Fundamentals of Cryptography

Homework 2

Sepehr Ebadi 9933243

Question 1

1)

هدف از جایگشت‌ها :

جایگشت‌های اولیه (IP) و نهایی (FP) در DES از نظر امنیتی اهمیت چندانی ندارند؛ این جایگشت‌ها قدرت رمزنگاری DES را افزایش نمی‌دهند. اما چند هدف عملیاتی و ساختاری دارند:

بازتوزیع بیت‌ها: این جایگشت‌ها به شکلی قابل پیش‌بینی بیت‌ها را جابجا می‌کنند که باعث می‌شود ساختار ورودی و خروجی منظم‌تر شود و DES به عنوان یک ساختار فایستل معکوس‌پذیر عمل کند.

استانداردسازی و سازگاری: وجود این جایگشت‌ها استانداردسازی DES را ممکن می‌سازد و اطمینان می‌دهد که پیاده‌سازی‌های مختلف DES سازگاری و انطباق کاملی با هم داشته باشند.

عدم تأثیر بر امنیت: از آنجایی که IP و FP به راحتی قابل پیش‌بینی و معکوس هستند، هیچ قدرت رمزنگاری اضافه‌ای به DES نمی‌دهند، اما به سازگاری و قابلیت بازتولید آن کمک می‌کنند.

2)

جایگشت غیرخطی : S-Box 8 در DES به رمزنگاری غیرخطی اضافه می‌کنند، به این معنی که تغییرات کوچک در بیت‌های ورودی تغییرات غیرقابل پیش‌بینی در خروجی ایجاد می‌کند که موجب «avalanche effects» شده و امنیت را تقویت می‌کند.

منحصربه‌فرد بودن نگاشت‌ها : هر S-Box بیت‌های ورودی را به طور منحصر به فردی به بیت‌های خروجی تبدیل می‌کند که مقاومت در برابر الگوهای معمول که ممکن است توسط مهاجمین استفاده شود را افزایش می‌دهد.

ویژگی‌های کلیدی S-Box ها:

- Strict Avalanche Criterion (SAC): تغییر یک بیت در ورودی باید با احتمال ۵۰ درصد هر بیت خروجی را تغییر دهد. این ویژگی باعث می‌شود خروجی‌ها به‌طور تصادفی تغییر کنند و مقاومت رمز در برابر حملات افزایش یابد.
- معیار استقلال بیت‌ها: بیت‌های خروجی S-Box ها باید به طور مستقل برای هر بیت ورودی تغییر یافته تغییر کنند، که پیش‌بینی رفتار بیت‌ها را در هر دور برای مهاجم سخت‌تر می‌کند.
- مقاومت در برابر تحلیل تفاضلی: طراحی S-Box ها به گونه‌ای است که احتمال ایجاد الگوهای خاص در اختلافات ورودی و خروجی کاهش یابد تا مهاجمین نتوانند الگوهای تفاضلی را به راحتی در چند دور ردیابی کنند.

3)

Feistel Structure and Diffusion: ساختار Diffusion در DES تضمین می‌کند که هر بیت خروجی هر دور به بیت‌های دورهای قبلی وابسته باشد. در هر دور، نیمی از داده‌ها از خروجی دور قبلی و کلید استفاده می‌شود.

مکانیزم انتشار:

در طول ۱۶ دور، هر بیت ورودی تأثیر خود را به صورت متناوب در چندین بیت خروجی گسترش می‌دهد. به این ترتیب، هر بیت از متن ساده و کلید بر خروجی نهایی تأثیر می‌گذارد.

Avalanche Effect: به دلیل ماهیت S-Box ها و تابع فایستل، تغییر هر بیت در متن ساده یا کلید به صورت تصادفی و پیچیده‌ای در خروجی توزیع می‌شود.

عملیات گسترش و XOR : در هر دور، نیمه ۳۲ بیتی راست گسترش یافته، با کلید ترکیب شده و از طریق S-Box ها عبور داده می‌شود تا وابستگی بیت‌ها به طور گسترده‌تر انتشار یابد.

