Classical Encryption Techniques

CSS322: Security and Cryptography

Sirindhorn International Institute of Technology
Thammasat University

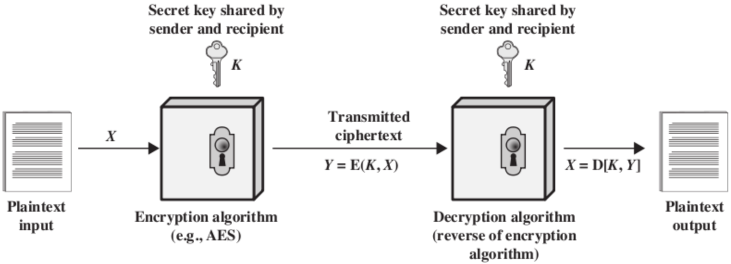
Prepared by Steven Gordon on 29 December 2011
CSS322Y11S2L02, Steve/Courses/2011/S2/CSS322/Lectures/classical.tex, r2070

Contents

Terminology

Plaintext	original message
Ciphertext	encrypted or coded message
Encryption	convert from plaintext to ciphertext (enciphering)
Decryption	restore the plaintext from ciphertext (deciphering)
Key	information used in cipher known only to sender/receiver
Cryptography	study of algorithms used for encryption
Cipher	a particular algorithm (cryptographic system)
Cryptanalysis	study of techniques for decryption without knowledge of plaintext
Cryptology	areas of cryptography and cryptanalysis

Simplified Model of Symmetric Encryption



Requirements and Assumptions

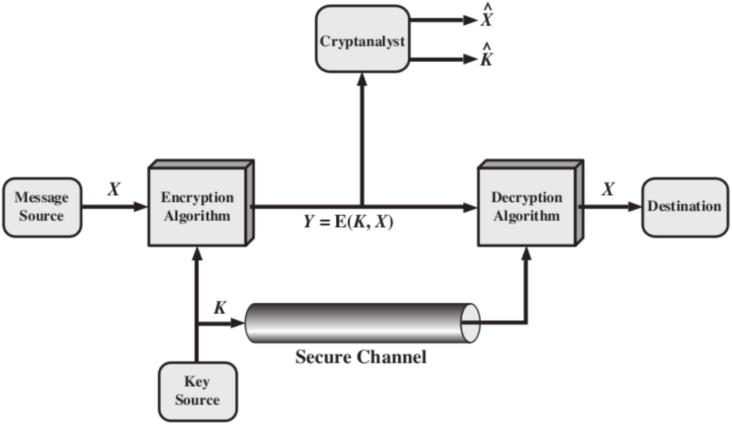
Requirements for secure use of symmetric encryption:

1. Strong encryption algorithm: Given the algorithm and ciphertext, an attacker cannot obtain key or plaintext
2. Sender/receiver know secret key (and keep it secret)

Assumptions:

- Cipher is known
- Secure channel to distribute keys

Model of Symmetric Cryptosystem



- Intended receiver can calculate: $X = \mathrm{D}(K, Y)$
- Attacker knows E, D and Y. Aim:
  - Determine plaintext: $\hat{X}$
  - Determine key: $\hat{K}$

Characterising Cryptographic Systems

Operations used for encryption:

Substitution replace one element in plaintext with another

Transposition re-arrange elements

Product systems multiple stages of substitutions and transpositions

Number of keys used:

Symmetric sender/receiver use same key (single-key, secret-key, shared-key, conventional)

Public-key sender/receiver use different keys (asymmetric)

Processing of plaintext:

Block cipher process one block of elements at a time

Stream cipher process input elements continuously

Cryptanalysis and Brute-Force Attacks

- Objective of attacker: recover key (not just message)
- Approaches of attacker:

  Cryptanalysis Exploit characteristics of algorithm to deduce plaintext or key

  Brute-force attack Try every possible key on ciphertext until intelligible translation into plaintext obtained

