LECTURE 5, CS442.

CONTENTS:

We will cover the Digital Signature Algorithm in detail in Lecture 6.

DISCRETE LOGARITHM is used both in the Digital Signature Algorithm and in Diffie Hellman key exchange (Lecture 7).

All the above material is from:
"CRYPTOGRAPHY AND NETWORK SECURITY"
3rd EDITION, BY WILLIAM STALLINGS.
PRENTICE HALL, ISBN 0-13-091429-0.

Let S be the set of even integers (positive, negative, and 0) under the usual operations of addition and multiplication. S is a commutative ring. The set of all n-square matrices defined in the preceding example is not a commutative ring.

Next, we define an **integral domain**, which is a commutative ring that obeys the following axioms:

(M5) Multiplicative identity: There is an element 1 in R such that $a1 = 1a = a$ for all a in R.

(M6) No zero divisors: If a, b in R and $ab = 0$, then either $a = 0$ or $b = 0$.

Let S be the set of integers, positive, negative, and 0, under the usual operations of addition and multiplication. S is an integral domain.

Fields

A **field** F, sometimes denoted by $\{F, +, \times\}$, is a set of elements with two binary operations, called *addition* and *multiplication*, such that for all a, b, c in F the following axioms are obeyed:

(A1–M6) F is an integral domain; that is, F satisfies axioms A1 through A5 and M1 through M6.

(M7) Multiplicative inverse: For each a in F, except 0, there is an element a^{-1} in F such that $aa^{-1} = (a^{-1})a = 1$.

In essence, a field is a set in which we can do addition, subtraction, multiplication, and division without leaving the set. Division is defined with the following rule: $a/b = a(b^{-1})$.

Familiar examples of fields are the rational numbers, the real numbers, and the complex numbers. Note that the set of all integers is not a field, because not every element of the set has a multiplicative inverse; in fact, only the elements 1 and -1 have multiplicative inverses in the integers.

Figure 4.1 summarizes the axioms that define groups, rings, and fields.

4.2 MODULAR ARITHMETIC

Given any positive integer n and any integer a, if we divide a by n, we get an integer quotient q and an integer remainder r that obey the following relationship:

$$a = qn + r \qquad 0 \le r < n; q = \lfloor a/n \rfloor$$

where $\lfloor x \rfloor$ is the largest integer less than or equal to x.

Figure 4.2 demonstrates that, given a and positive n, it is always possible to find q and r that satisfy the preceding relationship. Represent the integers on the number line; a will fall somewhere on that line (positive a is shown, a similar demonstration can be made for negative a). Starting at 0, proceed to n, $2n$, up to qn such that $qn \le a$ and $(q + 1)n > a$. The distance from qn to a is r, and we have found the unique values of q and r. The remainder r is often referred to as a **residue**.

$a = 11;$	$n = 7;$	$11 = 1 \times 7 + 4;$	$r = 4$
$a = -11;$	$n = 7;$	$-11 = (-2) \times 7 + 3;$	$r = 3$

If a is an integer and n is a positive integer, we define $a \bmod n$ to be the remainder when a is divided by n. Thus, for any integer a, we can always write

$$a = \lfloor a/n \rfloor \times n + (a \bmod n)$$

$11 \bmod 7 = 4; \qquad -11 \bmod 7 = 3$

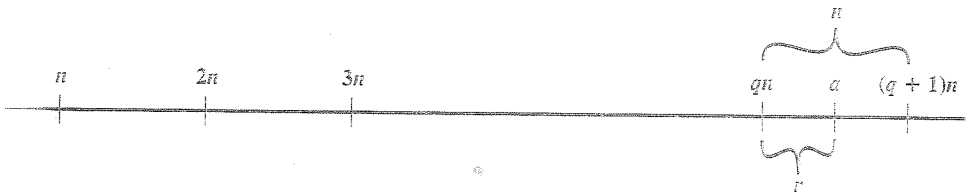
Two integers a and b are said to be **congruent modulo** n, if $(a \bmod n) = (b \bmod n)$. This is written as $a \equiv b \bmod n$.

$73 \equiv 4 \bmod 23; \qquad 21 \equiv -9 \bmod 10$

Divisors

We say that a nonzero b divides a if $a = mb$ for some m, where a, b, and m are integers. That is, b divides a if there is no remainder on division. The notation $b|a$ is commonly used to mean b divides a. Also, if $b|a$, we say that b is a **divisor** of a.

The positive divisors of 24 are 1, 2, 3, 4, 6, 8, 12, and 24.



The Relationship $a = qn + r$, $0 \le r < n$

The following relations hold:

- If $a|1$, then $a = \pm 1$.
- If $a|b$ and $b|a$, then $a = \pm b$.
- Any $b \neq 0$ divides 0.
- If $b|g$ and $b|h$, then $b|(mg + nh)$ for arbitrary integers m and n.

To see this last point, note that

If $b|g$, then g is of the form $g = b \times g_1$ for some integer g_1.
If $b|h$, then h is of the form $h = b \times h_1$ for some integer h_1.

So

$$mg + nh = mbg_1 + nbh_1 = b \times (mg_1 + nh_1)$$

and therefore b divides $mg + nh$.

$b = 7; g = 14; h = 63; m = 3; n = 2.$
$7|14$ and $7|63$. To show: $7|(3 \times 14 + 2 \times 63)$
We have $(3 \times 14 + 2 \times 63) = 7(3 \times 2 + 2 \times 9)$
And it is obvious that $7|(7(3 \times 2 + 2 \times 9))$

Note that if $a \equiv 0 \bmod n$, then $n|a$.

Properties of the Modulo Operator

The modulo operator has the following properties:

1. $a \equiv b \bmod n$ if $n|(a - b)$.
2. $a \equiv b \bmod n$ implies $b \equiv a \bmod n$.
3. $a \equiv b \bmod n$ and $b \equiv c \bmod n$ imply $a \equiv c \bmod n$.

To demonstrate the first point, if $n|(a - b)$, then $(a - b) = kn$ for some k. So we can write $a = b + kn$. Therefore, $(a \bmod n)$ = (remainder when $b + kn$ is divided by n) = (remainder when b is divided by n) = $(b \bmod n)$.

$23 \equiv 8 \pmod 5$	because	$23 - 8 = 15 = 5 \times 3$
$-11 \equiv 5 \pmod 8$	because	$-11 - 5 = -16 = 8 \times (-2)$
$81 \equiv 0 \pmod{27}$	because	$81 - 0 = 81 = 27 \times 3$

The remaining points are as easily proved.

