第三章　分组密码与高级加密标准

张磊

华东师范大学 • 软件学院

§3.1　引言

现代分组密码的基本构成

现代分组密码大多数都是乘积密码, 其通常伴随一系列的置换和代换操作, 常见的乘积密码是迭代密码: 这种密码明确定义了一个轮函数和一个密钥编排方案, 一个明文的加密将经过 N_r 轮类似的迭代过程.

乘积密码体制

设 $S_1 = (\mathcal{P}, \mathcal{P}, \mathcal{K}_1, \mathcal{E}_1, \mathcal{D}_1)$, $S_2 = (\mathcal{P}, \mathcal{P}, \mathcal{K}_2, \mathcal{E}_2, \mathcal{D}_2)$ 为两个具有相同明文空间的内嵌式密码体制, 其乘积密码体制定义为

$$S_1 \times S_2 \stackrel{\triangle}{=} (\mathcal{P}, \mathcal{P}, \mathcal{K}_1 \times \mathcal{K}_2, \mathcal{E}, \mathcal{D}),$$

其密钥形式为 $K = (K_1, K_2) \in \mathcal{K}_1 \times \mathcal{K}_2$. 对任意的 $K = (K_1, K_2)$, 加密和解密规则定义为:

$$E_K(x) = E_{K_2}(E_{K_1}(x))$$

和

$$D_K(y) = D_{K_1}(D_{K_2}(y)).$$

Examp-2-2-0

由以下定义的乘法密码体制 (记为 M) 和移位密码体制 (记为 S) 构造的乘积密码体制 $M \times S$ 就是仿射密码体制.

Examp-2-2-0

由以下定义的乘法密码体制 (记为 M) 和移位密码体制 (记为 S) 构造的乘积密码体制 $M \times S$ 就是仿射密码体制.

乘法密码

设 $\mathcal{P} = \mathcal{C} = \mathbb{Z}_{26}$,

$$\mathcal{K} = \{a \in \mathbb{Z}_{26} : \gcd(a, 26) = 1\}.$$

对于 $a \in \mathcal{K}$, 定义

$$E_a(x) = ax \bmod 26$$

和

$$D_a(y) = a^{-1}y \bmod 26.$$

乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;

乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;
2. 一个密码体制 S 称为幂等的, 如果 $S^2 \triangleq S \times S = S$.

乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;
2. 一个密码体制 S 称为幂等的, 如果 $S^2 \triangleq S \times S = S$. 前面研究过的移位密码、代换密码、仿射密码、希尔密码、维吉尼亚密码、置换密码以及乘法密码都是幂等的;

乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;
2. 一个密码体制 S 称为幂等的, 如果 $S^2 \triangleq S \times S = S$. 前面研究过的移位密码、代换密码、仿射密码、希尔密码、维吉尼亚密码、置换密码以及乘法密码都是幂等的;
3. 如果密码体制不是幂等的, 使用多次迭代后得到的密码体制就有可能获得更高的安全性;

乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;
2. 一个密码体制 S 称为幂等的, 如果 $S^2 \triangleq S \times S = S$. 前面研究过的移位密码、代换密码、仿射密码、希尔密码、维吉尼亚密码、置换密码以及乘法密码都是幂等的;
3. 如果密码体制不是幂等的, 使用多次迭代后得到的密码体制就有可能获得更高的安全性;
4. 一个构造非幂等的密码体制的方法是对两个不同的简单的密码体制做乘积;

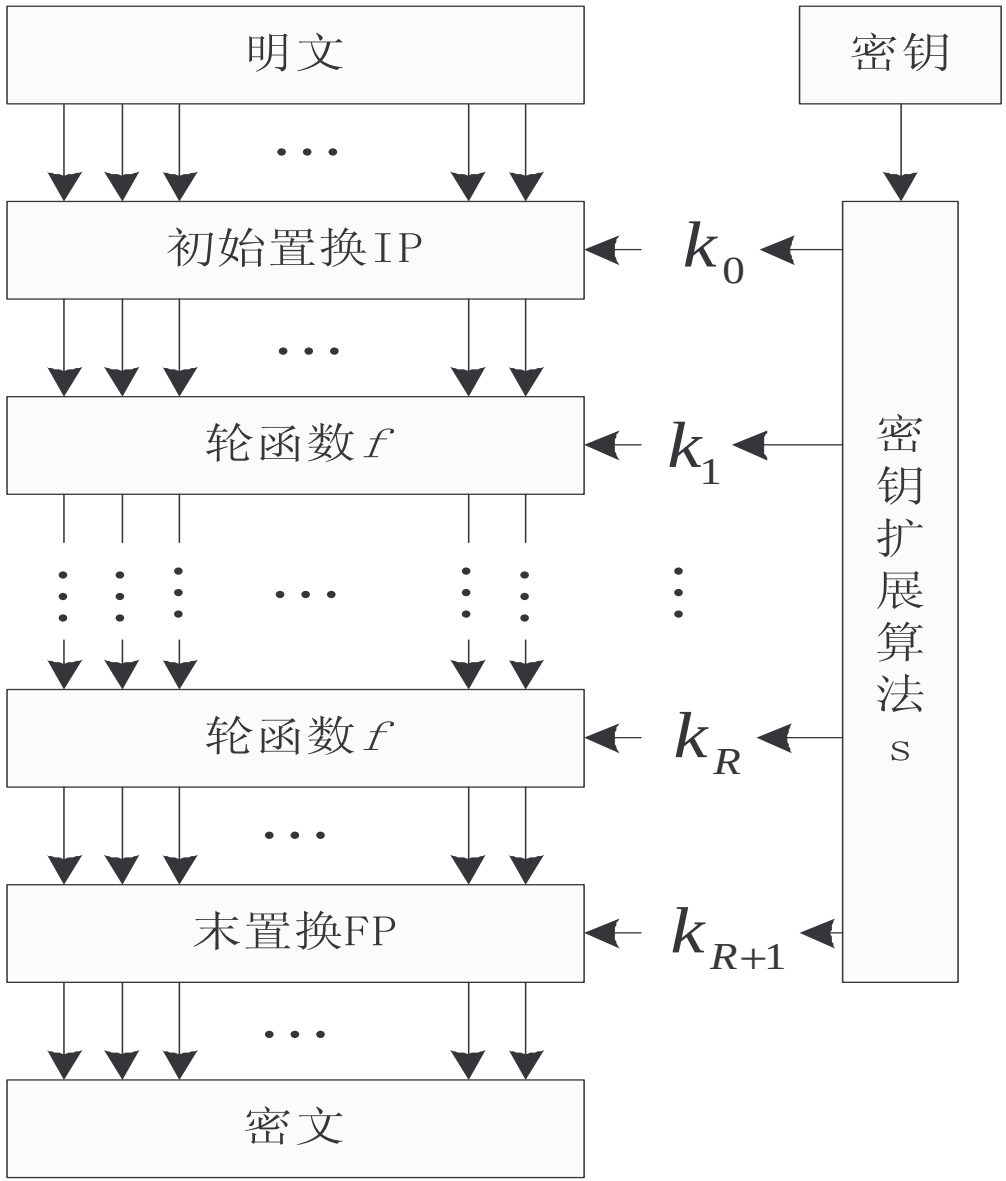
乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;
2. 一个密码体制 S 称为幂等的, 如果 $S^2 \triangleq S \times S = S$. 前面研究过的移位密码、代换密码、仿射密码、希尔密码、维吉尼亚密码、置换密码以及乘法密码都是幂等的;
3. 如果密码体制不是幂等的, 使用多次迭代后得到的密码体制就有可能获得更高的安全性;
4. 一个构造非幂等的密码体制的方法是对两个不同的简单的密码体制做乘积;
5. 幸运的是, 有许多简单的密码体制适合这种类型的构造, 通常使用的技术是将代换密码体制 与置换密码体制做乘积.

乘积密码体制相关概念

1. 如果 $S_1 \times S_2 = S_2 \times S_1$, 则称密码体制 S_1 和 S_2 可交换;
2. 一个密码体制 S 称为幂等的, 如果 $S^2 \triangleq S \times S = S$. 前面研究过的移位密码、代换密码、仿射密码、希尔密码、维吉尼亚密码、置换密码以及乘法密码都是幂等的;
3. 如果密码体制不是幂等的, 使用多次迭代后得到的密码体制就有可能获得更高的安全性;
4. 一个构造非幂等的密码体制的方法是对两个不同的简单的密码体制做乘积;
5. 幸运的是, 有许多简单的密码体制适合这种类型的构造, 通常使用的技术是将代换密码体制 与置换密码体制做乘积. DES, AES 就是通过这种方法构造的.

