Information Security

CS3002

Lecture 4

28th August 2024

Dr. Rana Asif Rehman

Email: r.asif@lhr.nu.edu.pk

Classical Symmetric Ciphers

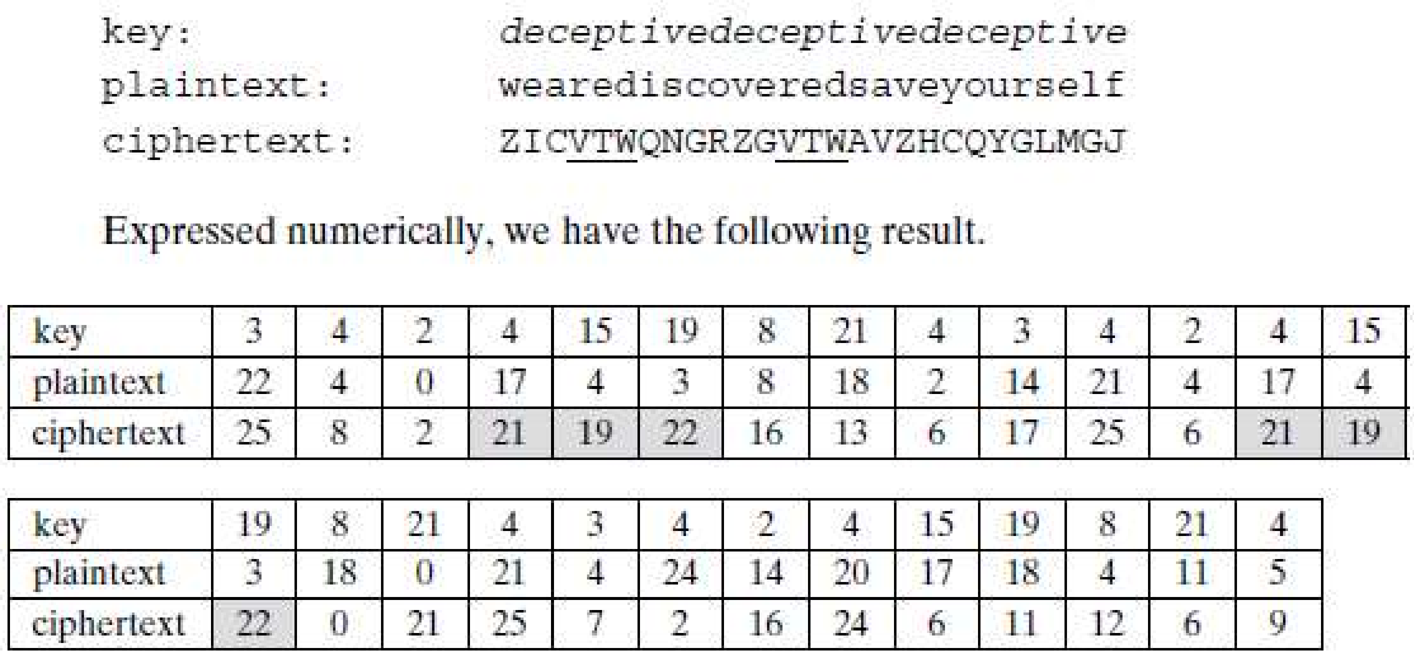
- Classical Substitution Ciphers
  - Caesar Cipher
  - Monoalphabetic Cipher
  - Vigenere Cipher
  - One-Time Pad

3. Vigenère Cipher

- Simplest polyalphabetic substitution cipher.
- Effectively multiple caesar ciphers.
- Key is multiple letters long $K = k_1\ k_2\ \ldots\ k_d$
- i^{th} letter specifies i^{th} alphabet to use.
- Use each alphabet in turn.
- Repeat from start after d letters in message.
- Decryption simply works in reverse.

Example of Vigenère Cipher

$$C_i = (p_i + k_{i \bmod m}) \bmod 26$$



```
key:            deceptivedeceptivedeceptive
plaintext:      wearediscoveredsaveyourself
ciphertext:     ZICVTWQNGRZGVTWAVZHCQYGLMGJ
```

Expressed numerically, we have the following result.

key	3	4	2	4	15	19	8	21	4	3	4	2	4	15
plaintext	22	4	0	17	4	3	8	18	2	14	21	4	17	4
ciphertext	25	8	2	21	19	22	16	13	6	17	25	6	21	19

key	19	8	21	4	3	4	2	4	15	19	8	21	4
plaintext	3	18	0	21	4	24	14	20	17	18	4	11	5
ciphertext	22	0	21	25	7	2	16	24	6	11	12	6	9

$$p_i = (C_i - k_{i \bmod m}) \bmod 26$$

Security of Vigenère Ciphers

- Have multiple ciphertext letters for each plaintext letter.
- Hence letter frequencies are obscured.
- But not totally lost (see frequency distribution)

Kasiski Method

- Method developed by Babbage / Kasiski.
- Repetitions in ciphertext give clues to period.
- Find same plaintext an exact period apart which results in the same ciphertext eg repeated “VTW” in previous example suggests size of 3 or 9
- Then attack each monoalphabetic cipher individually using same techniques as before.