Modular Arithmetic Operations

Note that, by definition (Figure 4.2), the (mod n) operator maps all integers into the set of integers $\{0, 1, \ldots (n - 1)\}$. This suggests the question: Can we perform arith

metic operations within the confines of this set? It turns out that we can; this technique is known as **modular arithmetic**.

Modular arithmetic exhibits the following properties:

1. $[(a \bmod n) + (b \bmod n)] \bmod n = (a + b) \bmod n$
2. $[(a \bmod n) - (b \bmod n)] \bmod n = (a - b) \bmod n$
3. $[(a \bmod n) \times (b \bmod n)] \bmod n = (a \times b) \bmod n$

We demonstrate the first property. Define $(a \bmod n) = r_a$ and $(b \bmod n) = r_b$. Then we can write $a = r_a + jn$ for some integer j and $b = r_b + kn$ for some integer k. Then

$$\begin{aligned}(a + b) \bmod n &= (r_a + jn + r_b + kn) \bmod n \\ &= (r_a + r_b + (k + j)n) \bmod n \\ &= (r_a + r_b) \bmod n \\ &= [(a \bmod n) + (b \bmod n)] \bmod n\end{aligned}$$

The remaining properties are as easily proved. Here are examples of the three properties:

$11 \bmod 8 = 3; \quad 15 \bmod 8 = 7$
$[(11 \bmod 8) + (15 \bmod 8)] \bmod 8 = 10 \bmod 8 = 2$
$(11 + 15) \bmod 8 = 26 \bmod 8 = 2$
$[(11 \bmod 8) - (15 \bmod 8)] \bmod 8 = -4 \bmod 8 = 4$
$(11 - 15) \bmod 8 = -4 \bmod 8 = 4$
$[(11 \bmod 8) \times (15 \bmod 8)] \bmod 8 = 21 \bmod 8 = 5$
$(11 \times 15) \bmod 8 = 165 \bmod 8 = 5$

Exponentiation is performed by repeated multiplication, as in ordinary arithmetic. (We have more to say about exponentiation in Chapter 8.)

To find $11^7 \bmod 13$, we can proceed as follows:
$11^2 = 121 \equiv 4 \bmod 13$
$11^4 \equiv 4^2 \equiv 3 \bmod 13$
$11^7 \equiv 11 \times 4 \times 3 \equiv 132 \equiv 2 \bmod 13$

Thus, the rules for ordinary arithmetic involving addition, subtraction, and multiplication carry over into modular arithmetic.

Table 4.1 provides an illustration of modular addition and multiplication modulo 8. Looking at addition, the results are straightforward and there is a regular pattern to the matrix. Also, as in ordinary addition, there is an additive inverse, or negative, to each integer in modular arithmetic. In this case, the negative of an integer x is the integer y such that $x + y \equiv 0 \bmod 8$. To find the additive inverse of an integer in the left-hand column, scan across the corresponding row of the matrix to find the value 0; the integer at the top of that column is the additive inverse; thus $2 + 6 = 0 \bmod 8$. Similarly, the entries in the multiplication table are straightforward. In ordinary arithmetic, there is a multiplicative inverse, or reciprocal, to each integer. In modular arithmetic mod 8, the multiplicative inverse of x is the integer y such that $x \times y \equiv 1 \bmod 8$. Now, to find the multiplicative inverse of an integer from the multiplication table, scan across the matrix in the row for that integer to find the value 1; the integer at the top of that column is the multiplicative inverse; thus $3 \times 3 = 1 \bmod 8$. Note that not all integers mod 8 have a multiplicative inverse; more about that later.

Properties of Modular Arithmetic

Define the set Z_n as the set of nonnegative integers less than n:

$$Z_n = \{0, 1, \ldots, (n-1)\}$$

This is referred to as the **set of residues**, or **residue classes** modulo n. To be more precise, each integer in Z_n represents a residue class. We can label the residue classes modulo n as $[0], [1], [2], \ldots, [n-1]$, where

$$[r] = \{a: a \text{ is an integer}, a \equiv r \bmod n\}$$

The residue classes modulo 4 are

$$[0] = \{\ldots, -16, -12, -8, -4, 0, 4, 8, 12, 16, \ldots\}$$
$$[1] = \{\ldots, -15, -11, -7, -3, 1, 5, 9, 13, 17, \ldots\}$$
$$[2] = \{\ldots, -14, -10, -6, -2, 2, 6, 10, 14, 18, \ldots\}$$
$$[3] = \{\ldots, -13, -9, -5, -1, 3, 7, 11, 15, 19, \ldots\}$$

Of all the integers in a residue class, the smallest nonnegative integer is the one usually used to represent the residue class. Finding the smallest nonnegative integer to which k is congruent modulo n is called **reducing k modulo n**.

If we perform modular arithmetic within Z_n, the properties shown in Table 4.2 hold for integers in Z_n. Thus, Z_n is a commutative ring with a multiplicative identity element (Figure 4.1).

There is one peculiarity of modular arithmetic that sets it apart from ordinary arithmetic. First, observe that, as in ordinary arithmetic, we can write the following:

Table 4.1 Arithmetic Modulo 8

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

(a) Addition modulo 8

×	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

(b) Multiplication modulo 8

w	$-w$	w^{-1}
0	0	—
1	7	1
2	6	—
3	5	3
4	4	—
5	3	5
6	2	—
7	1	7

(c) Additive and multiplicative inverses modulo 8

$$\textbf{if } (a + b) \equiv (a + c) \bmod n \quad \textbf{then} \quad b \equiv c \bmod n \tag{4.1}$$

$$(5 + 23) \equiv (5 + 7) \bmod 8; \qquad 23 \equiv 7 \bmod 8$$

Equation (4.1) is consistent with the existence of an additive inverse. Adding the additive inverse of a to both sides of Equation (4.1), we have

$$((-a) + a + b) \equiv ((-a) + a + c) \bmod n$$
$$b \equiv c \bmod n$$

However, the following statement is true only with the attached condition:

$$\textbf{if } (a \times b) \equiv (a \times c) \bmod n \textbf{ then } b \equiv c \bmod n \qquad \textbf{if } a \text{ is relatively prime to } n \tag{4.2}$$

