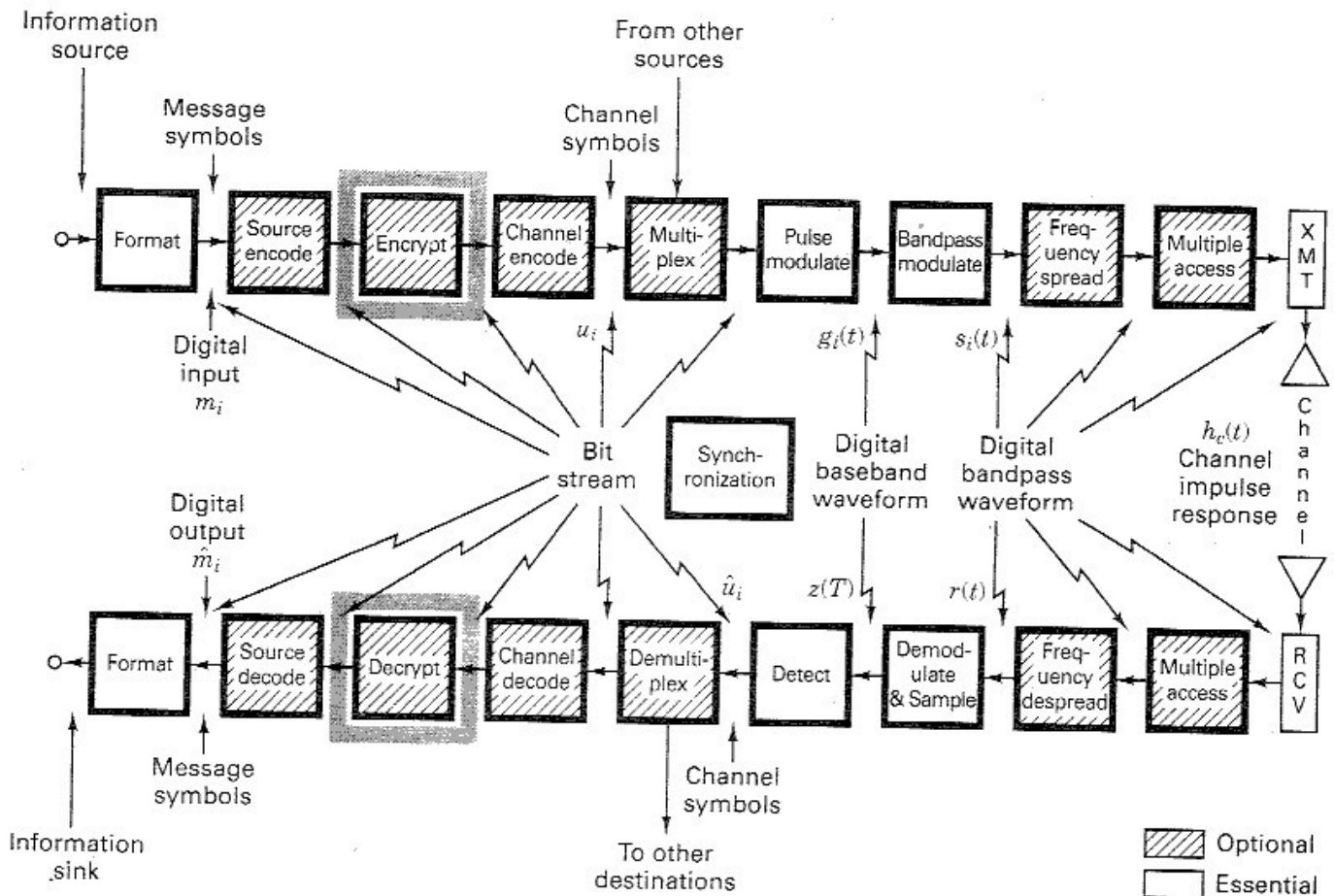


Encryption and Decryption



14.1 MODELS, GOALS, AND EARLY CIPHER SYSTEMS

14.1.1 A Model of the Encryption and Decryption Process

The desire to communicate privately is a human trait that dates back to earliest times. Hence the history of secret communications is rich with unique inventions and colorful anecdotes [1]. The study of ways to disguise messages so as to avert unauthorized interception is called *cryptography*. The terms *encipher* and *encrypt* refer to the message transformation performed at the transmitter, and the terms *decipher* and *decrypt* refer to the inverse transformation performed at the receiver. The two primary reasons for using cryptosystems in communications are (1) *privacy*, to prevent unauthorized persons from extracting information from the channel (eavesdropping); and (2) *authentication*, to prevent unauthorized persons from injecting information into the channel (spoofing). Sometimes, as in the case of electronic funds transfer or contract negotiations, it is important to provide the electronic equivalent of a *written signature* in order to avoid or settle any dispute between the sender and receiver as to what message, if any, was sent.

Figure 14.1 illustrates a model of a cryptographic channel. A message, or plaintext, M , is encrypted by the use of an invertible transformation, E_K , that produces a ciphertext, $C = E_K(M)$. The ciphertext is transmitted over an insecure or *public channel*. When an authorized receiver obtains C , he decrypts it with the inverse transformation, $D_K = E_K^{-1}$, to obtain the original plaintext message, as follows:

$$D_K(C) = E_K^{-1}[E_K(M)] = M \quad (14.1)$$

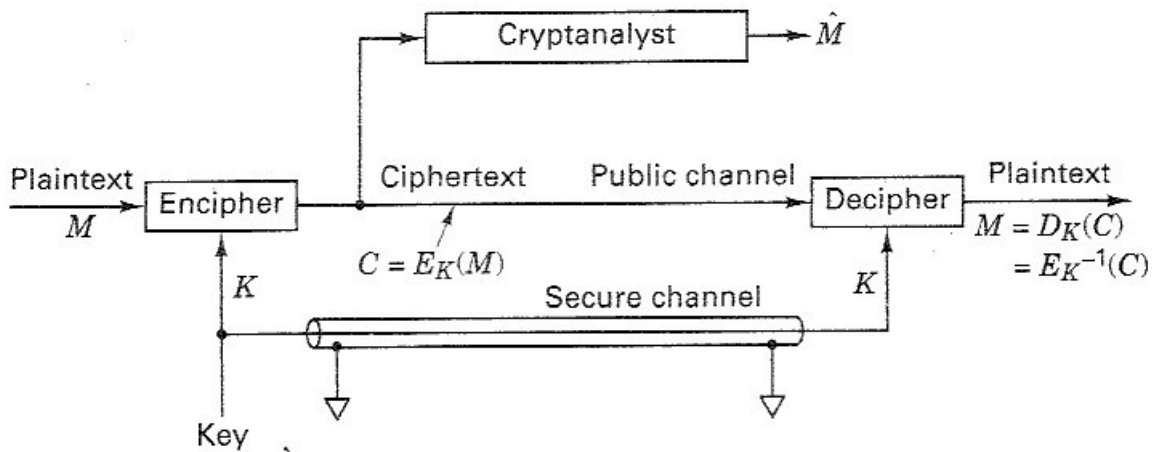


Figure 14.1 Model of a cryptographic channel.

The parameter K refers to a set of symbols or characters called a *key*, which dictates a specific encryption transformation, E_K , from a family of cryptographic transformations. Originally, the security of cryptosystems depended on the secrecy of the entire encryption process, but eventually systems were developed for which the general nature of the encryption transformation or algorithm could be publicly revealed, since the security of the system depended on the specific key. The key is supplied along with the plaintext message for encryption, and along with the ciphertext message for decryption. There is a close analogy here with a general-purpose computer and a computer program. The computer, like the cryptosystem, is capable of a large variety of transformations, from which the computer program, like the specific key, selects one. In most cryptosystems, anyone with access to the key can both encrypt and decrypt messages. The key is transmitted to the community of authorized users over a secure channel (as an example, a courier may be used to hand-carry the sensitive key information); the key usually remains unchanged for a considerable number of transmissions. The goal of the *cryptanalyst* (eavesdropper or adversary) is to produce an estimate of the plaintext, \hat{M} , by analyzing the ciphertext obtained from the public channel, without benefit of the key.