Autokey Cipher

- Ideally want a key as long as the message
- Vigenère proposed the **autokey** cipher with keyword is prefixed to message as key knowing keyword can recover the first few letters use these in turn on the rest of the message.
- But still have frequency characteristics to attack
- E.g. given key *deceptive*

```
key:        deceptivewearediscoveredsav
plaintext:  wearediscoveredsaveyourself
ciphertext:ZICVTWQNGKZEIIGASXSTSLVVWLA
```

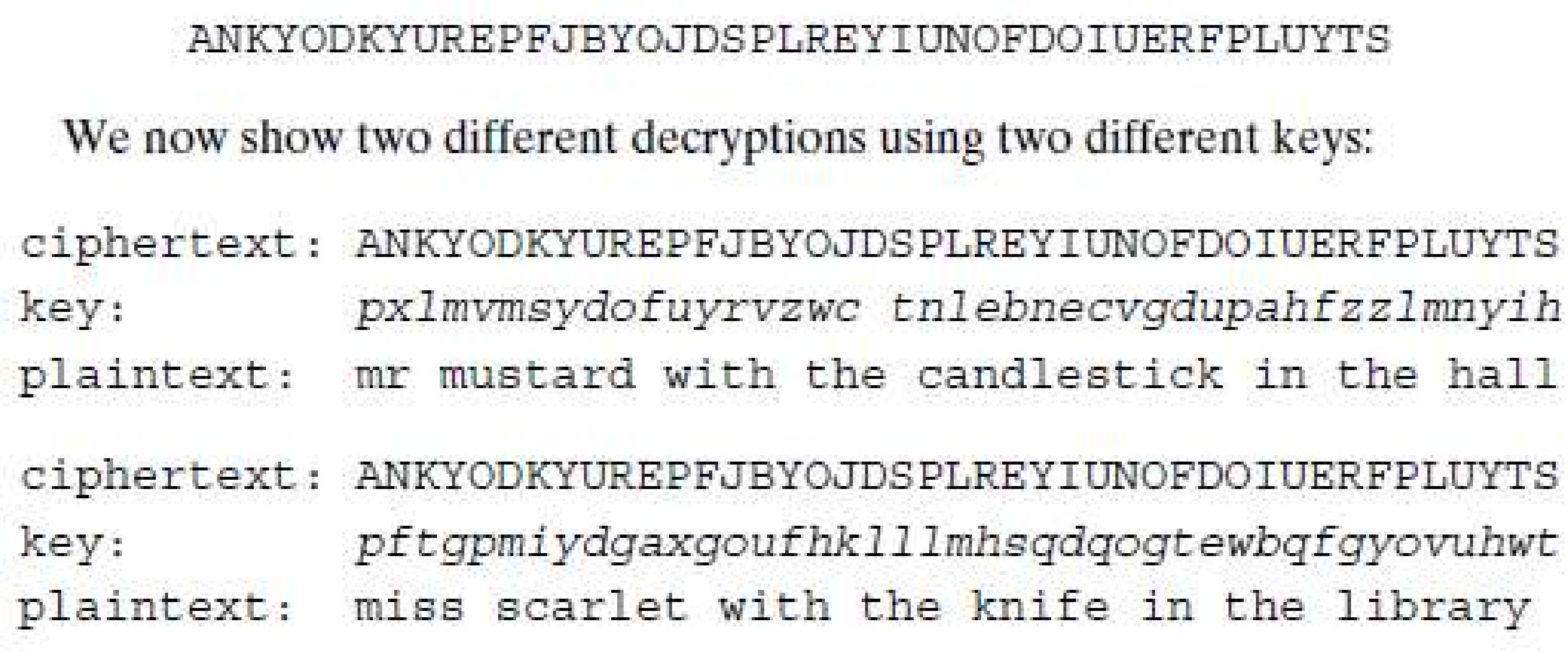
Autokey Cipher (cont.)

