Block Cipher

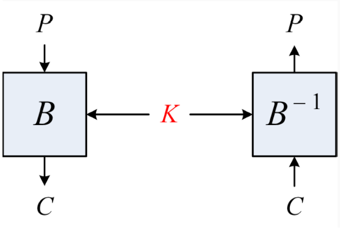
CUI TINGTING

School of Cypberspace, Hangzhou Dianzi University

January 9, 2024

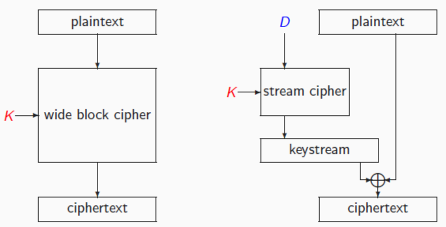
Contents

Block Cipher



- Permutation B_K operating on $\{0,1\}^b$ with the block length b
- Parameterized by a secret key: K
- Computing $C = B_K(P)$ or $P = B_K^{-1}(C)$ should be:
  - Effcient knowing the secret key K
  - Infeasible otherwise

Block vs. stream encryption

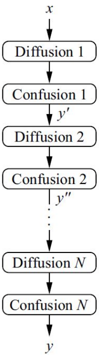


- NOT use D vs. use D (initial value).
- Block-wise vs. bit-wise.
- Different encryption algorithms vs. same encryption algorithm.
- Simple keyschedule vs. complex keystream process.

Block Cipher Primitives: Confusion and Diffusion

Claude Shannon: There are two primitive operations with which strong encryption algorithms can be built.

- Confusion: An encryption operation where the relationship between key and ciphertext is obscured.
- Diffusion: An encryption operation where the influence of one plaintext symbol is spread over many ciphertext symbols.
- Both operations by themselves CANNOT provide security.
- Product ciphers: concatenate confusion and diffusion elements.

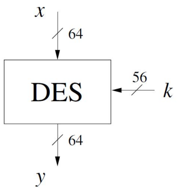


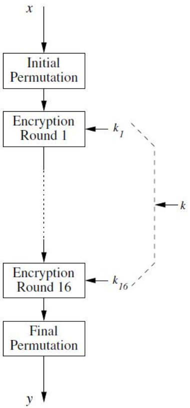
History of DES

- Data Encryption Standard (DES) encrypts blocks of size 64 bits.
- Developed by IBM based on the cipher Lucifer under influence of NSA, the design criteria have not been published.
- Standardized 1977 by the National Bureau of Standards (NBS, today called NIST).
- Most popular block cipher for most of the last 30 years
- By far best studied symmetric algorithm.
- Nowadays considered insecure due to the small key length of 56 bit.
- But: 3DES yields very secure cipher, still widely used today.
- Replaced by the Advanced Encryption Standard (AES) in 2000.

Overview of the DES Algorithm

- Block size: 64 bits, key size: 56 bits
- Totally 16 rounds
- Construction: Feistel network
- Different subkeys in each round derived from master key



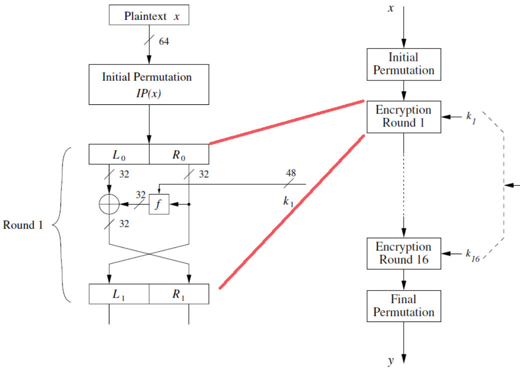


Feistel Network

- Expression:

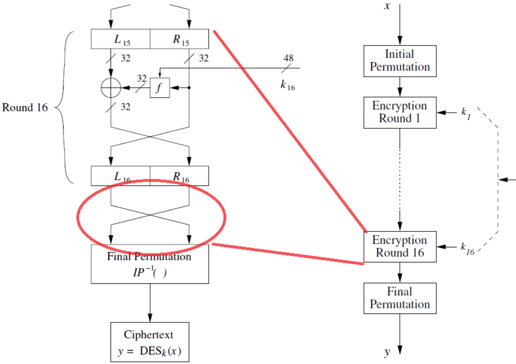
$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, k_i)$$

- Advantage: encryption and decryption differ only in key schedule



Feistel Network

- Note: Omit the last swap in round 16.



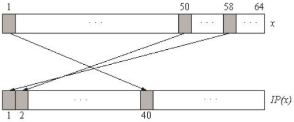
Initial and Final Permutation

- Bitwise permutations
- Finial permutation IP^{-1} is the inverse of IP.

