جامعة حماة

الكلية التطبيقية

قسم تقنيات الحاسوب

- **المقرر: أمن المعلومات**
- **المحاضرة : الثالثة**

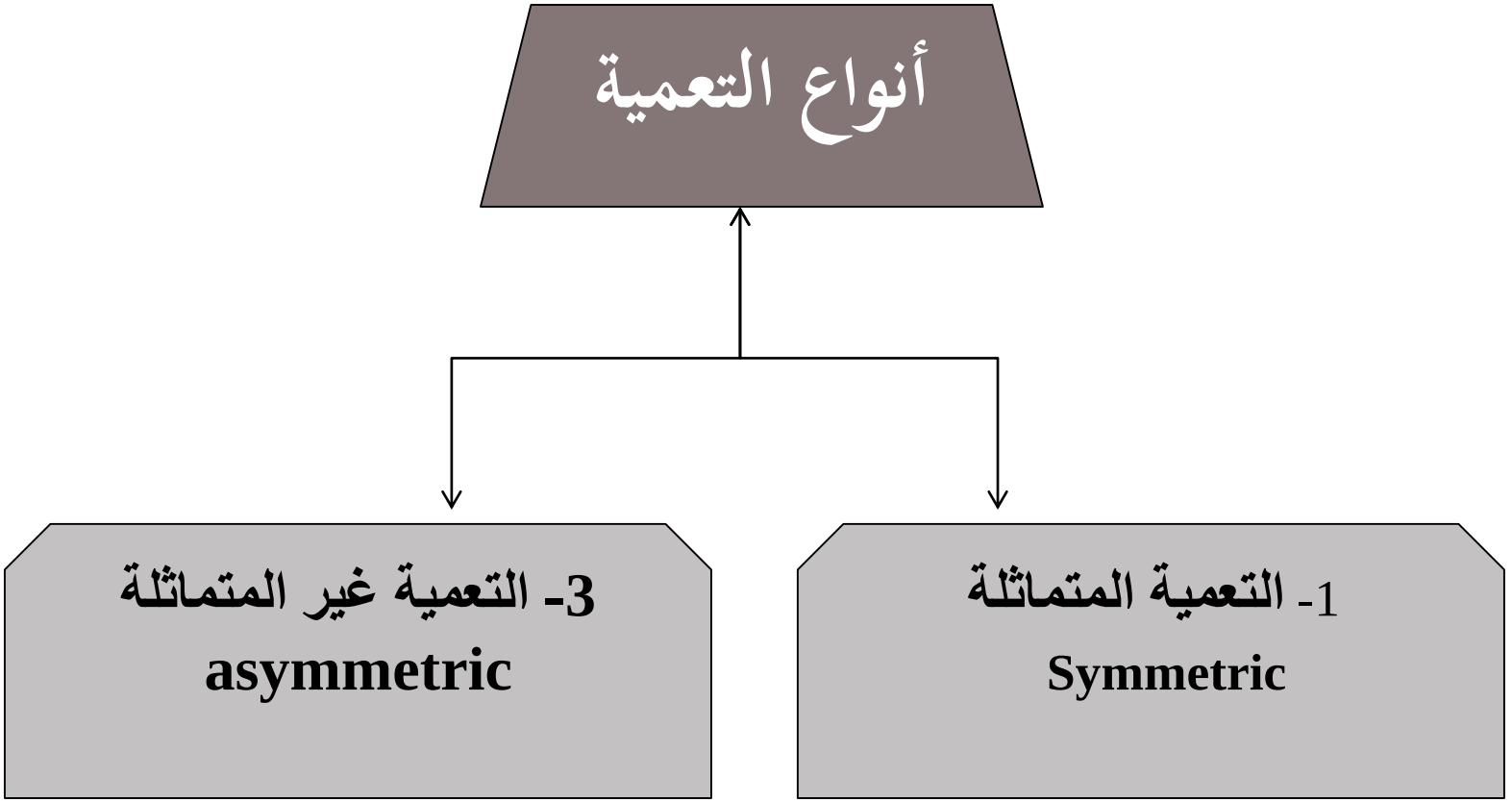
Dr. Mohammed AL-Mohammed

Chapter 3

التعمية وأمن المعلومات

مبادئ وتطبيقات

علم التعمية Cryptography



علم التعمية Cryptography

أنواع التعمية

1- التعمية المتماثلة

Symmetric Cryptography

علم التعمية Cryptography

1- نظام التعمية المتناظر **Symmetric Cipher Model:** وهو نوع من التعمية يكون فيه كل من المرسل والمستقبل لديهم نفس المفتاح "Single_Key".

يسمى أيضاً التشفير التقليدي (Conventional Cryptography):

وهو يستخدم مفتاح واحد لعملية التشفير وفك التشفير للبيانات. ويعتمد هذا النوع من التشفير على سرية المفتاح المستخدم .حيث أن الشخص الذي يملك المفتاح بإمكانه فك التشفير وقراءة محتوى الرسائل أو الملفات. أما الثغرة الكبيرة في هذا النوع من التشفير فكانت تكمن في تبادُل المفتاح السري دون أمان