Encryption schemes fall into two generic categories: *block encryption*, and *data-stream* or simply *stream encryption*. With block encryption, the plaintext is segmented into blocks of fixed size; each block is encrypted independently from the others. For a given key, a particular plaintext block will therefore be carried into the same ciphertext block each time it appears (similar to block encoding). With data-stream encryption, similar to convolutional coding, there is no fixed block size. Each plaintext bit, m_i , is encrypted with the i th element, k_i , of a sequence of symbols (key stream) generated with the key. The encryption is *periodic* if the key stream repeats itself after p characters for some fixed p ; otherwise, it is nonperiodic.

In general, the properties desired in an encryption scheme are quite different from those desired in a channel coding scheme. For example, with encryption, plaintext data should never appear directly in the ciphertext, but with channel coding, codes are often in *systematic form* comprising unaltered message bits plus par-

ity bits (see Section 6.4.5). Consider another example of the differences between encryption and channel coding. With block encryption, a single bit error at the input of the decryptor might change the value of many of the output bits in the block. This effect, known as *error propagation*, is often a desirable cryptographic property since it makes it difficult for unauthorized users to succeed in spoofing a system. However, in the case of channel coding, we would like the system to correct as many errors as possible, so that the output is relatively unaffected by input errors.

14.1.2 System Goals

The major requirements for a cryptosystem can be stated as follows:

1. To provide an *easy* and *inexpensive* means of encryption and decryption to all authorized users in possession of the appropriate key
2. To ensure that the cryptanalyst's task of producing an estimate of the plaintext without benefit of the key is made *difficult* and *expensive*

Successful cryptosystems are classified as being either *unconditionally secure* or *computationally secure*. A system is said to be *unconditionally secure* when the amount of information available to the cryptanalyst is insufficient to determine the encryption and decryption transformations, no matter how much computing power the cryptanalyst has available. One such system, called a *one-time pad*, involves encrypting a message with a random key that is used one time only. The key is never reused; hence the cryptanalyst is denied information that might be useful against subsequent transmissions with the same key. Although such a system is unconditionally secure (see Section 14.2.1), it has limited use in a conventional communication system, since a new key would have to be distributed for each new message—a great logistical burden. The distribution of keys to the authorized users is a major problem in the operation of any cryptosystem, even when a key is used for an extended period of time. Although some systems can be proven to be unconditionally secure, currently there is no known way to demonstrate security for an arbitrary cryptosystem. Hence the specifications for most cryptosystems rely on the less formal designation of *computational security* for x number of years, which means that under circumstances favorable to the cryptanalyst (i.e., using state-of-the-art computers) the system security could be broken in a period of x years, but could not be broken in less than x years.

14.1.3 Classic Threats

The weakest classification of cryptanalytic threat on a system is called a *ciphertext-only attack*. In this attack the cryptanalyst might have *some* knowledge of the general system and the language used in the message, but the only significant data available to him is the encrypted transmission intercepted from the public channel.

A more serious threat to a system is called a *known plaintext attack*; it involves knowledge of the plaintext *and* knowledge of its ciphertext counterpart. The

rigid structure of most business forms and programming languages often provides an opponent with much a priori knowledge of the details of the plaintext message. Armed with such knowledge and with a ciphertext message, the cryptanalyst can mount a known plaintext attack. In the diplomatic arena, if an encrypted message directs a foreign minister to make a particular public statement, and if he does so without paraphrasing the message, the cryptanalyst may be privy to both the ciphertext *and* its exact plaintext translation. While a known plaintext attack is not always possible, its occurrence is frequent enough that a system is not considered secure unless it is designed to be secure against the plaintext attack [2].

