Pokročilá kryptologie

Symetrická kryptografie

prof. Ing. Róbert Lórencz, CSc.

České vysoké učení technické v Praze, Fakulta informačních technologií
Katedra informační bezpečnosti

Obsah přednášky

- Bloková šifra
- DES
- AES

Blokové symetrické šifry (1)

Bloková symetrická šifra

- Nechť A je abeceda q symbolů, $t \in N$ a $M = C$ je množina všech řetězců délky t nad A. Nechť K je množina klíčů.
- Bloková šifra: šifrovací systém (M, C, K, E, D), kde E a D jsou zobrazení, definující pro každé $k \in K$ transformaci zašifrování E_k a dešifrování D_k tak, že zašifrování bloků OT $m_1, m_2, m_3, \ldots$, kde $m_i \in M$ pro každé $i \in N$, probíhá podle vztahu $c_i = E_k(m_i)$ pro každé $i \in N$.
- Dešifrování probíha podle vztahu $m_i = D_k(c_i)$ pro každé $i \in N$.
- Pro blokovou šifru je podstatné, že všechny bloky OT jsou šifrovány toutéž transformací a všechny bloky ŠT jsou dešifrovány toutéž transformací.

Blokové symetrické šifry (2)

OT	ŠT
0000	0111
0001	1110
0010	0000
0011	0100
0100	1001
0101	1101
0110	0101
0111	0001
1000	1100
1001	0010
1010	0110
1011	1111
1100	1010
1101	1011
1110	0011
1111	1000

ŠT	OT
0000	0010
0001	0111
0010	1001
0011	1110
0100	0011
0101	0110
0110	1010
0111	0000
1000	1111
1001	0100
1010	1100
1011	1101
1100	1000
1101	0101
1110	0001
1111	1011

Bloková šifra

- Převádí n-bitový OT na n-bitový ŠT.
- Blok o délce n-bitů vytváří 2^n různých bloků OT.
- Transformuje 2^n bloků OT na 2^n bloků ŠT.
- Pro $n = 4$ je 16 b OT převedeno na 16 b ŠT → jednoduché prolomení.

- U jednoduché „substituce“ můžeme 2. sloupec považovat za klíč o délce 64 b.
- Pro n-bitovou substituční šifru je klíč $n \times 2^n$ bitový.
- Pro 64 bitovou šifru je klíč $64 \times 2^{64} = 2^{70}$ b.

- Problém velkých klíčů řeší → Feistelová bloková šifra

Blokové symetrické šifry (3)

Feistelová bloková šifra

- Řeší problém velkých klíčů (její struktura).
- Aproximuje ideální blokovou šifru po velká n.
- Je složená šifra, která využíva posloupnost dvou resp. více šifer pro dosažení kryptografický silnější šifry.
- Redukuje délku klíče ideální blokové šifry.
- Využívá střídavě substituci a transpozici.
- Má parametry:
  - velikost bloku,
  - délka klíče,
  - počet rund,
  - algotitmus generování klíčů rund,
  - složitost operací v rundě.
- Větší blok, větší délka klíče, větší počet rund, složitější algoritmus a složitější operace v rundě zvyšuji bezpečnost, ale snižuji rychlost šifrovaní a dešifrování.

Blokové symetrické šifry (3)

Feistelová bloková šifra (1)

- šifrovací systém LUCIFER (projekt H. Feistela), předchůdce DES (Data Encryption Standard), má 64 b blok a 128 b klíč
- v současné době se přechází na blok 128 bitů, který používá standard AES (není v něm použit princip Feistela).
- blokové symetrické šifry využívající principy algoritmů Feistelova typu umožňují postupnou aplikací relativně jednoduchých transformací na bázi nelineárních posuvných registrů vytvořit složitý kryptografický algoritmus.
- tento přístup je využíván také v jiných oblastech: zabezpečovací kódy

Blokové symetrické šifry (4)

Definice – Feistelův kryptosystém

Nechť množina zpráv M je složená ze všech možných $2n$-tic V_{2n} prostoru a prostor K klíčů k je tvořen všechnymi možnými h-ticeme funkci $\{f_1, f_2, \ldots, f_h\}$, kde $f_i : V_n \to V_n$ pro každé $i = 1, 2, \ldots, h$ a kde $C = V_{2n}$ je prostor zašifrovaných textů. Zobrazení $T_k : K \times V_{2n} \to V_{2n}$, definované rekurentně vztahy

$$\begin{aligned} m_{i+1} &= m_{i-1} + f_i(m_i), \quad \text{pro} \quad i = 1, 2, \ldots, h \\ T_k(m) &= (m_h \; m_{h+1}), \end{aligned}$$

kde $m = (m_0 \; m_1) \in M$, definuje Feistelův kryptosystem.

