بسمه تعالی

مبانی رمزنگاری و امنیت شبکه

دکتر میرمحسنی

تمرین سری چهارم

پوریا دادخواه _ ۹۴۵۲۴۲۰۵

۱- می‌دانیم رمز playfair، حروف در جدول 5x5 قرار گرفته و براساس 5 قانون، هر دو حرف متن اصلی را به دو حرف جدید تبدیل می‌کنیم. بنابراین چون قوانین ثابت اند، تعداد کلیدهای ممکن برابر تعداد جدول‌های اولیه است. برای پرکردن این جدول !25 حالت داریم. اما دقت داریم که حالت‌هایی که ستون‌ها یا سطرها چرخش پیدا کند خاصیت مشابه خواهد بود. زیرا با چرخش سطر یا ستون اولاً داده‌های متوالی هیچ تغییری نمی‌کنند. هم چنین چهار حرف متناظر با گوشه‌های یک مستطیل همیشه با یکدیگر تشکیل یک مستطیل می‌دهند. زیرا آن هم سطرها یا ستون‌ها چرخش یافته و حروف‌های هم ستون یا هم سطر بازهم با یکدیگر هم ستون یا هم سطر می‌مانند. برای پرکردن جدول ابتدا یک حرف را به دلخواه در یک خانه قرار می‌دهیم. سپس برای پرکردن بقیه خانه‌ها !24 حالت داریم.

$$No = \frac{lg(24!)}{3.2} = 24.67$$

۲- a) شکل و ظاهر شکل:

(تعداد ماتریس‌های که حداقل یک سطر زوج باشد) = (تعداد ماتریس‌های که یک سطر زوج) − (هردو سطر زوج باشد)

$$\Rightarrow \quad = 2\times13^2\times26^2 - 13^4 = 13^4(8-1) = 7\times13^4$$

b) مشابه قسمت a داریم:

$$= 2\times13^2\times26^2 - 13^4 = 7\times13^4$$

c) تعداد ماتریس‌هایی که همه درایه‌هایش زوج باشد برابر 13^4 است.

d) برای این که تبدیل زوج باشه یا باید هر دو به فرد باشه یا یکی از حالت‌های قسمت a یا b باشه

بنابراین اجتماع حالت a و b باشه ⇐

$$= \underbrace{13^4}_{1} + \underbrace{7\times13^4 + 7\times13^4 - 5\times13^2\times13^2}_{2}$$

و یا مشابها $= \underbrace{13^4}_{1} + \underbrace{7\times13^4}_{\text{تبدیل‌های قسمت a}} + \underbrace{2\times13^2\times13^2}_{\text{تبدیل‌هایی که یک شعاع فرد و دیگری زوج}} = 10\times13^4$

---

e) در بازه موقت یعنی $[0, 25]$ ، اعداد 0 و 13 بر 13 بخش‌پذیرند پس دایره‌ای شعاع اول باید 0 یا 13 باشه. (۲ انتخاب داره هرکدام) دسته دوم دایره‌ها:

$$= 2^2 \times 26^2 = 2^4 \times 13^2$$

---

f) شعاع اول نباید بر 13 بخش‌پذیر باشه پس شعاع اول باید هم‌نهشت 0 یا 13 باشه ⇐ 4 حالت قبل دایره‌ای

نباید از میان 26^2 تا نقاط بیفته. هم‌چنین $ax \pmod{13}$ هم هیچ‌وقت بین 0 تا 13 نداره و برای همین از هر دو عدد دو شعاع اول، شعاع دوم 13 حالت (بین 0 تا 12) قرار نداره. هم‌چنین جمع آن‌ها با 13 نیز آن‌ها را با پیمانه 13 تغییر نخواهد داد و قابل قبول است ⇐ برای هر حالت شعاع اول، شعاع دوم 26 حالت مختلف داره ⇐

$$= (26^2 - 4) \times 26 \times 2 = 2^4 \times (13^2 - 1) \times 13$$

---

g) از جمع دو حالت قبل تعداد حالات خواسته شده به دست می‌آید. (یا شعاع اول نخش‌پذیر است B یا نیست که اولی شمول حالت "e" و دومی "f" بنابراین.)

