



فصل دو: رموزهای متقارن از گزشته تا به امروز

امنیت سیستم‌های کامپیوتری

ابوالفضل دیانت

آخرین ویرایش: ۱۳ آبان ۱۴۰۱ در ساعت ۱۷ و ۱۱ دقیقه - نسخه ۲.۵.۱

مزنگاری ت قبیل از شانوں

تاریخچه رمزنگاری (Cryptography)

☞ نخستین مفهوم در امنیت: رمزنگاری (Cryptography)

☞ قدمتی هزاران ساله از هیروگلیف‌ها گرفته تا Atabash و رمزنگاری آرشیلوس.

☞ نخستین الگوی مدون سامانه‌های رمزنگاری: الگوریتم سزار (Ceaser)

This is an example → Wklv lv dq hadpsoh.



رمزنگاری (Cryptography) که برگرفته از دو کلمه یونانی krypto به معنای محرمانه و graphien به معنای نوشتن است، به جرات می‌توان گفت قدمتی هزاران ساله دارد؛ گرچه باید گفت که عمر نگاه علمی به این موضوع، از صد سال تجاوز نمی‌کند. هیروگلیف‌های حکشده بر روی سنگ‌ها (حروفی که با کشیدن تصویرهایی از جانوران و اشیاء پدید آمده باشد-Hieroglyph)، در ۱۹۰۰ سال پیش از میلاد مسیح در تمدن باستانی مصر، شاید نخستین تلاش بشر در مسیر علم رمزنگاری بود، گرچه به نظر می‌رسد هدف مصریان باستان از این کار مخفی کردن پیام نبوده، بلکه بر عکس افزایش جذابیت کتبیه‌ها بوده است. اولین نمونه از این نوع کدگذاری را در آرامگاه مربوط به یکی از اشراف زادگان مصری به نام خنوم‌هتپ دوم (Khnumhotep II) یافت شده است.

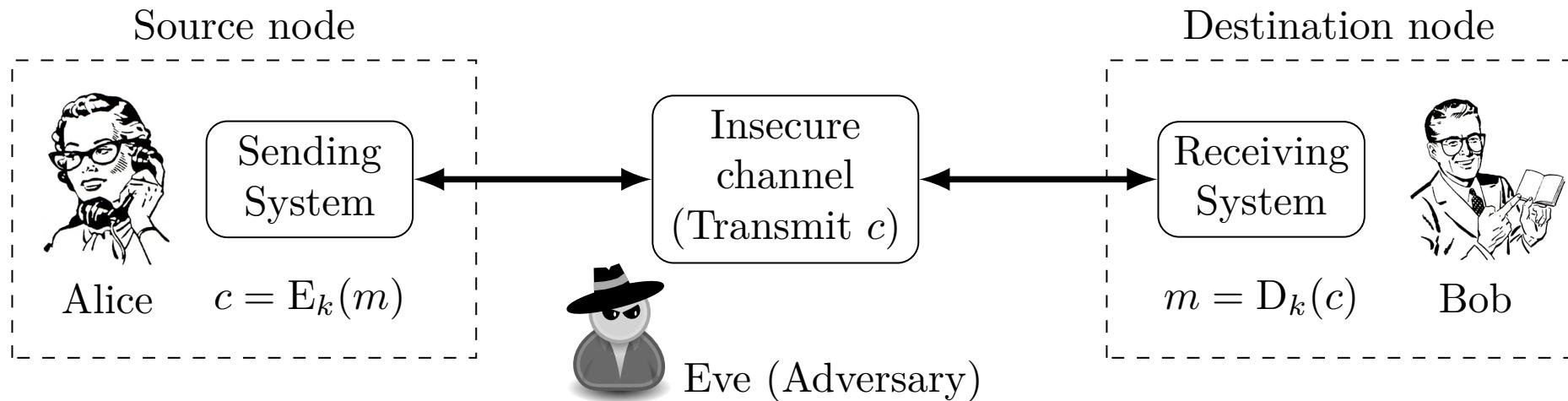
عبری‌ها در نوشتن کتاب مقدس ارمیای نبی، از یک شیوه رمزنگاری به نام Atbash استفاده می‌کردند، بدینسان که نام بسیاری از مکان‌ها و افراد در کتاب مقدس عبری‌ها عمداً با رمز Atbash به صورت مخفی و مبهم نوشته شده است. این رمز بسیار شبیه به رمز جانشینی است، بدین نحو که در Atbash اولین حرف از الفبای عبری با آخرین حرف جدول الفبا جانشین می‌شد. به همین نحو دومین حرف با حرف ما قبل آخر و این

روال به همین ترتیب تکرار می شد تا متن رمز شده به دست آید.

در یکی از شهرهای یونان باستان به نام اسپارتا، پیامها از طریق نوشته شدن روی یک نوار کاغذی و پیچیدن آن دور یک استوانه‌ی با قطر مشخص رمزگذاری می‌شدند. نوار کاغذی تا زمانی که توسط گیرنده آن، روی یک استوانه با همان قطر قرار نمی‌گرفت، به صورت ناخوانا باقی می‌ماند. به این نوع از رمزگاری اصطلاحاً آرشیلوس گفته می‌شد. جالب است بدانید که تا حدود پانصد سال راز این روش مخفی ماند تا عاقبت در ۱۲۰ قبل از میلاد راز این روش نگارش بر ملا شد.

در حوالی ۱۰۰ سال پیش از میلاد مسیح، ژولیس سزار (Julius Caesar)، در مکاتبات خود در هنگام جنگ از یک شیوه نوینی از رمزگاری استفاده می‌کرد که در آن جای حروف الفبا تغییر پیدا می‌کرد. روش او در عین سادگی اما کارا و مفید بود، و عملاً اولین الگوی رمزگذاری ثبت شده در تاریخ به شمار می‌آید. در این الگو هر حرف با حرفی به فاصله n از خودش جانشین می‌شد. الگوی رمزگاری ژولیوس سزار برای دورانی که از هر قوم و قبیله به ندرت کسانی با سواد بودند، به قدر کافی امنیت داشت ولی امروز بیشتر به یک شوخی شبیه است.

مدل یک سامانه رمزنگاری (Cryptography)



Alice قصد دارد تا متن اصلی (Plain Text) m را برای Bob به صورت رمز شده ارسال کند.

Alice با استفاده از الگوریتم رمزگذاری E , و با قراردادن ورودی m و کلید (k) , خروجی تابع که همانا متن رمز (Ciphertext) c را، برای Bob ارسال می‌کند.

$$c = E_k(m) = (m + k) \mod 26 \implies E_3(L_T) = (20 + 3) \mod 26 = L_W$$

الگوریتم رمزگذاری (Encryption) و رمزگشایی (Decryption) که عکس یکدیگرند.

تاریخچه رمزنگاری (Cryptography) (ادامه)



رمز سزار خیلی ساده بود. به نظر می‌رسد که براحتی می‌توان آن را شکست.

Pm ol ohk hufaopun jvumpkluaphs av zhf, ol dyval pa pu jpwoly, aoha pz, if zv johunpun aol vykly vm aol slaalyz vm aol hswohila, aoha uva h dvyk jvbsk il thkl vba.



☞ طول کلید رمز سزار برابر با $\log_2 26 = 4.7$ bit

☞ شاید بتوان رمز را پیچیده تر کرد رمز مُستَوی (Affine Cipher)

$$E_k(m) = (a \times m + k) \mod 26$$

☞ اما a و k چه مقداری می‌تواند داشته باشد؟

☞ اگر یک نگاشت کلی در نظر بگیریم؟ $\log_2(26!) \approx \log_2(2^{88.4}) \approx 88$ bit

☞ آیا ایده‌های دیگری هم دارید؟

برای این‌که معادل‌های فارسی کلمات بکار رفته از یک استاندارد مشخص پیروی کند، ما از واژه‌نامه و فرهنگ امنیت فضای تولید و تبادل اطلاعات (افتا) استفاده می‌کنیم ([این پیوند](#)). برخی واژه و تعاریف اولیه:

متن اصلی (Plain Text) پیام رمزنشده

متن رمز (Ciphertext) پیام رمزشده

رمزنگاری (Ciphering) الگوریتم تبدیل متن اصلی به متن رمز

رمزنگاری (Encryption) یا رمزنگاری (Enciphering) تبدیل متن اصلی به متن رمز

رمزنگشایی (Decryption) یا رمزنگشایی (Deciphering) تبدیل متن رمز به متن اصلی

رمزنگاری (Cryptography) علم اصول و روش‌های رمزنگاری

تحلیل رمز (Cryptanalysis) علم اصول و روش‌های رمزنگشایی متن رمز بدون اطلاع از کلید

رمزناسی (Cryptology) علمی است متشكل از علوم رمزنگاری و تحلیل رمز.

یکی از مهم‌ترین نکاتی که وجود دارد این است که تابع رمزگذاری یعنی E می‌بایست عکس تابع رمزگشایی یعنی D باشد. در واقع تابع E می‌بایست عکس‌پذیر باشد. از سوی دیگر، پیچیدگی (Complexity) اجرای هر دو تابع، باید بسیار پایین باشد.

رمز مُستَوی یک حالت کلی‌تر از رمز سِزار است. البته باید دقต کرد که در این نوع از رمزنگاری می‌بایست، مقادیر a و $n = 26$ باید نسبت به هم اول باشند. پس ما برای a تنها ۱۲ حالت بیشتر نمی‌توانیم داشته باشیم. برای b نیز ۲۶ حالت. پس در حالت کلی 26×12 حالت داریم. البته این مقدار نیز منجر به طول کلید بسیار کوتاهی می‌شود ($\log_2 12 \times 26 \approx 8.3$ bit) و براحتی با حمله Brute-force قابل شکستن است. همان‌طور که مشاهده می‌شود یکی از معیارهای مقایسه بین انواع الگوریتم‌های رمزگذاری از لحاظ قدرت، بحث اندازه فضای کلید (Key Space) است. هر چقدر فضای کلید بزرگ‌تر باشد، آن‌گاه الگوریتم مورد نظر نسبت به حمله-Brute-force مقاوم‌تر است. البته دقت کنید این واقعاً بدان معنا نیست که الگوریتم موردنظر از یک الگوریتم با فضای کلید کوچک‌تر قوی‌تر است، چراکه ممکن است روش هوشمندانه‌ای وجود داشته باشد که بتواند از ضعف‌های

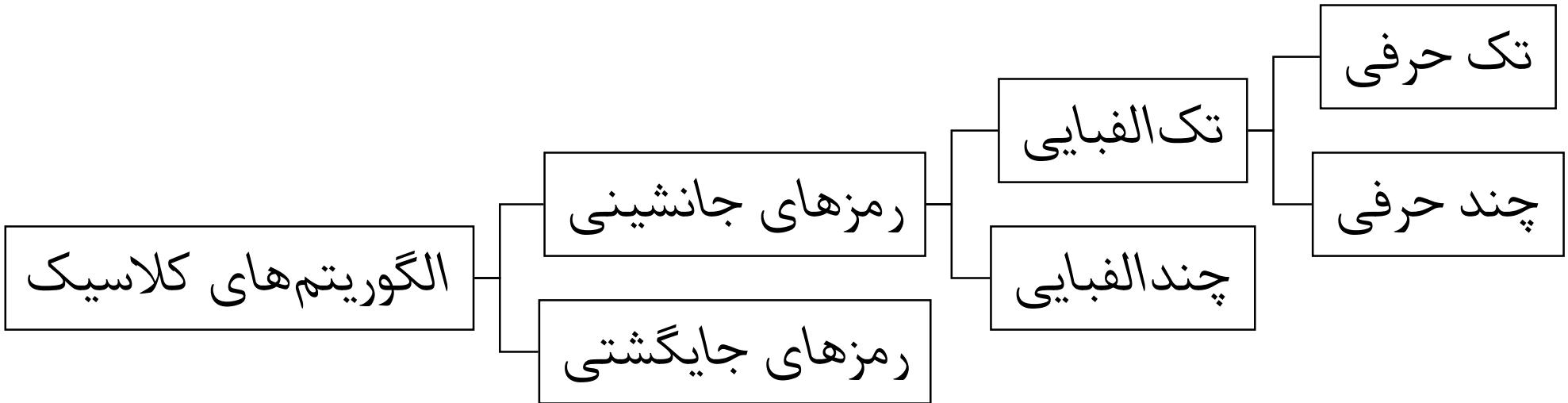
موجود در الگوریتم استفاده کرده و آن را در زمانی کمتر از Brute-force بشکند. در ضمن ما پارامتری به نام طول کلید را نیز در این میان معرفی کردیم. طول کلید (L) برابر است با

$$L = \log_2(|\mathcal{K}|) \text{ [bit]}, \quad (1)$$

که در عبارت فوق $|\mathcal{K}|$ بیانگر ابعاد فضای کلید است.

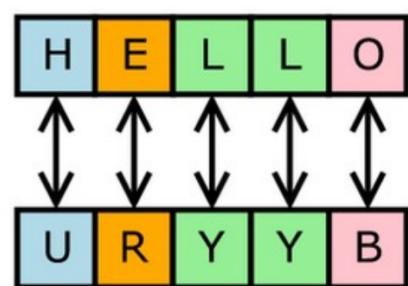
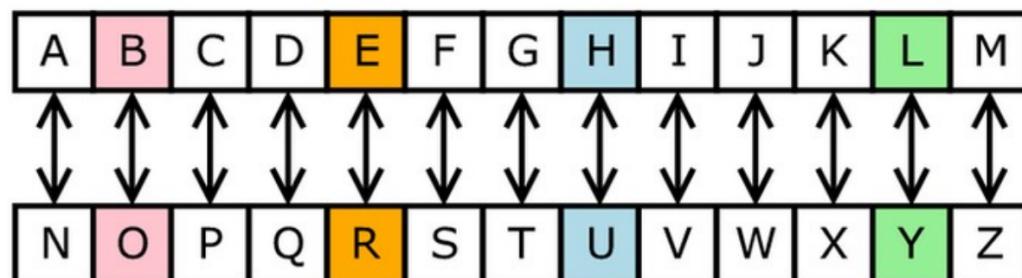
Al-Kindi ریاضی دان عرب‌زبانی بود که مخترع روش تحلیل فرکانسی حروف به عنوان روشی برای شکستن دسته‌ای از الگوریتم‌های رمزنگاری بود. او کتابی درباره رمزنگاری تحت عنوان [رساخ فی الاستخراج معما](#) نوشت که در آن توضیحاتی درباره تحلیل فرکانسی را ارایه داد.

دسته‌بندی سامانه‌های رمزنگاری کلاسیک



در این نوع از رمزگذاری، جایگاه حروف در یک متن

بهم نمی‌خورد، تنها هر حرف یا گروهی از حروف جابجا می‌شوند.

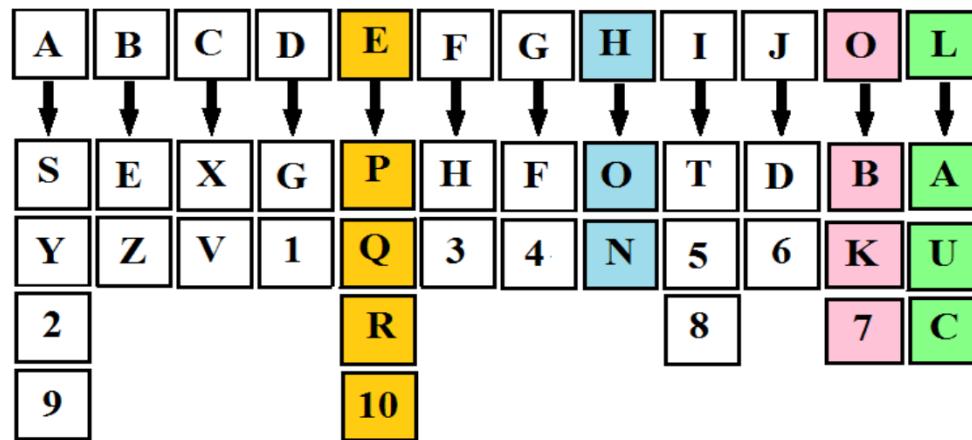


: (Monoalphabetic)

- تک حرفی: به مانند رمز سزار یا رمز مُستَوی

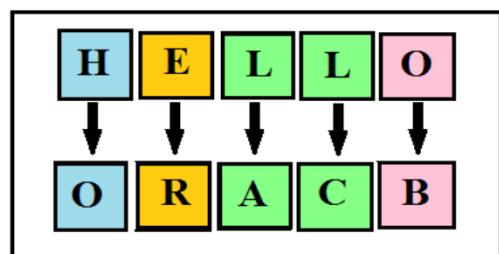
- چند حرفی: برای مثال به جای هر دو حرف، دو حرف دیگر را جایگزین کنیم. فضای کلید بزرگ و طول کلید بزرگ $.L = \log_2 ((26 \times 26)!) [bit]$

رمز جانشینی



رمز چندالفبایی (Polyalphabetic)

- یک حرف، با چند حرف جایگزین می‌شود.
- در حقیقت نگاشت ما یک به یک نیست.
- به مانند ماشین Enigma و رمز Vigenère.



