Advanced cryptology

Symmetric Cryptography

prof. Ing. Róbert Lórencz, CSc.

České vysoké učení technické v Praze, Fakulta informačních technologií
Katedra počítačových systémů

Příprava studijních programů Informatika pro novou fakultu ČVUT je spolufinancována Evropským sociálním fondem a rozpočtem Hlavního města Prahy v rámci Operačního programu Praha — adaptabilita (OPPA) projektem CZ.2.17/3.1.00/31952 – „Příprava a zavedení nových studijních programů Informatika na ČVUT v Praze“.
Praha & EU: Investujeme do vaší budoucnosti

Tato přednáška byla rovněž podpořena z prostředků projektu č. 347/2013/B1a Fondu rozvoje vysokých škol Ministerstva školství, mládeže a tělovýchovy

Obsah přednášky

- Block ciphers
- DES
- AES

Block symmetric ciphers (1)

Block cipher

- Let A be an alphabet of q symbols, $t \in N$ a $M = C$ is a set of all chains with length t over A. Let K be a set of keys.
- Block cipher is an encryption system (M, C, K, E, D), where E and D are representations, defining for every $k \in K$ encryption transformation E_k and decryption D_k so encryption of blocks of PT $m_1, m_2, m_3, \ldots$, where $m_i \in M$ for every $i \in N$, is defined with $c_i = E_k(m_i)$ for every $i \in N$ a
- Decryption is defined with $m_i = D_k(c_i)$ for every $i \in N$.
- For block cipher is essential, that all blocks of PT are encrypted with the same transformation and all blocks of CT are decrypted also with the same transformation.
- In certain circumstances we can label substitution and transposition ciphers as block ciphers.

Block symmetric ciphers (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

Block symmetric ciphers (3)

A block cipher

- Converts n-bit PT on the n-bit CT.
- The block length n bits generates $2n$ different Block PT.
- Converts n-bit PT on the n-bit CT.
- The block length n bits generates $2n$ different block PT.
- Transforms 2^n blocks in PT to 2^n blocks CT.
- For $n = 4$ 16 b is transferred PT 16 b CT $\rightarrow$ easy breaking.
- For simple "substitution"can second column considered as 64 b key.
- For n-bit substitution cipher is key nx2^n b.
- For 64-bit encryption is key 64x2^{64} = 2^{70} bit.
- The problem of large keys solves $\rightarrow$ Feistel block cipher.

Block symmetric ciphers (4)

Feistel block cipher

- Solves the problem of large keys (its structure).
- Approximates the ideal block cipher for large n.
- It is composed cipher that uses a series two or more ciphers to achieve stronger cryptographic cipher.
- Reduce the key length of the ideal block cipher.
- Uses alternating substitution and transposition.
- Has parameters:
  - block size,
  - key length,
  - number of rounds,
  - algorithm round key generation,
  - complexity of operations in round.
- A larger block, a larger key length, a larger number of rounds, complicated algorithm and complex operations in round increases security and reduces the speed of encryption and decryption.

Block symmetric ciphers (5)

Feistel block cipher (1)

- encryption system LUCIFER (project H. Feistel) predecessor DES (Data Encryption Standard), has block 64 b and 128 b key
- currently being passes to block of 128 bits, which uses standard AES (not used there the principle Feistel).
- symmetric block cipher using the principles of algorithms Feistelova type allow for the gradual application of relatively simple transformation based on non-linear shift registers to create a complex cryptographic algorithm.
- This approach is also used in other areas of: security codes

Block symmetric ciphers (6)

Definition – Feistel cryptosystem

Let a set of messages M is composed of all possible $2n$-tic V_{2n} and space of keys is created by all possible h-tice of functions $k = f_1, f_2, \ldots, f_h$, $f_i : V_n \to V_n$ for every $i = 1, 2, \ldots, h$ and space of encrypted texts $C = V_{2n}$. Representation $T_k : K \times V_{2n} \to V_{2n}$, defined recurrently by equations

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

where $m = (m_0 \; m_1) \in M$, defines Feistel cryptosystem.

Example: For $m = \texttt{1001 1101}$ we get successively:

$c_1 = (\texttt{1101 1110})$, where $m_2 = \texttt{1001} \oplus f_1(\texttt{1101}) = \texttt{1001} \oplus \texttt{0111} = \texttt{1110}$

$c_2 = (\texttt{1110 0000})$, where $m_3 = \texttt{1101} \oplus f_2(\texttt{1110}) = \texttt{1101} \oplus \texttt{1101} = \texttt{0000}$

f_1 a f_2 – permutation function (confusion), $\oplus$ – function xor (diffusion)

DES (1)

