Cryptography

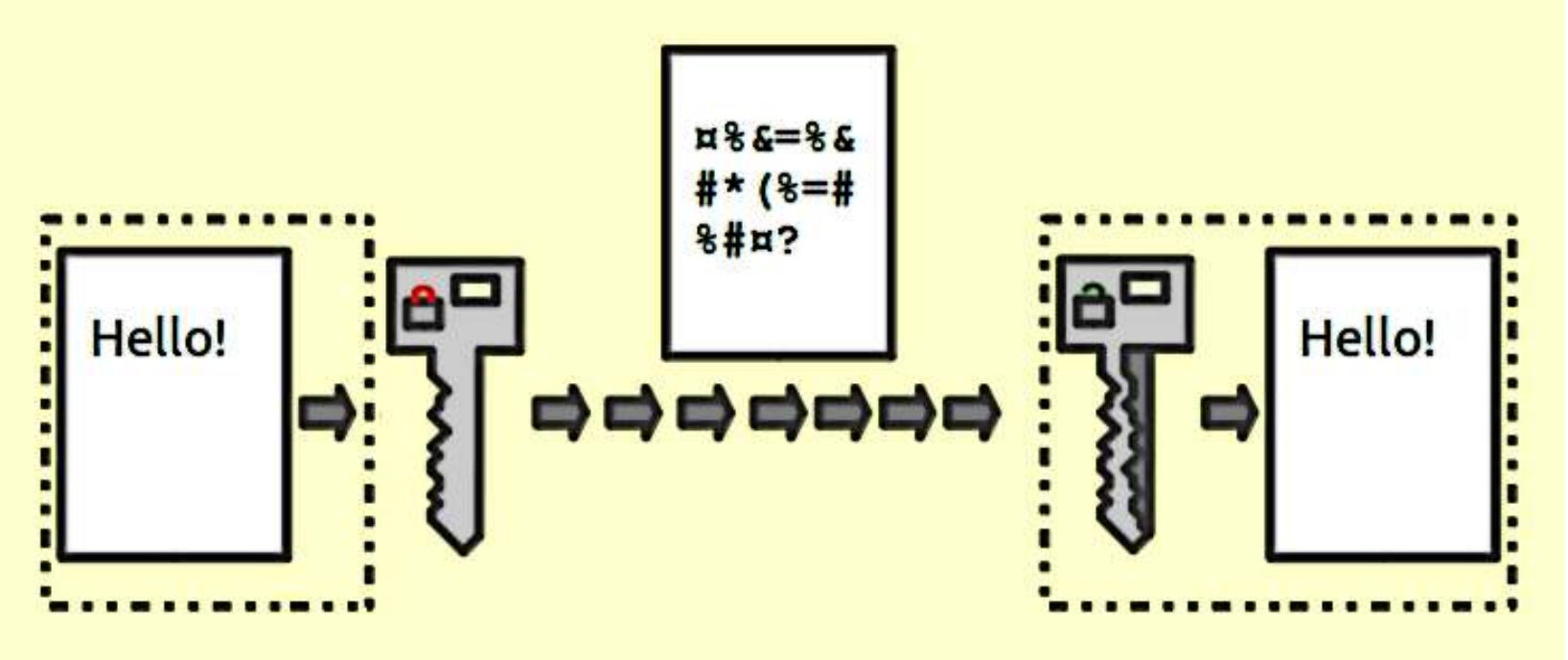
CYS401

- What is Encryption/Decription?
- Foundations of Cryptology
- Cipher Methods
- Cryptographic Algorithm,
- Cryptographic Tools,
- Protocols for secure communication,
- Attacks on cryptosystems

Cryptography

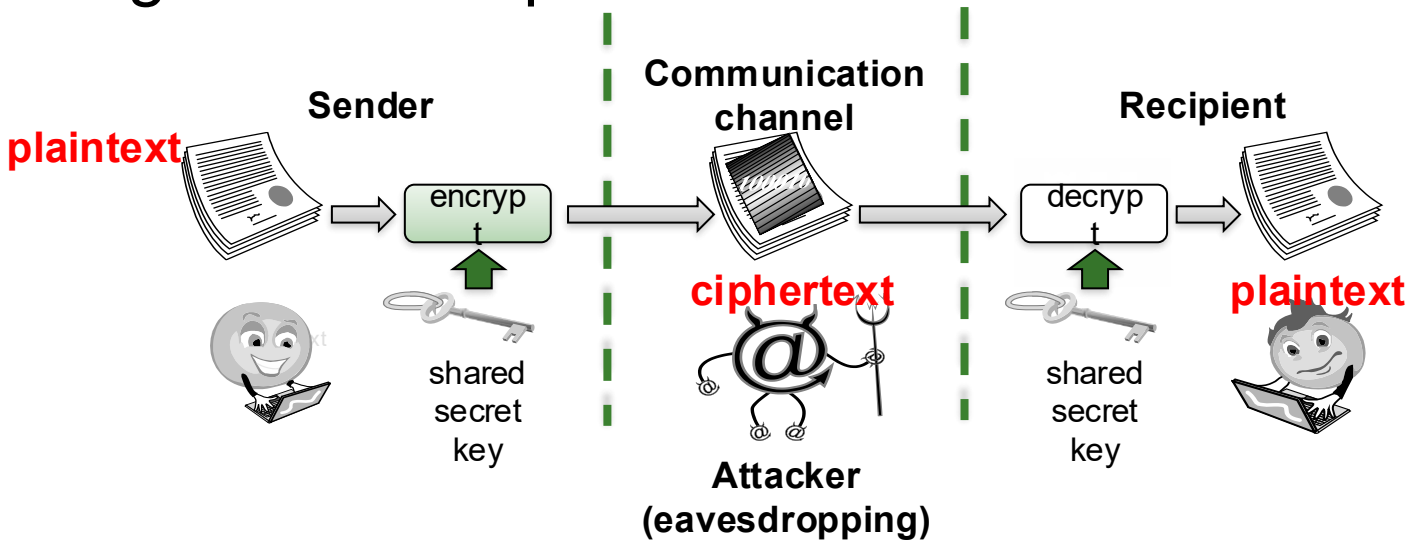
- Cryptography, a word with Greek origins, means "secret writing." However, we use the term to refer to the science and art of transforming messages to make them secure and immune to attacks.

Encryption in a nutshell



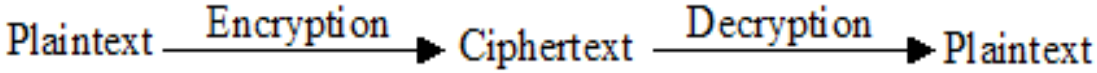
Basic Concepts

- **Encryption:** converting original message into a form unreadable by unauthorized individuals.
  - allow two parties to establish confidential communication over an insecure channel that is subject to eavesdropping.
- **Decryption:** the process of converting the ciphertext message back into plaintext



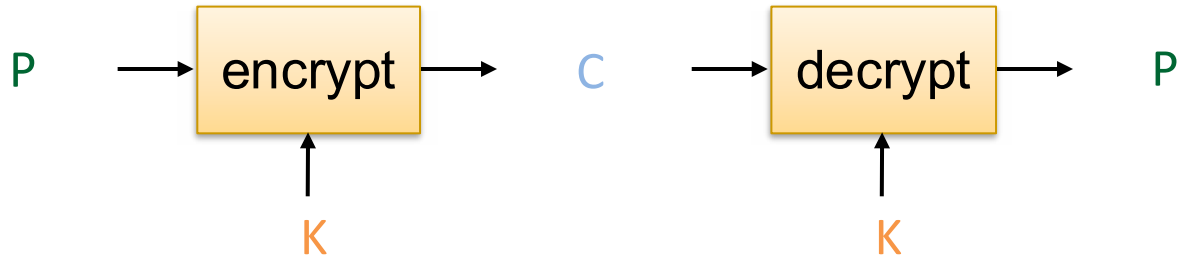
Some Basic Terminology

- **plaintext** - original message
- **ciphertext** - coded message
- **cipher** - algorithm for transforming plaintext to ciphertext
- **key** - info used in cipher known only to sender/receiver
- **encipher (encrypt)** - converting plaintext to ciphertext
- **decipher (decrypt)** - recovering ciphertext from plaintext
- **cryptography** - study of encryption principles/methods
- **cryptanalysis (codebreaking)** - study of principles/ methods of deciphering ciphertext *without* knowing key
- **cryptology** - field of both cryptography and cryptanalysis



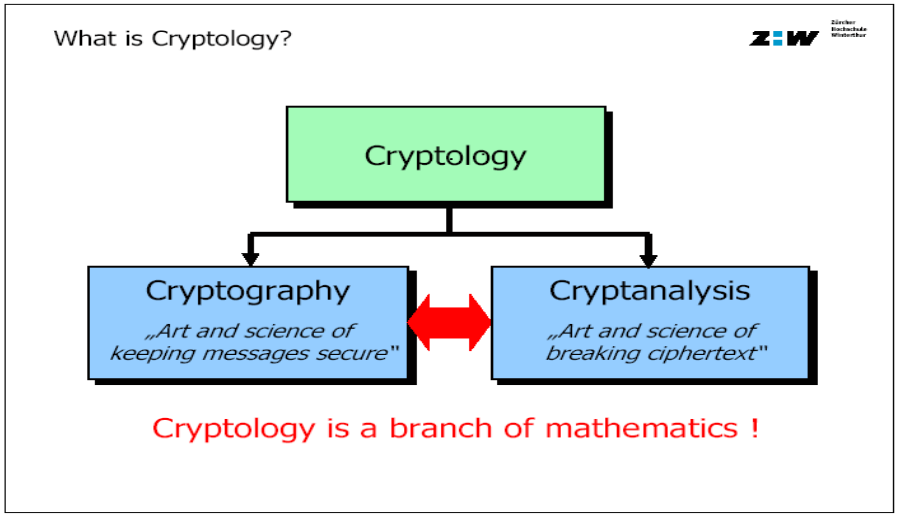
Basic Concepts

- Data Encryption and Decryption Notation
  - Secret key K
  - Encryption function $E_K(P) = C$
  - Decryption function $D_K(C) = P$
  - Plaintext length typically the same as ciphertext length



Basic Concepts

- **Cryptography:** process of making and using codes to secure transmission of information
  - Can **protect confidentiality and integrity**, but not availability
- **Cryptology**: science of encryption; combines cryptography and cryptanalysis
  - **Cryptanalysis:** process of obtaining the original message from encrypted message without access to the required secret information

