUKPESZOKAESPEUUSKORLLXAPVEEOX

SKAKA	LPESP	RESOV	ESPRE	SZELE	NOULO	UKUXX	SLRESNU	KPESZOK	AESPEUU	KSORLLX	APVEEOX
-------	-------	-------	-------	-------	-------	-------	---------	---------	---------	---------	---------

Transpoziční šifra

Dešifrování

Protože měl obdélník 5 sloupců, tak má po transformaci 5 řádek. Zapišme tedy došly ŠT do obdélníku s pěti řádky – máme 35 znaků, tzn. 7 sloupců. Tímto jsme dosáhli toho, že máme před sebou obdélník, který měl šifrující na pravé straně od šipky (vizte obrázek). Bud' ho ještě otočíme nebo rovnou čtěme po sloupcích.

Útok pomocí vycpávky

Nejfektivnější je zaútočit na tento typ kryptosystému pomocí paddingu (vycpávky). Protože víme, že byly vloženy znaky X na konec textu (aby se text zaroval na plný obdélník) a z ŠT víme, že po transformaci je padding od sebe vzdálen 7 pozic a má celkem 35 znaků, tak je zřejmé, že měl původní obdélník původně 7 řádek a 5 sloupců – můžeme dešifrovat.

Útok faktorizací

Protože se každé číslo dá rozložit pouze na omezený počet dělitelů a známe počet znaků ŠT, tak také můžeme snadno prozkoušet všechny kombinace počtu řádků a sloupců.

8. One time Pad + útok (Vernamova šifra)

Vernamova šifra nebo také **jednorázová tabulková šifra** (anglicky *one-time pad*) je jednoduchý šifrovací postup patentovaný v roce 1917 Gilbertem Vernamem. Spočívá v posunu každého znaku zprávy o náhodně zvolený počet míst v abecedě. To se prakticky rovná náhradě zcela náhodným písmenem a na tomto faktu je založen důkaz, že Vernamova šifra je v principu nerozluštiteLNÁ.

Postup šifrování [editovat | editovat zdroj]

Vezmeme jednotlivá písmena tajné zprávy a každé z nich posuneme o několik pozic v abecedě. Například první písmeno je posunuto o 5 pozic, druhé o 1, třetí o 14, čtvrté o 24, další o 9, 0, 3, 9, 19. Když při posouvání překročíme konec abecedy, pokračujeme od jejího začátku. Ze slova ALDEBARAN tak dostaneme šifrový text FMRCKAUJG. Posloupnost 5, 1, 14, 24, 9, 0, 3, 9, 19 je klíčem k rozluštění zprávy. Kdo ji zná, dokáže snadno posunout písmena opačným směrem a získat původní text. Bez znalosti klíče je luštění odposlechnuté zprávy nemožné.

CIHJT UUHML F
DCXAC JSJUK B
UOLYQ OKOXH P
DZYYO YKAIE L
MHQHL OHQQD S
AWSSZ YMFDI A
HQVHU OCJGU Q
Text zašifrovan

Podmínky spolehlivosti [editovat | editovat zdroj]

Porušení kterékoli z následujících podmínek má za následek oslabení šifry, takže by již nebyla nerozluštiteLNÁ.

1. **Klíč je tak dlouhý jako přenášená zpráva.** Jiné šifrovací systémy používají kratší klíče, což znamená, že počet možných klíčů je menší než počet možných zpráv. Kratší klíč v principu umožňuje útok hrubou silou.
2. **Klíč je dokonale náhodný.** Nepřipadají v úvahu počítacové generátory pseudonáhodných čísel, neboť jejich činnost lze předvídat. Nevhodnější je užití fyzikálních metod - hardware generátor náhodných čísel.
3. **Klíč nelze použít opakováně.** Tato podmínka je vlastně důsledkem předchozí, protože opakováný klíč není náhodný. Dostane-li útočník do ruky dvě zprávy zašifrované týmž klíčem, má často velmi snadnou cestu k rozluštění.

Vernamova šifra je symetrická proudová šifra spočívající v binární operaci XOR nad otevřeným textem a předem smluvěným náhodným klíčem (šumem).

Klíč se za žádných okolností nesmí recyklovat - na každou komunikaci se použije vždy nový - protože jinak by útočník mohl XORovat obě zašifrované zprávy, čímž by získal XOR obou nezašifrovaných zpráv, z čehož lze statistickými metodami získat otevřený text obou zpráv. Pokud se dodrží tato zásada a klíč je vygenerován skutečně náhodným způsobem (nikoliv pseudonáhodným), pak šifru nelze prolomit.

One time pad - Vernamova šifra

Abeceda $A = \mathbb{Z}_2$.

Množina klíčov i zašifrovaných textov je \mathbb{Z}_2 .

$x_1, x_2, \dots, x_n, \dots$ – prúd znakov priameho textu

$k_1, k_2, \dots, k_n, \dots$ – prúd klíčov, $P(k_i = 0) = P(k_i = 1) = \frac{1}{2}$

$y_1, y_2, \dots, y_n, \dots$ – prúd znakov zašifrovaného textu

$$\begin{array}{ccccccc} x_1, & x_2, & x_3, & \dots, & x_i, & \dots \\ k_1, & k_2, & k_3, & \dots, & k_i, & \dots \\ y_1 = x_1 \otimes k_1, & y_2 = x_2 \otimes k_2, & y_3 = x_3 \otimes k_3, & \dots, & y_i = x_i \otimes k_i, & \dots \end{array}$$

Ak sú klíče $k_1, k_2, \dots, k_n, \dots$ vyberané náhodne s rovnomerným rozdelením pravdepodobnosti, niet šance na prelomenie Vernamovej šifry.

Nevýhody:

- Klíč musí být aspoň tak dlhý, ako je správa
- Klíč sa nesmie použiť viac, ako raz

Získavanie náhodných postupností

- z výstupu Gieiger-Mullerovho počítača
- meranie nepravidelností silne zamestnaného servera
- meranie teplotných fluktuácií

Výsledky takýchto meraní budú súčasťou náhodné, ale pravdepodobnosti nul a jedničiek nemusia byť rovnaké.

Jeden spôsob vyrovnávania prevadepodobností je tento:

$\underbrace{00}_{-} | \underbrace{00}_{-} | \underbrace{10}_1 | \underbrace{11}_{-} | \underbrace{01}_0 | \underbrace{01}_0 | \underbrace{00}_{-} | \underbrace{11}_{-} | \underbrace{10}_1 | \underbrace{00}_{-} | \underbrace{10}_1 |$

Generátor RC4

Máme 256 S-boxov $S[0], S[1], \dots, S[255]$, ktoré obsahujú niektorú permutáciu čísel 0 až 255.

 Kľúč môže byť až $256 \times 8 = 2048$ bitov. Týmito bitmi sa napĺňajú postupne 8-bitové čísla $K[0], K[1], \dots, K[255]$.

Útok pri viacnásobnom použití toho istého prúdu kľúčov

Predpokladajme, že dve postupnosti znakov priameho textu

$$a_1, a_2, \dots, a_n, \dots, \quad b_1, b_2, \dots, b_n, \dots$$

boli zašifrované tým istým prúdom kľúčov $k_1, k_2, \dots, k_n, \dots$

Kryptoanalytik dostane dva zašifrované texty $y_1, y_2, \dots, y_n, \dots, z_1, z_2, \dots, z_n, \dots$ také, že

$$y_i = a_i \otimes k_i, \quad z_i = b_i \otimes k_i.$$

Kryptoanalytik si vypočíta postupnosť $w_1, w_2, \dots, w_n, \dots$, kde $w_i = y_i \otimes z_i$. Platí:

$$w_i = y_i \otimes z_i = (a_i \otimes k_i) \otimes (b_i \otimes k_i) = (a_i \otimes b_i) \otimes (k_i \otimes k_i) = (a_i \otimes b_i) \otimes 0 = (a_i \otimes b_i)$$

Postupnosť w_1, w_2, \dots je postupnosť znakov jedného priameho textu zašifrovaná postupnosťou iného priameho textu a takáto postupnosť nesie dostať informácie na odhalenie podstanej časti oboch priamych textov a v konečnom dôsledku aj postupnosti bitov kľúča.

Synchronizácia zašifrovaných textov

Kryptoanalytik zachytí dve postupnosti zašifrovaných textov

$$y_1, y_2, \dots, y_n, \dots, z_1, z_2, \dots, z_n, \dots,$$

ktoré boli zašifrované tým istým prúdom kľúčov, avšak sú navzájom posunuté o d pozícii, t.j.

$$y_i = a_i \otimes k_{i+d}, \quad z_i = b_i \otimes k_i.$$

Ak vytvorí postupnosť

$$w_i = y_i \otimes z_i = (a_i \otimes k_{i+d}) \otimes (b_i \otimes k_i) = (a_i \otimes b_i) \otimes (k_{i+d} \otimes k_i),$$

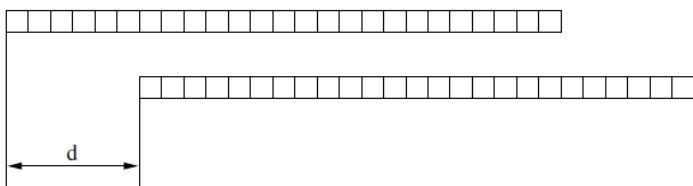
táto sa bude javiť ako postupnosť náhodných bitov.

