

به نام خدا



«امنیت داده و شبکه»

دکتر امینی

تمرین اول

علیرضا دهقانپور فراشاه

۹۸۱۰۱۵۵۵

فهرست مطالب

فهرست مطالب.....	1
سوال اول.....	2
سوال دوم.....	4
سوال سوم.....	8
سوال چهارم.....	13
سوال پنجم.....	14
سوال ششم.....	16
سوال هفتم.....	18

سوال اول

TCP sequence prediction attack

این حمله بدین صورت است که مهاجم سعی می‌کند شماره دنباله‌ی بسته‌های tcp را حدس بزند و بدین صورت بتواند بسته‌های جعلی درست کند و خود را جای نود فرستنده جا بزند. مهاجم سعی می‌کند با شنود بسته‌ها میان دو نود IP درست و Sequence number را حدس بزند و با این دو مورد بسته‌های جعلی درست کند. برای جلوگیری از زودتر رسیدن بسته‌های نود فرستنده مهاجم می‌تواند یک جمله DoS نیز به فرستنده بزند تا دیگر نتواند بسته‌ای برای گیرنده ارسال کند. در این حمله اصل صحت منبع و صحت داده نقض می‌شود.

راهکار مقابله

- استفاده از دیواری آتش به گونه‌ای که اجازه ندهیم بسته‌ای از درون شبکه با یک IP از خارج شبکه به مقصد برسد.
- استفاده از یک بستر امن مانند TLS یا SSL
- استفاده از اطلاعات لایه‌های پایین‌تر بگونه‌ای که بتوان بسته‌های جعلی را از باقی بسته‌ها تمیز داد.

Reflector attack

در این حمله فرد مهاجم نیاز به جعل IP و تعدادی سرور که در اصطلاح به آن‌ها reflector server می‌گویند دارد. همچنین فرد مورد حمله victim نامیده می‌شود. این حمله بدین صورت است که حمله‌کننده با جعل IP یک درخواست UDP به تعدادی reflector server می‌فرستد و بدین صورت این سرورها به IP جعلی پاسخ را می‌فرستند و به این روش حجم زیادی داده به سمت victim ارسال می‌شود. این حمله اصل دسترس‌پذیری را هدف قرار می‌دهد.

راهکار مقابله

- اگر از یک سرور استفاده می‌شود برای حمله سرورها می‌توانند بررسی کنند که ناگهان حجم زیادی درخواست از یک IP دریافت نکنند.
- استفاده از دیواری آتش و قرار دادن تعدادی قانون (Rule) مثلاً برای کنترل درخواست‌های DNS

Masquerading attack

این نوع حمله تمامی حملاتی را شامل می‌شود که یک فرد با جعل یا سرقت اطلاعات، امضای دیجیتال و ... یک فرد دیگر سعی دارد یک سامانه را فریب دهد یا با نام فرد دیگری کار خطایی را انجام دهد. در این حمله فرد مهاجم با هویت یک فردی که دسترسی‌های بالاتری دارد وارد یک سیستم می‌شود و از دسترسی‌های فرد دیگر استفاده می‌کند. در این حمله اصل صحت و محرمانگی نقض می‌شود.

راهکار مقابله

- بررسی‌های رفتاری به صورتی که اگر فرد رفتاری غیرقابل انتظار داشت بتوان تشخیص داد.
- می‌توان ورودها و login کردن‌های یک فرد را مورد بررسی قرار داد و اگر ناگهان فردی از یک IP جدید یا از یک منطقه‌ی دیگر وارد شد آن را متوجه شد.
- استفاده از راهکارهای پیچیده‌تر Authentication.

Fault Injection attack

این حمله از دسته حملات active است. در این حمله مهاجم با نفوذ فیزیکی و تغییر مثلاً ولتاژ یا فرکانس پردازنده سبب ایجاد خطا می‌شود. همچنین به صورت نرم‌افزاری می‌تواند نفوذ کند و از Robust نبودن کد استفاده کند و مثلاً باعث پر شدن بافر حافظه یا overflow شود. در این حمله اصل دسترس‌پذیری نقض می‌شود. البته می‌توان گفت اگر خطای ایجاد شده سبب تغییر داده شود اصل صحت نیز نقض می‌شود.

راهکار مقابله

- اعمال تست‌های آسیب‌پذیری نرم‌افزاری و بررسی آسیب‌های کد.
- کنترل سخت‌افزار سیستم و جلوگیری از تغییر پارامترهای کاری آن.
- جلوگیری از دسترسی‌های زیاد به کاربران یک سامانه یا دیتابیس.
- استفاده از مفاهیمی مانند Virtual Page Address و جلوگیری از پیش‌بینی پذیری در حافظه و سیستم.

Man in the middle attack

در این حمله یک فرد در میان ارتباط دو نود در شبکه قرار می‌گیرد و به گونه‌ای هر دو نود را فریب می‌دهد بدین صورت که با دریافت بسته از فرستنده و مشاهده آن و تغییر آن یک بسته جدید را به گیرنده می‌دهد به گونه‌ای که گیرنده متوجه وجود مرد میانی نشود و همچنین پاسخ گیرنده را به گونه‌ای تغییر می‌دهد که فرستنده متوجه تغییر درخواستش نشود. در این حمله هر سه اصل می‌تواند نقض شود یعنی محرمانگی و صحت و دسترس‌پذیری.

راهکار مقابله

- استفاده از رمزنگاری نامتقارن
- استفاده از یک شخص ثالث مورد اعتماد برای توزیع کلید متقارن
- استفاده از گواهی دیجیتال

منابع:

https://en.wikipedia.org/wiki/TCP_sequence_prediction_attack

<https://networkengineering.stackexchange.com/questions/68346/prevent-tcp-sequence-prediction-attack>

<https://www.rfc-editor.org/rfc/rfc6528>

<https://www.a10networks.com/blog/how-defend-against-amplified-reflection-ddos-attacks/>

<https://www.tutorialspoint.com/the-reflection-attack>

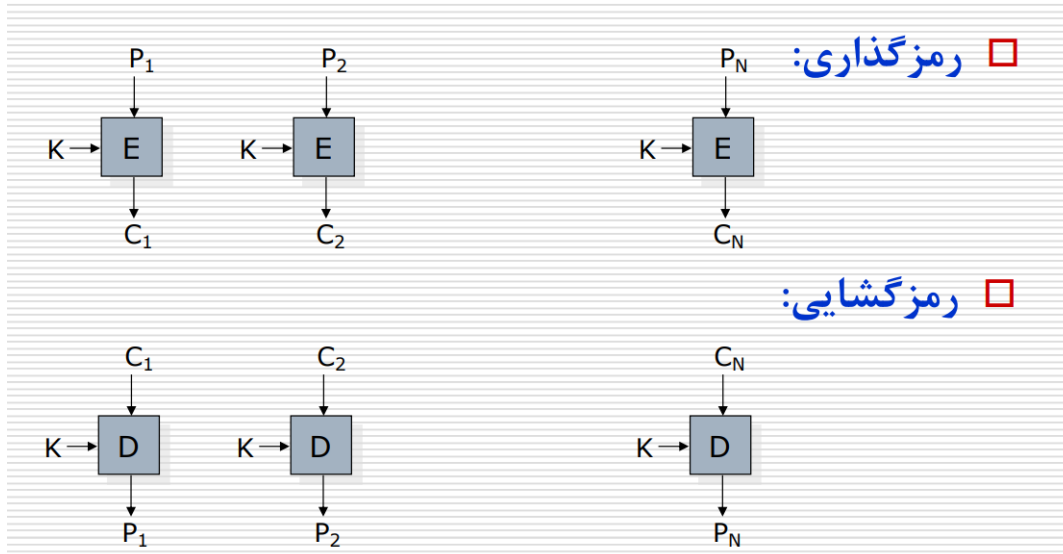
<https://www.netscout.com/what-is-ddos/what-is-reflection-amplification-attack>