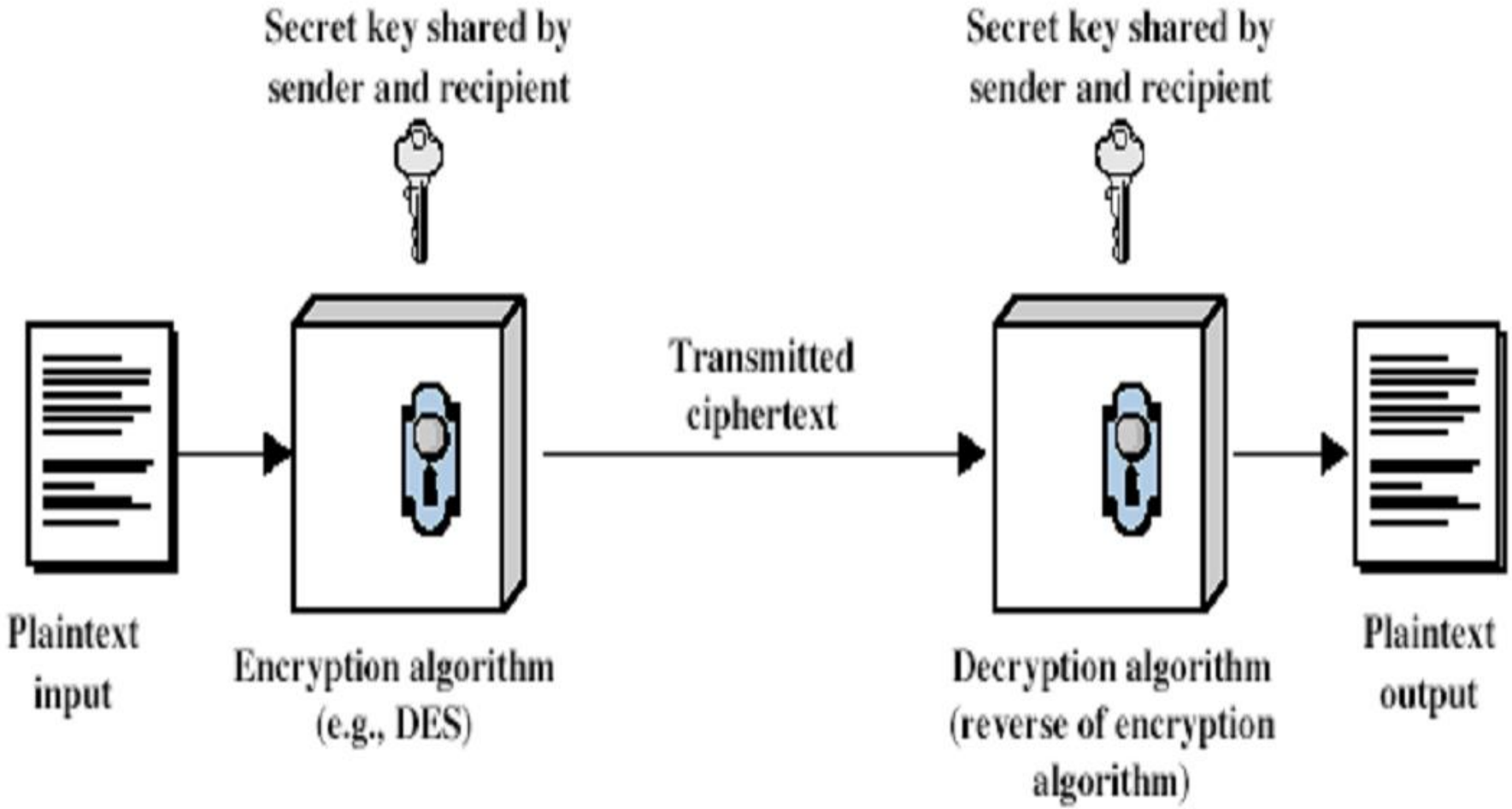
علم التعمية Cryptography

المتطلبات الأساسية لاستخدام التشفير الآمن Requirement of Encryption:

- وجود خوارزمية قوية للتشفير : بحيث إذا تم كشف الخوارزمية وكشف النص لا يمكن الوصول إلى النص الصريح ولا يمكن الوصول إلى المفتاح .
- كل من المرسل والمستقبل يجب أن يكون لديهم نسخ من المفتاح السري بشكل آمن ويحفظ لديهم بسرية تامة.

$$Encryption: C = E_k(P)$$
$$Decryption: P = D_k(C)$$
$$D_k(E_k(x)) = E_k(D_k(x)) = x$$

علم التعمية Cryptography



أنواع التعمية

1- التعمية المتماثلة
Symmetric

- **(التقنيات الكلاسيكية)**

معميات الإعاضة Substitution ciphers

معميات إبدال الموقع Transposition ciphers

(التقنيات الحديثة)

- معيار تعمية المعطيات Data Encryption Standard (DES)
- الخوارزمية العالمية لتعمية المعطيات (IDEA): International Data Encryption Algorithm
- معيار تعمية المعطيات الثلاثي Triple -DES (3DES)
- **المعيار AES**

علم التعمية Cryptography

- من أشهر خوارزميات منظومات التعمية التقليدية أوالمتماثلة

Caesar Cipher, DES, 3DES, IDEA,Towfish,RC4, Rijndael وجميعها تستخدم المعيار الأمريكي في التشفير. تدعى هذه الخوارزميات بخوارزميات التشفير متناظرة المفاتيح لكوننا نستخدم مفتاح تشفير واحد لعمليتي التشفيرencryption-key وفك التشفيرdecryption-key.

3- التعمية غير المتماثلة
aSymmetric

علم التعمية Cryptography

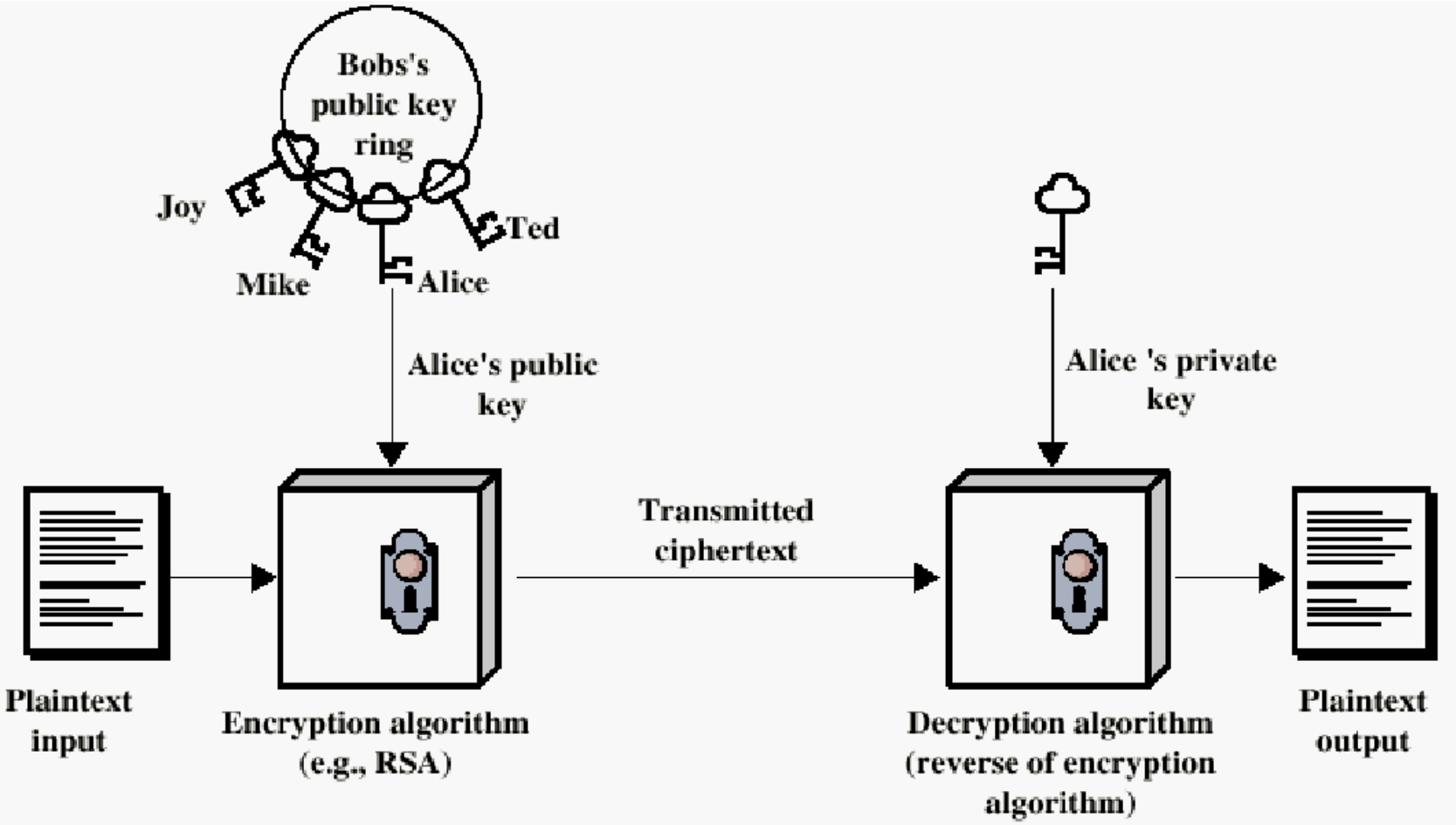
2- **نظام التعمية غير المتناظر Asymmetric Cipher Model** :كل من المرسل والمستقبل لديهما مفتاحين مختلفين واحد خاص بالمرسل وآخر مختلف للمستقبل ولا يعرف كل واحد منها مفتاح الآخر