DES algorithm

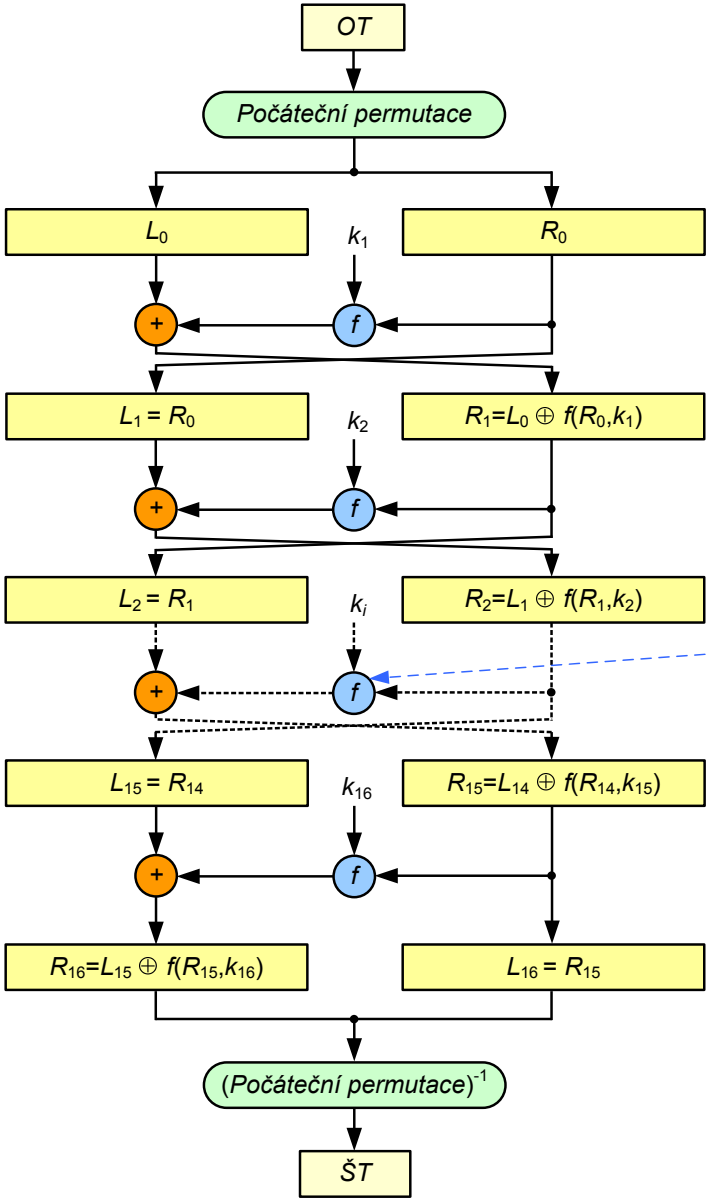
- Public tender (1977): encryption standard (FIPS 46-3) in USA for security of sensitive, but not secret data in government.
- Part of industry, internet and bank standards.
- 1977: warning – too short key length of 56b, which was injected into original design of IBM because of influence of American secret service NSA.
- DES – intensive research and attacks $\Rightarrow$ theoretical negative properties were discovered, like: so called weak and half-weak keys, complementarity and theoretically successful linear and differential cryptanalysis.
- In practice only serious disadvantage is short key.
- 1998: machine – DES-Cracker, decrypting DES with brute force.
- DES ended as American standard (only in "running out"systems because of compatibility) and replaced by: Triple-DES, (FIPS 46-3).
- From 26. 5. 2002 – encrypting standard of new generation AES.

DES (2)

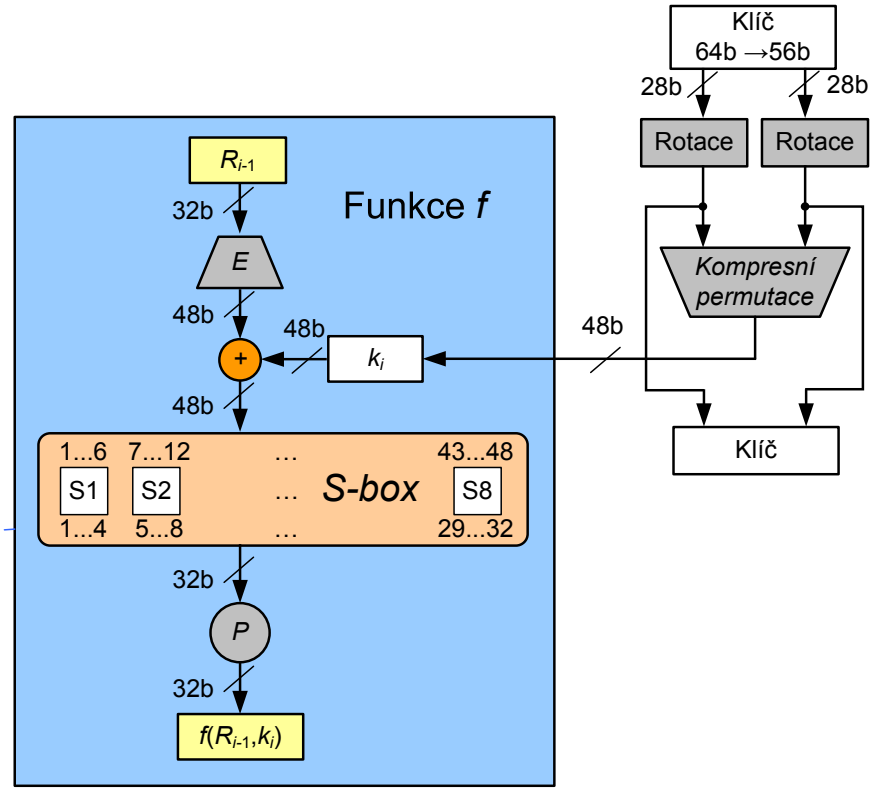
Construction blocks of DES (1)