$$= (26^2 - 4) \times 26 \times 2 + 4 \times 26^2$$

$$= 37648$$

h) در قسمت d دیدیم ۱۰ حالت خواهیم داشت:

oo - oo - ee - eo - oe
oo   ee   oo   eo   oe

ee - ee - eo - oe - ee
eo   oe   ee   ee   ee

زوج e
فرد o

حل شروط e , f را همزمان اعمال می‌کنیم.

(e) ستون اول مضرب 13 در هر کدام از این ۱۰ حالت عدد فرد مضرب 13 خود 13 و عدد زوج ۰ می‌شود

پس در هر کدام از این ۱۰ حالت، برای ستون اول یک انتخاب داریم. برای ستون دوم هم 13^2

پس اشتراک d و e برابر 10×13^2 است *

f) ستون اول نباید مضرب 13 باشد پس از 13^2 حالت، یکی غیرقابل قبول است. یکی از درایه‌ها

ستون دوم را به دلخواه با رعایت نکته قسمت f انتخاب می‌کنیم. این درایه 13 حالت دارد. درایه آخر هم یکتا (از ۲ حالت

ممکن آن با زوج یا فرد بودن یکی انتخاب می‌شود) در کل این حالت $10 \times (13^2-1) \times 13 \times 1$. **

جمع $= (*) + (**) = 23530$

i) از اصل متمم استفاده می‌کنیم.

= کل ترتیب‌ها − (ترتیب‌های بخش‌پذیر بر ۲) (d) − (ترتیب‌های بخش‌پذیر بر 13) (g) + (ترتیب‌های بخش‌پذیر بر 26) (h)

$= 26^4 - 10 \times 13^4 - 37648 + 23530 = 157248$

۳_ بزرگترین عدد در کلید 8 است. حروف کل متن را به ترتیب به دسته‌های 8 تایی تقسیم می‌کنیم و هر یک از اعداد موجود در کلید را به ترتیب به یکی از آن‌ها اختصاص می‌دهیم. از هر یک از این دسته‌های 8 تایی، حرفی که عدد نسبت داده شده به آن را نمایش می‌دهد را انتخاب می‌کنیم.

ادامه صفحه بعد

مثلاً اگر قسمت اول "ithought" باشد و شماره طبد آن 7 باشد حرف h را انتخاب

می‌کنیم و همین صورت ادامه می‌دهیم تا هر 73 حرف پیام رمز شده (متناظر با طول طبد) را بدست آوریم.

برای تشکیل دومین کار برنامه‌ای برای انجام کارهای فوق نوشتیم و در آن ابتدا علامت گذاری و فاصله و...

حذف شده و به ترتیب گفته شده انجام می‌شود. متن بدست آمده به صورت زیر است:

Hesitteth betweenthecherubimstheisleSmaybegladthereofa

Stheriversinthesouth

حال فاصله‌ها را دستی می‌گذاریم =>

He Sitteth between the cherubims. the isles be glad there of

as the rivers in the South.

۴- می‌دانیم که در رمز خانه‌ای تبدیل کلیدی از نوع involution است یعنی برای رمز نمودن

و گرفتن ابتدا دو بخش متن رمز شده را جابجا کنیم و سپس این متن رمز شده را وارد همان سیستم رمز

گذاری می‌کنیم. با این تفاوت که باید طبد‌ها را از آخر به اول (برعکس حالت رمز گذاری) متن رمز شده

اعمال شوند. در این صورت می‌دانیم از همان سیستم رمز گذاری با عمل رمز شدن استفاده کنیم.