Initial Permutation

IP

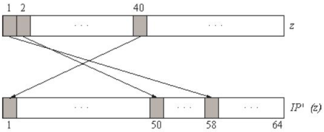
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7



Final Permutation

IP^{-1}

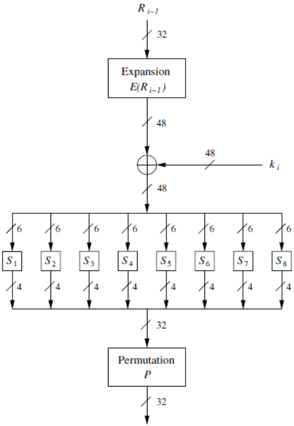
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25



f-Function in Feistel Network

Four steps:

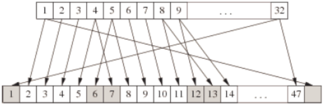
1. Expansion E
2. XOR with round key
3. Substitution (8 S-boxes)
4. Permutation P

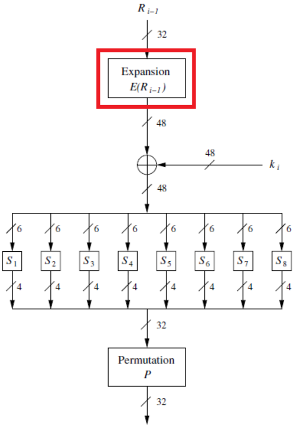


f-Function: Expansion Function E

Main purpose: diffision.

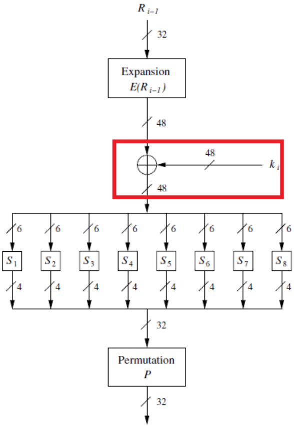
E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1





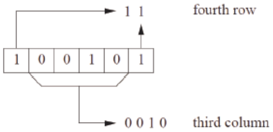
f-Function: Add Round Key

- Bitwise XOR of the round key and the output of the expansion function E.
- Round keys are derived from the main key in the DES key schedule.

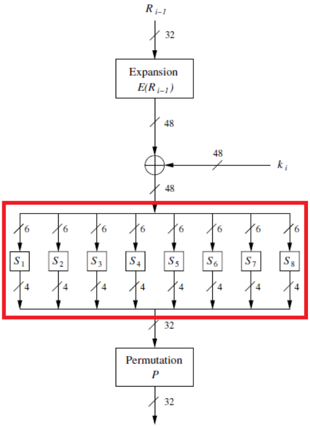


f-Function: S-boxes Layer

- Eight different S-box.
- Nonlinear, non-bijective.
- Main purpose: confusion.



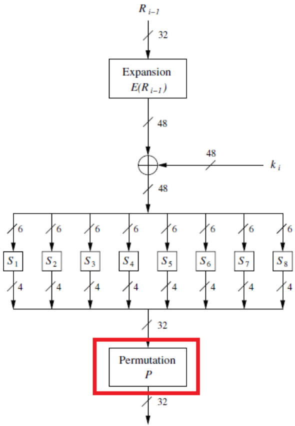
S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



f-Function: Permutation P

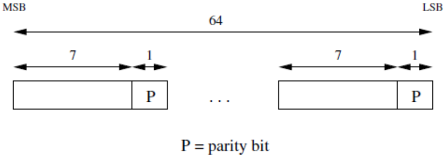
- Bitwise permutation.
- Output bits of one S-Box effect several S-Boxes in next round.
- Main purpose: diffusion.

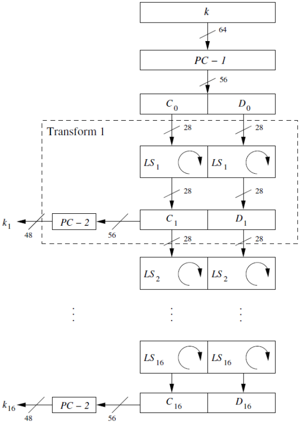
P							
16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25



Key Schedule

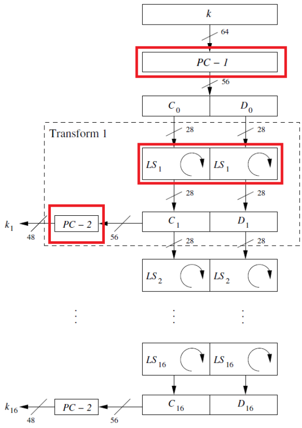
- Derives 16 round keys (or subkeys) k_i of 48 bits each from the original 56-bit key.
- The input key size of the DES is 64 bit: 56-bit key and 8-bit parity.





Key Schedule

- Split key into 28-bit halves C_0, D_0.
- Two rotations LS_i per round:
  - Rounds 1,2,9,16: $\lll 1$.
  - In other rounds: $\lll 2$.
- Two permutations $PC-1$ and $PC-2$.



$PC-1$							
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

$PC-2$							
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

Decryption

1. Due to Feistel construction, decryption only differs with encryption in key schedule.
2. Generate the same 16 round keys in reverse order.

Security of DES

- After proposal of DES two major criticisms arose:
  1. Key space is too small (2^{56} keys);
  2. S-box design criteria have been kept secret: Are there any hidden analytical attacks (backdoors), only known to the NSA?
- Analytical Attacks:
  1. Differential attack (1990, chosen plaintext attack, 2^{47})
  2. Linear attack (1992, known plaintext attack, 2^{43})
- Exhaustive key search:
  1. Definition: for a given pair of plaintext-ciphertext (x, y) test all 2^{56} keys until the condition $DES_k^{-1}(y) = x$ is fulfilled.
  2. Relatively easy given today's computer technology!