- Even this scheme is vulnerable to cryptanalysis. Because the key and the plaintext may share the same frequency distribution of letters, a statistical technique can be applied. For example, e enciphered by *e* can be expected to occur with a frequency of $(0.127)^2$ 0.016, whereas t enciphered by *t* would occur only about half as often. These regularities can be exploited to achieve successful cryptanalysis.

4. One-Time Pad

- If a truly random key as long as the message is used, the cipher will be secure called a One-Time pad.
- It is unbreakable since ciphertext bears no statistical relationship to the plaintext.
- Since for **any plaintext** & **any ciphertext** there exists a key mapping one to other.
- Can only use the key **once** though.
- Cannot use extensively and bears perfect secrecy.

One-Time Pad Security



ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS

We now show two different decryptions using two different keys:

```
ciphertext: ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS
key:        pxlmvmsydofuyrvzwc tnlebnecvgdupahfzzlmnyih
plaintext:  mr mustard with the candlestick in the hall

ciphertext: ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS
key:        pftgpmiydgaxgoufhkllmhsqdqogtewbqfgyovuhwt
plaintext:  miss scarlet with the knife in the library
```

Why not practical??

- Large number random key formation
- Distribution & protection among parties

Classical Symmetric Ciphers

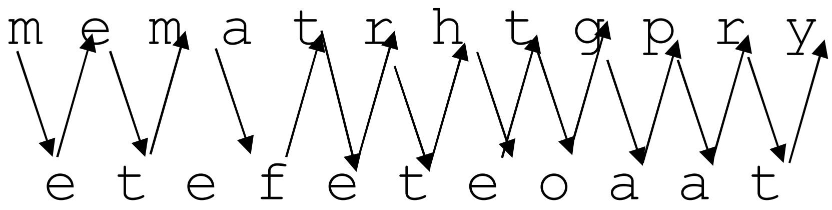
- Classical Transposition Ciphers
  - Rail Fence Cipher
  - Row Transposition Cipher

Transposition Ciphers

- Now consider classical **transposition** or **permutation** ciphers.
- A very different kind of mapping is achieved by performing some sort of permutation on the plaintext letters. This technique is referred to as a transposition cipher.
- These hide the message by rearranging the letter order without altering the actual letters used.
- Can recognise these since have the same frequency distribution as the original text.

1. Rail Fence Cipher

- Write message letters out diagonally over a number of rows
- Then read off cipher row by row
- The message ‘meet me after the toga party’
- e. g. write message out as:

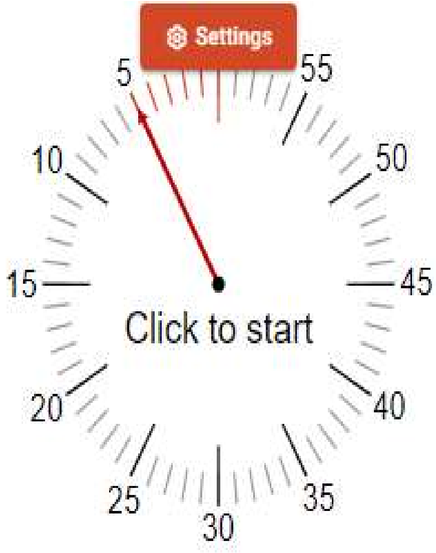


```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

- Giving ciphertext

MEMATRHTGPRYETEFETEOAAT

Activity Time



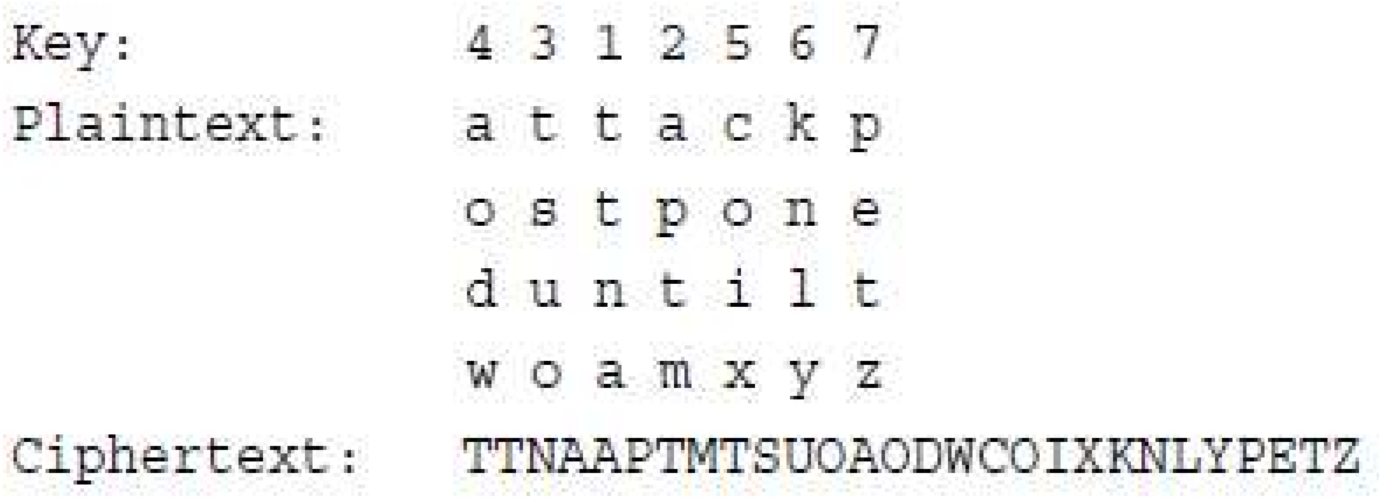
Decipher the following ciphertext by applying Rail Fence Cipher Algorithm using Key = 3

Plaintext: SIR RANA ASIF YOU ARE GREAT

Ciphertext: SASOEAIRNAIYURGETRAFAR

2. Row Transposition Ciphers

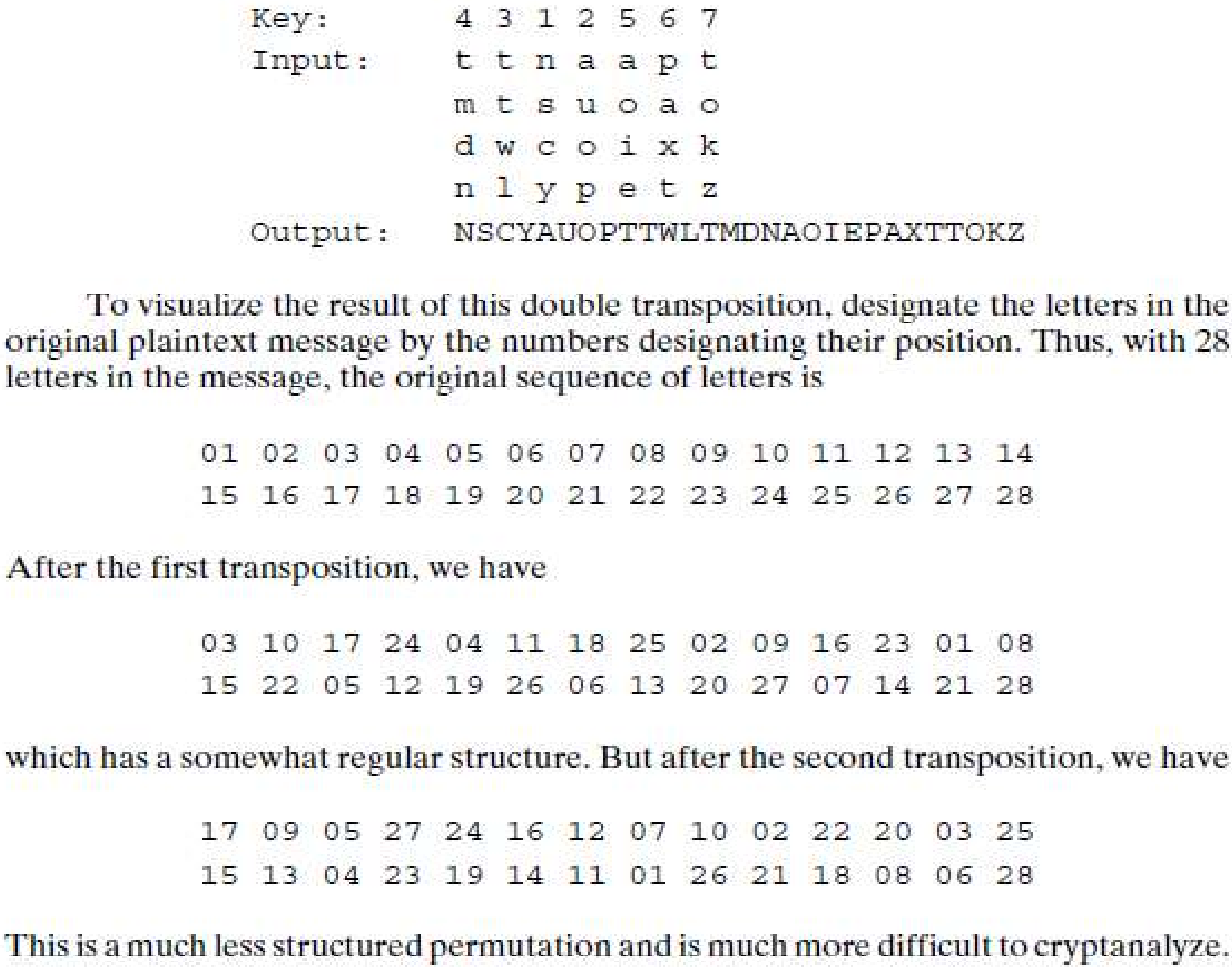
- A more complex transposition
- Write letters of message out in rows over a specified number of columns
- Then reorder the columns according to some key before reading off the rows