Ak však posunie zašifrovaný text $z_1, z_2, \dots, z_n, \dots$, oproti textu $y_1, y_2, \dots, y_n, \dots$ od d pozícii dozadu, a vytvorí postupnosť

$$w_i = y_i \otimes z_{i+d} = (a_i \otimes k_{i+d}) \otimes (b_{i+d} \otimes k_{i+d}) = (a_i \otimes b_{i+d}) \otimes (k_{i+d} \otimes k_{i+d}) = a_i \otimes b_{i+d},$$

počet nul v tejto postupnosti nápadne stúpne, lebo pravdepodobnosť nuly je pravdepodobnosťou, že $a_i = b_{i+d}$, čo sa rovná príslušnému indexu koincidencie.

Posúvame proti sebe oba zašifrované texty. Pri zasynchronizovaní – nájdení správnej vzdialenosťi d počet zhôd nápadne stúpne.



9.Pseudogeneratory (linearny kongruencny, kvadraticky, kubicky, RC4)

Použitie generátorov náhodných čísel

Lineárny kongruenčný generátor

$$X_n = (aX_{n-1} + b) \mod m$$

Periódna max $m - 1$.

Kvadratický kongruenčný generátor

$$X_n = (aX_{n-1}^2 + bX_{n-1} + c) \mod m$$

Kubický kongruenčný generátor

$$X_n = (aX_{n-1}^3 + bX_{n-1}^2 + cX_{n-1} + d) \mod m$$

Joan Boyar dokázala, že lineárny a nesjkôr aj ostatné kongruenčné generátory sú kryptograficky slabé. Nesmú sa používať v silnej kryptografii!!

Máme 256 S-boxov $S[0], S[1], \dots, S[255]$, ktoré obsahujú niektorú permutáciu čísel 0 až 255.

```
rand()
i=i+1 mod 256
j=j+S[i] mod 256
swap(S[i],S[j])
t=(S[i]+S[j]) mod 256
k=S[t]
return k
```

Inicializačná procedúra pre RC4

Kľúč môže byť až $256 \times 8 = 2048$ bitov. Týmito bitmi sa napĺnia postupne 8-bitové čísla $K[0], K[1], \dots, K[255]$.

Inicializačná procedúra je takáto:

```
for i=0 to 255
{
    S[i]=i
}
j=0
for i=0 to 255
{
    j=(j+S[i]+K[i]) mod 256
    swap(S[i],S[j])
}
i=0
j=0
```

Podobné sú generátory pseudonáhodných čísel označované ako VMPC.

Je tu to isté nebezpečenstvo pri viacnásobnom používaní rovnakého kľúča ako pri Vernamovej šifre.

10. Testy generatorov (frekvenčny, TwoBits, Runs, Poker, FIPS 140-1, Autokorelacny)



Frekvenčný test

Máme postupnosť bitov $\mathbf{b} = b_1, b_2, \dots, b_n$.

$$n_0 - \text{počet núl} \quad n_1 - \text{počet jednotiek} \quad n = n_0 + n_1$$

Za predpokladu, že \mathbf{b} je náhodná postupnosť s rovnakou pravdepodobnosťou núl a jednotiek má štatistika

$$X_1 = \frac{(n_0 - n_1)^2}{n}$$

$\chi^2(1)$ rozdelenie s jedným stupňom voľnosti pre $n \geq 10$ a testovaná hypotéza H je že $X_1 = 0$.



Dvojbitový sériový test

Dvojbitový sériový test

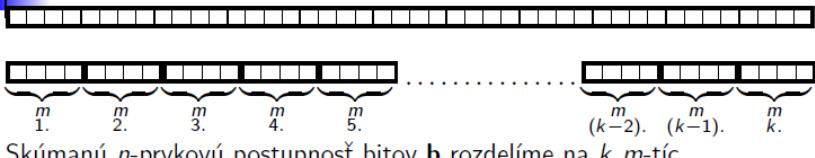
$n_{00}, n_{01}, n_{10}, n_{11}$ – počet výskytov dvojíc 00, 01, 10, 11 v postupnosti \mathbf{b} .

Platí $n_{00} + n_{01} + n_{10} + n_{11} = n - 1$.

$$X_2 = \frac{4}{n-1} (n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2) - \frac{2}{n} (n_0^2 + n_1^2) + 1$$

Pre $n \geq 21$ má štatistika X_2 rozdelenie $\chi^2(2)$ s dvoma stupňami voľnosti. Testujeme platnosť hypotézy $X_2 = 0$.

Poker test



Skúmanú n -prvkovú postupnosť bitov \mathbf{b} rozdelíme na k m -tíc.

Zrejme je $k \cdot m \leq n$.

Číslo m musí byť zvolené tak, aby $k \geq 5 \cdot 2^m$.

Každá m -tica bitov predstavuje číslo v rozmedzí 0 až $2^m - 1$.

Pre $i = 0, 1, 2, \dots, 2^m - 1$ označme n_i počet m -tíc takých, že predstavujú binárny rozvoj čísla i .

$$X_3 = \frac{2^m}{k} \cdot \left(\sum_{i=0}^{2^m-1} n_i^2 \right) - k$$

Štatistika X_3 má rozdelenie $\chi^2(2^m - 1)$ a testujeme hypotézu $X_3 = 0$.

Runs test

Blok dĺžky n je postupnosť n jednotiek v postupnosti \mathbf{b} z oboch strán ohraňičená nulou alebo začiatkom alebo koncom postupnosti b .

Medzera (Gap) dĺžky n je postupnosť n núl v postupnosti \mathbf{b} z oboch strán ohraňičená jednotkou alebo začiatkom alebo koncom postupnosti b .
Pravdepodobnosť výskytu bloku dĺžky i : $\dots 0 \underbrace{1 1 \dots 1}_i 0 \dots$

v nekonečne dlhej náhodnej postupnosti bitov je $\frac{1}{2^{i+2}}$.

Očakávaný počet blokov dĺžky i v n -prvkovej postupnosti \mathbf{b} je $e_i = \frac{n-i+3}{2^{i+2}}$.

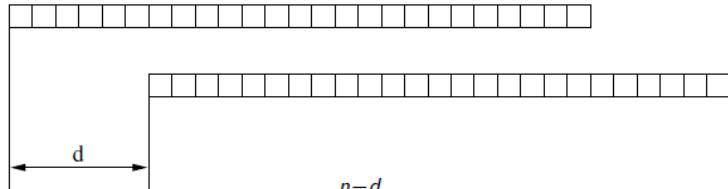
$$X_4 = \sum_{i=1}^k \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^k \frac{(G_i - e_i)^2}{e_i}$$

kde k je najväčší také, že $e_i \geq 5$ a B_i, G_i je skutočný počet blokov, resp. medzier dĺžky i v postupnosti \mathbf{b} .

Štatistika X_4 má rozdelenie $\chi^2(2k - 2)$, testovaná hypotéza je $X_4 = 0$.

Autokorelačný test

d – pevné číslo $1 \leq d \leq [\frac{n}{2}]$



$$A(d) = \sum_{i=1}^{n-d} b_i \oplus b_{i+d}$$

$$X_5 = 2 \cdot \frac{A(d) - \frac{n-d}{2}}{\sqrt{n-d}}$$

Štatistika X_5 má normálne rozdelenie $N(0, 1)$.

Testujeme hypotézu $X_5 = 0$.

FIPS 140–1 štatistický test

Test je určený pre reťaze **b** dlhé 20000 bitov.

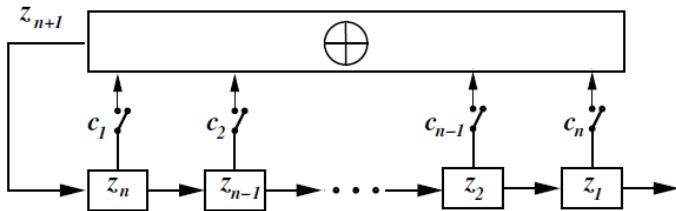
- ➊ Monobit test: $1 < n_1 < 10346$
- ➋ Poker test pre $m = 4$: $1.03 < X_3 < 57.4$
- ➌ Runs test.
Pre $i = 1, 2, 3, 4, 5$ B_i resp. G_i – počet blokov resp. medzier dĺžky i .
Pre $i = 6$ B_6 resp. G_6 počet blokov resp. medzier dĺžky 6 a viac.

i	Dovolený rozsah B_i, G_i
1	2267 – 2733
2	1079 – 1421
3	502 – 748
4	223 – 402
5	90 – 223
6	90 – 223

- ➍ Long run test. Nesmie existovať blok alebo medzera dĺžky 34 alebo viac.