اثبات وابستگی کامل: پس از ۱۶ دور، هر بیت در خروجی به هر بیت از ورودی اولیه و کلید وابسته است، به دلیل تبدیل‌های غیرخطی و انتشار وابستگی که تضمین می‌کنند تمام بیت‌ها به‌صورت پیچیده و کامل به هم وابسته باشند.

فرض کنید که یک بیت در ورودی تغییر می کند، برای مثال اولین بیت در متن ساده. با فرض ساختار فایستل، در هر دور، تغییرات به‌صورت غیرخطی توسط S-Box ها و عملیات XOR گسترش می‌یابد. به طور خلاصه، هر بیت از متن ساده و کلید حداقل یک بار در هر دور از طریق ساختار فایستل به تمام بیت‌های نیمه چپ و راست وابسته می‌شود.

برای مثال، اگر ورودی اولیه ۶۴ بیت و به‌صورت P=P1,P2,…,P64 باشد و P1 تغییر کند، این تغییر پس از ۱۶ دور به این شکل گسترش می‌یابد که همه بیت‌های L16 و R16 به P1 و کلیدها K1,K2,…,K16K وابسته هستند.

بنابراین، پس از ۱۶ دور، هر بیت خروجی تابعی از تمامی بیت‌های ورودی و کلیدها است و به‌صورت کامل به آن‌ها وابسته شده است، که این ویژگی انتشار در DES امنیت و مقاومت رمزنگاری را افزایش می‌دهد.

Question 2

1)

روش جستجوی کلید کامل (Exhaustive Key Search):

حجم فضای کلیدها: DES از یک کلید ۵۶ بیتی استفاده می کند، که به معنی $۲^{۵۶}$ (حدود ۷۲ کوادریلیون) ترکیب ممکن برای کلید است. در روش حمله‌ی آزمون و خطا، مهاجم با استفاده از هر یک از این ترکیب‌های ممکن سعی می کند کلید صحیح را پیدا کند.

ساختار DES و امکان‌پذیری حمله: ساختار DES به گونه‌ای است که عملیات رمزگشایی سریع انجام می‌شود و هر کلید به راحتی می تواند بررسی شود. از آنجا که فضای کلیدهای DES نسبتاً کوچک است، این امکان وجود دارد که یک مهاجم با استفاده از توان محاسباتی کافی همه ترکیب‌های کلید را امتحان کند.

چالش‌های عملیاتی و پیچیدگی پیاده‌سازی:

منابع محاسباتی مورد نیاز: هرچند جستجوی کامل کلیدهای DES از نظر تئوری قابل اجرا است، اما به توان محاسباتی قابل توجهی نیاز دارد. برای مثال، با سخت‌افزارهای معمولی ممکن است زمان زیادی برای جستجو نیاز باشد.

2)

معرفی Triple DES :

- ساختار Triple DES : 3DES برای افزایش امنیت DES ، از سه مرحله رمزگذاری و رمزگشایی استفاده می‌کند. یک بار رمزگذاری، سپس رمزگشایی، و دوباره رمزگذاری. این فرآیند به دو یا سه کلید ۵۶ بیتی نیاز دارد.
- افزایش فضای کلیدها: با استفاده از 3DES، فضای کلید به 2^{112} یا 2^{168} ترکیب افزایش می‌یابد که فضای جستجو را بزرگ‌تر و امنیت را در برابر جستجوی کامل تقویت می‌کند.

محدودیت‌های Triple DES :

کارایی محاسباتی: اجرای سه‌باره DES باعث افزایش زمان پردازش و کاهش کارایی رمزنگاری می‌شود. این امر برای سیستم‌هایی که به سرعت بالایی نیاز دارند، مشکل‌ساز است و باعث می‌شود که 3DES در برخی کاربردها مناسب نباشد.

مقاومت در برابر حملات مدرن: در حالی که 3DES امنیت بیشتری نسبت به DES دارد، هنوز در برابر برخی از حملات جدیدتر مانند حملات جستجوی میانجی (Meet-in-the-Middle) آسیب‌پذیر است، زیرا تنها امنیت محدودتری نسبت به الگوریتم‌های مدرن مانند AES ارائه می‌دهد.