```
Key:          4 3 1 2 5 6 7
Plaintext:    a t t a c k p
              o s t p o n e
              d u n t i l t
              w o a m x y z
Ciphertext:   TTNAAPTMTSUOAODWCOIXKNLYPETZ
```

Problem with Transposition approach

- A pure transposition cipher is easily recognized because it has the same letter frequencies as the original plaintext.
- The transposition cipher can be made significantly more secure by performing more than one stage of transposition.



```
Key:        4 3 1 2 5 6 7
Input:      t t n a a p t
            m t s u o a o
            d w c o i x k
            n l y p e t z
Output:     NSCYAUOPTTWLTMDNAOIEPAXTTOKZ
```

To visualize the result of this double transposition, designate the letters in the original plaintext message by the numbers designating their position. Thus, with 28 letters in the message, the original sequence of letters is

```
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 26 27 28
```

After the first transposition, we have

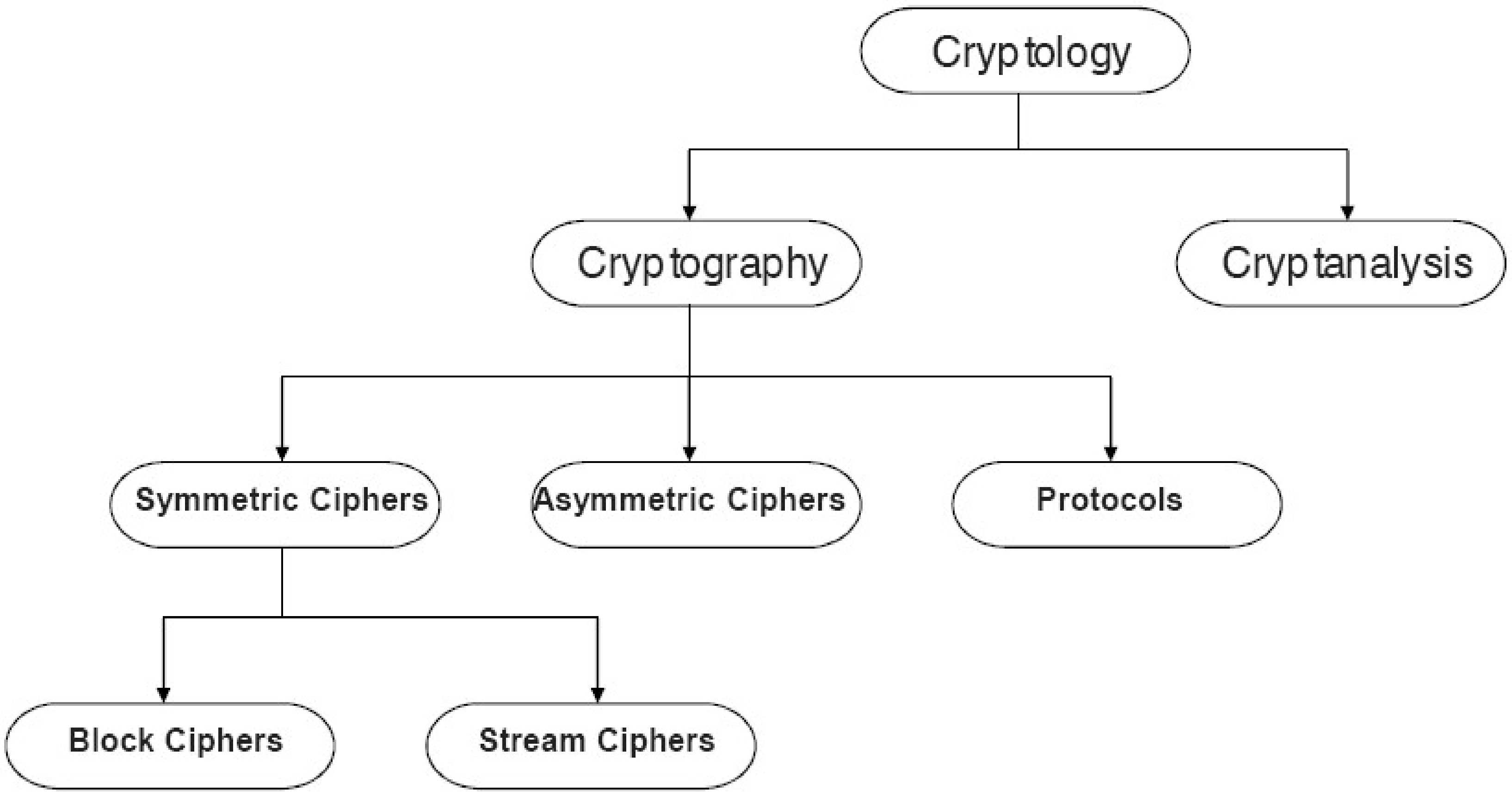
```
03 10 17 24 04 11 18 25 02 09 16 23 01 08
15 22 05 12 19 26 06 13 20 27 07 14 21 28
```

which has a somewhat regular structure. But after the second transposition, we have

