CHAPTER 12
PUBLIC-KEY CRYPTOGRAPHY

PREREQUISITE: Section 2.3.

Codes have been used for centuries by merchants, spies, armies, and diplomats to transmit secret messages. In recent times, the large volume of sensitive material in government and corporate computerized data banks (much of which is transmitted by satellite or over telephone lines) has increased the need for efficient, high-security codes.

It is easy to construct unbreakable codes for one-time use. Consider this "code pad":

Actual Word:	morning	evening	Monday	Tuesday	attack
Code Word:	bat	glxt	king	button	figle

If I send you the message FIGLE BUTTON BAT, there is no way an enemy can know for certain that it means "attack on Tuesday morning" unless he or she has a copy of the pad. Of course, if the same code is used again, the enemy might well be able to break it by analyzing the events that occur after each message.

Although one-time code pads are unbreakable, they are cumbersome and inefficient when many long messages must be routinely sent. Even if the encoding and decoding are done by a computer, it is still necessary to design and supply a new pad (at least as long as the message) to each participant for every message and to make all copies of these pads secure from unautho-

rized persons. This is expensive and impractical when hundreds of thousands of words must be encoded and decoded every day.

For frequent computer-based communication among several parties, the ideal code system would be one in which

1. Each person has efficient, reusable, computer algorithms for encoding and decoding messages.
2. Each person's decoding algorithm is *not* obtainable from his or her encoding algorithm in any reasonable amount of time.

A code system with these properties is called a **public-key system.** Although it may not be clear how condition 2 could be satisfied, it is easy to see the advantages of a public-key system.

The *encoding* algorithm of each participant could be publicly announced — perhaps published in a book (like a telephone directory) — thus eliminating the need for couriers and the security problems associated with the distribution of code pads. This would not compromise secrecy because of condition 2: Knowing a person's *encoding* algorithm would not enable you to determine his or her *decoding* algorithm. So you would have no way of decoding messages sent to another person in his or her code, even though you could send coded messages to that person.

Since the encoding algorithms for a public-key system are available to everyone, forgery appears to be a possibility. Suppose, for example, that a bank receives a coded message claiming to be from Anne and requesting the bank to transfer money from Anne's account into Tom's account. How can the bank be sure the message was actually sent by Anne?

The answer is as simple as it is foolproof. Coding and decoding algorithms are inverses of each other: Applying one after the other (in either order) produces the word you started with. So Anne first uses her secret *decoding* algorithm to write her name; say it becomes Gybx. She then applies the bank's public encoding algorithm to Gybx and sends the result (her "signature") along with her message. The bank uses its secret decoding algorithm on this "signature" and obtains Gybx. It then applies Anne's public *encoding* algorithm to Gybx, which turns it into Anne. The bank can then be sure the message is from Anne, because no one else could use her *decoding* algorithm to produce the word Gybx that is encoded as Anne.

One public-key system was developed by R. Rivest, A. Shamir, and L. Adleman in 1977. Their system, now called the RSA system, is based on elementary number theory. Its security depends on the difficulty of factoring large integers (as described in Section 1.4). Here are the mathematical preliminaries needed to understand the RSA system.

LEMMA 12.1 *Let $p, r, s, c \in \mathbf{Z}$ with p prime. If $p \nmid c$ and $rc \equiv sc \pmod{p}$, then $r \equiv s \pmod{p}$.*

Proof Since $rc \equiv sc \pmod{p}$, p divides $rc - sc = (r - s)c$. By Theorem 1.8, $p \mid (r - s)$ or $p \mid c$. Since $p \nmid c$, we have $p \mid (r - s)$, and hence $r \equiv s \pmod{p}$. ∎

LEMMA 12.2 *(FERMAT'S LITTLE THEOREM) If p is prime, $a \in \mathbf{Z}$, and $p \nmid a$, then $a^{p-1} \equiv 1 \pmod{p}$.*

Proof* None of the numbers $a, 2a, 3a, \ldots, (p-1)a$ is congruent to 0 modulo p by Exercise 1. Consequently, each of them must be congruent to one of $1, 2, 3, \ldots, p-1$ by Corollary 2.5 and Theorem 2.3. If two of them were congruent to the same one, say $ra \equiv i \equiv sa \pmod{p}$ with

$$1 \leq i, r, s \leq p - 1,$$

then we would have $r \equiv s \pmod{p}$ by Lemma 12.1 (with $c = a$). This is impossible because no two of the numbers $1, 2, 3, \ldots, p-1$ are congruent modulo p (the difference of any two is less than p and hence not divisible by p). Therefore in some order $a, 2a, 3a, \ldots, (p-1)a$ are congruent to $1, 2, 3, \ldots, p-1$. By repeated use of Theorem 2.2,

$$a \cdot 2a \cdot 3a \cdots (p-1)a \equiv 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod{p}.$$