رمز Vigenère

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



یک نوع رمزگاری : Vigenère Cipher

چند الفبایی.

 نحوه رمز کردن: حروف متن اصلی

تعیین کننده ستون و حرف عبارت کلید

تعیین کننده سطر است.

این الگوریتم رمزگذاری، در اصل توسط Giovan Battista Bellaso در سال ۱۵۵۳ در کتاب la cifra del sig نسبت Blaise de Vigenère ارایه شد، گرچه طرح وی ناموفق بود؛ بعدها در قرن پانزدهم به یک فرانسوی به نام Charles Babbage موفق به شکستن چندین رمز تا اوایل ۱۸۵۴ داده شد، و به همین نیز مشهور شد. در نهایت، Kasiski به طور کامل کد رمز را شکست بود، اما در انتشار راه حل کلی برای آن با شکست مواجه شد. در سال ۱۸۶۳ منتشر کرد: و روش آن را در قرن نوزدهم (سال ۱۸۶۳) منتشر کرد.

برای توصیف رمز Vigenère عبارت زیر را به عنوان متن اصلی در نظر بگیرید:

THISISANEXAMPLE

Alice به عنوان فرستنده پیام، یک کلمه را به عنوان کلید انتخاب می‌کند و آن را تکرار می‌کند تا اینکه با طول متن اصلی برابر شود. به عنوان مثال، اگر واژه IUST را بعنوان کلید استفاده کنیم، خواهیم داشت:

IUSTIUSTIUSTIUS

برای رمزگاری پیام، باید متن آن را به صورت پیوسته و بدون فاصله نوشته و در خط بعد حروف کلمه کلیدی

را به ترتیب و پشت سر هم زیر همین متن تکرار کرد. برای دستیابی به متن رمز شده، حروف کلمه کلید را در ردیف‌های جدول و حروف پیام را در ستون‌های جدول می‌یابیم، از کنار هم قراردادن حروف موجود در تقاطع سطر و ستون، متن پیام رمز مشخص می‌شود. به عنوان مثال، اولین حرف متن اصلی T است که با اولین حرف کلید یعنی I، تناظر دارد. بدین‌سان باید با مراجعه به خانه (T,I)، با حرف B مواجه خواهیم شد. در نهایت با طی این فرایند تا انتهای، متن رمز به صورت زیر در خواهد آمد:

BBALQMSGMRSFXFW

مراجع مفید



برای تست الگوریتم برای تست این الگوریتم، می‌توانید از [این پیوند](#) استفاده کنید. همچنین برای مطالعه بیشتر در مورد الگوریتم رمزشکنی Kasiski، می‌توانید [این پیوند](#) را مطالعه کنید.

تعريف ۲

رمز جایگشتی (Transposition Cipher) در این نوع از رمزگذاری، ترتیب حروف متن را بهم



میریزد ولی شکل آنها را تغییر نمی‌کند.

به عنوان نمونه، متن در k ستون در کنار یکدیگر به صورت سطrix نوشته و به صورت ستونی خوانده می‌شود.

Plaintext: Security protects confidentiality

S	E	C	U	R	I	T
Y	P	R	O	T	E	C
T	S	C	O	N	F	I
D	E	N	T	I	A	L
I	T	Y				

Ciphertext: SYTDIEPSETCRCNYUOOTRTNIIEFATCIL

در رمز جایگشتی (Transposition Cipher)، در یک حالت کلی می‌توان نخست متن را به یک سری قطعه (Block) ها با طول N تقسیم‌بندی کرد. سپس سعی کرد بر روی هر قطعه، رمز مورد نظر اعمال کرد. بدینهی است که رمز جایگشتی در حالت کلی برای یک قطعه با طول N ، تعداد $N!$ حالت وجود دارد. در این میان طول کلید برابر با $L = \log_2(N!)$ خواهد شد.

رمز جایگشتی که در اینجا به عنوان نمونه، مثال زده شد، الگوریتم بسیار ساده‌ای دارد. کافی است که عبارت متن اصلی را به صورت سطري بنویسیم و سپس عبارت متن رمز به صورت ستونی بدست خواهد آمد. دقت کنید که در اینجا کلید در حقیقت تعداد ستون‌ها (M) است و بدین‌سان فضای کلید برابر با M خواهد شد. بدینهی است که این رمز بسیار ساده و قابل شکستن است. در حوزه رمز جایگشتی، الگوریتم‌های جالب و قوی‌تری نیز وجود دارد. به عنوان نمونه، یک ایده بهتر آن است که ستون‌ها به ترتیب خوانده نشود. مثلاً اگر تعداد ستون‌ها $M = 6$ است، یک دنباله دیگر به عنوان کلید داشته باشیم که ترتیب خواندن ستون‌ها را مشخص کند، مثلاً:

$$[6, 3, 2, 4, 1, 5]$$

پر واضح است که برای یک طول ثابت، این ایده برای ما $M!$ فضای کلید به ارمغان می‌آورد.



سوال اول: امروزه تا چه میزان می‌توان در یک زمان معقول حمله Brute-force انجام داد؟ هدف از این سوال این است که، بگویید Graphics Central Processing Unit (CPU) یا Processing Unit (GPU) های فعلی تا چه میزان می‌توانند محاسبات را در ثانیه انجام دهند؟ به عنوان نمونه یکی دو مدل به همراه آن مثال بزنید.

سوال دوم: آلمان‌ها در طول جنگ جهانی اول از یک سامانه رمزگذاری به نام Double Transposition استفاده می‌کردند. در مورد این الگوریتم تحقیق کنید و به طور مختصر آن را توضیح دهید.

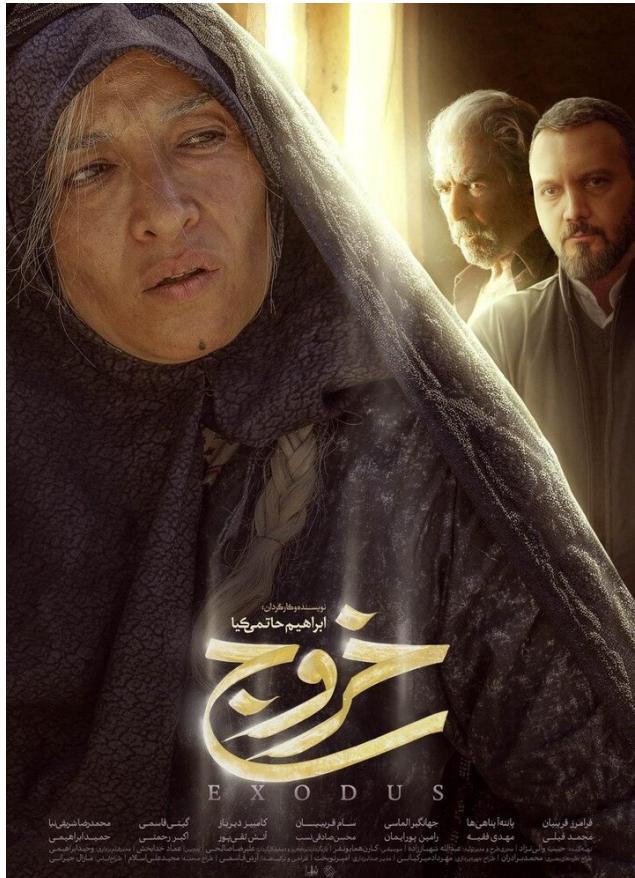
سوال سوم: رمز زیر را بشکنید و نوع آن را نیز مشخص کنید. از تحلیل فرکانسی سعی کنید بهره بگیرید، و روند کار را نیز توضیح دهید.

CIMWB DWQPW TPAMS DDAJP TKPKJ KPWMA ZPWJK NEPWD WJMWQ AXTPF IM-
RWV WRAZQ KDKQI PEKXT RKPWX QEYIR RNWSX DJWQW TWXPW TKXTY IRRWB
PWXTP FWDWJ ZAJUK XQWAZ CKDDR IQKPI AXMKR AXCYI PFWBD KXTIX CPFWM
QADWA ZQKDK NIRIP IWMIX MSDDA JPAZI XQJWK MIXCR EXWYK XTIXX AVKPI VWKDD
RIQKP IAXMK QJAMM PFWJW KRUMA ZYIJW RWMMQ AXXWQ PIVIP EQACX IPIAX
MWXMI XCKXT IUKCI XCCFI CFWJZ JWGSW XQIWM YIRRW XKNRW USQFZ KMPWJ
MKUDR IXCJK PWMIX KTTIP IAXPA DJAVI TIXCM ICXIZ IQKXP RENWP PWJPF JASCF
DSPKX TFICF WJTKP KJKPW MPFWQ AUNIX KPIAX AZMSN UUYKV WWCYK VWRWX
CPFMM UKRRW JPFKX AXWUI RRIUW PWJKX TPFWS MWAZZ JWGSW XQEMW RWQPI
VIPEP ATWPW JUIXW JWRKP IVWWR WQPJA UKCXW PIQKN MAJDP IAXJK PWMIM
WBDWQ PWTPA RWKTP ADAPW XPIKR REMIC XIZIQ KXPKT VKXQW MIXYI JWRWM

MMWXM IXCPW QFXAR ACEKT TIPIA XKRRE YFWJW KMPFW KTTIP IAXAZ UANIR
WWTCW QAUDS PIXCI MKDAI XPAZQ AXMIT WJKPI AXKMK XKTTI PIAXP ACXWP YA-
JOM UANIR WWTCW QAUDS PIXCY IRRNW NSIRP IXPAK RRCXW PYAJO MWTCW KX-
TQA JWQAU DSPIX CYIRR NWQAU WUSQF UAJWM WKURW MMREI XPWCJ KPWTK MD-
KJP AZKQA UNIXW TQAUU SXIQK PIAXM QAUDS PKPIA XIXZJ KMPJS QPSJW ZJKUW YA-
JON EPFWP IUWCX WPYAJ OMKJW TWDRA EWTPF IMYIR RDJAV ITWUK XEDAP WXPIK
RKTVK XPKCW MKMCP WQFXA RACEN WQAUW MADWJ KPIAX KRIXQ RSTIX CIUDJ
AVWTK QQWMM PAKJP IZIQI KRIXP WRRIC WXQWQ KDKNI RIPIWM

میاں پر نامہ

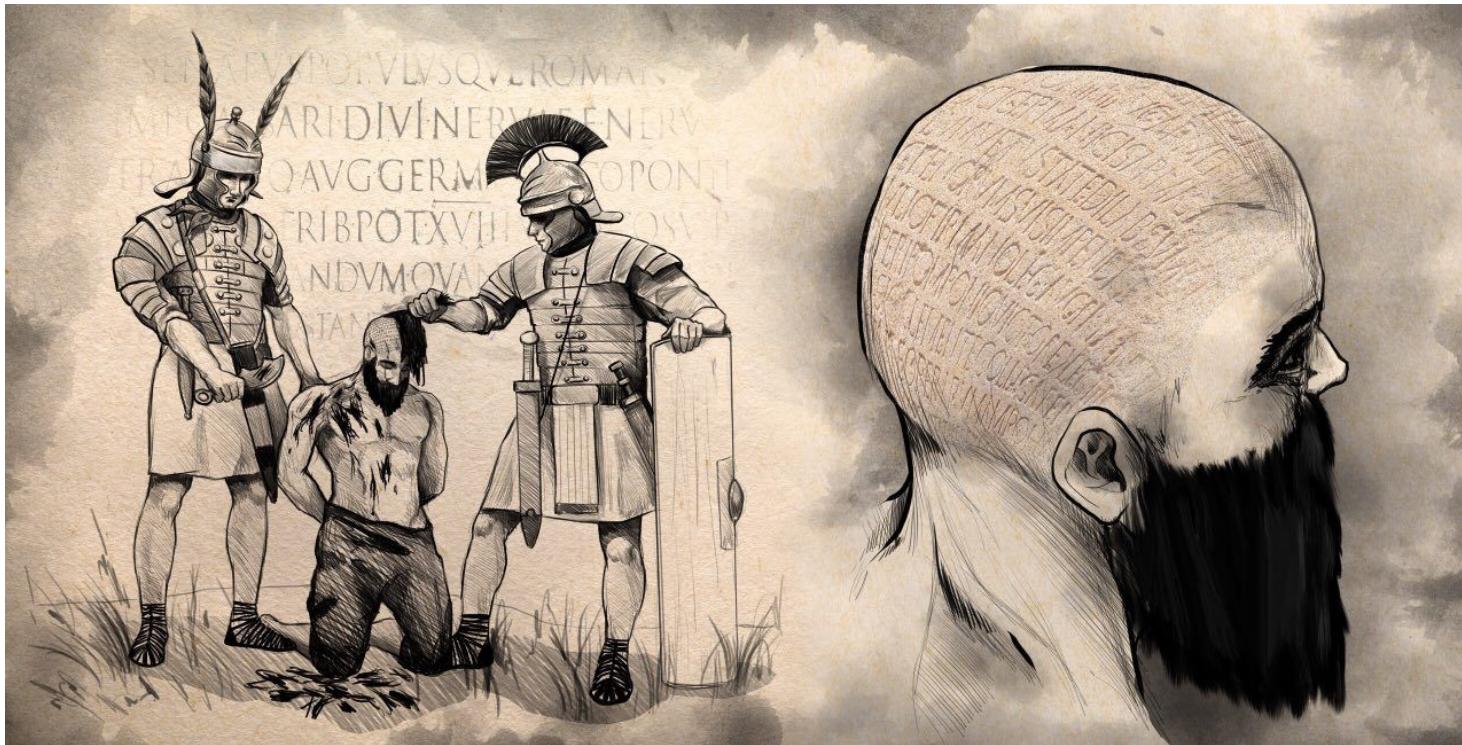
چه راهکاری دارد؟



❶ در آلمان ۴۵ کیلوگرم هروئین به طرزی ماهرانه در نه قالیچه جاساز شده بود. ماموران متوجه شدند که در هر قالیچه پنج کیلو مواد، در ریسمان‌های نازکی در تار و پود کناره فرش‌ها جاساز شده بود.

❷ چگونه می‌توان فهمید که فیلم خروج حاتمی‌کیا، از کدام سینما و از کدام منبع لورفته است؟

نهان‌نگاری (Steganography) چیست؟



در ۴۴۰ سال قبل از میلاد، در کتاب خود به تراشیدن سربردگان، خال کوبی پیام بر روی سرآنها Herodotus  و رشد مجدد موی سر برده کان، اشاره کرده که به وسیله آن پادشاه می‌توانست، بدون هیچ‌گونه شکی از ناحیه دشمن، پیام خود را انتقال دهد.

نهان‌نگاری (Steganography) چیست؟



(ب) تصویر استخراج شده



(آ) تصویر نهان‌نگاری شده

نهان‌نگاری (Steganography)

تعريف ۳

يعنى علم و هنر پنهان‌سازى يك پيام در داخل يك پيام و يا يك شى ديگر.

تفاوت رمزنگارى (Cryptography) و نهان‌نگارى در آن است که در رمزنگارى هدف مخفی‌سازی محتواي پيام است و نه وجود آن، اما در نهان‌نگارى هدف مخفی‌سازی هر گونه نشانه‌اي از وجود پيام است.

نشان‌گذاری (Watermarking) چیست؟



- تفاوت بین نشان‌گذاری (Watermarking) و نهان‌نگاری (Steganography)
- انواع دسته‌بندی‌ها: Visible و Invisible و در سوی دیگر شکننده و غیرشکننده.