<https://www.socinvestigation.com/masquerade-attack-everything-you-need-to-know-in-2022/>

<https://seon.io/resources/dictionary/masquerade-attack/>

https://en.wikipedia.org/wiki/Fault_injection

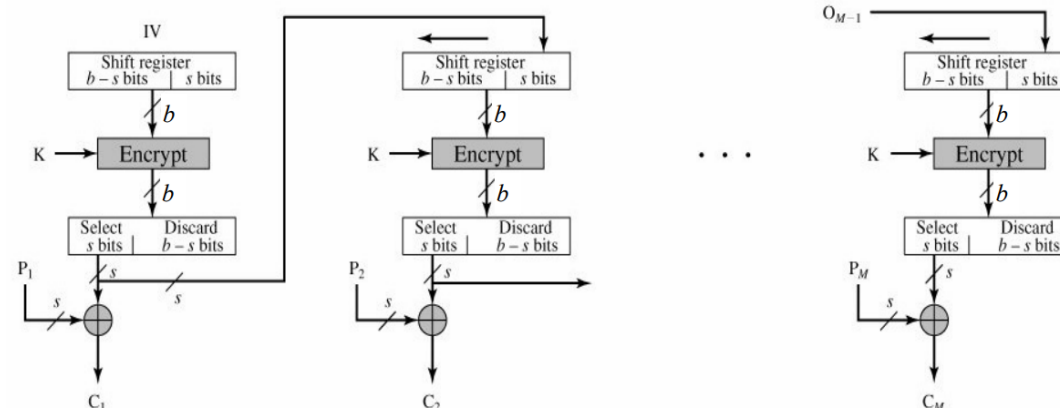
ECB •



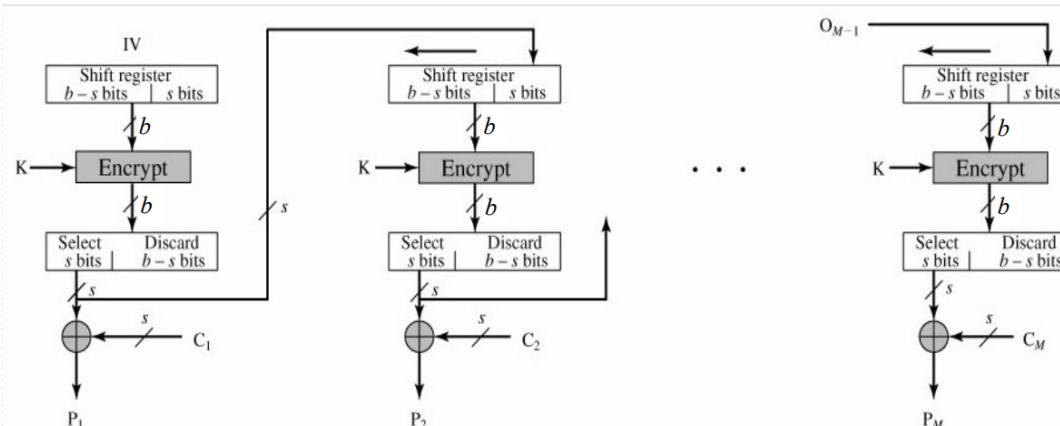
الف) رمزگشایی برای همان بلوک که در آن یک بیت دچار خطا شده است دارای ایراد خواهد بود و برای سایر بلوکها مشکلی رخ نخواهد داد.

- ب) به همان دلیل قسمت قبل فقط رمزگشایی همان بلوک دچار مشکل خواهد شد و سایر بلوکها قابل رمزگشایی هستند.
- پ) در رمزگذاری وابستگی میان بلوکها وجود ندارد و بنابراین می توان موازی رمزگذاری کرد.
- ت) در رمزگشایی وابستگی میان بلوکها وجود ندارد و بنابراین می توان موازی رمزگشایی کرد.
- ث) این امکان وجود دارد زیرا هر بلوک کاملاً مستقل از بلوک دیگری رمز و رمزگشایی می شود.

رمزگذاری □



رمزگشایی □

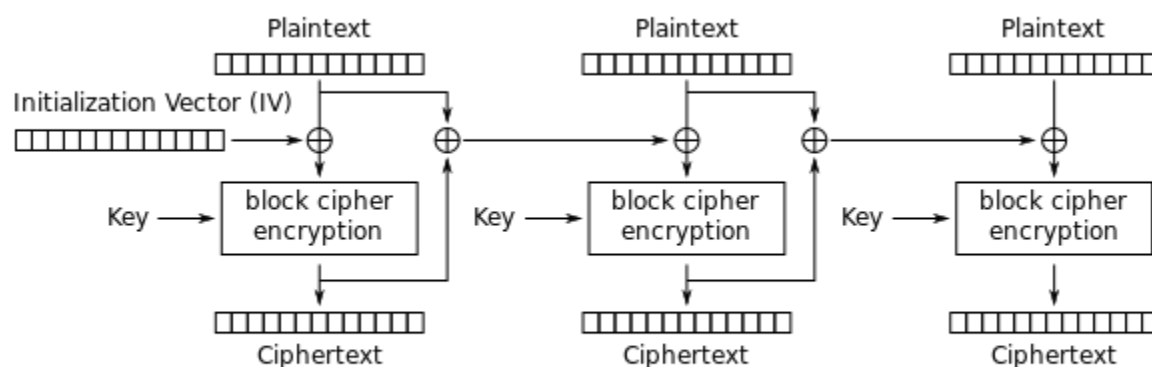


الف) رمزگشایی برای همان بلوک که در آن یک بیت دچار خطا شده است دارای ایراد خواهد بود و برای سایر بلوکها مشکلی رخ نخواهد داد. زیرا در این مد C_i فقط در P_i تاثیر خواهد داشت.