Question 3

1)

$$A(x) + B(x) = (x^2 + 1) + (x^3 + x^2 + 1) mod\, p(x) = x^3$$

$$A(x) * B(x) = (x^2 + 1) * (x^3 + x^2 + 1) = x^5 + x^4 + x^3 + x^2 + x^2 + 1 = x^5 + x^4 + x^3 + 1$$

$$x^4 + x + 1 = 0\, mod\, p(x) \rightarrow$$

$$x^4 = x + 1 \, mod \, p(x)$$
$$A(x) * B(x) \, mod \, p(x)$$
$$x^5 + x^4 + x^3 + 1 \, mod \, p(x)$$
$$x^2 + x + x + 1 + x^3 + 1 \, mod \, p(x)$$
$$x^3 + x^2$$

2)

$$A(x) + B(x) = (x^2 + 1) + (x + 1) \, mod \, p(x) = x^2 + x$$
$$A(x) * B(x) = (x^2 + 1) * (x + 1) = x^3 + x^2 + x + 1$$
$$A(x) * B(x) \, mod \, p(x) = \ x^3 + x^2 + x + 1$$

Question 4

1)

هدف پیدا کرن p(x) های درجه ۳ است($a_3 = 1$). همچنین باید در نظر بگیریم که چند جمله ای مورد نظر باید irreducible باشد.

$$p(1) \neq 0 \, , p(0) \neq 0$$
$$p(0) \neq 0 \rightarrow a_0 \neq 0 \rightarrow a_0 = 1$$
$$p(1) \neq 0 \rightarrow 1 * 1 + a_2 + a_1 + 1 \neq 0 \rightarrow \ a_2 + a_1 \neq 0 \, mod \, 2$$

دو حالت داریم:

$$a_2 = 1 \rightarrow p(x) = x^3 + x^2 + 1$$
$$a_1 = 1 \rightarrow p(x) = x^3 + x + 1$$

$$x^3 + x^2 + 1$$
$$x^3 + x + 1$$

2)

هدف پیدا کرن p(x) های درجه ۴ است($a_4 = 1$). همچنین باید در نظر بگیریم که چند جمله ای مورد نظر باید irreducible باشد.

$$p(1) \neq 0 \, , p(0) \neq 0$$

$$p(0) \neq 0 \rightarrow a_0 \neq 0 \rightarrow a_0 = 1$$
$$p(1) \neq 0 \rightarrow 1 * 1 + a_3 + a_2 + a_1 + 1 \neq 0 \rightarrow \ a_3 + a_2 + a_1 \neq 0 \, mod \, 2$$

$$a_3 = 1, a_2 = 1, a_1 = 1 \rightarrow p(x) = x^4 + x^3 + x^2 + x + 1$$
$$a_3 = 1 \rightarrow p(x) = x^4 + x^3 + 1$$
$$a_1 = 1 \rightarrow p(x) = x^4 + x + 1$$

$$p(x) = x^4 + x^2 + 1 \; mod \; 2 = (x^2 + x + 1)^2$$

$$x^4 + x^3 + x^2 + x + 1$$
$$x^4 + x^3 + 1$$
$$x^4 + x + 1$$

Question 5

با استفاده از جدول داخل کتاب میتوان خروجی S-box را بدست آورد.

Table 4.3 AES S-Box: Substitution values in hexadecimal notation for input byte (xy)

	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
x 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

$$B=ByteSub(A)=\begin{matrix}16 & 16 & 16 & 16\\ 16 & 16 & 16 & 16\\ 16 & 16 & 16 & 16\\ 16 & 16 & 16 & 16\end{matrix}$$

عملیات ShiftRow تاثیری ندارد.

$$C = MixColumn(B) = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \end{bmatrix}$$
$$= \begin{bmatrix} (02 + 03 + 01 + 01) * 16 \\ (01 + 02 + 03 + 01) * 16 \\ (01 + 01 + 02 + 03) * 16 \\ (03 + 01 + 01 + 02) * 16 \end{bmatrix}$$

در میدان توسعه یافته $GF(2^8)$:

$$01 \equiv 0000\ 0001 \equiv 1, 02 \equiv 0000\ 0010 \equiv x, 03 \equiv 0000\ 0011 \equiv x + 1$$
$$\rightarrow 01 + 01 + 02 + 03 \equiv 1 + 1 + x + x + 1 \equiv 1\ mod\ 2$$
$$\rightarrow 01 * 16 = 16$$

عملیات MixColumn تغییری نمیکند.