مطمئنا رمزنگاری (Cryptography) به تنها یی نمی توانست امنیت پیام را در یک جنگ، تضمین کند، چراکه Herodotus کوچکترین شک دشمن مبنی بر ارسال هرگونه پیام محترمانه، موجب قطع کanal مخابراتی او می شد. در ۴۴۰ سال قبل از میلاد، در کتاب خود دو مثال ذکر کرده است که توسط آن فرد می توانسته به طور امن، پیغام خود را ارسال کند. تراشیدن سربردگان، خالکوبی پیام بر روی سرآنها و رشد مجدد موی سر بردگان، به پادشاه Herodotus تضمین می داد که بدون هیچ گونه شکی از ناحیه دشمن می تواند پیام خود را انتقال دهد. کاری که به آن اشاره کرد، را امروزه نهان نگاری (Steganography) می نامیم.

اصل کلمه نهان نگاری را باید در یونان باستان جستجو کرد. Steganography در حقیقت از دو واژه یونانی "steganos" (στεγανός) به معنای نوشتن، سرچشمeh گرفته "graphei" (γράφη) به معنای محافظت شده و است. پس در کل Steganography یعنی هنر پنهان نوشتن. اولین کسی که در تاریخ از این واژه استفاده نمود، فردی به نام Johannes Trithemius که در سال ۱۴۹۹ در کتابی در مورد جادو، از این واژه استفاده نمود.

نهان نگاری به مانند رمزنگاری علمی است چالش برانگیز؛ چراکه در نقطه مقابل نهان ساز، فرد یا افرادی وجود

دارند، که می‌خواهند کار نهان‌ساز را با شکست مواجه کنند. اصطلاحاً به این علم، نهان‌کاوی (Steganalysis) گفته می‌شود. در حقیقت نهان‌کاو، شخصی و یا سازمانی است غیر مجاز (غیر مجاز از دیدگاه گیرنده و فرستنده سیگنال) که به سیگنال ارسالی دسترسی دارد. او تلاش می‌کند تا جواب هر یک از سوالات زیر را بیابد:

❶ آیا پیامی در سیگنال پنهان شده است یا نه؟ مهم‌ترین وظیفه یک نهان‌کاو تشخیص پاک (Clear) یا غیرپاک (Stego) بودن سیگنال است.

❷ بدست آوردن اطلاعات جانبی از پیام، همچون طول پیامی که پنهان شده است؟
❸ بدست آوردن اصل پیام پنهان شده.

البته نهان‌نگاری خود زیرمجموعه علمی بزرگتر به نام نهان‌سازی اطلاعات (Information Hiding) است. دو علم کلیدی در نهان‌سازی اطلاعات، را می‌توان علوم نشان‌گذاری (Watermarking) و نهان‌نگاری دانست.



سوال اول: پیاده‌سازی یک روش نهان‌کاوی به عنوان آشکارسازی بر نهان‌نگاری به روش LSB. یعنی فرض کنید که ما با روش LSB عملیات نهان‌نگاری را انجام دادیم، شما باید یک روش نهان‌کاوی به منظور تشخیص آن پیاده‌سازی کنید.

سوال دوم: پیاده‌سازی یک روش نشان‌گذاری از نوع Visible و شکننده . البته بدیهی است که ابتدا باید تحقیق کنید و یک روش نشان‌گذاری از نوع Visible و شکننده را پیدا کنید و سپس آن را پیاده‌سازی کنید. پیاده‌سازی باید به زبان‌های C++ یا Python باشد.

الگو، پیتمهای رمزگاری نویس - پنهان کردن



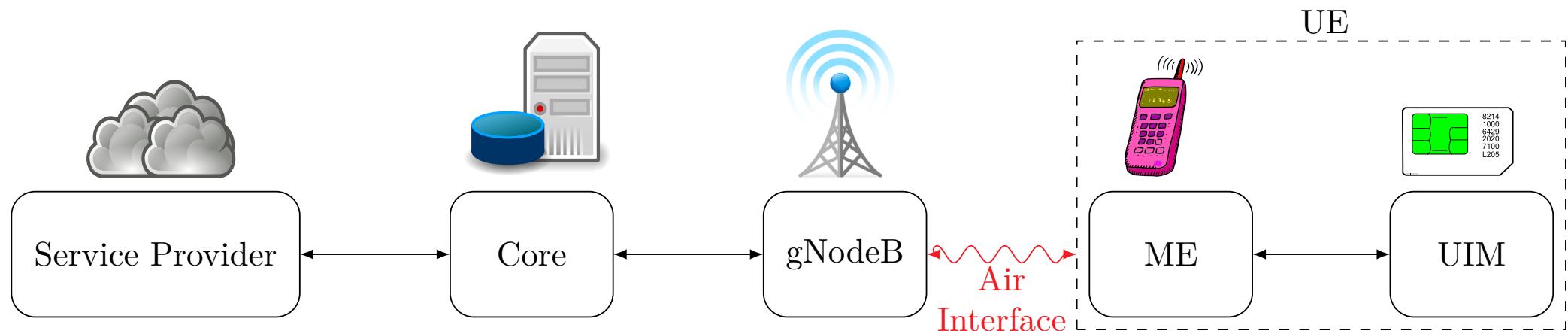
Shannon ریاضی‌دان، مهندس الکترونیک و رمزنگار معروف آمریکایی است که به عنوان پدر نظریه اطلاعات شناخته می‌شود. او در مقاله ۱۹۴۸ خود علم نظریه اطلاعات را پایه‌گذاری می‌کند و در مقاله ۱۹۴۹ خود علم رمزنگاری را بنیان‌گذاری می‌کند. شانون در هر دو مقاله به مبحث ارسال پیام در یک سامانه مخابراتی می‌پردازد، اما با دو دیدگاه مختلف.

Shannon, Claude Elwood. “A mathematical theory of communication.” *Bell system technical journal* 27, no. 3 (1948): 379-423.

Shannon, Claude Elwood. “Communication theory of secrecy systems.” *Bell system technical journal* 28, no. 4 (1949): 656-715.



نگاه شانون به امنیت: سخن اول



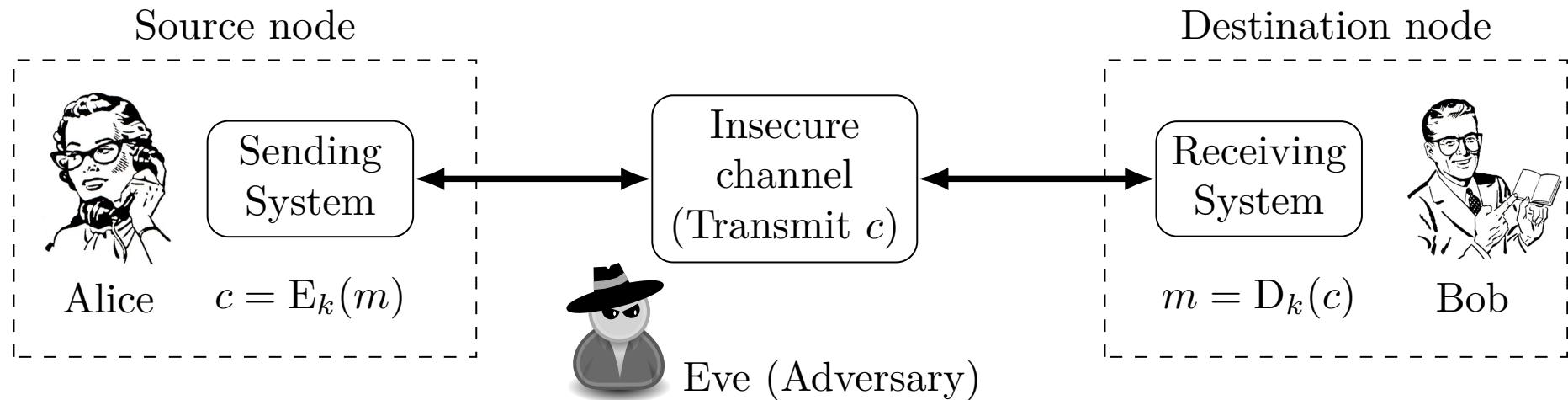
دو رویه رمزگذاری و رمزگشایی باید پیچیدگی (Complexity) کم و روند ساده‌ای داشته باشند، تا بتوانیم به صورت بی‌درنگ (Realtime)، عملیات مذکور را انجام دهیم.

مثلا در یک شبکه تلفن همراه نسل پنجم باید تا 10Gbps گذردهی (Throughput) توسط (UE) (User Equipment) انجام شود!

پشتیبانی شود!

اما کار حمله‌گر (Attacker) باید خیلی سخت باشد. اما چقدر سخت؟!

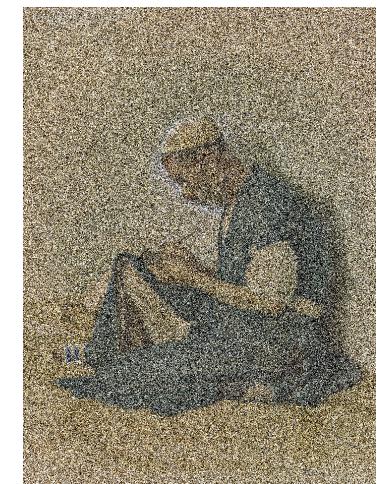
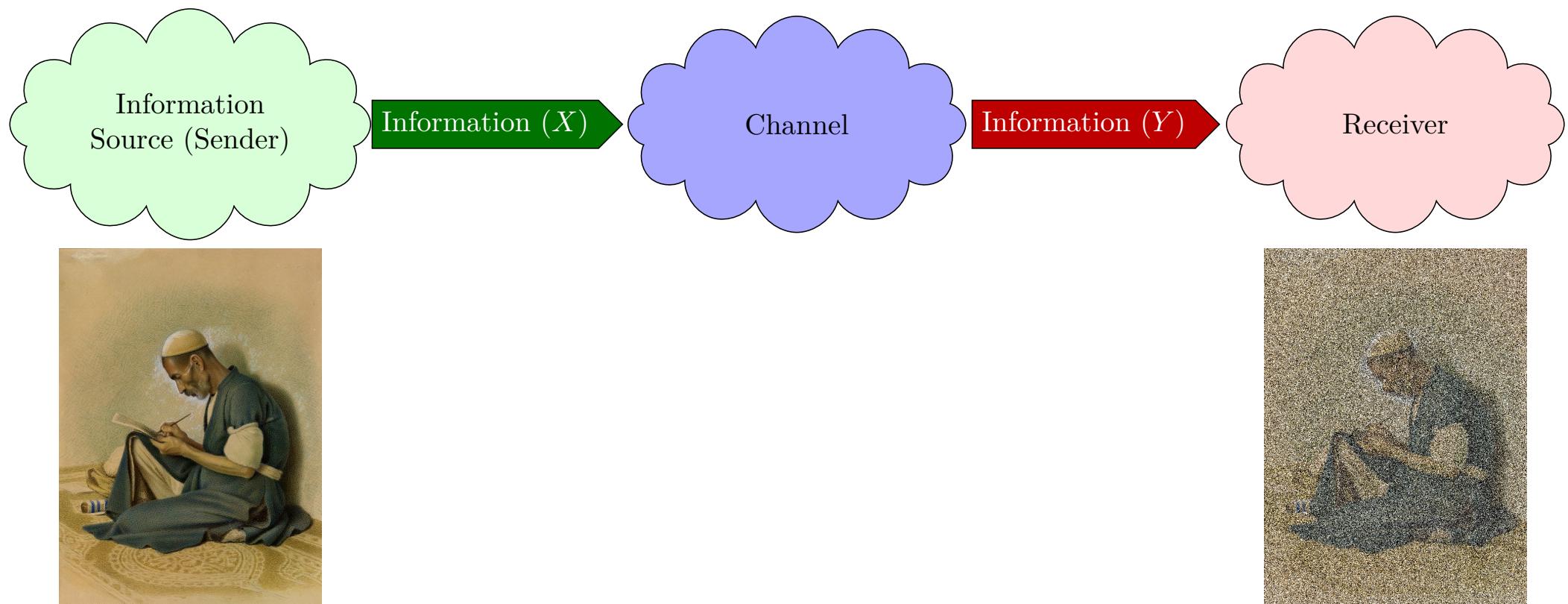
نگاه شانون به امنیت: سخن دوم



اصل Kerckhoffs: امنیت یک الگوریتم رمزگذاری باید مبتنی بر مخفی ماندن کلید باشد، حتی اگر حمله‌گر دانش کافی راجع به کل فرایند رمزگذاری و رمزگشایی داشته باشد.

شانون نیز گفت: *The enemy knows the system.*

نگاه شانون به امنیت: سخن سوم

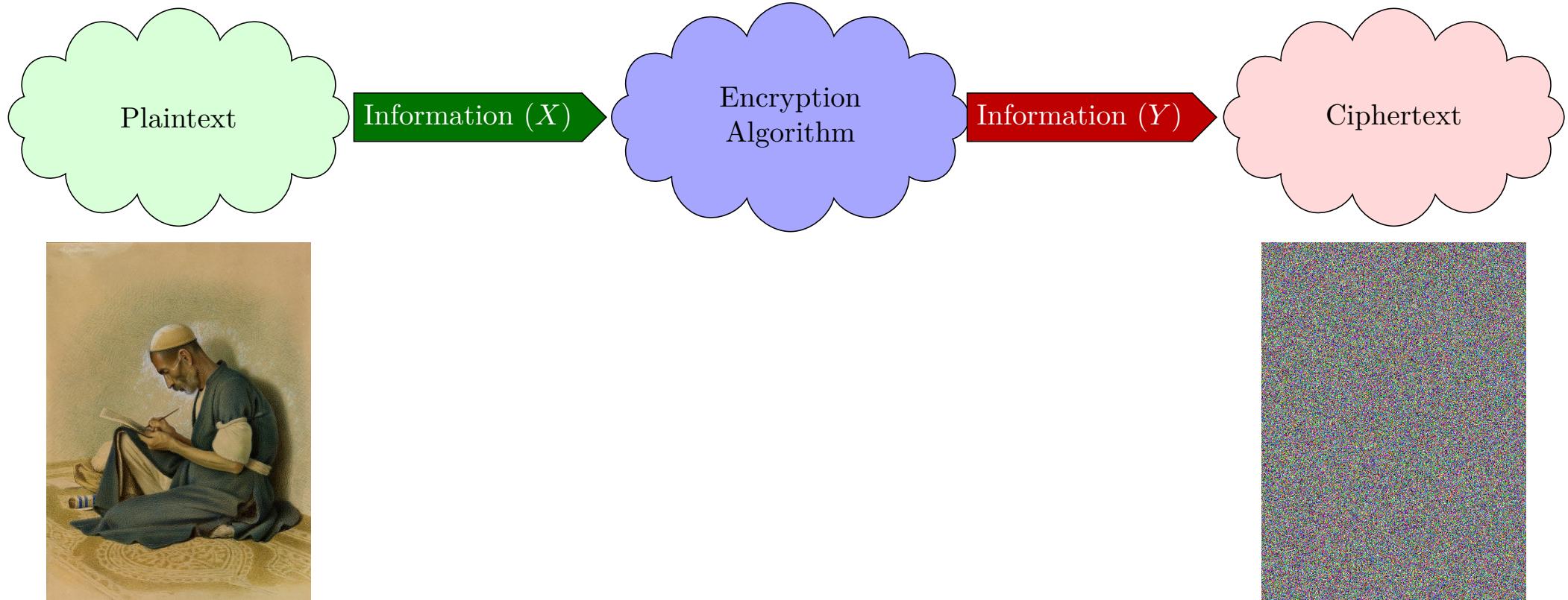


در یک سامانه مخابراتی، پیام از سوی منبع ارسال شده، و پس از گذر از کanal، به دست مقصد می‌رسد.

بدیهی است که در یک سامانه مخابراتی ما تلاش می‌کنیم که سیگنال را به قدری محافظت کنیم که دچار

خرابی کمتری شود.

نگاه شانون به امنیت: سخن سوم (ادامه)



❸ شانون تلاش کرد تا یک سامانه رمزگذاری را به صورت یک سامانه مخابراتی مدل کند. در یک سامانه رمزگذاری، ما به عمد می‌خواهیم یک نویز به متن اصلی اضافه کنیم. حمله‌گر در صورت مشاهده متن رمز، نباید به هیچ‌گونه اطلاعاتی در مورد متن اصلی پی ببرد.