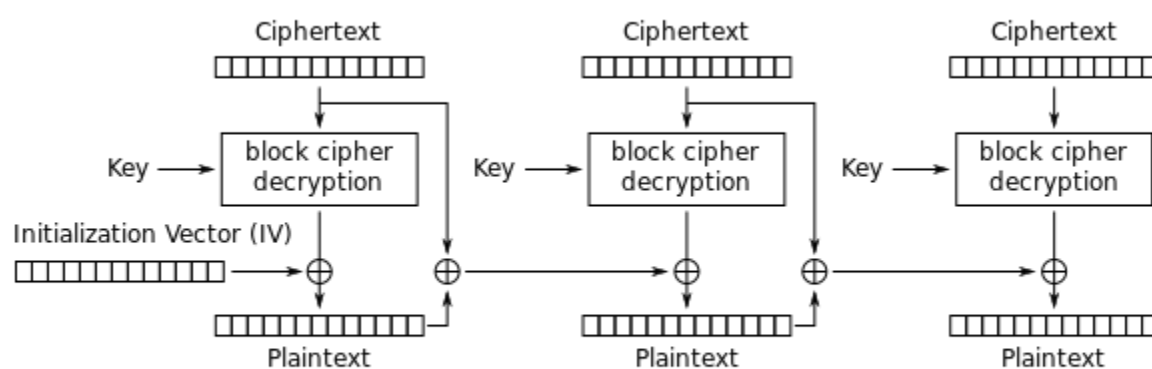
ب) به همان دلیل قسمت قبل فقط رمزگشایی همان بلوک دچار مشکل خواهد شد و سایر بلوکها قابل رمزگشایی هستند. پ) در رمزگذاری وابستگی میان بلوکها وجود دارد و بنابراین نمیتوان موازی رمزگذاری کرد. البته با این فرض که با داشتن Initialization Vector (IV) عملیات به دست آوردن s بیت هر مرحله را انجام داده باشیم میتوان به صورت موازی رمز کرد ولی در صورت تولید IV همراه با آمدن Plaintext این امکان میسر نیست.

ت) دقیقا مانند بخش قبل است یعنی اگر IV را داشته باشیم میتوان s بیت مورد نیاز هر مرحله را از قبل تولید کنیم و با آمدن Ciphertext آن را رمزگشایی کنیم.

ث) این امکان وجود دارد در صورتی که IV را داشته باشیم میتوان s بیت مورد نیاز هر مرحله را از قبل تولید کنیم و با آمدن Ciphertext بلاک مورد نظر را رمزگشایی کنیم.



Propagating Cipher Block Chaining (PCBC) mode encryption



Propagating Cipher Block Chaining (PCBC) mode decryption

الف) در اینجا همانطور که مشخص است تغییر یک بیت از ciphertext باعث خرابی plaintext همان بلوک می شود ولی از این plaintext در بلوک بعدی برای رمزگشایی استفاده می شود بنابراین تغییر این بیت سبب انتشار خطا در تمامی بلوک های بعدی می شود.

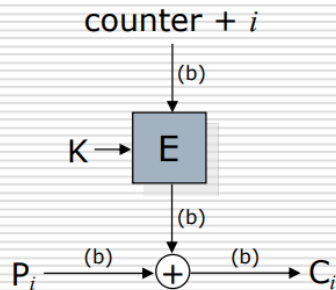
ب) به دلیل وابستگی به متن رمزگشایی شده بلوک قبلی تمامی بلوک های بعد از بلوک از دست رفته دیگر قابل رمزگشایی نیستند.

پ) به دلیل وابستگی های توضیح داده شده این امکان میسر نیست.

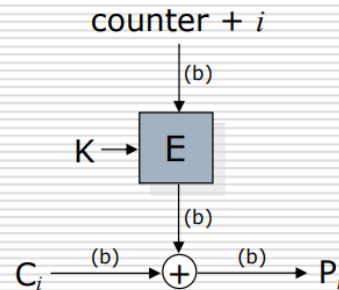
ت) به دلیل وابستگی های توضیح داده شده این امکان میسر نیست.

ث) نیاز است تمامی بلوک های قبلی رمزگشایی شوند.

رمز گذاری □ ↓



رمز گشایی □ ↓



الف) در اینجا تغییر یک بیت در ciphertext سبب خطا در plaintext همان بلاک می‌شود و بلوک‌ها نسبت به هم مستقل هستند بنابراین سایر بلوک‌ها دچار خطا نمی‌شوند.

ب) بلوک‌ها به یکدیگر وابستگی ندارند بنابراین از دست دادن یک بلوک تاثیری در رمزگشایی سایر بلوک‌ها ندارد.

پ) می‌توان بلوک‌ها را به صورت موازی رمز کرد زیرا i و counter را داریم و هر بلوک مستقل از دیگری است.

ت) فرآیند رمزگذاری هم مانند رمزنگاری است و بنابراین می‌توان به صورت موازی رمزگشایی کرد.

ث) با توجه به مستقل بودن هر بلوک برای رمزنگاری و رمزگشایی می‌توان هر بلوک دلخواه را با داشتن counter و index رمزگشایی کرد.

منابع

https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation

سوال سوم

(الف)

این رمز جزو رمزهای جانشینی چند الفبایی است. زیرا که در این الگوریتم با استفاده از یک ماتریس 5×5 در هر دو حرف جایگزین دو حرف دیگر می‌شوند و از رمزنگاری‌های تک الفبایی پیچیده‌تر است.

(ب)

• مزایا

- شکستن آن سخت‌تر است زیرا که به جای ۲۵ حرف گوی 25×25 حرف داریم و بدین ترتیب تحلیل فرکانسی نیز سخت‌تر خواهد بود زیرا که به متن بیشتری نیاز است.
- رمزنگاری و رمزگشایی آن الگوریتم پیچیده‌ای ندارد و از این رو به راحتی قابل پیاده‌سازی است.

• معایب

- از معایب آن می‌توان به این اشاره کرد که اگر مثلاً AB به XY رمز شود آنگاه BA به YX رمز می‌شود.
- کاراکترهایی مانند Space و علائم نگارشی و اعداد قابل رمز نیستند.

(پ)

ساختن ماتریس: این ماتریس به وسیله‌ی حروف یکتای کلید پر می‌شود و ادامه‌ی آن حروف الفبا به ترتیب قرار می‌گیرند. در اینجا قابل ذکر است که یک حرف الفبا حذف می‌شود و معمولاً حذف می‌شود و سپس باقی حروف در جدول 5×5 قرار می‌گیرند.

الگوریتم رمز: ابتدا هر دوتا دوتا حروف plaintext را جدا می‌کنیم. اگر یک حرف تکرار شده بود باید حرف اول را برداشت و یک حرف ساختگی به آن افزود و سپس حرف دوم را با باقی متن جدا کنیم. برای مثال کلمه‌ی hello داریم he, lx, lo که x همان حرف ساختگی است. اگر یک حرف در انتها باقی ماند نیز به آن یک حرف جعلی می‌افزاییم.