عملیات AddRoundKey :

$$C \oplus K = \begin{bmatrix} 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 \end{bmatrix} \oplus \begin{bmatrix} FF & FF & FF & FF \\ FF & FF & FF & FF \\ FF & FF & FF & FF \\ FF & FF & FF & FF \end{bmatrix} = \begin{bmatrix} E9 & E9 & E9 & E9 \\ E9 & E9 & E9 & E9 \\ E9 & E9 & E9 & E9 \\ E9 & E9 & E9 & E9 \end{bmatrix}$$

Question 6

1)

در الگوریتم رمزنگاریAES ، لایه‌های ShiftRows و MixColumns نقش مهمی در ایجاد انتشار یا Diffusionایفا می کنند. هر دو این لایه‌ها به گسترش تغییرات کوچک در داده‌های ورودی کمک می کنند، که در نهایت امنیت الگوریتم AES را در برابر تحلیل‌های رمزنگاری افزایش می‌دهد.

"ShiftRows":

در این لایه، هر ردیف از ماتریس وضعیت (state matrix) در AES به تعداد مشخصی جابجا می‌شود. به طور خاص، ردیف اول بدون تغییر باقی می‌ماند، ردیف دوم یک ستون به چپ جابجا می‌شود، ردیف سوم دو ستون و ردیف چهارم سه ستون به چپ جابجا می‌شوند.

این جابجایی ردیف‌ها باعث می‌شود که بیت‌ها از هر بایت به مکان‌های متفاوتی منتقل شوند و در نتیجه وابستگی بین بیت‌ها از بایت‌های متفاوت افزایش یابد. این امر باعث گسترش تغییرات و عدم تمرکز تغییرات کوچک در یک بخش می‌شود.

"MixColumns":

در این مرحله، هر ستون از ماتریس وضعیت با استفاده از تبدیل‌های ریاضی در میدان (GF) تغییر داده می‌شود. این تبدیل به صورتی است که هر بایت جدید به صورت ترکیبی از تمام بایت‌های همان ستون محاسبه می‌شود.

لایه MixColumns باعث می‌شود که تغییر در یک بایت به تمام بایت‌های ستون‌های مربوطه منتقل شود. این کار باعث می‌شود که تغییرات کوچک در ورودی به سرعت به تمام بایت‌های وضعیت گسترش پیدا کنند.

اهمیت انتشار (Diffusion) در امنیتAES:

انتشار در AES موجب می‌شود که تغییر در یک بیت ورودی به مرور زمان بر کل بیت‌های خروجی تأثیر بگذارد. این ویژگی برای مقاومت در برابر حملات تفاضلی و خطی که به دنبال شناسایی الگوهای پایدار در تغییرات هستند، حیاتی است.

انتشار باعث می‌شود که رابطه بین بیت‌های ورودی و بیت‌های خروجی غیرقابل پیش‌بینی شود، و تحلیلگر نتواند با استفاده از الگوهای ساده به کلید یا اطلاعات اصلی دسترسی پیدا کند.

2)

در الگوریتمAES ، لایه) که به عنوان مرحله "AddRoundKey" نیز شناخته می‌شود یکی از مراحل اصلی است که در آن کلید رمزنگاری به ماتریس وضعیت (state matrix) اضافه می‌شود. این لایه در هر دور از فرآیند رمزنگاری AES اعمال می‌شود و اطمینان حاصل می‌کند که داده‌های ورودی به‌طور مستقیم تحت تأثیر کلید قرار بگیرند.