Příklad:

Pro $m = (m_0 \; m_1) = (\texttt{1001 1101})$ dostáváme postupně:
$c_1 = (\texttt{1101 1110})$, kde $m_2 = \texttt{1001} \oplus f_1(\texttt{1101}) = \texttt{1001} \oplus \texttt{0111} = \texttt{1110}$
$c_2 = (\texttt{1110 0000})$, kde $m_3 = \texttt{1101} \oplus f_2(\texttt{1110}) = \texttt{1101} \oplus \texttt{1101} = \texttt{0000}$
f_1 a f_2 – premutační funkce (konfuze), $\oplus$ – funkce xor (difuze)

DES (1)

Algoritmus DES

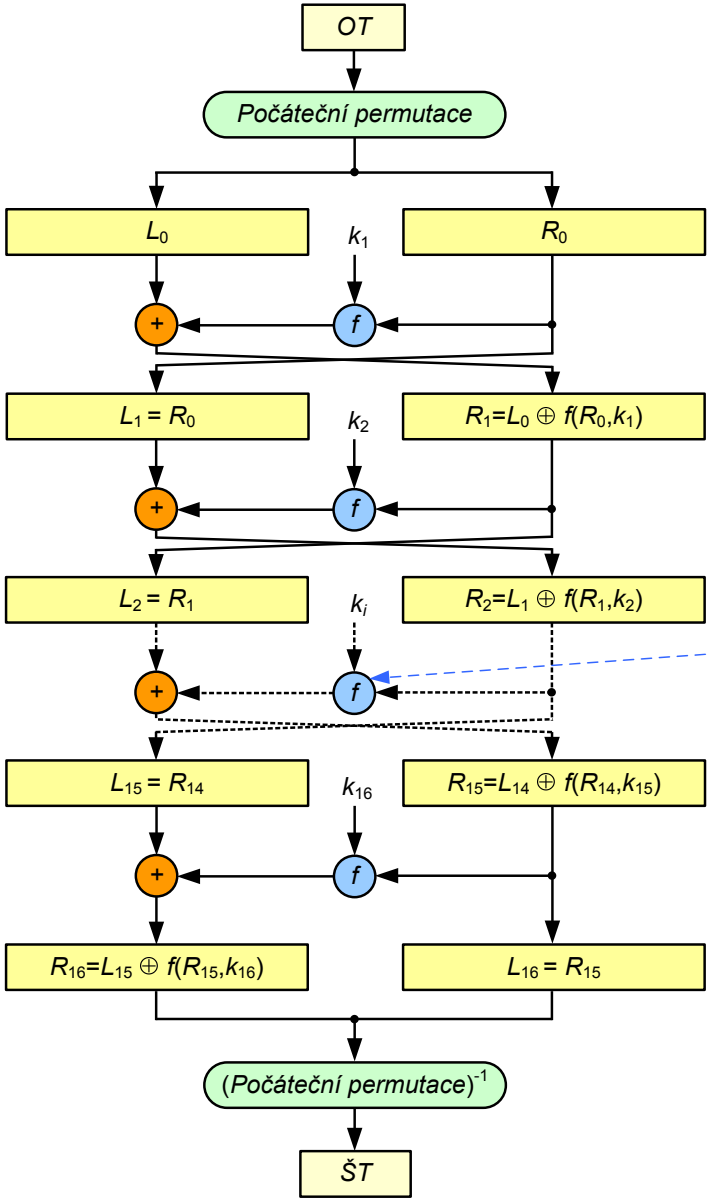
- Veřejná soutěž (1977): šifrovací standard (FIPS 46-3) v USA pro ochranu citlivých, ale neutajovaných dat ve státní správě.
- Součást průmyslových, internetových a bankovních standardů.
- 1977: varování – příliš krátký klíč 56b, který byl do původního návrhu IBM zanesen vlivem americké tajné služby NSA.
- DES – intenzivní výzkum a útoky $\Rightarrow$ objeveny teoretické negativní vlastnosti jako: tzv. slabé a poloslabé klíče, komplementárnost a teoreticky úspěšná lineární a diferenciální kryptoanalýza.
- V praxi jedinou zásadní nevýhodou je pouze krátký klíč.
- 1998: stroj – DES-Cracker, luštící DES hrubou silou.
- DES jako americký standard skončil (jen v "dobíhajících"systémech a kvůli kompatibilitě) a místo něj: Triple-DES, (1999, FIPS 46-3).
- 3DES (3xDES) - až 3x delší klíč.
- Od 26. 5. 2002 – šifrovací standard nové generace AES.

DES (2)

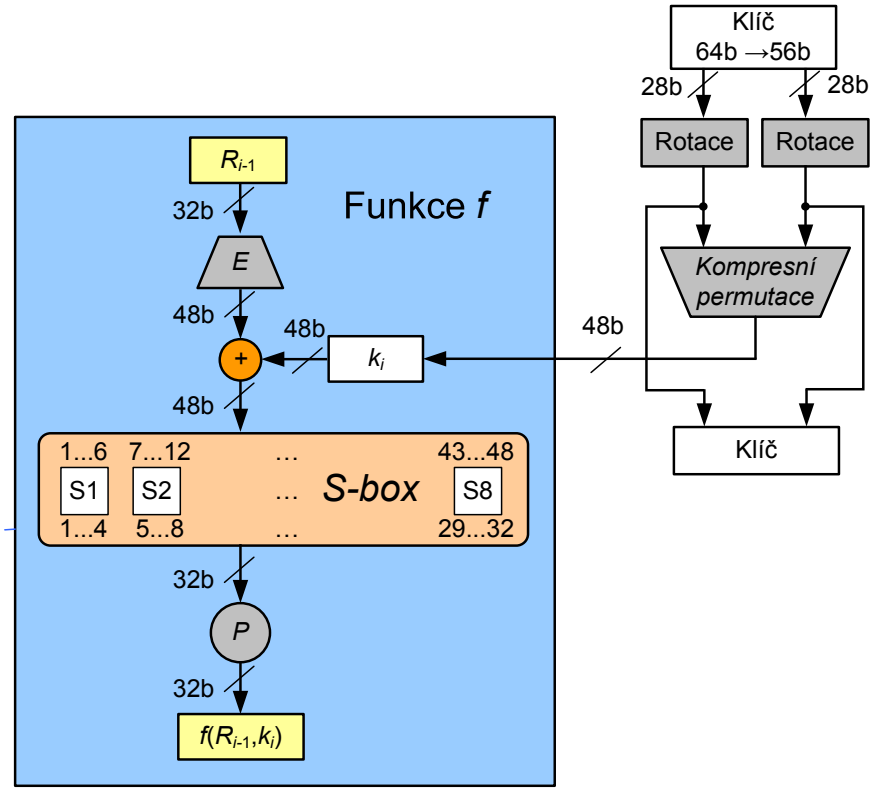
Charakteristika DES (1)

