



امنیت داده و شبکه

رمزنگاری متقارن (مدرن)

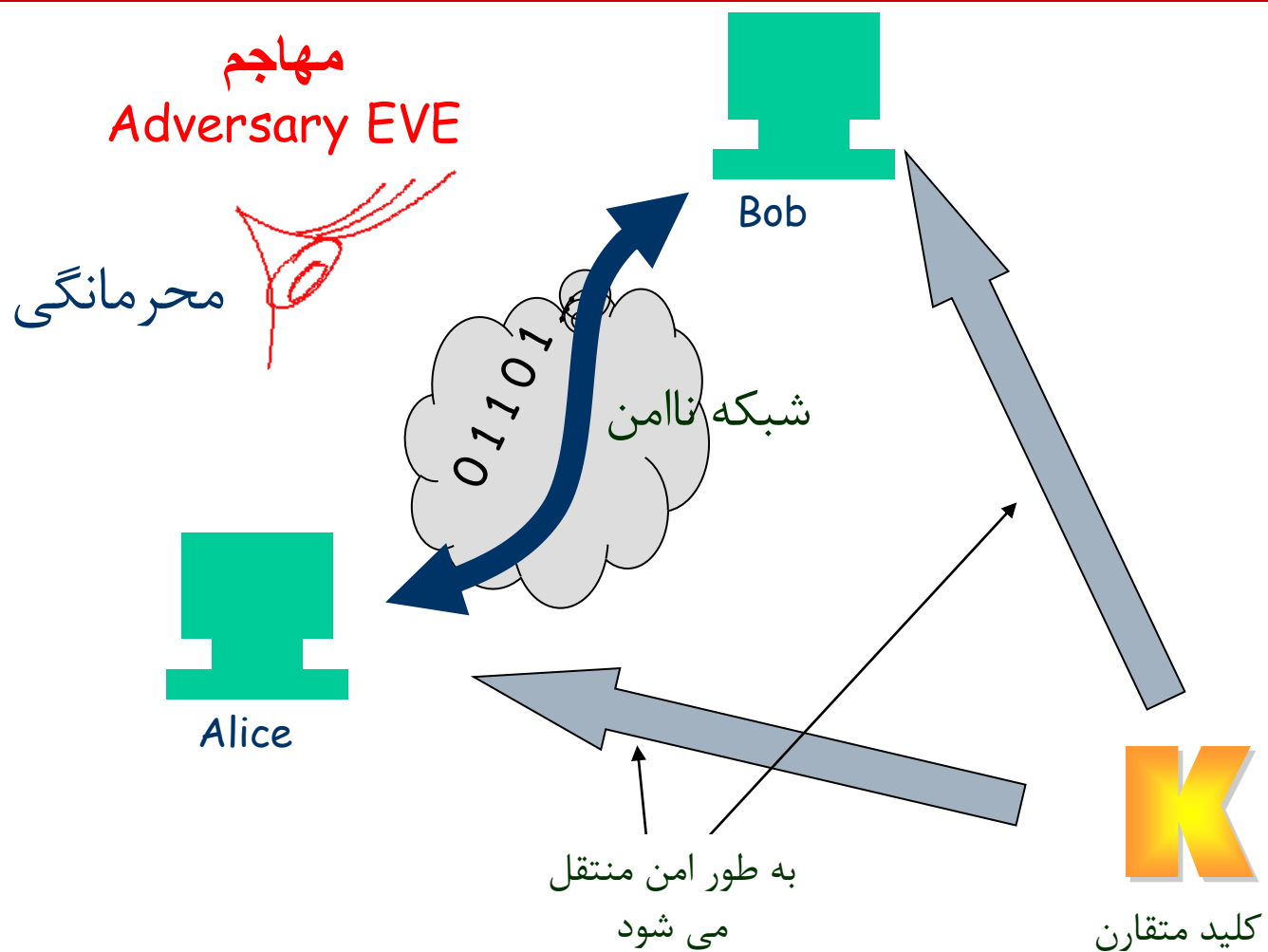


فهرست مطالب

- رمزهای متقارن و قالبی
- ساختار رمزهای فیستل
- استاندارد رمزگذاری داده DES
- الگوریتم رمز 2DES و 3DES
- استاندارد رمزگذاری پیشرفته AES
- رمزهای متقارن معروف
- مدهای کاری رمزهای متقارن



رمزنگاری متقارن





الگوریتم‌های رمز متقارن

□ رمزهای متقارن را می‌توان با دو روش عمده تولید کرد:

■ رمزهای قطعه‌ای یا قالبی (Block Cipher)

□ پردازش پیغام‌ها بصورت قطعه به قطعه

□ اندازه متعارف مود استفاده برای قطعات ۶۴، ۱۲۸ یا ۲۵۶ بیتی است.

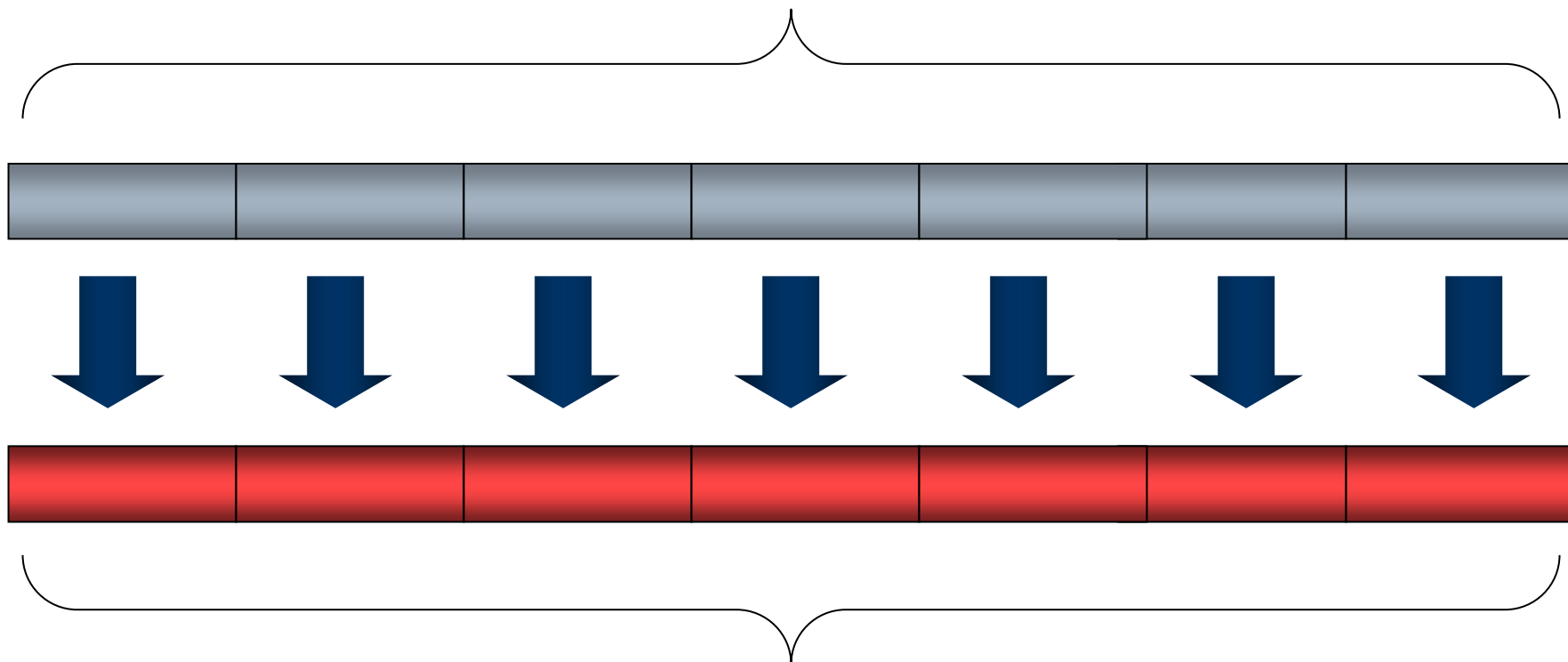
■ رمزهای جریانی (Stream Cipher)

■ پردازش پیغام‌ها بصورت پیوسته



رمزهای قالبی

متن آشکار (تقسیم شده به قطعات)



قطعات خروجی

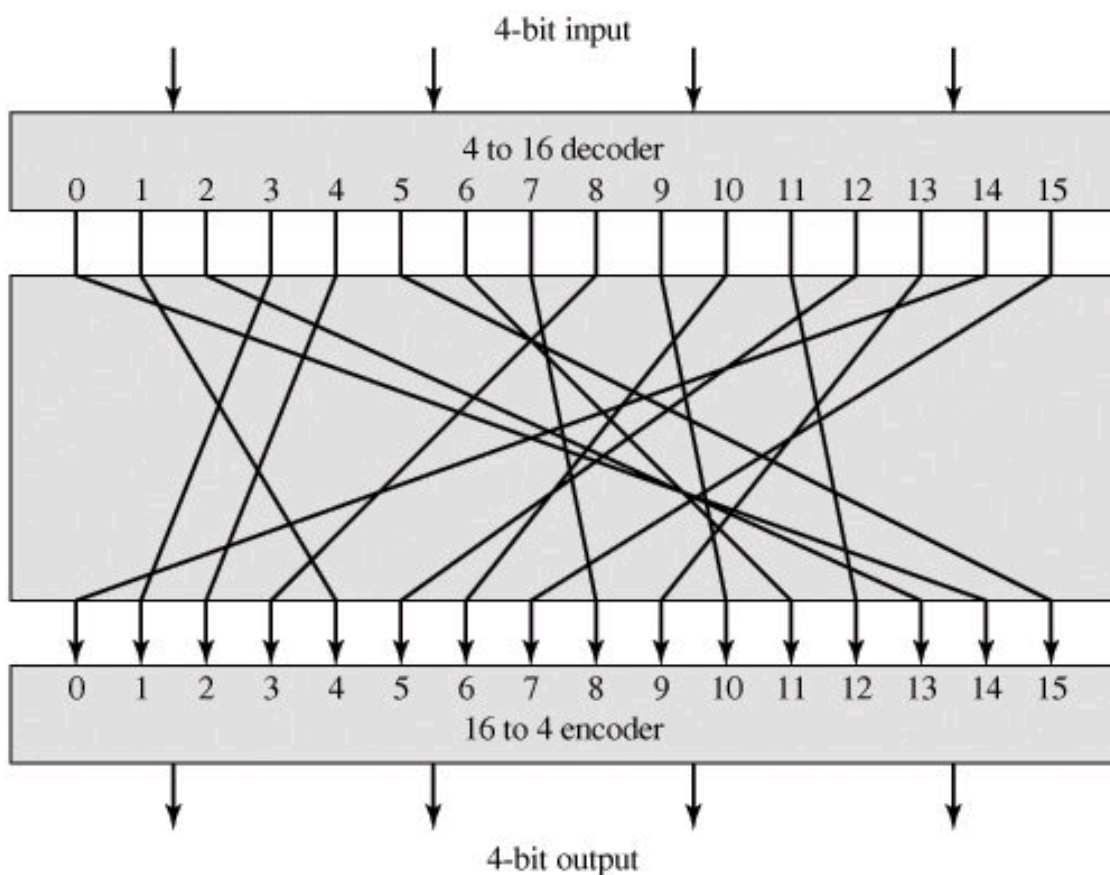


رمزگذاری کلاسیک-رمزگذاری مدرن

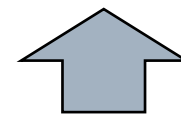
- در روشهای رمزگذاری مدرن، علاوه بر اعمال جانشینی و جایگشت از توابع ساده مانند XOR استفاده می شود.
- مجموعه اعمال فوق طی مراحل متوالی روی متن اولیه اعمال می شوند.
- تکنیک بکارگرفته شده در Rotor Machine ها الهام بخش روشهای رمزگذاری مدرن بوده است.



رمز قالبی ایده‌آل



- ☐ یک جانشینی عمده
- ☐ طول کلید $n \cdot 2^n$ برای قطعات n بیتی
- ☐ نیاز به کاهش طول کلید و ایجاد تقریبی از رمز قالبی ایده‌آل



ایده رمز فستل



اصول رمزهای قالبی

- اغلب مبتنی بر ساختار رمز فیستل هستند.
- نگاشت قطعات متن آشکار به قطعات متن رمز شده باید (برای ممکن بودن رمز گشایی) برگشت پذیر باشد.
- ایده رمز محصولی (Product Cipher): الگوریتم رمز، قطعات ورودی را در چند مرحله ساده و متوالی پردازش می کند. به این مراحل دور می گوئیم.
- هر دور عموماً مبتنی بر ترکیب اعمال ساده ای همچون جایگزینی و جایگشت استوار است.



شانون و رمز جانشینی و جایگشت

- شانون ایده استفاده از شبکه اعمال جانشینی و جایگشت را در سال ۱۹۴۹ مطرح کرد.
- پایه رمزهای مدرن بر اساس این دو عمل است:
 - جانشینی (S-box)
 - جایگشت (P-box)
- این دو عمل، گمراه‌کنندگی (Confusion) و پراکندگی (Diffusion) پیام موردنظر و کلید را موجب می‌شوند.



گمراه‌کنندگی و پراکندگی

□ الگوریتم‌های رمز باید خصوصیات آماری پیام اصلی (متن آشکار) را به طور کامل مخفی کنند.

□ **گمراه‌کنندگی (Confusion):** رابطه بین متن رمز شده و کلید تا حد امکان پیچیده باشد.

□ **پراکندگی (Diffusion):** ساختار آماری متن آشکار بر روی حجم وسیعی از متن‌های رمز شده ممکن پراکنده شود.



استانداردهای رمزهای قالبی آمریکا

□ رمزهای قالبی استاندارد

■ استاندارد رمزگذاری داده DES

■ استاندارد رمزگذاری پیشرفته AES

□ تحت نظارت

National Institute of Science and Technology (NIST)



فهرست مطالب

- رمزهای متقارن و قالبی
- ساختار رمزهای فیستل
- استاندارد رمزگذاری داده DES
- الگوریتم رمز 2DES و 3DES
- استاندارد رمزگذاری پیشرفته AES
- رمزهای متقارن معروف
- مدهای کاری رمزهای متقارن