حال هر دو حرف را به سه حالت زیر رمز می‌کنیم:

1. اگر هر دو حرف روی یک سطر از ماتریس بودند از حرف بعدی هر کدام در همان سطر برای رمز استفاده می‌کنیم.
2. اگر هر دو حرف در یک ستون از ماتریس بودند از حرف بعدی هر کدام در همان ستون برای رمز استفاده می‌کنیم.
3. اگر هیچ کدام از دو حالت بالا نبود با این دو حرف یک مربع می‌سازیم و گوشه‌های مستطیل را برای رمز انتخاب می‌کنیم.

رمزگشایی: در اینجا دقیقاً با همان ماتریس به مانند رمزنگاری عمل می‌کنیم:

1. اگر هر دو حرف روی یک سطر از ماتریس بودند از حرف قبلی هر کدام در همان سطر برای رمزگشایی استفاده می‌کنیم.
2. اگر هر دو حرف در یک ستون از ماتریس بودند از حرف قبلی هر کدام در همان ستون برای رمزگشایی استفاده می‌کنیم.
3. اگر هیچ کدام از دو حالت بالا نبود با این دو حرف یک مربع می‌سازیم و گوشه‌های مستطیل را برای رمزگشایی انتخاب می‌کنیم.

ت)

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

V	I	C	T	O
R	Y	A	B	D
E	F	G	H	K
L	M	N	P	Q
S	U	W	X	Z

DI SY GL YO WL GR YD LV SY GR DT XI CO = YO UR EN DI SN EA RB ES UR EA BO UT IT

متن آشکار: YOUR END IS NEAR BE SURE ABOUT IT

(ث)

برای حملات فرکانسی به این روش رمزنگاری می‌بایست فرکانس تمامی ۲۵ در ۲۵ جفت حروف را بررسی کرد و متوجه شد که هر کدام به کدام مپ شده‌اند. ولی می‌توان با استفاده از آن عیب که در این روش مطرح شد فضای حالت را کوچک‌تر کرد یعنی جفت AB با جفت BA به یک جفت معکوس نسبت داده می‌شوند. البته قابل ذکر است که حملات فرکانسی روی این روش به مراتب سخت‌تر از روش‌های تک الفبایی است.

منابع

<https://www.geeksforgeeks.org/playfair-cipher-with-examples/>

سوال چهارم

$$C = P \oplus K$$

$$C' = P \oplus K' = \neg (P \oplus K) = \neg C$$

بنابراین دو عبارت رمزشده نقیض یکدیگرند ولی بدون داشتن کلید نمی‌توان پیام را رمزگشایی کرد و هیچ استنتاجی نمی‌توان در مورد پیام داشت.

سوال پنجم

خاصیت بهمنی اکید: طبق تعریف می‌گوییم یک الگوریتم رمز خاصیت بهمنی اکید دارد اگر تغییر یک بیت از ورودی سبب شود هر بیت از خروجی به احتمال ۵۰ درصد تغییر کند.

خاصیت تمامیت: می‌گوییم یک تابع بولین کامل است اگر هر بیت خروجی تابعی از تمامی بیت‌های ورودی باشد.

Random Cipher: این مدل به این صورت است که یک تابع تصادفی است که متن ورودی را به یک ciphertext مپ می‌کند و احتمال مپ شدن هر ciphertext به یک متن یکسان است و خروجی از برد تابع به صورت یکنواخت انتخاب شده است.

در این مدل داده شده بعد از یک مرحله اجرا هیچ یک از خواص را نداریم زیرا بخش‌هایی از خروجی هیچ وابستگی‌ای به برخی از بیت‌های ورودی ندارند و به طور واضح خاصیت بهمنی اکید و تمامیت برقرار نیست.

اگر چهار مرحله این الگوریتم رمز را اجرا کنیم خواهیم داشت:

$$y_0 = x_0 \oplus F(x_1), y_1 = x_1 \oplus F(x_2), y_2 = x_2 \oplus F(x_3), y_3 = x_3 \oplus F(F(x_1) \oplus x_0)$$

پس از اجرای ۴ مرحله‌ی دیگر خواهیم داشت:

$$y_0 = x_0 \oplus F(x_1) \oplus F(x_1 \oplus F(x_2))$$

$$y_1 = x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3))$$

$$y_2 = x_2 \oplus F(x_3) \oplus F(x_3 \oplus F(F(x_1) \oplus x_0))$$

$$y_3 = x_3 \oplus F(F(x_1) \oplus x_0) \oplus F((x_0 \oplus F(x_1)) \oplus F(x_1 \oplus F(x_2)))$$

با اجرای دو مرحله‌ی دیگر می‌رسیم به:

$$y_0 = x_2 \oplus F(x_3) \oplus F(x_3 \oplus F(F(x_1) \oplus x_0))$$

$$y_1 = x_3 \oplus F(F(x_1) \oplus x_0) \oplus F((x_0 \oplus F(x_1)) \oplus F(x_1 \oplus F(x_2)))$$

$$y_2 = (x_0 \oplus F(x_1) \oplus F(x_1 \oplus F(x_2))) \oplus F(x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3)))$$

$$y_3 = (x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3))) \oplus F(x_2 \oplus F(x_3) \oplus F(x_3 \oplus F(F(x_1) \oplus x_0)))$$

توجه کنید که با ده مرحله خاصیت تمامیت را ندارد زیرا مثلاً اگر یک بیت از $2x$ تغییر کند فقط در بیت متناظر در $0y$ تغییر ایجاد می‌شود و یعنی هر بیت $0y$ تابعی از تمامی بیت‌های $2x$ نیست و فقط تابعی از بیت‌های متناظر است.

در مورد خاصیت بهمنی اکید توجه کنید که با اجرای ده مرحله این خاصیت ایجاد نشده است زیرا برای مثال در $0y$ اگر یک بیت $2x$ تغییر کند حتماً همان بیت متناظر در $0y$ تغییر می‌کند و با احتمال یک این اتفاق می‌افتد.