Table 4.2 Properties of Modular Arithmetic for Integers in Z_n

Property	Expression
Commutative laws	$(w + x) \bmod n = (x + w) \bmod n$ $(w \times x) \bmod n = (x \times w) \bmod n$
Associative laws	$[(w + x) + y] \bmod n = [w + (x + y)] \bmod n$ $[(w \times x) \times y] \bmod n = [w \times (x \times y)] \bmod n$
Distributive laws	$[w \times (x + y)] \bmod n = [(w \times x) + w \times y)] \bmod n$ $[w + (x \times y)] \bmod n = [(w + x) + w + y)] \bmod n$
Identities	$(0 + w) \bmod n = w \bmod n$ $(1 \times w) \bmod n = w \bmod n$
Additive inverse $(-w)$	For each $w \in Z_n$, there exists a z such that $w + z \equiv 0 \bmod n$.

where the term *relatively prime* is defined as follows: Two integers are **relatively prime** if their only common positive integer factor is 1. Similar to the case of Equation (4.1), we can say that Equation (4.2) is consistent with the existence of a multiplicative inverse. Applying the multiplicative inverse of a to both sides of Equation (4.1), we have

$$((a^{-1})ab) \equiv ((a^{-1})ac) \bmod n$$

$$b \equiv c \bmod n$$

To see this, consider an example in which the condition of Equation (4.2) does not hold. The integers 6 and 8 are not relatively prime, since they have the common factor 2. We have the following:

$$6 \times 3 = 18 \equiv 2 \bmod 8$$
$$6 \times 7 = 42 \equiv 2 \bmod 8$$

Yet $3 \neq 7 \bmod 8$.

The reason for this strange result is that for any general modulus n, a multiplier a that is applied in turn to the integers 0 through $(n - 1)$ will fail to produce a complete set of residues if a and n have any factors in common.

With $a = 6$ and $n = 8$.

Z_8	0	1	2	3	4	5	6	7
Multiply by 6	0	6	12	18	24	30	36	42
Residues	0	6	4	2	0	6	4	2

Because we do not have a complete set of residues when multiplying by 6, more than one integer in Z_8 maps into the same residue. Specifically, $6 \times 0 \bmod 8 = 6 \times 4 \bmod 8$; $6 \times 1 \bmod 8 = 6 \times 5 \bmod 8$; and so on. Because this is a many-to-one mapping, there is not a unique inverse to the multiply operation.

However, if we take $a = 5$ and $n = 8$, whose only common factor is 1,

Z_8	0	1	2	3	4	5	6	7
Multiply by 5	0	5	10	15	20	25	30	35
Residues	0	5	2	7	4	1	6	3

The line of residues contains all the integers in Z_8, in a different order.

In general, an integer has a multiplicative inverse in Z_n if that integer is relatively prime to n. Table 4.1c shows that the integers 1, 3, 5, and 7 have a multiplicative inverse in Z_8, but 2, 4, and 6 do not.

4.3 EUCLID'S ALGORITHM

One of the basic techniques of number theory is Euclid's algorithm, which is a simple procedure for determining the greatest common divisor of two positive integers.

Greatest Common Divisor

We will use the notation gcd(a, b) to mean the **greatest common divisor** of a and b. The positive integer c is said to be the greatest common divisor of a and b if

1. c is a divisor of a and of b;
2. any divisor of a and b is a divisor of c.

An equivalent definition is the following:

$$\gcd(a, b) = \max[k, \text{ such that } k|a \text{ and } k|b]$$

Because we require that the greatest common divisor be positive, $\gcd(a, b) = \gcd(a, -b) = \gcd(-a, b) = \gcd(-a, -b)$. In general, $\gcd(a, b) = \gcd(|a|, |b|)$.

$$\gcd(60, 24) = \gcd(60, -24) = 12$$

Also, because all nonzero integers divide 0, we have $\gcd(a, 0) = |a|$.

We stated that two integers a and b are relatively prime if their only common positive integer factor is 1. This is equivalent to saying that a and b are relatively prime if $\gcd(a, b) = 1$.

8 and 15 are relatively prime because the positive divisors of 8 are 1, 2, 4, and 8, and the positive divisors of 15 are 1, 3, 5, and 15, so 1 is the only integer on both lists.

Finding the Greatest Common Divisor

Euclid's algorithm is based on the following theorem: For any nonnegative integer a and any positive integer b,

$$\gcd(a, b) = \gcd(b, a \bmod b) \tag{4.3}$$

$$\boxed{\gcd(55, 22) = \gcd(22, 55 \bmod 22) = \gcd(22, 11) = 11}$$

To see that Equation (4.3) works, let $d = \gcd(a, b)$. Then, by the definition of gcd, $d|a$ and $d|b$. For any positive integer b, a can be expressed in the form

$$a = kb + r \equiv r \bmod b$$
$$a \bmod b = r$$

with k, r integers. Therefore, $(a \bmod b) = a - kb$ for some integer k. But because $d|b$, it also divides kb. We also have $d|a$. Therefore, $d|(a \bmod b)$. This shows that d is a common divisor of b and $(a \bmod b)$. Conversely, if d is a common divisor of b and $(a \bmod b)$, then $d|kb$ and thus $d|[kb + (a \bmod b)]$, which is equivalent to $d|a$. Thus, the set of common divisors of a and b is equal to the set of common divisors of b and $(a \bmod b)$. Therefore, the gcd of one pair is the same as the gcd of the other pair, proving the theorem.

Equation (4.3) can be used repetitively to determine the greatest common divisor.

$$\boxed{\begin{aligned}&\gcd(18, 12) = \gcd(12, 6) = \gcd(6, 0) = 6\\&\gcd(11, 10) = \gcd(10, 1) = \gcd(1, 0) = 1\end{aligned}}$$

Euclid's algorithm makes repeated use of Equation (4.3) to determine the greatest common divisor, as follows. The algorithm assumes $a > b > 0$. It is acceptable to restrict the algorithm to positive integers because $\gcd(a, b) = \gcd(|a|, |b|)$.