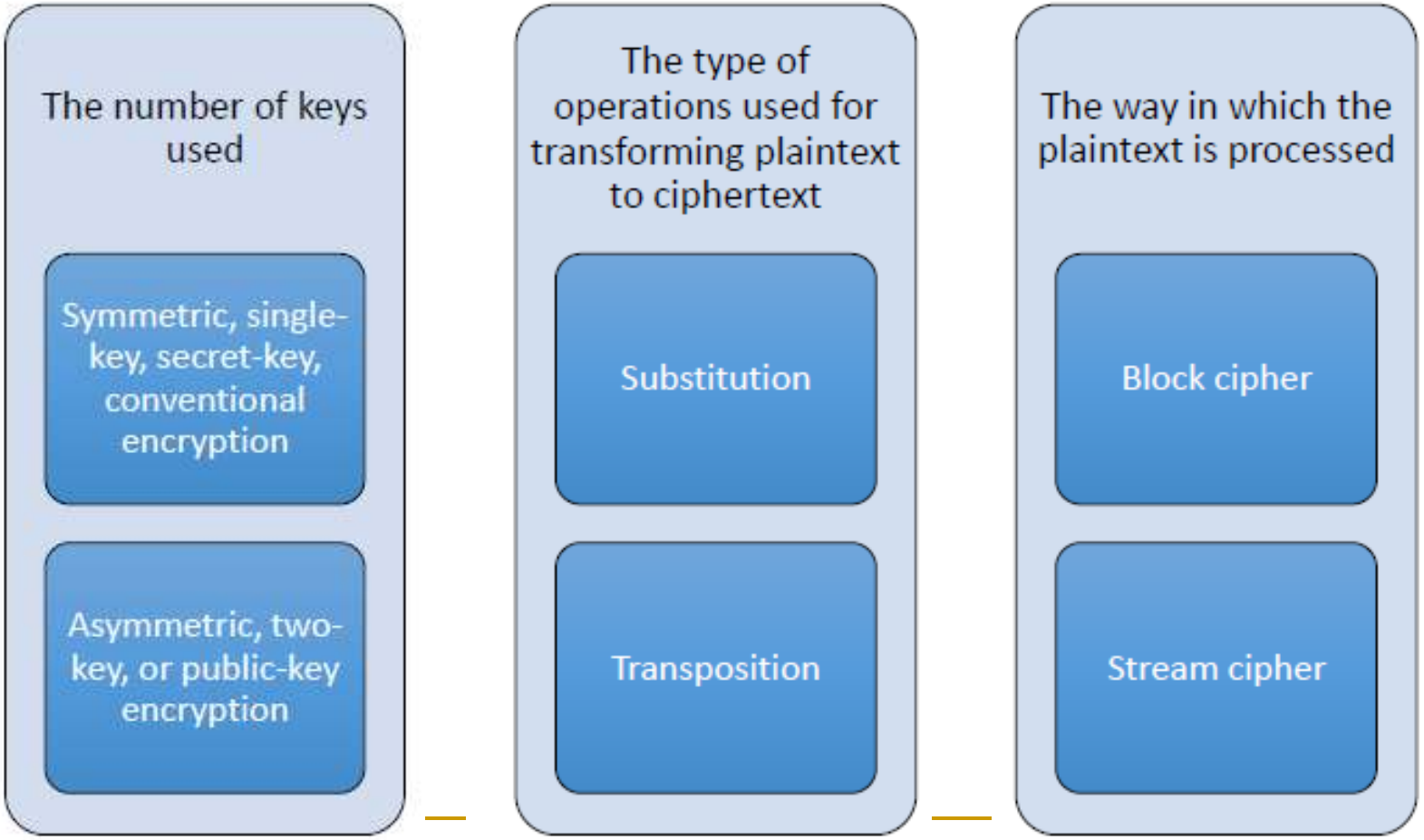


Kerckhoffs's principle in cryptography

- Cryptography algorithm should be secure even if everyone knows how it works, the security of a cryptosystem should depend solely on the secrecy of the key and the private randomizer
- Based on Kerckhoff's principle, one should always assume that the adversary, Eve, knows the encryption/decryption algorithm. The resistance of the cipher to attack must be based only on the secrecy of the key.
- In short:
  - Algorithm must be made public
  - Only the key is kept secret

Classifying Cryptographic Systems

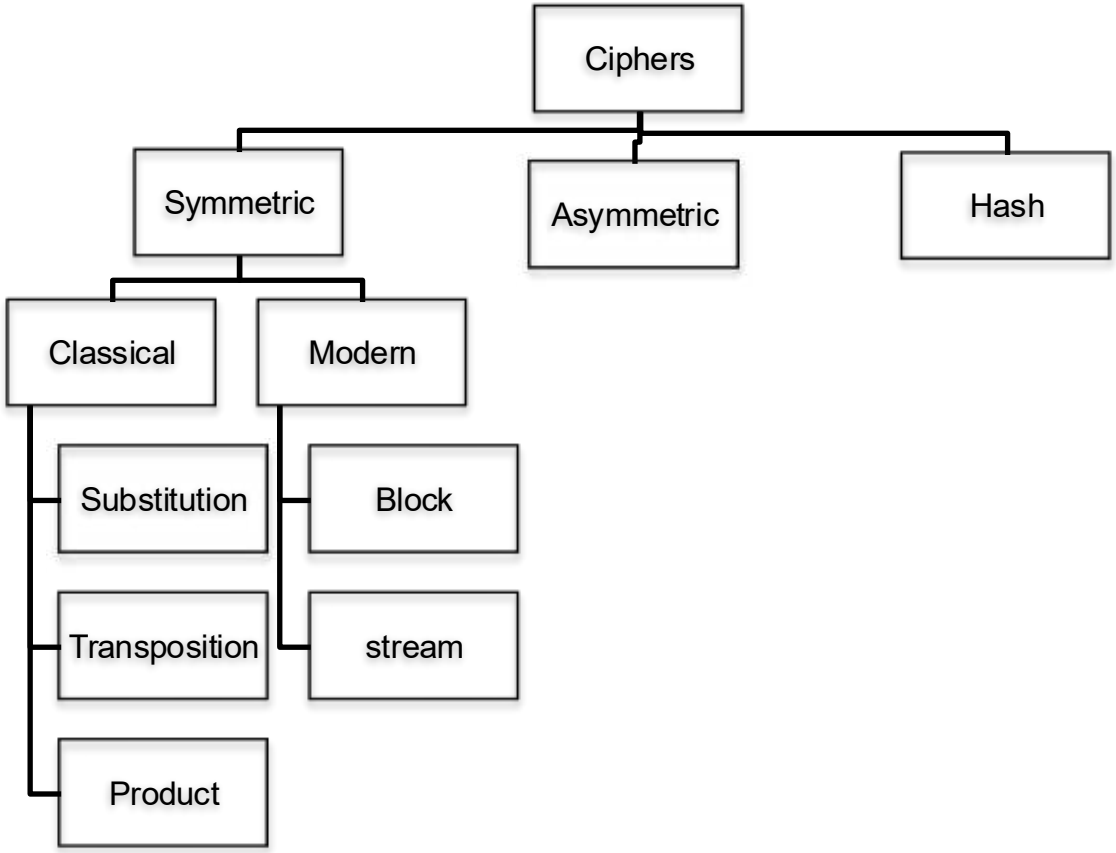
Cryptosystems are characterized along three independent dimensions



Classifying Cryptographic Systems

- Number of Keys Used
  - Same secret key is used for encryption and decryption → **Symmetric/ Conventional Encryption**
  - Different keys for encryption and decryption → **Asymmetric/ Public Key Encryption**
- Type of operation used for encryption
  - **Substitution –** Each letter in the plaintext (bit, letter, group of bits or letters) is mapped into another element
  - **Transposition/Permutation –** Rearranging elements of the plaintext
- Plaintext processing
  - ***Blocks*** **-** Block ciphers processes one block of input at a time and produces an output for each input block
  - ***Bit by bit*** **–** Stream Cipher processes the input elements simultaneously, producing output one element at a time, as it goes along

Cipher Classification



Cryptosystem Types

- Symmetric Cryptosystem
  - Substitution: Replacing
  - Transposition: Change the positions
- Asymmetric Cryptosystem
  - Public key cryptosystems

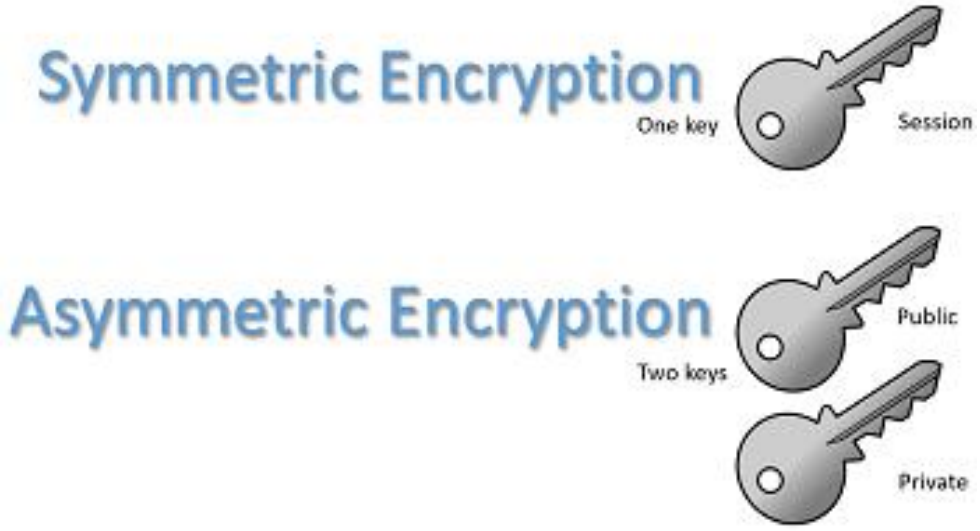
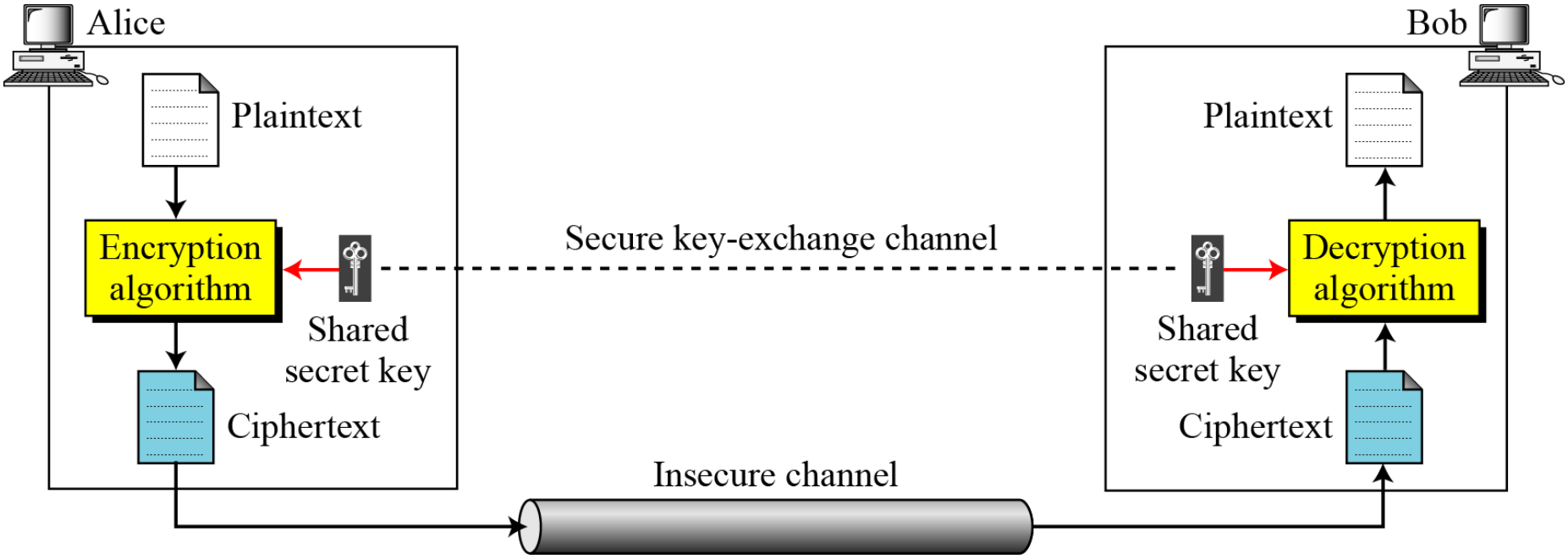


Figure 5.1 *General idea of symmetric-key cipher*



symmetric-key encipherment uses a single key for both encryption and decryption.