迭代密码的一般形式



迭代密码的两种类型

根据轮函数的形式, 迭代密码分为两种类型:

- Feistel 型密码: 代表密码是 DES
- 代换-置换网络 (SPN): 代表密码是 AES

本章内容

§ 3.2　代换–置换网络 (SPN)

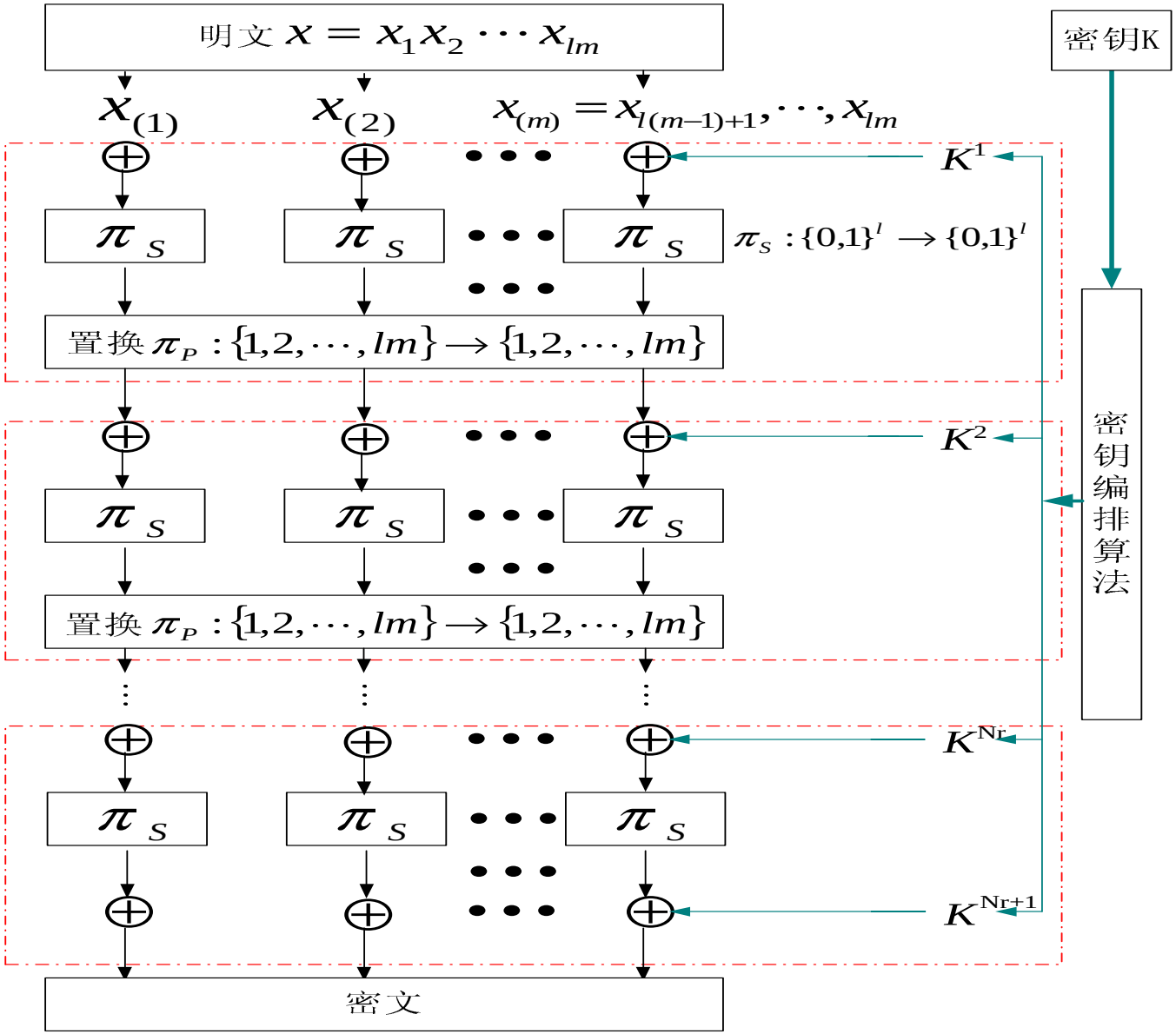
代换-置换网络的一般形式

$\pi_S : \{0,1\}^l \to \{0,1\}^l$, 双射, 非线性运算, 代换层, S 盒.
$\pi_P : \{1,2,\ldots,lm\} \to \{1,2,\ldots,lm\}$, 线性运算, 置换层.

代换-置换网络的一般形式

$\pi_S : \{0,1\}^l \to \{0,1\}^l$, 双射, 非线性运算, 代换层, S 盒.

$\pi_P : \{1,2,\ldots,lm\} \to \{1,2,\ldots,lm\}$, 线性运算, 置换层.



Examp-3-2-1

参数定义: $l = m = N_r = 4$

S 盒 π_S 定义: 输入 z 和输出 $\pi_S(z)$ 都以十六进制表示，即

$$0 \leftrightarrow \{0,0,0,0\}, 1 \leftrightarrow \{0,0,0,1\}, \cdots, F \leftrightarrow \{1,1,1,1\}$$

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

置换 $\pi_P : i + 4j \rightarrow 4i + j - 3, \quad 1 \leq i \leq 4, \quad 0 \leq j \leq 3$

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

密钥编排算法定义:

$$32 \text{ 比特密钥 } K = K_1||K_2||K_3||K_4||K_5||K_6||K_7||K_8$$

轮密钥:

$$K^1 = K_1||K_2||K_3||K_4 \quad K^2 = K_2||K_3||K_4||K_5 \quad K^3 = K_3||K_4||K_5||K_6$$
$$K^4 = K_4||K_5||K_6||K_7 \quad K^5 = K_5||K_6||K_7||K_8$$

Examp-3-2-1

参数定义: $l = m = N_r = 4$

S 盒 π_S 定义: 输入 z 和输出 $\pi_S(z)$ 都以十六进制表示，即

$$0 \leftrightarrow \{0,0,0,0\}, 1 \leftrightarrow \{0,0,0,1\}, \cdots, F \leftrightarrow \{1,1,1,1\}$$

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

置换 $\pi_P : i + 4j \rightarrow 4i + j - 3, \quad 1 \leq i \leq 4, \quad 0 \leq j \leq 3$

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

密钥编排算法定义:

$$32 \text{ 比特密钥 } K = K_1||K_2||K_3||K_4||K_5||K_6||K_7||K_8$$

轮密钥:

$$K^1 = K_1||K_2||K_3||K_4 \quad K^2 = K_2||K_3||K_4||K_5 \quad K^3 = K_3||K_4||K_5||K_6$$
$$K^4 = K_4||K_5||K_6||K_7 \quad K^5 = K_5||K_6||K_7||K_8$$

Examp-3-2-1

参数定义: $l = m = N_r = 4$

S 盒 π_S 定义: 输入 z 和输出 $\pi_S(z)$ 都以十六进制表示，即

$$0 \leftrightarrow \{0,0,0,0\}, 1 \leftrightarrow \{0,0,0,1\}, \cdots, F \leftrightarrow \{1,1,1,1\}$$

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

置换 $\pi_P : i + 4j \rightarrow 4i + j - 3, \quad 1 \leq i \leq 4, \quad 0 \leq j \leq 3$

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

密钥编排算法定义:

32 比特密钥 $K = K_1||K_2||K_3||K_4||K_5||K_6||K_7||K_8$

轮密钥:

$$K^1 = K_1||K_2||K_3||K_4 \quad K^2 = K_2||K_3||K_4||K_5 \quad K^3 = K_3||K_4||K_5||K_6$$
$$K^4 = K_4||K_5||K_6||K_7 \quad K^5 = K_5||K_6||K_7||K_8$$

Examp-3-2-1

参数定义: $l = m = N_r = 4$

S 盒 π_S 定义: 输入 z 和输出 $\pi_S(z)$ 都以十六进制表示，即

$$0 \leftrightarrow \{0,0,0,0\}, 1 \leftrightarrow \{0,0,0,1\}, \cdots, F \leftrightarrow \{1,1,1,1\}$$

z	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$\pi_S(z)$	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

置换 $\pi_P : i + 4j \rightarrow 4i + j - 3, \quad 1 \leq i \leq 4, \quad 0 \leq j \leq 3$

z	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\pi_P(z)$	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