أو التشفير اللا متماثل Asymmetric Cryptography

ويعرف بتشفير المفتاح العام (العمومي) Public Key Cryptography :

والتي يختلف فيها مفتاح التشفير عن مفتاح فك التشفير أي أن هنالك مفتاحين: مفتاح خاص (private key) ومفتاح عام (public key). المفتاح العام يوزع على جميع الناس أما المفتاح الخاص فهو شخصي يحتفظ به صاحبه ولا يرسله لأحد, تم تطوير هذا النظام في السبعينات في بريطانيا وكان استخدامه حكراً على قطاعات معينة من الحكومة, حيث أن المفتاح العام هو لتشفير الرسائل والمفتاح الخاص لفك تشفير الرسائل. وأي رسالة أو ملف يتم تشفيره بالمفتاح العام لا يمكن فك تشفيره إلا بالمفتاح الخاص والعكس صحيح. فالتشفير اللا متماثل جاء حلاً لمشكلة التوزيع غير الآمن للمفاتيح في التشفير المتماثل.

علم التعمية Cryptography



علم التعمية Cryptography

- يعد تطور تعمية المفتاح العمومي الأعظم ولربما كان الثورة الحقيقية الوحيدة في تاريخ علم التعمية بأكمله وتقدم تعمية المفتاح العمومي تحولاً ثورياً عن كل ما سبقها. وذلك لسببين، أولهما أن خوارزميات تعمية المفتاح العمومي مبنية على توابع رياضية إضافة إلى الإعاضة والإبدال.

علم التعمية Cryptography

- تدعى منظومات تعمية المفتاح العمومي بالمنظومات غير المتماثلة asymmetricويكون فيها مفتاح التعمية encryption key مغايراً لمفتاح الإظهارdecryption key وبالتالي يمكن الإعلان عن أحدهما (المفتاح العمومي public-key) دون الخوف على الآخر(المفتاح الخصوصيprivate-key) .
- ويتم الحصول على المفتاح العمومي من المفتاح الخصوصي بواسطة تحويل باتجاه واحد فقط One-way Transformationبينما يجب أن يكون من الصعوبة بمكان تحديد المفتاح الخصوصي من المفتاح العمومي

علم التعمية Cryptography

أنواع التعمية

3- التعمية غير المتماثلة
aSymmetric

⟺

منظومات تعمية المفتاح العمومي Public-Key Cryptographic Systems

أشهر الخوارزميات

- خوارزمية RSA (Rivest, Shamir and Adleman)
- خوارزمية الجمل ELGamal Algorithm
- خوارزمية المنحني الإهليلجي Elliptic Curve Algorithm EC

مبدأ كيرشوف Kerckhoff's Principle

قد يبدو أن من الآمن أكثر أن نقوم بتخبئة خوارزميتي التشفير وفك التشفير والمفتاح السري كلها دون الإعلان عنها وهو الأمر الذي لا ننصح به.

يجب علينا وفقًا لمبدأ كيرشوف أن نفترض أن الآخرين يعلمون بخوارزمية التشفير أو فك التشفير. وبالتالي تنحصر مهمتنا في مقاومة هجومات فك التشفير في الحفاظ على المفاتيح السرية. سنتبين أهمية هذا المبدأ عندما سنتحدث فيما بعد عن أساليب التشفير الحديثة التي تتمتع بمجال لتوليد المفاتيح واسعًا مما يجعل من الصعب الوصول إليها.

1. التشفير بالاستبدال Substitution Ciphers

يهدف هذا الأسلوب من التشفير إلى استبدال رمز بآخر. إذا كان الرمز هو عبارة عن حرف من الحروف الأبجدية ضمن نص غير مشفر فإننا نقوم باستبداله بحرف آخر. يمكن مثلا استبدال الحرف A بالحرف D والحرف T بالحرف Z وهكذا.. إذا كانت الرموز عبارة عن خانات عددية (من **0** حتى **9**)، فيمكن مثلا استبدال 3 ب **7** و **3** ب **6** وهكذا

يمكن تقسيم طرق الاستبدال ضمن فئتين، هما:

1. المشفرات وحيدة الحرف **Monoalphabetic ciphers**
3. المشفرات متعددة الحروف **Polyalphabetic ciphers**

المشفرات وحيدة الحرف

يجري هنا استبدال حرف (أو رمز) ضمن النص غير المشفر دائمًا بحرف (أو رمز) آخر هو نفسه دائمًا بغض النظر عن موضعه ضمن النص. أي ان العلاقة بين الحروف ضمن النص غير المشفر والنص المشفر هي علاقة واحد لواحد.

مثال:

hello :Plaintext نص غير مشفر

KHOOR :Ciphertext نص مشفر

وتشمل:

1. المشفرات بالجمع.
3. المشفرات بالضرب.
3. المشفرات بالتبديل وحيدة الحرف.