11. LFSR

Lineárny posuvný register



Postupnosť c_1, c_2, \dots, c_n – zpätno-väzbová sekvenčia – tap sequence

$$z_{n+1} = c_1 z_n + c_2 z_{n-1} + \dots + c_{n-1} z_2 + c_n z_1 \quad (1)$$

Maximálna períoda LFSR dĺžky n je $2^n - 1$.

Zpätno-väzbový polynom – connection polynomial – je polynom nad \mathbb{Z}_2 :

$$1 + c_1x + c_2x^2 + c_3x^3 + \dots + c_nx^n$$

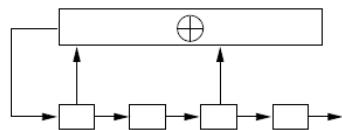
Primitívny polynom stupňa n je taký polynom ktorý je

- irreducibilný
- je deliteľom polynómu $x^{2^n-1} + 1$
- nie je deliteľom žiadneho polného tvaru $x^d + 1$, kde d delí $2^n - 1$

Zpätnoväzobný polynom

Platí: Lineárny posuvný register dĺžky n má maximálnu períodu $2^n - 1$ práve vtedy, keď jeho zpätnoväzobný polynom je primitívny.

Singulárny LFSR je taký LFSR, ktorého dĺžka je väčšia než stupeň väzobného polynómu.

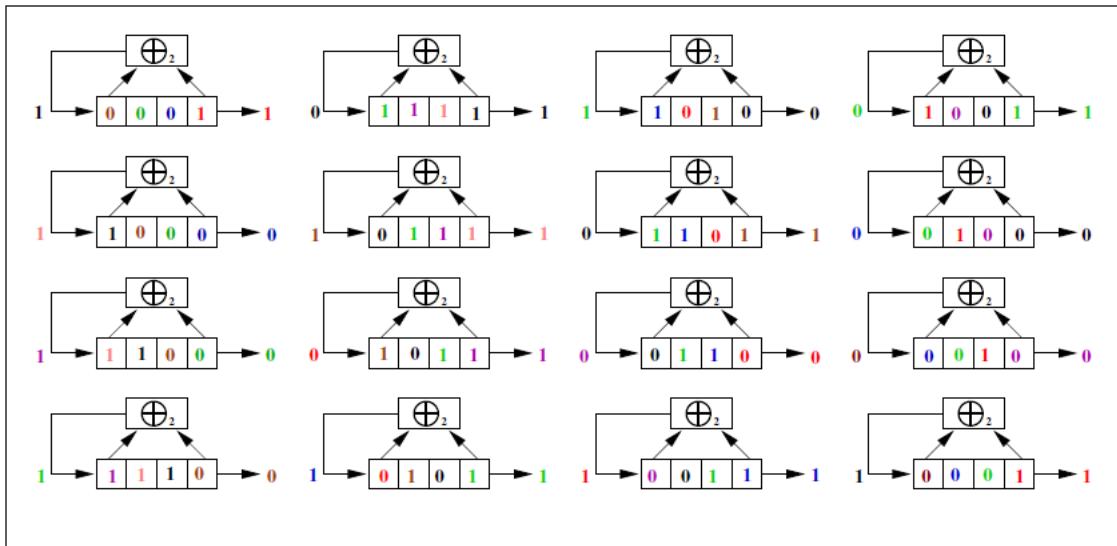


Nie sú zaručená periodicitá pre každý počiatočný stav singulárnych LFSR, preto sa v kryptografii nepoužívajú.

Zistiť, či je daný polynom primitívny je algoritmicky riešiteľný problém.

Hľadanie primitívnych polynómov je ťažké.

Príklad práce LFSR



LFSR v tabuľkovom procesore

	A	B	C	D	E
1	0	0	0	0	0
2	=MOD(A1+D1+E1;2)	=A1	=B1	=C1	=D1

Druhý riadok tabuľky sa rozkopíruje do ďalších riadkov stĺpcov A až E.

Výstupné bity z LFSR sa použijú ako prúd pseudonáhodných binárnych čísel.

Kľúč:

- Počiatočné nastavenie registra – n bitov z_1, z_2, \dots, z_n
- Nastavenie zpätnoväzbovej postupnosti n bitov c_1, c_2, \dots, c_n

Ak poznáme zpätnoväzobnú postupnosť a ak a odchytíme porade n bitov z LFSR, ďalšie bity ľahko vypočítame podľa rovnice (1).

Útok na LFSR ak poznáme $2n$ bitov

Ak poznáme len dĺžky LFSR postupujeme nasledovne:
Predpokladajme, že poznáme n – dĺžku LFSR a $2n$ výstupných bitov:

$$z_{2n}, z_{2n-1}, \dots, z_2, z_1$$

$$\begin{aligned} z_{n+1} &= c_1 z_n \oplus \dots \oplus c_{n-1} z_2 \oplus c_n z_1 \\ z_{n+2} &= c_1 z_{n+1} \oplus \dots \oplus c_{n-1} z_3 \oplus c_n z_2 \\ &\dots \\ z_{2n} &= c_1 z_{2n-1} \oplus \dots \oplus c_{n-1} z_n z_{n-1} \oplus c_n z_n \end{aligned}$$

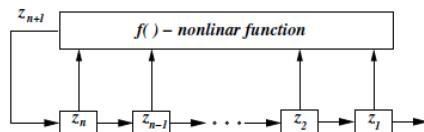
$$\begin{pmatrix} z_n & z_{n-1} & \dots & z_1 \\ z_{n-1} & z_{n-2} & \dots & z_2 \\ \dots & \dots & \dots & \dots \\ z_{2n-1} & z_{2n-2} & \dots & z_n \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix}$$

$$Z\mathbf{c} = \mathbf{z} \quad \mathbf{c} = Z^{-1}\mathbf{z}$$

Dôsledok: Kryptografia pomocou LFSR je veľmi slabá a nesmie sa používať.

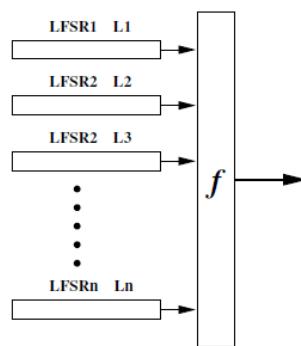
Pokusy o zlepšenie bezpečnosti LFSR

Náhrada \oplus nelineárnu funkciou:



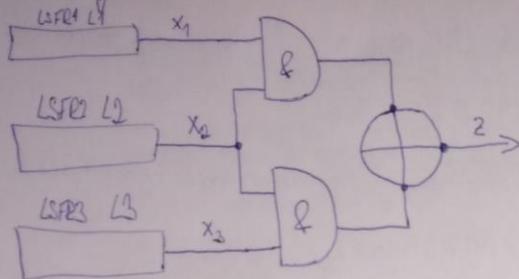
Nevýhoda: Ťažko sa teoreticky študujú, ťažko sa dokazujú vlastnosti ako napr. existencia krátkych cyklov.

Výstupy z viacerých LFSR použiť ako vstupy do nelineárnej funkcie.



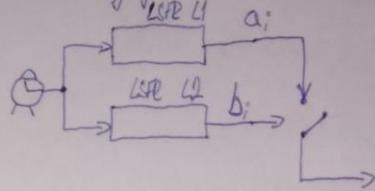
Modifikácie LSR

1) Gelfredo generator



$$z = x_1 \cdot x_2 \oplus (1 \oplus x_1) \cdot x_3$$

2) Shrinking generator



Uchádza len o a_i , pre ktorú je $b_i = 1$.

Ak L_1, L_2 nesúdeliteľné, tak generátor má periodu $(2^{L_1}-1) \cdot (2^{L_2}-1)$

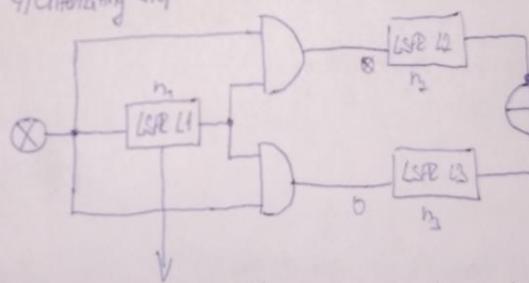
3) Blum-Micali generator

g, p - dve príve veľké prírodné

$$x_{i+1} = g^x \pmod{p}$$

$$b_i = \begin{cases} 1 & \text{ak } x_i < \frac{p-1}{2} \\ 0 & \text{inak} \end{cases}$$

4) Alternating step



Roznie sa iba LSF

alebo 3

ak L_1, L_2, L_3 nesúdeliteľné
tak perioda je $2^{n_1}(2^{n_2}-1)$

- upravený tak, že po $n-1$ ruboch následuje jednú nulu

- bezpečný

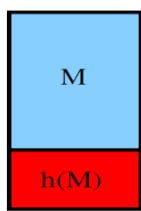
12. Hash funkcie vseobecne, vlastnosti,(vseobecne, MD5, SHA)

Vlastnosti hashu by prednáška:

▀ kryptografii sa na zaistenie správ proti zmenám pridáva ku každej správe M ďalšia (redundantná) časť, nazývaná **odtlačok správy**.