- If either attack finds key, all future/past messages are compromised

Cryptanalytic Attacks

Type of Attack	Known to Cryptanalyst
Ciphertext Only	• Encryption algorithm • Ciphertext
Known Plaintext	• Encryption algorithm • Ciphertext • One or more plaintext-ciphertext pairs formed with the secret key
Chosen Plaintext	• Encryption algorithm • Ciphertext • Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key
Chosen Ciphertext	• Encryption algorithm • Ciphertext • Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key
Chosen Text	• Encryption algorithm • Ciphertext • Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key • Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key

Measures of Security

Unconditionally Secure

- Ciphertext does not contained enough information to derive plaintext or key
- One-time pad is only unconditionally secure cipher (but not very practical)

Computationally Secure

- If either:
  - Cost of breaking cipher exceeds value of encrypted information
  - Time required to break cipher exceeds useful lifetime of encrypted information
- Hard to estimate value/lifetime of some information
- Hard to estimate how much effort needed to break cipher

Brute-Force Attacks

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

On average, number of guesses is half the key space

Contents

Classical Substitution Ciphers

- Letters of plaintext are replaced by others letters or by numbers of symbols
- If plaintext viewed as sequence of bits, replace plaintext bit patterns with ciphertext bit patterns

Caesar Cipher

- Earliest known cipher, used by Julius Caesar (Roman general 2000 years ago)
- Replace each letter by the letter three positions along in alphabet

```
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
```

Generalised Caesar Cipher

- Allow shift by k positions
- Assume each letter assigned number ($a = 0$, $b = 1$, ...)

$$C = \mathrm{E}(k, p) = (p + k) \bmod 26$$

$$p = \mathrm{D}(k, C) = (C - k) \bmod 26$$

Breaking the Caesar Cipher

- Brute force attack
  - Try all 25 keys, e.g. $k = 1$, $k = 2$, ...
  - Plaintext should be recognised
- Recognising plaintext in brute force attacks
  - Need to know "structure" of plaintext
  - Language? Compression?
- How to improve against brute force?
  - Hide the encryption/decryption algorithm: Not practical
  - Compress, use different language: Limited options
  - Increase the number of keys

Monoalphabetic (Substitution) Ciphers

- Monoalphabetic: use a single alphabet for both plaintext and ciphertext
- Arbitrary substitution: one element maps to any other element
  - n element alphabet allows $n!$ permutations or keys
- Example:

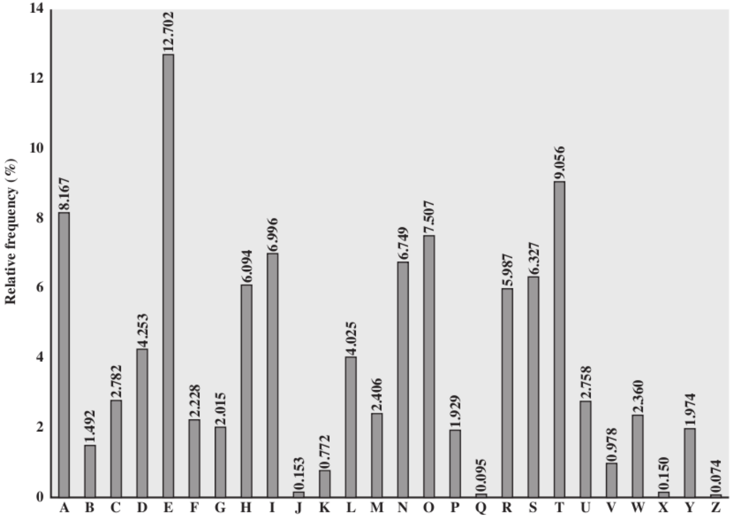
```
Plain :a b c d e ... w x y z
Cipher:D Z G L S ... B T F Q
```

- Try brute force ...
  - Caesar cipher: 26 keys
  - Monoalphabetic (English alphabet): 26! keys ($> 4 \times 10^{26}$)