المشفرات بالجمع Additive Ciphers

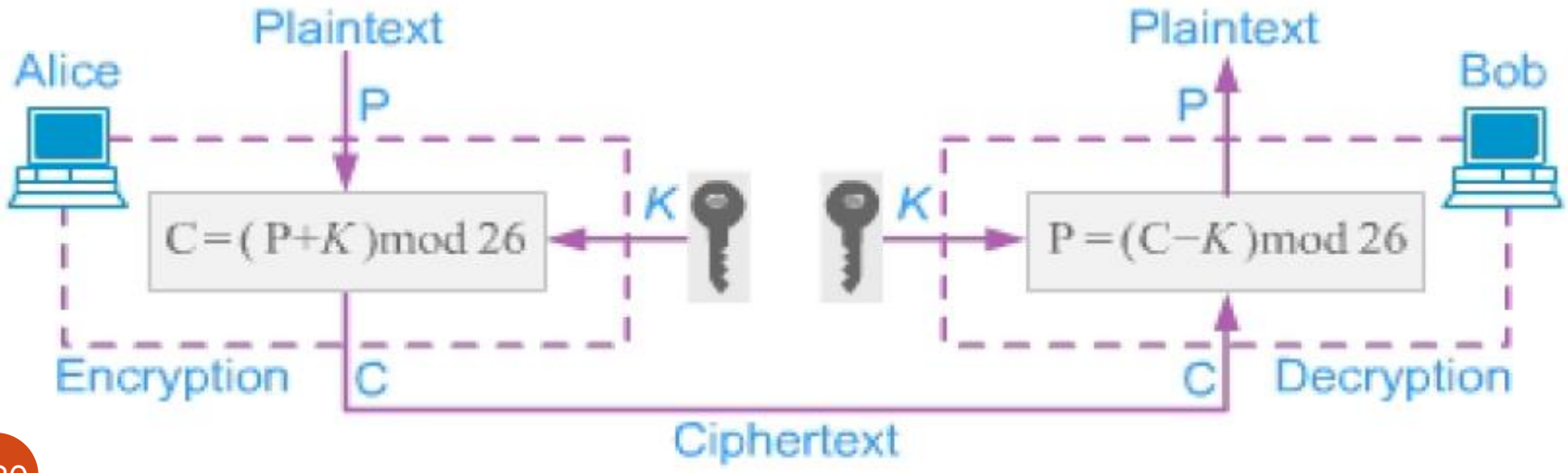
يعد هذا النوع من المشفرات أبسط مشفرات الاستبدال وحيدة الحرف. وتسمى أحيانًا مشفرات الإزاحة **Shift ciphers** أو مشفرات قيصر **Caeser ciphers** ولكن تبقى التسمية مشفرات الجمع أكثر تعبيرًا عن الطبيعة الرياضية لهذا النوع من المشفرات. بفرض أن النص غير المشفر يتضمن حرف أبجدية صغيرة (من a إلى z) وأن النص المشفر يتضمن حروفًا كبيرة (من A إلى Z) فبالتالي من أجل تطبيق عمليات حسابية ورياضية على النصين غير المشفر والمشفر يجب أن نخصص قيمًا عددية لكل حرف صغير أو كبير كما هو مبين في الشكل التالي:

Plaintext →	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
Ciphertext →	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
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

خصص لكل حرف كبير وصغير ضمن الشكل السابق عددًا صحيحًا ضمن المجموعة Z_{36} (وهي مجموعة الأعداد الصحيحة من **0** حتى **35** : $Z_n = \{0,1, 3,3,\ldots, n-1\}$).

المفتاح السري بين Alice و Bob هو أيضًا عدد صحيح من المجموعة Z_{36}

تقوم خوارزمية التشفير بإضافة المفتاح إلى حروف النص غير المشفر. أما خوارزمية فك التشفير فتقوم بطرح هذا المفتاح من حروف النص المشفر. تجري كافة العمليات الحسابية ضمن المجموعة Z_{36} يوضح الشكل التالي كيفية تنفيذ ذلك:



خوارزمية قيصر Caesar Cipher

أبسط أنواع طرق التبديل تنتج عن طريق تبديل أحرف الرسالة الأصلية بأحرف أخرى تبعد عنها مسافة ثابتة في الترتيب الأبجدي. أول من استخدم هذه الطريقة يوليوس قيصر.

قام القيصر بعمل إزاحة للأحرف ثلاث خانات، حيث استبدل الحرف A بالحرف D، والحرف B بالحرف E، والحرف C بالحرف F، وهكذا... كما موضح في الجدول:

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
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	c

تبديل الحروف بطريقة قيصر.

يمكن أن نبرهن بسهولة أن عمليتي التشفير وفك التشفير هما عمليتان متعاكستان وأن النص غير المشفر $P1$ والمولد من قبل **Bob** هو نفسه النص P المرسل من قبل **Alice** :

$$P1 = (C - k)\bmod 36 = (P + k - k)\bmod 36 = P$$

مثال:

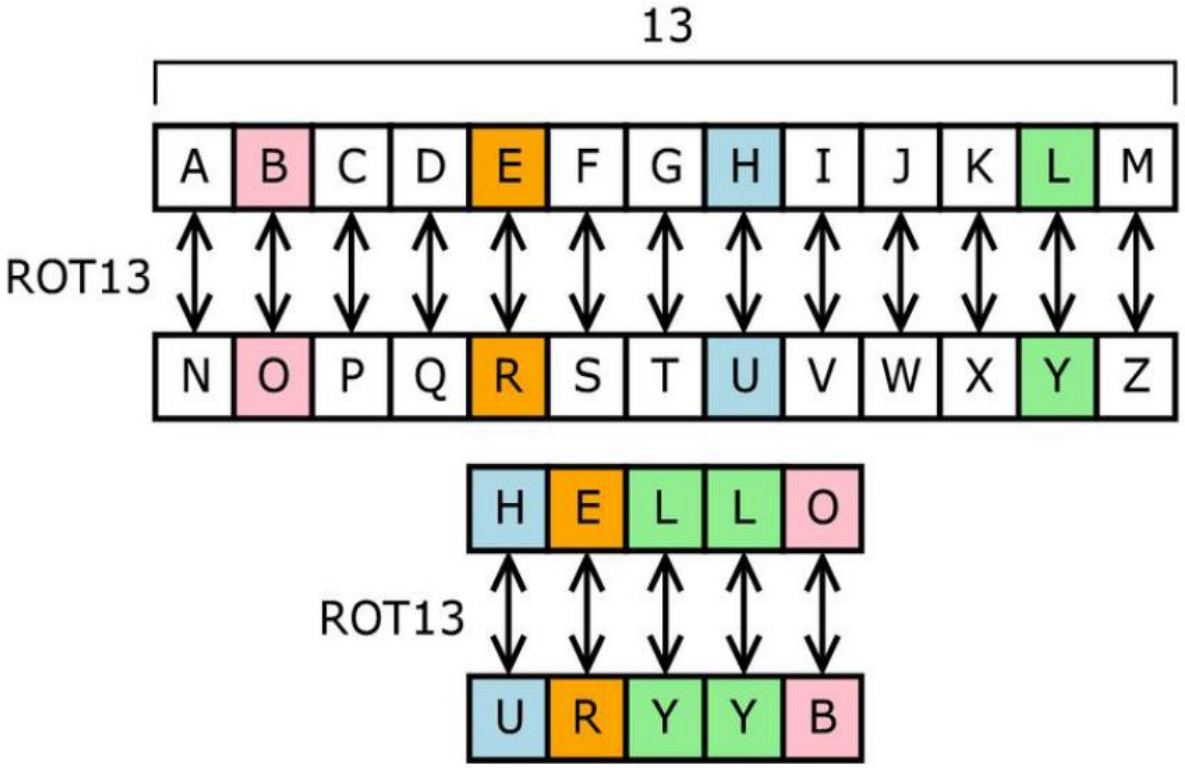
K = 3 , P = RUNAWAY

E(RUNAWAY) ⟶ UXQDZDB

D(UXQDZDB) ⟶ RUNAWAY

خوارزمية الإزاحة 13 ROT13

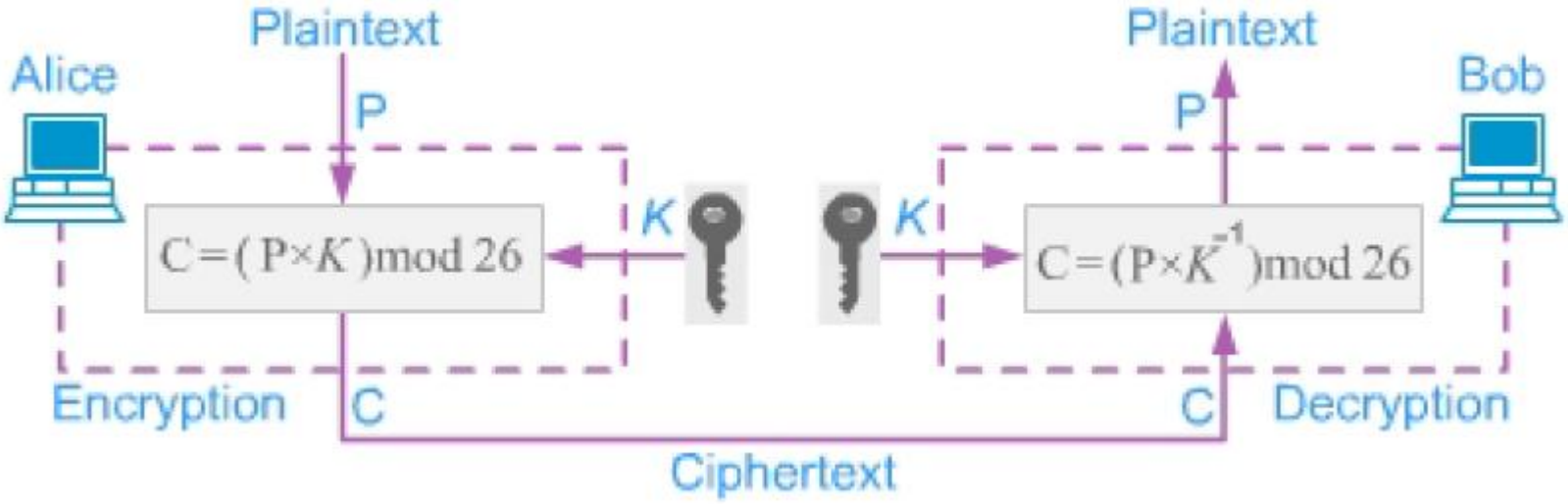
في هذه الطريقة يتم إزاحة Rotate كل حرف من الرسالة الأصلية ثلاث عشرة مرة فنحصل على الـــنص المشفر وإذا أردنا إرجاع النص المشفر إلى أصله نقوم بعمل إزاحة لكل حرف ثلاث عشرة مـــرة أخـــرى، الشكل يوضح عملية الإزاحة المتبعة.



عمل خوارزمية الإزاحة 13.

المشفرات بالضرب Multiplicative Ciphers

تعتمد خوارزميات التشفير هنا على ضرب قيمة حروف النص غير المشفر بقيمة المفتاح للحصول على النص المشفر. أما خوارزميات فك التشفير فتعتمد على قسمة قيم حروف النص المشفر على قيمة المفتاح للحصول على النص غير المشفر، كما هو مبين في الشكل التالي:



يجب أن تجري العمليات الحسابية ضمن المجموعة Z_{36} كما يجب أن يكون المفتاح عنصراً من عناصر المجموعة Z_{36} التي تضم فقط **13** عنصرًا، وهي:

1, 3, 5, 7, 9, 11, 15, 17 , 19 , 31 ,33 ,35

ملاحظة:

المجموعة Z_{36}^{*} هي مجموعة جزئية من Z_{36} وتضم فقط الأعداد التي لها أزواج جداء عكسية، أي أزواج التي يكون نتيجة جدائها مساوي للقيمة **1**، كما هو مبين في الجدول التالي:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
2	0	2	4	6	8	10	12	14	16	18	20	22	24	0	2	4	6	8	10	12	14	16	18	20	22	24
3	0	3	6	9	12	15	18	21	24	1	4	7	10	13	16	19	22	25	2	5	8	11	14	17	20	23
4	0	4	8	12	16	20	24	2	6	10	14	18	22	0	4	8	12	16	20	24	2	6	10	14	18	22
5	0	5	10	15	20	25	4	9	14	19	24	3	8	13	18	23	2	7	12	17	22	1	6	11	16	21
6	0	6	12	18	24	4	10	16	22	2	8	14	20	0	6	12	18	24	4	10	16	22	2	8	14	20
7	0	7	14	21	2	9	16	23	4	11	18	25	6	13	20	1	8	15	22	3	10	17	24	5	12	19
8	0	8	16	24	6	14	22	4	12	20	2	10	18	0	8	16	24	6	14	22	4	12	20	2	10	18
9	0	9	18	1	10	19	2	11	20	3	12	21	4	13	22	5	14	23	6	15	24	7	16	25	8	17
10	0	10	20	4	14	24	8	18	2	12	22	6	16	0	10	20	4	14	24	8	18	2	12	22	6	16
11	0	11	22	7	18	3	14	25	10	21	6	17	2	13	24	9	20	5	16	1	12	23	8	19	4	15
12	0	12	24	10	22	8	20	6	18	4	16	2	14	0	12	24	10	22	8	20	6	18	4	16	2	14
13	0	13	0	13	0	13	0	13	0	13	0	13	0	13	0	13	0	13	0	13	0	13	0	13	0	13
14	0	14	2	16	4	18	6	20	8	22	10	24	12	0	14	2	16	4	18	6	20	8	22	10	24	12
15	0	15	4	19	8	23	12	1	16	5	20	9	24	13	2	17	6	21	10	25	14	3	18	7	22	11
16	0	16	6	22	12	2	18	8	24	14	4	20	10	0	16	6	22	12	2	18	8	24	14	4	20	10
17	0	17	8	25	16	7	24	15	6	23	14	5	22	13	4	21	12	3	20	11	2	19	10	1	18	9
18	0	18	10	2	20	12	4	22	14	6	24	16	8	0	18	10	2	20	12	4	22	14	6	24	16	8
19	0	19	12	5	24	17	10	3	22	15	8	1	20	13	6	25	18	11	4	23	16	9	2	21	14	7
20	0	20	14	8	2	22	16	10	4	24	18	12	6	0	20	14	8	2	22	16	10	4	24	18	12	6
21	0	21	16	11	6	1	22	17	12	7	2	23	18	13	8	3	24	19	14	9	4	25	20	15	10	5
22	0	22	18	14	10	6	2	24	20	16	12	8	4	0	22	18	14	10	6	2	24	20	16	12	8	4
23	0	23	20	17	14	11	8	5	2	25	22	19	16	13	10	7	4	1	24	21	18	15	12	9	6	3
24	0	24	22	20	18	16	14	12	10	8	6	4	2	0	24	22	20	18	16	14	12	10	8	6	4	2
25	0	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