☞ دو نوع امنیت برای یک سامانه قابل تعریف است:

- **امنیت بدون شرط (Unconditional Security):** در صورتی که علی‌رغم توان زیاد محاسباتی دشمن، نتواند بر اساس متن رمز شده سیستم را بشکند، چرا که هیچ‌گونه اطلاعاتی از متن اصلی توسط متن رمز درز نمی‌کند.

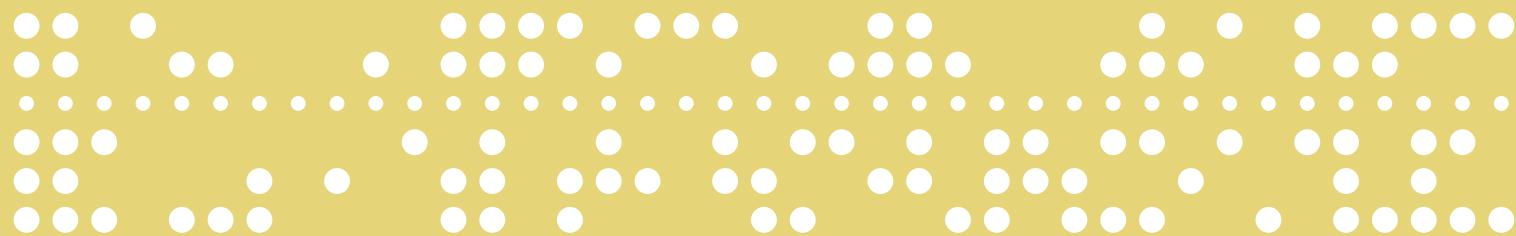
- **امنیت محاسباتی (Computational Security):** در صورتی که شکستن سیستم رمز عملاً از نظر محاسباتی پیچیده و طولانی باشد.

- ☞ تنها سامانه بدون شرط امن شناخته شده، سامانه One Time Pad یا Vernam یا Shannon است. این موضوع را در مقاله خود اثبات کرد.

سامانه Vernum



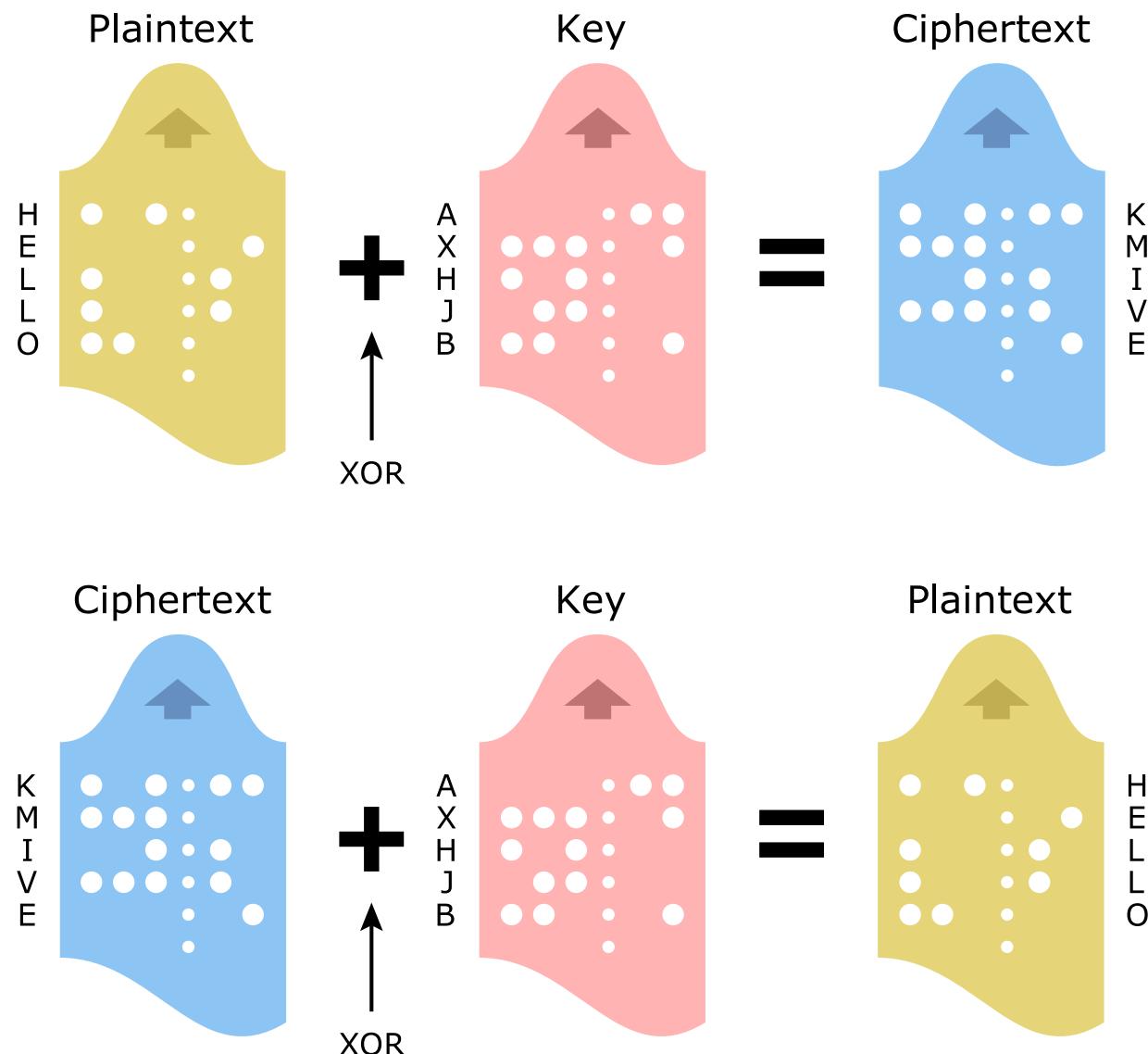
NUL
CR
LF
SP
FIGS
LTRS



H E L L O

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

سامانه Vernum (ادامه)



- نیاز به دنباله کلید (Key) کاملاً تصادفی با طولی Sequence) برابر با طول متن اصلی داریم.
- برای رمزگذاری یک کاراکتر از متن اصلی با یک کاراکتر از کلید، بیت به بیت XOR شود.
- برای رمزگشایی نیز کافی است متن رمز را با کلید ترکیب کنیم.



☞ اجازه دهید کمی در مورد سامانه Vernum بحث کنیم. مزایا و معایب این سامانه چیست؟ گفته نسبت به حمله نوع اول فقط ایمنی دارد!

یکی از دستگاه‌های مکمل تلگراف دستگاه Teletype (TTY) و بعدها مدل ارتقا یافته آن Telex بود. کاربر این دستگاه، قبل از ارسال پیام خود را تایپ می‌کرد. در ضمن تایپ پیام، ماشین روی نوار کاغذی باریکی، به ازای هر حرف سوراخ‌هایی را ایجاد می‌کرد (Baudot code). سپس ابتدای این نوار کاغذی را، در فرستنده قرار می‌دادند. دستگاه فرستنده، به ترتیب با خواندن این سوراخ‌ها پیام مناسب را ارسال می‌کرد.

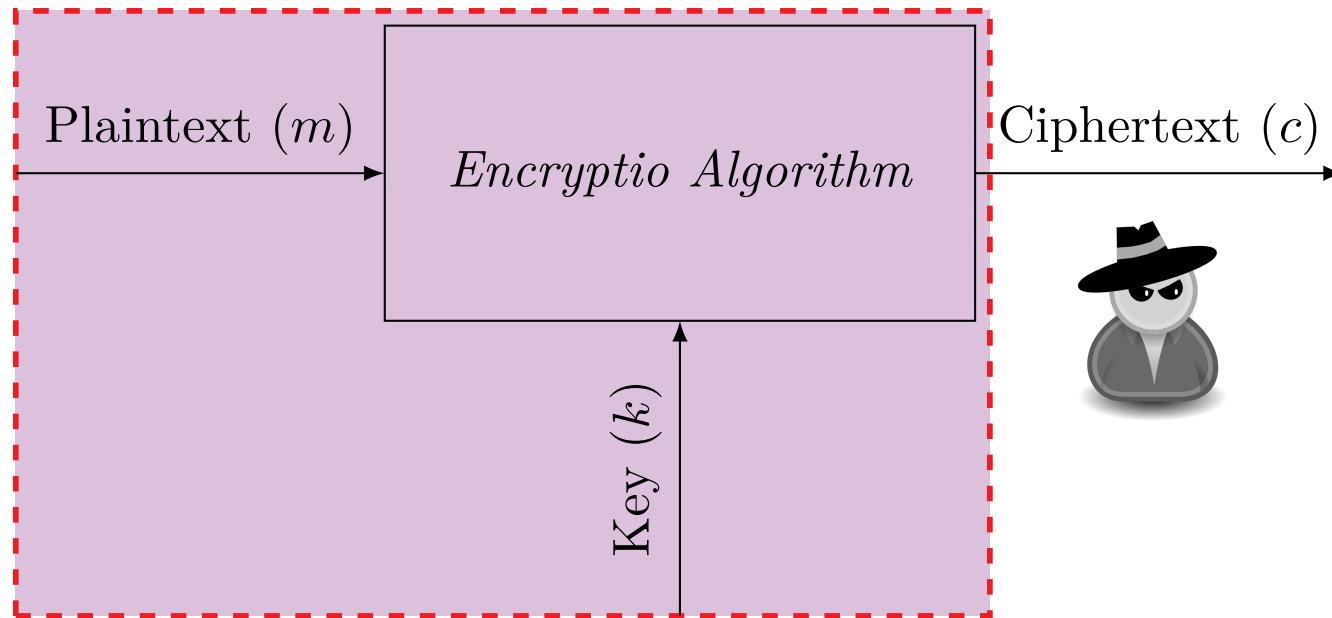
رمز Vernum نخستین بار توسط Gilbert Sandford Vernam در سال ۱۹۱۷ برای بکارگیری در دستگاه‌های Teletype، ابداع شد. در این سامانه، یک دنباله کلید (Key Sequence) داریم به اندازه طول متن اصلی. این الگوریتم، بر این اصل استوار است که هر کاراکتر متن اصلی با یک کاراکتر از کلید ترکیب می‌شود (بیت به بیت با یکدیگر XOR شود). اگر از یک دنباله کلید واقعاً تصادفی استفاده شود، نتیجه یک متن رمز واقعاً «تصادفی» خواهد بود، که هیچ ارتباطی با متن اصلی ندارد.

از دیدگاه شانون، چنین سامانه‌ای ویژگی امنیت بدون شرط را دارد. بدین‌سان می‌توان این متن رمز را بدون خطر شنود (Eavesdropping)، برای گیرنده ارسال کرد. تمام کاری که گیرنده باید انجام دهد این است که متن

رمز شده را با همان دنباله کلید دوباره ترکیب کند (بیت به بیت با یکدیگر XOR شود).

یکی از مشکلات سامانه Vernum، طول کلید این الگوریتم است، چراکه می‌باشد طولی برابر با متن اصلی داشته باشد. از دیگر مشکلات این سامانه، این است که تنها نسبت به حمله نوع اول مقاوم است، و در برابر دیگر حملات به سادگی شکسته می‌شود. در سال ۱۹۴۴-۱۹۴۵، سرویس جاسوسی ارتش آمریکا، موفق شد یک سیستم One Time Pad را که توسط وزارت امور خارجه آلمان برای مخابرات سطح بالا استفاده شده بود و نام رمزی آن GEE بود، حل کند. این سیستم ناامن بود چون دنباله کلید به اندازه کافی تصادفی نبود.

سه نوع حمله - نوع اول



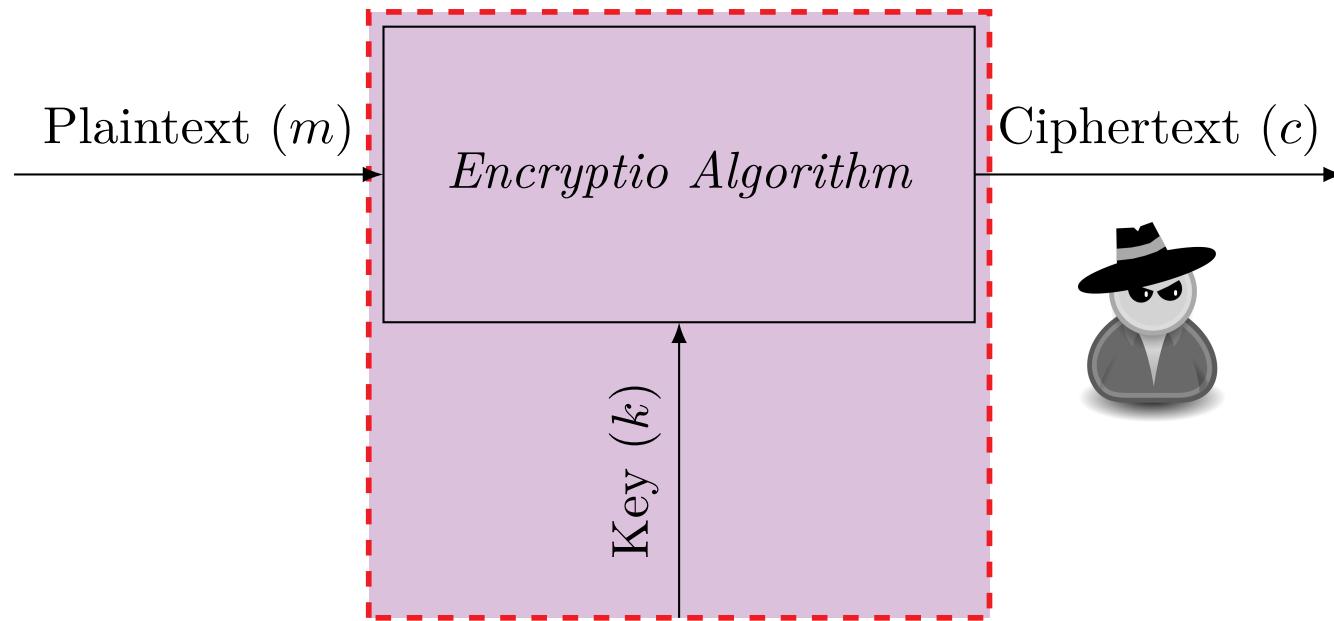
حمله براساس فقط متن رمزشده (Ciphertext Only Attack): در این حمله دشمن فقط متن رمزشده را دارد و از آن باید بتواند کلید را بدست آورد.

WEP به مانند تحلیل فرکانسی حروف در رمز جانشینی ساده تک حرفی یا به مانند حمله به الگوریتم

مثال ۱

.(Wired Equivalent Privacy)

سه نوع حمله - نوع دوم



حمله با متن اصلی معلوم (Known Plaintext Attack): دشمن به یک یا چند متن اصلی و متون رمز شده متناظر دسترسی دارد. استفاده از Crib (/krɪb/) از

مثال ۲

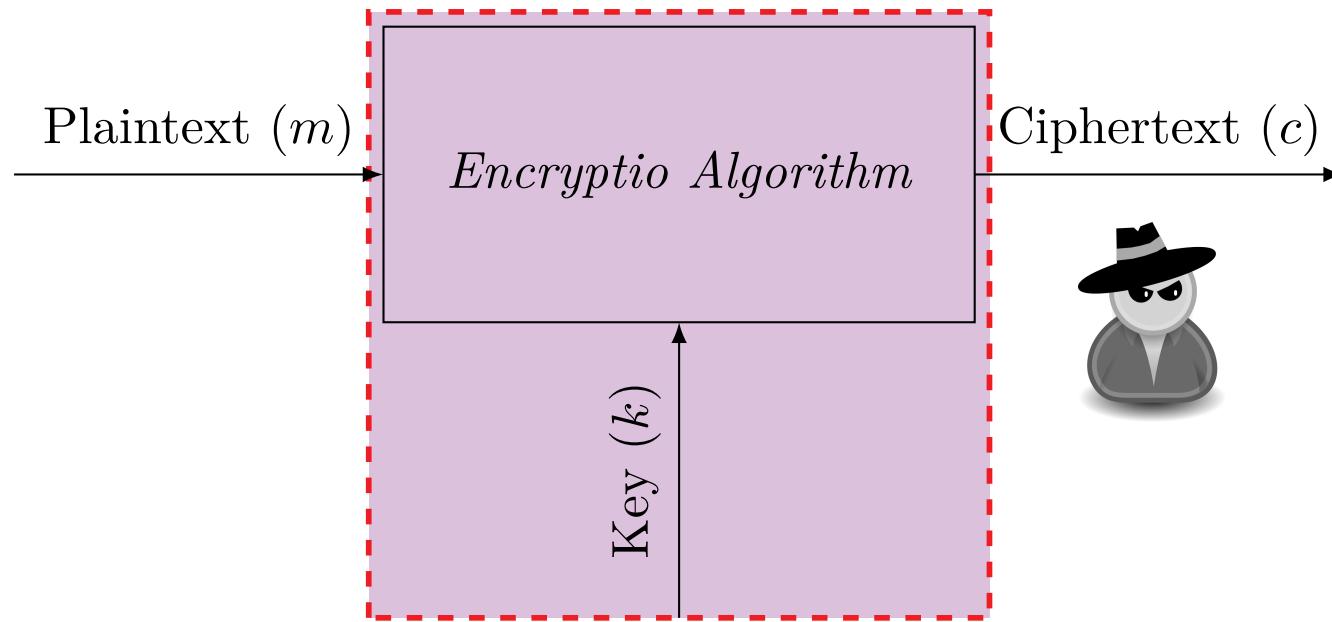
دو مثال مشهور، در این زمینه رمزشکنی ماشین Enigma و A5/2 در شبکه‌های نسل دو (Global System for Mobile Communication (GSM)) است. در هر دو، بخشی از متن متن اصلی معلوم بوده است.