اگر سه مرحله‌ی دیگر این دور را اجرا کنیم یعنی در کل ۱۳ مرحله به روابط زیر می‌رسیم.

$$y_0 = x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3)) \oplus F(x_2 \oplus F(x_3) \oplus F(x_3 \oplus F(x_0 \oplus F(x_1))))$$

$$y_1 = x_2 \oplus F(x_3) \oplus F(x_3 \oplus F(x_0 \oplus F(x_1))) \oplus F(x_3 \oplus F(x_0 \oplus F(x_1)) \oplus F(x_0 \oplus F(x_1) \oplus F(x_1 \oplus F(x_2))))$$

$$y_2 = x_3 \oplus F(x_0 \oplus F(x_1)) \oplus F(x_0 \oplus F(x_1) \oplus F(x_1 \oplus F(x_2))) \oplus F(x_0 \oplus F(x_1) \oplus F(x_1 \oplus F(x_2)) \oplus F(x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3))))$$

$$y_3 = x_0 \oplus F(x_1) \oplus F(x_1 \oplus F(x_2)) \oplus F(x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3))) \oplus F(x_1 \oplus F(x_2) \oplus F(x_2 \oplus F(x_3)) \oplus F(x_2 \oplus F(x_3) \oplus F(x_3 \oplus F(x_0 \oplus F(x_1))))$$

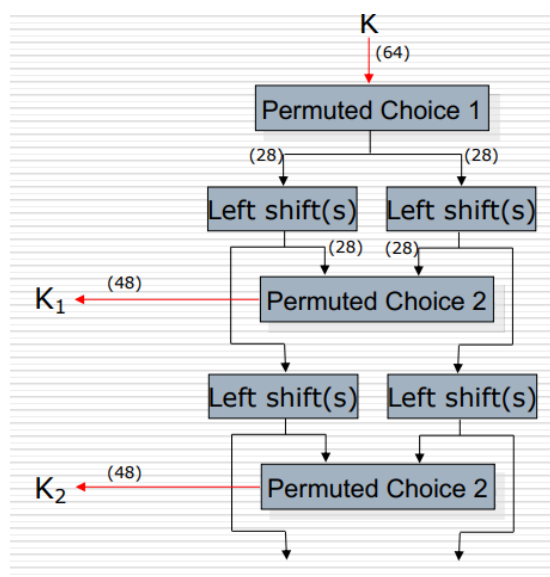
که همانطور که مشخص است اگر F خاصیت تمامیت را دارا باشد این ساختار نیز خاصیت تمامیت را دارد. با فرض این که تابع F خاصیت بهمنی اکید را داراست اگر هر بیت از ورودی تغییر کند چون حداقل یک بار در F آمده است آنگاه تمامی بیت‌های خروجی با احتمال ۵۰ درصد ممکن است تغییر کنند.

برای خاصیت random cipher اجرای چهار دور کافی و ضروری است زیرا کمتر از چهار دور باعث می‌شود خروجی تابعی مستقیم از ورودی باشد و به صورت یکنواخت متن رمز تولید نشود. اما اجرای چهار دور از الگوریتم کافی است زیرا که اگر F با تغییر یک بیت ورودی به صورت تصادفی یک متن رمز دیگر تولید کند ترکیبات مختلف آن و xor آن با ورودی نیز این ویژگی را حفظ می‌کند. علت حفظ این ویژگی نیز این است که F به صورت تصادفی خروجی تولید می‌کند و xor با بخش‌هایی از ورودی تاثیری در تصادفی بودن خروجی ندارد.

در واقع می‌توان به این صورت گفت اگر $3y$ را داشته باشیم چون تابع F این ویژگی را دارد نمی‌توانیم استنباطی در مورد $0x$ انجام دهیم و چون با هر دور خروجی‌ها شیف‌ت می‌خورند $3y$ مرحله فعلی $2y$ مرحله بعد است بنابراین با 4 مرحله به حالتی می‌رسیم که نتوان در مورد ورودی استنتاجی انجام داد.

سوال ششم

در رمزنگاری DES که براساس ساختار رمزهای فیستل است طول کلید ۵۶ بیت است و از آن با استفاده از یک key scheduler برای هر دور که در مجموع ۱۶ دور داریم یک کلید بدست می آوریم. طول کلید هر دور ۴۸ بیت است. برای کلید اصلی که ۵۶ بیت است از ۶۴ بیت استفاده می کنیم و بیت های (8, 16, 24, 32, 40, 48, 56, 64) را صرف نظر می کنیم و از آن ها می توان به عنوان parity bit استفاده کرد. به این عملیات Permuted Choice 1 یا به اختصار 1-PC گفته می شود. بعد از این مرحله کلید بدست آمده که ۵۶ بیت است را به دو قسمت ۲۸ بیتی تقسیم می کنیم و هر کدام از این بخش ها را به تعداد ۱ یا ۲ بار شیفت چپ می دهیم و در انتها روی آن عملیات Permuted Choice 2 که به اختصار 2-PC گفته می شود را اعمال می کنیم و به یک کلید ۴۸ بیتی می رسیم. این روند را در تصویر زیر مشاهده می کنید.



در این الگوریتم اگر کلید ۵۶ بیتی از کلیدهای مقابل باشد این الگوریتم دچار ایراد خواهد شد و به کلیدهای مقابل کلید ضعیف می گویند.

- 0x 11111111 11111111
- 0x 00000000 00000000
- 0x 11111111 00000000
- 0x 00000000 11111111

ایراد این کلیدها این است که در هنگام شیفت خوردن هر نیمه ثابت می مانند یعنی هر نیمه ی ۲۸ بیتی یا تماماً ۰ یا تماماً ۱ است که سبب می شود عملیات شیفت بی معنی شود. بدین ترتیب کلید هر دور یکسان می شود یعنی ۱۶ کلید ۴۸ بیتی یکسان. از آنجایی که در ساختار رمزهای فیستل رمزگشایی همان عملیات رمزنگاری است اگر کلید هر دور یکسان باشد پس از اعمال هر زوج دور به plaintext می رسیم.

PC-1

<i>Left</i>							<i>Right</i>						
57	49	41	33	25	17	9	63	55	47	39	31	23	15
1	58	50	42	34	26	18	7	62	54	46	38	30	22
10	2	59	51	43	35	27	14	6	61	53	45	37	29
19	11	3	60	52	44	36	21	13	5	28	20	12	4

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

منابع

https://en.wikipedia.org/wiki/Weak_key#Weak_keys_in_DES

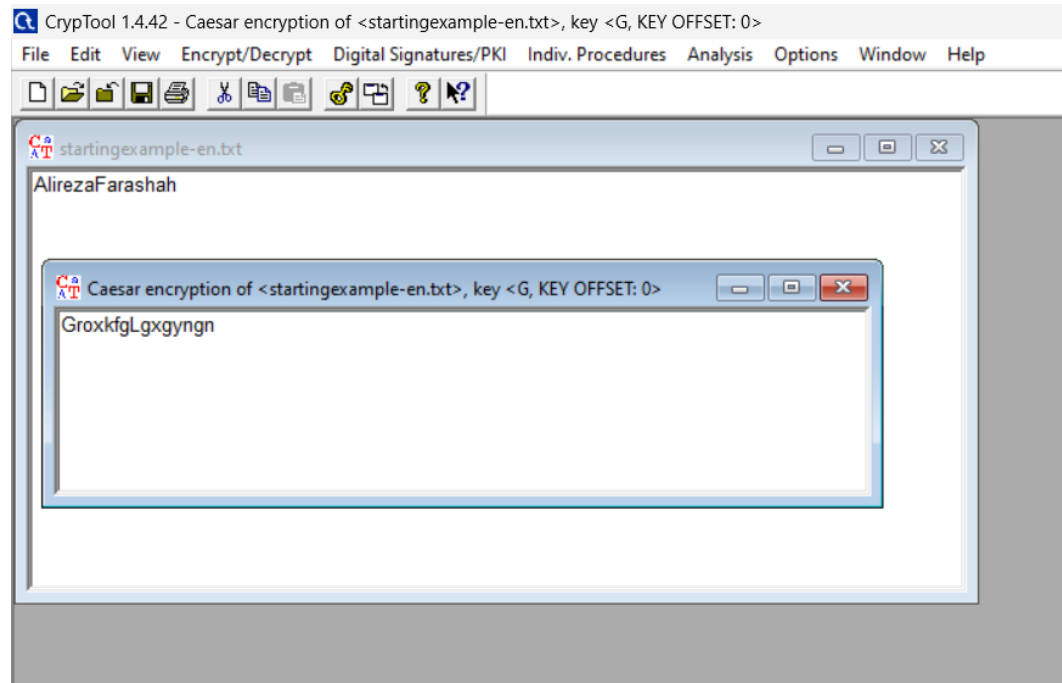
https://en.wikipedia.org/wiki/Key_schedule