Rearranging the left side shows that

$$a \cdot a \cdot a \cdots a \cdot 1 \cdot 2 \cdot 3 \cdots (p-1) \equiv 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod{p}$$
$$a^{p-1}(1 \cdot 2 \cdot 3 \cdots (p-1)) \equiv 1(1 \cdot 2 \cdot 3 \cdots (p-1)) \pmod{p}.$$

Now $p \nmid (1 \cdot 2 \cdot 3 \cdots (p-1))$ (if it did, p would divide one of the factors by Corollary 1.9). Therefore $a^{p-1} \equiv 1 \pmod{p}$ by Lemma 12.1 (with $c = 1 \cdot 2 \cdot 3 \cdots (p-1)$). ∎

Throughout the rest of this discussion p and q are distinct positive primes. Let $n = pq$ and $k = (p-1)(q-1)$. Choose d such that $(d,k) = 1$. Then the equation $dx = 1$ has a solution in $\mathbf{Z}_k$ by Corollary 2.9. Therefore the congruence $dx \equiv 1 \pmod{k}$ has a solution in $\mathbf{Z}$; call it e.

THEOREM 12.3 *Let p, q, n, k, e, d be as above. Then $b^{ed} \equiv b \pmod{n}$ for every $b \in \mathbf{Z}$.*

Proof Since e is a solution of $dx \equiv 1 \pmod{k}$, $de - 1 = kt$ for some t. Hence $ed = kt + 1$, so that

$$b^{ed} = b^{kt+1} = b^{kt}b^1 = b^{(p-1)(q-1)t}b = (b^{p-1})^{(q-1)t}b.$$

If $p \nmid b$, then by Lemma 12.2,

$$b^{ed} = (b^{p-1})^{(q-1)t}b \equiv (1)^{(q-1)t}b \equiv b \pmod{p}.$$

* A shorter alternate proof, using group theory, is outlined in Exercise 15 of Section 7.8.

If $p \mid b$, then b and every one of its powers are congruent to 0 modulo p. Therefore in every case, $b^{ed} \equiv b \pmod{p}$. A similar argument shows that $b^{ed} \equiv b \pmod{q}$. By the definition of congruence,

$$p \mid (b^{ed} - b) \qquad \text{and} \qquad q \mid (b^{ed} - b).$$

Therefore $pq \mid (b^{ed} - b)$ by Exercise 2. Since $pq = n$, this means that $n \mid (b^{ed} - b)$, and hence $b^{ed} \equiv b \pmod{n}$. ∎

The least residue modulo n of an integer c is the remainder r when c is divided by n. By the Division Algorithm, $c = nq + r$, so that $c - r = nq$, and hence $c \equiv r \pmod{n}$. Since two numbers strictly between 0 and n cannot be congruent modulo n, the least residue of c is the only integer between 0 and n that is congruent to c modulo n.

We can now describe the mechanics of the RSA system, after which we shall show how it satisfies the conditions for a public-key system. The message to be sent is first converted to numerical form by replacing each letter or space by a two-digit number:*

$$\text{space} = 00,\ \text{A} = 01,\ \text{B} = 02,\ \ldots,\ \text{Y} = 25,\ \text{Z} = 26.$$

For instance, the word GO is written as the number 0715 and WEST is written 23051920, so that the message "GO WEST" becomes the number 07150023051920, which we shall denote by B.

Let p, q, n, k, d, e be as in Theorem 12.3, with p and q chosen so that $B < pq = n$. To encode message B, compute the least residue of B^e modulo n; denote it by C. Then C is the coded form of B. Send C in any convenient way.

The person who receives C decodes it by computing the least residue of C^d modulo n. This produces the original message for the following reasons. Since B^e is congruent modulo n to its least residue C, Theorem 12.3 shows that

$$C^d \equiv (B^e)^d = B^{ed} \equiv B \pmod{n}.$$

The least residue of C^d is the only number between 0 and n that is congruent to C^d modulo n and $0 < B < n$. So the original message B is the least residue of C^d.

