Cryptography

10

Key Topics

Caesar Cipher
Enigma Codes
Bletchley Park
Turing
Public and Private Keys
Symmetric Keys
Block Ciphers
RSA

10.1 Introduction

Cryptography was originally employed to protect communication of private information between individuals. Today, it consists of mathematical techniques that provide secrecy in the transmission of messages between computers, and its objective is to solve security problems such as privacy and authentication over a communications channel.

It involves enciphering and deciphering messages, and it employs theoretical results from number theory to convert the original message (or plaintext) into cipher text that is then transmitted over a secure channel to the intended recipient. The cipher text is meaningless to anyone other than the intended recipient, and the recipient uses a key to decrypt the received cipher text and to read the original message.

G. O'Regan, *Guide to Discrete Mathematics*, Texts in Computer Science,
DOI 10.1007/978-3-319-44561-8_10

Alphabet Symbol	abcde fghij klmno pqrst uvwxyz
Cipher Symbol	dfegh ijklm nopqr stuvw xyzabc

Fig. 10.1 Caesar Cipher

The origin of the word "cryptography" is from the Greek "kryptos" meaning hidden, and "graphein" meaning to write. The field of cryptography is concerned with techniques by which information may be concealed in cipher texts and made unintelligible to all but the intended recipient. This ensures the privacy of the information sent, as any information intercepted will be meaningless to anyone other than the recipient.

Julius Caesar developed one of the earliest ciphers on his military campaigns in Gaul. His objective was to communicate important messages safely to his generals. His solution is one of the simplest and widely known encryption techniques, and it involves the substitution of each letter in the plaintext (i.e., the original message) by a letter a fixed number of positions down in the alphabet. The Caesar cipher involves a shift of three positions and this leads to the letter B being replaced by E, the letter C by F, and so on.

The Caesar cipher (Fig. 10.1) is easily broken, as the frequency distribution of letters may be employed to determine the mapping. However, the Gaulish tribes who were mainly illiterate, and it is likely that the cipher provided good security. The translation of the Roman letters by the Caesar cipher (with a shift key of 3) can be seen by the following table.

The process of enciphering a message (i.e., the plaintext) simply involves going through each letter in the plaintext and writing down the corresponding cipher letter. The enciphering of the plaintext message "summer solstice" involves the following:

Plaintext:	Summer Solstice
Cipher Text	vxpphu vrovwleh

The process of deciphering a cipher message involves doing the reverse operation: i.e., for each cipher letter the corresponding plaintext letter is identified from the table.

Cipher Text	vxpphu vrovwleh
Plaintext:	Summer Solstice

The encryption may also be represented using modular arithmetic. This involves using the numbers 0–25 to represent the alphabet letters, and the encryption of a letter is given by a shift transformation of three (modulo 26). This is simply addition (modula 26): i.e., the encoding of the plaintext letter x is given by

$$x + 3 (\text{mod } 26) = a$$

Similarly, the decoding of the cipher letter a is given by

$$a - 3 (\text{mod } 26) = x$$

The Caesar cipher was still in use up to the early twentieth century. However, by then frequency analysis techniques were available to break the cipher. The Vignère cipher uses a Caesar cipher with a different shift at each position in the text. The value of the shift to be employed with each plaintext letter is defined using a repeating keyword.

10.2 Breaking the Enigma Codes

The Enigma codes were used by the Germans during the second world war for the secure transmission of naval messages to their submarines. These messages contained top-secret information on German submarine and naval activities in the Atlantic, and the threat that they posed to British and Allied shipping.

The codes allowed messages to be passed secretly using encryption, and this meant that any unauthorized interception was meaningless to the Allies. The plaintext (i.e., the original message) was converted by the Enigma machine (Fig. 10.2) into the encrypted text, and these messages were then transmitted by the German military to their submarines in the Atlantic, or to their bases throughout Europe.

The Enigma cipher was invented in 1918 and the Germans believed it to be unbreakable. A letter was typed in German into the machine, and electrical impulses through a series of rotating wheels and wires produced the encrypted letter which was lit up on a panel above the keyboard. The recipient typed the received message into his machine and the decrypted message was lit up letter by letter above the keyboard. The rotors and wires of the machine could be configured in

Fig. 10.2 The Enigma machine

Fig. 10.3 Bletchley park

Fig. 10.4 Alan Turing

many different ways, and during the war the cipher settings were changed at least once a day. The odds against anyone breaking the Enigma machine without knowing the setting were 150×10^{18} to 1.