[https://en.wikipedia.org/wiki/DES_supplementary_material#Permutation_\(P\)](https://en.wikipedia.org/wiki/DES_supplementary_material#Permutation_(P))

<https://www.tutorialspoint.com/what-are-the-weaknesses-of-data-encryption-standard>

سوال هفتم

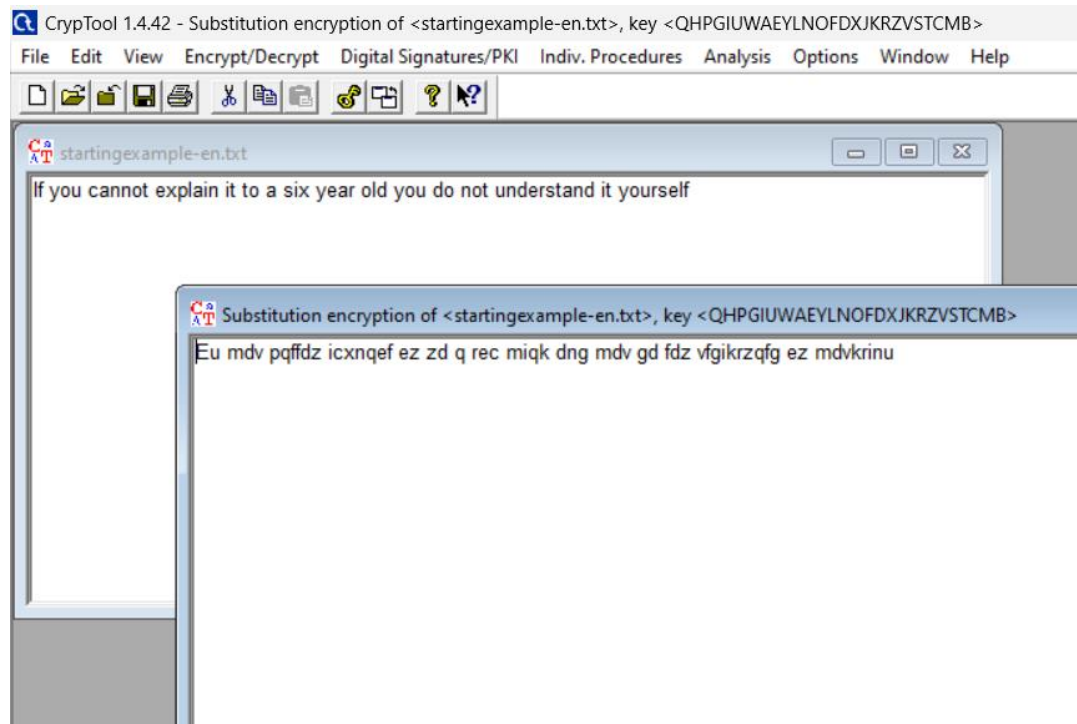
(الف)



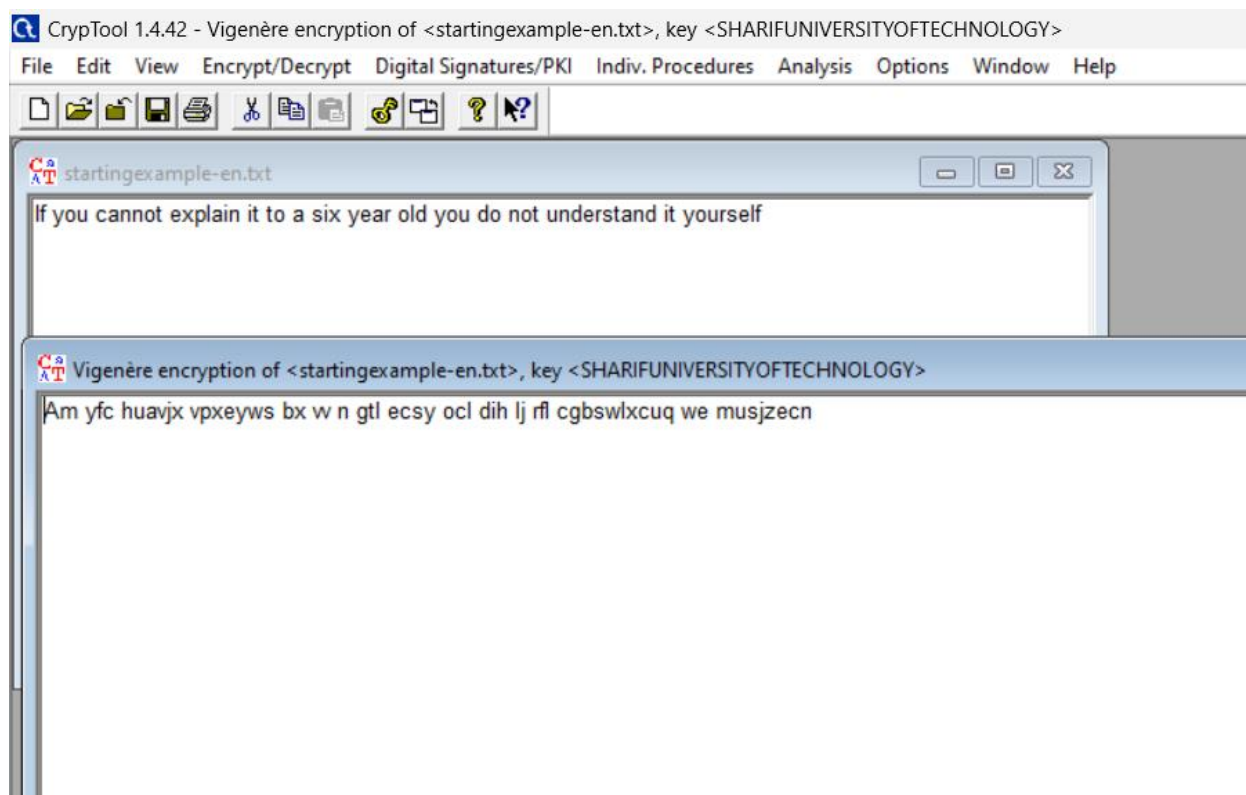
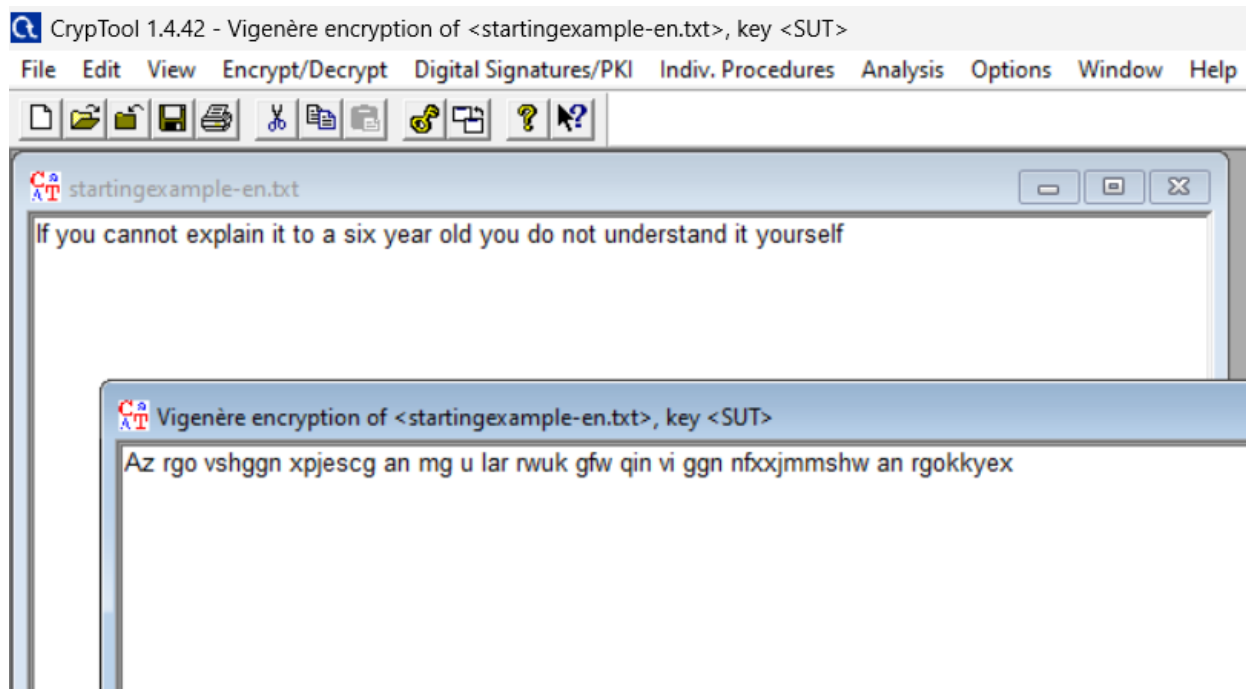
متن: AlirezaFarashah