- DES je iterovaná šifra typu $E_{k_{16}}(E_{k_{15}}(\ldots(E_{k_1}(m_i)\ldots)))$.
- Používá 16 rund a 64b bloky OT a ŠT. Šifrovací klíč k má délku 56 b (vyjadřuje se ale jako 64b číslo, kde každý 8. bit je bit parity)
- 56b klíč k je v inicializační fázi nebo za chodu algoritmu expandován na 16 rundovních klíčů k_1 až k_{16}, které jsou řetězci 48 bitů, každý z těchto bitů je některým bitem původního klíče k.
- Místo počátečního zašumění OT se používá bezklíčová pevná permutace *Počáteční permutace* a místo závěrečného zašumění permutace k ní inverzní $(\textit{Počáteční permutace})^{-1}$.
- Po *počáteční permutaci* je blok rozdělen na dvě 32b poloviny (L_0, R_0). Každá ze 16 rund $i = 1, 2, \ldots, 16$ transformuje (L_i, R_i) na novou hodnotu $(L_{i+1}, R_{i+1}) = (R_i, L_i \oplus f(R_i, k_{i+1}))$, liší se jen použitím jiného rundovního klíče k_i.
- Ve smyslu definice Feistelova kryptosystému je v tomto případě $h = 16$ a $2n = 64$.

DES (3)



Algoritmus DES



DES (4)

Charakteristika DES (2)

- Po 16. rundě dochází ještě k výměně pravé a levé strany: $(L_{16}, R_{16}) = (R_{15}, L_{15} \oplus f(R_{15}, k_{16}))$ a závěrečné permutaci $(\textit{Počáteční permutace})^{-1}$.
- Dešifrování probíhá stejným způsobem jako zašifrování, pouze se obrátí pořadí výběru rundovních klíčů.

Rundovní funkce *f*

- Rundovní funkce se skládá z binárního načtení klíče k_i na vstup. 48b klíč k_i je vytvořen po kompresi ze 2 28b rotovaných částí původního klíče k, kde počet bitů rotace je závislý na čísle rundy.
- Tento klíč k_i je dál xorován s expandovanou 32b částí R_{i-1}, která je expanzně permutovaná v bloku E z 32b na 48b. Tato operace kromě rozšíření daného 32b slova také permutuje bity tohoto slova tak, aby se dosáhlo lavinového efektu.

DES (5)

Charakteristika DES (3)

- Následně je prováděna pevná, nelineární substituce na úrovni 6b znaků do 4b znaků s následnou transpozicí na úrovni bitů. Těmito operacemi se dosahuje dobré difúze i konfúze.
- Použité substituce se nazývají substituční boxy: S-boxy, jsou jediným nelineárním prvkem schématu. Pokud bychom substituce vynechali, mohli bychom vztahy mezi ŠT, OT a klíčem popsat pomocí operace binárního sčítání $\oplus$, tedy lineárními vztahy.
- Tato nelinearita je překážkou jednoduchého řešení rovnic, vyjadřující vztah mezi OT, ŠT a K.
- Následně 32b výsledné slovo z S-boxu je permutováno c bloku P. Tato permutace převádí každý vstupní bit do výstupu, kde žádbý vstupní bit se nepoužije $2\times$.
- Nakonec se výsledek permutace sečte modulo 2 s levou 32b polovinou a začne další runda.

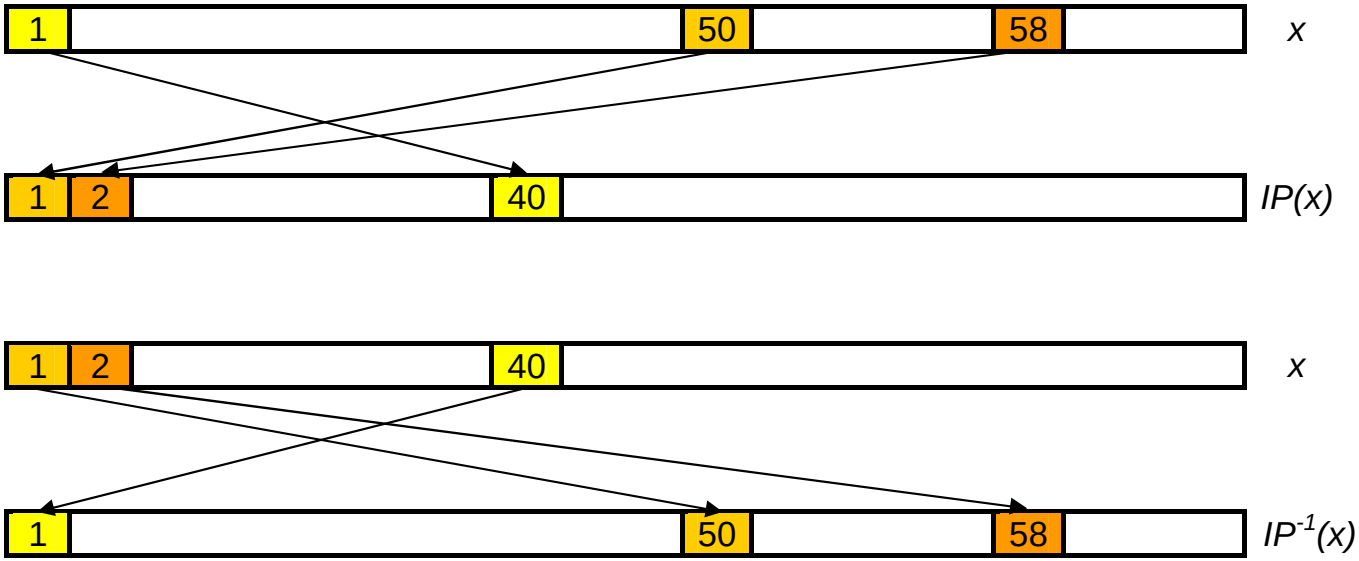
DES (6)