History of Attacks on DES

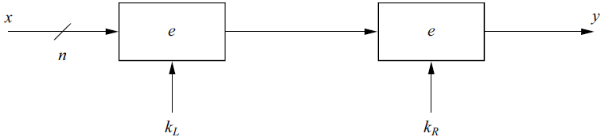
Year	Proposed/ implemented DES Attack
1977	Diffie & Hellman, (under-)estimate the costs of a key search machine
1990	Biham & Shamir propose differential cryptanalysis (2^{47} chosen ciphertexts)
1993	Mike Wiener proposes design of a very efficient key search machine: Average search requires 36h. Costs: $1.000.000
1993	Matsui proposes linear cryptanalysis (2^{43} chosen ciphertexts)
Jun. 1997	DES Challenge I broken, 4.5 months of distributed search
Feb. 1998	DES Challenge II--1 broken, 39 days (distributed search)
Jul. 1998	DES Challenge II--2 broken, key search machine *Deep Crack* built by the Electronic Frontier Foundation (EFF): 1800 ASICs with 24 search engines each, Costs: $250 000, 15 days average search time (required 56h for the Challenge)
Jan. 1999	DES Challenge III broken in 22h 15min (distributed search assisted by *Deep Crack*)
2006-2008	Reconfigurable key search machine *COPACOBANA* developed at the Universities in Bochum and Kiel (Germany), uses 120 FPGAs to break DES in 6.4 days (avg.) at a cost of $10 000.

Double DES

How to enhence the security of DES?

Double DES

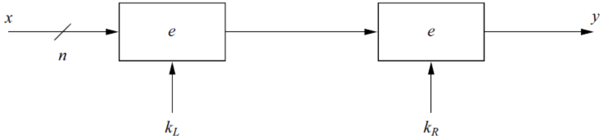
How to enhence the security of DES?



- Double DES: A plaintext x is first encrypted with a key k_L, then is encrypted again using a second key k_R to product the ciphertext y.

Double DES

How to enhence the security of DES?



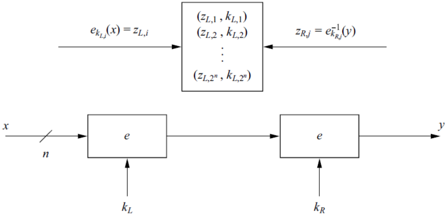
- Double DES: A plaintext x is first encrypted with a key k_L, then is encrypted again using a second key k_R to product the ciphertext y.
- Exhaustive key search would require $2^{56} \cdot 2^{56} = 2^{112}$ encryptions or decryptions.

Meet-in-the-Middle Attack

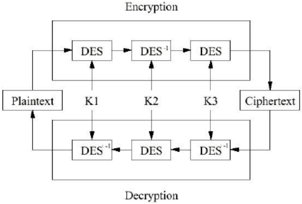
MITM Attack

- **Phase I: Table Computation.** Compute a lookup table for all pairs $(k_{L,i}, z_{L,i})$, where $e_{k_{L,i}}(x) = z_{L,i}$, $i = 1, 2, \ldots, 2^k$
- **Phase II: Key Matching.** Compute $e^{-1}_{k_{R,i}}(x) = z_{R,i}$ and check if $z_{R,i}$ matchs with any $z_{L,i}$ in the table.

- Data comp.: $3 \sim 4$ known plaintexts.
- Time comp.: 2^{57}.
- Memory comp.: 2^{56}.



Triple DES – 3DES



- Triple encryption using DES is often used in practice to extend the effective key length of DES to 112
- Advantage: choosing $k_1 = k_2 = k_3$ performs single DES encryption.
- No practical attack known today.
- Used in many legacy applications, i.e., in banking systems.

AES Selection Process

- NIST launches the AES open contest to replace DES in 1997
  - 128-bit block length, 128-, 192- and 256-bit keys
  - specs, code, design rationale and preliminary analysis
- First round: August 1998 to August 1999
  - 15 candidates at 1st AES conference in Ventura, California
  - analysis presented at 2nd AES conf. in Rome, March 1999
  - NIST narrowed down to 5 finalists using this analysis
- Second round: August 1999 to summer 2000
  - analysis presented at 3rd AES conf. in New York, April 2000
  - NIST selected winner using this analysis: Rijndael
- NIST motivated their choice in two reports

AES Winner — Rijndael Cipher

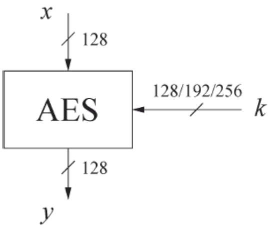
- Block and key lengths $\in \{128, 160, 192, 224, 256\}$
  - set of 25 block ciphers
  - AES limits block length to 128 and key length to multiples of 64


Vincent Rijmen


Joan Daemen

AES: Overview

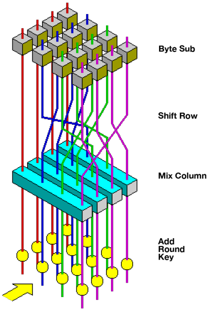
Three AES versions according to key size.

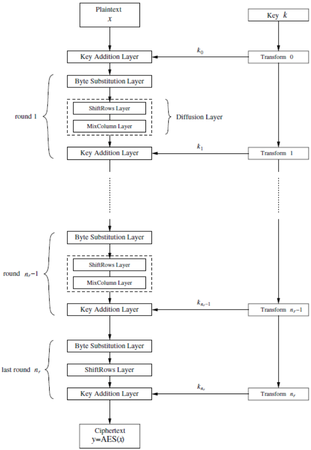


Block size (bit)	Key size (bit)	#Rounds
128	128	10
128	192	12
128	256	14

AES: Overview

- Iterated cipher with 10/12/14 rounds.
- Each round consists of "Layers".
- all operations on $GF(2^8)$ field.