When the cryptanalyst is in the position of *selecting* the plaintext, the threat is termed a *chosen plaintext attack*. Such an attack was used by the United States to learn more about the Japanese cryptosystem during World War II. On May 20, 1942, Admiral Yamamoto, Commander-in-Chief of the Imperial Japanese Navy, issued an order spelling out the detailed tactics to be used in the assault of Midway island. This order was intercepted by the Allied listening posts. By this time, the Americans had learned enough of the Japanese code to decrypt most of the message. Still in doubt, however, were some important parts, such as the *place* of the assault. They suspected that the characters “AF” meant Midway island, but to be sure, Joseph Rochefort, head of the Combat Intelligence Unit, decided to use a chosen plaintext attack to trick the Japanese into providing concrete proof. He had the Midway garrison broadcast a distinctive plaintext message in which Midway reported that its fresh-water distillation plant had broken down. The American cryptanalysts needed to wait only two days before they intercepted a Japanese ciphertext message stating that AF was short of fresh water [1].

14.1.4 Classic Ciphers

One of the earliest examples of a monoalphabetic cipher was the *Caesar Cipher*, used by Julius Caesar during the Gallic wars. Each plaintext letter is replaced with a new letter obtained by an *alphabetic shift*. Figure 14.2a illustrates such an encryption transformation, consisting of three end-around shifts of the alphabet. When using this Caesar’s alphabet, the message, “now is the time” is encrypted as follows:

Plaintext:	N	O	W	I	S	T	H	E	T	I	M	E
Ciphertext:	Q	R	Z	L	V	W	K	H	W	L	P	H

The decryption key is simply the number of alphabetic shifts; the code is changed by choosing a new key. Another classic cipher system, illustrated in Figure 14.2b, is called the *Polybius square*. Letters I and J are first combined and treated as a single character since the final choice can easily be decided from the context of the message. The resulting 25 character alphabet is arranged in a 5×5 array. Encryption of any character is accomplished by choosing the appropriate row-column (or column-row) number pair. An example of encryption with the use of the Polybius square follows:

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
 Chiphertext: 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

(a)

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I	J
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

(b)

Figure 14.2 (a) Caesar's alphabet with a shift of 3. (b) Polybius square.

Plaintext: N O W I S T H E T I M E
 Ciphertext: 33 43 25 42 34 44 32 51 44 42 23 51

The code is changed by a rearrangement of the letters in the 5 × 5 array.

The *Trithemius progressive key*, shown in Figure 14.3, is an example of a *polyalphabetic cipher*. The row labeled shift 0 is identical to the usual arrangement of the alphabet. The letters in the next row are shifted one character to the left with an end-around shift for the leftmost position. Each successive row follows the same pattern of shifting the alphabet one character to the left as compared to the prior row. This continues until the alphabet has been depicted in all possible arrangements of end-around shifts. One method of using such an alphabet is to select the first cipher character from the shift 1 row, the second cipher character from the shift 2 row, and so on. An example of such encryption is

Plaintext: N O W I S T H E T I M E
 Ciphertext: O Q Z M X Z O M C S X Q

There are several interesting ways that the Trithemius progressive key can be used. One way, called the *Vigenere key method*, employs a keyword. The key dictates the row choices for encryption and decryption of each successive character in the message. For example, suppose that the word "TYPE" is selected as the key; then an example of the Vigenere encryption method is

Key: T Y P E T Y P E T Y P E
 Plaintext: N O W I S T H E T I M E
 Ciphertext: G M L M L R W I M G B I