• در سیستم رمز گذاری مطرح شده در صورت سوال، در طبدها رابطه $K7=K10$ ، $K6=K11$ ، $K1=K16$

... $K8=K9$ برقرار است و در نتیجه برای رمز گشایی نیازی به برعکس کردن ترتیب طبدها نیست

(زیرا ست کرنر تقارن دارد) یعنی اگر متن رمز شده را (همان encryption oracle) اعمال کنیم، متن پیام اولیه

بدست می‌دهد.

۵- (a) اگر دو کلید k و k' برابر باشند در تمام $(N-t)$ متن اصلی دیگر با یکسان باشند.

برای $(N-t)$ متن اصلی دیگر $(N-t)!$ حالت جایگشت وجود دارد که فقط یک حالت آن مطلوب

است. پس احتمال این که k و k' برابر باشند برابر است با : $\frac{1}{(N-t)!}$

(b) ابتدا تعداد حالت هایی را به دست می آوریم که در بین M متن اصلی معلوم، هیچ کدام از متن اصلی خود نرود

متمم این حالتی می شود که حداقل یکی به متن خود رفته. با شمول و عدم شمول. (A_i : متن اصلی i ام به متن رمز متناظرش رمز شود)

$$N = N_{کل} - \sum N(A_i) + \sum N(A_i \cap A_j) - \dots + (-1)^M \sum N(\bigcap_{i=1}^{M} A_i)$$

$N(A_i) = (M-1)! \rightarrow M$ تا

$N(A_i \cap A_j) = (M-2)! \rightarrow \binom{M}{2}$ تا

$N(\bigcap_{i=1}^{M} A_i) = 1 \rightarrow \binom{M}{M}$ تا

* ادامه سؤال بعد از سؤال 7 *

۶- اساس کار DES رمز فایستل است و می دانیم که در رمز فایستل تبدیل های از نوع involution است

$$h_{i-1} = (m_{i-1}, m_i) \rightarrow h_i = (m_i, m_{i-1} + F_{k_i}(m_i)) = (m_i, m_{i+1})$$

$$m_{i+1} = m_{i-1} + F_{k_i}(m_i)$$

$$m_{i-1} = m_{i+1} + F_{k_i}(m_i)$$

این رابطه علی رغم پیچیدگی F در DES همچنان برقرار است یعنی در عملیات رمزگشایی این تابع حذف می شود

(هر دور از نوع involution است.) یعنی کافیست در رمزگشایی جای دو بخش خروجی را عوض کنیم و به ورودی

یک دور با همان کلید متناظر اعمال کنیم و در خروجی، ورودی اولیه دور رمزگذاری را خواهیم داشت:

Encryption: $\begin{cases} R_{i+1} = L_i \oplus f(R_i, k_i) \\ L_{i+1} = R_i \end{cases}$

Decryption: $\begin{cases} R_i' = L_{i+1} = R_i \\ L_i' = R_{i+1} \oplus f(L_{i+1}, k_i) = L_i \oplus (f(R_i, k_i) \oplus f(R_i, k_i)) = L_i \end{cases}$

150

برای رمزنگاری کافیست ابتدا جایگشت IP^{-1} را به پیام رمز شده اعمال کنیم. سپس دور آخر را در جهت برعکس (کلید آخر) قرار دهیم و در نهایت جایگشت IP را بر آن اعمال کنیم تا در خروجی همان رمز اولیه را داشته باشیم.

۷- a) اساس کار DES رمز فایستل است. ابتدا ثابت می‌کنیم با NOT کردن ورودی و طبق در رمز فایستلی خروجی نیز NOT می‌شود. می‌دانیم که در رمز فایستلی رابطه ورودی و خروجی هر دور به صورت زیر است:

$$\begin{cases} R_i = L_0 \oplus f(R_0, k_0) \\ L_i = R_0 \end{cases}$$

عملیات‌های جایگشتی (Permutation) و شیفت دوری طبق و از این قبیل (Expansion Permutation) هستند که بر بیت‌ها عمل می‌کنند و اگر بیت در ورودی NOT شود، همان بیت در خروجی هم‌تنظیر از ورودی همیشه به همین شکل عمل می‌کند و اگر بیت در ورودی NOT شود، همان بیت در خروجی NOT خواهد شد. البته این برای S-box درست نیست. (زیرا در S-box اگر یک بیت تغییر کند حداقل دو بیت در خروجی تغییر می‌کند.)