Group

A group $< G, \circ >$ is a set of elements G with an operation $\circ$ which combines two elements of G. A group has the following properties:

1. The group operation $\circ$ is closed. That is, for all $a, b \in G$, it holds that $a \circ b = c \in G$.
2. The group operation is associative. That is, $a \circ (b \circ c) = (a \circ b) \circ c$.
3. There is an element $1 \in G$, called the neutral element (or identity element), such that $a \circ 1 = 1 \circ a = a$ for all $a \in G$.
4. For each $a \in G$ there exists an element $a^{-1} \in G$, called the inverse of a, such that $a \circ a^{-1} = a^{-1} \circ a = 1$.

A group G is abelian (or commutative) if, furthermore, $a \circ b = b \circ a$ for all $a, b \in G$.

Field

A field $< F, +, \times >$ is a set of elements with the following properties:

1. All elements of F form an additive group with the group operation $+$ and the neutral element 0.
2. All elements of F except 0 form a multiplicative group with the group operation $\times$ and the neutral element 1.
3. When the two group operations are mixed, the distributivity law holds, i.e., for all $a, b, c \in F : a(b + c) = (ab) + (ac)$.

Field

A field $< F, +, \times >$ is a set of elements with the following properties:

1. All elements of F form an additive group with the group operation $+$ and the neutral element 0.
2. All elements of F except 0 form a multiplicative group with the group operation $\times$ and the neutral element 1.
3. When the two group operations are mixed, the distributivity law holds, i.e., for all $a, b, c \in F : a(b + c) = (ab) + (ac)$.

Example 1

$< \mathbb{R}, +, \times >$

Galois Fields (Finite Fields)

Definition 2 (order/cardinality)

The number of elements in the field is called the order or cardinality of the field.

Galois Fields (Finite Fields)

Definition 2 (order/cardinality)

The number of elements in the field is called the order or cardinality of the field.

Theorem 3

A field with order m only exists if m is a prime power, i.e., $m = p^n$, for some positive integer n and prime integer p. p is called the characteristic of the finite field.

Example 4

Finite fields with 11, 81, 256, 12?

Prime Fields $GF(p)$

$GF(p)$

Let p be a prime, $GF(p)$ consists of

1. Elements: $0, 1, \ldots, p-1$.
2. Two operations: modular integer addition and integer multiplication modulo p.

Example 5

GF(5)={0,1,2,3,4}
GF(2)={0,1} (addition: XOR gate, multiplication: logical AND gate)

Extension Fields $GF(2^m)$

Extension fields

$GF(2^m)$ with $m > 1$:

- Elements of extension fields can be represented as polynomials.
- Computation in the extension field is achieved by performing a certain type of polynomial arithmetic.

For example: each element $A \in GF(2^8)$ is represented as:

$$A(x) = a_7x^7 + \ldots, +a_1x^1 + a_0, a_i \in GF(2)$$

Another form is an 8-bit vector:

$$A(x) = (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$$

Addition and Subtraction in $GF(2^m)$

Definition 6

Let $A(x), B(x) \in GF(2^m)$. The sum of the two elements is computed according to:

$$C(x) = A(x) + B(x) = \sum_{i=0}^{m-1} c_i x^i, c_i \equiv a_i + b_i \mod 2$$

and the difference is computed according to:

$$C(x) = A(x) - B(x) = \sum_{i=0}^{m-1} c_i x^i, c_i \equiv a_i - b_i \equiv a_i + b_i \mod 2$$

Addition and Subtraction in $GF(2^m)$

Example 7

Assume $A(x) = x^7 + x^6 + x^4 + 1$, $B(x) = x^4 + x^2 + 1$ in $GF(2^8)$,
How about $A(x) + B(x)$?

Addition and Subtraction in $GF(2^m)$

Example 7

Assume $A(x) = x^7 + x^6 + x^4 + 1$, $B(x) = x^4 + x^2 + 1$ in $GF(2^8)$, How about $A(x) + B(x)$?

$$
\begin{array}{rcccc}
A(x) = & x^7 + x^6 + & x^4 + & & 1 \\
B(x) = & & x^4 + & x^2 + & 1 \\
\hline
C(x) = & x^7 + x^6 + & & x^2 &
\end{array}
$$

Multiplication in $GF(2^m)$

Standard polynomial multiplication rule:

$$\begin{aligned} A(x) \cdot B(x) &= (a_{m-1}x^{m-1} + \ldots + a_0) \cdot (b_{m-1}x^{m-1} + \ldots + b_0) \\ &= c_{2m-2}x^{2m-2} + \ldots + c_0 \end{aligned}$$

where

$$\begin{aligned} c_0 &= a_0 b_0 \mod 2, \\ c_1 &= a_0 b_1 + a_1 b_0 \mod 2, \\ &\ldots\ldots \\ c_{2m-2} &= a_{m-1} b_{m-1} \mod 2 \end{aligned}$$

Multiplication in $GF(2^m)$

Standard polynomial multiplication rule:

$$
\begin{aligned}
A(x) \cdot B(x) =& (a_{m-1}x^{m-1} + \ldots + a_0) \cdot (b_{m-1}x^{m-1} + \ldots + b_0) \\
=& c_{2m-2}x^{2m-2} + \ldots + c_0
\end{aligned}
$$

where

$$
\begin{aligned}
c_0 &= a_0 b_0 \mod 2, \\
c_1 &= a_0 b_1 + a_1 b_0 \mod 2, \\
&\ldots\ldots \\
c_{2m-2} &= a_{m-1} b_{m-1} \mod 2
\end{aligned}
$$

Problem here?