The encryption and decryption algorithms are inverses of each other

If P is the plaintext, C is the ciphertext, and K is the key,

$$\text{Encryption: } C = E_k(P) \qquad \text{Decryption: } P = D_k(C)$$

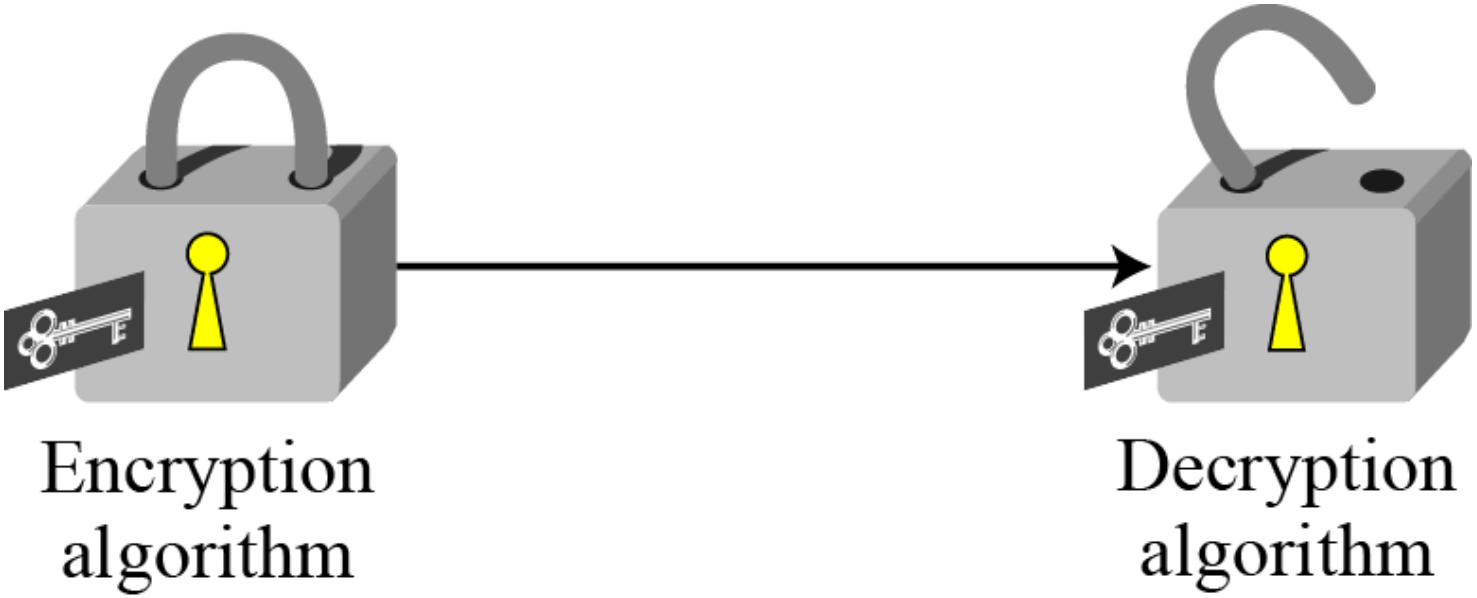
$$\text{In which, } D_k(E_k(x)) = E_k(D_k(x)) = x$$

We assume that Bob creates P_1; we prove that $P_1 = P$:

$$\textbf{Alice: } C = E_k(P)$$

$$\textbf{Bob: } P_1 = D_k(C) = D_k(E_k(P)) = P$$

Figure 3.2 ***Locking and unlocking with the same key***



Symmetric Cryptosystem Characteristics

- **Efficiency**
  - Functions E_K and D_K should have efficient algorithms
  - Symmetric key encryption is fast compared to public key encryption
  - How fast? Almost 30,000 times faster
- **Consistency**
  - Decrypting the ciphertext yields the plaintext
  - $D_K(E_K(P)) = P$

Substitution Ciphers: Caesar Cipher

- Caesar Cipher (1st Century B.C. – used by Julius Caesar to encrypt letters sent to Cicero).
  - Basic Idea: The cipher alphabet is the plain alphabet shifted n spaces right.
  - For example, if we take n = 14, we get:

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	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	N

  - Plaintext: "attack at dawn"
  - Ciphertext: "OHHOQY OH ROKB"

$E_n(x) = (x + n) \bmod 26$ → E(a)=1+14 mod26 =15 = "o"
$D_n(x) = (x – n) \bmod 26$ →D(O)= 1-14 mod 26= 13 ="a"

Caesar Cipher

Shift =3

Plain	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
Cipher	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

To encrypt a message, replace every letter in the message with the one shown below it in the table. For example,

```
THE SECRET IS OUT
WKH VHFUHW LV RXW
```

Caesar Cipher

- Example, let n=1 and plaintext:

  Dear, shall we have dinner at McD?

- Encryption:
  - E(D) = (3 + 1) mod = 4 or E
  - E(e) = (4 + 1) mod = 5 or f
  - E(a) = (0 + 1) mod= 1 b

- Ciphertext: Efbs-!tibmm!xf!ibwf!ejoofs!bu!NdE@
- Decryption:
  - D(E) = (4 - 1) mod = 3 or D
  - D(f) = (5 - 1) mod = 4 or e
  - D(b) = (1 - 1) mode = 0 a

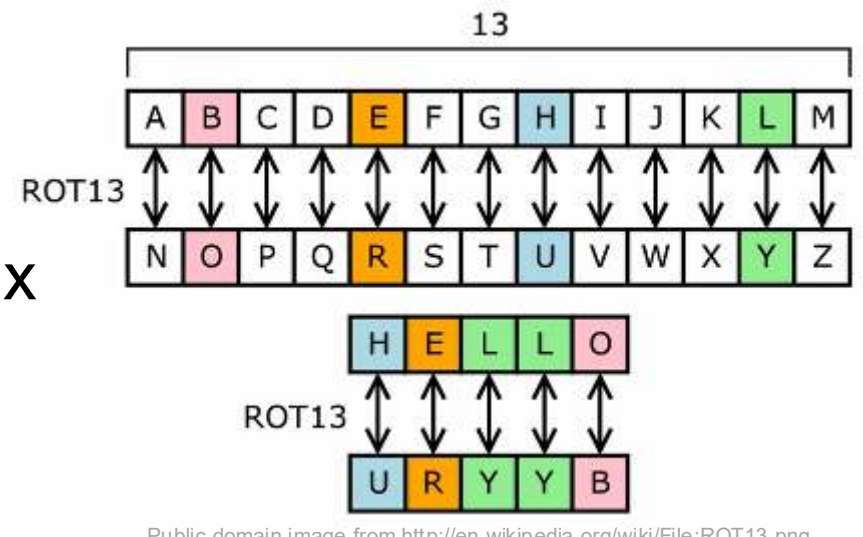
Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(	40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
;	59	0073	0x3b	[	91	0133	0x5b	{	123	0173	0x7b
<	60	0074	0x3c	\	92	0134	0x5c	|	124	0174	0x7c
=	61	0075	0x3d	]	93	0135	0x5d	}	125	0175	0x7d
>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
?	63	0077	0x3f	_	95	0137	0x5f				

First, build the mapping table with all the shifts to get it easier!

Public domain image from http://commons.wikimedia.org/wiki/File:Caesar3.svg

Substitution Ciphers: ROT13

- Each letter is uniquely replaced by another.
- There are 26! possible substitution ciphers.
- There are more than 4.03 x 10^{26} such ciphers.
- One popular substitution "cipher" is ROT13.



Public domain image from http://en.wikipedia.org/wiki/File:ROT13.png

Encrypt the sentence "We love PSU"

Substitution Ciphers : Keyword Ciphers

- Use a keyword or phrase as the basis of the encryption scheme.
- For example, let "**PSU IS MY CHOICE**" be the keyword.
- Remove repeated letters in the keyword
- and add in remaining letters of alphabet in order.

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

ENCRYPT

DECRYPT

- Encrypt
  - HELLO=HMBBG
  - SMART=
  - STUDYHARD=
- DECRYPT
  - BPZX= Lazy

Public domain image from http://commons.wikimedia.org/wiki/File:Caesar3.svg

Examples

Given "**Zebras**" as a keyword, Decrypt using the keyword cipher

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

Examples

Given "**Zebras**" as a keyword, Decrypt using the keyword cipher

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

Playfair Cipher

- not even the large number of keys in a monoalphabetic cipher provides security
- one approach to improving security was to encrypt multiple letters
- the **Playfair Cipher** is an example
- invented by Charles Wheatstone in 1854, but named after his friend Baron Playfair

Playfair Key Matrix

- a 5X5 matrix of letters based on a keyword
- fill in letters of keyword (sans duplicates)
- fill rest of matrix with other letters
- eg. using the keyword MONARCHY

Encrypting and Decrypting

- plaintext is encrypted two letters at a time
  1. if a pair is a repeated letter, insert filler like 'X'
  2. if both letters fall in the same row, replace each with letter to right (wrapping back to start from end)
  3. if both letters fall in the same column, replace each with the letter below it (wrapping to top from bottom)
  4. otherwise each letter is replaced by the letter in the same row and in the column of the other letter of the pair

Playfair Encryption and Decryption

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

HELLO -> HE LX LO

Encryption:
HE -> CF
LX -> SU
LO -> MP

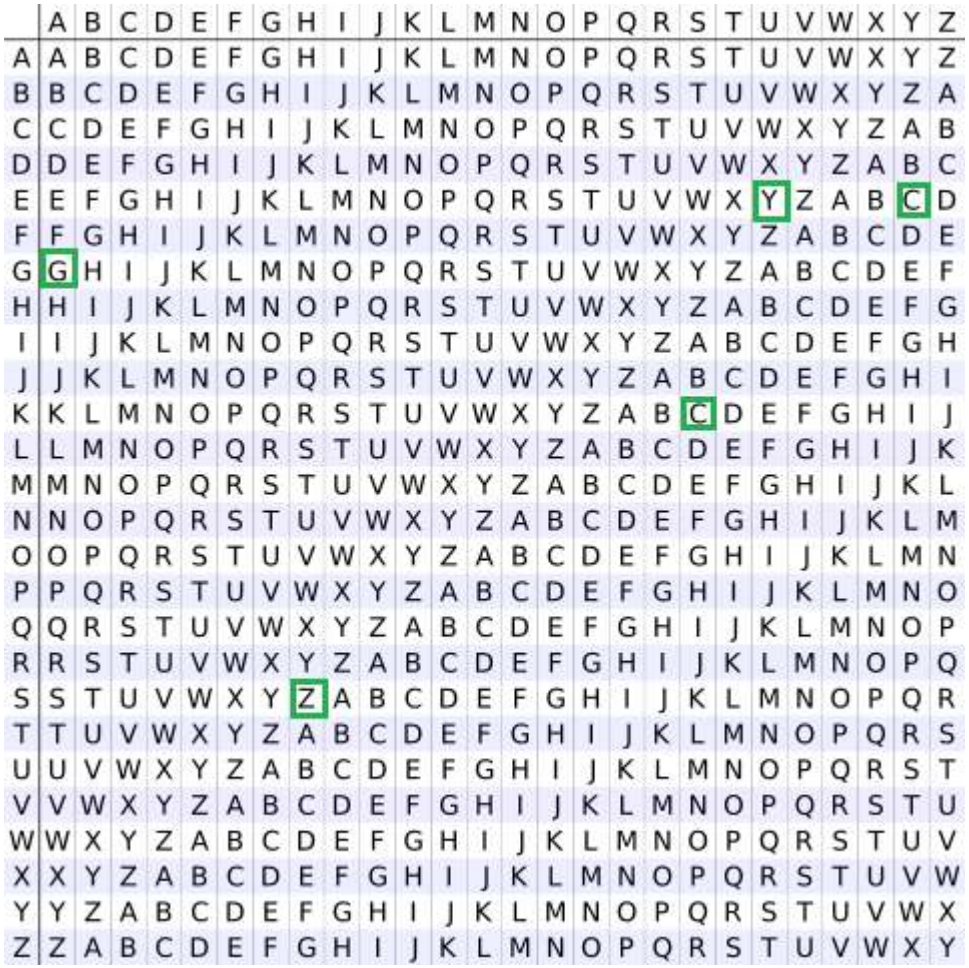
Decryption:
CF -> HE
SU -> LX
MP -> LO

Security of Playfair Cipher

- security much improved over monoalphabetic
- since have 26 x 26 = 676 digrams
- would need a 676 entry frequency table to analyse (verses 26 for a monoalphabetic)
- and correspondingly more ciphertext
- was widely used for many years
  - eg. by US & British military in WW1
- it **can** be broken, given a few hundred letters
- since still has much of plaintext structure