EUCLID(a, b)

1. A $\leftarrow a$; B $\leftarrow b$
2. **if** B $= 0$ **return** A $= \gcd(a, b)$
3. R $=$ A mod B
4. A $\leftarrow$ B
5. B $\leftarrow$ R
6. **goto** 2

The algorithm has the following progression:

$$A_1 = B_1 \times Q_1 + R_1$$
$$A_2 = B_2 \times Q_2 + R_2$$
$$A_3 = B_3 \times Q_3 + R_3$$
$$A_4 = B_4 \times Q_4 + R_4$$

To find gcd(1970, 1066)

$1970 = 1 \times 1066 + 904$	gcd(1066, 904)
$1066 = 1 \times 904 + 162$	gcd(904, 162)
$904 = 5 \times 162 + 94$	gcd(162, 94)
$162 = 1 \times 94 + 68$	gcd(94, 68)
$94 = 1 \times 68 + 26$	gcd(68, 26)
$68 = 2 \times 26 + 16$	gcd(26, 16)
$26 = 1 \times 16 + 10$	gcd(16, 10)
$16 = 1 \times 10 + 6$	gcd(10, 6)
$10 = 1 \times 6 + 4$	gcd(6, 4)
$6 = 1 \times 4 + 2$	gcd(4, 2)
$4 = 2 \times 2 + 0$	gcd(2, 0)

Therefore, gcd(1970, 1066) = 2

The alert reader may ask how we can be sure that this process terminates. That is, how can we be sure that at some point B divides A? If not, we would get an endless sequence of positive integers, each one strictly smaller than the one before, and this is clearly impossible.

4.4 FINITE FIELDS OF THE FORM GF(*p*)

In Section 4.1, we defined a field as a set that obeys all of the axioms of Figure 4.1 and gave some examples of infinite fields. Infinite fields are not of particular interest in the context of cryptography. However, finite fields play a crucial role in many cryptographic algorithms. It can be shown that the order of a finite field (number of elements in the field) must be a power of a prime p^n, where n is a positive integer. We discuss prime numbers in detail in Chapter 8. Here, we need only say that a prime number is an integer whose only positive integer factors are itself and 1.

The finite field of order p^n is generally written GF(p^n); GF stands for Galois field, in honor of the mathematician who first studied finite fields. Two special cases are of interest for our purposes. For $n = 1$, we have the finite field GF(p); this finite field has a different structure than that for finite fields with $n > 1$ and is studied in this section. In Section 4.6, we look at finite fields of the form GF(2^n).

Finite Fields of Order p

For a given prime, p, the finite field of order p, GF(p) is defined as the set Z_p of integers $\{0, 1, \ldots, p-1\}$, together with the arithmetic operations modulo p.

Recall that we showed in Section 4.2 that the set Z_n of integers $\{0, 1, \ldots, n-1\}$, together with the arithmetic operations modulo n, is a commutative ring (Table 4.2). We further observed that any integer in Z_n has a multiplicative inverse if and only if that integer is relatively prime to n [see discussion of Equation (4.2)]. If n is prime, then all of the nonzero integers in Z_n are relatively prime to n, and therefore there exists a multiplicative inverse for all of the nonzero integers in Z_n. Thus, we can add the following properties to those listed in Table 4.2 for Z_p:

Multiplicative inverse (w^{-1})	For each $w \in Z_p$, $w \neq 0$, there exists a $z \in Z_p$ such that $w \times z \equiv 1 \bmod p$

Because w is relatively prime to p, if we multiply all the elements of Z_p by w, the resulting residues are all of the elements of Z_p permuted. Thus, exactly one of the residues has the value 1. Therefore, there is some integer in Z_p that, when multiplied by w, yields the residue 1. That integer is the multiplicative inverse of w, designated w^{-1}. Therefore, Z_p is in fact a finite field. Further, Equation (4.2) is consistent with the existence of a multiplicative inverse and can be rewritten without the condition

$$\textbf{if } (a \times b) \equiv (a \times c) \bmod p \textbf{ then } b \equiv c \bmod p \tag{4.4}$$

Multiplying both sides of Equation (4.4) by the multiplicative inverse of a, we have

$$((a^{-1}) \times a \times b) \equiv ((a^{-1}) \times a \times c) \bmod p$$
$$b \equiv c \bmod p$$

The simplest finite field is GF(2). Its arithmetic operations are easily summarized as follows:

+	0	1
0	0	1
1	1	0

Addition

×	0	1
0	0	0
1	0	1

Multiplication

w	$-w$	w^{-1}
0	0	—
1	1	1

Inverses

In this case, addition is equivalent to the exclusive-OR (XOR) operation, and multiplication is equivalent to the logical AND operation.

Table 4.3 shows GF(7).

Table 4.3 Arithmetic in GF(7)

+	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

(a) Addition modulo 7

×	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

(b) Multiplication modulo 7

w	$-w$	w^{-1}
0	0	—
1	6	1
2	5	4
3	4	5
4	3	2
5	2	3
6	1	6

(c) Additive and multiplicative inverses modulo 7

Finding the Multiplicative Inverse in GF(*p*)

It is easy to find the multiplicative inverse of an element in GF(p) for small values of p. You simply construct a multiplication table, such as shown in Table 4.3b, and the desired result can be read directly. However, for large values of p, this approach is not practical.

If gcd(m, b) = 1, then b has a multiplicative inverse modulo m. That is, for positive integer $b < m$, there exists a $b^{-1} < m$ such that $bb^{-1} = 1 \bmod m$. Euclid's algorithm can be extended so that, in addition to finding gcd(m, b), if the gcd is 1, the algorithm returns the multiplicative inverse of b.

EXTENDED EUCLID(m, b)

1. (A1, A2, A3) $\leftarrow$ (1, 0, m); (B1, B2, B3) $\leftarrow$ (0, 1, b)
2. **if** B3 = 0 **return** A3 = gcd(m, b); no inverse
3. **if** B3 = 1 **return** B3 = gcd(m, b); B2 = $b^{-1} \bmod m$
4. $Q = \left\lfloor \frac{A3}{B3} \right\rfloor$

5. (T1, T2, T3) ← (A1 − QB1, A2 − QB2, A3 − QB3)
6. (A1, A2, A3) ← (B1, B2, B3)
7. (B1, B2, B3) ← (T1, T2, T3)
8. **goto** 2

Throughout the computation, the following relationships hold:

$$mT1 + bT2 = T3 \qquad mA1 + bA2 = A3 \qquad mB1 + bB2 = B3$$

To see that this algorithm correctly returns gcd(m, b), note that if we equate A and B in Euclid's algorithm with A3 and B3 in the extended Euclid's algorithm, then the treatment of the two variables is identical. At each iteration of Euclid's algorithm, A is set equal to the previous value of B and B is set equal to the previous value of A mod B. Similarly, at each step of the extended Euclid's algorithm, A3 is set equal to the previous value of B3, and B3 is set equal to the previous value of A3 minus the integer quotient of A3 multiplied by B3. This latter value is simply the remainder of A3 divided by B3, which is A3 mod B3.