- DES is iterated cipher of type $E_{k_{16}}(E_{k_{15}}(\ldots(E_{k_1}(m_i)\ldots)))$.
- Uses 16 rounds and 64b blocks of PT and CT. Encryption key k has a length 56 b (described but as 64b number, where every 8th bit is parity bit)
- 56b key k is in initialization phase, or during execution of an algorithm, expanded to 16 round keys from k_1 to k_{16}, which are strings of 48 bits, everyone of these bits is some bit of original key k.
- Instead of initial diffusion of PT is used keyless fixed permutation *Initial permutation* and instead of final diffusion, permutation inverse to it $(\textit{Initial permutation})^{-1}$.
- After *initial permutation* is block split into halves of 32b (L_0, R_0). Everyone of 16 rounds $i = 1, 2, \ldots, 16$ transform (L_i, R_i) to new value $(L_{i+1}, R_{i+1}) = (R_i, L_i \oplus f(R_i, k_{i+1}))$, differs only in usage of different round key k_i.
- In meaning of Feistel cryptosystem definition, is in this case $h = 16$ and $2n = 64$.

DES (3)



Algoritmus DES



DES (4)

Construction blocks of DES (2)

- After 16. round is executed the exchange between right a left side: $(L_{16}, R_{16}) = (R_{15}, L_{15} \oplus f(R_{15}, k_{16}))$ and final permutation ($Initial\ permutation$)$^{-1}$.
- Decryption is done in a same way as encryption, only the process of choosing round keys is reversed.

Rund function *f*

- Rund function is composed from binary loading of the key k_i on input. 48b key k_i is created after compression from 2 28b rotated parts of original key k, where the number of bites rotated is dependent on round number.
- This key k_i is further XORed with expanded 32b part R_{i-1}, which is expansively permuted in block E from 32b to 48b. This operation beside expansion of given 32b word also permutes bites of this word, such that avalanche effect is achieved.

DES (5)

Construction blocks of DES (3)

- Then, fixed nonlinear substitution is executed on level of 6b characters to 4b characters with following transposition on level of bits. These operation helps to achieve good diffusion and confusion.
- Used substitutions are called substitution boxes: S-boxes, they are the only nonlinear part of the schema. If we skipped substitutions, we can describe relations between CT, PT and key with operation of binary adding $\oplus$, so with linear equations.
- This nonlinearity is a obstacle in simple solving of equations, describing relations between CT, PT and key.
- After that, 32b outcoming word from S-box is permuted in block P. This permutation transform every incoming bit to output, where no incoming bit is used $2\times$.
- At the end, the result of permutation is added modulo 2 with left 32b half and the next round is started.

