به نام خدا

دانشکده‌ی علوم ریاضی

مقدمه‌ای بر رمزنگاری ۱۳ آبان

جلسه‌ی ۱۴آ: رمزهای قالبی

مدرس: دکتر شهرام خزائی *نگارنده: پردیس ملک‌زاده و نیلوفر صفی صمغ آبادی*

یکی از ابزارهای مهم برای طراحی پروتکل‌های امنیتی را رمزهای قالبی[۱] تشکیل می‌دهند. در این جلسه ابتدا به معرفی رمزهای قالبی می‌پردازیم. سپس به معرفی دو روش طراحی رمزهای قالبی یعنی شبکه‌های جانشینی-جایگشتی (SPN[۲]) و شبکه‌های فایستلی[۳] می‌پردازیم.

۱ رمز قالبی

رمزهای قالبی، همان‌گونه که از نامشان پیداست روی قالب‌هایی از داده عمل می‌کنند. یک رمز قالبی دارای دو مؤلفه مهم می‌باشد:

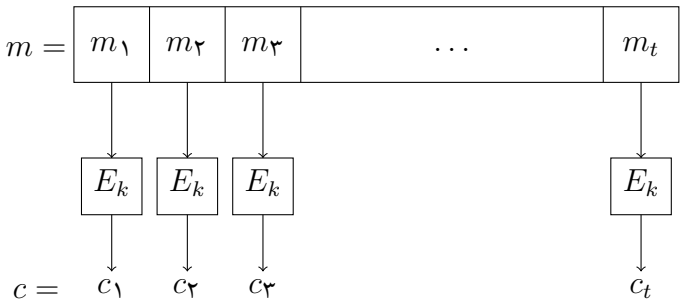
۱. اندازه قالب که آن را با n نشان می‌دهیم، و

۲. اندازه کلید (طول کلید) که آن را با $|k|$ نشان می‌دهیم.

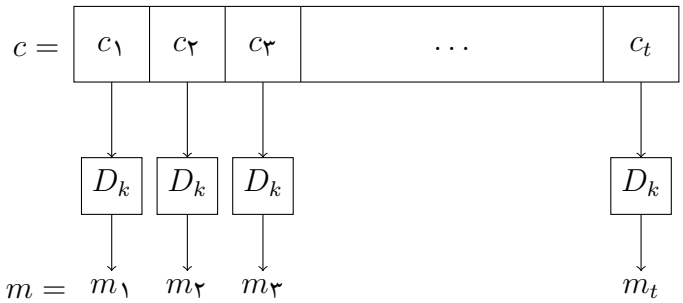
برای یک کلید داده شده، یک رمز قالبی با اندازه قالب n، مجموعه $\mathcal{M} = \{0,1\}^n$ از 2^n ورودی را به همان مجموعه $\mathcal{M}$ از 2^n خروجی تبدیل می‌کند، به‌طوری که هر خروجی ممکن فقط و فقط یک‌بار تولید می‌شود. در واقع این عمل یک جایگشت[۴] از مجموعه ورودی‌هاست و هنگامی که کلید تغییر کند، جایگشت‌های متفاوتی خواهیم داشت. بنابراین یک رمز قالبی روشی برای تولید خانواده‌ای از جایگشت‌هاست که توسط کلید محرمانه (k) نشانه‌گذاری می‌شوند.
ساده‌ترین روش استفاده از یک رمز قالبی، اعمال آن به قالب‌های متن اصلی است. برای رمزنگاری پیام m، ابتدا با استفاده از یک دنباله‌زنی[۵] مناسب (مثلاً اضافه کردن یک بیت یک و به تعداد لازم بیت صفر در انتهای پیام)، طول پیام به مضربی از طول قالب رسانده می‌شود. سپس متن اصلی به قالب‌های m_i تفکیک می‌شود. رمز قالبی، قالبی از یک پیام اصلی مانند m_i را به قالبی از یک پیام رمزشده مانند c_i تبدیل می‌کند. این نحوه به‌کارگیری رمز قالبی، سبک کتاب الکترونیک[۶] نامیده می‌شود که از امنیت قابل قبولی برخوردار نیست. اعمال رمز قالبی روی قالب iام

[۱] block ciphers
[۲] Substitution-Permutation Network (SPN)
[۳] Feistel network
[۴] permutation
[۵] padding
[۶] electronic code book mode

نیز اصطلاحاً رمزگذاری می‌گویند و به صورت $c_i = E_k m_i$ نشان داده می‌شود (شکل ۱). عکس عمل رمزگذاری، عمل رمزگشایی است که به صورت $m_i = D_k(c_i)$ نشان داده می‌شود (شکل ۲).



شکل ۱: نمای کلی الگوریتم رمزگذاری یک رمز قالبی



شکل ۲: نمای کلی الگوریتم رمزگشایی یک رمز قالبی

۲ روش‌های طراحی رمزهای قالبی

به‌طور کلی رمزهای قالبی دارای دو ساختار مهم هستند که از آن‌ها برای طراحی این رمزها استفاده می‌شود: شبکه‌های جانشینی-جایگشتی و شبکه‌های فایستلی که در ادامه به معرفی آن‌ها می‌پردازیم.