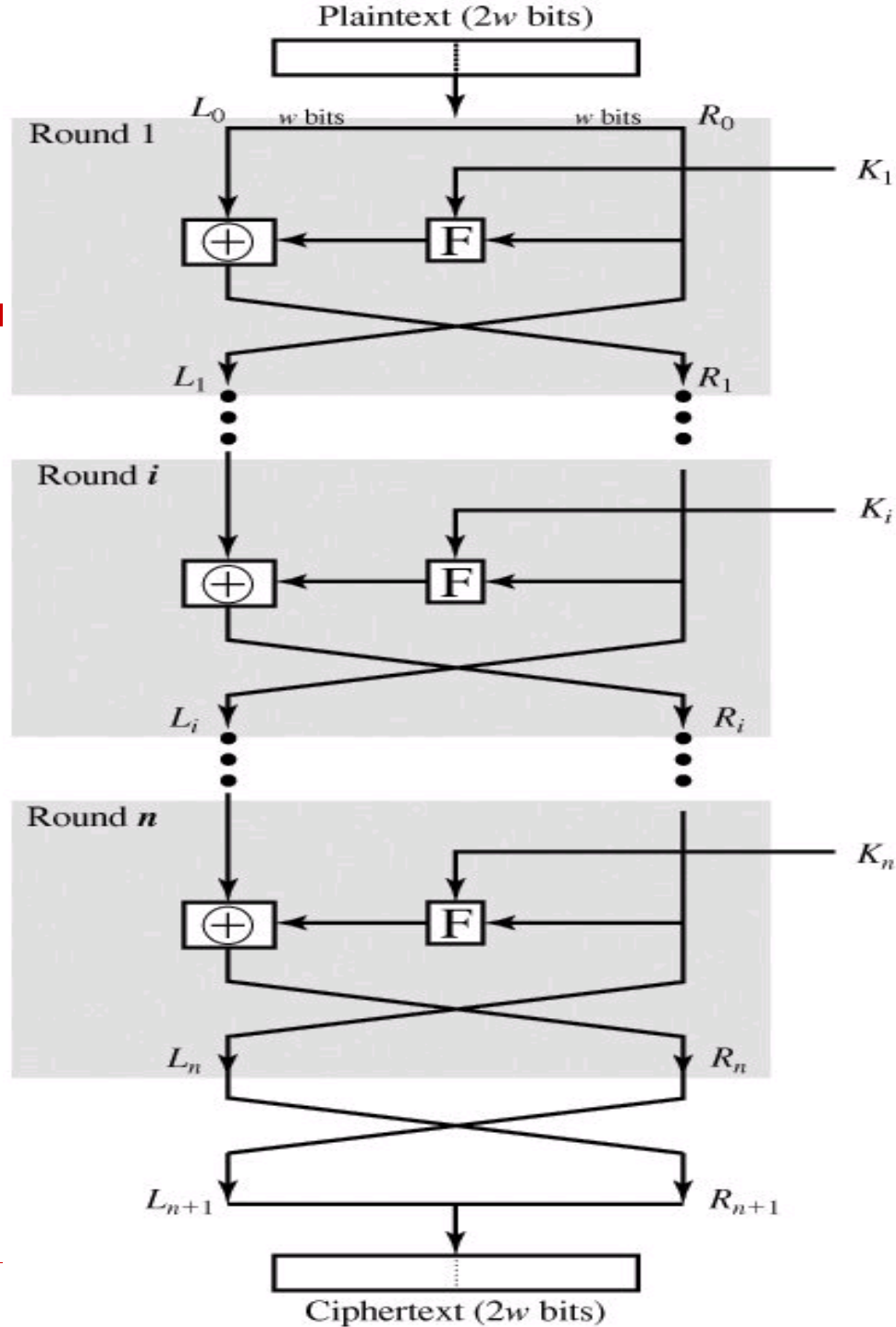


ساختار رمزهای فستل

- معمولاً الگوریتم‌های رمزنگاری از ساختاری تبعیت می‌کنند که توسط فستل در سال ۱۹۷۳ در IBM پیشنهاد شد.
- مبتنی بر رمز محصولی برگشت‌پذیر
- مبتنی بر مفهوم شبکه جانشینی و جایگشت
- هر قطعه ورودی را به دو نیمه تقسیم می‌کند:
 - پردازش در طی چند مرحله (دور)
 - انجام جانشینی بر روی نیمه چپ
 - جانشینی بر اساس تابع دور حاصل از زیرکلید هر دور و نیمه راست
 - جایگشت با معاوضه دو نیمه



ساختار رمز فایستل





ساختار رمزهای فایستل

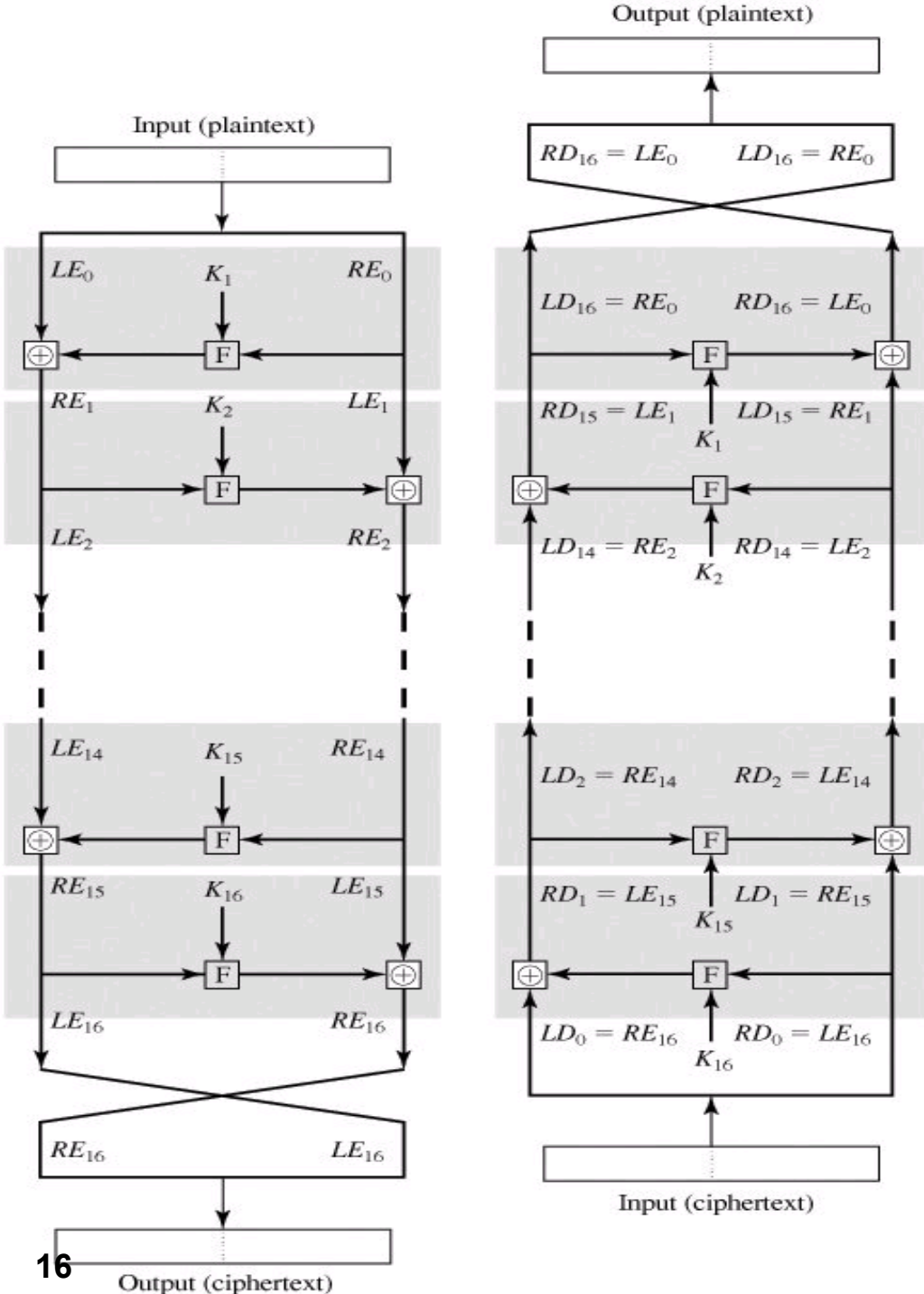
رمزهای فایستل به انتخاب پارامترهای زیر بستگی دارند.

- طول قطعه (بلوک): ۶۴ بیت تا ۱۲۸ بیت
- طول کلید: ۶۴ بیت یا کمتر در حال حاضر کافی نیست.
- تعداد دورها: معمولاً ۱۶ دور
- الگوریتم تولید زیرکلیدها
- هر چه پیچیده‌تر باشد، تحلیل هم سخت‌تر می‌شود.
- تابع دور (Round function): هر چه پیچیده‌تر تحلیل سخت‌تر
- سرعت رمزگذاری/رمزگشایی
- سادگی بررسی و درک درستی عملکرد



رمز گذاری و رمز گشایی در ساختار رمز فیستل

□ نیازی به برگشت پذیر بودن
تابع F نیست.





فهرست مطالب

- رمزهای متقارن و قطعه‌ای
- ساختار رمزهای فیستل
- **استاندارد رمزگذاری داده DES**
- الگوریتم رمز 2DES و 3DES
- استاندارد رمزگذاری پیشرفته AES
- رمزهای متقارن معروف
- مدهای کاری رمزهای متقارن



استاندارد رمزگذاری داده DES

□ مرور

- در سال ۱۹۷۴ توسط IBM تولید شد.
- پس از انجام تغییراتی توسط NSA، در سال ۱۹۷۶ NIST آن را پذیرفت.
- اساس الگوریتم ترکیبی از عملیات جایگزینی و جایگشت است.

□ مشخصات

- طول کلید ۵۶ بیت
- طول قطعه‌های ورودی و خروجی : ۶۴ بیت
- تعداد دورها: ۱۶ دور
- الگوریتم‌های رمزگذاری و رمزگشایی عمومی هستند، ولی مبانی ریاضی و اصول طراحی آنها فاش نشد.



DES امن نیست!

□ کلید ۵۶ بیتی دارای کل فضای حالت $2^{56} = 7.2 * 10^{16}$

■ حمله آزمون جامع هرچند مشکل ولی امکان پذیر است.

□ در ژانویه ۱۹۹۹ این الگوریتم توسط آزمون جامع فضای کلید در ۲۳ ساعت شکسته شد!

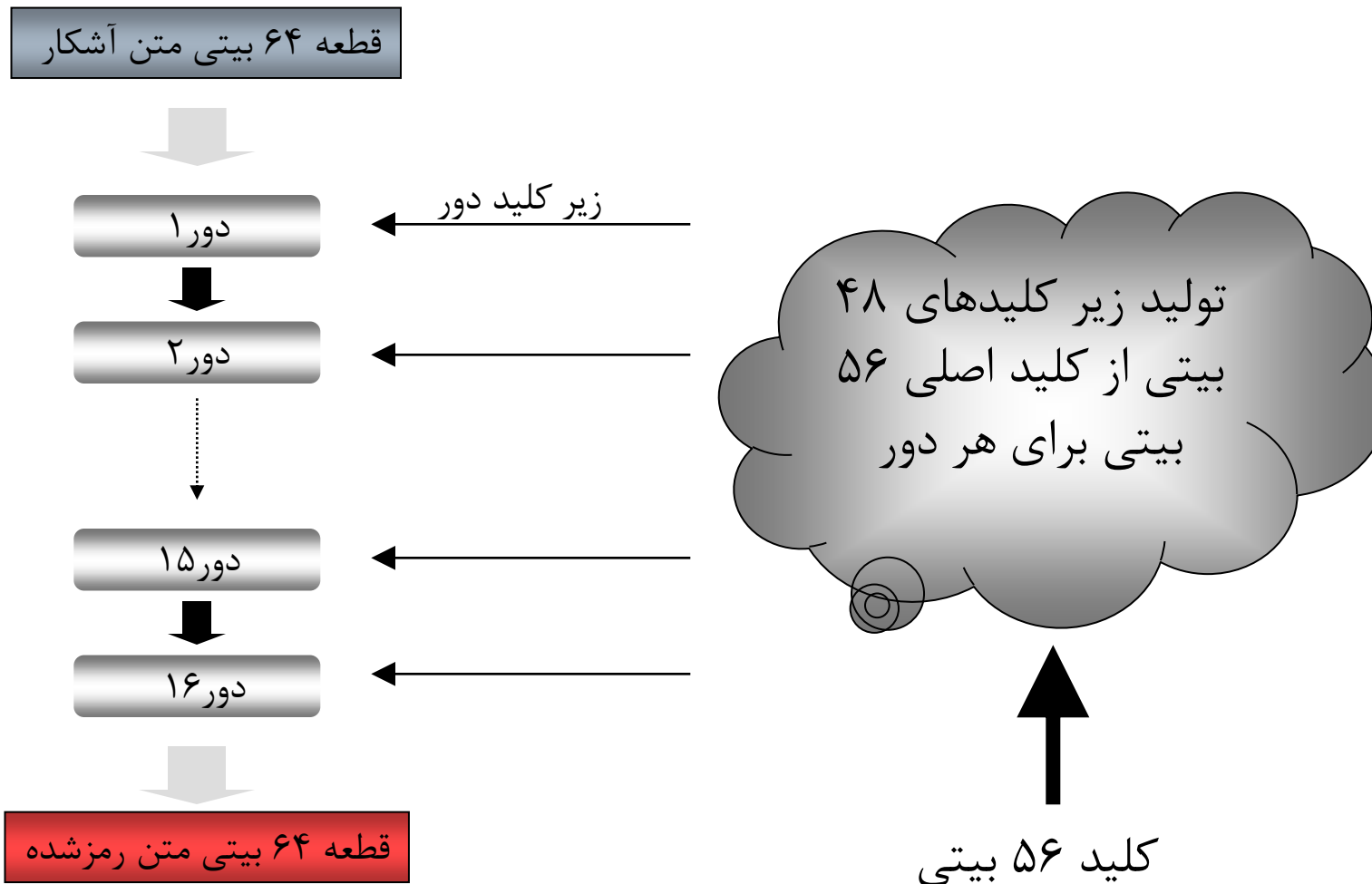
■ بیش از ۱۰۰۰ کامپیوتر بر روی اینترنت هر یک بخش کوچکی از کار جستجو را انجام دادند.

□ به الگوریتم‌های امن تر با طول کلید بیشتر نیاز داریم.

□ علاوه بر این، DES طراحی شفاف و روشن ندارد.

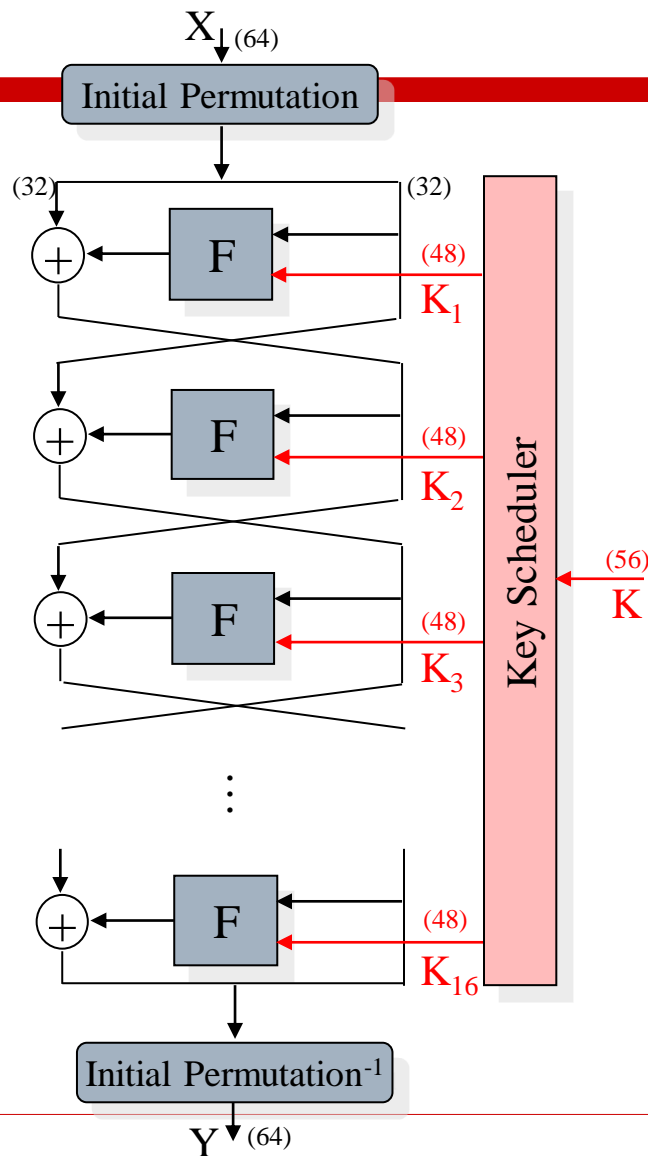


استاندارد رمزگذاری داده DES





ساخترافیستل رمز DES





حمله تحلیلی به DES

- عموماً حملات آماری هستند.
- از ساختار داخلی DES استفاده می‌کنند.
- تشخیص همه یا بعضی از بیت‌های کلید میانی
- جستجوی کامل روی بقیه بیت‌ها
- شامل
 - تحلیل تفاضلی
 - تحلیل خطی
- این روش‌ها هنوز به طور عملی امکان‌پذیر نیستند.
- جستجوی کامل ساده‌تر به نظر می‌رسد!



تحلیل تفاضلی و خطی DES

□ تحلیل تفاضلی

- ارائه شده توسط Murphy و دیگران در سال ۱۹۹۰
- مبتنی بر اینکه تغییرات ورودی چگونه به تغییرات در خروجی منتقل می‌شوند.
- نیاز به 2^{47} زوج plaintext/ciphertext انتخابی دارد.

□ تحلیل خطی

- ارائه شده توسط Matsui در سال ۱۹۹۱
- مبتنی بر یافتن یک تقریب خطی از تبدیلات انجام شده توسط DES
- نیاز به 2^{47} زوج plaintext/ciphertext انتخابی دارد.