The British code and cipher school relocated from London to Bletchley Park (Fig. 10.3) at the start of the second world war. It was located in the town of Bletchley near Milton Keynes (about 50 miles North West of London). It was commanded by Alistair Dennison and was known as Station X, and several thousands were working there during the second world war. The team at Bletchley Park broke the Enigma codes, and therefore made vital contributions to the British and Allied war effort.

Polish cryptanalysts did important work in breaking the Enigma machine in the early 1930s, and they constructed a replica of the machine. They passed their knowledge on to the British and gave them the replica just prior to the German invasion of Poland. The team at Bletchley built upon the Polish work, and the team included Alan Turing[1] (Fig. 10.4) and other mathematicians.

The code-breaking teams worked in various huts in Bletchley park. Hut 6 focused on air force and army ciphers, and hut 8 focused on naval ciphers. The deciphered messages were then converted into intelligence reports, with air force

[1]Turing made fundamental contributions to computing, including the theoretical Turing machine.

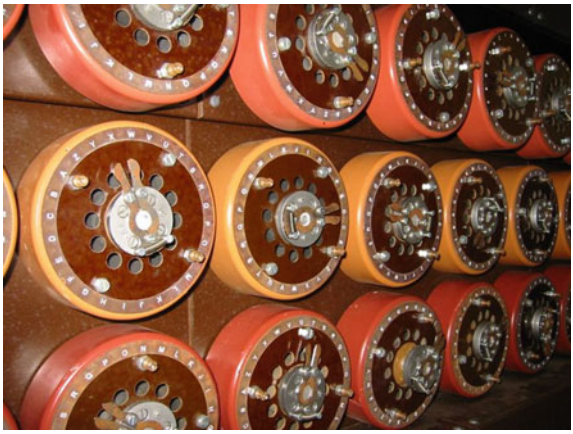

Fig. 10.5 Replica of bombe

and army intelligence reports produced by the team in hut 3, and naval intelligence reports produced by the team in hut 4. The raw material (i.e., the encrypted messages) to be deciphered came from wireless intercept stations dotted around Britain, and from various countries overseas. These stations listened to German radio messages, and sent them to Bletchley park to be deciphered and analyzed.

Turing devised a machine to assist with breaking the codes (an idea that was originally proposed by the Polish cryptanalysts). This electromechanical machine was known as the bombe (Fig. 10.5), and its goal was to find the right settings of the Enigma machine for that particular day. The machine greatly reduced the odds and the time required to determine the settings on the Enigma machine, and it became the main tool for reading the Enigma traffic during the war. The bombe was first installed in early 1940 and it weighed over a ton. It was named after a cryptological device designed in 1938 by the Polish cryptologist, Marian Rejewski.

A standard Enigma machine employed a set of rotors, and each rotor could be in any of 26 positions. The bombe tried each possible rotor position and applied a test. The test eliminated almost all of the positions and left a smaller number of cases to be dealt with. The test required the cryptologist to have a suitable "crib": i.e., a section of ciphertext for which he could guess the corresponding plaintext.

For each possible setting of the rotors, the bombe employed the crib to perform a chain of logical deductions. The bombe detected when a contradiction had occurred and it then ruled out that setting and moved onto the next. Most of the possible settings would lead to contradictions and could then be discarded. This would leave only a few settings to be investigated in detail.

The Government Communication Headquarters (GCHQ) was the successor of Bletchley Park, and it opened after the war. The site at Bletchley park was then used for training purposes.

The codebreakers who worked at Bletchley Park were required to remain silent about their achievements until the mid-1970s when the wartime information was declassified. The link between British Intelligence and Bletchley Park came to an end in the mid-1980s.

It was decided in the mid-1990s to restore Bletchley Park, and today it is run as a museum by the Bletchley Park Trust.

10.3 Cryptographic Systems

A cryptographic system is a computer system that is concerned with the secure transmission of messages. The message is encrypted prior to its transmission, which ensures that any unauthorized interception and viewing of the message is meaningless to anyone other than the intended recipient. The recipient uses a key to decrypt the cipher text, and to retrieve the original message.

There are essentially two different types of cryptographic systems employed, and these are public key cryptosystems and secret key cryptosystems. A *public key cryptosystem* is an asymmetric cryptosystem where two different keys are employed: one for encryption and one for decryption. The fact that a person is able to encrypt a message does not mean that the person is able to decrypt a message.