سه نوع حمله - نوع دوم (ادامه)



فیلم بازی تقلید (The Imitation Game)

سه نوع حمله - نوع سوم



حمله با متن اصلی منتخب (Chosen Plaintext Attack): در این نوع از حمله دشمن به هر متن رمز و متن اصلی معادل آن دسترسی دارد، و تنها کلید را ندارد (قویترین حمله).

مثال ۳ مثلاً ما یک ماشین رمزگذاری داریم که کاربر آن در حقیقت یک جاسوس است که کلید را ندارد، ولی ما می‌توانیم از او بخواهیم هر متن اصلی دلخواهی را ارسال کند.

در حمله نوع اول یا حمله براساس فقط متن رمزشده (Ciphertext Only Attack)، فرض می‌شود که حمله‌گر تنها به متن رمز دسترسی دارد. البته دقیق‌تر کنید که این منافاتی با اصل Kerckhoffs ندارد. یعنی ما فرض می‌کنیم که حمله‌گر به طور کامل از جزئیات سامانه آگاهی دارد، ولی برای متن رمزای که در اختیار او است، متن اصلی معادل و کلید را نمی‌داند. به عنوان مثال می‌توان از حملات تحلیل فرکانسی در رمزهای کلاسیک نام برد. همان‌طور که به یاد دارید در این نوع از حملات متنها به متن رمز دسترسی داریم و با تحلیل فرکانسی حروف در آن، سعی در بدست آوردن اطلاعاتی در مورد متن اصلی و کلید داریم. مثال دیگر در این حوزه حمله به الگوریتم (WEP) Wired Equivalent Privacy بود. (WEP) یک الگوریتم برای تامین امنیت شبکه‌های IEEE 802.11 بود که در سال ۱۹۹۷ معرفی شد. تلاش این پروتکل بر آن بود که ویژگی محترمانگی را نسبت به سامانه‌های سنتی برآورده سازد.

در حملات نوع دوم یا حمله با متن اصلی معلوم، فرض می‌شود که حمله‌گر علاوه بر این که به متن رمز دسترسی دارد، به یک یا چند متن اصلی به همراه متن رمز معادل آن نیز دسترسی دارد. ما به متون اصلی یی که حمله‌گر

متن رمز معادل آن را حدس می‌زند، اصطلاحا Crib می‌گوییم. واژه Crib، بیشتر از زبان عامیانه گرفته شده است. داستان حضور این واژه در حوزه امنیت به داستان ماشین Enigma و شکستن رمز آن توسط تیم Turing در Bletchley Park بر می‌گردد. یک ماشین بسیار قوی برای رمزگاری بود که آلمان‌ها در طول جنگ جهانی دوم، از آن بھره زیادی برداشتند. فرماندهان آلمانی نسبت به امنیت Enigma بسیار حساس بودند، گرچه این حساسیت توسط کاربر سامانه، کمتر رعایت می‌شد. به عنوان نمونه، تنظیمات دستگاه باید هر روز تغییر پیدا می‌کرد، ولی این اتفاق نمی‌افتد. از سوی دیگر، حوالی یک ساعت خاص در طول روز پیامی مخابره می‌شد.

Turing تقریباً مطمئن بود که این پیام به منظور گزارش وضعیت آب و هوا است و حتماً کلمه Wetter (معادل آلمانی Wether) در آن بکار رفته است. پس عبارت Wetter می‌توانست برای او نقش یک crib را ایفا کند و او می‌توانست متن رمز معادل آن را داشته باشد. این رخداد می‌توانست کمک فراوانی به آن‌ها بکند، و می‌توانست به آن‌ها کمک کند تا حدس‌های غلط را از مساله حذف کنند و فضای کلید را کوچک‌تر کنند. برای مطالعه بیشتر در این زمینه به [۱، فصل دوم] مراجعه کنید.

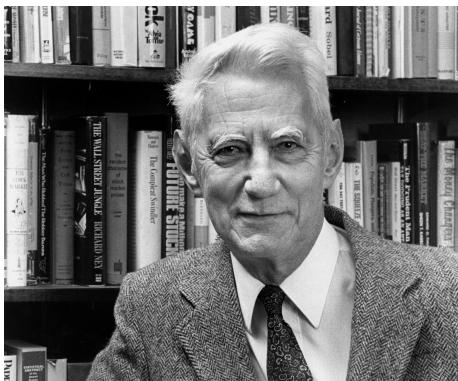
در حمله نوع سوم یا حمله با متن اصلی منتخب (Chosen Plaintext Attack)، فرض می‌شود که حمله‌گر به هر متن رمز و متن اصلی معادل آن دسترسی دارد، و تنها کلید را در اختیار ندارد. به عنوان نمونه فرض کنید که جاسوس به عنوان کاربر یک سامانه رمزگذاری در یک سازمان نفوذ کرده. او به کلید دسترسی ندارد، ولی می‌تواند به دلخواه متن اصلی در سامانه وارد کند و متن رمز معادل آن را مشاهده کند.

لازم به ذکر است، سامانه‌های رمزگذاری مدرن نظیر AES (Advanced Encryption Standard)، می‌بایست در برابر حملات حمله با متن اصلی منتخب نیز مقاوم باشند. نکته دیگری که در این مجال باید ذکر شود، این است که سامانه Vernum، به سادگی در برابر حملات نوع دوم و سوم شکسته می‌شود و هیچ‌گونه مقاومتی ندارد. می‌توانید بگویید چرا؟

حمله دیگری داریم که آن را حمله بر اساس حمله با متن رمز منتخب (Chosen Ciphertext Attack) می‌نامیم. در این نوع از حمله، حمله‌گر می‌تواند یک متن رمز دلخواه به مانند c را انتخاب کند، و متن اصلی معادل آن را داشته باشد. در حقیقت می‌توان گفت که این حمله به نوعی عکس حمله حمله با متن اصلی منتخب است.

به عنوان مثال، حمله تفاضلی شامیر-بیهام در سال ۱۹۹۱ به (Data Encryption Standard (DES)) از نوع از حملات محسوب می‌شود.

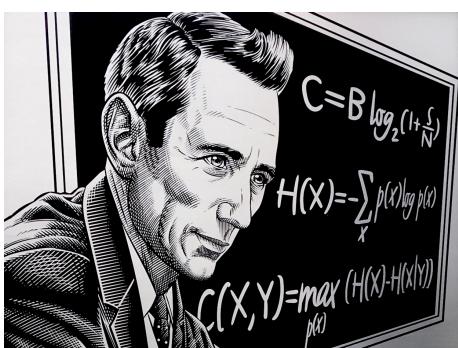
نگاه شانون به امنیت: سخن چهارم



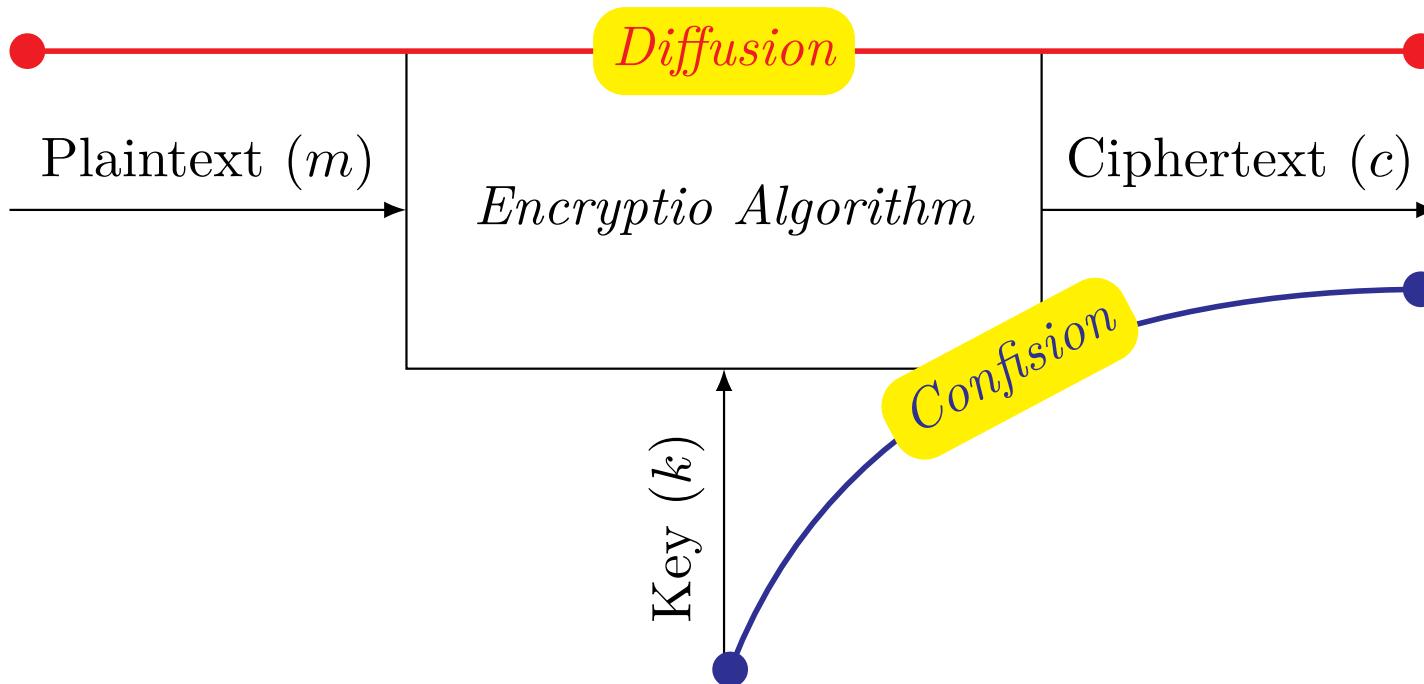
کلید باید به صورت مطمئن و غیرقابل دسترس توسط دشمن در اختیار گیرنده قرار گیرد.



کلید حتی المقدور کوتاه و ساده باید باشد، تا بتوان آن راه ساده‌تر ذخیره و یا ارسال کرد.



ممکن است رمز کردن موجب افزایش طول پیام گردد که باید از این کار پرهیز شود.



شانون گفت که یک سامانه رمزگذاری قوی باید دو ویژگی زیر را داشته باشد:

- گمراه‌کنندگی (Confusion): یعنی رابطه بین متن رمز و کلید تا حد امکان پیچیده باشد.
- انتشار (Diffusion) ساختار آماری متن آشکار بر روی حجم وسیعی از متنهای رمزشده ممکن پراکنده گردد.

ویژگی گمراه‌کنندگی، به معنای آن است که هر بیت از متن رمز، باید به چندین بخش از کلید وابسته باشد. به عبارت بهتر، این ویژگی سعی در محو ارتباط بین متن رمز و کلید را دارد. برای دستیابی به این ویژگی می‌بایست کاری کرد که اگر یک بیت از کلید را تغییر کند، تقریباً اکثر بیت‌های متن رمز تحت تاثیر قرار گیرد. الگوریتم‌های رمز جانشینی مثال بسیار ساده‌ای است که ویژگی گمراه‌کنندگی را ارضاء می‌کند. برای فهم بهتر مطلب، عبارت زیر را در نظر بگیرید:

This is an example.

با رمز مُستَوی با پارامتر $5 = b$ ، عبارت زیر را داریم:

Icjbjb fs hkflgeh.

اکنون اجازه دهید یک پارامتر از کلید را کمی تغییر دهیم. مثلاً $b = 4$ را به $b = 5$ تبدیل کنیم. نتیجه متن رمز به صورت زیر در خواهد آمد.

Hbia ia er gjekfdg.

همان طور که مشاهده می‌کنید، با یک تغییر کوچک در کلید، تقریباً کل متن رمز تحت تاثیر قرار گرفت.

اما انتشار، به معنای آن است که اگر یک بیت از متن اصلی تغییر کند، آن‌گاه تقریباً حدود نیمی از بیت‌های متن رمز تغییر کند، و بالعکس. یعنی اگر یک بیت از متن رمز تغییر کند، حدود نیمی از بیت‌های متن اصلی تغییر کند. هدف غایی از این ویژگی، مخفی کردن ارتباط آماری بین متن رمز و متن اصلی است.

ساختر آماری متن اصلی را بر روی حجم وسیعی از متن رمز پراکنده می‌کند. به عنوان مثالی از الگوریتم‌هایی که ویژگی انتشار را دارند، می‌توانند از الگوریتم Hill یاد کرد. در این الگوریتم، متن اصلی را به قطعه‌های n کاراکتری تقسیم‌بندی می‌کنیم. سپس به هر قطعه را به صورت یک بردار $n \times 1$ توصیف می‌شود. هر عدد در این بردار بیانگر شاخص کاراکتر مربوطه است. ما برای حرف A شاخص 0 و برای B شاخص 1 را در نظر گرفتیم و همین روند را تا حرف Z با شاخص 25 ادامه می‌دهیم. فرض کنید که $n = 3$ داریم و عبارت یک قطعه کلمه ACT

است. در ادامه این بردار در یک ماتریس $n \times n$ باید ضرب شود. این ماتریس را به صورت زیر در نظر بگیرید:

$$\begin{pmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{pmatrix}$$

اکنون برای بدست آوردن متن رمز کافی است که ماتریس یاد شده را در بردار عبارت CAT ضرب کنیم.

$$\begin{pmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{pmatrix} \begin{pmatrix} 2 \\ 0 \\ 19 \end{pmatrix} = \begin{pmatrix} 31 \\ 216 \\ 325 \end{pmatrix} \equiv \begin{pmatrix} 5 \\ 8 \\ 13 \end{pmatrix} \pmod{26}$$

این بدان معنا است که متن رمز به صورت FIN خواهد شد. اکنون برای این که درک کنیم که آیا الگوریتم رمز Hill

ویژگی انتشار را دارد، کافی است که یک تغییر کوچک در متن اصلی بدهیم. این بار تلاش می‌کنیم که به جای

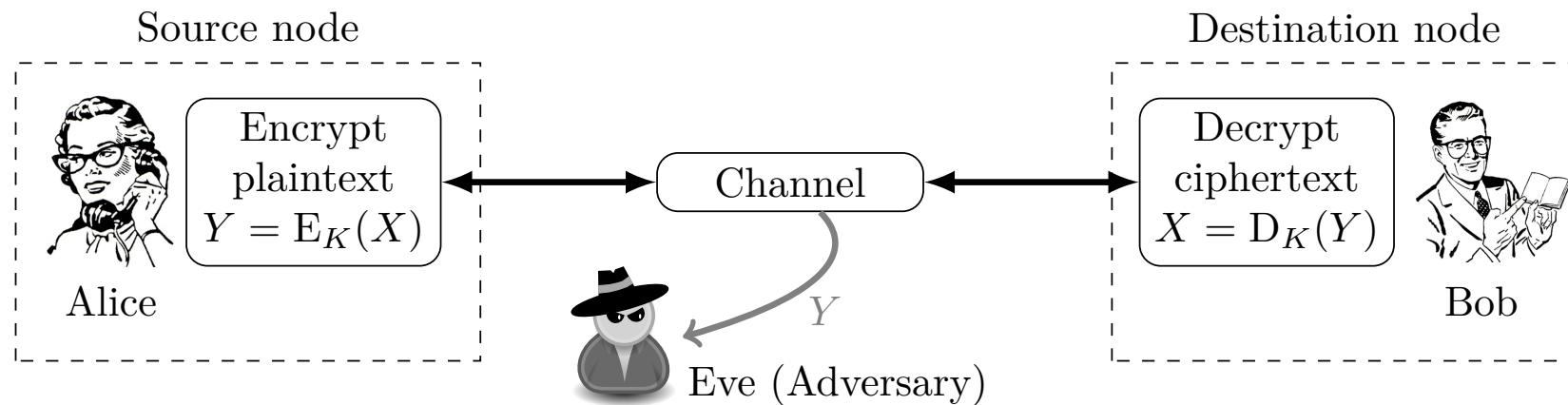
CAT عبارت CAS را ارسال کنیم. پس خواهیم داشت:

$$\begin{pmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{pmatrix} \begin{pmatrix} 2 \\ 0 \\ 18 \end{pmatrix} = \begin{pmatrix} 30 \\ 206 \\ 310 \end{pmatrix} \equiv \begin{pmatrix} 4 \\ 24 \\ 24 \end{pmatrix} \pmod{26}$$

که متن رمز معادل عبارت EYY خواهد شد. براحتی می‌توان مشاهده کرد که اندکی تغییر در متن اصلی تقریباً کل متن رمز را تحت تاثیر قرار داد. ذکر این نکته ضروری است که برای عملکرد موفق الگوریتم Hill، ماتریس موردنظر، باید با دقت طراحی شود.