```
17 09 05 27 24 16 12 07 10 02 22 20 03 25
15 13 04 23 19 14 11 01 26 21 18 08 06 28
```

This is a much less structured permutation and is much more difficult to cryptanalyze.

Classification of the Field of Cryptology



Adopted with thanks from: Chapter 1 of Understanding Cryptography by Christof Paar and Jan Pelzl

Why need bit-oriented ciphers?

- In previous lectures we discussed *character-oriented ciphers the problem is*
  - Keep language statistics
- Need bit-oriented ciphers
  - Numbers, graphics, audio and video data
  - Mixing at larger number of symbols increases security

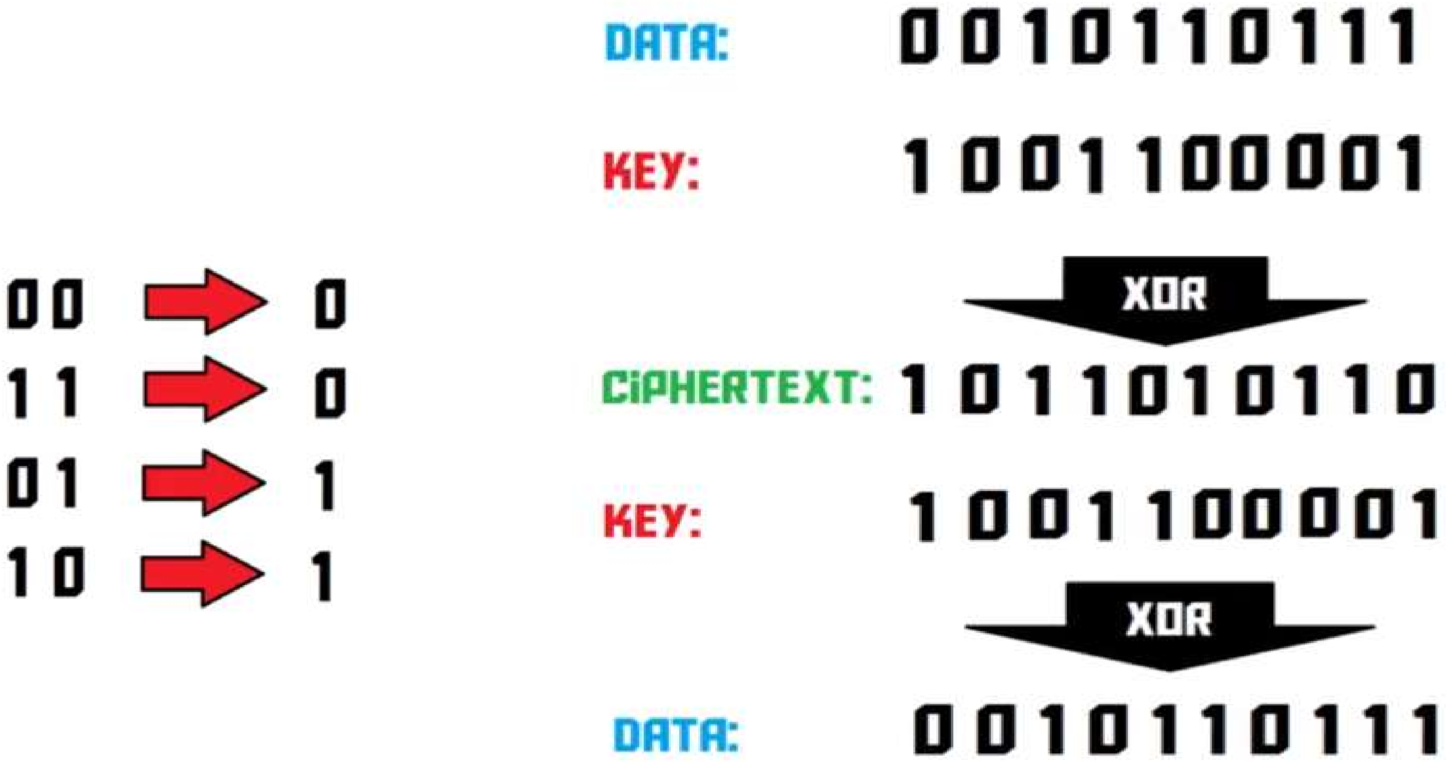
Stream and Block Ciphers