In a *secret key cryptosystem* the same key is used for both encryption and decryption. Anyone who has knowledge on how to encrypt messages has sufficient knowledge to decrypt messages. The following notation is employed (Table 10.1).

The encryption and decryption algorithms satisfy the following equation:

$$\mathrm{D}d_k(\mathrm{C}) = \mathrm{D}d_k(\mathrm{E}e_k(\mathrm{M})) = \mathrm{M}$$

There are two different keys employed in a public key cryptosystem. These are the encryption key e_k and the decryption key d_k with $e_k. \neq d_k.$ It is called asymmetric since the encryption key differs from the decryption key.

There is just one key employed in a secret key cryptosystem, with the same key $e_k.$ is used for both encryption and decryption. It is called *symmetric* since the encryption key is the same as the decryption key: i.e., $e_k. = d_k.$

Table 10.1 Notation in cryptography

Symbol	Description
M	Represents the message (plaintext)
C	Represents the encrypted message (cipher text)
e_k	Represents the encryption key
d_k	Represents the decryption key
E	Represents the encryption process
D	Represents the decryption process

10.4 Symmetric Key Systems

A symmetric key cryptosystem (Fig. 10.6) uses the same secret key for encryption and decryption. The sender and the receiver first need to agree a shared key prior to communication. This needs to be done over a secure channel to ensure that the shared key remains secret. Once this has been done they can begin to encrypt and decrypt messages using the secret key. Anyone who is able to encrypt a message has sufficient information to decrypt the message.

The encryption of a message is in effect a transformation from the space of messages $\mathcal{M}$ to the space of cryptosystems $\mathbb{C}$. That is, the encryption of a message with key k is an invertible transformation f such that:

$$f : \mathcal{M} \overset{k}{\rightarrow} \mathbb{C}$$

The cipher text is given by $C = E_k(M)$ where $M \in \mathcal{M}$ and $C \in \mathbb{C}$. The legitimate receiver of the message knows the secret key k (as it will have transmitted previously over a secure channel), and so the cipher text C can be decrypted by the inverse transformation f^{-1} defined by:

$$f^{-1} : \mathbb{C} \overset{k}{\rightarrow} \mathcal{M}$$

Therefore, we have that $D_k(C) = D_k(E_k(M)) = M$ the original plaintext message.

There are advantages and disadvantages to symmetric key systems (Table 10.2), and these include

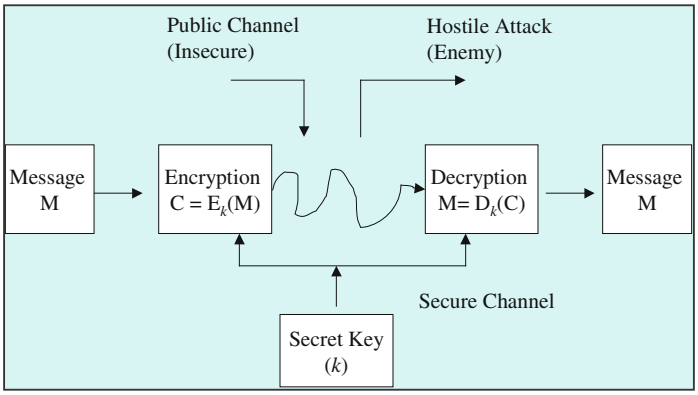

Fig. 10.6 Symmetric key cryptosystem

Table 10.2 Advantages and disadvantages of symmetric key systems

Advantages	Disadvantages
Encryption process is simple (as the same key is used for encryption and decryption)	A shared key must be agreed between two parties
It is faster than public key systems	Key exchange is difficult as there needs to be a secure channel between the two parties (to ensure that the key remains secret)
It uses less computer resources than public key systems	If a user has n trading partners then n secret keys must be maintained (one for each partner)
It uses a different key for communication with every different party	There are problems with the management and security of all of these keys (due to volume of keys that need to be maintained)
	Authenticity of origin or receipt cannot be proved (as key is shared)