Multiplication in $GF(2^m)$

Standard polynomial multiplication rule:

$$\begin{aligned} A(x) \cdot B(x) =& (a_{m-1}x^{m-1} + \ldots + a_0) \cdot (b_{m-1}x^{m-1} + \ldots + b_0) \\ =& c_{2m-2}x^{2m-2} + \ldots + c_0 \end{aligned}$$

where

$$\begin{aligned} c_0 &= a_0 b_0 \mod 2, \\ c_1 &= a_0 b_1 + a_1 b_0 \mod 2, \\ &\ldots\ldots \\ c_{2m-2} &= a_{m-1} b_{m-1} \mod 2 \end{aligned}$$

Problem here?
NOT closed.

Multiplication in $GF(2^m)$

Multiplication

Let $A(x), B(x) \in GF(2^m)$ and let

$$P(x) \equiv \sum_{i=0}^{m} p_i x^i, p_i \in GF(2)$$

be an irreducible polynomial. Multiplication of the two elements $A(x), B(x)$ is performed as

$$C(x) \equiv A(x) \cdot B(x) \mod P(x).$$

Multiplication in $GF(2^m)$

Example 8

Multiply the two polynomials $A(x) = x^3 + x^2 + 1$ and $B(x) = x^2 + x$ in the field $GF(2^4)$, where the irreducible polynomial is $P(x) = x^4 + x + 1$.

Multiplication in $GF(2^m)$

Example 8

Multiply the two polynomials $A(x) = x^3 + x^2 + 1$ and $B(x) = x^2 + x$ in the field $GF(2^4)$, where the irreducible polynomial is $P(x) = x^4 + x + 1$.

$$\begin{aligned} A(x) \cdot B(x) &= x^5 + x^3 + x^2 + x \\ &= x \cdot (x^4 + x + 1) + (x^2 + x) + x^3 + x^2 + x \\ &= x^3 \mod p(x) \end{aligned}$$

Multiplication in $GF(2^m)$

Example 8

Multiply the two polynomials $A(x) = x^3 + x^2 + 1$ and $B(x) = x^2 + x$ in the field $GF(2^4)$, where the irreducible polynomial is $P(x) = x^4 + x + 1$.

$$\begin{aligned} A(x) \cdot B(x) &= x^5 + x^3 + x^2 + x \\ &= x \cdot (x^4 + x + 1) + (x^2 + x) + x^3 + x^2 + x \\ &= x^3 \mod p(x) \end{aligned}$$

Vector form: $(1101) \cdot (0110) = (1000)$ or $0xd \cdot 0x6 = 0x8$

Inversion in $GF(2^m)$

Inversion

For a given finite field $GF(2^m)$ and the corresponding irreducible reduction polynomial $P(x)$, the inverse A^{-1} of a nonzero element $A \in GF(2^m)$ is defined as:

$$A^{-1}(x) \cdot A(x) = 1 \mod P(x).$$

Inversion in $GF(2^m)$

Example 9

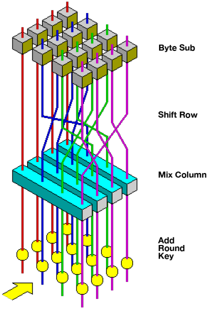
The table contains all inverses in $GF(2^8)$ modulo $P(x) = x^8 + x^4 + x^3 + x + 1$ in hexadecimal notation.

X \ Y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	00	01	8D	F6	CB	52	7B	D1	E8	4F	29	C0	B0	E1	E5	C7
1	74	B4	AA	4B	99	2B	60	5F	58	3F	FD	CC	FF	40	EE	B2
2	3A	6E	5A	F1	55	4D	A8	C9	C1	0A	98	15	30	44	A2	C2
3	2C	45	92	6C	F3	39	66	42	F2	35	20	6F	77	BB	59	19
4	1D	FE	37	67	2D	31	F5	69	A7	64	AB	13	54	25	E9	09
5	ED	5C	05	CA	4C	24	87	BF	18	3E	22	F0	51	EC	61	17
6	16	5E	AF	D3	49	A6	36	43	F4	47	91	DF	33	93	21	3B
7	79	B7	97	85	10	B5	BA	3C	B6	70	D0	06	A1	FA	81	82
8	83	7E	7F	80	96	73	BE	56	9B	9E	95	D9	F7	02	B9	A4
9	DE	6A	32	6D	D8	8A	84	72	2A	14	9F	88	F9	DC	89	9A
A	FB	7C	2E	C3	8F	B8	65	48	26	C8	12	4A	CE	E7	D2	62
B	0C	E0	1F	EF	11	75	78	71	A5	8E	76	3D	BD	BC	86	57
C	0B	28	2F	A3	DA	D4	E4	0F	A9	27	53	04	1B	FC	AC	E6
D	7A	07	AE	63	C5	DB	E2	EA	94	8B	C4	D5	9D	F8	90	6B
E	B1	0D	D6	EB	C6	0E	CF	AD	08	4E	D7	E3	5D	50	1E	B3
F	5B	23	38	34	68	46	03	8C	DD	9C	7D	A0	CD	1A	41	1C