۱.۲ شبکه جانشینی-جایگشتی

۱.۱.۲ الگوی پریشانی-پخش

شانون[۷] یک طرح کلی برای ساخت جایگشت‌های شبه‌تصادفی معرفی کرد. ایده‌ی اصلی، ساخت جایگشت شبه‌تصادفی E با طول قالب بزرگ از تعداد زیادی جایگشت تصادفی یا شبه‌تصادفی کوچکتر $\{E_k^i\}$ با طول قالب کوچک‌تر است. فرض کنید بخواهیم رمزقالبی E با طول قالب ۱۲۸ بیت بسازیم. رمزقالبی E را به صورت زیر تعریف می‌کنیم: کلید k برای E، ۱۶ جایگشت تصادفی $E_k^{۱}, \cdots, E_k^{۱۶}$ را مشخص می‌کند که طول قالب هر یک ۸ بیت است. سپس ورودی $x \in \{۰,۱\}^{۱۲۸}$ را به ۱۶ قسمت ۸ بیتی متوالی $x_{۱۶}, ..., x_{۱}$ تجزیه می‌کنیم. داریم:

$$E_k(x) = E_k^{۱}(x_{۱}) \cdots E_k^{۱۶}(x_{۱۶}).$$

می‌گوییم $\{E_k^i\}$ ها پریشانی[۸] را برای E ایجاد می‌کنند.
جایگشت E که در مثال بالا تعریف کردیم شبه‌تصادفی نیست. زیرا اگر دو ورودی x و x' تنها در ۱ بیت اختلاف داشته باشند، $E_k(x)$ و $E_k(x')$ تنها در اولین بایت خود اختلاف خواهند داشت. برای حل این مشکل دو تغییر در روند بالا ایجاد می‌کنیم:

۱. بیت‌های خروجی $\{E_k^i\}$ها را با هم قاطی کرده و یک جایگشت از آن‌ها را در نظر می‌گیریم. به این مرحله پخش[۹] گفته می شود.

۲. پخش و پریشانی که با هم دور[۱۰] نامیده می‌شوند چندین بار تکرار می‌شوند.

استفاده‌ی متوالی از پخش و پریشانی تضمین می‌کند اعمال تغییرات کوچک در ورودی باعث اعمال تغییرات اساسی در خروجی می شود.

۲.۱.۲ شبکه جانشینی-جایگشتی

شبکه جانشینی-جایگشتی از اجرای مستقیم طرح پریشانی-پخش حاصل می‌شود. تفاوت اصلی در این است که در اینجا از جایگشت‌های ثابتی که به کلید وابسته نیستند و اصطلاحاً S-box نامیده می‌شوند استفاده می‌شوند. این کار نه تنها پیاده‌سازی را راحت‌تر می‌کند، بلکه باعث می‌شود با انتخاب مناسب اجزای رمزقالبی، بتوان آنها را امن‌تر طراحی کرد. شکل ۳ ساختار کلی یک شبکه جانشینی-جایگشتی را نشان می‌دهد که S-box با S_{ij} نمایش داده شده‌اند. کلید اولیه یک رمز قالبی کلید اصلی[۱۱] نامیده می شود و کلیدهای دور[۱۲] طبق طرح کلید[۱۳] از کلید اصلی استخراج می‌شوند. طرح کلید محاسباتی است که کلید دور مورد نیاز برای هر تکرار رمز قالبی را با استفاده از کلید اصلی تأمین می‌کند. هر دور از یک شبکه جانشینی-جایگشتی از تکرار سه مرحله زیر حاصل می‌شود:

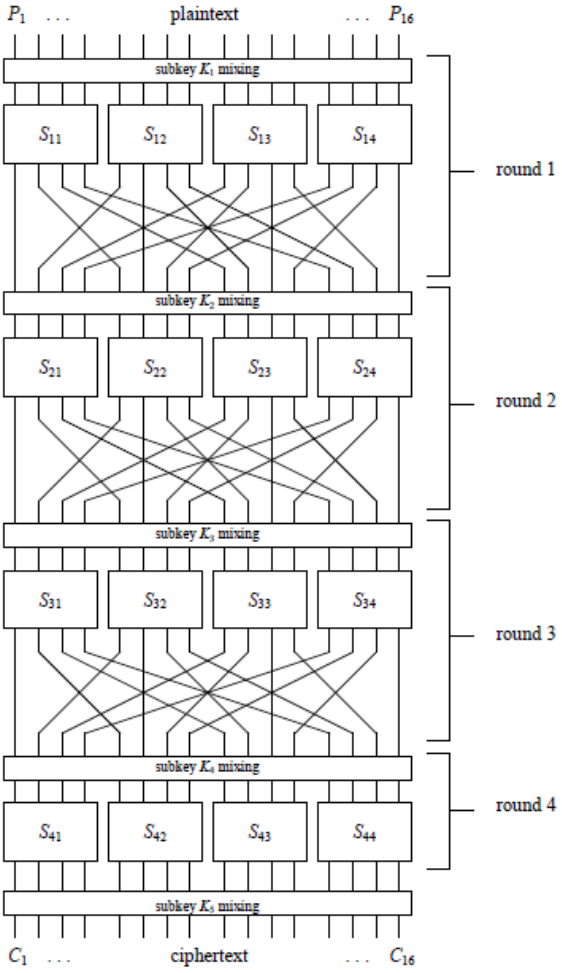
- اضافه شدن کلیدهای دور،

---

[۷] Shannon
[۸] confusion
[۹] diffusion
[۱۰] round
[۱۱] master key
[۱۲] round keys
[۱۳] key schedule

- جانشین‌سازی مناسب با استفاده از S-boxها،
- اعمال یک جایگشت (یا درحالت کلی یک تبدیل خطی) مناسب.

البته پس از آخرین دور در شبکه‌های جانشینی-جایگشتی، همواره یک زیر کلید دیگر نیز اضافه خواهد شد که سفیدسازی[۱۴] نامیده می‌شود. این کار باعث می‌شود تا جانشین‌سازی و جایگشت دور آخر در امنیت ساختار مؤثر باشد.