V anglickej literatúre MAC(M) – Message Atentication Code,
MD(M) – Message Digest, Fingerprint.

Tieto sa vypočítavajú pomocou jednocestných hashovacích funkcií.



Požadované vlastnosti hashovacej funkcie $h(M)$

- ① Pre každé M je ľahké vypočítať $h(M)$
- ② Pre každé h je ľažké nájsť také M , že $h = h(M)$
- ③ Pre každé M je ľažké nájsť iné M' také, že $h(M) = h(M')$
- ④ Je ľažké nájsť dve rôzne náhodné správy $M \neq M'$ také, že $h(M) = h(M')$

Dvojica správ M, M' s vlastnosťou $h(M) = h(M')$ sa nazýva kolízia.
Vlastnosť 4. sa nazýva odolnosť voči kolízii – collision resistance.

Medzi hlavné vlastnosti hashovacej funkcie patrí:

- Rýchlosť transformácie – zo vstupu sa rýchlo vyráta HASH
- Lavínovitosť - malou zmenou vstupných dát dosiahneme veľké zmeny na výstupe (odtlačok je odlišný od pôvodného na prvý pohľad)
- Jednocestnosť - z hashu je prakticky nemožné rekonštruovať pôvodný text správy (rozdier oproti klasickému šifrovaniu)
- Bezkolíznosť - odolnosť voči kolízii (dvom rôznym správam odpovedá rovnaký hash ... v praxi veľmi nepravdepodobné, pomocí hashu v praxi identifikovať práve jednu správu (overiť jej správnosť))

Hash pomocou kryptosystému:

$$h_i = E_{h_{i-1}}(m_i) \bigcirc (\text{znamienko + v krúžku}) m_i$$

↑ bude budúci kľúč, takže dĺžka kľúča musí byť rovná dĺžke bloku.

$$\begin{aligned} h_0 &= IV & h_i &= E_{m_i}(h_{i-1}) \oplus h_{i-1} \\ h_i &= f(m_i, h_{i-1}) & h_i &= E_{m_i}(m_i \oplus h_{i-1}) \oplus h_{i-1} \\ h_i &= E_{h_{i-1}}(m_i) \oplus m_i & h_i &= E_{m_i \oplus h_{i-1}}(m_i) \oplus m_i \\ h_i &= E_{h_{i-1}}(m_i) \oplus m_i \oplus h_{i-1} & h_i &= E_{m_i \oplus h_{i-1}}(h_{i-1}) \oplus h_{i-1} \\ h_i &= E_{h_{i-1}}(m_i \oplus h_{i-1}) \oplus m_i \oplus h_{i-1} & h_i &= E_{m_i \oplus h_{i-1}}(m_i) \oplus h_{i-1} \\ h_i &= E_{h_{i-1}}(h_{i-1}) \oplus m_i & h_i &= E_{m_i \oplus h_{i-1}}(h_{i-1}) \oplus m_i \end{aligned}$$

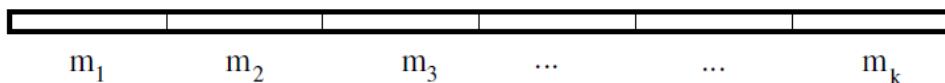
Prelomená schéma $h_i = E_{m_i}(h_{i-1})$.

Ak by mal odtlačok správy 64 bitov, t.j. $n = 2^{64}$, stačí vytvoriť $1,17 * 2^{32} \approx 5 * 10^9$ náhodných správ, aby sme s pravdepodobnosťou $1/2$ našli kolíziu.

Preto sa používajú odtlačky dĺhé 128, 160, 256 bitov.

Všeobecný postup tvorby odtlačku správy

- ❶ Správa M , pre ktorú sa robí odtlačok, sa rozdelí na rovnako dĺhé bloky m_1, m_2, \dots



- ❷ Hashovací algoritmus má pevne stanovený inicializačný vektor IV . Položíme $h_0 = IV$.
- ❸ Rekurzívne počítame $h_i = f(m_i, h_{i-1})$.
- ❹ Výsledný odtlačok celej správy $h(M) = h_k$.

dĺžka odtlačku 3 kola

narodeninový paradox – v skupine 23 ľudí sa s pravdepodobnosťou $p > \frac{1}{2}$ nájde dvojica ktorá má v ten istý deň narodeniny



MD5

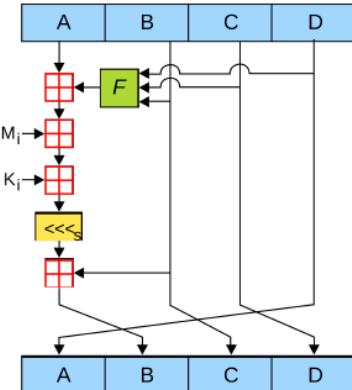
- ❶ Je zosilnením algoritmu MD4.
- ❷ Dáva 128-bitový hash.
- ❸ Pracuje s 512-bitovým blokom textu
- ❹ Namiesto troch kôl má 4 kolá
- ❺ Má pozmenené funkcie takto

$$F(X, Y, Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$

$$G(X, Y, Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$$

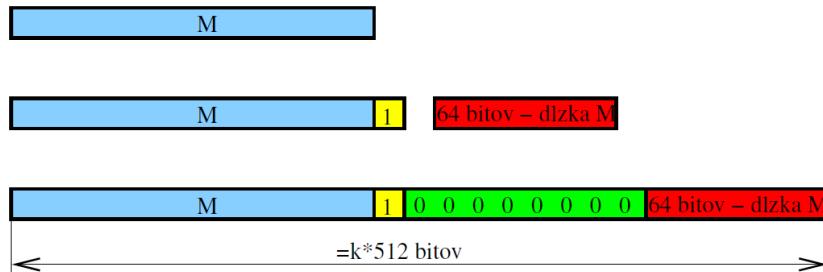
$$H(X, Y, Z) = X \oplus Y \oplus Z$$

$$I(X, Y, Z) = Y \oplus (X \vee \neg Z)$$
- ❻ Beží asi o 30% pomalšie než MD4



MD4

Správa sa pred výpočtom odtlačku musí upraviť takto:



- ❶ Pridá sa jeden bit s hodnotou 1.
- ❷ Vytvorí sa 64-bitové číslo obsahujúce dĺžku správy. Týmto číslom bude upravená správa končiť.
- ❸ Medzi doplnenú jednotku a 64 bitov dĺžky sa vloží toľko núl, aby výsledná dĺžka správy bola násobkom 512.

- ❹ Dĺžka odtlačku algoritmu MD4 je 128 bitov, t.j. h_i má 128 bitov.
- ❺ h_i sa pracuje ako so štvoricou (A, B, C, D) 32-bitových čísel
- ❻ Spracovávaná dĺžka bloku správy m_i je 512 bitov.
- ❼ S blokom textu sa pracuje ako so 16-ticou

$$X[0], X[1], X[2], \dots, X[15]$$

32-bitových čísel.

- ❽ Inicializačne sa nastaví hodnota $h_0 \equiv (A, B, C, D)$
- ❾ i -tý 512-bitový blok textu m_i sa vyjadri v tvare šestnásťich 32-bitových čísel $X[0], X[1], X[2], \dots, X[15]$ a rekurentne sa vypočíta

$$h_i = f(m_i, h_{i-1})$$

- ❿ Ak m_k je posledný blok správy, potom h_k je odtlačok celej správy.

SHA:

- Vycinuli v NSA
- Náhrada MD5
- SHA-0 prelomený => použ. SHA-1 / SHA-2 / SHA-3



SHA algoritmus

- ➊ SHA produkuje 160-bitový hash
- ➋ Spracováva 512 bitový blok textu $W[0], W[1], \dots, W[15]$, ktorý expanduje do 80 takto: pre $16 \geq j \leq 79$

$$W[j] = W[j - 3] \oplus W[j - 8] \oplus W[j - 14] \oplus W[j - 16]$$

- ➌ Má 4 kolá po 20 krokov

SHA-256:

- dĺžka hashu 256
- patrí k najviac používaným
- prelomenie sa považuje za prakticky nemožné
- konštruovaná ako vysoko nelineárna
- patrí do „rodiny“ SHA-2

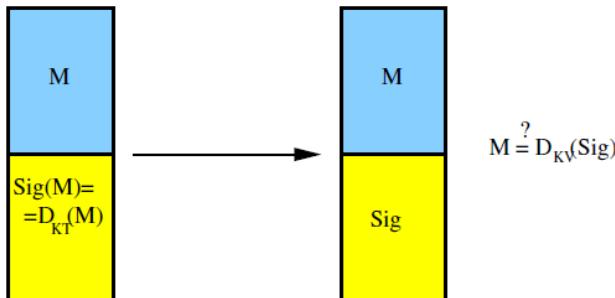
13. Digitalny podpis - (plus Birdthday attack, zmenky)

Používa sa na overenie integrity údajov, neumožnuje overiť pôvod údajov - autentifikáciu

Digitálny podpis

- Účastník A s dvojicou kľúčov KV_A, KT_A podpíše správu M tak, že k nej pripojí výsledok dešifrovania správy M kľúčom KT_A . Teda

$$Sig(M) = D_{KT_A}(M).$$



- Účastník B overí pravosť podpisu tak, že vypočíta $M' = E_{KV_A}(Sig(M))$ a skontroluje, či $M = M'$.