Internal Structure of AES

- AES is a byte-oriented cipher
- The state A (i.e., the 128-bit data path) can be arranged in a 4×4 matrix:

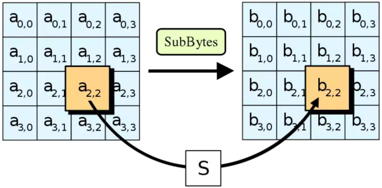
$$\begin{pmatrix} a_{00} & a_{01} & a_{42} & a_{43} \\ a_{11} & a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{32} & a_{33} \end{pmatrix}$$

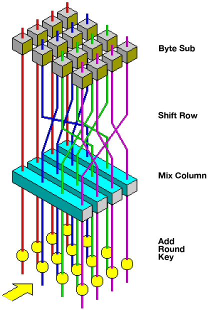


Non-linear layer: SubBytes

SubBytes consists of 16 S-boxes:

- identical
- nonlinear
- bijective

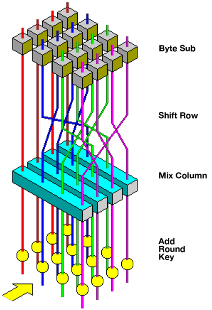




Non-linear layer: SubBytes

S-box: $y = A(x^{-1}) + b$ in $GF(2^8)$ with
$p(x) = x^8 + x^4 + x^3 + x + 1$.

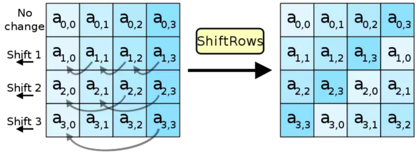
x \ y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

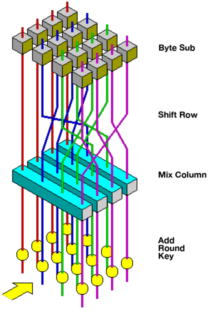


Example: $S(C2) = 25$

Shuffing layer: ShiftRows

- Each row is shifted by a different amount
- Moves bytes in a given column to 4 different columns
- Contribute to fast diffusion

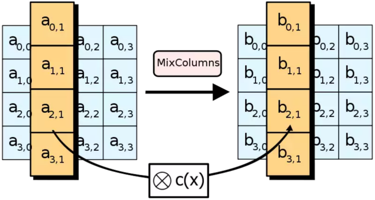


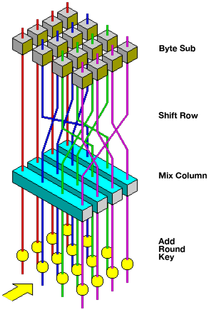


Mixing layer: MixColumns

Same invertible mapping applied to all 4 columns with $p(x) = x^8 + x^4 + x^3 + x + 1$

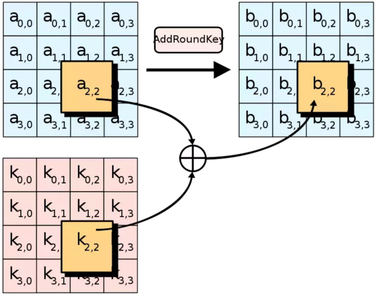
$$c(x) = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix}$$

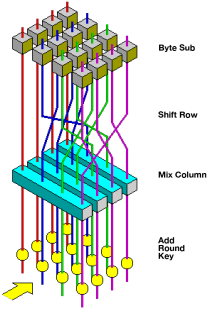




Round key addition: AddRoundKey

The subkeys are generated in the key schedule.



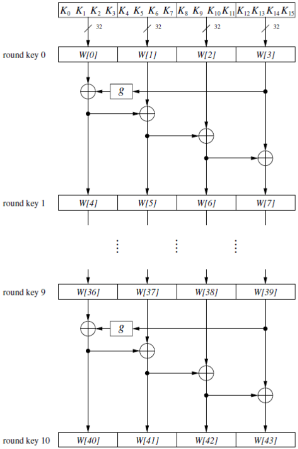


Key Schedule

- Subkeys are derived recursively from the original master key.
- Each round has 1 subkey, plus 1 whitening subkey at the beginning of AES.
- There are different key schedules for the different key sizes.

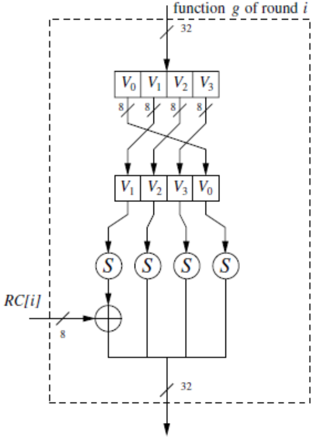
block size (bit)	key size (bit)	#rounds	#subkeys
128	128	10	11
128	192	12	13
128	256	14	15