شکل ۳: یک شبکه جانشینی جایگشتی

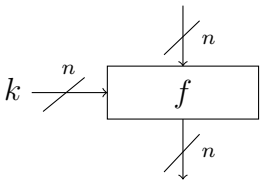
۳.۱.۲ امنیت شبکه جانشینی-جایگشتی

در صورتی که S-box ها، جایگشت استفاده شده و طرح کلید با دقت انتخاب شوند، شبکه‌های جانشینی جایگشتی گزینه‌ی مناسبی برای طراحی رمزهای قالبی هستند. امنیت این ساختار شدیداً به تعداد دورهای استفاده شده در آن وابسته است. رمز قالبی AES با استفاده از این روش طراحی شده است.

[۱۴] whitening

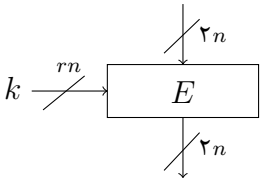
۲.۲ شبکه‌های فایستلی

در این بخش به مسأله طراحی رمز قالبی با استفاده از یک تابع شبه تصادفی می‌پردازیم. فرض کنید تابع شبه‌تصادفی $f_k(\cdot)$ را داریم که n بیت را به صورت زیر به n بیت می‌نگارد (شکل ۴).



شکل ۴: یک تابع شبه تصادفی با ورودی و خروجی n بیت

می‌خواهیم رمزی قالبی طراحی کنیم که مانند شکل ۵، $۲n$ بیت را به $۲n$ بیت می‌نگارد و طول کلیدش rn است.



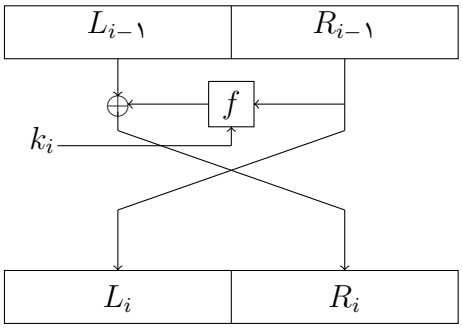
شکل ۵: یک رمز قالبی با ورودی و خروجی $۲n$ بیت

برای این منظور می‌توان از طراحی شبکه‌های فایستلی استفاده کرد که در ادامه به چگونگی آن می‌پردازیم:
یک رمز قالبی با اندازه قالب n که دارای ساختار شبکه‌های فایستلی می‌باشد، از تکرار r دور با ساختار یکسان تشکیل شده است. هر دور نیز دارای یک تابع دور[۱۵] و یک عمل‌گر جابجایی می‌باشد که نیمه‌های سمت چپ و راست ورودی خود را جابجا می‌کند (البته در دور آخر شبکه‌های فایستلی جابجایی وجود ندارد). تابع دور، n بیت نیمه‌ی سمت راست را تحت تأثیر زیرکلید دور به n بیت خروجی می‌نگارد که از این خروجی برای تغییر نیمه دیگر متن استفاده می‌شود. به عبارت دقیقتر دور i-ام یک شبکه‌ی فایستلی به صورت زیر اجرا می‌شود:

$$L_i = R_{i-۱},$$
$$R_i = L_{i-۱} \oplus f_{k_i}(R_{i-۱}).$$

که در آن همان‌طور که در شکل ۶ نشان داده شده است، k_i زیرکلید دور i، L_i نیمه چپ خروجی دور i و R_i نیمه راست خروجی دور i می‌باشند. این کار به صورت بازگشتی با شروع از دور اول، به تعداد r بار انجام می‌شود تا نهایتا خروجی تولید گردد. نمونه‌ای از رمزهای قالبی که دارای چنین ساختاری هستند، رمز قالبی DES می‌باشد.

[۱۵] round function



شکل ۶: یک دور از یک شبکه فایستلی

نکته قابل توجه این است که لزومی ندارد تابع دور یک شبکه فایستلی، جایگشت باشد تا خود شبکه فایستلی نیز یک جایگشت شود. در واقع معکوس یک شبکه فایستلی با استفاده از معکوس روابط بالا و بکارگیری روابط بازگشتی زیر انجام می‌گیرد:

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

نکته ۱ *تعداد کل جایگشت‌هایی که n بیت را به n بیت می‌نگارند $!2^n$ است؛ در حالیکه ما خانواده ای از جایگشت‌ها ایجاد کردیم که این تعداد را به 2^{rn} کاهش می‌دهد. بخصوص اگر r کوچک باشد این تعداد بسیار کمتر از تعداد کل جایگشت‌ها است. در عمل کلیدهای دور با استفاده از یک الگوریتم طرح کلید، از کلید اصلی استخراج می‌شوند.*

سؤالی که در اینجا مطرح است این است که تعداد دورها در شبکه‌ی فایستلی چقدر باشد، تا با فرض اینکه f شبه‌تصادفی است، خانواده جایگشتی که ایجاد کردیم شبه‌تصادفی باشد؟
شبکه‌های فایستل تک‌دوری و دو-دوری، تنها با دو پرسمان[۱۶] از E از یک جایگشت کاملاً تصادفی قابل تشخیص هستند.

سؤال ۱ *چگونه می‌توان شبکه فایستل سه‌دوری را از یک جایگشت کاملاً تصادفی تمیز داد؟*

شبکه فایستل سه‌دوری را نمی‌توان فقط با پرسمان از E از یک جایگشت کاملاً تصادفی تمیز داد.