密钥编排算法定义:

$$32 \text{ 比特密钥 } K = K_1||K_2||K_3||K_4||K_5||K_6||K_7||K_8$$

轮密钥:

$$K^1 = K_1||K_2||K_3||K_4 \quad K^2 = K_2||K_3||K_4||K_5 \quad K^3 = K_3||K_4||K_5||K_6$$
$$K^4 = K_4||K_5||K_6||K_7 \quad K^5 = K_5||K_6||K_7||K_8$$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$
密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$

密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

加密过程:

$$
\begin{array}{lllll}
x & & = & 0010\ 0110\ 1011\ 0111 & \\
K^1 & & = & 0011\ 1010\ 1001\ 0100 & \\
u^1 & = x \oplus K^1 & = & 0001\ 1100\ 0010\ 0011 & = 1\ C\ 2\ 3 \\
v^1 & = \pi_S(u^1) & = & 0100\ 0101\ 1101\ 0001 & = 4\ 5\ D\ 1 \\
w^1 & = \pi_P(v^1) & = & 0010\ 1110\ 0000\ 0111 & \\
 & & \vdots & & \\
w^3 & & = & 1110\ 0100\ 0110\ 1110 & \\
K^4 & & = & 0100\ 1101\ 0110\ 0011 & \\
u^4 & = w^3 \oplus K^4 & = & 1010\ 1001\ 0000\ 1101 & = A\ 9\ 0\ B \\
v^4 & = \pi_S(u^4) & = & 0110\ 1010\ 1110\ 1001 & = 6\ A\ E\ C \\
K^5 & & = & 1101\ 0110\ 0011\ 1111 & \\
y & = v^4 \oplus K^5 & = & 1011\ 1100\ 1101\ 0110 &
\end{array}
$$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$

密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

加密过程:

$$
\begin{array}{lcll}
x & & = & 0010\ 0110\ 1011\ 0111 \\
K^1 & & = & 0011\ 1010\ 1001\ 0100 \\
u^1 & = x \oplus K^1 & = & 0001\ 1100\ 0010\ 0011 \quad = \quad 1\ C\ 2\ 3 \\
v^1 & = \pi_S(u^1) & = & 0100\ 0101\ 1101\ 0001 \quad = \quad 4\ 5\ D\ 1 \\
w^1 & = \pi_P(v^1) & = & 0010\ 1110\ 0000\ 0111 \\
 & & \vdots & \\
w^3 & & = & 1110\ 0100\ 0110\ 1110 \\
K^4 & & = & 0100\ 1101\ 0110\ 0011 \\
u^4 & = w^3 \oplus K^4 & = & 1010\ 1001\ 0000\ 1101 \quad = \quad A\ 9\ 0\ B \\
v^4 & = \pi_S(u^4) & = & 0110\ 1010\ 1110\ 1001 \quad = \quad 6\ A\ E\ C \\
K^5 & & = & 1101\ 0110\ 0011\ 1111 \\
y & = v^4 \oplus K^5 & = & 1011\ 1100\ 1101\ 0110
\end{array}
$$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$

密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

加密过程:

$$
\begin{array}{lllll}
x & & = & 0010\ 0110\ 1011\ 0111 & \\
K^1 & & = & 0011\ 1010\ 1001\ 0100 & \\
u^1 & = x \oplus K^1 & = & 0001\ 1100\ 0010\ 0011 & = 1\ C\ 2\ 3 \\
v^1 & = \pi_S(u^1) & = & 0100\ 0101\ 1101\ 0001 & = 4\ 5\ D\ 1 \\
w^1 & = \pi_P(v^1) & = & 0010\ 1110\ 0000\ 0111 & \\
 & & \vdots & & \\
w^3 & & = & 1110\ 0100\ 0110\ 1110 & \\
K^4 & & = & 0100\ 1101\ 0110\ 0011 & \\
u^4 & = w^3 \oplus K^4 & = & 1010\ 1001\ 0000\ 1101 & = A\ 9\ 0\ B \\
v^4 & = \pi_S(u^4) & = & 0110\ 1010\ 1110\ 1001 & = 6\ A\ E\ C \\
K^5 & & = & 1101\ 0110\ 0011\ 1111 & \\
y & = v^4 \oplus K^5 & = & 1011\ 1100\ 1101\ 0110 &
\end{array}
$$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$

密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

加密过程:

$$
\begin{array}{lllll}
x & & = & 0010\ 0110\ 1011\ 0111 & \\
K^1 & & = & 0011\ 1010\ 1001\ 0100 & \\
u^1 & = x \oplus K^1 & = & 0001\ 1100\ 0010\ 0011 & = 1\ C\ 2\ 3 \\
v^1 & = \pi_S(u^1) & = & 0100\ 0101\ 1101\ 0001 & = 4\ 5\ D\ 1 \\
w^1 & = \pi_P(v^1) & = & 0010\ 1110\ 0000\ 0111 & \\
 & & \vdots & & \\
w^3 & & = & 1110\ 0100\ 0110\ 1110 & \\
K^4 & & = & 0100\ 1101\ 0110\ 0011 & \\
u^4 & = w^3 \oplus K^4 & = & 1010\ 1001\ 0000\ 1101 & = A\ 9\ 0\ B \\
v^4 & = \pi_S(u^4) & = & 0110\ 1010\ 1110\ 1001 & = 6\ A\ E\ C \\
K^5 & & = & 1101\ 0110\ 0011\ 1111 & \\
y & = v^4 \oplus K^5 & = & 1011\ 1100\ 1101\ 0110 &
\end{array}
$$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$

密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

加密过程:

$$
\begin{array}{lllll}
x & & = & 0010\ 0110\ 1011\ 0111 & \\
K^1 & & = & 0011\ 1010\ 1001\ 0100 & \\
u^1 & = x \oplus K^1 & = & 0001\ 1100\ 0010\ 0011 & = 1\ C\ 2\ 3 \\
v^1 & = \pi_S(u^1) & = & 0100\ 0101\ 1101\ 0001 & = 4\ 5\ D\ 1 \\
w^1 & = \pi_P(v^1) & = & 0010\ 1110\ 0000\ 0111 & \\
 & & \vdots & & \\
w^3 & & = & 1110\ 0100\ 0110\ 1110 & \\
K^4 & & = & 0100\ 1101\ 0110\ 0011 & \\
u^4 & = w^3 \oplus K^4 & = & 1010\ 1001\ 0000\ 1101 & = A\ 9\ 0\ B \\
v^4 & = \pi_S(u^4) & = & 0110\ 1010\ 1110\ 1001 & = 6\ A\ E\ C \\
K^5 & & = & 1101\ 0110\ 0011\ 1111 & \\
y & = v^4 \oplus K^5 & = & 1011\ 1100\ 1101\ 0110 &
\end{array}
$$

Examp-3-2-1 (Cont.)

输入: 明文: $x = 0010\ 0110\ 1011\ 0111$

密钥: $K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$

加密过程:

$$
\begin{array}{lllll}
x & & = & 0010\ 0110\ 1011\ 0111 & \\
K^1 & & = & 0011\ 1010\ 1001\ 0100 & \\
u^1 & = x \oplus K^1 & = & 0001\ 1100\ 0010\ 0011 & = 1\ C\ 2\ 3 \\
v^1 & = \pi_S(u^1) & = & 0100\ 0101\ 1101\ 0001 & = 4\ 5\ D\ 1 \\
w^1 & = \pi_P(v^1) & = & 0010\ 1110\ 0000\ 0111 & \\
 & & \vdots & & \\
w^3 & & = & 1110\ 0100\ 0110\ 1110 & \\
K^4 & & = & 0100\ 1101\ 0110\ 0011 & \\
u^4 & = w^3 \oplus K^4 & = & 1010\ 1001\ 0000\ 1101 & = A\ 9\ 0\ B \\
v^4 & = \pi_S(u^4) & = & 0110\ 1010\ 1110\ 1001 & = 6\ A\ E\ C \\
K^5 & & = & 1101\ 0110\ 0011\ 1111 & \\
y & = v^4 \oplus K^5 & = & 1011\ 1100\ 1101\ 0110 &
\end{array}
$$

SPN 的特点: 简单有效

- S 盒 π_S 可以用查表方式实现, $\pi_S : \{0,1\}^l \to \{0,1\}^l$ 所需的存储空间是 $l2^l$.

SPN 的特点: 简单有效

- S 盒 π_S 可以用查表方式实现, $\pi_S : \{0,1\}^l \to \{0,1\}^l$ 所需的存储空间是 $l2^l$.
- S 盒 π_S 的作用是把明文和密钥局部混淆, 是非线性运算, π_P 起全局扩散作用, 是线性运算.