مثال:

K = 7, P = hello
E(hello) → XCZZU
D(XCZZU) → hello

المشفرات بالتبديل وحيدة الحرف Monoalphabetic Substitution Ciphers

على اعتبار أن المشفرات بالجمع وبالضرب لديها مجال صغير لاختيار المفتاح المرتبط بها، فإنه من الممكن أن يكون من السهل كسرها. يمكن أن نوفر حلاً أفضل بأن يتفق كلا الطرفين على جدول للمطابقة بين الحروف المتبادلة. يعرض الشكل التالي مثالا عن هذا الجدول:

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

مثال:

“runaway” ⇨ “HJGNPNS”

المشفرات متعددة الحروف Polyalphabetic Ciphers

مع هذا النوع من المشفرات، يمكن أن يتوفر عدة خيارات لاستبدال كل حرف من الحروف. فالعلاقة بين أي حرف من حروف النص غير المشفر مع أي حرف ضمن النص المشفر هي علاقة حرف واحد مع أكثر **one-to-many** يمكن مثلاً استبدال الحرف **"a"** بالحرف D إذا ورد في بداية النص أو بالحرف **N** إذا ورد في منتصفه.

- يضمن هذا النوع من المشفرات إخفاء تواتر ورود الحروف ضمن النص. وبالتالي تصبح محاولات كسر التشفير المعتمدة على تواتر ورود الحروف ضمن النص غير فعالة.
- لبناء مشفر متعدد الحروف، يجب الربط بين الحرف ضمن النص المشفر والحرف المقابل له ضمن النص غير المشفر ومكان تواجده ضمن هذا النص. هذا يعني أن المفتاح في هذهالحا لة، هو عبارة عن مجموعة من المفاتيح الجزئية. يرتبط كل واحد منها بموقع الحرف ضمن النص غير المشفر. بمعنى آخر:

فإن المفتاح $k = (k1, k\,3, k3, \ldots.)$ يفيد أنه يمكن استخدام $k\ i$ لتشفير الحرف رقم i ضمن النص غير المشفر لبناء الحرف رقم i ضمن النص المشفر.

أمثلة:

التشفير بمفتاح ذاتي التوليدAutokey Cipher: يتكون المفتاح مع هذا النوع من المشفرات من مجموعة من المفاتيح الجزئية متتابعة حيث يجري استخدام كل مفتاح جزئي لتشفير الحرف الموافق لترتيب المفتاح ضمن النص غير المشفر. يجري الاتفاق حول قيمة المفتاح الجزئي الأول. ويكون المفتاح الجزئي الثاني هو قيمة الحرف الأول ضمن النص غير المشفر. أما المفتاح الجزئي الثالث، فهو قيمة الحرف الثاني ضمن النص غير المشفر وهكذا دواليك.

$P = P_1P_3P_3\ldots.$

$C = C_1C_3C_3\ldots.$

$k = (k_1, P_1, P_3, \ldots)$

$Encryption : C_i = (P_i + k_i) \bmod 36$

$Decryption : P_i = (C_i - k_i) \bmod 36$

أمثلة:

- **المشفر فيجينر Vigenere Cipher:** جرى تصميم هذا المشفر من قبل الرياضي الفرنسي فيجينر في القرن السادس عشر. يعتمد هذا المشفر على تكرار استخدام مفتاح مع دفق حروف النص غير المشفر بشكل متتالي. يجب أن يكون طول المفتاح **m** محقق للشرط $1 \leq m \leq 36$.
  - نساوي كل حرف من المفتاح والنص بقيمته العددية من 0 إلى 25 حسب ترتيب حروف أبجدية اللغة الإنجليزية، مثلاً نجعل: a = 0، b=1، c= 2 وهكذا...
  - نجمع المفتاح مع النص الأصلي للحصول على النص المشفر، كما هو موضح في المعادلة التالية:

النص المشفر = النص الأصلي + الحرف الموازي له من المفتاح.

  - لفك التشفير نستخدم المعادلة التالية:

النص الأصلي = النص المشفر – الحرف الموازي له من المفتاح.

$$P = P_1 P_2 P_3 \cdots$$

$$C = C_1 C_2 C_3 \cdots$$

$$K = [(k_1, k_2, \ldots, k_m), (k_1, k_2, \ldots, k_m), \cdots]$$

$$Encryption : C_i = (P_i + k_i) \bmod 26$$

$$Decryption : P_i = (C_i - k_i) \bmod 26$$

مثال:

Key: “python”
Plaintext: “rabbitwithbigpointyteeth”
Ciphertext:

R	a	b	b	i	t	w	i	t	h	b	i	g	p	o	i	n	t	y	t	e	e	t	H
p	y	t	h	o	n	p	y	t	h	o	n	p	y	t	h	o	n	p	y	t	h	o	N
G	Y	U	I	V	G	L	G	M	Y	M	V	V	N	H	P	B	G	N	R	P	L	H	U

مثال:

لنفرض أن المفتاح المستخدم للتشفير هو كلمة danger، والرسالة المراد تشفيرها هي:

Tomorrow is the time

نقوم بإتباع نفس الخطوات السابقة للتشفير ولفك التشفير:

1. نكرر المفتاح حسب عدد حروف النص الأصلي، كالتالي:

key : dangerdangerdange

plaintext : tomorrowisthetime

2. نجمع الحروف المتناظرة معاً للحصول على النص المشفر، بعد تحويل جميع الحروف إلى أرقام،

مثلا: t+d تساوي 3+19=22 وهي قيمة الحرف w، إذا w=t+d، كذلك o+a تساوي 0+14=14

وهو حرف o، وهكذا... كما هو موضح في الجدول التالي:

d	a	n	g	e	r	d	a	n	g	e	r	d	a	n	g	e
t	o	m	o	r	r	o	w	i	s	t	h	e	t	i	m	e
w	o	z	u	v	i	r	w	v	y	x	y	h	t	v	s	i

وبعد تشفير الرسالة نحصل على النص المشفر:

Wozuvirwvyxyhtvsi

3. لفك التشفير نطرح كل حرفين متناظرين من بعضهما بعد تحويلهما إلـــى أرقـــام، مـــثلا:

$$d - w = 3 - 22 = 19$$

والرقم 19 يساوي الحرف t، وهكذا حتى نحصل على النص الأصلي.