Examples of Symmetric Key Systems

(i) *Caesar Cipher*

The Caesar cipher may be defined using modular arithmetic. It involves a shift of three places for each letter in the plaintext, and the alphabetic letters are represented by the numbers 0–25. The encryption is carried out by addition (modula 26). The encryption of a plaintext letter x to a cipher letter c is given by[2]:

$$c = x + 3 (\text{mod } 26)$$

Similarly, the decryption of a cipher letter c is given by:

$$x = c - 3 (\text{mod } 26)$$

(ii) *Generalized Caesar Cipher*

This is a generalization of the Caesar cipher to a shift of k (the Caesar cipher involves a shift of three). This is given by

$$f_k = \mathrm{E}_k(x) \equiv x + k (\text{mod } 26) \qquad 0 \leq k \leq 25$$
$$f_k^{-1} = \mathrm{D}_k(c) \equiv c - k (\text{mod } 26) \qquad 0 \leq k \leq 25$$

[2]Here x and c are variables rather than the alphabetic characters 'x' and 'c'.

(iii) *Affine Transformation*

This is a more general transformation and is defined by

$$\begin{aligned} f_{(a,b)} &= \mathrm{E}_{(a,b)}(x) \equiv ax + b \,(\mathrm{mod}\, 26) \qquad 0 \le a, b, x \le 25 \text{ and } \gcd(a, 26) = 1 \\ f^{-1}_{(a,b)} &= \mathrm{D}_{(a,b)}(c) \equiv a^{-1}(c-b) \,(\mathrm{mod}\, 26) \quad a^{-1} \text{ is the inverse of } a \bmod 26 \end{aligned}$$

(iv) *Block Ciphers*

Stream ciphers encrypt a single letter at a time and are easy to break. Block ciphers offer greater security, and the plaintext is split into groups of letters, and the encryption is performed on the block of letters rather than on a single letter.

The message is split into blocks of n-letters: M_1, M_2,…, M_k, where each M_i ($1 \le i \le k$) is a block n-letters. The letters in the message are translated into their numerical equivalents, and the cipher text formed as follows:

$$\mathrm{C}_i \equiv \mathrm{AM}_i + \mathrm{B} \,(\mathrm{mod}\, \mathrm{N}) \qquad i = 1, 2, \ldots k$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \ldots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \ldots & a_{3n} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a_{n1} & a_{n2} & a_{n3} & \ldots & a_{nn} \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ \ldots \\ \ldots \\ m_n \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \ldots \\ \ldots \\ b_n \end{pmatrix} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \ldots \\ \ldots \\ c_n \end{pmatrix},$$

where (A, B) is the key, A is an invertible $n \times n$ matrix with gcd(det(A), N) = 1,[3] $\mathrm{M}_i = (m_1, m_2, \ldots, m_n)^{\mathrm{T}}$, $\mathrm{B} = (b_1, b_2, \ldots, b_n)^{\mathrm{T}}$, $\mathrm{C}_i = (c_1, c_2, \ldots, c_n)^{\mathrm{T}}$. The decryption is performed by

$$\mathrm{M}_i \equiv \mathrm{A}^{-1}(\mathrm{C}_i - \mathrm{B})(\mathrm{mod}\, \mathrm{N}) \qquad i = 1, 2, \ldots, k$$

$$\begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ \ldots \\ \ldots \\ m_n \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \ldots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \ldots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \ldots & a_{3n} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a_{n1} & a_{n2} & a_{n3} & \ldots & a_{nn} \end{pmatrix}^{-1} \begin{pmatrix} c_1 - b_1 \\ c_2 - b_2 \\ c_3 - b_3 \\ \ldots \\ \ldots \\ c_n - b_n \end{pmatrix}$$

[3]This requirement is to ensure that the matrix A is invertible.

(v) *Exponential Ciphers*

Pohlig and Hellman [1] invented the exponential cipher in 1976. This cipher is less vulnerable to frequency analysis than block ciphers.

Let p be a prime number and let M be the numerical representation of the plaintext, with each letter of the plaintext replaced with its two-digit representation (00–25). That is, A = 00, B = 01, …, Z = 25.

M is divided into blocks M_i (these are equal size blocks of m letters where the block size is approximately the same number of digits as p). The number of letters m per block is chosen such that

$$\underbrace{2525\ldots25}_{m \text{ times}} < p < \underbrace{2525\ldots25}_{m+1 \text{ times}}$$

For example, for the prime 8191 a block size of $m = 2$ letters (4 digits) is chosen since:

$$2525 < 8191 < 252525$$

The secret encryption key is chosen to be an integer k such that $0 < k < p$ and $\gcd(k, p - 1) = 1$. Then the encryption of the block M_i is defined by

$$C_i = E_k(\mathrm{M}_i) \equiv \mathrm{M}_i^k (\mathrm{mod}\, p)$$

The cipher text C_i is an integer such that $0 \leq \mathrm{C}_i < p$.