Vnitřní struktura DES (1)

Počátečni permutace - IP a koncová permutace - IP^{-1}

- Nezvyšují bezpečnost DES.
- Lehce implementovane v HW, ale ne v SW.
- Původ permutací je pravděpodobně v snaze přeuspořádat OT do podoby, která je lépe dále zpracovatelná (souviselo to s technologickou úrovní, dnes to již neplatí).

Příklad:



DES (7)

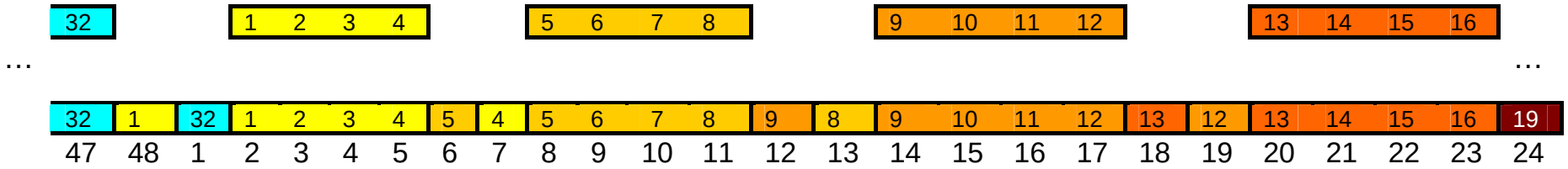
Vnitřní struktura DES (2)

Počátečni permutace - *IP* a koncová permutace - IP^{-1}

Počáteční permutace - IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Koncová permutace - IP^{-1}							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	28	26
33	1	41	9	49	17	57	25

Realizace expanzní funkce E



DES (8)

Vnitřní struktura DES (3)
Substituce pomocí S-boxů

Příklad dekódovaní vstupu „011011“

msb	1. – 4. bit				lsb
0	1	1	0	1	1

S-box S_5

		1. – 4. bit															
msb	lsb	**0000**	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
0	**0**	0010	1100	0100	0001	0111	1100	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
0	**1**	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1100	0011	1001	1000	0110
1	**0**	0100	0010	0001	1011	1100	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
1	**1**	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1100	0100	0101	0011

- Jediny nelineární element DES, který provádí konfuzi
- Výběr převodních tabulek nebylo nikdy úplně odhalen

DES (9)

Vnitřní struktura DES (3)

S-boxy byli navržený podle nasledujících kritérií:

1. Každý S-box má 6 vstupních bitů a 4 výstupní.
2. Žádný z výstupních bitů není lineární kombinací vstupních bitů.
3. Když vstupní msb = lsb a 4 prostřední bity se mění, potom každá možná 4-bitová výstupní hodnota se vyskytne jenom jednou.
4. Pokud jsou 2 vstupy do S-boxu rozílné jenom právě v jednom bitě, potom jejích výstup musí být rozdílny minimálně ve 2 bitech.
5. Pokud 2 vstupy do S-boxu jsou rozdílné ve 2 středních bitech, potom jejích výstup musí být rozdílný minimálně ve 2 bitech.
6. Pokud 2 vstupy do S-boxu se liši v prvních 2 bitech a jsou identické v posledních 2 bitech, potom musí být výstup rozdílný.
7. Pro jakýkoliv nenulový 6-bitový rozdílný vstupy je ne víc než 8 z 32 dvojíc vstupu
8. Nějaká kolize (žádný rozdil ve výstupech) 32-bitových výstupů 8 S-boxů je možná jenom pro 3 sousední S-boxy.

Vnitřní struktura DES (4)

Permutace P

P							
16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

- 32-bitový vstup a 32-bitový výstup $\rightarrow$
- příma permutace – žádný bit není vynechán
- zavádí difuzi

DES (10)

Transformace klíče (1) Transformace klíče zahrnuje:

- Upraví se 64-bitový klíč na 56-bitový vynecháním každého 8. bitů parity.
- Vykoná se permutace klíče
- 56-bitový klíč se rozdělí na 28-bitové polovice, které s posouvají rotaci o 1 (1., 2. 9. 16. runda) nebo 2 bity vlevo (ostatní rundy).
- Rotace zabezpečuje vytvoření různých klíčů pro každou rundů.
- Kompresní permutace vybere z 56 bitů 48.