SPN 的变体

- 使用多个 S 盒 π_S，如 DES 中使用了 8 个不同的 S 盒.

SPN 的变体

- 使用多个 S 盒 π_S，如 DES 中使用了 8 个不同的 S 盒.
- 每一轮中可包含一个可逆的线性运算, 该线性变换要么代替置换 π_P, 要么作为 π_P 的补充.

§ 3.5　数据加密标准 (DES)

DES 的历史

1. 1973 年 5 月 15 日美国国家标准局 (现为美国国家标准技术研究所, 即 NIST) 公开征集密码体制。

DES 的历史

1. 1973 年 5 月 15 日美国国家标准局 (现为美国国家标准技术研究所, 即 NIST) 公开征集密码体制。
2. 1977 年 2 月 15 日由 IBM 开发的 DES 被选择为标准，用在“非密级”应用中，DES 是对早期版本 Lucifer 的改进。

DES 的历史

1. 1973 年 5 月 15 日美国国家标准局 (现为美国国家标准技术研究所, 即 NIST) 公开征集密码体制。
2. 1977 年 2 月 15 日由 IBM 开发的 DES 被选择为标准，用在“非密级”应用中，DES 是对早期版本 Lucifer 的改进。
3. 每隔 5 年对 DES 进行一次评审。

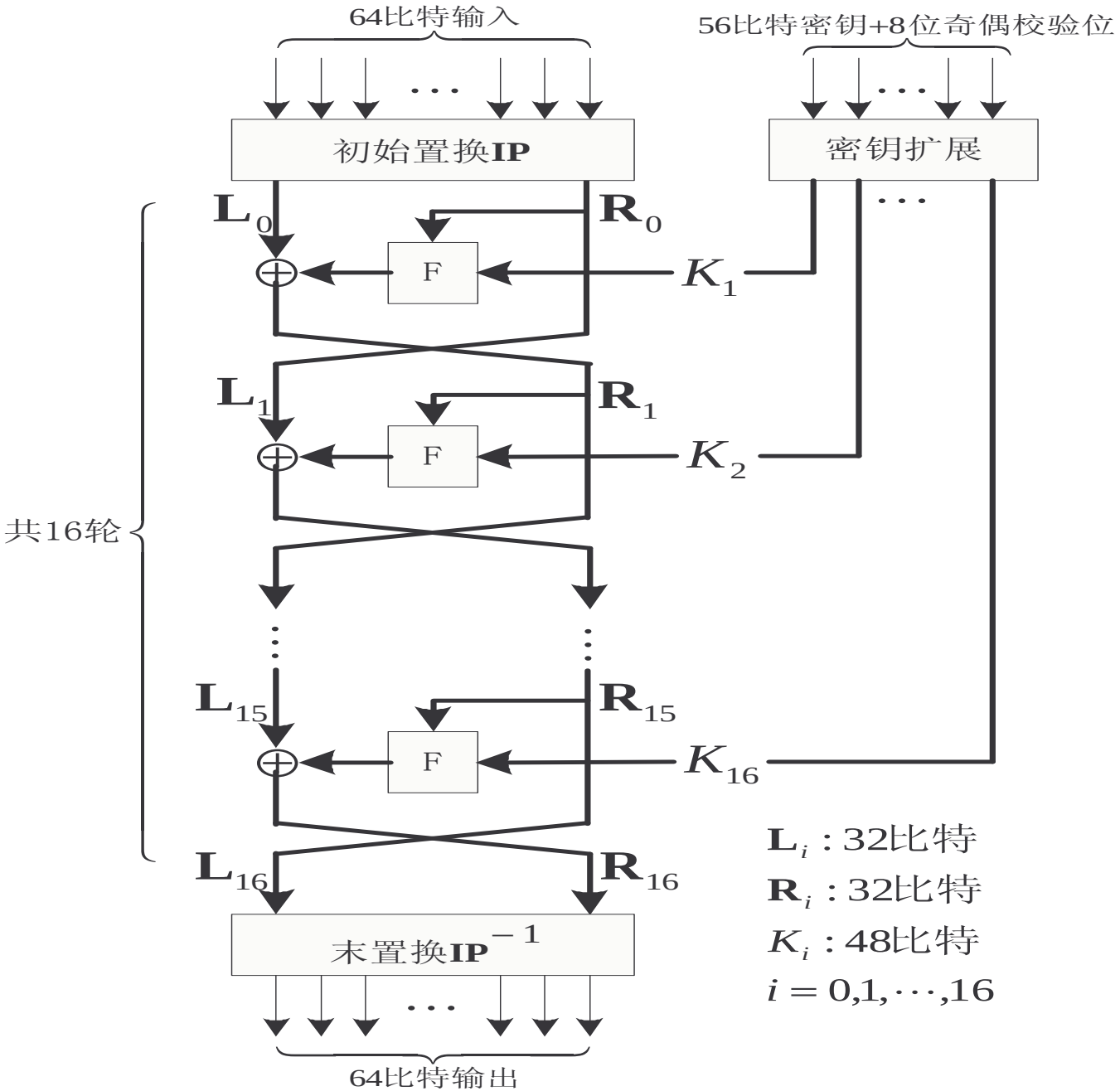
DES 的历史

1. 1973 年 5 月 15 日美国国家标准局 (现为美国国家标准技术研究所, 即 NIST) 公开征集密码体制。
2. 1977 年 2 月 15 日由 IBM 开发的 DES 被选择为标准，用在“非密级”应用中，DES 是对早期版本 Lucifer 的改进。
3. 每隔 5 年对 DES 进行一次评审。
4. 在 1999 年 1 日对 DES 最后一次评审。

DES 的历史

1. 1973 年 5 月 15 日美国国家标准局 (现为美国国家标准技术研究所, 即 NIST) 公开征集密码体制。
2. 1977 年 2 月 15 日由 IBM 开发的 DES 被选择为标准，用在“非密级”应用中，DES 是对早期版本 Lucifer 的改进。
3. 每隔 5 年对 DES 进行一次评审。
4. 在 1999 年 1 日对 DES 最后一次评审。
5. 2001 年 11 月 26 日 DES 最终被 AES 代替。

DES 算法描述: 算法结构图



DES 算法描述: 初始置换 (IP)

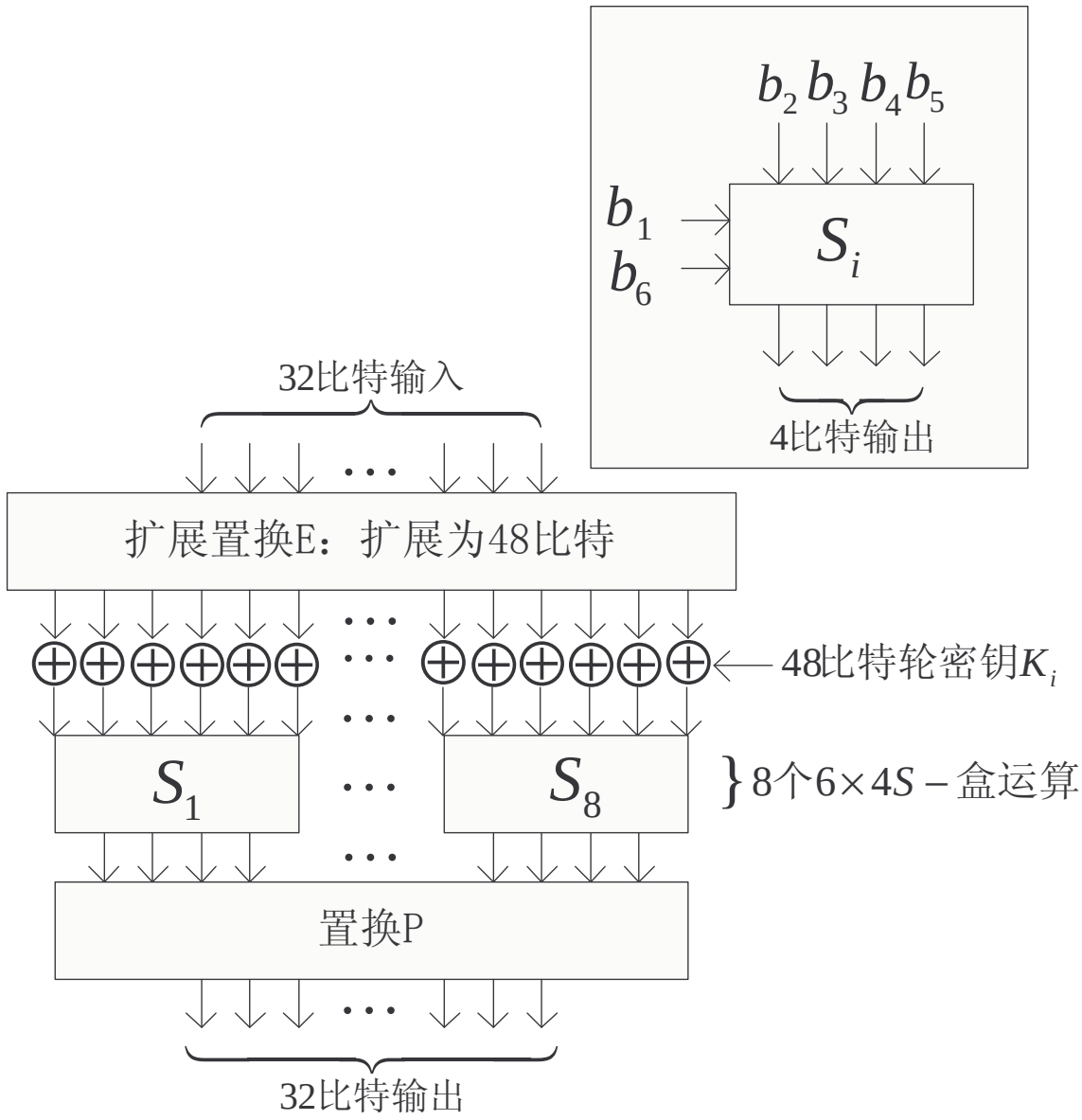
(a) Initial Permutation (IP)