Vigenère Cipher (Encryption)

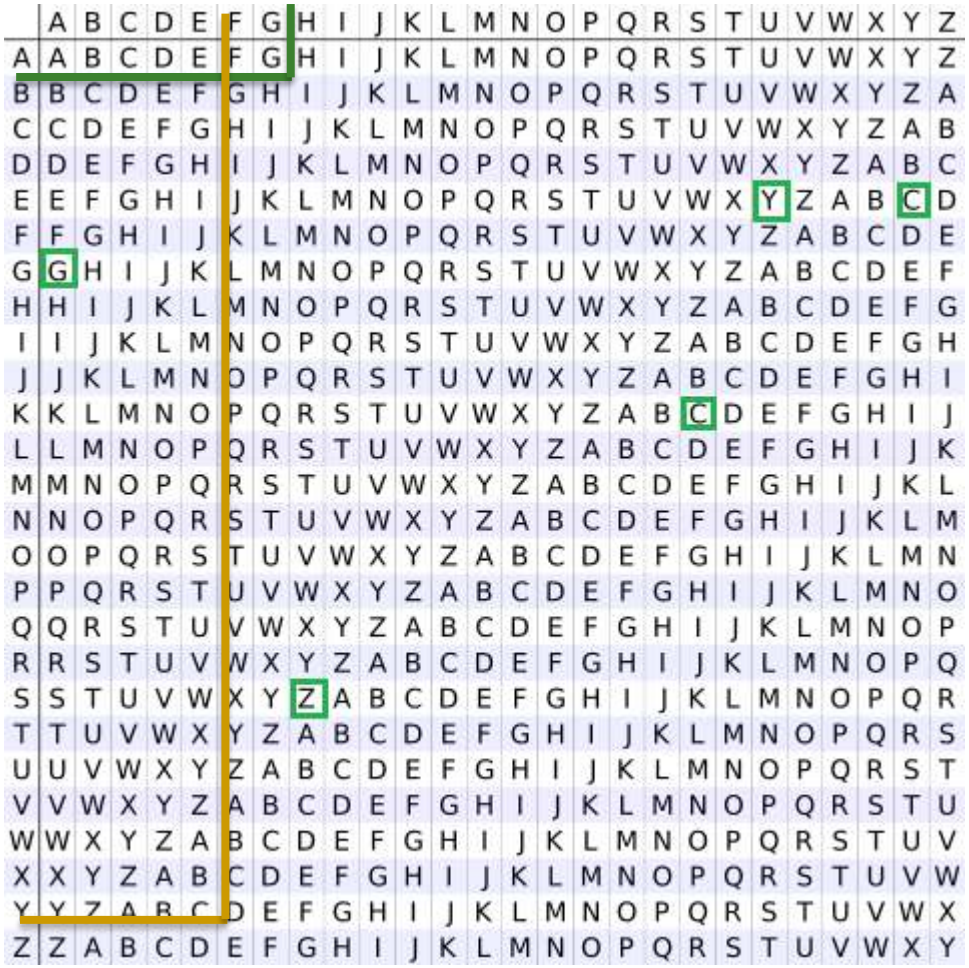
- Plaintext : GEEKSFORGEEKS
  - Length = 13
- Keyword : AYUSH
  - The keyword "AYUSH" should be repeated to match the plain text length
  - So, the key generated is "AYUSHAYUSHAYU"
  - The rows are the plain text
  - The columns are the key
  - The intersections are the cipher.
- Cipher is GCYCZFMLYLEIM



Public domain image from http://commons.wikimedia.org/wiki/File:Caesar3.svg

Vigenère Cipher (Decryption)

- Cipher is GCYCZFMLYLEIM
- the key is "AYUSHAYUSHAYU"
- Using the same table
  - Rows represent the key
  - The intersection represent the cipher
  - The corresponding column is the plain
- Plaintext : GEEKSFORGEEKS



Public domain image from http://commons.wikimedia.org/wiki/File:Caesar3.svg

How to attack substitution:

Frequency Analysis

- Letters in a natural language, like English, are not uniformly distributed.
- Knowledge of letter frequencies, including pairs and triples can be used in cryptologic attacks against substitution ciphers.

a:	8.05%	b:	1.67%	c:	2.23%	d:	5.10%
e:	12.22%	f:	2.14%	g:	2.30%	h:	6.62%
i:	6.28%	j:	0.19%	k:	0.95%	l:	4.08%
m:	2.33%	n:	6.95%	o:	7.63%	p:	1.66%
q:	0.06%	r:	5.29%	s:	6.02%	t:	9.67%
u:	2.92%	v:	0.82%	w:	2.60%	x:	0.11%
y:	2.04%	z:	0.06%				

8.1: Letter frequencies in the book *The Adventures of Tom Sawyer*, by Twain.

Substitution Ciphers: One-Time Pads

- There is one type of substitution cipher that is absolutely unbreakable.
  - The **one-time pad** was invented in 1917 by Joseph Mauborgne and Gilbert Vernam
  - We use a block of shift keys, $(k_1, k_2, \ldots, k_n)$, to encrypt a plaintext, M, of length n, with each shift key being chosen uniformly at random.
- Since each shift is random, every ciphertext is equally likely for any plaintext.

Example

Message:	S	E	N	D	H	E	L	P
Letters changed into corresponding numbers:	19	5	14	4	8	5	12	16
One-time pad:	7	9	5	2	12	1	0	6
Add the plaintext and the OTP:	26	14	19	6	20	6	12	22
Ciphertext:	Z	N	S	F	T	F	L	V

$E_n(x) = (x + n) \bmod 26$
$D_n(x) = (x - n) \bmod 26$

Substitution Ciphers: Hill Cipher

- Developed by mathematician Lester Hill in 1929, the encryption algorithm takes m successive plaintext letters and substitutes with m ciphertext letters.
- Each letter is represented by a number modulo 26. (E.g. A = 0, B = 1, …, Z = 25)
- To encrypt a message, each block of *n* letters is multiplied by an invertible $n \times n$ matrix, modulus 26.
- To decrypt the message, each block is multiplied by the inverse of the matrix used for encryption.

- $c_1=(k_{11}p_1+k_{12}p_2) \bmod 26$
- $c_2=(k_{21}p_1+k_{22}p_2) \bmod 26$

- This can be expressed in term of column vectors and matrices:

$$\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} mod\ 26$$

Example:

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
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

- Consider the plain text "**july**", and use the encryption key:

$$\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$$

- The first two letters (**ju**) of the plaintext are represented by the (9, 20) and the last two letters (**ly**) are (11, 24) . Then :

- $C_1 = E_k(P_1) = KP \ mod\ 26$

$$= (9, 20)\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix} = (99 + 60, 72 + 140) = (3, 4)\ mod\ 26$$

- $C_2 = E_k(P_2) = KP \bmod 26$

$$= (11, 24)\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix} = (121 + 72, 88 + 168) = (11, 22)\ mod\ 26$$

- so the ciphertext of "**july**" is "**DELW**".

- **Decryption** requires using the inverse of the matrix K, defined as $KK^{-1} = K^{-1}K = I$, where I is the matrix that is all 0's except for ones along the diagonal from upper left to lower right. Use the decryption formula:

$$P = D_k(C) = K^{-1}C \bmod 26 = K^{-1}\,KP = P$$

- $P_1 = (3,4)\begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix} = (9,20) mod\ 26$
- $P_2 = (11,22)\begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix} = (11,24) mod\ 26$
- The strength of the Hill cipher is that it completely hides single-letter frequencies and thus it is secure against ciphertext only attack. Indeed with Hill, the use of a larger matrix hides more frequency information

How to find the inverse key matrix

Assume the key matrix is K as follows $k = \begin{bmatrix} 11 & 8 \\ 3 & 7 \end{bmatrix}$

We calculate the determinant

Determinant $Det = 11 * 7 - 3 * 8 \ mod\ 26 = 53\ mod\ 26 = 1$

Then, we find Adj (k) $Adj(k) = \begin{bmatrix} 7 & -8 \\ -3 & 11 \end{bmatrix}$

Finally, we calculate inverse k

$$K^{-1} = \det * \ Adj(k)$$

$$k^{-1} = 1 * \begin{bmatrix} 7 & -8 \\ -3 & 11 \end{bmatrix} mod\ 26$$

$$k^{-1} = \begin{bmatrix} 7 & 18 \\ 23 & 11 \end{bmatrix} mod\ 26$$

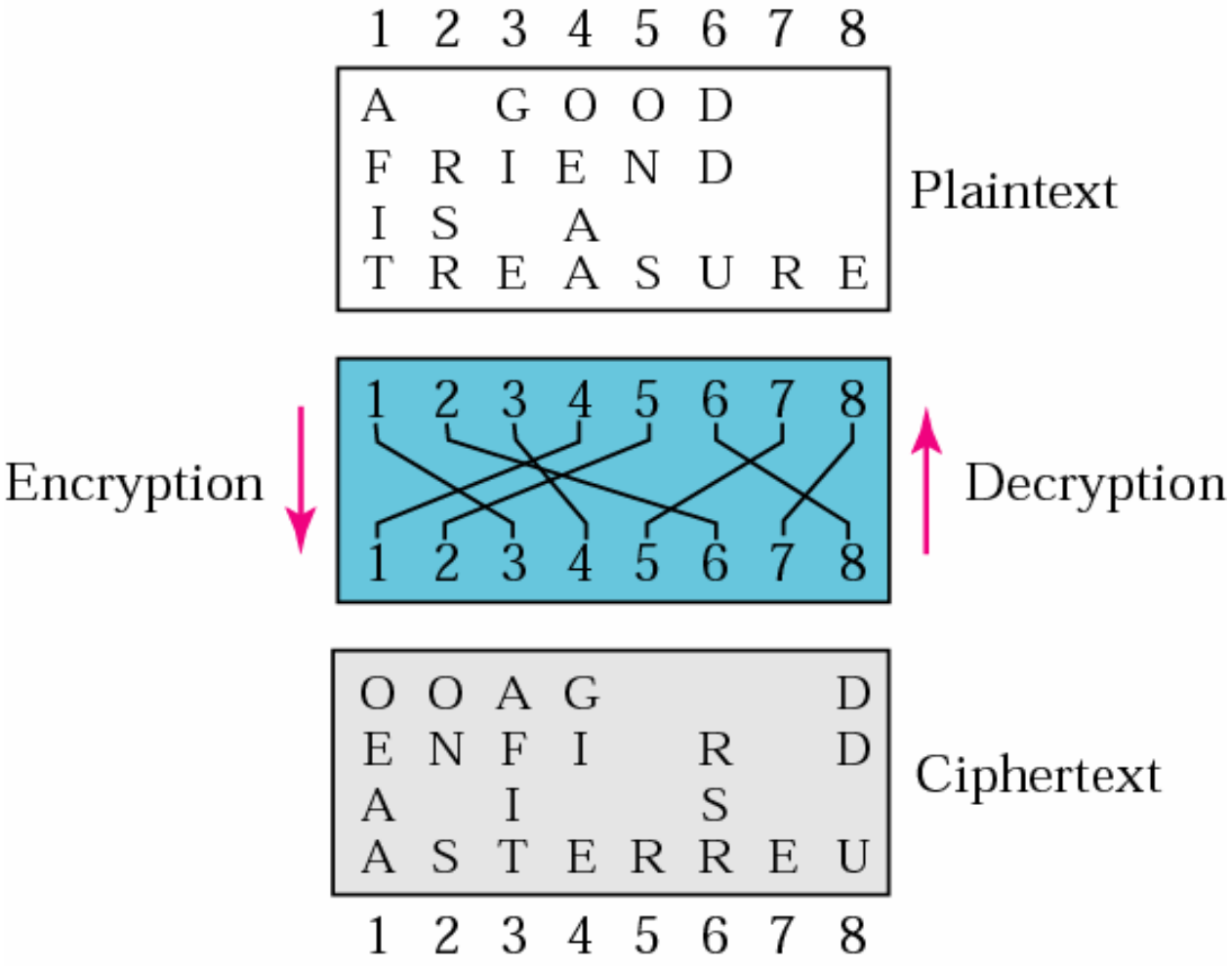
So, the decryption key is $\begin{bmatrix} 7 & 18 \\ 23 & 11 \end{bmatrix}$