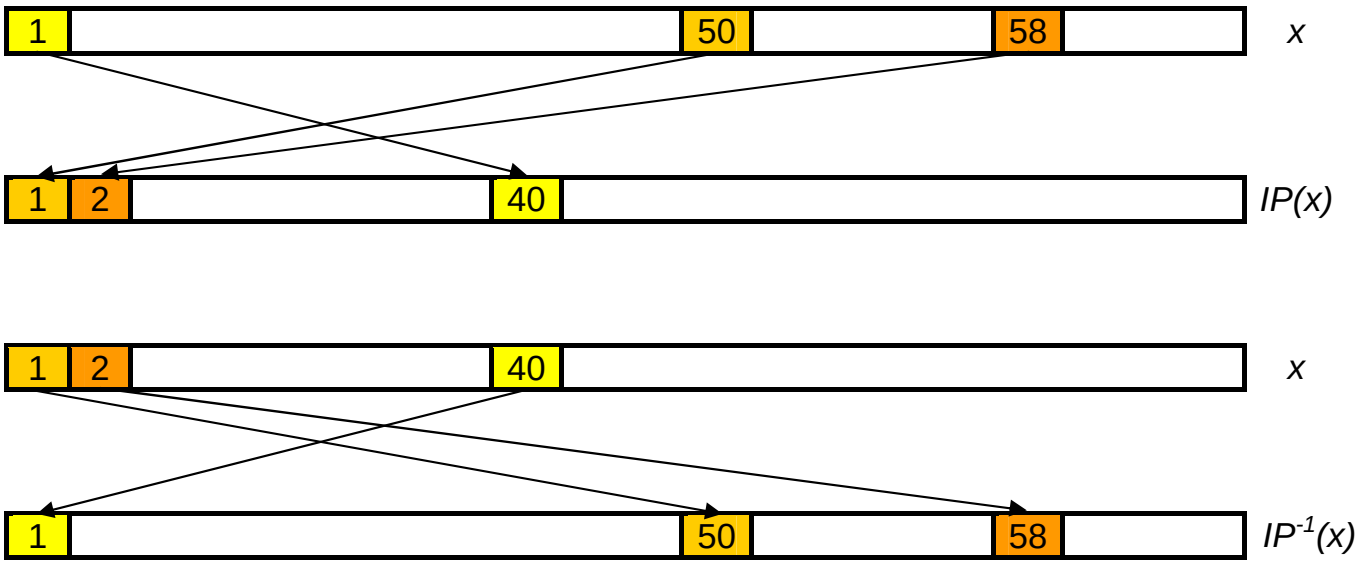
DES (6)

The internal structure of DES(1)

Initial permutation - IP and end permutation - IP^{-1}

- Do not increase the security of DES.
- Easy implemented in the hardware, but not in the software.
- The origin of permutations is probably in the an effort to rearrange PT into form, which is better further processable (related to the level of technology, today it is no longer valid).

Example:



DES (7)

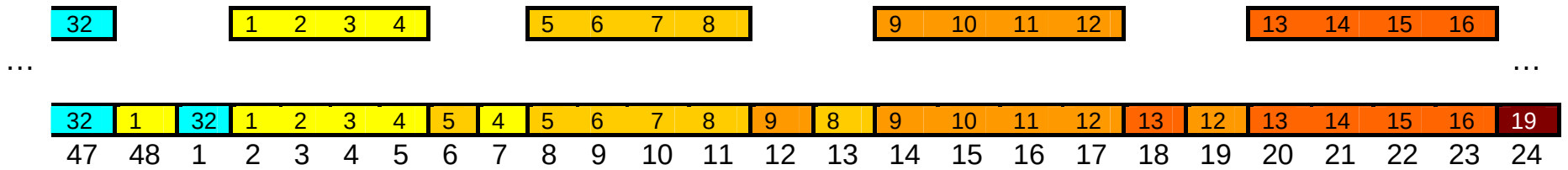
The internal structure of DES (2)

Initial permutation - IP and end permutation - 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

Implementation of the expansion function E



DES (8)

The internal structure of DES (3)
Substitution using S-boxes

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

- Only one nonlinear element DES which performs confusion
- Selection of conversion tables has never been fully disclosed

DES (9)

The internal structure of DES (3)

S-boxes were designed according to the following criteria:

1. Each S-box has 6 input bits and 4 output bits.
2. None of output bits is a linear combination of input bits.
3. When the input MSB = LSB and 4 middle bits are changing, then each possible 4-bit output value occurs only once.
4. If two inputs to an S-box are different only just one bite, then its output must be different in at least 2 bits.
5. If two inputs to an S-box are different in the two middle bits, then its output must be different in at least 2 bits.
6. If two inputs to an S-box are different in the the first two bits and are identical in the last 2 bits, then the output must be different.
7. For any nonzero 6-bit differential inputs is not more than 8 out of 32 pairs of input
8. A collision (no difference in output) 32 outputs 8-bit S-boxes is possible only for three neighboring S-boxes.

The internal structure of 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-bit input and 32-bit output $\rightarrow$
- directly permutations - no bit is omitted
- introduces diffusion

DES (10)

Key transformation (1) key transformation includes:

- Modifies the 64-bit key to 56-bit omitting every 8th bit of parity.
- Performs the permutations of key.
- 56-bit key is divided into 28-bit halves, which are rotated 1 (1st, 2nd, 9th 16th round) or 2 bits to the left (the other rounds).
- The rotation ensures the creation of different keys for each round.
- Compression permutation selects from 56 bits 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