قضیه ۱ (Luby-Rackoff) *فرض کنید تابع دور شبکه فایستل شبه‌تصادفی باشد. در این‌صورت:*

- *شبکه فایستل سه‌دوری با دسترسی اوراکلی به جایگشت، قابل تمایز از یک جایگشت کاملاً تصادفی نیست.*
- *شبکه فایستل چهار- دوری، یک جایگشت شبه‌تصادفی است.*

نکته ۲ *در عمل، توابع ساده‌ای برای f انتخاب می‌شوند که شبه‌تصادفی نیستند؛ در عوض تعداد دورها را زیاد می‌شوند.*

[۱۶] query

به نام خدا

دانشکده‌ی علوم ریاضی

مقدمه‌ای بر رمزنگاری ۲۱ آبان ۹۲

جلسه‌ی ۱۴ ب: AES و DES

مدرس: دکتر شهرام خزائی نگارنده: محمدامین شعبانی و حسین اورعی

۱ مقدمه

در جلسه‌ی قبل با روش‌های کلی طراحی رمزهای قالبی، از جمله طراحی مبتنی بر شبکه‌های فایستلی و جانشینی-جایگشتی آشنا شدیم. در این جلسه به بررسی دو الگوریتم رمز قالبی معروف یعنی AES و DES می‌پردازیم.

۲ رمز قالبی AES

رمز قالبی AES یا به عبارتی استاندارد رمزنگاری پیشرفته[۱]، الگوریتمی برای رمزنگاری داده‌های الکترونیکی است که در سال ۲۰۰۱ ایجاد گردید. این الگوریتم در ابتدا تحت نام رایندال[۲]، توسط دو رمزنگار بلژیکی به نام‌های یوآن دیمن[۳] و وینسنت رایمن[۴] طراحی شد. در رمز قالبی AES طول قالب ۱۲۸ بیت ثابت بوده و اندازه کلید می‌تواند ۱۲۸، ۱۹۲ و ۲۵۶ بیتی باشد. با توجه به طول کلید، تعداد دورها به ترتیب ۱۰، ۱۲ و ۱۴ می‌باشد.

B_0	B_4	B_8	B_{12}
B_1	B_5	B_9	B_{13}
B_2	B_6	B_{10}	B_{14}
B_3	B_7	B_{11}	B_{15}

شکل ۱: نحوه نمایش ورودی رمز قالبی AES

برای توضیح الگوریتم رمزنگاری AES ورودی این رمز قالبی را مانند شکل ۱ به صورت یک ماتریس مربعی 4×4 در نظر می‌گیریم. هر خانه این آرایه (با شروع از B_0 که کم ارزش‌ترین بایت ورودی است) نشان‌دهنده یک بایت

[۱] Advanced Encryption Standard
[۲] Rijndael
[۳] Joan Daemen)
[۴] Vincent Rijmen

از ورودی است. این ورودی همان‌طور که در شکل ۲ نشان داده شده است، تحت تأثیر عملگرهای SubBytes، ShiftRows، MixColumns و AddRoundKey به متن رمز شده تبدیل می‌گردد. در ادامه به تشریح این عملگرها می‌پردازیم. لازم به ذکر است که در ابتدای این رمز قالبی عمل سفید کردن نیز وجود دارد (شکل ۲).

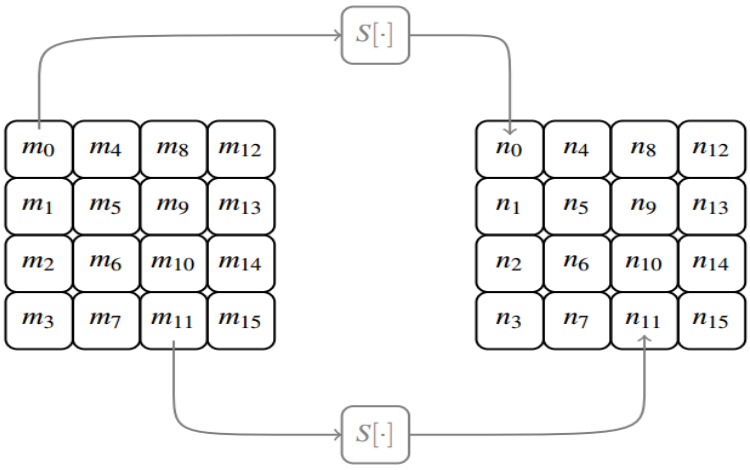
AES Encryption over r Rounds	
pre-whitening:	AddRoundKey
$r-1$ rounds:	SubBytes ShiftRows MixColumns AddRoundKey
final round:	SubBytes ShiftRows AddRoundKey

شکل ۲: نمای کلی الگوریتم رمز قالبی AES

۱.۲ SubBytes

این عملگر هر بایت از ورودی خود را تحت تأثیر S–جعبه به کار رفته در الگوریتم قرار می‌دهد (شکل ۳). به عبارت دقیقتر داریم:

$$n_i = S[m_i] \quad \text{for} \quad ۰ \leq i \leq ۱۵.$$



شکل ۳: نحوه کار عملگر SubBytes

تعریف $S-$جعبه AES نیز در شکل ۴ آمده است.

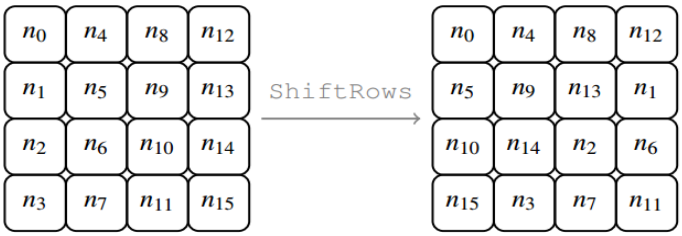
S[·]															
	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
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

شکل ۴: $S-$جعبه به کار رفته در رمز قالبی AES

می‌توان $S-$جعبه AES را با استفاده از میدان $GF(۲^۸) = GF(۲)[x]/(x^۸ + x^۴ + x^۳ + x + ۱)$ تیز تعریف کرد. بایت ورودی x به عنوان عضوی از میدان $GF(۲^۸)$ در نظر گرفته می‌شود. برای محاسبه $S(x)$ ابتدا معکوس x در $GF(۲^۸)$ محاسبه می‌شود (معکوس صفر، صفر در نظر گرفته می‌شود). سپس حاصل به عنوان یک بردار در $GF(۲)^۸$ در تفسیر می‌شود، تحت یک تبدیل مستوی قرار می‌گیرد و خروجی $S(x)$ می‌شود.