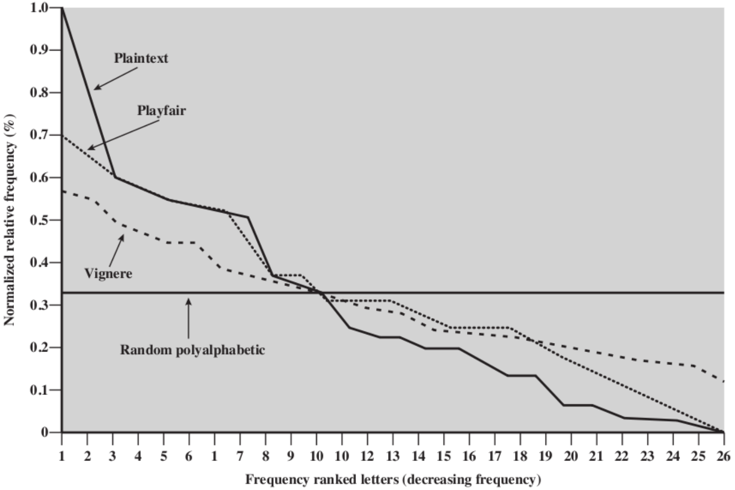
Attacks on Monoalphabetic Ciphers

- Exploit the regularities of the language
  - Frequency of letters, digrams, trigrams
  - Expected words
- Fundamental problem with monoalphabetic ciphers
  - Ciphertext reflects the frequency data of original plaintext
  - Solution 1: encrypt multiple letters of plaintext
  - Solution 2: use multiple cipher alphabets

Relative Frequency of Letters in English Text



Relative Frequency of Occurrence of Letters



Playfair Cipher

Initialisation

1. Create 5x5 matrix and write keyword (row by row)
2. Fill out remainder with alphabet, not repeating any letters
3. Special: Treat I and J as same letter

Encryption

1. Operate on pair of letters (digram) at a time
2. Special: if digram with same letters, separate by special letter (e.g. x)
3. Plaintext in same row: replace with letters to right
4. Plaintext in same column: replace with letters below
5. Else, replace by letter in same row as it and same column as other plaintext letter

Playfair Cipher Example

- Plaintext: `hello`
- Keyword: `thailand`
- Ciphertext: `LDAZEU`

Playfair Cipher - Is it Breakable?

- Better than monoalphabetic: relative frequency of digrams much less than of individual letters
- But relatively easy (digrams, trigrams, expected words)

Polyalphabetic Ciphers

- Use different monoalphabetic substitutions as proceed through plaintext
  - Set of monoalphabetic ciphers
  - Key determines which monoalphabetic cipher to use for each plaintext letter
- Examples:
  - Vigenère cipher
  - Vernam cipher (see textbook)
  - One time pad

Vigenère Cipher

- Set of 26 general Caesar ciphers
- Letter in key determines the Caesar cipher to use
  - Key must be as long as plaintext: repeat a keyword
- Example:

```
Plain:  internettechnologies
Key:    sirindhornsirindhorn
Cipher: AVKMEQLHKRUPEWYRNWVF
```

- Multiple ciphertext letters for each plaintext letter

Vigenère Cipher - Is it Breakable?

- Yes
- Monoalphabetic or Vigenère cipher? Letter frequency analysis
- Determine length of keyword
- For keyword length m, Vigenère is m monoalphabetic substitutions
- Break the monoalphabetic ciphers separately

Weakness is repeating, structured keyword

One Time Pad

- Similar to Vigenère, but use random key as long as plaintext
- Only known scheme that is unbreakable (unconditional security)
  - Ciphertext has no statistical relationship with plaintext
  - Given two potential plaintext messages, attacker cannot identify the correct message
- Two practical limitations:
  1. Difficult to provide large number of random keys
  2. Distributing unique long random keys is difficult
- Limited practical use

One Time Pad Example

Attacker knows the ciphertext:

```
ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS
```

Attacker tries all possible keys. Two examples:

```
key1:       pxlmvmsydofuyrvzwc tnlebnecvgdupahfzzlmnyih
plaintext1: mr mustard with the candlestick in the hall

key2:       mfugpmiydgaxgoufhklllmhsqdqogtewbqfgyovuhwt
plaintext2: miss scarlet with the knife in the library
```

There are many other legible plaintexts obtained with other keys. No way for attacker to know the correct plaintext