الگو، پیشنهادی لبیر متنها،

الگوریتم کلید متقارن



در الگوریتم کلید متقارن (Symmetric Key Algorithm)، گیرنده و فرستنده از یک کلید مشترک برای رمزگذاری و رمزگشایی استفاده می‌کنند.

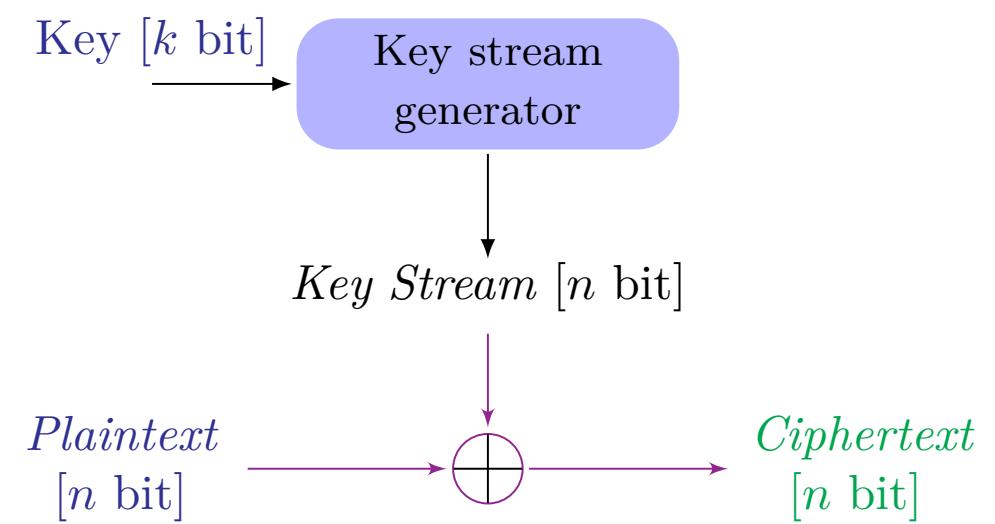
$$m = D_K(E_K(m))$$

همه الگوریتم‌های رمزنگاری قبل از دهه ۱۹۷۰ و همچنین الگوریتم‌های مشهوری به مانند AES, RC4, ...

الگوریتم کلید متقارن - یک دسته‌بندی کلی



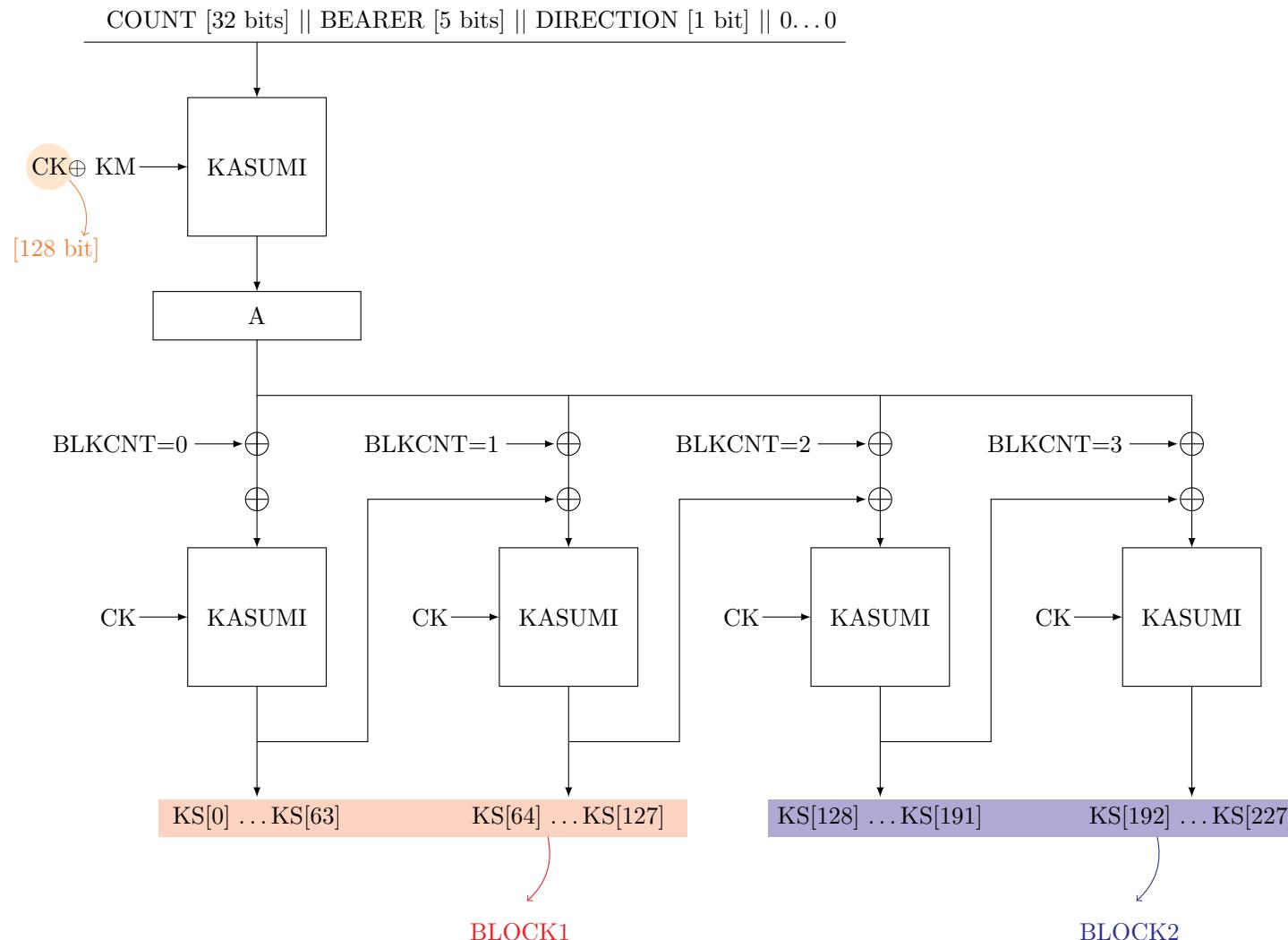
(ب) رمزنگاری بلوکی (Block Cipher)



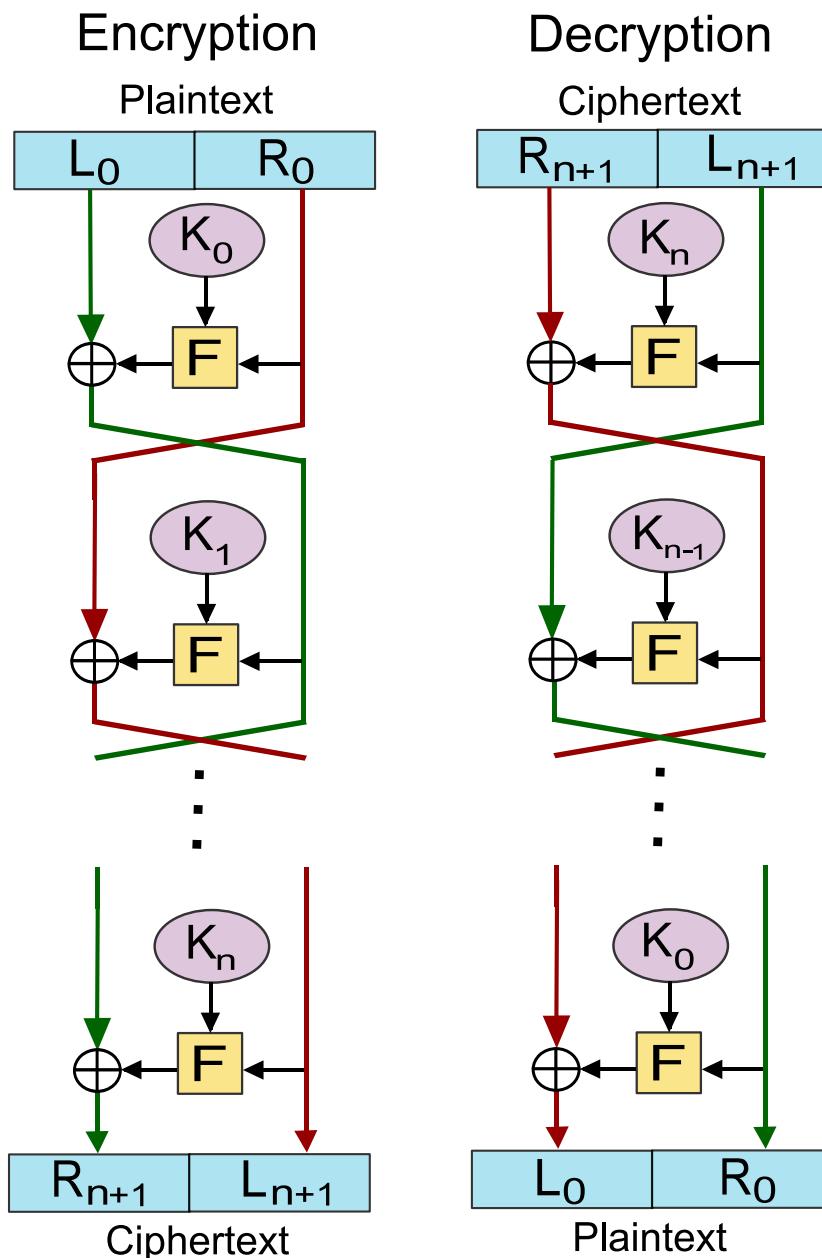
(آ) رمزنگاری جویباری (Stream Cipher)

مثالی از رمزنگاری جویباری (Stream Cipher) و تولیدکننده کلید

نمایی از الگوریتم f8 به عنوان تولیدکننده کلید در شبکه‌های تلفن همراه



الگوریتم کلید متقارن - رمزنگاری بلوکی (Block Cipher)



رمزنگاری بلوکی ایده‌آل، به صورت یک جانشینی کامل با طول n بیت با فضای 2^n ، اما ...

اندازه قالب‌ها 64,128,256 بیت.

شبکه فایستل (Feistel) از روش‌های معمول طراحی است.

$$L_n = R_{n-1}, \quad R_n = L_{n-1} \oplus F(R_{n-1}, K_n)$$

هر دور، شامل دو عملیات جانشینی و جایگشتی است.

DES (Data Encryption Standard) - معرفی

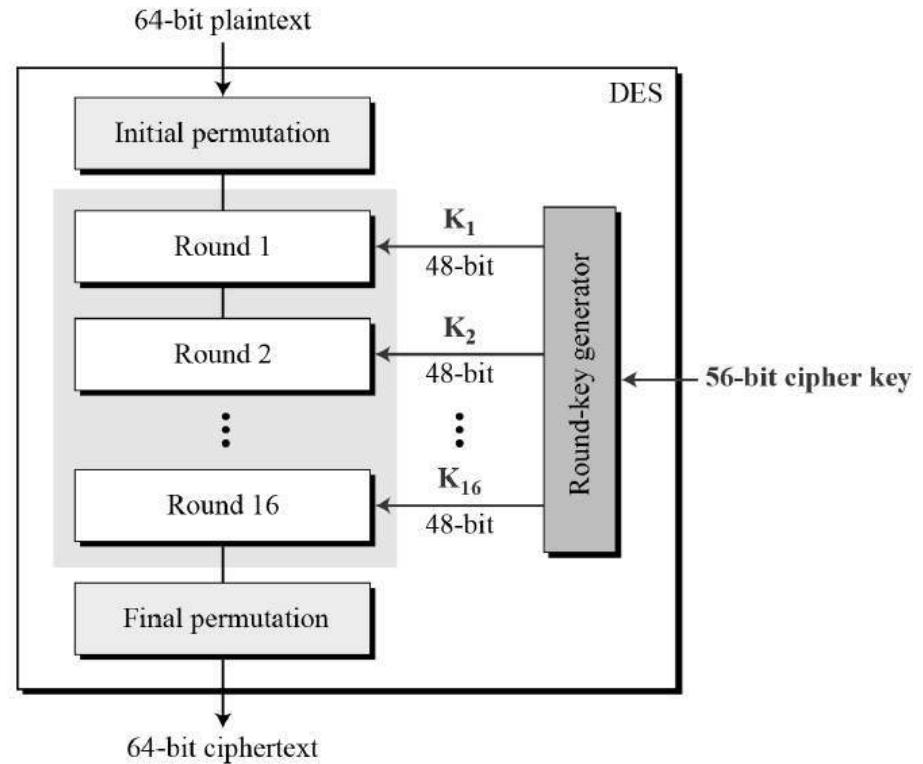
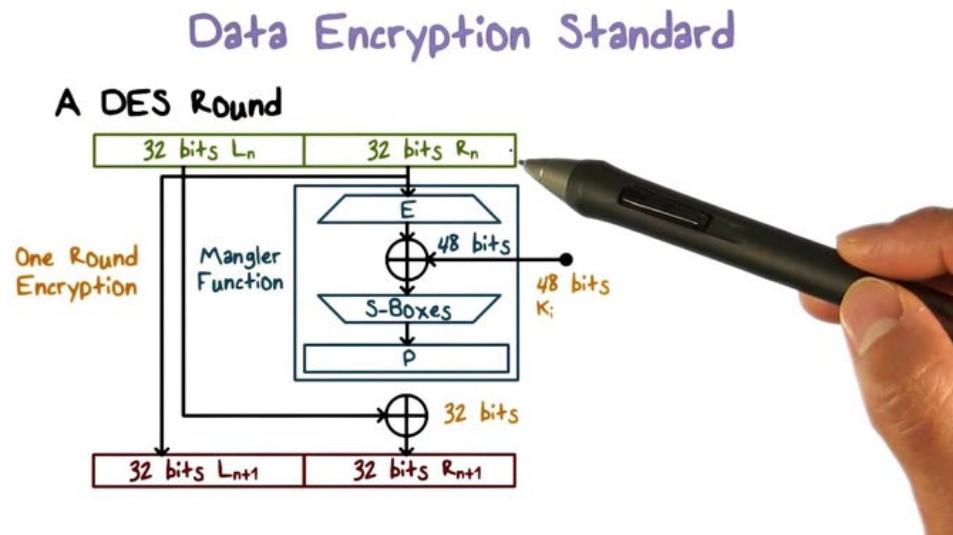
یک الگوریتم رمزگذاری متعلق به دهه ۱۹۷۰، که توسط IBM با همکاری Non-Standalone (NSA) توسعه یافته است.

بعد از شکسته شدن آن در سال ۲۰۰۱ استاندارد Advanced Encryption Standard (AES) جایگزین شد.

