1997

Cryptology Through Time

Susan Danelle Vaucher

CRYPTOLOGY THROUGH TIME

by

Susan Danelle Vaucher

CRYPTOLOGY THROUGH TIME

by

Susan Danelle Vaucher

A project submitted in partial fulfillment of the requirements of
the Southern Scholars Honors Program

Southern Adventist University

April, 1997

The Basics

There has always been a need for secrecy. Whether vital wartime military information, a bank transaction, or a letter to a friend, governments, organizations, and individuals have always been and always will be concerned with keeping their secret and transmitting private information securely. Hence, the science known today as cryptology was developed. Cryptology encompasses two opposite yet related disciplines: cryptography and cryptanalysis. The word cryptography is derived from the Greek words krypto (secret) and graphos (writing) [**4**, p. 13] and refers to the aspect of cryptology concerned with enciphering, or disguising, a message in such a way that only the sender and the intended recipient can decipher it. Cryptanalysis is the enemy of cryptography in that cryptanalysis is the science of deciphering the enciphered message without the key and without the consent of the sender or the receiver [**1**, p. 2].

It is important that we understand that cryptology is a science of ciphers and not codes. The difference between a code and a cipher is that in a cipher each letter of the original message is replaced with a different and distinct letter or symbol, or rearranged in such a way that it is unintelligible [**4**, p. 33]. A code replaces entire words or phrases with other words or phrases having a secret meaning. In order to understand a code, the recipient must have access to a code book that "translates" the code into an understandable and meaningful message. Because encoding and decoding require a hard copy, it is difficult to keep the code secret; if channels are not secure for sending messages, they are certainly not secure for transmitting the code. Another problem arises in the event that the code is broken. Another code language must be devised, written out, and distributed to all intended senders and receivers. This is a timely and insecure process which may set back the progress of the operation.

Ciphers, on the other hand, are mathematical systems of disguising a message using a key. The key is usually the confidential part of a mathematical algorithm that arranges the alphabet in a certain pattern. It is also used for deciphering the message.

The advantages of using a cipher instead of a code are numerous. Ciphers allow more explicit transmissions since every letter of the alphabet may be used distinctly to form any word desired. As you will see, the key to a cipher is also more secure to transmit than a code book. Keys are relatively short; they can be memorized or agreed upon ahead of time; and they can be changed frequently, which helps foil the efforts of any diligent cryptanalyst interested in communications between specific sources [**1**, p. 6].

Cryptology, in various forms and styles, has been used for ages to communicate secret information. One of the earliest cryptographic devices dates back to 405 B.C. and the reign of the great Lysander of Sparta [**4**, p. 28]. Lysander devised a simple yet effective apparatus called a scytale which he used to encrypt and decrypt messages sent from one territory to another. The scytale was composed of a cylindrical object such as a staff or baton around which the sender wound a long, thin strip of leather or parchment in such a way that it created a spiral. Then he wrote a message on the parchment lengthwise along the cylinder, placing one letter on each overlapping turn of the spiral, without leaving spaces between words, until he had written a message all the way around the tube, as shown in Figure 1 [**4**, p. 30].

Figure 1. The Spartan scytale.

When the strip was unwound, the messenger carried a long strip of parchment with a column of meaningless, evenly spaced letters down the left hand side, giving no indication of the method of encryption. Only a person with a cylinder of equal diameter could easily decipher the message. Lysander's scytale is the first known *transposition cipher*, a cipher in which "letters remain what they are, but not where they are [**1**, p. 4]."

An ambitious cryptanalyst could attempt to wind the parchment using cylinders of several different diameters, looking for the spiral to start forming words. However, he would first have to know that the key was indeed a cylinder. Several tactics such as writing each word backward or filling every other space with a nonsense letter would make his task much more difficult.

Additive Ciphers

Some of the most common and effective methods of encryption are *substitution ciphers,* which are ciphers involving permutations of the alphabet. A permutation is a mathematical algorithm that assigns to each member of a set a unique member of the same set.

Table 1 represents a permutation of a set, the alphabet. Each letter of the alphabet is assigned a new and unique letter. If we wanted to encode a message using the permutation in Table 1, we would first decide on the *cleartext*, which is the message in plain English. Then we would apply the permutation to arrive at a *ciphertext*, the encrypted message [**1**, p. 3].