رمز: GroxkfgLgxgyngn

(ب)



(پ)



هرچه طول کلید بیشتر باشد یافتن طول کلید برای رمزگشایی کار سخت‌تری خواهد بود برای مثال در دو حالت بالا در حالت اول دوبار کلمه you به rgo مپ شده‌است ولی در حالتی که طول کلید بیشتر است این اتفاق کمتر رخ می‌دهد.

ت)
رمز:

Key Entry: Playfair

Options

☒ Separate duplicate letters

First separator:


Second separator:

☒ Separate duplicate letters only within pairs

☒ Ignore duplicate letters in the key phrase

Playfair key

Short version of the Playfair key:



Key matrix

A	L	I	R	E	Z
D	H	G	N	P	O
U	F	S	9	8	1
0	5	B	C	J	K
M	Q	T	V	W	X
Y	2	3	4	6	7

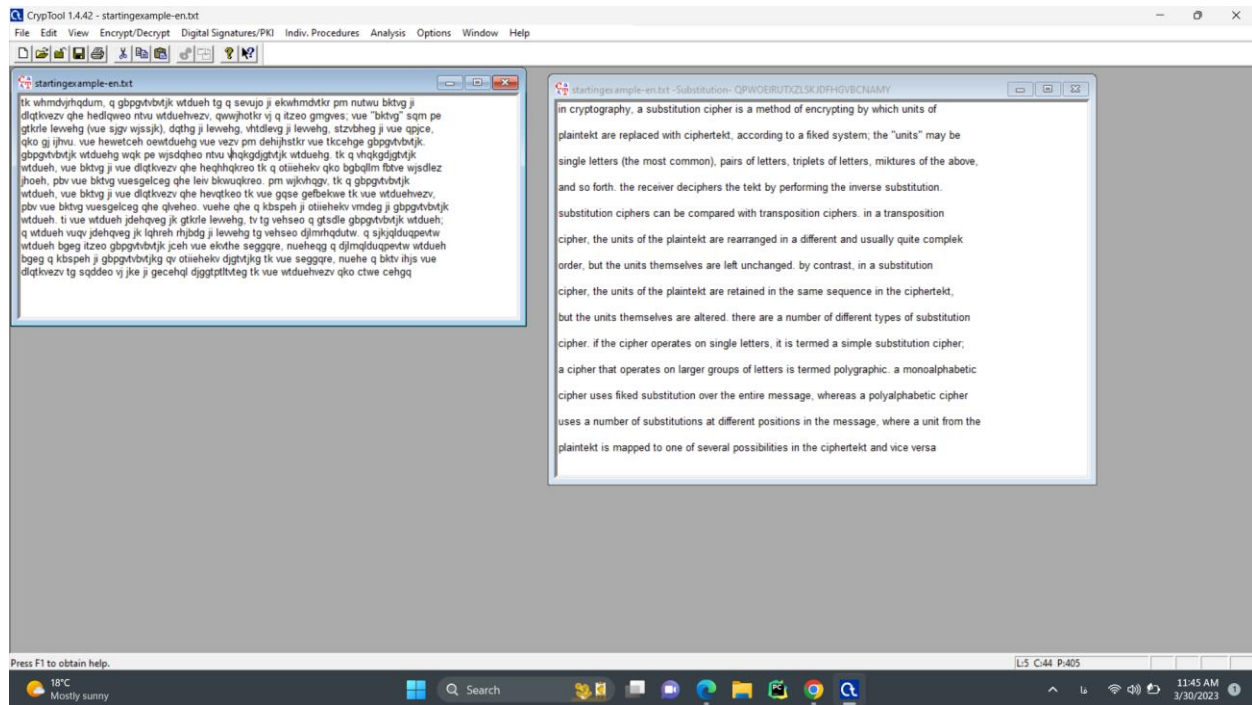
☐ 5x5 matrix

☒ 6x6 matrix

Encrypt Decrypt Cancel

رمز:

ث)
متن اول

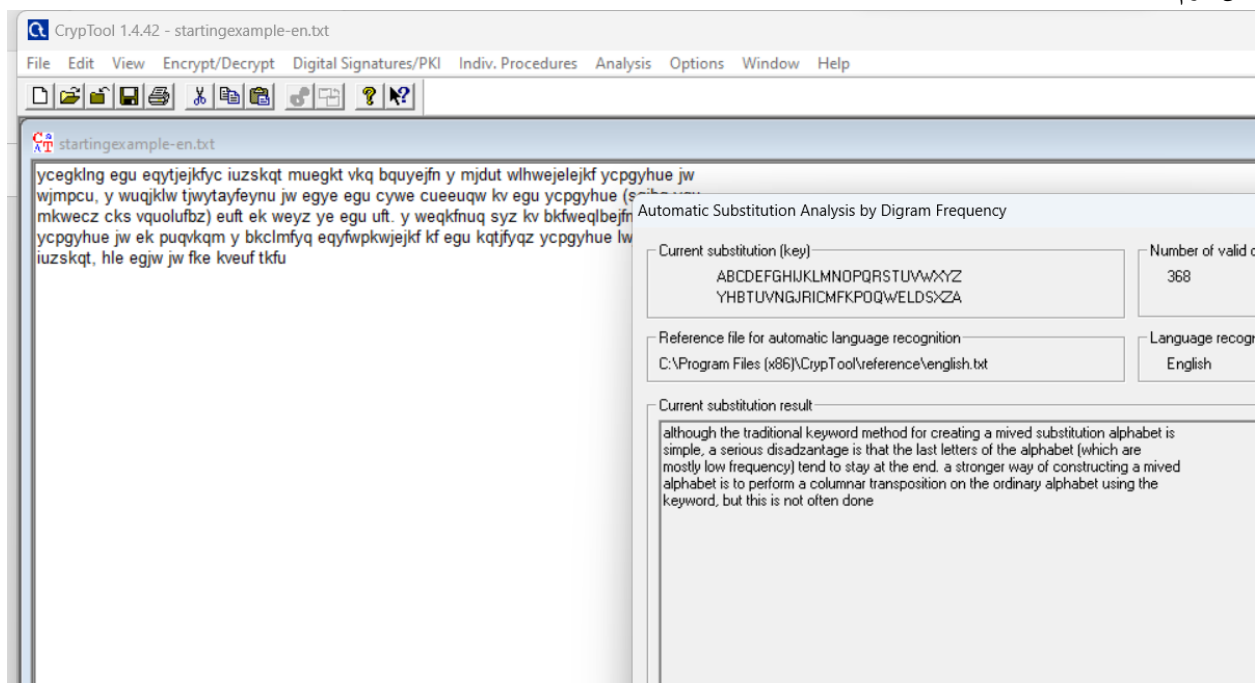


in cryptography, a substitution cipher is a method of encrypting by which units of plaintext are replaced with ciphertext, according to a fixed system; the "units" may be single letters (the most common), pairs of letters, triplets of letters, mixtures of the above, and so forth. the receiver deciphers the text by performing the inverse substitution.

substitution ciphers can be compared with transposition ciphers. in a transposition cipher, the units of the plaintext are rearranged in a different and usually quite complex order, but the units themselves are left unchanged. by contrast, in a substitution cipher, the units of the plaintext are retained in the same sequence in the ciphertext, but the units themselves are altered. there are a number of different types of substitution cipher. if the cipher operates on single letters, it is termed a simple substitution cipher; a cipher that operates on larger groups of letters is termed polygraphic. a monoalphabetic cipher uses fixed substitution over the entire message, whereas a polyalphabetic cipher uses a number of substitutions at different positions in the message, where a unit from the plaintext is mapped to one of several possibilities in the ciphertext and vice versa

uses a number of substitutions at different positions in the message, where a unit from the plaintext is mapped to one of several possibilities in the ciphertext and vice versa

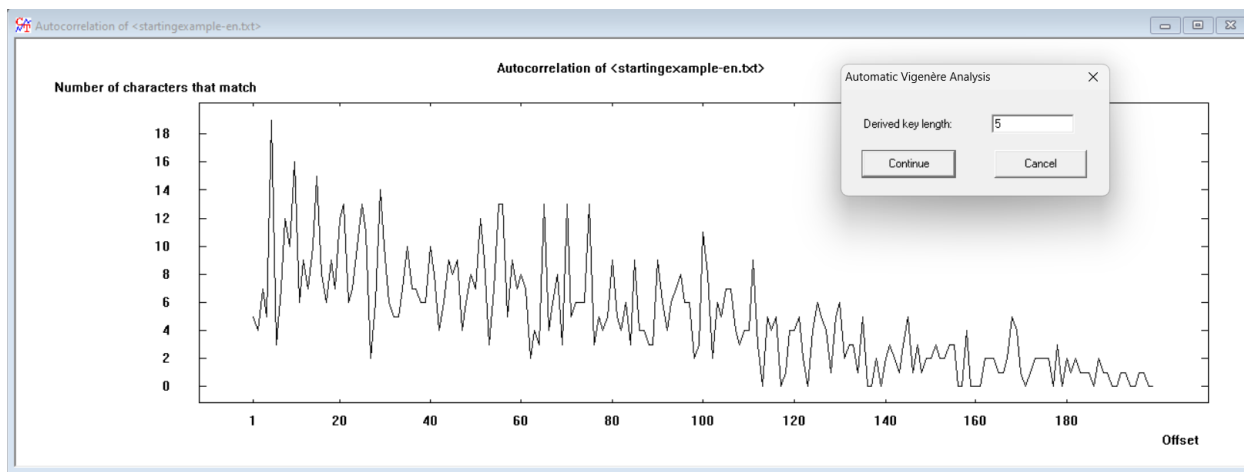
متن دوم



although the traditional keyword method for creating a mived substitution alphabet is simple, a serious disadzantage is that the last letters of the alphabet (which are mostly low frequency) tend to stay at the end. a stronger way of constructing a mived alphabet is to perform a columnar transposition on the ordinary alphabet using the keyword, but this is not often done

متن اول نزدیک تر است و علت نیز این است که متن اول بلندتر است و تحلیل فرکانسی روی آن دقیق تر خواهد بود.

(ج)



نمودار اول autocorrelation است. این معیار نشان‌دهنده‌ی تعداد matching characters بر اساس تعداد shift است که با توجه به نمودار بیشینه‌ی این مقدار در ۵ است یعنی نماینده‌ی طول کلید ۵ است. علت این فرض این است که اگر متن رمز شده را به اندازه طول کلید shift دهیم سبب می‌شود که حروف یکسان با کلید یکسانی رمز شوند و این اتفاق برای مضارب طول کلید رخ دهد دقیقاً به مانند نمودار که تعداد match ها در مضارب ۵ بیشینه می‌شود. سپس با تحلیل فرکانسی این رمز شکسته می‌شود و کلید پیشنهادی SMILE است.