The decryption of C_i involves first determining the inverse k^{-1} of the key k (mod $p - 1$), i.e., we determine k^{-1} such that $kk^{-1} \equiv 1 \pmod{p-1}$. The secret key k was chosen so that $(k, p - 1) = 1$, and this means that there are integers d and n such that $kd = 1 + n(p - 1)$, and so k^{-1} is d and $kk^{-1} = 1 + n(p - 1)$. Therefore,

$$D_{k^{-1}}(C_i) \equiv \mathrm{C}_i^{k^{-1}} \equiv (\mathrm{M}_i^k)^{k^{-1}} \equiv \mathrm{M}_i^{1+n(p-1)} \equiv \mathrm{M}_i (\mathrm{mod}\, p)$$

The fact that $\mathrm{M}_i^{1+n(p-1)} \equiv \mathrm{M}_i \pmod{p}$ follows from Euler's Theorem and Fermat's Little Theorem (Theorems 3.7 and 3.8), which were discussed in Chap. 3. Euler's Theorem states that for two positive integers a and n with $\gcd(a, n) = 1$ that $a^{\phi(n)} \equiv 1 \pmod{n}$.

Clearly, for a prime p we have that $\phi(p) = p - 1$. This allows us to deduce that

$$\mathrm{M}_i^{1+n(p-1)} \equiv \mathrm{M}_i^1 \mathrm{M}_i^{n(p-1)} \equiv \mathrm{M}_i \left(\mathrm{M}_i^{(p-1)}\right)^n \equiv \mathrm{M}_i (1)^n \equiv \mathrm{M}_i (\mathrm{mod}\, p)$$

(vi) *Data Encryption Standard (DES)*

DES is a popular cryptographic system [2] used by governments and private companies around the world. It is based on a symmetric key algorithm and uses a shared secret key that is known only to the sender and receiver. It was designed by IBM and approved by the National Bureau of Standards (NBS[4]) in 1976. It is a block cipher and a message is split into 64-bit message blocks. The algorithm is employed in reverse to decrypt each cipher text block.

Today, DES is considered to be insecure for many applications as its key size (56 bits) is viewed as being too small, and the cipher has been broken in less than 24 h. This has led to it being withdrawn as a standard and replaced by the Advanced Encryption Standard (AES), which uses a larger key of 128 bits or 256 bits.

The DES algorithm uses the same secret 56-bit key for encryption and decryption. The key consists of 56 bits taken from a 64-bit key that includes 8 parity bits. The parity bits are at position 8, 16, …, 64, and so every eighth bit of the 64-bit key is discarded leaving behind only the 56-bit key.

The algorithm is then applied to each 64-bit message block and the plaintext message block is converted into a 64-bit cipher text block. An initial permutation is first applied to M to create M′, and M′ is divided into a 32-bit left half L_0 and a 32-bit right half R_0. There are then 16 iterations, with the iterations having a left half and a right half:

$$\begin{aligned} L_i &= R_{i-1} \\ R_i &= L_{i-1} \oplus f(R_{i-1}, K_i) \end{aligned}$$

The function f is a function that takes a 32-bit right half and a 48-bit round key K_i (each K_i contains a different subset of the 56-bit key) and produces a 32-bit output. Finally, the pre-cipher text (R_{16}, L_{16}) is permuted to yield the final cipher text C. The function f operates on half a message block and involves Table 10.3.

The decryption of the cipher text is similar to the encryption and it involves running the algorithm in reverse.

DES has been implemented on a microchip. However, it has been superseded in recent years by AES due to security concerns with its small 56-bit key size. The AES uses a key size of 128 bits or 256 bits.

10.5 Public Key Systems

A public key cryptosystem (Fig. 10.7) is an asymmetric key system where there is a separate key e_k for encryption and d_k decryption with $e_k. \neq d_k$. Martin Hellman and Whitfield Diffie invented it in 1976. The fact that a person is able to encrypt a

[4]The NBS is now known as the National Institute of Standards and Technology (NIST).