Key Schedule for 128-Bit Key AES



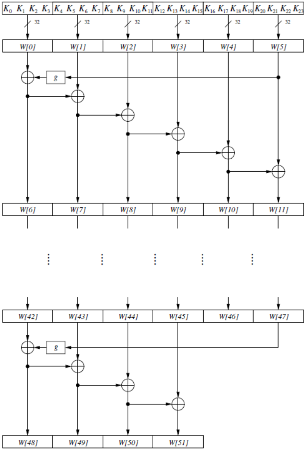
- Word-oriented: 1 word = 32 bits
- 11 subkeys are stored in $W[0], \ldots, W[43]$
- First subkey $W[0], \ldots, W[3]$ is the original AES key
- $W[4i] = W[4(i-1)] + g(W[4i-1])$.
- $W[4i+j] = W[4i+j-1] + W[4(i-1)+j]$

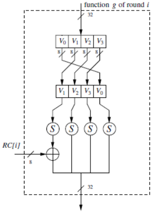
Key Schedule for 128-Bit Key AES



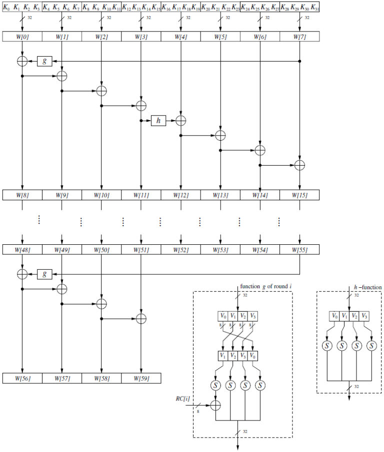
- The round coefficient is an element of the Galois field $GF(2^8)$ with $p(x) = x^8 + x^4 + x^3 + x + 1$.
- $RC[i] = x^{i-1}, i = 1, 2, \ldots$.
- $g()$ has two purposes:
  - add nonlinearity to the key schedule
  - removes symmetry in AES

Key Schedule for 192-Bit Key AES

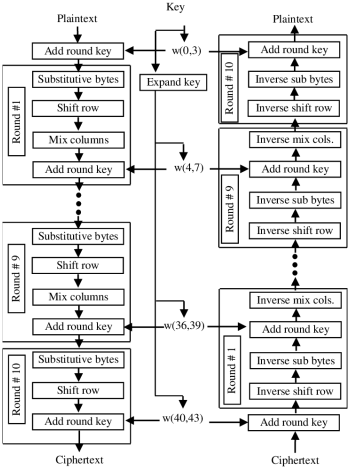




Key Schedule for 256-Bit Key AES



Decryption



- Inverse MixColumn Sublayer

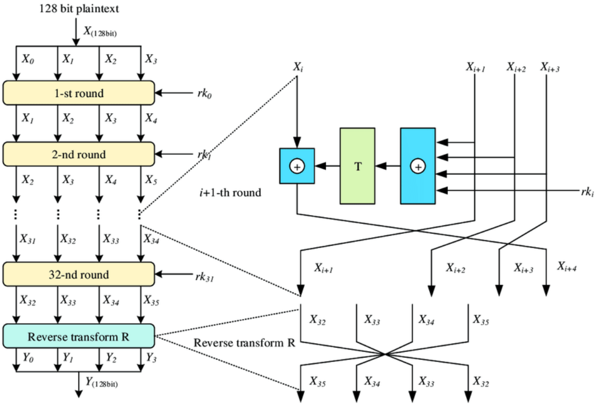
$$c(x) = \begin{pmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{pmatrix}$$

- Inverse ShiftRows Sublayer
- Inverse Byte Substitution Layer

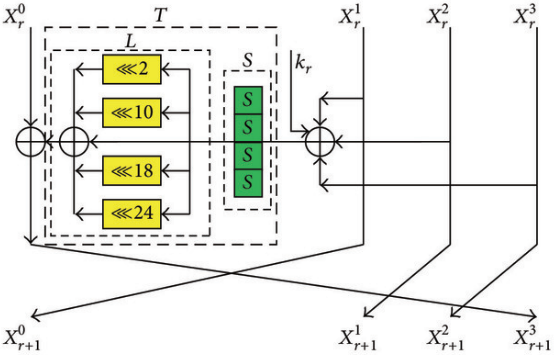
Security

- Brute-force attack:
  - Due to the key length of 128, 192 or 256 bits, a brute-force attack is not possible
- Analytical attacks:
  - There is no analytical attack known that is better than brute-force in single-key setting
  - There are related-key boomerang attack on AES-192 and AES-256.

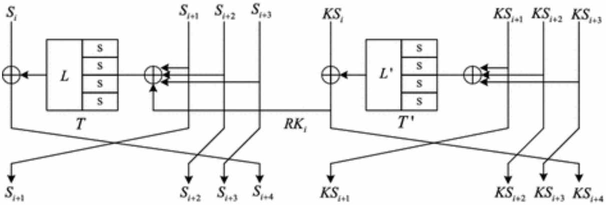
Overview of SM4 Cipher



Internal Structure of SM4



Key Schedule



Applications of Block Ciphers

A block cipher is much more than just an encryption algorithm, it can be used

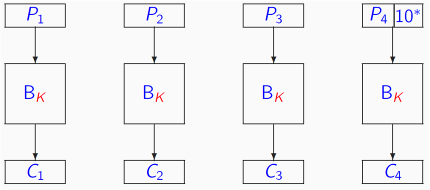
1. to build different types of block-based encryption schemes
2. to realize stream ciphers
3. to construct hash functions
4. to make message authentication codes (MAC)
5. to build key establishment protocols
6. to make a pseudo-random number generator
7. ...