Table 1. Alphabetic permutation.

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
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

cleartext: *Send the money on Tuesday*

ciphertext: *eqzpftqyazqkazfgqepmk*

By eliminating spaces between the words in the ciphertext, we give no indication of where words begin and end, making the cryptanalyst's job more difficult. We can assume that the intended receiver will be able to space the words once the message has been deciphered.

Julius Caesar used the above method in the first century B.C. to convey his secret messages. He permuted the alphabet by a standard shift of twenty-three letters to the right, in other words, three letters to the left [**3**, p. 250]. The shift was standard so the key did not change. A permutation of Caesar's alphabet would be represented by Table 2.

Table 2. Caesar's permutation.

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
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

If Caesar sent the message *ohdyhdwplgqljkw*, the recipient would know that to achieve the cleartext he must replace each letter of the ciphertext with the letter three positions before it in the alphabet. Thus the cleartext would read *leaveatmidnight*, which the recipient would easily separate into *leave at midnight*.

This method of permuting the alphabet is known as an *additive cipher* and can be defined using an algorithm in modular arithmetic. Modular arithmetic deals with number systems that repeat themselves in cycles. There are many common examples of such number systems. The days of the week are numbered in cycles; so are twelve and twenty-four hour clocks. Although most of us don't realize it, we tell time in modular arithmetic. Suppose it is 10:00 a.m.. If we want to know what time it will be in five hours according to a twelve hour cycle, we add two hours to 10 to reach the end of that twelve hour cycle and start a new cycle with the three remaining hours. So in five hours it will be 3:00 p.m..

We can represent this operation by the mathematical statement

$$3 \equiv 10 + 5 \pmod{12}.$$

What this statement tells us is that 3 is equal to the remainder of 10 + 5 divided by 12.

If we give each letter of the alphabet a numerical value as represented in Table 3, we notice that the alphabet can also be considered a cycle of numbers.

Table 3. Numerical values for additive ciphers.

a	b	c	d	e	f	g	h	i	j	k	l	m
1	2	3	4	5	6	7	8	9	10	11	12	13
n	o	p	q	r	s	t	u	v	w	x	y	z
14	15	16	17	18	19	20	21	22	23	24	25	26

The cycle consists of 26 numbers and can be permuted using modular arithmetic, specifically by the following algorithm [**3**, p. 252]:

$$y \equiv x + d \pmod{26}$$

y = the numerical value of the ciphertext letter
x = the numerical value of the cleartext letter
d = the number of units shifted

(mod 26) indicates that y will be equal to the remainder of $x + d$ divided by 26

(Note that in the algorithm y, x, and d represent variables, not letters of the alphabet.)

To encrypt the word *to* with a shift of ten units, we have:

For *t*,
$x = 20, \quad d = 10$
$y \equiv 20 + 10 \pmod{26}$
$y \equiv 30 \pmod{26}$
$y = 4$

For *o*,
$x = 15, \quad d = 10$
$y \equiv 15 + 10 \pmod{26}$
$y \equiv 25 \pmod{26}$
$y = 25$

The fourth and twenty-fifth letters of the alphabet are d and y, so *to* is encrypted to *dy*.

The key to this cipher lies in the value assigned to *d*. *d* may be agreed upon ahead of time and may vary. For example, *d* may be the sum of the digits of the date the message is sent. Changing the value of *d* makes it more difficult for the cryptanalyst to find the key to the cipher.

This additive cipher is useless if the cryptanalyst knows that messages are being encrypted using a simple shift of *d* units. There are only twenty-six possible permutations of the alphabet using an additive cipher, as you can see in Table 4. A cryptanalyst who suspects that an additive cipher is being used need only attempt 26 shifts on a small part of the message to determine the proper value for *d*.

Table 4. Permutations using an additive cipher.

d	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	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
1	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	a
2	c	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
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
.																										
.																										
.																										
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

There are other techniques that cryptanalysts use to attempt to decrypt secret messages. One of these involves the use of frequency tables. In the English language, as in any other language, there are certain letters that are used more than others. Researchers have analyzed large quantities of text in a variety of writing styles to determine the frequency with which each letter of the alphabet appears. Although different studies show slight differences in the exact numerical value of the frequencies, they agree on the popularity, so to speak, of the letters. A sample frequency chart is shown in Table 5 [**1**, p. 10].