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	
Shift:	0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	R	U	V	W	X	Y	Z
	1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	O	R	S	T	U	V	W	X	Y	Z	A
	2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	O	R	S	T	U	V	W	X	Y	Z	A	B
	3	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
	4	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	D
	5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
	6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
	7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
	8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
	9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
	10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
	11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
	12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
	13	N	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
	14	O	P	Q	R	T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	15	P	Q	R	S	S	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
	18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
	20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
	21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
	23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
	24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
	25	Z	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

Figure 14.3 Trithemius progressive key.

where the first letter, T, of the key indicates that the row choice for encrypting the first plaintext character is the row starting with T (shift 19). The next row choice starts with Y (shift 24), and so on. A variation of this key method, called the *Vigenere auto (plain) key method*, starts with a single letter or word used as a *priming key*. The priming key dictates the starting row or rows for encrypting the first or first few plaintext characters, as in the preceding example. Next, the *plaintext characters* themselves are used as the key for choosing the rows for encryption. An example using the letter “F” as the priming key is

Key:	F	N	O	W	I	S	T	H	E	T	I	M
Plaintext:	N	O	W	I	S	T	H	E	T	I	M	E
Ciphertext:	S	B	K	E	A	L	A	L	X	B	U	Q

With the auto key method, it should be clear that feedback has been introduced to the encryption process. With this feedback, the choice of the ciphertext is dictated by the contents of the message.

A final variation of the Vigenere method, called the *Vigenere auto (cipher) key method*, is similar to the plain key method in that a priming key and feedback are used. The difference is that after encryption with the priming key, each successive key character is the sequence is obtained from the prior *ciphertext character* instead of from the plaintext character. An example should make this clear; as before, the letter "F" is used as the priming key:

Key:	F	S	G	C	K	C	V	C	G	Z	H	T
Plaintext:	N	O	W	I	S	T	H	E	T	I	M	E
Ciphertext:	S	G	C	K	C	V	C	G	Z	H	T	X

Although each key character can be found from its preceding ciphertext character, it is functionally dependent on *all* the preceding characters in the message plus the priming key. This has the effect of diffusing the statistical properties of the plaintext across the ciphertext, making statistical analysis very difficult for a cryptanalyst. One weakness of the cipher key example depicted here is that the ciphertext contains key characters which will be exposed on the public channel "for all to see." Variations of this method can be employed to prevent such overt exposure [3]. By today's standards Vigenere's encryption schemes are not very secure; his basic contribution was the discovery that nonrepeating key sequences could be generated by using the messages themselves or functions of the messages.

14.2 THE SECRECY OF A CIPHER SYSTEM

14.2.1 Perfect Secrecy

Consider a cipher system with a finite message space $\{M\} = M_0, M_1, \dots, M_{N-1}$ and a finite ciphertext space $\{C\} = C_0, C_1, \dots, C_{U-1}$. For any M_i , the a priori probability that M_i is transmitted is $P(M_i)$. Given that C_j is received, the a posteriori probability that M_i was transmitted is $P(M_i|C_j)$. A cipher system is said to have *perfect secrecy* if for every message M_i and every ciphertext C_j , the a posteriori probability is equal to the a priori probability:

$$P(M_i|C_j) = P(M_i) \quad (14.2)$$

Thus, for a system with perfect secrecy, a cryptanalyst who intercepts C_j obtains no further information to enable him or her to determine which message was transmitted. A necessary and sufficient condition for perfect secrecy is that for every M_i and C_j ,

$$P(C_j|M_i) = P(C_j) \quad (14.3)$$

The schematic in Figure 14.4 illustrates an example of perfect secrecy. In this example, $\{M\} = M_0, M_1, M_2, M_3$, $\{C\} = C_0, C_1, C_2, C_3$, $\{K\} = K_0, K_1, K_2, K_3$, $N = U = 4$,