Encryption with Block Ciphers: Modes of Operation

There are several ways of encrypting long plaintexts, e.g., an e-mail or a computer file, with a block cipher ("modes of operation")

1. Electronic Code Book mode (ECB)
2. Cipher Block Chaining mode (CBC)
3. Output Feedback mode (OFB)
4. Cipher Feedback mode (CFB)
5. Counter mode (CTR)
6. Galois Counter Mode (GCM)

Electronic Code Book mode (ECB)



Advantages:

- simple: not require block synchronization between Alice and Bob.
- parallelizable: for high-speed implementations.

Disadvantages:

- equal plaintext blocks $\rightarrow$ equal ciphertext blocks: low-entropy
- problem in padded last block

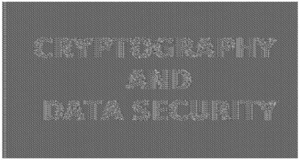
Attacks on ECB

Block #	1	2	3	4	5
	Sending Bank A	Sending Account #	Receiving Bank B	Receiving Account #	Amount $

Attack process:

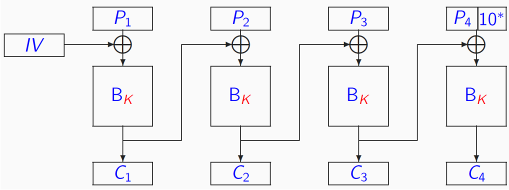
1. Oscar opens one account at bank A and one at bank B.
2. He sends $1.00 transfers from his account at bank A to his account at bank B
3. all transforms replaces block 4

CRYPTOGRAPHY AND DATA SECURITY



Statistical properties in the plaintext are preserved in the ciphertext.

Cipher Block Chaining mode (CBC)



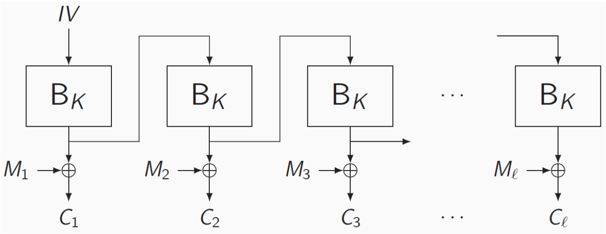
- ECB with plaintext block randomized by previous ciphertext block
- First plaintext block randomized with Initial Value (IV)
- Advantages: solve some leakage in ECB, paralleled decryption
- Disadvantages: encryption strictly serial, IV re-used attack

IV Re-used Attack on CBC

Block #	1	2	3	4	5
	Sending Bank A	Sending Account #	Receiving Bank B	Receiving Account #	Amount $

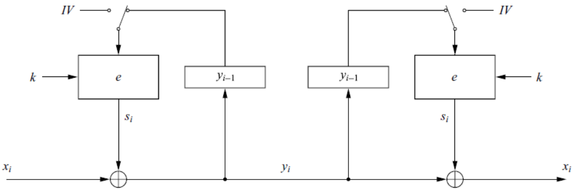
1. Oscar opens one account at bank A and one at bank B.
2. He sends $1.00 transfers from his account at bank A to his account at bank B
3. all transforms replaces block 5

Output Feedback mode (OFB)



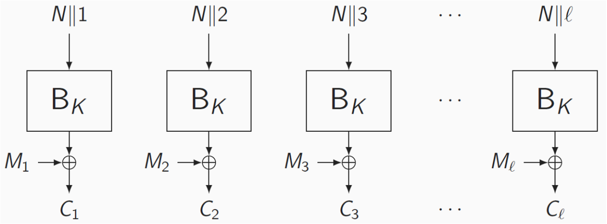
- to build a synchronous stream cipher from a block cipher
- The key stream is generated blockwise, not bitwise
- strictly serial for encryption and decrytion
- no need for B_K^{-1}, no padding

Cipher Feedback mode (CFB)



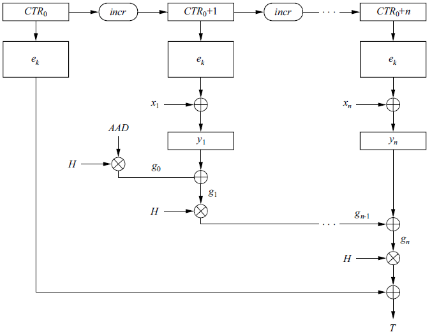
- to build a synchronous stream cipher from a block cipher
- The key stream is generated blockwise, not bitwise
- strictly serial for encryption and decrytion
- use in situations where short plaintext blocks are to be encrypted

Counter mode (CTR)



- use a block cipher as a stream cipher (like the OFB and CFB modes)
- fully parallelizable

Galois Counter Mode (GCM)



- Message Authentication: the receiver can make sure that the message was really created by the original sender
- Message Integrity: the receiver can make sure that nobody tampered with the ciphertext during transmission

Overview on Encryption Modes

Feature	ECB	CBC	OFB	CFB	CTR
parallel encryption	y	n	n	n	y
parallel decryption	y	y	n	n	y
padding-free	n	n	y	y	y
IV-violation tolerant	n.a.	y	n	n	n

Legend:

- random access: fast decryption of bits anywhere in the message
- bit errors limited: bitflips in C do not spread out in P

Thanks & Questions