نحوه عملکرد لایه"Key Addition"

در مرحلهKey Addition ، بیت‌های ماتریس وضعیت با بیت‌های کلید دور (Round Key) به صورت بیتی XOR می‌شوند. این بدان معناست که هر بیت در ماتریس وضعیت با بیت معادل در کلید دور مطابقت داده شده و با عملیات XOR ترکیب می‌شود.

فرض کنیم ماتریس وضعیت (state matrix) به‌صورت S و کلید دور به‌صورت K باشد. عملیات کلید اضافه به صورت زیر انجام می‌شود:

$$S \rightarrow S \oplus K$$

عملیات XOR یک روش ساده و کارآمد برای افزودن کلید به داده‌ها است. با این عمل، هر بیت از داده‌ها به‌صورت مستقیم به بیت معادل در کلید وابسته می‌شود. این وابستگی مستقیم باعث می‌شود که تغییر در هر بیت از کلید تأثیر قابل توجهی در خروجی نهایی داشته باشد.

به دلیل اینکه XOR یک عملیات برگشت‌پذیر است، تنها با کلید درست می‌توان فرآیند رمزگشایی را به درستی انجام داد. این ویژگی تضمین می‌کند که تنها افرادی که به کلید صحیح دسترسی دارند قادر به رمزگشایی متن رمزنگاری‌شده خواهند بود.

با تغییر یک بیت در کلید یا متن اصلی، به دلیل ترکیب مستقیم و غیرقابل پیش‌بینی کلید با داده از طریق XOR، تغییرات قابل توجهی در کل فرآیند رمزنگاری رخ می‌دهد. این اثر بهمن کمک می‌کند که الگوهای پیش‌بینی‌پذیر از بین بروند و رمزنگاری قوی‌تر شود.

3)

درAES ، لایه جایگزینی بایت از یک S-Box استفاده می‌کند که یک جایگزینی غیرخطی برای هر بایت در ماتریس وضعیت ارائه می‌دهد. این غیرخطی بودن برای امنیت AES حیاتی است و به‌عنوان یکی از ویژگی‌های کلیدی طراحی S-Box شناخته می‌شود.

:Cryptanalysis

یک S-Box غیرخطی باعث می‌شود که وابستگی بین بیت‌های ورودی و خروجی پیچیده و غیرقابل پیش‌بینی باشد. این پیچیدگی تحلیل الگوها و روابط میان ورودی و خروجی را برای مهاجمان دشوارتر می‌کند. به‌ویژه،

حملات خطی و تفاضلی که به دنبال الگوهای خطی و تفاضلی در ورودی و خروجی هستند، با یک S-Box غیرخطی ناکارآمد می‌شوند.

با استفاده از یک S-Box غیرخطی، یک تغییر کوچک در بیت‌های ورودی به تغییرات گسترده و غیرقابل پیش‌بینی در خروجی منجر می‌شود. این امر موجب می‌شود که یک تغییر کوچک در ورودی در دورهای بعدی به کل متن رمزنگاری شده منتشر شود.

اگر یک S-Box خطی استفاده شود، وابستگی بین بیت‌های ورودی و خروجی به‌صورت خطی خواهد بود، به این معنا که یک مهاجم می‌تواند از روش‌های تحلیل خطی برای تخمین بخش‌های مختلف کلید استفاده کند. این مسئله باعث می‌شود که مقاومت AES در برابر حملات تحلیل خطی و تفاضلی به شدت کاهش یابد.

یک S-Box خطی نمی‌تواند اثر بهمن را به درستی ایجاد کند. در نتیجه، تغییرات در بیت‌های ورودی به جای انتشار در کل داده، محدود و قابل پیش‌بینی باقی می‌ماند، که می‌تواند به مهاجمان اطلاعاتی درباره ساختار کلید و داده‌های ورودی ارائه دهد.

استفاده از یک S-Box غیرخطی در AES برای جلوگیری از تحلیل‌های ساده و افزایش امنیت رمزنگاری ضروری است. اگر S-Box خطی بود، کل الگوریتم در برابر حملات تحلیل خطی و تفاضلی آسیب‌پذیر می‌شد و انتشار نیز به درستی رخ نمی‌داد، که در نهایت امنیت AES را به شدت کاهش می‌داد.