Table 10.3 DES Encryption

Step	Description
1	Expansion of the 32-bit half block to 48 bits (by duplicating half of the bits)
2	The 48-bit result is combined with a 48-bit subkey of the secret key using an XOR operation
3	The 48-bit result is broken into 8 * 6 bits and passed through 8 substitution boxes to yield 8 * 4 = 32 bits (This is the core part of the encryption algorithm)
4	The 32-bit output is rearranged according to a fixed permutation

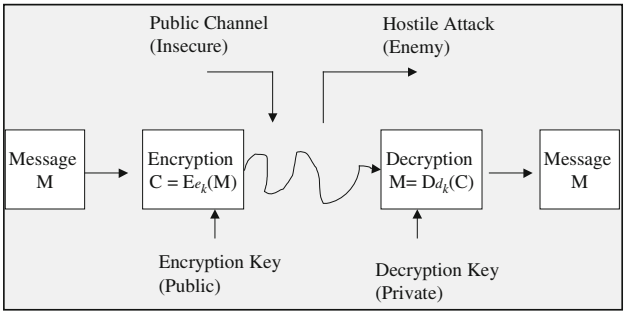

Fig. 10.7 Public key cryptosystem

Table 10.4 Public key encryption system

Item	Description
1	It uses the concept of a key pair (e_k, d_k)
2	One half of the pair can encrypt messages and the other half can decrypt messages
3	One key is private and one key is public
4	The private key is kept secret and the public key is published (but associated with trading partner)
5	The key pair is associated with exactly one trading partner

message does not mean that the person has sufficient information to decrypt messages.

The public key cryptosystem is based on the Table 10.4:

The advantages and disadvantages of public key cryptosystems Table 10.5:

The implementation of public key cryptosystems is based on *trapdoor one-way functions*. A function $f : X \rightarrow Y$ is a trapdoor one-way function if

- f is easy to computer
- f^{-1} is difficult to compute
- f^{-1} is easy to compute if a trapdoor (secret information associated with the function) becomes available.

Table 10.5 Advantages and disadvantages of public key cryptosystems

Advantages	Disadvantages
Only the private key needs to be kept secret	Public keys must be authenticated
The distribution of keys for encryption is convenient as everyone publishes their public key and the private key is kept private	It is slow and uses more computer resources
It provides message authentication as it allows the use of digital signatures (which enables the recipient to verify that the message is really from the particular sender)	Security Compromise is possible (if private key compromised)
The sender encodes with the private key that is known only to sender. The receiver decodes with the public key and therefore knows that the message is from the sender	Loss of private key may be irreparable (unable to decrypt messages)
Detection of tampering (digital signatures enable the receiver to detect whether message was altered in transit)	
Provides for nonrepudiation	

A function satisfying just the first two conditions above is termed a *one-way function*.

Examples of Trapdoor and One-way Functions

(i) The function $f : pq \rightarrow n$ (where p and q are primes) is a one-way function since it is easy to compute. However, the inverse function f^{-1} is difficult to compute problem for large n since there is no efficient algorithm to factorize a large integer into its prime factors (*integer factorization problem*).

(ii) The function $f_{g,\ N} : x \rightarrow g^x \pmod{N}$ is a one-way function since it is easy to compute. However, the inverse function f^{-1} is difficult to compute as there is no efficient method to determine x from the knowledge of $g^x \pmod{N}$ and g and N (*the discrete logarithm problem*).

(iii) The function $f_{k,\ N} : x \rightarrow x^k \pmod{N}$ (where N = pq and p and q are primes) and $kk' \equiv 1 \pmod{\varphi(n)}$ is a trapdoor function. It is easy to compute but the inverse of f (the kth root modulo N) is difficult to compute. However, if the trapdoor k' is given then f can easily be inverted as $(x^k)^{k'} \equiv x \pmod{N}$

10.5.1 RSA Public Key Cryptosystem

Rivest, Shamir and Adleman proposed a practical public key cryptosystem (RSA) based on primality testing and integer factorization in the late 1970s. The RSA algorithm was filed as a patent (Patent No. 4,405, 829) at the U.S. Patent Office in December 1977. The RSA public key cryptosystem is based on the following assumptions:

- It is straightforward to find two large prime numbers.
- The integer factorization problem is infeasible for large numbers

The algorithm is based on mod n arithmetic, where n is a product of two large prime numbers.