خوارزمية الكراسة الواحدة One time pad

ظهرت سنة 1917ف من قبل العالم فيرنام Vernam، في هذه الطريقة مفتاح التشفير يكون طويل جداً، أحيانا أطول من النص الأصلي. من الممكن استخدام كتاب أو قصة أو مجلة لتكون المفتاح، وذلك حسب الاتفاق بين المرسل والمستقبل.

يتم التشفير بأخذ جزء من المفتاح، جملة أو كلمة، بحيث يكون طوله مساوياً لطول النص الأصلي، وتحويل هذا الجزء إلى أرقام، بعد ذلك نزيح كل حرف من النص الأصلي بمقدار الرقم الذي يوازيه من الجزء المأخوذ من المفتاح. ولفك التشفير نستخدم نفس الطريقة ولكن بالمعكوس، أي أنه إذا كانت الإزاحة في التشفير إلى الأمام نرجع الأحرف إلي الخلف عند فك التشفير والعكس صحيح.

المشفر لمرة واحدة One-time Pad:يكون في هذا المشفر طول المفتاح السري العشوائي مساويًا لطول النص غير المشفر. مما يجعل من المستحيل عمليًا كسر هذا المشفر ومعرفة النص المرسل في الأصل وفقًا لما جاء في دراسة قام بها العالم شانون بهذا الخصوص. تزداد سرية هذا المشفر عند تغيير المفتاح عند تغيير النص المرسل حيث لا يجري استخدامه إلا مرة واحدة فقط. يؤدي ذلك إلى ضمان السرية بشكل كبير ولكن يستحيل عمليًا تحقيق ذلك بشكل سهل وتبادل المفتاح من وقت إلى آخر بين الطرفين المتصلين.

مثال:

Ciphertext: NZAKBMK

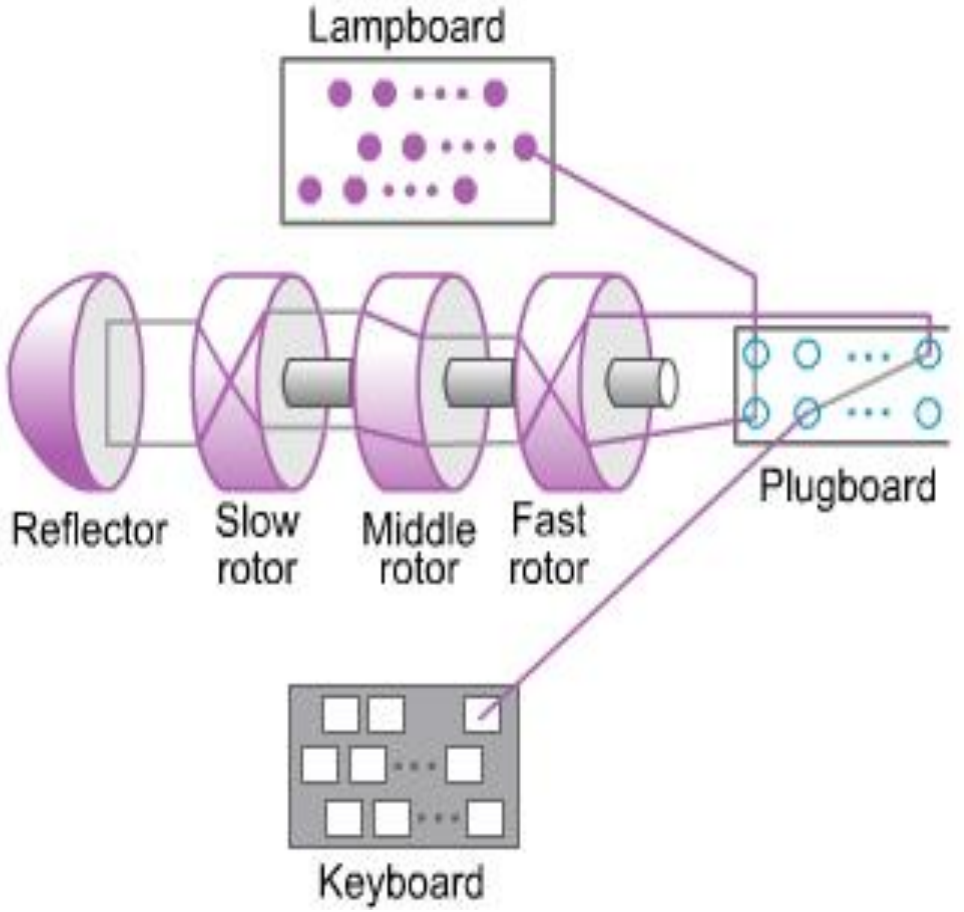
Possible Vigenére keys: wtnkxmm and nlvwker

Ciphertext: NZAKBMK NZAKBMK

Possible keys: nlvwker wtnkxmm

Plaintext: goforit runaway

- **آلة انيغما Enigma Machine:** جرى تطوير هذه الآلة من قبل الألمان أثناء الحرب العالمية الثانية. وهي آلة معقدة تعمل وفقاً لمبدأ التشفير بالتبديل. يبين الشكل التالي منظر الآلة التي سنشرح فيما يلي طريقة عملها:

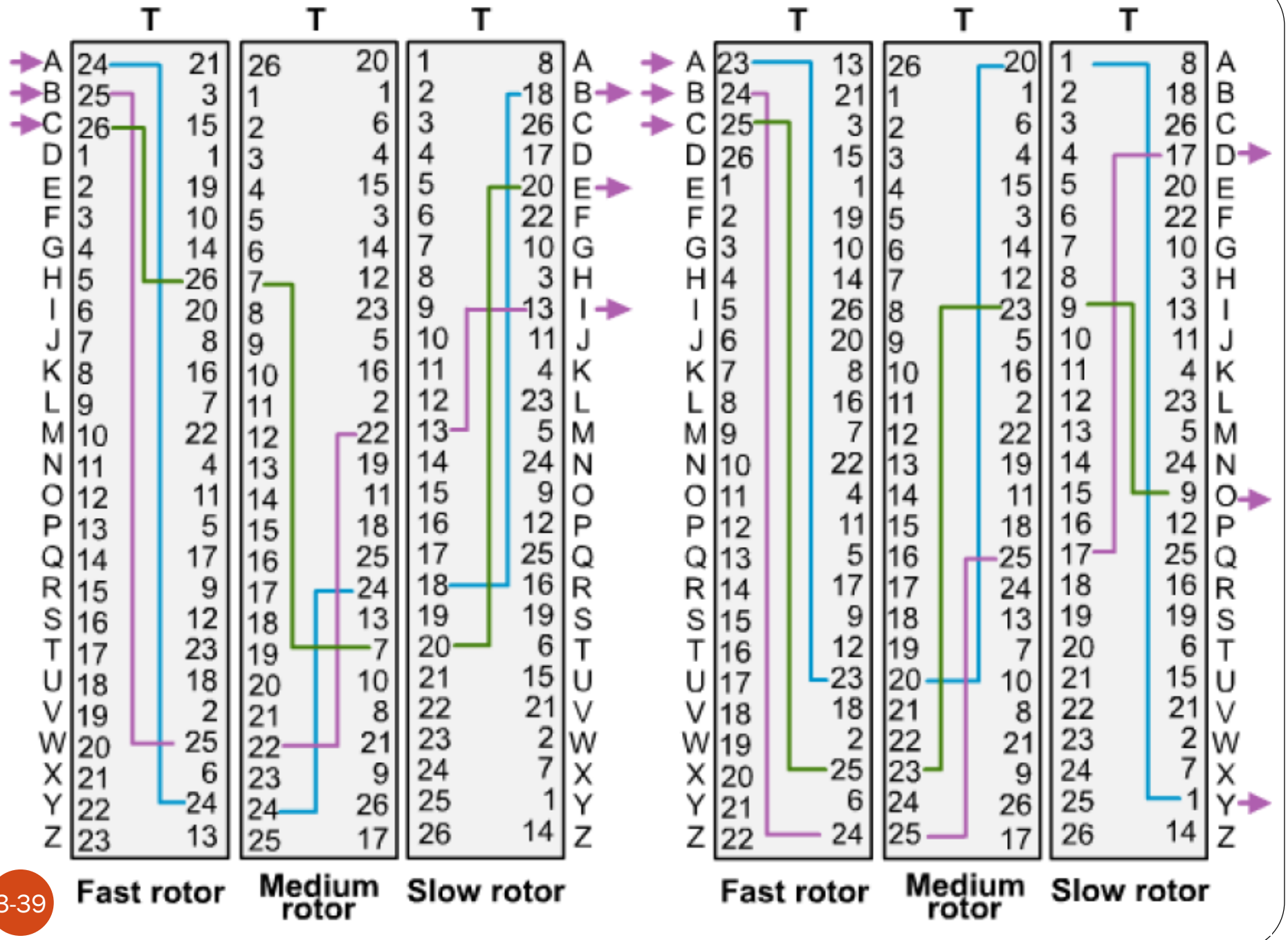


تتضمن الآلة ثلاثة دواليب من بين خمسة يمكن تركيبها. تعمل هذه الدواليب على الشكل التالي:

1. يقوم كل دولاب بعملية تشفير بالتبديل وحيدة الحرف.
3. يجري تمرير نتيجة تشفير كل دولاب إلى الدولاب التالي ليجري تشفيرها مرة أخرى وهكذا وصولاً إلى الدولاب الثالث.
3. تدور الدواليب من أجل كل حرف من حروف النص غير المشفر وفقًا لما يلي:

- الدولاب السريع **Fast rotor**: من أجل كل حرف.
- الدولاب المتوسط **Middle rotor**: من أجل كل **36** حرف.
- الدولاب البطيء **Slow rotor**:من أجل كل **676 = 26x 26** حرف.

- إذا جرى إدخال الحرف **C** على لوحة المفاتيح كحرف أول فإنه سيجري ما يلي:
  1. سيطابق هذا الحرف الرقم **26** على الدولاب السريع.
  2. الرقم **26** مرتبط بالرقم **7** على الدولاب الوسيط.
  3. الرقم **7** مرتبط بالرقم **20** على الدولاب البطيء.
  4. يوافق الرقم **20** الحرف **E** كنتيجة لعملية التشفير.
- إذا جرى إدخال الحرف **C** على لوحة المفاتيح كحرف ثاني فإنه سيجري ما يلي بعد دوران الدولاب السريع لحرف واحد:
  1. سيطابق هذا الحرف الرقم **25** على الدولاب السريع.
  2. الرقم **25** مرتبط بالرقم **23** على الدولاب الوسيط.
  3. الرقم **23** مرتبط بالرقم **9** على الدولاب البطيء.
  4. يوافق الرقم **9** الحرف **O** كنتيجة لعملية التشفير.
- يجري تكرار التشفير للحروف بعد **17,576** = **26 x 26 x 26** حرف.


T
T
T
A 24 21
B 25 3
C 26 15
D 1 1
E 2 19
F 3 10
G 4 14
H 5 26
I 6 20
J 7 8
K 8 16
L 9 7
M 10 22
N 11 4
O 12 11
P 13 5
Q 14 17
R 15 9
S 16 12
T 17 23
U 18 18
V 19 2
W 20 25
X 21 6
Y 22 24
Z 23 13
Fast rotor
26 20
1 1
2 6
3 4
4 15
5 3
6 14
7 12
8 23
9 5
10 16
11 2
12 22
13 19
14 11
15 18
16 25
17 24
18 13
19 7
20 10
21 8
22 21
23 9
24 26
25 17
Medium rotor
1 8 A
2 18 B
3 26 C
4 17 D
5 20 E
6 22 F
7 10 G
8 3 H
9 13 I
10 11 J
11 4 K
12 23 L
13 5 M
14 24 N
15 9 O
16 12 P
17 25 Q
18 16 R
19 19 S
20 6 T
21 15 U
22 21 V
23 2 W
24 7 X
25 1 Y
26 14 Z
Slow rotor
T
T
T
A 23 13
B 24 21
C 25 3
D 26 15
E 1 1
F 2 19
G 3 10
H 4 14
I 5 26
J 6 20
K 7 8
L 8 16
M 9 7
N 10 22
O 11 4
P 12 11
Q 13 5
R 14 17
S 15 9
T 16 12
U 17 23
V 18 18
W 19 2
X 20 25
Y 21 6
Z 22 24
Fast rotor
26 20
1 1
2 6
3 4
4 15
5 3
6 14
7 12
8 23
9 5
10 16
11 2
12 22
13 19
14 11
15 18
16 25
17 24
18 13
19 7
20 10
21 8
22 21
23 9
24 26
25 17
Medium rotor
1 8 A
2 18 B
3 26 C
4 17 D
5 20 E
6 22 F
7 10 G
8 3 H
9 13 I
10 11 J
11 4 K
12 23 L
13 5 M
14 24 N
15 9 O
16 12 P
17 25 Q
18 16 R
19 19 S
20 6 T
21 15 U
22 21 V
23 2 W
24 7 X
25 1 Y
26 14 Z
Slow rotor