Note also that if gcd(m, b) = 1, then on the final step we would have B3 = 0 and A3 = 1. Therefore, on the preceding step, B3 = 1. But if B3 = 1, then we can say the following:

$$\begin{aligned} mB1 + bB2 &= B3 \\ mB1 + bB2 &= 1 \\ bB2 &= 1 + mB1 \\ bB2 &\equiv 1 \bmod m \end{aligned}$$

And B2 is the multiplicative inverse of b, modulo m.

Table 4.4 is an example of the execution of the algorithm. It shows that gcd(550, 1759) = 1 and that the multiplicative inverse of 550 is 355; that is, $550 \times 355 \equiv 1 \bmod 1759$.

For a more detailed proof of this algorithm, see [KNUT97].

Table 4.4 Finding the Multiplicative Inverse of 550 in GF(1759)

Q	A1	A2	A3	B1	B2	B3
—	1	0	1759	0	1	550
3	0	1	550	1	−3	109
5	1	−3	109	−5	16	5
21	−5	16	5	106	−339	4
1	106	−339	4	−111	355	1

The Devil said to Daniel Webster: "Set me a task I can't carry out, and I'll give you anything in the world you ask for."

Daniel Webster: "Fair enough. Prove that for n greater than 2, the equation $a^n + b^n = c^n$ has no non-trivial solution in the integers."

They agreed on a three-day period for the labor, and the Devil disappeared.

At the end of three days, the Devil presented himself, haggard, jumpy, biting his lip. Daniel Webster said to him, "Well, how did you do at my task? Did you prove the theorem?'

"Eh? No . . . no, I haven't proved it."

"Then I can have whatever I ask for? Money? The Presidency?"

"What? Oh, that—of course. But listen! If we could just prove the following two lemmas—"

***—The Mathematical Magpie,* Clifton Fadiman**

A number of concepts from number theory are essential in the design of public-key cryptographic algorithms. This chapter provides an overview of the concepts referred to in other chapters. The reader familiar with these topics can safely skip this chapter.

8.1 PRIME NUMBERS[1]

A central concern of number theory are prime numbers. Indeed, whole books have been written on the subject (e.g., [CRAN01], [RIBE96]). In this section we provide an overview relevant to the concerns of this book.

An integer $p > 1$ is a prime number if and only if its only divisors are ± 1 and $\pm p$. Prime numbers play a critical role in number theory and in the techniques discussed in this chapter.

Table 8.1 shows the primes under 2000.

Any integer $a > 1$ can be factored in a unique way as

$$a = p_1^{a_1} p_2^{a_2} \ldots p_t^{a_t}$$

where $p_1 < p_2 < \ldots < p_t$ are prime numbers and where each a_i is a positive integer.

$91 = 7 \times 13$; $11011 = 7 \times 11^2 \times 13$

[1] In this section, unless otherwise noted, we deal only with the nonnegative integers. The use of negative integers would introduce no essential differences.

ıt, and I'll give you

ıan 2, the equation

'evil disappeared.
gard, jumpy, biting
my task? Did you

'ncy?"
ve the following two

Clifton Fadiman

l in the design of pub-
es an overview of the
miliar with these top-

:ed, whole books have
his section we provide

ly divisors are ±1 and
in the techniques dis-

$_i$ is a positive integer

ntegers. The use of negative

Table 8.1 Primes under 2000

2	3	5	7	11	13	17	19	23	29	31	37	41	43	47	53	59	61	67	71	73	79	83	89	97
101	103	107	109	113	127	131	137	139	149	151	157	163	167	173	179	181	191	193	197	199				
211	223	227	229	233	239	241	251	257	263	269	271	277	281	283	293									
307	311	313	317	331	337	347	349	353	359	367	373	379	383	389	397									
401	409	419	421	431	433	439	443	449	457	461	463	467	479	487	491	499								
503	509	521	523	541	547	557	563	569	571	577	587	593	599											
601	607	613	617	619	631	641	643	647	653	659	661	673	677	683	691									
701	709	719	727	733	739	743	751	757	761	769	773	787	797											
809	811	821	823	827	829	839	853	857	859	863	877	881	883	887										
907	911	919	929	937	941	947	953	967	971	977	983	991	997											
1009	1013	1019	1021	1031	1033	1039	1049	1051	1061	1063	1069	1087	1091	1093	1097									
1103	1109	1117	1123	1129	1151	1153	1163	1171	1181	1187	1193													
1201	1213	1217	1223	1229	1231	1237	1249	1259	1277	1279	1283	1289	1291	1297										
1301	1303	1307	1319	1321	1327	1361	1367	1373	1381	1399														
1409	1423	1427	1429	1433	1439	1447	1451	1453	1459	1471	1481	1483	1487	1489	1493	1499								
1511	1523	1531	1543	1549	1553	1559	1567	1571	1579	1583	1597													
1601	1607	1609	1613	1619	1621	1627	1637	1657	1663	1667	1669	1693	1697	1699										
1709	1721	1723	1733	1741	1747	1753	1759	1777	1783	1787	1789													
1801	1811	1823	1831	1847	1861	1867	1871	1873	1877	1879	1889													
1901	1907	1913	1931	1933	1949	1951	1973	1979	1987	1993	1997	1999												

It is useful for what follows to cast this another way. If P is the set of all prime numbers, then any positive integer can be written uniquely in the following form:

$$a = \prod_{p \in P} p^{a_p} \qquad \text{where each } a_p \geq 0$$

The right-hand side is the product over all possible prime numbers p; for any particular value of a, most of the exponents a_p will be 0.

$$3600 = 2^4 \times 3^2 \times 5^2$$

The value of any given positive integer can be specified by simply listing all the nonzero exponents in the foregoing formulation.

The integer 12 is represented by $\{a_2 = 2, a_3 = 1\}$.
The integer 18 is represented by $\{a_2 = 1, a_3 = 2\}$.