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

(b) Inverse Initial 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

DES 算法描述: F 函数


$b_2\ b_3\ b_4\ b_5$
b_1
b_6
S_i
4比特输出
32比特输入
扩展置换E：扩展为48比特
48比特轮密钥K_i
S_1
S_8
8个6×4 S – 盒运算
置换P
32比特输出

DES 算法描述: 扩展置换 *E* 和置换 *P*

扩展置换 *E*:

32	01	02	03	04	05
04	05	06	07	08	09
08	09	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	01

置换 *P*:

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

DES 算法描述: 8 个 S 盒

S_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

S_2

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

S_3

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

S_4

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

S_5

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

S_6

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

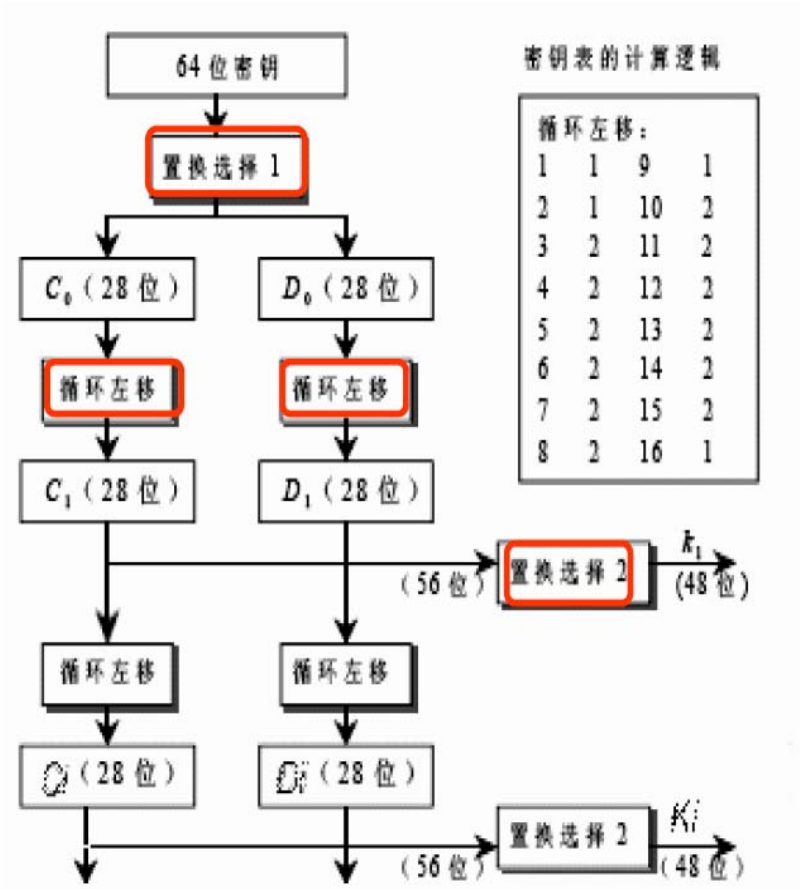
S_7

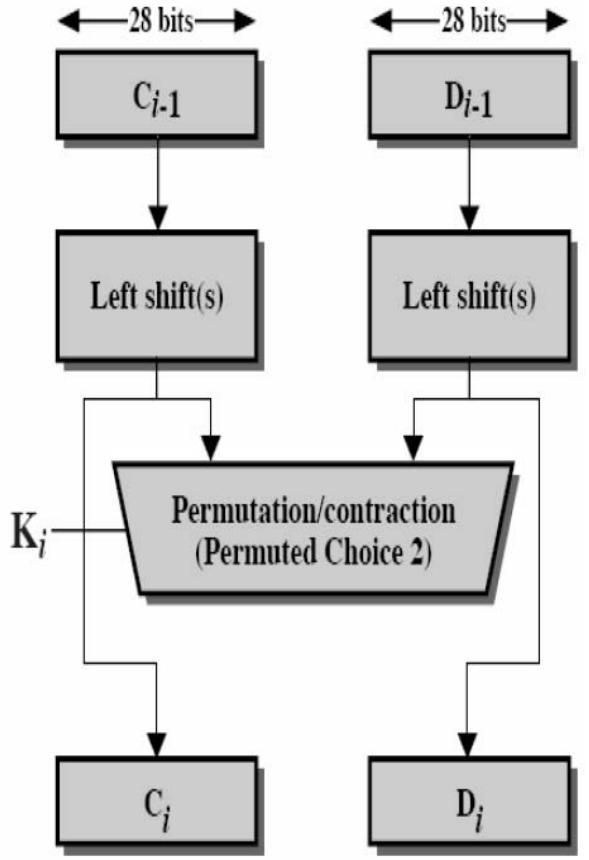
4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S_8

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

DES 算法描述: 密钥编排算法





DES 算法描述: 置换选择 1 和 2

置换选择 1: 64 位 $\Longrightarrow$ 56 位, 舍弃了奇偶校验位 (即第 8, 16, $\cdots$, 64 位).

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

置换选择 2: 56 位 $\Longrightarrow$ 48 位, 舍弃了第 9, 18, 22, 25, 35, 38, 43, 54 比特位.

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

S 盒的“陷门”争议

- S 盒是整个 DES 算法的关键部件，DES 靠它实现非线性变换。关于 S 盒的设计准则当时没有完全公开。
- 许多密码学家怀疑美国国家安全局 (NSA) 设计 S 盒时隐藏了“陷门”，这样只有他们才可以破译算法，但没有证据能标明这点。
- 在 1976 年, NSA 披露了 S 盒几条设计准则. 到 1990 年, Biham 和 Shamir 提出对 DES 的差分密码分析后, IBM 公布了 S 盒和 P 置换的设计准则.