فهرست مطالب

- رمزهای متقارن و قطعه‌ای
- ساختار رمزهای فیستل
- استاندارد رمزگذاری داده DES
- **الگوریتم رمز 2DES و 3DES**
- استاندارد رمزگذاری پیشرفته AES
- رمزهای متقارن معروف
- مدهای کاری رمزهای متقارن



الگوریتم 2DES و 3DES

مسئله:

■ آسیب پذیری DES در مقابل حمله آزمون جامع

راه حل:

- پیچیده کردن الگوریتم DES از طریق اضافه کردن مراحل رمزنگاری و افزایش طول کلید
- یا استفاده از الگوریتم‌های رمزنگاری مناسب دیگر



الگوریتم 2DES

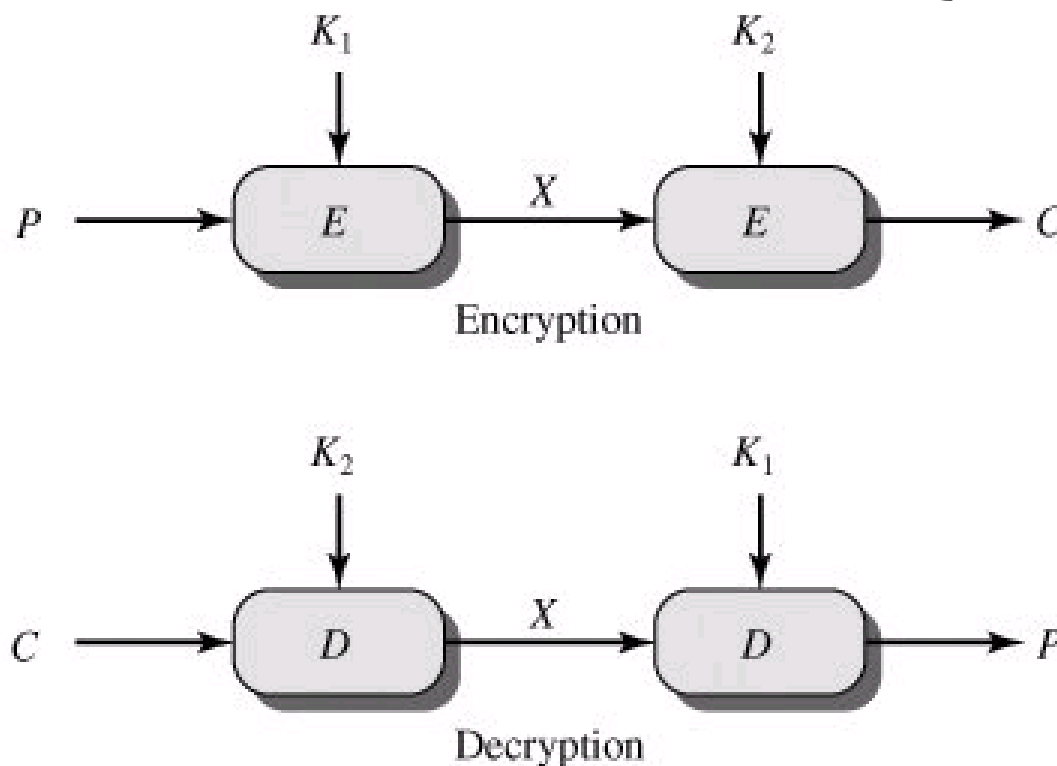
□ افزایش قدرت DES با رمزگذاری چندمرحله‌ای با DES و استفاده از کلیدهای متعدد

2DES

$$C = E(K_2, E(K_1, P))$$

$$P = D(K_1, D(K_2, C))$$

طول کلید = ۱۱۲ بیت





تحلیل الگوریتم رمز 2DES

□ حمله ملاقات در میانه (Meet-in-the-Middle)

□ $C = E(K_2, E(K_1, P))$

□ $X = D(K_2, C) = E(K_1, P)$

□ با داشتن یک زوج (P, C) ,

□ P را با 2^{56} کلید ممکن برای K_1 رمزگذاری کن و مقادیر X را ذخیره کن.

□ C را با 2^{56} کلید ممکن برای K_2 رمزگشایی کن و مقادیر حاصله با مقادیر ذخیره شده مقایسه کن.

□ در صورت تطابق، درستی زوج کلید یافت شده را چک کن.

□ پیچیدگی انجام عملیات فوق $O(2^{56})$ است.



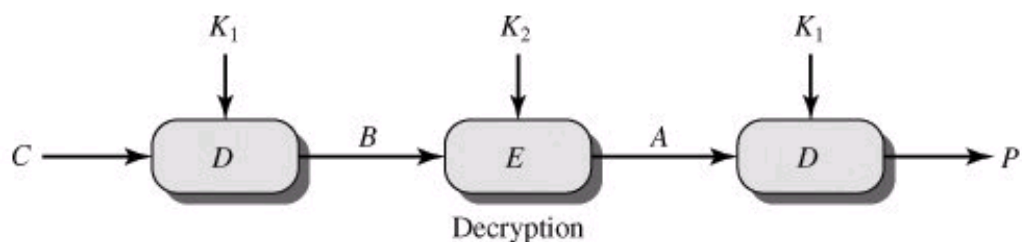
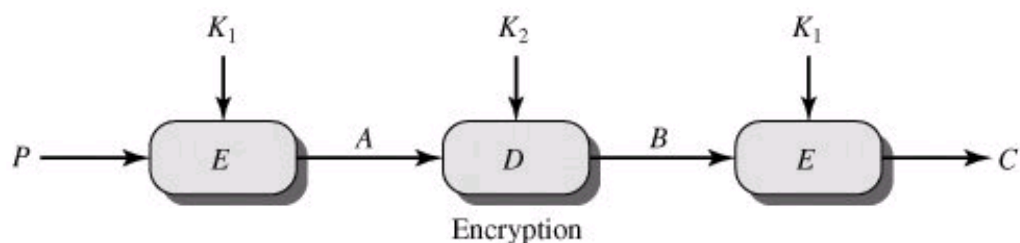
الگوریتم 3DES با دو کلید

❑ حل مشکل 2DES با سه مرحله رمزگذاری با DES

❑ امکان بهره‌گیری از DES به صورت زیر:

$$C = E(K_1, D(K_2, E(K_1, P)))$$

$$P = D(K_1, E(K_2, D(K_1, C)))$$





الگوریتم 3DES با سه کلید

□ استفاده از سه کلید مختلف

$$C = E(K_3, D(K_2, E(K_1, P)))$$

□ طول کلید = ۱۶۸ بیت

□ استفاده در برخی برنامه‌های تحت اینترنت

PGP ■

S/MIME ■



فهرست مطالب

- رمزهای متقارن و قطعه‌ای
- ساختار رمزهای فیستل
- استاندارد رمزگذاری داده DES
- الگوریتم رمز 2DES و 3DES
- **استاندارد رمزگذاری پیشرفته AES**
- رمزهای متقارن معروف
- مدهای کاری رمزهای متقارن



استاندارد رمزگذاری پیشرفته AES

□ NIST در سال ۱۹۹۷ مسابقه‌ای دو مرحله‌ای برای طراحی استاندارد جدید برگزار کرد.

■ تمام طراحی‌ها باید بر اساس اصول کاملاً روشن انجام شوند.

■ سازمانهای دولتی آمریکا حق هیچ گونه دخالتی در طراحی الگوریتم ندارند.

ارایه شده توسط:
Vincent Rijmen•
Joan Daemen•

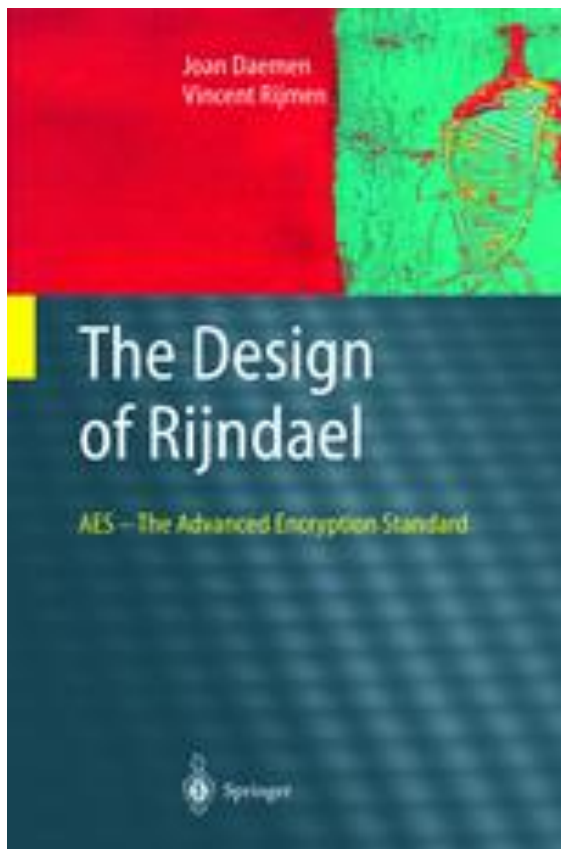
□ در سال ۲۰۰۰ رایندهال (Rijndael) به عنوان برنده اعلام شد.

■ استاندارد جدید تحت عنوان استاندارد رمزگذاری پیشرفته AES مورد قبول واقع شد.



اصول طراحی Rijndael

- J. Daemen and V. Rijmen. **The Design of Rijndael**. Springer-Verlag Berlin, 2002.



Joan Daemen
(1965 –)



Vincent Rijmen
(1970 –)



مشخصات استاندارد AES

256	192	128	طول کلید
128	128	128	طول قطعه ورودی و خروجی
14	12	10	تعداد دور
128	128	128	طول کلید هر دور

در الگوریتم اصلی Rijndael طول قطعه می تواند ۱۲۸، ۱۹۲ و یا ۲۵۶ بیت باشد، ولی در استاندارد **FIPS PUB 197** طول آن به ۱۲۸ بیت محدود شده است.



مشخصات استاندارد AES

- مبتنی بر ساختار رمز **فistel** نیست و کل قطعه داده پردازش می شود.
- کلید ۱۲۸ بیتی (۴ کلمه ای)، به یک آرایه W با ۴۴ عنصر از کلمات ۳۲ بیتی بسط داده می شود.
- کلید هر دور ۴ عنصر این آرایه (۱۲۸ بیت) است.



نحوه کار AES-128

- الگوریتم زمان بندی کلید نقش تهیه کلید برای هر دور بر اساس کلید اصلی را بر عهده دارد.
- برخلاف DES و بسیاری از رمزهای دیگر، اعمال لازم بر روی بایتهای انجام می شود نه بیتها.
- متن آشکار ۱۲۸ بیتی به شکل یک ماتریس حالت 4×4 در می آید.
 - هر درایه یک بایت از متن آشکار را نشان می دهد.
 - این ماتریس به صورت ستونی پر می شود.
 - این ماتریس در انتها مولد متن رمز است.



نحوه کار AES-128

□ متن آشکار ورودی به صورت **ستونی** در ماتریس حالت ذخیره می شود.

Input = 32 43 f6 a8 88 5a 30 8d
31 31 98 a2 e0 37 07 34

32	88	31	e0
43	5a	31	37
f6	30	98	07
a8	8d	a2	34

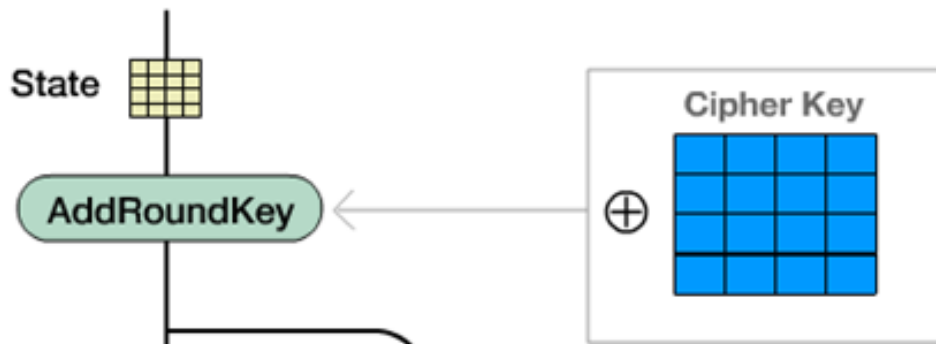


مراحل رمزگذاری AES-128

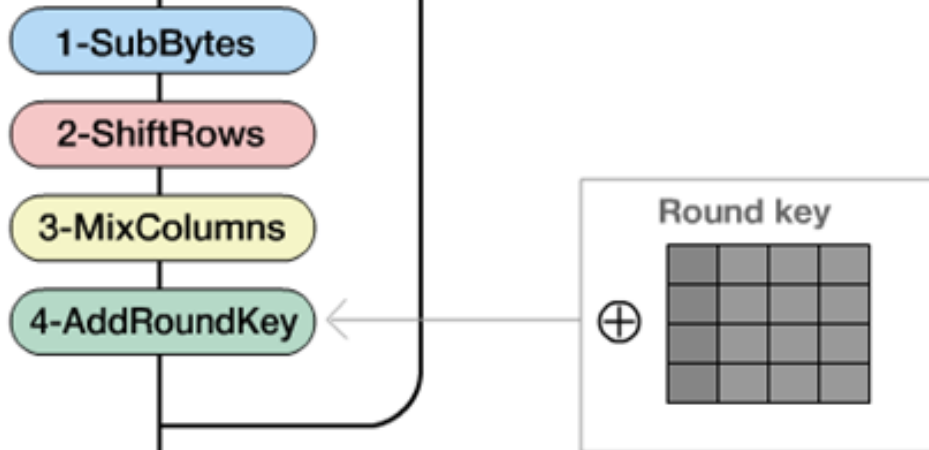
- در هر دور ۴ عمل بر روی ماتریس حالت اعمال می شود.
- **جایگزینی بایتهای:** جایگزینی درایه های ماتریس حالت با استفاده از یک s-box
- **شیفت سطری**
- **ترکیب ستونها:** ترکیب خطی ستونها با استفاده از ضرب ماتریسی
- **اضافه نمودن کلید دور:** جمع مبنای دو ماتریس حالت با کلید دور
- هر چهار عمل برگشت پذیر بوده، لذا هر دور برگشت پذیر است.