ShiftRows ۲.۲

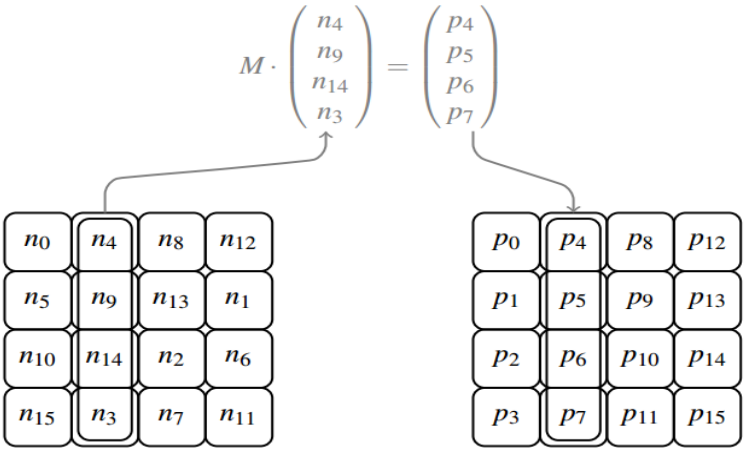
این عملگر روی سطرهای ماتریس ورودی خود عمل می‌کند. در این مرحله بایت‌های هر سطر به وسیله یک مقدار معین به صورت چرخشی شیفت می‌یابد؛ بدین صورت که سطر اول بدون تغییر باقی می ماند، سطر دوم یک واحد به سمت چپ شیفت می‌خورد و به همین ترتیب سطر n به تعداد $n - ۱$ بایت به صورت چرخشی به چپ شیفت می‌یابد (شکل ۵).



شکل ۵: نحوه کار عملگر ShiftRows

MixColumns ۳.۲

در این مرحله هر کدام از ستون‌های ماتریس ورودی، در ماتریس M که یک ماتریس ۴×۴ است در میدان $GF(۲^*)$ ضرب شده و ماتریس خروجی را تشکیل می‌دهد (شکل ۶). مرحله ShiftRowsبه همراه این مرحله خاصیت آشفتگی[۵] و پخش[۶] را در این رمز قالبی فراهم می‌آورند.



شکل ۶: نحوه کار عملگر MixColumns

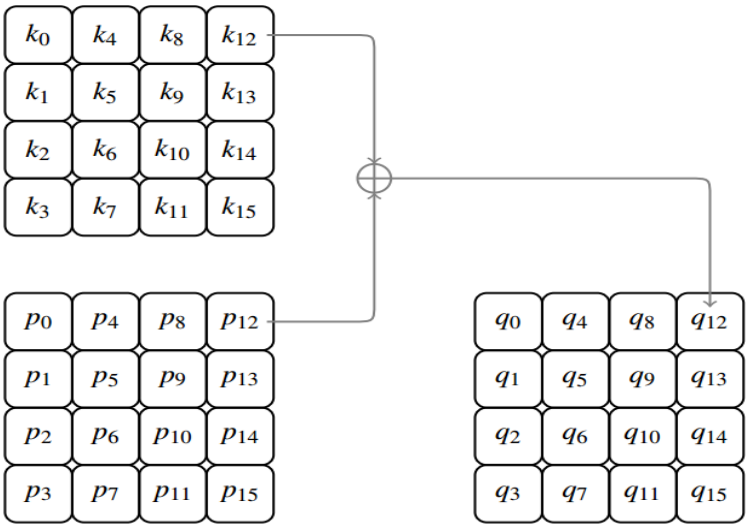
تعریف ماتریس M نیز به صورت زیر می‌باشد:

[۵] confusion
[۶] diffusion

$$M = \begin{pmatrix} ۰۲ & ۰۳ & ۰۱ & ۰۱ \\ ۰۱ & ۰۲ & ۰۳ & ۰۱ \\ ۰۱ & ۰۱ & ۰۲ & ۰۳ \\ ۰۳ & ۰۱ & ۰۱ & ۰۲ \end{pmatrix}$$

۴.۲ AddRoundKey

در این عملگر ماتریس ورودی با زیرکلید دور تولید شده توسط الگوریتم طرح کلید[۷] ترکیب می‌شود. بدین صورت که هر بایت از ماتریس ورودی با بایت متناظر زیرکلید دور XOR شده و ماتریس خروجی را تشکیل می‌دهند (شکل ۷).



شکل ۷: نحوه کار عملگر AddRoundKey

همان‌طور که ذکر شد اندازه کلید استفاده شده در رمز AES، تعداد دورها را تعیین می‌کند. یعنی برای مثال برای کلید ۱۲۸ بیتی مرحله میانی ۹ بار اجرا می‌شود که با اجرای مرحله پایانی ۱۰ دور تکمیل می‌شود.
برای رمزگشایی نیز مجموعه‌ای از دورهای معکوس برای تبدیل متن رمز شده به متن اصلی با استفاده از همان کلید رمزنگاری به کار گرفته می‌شود.