DES 的分析

- 对 DES 最有效的攻击还是强力攻击，因为无论对差分或是线性密码分析，所需要的海量选择明文–密文对是不现实的，而且单是存贮明文–密文对就至少需要 140,000 GB。
- 美国克罗多州的程序员 Verser 从 1997 年 3 月 13 日起，用了 96 天的时间，在 Internet 上数万名志愿者的协同工作下，于 6 月 17 日成功地找到了美国 RSA 公司悬赏一万美金破译的密钥长度为 56 比特的 DES 的密钥。
- 1998 年电子边境基金会 (EFF) 使用一台 25 万美元的电脑在 56 小时内再次破解了 56 比特的 DES。
- 1999 年 1 月 RSA 数据安全会议期间，EFF 用 22 小时 15 分钟就宣告成功破解 DES。
- 如果 DES 的密钥长度为 128 比特，那么它可以在 1018 年内攻破。

§ 3.6　高级加密标准 (AES)

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

AES 的遴选过程

1. 1997 年 1 月，美国 NIST 开始对 AES 进行研究，并成立了 AES 工作室。
2. 1997 年 9 月 12 日，NIST 发布征集算法的正式公告，至 1998 年 6 月 15 日收到了 21 个侯选算法。
3. 1998 年 8 月 20 日召开第一次 AES 候选会议，确定了来自全世界的 15 个候选算法。
4. 1999 年 3 月开始的第二次候选会议，选出了五个决赛算法：MARS, RC6, Rijndael, Serpent 和 Twofish。
5. 2000 年 4 月召开第三次 AES 候选会议, 同年 10 月 2 日美国商业部长宣布 Rijndeal 最终获胜。
6. 2001 年 11 月 26 日，NIST 发布了联邦信息处理标准，正式公告了 AES，2002 年 5 月 26 日公告正式生效。
7. 每隔 5 年重新进行一次正式评估。

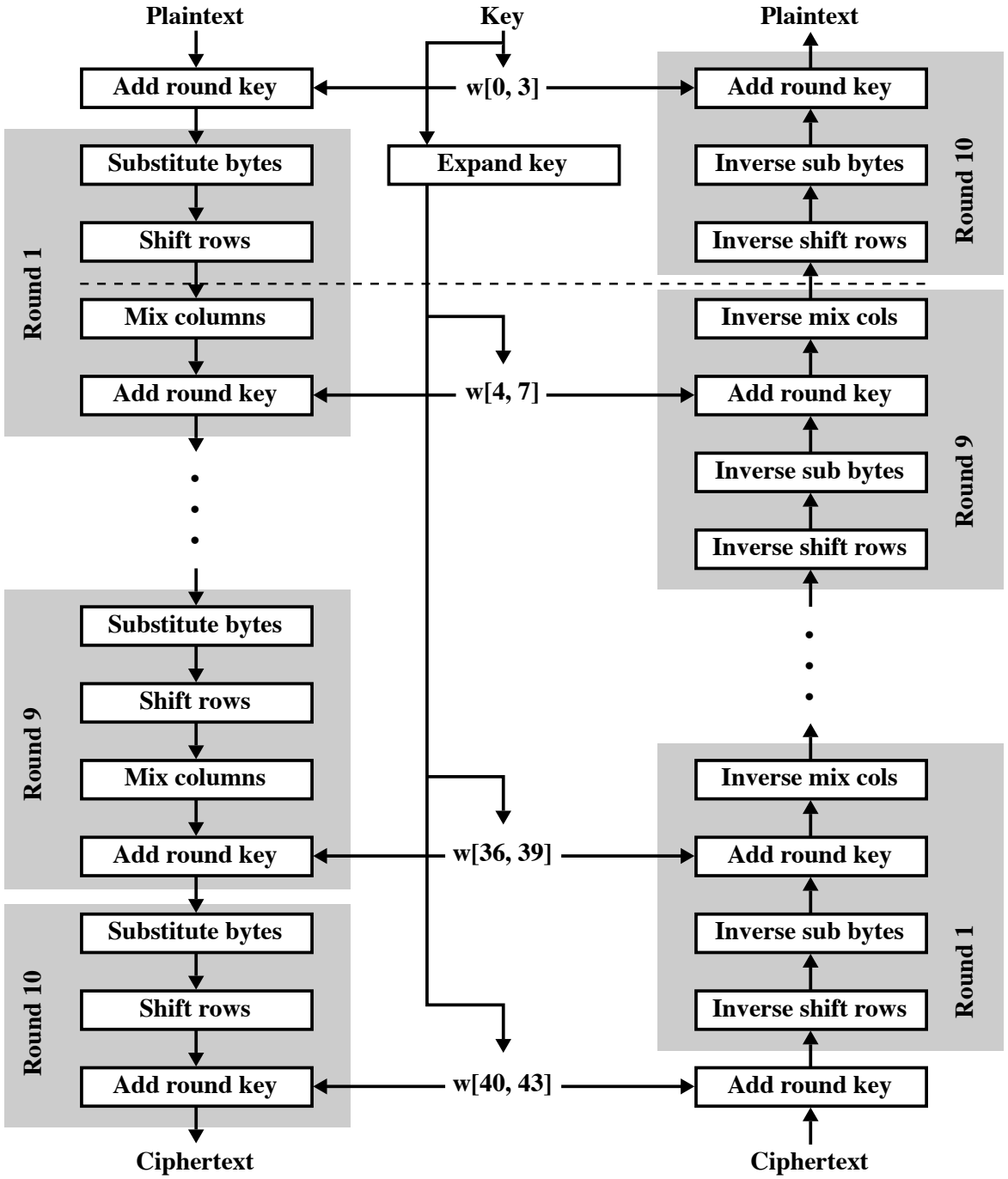
AES 的遴选要求

1. 公开
2. 分组加密单钥体制, 支持 128 比特长分组和 128, 192 和 256 比特三种不同长度密钥
3. 可用软件和硬件实现
4. 可自由使用，或依据符合美国 NIST 策略的条件使用

Rijndeal 胜出原因

- 其软、硬件实现对计算环境的适应性强，性能稳定、优良。
- 密钥建立时间短，密钥灵活性好。
- 存储量要求低使它适合资源紧缺环境，且保持优秀的性能。
- Rijndeal 在所有提交算法中最易于抵抗能量和计时攻击。
- Rijndeal 可灵活组合不同的分组长和密钥长，对算法所作的改动仅是增加它的轮数。