Table 5. Relative frequencies of the letters of the alphabet.

letter	relative frequency %	letter	relative frequency %
a	8.167	n	6.749
b	1.493	o	7.507
c	2.782	p	1.929
d	4.253	q	0.095
e	12.707	r	5.987
f	2.228	s	6.327
g	2.015	t	9.056
h	6.094	u	2.758
i	6.966	v	0.978
j	0.153	w	2.360
k	0.772	x	0.150
l	4.025	y	1.974
m	2.406	z	0.074

According to Table 5, the letter e is used more than any other letter in the English language. It has a frequency of approximately 13%, which means that approximately thirteen out of every one hundred letters written in English is the letter e.

Frequency tables are useful tools to cryptanalysts. If the cryptanalyst suspects an additive cipher is being used, his first step is to look through the message and note what letter appears the most. If, for instance, the letter y appears more than any other, he next looks at a table similar to Table 4 and finds the letter y under the cleartext letter e. He notices that the shift for this permutation is $d = 20$. By decrypting a small portion of the message with a shift of 20 units to the left, he can soon tell if he guessed the correct shift. If the message does not make sense, it may be that the letter y was substituted for the letter t, which is the next most frequently used letter. Another possibility that the cryptanalyst must face is that the message was encrypted by some other method.

Affine Ciphers

Affine ciphers were developed to add to the security of ciphers such as the above additive cipher. An affine cipher incorporates multiplication into the algorithm for an additive cipher. For this process, our key consists of two numbers d and m. The algorithm [**3**, p. 253] is:

$$y \equiv mx + d \pmod{26}$$

y = the numerical value of the ciphertext letter
m = a positive integer, relatively prime to 26
x = the numerical value of the cleartext letter
d = the number of units shifted

(mod 26) indicates that y will be equal to the remainder of $mx + d$ divided by 26

(Remember that m, x, y, and d are variables, not letters of the alphabet.)

I will demonstrate the use of this algorithm by encrypting the letter g using an affine cipher with $m = 5$ and $d = 11$.

To encrypt g, which is the seventh letter of the alphabet, we have

$x = 7, \quad m = 5, \quad d = 11$
$y \equiv 5(7) + 11 \pmod{26}$
$y \equiv 46 \pmod{26}$
$y = 20.$

Thus, g is encrypted to t, the twentieth letter. Table 6 shows the permutation of the alphabet achieved similarly.

Table 6. Permutation using an affine cipher with $m = 5$ and $d = 11$.

a	b	c	d	e	f	g	h	i	j	k	l	m
p	u	z	e	j	o	t	y	d	i	n	s	x

n	o	p	q	r	s	t	u	v	w	x	y	z
c	h	m	r	w	b	g	l	q	v	a	f	k

By looking at the table, we can see that the permutation does not follow in alphabetical order. A cryptanalyst does not know the key, the values of m and d. If he analyzes a portion of a message and notices that the letter j appears more than any other, he will be correct in assuming that j is the letter assigned to e. Assuming that the message was encrypted using and additive cipher, he will deduce that the value of $d = 5$, since there is a shift of five units between e and j. However, when he decrypts the letter h with a shift of five units, he will arrive at the letter c. From the table we see that this is incorrect; h is actually assigned to o.

There are many possibilities for m and d that will encrypt e to j. The cryptanalyst must make a correct assumption as to the decryption of at least one other letter of the ciphertext in order to begin formulating possibilities.

The Enigma Ciphers

As we have stated, military operations rely largely on transmitting tactical information secretly. An intercepted message could mean a drastic change of plans, at best. Imagine the

consequences of a military power sending messages using a system they believe is completely secure while every single message is being intercepted and deciphered by its opponent. This was the case for Germany during World War II. The German military devised the Enigma machine to encipher and decipher messages. These machines were actual physical pieces of machinery distributed to any stations that sent or received messages. The system was thought to be so secure that the enciphered messages were transmitted over open channels.