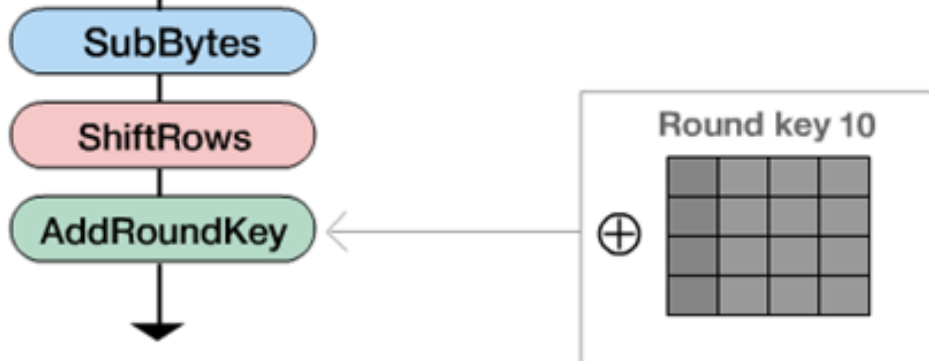
initial
round



ROUND
1..9



9
rounds

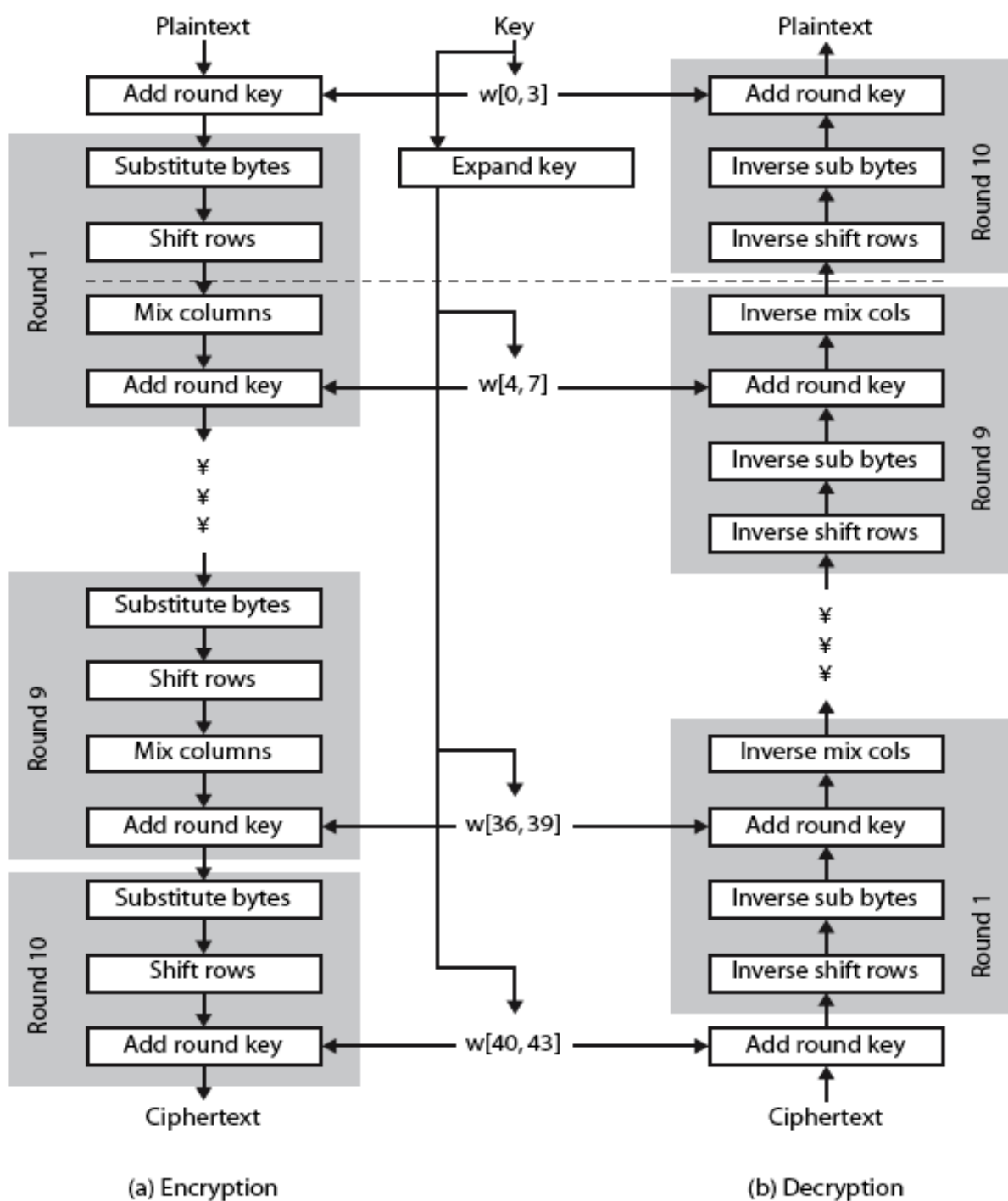


final
round

رمزگذاری در AES



رمزگذاری و رمزگشایی در AES





جایگزینی بایتها (S-box) در AES

- نوعی تابع غیرخطی محسوب می شود
- توسط یک جدول 16×16 پیاده سازی می شود.
- این جدول بر اساس تبدیل مقادیر در میدان متناهی 2^8 ساخته می شود و در مقابل حملات شناخته شده مقاوم است.



جایگزینی بایتها (S-box) در AES

□ ورودی تابع سطر و ستون درایه جدول را معین کرده و مقدار ذخیره شده در این درایه خروجی تابع است.

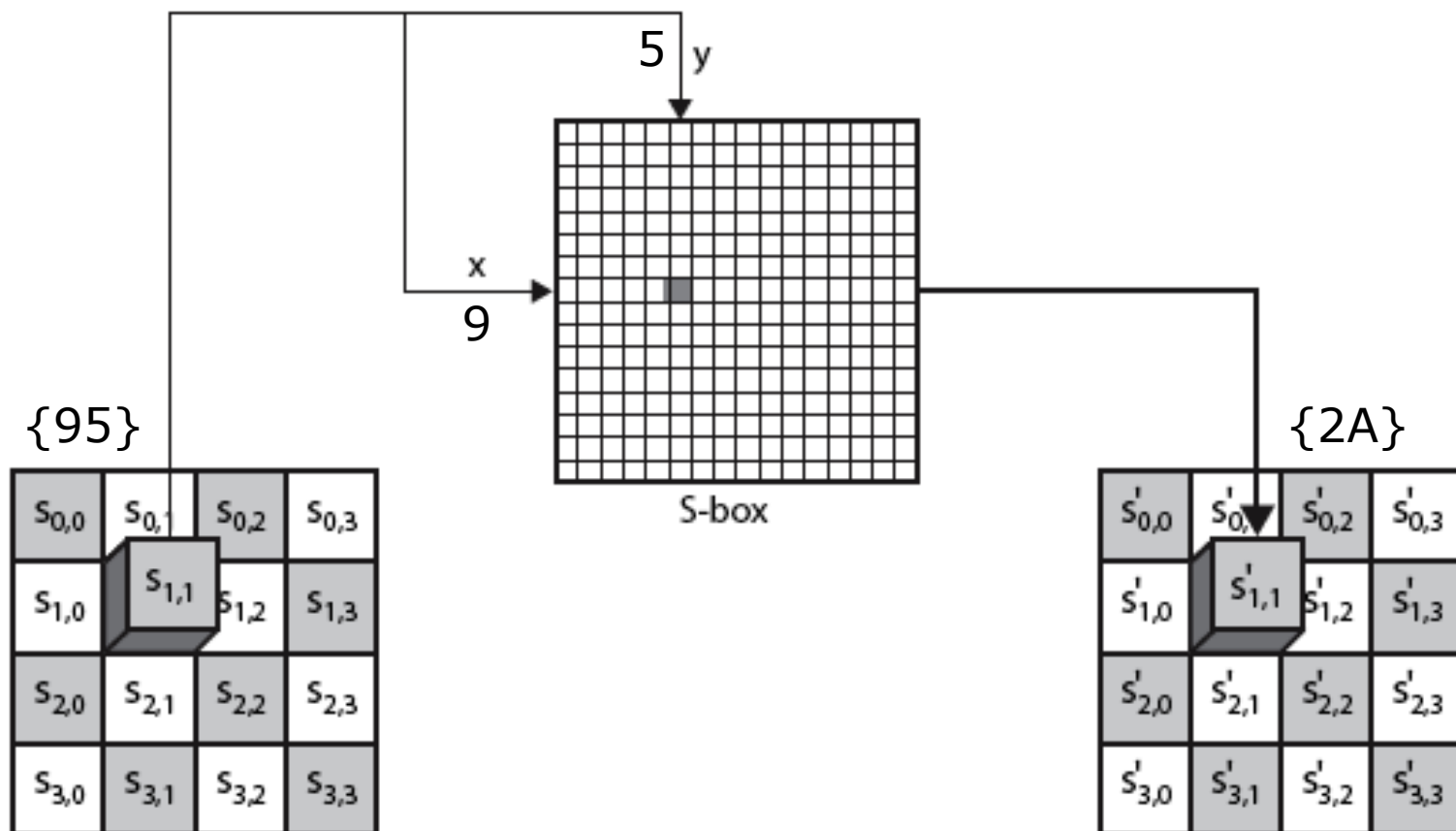
□ با داشتن یک عنصر از ماتریس حالت

■ سطر جدول = ۴ بیت سمت چپ عنصر

■ ستون جدول = ۴ بیت سمت راست عنصر

□ برای رمزگشایی از جدول معکوس استفاده می شود.

جایگزینی بایتها (S-box) در AES





جداول جایگزینی در AES

(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	4									
	2	B7	FD	93	26	36	3F	F									
	3	04	C7	23	C3	18	96	0									
	4	09	83	2C	1A	1B	6E	5									
	5	53	D1	00	ED	20	FC	B									
	6	D0	EF	AA	FB	43	4D	3									
	7	51	A3	40	8F	92	9D	3									
	8	CD	0C	13	EC	5F	97	4									
	9	60	81	4F	DC	22	2A	9									
	A	E0	32	3A	0A	49	06	2									
	B	E7	C8	37	6D	8D	D5	4									
	C	BA	78	25	2E	1C	A6	B									
	D	70	3E	B5	66	48	03	F									
	E	E1	F8	98	11	69	D9	8									
	F	8C	A1	89	0D	BF	E6	4									

(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



شیفت سطری در AES

□ شیفت چرخشی به چپ که در آن

■ سطر اول بدون تغییر

■ سطر دوم یک بایت شیفت چرخشی به چپ

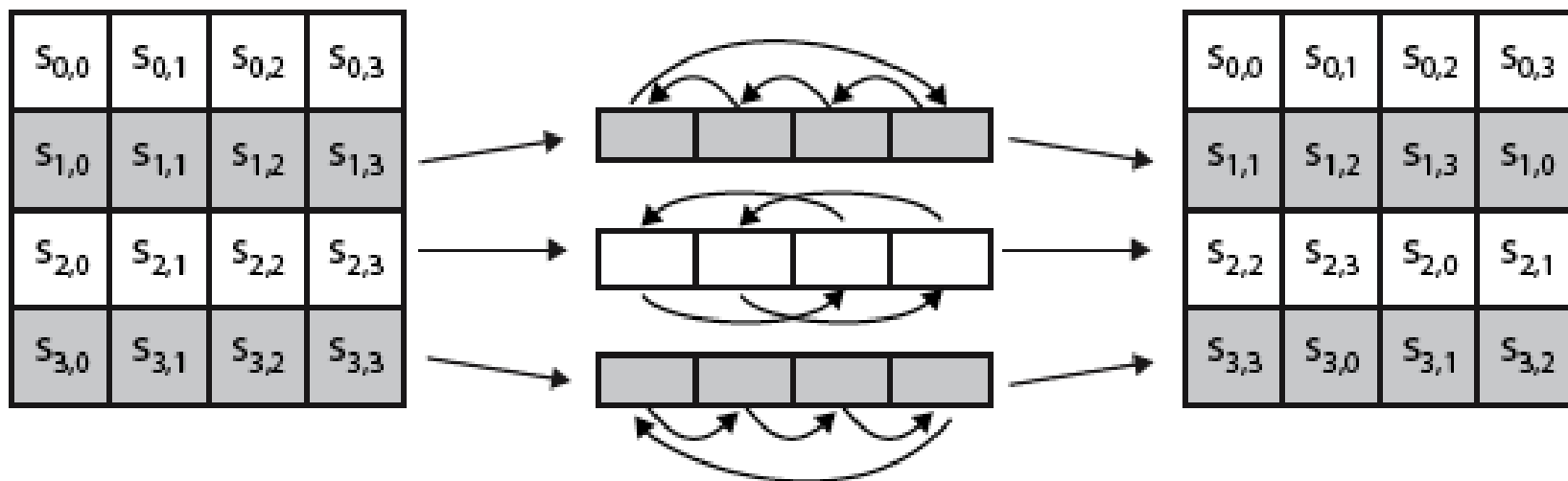
■ سطر سوم دو بایت شیفت چرخشی به چپ

■ سطر چهارم سه بایت شیفت چرخشی به چپ

□ در رمزگشایی، شیفت به راست انجام می شود.

□ از آنجا که داده به صورت ستونی در ماتریس حالت ذخیره شده،
لذا این مرحله یک جایگشت بین ستونها انجام می دهد.

شيفت سطری در AES

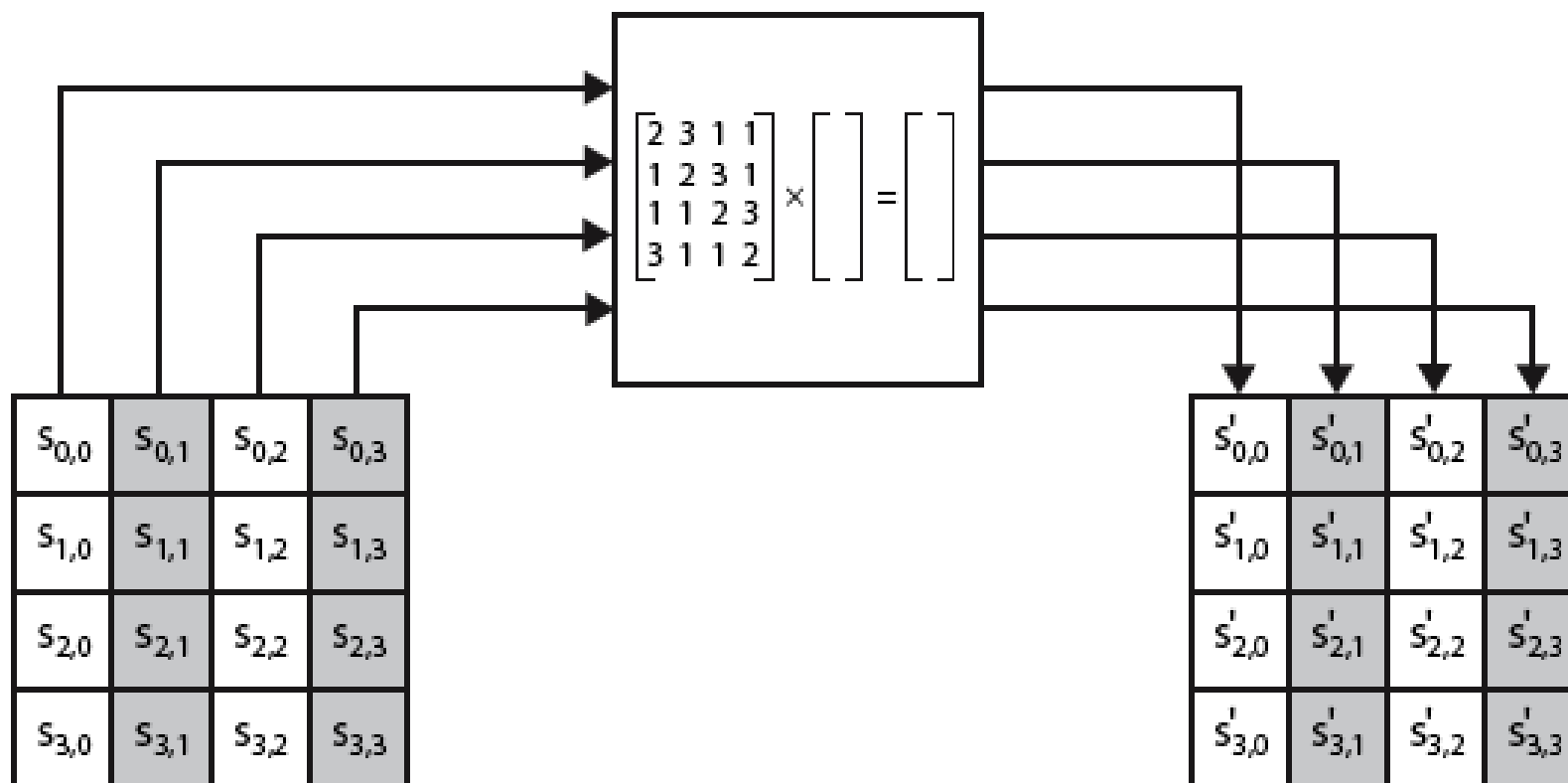




ترکیب ستونها در AES

- هر ستون جداگانه پردازش می شود.
- هر بایت با مقداری (وابسته به هر چهار عنصر آن ستون) جایگزین می شود.
- با ضرب ماتریسی این کار انجام می شود.

ترکیب ستونها در AES





ترکیب ستونها در AES

جمع همان XOR است ولی ضرب باید در میدان متناهی 2^8 انجام شود که آن هم با تعدادی XOR و شیفتهی قابل انجام است (برای اطلاع از نحوه چگونگی مراجعه شود به فصل ۴ کتاب Stallings).

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

$$s'_{0,j} = (2 \cdot s_{0,j}) \oplus (3 \cdot s_{1,j}) \oplus s_{2,j} \oplus s_{3,j}$$

$$s'_{1,j} = s_{0,j} \oplus (2 \cdot s_{1,j}) \oplus (3 \cdot s_{2,j}) \oplus s_{3,j}$$

$$s'_{2,j} = s_{0,j} \oplus s_{1,j} \oplus (2 \cdot s_{2,j}) \oplus (3 \cdot s_{3,j})$$

$$s'_{3,j} = (3 \cdot s_{0,j}) \oplus s_{1,j} \oplus s_{2,j} \oplus (2 \cdot s_{3,j})$$



ترکیب ستونها در AES

برای رمزگشایی از ماتریس دیگری در ضرب استفاده می‌شود.