Transposition Cipher

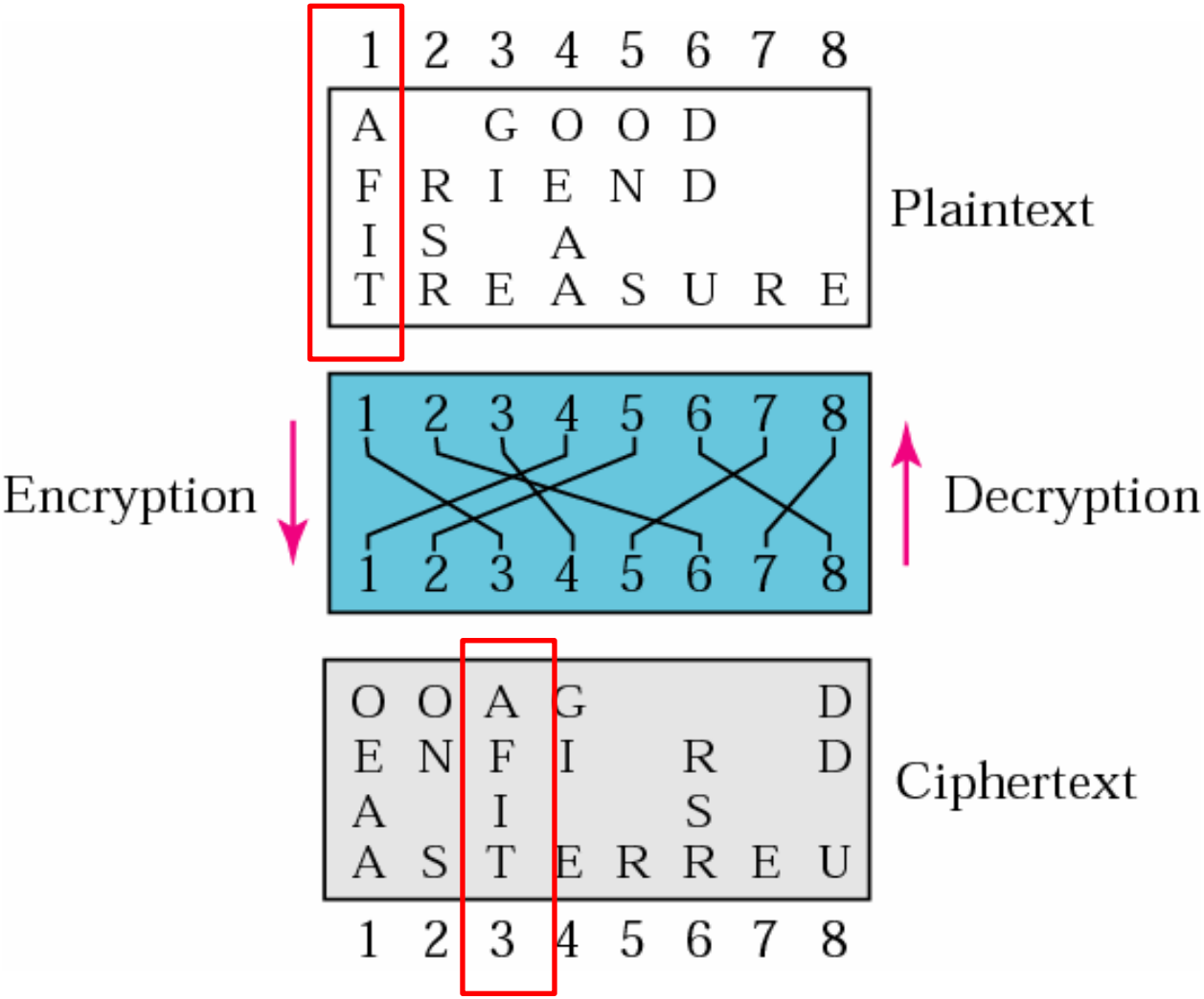
Transposition Cipher

- In a transposition cipher, the characters retain their plaintext form but change their positions to create the ciphertext
- The text is organized into a two-dimensional table, and the columns are interchanged according to a key
- The key defines which columns should be swapped

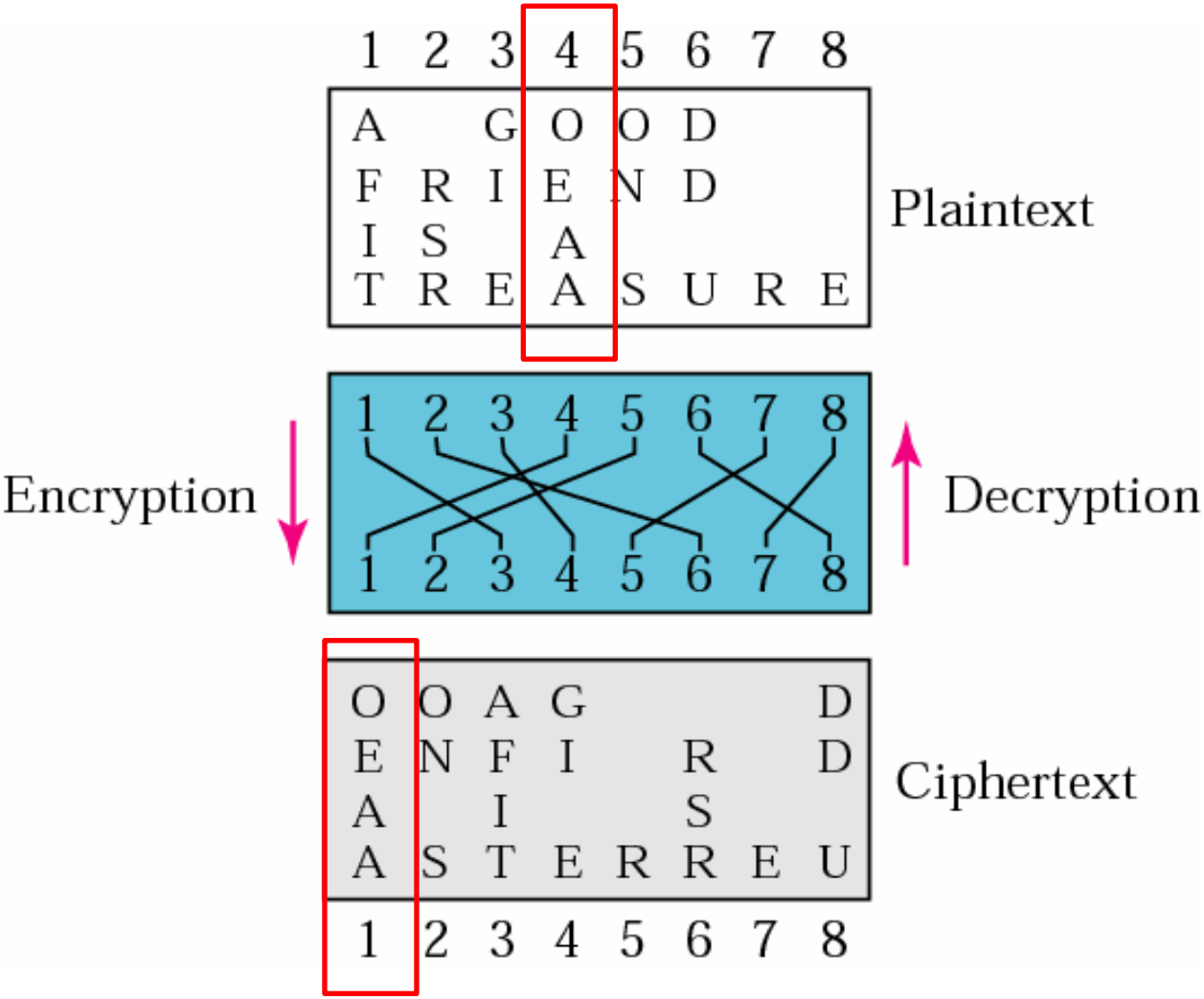
Example


1 2 3 4 5 6 7 8
A G O O D
F R I E N D
I S A
T R E A S U R E
Plaintext
Encryption
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8
Decryption
O O A G D
E N F I R D
A I S
A S T E R R E U
Ciphertext
1 2 3 4 5 6 7 8

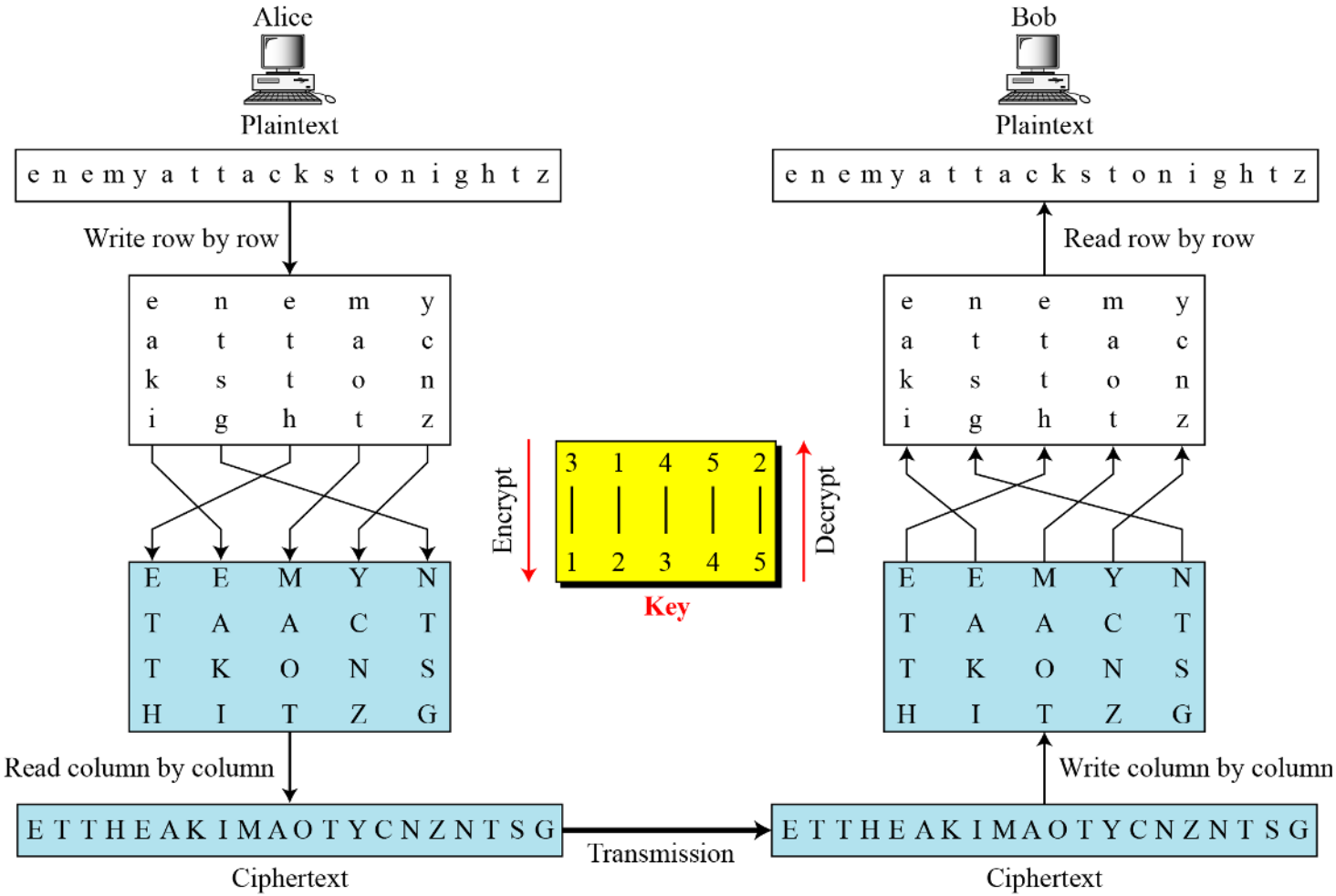
Example


1 2 3 4 5 6 7 8
A G O O D
F R I E N D
I S A
T R E A S U R E
Plaintext
Encryption
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8
Decryption
O O A G D
E N F I R D
A I S
A S T E R R E U
Ciphertext
1 2 3 4 5 6 7 8