Multiplication of two numbers is equivalent to adding the corresponding exponents:

$$k = mn \rightarrow \qquad k_p = m_p + n_p \qquad \text{for all } p \in P$$

$k = 12 \times 18 = 216$
$k_2 = 2 + 1 = 3; k_3 = 1 + 2 = 3$
$216 = 2^3 \times 3^3$

What does it mean, in terms of these prime factors, to say that a divides b ($a|b$)?[2] Any integer of the form p^k can be divided only by an integer that is of a lesser or equal power of the same prime number, p^j with $j \leq k$. Thus, we can say

$$a|b \rightarrow \qquad a_p \leq b_p \qquad \text{for all } p$$

$a = 12; b = 36; 12|36; 12 = 2^2 \times 3; 36 = 2^2 \times 3^2$
$a_2 = 2 = b_2$
$a_3 = 1 \leq 2 = b_3$

It is easy to determine the greatest common divisor[3] of two positive integers if we express each integer as the product of primes.

$$300 = 2^2 \times 3^1 \times 5^2$$
$$18 = 2^1 \times 3^2$$
$$\gcd(18, 300) = 2^1 \times 3^1 \times 5^0 = 6$$

[2]Recall from Chapter 4 that a divides b if there is no remainder on division.
[3]Recall from Chapter 4 that the greatest common divisor of integers a and b, expressed $\gcd(a, b)$, is an integer c that divides both a and b without remainder and that any divisor of a and b is a divisor of c.

In general,

$$k = \gcd(a, b) \quad \rightarrow \quad k_p = \min(a_p, b_p) \text{ for all } p$$

Determining the prime factors of a large number is no easy task, so the preceding relationship does not directly lead to a way of practical method of calculating the greatest common divisor.

8.2 FERMAT'S AND EULER'S THEOREMS

Two theorems that play important roles in public-key cryptography are Fermat's theorem and Euler's theorem.

Fermat's Theorem[4]

Fermat's theorem states the following: If p is prime and a is a positive integer not divisible by p, then

$$a^{p-1} \equiv 1 \bmod p \tag{8.1}$$

Proof: From Chapter 4, we know that if all of the elements of Z_p, where Z_p is the set of integers $\{0, 1, \ldots, p-1\}$, are multiplied by a, modulo p, the result consists of all of the elements of Z_p in some sequence. Furthermore, $a \times 0 \equiv 0 \bmod p$. Therefore, the $(p-1)$ numbers $\{a \bmod p, 2a \bmod p, \ldots (p-1)a \bmod p\}$ are just the numbers $\{1, 2, \ldots, (p-1)\}$ in some order. Multiplying the numbers in both sets and taking the result mod p yields

$$\begin{aligned} a \times 2a \times \ldots \times ((p-1)a) &\equiv [(a \bmod p) \times (2a \bmod p) \times \ldots \times \\ &\quad ((p-1)a \bmod p)] \bmod p \\ &\equiv [1 \times 2 \times \ldots \times (p-1)] \bmod p \\ &\equiv (p-1)! \bmod p \end{aligned}$$

But

$$a \times 2a \times \ldots \times ((p-1)a) = (p-1)!a^{p-1}$$

Therefore,

$$(p-1)!a^{p-1} \equiv (p-1)! \bmod p$$

We can cancel the $(p-1)!$ term because it is relatively prime[5] to p [see Equation (4.5)]. This yields Equation (8.1).

[4] Sometimes referred to as Fermat's little theorem.

[5] From Chapter 4 that two numbers are relatively prime if they have no prime factors in common: their only common divisor is 1. This is equivalent to saying that two numbers are relatively prime if their greatest common divisor is 1.

$a = 7, p = 19$
$7^2 = 49 \equiv 11 \bmod 19$
$7^4 \equiv 121 \equiv 7 \bmod 19$
$7^8 \equiv 49 \equiv 11 \bmod 19$
$7^{16} \equiv 121 \equiv 7 \bmod 19$
$a^{p-1} = 7^{18} = 7^{16} \times 7^2 \equiv 7 \times 11 \equiv 1 \bmod 19$

An alternative form of the theorem is also useful: If p is prime and a is any positive integer, then

$$a^p \equiv a \bmod p \tag{8.2}$$

$p = 5, a = 3, 3^5 = 243 \equiv 3 \bmod 5$
$p = 5, a = 10, 10^5 = 100000 \equiv 10 \bmod 5 \equiv 0 \bmod 5$

Euler's Totient Function

Before presenting Euler's theorem, we need to introduce an important quantity in number theory, referred to as Euler's totient function and written $\phi(n)$, where $\phi(n)$ is the number of positive integers less than n and relatively prime to n.

Determine $\phi(37)$ and $\phi(35)$.
Because 37 is prime, all of the positive integers from 1 through 36 are relatively prime to 37. Thus $\phi(37) = 36$.
To determine $\phi(35)$, we list all of the positive integers less than 35 that are relatively prime to it:

1, 2, 3, 4, 6, 8, 9, 11, 12, 13, 16, 17, 18,
19, 22, 23, 24, 26, 27, 29, 31, 32, 33, 34.

There are 24 numbers on the list, so $\phi(35) = 24$.

Table 8.2 lists the first 30 values of $\phi(n)$. The value $\phi(1)$ is without meaning but is defined to have the value 1.

It should be clear that for a prime number p,

$$\phi(p) = p - 1$$

Now suppose that we have two prime numbers p and q, with $p \neq q$. Then, for n

$$\phi(n) = \phi(pq) = \phi(p) \times \phi(q) = (p - 1) \times (q - 1)$$

To see that $\phi(n) = \phi(pq)$, consider that the set of residues in Z_n is $\{0, 1, \ldots, (pq$ The residues that are not relatively prime to n are the set $\{p, 2p, \ldots, (q - 1)p\}$ set $\{q, 2q, \ldots, (p - 1)q\}$, and 0. Accordingly,

Table 8.2 Some Values of Euler's Totient Function $\phi(n)$

n	$\phi(n)$	n	$\phi(n)$	n	$\phi(n)$
1	1	11	10	21	12
2	1	12	4	22	10
3	2	13	12	23	22
4	2	14	6	24	8
5	4	15	8	25	20
6	2	16	8	26	12
7	6	17	16	27	18
8	4	18	6	28	12
9	6	19	18	29	28
10	4	20	8	30	8

$$\begin{aligned}\phi(n) &= pq - [(q-1) + (p-1) + 1] \\ &= pq - (p+q) + 1 \\ &= (p-1) \times (q-1) \\ &= \phi(p) \times \phi(q)\end{aligned}$$

$\phi(21) = \phi(3) \times \phi(7) = (3-1) \times (7-1) = 2 \times 6 = 12$
where the 12 integers are {1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20}

Euler's Theorem

Euler's theorem states that for every a and n that are relatively prime:

$$a^{\phi(n)} \equiv 1 \bmod n \tag{8.3}$$

$a = 3;\ n = 10;\ \phi(10) = 4;\ 3^4 = 81 \equiv 1 \bmod 10$
$a = 2;\ n = 11;\ \phi(11) = 10;\ 2^{10} = 1024 \equiv 1 \bmod 11$

Proof: Equation (8.3) is true if n is prime, because in that case $\phi(n) = (n-1)$ and Fermat's theorem holds. However, it also holds for any integer n. Recall that $\phi(n)$ is the number of positive integers less than n that are relatively prime to n. Consider the set of such integers, labeled as follows:

$$R = \{x_1, x_2, \ldots, x_{\phi(n)}\}$$

Now multiply each element by a, modulo n:

$$S = \{(ax_1 \bmod n), (ax_2 \bmod n), \ldots, (ax_{\phi(n)} \bmod n)\}$$

The set S is a permutation of R, by the following line of reasoning:

1. Because a is relatively prime to n and x_i is relatively prime to n, ax_i must also be relatively prime to n. Thus, all the members of S are integers less than n that are relatively prime to n.