Contents

Rail Fence Transposition

- Plaintext letters written in diagonals over N rows (depth)
- Ciphertext obtained by reading row-by-row
- Easy to break: letter frequency analysis to determine depth
- Example:

```
plaintext: internettechnologiesandapplications
depth: 3
```

Rows/Columns Transposition

- Plaintext letters written in rows
- Ciphertext obtained by reading column-by-column, but re-arranged
- Key determines order of columns to read
- Easy to break using letter frequency (try different column orders)
- Example:

```
plaintext: securityandcryptography
key: 315624
```

Rows/Columns Transposition

Transposition ciphers can be made stronger by using multiple stages of transposition

```
plaintext:  attackpostponeduntiltwoamxyz
key: 4312567
ciphertext: TTNAAPTMTSUOAODWCOIXKNLYPETZ
```

Transpose again using same key:

```
output:     NSCYAUOPTTWLTMDNAOIEPAXTTOKZ
```

Original plaintext letters, by position:

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 first transposition:

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

After second transposition:

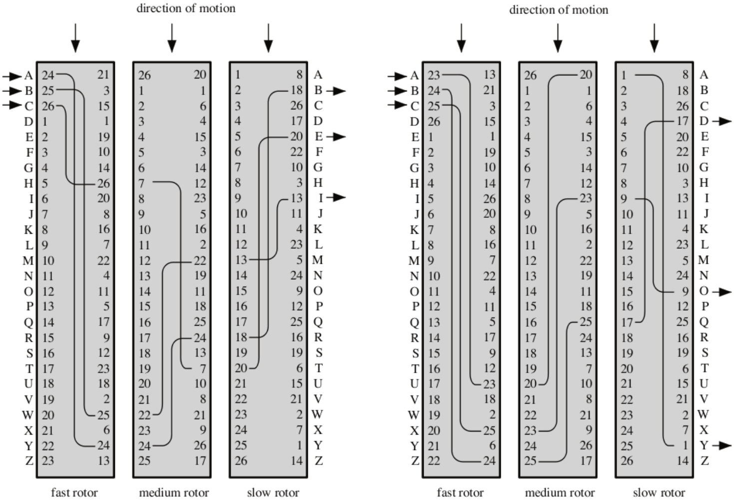
17	09	05	27	24	16	12	07	10	02	22	20	03	25
15	13	04	23	19	14	11	01	26	21	18	08	06	28

Contents

Rotor Machines

- Multiple stages of encryption can be used for substitution and transposition ciphers
- Rotor machines were early application of this
  - Principle was basis for Enigma cipher used by Germany in WW2
- Machine has multiple cylinders
  - Monoalphabetic substitution cipher for each cylinder
  - Output of one cylinder is input to next cylinder
  - Plaintext is input to first cylinder; ciphertext is output of last cylinder
  - Entering a plaintext letter causes last cylinder to rotate its cipher
  - Complete rotation of one cylinder causes previous cylinder to rotate its cipher
- Principle is used in Data Encryption Standard (DES)

Three-Rotor Machine



(a) Initial setting **(b) Setting after one keystroke**

Contents

Steganography

- Hide a real message in a fake, but meaningful, message
- Assumes recipient knows the method of hiding
- Examples:
  - Selected letters in a document are marked to form the hidden message
  - Invisible ink (letters only become visible when exposed to a chemical or heat)
  - Using selected bits in images or videos to carry the message
- Advantages
  - Does not *look like* you are hiding anything
- Disadvantages
  - Once attacker knows your method, everything is lost
  - Can be inefficient (need to send lot of information to carry small message)

Steganography Example

Dear George,
Greetings to all at Oxford. Many thanks for your letter and for the Summer examination package. All Entry Forms and Fee Forms should be ready for final despatch to the Syndicate by Friday 20th or at the very latest, I'm told, by the 21st. Admin has improved here, though there's room for improvement still; just give us all two or three more years and we'll really show you! Please don't let these wretched 16+ proposals destroy your basic O and A pattern. Certainly this sort of change, if implemented immediately, would bring chaos.
Sincerely yours.