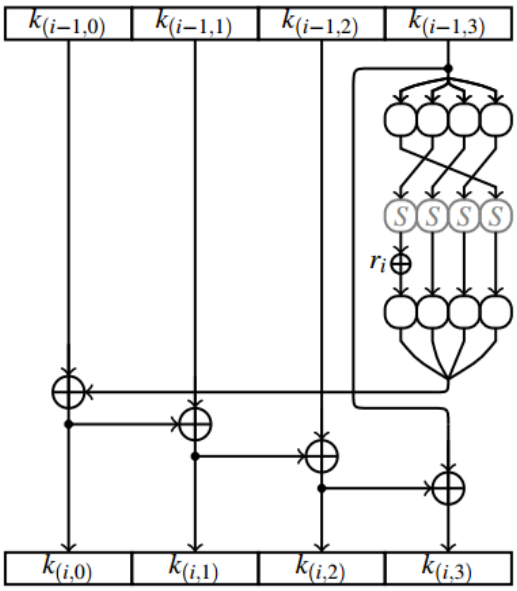
۵.۲ الگوریتم طرح کلید

الگوریتم طرح کلید رمز قالبی AES با توجه به طول کلید آن فرق می‌کند. در این قسمت الگوریتم طرح کلید AES-128 را توضیح می‌دهیم. در اینجا کلیدهای دور را به صورت زیر نشان می‌دهیم:

$$k_{(i,۰)}||k_{(i,۱)}||k_{(i,۲)}||k_{(i,۳)} \qquad ۰ \leq i \leq r$$

[۷] Key Schedule

که در آن i شماره دور و $k_{(i,0)}$، $k_{(i,1)}$، $k_{(i,2)}$ و $k_{(i,3)}$ کلمه‌های **۳۲** بیتی می‌باشند. زیرکلید هر دور با استفاده از زیرکلید دور قبل و به صورتی که در شکل ۸ نشان داده شده است به دست می‌آید.



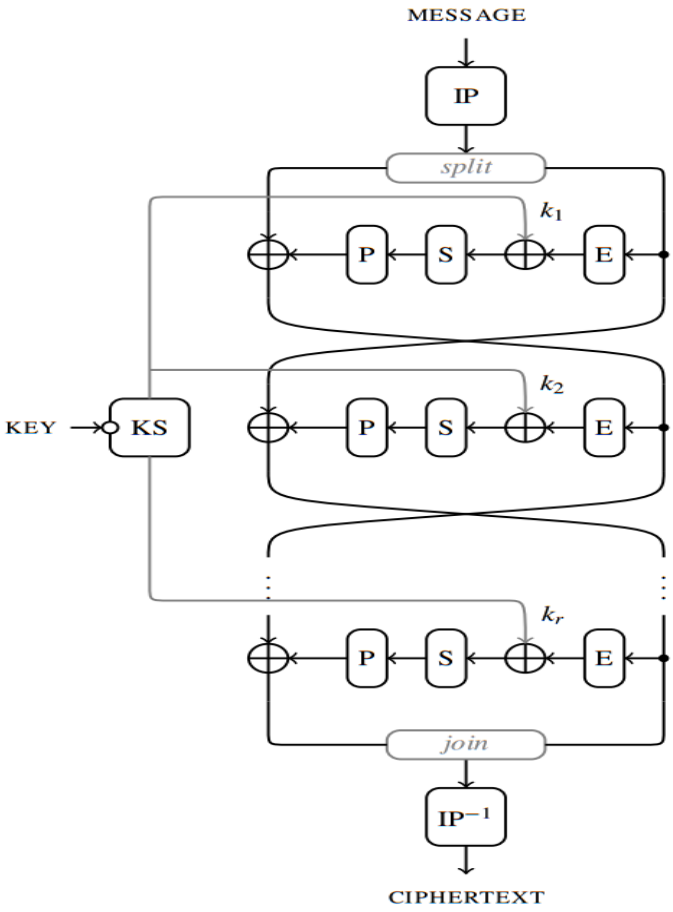
شکل ۸: طرح کلید AES

که در آن r_i برابر است با $۰۲^{i-۱}$ در میدان $GF(۲^۸)$.

۳ رمز قالبی DES

رمز قالبی DES یا همان استاندارد رمزنگاری داده[۸] در سال **۱۹۷۵** توسط شرکت IBM مطرح شد. الگوریتم DES یک شبکه‌ی فایستلی **۱۶** دوری می‌باشد. در این رمز قالبی طول قالب‌ها **۶۴** بیت است. کلید نیز شامل **۶۴** بیت می‌باشد ولی همان‌طور که خواهد آمد، در عمل تنها از **۵۶** بیت آن استفاده می‌شود. همان‌طور که در جلسه قبل گفته شد تابع دور شبکه فایستلی که آن را با f نشان می‌دهیم در هر دور بر روی نیمی از قالب ورودی عمل می‌کند. در نتیجه در رمز قالبی DES، طول ورودی تابع f در هر دور **۳۲** بیت می‌باشد. الگوریتم طرح کلید نیز برای هر دور با استفاده از کلید اصلی یک کلید **۴۸** بیتی را تولید می‌کند. در ادامه به بررسی تابع داخلی DES می‌پردازیم.