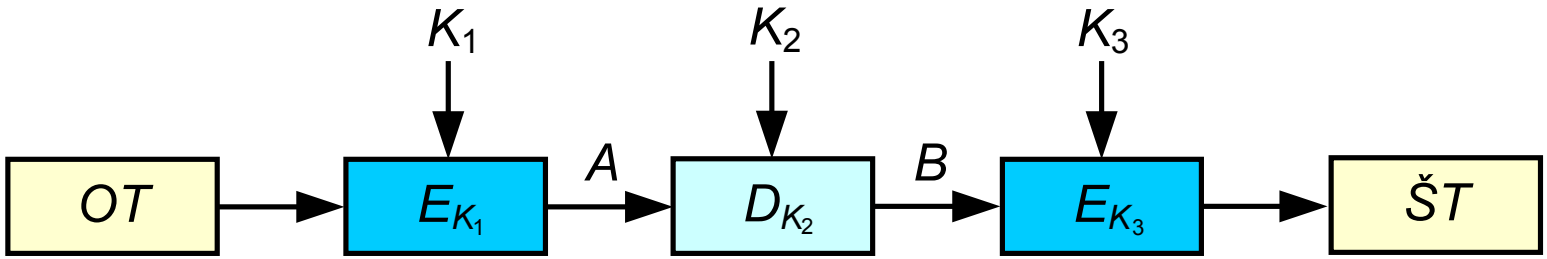
- TripleDES (3DES) is prolonging the key of original DES by using DES as construction block for 3$\times$ with 2 or 3 different keys.
- Most commonly is used the EDE version of this cipher, which is defined in standard FIPS PUB 46-3 (in bank norm X9.52).
- Incoming data PT are encrypted with $CT = E_{K_3}(D_{K_2}(E_{K_1}(PT)))$, where K_1, K_2 a K_3 are even 3 different keys, or $K3 = K1$. Version EDE is being used from compatibility reasons $\rightarrow$ in case of equality of all keys 3DES = DES.
- Key 3DES is either 112 bits (2 keys) or 168 bits (3 keys). 3DES is reliable $\rightarrow$ key is sufficiently long and theoretical weaknesses (complementarity, weak keys) can be prevented $\Rightarrow$
- 3DES and AES $\rightarrow$ is official standard replacing DES.
- 3DES can be used, as any other block cipher, in different operational modes (CBC mod $\Rightarrow$ 3DES-EDE-CBC).

TripleDES (2)

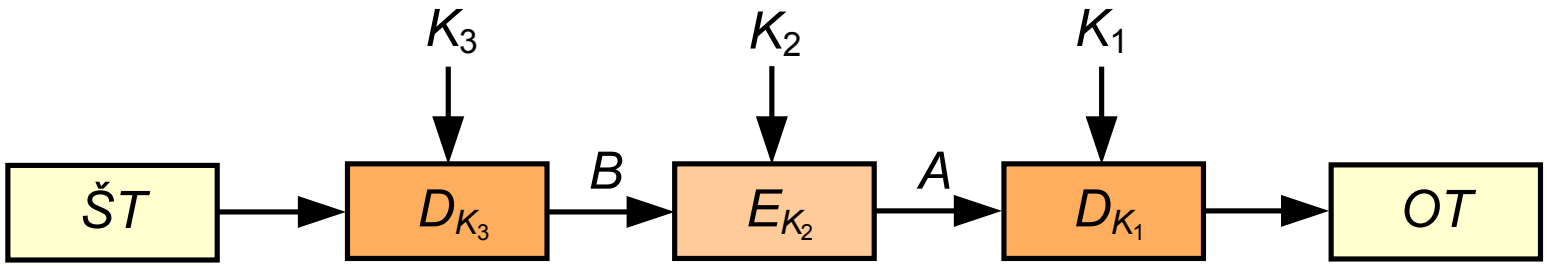
Algorithm 3DES

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)

- After brute-force attacks on DES, American standardization office prepared substitution - Advanced Encryption Standard (AES).
- 2.1. 1997 tender on AES – 15 candidates.
- From 5 finalists the Rijndael algorithm was chosen (authors J. Daemen a V. Rijmen).
- As AES way accepted with effect from 26th May 2002 and published as standard in official publication FIPS PUB 197.
- AES is block cipher with block length 128 bits, by which it differs from current block ciphers, which had block length of 64 bits.
- AES supports 3 key lengths: 128, 192 a 256 bits $\Rightarrow$ partly changes the algorithm (round count is as follows 10, 12 and 14).
- Bigger length of block and key prevent the attacks, which were applied on DES. AES doesn't have weak keys, is resistant against known attacks and methods of linear and differential cryptanalysis.