طراحی آن مبتنی بر شبکه‌های Feistel با ۱۶ دور بود، گرچه جزئیات الگوریتم تا مدت‌های زیادی توسط آمریکا منتشر نشد.

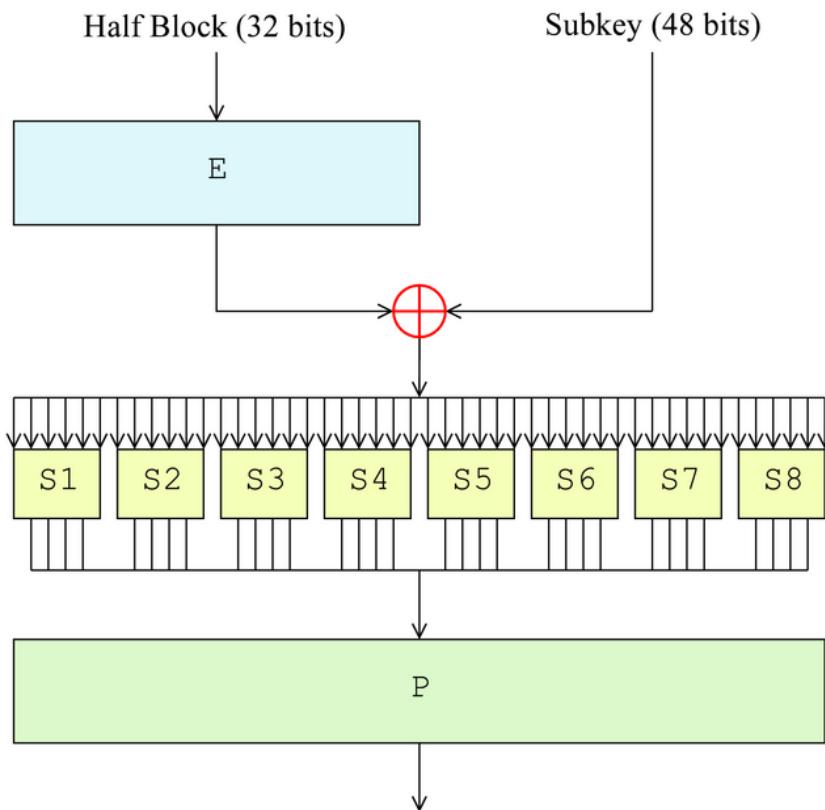


طراحی - DES



(آ) ساختار الگوریتم DES با ۱۶ دور. ورودی هر دور، خروجی ۶۴ (ب) ساختار هر دور در DES که مبتنی بر شبکه فایستل است. اصلی‌ترین بخش طراحی همان تابع F خواهد بود. بیتی مرحله قبلی و یک زیرکلید ۴۸ بیتی است.

تابع F - DES



تابع E با هدف بسط دادن ۳۲ بیت به ۴۸ بیت.

هشت S-Box که باعث غیرخطی شدن سامانه می‌شود، و در حقیقت عملیات جانشینی در این قسمت انجام می‌شود (گمراه‌کنندگی (Confusion)).

بخش P-Box: عملیات جایگشت را انجام می‌دهد (انتشار (Diffusion))

P-Box و S-Box - بخش DES

S ₅		Middle 4 bits of input															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Outer bits	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

شماره پنج در DES با ورودی شش بیتی و خروجی چهاربیتی S-Box

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	

نحوه کارکرد بخش P-Box

یک الگوریتم رمزگذاری است که در دهه ۱۹۷۰ توسط تیمی در IBM با کمک (National Security Association) توسعه یافت. این کار در حقیقت به منظور پاسخی به نیاز NIST (National Institute of Standards and Technology) و درخواست (National Security Association (NSA)، به منظور توسعه یک الگوریتم رمزگذاری استاندارد، بود [۲، Chapter 6]. گرچه حضور DES و عدم فاش شدن نقش او در طراحی DES، این شبکه را به ذهن‌ها تداعی کرد که NSA حتماً راه گریزی برای رمزگشایی DES داشتن کلید برای خود، قرار داده است.

برمبانای الگوریتمی به نام Lucifer که توسط Horst Feistel در IBM توسعه یافته بود، ولی هیچ‌گاه استاندارد نشده بود، طراحی شد. برای پذیرش Lucifer چندین تغییر مهم در آن ایجاد شد. یکی از این تغییرات این بود که کلید ۱۲۸ بیتی این الگوریتم به ۶۴ بیت تقلیل پیدا کرد. از دیگر این موارد، تغییر در طراحی S-Box‌ها (یکی از بخش‌های الگوریتم) بود. بعدها فاش شد که NSA از حملات تفاضلی (Differential Attack) که ذکر آن بعدها گفته خواهد شد، آگاهی داشته و این تغییرات به منظور امن کردن بیشتر الگوریتم نسبت به این گونه

حملات صورت گرفته بود. در کل، جزئیات طراحی و پیاده‌سازی DES تا مدت‌ها توسط دولت آمریکا مخفی نگه داشته شد. همین عامل سبب شد که افراد بسیاری به روند طراحی و همکاری NSA مشکوک باشند و همواره با دید تردید به این الگوریتم نگاه کنند [§§3.3.2، ۳].

DES در حقیقت یک رمزنگاری بلوکی مبتنی بر شبکه‌های Feistel است و بدین‌سان در دسته الگوریتم‌های کلید متقارن قرار می‌گیرد. طول ورودی متن اصلی برابر با ۶۴ بیت، و طول کلید آن نیز ۶۴ بیت است. البته باید دقت کرد که از ۶۴ بیت کلید، هشت بیت به عنوان Parity قرار داده شده، که این خود سبب می‌شود که طول واقعی کلید برابر با ۵۶ بیت گردد. برطبق سخنان شانون که پیشتر گذشت، طول متن رمز خروجی برابر با طول ورودی و همان ۶۴ بیت خواهد شد [Chapter 3، ۴].

DES مبتنی بر شبکه‌های Feistel است. ساختار کلی این الگوریتم در شکل زیر نشان داده شده است. دو بخش Initail Permutation و Final Permutation که عکس یکدیگر هستند، تنها عملیات جایگشت را انجام می‌دهند. در این میان دورهای الگوریتم وجود دارد. تعداد دورهای این الگوریتم برابر با ۱۶ دور، و طول زیرکلید

هر دور نیز برابر با 48 بیت در نظر گرفته شده است.

همان طور که می‌دانید یکی از مهم‌ترین بخش‌های طراحی رمزهای مبتنی بر شبکه‌های Feistel، بحث مربوط به طراحی تابع F است. در نخستین گام در تابع F، بخش E یا همان Expansion function وجود دارد. هدف از این بخش این است که ۳۲ بیت ورودی نیم قطعه متن را به ۴۸ بیت تبدیل کند، چرا که قرار است در گام بعدی این ۴۸ بیت با ۴۸ بیت زیرکلید دور، XOR شود. همان‌طور که در جدول زیر مشاهده می‌شود، برای رسیدن به این مقصود برحی از بیت‌ها به مانند بیت ۳۲ و ۲۸ و ۱۲ و ... دو بار تکرار می‌شود.

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

مهم‌ترین بخش الگوریتم DES را در هر دور باید S-Box و P-Box دانست. در حقیقت S-Box و P-Box یک رمز جانشینی و P-Box یک رمز جایگشتی است. این دو برای ما ویژگی‌های گمراه‌کنندگی و انتشار را که شانون در مقاله خود برای دست‌یابی به یک رمز قدرتمند ارایه کرد را به ارمغان می‌آورد.

S-Box‌ها یکی از مهم‌ترین عناصر بکار رفته در بسیاری از سامانه‌های رمز مبتنی بر رمزنگاری بلوکی است. در DES، هشت S-Box با شش بیت ورودی و چهار بیت خروجی، به ما غیرخطی شدن الگوریتم و ویژگی گمراه‌کنندگی را تضمین می‌دهد. به دلیل این‌که تنها بخش غیرخطی سامانه همین S-Box‌ها هستند، امنیت DES عملاً وابسته به این بخش است. تاکنون تحقیقات بسیاری در مورد دستیابی به یک S-Box مناسب صورت پذیرفته است که خارج از موضوعات این نوشتار محسوب می‌شود. شکل زیر نگاشت شش بیت به چهاربیت در S-Box پنجم الگوریتم DES را به ما نشان می‌دهد. براحتی می‌توان دید که دنباله شش بیتی 011011 به 1001، نگاشت می‌شود. اطلاعات مربوط به مابقی S-Box‌ها را می‌توانید در [این پیوند](#) پیدا کنید.

S ₅		Middle 4 bits of input															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Outer bits	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

بخش بعدی که در DES به آن برخورد می‌کنیم، بخش P-Box است. اگر جایگاه بیت‌هارا از یک تا ۳۲ شماره‌گذاری کنیم، نحوه جایگشت (Permutation) به صورت جدول زیر خواهد بود.

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



- آژانس امنیت ملی آمریکا (National Security Agency)، یک سازمان اطلاعاتی در زیرمجموعه وزارت دفاع آمریکا است که در سال ۱۹۵۲ تشکیل شد.
- با وظایفی نظیر شنود الکترونیک، نظارت جهانی، گردآوری و پردازش اطلاعات، حفاظت از شبکه‌های ارتباطی و سامانه‌های اطلاعاتی ایالات متحده آمریکا.
- بر عکس آژانس اطلاعات مرکزی آمریکا (Central Intelligence Agency)، اطلاعات را توسط مامور اطلاعاتی جمع‌آوری نمی‌کنند.



مؤسسه ملی فناوری و استانداردها یک موسسه دولتی آمریکا است، که در سال ۱۹۰۱ تشكیل شد، وزیرنظر وزارت بازرگانی آمریکا کار می‌کند، با هدف نوآوری و ایجاد رقابت،

این نهاد، مسئول وضع استاندارد برای امنیت اطلاعات همه عملیات و دارایی‌های سازمان‌های ایالات متحده آمریکا است.

ایالات متحده آمریکا به طور مشخص دارای ۱۷ سازمان اطلاعاتی است که از آن با عنوان جامعه اطلاعاتی ایالات متحده آمریکا (United States Intelligence Community) یاد می‌کنیم. ریاست تمامی این سازمان‌ها بر عهده یاست آن بر عهده اداره کننده اطلاعات ملی (Director of National Intelligence) است، که از مقامات دولت ایالات متحده آمریکا و تحت هدایت و کنترل رئیس جمهور آمریکا کار می‌کند. چند مورد از این سازمان‌های اطلاعاتی به شرح زیر است:

- سازمان مرکزی اطلاعات آمریکا (Central Intelligence Agency)
- سازمان امنیت ملی آمریکا (National Security Agency)
- سازمان اطلاعات دفاعی (Defense Intelligence Agency)
- دفتر تروریسم و اطلاعات مالی (Office of Terrorism and Financial Intelligence)

در می ۲۰۱۳، ادوارد اسنودن از محل کارش که متعلق به سازمان امنیت ملی آمریکا در هاوایی بود به هنگ کنگ پرواز کرد، و هزاران مدرک طبقه‌بندی شده را در اختیار روزنامه‌نگاران گذاشت. او خیلی زود، مورد توجه

خبرگزاری‌ها قرار گرفت. این اسناد نشان‌دهنده جزئیات چگونگی تلاش گسترده آمریکا با همکاری انگلیس، کانادا، استرالیا و نیوزیلندلس برای نظارت و جاسوسی گسترده، به ویژه از طریق اینترنت است. برخی از اسناد منتشر شده نشان می‌دهد که ارتباطات در مقیاسی بسیار وسیع، رهگیری و شنود می‌شود، حتی در برابر شهروندان خود آمریکا و متحدانش نظیر کشورهای عضو ناتو و کشورهای اتحادیه اروپا.

طبق اطلاعات گاردین، بیشترین مقدار اطلاعات از ایران جمع‌آوری شده. به‌گونه‌ای که در پروژه خبرچین بیکران (Boundless Informant)، تنها در مارس ۲۰۱۳ تعداد ۱۴ میلیارد گزارش از ایران وجود دارد. اسنودن در مصاحبه‌ای گفت که NSA و اسرائیل به کمک یکدیگر بدافزار استاکسنت را برای ضربه زدن به تأسیسات هسته‌ای ایران و جاسوسی از آن تولید کردند.

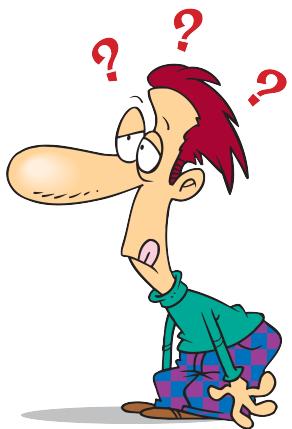
برخی از پروژه‌های NSA به شرح زیر است:

- خبرچین بیکران (Boundless Informant): یک سامانه تحلیل و نمایش کلان داده (Big Data)، که به خلاصه‌سازی اطلاعات جمع‌آوری شده کمک فراوانی می‌کرد.

● منشور (Prism): هدف آن جمع‌آوری اطلاعات تبادل شده در اینترنت کل دنیا با کمک شرکت‌های اینترنتی Microsoft, Google, Yahoo, Facebook, ICP (Internet Content Provider) هایی نظیر YouTube, Skype, Apple, PalTalk, AOL (Routing) از سمت آمریکا و این‌که بسیاری از زیرساخت‌های ارتباطی جهان در آمریکا وجود دارد، بسیاری از ارتباطات اینترنت جهانی از آمریکا می‌گذرد.

● XKeyscore: در ۲۰۱۴ ژانویه ادوارد اسنودن در یک گفتگوی تلویزیونی در مورد این پروژه گفت: شما می‌توانید ایمیل‌های هر کسی را که آدرس ایمیل دارد بخوانید. هر وبگاه، ترافیک ورودی به آن و خروجی از آن. هر رایانه‌ای که کسی از آن استفاده می‌کند. هر لپتاپی که از هر جای جهان به جای دیگر می‌رود می‌توانید تک‌تک مردم را علامت‌گذاری کنید. برای نمونه شما در یک ابرشرکت آلمانی کار می‌کنید و من می‌خواهم به شبکه آن دسترسی داشته باشم. من می‌توانم نام کاربری شما در فرمی در جایی از وبگاهی که به آن می‌روید، و نام واقعی شما را شناسایی کنم. من می‌توانم همراهی شما با دوستانتان را ردگیری کنم و

ردپای دیجیتالی بسازم که رفتارهای ویژه شما را نشان می‌دهد. یعنی هرجایی در جهان که می‌روید حتی اگر هویت یا حضور برخط خود را پنهان کنید.



قطعاً یک روش غیرهشمندانه برای شکستن DES همان Bruteforce است، که در آن باید 2^{56} حالت فضای کلید را چک کرد.

با یک کارت گرافیکی NVidia A100 با قدرت PFLOPS 10 خواهیم داشت:

$$\text{Time} = \frac{2^{56}}{10 \times 10^{15}} = 7.2 \text{ sec}$$

روش‌های رمزشکنی را می‌توان خیلی خوب توزیع کرد. در ژانویه ۱۹۹۹ چندین هزار کامپیوتر با مشارکت همدیگر توانستند DES را در طول ۲۲ ساعت و ۱۵ دقیقه بشکنند (www.distributed.net).

چکار کنیم که روند چک کردن طولانی نشود؟



﴿ حملاتی که می‌تواند در زمانی کمتر از Brute-force رمز را بشکند.