Ak $M' \neq M$, potom buď správa bola zmenená, alebo podpis nie je pravý.

Ak $M' = M$, potom je podpis pravý a správa nezmenená.

Jediný človek – účastník A – mohol ku správe M vytvoriť

$Sig(M) = D_{KT_A}(M)$, pretože on jediný má kľúč KT_A .

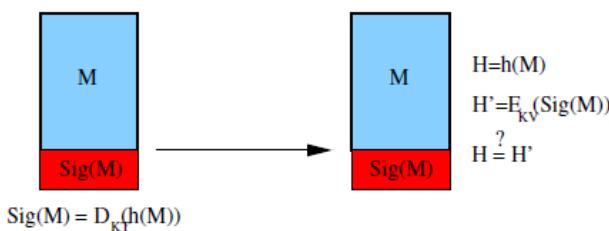
Digitálny podpis

- Účastník A s dvojicou kľúčov KV_A, KT_A podpíše správu M tak, že

- Vypočíta $h(M)$ odtlačok správy M .
- Odtlačok správy $h(M)$ zašfriuje svojim tajným kľúčom:

$$Sig(M) = D_{KT_A}(h(M)).$$

- $Sig(M)$ pripojí k správe M ako svoj digitálny podpis



- Účastník B overí pravosť podpisu tak, že

- Vypočíta $H = h(M)$
- Vypočíta $H' = E_{KV_A}(h(M))$.
- Skontroluje, či $H' = H$.

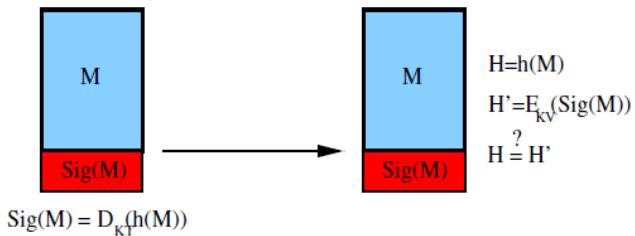
Ak $H' \neq H$, potom buď správa bola zmenená, alebo podpis nie je pravý.

Ak $H' = H$, potom je podpis pravý a správa nezmenená.

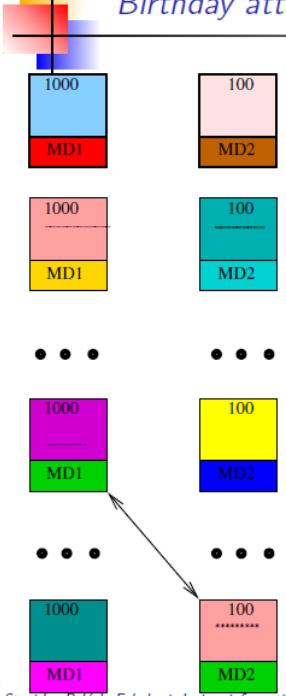
Jediný človek – účastník A – mohol ku správe M vytvoriť

$Sig(M) = D_{KT_A}(h(M))$, pretože on jediný má kľúč KT_A .

Digitálny podpis



Birthday attack - Narodeninový útok



- ➊ Útočník vytvorí dve zmenky – jednu na 100 euro, druhú na 1000 euro
- ➋ Z obidvoch zmeniek bezvýznamy zmenami vytvára ich ďalšie varianty dokedy nenájde kolíziu – dvojicu 100 eurového a 1000 eurového variantu s rovnakým odtlačkom h . Ak má odtlačok n možných hodnôt, stačí mu vytvoriť $1.17\sqrt{n}$ dvojíc variantov zmeniek, aby s pravdepodobnosťou $> \frac{1}{2}$ našiel kolíziu.
- ➌ Dížníkovi dá potvrdiť 100 eurový variant s odtlačkom h .
- ➍ Po čase vymáha 1000 euro na základe toho, že mu dížník potvrdil odtlačok h prislúchajúci 1000 eurovému variantu.

Poučenie: Pred podpisom digitálneho dokumentu vždy v ňom urobiť malú zmenu.

Deffie – Hellmanova výmena kľúčov



Diffie - Hellmanova výmena kľúčov

A a B sa dohodnú na veľkom prvočíslle p a čísle s, $1 < s < p$.

Čísla s, p môžu byť verejné, použiteľné opakovane aj pre viac používateľov.

A

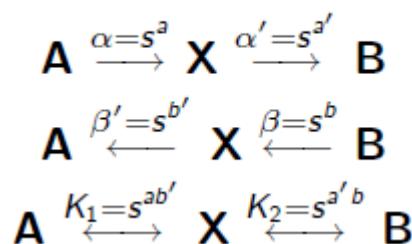
B

- | | |
|---|--|
| <ul style="list-style-type: none">• Zvolí $a < p$ tajné.• Vypočíta $\alpha = s^a \pmod{p}$.• Odošle α.• Prijme β.• Vypočíta kľúč $K_A = \beta^a \pmod{p}$
Je $K_A = K_B$? | <ul style="list-style-type: none">• Zvolí $b < p$ tajné.• Vypočíta $\beta = s^b \pmod{p}$.• Odošle β.• Prijme α.• Vypočíta kľúč $K_B = \alpha^b \pmod{p}$ |
|---|--|

Platí:

$$K_A = \beta^a = (s^b)^a = s^{ab} = (s^a)^b = \alpha^b = K_B \pmod{p}$$

Nebezpečenstvo: Intruder in the middle attack



14. AES

- Symetrická bloková šifra (zrejme najviac použ. sym. blok. šifra súčasnosti)
- dĺžka bloku je 128 bitov
- dĺžka kľúča 128/192/256 – voliteľná

Výhody:

- Výkonnosť v hardvérovej i softvérovej implementácii
- Nízke pamäťové nároky
- Možnosť ochrany pred útokmi parazitnými kanálmi

128-bit blok namiesto textu berieme ako postupnosť 16 bajtov:

$a_{00} \ a_{10} \ a_{20} \ a_{30} \ | \ a_{01} \ a_{11} \ a_{21} \ a_{31} \ | \ a_{02} \ a_{12} \ a_{22} \ a_{32} \ | \ a_{03} \ a_{13} \ a_{23} \ a_{33}$

Tie sa usporiadajú do tabuľky, ktorá sa volá stav:

a_{00}	a_{01}	a_{02}	a_{03}
a_{10}	a_{11}	a_{12}	a_{13}
a_{20}	a_{21}	a_{22}	a_{23}
a_{30}	a_{31}	a_{32}	a_{33}

Stav

k_{00}			
			k_{33}

Kolový kľúč

S týmto stavom sa iteračne vykonáva niekoľko kôl operácií (10,12,14 – podľa dĺžky kľúča), niektoré z nich závisia na kolovom kľúči reprezentovanom ako matica bajtov.

AES je jediný verejne dostupný šifrovací algoritmus schválený NSA pre najtajnejšie informácie.

Pre stavovú maticu sú definované 4 elementárne operácie:

- substitúcia bajtov (SubBytes)
- rotácia riadkov (MixColumns)
- substitúcia stĺpcov (ShiftRows)
- pričítavanie iteračného kľúča (AddRoundKey)

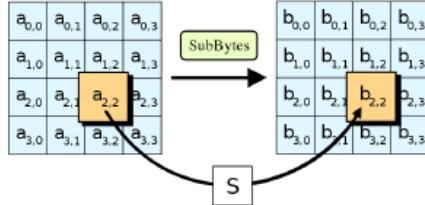
Operácia SubBytes: (Substitúcia bajtov)



AES - Operácia SubBytes

S každým bajtom a tabuľky Stav sa vykonajú dve operácie:

- ① Najprv sa k hodnote a najde v poli $GF(2^8)$ inverzný prvk $x = a^{-1}$, ak $a \neq 0$. Ak $a = 0$, položíme $x = 0$.



- ② Potom sa vypočítia byte $b = b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7$

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

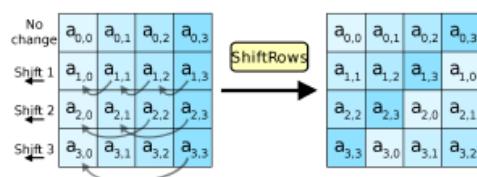
Operácia ShiftRows: (Rotácia riadkov)



AES - Operácia ShiftRows

Na riadky tabuľky Stav sa aplikujú nasledujúce Ľavé rotačné posuny

- ① 1. riadok ostáva bez zmeny
- ② 2. riadok - posun o 1 bajt - t.j. 8 bitov
- ③ 3. riadok - posun o 2 bajty - t.j. 16 bitov
- ④ 4. riadok - posun o 3 bajty - t.j. 24 bitov



Operácia MixColumns: (Substitúcia stĺpcov)

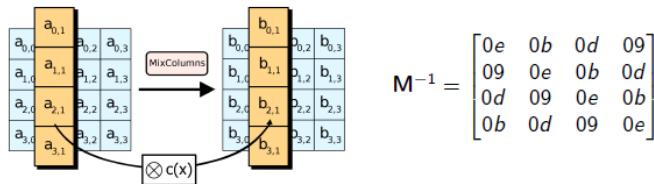
AES - Operácia MixColumns

Pri tejto operácii považujeme maticu Stav za maticu prvkov poľa $GF(2^8)$.