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$



افزودن کلید دور در AES

□ ماتریس حالت با کلید دور XOR می شود.

□ به صورت ستونی انجام می شود.

□ برای رمزگشایی نیز همین عمل انجام می شود.

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

 \oplus

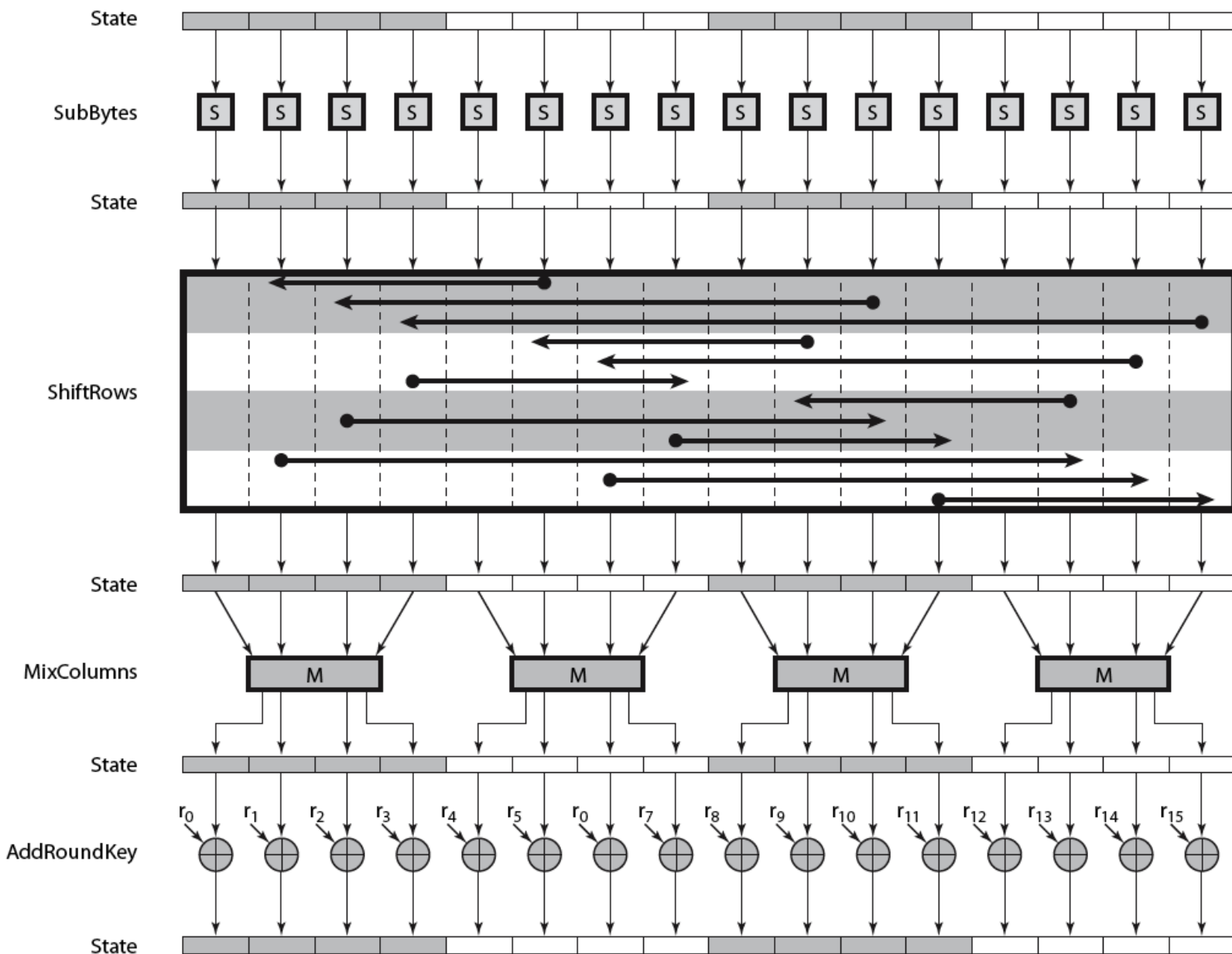
w_i	w_{i+1}	w_{i+2}	w_{i+3}
-------	-----------	-----------	-----------

 $=$

$s'_{0,0}$	$s'_{0,1}$	$s'_{0,2}$	$s'_{0,3}$
$s'_{1,0}$	$s'_{1,1}$	$s'_{1,2}$	$s'_{1,3}$
$s'_{2,0}$	$s'_{2,1}$	$s'_{2,2}$	$s'_{2,3}$
$s'_{3,0}$	$s'_{3,1}$	$s'_{3,2}$	$s'_{3,3}$



یک دور الگوریتم AES





بسط کلید در AES

- یک کلید ۱۲۸ بیتی (۱۶ بایتی) دریافت می‌کند و آن را به یک آرایه ۴۴ عنصره (از کلمات ۳۲ بیتی) بسط می‌دهد.
- شروع: کپی کلید در ۴ عنصر (کلمه) اول آرایه
- تکرار: تولید هر عنصر (کلمه $w[i]$) بر اساس $w[i-1]$ و $w[i-4]$
- عناصر موجود در درایه های مضرب ۴ با تابع پیچیده g محاسبه می‌شوند.



بسط کلید در AES

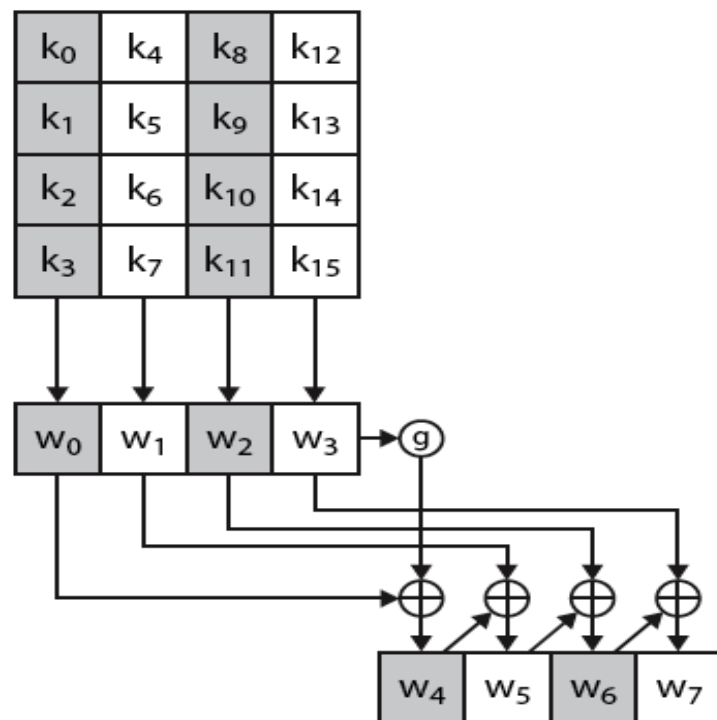
□ If $i=4k$:

g

$$w[i] = \text{SubWord}(\text{RotWord}(w[i-1])) \oplus \text{Rcon}[i/4] \oplus w[i-4]$$

□ Otherwise:

$$w[i] = w[i-1] \oplus w[i-4]$$





بسط کلید در AES

□ تابع پیچیده g شامل زیرتوابع زیر است:

1. (RotWord) شیفت چرخشی به چپ به اندازه یک بایت
2. (SubWord) جایگزینی هر بایت بر اساس جدول S-box مورد استفاده در رمزگذاری
3. ترکیب XOR مقدار حاصل از انجام اعمال ۱ و ۲ با مقدار ثابت $Rcon[i/4]$

$$Rcon[i/4] = (RC[i/4], 0, 0, 0)$$

j	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36



امنیت AES

□ تا کنون حمله ای بر روی آن کشف نشده و در مقابل همه حملات معمول امن طراحی شده است.

□ از لحاظ مقایسه با DES:

■ فرض کنید ماشینی وجود دارد که کلید DES را از طریق آزمون جامع در یک ثانیه بازیابی می کند، یعنی در هر ثانیه 2^{56} کلید را امتحان می کند. این ماشین کلید AES را در $10^{12} \times 149$ سال بازیابی می نماید.



جنبه‌های پیاده‌سازی AES

- قابلیت پیاده‌سازی روی پردازنده‌های ۸ بیتی
- قابلیت پیاده‌سازی کارا روی پردازنده‌های ۳۲ بیتی
- همه اعمال با شیفِت، XOR و استفاده از یک سری جداول look-up قابل انجام است.
- به اعتقاد طراحان آن، قابلیت پیاده‌سازی بسیار کارای آن باعث انتخاب آن شده است.



مجموعه دستورالعمل AES

□ اینتل در سال ۲۰۰۸ مجموعه دستورالعملهای AES را به CPUهای خود افزود ← افزایش چشمگیر سرعت AES

□ مجموعه دستورالعملهای مشابهی برای سایر معماریهای CPU نظیر ARM وجود دارد.

دستورالعمل	توصیف
AESENC	اجرای یک دور عادی از رمزگذاری AES
AESENCLAST	اجرای دور آخر از رمزگذاری AES
AESDEC	اجرای یک دور عادی از رمزگشایی AES
AESDECLAST	اجرای دور آخر از رمزگشایی AES
AESKEYGENASSIST	کمک در تولید کلید دور AES
AESIMC	کمک در عملیات Inverse Mix Columns
PCLMULQDQ	ضرب بدون رقم نقلی (عملیات در میدانهای متناهی)



فهرست مطالب

- رمزهای متقارن و قطعه‌ای
- ساختار رمزهای فیستل
- استاندارد رمزگذاری داده DES
- الگوریتم رمز 2DES و 3DES
- استاندارد رمزگذاری پیشرفته AES
- **رمزهای متقارن معروف**
- مدهای کاری رمزهای متقارن



IDEA

- ابداع شده توسط Lai و Messay در سال ۱۹۹۰
- ویژگیها
 - طول کلید : ۱۲۸ بیت
 - طول بلاک : ۶۴ بیت
 - تعداد دورها : ۸ دور
 - انجام عملیات روی عملوندهای ۱۶ بیتی

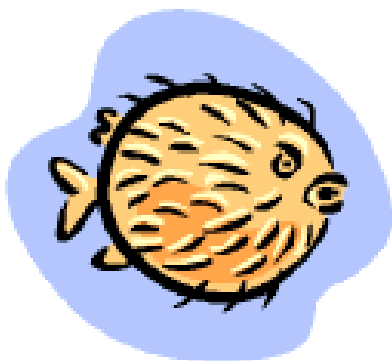


تحليل IDEA

- تا کنون هیچ حمله عملی علیه IDEA شناخته نشده است.
- به نظر می رسد تا مدت‌ها نسبت به حملات امن باشد.
- طول کلید ۱۲۸ بیتی حمله آزمون جامع را غیرممکن می کند
(حداقل با تکنولوژی‌های موجود).

Blowfish

- طراحی شده توسط Schneier در سال ۱۹۹۳/۹۴
- وجود پیاده‌سازی‌های پرسرعت روی پردازنده‌های ۳۲بیتی
- فشردگی: نیاز به کمتر از 5k حافظه
- پیاده‌سازی آسان
- تحلیل الگوریتم آسان
- طول کلید متغیر: **درجه امنیت قابل تغییر است.**





ویژگیهای Blowfish

- ☐ طول بلاک: ۶۴ بیت
- ☐ تعداد دورها: ۱۶ دور
- ☐ طول کلید متغیر: ۳۲ تا ۴۴۸ بیت
- ☐ تولید زیرکلید و S-Box های وابسته به کلید



RC5

- ☐ انطباق با نرم افزارها و سخت افزارهای مختلف
- ☐ سرعت اجرای زیاد : عملیات روی کلمه‌ها انجام می‌شوند.
- ☐ انطباق با پردازنده‌های با تعداد بیت‌های متفاوت
- ☐ طول بلاک متغیر
- ☐ طول کلید متغیر
- ☐ تعداد دورها متغیر
- ☐ نیاز به حافظه کم
- ☐ طراحی و تحلیل الگوریتم ساده
- ☐ تعداد دورهای وابسته به داده: تحلیل رمز را مشکل می‌کند.



CAST-128

- ابداع شده توسط Tavares و Adams در سال ۱۹۹۷
- طول کلید متغیر: از ۴۰ تا ۱۲۸ بیت (افزایش ۸ بیتی)
- تعداد دور: ۱۶ دور
- مشابه ساختار کلاسیک فیستل است با دو تفاوت زیر:
 - در هر دور از دو زیرکلید استفاده می‌کند.
 - تابع F به دور بستگی دارد.
- در حال استفاده در PGP (امن سازی سرویس ایمیل)



مقایسه سرعت الگوریتمها

Algorithm	Clock cycles per round	# of rounds	#of clock cycles per byte encrypted
Blowfish	9	16	18
RC5	12	16	23
DES	18	16	45
IDEA	50	8	50
3DES	18	48	108



فهرست مطالب

- رمزهای متقارن و قطعه‌ای
- ساختار رمزهای فیستل
- استاندارد رمزگذاری داده DES
- الگوریتم رمز 2DES و 3DES
- استاندارد رمزگذاری پیشرفته AES
- رمزهای متقارن معروف
- مدهای کاری رمزهای متقارن



استفاده از رمزهای قطعه‌ای

❑ رمزهای قطعه‌ای به طور مستقل امنیت زیادی را به ارمغان نمی‌آورند. بلکه باید در مدهای کاری مناسب مورد استفاده قرار گیرند.

❑ مدهای کاری که متن‌های مشابه را به متن‌های رمز شده یکسان تبدیل می‌کنند، امن نیستند. صرف نظر از رمز قطعه‌ای مورد استفاده!



وضعیت ایده آل

□ ساختار الگوریتم رمزنگاری متقارن (مد کاری) به گونه‌ای باشد که قابلیت‌های عناصر سازنده خود (رمزهای قطعه‌ای) را به ارث ببرد.

■ یعنی با اطمینان از رمزهای قطعه‌ای، بتوانیم از الگوریتم رمزنگاری نیز مطمئن شویم.



مدهای کاری رمزهای قطعه ای

□ امروزه مدهای کاری با توجه به امنیت قابل اثبات طراحی می شوند.

□ مدهای کاری می توانند از رمزهای قطعه ای AES، DES، CAST-128، ... استفاده کنند.