The machine consisted of three rotors about the size and shape of hockey pucks. On either side of these rotors were 26 electrical contacts, representing each letter of the alphabet. The Enigma machine also had an input device, similar to a typewriter keyboard, and an output device, which consisted of a series of lights indicating the enciphered letter.

The cleartext letter was entered, sent through the rotors and was converted into ciphertext. The rotors of the Enigma machine were designed in such a way that the first rotor shifted by one space after each letter was entered. When 26 letters of the message had been entered, the rotor completed one full circle. This process was repeated by the second rotor and then the third. No two letters were ever enciphered using the same circuit. Figure 2 [**3**, p. 258] is a diagram visually demonstrating this process for a smaller, six-letter alphabet. Figure 3 [**3**, p. 259] shows the position of the rotors after encrypting one letter.

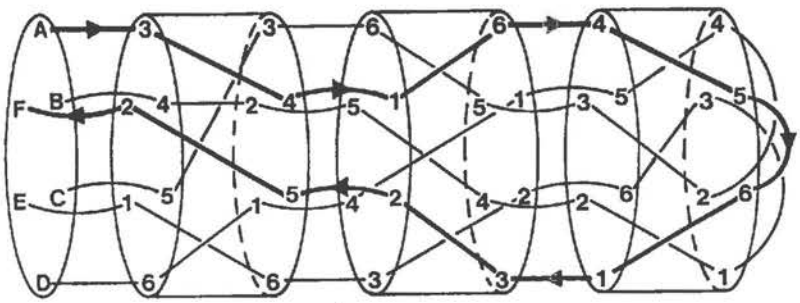

Figure 2. Diagram of a possible Enigma circuit.

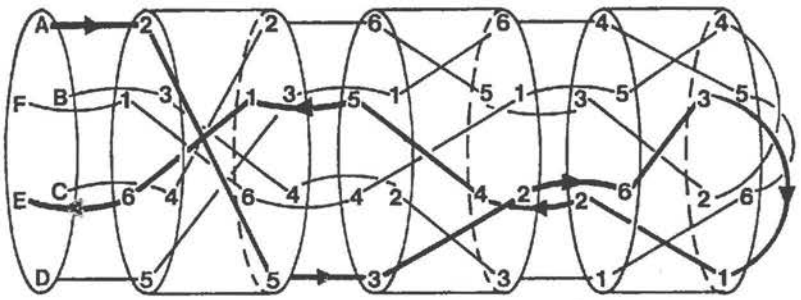

Figure 3. Enigma circuit after encrypting one letter.

The machine also had several settings at which the rotors could be placed to initialize sending or receiving the message. There were basic settings which were standard for all machines in the system. The initial portion of the message was encrypted using one of the basic settings. This portion of the message would include the rotor settings at which the rest of the message was enciphered.

Another feature of the Enigma machine was that the process was completely symmetrical. If the machine encrypted the letter *g* to the letter *b*, it would in turn decrypt *b* to *g*. Although this was essential for using the machines to decipher messages, it was one of the factors that made it possible for the cipher to be broken.

The key to the cipher was discovered by a group of British mathematicians led by Alan Turing. To this day little is known about their activities in breaking the cipher. However, we do know that the group achieved access to a reconstruction of an Enigma machine. They did not know the original setting or the order of the rotors. To solve the problem Turing built a machine of his own. A machine called the Colossus was developed to rapidly check all possible rotor settings for the messages of the day. The Colossus was an ancestor to what we know today as the modern computer [3, pp. 256-62].

Public Key Cryptography

The methods of cryptography that we have studied so far have a common downfall. They are all *symmetrical ciphers*. This means that the same algorithm and key are used to encrypt and decrypt the message. Obviously, the security of the key becomes an issue. Anyone who sends a message can decipher a message. Distributing the key to those with whom I choose to communicate becomes a hazardous procedure. Even if the key is secure, the increased capability of computers makes it more likely for the key to be determined.

With the advance of computer technology, electronic financial transfers, and the Internet, sending messages or data securely has taken on a whole new meaning. The government, corporations and individuals digitally transmit information that, for understandable reasons, must be kept private.