- Way in which plaintext is processed
  - Streamwise
  - Blockwise

Stream Ciphers

- Stream ciphers process messages a *bit or byte* at a time when en/decrypting
  - Vernam Cipher
- If the cryptographic keystream is random, cipher is unbreakable.
- Bit-stream generator is implemented as an algorithm, can be produced by both users.
- In this approach the bit-stream generator is a key-controlled algorithm and must produce a bit stream that is cryptographically strong.

Stream Ciphers



00 →	0
11 →	0
01 →	1
10 →	1

DATA: 0010110111

KEY: 1001100001

XOR

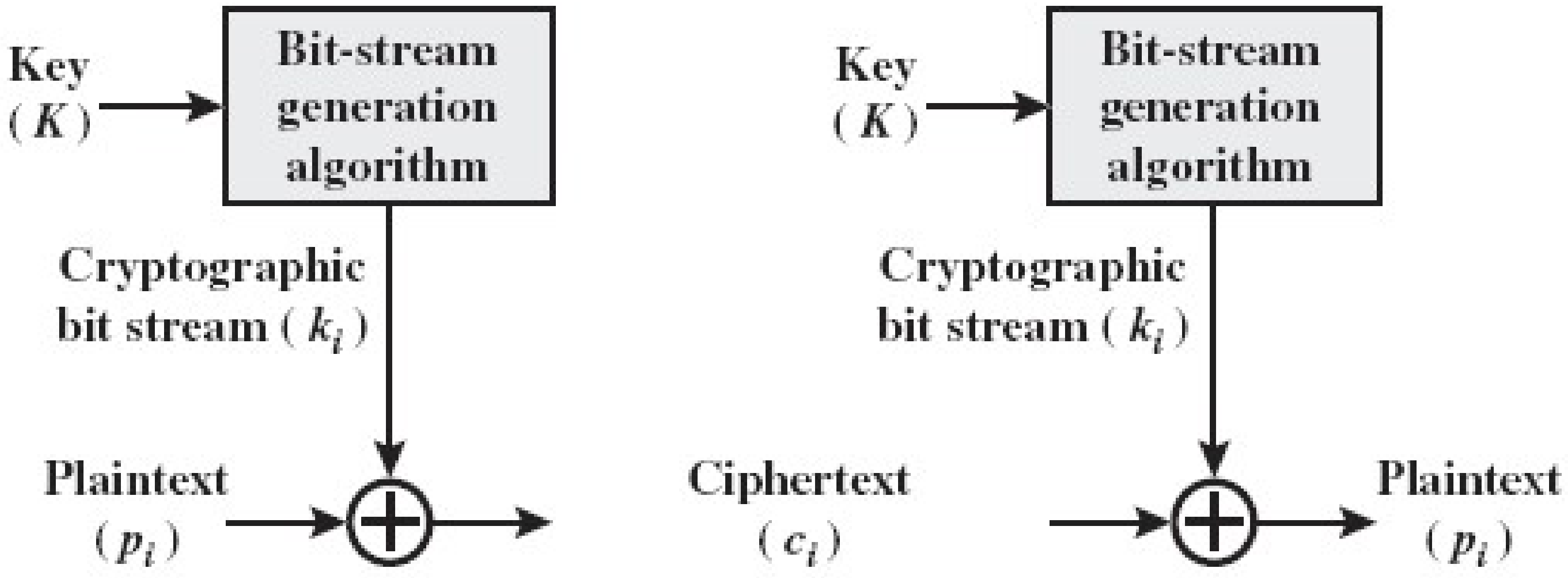
CIPHERTEXT: 1011010110

KEY: 1001100001

XOR

DATA: 0010110111

Stream Ciphers