□ برخی مدهای کاری پراهمیت عبارتند از :

ECB: Electronic Code Book ■

CBC: Cipher Block Chaining ■

CTR: Counter Mode ■

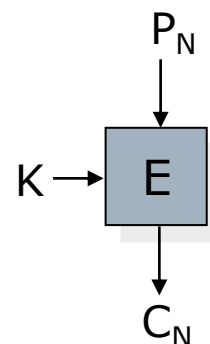
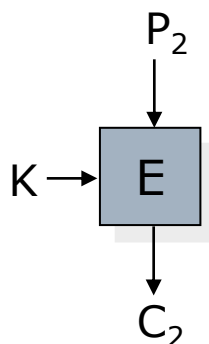
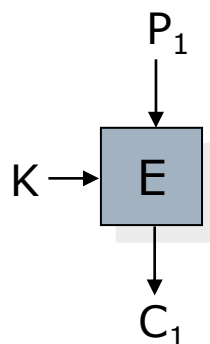
CFB: Cipher Feed Back ■

OFB: Output Feed Back ■

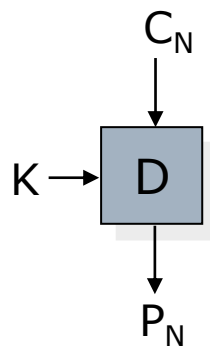
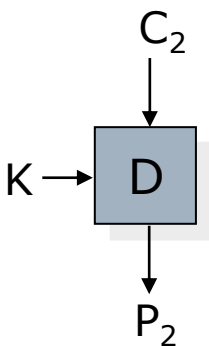
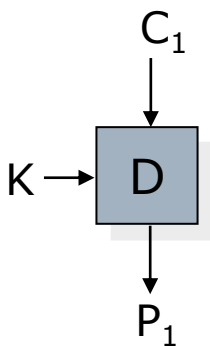
مد کاری ECB (Electronic Code Book)



رمزگذاری: □



رمزگشایی: □





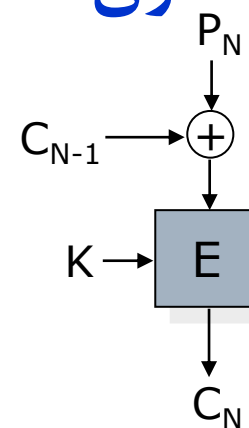
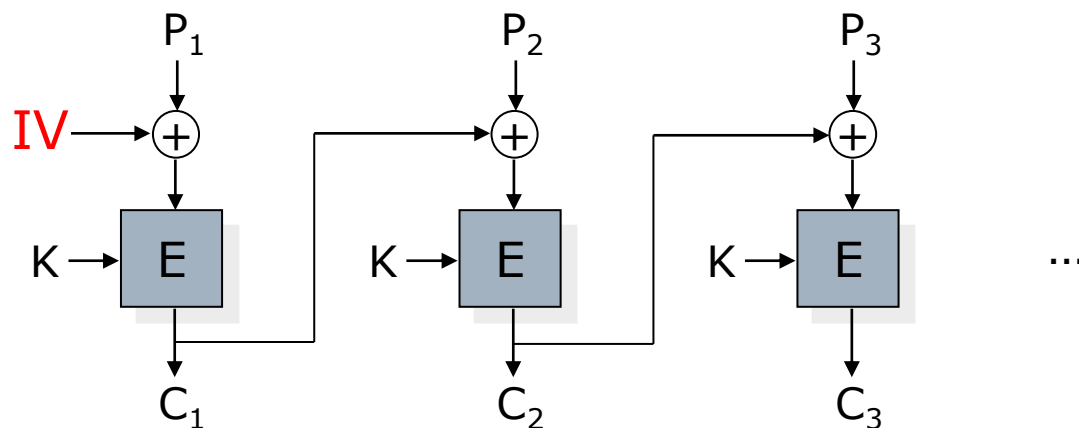
بررسی مد کاری ECB

- اشکال اساسی: هر متن آشکار به ازاء کلید ثابت همیشه به یک متن رمز شده نگاشته می شود.
- دشمن می تواند دریابد که پیام های یکسان ارسال شده اند.
- این مد امن محسوب نمی شود حتی اگر از یک رمز قطعه ای قوی استفاده کنیم.
- ECB مثالی از مواردی است که علی رغم بهره برداری از عناصر مرغوب، کیفیت نهایی دلخواه نیست.

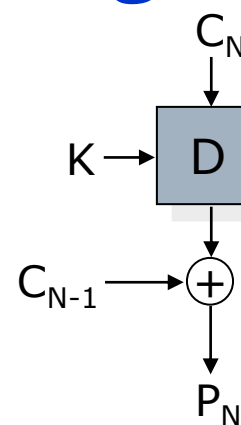
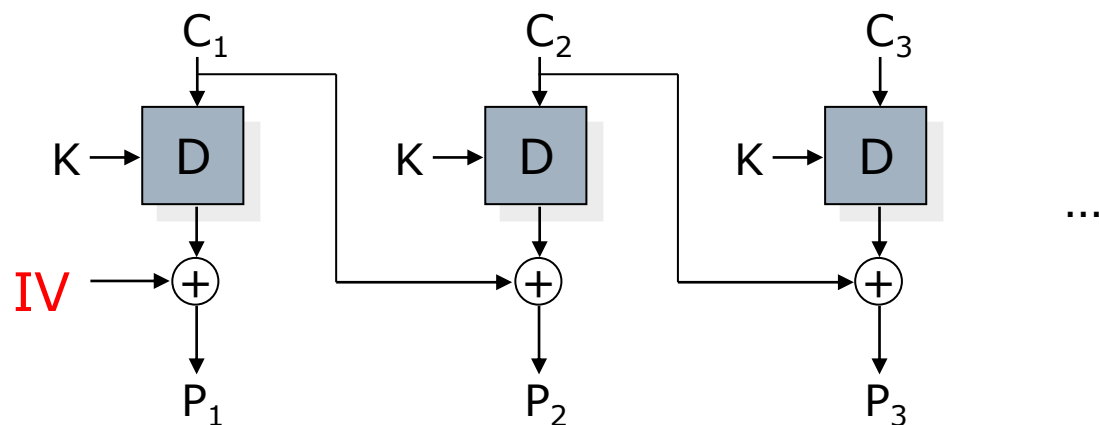
مد کاری CBC (Cipher Block Chaining)



رمزگذاری: □



رمزگشایی: □





مد کاری CBC

- این مد از یک مقدار دهی اولیه تصادفی (IV) بهره می گیرد.
- مقدار IV در هر بار رمزگذاری به صورت تصادفی تغییر می کند.
- IV همراه با متن رمز شده ارسال می شود.
- در صورت ارسال IV بصورت متن آشکار، تحلیلگر ممکن است بتواند با فرستادن IV جعلی موردنظر خود، منجر به تغییر خاصی در پیغام واگشایی شده در سمت گیرنده شود.
- IV نیز باید بصورت رمز شده ارسال شود. برای اینکار می توان از مد کاری ECB استفاده کرد.
- هر متن آشکار به ازاء کلید ثابت هر بار به یک متن رمز شده متفاوت نگاشته می شود (زیرا مقدار IV تغییر می نماید).



بررسی مد کاری CBC

□ ملزومات امنیتی:

- IV باید کاملاً غیر قابل پیش‌بینی باشد.

□ رمزگذاری:

- عملیات رمزگذاری قابل موازی‌سازی نیست.
- مقدار IV و متن آشکار باید در دسترس باشند.

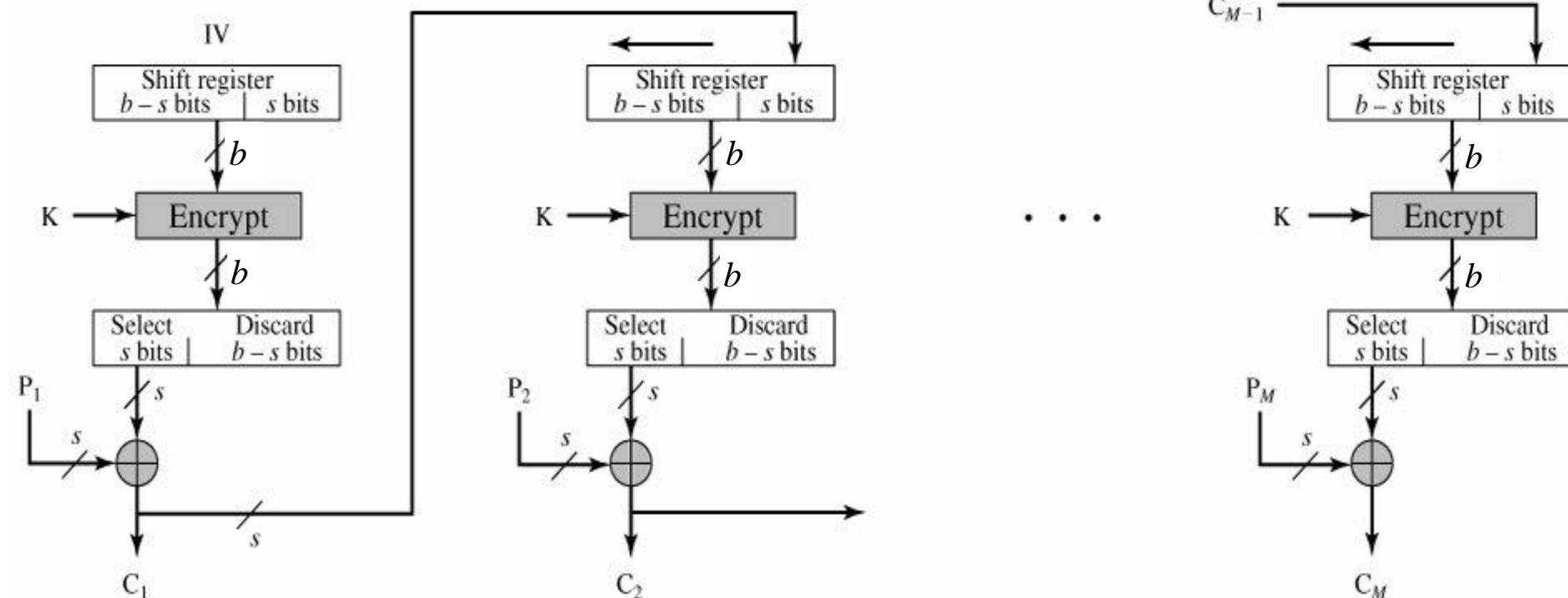
□ رمزگشایی:

- عملیات رمزگشایی قابل موازی‌سازی است.
- مقدار IV و متن رمز شده باید در دسترس باشند.

مد کاری CFB (Cipher Feed Back)



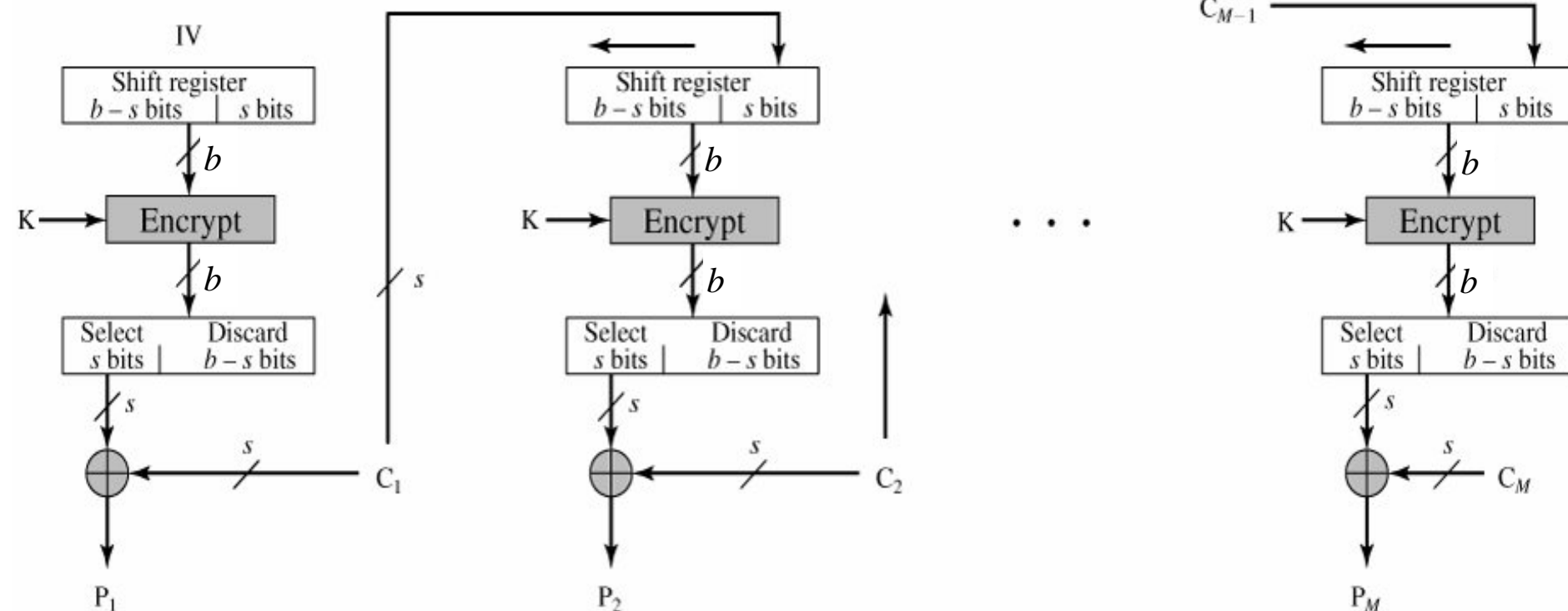
رمزگذاری □





مد کاری CFB

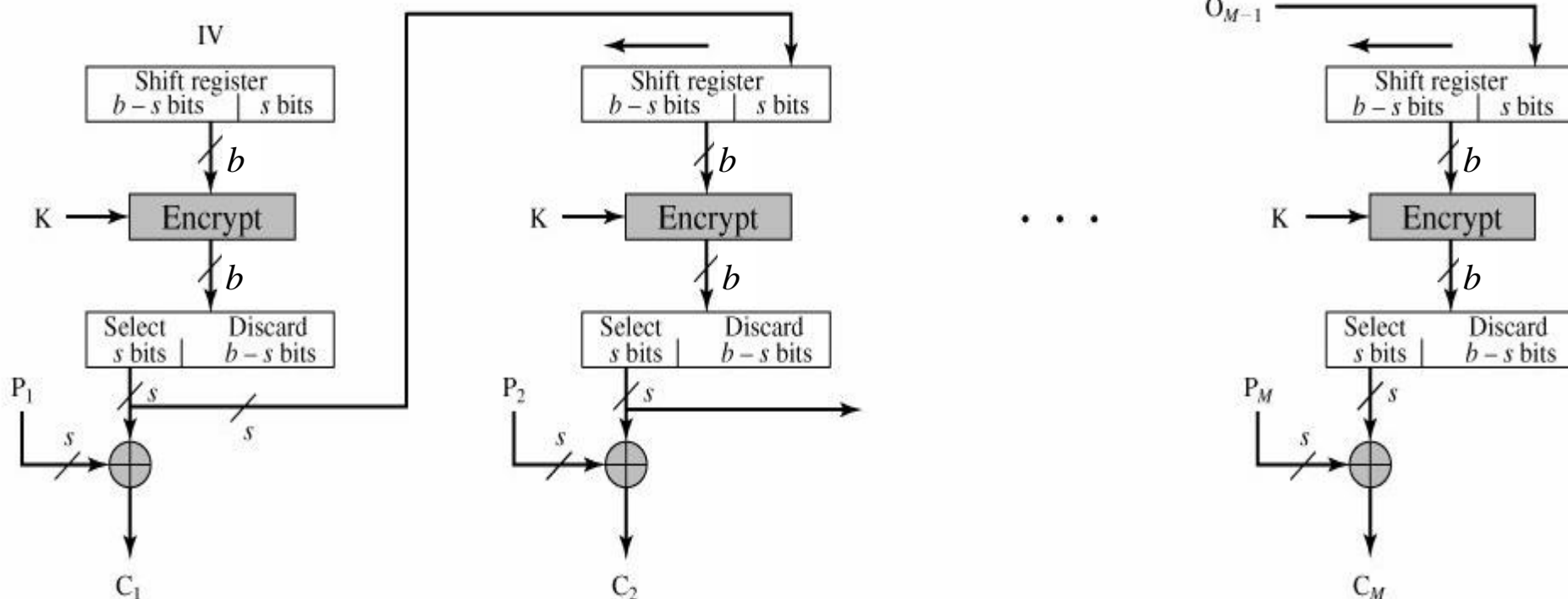
□ رمزگشایی



مد کاری OFB (Output Feed Back)



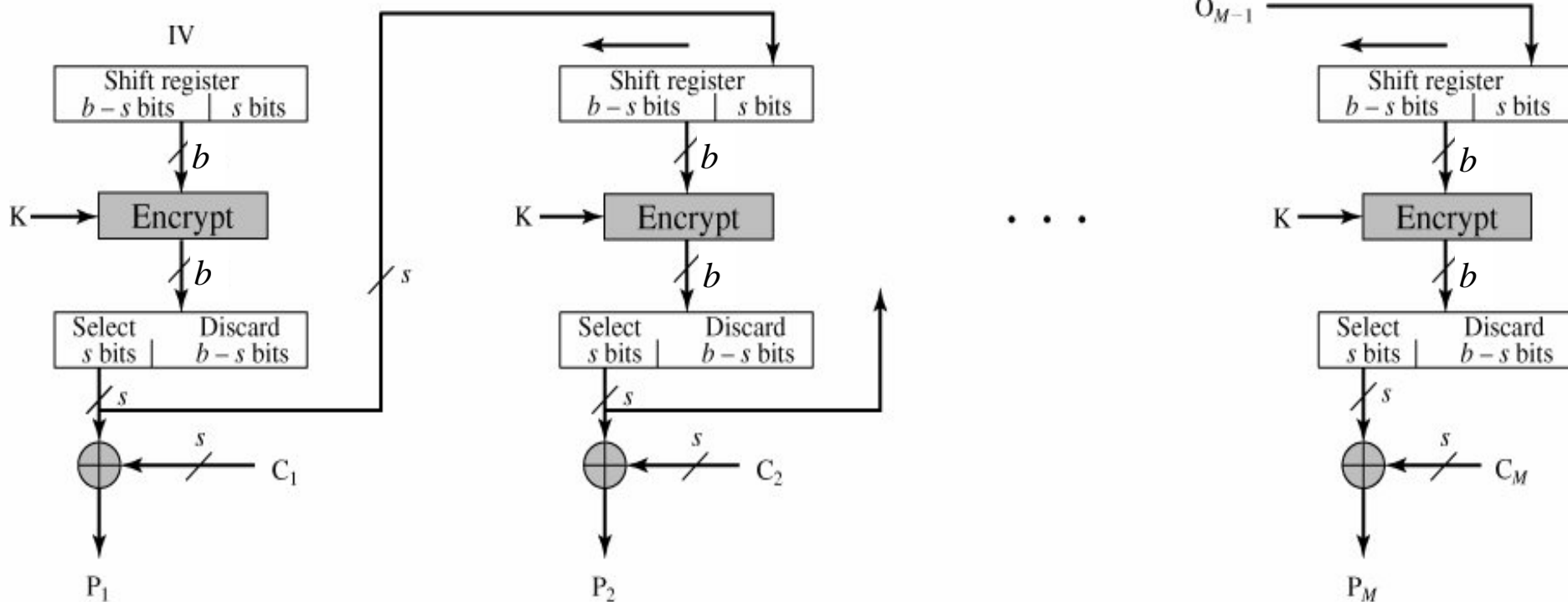
رمزگذاری □





مد کاری OFB

□ رمزگشایی





مقایسه CFB و OFB

□ موارد استفاده CFB و OFB

■ رمز جریانی

■ کاربردهای بی درنگ

□ عیب CFB: انتشار خطای انتقال

□ OFB این عیب را برطرف می کند.