[۸] Data Encryption Standard



شکل ۹: ساختار کلی الگوریتم رمز قالبی DES

همان‌طور که در شکل ۹ نشان داده شده است، تابع f در ابتدا ورودی ۳۲ بیتی خود را گرفته و آن را توسط انتشار E به ۴۸ بیت تبدیل می‌کند. سپس مقدار بدست آمده را با زیرکلید دور ۴۸ بیتی بدست آمده در الگوریتم طرح کلید، XOR کرده و حاصل بدست آمده را به ۸ قسمت ۶ بیتی تقسیم می‌کند. حال هر کدام از این قسمت‌ها تحت تأثیر $S-$جعبه‌های متفاوتی که ۶ بیت را به ۴ بیت تبدیل می‌کنند، قرار می‌گیرند. سپس با کنار هم گذاشتن بیت‌های بدست آمده یک رشته ۳۲ بیتی تولید می‌شود. توجه کنید که چون $S-$جعبه‌ها ۶ بیت را به ۴ بیت تبدیل می‌کنند، در نتیجه برگشت‌پذیر نمی‌باشند. بعد از اثر $S-$جعبه‌ها، جایگشت P برای جابه‌جا کردن بیت‌ها بر روی آن‌ها اعمال می‌شود. در ادامه جزئیات الگوریتم رمز قالبی DES را شرح می‌دهیم.

۱.۳ جایگشت IP و وارون آن

همان‌طور که در شکل ۹ مشاهده می‌شود، در ابتدا و انتهای الگوریتم رمز قالبی DES به ترتیب جایگشت و وارون آن به کار گرفته می‌شوند. این جایگشت و وارون آن در شکل ۱۰ قابل مشاهده‌اند.

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

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

شکل ۱۰: جایگشت IP و وارون آن

به عنوان مثال تحت اثر IP، بیتی که در مکان ۵۸ ورودی است به مکان اول آورده می‌شود. در ادامه به بیان جزئیات تابع دور می‌پردازیم.

۲.۳ تعریف انتشار E

با توجه به شکل ۹، در هر دور DES ابتدا انتشار E روی ورودی تابع دور عمل می‌کند. تعریف انتشار E در شکل ۱۱ آمده است.

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

شکل ۱۱: تعریف انتشار E

همان‌طور که در شکل ۱۱ مشاهده می‌شود، ۳۲ بیت ورودی به ۴۸ بیت گسترش می‌یابد. به عنوان مثال تحت اثر E، بیتی که در مکان اول ورودی است به مکان‌های ۲ و ۴۸ برده می‌شود.

۳.۳ تعریف $S-$جعبه‌ها

همان‌طور که ذکر شد در هر دور، ۴۸ بیت خروجی E پس از ترکیب (XOR) با زیرکلید دور، به ۸ قسمت ۶ بیتی تقسیم می‌شود و روی هر قسمت یک $S-$جعبه متفاوت عمل می‌کند. تعریف این $S-$جعبه‌ها در شکل ۱۲ قابل مشاهده می‌باشد.

S1	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	e	4	d	1	2	f	b	8	3	a	6	c	5	9	0	7
p_1	0	f	7	4	e	2	d	1	a	6	c	b	9	5	3	8
p_2	4	1	e	8	d	6	2	b	f	c	9	7	3	a	5	0
p_3	f	c	8	2	4	9	1	7	5	b	3	e	a	0	6	d

S2	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	f	1	8	e	6	b	3	4	9	7	2	d	c	0	5	a
p_1	3	d	4	7	f	2	8	e	c	0	1	a	6	9	b	5
p_2	0	e	7	b	a	4	d	1	5	8	c	6	9	3	2	f
p_3	d	8	a	1	3	f	4	2	b	6	7	c	0	5	e	9

S3	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	a	0	9	e	6	3	f	5	1	d	c	7	b	4	2	8
p_1	d	7	0	9	3	4	6	a	2	8	5	e	c	b	f	1
p_2	d	6	4	9	8	f	3	0	b	1	2	c	5	a	e	7
p_3	1	a	d	0	6	9	8	7	4	f	e	3	b	5	2	c

S4	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	7	d	e	3	0	6	9	a	1	2	8	5	b	c	4	f
p_1	d	8	b	5	6	f	0	3	4	7	2	c	1	a	e	9
p_2	a	6	9	0	c	b	7	d	f	1	3	e	5	2	8	4
p_3	3	f	0	6	a	1	d	8	9	4	5	b	c	7	2	e

S5	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	2	c	4	1	7	a	b	6	8	5	3	f	d	0	e	9
p_1	e	b	2	c	4	7	d	1	5	0	f	a	3	9	8	6
p_2	4	2	1	b	a	d	7	8	f	9	c	5	6	3	0	e
p_3	b	8	c	7	1	e	2	d	6	f	0	9	a	4	5	3

S6	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	c	1	a	f	9	2	6	8	0	d	3	4	e	7	5	b
p_1	a	f	4	2	7	c	9	5	6	1	d	e	0	b	3	8
p_2	9	e	f	5	2	8	c	3	7	0	4	a	1	d	b	6
p_3	4	3	2	c	9	5	f	a	b	e	1	7	6	0	8	d

S7	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	4	b	2	e	f	0	8	d	3	c	9	7	5	a	6	1
p_1	d	0	b	7	4	9	1	a	e	3	5	c	2	f	8	6
p_2	1	4	b	d	c	3	7	e	a	f	6	8	0	5	9	2
p_3	6	b	d	8	1	4	a	7	9	5	0	f	e	2	3	c

S8	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
p_0	d	2	8	4	6	f	b	1	a	9	3	e	5	0	c	7
p_1	1	f	d	8	a	3	7	4	c	5	6	b	0	e	9	2
p_2	7	b	4	1	9	c	e	2	0	6	a	d	f	3	5	8
p_3	2	1	e	7	4	a	8	d	f	c	9	0	3	5	6	b

شکل ۱۲: تعریف S–جعبه‌های رمز قالبی DES

همان‌طور که مشاهده می‌شود، هر S–جعبه دارای ۴ سطر و ۱۶ ستون است. از شش بیت ورودی به هر S–جعبه، دو بیت خارجی سطر را مشخص می‌کنند و چهار بیت وسط مشخص کننده ستون می‌باشند. به عنوان مثال ورودی ۱۰۰۰۰۱ تحت اثر S_1 به f=1111 تصویر می‌شود.