AES (2)

- Algorithm on encryption and decryption can be efficiently programmed on different types of processors, focused on small memory or code-size and is also suitable for parallel execution.
- AES will be probably valid encryption standard for several decades and will have a huge impact on computer safety.
- If we let N_k be a key length of 32b words, thus we have $N_k = 4, 6$ and 8 for key length 128, 192 a 256 bits.
- AES is an iterative cipher, number of rounds N_r is changing based on key-length: $N_r = N_k + 6$, i.e. it is 10, 12 or 14 round.
- This fact reflects the necessity of confusion due to key. Algorithm works with elements of Galois field $GF(2^8)$ and with polynomials, whose coefficients are elements of $GF(2^8)$. Byte with bits $(b_7, ..., b_0)$ is thus understood as polynomial $b_7 x^7 + \cdots + b_1 x^1 + b_0$ a operation "byte multiplication"is multiplication of there polynomials modulo $m(x) = x^8 + x^4 + x^3 + x^1 + 1$.

AES (3)

Rund keys

- AES uses $4 + N_r \times 4$ round keys with length of 32b, which are be defined way derived from encryption key.
- Before execution of 1st round of encryption is executed the „initial noise“, where the PT is xored with first 4 round keys (128b on 128b)$\Rightarrow$
- N_r similar rounds (with exception for the last, where the operation MixColumns isn't executed), in which output from every previous round is used as an input for the following round. By doing this the gradual adding of complexity is done.

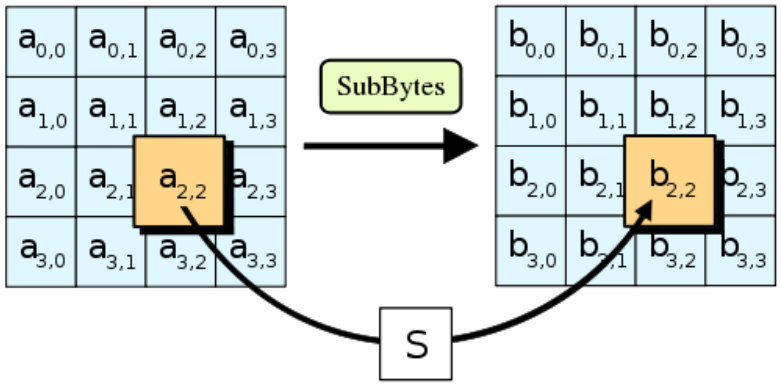
Rund (1)

- At beginning of every round the input (16 B) is always filled from left to right and from up to down by columns into the matrix 4x4 B $\mathbf{A} = (a_{ij})\ i, j = 0, 1, 2, 3$.
- On every byte of matrix **A** is separately applied substitution, given with fixed substitution table SubBytes.

AES (4)

Substitute bytes

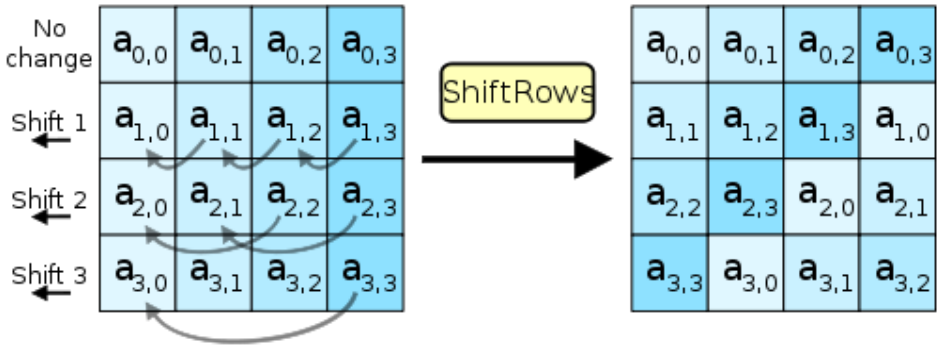
- Transformation in the direct and inverse form for direct and inverse substitution.
- S-box is organized in the form of a matrix of 16x16 (4b x 4b) for transforming all 8-bit values
- in the matrix is line number determined by 4 higher bits of the current byte and column number is determined by the lower 4 bits of the current byte
- S-box is filled a such way that every byte can be calculated as the multiplicative inverse in $GF(2^8)$ modulo $m(x) = x^8 + x^4 + x^3 + x + 1$ and zero is transformed on itself



AES (5)

Shift Rows

- The transformation is applied to the rows with individual byte.
- 1st row remains unchanged, the second row is shifted by one place to the left, 3rd row is moved 2 places to the left and the third row is moved 3 places to the left.
- inverse transformation performs the same shift lines, but to the right.
- Since the first 4 bytes of PT are written in a first column, and so on. Bytes of each column are distributed to 4 bytes of different columns, then there is a complete mixing of bytes PT - transposition occurs on byte level.