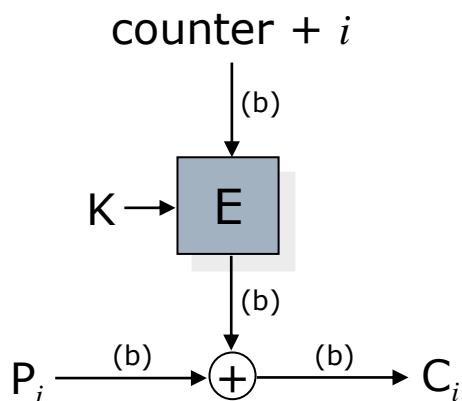


مد کاری CTR (Counter Mode)

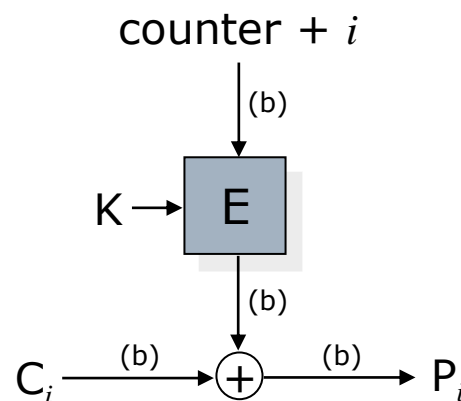
□ شمارنده به طول قطعه (b بیت) انتخاب شده و می تواند با مقدار اولیه صفر یا بصورت تصادفی انتخاب شود.

□ برای هر قطعه به شمارنده یک واحد اضافه می شود (در پیمانه 2^b)

□ رمز گذاری ↓



□ رمز گشایی ↓





بررسی مد کاری CTR

□ ملزومات امنیتی:

- مقادیر شمارنده، در بازه طول عمر کلید، باید مجزا باشند.

□ رمزگذاری:

- عملیات رمزگذاری قابل موازی سازی است.
- برای عملیات رمزگذاری نیازی به متن آشکار نیست.



بررسی مد کاری CTR

□ رمزگشایی:

- عملیات رمزگشایی قابل موازی سازی است.
- برای عملیات رمزگشایی نیازی به متن رمز شده نیست.

□ پیاده سازی:

- به شکل کارایی می تواند پیاده سازی سخت افزاری و نرم افزاری شود.
- از پردازش موازی می توان در آن استفاده کرد.



مقایسه کاربرد انواع مدهای کاری

کاربرد	مد کاری
ارسال مقادیر کوچک مانند کلید	ECB (Electronic Code Book)
ارسال قطعه-گرای هر گونه داده احراز صحت	CBC (Cipher Block Chaining)
ارسال جریانی هر گونه داده احراز صحت	CFB (Cipher Feed Back)
ارسال جریانی بر روی کانال نویزی (مانند ارتباطات ماهواره‌ای)	OFB (Output Feed Back)
ارسال قطعه-گرای هر گونه داده مناسب برای ارسال با سرعت بالا	CTR (Counter)



پایان

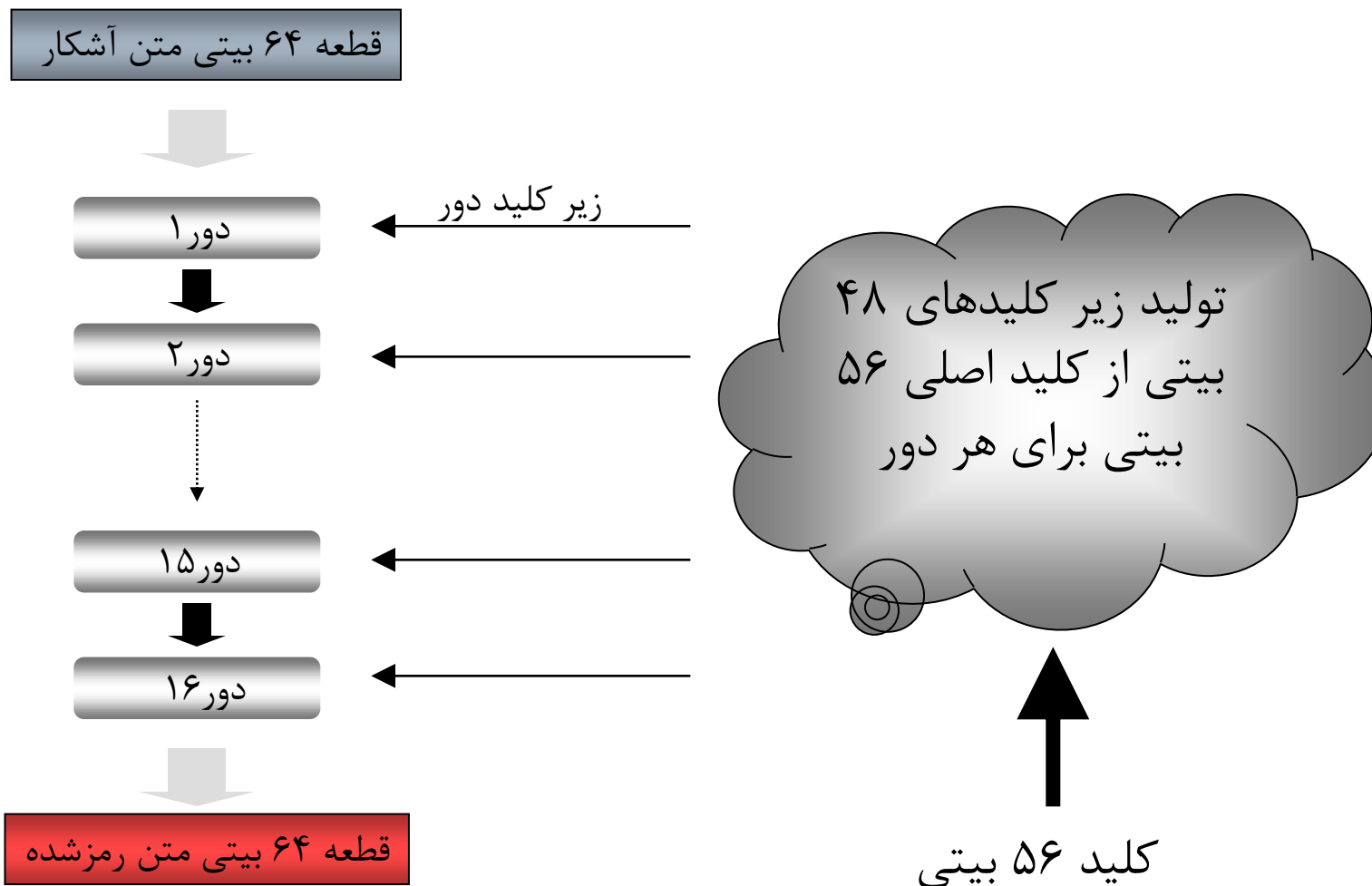


پیوست

الگوریتم رمزنگاری DES

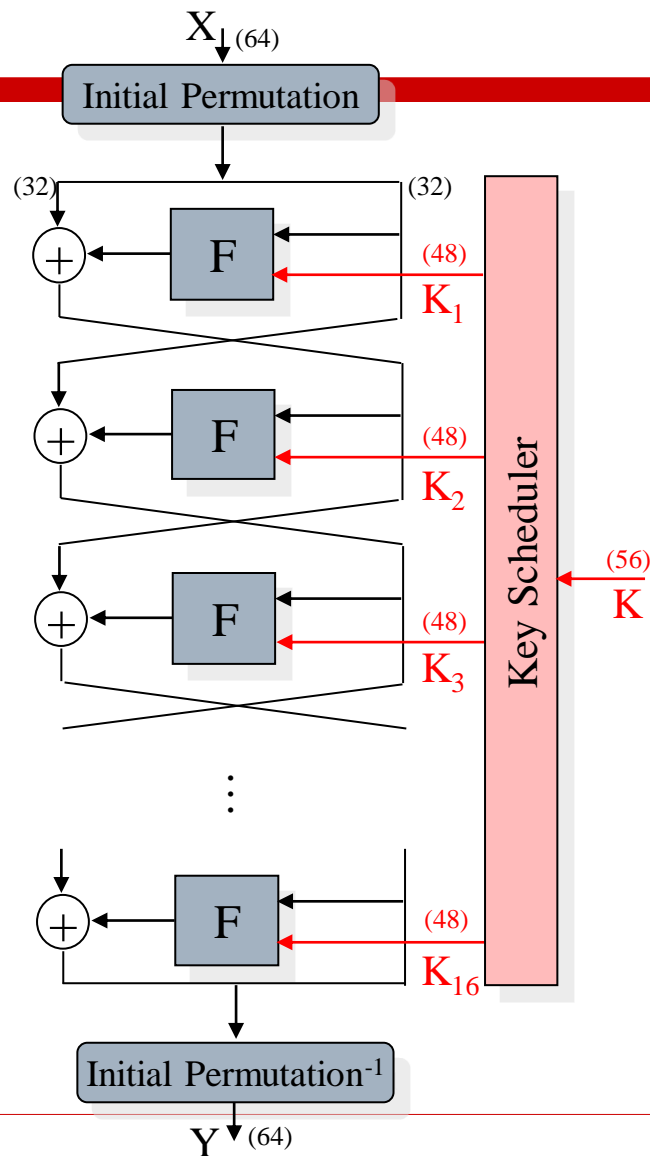


استاندارد رمزگذاری داده DES





ساختار فیستل رمز DES





جداول جایگشت اولیه

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

Initial Permutation (IP-1)

8	40	32	24	16	8	24	64	32									
1	33	25	17	9	1				23	63	31						
3	35	27	19	11	3							22	62	30			
5	37	29	21	13	5										21	61	29
7	39	31	23	15	7												
35		3	43	11	51	19	59	27									
34		2	42	10	50				18	58	26						
33		1	41	9	49							17	57	25			

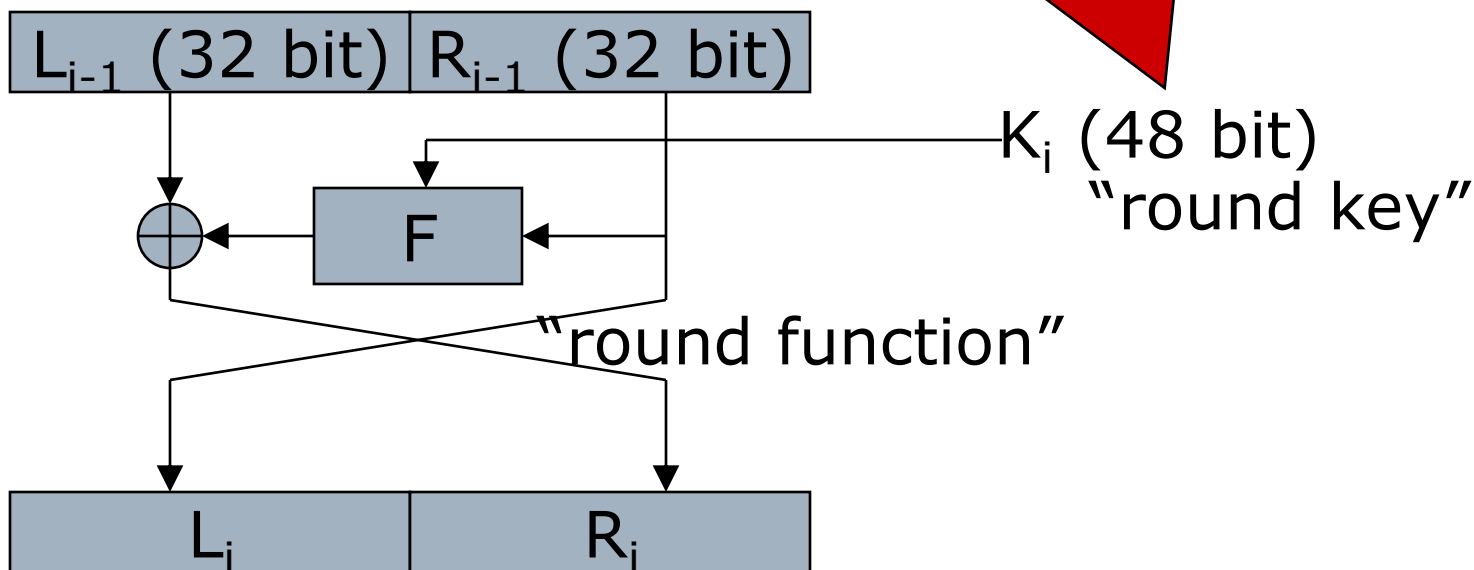
□ تاثیری در رمز ندارند.

□ صرفاً جهت تسهیل در بارگذاری بلوکها
در سخت افزارهای دهه ۷۰.



یک دور از DES

توسط زمانبند کلید
تولید میشود.