S každým jej stĺpcom $\mathbf{a}_i = [a_{0i} \ a_{1i} \ a_{2i} \ a_{3i}]^T$ vykonáme

$$\underbrace{\begin{bmatrix} b_{0i} \\ b_{1i} \\ b_{2i} \\ b_{3i} \end{bmatrix}}_{\mathbf{b}_i} = \underbrace{\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}}_{M} \otimes_{GF(2^8)} \underbrace{\begin{bmatrix} a_{0i} \\ a_{1i} \\ a_{2i} \\ a_{3i} \end{bmatrix}}_{\mathbf{a}_i} \quad \text{t. j. } \mathbf{b}_i = M \otimes \mathbf{a}_i$$

V maticovom tvare: $\mathbf{B} = M \cdot \mathbf{A}$



Operácia AddRoundKey: (Pričítavanie iteračného klúča)

AES – Funkcia AddRoundKey

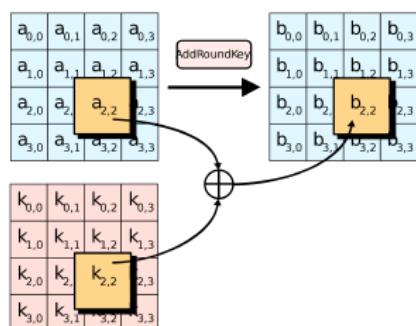
V tomto kole sa pre každý prvek a_{ij} Stavu vykoná

$$b_{ij} = a_{ij} \oplus k_{ij},$$

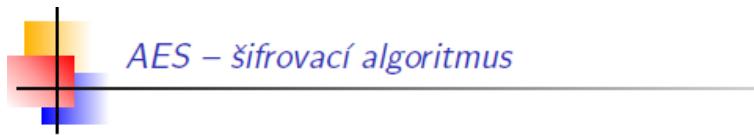
kde k_{ij} je prvek matice príslušného kolového klúča.

V maticovom tvare:

$$\mathbf{B} = \mathbf{A} \oplus \mathbf{K}.$$



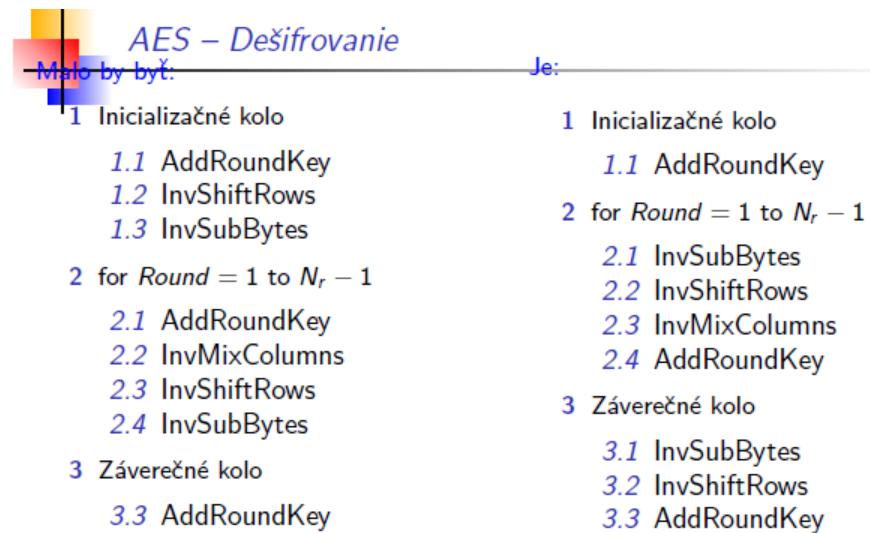
Šifrovanie:



- 1 Inicializačné kolo
 - 1.1 AddRoundKey
- 2 for $Round = 1$ to $N_r - 1$
 - 2.1 SubBytes
 - 2.2 ShiftRows
 - 2.3 MixColumns
 - 2.4 AddRoundKey
- 3 Záverečné kolo (bez MixColumns)
 - 3.1 SubBytes
 - 3.2 ShiftRows
 - 3.3 AddRoundKey

Dĺžka kľúča	128	192	256
Počet kôl N_r	10	12	14

Dešifrovanie:



Poradie operácií InvShiftRows a InvSubBytes je zameniteľné.

$$\text{AddRoundKey}(\text{InvMixcolumns}(B)) = K \oplus M^{-1} \cdot B.$$
$$\text{InvMixcolumns}(\text{AddRoundKey}(B)) = M^{-1} \cdot (K \oplus B) = M^{-1}K \oplus M^{-1}B.$$

15. Asymetricka kryptografia (vseobecne principy, RSA)

- Pre šifrovanie a dešifrovanie využíva dva odlišné kľúče

Všeobecne:

Základná veta aritmetiky hovorí, že každé prirodzené číslo $m > 1$ sa dá jednoznačne napísať ako $m = p_1^{\alpha_1} * p_2^{\alpha_2} \dots p_k^{\alpha_k}$, kde

$p_1 \dots p_k$ sú navzájom rôzne prvočísla a α_i sú prirodzené čísla.

Najväčší spoločný deliteľ:

Hovoríme, že prirodzené číslo $d \in \mathbb{N}$ je najväčším spoločným deliteľom celých čísel $a \in \mathbb{Z}$, $b \in \mathbb{Z}$ a píšeme $d = NSD(a, b)$, ak platí

- ① $d | a$ a tiež $d | b$.
- ② Ak $d_1 \neq d$ a $d_1 | a$, $d_1 | b$, potom aj $d_1 | d$.

Euklidov algoritmus pre výpočet $NSD(a, b)$ $r_0 = a$, $r_1 = b$

$$\begin{aligned} r_0 &= r_1 \cdot q_1 + r_2, & r_2 < r_1 \\ r_1 &= r_2 \cdot q_2 + r_3, & r_3 < r_2 \\ &\dots \\ r_{i-1} &= r_i \cdot q_i + r_{i+1}, & r_{i+1} < r_i \\ &\dots \\ r_{m-1} &= r_m \cdot q_m + 0 \end{aligned}$$

$$r_m = NSD(r_0, r_1) = NSD(a, b)$$

Nevýhody symetrickej kryptografie:

- každá dvojica účastníkov musí udržiavať svoj kľúč a teda kľúčov je veľmi veľa a všetky sa musia držať v tajnosti.

Princíp kryptografie s verejným kľúčom:

- Každý účastník A ma jednu dvojicu kľúčov – verejný kľúč $KV(A)$ a tajný kľúč $KT(A)$. Verejný kľúč KV sa zverejní a tajný KT sa utají.

- Účastník A šifruje správu x účastníkovi B tak, že nájde verejný kľúč $KV(B)$ a pošle správu $y = E_{KV(B)}(x)$

-Účastník B dešifruje správu y predpisom $x = D_{KT(B)}(y)$.

RSA algoritmus:

- 1) Účastník A zvolí 2 veľké prvočísla p,q.
- 2) $n = p * q$
- 3) Vypočítame Eulerovu funkciu $\phi(n) = (p-1)(q-1)$
- 4) Ďalej zvolí 2 čísla $0 < e < \phi(n)$, $0 < d < \phi(n)$ také, že:
$$e * d \equiv 1 \pmod{\phi(n)}$$
- 5) Verejný kľúč účastníka A je dvojica (e, n) , jeho tajný kľúč je dvojica (d, n) .
- 6) Účastník B bude správu $x < n$ pre účastníka A šifrovať predpisom:
$$y = x^e \pmod{n}$$
- 7) Účastník A dešifruje správu y predpisom:
$$x = y^d \pmod{n}$$

Volba prvočísel p,q v RSA:



RSA algoritmus – voľba prvočísel p, q

Problém voľby prvočísel p, q.

- Dostatočná veľkosť – aspoň 512 – 1024 bitov.
- Zisťovanie prvočíselnosti – použiť niektorý pravdepodobnosťny test.
- Je ich dosť? Počet prvočísel menších než $n \approx \frac{n}{\ln n}$.

Problém voľby čísel e, d.

- Veľmi často sa volí $e = 65537 = 2^{16} + 1$. e je prvočíslo.
- Číslo d také, že $e \cdot d \equiv 1 \pmod{\phi(n)}$ sa nájde rozšíreným Euklidovým algoritmom.

Bezpečnosť proti útokom je založená na tom, že rovnicu $y = x^e \pmod{n}$ nevieme vyriešiť ani keď poznáme c. Potrebujeme faktorizovať n (rozložiť na prvočísla).

Kongruencia:

$a \equiv b \pmod{n}$ platí práve vtedy, keď obe čísla a, b dávajú po delení číslom n ten istý zvyšok.