Example


1 2 3 4 5 6 7 8
A G O O D
F R I E N D
I S A
T R E A S U R E
Plaintext
Encryption
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8
Decryption
O O A G D
E N F I R D
A I S
A S T E R R E U
Ciphertext
1 2 3 4 5 6 7 8

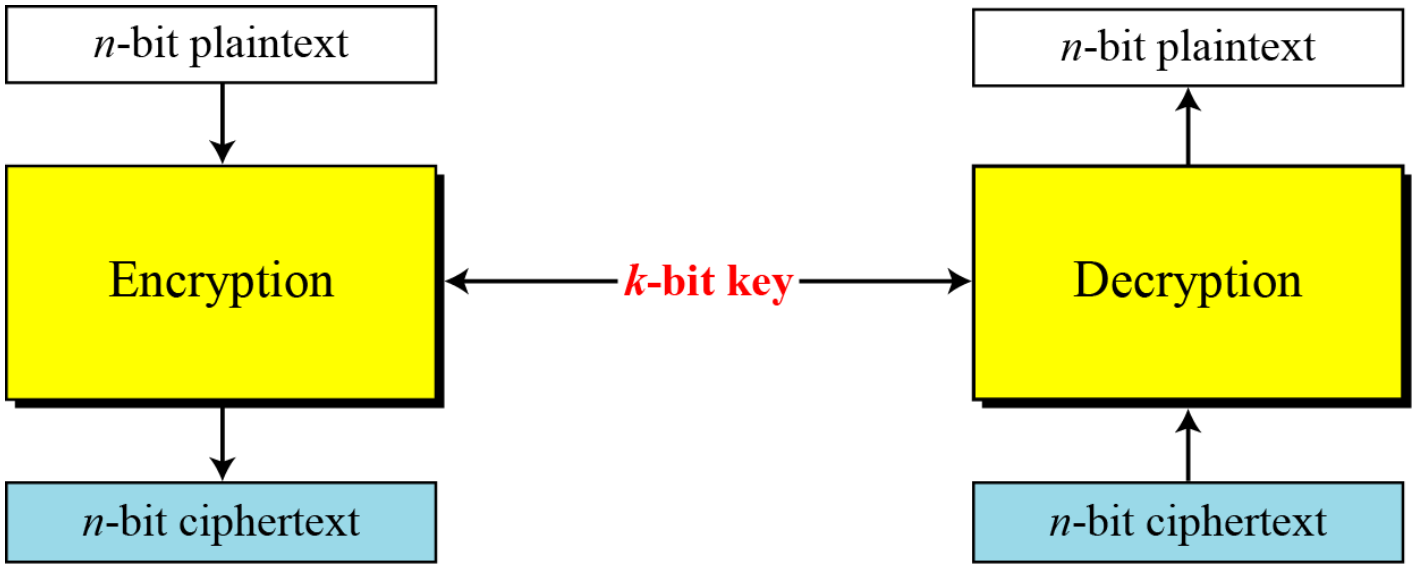
Example


Alice
Plaintext
e n e m y a t t a c k s t o n i g h t z
Write row by row
e n e m y
a t t a c
k s t o n
i g h t z
E E M Y N
T A A C T
T K O N S
H I T Z G
Read column by column
E T T H E A K I M A O T Y C N Z N T S G
Ciphertext
Encrypt
3 1 4 5 2
1 2 3 4 5
Key
Decrypt
Transmission
Bob
Plaintext
e n e m y a t t a c k s t o n i g h t z
Read row by row
e n e m y
a t t a c
k s t o n
i g h t z
E E M Y N
T A A C T
T K O N S
H I T Z G
Write column by column
E T T H E A K I M A O T Y C N Z N T S G
Ciphertext

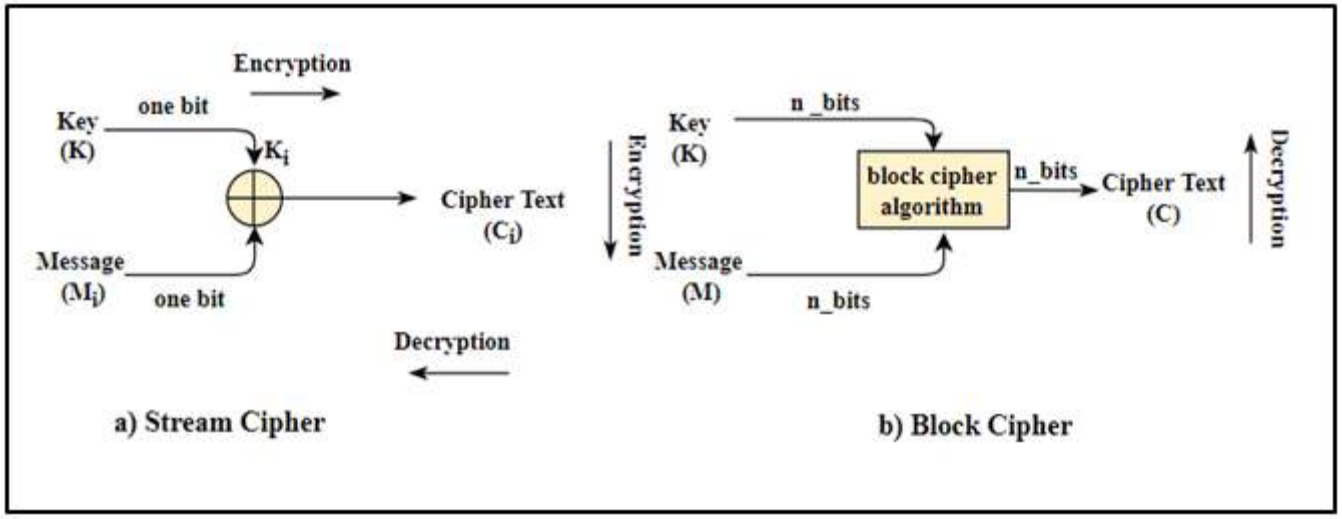
MODERN BLOCK CIPHERS

- *A symmetric-key modern block cipher encrypts an n-bit block of plaintext or decrypts an n-bit block of ciphertext.*
- *The encryption or decryption algorithm uses a k-bit key.*

A modern block cipher



Block Ciphers



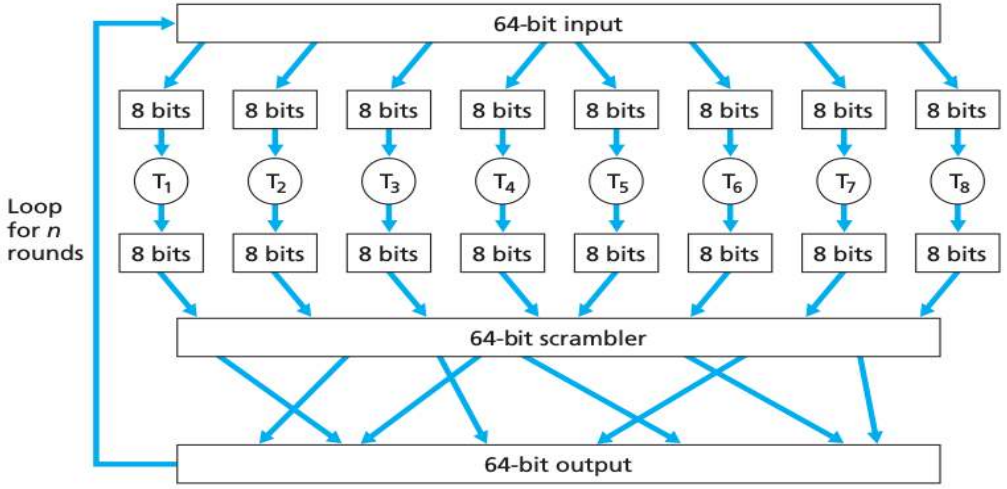
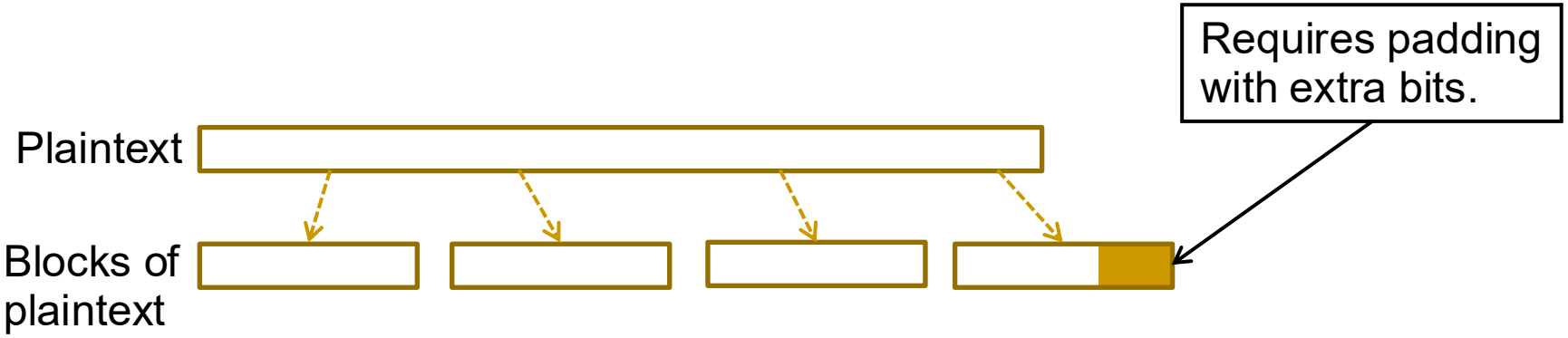


Figure 8.5 ♦ An example of a block cipher

Block Ciphers

- In a **block cipher:**
  - Plaintext and ciphertext have fixed length b (e.g., 128 bits)
  - A plaintext of length n is partitioned into a sequence of m **blocks**, P[0], …, P[m–1], where $n \leq bm < n + b$
- Each message is divided into a sequence of blocks and encrypted or decrypted in terms of its blocks.