128 比特密钥 (10 轮) AES 算法结构



AES 中的四个运算

- S 盒运算：SubByte
- 行移位：ShiftRow
- 列混合：MixColumn
- 轮密钥异或：AddRoundKey

S 盒运算：SubByte

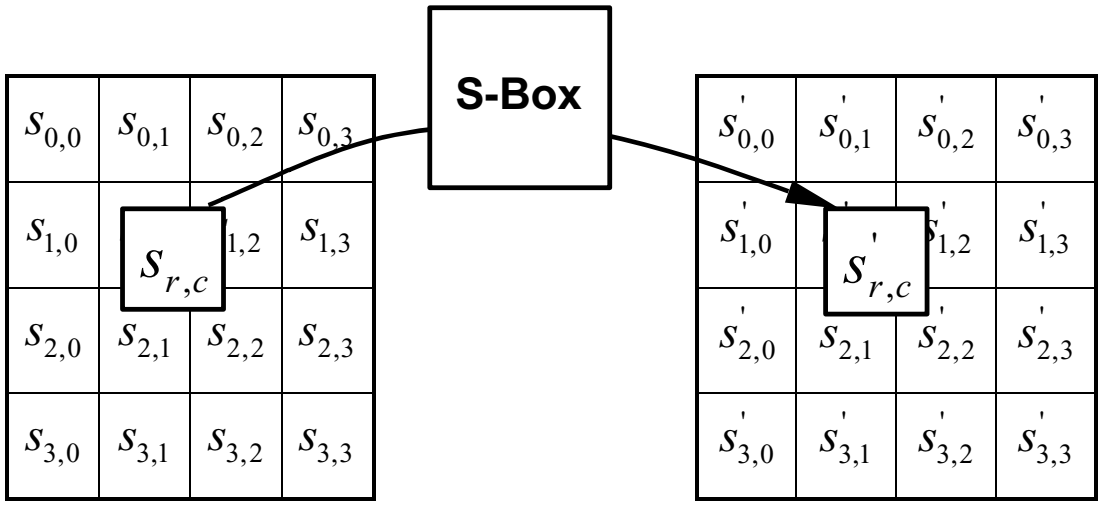


Figure: `SubByte()`作用在状态的单个字节上

Table 5.4 AES S-Boxes

(a) S-box

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

(b) Inverse S-box

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

行移位：ShiftRow

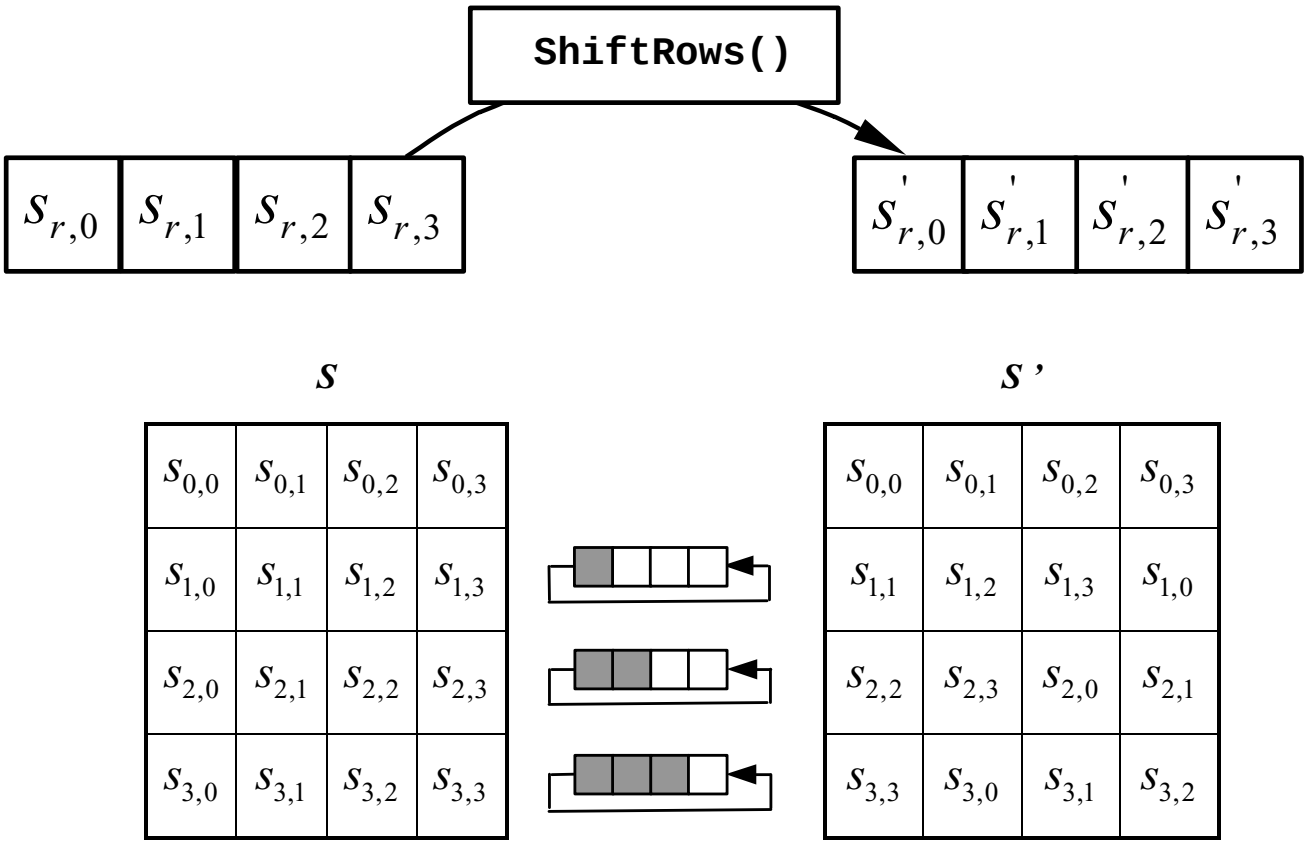


Figure: ShiftRow()作用在状态的行上

列混合：MixColumn

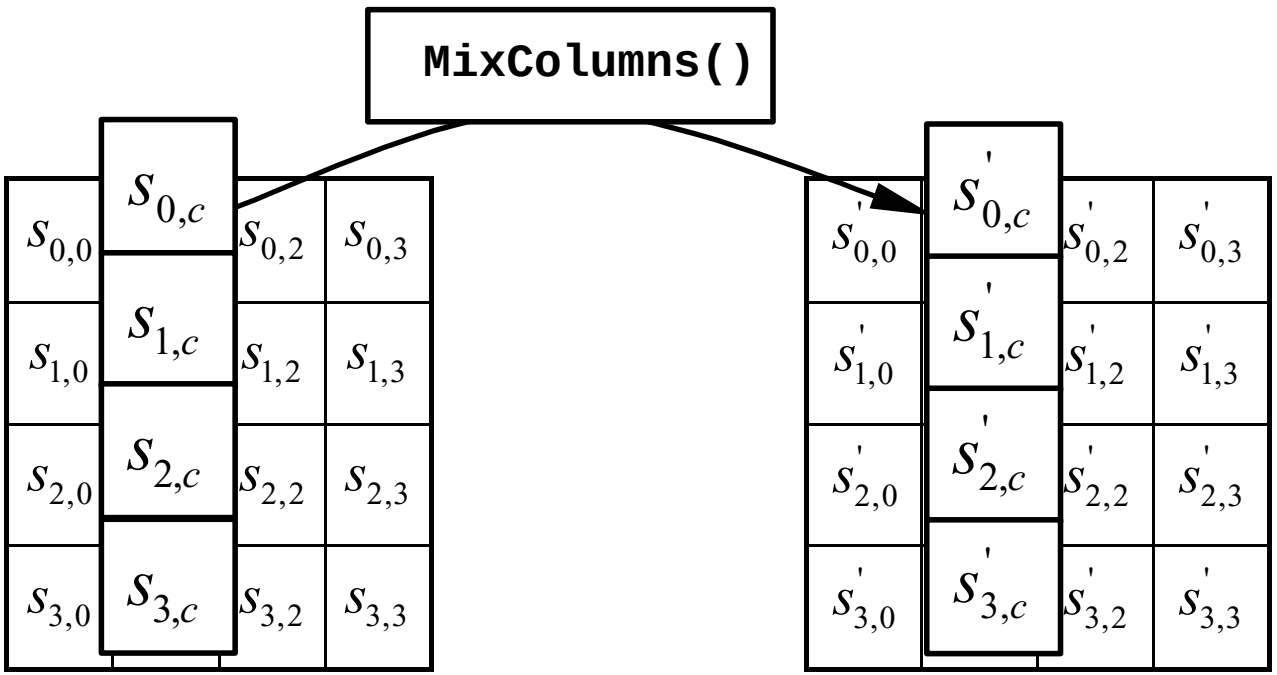


Figure: `MixColumn()`在状态的列上运算

密钥异或：AddRoundKey

$$
\begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} \oplus \begin{bmatrix} k_{0,0} & k_{0,1} & k_{0,2} & k_{0,3} \\ k_{1,0} & k_{1,1} & k_{1,2} & k_{1,3} \\ k_{2,0} & k_{2,1} & k_{2,2} & k_{2,3} \\ k_{3,0} & k_{3,1} & k_{3,2} & k_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}
$$

密钥编排方案 (书上算法 3.6)

KeyExpansion(key): // |key|=128, |key[i]|=8

external RotWord (字节循环左移)，SubWord (对每个字节进行 AES 的 S 盒代换)

常数 RCon[1], . . . , RCon[10] // |RCon[i]|=32

for $i \leftarrow 0$ to 3 // |w[i]|=32
do $w[i] \leftarrow (\text{key}[4i], \text{key}[4i+1], \text{key}[4i+2], \text{key}[4i+3])$
for $i \leftarrow 4$ **to** 43

$$\textbf{do} \begin{cases} \text{temp} \leftarrow w[i-1] \\ \text{if } i \equiv 0 \pmod 4 \\ \quad \text{then } \text{temp} \leftarrow \text{SubWord}(\text{RotWord}(\text{temp})) \oplus \text{Rcon}[i/4] \\ w[i] \leftarrow w[i-4] \oplus \text{temp} \end{cases}$$

return $w[0], \ldots, w[43]$

AES 解密函数

1. 所有操作逆序进行
2. 轮密钥顺序与加密函数的轮密钥相反
3. 所有操作均为加密函数的逆操作

AES 的分析

能抵抗所有已知攻击，如线性和差分分析：

- S 盒使得的线性逼近和差分分布表趋于均匀；
- 列混合使用了宽轨道策略，使得找到包含较少活动 S 盒的差分和线性分析成为不可能。

§3.7　工作模式

考虑作为 AES 标准的工作模式

- 电码本模式（ECB 模式）
- 密码分组链接模式（CBC 模式）
- 输出反馈模式（OFB 模式）
- 密码反馈模式（CFB 模式）
- 计数模式（CTR）
- 计数密码分组链接模式 (CCM 模式), 它是计数模式和 CBC 模式的组合使用.