2. There are no duplicates in S. Refer to Equation (4.2). If $ax_i \bmod n = ax_j \bmod n$, then $x_i = x_j$.

Therefore,

$$\prod_{i=1}^{\phi(n)} (ax_i \bmod n) = \prod_{i=1}^{\phi(n)} x_i$$

$$\prod_{i=1}^{\phi(n)} ax_i \equiv \prod_{i=1}^{\phi(n)} x_i \pmod{n}$$

$$a^{\phi(n)} \times \left[\prod_{i=1}^{\phi(n)} x_i\right] \equiv \prod_{i=1}^{\phi(n)} x_i \pmod{n}$$

$$a^{\phi(n)} \equiv 1 \pmod{n}$$

An alternative form of the theorem is also useful:

$$a^{\phi(n)+1} \equiv a \pmod{n} \tag{8.4}$$

We can develop a corollary to Euler's theorem that is useful in demonstrating the validity of the RSA algorithm (Chapter 9). Given two prime numbers p and q, and integers $n = pq$ and m, with $0 < m < n$, the following relationship holds:

$$m^{\phi(n)+1} = m^{(p-1)(q-1)+1} \equiv m \bmod n \tag{8.5}$$

If $\gcd(m, n) = 1$, that is, if m and n are relatively prime then the relationship holds by virtue of Euler's theorem [Equation (8.3)]. Suppose $\gcd(m, n) \neq 1$. What does this mean? Because $n = pq$, the equality $\gcd(m, n) = 1$ is equivalent to the logical expression (m is not a multiple of p) AND (m is not a multiple of q). If m is a multiple of p, then n and m share the prime factor p and are not relatively prime, and if m is a multiple of q, then n and m share the prime factor q and are not relatively prime. Therefore, the expression $\gcd(m, n) \neq 1$ must be equivalent to the negation of the foregoing logical expression. Therefore, $\gcd(m, n) \neq 1$ is equivalent to the logical expression (m is a multiple of p) OR (m is a multiple of q).

Let us look at the case in which m is a multiple of p, so that the relationship $m = cp$ holds for some positive integer c. In this case, we must have $\gcd(m, q) = 1$. Otherwise, we have m a multiple of p and m a multiple of q and yet $m < pq$. If $\gcd(m, q) = 1$, then Euler's theorem holds and

$$m^{\phi(q)} \equiv 1 \bmod q$$

But then, by the rules of modular arithmetic,

$$[m^{\phi(q)}]^{\phi(p)} \equiv 1 \bmod q$$

$$m^{\phi(n)} \equiv 1 \bmod q$$

Therefore, there is some integer k such that

$$m^{\phi(n)} = 1 + kq$$

Multiplying each side by $m = cp$,

$$m^{\phi(n)+1} = m + kcpq = m + kcn$$
$$m^{\phi(n)+1} \equiv m \bmod n$$

A similar line of reasoning is used for the case in which m is a multiple of q. Thus, Equation (8.5) is proven. An alternative form of this corollary is directly relevant to RSA:

$$\begin{aligned} m^{k\phi(n)+1} &\equiv [(m^{\phi(n)})^k \times m] \bmod n \\ &\equiv [(1)^k \times m] \bmod n \qquad \text{by Euler's theorem} \\ &\equiv m \bmod n \end{aligned} \tag{8.6}$$

8.3 TESTING FOR PRIMALITY

In a number of cryptographic algorithms, it is necessary to select one or more very large prime numbers at random. Thus we are faced with the task of determining whether a given large number is prime. There is no simple yet efficient means of accomplishing this task.

In this section, we present one attractive and popular algorithm. You may be surprised to learn that this algorithm yields a number that is not necessarily a prime. However, the algorithm can yield a number that is almost certainly a prime. This will be explained presently. The algorithm due to Miller and Rabin [MILL75, RABI80] exploits Fermat's theorem [Equation (8.1)], which states that $a^{n-1} \equiv 1 \bmod n$ if n is prime.

The algorithm can be explained as follows. For a candidate odd integer $n \geq 3$, consider the even number $(n - 1)$. This number can be expressed in the form of some power of 2 times an odd number:

$$n - 1 = 2^k q \qquad \text{with } k > 0,\ q \text{ odd}$$

That is, we divide $(n - 1)$ by 2 until the result is an odd number, for a total of k divisions.

Next, we choose an integer a in the range $1 < a < n - 1$. The algorithm then involves computation of the residues modulo n of the following sequence of powers:

$$a^q, a^{2q}, \ldots, a^{2^{k-1}q}, a^{2^k q} \tag{8.7}$$

If n is prime, we know from Fermat's theorem that $a^{2^k q} \bmod n = a^{n-1} \bmod n = 1$. There may or may not be an earlier element of the sequence (8.7) that has a value of 1. To clarify what follows, we characterize the sequence (8.7) in the following form: $\{a^{2^j q},\ 0 \leq j \leq k\}$. Then, if n is prime, there is a smallest value of j (possibly $j = k$) such that $a^{2^j q} \bmod n = 1$. There are two cases to consider.

8.5 DISCRETE LOGARITHMS

Discrete logarithms are fundamental to a number of public-key algorithms, including Diffie-Hellman key exchange and the digital signature algorithm (DSA). This section provides a brief overview of discrete logarithms. For the interested reader, more detailed developments of this topic can be found in [ORE67] and [LEVE90].