Transformace a permutace klíče							
57	49	41	33	25	17	9	1
58	50	42	34	26	18	10	2
59	51	43	35	27	19	11	3
60	52	44	36	63	55	47	39
31	23	15	7	62	54	46	38
30	22	14	6	61	53	45	37
29	21	13	5	28	20	12	4

Kompresní permutace klíče							
14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

TripleDES (1)

TripleDES

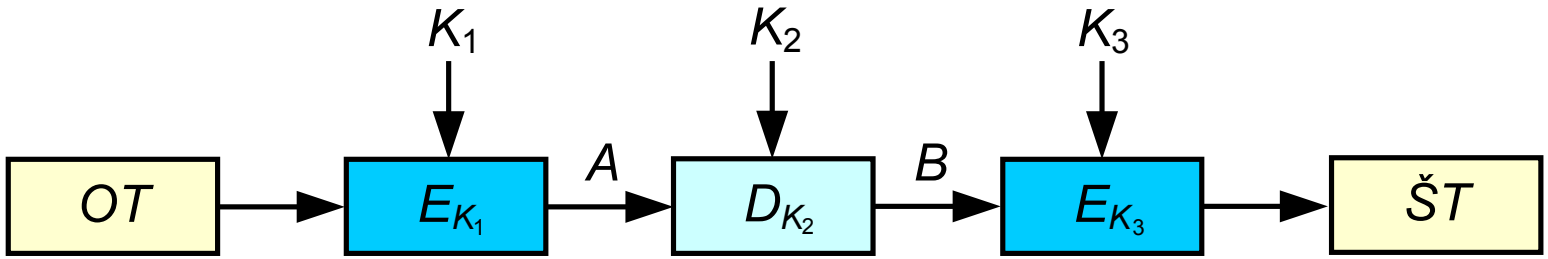
- TripleDES (3DES) prodlužuje klíč originální DES tím, že používá DES jako stavební prvek celkem 3× s 2 nebo 3 různými klíči.
- Nejčastěji se používá varianta EDE této šifry, která je definována ve standardu FIPS PUB 46-3 (v bankovní normě X9.52).
- Vstupní data OT jsou zašifrována podle vztahu $ŠT = E_{K_3}(D_{K_2}(E_{K_1}(OT)))$, kde K_1, K_2 a K_3 jsou buď 3 různé klíče nebo $K3 = K1$. Varianta EDE byla zavedena z důvodu kompatibility → při rovnosti všech klíčů 3DES = DES.
- Klíč 3DES je tedy buď 112 bitů (2 klíče) nebo 168 bitů (3 klíče). 3DES je spolehlivá → klíč je dostatečně dlouhý a teoretickým slabinám (komplementárnost, slabé klíče) se dá předcházet ⇒
- 3DES a AES → platný oficiální standard nahrazující DES.
- 3DES lze, jako jakoukoliv jinou blokovou šifru, použít v různých operačních modech (CBC mod ⇒ 3DES-EDE-CBC).

TripleDES (2)

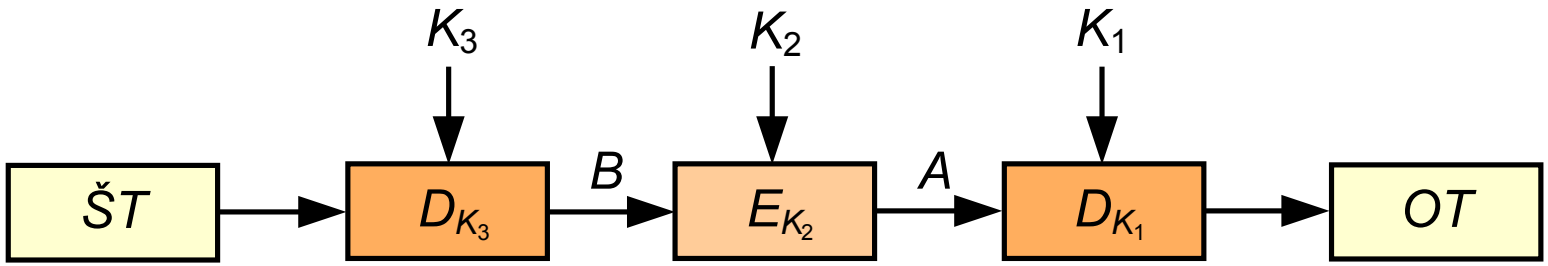
Algoritmus 3DES

Šifrování

3DES-EDE



Dešifrování



$K_1 \neq K_2 \neq K_3 \Rightarrow$ 3 x klíč = 168b $\rightarrow$ 3DES
$K_1 = K_3 \neq K_2 \Rightarrow$ 2 x klíč = 112b $\rightarrow$ 3DES
$K_1 = K_2 = K_3 \Rightarrow$ 1 x klíč = 56b $\rightarrow$ DES

AES (1)

AES (1)

- Po útocích hrubou silou na DES, americký standardizační úřad připravil náhradu - Advanced Encryption Standard (AES).
- 2.1. 1997 výběrové řízení na AES – 15 kandidátů.
- Z 5 finalistů byl vybrán algoritmus Rijndael [rájndol] (autoři J. Daemen a V. Rijmen).
- Jako AES byl přijat s účinností od 26. května 2002 a byl vydán jako standard v oficiální publikaci FIPS PUB 197.
- AES je bloková šifra s délkou bloku 128 bitů, čímž se odlišuje od současných blokových šifer, které měly blok 64 bitový.
- AES podporuje tři délky klíče: 128, 192 a 256 bitů $\Rightarrow$ se částečně mění algoritmus (počet rund je po řadě 10, 12 a 14).
- Větší délka bloku a klíče zabraňují útokům, které byly aplikované na DES. AES nemá slabé klíče, je odolný proti známým útokům a metodám lineární a diferenciální kryptoanalýzy.