Hence, mathematicians formulated what is known as *public key cryptography*. Public key cryptography uses an *asymmetrical algorithm.* In other words, in these algorithms, the key used to encrypt is different than the key used to decrypt. Using this method, the key to the cipher is made public. Anyone can use it to send an encrypted message. Although the key for encrypting the message is public, the key for decrypting the message is private, and only the intended recipient can decrypt the message.

There are many methods of public key cryptography. One of the oldest and most popular is a system known as the RSA method. This method was discovered at the Massachusetts Institute of Technology by Ron Rivest, Adi Shamir and Leonard Adleman in 1978 [**3**, p. 267]. It has withstood many years of cryptanalysis and is still frequently used because it is easy to understand and implement [**5**, p. 467].

The algorithms behind this method of public key cryptography are based in the theory of prime numbers and modular arithmetic. Prime numbers are positive whole numbers that are divisible only by themselves and by the number 1. Composite numbers are divisible by at least one number other than itself and the number 1. A composite number created by two prime factors is divisible only by itself, by 1, and by its two prime factors.

Computer programs exist that find very large prime numbers and multiply them together to achieve a much larger composite number. However, it is practically impossible for even the most powerful computers to factor products of large prime numbers. On a much smaller scale, we can easily see that the numbers 53 and 71 are prime. But it is not so easy to tell that the number 3763 factors into $53 \cdot 71$.

This difficulty in factoring products of large prime numbers is the basis of public key cryptography. We can easily create a number that is the product of two large prime numbers, distribute it in a public key directory, and receive messages encoded using our public number. In order to decipher the message, we must have access to the factors of this very large composite number, prime numbers of fifty digits or more.

I will demonstrate the RSA method of public key cryptography with an example using small prime numbers in order to perform the calculations without the aid of a computer. (A similar example appears in [3, p. 269].)

First, I will choose two prime numbers p and q. I will keep these numbers secret, while the product pq I will publish in a directory of public keys. I must also choose a secret number N that is *relatively prime* to the product (p-1)(q-1). Two numbers are relatively prime if they have no common prime factors. I will also publish a number M such that $MN \equiv 1 \bmod (p-1)(q-1)$.

I select p = 17 and q = 19. So pq = (17)(19) = 323. The product (p-1)(q-1) = (16)(18) = 288. I can choose N to be any number relatively prime to 288. I choose N = 11. Now I must find the value of M. By the formula above, M must satisfy $11M \equiv 1 \bmod 288$. This means that M must be a number such that when M is multiplied by 11 and divided by 288 the remainder equals 1. To find the value of M, I can use the Euclidean algorithm:

$$\begin{aligned} a &= q_1 b + r_1 \\ b &= q_2 r_1 + r_2 \\ r_1 &= q_3 r_2 + r_3 \\ &\vdots \end{aligned}$$

The algorithm is used in this continued fashion until the desired remainder is achieved.

In our particular case, a = 288, b = 11. I must find the other values.

$$\begin{aligned} & a = q_1 b + r_1 \\ \text{So,} \quad & 288 = q_1(11) + r_1 \\ & 288 = 26(11) + 2. \end{aligned}$$

$$\begin{aligned} & b = q_2 r_1 + r_2 \\ \text{Now,} \quad & 11 = q_2(2) + r_2 \\ & 11 = 5(2) + 1. \end{aligned}$$

I have achieved the desired remainder, 1. Next I substitute from the algorithm above to get 1 equal to the difference of multiples of 11 and 288.

$$\begin{aligned} & 1 = 11 - 5(2). \\ \text{But,} \quad & 2 = 288 - 26(11). \\ \text{So,} \quad & 1 = 11 - 5(288 - 26(11)) \\ & 1 = 131(11) - 5(288). \end{aligned}$$

From this I have established that M = 131.

Now I have all my numbers, private and public, and am ready to send and receive messages. The numbers that I will publish in the public directory are pq = 323 and M = 131. My private numbers are (p-1)(q-1) = 288 and N = 11. Note that a cryptanalyst cannot deduce these secret

numbers from any of my public information. He does not know the value of p or q, only that pq = 323.