$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$

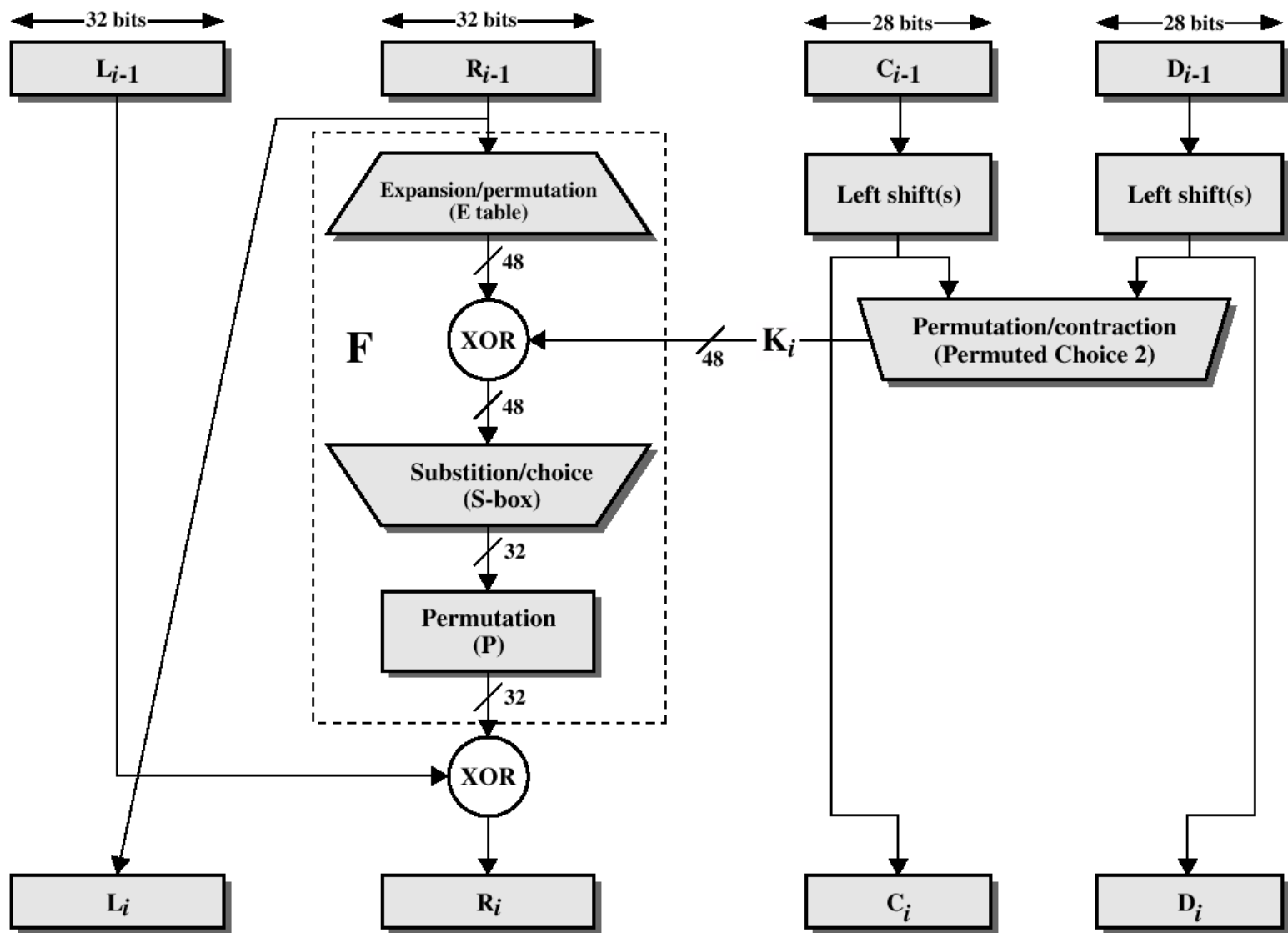
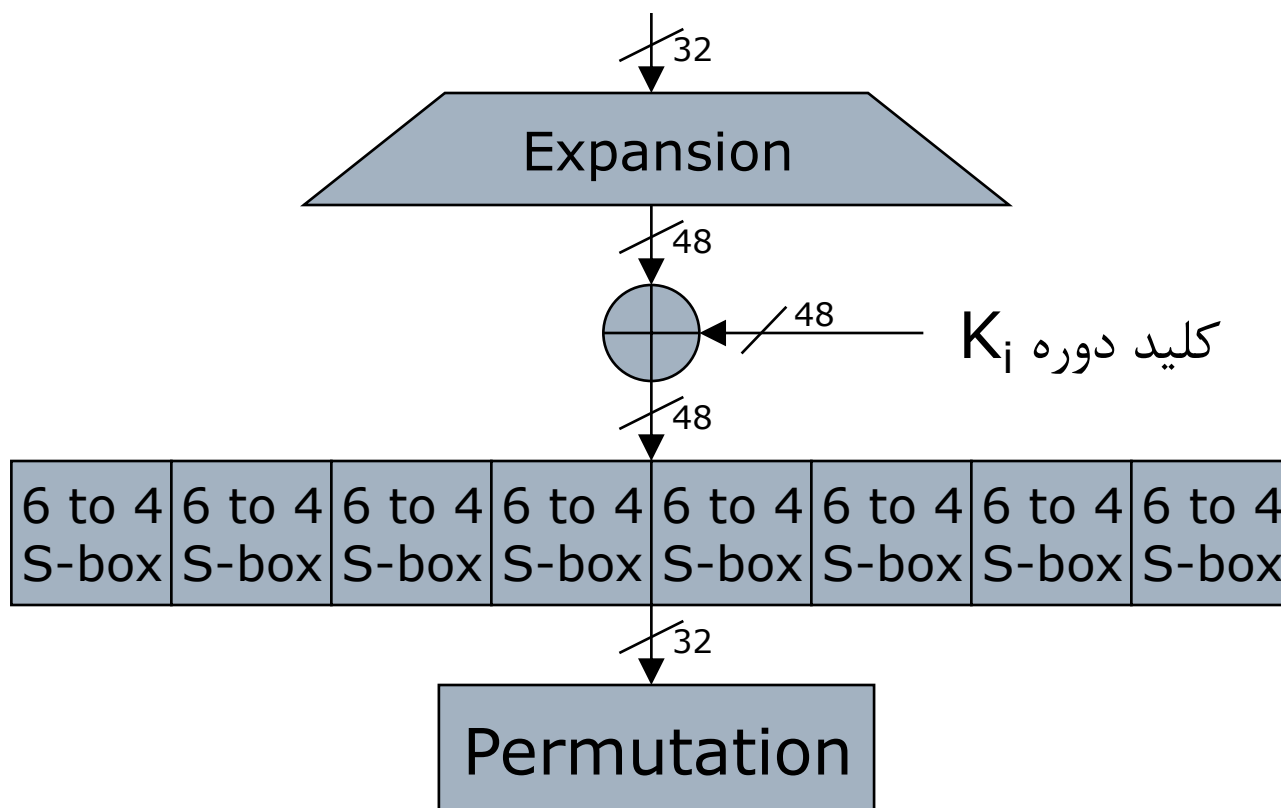


Figure 2.4 Single Round of DES Algorithm

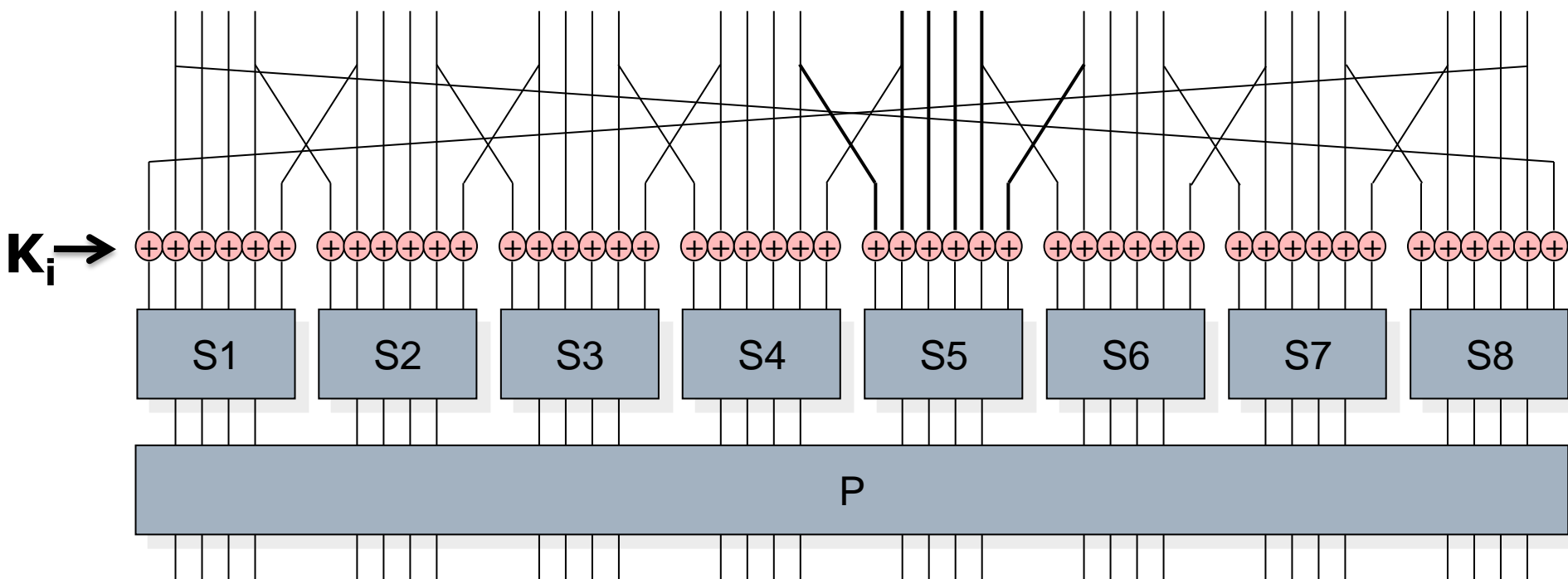


تابع دور DES





تابع دور DES





بررسی S-Box در DES

- تنها بخش غیرخطی از الگوریتم DES هستند.
- غیرقابل برگشت هستند.
- اصول طراحی آنها سری هستند.
- استفاده از ۸ S-Box که هر یک ۶ بیت ورودی را به ۴ بیت خروجی تبدیل می کنند.
- بیت های ۱ و ۶ : انتخاب یکی از ۴ سطر ماتریس
- بیت های ۲ تا ۵ : انتخاب یکی از ۱۶ ستون ماتریس
- برگرداندن عدد موجود در آن خانه از ماتریس به عنوان خروجی
- در مجموع ۴۸ بیت ورودی از هشت S-Box مختلف عبور می کنند و ۳۲ بیت برمی گردانند.



یک S-Box از DES

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

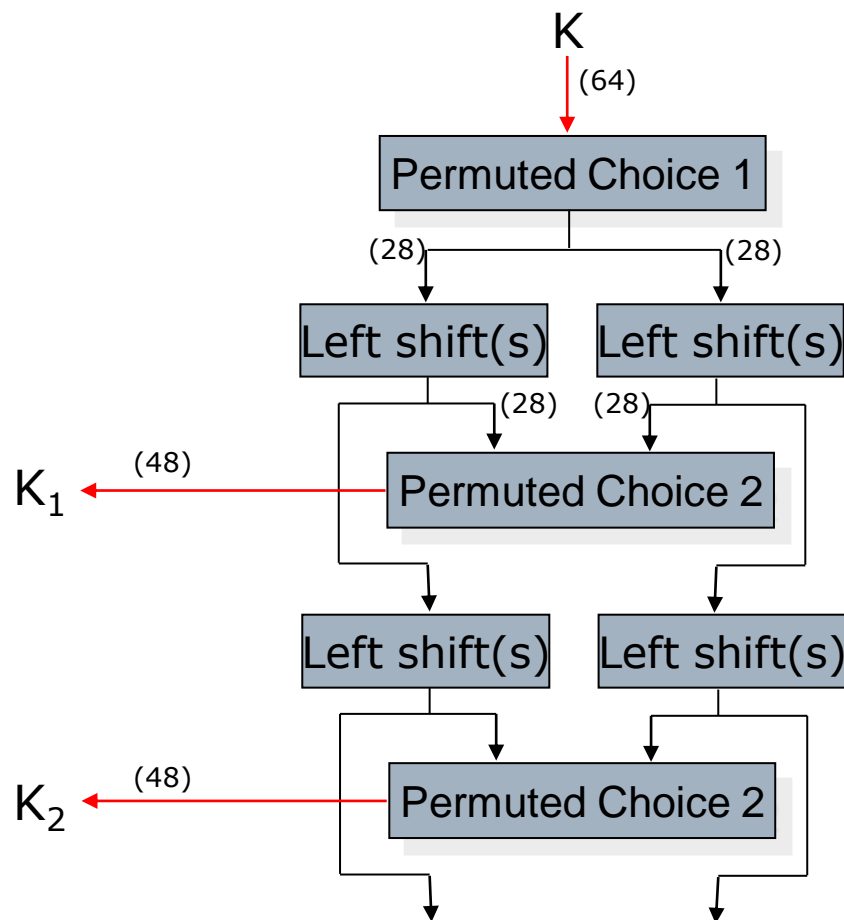


جدول جایگشت

□ جدول جایگشت مورد استفاده در هر دور DES

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

زمانبندی کلید



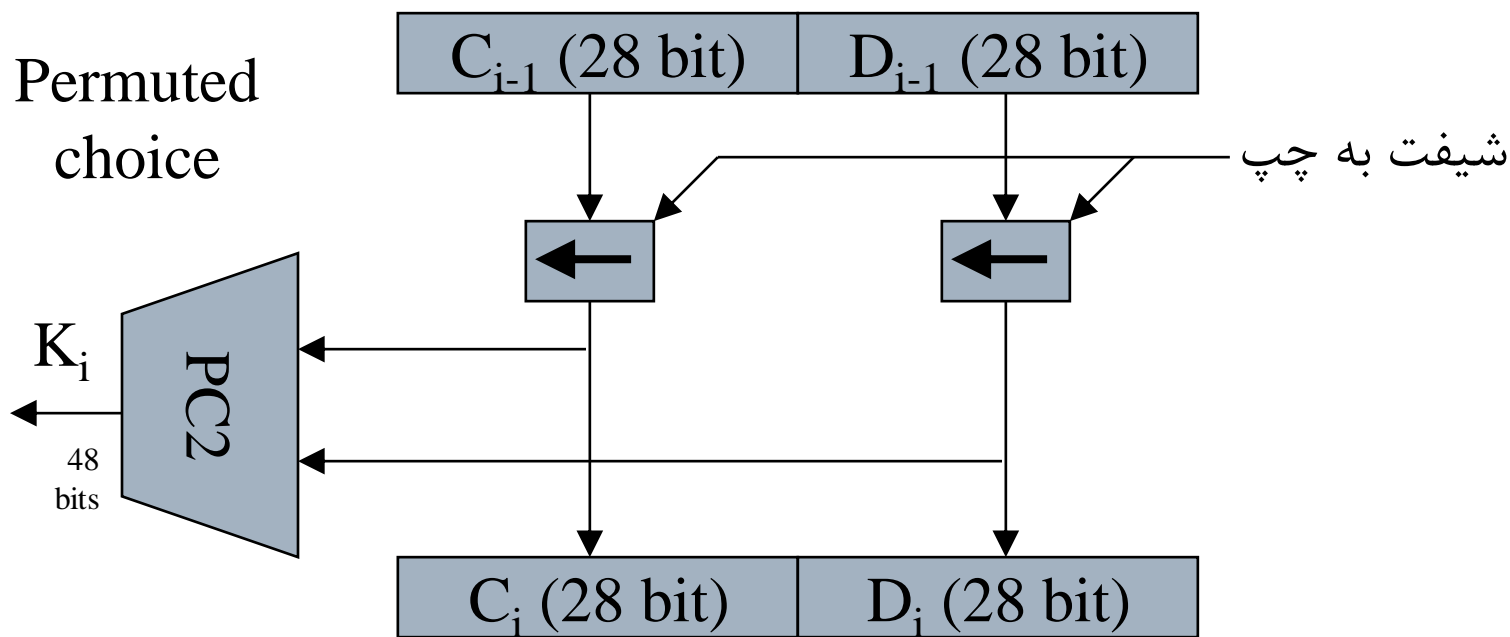
□ هر بیت کلید حدوداً در ۱۴ دور از ۱۶ دور استفاده می شود.

□ تابع تعبیه شده برای زمانبندی کلید، یک مقدار ۶۴ بیتی را به عنوان کلید می پذیرد ولیکن فقط ۵۶ بیت آن را استفاده می کند و بقیه به عنوان parity می تواند مورد استفاده قرار گیرد.



زمانبندی کلید

✓ کلید اصلی ۵۶ بیت
✓ کلید هر دور ۴۸ بیت





عناصر زمانبند کلید

□ شیفت چرخشی به چپ بر اساس جدول زیر

شماره دور

تعداد بیت شیفت

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

□ جداول جایگشت

Permuted Choice One (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

Permuted Choice Two (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



میزان توانمندی DES

□ اندازه کلید

■ ۵۶ بیت دارای کل فضای حالت $2^{56} = 7.2 * 10^{16}$

■ حمله آزمون جامع هرچند مشکل، ولی امکان پذیر است.

□ آخرین گزارش ثبت شده در سال ۱۹۹۹ نشان از کشف کلید تنها در عرض ۲۳ ساعت داده اند!

□ حمله زمانی

■ پیاده سازی الگوریتم رمز را مورد هدف قرار می دهند.

■ الگوریتم برای ورودی های مختلف مدت زمان متفاوتی صرف رمزگذاری می کند.

■ بیشتر در کارتهای هوشمند مشکل را می شوند.

■ DES در مقابل حمله زمانی مقاوم است.

Time to break a code (10^6 decryptions/ μ s)