AES (2)

- Algoritmus zašifrování i odšifrování se dá výhodně programovat na různých typech procesorů, má malé nároky na paměť i velikost kódu a je vhodný i pro paralelní zpracování.
- AES bude pravděpodobně platným šifrovacím standardem několik desetiletí a bude mít obrovský vliv na počítačovou bezpečnost.
- Označíme-li délku klíče N_k jako počet 32b slov, máme $N_k = 4, 6$ a 8 pro délku klíče 128, 192 a 256 bitů.
- AES je iterativní šifra, počet rund N_r se mění podle délky klíče: $N_r = N_k + 6$, tj. je to 10, 12 nebo 14 rund.
- Tato skutečnost odráží nutnost zajistit konfúzi vzhledem ke klíči. Algoritmus pracuje s prvky Galoisova tělesa $GF(2^8)$ a s polynomy, jejichž koeficienty jsou prvky z $GF(2^8)$. Bajt s bity $(b_7, ..., b_0)$ je proto chápán jako polynom $b_7 x^7 + \cdots + b_1 x^1 + b_0$ a operace "násobení bajtů"odpovídá násobení těchto polynomů modulo $m(x) = x^8 + x^4 + x^3 + x^1 + 1$.

AES (3)

Rundovní klíče

- Rundovní klíče AES využívá $4 + N_r \times 4$ rundovních 32b klíčů, které se definovaným způsobem derivují ze šifrovacího klíče.
- Před zahájením 1. rundy zašifrování se provede úvodní zašumění, kdy se na OT naxorují první 4 rundovní klíče (128b na 128b)$\Rightarrow$
- N_r shodných rund (s výjimkou poslední, kdy se neprovede operace MixColumns), při kterých výstup z každé předchozí rundy slouží jako vstup do rundy následující. Tím dochází k postupnému mnohonásobnému zesložiťování výstupu.

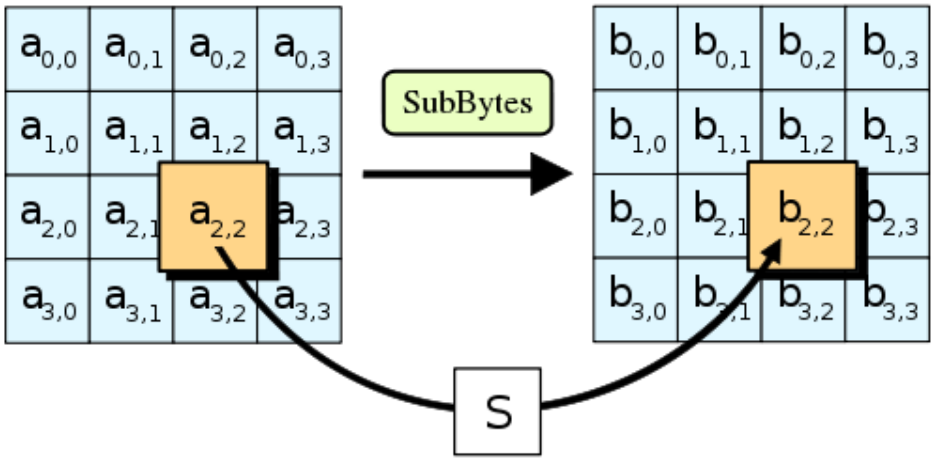
Runda (1)

- Na počátku každé rundy se vždy vstup (16 B) naplní postupně shora dolů a zleva doprava po sloupcích do matice 4x4 B $\mathbf{A} = (a_{ij})$ $i, j = 0, 1, 2, 3$.
- Na každý bajt matice **A** se zvlášť aplikuje substituce, daná pevnou substituční tabulkou SubBytes.

AES (4)

Substitute bytes

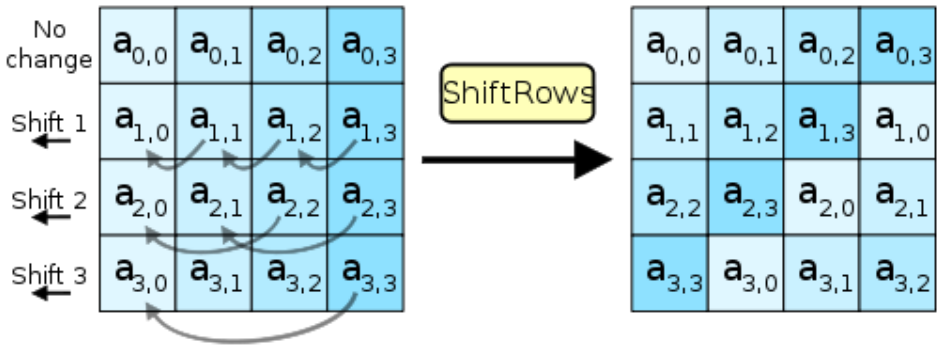
- Transformace v přímem a inverzním tvaru pro přímou a inverzní substituci
- S-box organizován ve tvaru matice 16x16 (4b x 4b) pro transformaci vsech 8-bitových hodnot
- v matici je číslo řádku určeno 4 vyššími bity aktualného bytu a číslo sloupce je určeno 4 nižšími bity aktualneho bytu
- S-box je vyplněn tak, že každý byt se dá vypočítat jako multiplikatívní inverze v $GF(2^8)$ modulo $m(x) = x^8 + x^4 + x^3 + x + 1$ a nula se tranformuje sama na sebe



AES (5)

Shift Rows

- Transformace se aplikuje na řádky s jednotlivými byty
- 1. řádek zůstava beze změny, 2. řádek se posune o 1 místo do leva, 3. řádek se posune o 2 místa do leva a 3. řádek se posune o 3 místa doleva
- inverzní transformace vykonáva stejný posuv řádků, ale do prava
- Vzhledem k tomu, že první 4 byty OT jsou zapsány v prvním sloupci atd. jsou každého sloupce distribuované do 4 bytu různých sloupců, dochází úplnemu promíchání bytů OT - dochází k transpozici na úrovni bajtů.