电码本模式（ECB 模式）

对给定的明文分组序列：

$$x_1, x_2, \ldots$$

密文序列为：

$$y_1 = E_K(x_1), y_2 = E_K(x_2), \ldots$$

解密：

$$x_1 = D_K(y_1), x_2 = D_K(y_2), \ldots$$

密码分组链接模式（CBC 模式）

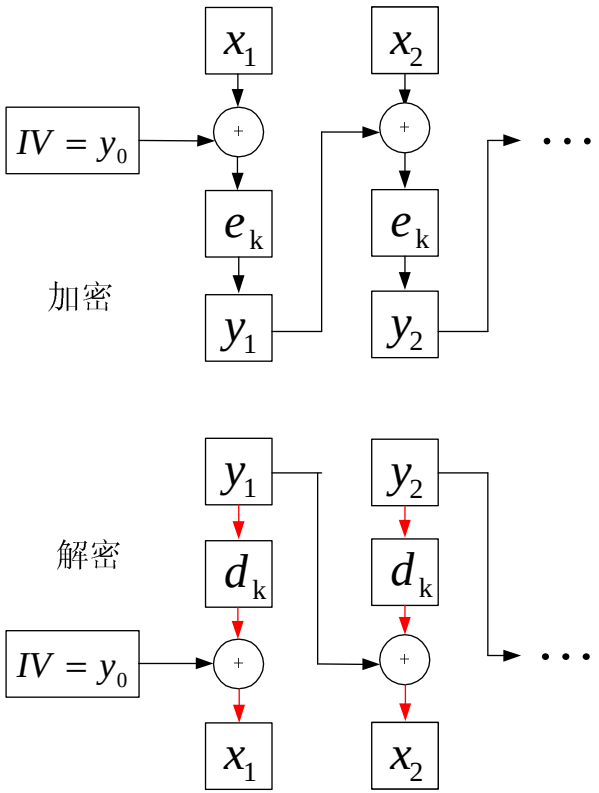
初始向量：IV

加密：

$$
\begin{aligned}
y_0 &= IV \\
y_1 &= E_K(y_0 \oplus x_1) \\
y_2 &= E_K(y_1 \oplus x_2) \\
&\vdots
\end{aligned}
$$

解密：

$$
\begin{aligned}
y_0 &= IV \\
x_1 &= D_K(y_1) \oplus y_0 \\
x_2 &= D_K(y_2) \oplus y_1 \\
&\vdots
\end{aligned}
$$



输出反馈模式（OFB 模式）

初始向量：IV

加密：

$$\begin{aligned} z_0 &= IV, \quad z_i = E_K(z_{i-1}) \\ y_i &= x_i \oplus z_i, \quad i = 1, 2, \ldots \end{aligned}$$

解密：

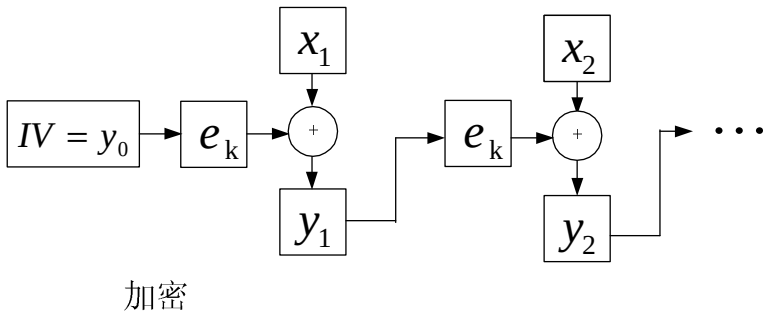
$$\begin{aligned} z_0 &= IV, \quad z_i = E_K(z_{i-1}) \\ x_i &= y_i \oplus z_i, \quad i = 1, 2, \ldots \end{aligned}$$

密码反馈模式（CFB 模式）

初始向量：IV

加密：

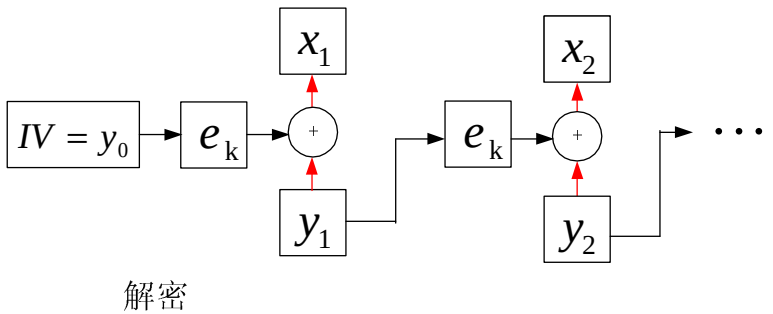
$$\begin{aligned} y_0 &= IV, \quad z_i = E_K(y_{i-1}) \\ y_i &= x_i \oplus z_i, \quad i = 1, 2, \ldots \end{aligned}$$



加密

解密：

$$\begin{aligned} y_0 &= IV, \quad z_i = E_K(y_{i-1}) \\ x_i &= y_i \oplus z_i, \quad i = 1, 2, \ldots \end{aligned}$$



解密

计数模式（CTR 模式）

计数模式类似于 OFB 模式，差别是密钥流产生机制不同。

计数模式（CTR 模式）

计数模式类似于 OFB 模式，差别是密钥流产生机制不同。

计数模式的工作方式：

1. 计数器 ctr：m 长比特串
2. 由 ctr 递归构造 m 长比特串：

$$T_i = ctr + i - 1 \bmod 2^m$$

3. 对所有的 $i \geq 1$，如下加密明文分组 $x_1, x_2, \ldots$

$$y_i = x_i \oplus E_K(T_i)$$

改变一个明文分组对四种工作模式的影响

- ECB：只影响当前分组，但相同的明文分组产生相同的密文分组的特点有时是一个严重的安全性弱点
- CBC：当前分组和后续分组都受影响，可用作认证码
- OFB：只影响当前分组，可用在卫星通信中
- CFB：当前分组和后续分组都受影响，可用作认证码

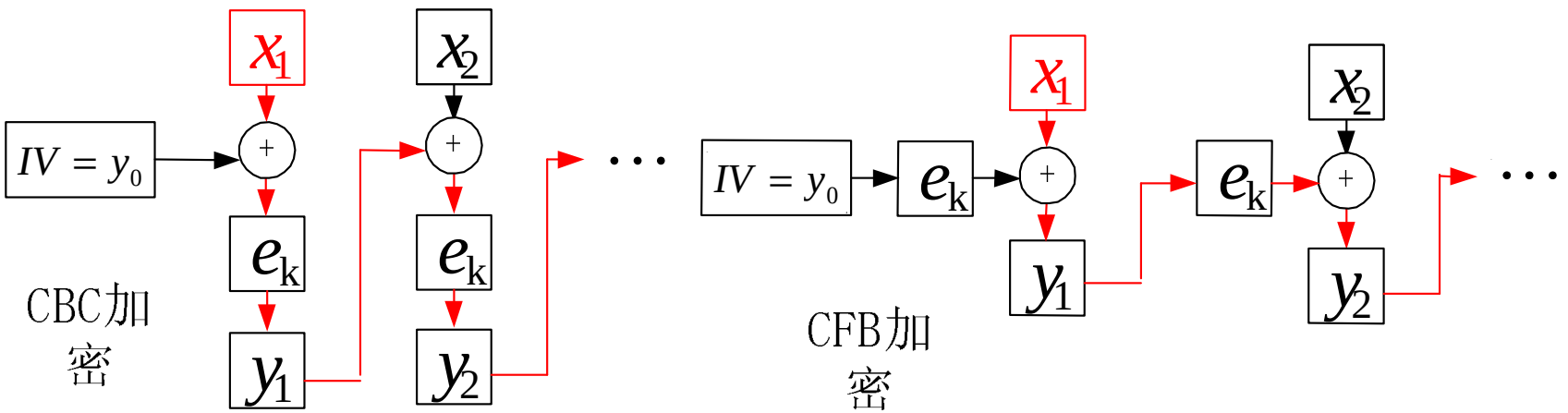


Figure: 一个明文分组的错误对密文的影响

作业 5 (共 2 题)

5.1. 此题目给出了一轮 DES 加密的例子. 假设明文和密钥都是

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

1. 推导第一轮的子密钥 K_1;
2. 计算 L_0, R_0 和 $E(R_0)$;
3. 计算 $E(R_0) \bigoplus K_1$ (结果记为 A);
4. 把得到的 A 分成 6 位 (数据) 的集合, 求对应 S 盒代换的值 (结果串连之后记为 B);
5. 应用置换求 $P(B)$;
6. 计算 $R_1 = P(B) \bigoplus L_0$;
7. 写出密文 $L_1 || R_1$.

作业 5 (共 2 题)

5.2. (书上习题 3.5) 用下列十六进制表示的 128 比特的 AES 种子密钥构造一个完整的密钥编排方案:

2B7E 1516 28AE D2A6 ABF7 1588 09CF 4F3C.