$$\bar{A} = 1 \oplus A$$

با توجه به قواعد منطقی داریم:

$$\Rightarrow \bar{A} \oplus \bar{B} = 1 \oplus A \oplus 1 \oplus B = (1 \oplus 1) \oplus (A \oplus B) = A \oplus B$$

$$\overline{A \oplus B} = 1 \oplus (A \oplus B) = \overline{(A \oplus B)}$$

اگر فرض کنیم طبق ورودی NOT شوند و با توجه به اینکه تمامی عملگر f بیت عملگر bit-wise است:

$$L_1 = \overline{R_0}$$

$$f(\overline{R_0}, \overline{k_0}) = f(\overline{R_0} \oplus \overline{k_0}) = f(R_0 \oplus k_0)$$

$$R_1 = \overline{L_0} \oplus f(R_0, k_0) = \overline{L_0 \oplus f(R_0, k_0)} = \overline{R_1}$$

ادامه ضمیمه

نتیجه می‌شود که اگر ترکیب بلوک ورودی و کلید NOT شوند، خروجی نیز NOT می‌شود.

این مراحل 16 بار تکرار می‌شود و در نتیجه خروجی نهایی NOT خروجی اولیه خواهد بود.

بلوک کلی (Initial permutation(IP و IP^{-1} نیز همانطور که گفته شد تنها تغییر مکان بیت‌ها را انجام می‌دهند و تأثیری در عملیات کلی رمزنگاری ندارند. در نتیجه اگر ورودی و کلید NOT شوند، خروجی DES نیز NOT خواهد شد.

b) در حمله جستجوی فراگیر اگر دو جفت (C_1, P_1) و (C_2, P_2) را داشته باشیم به طوری که $P_1 = \bar{P}_2$ و $C_1 = \bar{C}_2$ (یعنی مثلاً کلید MSB آن‌ها است را دور می‌زنیم) اگر کلید K روی P_1 نتیجه C_1 را بدهد احتمالاً این کلید همان کلید استفاده شده در رمزنگاری است. حال اگر این کلید روی P_1 نتیجه $C_2 = \bar{C}_1$ را بدهد، احتمالاً کلید $\bar{K}$ کلید استفاده‌شده در رمزنگاری است. در نتیجه فضای کلید عملاً نصف می‌شود.

ادامه سؤال 15 ←

$$\Rightarrow N = M! - M(M-1)! + \binom{M}{2}(M-2)! - \cdots + (-1)^M \binom{M}{M}$$

$$= \sum_{i=0}^{M} (-1)^i (M-i)! \binom{M}{i} = \sum_{0}^{M} (-1)^i \frac{M!}{i!} = M! \sum_{0}^{M} \frac{(-1)^i}{i!}$$

حال به مسئله اصلی می‌پردازیم؛ ابتدا t' تا از P_i ها را انتخاب می‌کنیم تا برقرار شرط مسئله یعنی P_i با C_i برابر باشند. $\binom{N-t}{t'}$ تعداد حالت‌هایی که $N-t-t'$ رقم اضافی باقی‌مانده هیچ‌کدام همزمان برابر نشوند.

برابر است با: $\binom{N-t}{t'} \times (N-t-t')! \sum_{0}^{N-t-t'} \frac{(-1)^i}{i!}$ تعداد حالت‌هایی که t' تا صحیح باشند

$$\Rightarrow P_r(t' \text{ تا صحیح}) = \frac{\binom{N-t}{t'} \times (N-t-t')! \sum_{0}^{N-t-t'} \frac{(-1)^i}{i!}}{(N-t)!}$$

$$\Rightarrow P_r = \frac{1}{t'!} \times \sum_{i=0}^{N-t-t'} \frac{(-1)^i}{i!}$$