First of all, users of this public key system must agree upon a numbering system for the letters of the alphabet. The system previously described in Table 3 will not suffice because we must make a distinction between letters such as *ac*, with numerical value 13, and the letter *m*, with numerical value 13. The most common numbering system is to assign numerical values to the alphabet as shown in Table 7.

Table 7. Numerical values assigned to the alphabet for RSA encryption.

a	b	c	d	e	f	g	h	i	j	k	l	m
00	01	02	03	04	05	06	07	08	09	10	11	12
n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

For an example of how this system works, I will demonstrate the steps necessary for someone to encrypt the letter *d* and for me to decrypt it. To encrypt the letter *d*, the sender must first assign it the numerical value 03. The sender must look up my public numbers, pq = 323 and M = 131, in the directory and encrypt the numerical value of *d* in the following manner:

$$3^{131} \equiv z \pmod{323}.$$

The sender must now calculate the correct value for *z*, the numerical value of the ciphertext letter. The process is tedious by hand calculations but quite simple for a computer. I will go through the steps for finding the value of *z* by hand. Those of you not familiar with modular or exponential arithmetic may want to skip the following explanation.

I know that $3^{131} \equiv 3^{(128+2+1)} \equiv 3^{128} \cdot 3^2 \cdot 3^1 \pmod{323}$. The values of $3^1 \pmod{323}$ and $3^2 \pmod{323}$ are elementary. I can find the value of $3^{128} \pmod{323}$ in the following way:

$3^1 \equiv 3 \pmod{323}$
$3^2 \equiv 9 \pmod{323}$
$3^4 \equiv 9^2 \equiv 81 \pmod{323}$
$3^8 \equiv 81^2 \equiv 101 \pmod{323}$
$3^{16} \equiv 101^2 \equiv 188 \pmod{323}$
$3^{32} \equiv 188^2 \equiv 137 \pmod{323}$
$3^{64} \equiv 137^2 \equiv 35 \pmod{323}$
$3^{128} \equiv 35^2 \equiv 256 \pmod{323}$

From the above steps, I see that

$$\begin{aligned} 3^{131} &\equiv 3^{128} \cdot 3^2 \cdot 3^1 \\ &\equiv 256 \cdot 9 \cdot 3 \\ &\equiv 129 \pmod{323}. \end{aligned}$$

So $z = 129$ is the encrypted numerical value sent to me through my computer system.

But how do I know that 129 represents the letter *d*? Here is where my secret number N comes into play. Remember that my secret numbers are $(p-1)(q-1) = 288$ and $N = 11$. When I receive the number 129, I use N to decipher 129 in the following way:

$$129^{11} \equiv w \pmod{323}.$$

The value that I find for w will be the numerical value for the cleartext letter sent. This again is a tedious process by hand, but it is fast and simple when done by computer. I know that $129^{11} = 129^8 \cdot 129^2 \cdot 129^1$. By finding the values of each of these factors (mod 323), I can arrive at the value for w.

First I see that

$129^1 \equiv 129 \pmod{323}$
$129^2 \equiv 168 \pmod{323}$
$129^4 \equiv 168^2 \equiv 123 \pmod{323}$
$129^8 \equiv 123^2 \equiv 271 \pmod{323}$.

So,

$$\begin{aligned} 129^{11} &\equiv 129^8 \cdot 129^2 \cdot 129^1 \\ &\equiv 271 \cdot 168 \cdot 129 \\ &\equiv 3 \pmod{323}. \end{aligned}$$

I arrive at $w = 3$, or 03, which is the numerical value for the letter *d*, the letter that was originally encrypted and sent.

In reality the numbers used for p and q are as large as 50 digits long, making the products pq and (p-1)(q-1) over one hundred digits long. Remember that p and q are prime, thus the only factors of pq are p and q. This number pq is practically impossible to factor. A cryptanalyst must know the values of p and q to find the value of (p-1)(q-1). Remember that I chose N and found M based on (p-1)(q-1) = 288.