Eulerova Funkcia:



Eulerova funkcia $\phi(n)$

Definícia. Nech $n \in \mathbb{N}$. Eulerova funkcia $\phi(n)$ je počet prirodzených čísel menších alebo rovných než n nesúdeliteľných s n .

n	1	2	3	4	5	6	7	8	9	10	11	12	13	...
$\phi(n)$	1	1	2	2	4	2	6	4	6	4	10	4	12	...

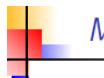
Ak p je prvočíslo, potom všetky čísla $1, 2, \dots, p - 1$ sú neúdeliteľné s p .

Ak p je prvočíslo, potom všetky súdeliteľné čísla s p menšie alebo rovne než p sú $1p, 2p, 3p, \dots, p^{n-1} \cdot p$ – je ich presne p^{n-1} .

Tvrdenie. Nech $p \in \mathbb{N}$ je prvočíslo, $n \in \mathbb{N}$, $n \geq 1$. Potom platí:

$$\begin{aligned}\phi(p) &= p - 1 \\ \phi(p^n) &= p^n - p^{n-1} = p^n \left(1 - \frac{1}{p}\right)\end{aligned}$$

Malá Fermatova veta:



Malá Fermatova veta

Malá Fermatova veta. Nech p je prvočíslo, nech c je ľubovoľné prirodzené číslo. Potom

$$c^p \equiv c \pmod{p}.$$

Ak navyše $c \in \{1, 2, \dots, p - 1\}$, potom

$$c^{p-1} \equiv 1 \pmod{p}.$$

Eulerova Veta:

Eulerova veta. Pre ľubovoľné číslo x nesúdeliteľné s číslom m platí

$$x^{\phi(m)} \equiv 1 \pmod{m}.$$

Fermatov test prvočíselnosti:

Fermatov test prvočíselnosti.

1. Ak pre niektoré $c < M$ je $c^{M-1} \not\equiv 1 \pmod{M}$, potom je c určite zložené číslo.
2. Ak pre dostatočne veľa čísel $c < M$ platí $c^{M-1} \equiv 1 \pmod{M}$, potom c je pravdepodobne prvočíslo.

Carmichaelove čísla:

Carmichaelove číslo – také zložené číslo M , že pre všetky $c < M$,
 c nesúdeliteľné s M platí $c^{M-1} \equiv 1 \pmod{M}$.

561	=	$3 \cdot 11 \cdot 17$	Vlastnosti Carmichaelovho čísla M :
1105	=	$5 \cdot 13 \cdot 17$	• M je zložené z aspoň troch prvočísel
1729	=	$7 \cdot 13 \cdot 19$	• Žiadne prvočíslo sa v rozklade M neopakuje
2465	=	$5 \cdot 17 \cdot 29$	• Carmichaelove čísla sú zriedkavé – medzi 1 a 10^{21} je
2821	=	$7 \cdot 13 \cdot 31$	ich najviac 20,138,200. Pravdepodobnosť, že číslo
6601	=	$7 \cdot 23 \cdot 41$	z intervalu $\langle 1, 10^{21} \rangle$ je Carmichaelovo je
8911	=	$7 \cdot 19 \cdot 67$	$\frac{10^{21}}{2 \cdot 10^7} = 5 \cdot \frac{1}{10^{13}}$

Rabin Miller test prvočiselnosti:



Pravdepodobnostný test prvočiselnosti - RABIN – MILLER

1. Vyjadri p v tvare $p = 1 + 2^s \cdot r$, r nepárne
2. For $i = 1$ to t urob:
 - 2.1 Vyber náhodné číslo a také, že $2 \leq a \leq p - 2$
 - 2.2 Polož $y = a^r \pmod{p}$
 - 2.3 Ak $y \neq 1$ and $y \neq p - 1$ urob:
[
 j=1
 WHILE ($j \leq s - 1$) and ($y \neq p - 1$)
 $y = y^2 \pmod{p}$
 Ak $y = 1$, RETURN ZLOŽENÉ
 $j = j + 1$
 Ak $y \neq p - 1$ RETURN ZLOŽENÉ
]
3. RETURN PRVOČÍSLO S PRAVDEPODOBNOSŤOU $1 - \left(\frac{1}{4}\right)^t$

Lehmannov test

16. Symetricku kryptografiu - Feistelove kola (hlavne DES - komplet, GOST, IDEA- posledne dve hlavne dlzka bloku a dlzka kluca plus omacka)

Na šifrovanie aj dešifrovanie sa využíva jedený kľúč čož ma za následok nízku výpočtovú náročnosť avšak nutnosť zdieľania tajného kľúča.

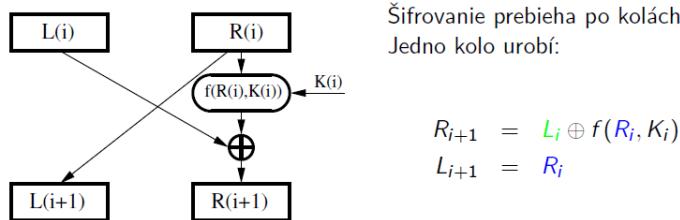
Všeobecný princíp symetrickej kryptografie

- ① A a B sa dohodnú na kryptosystéme
- ② A a B sa dohodnú na kľúči
- ③ A (resp. B) šifruje priamy text x ako $y = E_K(x)$
- ④ B (resp. A) dešifruje zašifrovaný text y ako $x = D_K(y)$

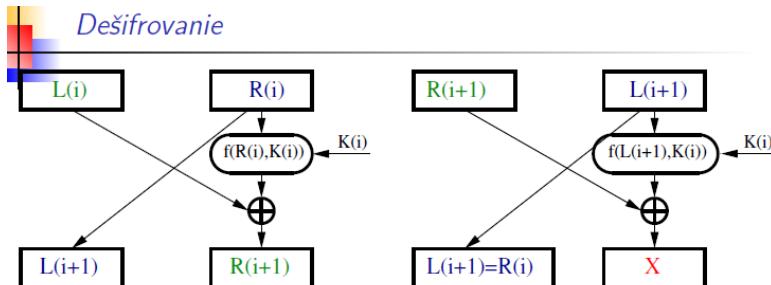
Kryptosystémy Feistelovho typu

Sú to systémy s blokovou šifrou – šifrujú sa celé bloky priameho textu. Pre kryptosystémy Feistelovho typu musí mať blok párny počet bitov.

Blok sa rozdelí na dve rovnako dlhé časti – ľavú L_i a pravú R_i .



Dešifrovanie:

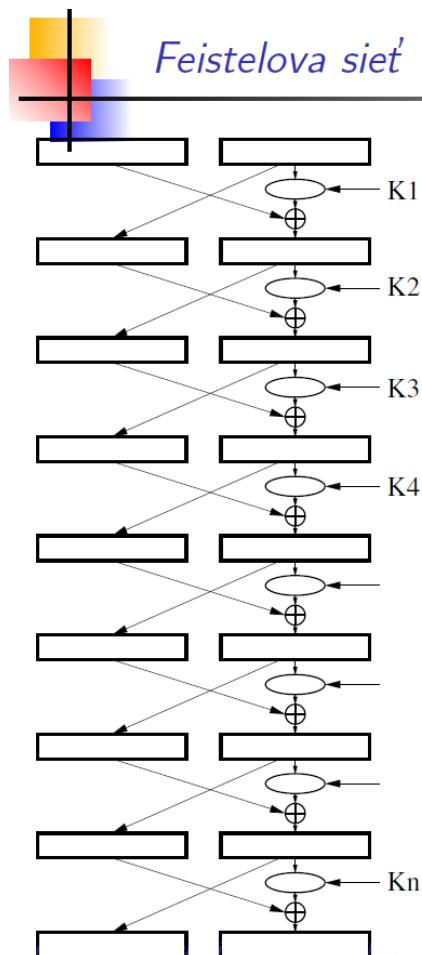


Počítajme X .

$$X = R_{i+1} \oplus f(L_{i+1}, K_i) = L_i \oplus \underbrace{f(R_i, K_i) \oplus f(R_i, K_i)}_{=0} = L_i \oplus f(R_i, K_i) = R_i$$

Dôsledok: Ak kolovému algoritmu vložíme kolový kľúč K_i , na miesto pravej časti L_{i+1} a na miesto ľavej časti R_{i+1} , dostaneme na jeho výstupe na pravej a ľavej časti poradie pôvodné L_i a R_i . Ten istý kolový algoritmus (s prehodenou ľavou a pravou stranou a tým istým kolovým kľúčom) teda môžeme použiť ako inverznú funkciu.

Feistelová siet' :



Feistelova siet' je iterované niekoľkonásobné opakovanie kolových algoritmov, každý s iným kolovým kľúčom.