AES (6)

Mix Columns

- Dále se na každý jednotlivý sloupec matice aplikuje operace MixColumns, která je substitucí 32 bitů na 32 bitů. Tuto substituci lze však popsat lineárními vztahy – všechny výstupní bity jsou nějakou lineární kombinací vstupních bitů. Označíme-li jednotlivé bajty v rámci daného sloupce matice **A** (shora dolů) jako a_0 až a_3, pak výstupem budou jejich nové hodnoty b_0 až b_3, podle vztahů

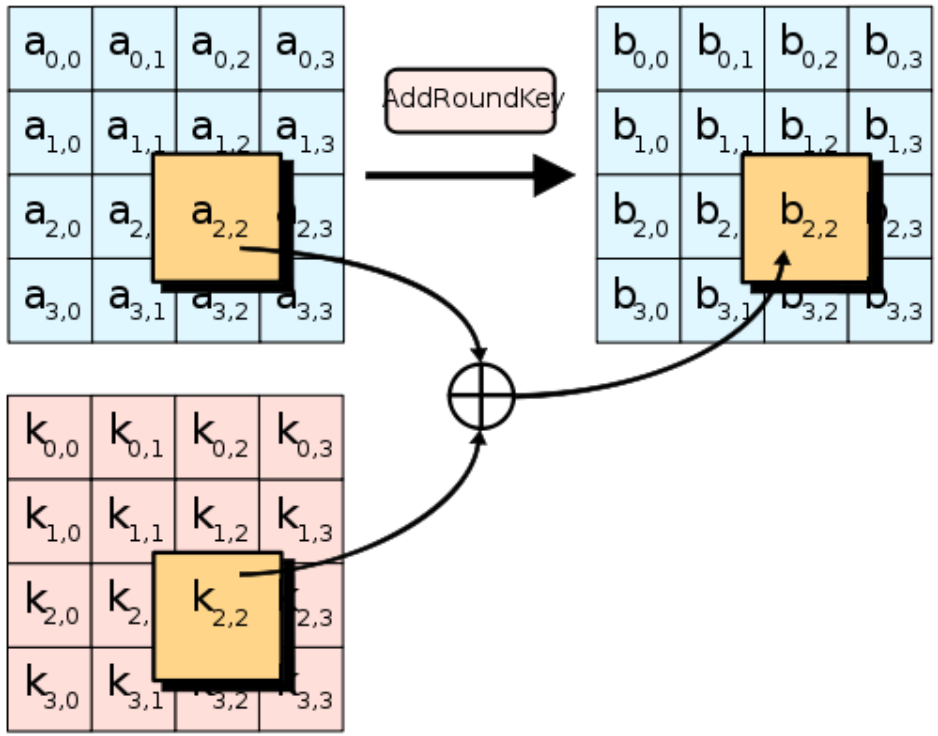
$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} '02' & '03' & '01' & '01' \\ '01' & '02' & '03' & '01' \\ '01' & '01' & '02' & '03' \\ '03' & '01' & '01' & '02' \end{pmatrix} \times \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix}$$

- Násobení je násobení prvků $GF(2^8)$. Konstantní prvky tohoto pole jsou vyjádřeny hexadecimálně.

AES (7)