The encryption of a plaintext message M to produce the cipher text C is given by

$$C \equiv M^e (\text{mod } n),$$

where e is the public encryption key, M is the plaintext, **C** is the cipher text, and n is the product of two large primes p and q. Both e and n are made public, and e is chosen such that $1 < e < \phi(n)$, where $\phi(n)$ is the number of positive integers that are relatively prime to n.

The cipher text C is decrypted by

$$M \equiv C^d (\text{mod } n),$$

where d is the private decryption key that is known only to the receiver, and $ed \equiv 1 \pmod{\phi(n)}$ and d and $\phi(n)$ are kept private.

The calculation of $\phi(n)$ is easy if both p and q are known, as it is given by $\phi(n) = (p - 1)(q - 1)$. However, its calculation for large n is infeasible if p and q are unknown.

$$\begin{aligned} ed &\equiv 1 \ (\text{mod } \phi(n)) \\ \Rightarrow ed &= 1 + k\phi(n) \text{for some } k \in \mathbb{Z} \end{aligned}$$

We discussed Euler' Theorem in Chap. 3, and this result states that if a and n are positive integers with gcd(a, n) = 1 then $a^{\phi(n)} \equiv 1 \pmod{n}$. Therefore, $\mathrm{M}^{\phi(n)} \equiv 1 \pmod{n}$ and $\mathrm{M}^{k\phi(n)} \equiv 1 \pmod{n}$. The decryption of the cipher text is given by:

$$\begin{aligned} C^d (\text{mod } n) &\equiv \mathrm{M}^{ed} (\text{mod } n) \\ &\equiv \mathrm{M}^{1+k\phi(n)} (\text{mod } n) \\ &\equiv \mathrm{M}^1 \mathrm{M}^{k\phi(n)} (\text{mod } n) \\ &\equiv \mathrm{M}.1 (\text{mod } n) \\ &\equiv \mathrm{M} (\text{mod } n) \end{aligned}$$

10.5.2 Digital Signatures

The RSA public key cryptography may also be employed to obtain digital signatures. Suppose A wishes to send a secure message to B as well as a digital signature. This involves signature generation using the private key, and signature verification using the public key. The steps involved are: (Table 10.6):

Table 10.6 Steps for A to send secure message and signature to B

Step	Description
1	A uses B's public key to encrypt the message
2	A uses its private key to encrypt its signature
3	A sends the message and signature to B
4	B uses A's public key to decrypt A's signature
5	B uses its private key to decrypt A's message

The National Institute of Standards and Technology (NIST) proposed an algorithm for digital signatures in 1991. The algorithm is known as the Digital Signature Algorithm (DSA) and later became the Digital Signature Standard (DSS).

10.6 Review Questions

1. Discuss the early ciphers developed by Julius Caesar and Augustus. How effective were they at that period in history, and what are their weaknesses today?
2. Describe how the team at Bletchley Park cracked the German Enigma codes.
3. Explain the differences between a public key cryptosystem and a private key cryptosystem.
4. What are the advantages and disadvantages of private (symmetric) key cryptosystems?
5. Describe the various types of symmetric key systems.
6. What are the advantages and disadvantages of public key cryptosystems?
7. Describe public key cryptosystems including the RSA public key cryptosystem.
8. Describe how digital signatures may be generated.

10.7 Summary

This chapter provided a brief introduction to cryptography, which is the study of mathematical techniques that provide secrecy in the transmission of messages between computers. It was originally employed to protect communication between individuals, and today it is employed to solve security problems such as privacy and authentication over a communications channel.

It involves enciphering and deciphering messages, and theoretical results from number theory are employed to convert the original messages (or plaintext) into cipher text that is then transmitted over a secure channel to the intended recipient. The cipher text is meaningless to anyone other than the intended recipient, and the received cipher text is then decrypted to allow the recipient to read the message.

A public key cryptosystem is an asymmetric cryptosystem. It has two different encryption and decryption keys, and the fact that a person has knowledge on how to encrypt messages does not mean that the person has sufficient information to decrypt messages.

In a secret key cryptosystem the same key is used for both encryption and decryption. Anyone who has knowledge on how to encrypt messages has sufficient knowledge to decrypt messages, and it is essential that the key is kept secret between the two parties.

References

1. An Improved Algorithm for Computing Algorithms over GF(p) and its Cryptographic Significance. S. Pohlig and M. Hellman (1978). IEEE Transactions on Information Theory (24): 106–110.
2. Data Encryption Standard. FIPS-Pub 46. National Bureau of Standards. U.S. Department of Commerce. January 1977.