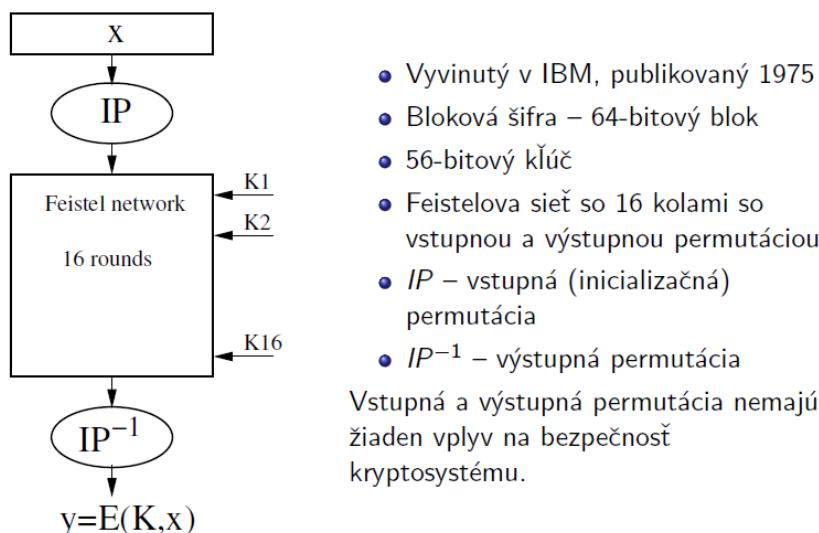
Dešifrovanie sa urobí tou istou sieťou, ktorej sa na vstup vloží zašifrovaný text s poradím kolových kľúčov K_n, K_{n-1}, \dots, K_1 a so zameneným poradím pravej a ľavej časti.

Dôležité: Práve popísaný inverzny mechanizmus nezáleží na tvare funkcie $f(R_i, K_i)$.

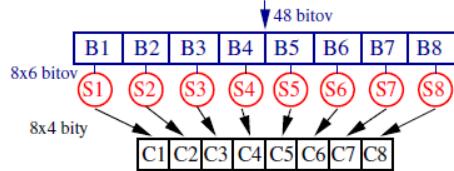
Na funkcii $f(R_i, K_i)$ však podstatne závisia kryptografické vlastnosti Feistelovej siete.

5/

DES:



DES – Použitie S-boxov



- S-box je tabuľka so štyrmi riadkami a šestnástimi stĺpcami.
- Riadky sú číslované od 0 do 3, stĺpce sú číslované od 0 do 15.
- DES používa 8 S-boxov, bloku B_i je priradený S-box S_i .
- Každé B_i je 6-bitové číslo $b_1 b_2 b_3 b_4 b_5 b_6$ a predstavuje adresu príslušného štvorbitového čísla C_i v S-boxe S_i .

DES – Adresovanie v S-boxe

Adresa sa vypočíta takto:

Nech $B_1 = b_1 b_2 b_3 b_4 b_5 b_6$.

$b_1 b_6$ je číslo riadku, $b_2 b_3 b_4 b_5$ je číslo stĺpca v príslušnom S-boxe.
(Riadky i stĺpce sú číslované od 0 po 3 resp. od 0 po 15.)

S-box 1:

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Príklad:

$B_1 = 101011$. $b_1 b_6 = (11)_2 = 3$, $b_2 b_3 b_4 b_5 = (0101)_2 = 5$.

V S-boxe S_1 je v riadku 3 a stĺpci 5 číslo 9 (pozor, riadky a stĺpce sa číslujú od 0), ktorého binárny rozvoj je 1001. Je teda

$$S_1(B_1) = S_1(101011) = 1001 = C_1.$$

DES – Pravidlá tvorby S-boxov

Jediná nelineárna časť DESu je v S-boxoch. Na nich závisí odolnosť DESu.

- Každý riadok je permutáciou čísel 0 – 15.
- Žiaden S-box nie je lineárnej alebo afinnej funkciou vstupov
- Zmena jedného vstupného bitu S-boxu spôsobí zmenu aspoň dvoch bitov výstupu
- Pre každý S-box a pre každé šestbitové x $S(x)$ a $S(x \oplus 001100)$ sa líšia aspoň v dvoch bitoch
- Pre každý S-box a pre každé šestbitové x a pre ľubovoľné bity $r, s \in \{0, 1\}$ $S(x) \neq S(x \oplus 11rs00)$.
- Ak fixujeme hodnotu jedného vstupného bitu, potom počet vstupných hodnôt, pre ktoré je ľubovoľný určený bit rovný 0 (alebo 1), je medzi 13 a 19.

DES – Generovanie kolových kľúčov:



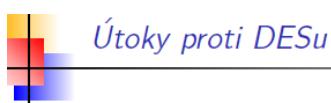
Kľúč pre systém DES je 56-bitový, ale ukladá sa ako 64 bitov s tým, že v každom bajte je 7 bitov kľúča a jeden kontrolný bit doplňujúci bajt na napárnu paritu. Po odstránení paritných bitov sa získa 56 bitov kľúča, ktorých poradie sa zmení podľa permutácie PC-1.

Potom sa 56 bitov kľúča rozdelí na dve 28-bitové časti C_0, D_0 , na každú z nich sa postupne aplikuje Ľavý rotačný posun $LS_1, LS_2, \dots, LS_{16}$. Pre $i = 1, 2, 9, 16$ je LS_i posun o jedno miesto, inak o 2 miesta.

Získa sa tak postupnosť

$C_1D_1, C_2D_2, \dots, C_{16}D_{16}$ 56 bitových reťazcov, z ktorých operácia PC-2 výberom 48 bitov a ich permutáciou vytvorí postupne kľúče K_1, K_2, \dots, K_{16} .

Útoky proti DESu:



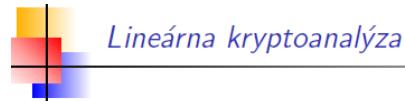
Útok hrubou silou.

Počet kľúčov 2^{56} sa ukazuje v dnešnej dobe malý. Podarilo sa prelomiť DES distribuovaným výpočtom na Internete.

Diferenciálna kryptoanalýza.

Je to útok typu "chosen plaintext attack". Šifrovaciemu algoritmu s neznámym kľúčom sa dávajú šifrovať dvojice priamych textov P_1, P_2 s určitou diferenciou $P_1 \oplus P_2$ a na základe diferencie príslušných zašifrovaných textov sa usudzuje na niektoré vlastnosti kľúča.

Lineárna kryptoanalýza:



Lineárna kryptoanalýza.

Ak pre priamy text $x_1x_2 \dots x_{64}$, kľúč $k_1k_2 \dots k_{56}$ a pre príslušný zašifrovaný text $y_1y_2 \dots y_{64}$ platí

$$\bigoplus_{i=1}^{64} a_i x_i \oplus \bigoplus_{i=1}^{64} b_i y_i = \bigoplus_{i=1}^{56} c_i k_i$$

s pravdepodobnosťou rôznou od $\frac{1}{2}$, dá sa to využiť pri kryptoanalýze.

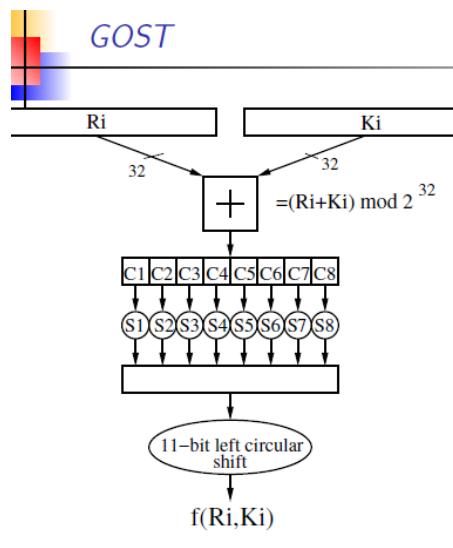
Pre DES platí

$$x_{17} \oplus y_3 \oplus y_8 \oplus y_{14} \oplus y_{25} = K_{i,26}$$

s pravdepodobnosťou $\frac{1}{2} - \frac{5}{16} = \frac{3}{16}$.

Na základe tohto faktu bol navrhnutý chosen plaintext attack analyzujúci priemerne 2^{43} známych priamych textov, ktorý odhalil kľúč za 50 dní práce 12 počítačov HP9735 (v roku 1994).

Gost:



Sovietsky kryptovací systém používaný v časoch studenej vojny.

Bloková šifra.

64 bitový blok, 256 bitový kľúč.

Feistelova sieť s 32 kolami.

S-boxy sú jednoriadkové tabuľky obsahujúce permutácie čísel $0, 1, \dots, 15$.

IDEA:



IDEA – Špecifikácia

IDEA – International Data Encryption Algorithm (Xuejia Lai and James Massey) - 1992.

Je patentovaný, US patent vyprší 7.1.2012.

Bloková šifra – blok 64 bitov

Kľúč 128 bitov.

64-bitový blok sa rozdelí na 4 16-bitové časti x_1, x_2, x_3, x_4 , s ktorými s urobí 8 kôl algoritmu plus záverečné "polovičné kolo."

V kolách sa používajú tieto operácie:

\oplus – XOR po bitoch

\boxplus – sčítanie mod 2^{16}

\odot – násobenie mod $(2^{16} + 1)$ pričom sa 16-bitové slovo pozostávajúce zo samých 0 považuje za reprezentáciu čísla 2^{16} .

Jedno kolo algoritmu IDEA