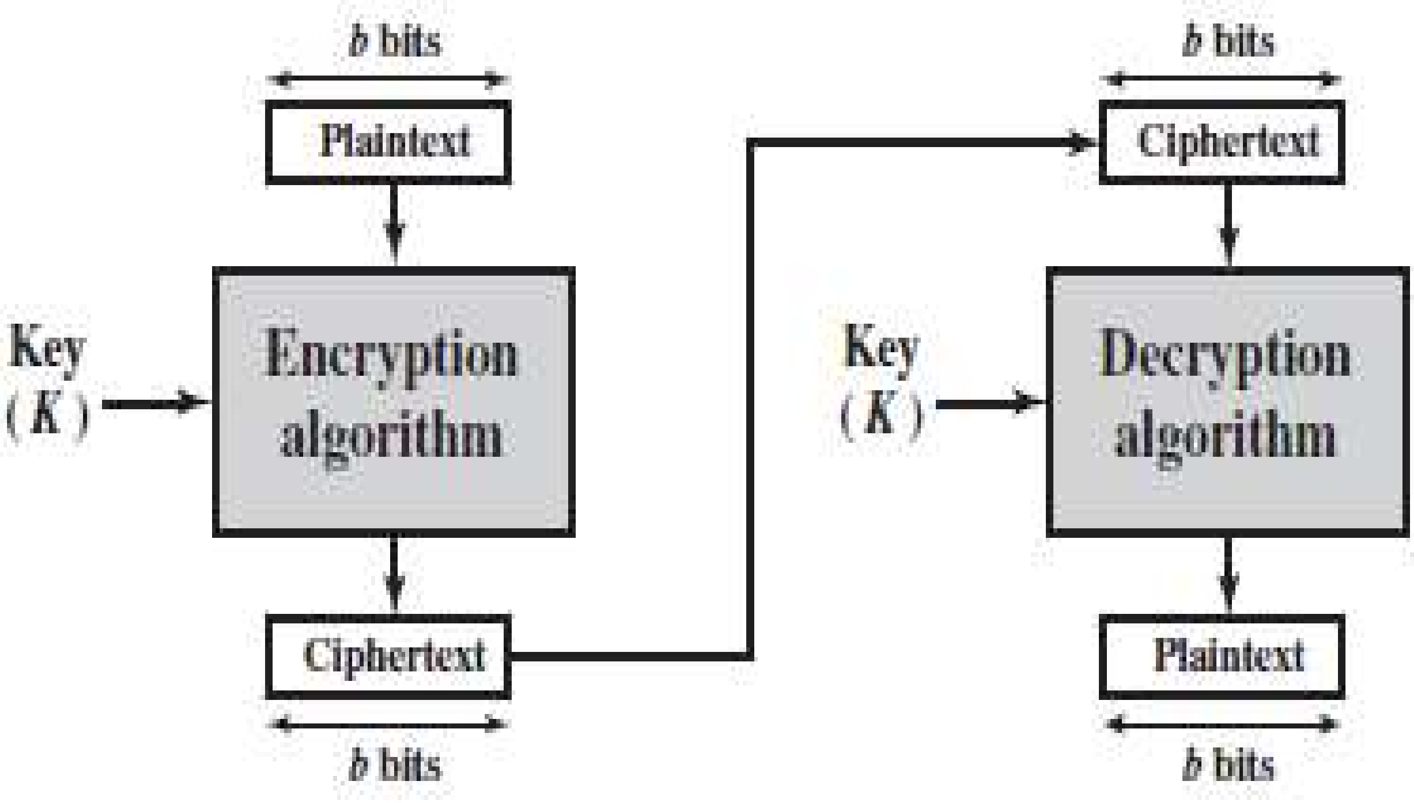

Key
(K)
Bit-stream generation algorithm
Cryptographic bit stream (k_i)
Plaintext
(p_i)
Ciphertext
(c_i)
Key
(K)
Bit-stream generation algorithm
Cryptographic bit stream (k_i)
Plaintext
(p_i)

(a) Stream cipher using algorithmic bit-stream generator

Block Ciphers

- Block ciphers process messages in blocks, each of which is then en/decrypted
- Like a substitution on very big characters
  - 64-bits or 128
- Many current ciphers are block ciphers.
- Broader range of applications.

Block Ciphers



b bits
Plaintext
Key
(*K*)
Encryption algorithm
Ciphertext
b bits
b bits
Ciphertext
Key
(*K*)
Decryption algorithm
Plaintext
b bits