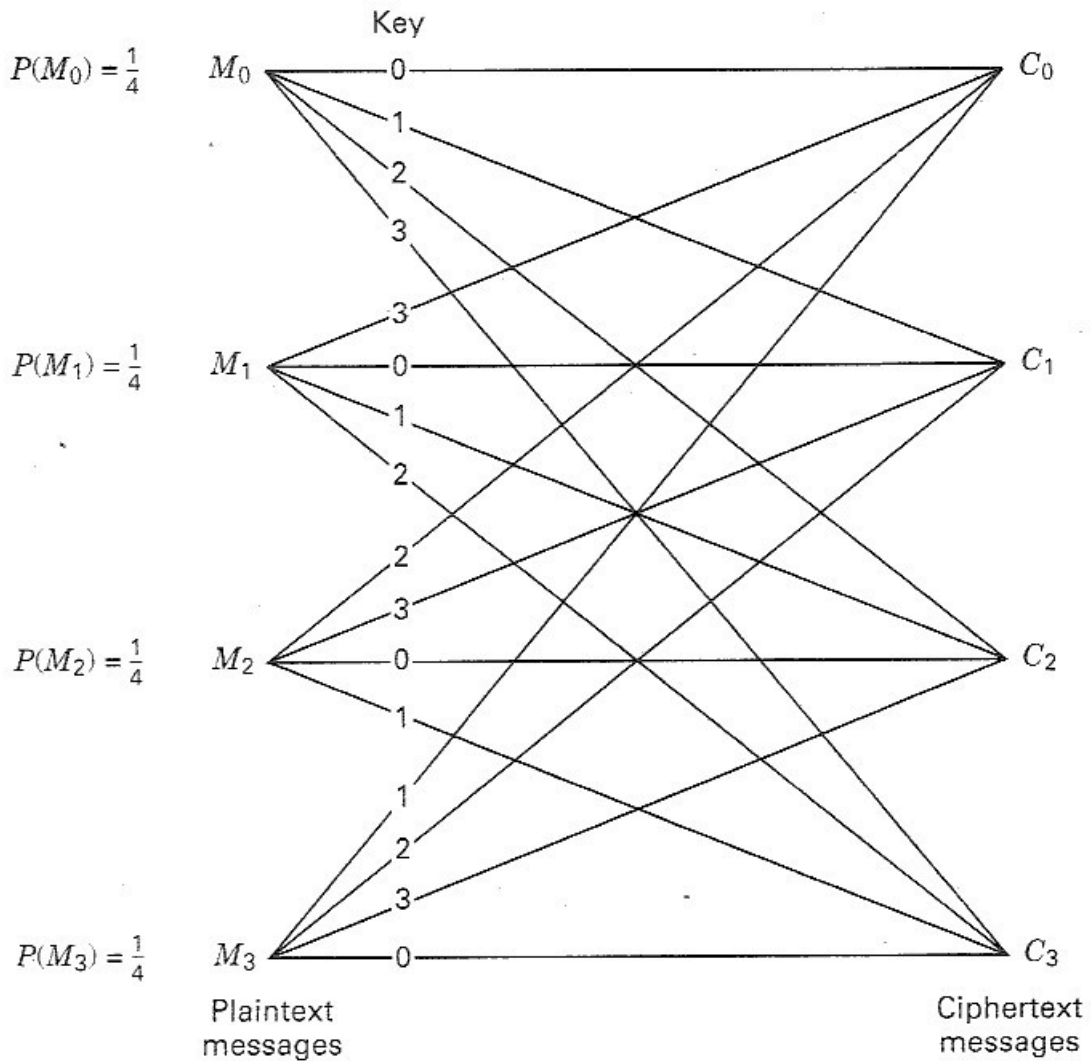


Figure 14.4 Example of perfect secrecy.