۴.۳ تعریف جایگشت P

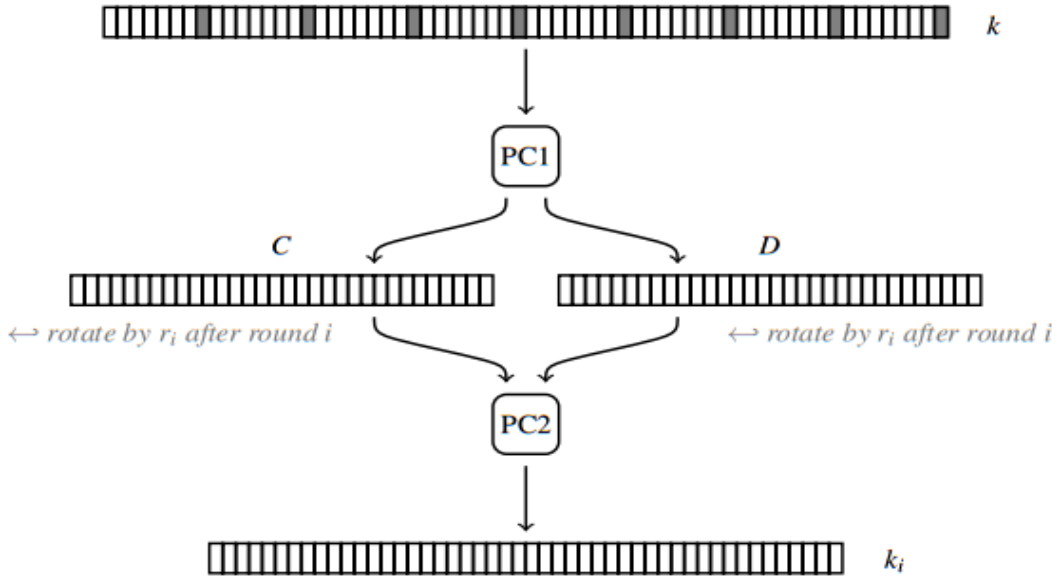
در انتهای تابع دور، جایگشت P روی ۳۲ بیت خروجی S–جعبه‌ها عمل می‌کند. تعریف این جایگشت در شکل ۱۳ آمده است.

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

شکل ۱۳: تعریف جایگشت P

۵.۳ طرح کلید DES

در این به بخش به توضیح طرح کلید رمز قالبی DES می‌پردازیم. در دور i، زیرکلید دور متناظر (k_i) به صورت زیر از روی کلید اولیه ساخته می‌شود:
با توجه به شکل ۱۴، از ۶۴ بیت کلید اولیه بیت‌های ۸، ۱۶، ..., ۶۴ حذف می‌شوند. سپس روی رشته ۵۶ بیتی باقیمانده جایگشت PC_1 که تعریف آن در شکل ۱۴ آمده است، اثر می‌کند و رشته حاصل به دو نیمه ۲۸ بیتی تقسیم می‌شود. سپس این دو نیمه به اندازه r_i به سمت چپ دوران می‌یابند. مقدار r_i به ازای دورهای مختلف در شکل ۱۴ آمده است. پس از دوران، این دو نیمه به یکدیگر الحاق[۹] می‌شوند. در نهایت رشته ۵۶ بیتی به دست آمده طبق جدول PC_2 به ۴۸ بیت تبدیل شده و زیرکلید دور k_i را تشکیل می‌دهد.



PC1							PC2					
57	49	41	33	25	17	9	14	17	11	24	1	5
1	58	50	42	34	26	18	3	28	15	6	21	10
10	2	59	51	43	35	27	23	19	12	4	26	8
19	11	3	60	52	44	36	16	7	27	20	13	2
63	55	47	39	31	23	15	41	52	31	37	47	55
7	62	54	46	38	30	22	30	40	51	45	33	48
14	6	61	53	45	37	29	44	49	39	56	34	53
21	13	5	28	20	12	4	46	42	50	36	29	32

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

شکل ۱۴: ساختار طرح کلید DES

[۹] concatenation

۴ امنیت رمزهای قالبی AES و DES

برای رمز قالبی AES بهترین حمله تا کنون در سال ۲۰۱۱ با استفاده از حمله دوبخشی[۱۰] ارائه شده است. برای مثال این حمله روی AES-128 به زمان $۲^{۱۲۶/۱۸}$ و حافظه $۲^{۸}$ نیاز دارد. پیچیدگی داده نیز $۲^{۸۸}$ می‌باشد. بنابراین AES از امنیت قابل ملاحظه‌ای برخوردار می‌باشد.
برای رمز قالبی DES اولین حمله در سال ۱۹۹۱ توسط بیهام[۱۱] و شامیر[۱۲] ارائه شد که پیچیدگی داده آن $۲^{۴۷}$ بود. حمله‌ی دیگری در سال ۱۹۹۳ توسط ماتسویی[۱۳] مطرح شد که نیاز به دانستن $۲^{۴۳}$ داده داشت. علاوه بر این حملات، حمله‌ای که همواره بر الگوریتم‌های رمزنگاری عمل می‌کند جستجوی تمامی کلیدهای ممکن می‌باشد. امروزه جستجوی فضای کلید ۵۶ بیتی با هزینه و زمان معقولی به راحتی امکان پذیر می‌باشد.

[۱۰] biclique
[۱۱] Eli Biham
[۱۲] Adi Shamir
[۱۳] Mitsuru Matsui