One of the important features of the RSA method is that it allows users to sign their messages in a way that eliminates impostors. The signature is applied at the end of the message and uses the private and public numbers of both the sender and the recipient in a fashion similar to that of sending a message. For example, Adam sends Betty a message and wishes to sign it so that Betty knows that it is from him and not from Chuck. After completing the message, Adam breaks his name down into numerical values and uses his secret number to encipher it. He then enciphers the new numbers using Betty's public numbers and sends the message to her. Betty receives the twice encrypted signature. She first decrypts it using her private numbers and arrives at the signature encrypted with Adam's private numbers. Since the message was enciphered using Adam's private numbers, it can be deciphered using his public numbers which are accessible to Betty through the directory. Adam's public numbers will only be useful in deciphering a message that was enciphered using Adam's private numbers. If Betty is successful in

deciphering the signature she is certain that it was sent by Adam.

Several questions arise concerning the use of the RSA method. First of all, how do we find two 50-digit prime numbers? Mathematicians and computer programmers have developed computer software that is capable of finding these very large numbers. The software must be used on the proper hardware and very powerful computers are required. Companies specializing in finding these numbers sell them to qualified buyers.

Will we ever run out of usable prime numbers? The answer is no. Prime numbers of up to 512 bits in length can be used for this method of encryption. Approximately 10^{151} of these prime numbers exist. Compare that to 10^{77} atoms in the universe. This large number of primes virtually eliminates the chance of two people accidentally picking the same number [**5**, p. 258].

There are legal questions that arise from such secure methods of encryption. As software used for public key cryptology becomes faster and more refined, it is more practical for private citizens and corporations to implement. The government is concerned by the increased availability of these methods and has placed certain restrictions on the sale and distribution of software used to implement them. The main concern is that by allowing these encryption systems to be distributed publicly, the government will lose access to information it may need.

Although most of us feel that the government should not have access to our information, there are certain instances when it is necessary. Our government runs intelligence operations that keep track of the activities of foreign and domestic organizations and governments that may possibly pose a threat to our national security. Also, if these powerful encryption methods were publicly available, organized crime operations right here in our country would be able to secure their transactions and transmit plans in such a way that they would be unobtainable for prosecution.

For this reason, the Clinton administration has passed laws regulating domestic and foreign distribution of powerful encryption software [2, p. 33].

Developers and potential users of this software hold a different point of view. By limiting distribution, the government has regulated the right to privacy and has limited the developers' financial gain. Regardless of regulations by the U.S. government, foreign countries will eventually develop the software to implement this type of encryption system. The same technology will eventually be used, without any benefit to our national developers and our economic system.

Another argument against the government's position arises from corporations who wish to transmit financial, legal and development information securely, without their competition eavesdropping. Many corporations have bases around the world and transmit information electronically. If the information is not highly secure, rival corporations can intercept and decipher messages vital to a corporation's success. The government spends millions of dollars each year in efforts to stop this violation of privacy and to prosecute guilty parties. This problem would not exist if messages were transferred using the restricted cryptographic techniques.

One possible compromise between these points of view would be to devise an escrow system. Upon receiving prime numbers, the buyer must register them with a neutral agency. The numbers would be accessed only as required and permissible by law. If the market is opened promptly, the American corporations will supply the hardware and software. Since the technology would be based domestically, foreign subscribers would be required to adhere to the same procedure. A system such as this one would give the public the right to privacy while allowing the government to access delinquent operations and maintain national security.

Computer technology is increasing everyday. The RSA method and other methods of encryption that depend on the inability of computers to perform certain functions may one day become obsolete. Already computers and computer software have been built that will factor numbers originally used for the RSA method. Although the use of larger numbers has temporarily eliminated this problem, public key cryptography may one day be useless. However, the need for privacy will not become obsolete. New methods will develop to replace the old, the science of secrecy will continue evolving to meet the needs of society.

References

1. A. Beutelspacher, *Cryptology,* Mathematical Association of America, Washington, D.C., 1994.
2. K. Dam and H. Lin, National Cryptography Policy for the Information Age, *Issues in Science and Technology*, 12 (Summer 1996), 33-38.
3. D. M. Davis, *The Nature and Power of Mathematics,* Princeton University Press, New Jersey, 1993.
4. J. Laffin, *Codes and Ciphers,* Abelard-Schuman, New York, 1973.
5. B. Schneier, *Applied Cryptography* (2nd ed.), Wiley, New York, 1996.