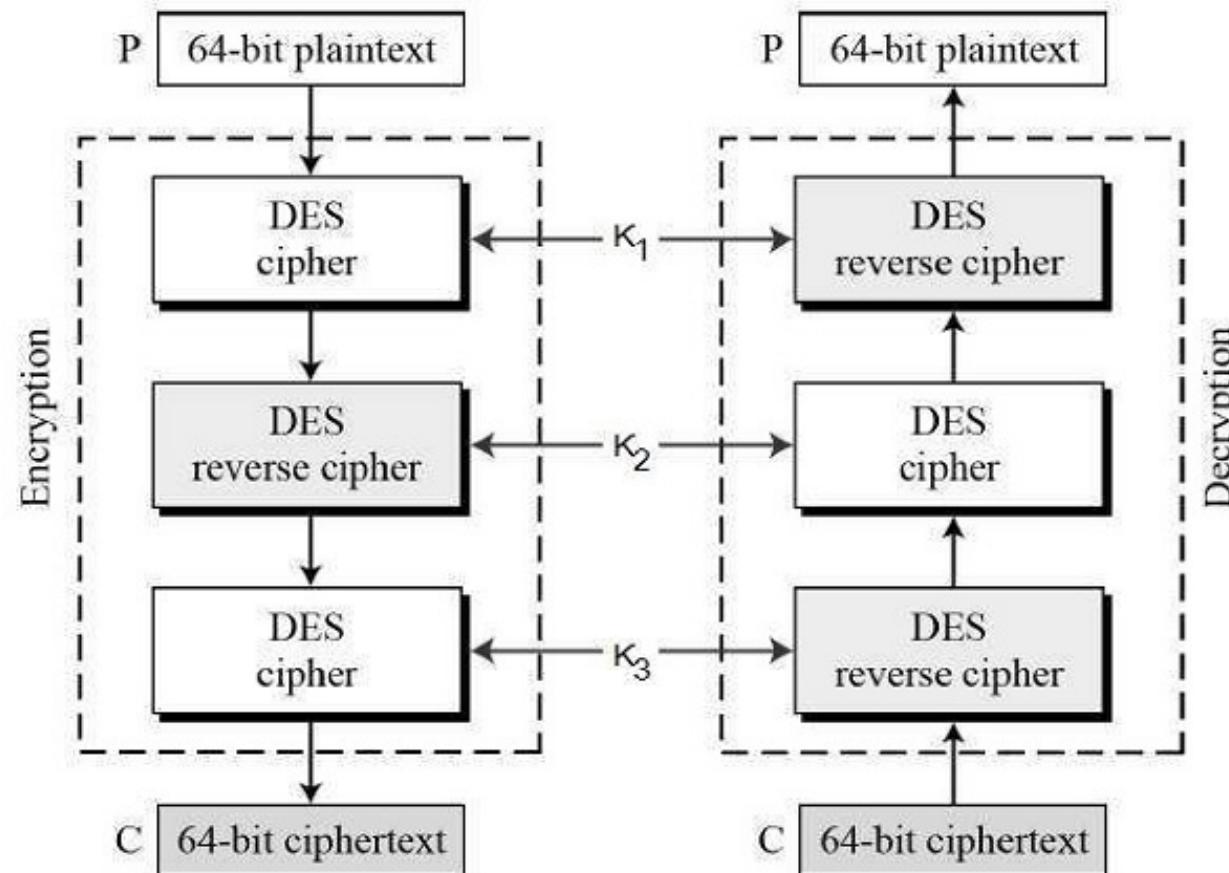
- حمله تفاضلی (Differential Attack)
- حمله خطی (Linear Attack)
- حمله همبستگی (Correlation Attack)



﴿ حمله تفاضلی از نوع حمله با متن رمز منتخب (Chosen Ciphertext Attack) است،
که توانست در زمان $^{37}2^{37}$ و با داشتن 247 متن اصلی، الگوریتم DES را بشکند. ایده اصلی
این بود که توزیع $m_1 \oplus m_2$ خاصی که خروجی آن $c_1 \oplus c_2$ می‌شود، اطلاعاتی را در مورد
کلید لو دهد.

3-DES

افزایش قدرت DES با استفاده از چندین بلوک DES 



DES در زمان خودش، الگوریتمی سریع و کارآمد محسوب می‌شد. اما به مرور و با قوی‌ترین شدن کامپیوترها و سریع‌تر شدن محاسبات دیگر پاسخ‌گوی نیازها نبود و در نهایت در سال ۲۰۰۱، با الگوریتم AES جایگزین شد. به عنوان مثال، طول کلید ۵۶ بیتی آن، طول کلید کوچکی محسوب می‌شود، و ما می‌دانیم که^{۵۶} ۲ محاسبه در حمله Brute-force، را می‌توانیم امروزه انجام دهیم. نکته جالب در این میان است که عملیات تحلیل رمز، عملیاتی است که برای براحتی می‌توان آن را موازی ساخت. به همین دلیل به جای متمرکز شدن بر یک ابرکامپیوتر برای بکارگیری در تحلیل رمز، می‌توان محاسبات را بین هزاران کامپیوتر کوچک تقسیم نمود.

از دیدگاه هوشمندانه، حملات بسیاری تاکنون به DES انجام پذیرفته است، از قبیل حمله تفاضلی (Differential Attack)، حمله خطی (Linear Attack)، حمله همبستگی (Correlation Attack)، حمله کدینگ صوری، حمله تقارن در کلیدهای مکمل و این‌ها را به عنوان حملات هوشمندانه‌ای می‌توان برشمرد که می‌توانند در زمان کمتری از Brute-force، برای ما عملیات رمزشکنی را انجام دهند.

و Eli Biham در [۵] حمله تفاضلی، را ارایه دادند، و توسط آن توانستند در زمان اجرای^{۳۷} Adi Shamir

به نسبت⁵⁶ 2 جستجوی جامع، الگوریتم DES را بشکنند. این حمله از نوع حمله با متن رمز منتخب (Chosen Plaintext Attack) است. درواقع جفت متون رمز، به نحوی انتخاب می‌شوند که XOR متون اصلی متناظر، مقدار خاصی شود. آن‌ها ادعا کردند که توزیع $m_1 \oplus m_2$ خاصی که خروجی آن $c_1 \oplus c_2$ می‌شود، اطلاعاتی را در مورد کلید لو دهد. گرچه برای اجرای این حمله نیاز است تا⁴⁷ 2 متن اصلی منتخب (Chosen Plaintext) تولید شود، تا بتوان³⁶ 2 متن رمز مناسب برای حمله انتخاب گردد، که این مورد چندان از لحاظ عملی قابل دستیابی نیست و عملاً این حمله را به صورت تئوری باقی نگه می‌دارد. جالب این است که با انتشار این مقاله، NSA فاش کرد که در زمان طراحی DES از این نوع حمله آگاه بوده، و تا حدی الگوریتم را نسبت به این حمله مقاوم ساخته است.

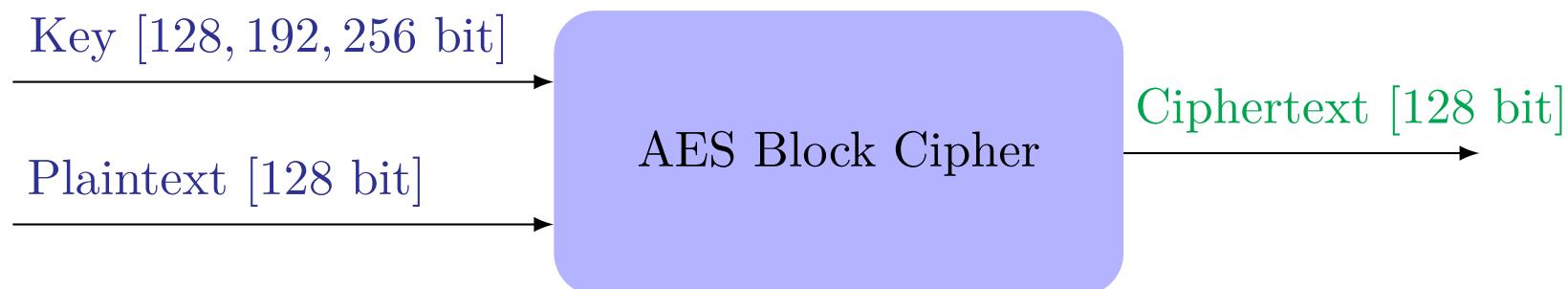
Matsui در [۶]، ایده حمله خطی را مطرح کرد و ادعا کرد که می‌تواند با بهره‌گیری از⁴³ 2 متن اصلی معلوم (Known Plaintext)، الگوریتم DES را بشکنند. این شیوه نیز باز از لحاظ عملی چندان قابل دستیابی نیست. به همین دلیل ما باید به همان Brute-force امید داشته باشیم.

Name	Time	Ciphertext Only	Known Plaintext	Chosen Plaintext
Brute-force	2^{56}	0	1	0
Differential Attack [1992]	2^{37}	0	0	2^{47}
Linear Attack [1993]	2^{43}	0	2^{43}	0

بالاخره باید پذیرفت که مهمترین چالش در DES، طول کلید کوتاه آن است. برای جبران این ضعف، ایده‌هایی نظیر DES-2 و DES-3 مطرح شد، تا بتواند برای ما امنیت بیشتری را به ارمغان بیاورد. یعنی به نوعی مبنای کار همان DES باشد ولی از چندین بلوک DES به صورت سری استفاده کنیم.

در سال ۱۹۹۷ NIST یک مسابقه برای استانداردسازی یک سامانه جدید رمزگاری، که بتواند برای کاربردهای عمومی برگزار می‌کند.

- در مرحله اول، تعداد ۱۵ پیشنهاد و در مرحله بعدی تنها پنج پیشنهاد پذیرفته شد.
- در نهایت، در دوم اکتبر ۲۰۰۰، الگوریتم Rijndael (['rɛində:l]) به عنوان برنده انتخاب شد.



Rijndael - الگوریتم AES



Vincent Rijmen و Joan Daemen

Rijndael مبتنی بر شبکه‌های Feistel نیست.

تعداد دور بر حسب طول کلید برابر با 12, 10, 14 است.

در دوم ژانویه ۱۹۹۷، NIST اعلام کرد که قصد دارد جایگزینی برای DES به نام AES انتخاب کند ([این پیوند](#)). یک الگوریتم رمز طبقه‌بندی نشده که طراحی آن به طور کامل مشخص باشد و بتواند داده‌ها را قرن‌ها محافظت کند. همگی الگوریتم‌ها می‌باشند مبتنی بر رمزنگاری بلوکی، با ورودی ۱۲۸ بیتی و طول کلید ۱۲۸، ۱۹۲ و ۲۵۶ بیتی، طراحی شوند. البته پارامترهای دیگری نظیر کارایی، سادگی، انعطاف‌پذیری، سهولت پیاده‌سازی سخت‌افزاری و نرم‌افزاری، به عنوان معیارهای مسابقه در نظر گرفته شد.

در طول نه ماه پیشنهادات بسیاری ارایه شد. NIST دو کنفرانس در سال‌های ۱۹۹۸ و ۱۹۹۹ برگزار کرد. در مرحله اول ۱۵ الگوریتم و در مرحله دوم، پنج گزینه به عنوان کاندیدای نهایی انتخاب شدند. در کنفرانس AES3 که در سال ۲۰۰۰ برگزار شد، بررسی‌های نهایی انجام شد و سرانجام در دوم اکتبر ۲۰۰۰، الگوریتم Rijndael عنوان برنده مسابقه اعلام گشت.

سوال اول: یکی از مهمترین حملات به DES، حمله تفاضلی است که توسط Adi Shamir و Eli Biham در دهه ۱۹۹۰ مطرح شد [۵]. در مورد این حمله تحقیق کنید و نحوه این حمله را با یک مثال ساده شده DES بیان کنید. مثلا با DES سه دور یا شش دور.

سوال دوم: در زبان C++ یا Python یک پیام را با الگوریتم AES، رمزگذاری یا رمزگشایی کنید. در ضمن نحوه کارکرد AES را به طور خلاصه بیان کنید.

- [1] M. Stamp and R. Low. *Applied Cryptanalysis: Breaking Ciphers in the Real World*. IEEE Press, Wiley, 2007.
- [2] S. Singh. *The Code Book: The Secrets Behind Codebreaking*. Random House Children's Books, 2002.
- [3] M. Stamp. *Information Security: Principles and Practice*. Wiley, 2005.
- [4] C. Paar and J. Pelzl. *Understanding Cryptography: A Textbook for Students and Practitioners*. Springer Berlin Heidelberg, 2009.
- [5] E. Biham and A. Shamir, “Differential cryptanalysis of the full 16-round des,” in *Annual international cryptology conference*, pp.487–496, Springer, 1992.
- [6] M. Matsui, “Linear cryptanalysis method for des cipher,” in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp.386–397, Springer, 1993.

فهرست اختصارات

A

AES Advanced Encryption Standard

C

CPU Central Processing Unit

D

DES Data Encryption Standard

G

GPU Graphics Processing Unit

GSM Global System for Mobile Communication

I

ICP Internet Content Provider

N

NIST National Institute of Standards and Technology

NSA Non-Standalone

U

UE User Equipment

W

WEP Wired Equivalent Privacy

واژه‌نامه انگلیسی به فارسی

Block قطعه A

Block Cipher رمزگاری بلوکی Affine Cipher رمز مُستَوی

Attack حمله

C حمله‌گر

Chosen Ciphertext Attack حمله با متن رمز منتخب

Chosen Plaintext B متن اصلی منتخب

Chosen Plaintext حمله با متن اصلی منتخب Big Data کلان داده

Attack

Cryptology	رمزنگاری	Ciphering رمزشناسی
D	Ciphertext	متن رمز
	Ciphertext Only	حمله براساس فقط متن رمز شده ..
Deciphering	رمزگشایی	Attack
Decryption	رمزگشایی	امنیت محاسباتی
Destination	مقصد ..	پیچیدگی
Differential Attack	حمله تفاضلی ..	گمراه کندگی
Diffusion	انتشار ..	محرمانگی
		Correlation Attack
		تحلیل رمز
		Cryptanalysis
		Cryptography

K E

Key Sequence	دنباله کلید	Eavesdropping	شنود....
Key Space	فضای کلید	Enciphering	رمزگذاری.....
Known Plaintext	متن اصلی معلوم	Encryption	رمزگذاری
Known Plaintext Attack	حمله با متن اصلی معلوم		

I

L	Index	شاخص
Linear Attack	حمله خطی	نهان‌سازی اطلاعات
	Information Hiding	نظریه اطلاعات

M

R

پیام Message

تکالفبایی Monoalphabetic

بسیربابی Routing

P

S

کارایی Performance

جایگشت Permutation

متن اصلی Plain Text

متن اصلی Plaintext

چندالفبایی Polyalphabetic

User	کاربر	Steganalysis	نهان کاوی
		Steganography	نهان نگاری
W		Stream Cipher	رمزنگاری جویباری
Watermarking	نشان گذاری		
	T		
		Throughput	گذردهی
		Transposition Cipher	رمز جایگشتی
	U		
		UnconditionalSecurity	امنیت بدون شرط

واژه‌نامه فارسی به انگلیسی

ب

Realtime الگوریتم کلید متقارن . Symmetric Key Algorithm . بی‌درنگ ..

امنیت Security

امنیت بدون شرط UnconditionalSecurity

پ

Message پیام

امنیت محاسباتی Computational Security

Complexity پیچیدگی

انتشار Diffusion

ت

Attack	حمله ..	Cryptanalysis	تحليل رمز
Known Plaintext Attack	حمله با متن اصلی معلوم	Monoalphabetic	تک الفبایی
Chosen Plaintext	حمله با متن اصلی منتخب ..		

Attack

Chosen Ciphertext Attack	حمله با متن رمز منتخب	جایگشت
Ciphertext Only ..	حمله براساس فقط متن رمزشده ..	Permutation

Attack

Differential Attack	حمله تفاضلی ..	ج ..
Linear Attack	حمله خطی ..	چند الفبایی
Correlation Attack	حمله همبستگی ..	

Cryptology	رمزناسی	Attacker	حمله‌گر
Ciphering	رمزگذاری		
Enciphering	رمزگذاری		
Encryption	رمزگذاری		دنباله کلید
Decryption	رمزگشایی	Key Sequence	
Cryptography	رمزنگاری		
Block Cipher	رمزنگاری بلوکی		ر
Stream Cipher	رمزنگاری جویباری	Substitution Cipher	رمز جانشینی
		Transposition Cipher	رمز جایگشتی
		Affine Cipher	رمز مُستَوی

ش

ک

Performance	کارایی	Index	شاخص
User	کاربر	Eavesdropping	شنود
Big Data	کلان داده		

ف

فضای کلید

گ سpace

Throughput	گذردهی
Confusion	گمراه کندگی

ق

قطعه

Block

Watermarking	Plain Text	نشانگذاری
Information Theory	Known Plaintext	نظریه اطلاعات
Information Hiding	Chosen Plaintext	نهانسازی اطلاعات
Steganalysis	Ciphertext	نهانکاوی
Steganography	Confidentiality	محرمانگی
	Routing	مسیریابی
	Destination	مقصد
	Source	منبع