and $P(M_i) = P(C_j) = \frac{1}{4}$. The transformation from message to ciphertext is obtained by

$$C_s = T_{K_j}(M_i) \quad (14.4)$$

$$s = (i + j) \text{ modulo-} N$$

where T_{K_j} indicates a transformation under the key, K_j , and x modulo- y is defined as the remainder of dividing x by y . Thus $s = 0, 1, 2, 3$. A cryptanalyst intercepting one of the ciphertext messages $C_s = C_0, C_1, C_2,$ or C_3 would have no way of determining which of the four keys was used, and therefore whether the correct message is $M_0, M_1, M_2,$ or M_3 . A cipher system in which the number of messages, the number of keys, and the number of ciphertext transformations are all equal is said to have perfect secrecy if and only if the following two conditions are met:

1. There is only one key transforming each message to each ciphertext.
2. All keys are equally likely.

If these conditions are not met, there would be some message M_i such that for a given C_j , there is no key that can decipher C_j into M_i , implying that $P(M_i|C_j) = 0$ for some i and j . The cryptanalyst could then eliminate certain plaintext messages from consideration, thereby simplifying the task. Perfect secrecy is a very desirable objective since it means that the cipher system is unconditionally secure. It should be apparent, however, that for systems which transmit a large number of messages, the amount of key that must be distributed for perfect secrecy can result in formidable management problems, making such systems impractical. Since in a system with perfect secrecy, the number of different keys is at least as great as the number of possible messages, if we allow messages of unlimited length, perfect secrecy requires an infinite amount of key.

Example 14.1 Breaking a Cipher System When the Key Space is Smaller Than the Message Space

Consider that the 29-character ciphertext

G R O B O K B O D R O R O B Y O C Y P I O C D O B I O K B

was produced by a Caesar cipher (see Section 14.1.4) such that each letter has been shifted by K positions, where $1 \leq K \leq 25$. Show how a cryptanalyst can break this code.

Solution

Because the number of possible keys (there are 25) is smaller than the number of possible 29-character meaningful messages (there are a myriad), perfect secrecy cannot be achieved. In the original polyalphabetic cipher of Figure 14.3, a plaintext character is replaced by a letter of increasingly higher rank as the row number (K) increases. Hence, in analyzing the ciphertext, we reverse the process by creating rows such that each ciphertext letter is replaced by letters of decreasing rank. The cipher is easily broken by trying all the keys, from 1 to 25, as shown in Figure 14.5, yielding only one key ($K = 10$) that produces the meaningful message: WHERE ARE THE HEROES OF YESTERYEAR (The spaces have been added.)

Example 14.2 Perfect Secrecy

We can modify the key space of Example 14.1 to create a cipher having perfect secrecy. In this new cipher system each character in the message is encrypted using a *randomly selected* key value. The key, K , is now given by the sequence k_1, k_2, \dots, k_{29} , where each k_i is a random integer in the range (1, 25) dictating the shift used for the i th character; thus there are a total of $(25)^{29}$ different key sequences. Then the 29-character ciphertext in Example 14.1 could correspond to *any* meaningful 29-character message. For example, the ciphertext could correspond to the plaintext (the spaces have been added)

ENGLISH AND FRENCH ARE SPOKEN HERE

derived by the key 2, 4, 8, 16, 6, 18, 20, Most of the 29-character possibilities can be ruled out because they are not meaningful messages (this much is known without the ciphertext). Perfect secrecy is achieved because interception of the ciphertext in this system reveals no additional information about the plaintext message.

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

Figure 14.5 Example of breaking a cipher system when the key space is smaller than the message space.

14.2.2 Entropy and Equivocation

As discussed in Chapter 9, the amount of information in a message is related to the probability of occurrence of the message. Messages with probability of either 0 or 1 contain no information, since we can be very confident concerning our prediction of their occurrence. The more uncertainty there is in predicting the occurrence of a message, the greater is the information content. Hence when each of the messages in a set is equally likely, we can have *no* confidence in our ability to predict the occurrence of a particular message, and the uncertainty or information content of the message is maximum.