Before presenting a numerical example, we show that the RSA system satisfies the conditions for a public-key system:

1. When the RSA system is used in practice, p and q are large primes (on the order of 100 digits each). As noted in Section 1.4 such primes can be quickly identified by a computer. Even though B, e,

* More numbers could be used for puncutation marks, numerals, special symbols, etc. But this will be sufficient for illustrating the basic concepts.

C, d are large numbers, there are fast algorithms for finding the least residues of B^e and C^d modulo n. They are based on binary representation of the exponent and do *not* require direct computation of B^e or C^d (which would be gigantic numbers). See Knuth [33] for details. So the encoding and decoding algorithms of the RSA system *are* computationally efficient.

2. To use the RSA system, each person in the network uses a computer to choose appropriate p, q, d and then determines n, k, e. The numbers e and n for the encoding algorithm are publicly announced, but the prime factors p, q of n and the numbers d and k are kept secret. Anyone with a computer can encode messages by using e and n. But there is no practical way for outsiders to determine d (and hence the decoding algorithm) without first finding p and q by factoring n.* With present technology this would take millions of years, as explained in Section 1.4. So the RSA system appears secure, as long as new and very fast methods of factoring are not developed.

Even when n is chosen as above, there may be some messages that in numerical form are larger than n. In such cases the original message is broken into several blocks, each of which is less than n. Here is an example, due to Rivest-Shamir-Adleman.

EXAMPLE Let $p = 47$ and $q = 59$. Then $n = pq = 47 \cdot 59 = 2773$ and $k = (p-1)(q-1) = 46 \cdot 58 = 2668$.** *Let* $d = 157$, which is easily shown to be relatively prime to 2668. Solving the congruence $157x \equiv 1 \pmod{2668}$ shows that $e = 17$ (see Exercise 15 in Section 13.1). We shall encode the message "IT'S ALL GREEK TO ME." We can encode only numbers less than $n = 2773$. So we write the message in two-letter blocks (and denote spaces by #):

I T	S #	A L	L #	G R
0920	1900	0112	1200	0718
E E	K #	T O	# M	E #
0505	1100	2015	0013	0500.

Then each block is a number less than 2773. The first block, 0920, is encoded by using $e = 17$ and a computer to calculate the least residue of 920^{17} modulo 2773:

$$920^{17} \equiv 948 \pmod{2773}.$$

* Alternatively, one might try to find k and then solve the congruence $ex \equiv 1 \pmod{k}$ to get d. But this can be shown to be computationally equivalent to factoring n, so no time is saved.

** These numbers will illustrate the concepts. But they are too small to provide a secure code since 2773 can be factored by hand.

The other blocks are encoded similarly, so the coded form of the message is

0948	2342	1084	1444	2663
2390	0778	0774	0219	1655.

A person receiving this message would use $d = 157$ to decode each block. For instance, to decode 0948, the computer calculates

$$948^{157} \equiv 920 \pmod{2773}.$$

This is the original first block 0920 = IT.

For more information on cryptography and the RSA system, see DeMillo-Davida [36], Diffie-Hellman [37], Rivest-Shamir-Adleman [38], and Simmons [39].

EXERCISES

A. 1. Let p be a prime and $k, a \in \mathbf{Z}$ such that $p \nmid a$ and $0 < k < p$. Prove that $ka \not\equiv 0 \pmod{p}$. [*Hint:* Theorem 1.8.]

2. If p and q are distinct primes such that $p \mid c$ and $q \mid c$, prove that $pq \mid c$. [*Hint:* If $c = pk$, then $q \mid pk$; use Theorem 1.8.]

3. Use a calculator and the RSA encoding algorithm with $e = 3$, $n = 2773$ to encode these messages:

   (a) GO HOME (b) COME BACK (c) DROP DEAD

   [*Hint:* Use 2-letter blocks and don't omit spaces.]

4. Prove this version of Fermat's Little Theorem: If p is a prime and $a \in \mathbf{Z}$, then $a^p \equiv a \pmod{p}$. [*Hint:* Consider two cases, $p \mid a$ and $p \nmid a$; use Lemma 12.2 in the second case.]

B. 5. Find the decoding algorithm for the code in Exercise 3.

6. Let C be the coded form of a message that was encoded by using the RSA algorithm. Suppose that you discover that C and the encoding modulus n are *not* relatively prime. Explain how you could factor n and thus find the decoding algorithm. [The probability of such a C occurring is less than 10^{-99} when the prime factors p, q of n have more than 100 digits.]