Padding

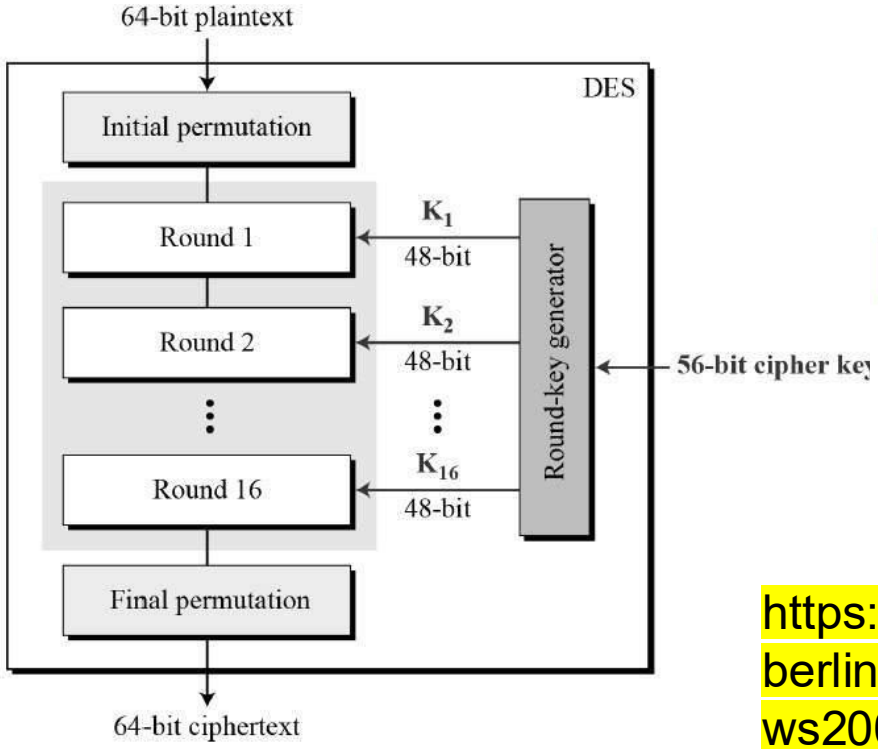
- Block ciphers require the length n of the plaintext to be a multiple of the block size b
- Padding the last block needs to be unambiguous (cannot just add zeroes)

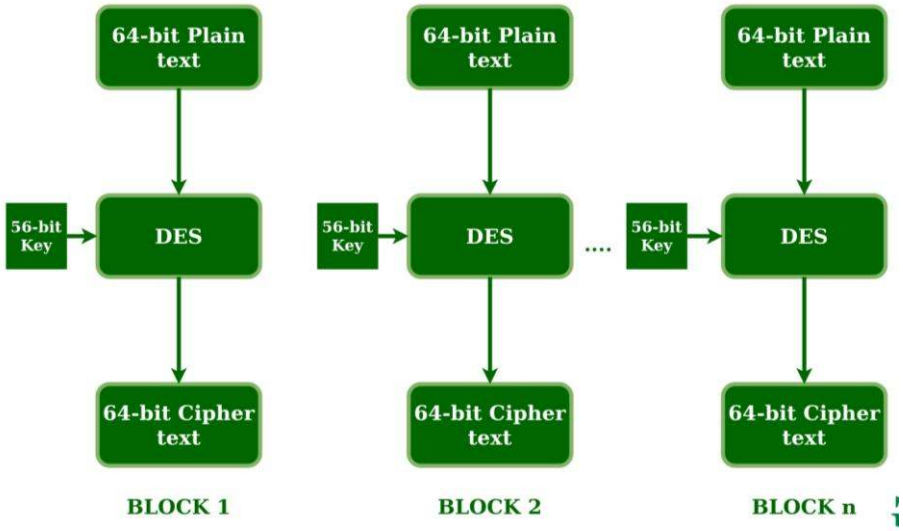
Padding

- Example for b = 64 bits (8 bytes)
  - Plaintext: “Roberto” (7 bytes)
  - Padded plaintext: “Roberto9” (8 bytes), where 9 denotes the number and not the character
- We need to always pad the last block, which may consist only of padding

Block Ciphers in Practice

- Data Encryption Standard (DES)
  - Developed by IBM and adopted by NIST in 1977
  - 64-bit blocks and 56-bit keys
  - Small key space makes exhaustive search attack feasible since late 90s

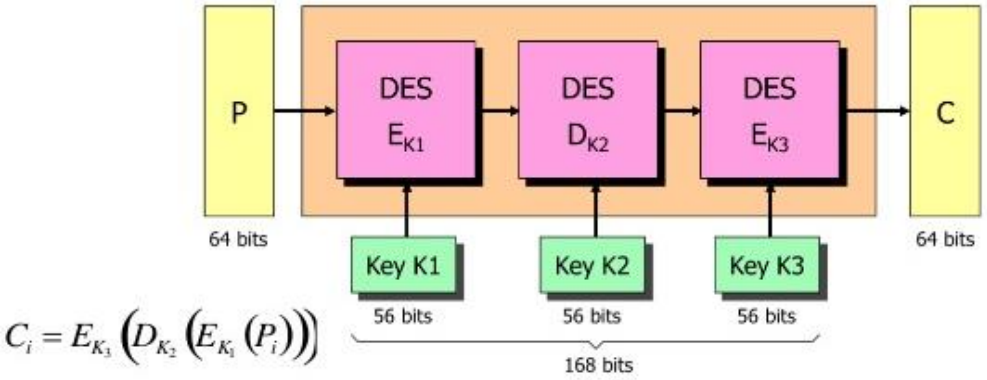




https://page.math.tu-berlin.de/~kant/teaching/hess/krypto-ws2006/des.htm

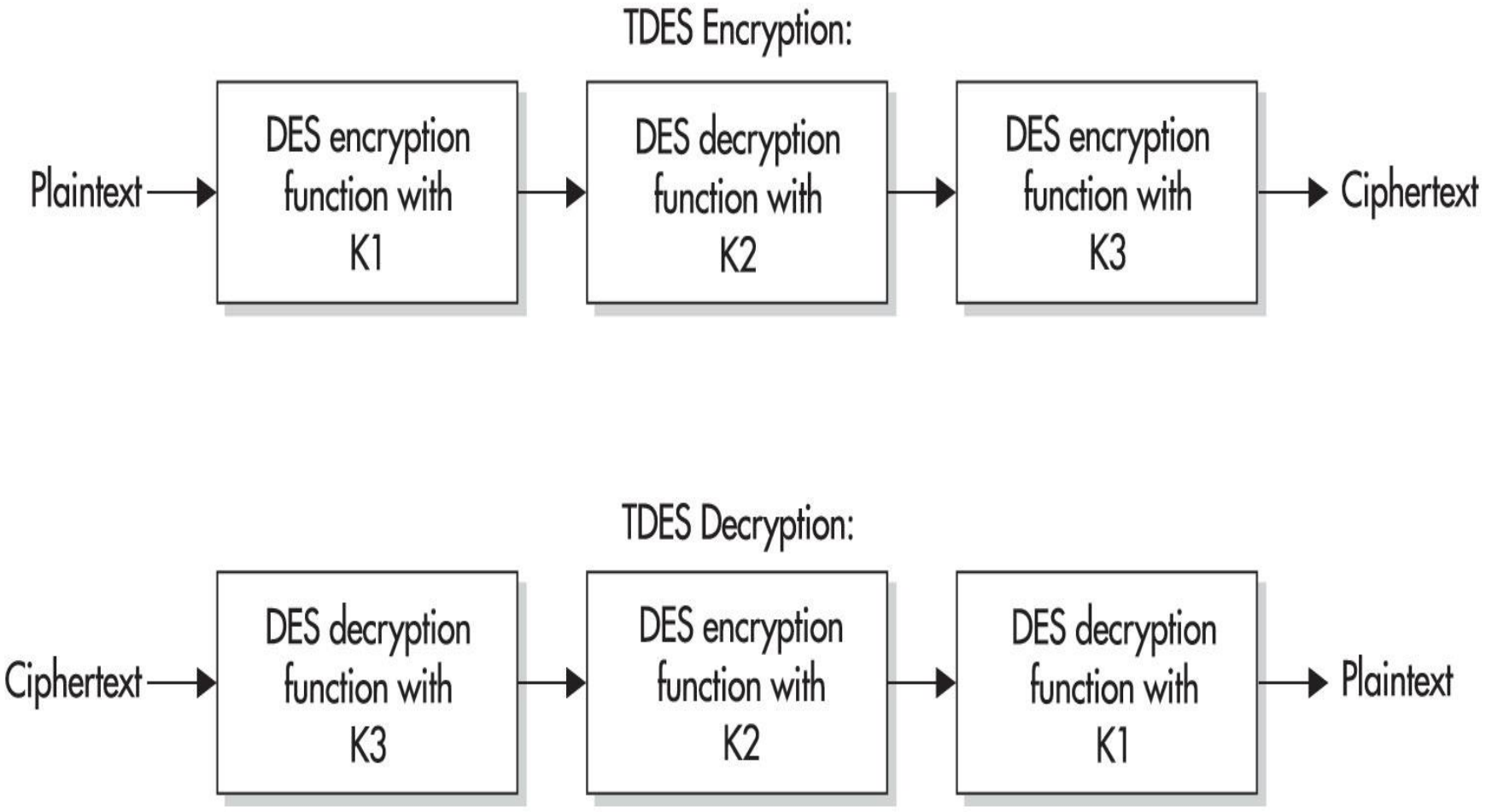
Block Ciphers in Practice

- Triple DES (3DES)
  - Nested application of DES with three different keys KA, KB, and KC
  - Effective key length is **168 bits**, making exhaustive search attacks unfeasible
  - $C = E_{KC}(D_{KB}(E_{KA}(P)))$; $P = D_{KA}(E_{KB}(D_{KC}(C)))$
  - Equivalent to DES when KA=KB=KC (backward compatible)



True cryptographic strength of 3DES key is 2x56 bits = 112 bits

Triple DES

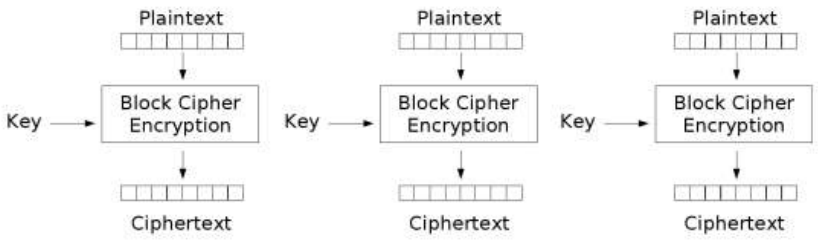

TDES Encryption:
Plaintext
DES encryption function with K1
DES decryption function with K2
DES encryption function with K3
Ciphertext
TDES Decryption:
Ciphertext
DES decryption function with K3
DES encryption function with K2
DES decryption function with K1
Plaintext

Block Ciphers in Practice

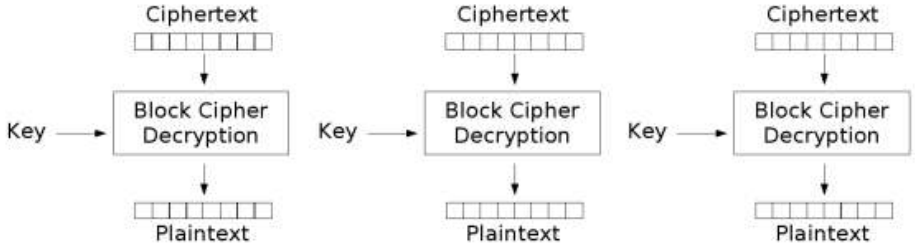
- Advanced Encryption Standard (AES)
  - Selected by NIST in 2001 through open international competition and public discussion
  - 128-bit blocks and several possible key lengths: 128, 192 and 256 bits
  - Exhaustive search attack not currently possible
- International Data Encryption Algorithm (IDEA)
  - Uses a 128-bit key and is used in Pretty Good Privacy (PGP) encryption for e-mail systems
- RC5
  - Developed at MIT, and allows for variable length keys.
- Blowfish
  - Designed by Bruce Schneier in 1993 allows for variable length keys up to 448 bits

Block Cipher Modes

- A block cipher mode describes the way a block cipher encrypts and decrypts a sequence of message blocks.
- **Electronic Code Book (ECB) Mode:**
  - Block P[i] encrypted into ciphertext block $C[i] = E_K(P[i])$
  - Block C[i] decrypted into plaintext block $M[i] = D_K(C[i])$