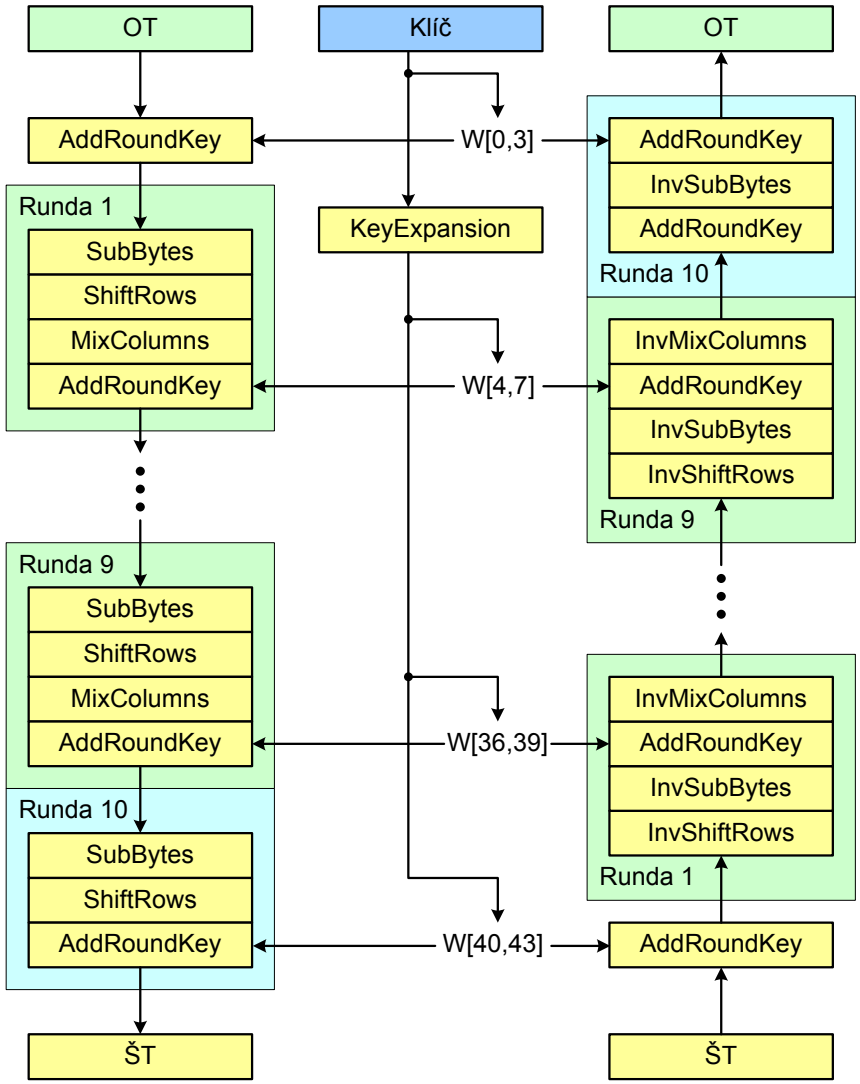
Add Round Key

- Jako poslední operace rundy se vykoná transformace AddRoundKey, v rámci níž se na jednotlivé sloupce matice **A** zleva doprava naxorují 4 odpovídající rundovní klíče. Tím je jedna runda popsána a začíná další. Po poslední rundě se ŠT jen vyčte z matice **B**.



Algoritmus AES

AES – Struktura šifrování a dešifrování



AES (8)

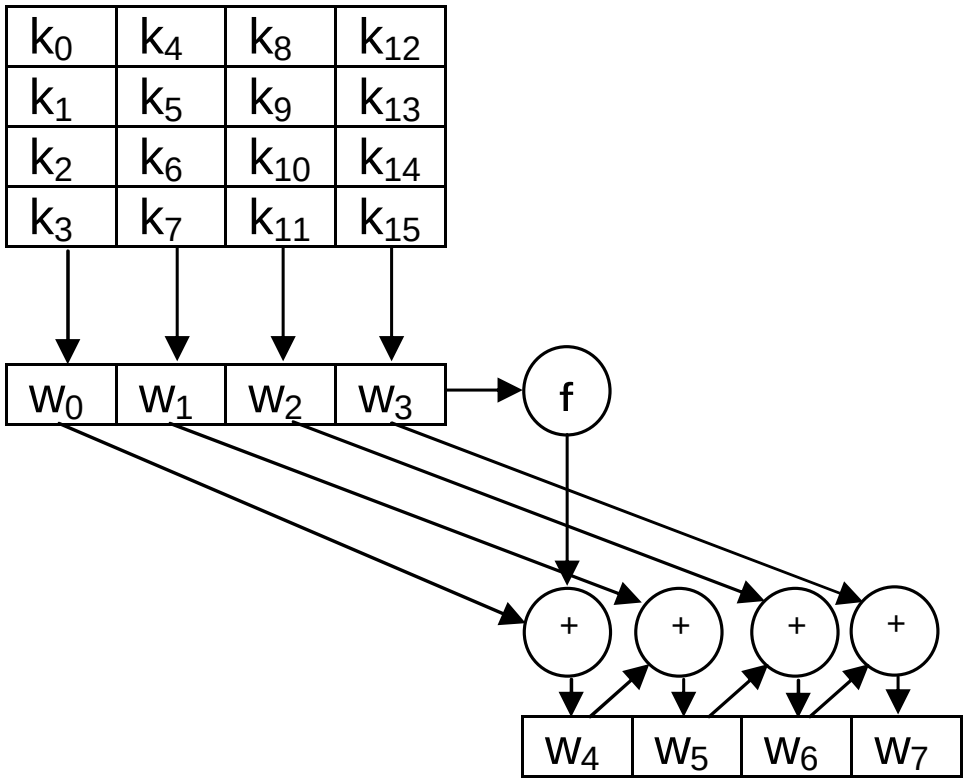
- Při odšifrování se používají operace inverzní k operacím, použitým při zašifrování, neboť všechny jsou reverzibilní.
- Nelinearity v AES se objevují pouze v substituci SubBytes. V roce 2002 bylo zjištěno, že vzájemné vztahy výstupních (y1,..,y8) a vstupních (x1,..,x8) bitů lze popsat implicitními rovnicemi f(x1,..,x8,y1,..,y8) = 0 pouze druhého řádu.

AES (9)

Key Expansion (1)

- Operace realizuje expanzi klíče s délkou 16 bytů = 128 bitů.
- Klíč ma 4 slova s délkou 4 byty.
- Při počte rund 10 je potřená expanze na 44 slov

Proces expanze klíče



AES (10)

Key Expansion (2)
Operace tvořící funkci f

- Operace Rot Word realizuje cyklický posuv bytů slova w_3 o jednu pozici vlevo.
- Operace Sub Word realizuje substitucí posunutých bytů podle předpisu v S-boxe.
- Výsledek předchozích operací se XORuje s konstantou $RC(j)$, které hodnoty jsou definované pro každou rundu.
- $RC(j) = [B_j, 0, 0, 0]$

i	1	2	3	4	5	6	7	8	9	10
B_j	01	02	04	08	10	20	40	80	1B	36