AES (6)

Mix Columns

- Father, at each column of the matrix operations applied MixColumns, that the substitution of 32 bits to 32 bits. This substitution can however described by linear relationships - all output bits are a linear combination of input bits. If we denote individual bytes in the within the columns of the matrix **A** (top to bottom) as a_0 to a_3, then the output will be the new value b_0 to b_3, according to relations

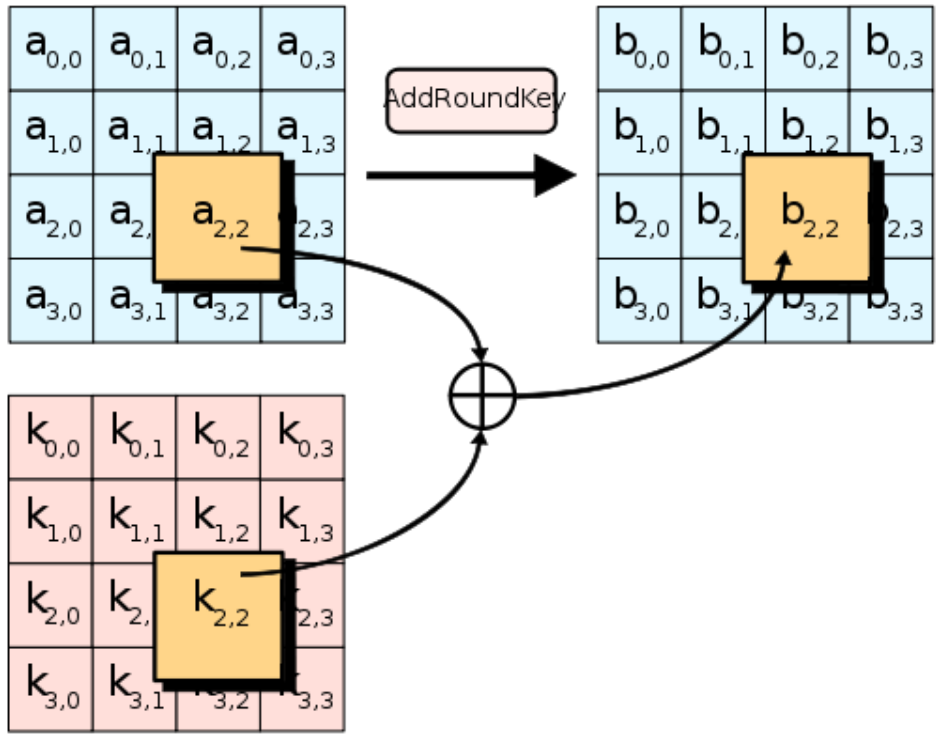
$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} '02' & '03' & '01' & '01' \\ '01' & '02' & '03' & '01' \\ '01' & '01' & '02' & '03' \\ '03' & '01' & '01' & '02' \end{pmatrix} \times \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix}$$

- Multiplication is multiplication by elements of $GF(2^8)$. constant elements this field are expressed in hexadecimal.

AES (7)

Add Round Key

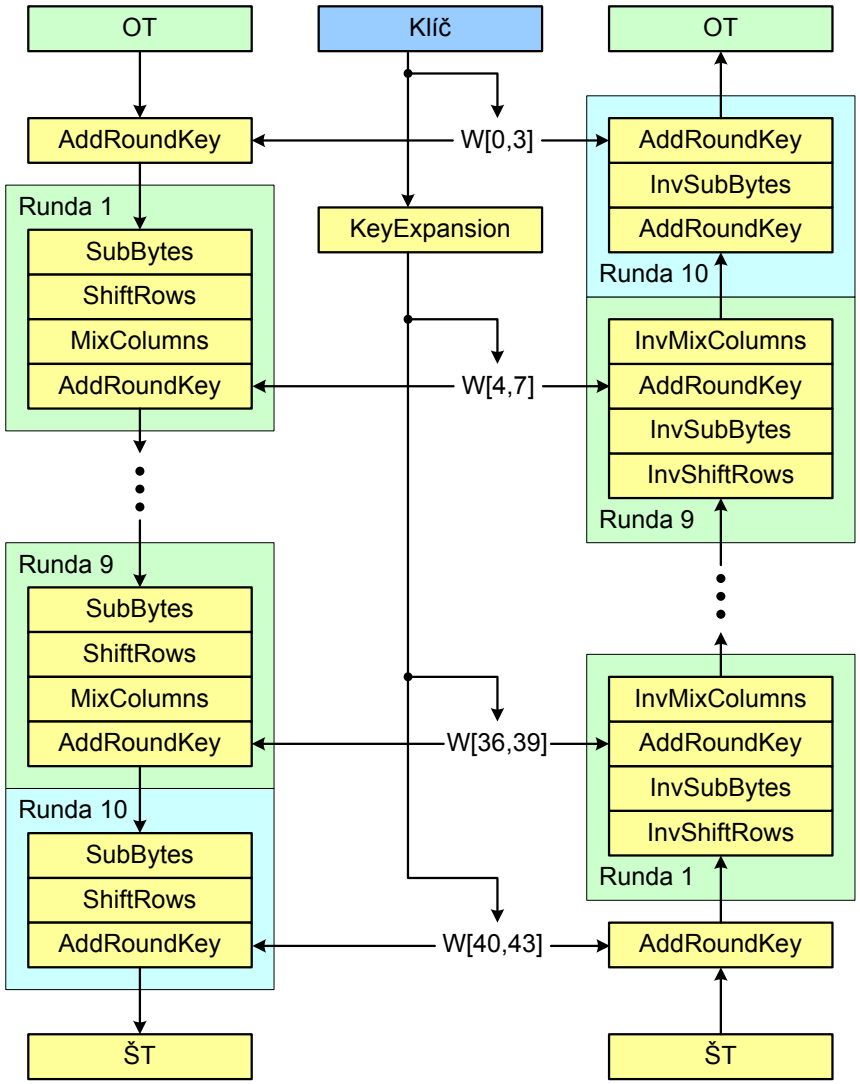
- The last round operation is performed transformation AddRoundKey, within which the individual columns of the matrix **A** from left to right xor 4 corresponding round keys. This is one round described and another begins. After the last round is CT only