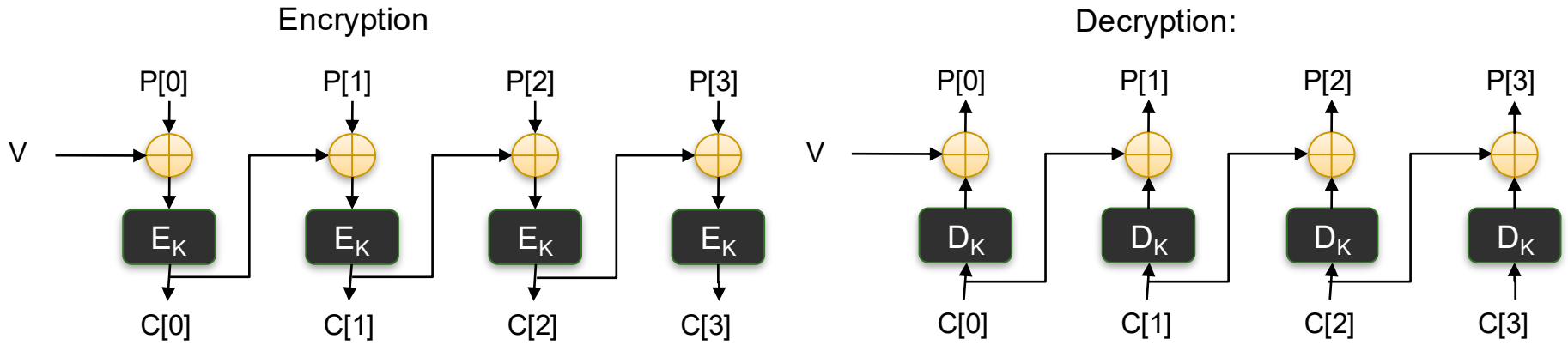
Electronic Codebook (ECB) mode encryption



Electronic Codebook (ECB) mode decryption

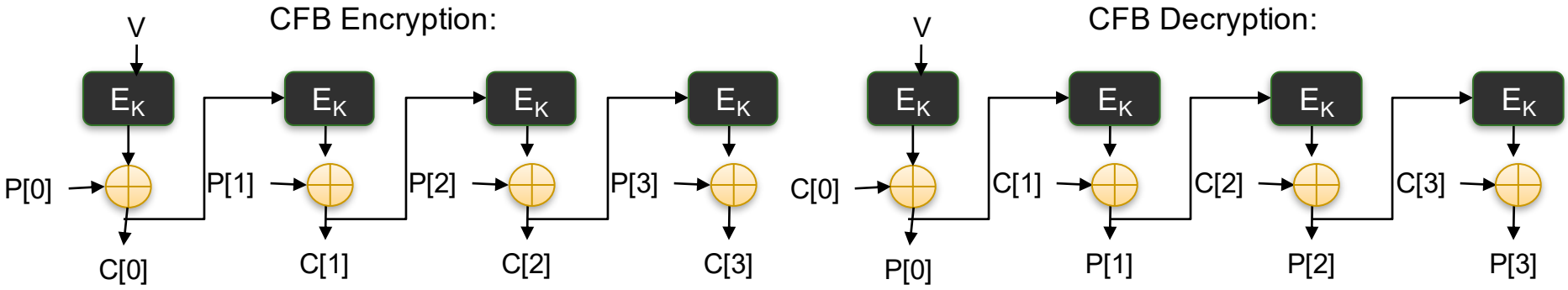
Cipher Block Chaining (CBC) Mode

- **Cipher Block Chaining (CBC) Mode**
  - The 1st block is XOR with an initialization vector
  - The previous ciphertext block is XOR the current plaintext block $C[i] = E_K(C[i-1] \oplus P[i])$
  - Decryption: $P[i] = C[i-1] \oplus D_K(C[i])$



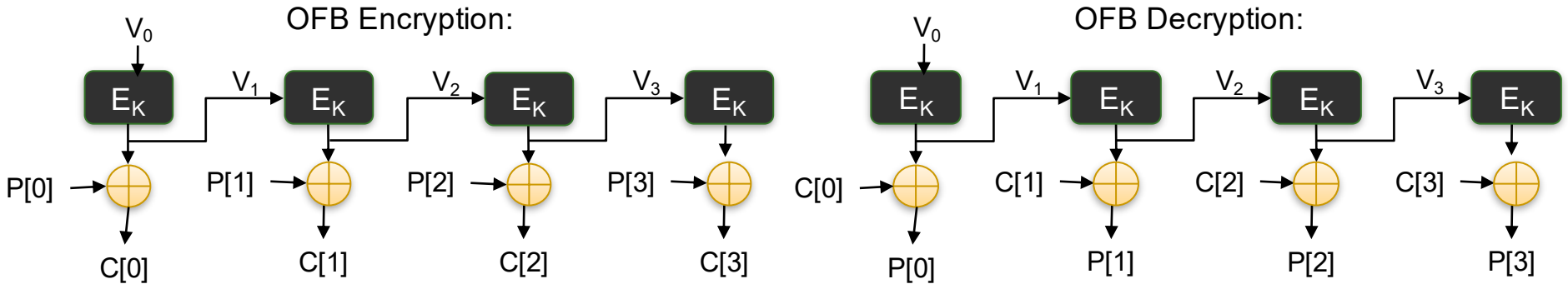
Cipher Feedback (CFB) Mode

- **Cipher Feedback (CFB) Mode**
  - Similar to Cipher-Block Chaining
  - Encrypt the current block with previous ciphertex block $C[i] = E_K(C[i-1]) \oplus P[i]$
  - Decryption: $P[i] = C[i] \oplus E_K(C[i-1])$



Output Feedback (OFB) Mode

- **Output Feedback (OFB) Mode**
  - Similar to one-time pad, but pad with generated cipher blocks $V_1=E_k(V_{i-1})$ and begin with initialization vector V_0.
  - Encrypt with $C_i = V_i \oplus B_i$
  - Decryption with $B_i = V_i \oplus C_i$



Counter (CTR) Mode

- Counter (CTR) Mode
  - Similar to OFB, but uses a seed s
  - $V_i = E_k(s + i - 1)$
  - CTR mode can be performed in parallel and can recover from dropped blocks

ATTACKING CONVENTIONAL ENCRYPTION SCHEMES

Objective:

To recover the key and not just the messages, so that the attacker can easily compromise all future and past cipher texts.

General Approaches

- Cryptanalysis
- Brute force attack

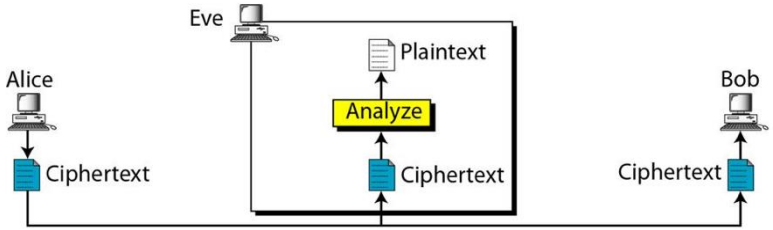
1. Cryptanalysis

- Based on what is known to the attacker, cryptanalytic attempts are classified as:
  - Ciphertext Only
  - Known Plaintext
  - Chosen Plaintext
  - Chosen Ciphertext
- Encryption algorithm is assumed to be known for all the attacks.
- In each case, the object is to determine the key (and plaintext) that was used.

Ciphertext only Attack

The attacker has access only to the ***ciphertext*** of several messages encrypted through same encryption scheme. The knowledge of the plaintext is minimal. His job is to find plaintext, or key.

Known



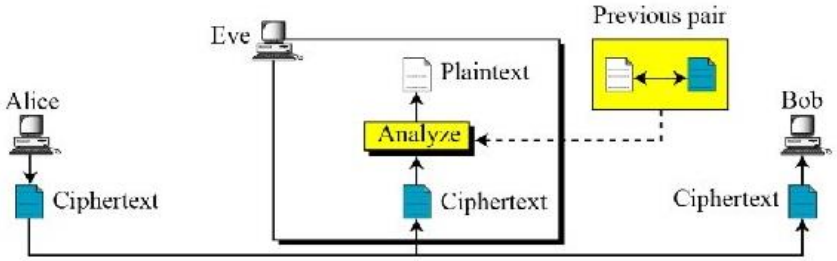
$C_1=E_k(P_1)$, $C_2=E_k(P_2)$,……, $C_J=E_k(P_J)$

To be Known

Key (K) and/or P_1, P_2, P_3,……., P_j

Known Plaintext Attack

The attacker has access to the plaintexts as well as to their corresponding ciphertexts. He intends to find key.



Known

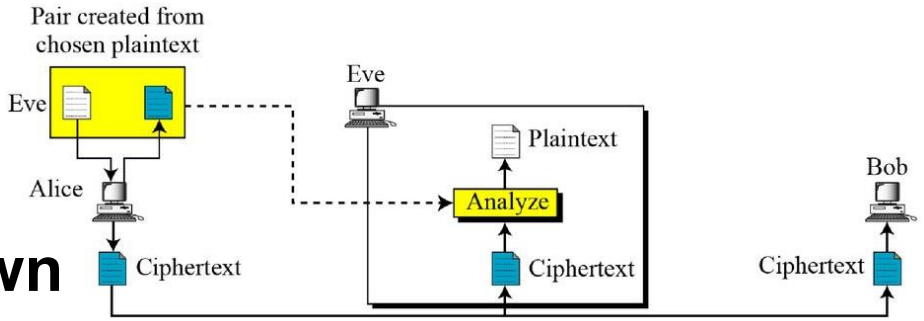
$P1, C_1=E_k(P_1),\ P_2, C_2=E_k(P_2), \ldots\ldots, P_J, C_J=E_k(P_J)$

To be Known

Key (K) to get P1, P2, P3,……., P_j

Chosen Plaintext Attack

The attacker has access to the plaintext as well as to their corresponding ciphertext and also he has ability to encrypt texts of his own choice. That is possible when an encryption box embedded with a secure key comes in the hands of attacker or the attacker can send his own plaintexts to the owner of the secret key to encrypt. His job is to deduce either key to get plaintexts.



Known

$P_1, C_1 = E_k(P_1)$, $P_2, C_2 = E_k(P_2)$, ……, P_J, $C_J = E_k(P_J)$

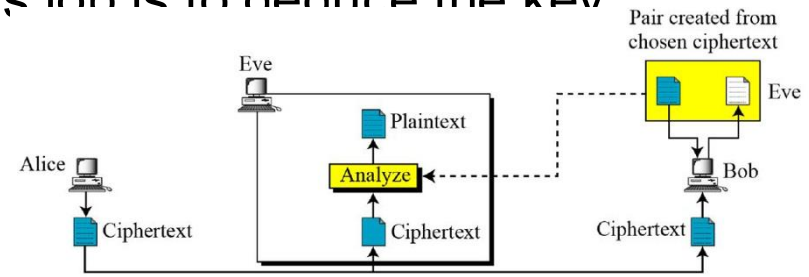
(attacker can choose P_1, P_2, ….,$P_{j,}$)

To be Known

Key (K) to get P1, P2, P3,……., P_j

Chosen Ciphertext Attack

In contrast to chosen plaintext attack, here an attacker can choose different ciphertexts to be decrypted and he has access to the decrypted plaintexts. The attacker has access to a decryption box or can send to the owner his ciphertexts to decrypt. His job is to deduce the key



Known

$C_1, P_1 = D_k(C_1),\ C_2, P_2 = D_k(C_2), \ldots\ldots, C_J, P_J = D_k(C_J)$

(attacker can choose $C_1, C_2, \ldots, C_j$)

To be Known

Key (K)

2. Brute Force Search

- A brute-force attack involves trying every possible key until an intelligible translation of the ciphertext into plaintext is obtained.
- Most basic attack, proportional to key size
- Assume either know / recognise plaintext

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/μs	Time required at 10^6 decryptions/μs
32	$2^{32} = 4.3 \times 10^9$	2^{31} μs = 35.8 minutes	2.15 milliseconds
56 (DES)	$2^{56} = 7.2 \times 10^{16}$	2^{55} μs = 1142 years	10.01 hours
128 (AES)	$2^{128} = 3.4 \times 10^{38}$	2^{127} μs = 5.4×10^{24} years	5.4×10^{18} years
168 (Triple DES)	$2^{168} = 3.7 \times 10^{50}$	2^{167} μs = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	2×10^{26} μs = 6.4×10^{12} years	6.4×10^6 years

Thank you