reads from the matrix **B**.

Algoritmus AES

AES – Struktura šifrování a dešifrování



AES (8)

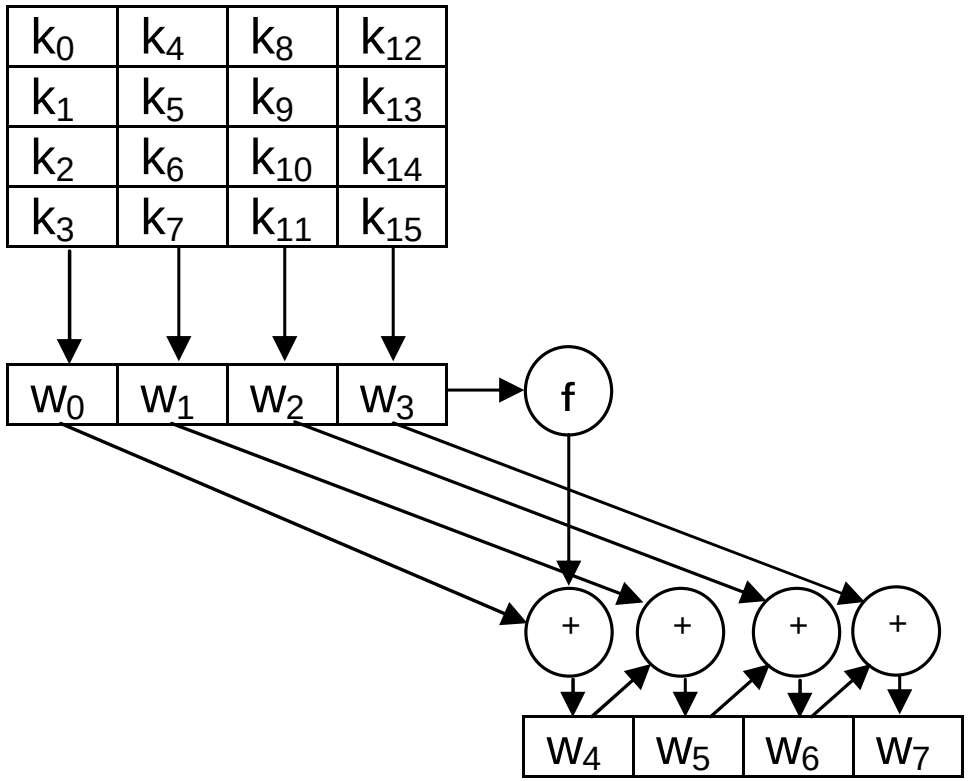
- During decryption are used operations to inverse operations used in the encryption, since all are reversible.
- Nonlinearities in the AES appear only in the the substitution of SubBytes. In 2002 it was found that interrelations output (y1, .., Y8) and input (x1, .., x8) bits can be described by implicit equations f(x1, .., x8, y1, .., y8) = 0 only second order.

AES (9)

Key Expansion (1)

- Operation implements expansion key with a length of 16 B - 128 b.
- Key has 4 words with a length of 4 bytes.
- If the number rund is 10 then is needed expansion to 44 words

Key expansion process



AES (10)

Key Expansion (2)
Operations creating of function f

- Operations Rot Word realizes cyclic shift word bytes w_3 about 1 position in the left.
- Operation Sub Word realizes substitution of shifted bytes according rules in S-boxe.
- The result of the previous operation is XORed with constant $RC(j)$, which values are defined for each round.
- $RC(j) = [B_j, 0, 0, 0]$

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