The Powers of an Integer, Modulo n

Recall from Euler's theorem [Equation (8.3)] that, for every a and n that are relatively prime,

$$a^{\phi(n)} \equiv 1 \bmod n$$

where $\phi(n)$, Euler's totient function is the number of positive integers less than n and relatively prime to n. Now consider the more general expression

$$a^m \equiv 1 \bmod n \tag{8.11}$$

If a and n are relatively prime, then there is at least one integer m that satisfies Equation (8.11), namely, $m = \phi(n)$. The least positive exponent m for which Equation (8.11) holds is referred to in several ways:

- the order of a (mod n)
- the exponent to which a belongs (mod n)
- the length of the period generated by a

To see this last point, consider the powers of 7, modulo 19:

$$\begin{aligned}
7^1 &= & 7 \bmod 19\\
7^2 &= 49 = 2 \times 19 + 11 = & 11 \bmod 19\\
7^3 &= 343 = 18 \times 19 + 1 = & 1 \bmod 19\\
7^4 &= 2401 = 126 \times 19 + 7 = & 7 \bmod 19\\
7^5 &= 16807 = 884 \times 19 + 11 = & 11 \bmod 19
\end{aligned}$$

There is no point in continuing because the sequence is repeating. This can be proven by noting that $7^3 \equiv 1 \pmod{19}$ and therefore $7^{3+j} \equiv 7^3 7^j \equiv 7^j \pmod{19}$, and hence any two powers of 7 whose exponents differ by 3 (or a multiple of 3) are congruent to each other (mod 19). In other words, the sequence is periodic, and the length of the period is the smallest positive exponent m such that $7^m \equiv 1 \pmod{19}$.

Table 8.3 shows all the powers of a, modulo 19 for all positive $a < 19$. The length of the sequence for each base value is indicated by shading. Note the following:

1. All sequences end in 1. This is consistent with the reasoning of the preceding few paragraphs.

2. The length of a sequence divides $\phi(19) = 18$. That is, an integral number of sequences occur in each row of the table.
3. Some of the sequences are of length 18. In this case, it is said that the base integer a generates (via powers) the set of nonzero integers modulo 19. Each such integer is called a primitive root of the modulus 19.

More generally, we can say that the highest possible exponent to which a number can belong (mod n) is $\phi(n)$. If a number is of this order, it is referred to as a **primitive root** of n. The importance of this notion is that if a is a primitive root of n, then its powers

$$a, a^2, \ldots, a^{\phi(n)}$$

are distinct (mod n) and are all relatively prime to n. In particular, for a prime number p, if a is a primitive root of p, then

$$a, a^2, \ldots, a^{p-1}$$

are distinct (mod p). For the prime number 19, its primitive roots are 2, 3, 10, 13, 14, and 15.

Not all integers have primitive roots. In fact, the only integers with primitive roots are those of the form 2, 4, p^α, and $2p^\alpha$, where p is any odd prime and α is a positive integer.

Indices

With ordinary positive real numbers, the logarithm function is the inverse of exponentiation. An analogous function exists for modular arithmetic.

Table 8.3 Powers of Integers, Modulo 19

a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^{10}	a^{11}	a^{12}	a^{13}	a^{14}	a^{15}	a^{16}	a^{17}	a^{18}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	16	8	13	7	14	9	18	17	15	11	3	6	12	5	10	1	
9	5	8	15	7	2	6	18	16	10	11	14	4	12	17	13	1	
16	9	7	17	11	6	5	1	4	16	7	9	17	11	6	5	1	
6	17	11	9	7	16	4	1	5	6	11	17	9	7	16	4	1	
17	4	7	5	11	9	16	1	6	17	7	4	5	11	9	16	1	
11	7	1	7	11	1	7	11	1	7	11	1	7	11	1	7	11	1
7	11	18	11	12	1	8	7	18	11	12	1	8	7	18	11	12	1
5	6	7	6	16	11	4	17	1	9	5	7	6	16	11	4	17	1
9	5	12	6	3	11	15	17	18	9	14	7	13	16	8	4	2	1
1	1	1	11	7	1	11	7	1	11	7	1	11	7	1	11	7	1
11	7	18	7	8	1	12	11	18	7	8	1	12	11	18	7	8	1
17	4	12	4	14	11	10	16	18	6	2	7	15	5	8	9	3	1
4	16	8	17	10	7	3	4	18	5	13	11	2	9	12	16	15	1
16	9	12	9	2	11	13	5	18	4	3	7	10	17	8	6	14	1
6	17	11	5	4	7	17	6	1	16	9	11	5	4	7	17	6	1
17	4	11	16	6	7	5	9	1	17	4	11	16	6	7	5	9	1
7	11	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1	18

Let us briefly review the properties of ordinary logarithms. The logarithm of a number is defined to be the power to which some positive base (except 1) must be raised in order to equal the number. That is, for base x and for a value y,

$$y = x^{\log_x(y)}$$

The properties of logarithms include the following:

$$\log_x(1) = 0$$
$$\log_x(x) = 1$$
$$\log_x(yz) = \log_x(y) + \log_x(z) \tag{8.12}$$
$$\log_x(y^r) = r \times \log_x(y) \tag{8.13}$$

Consider a primitive root a for some prime number p (the argument can be developed for nonprimes as well). Then we know that the powers of a from 1 through $(p - 1)$ produce each integer from 1 through $(p - 1)$ exactly once. We also know that any integer b can be expressed in the form

$$b \equiv r \bmod p \qquad \text{where } 0 \leq r \leq (p - 1)$$

by the definition of modular arithmetic. It follows that for any integer b and a primitive root a of prime number p, we can find a unique exponent i such that

$$b \equiv a^i \bmod p \qquad \text{where } 0 \leq i \leq (p - 1)$$

This exponent i is referred to as the index of the number b for the base a (mod p). We denote this value as $\text{ind}_{a,p}(b)$.

Note the following:

$$\text{ind}_{a,p}(1) = 0, \text{ because } a^0 \bmod p = 1 \bmod p = 1 \tag{8.14}$$
$$\text{ind}_{a,p}(a) = 1, \text{ because } a^1 \bmod p = a \tag{8.15}$$

Here is an example using a nonprime modulus, $n = 9$. Here $\phi(n) = 6$ and $a = 2$ is a primitive root. We compute the various powers of a and find

$$2^0 = 1 \qquad 2^4 = 7$$
$$2^1 = 2 \qquad 2^5 = 5 \pmod 9$$
$$2^2 = 4 \qquad 2^6 = 1$$
$$2^3 = 8$$

This gives us the following table of the numbers with given indices (mod 9) for the root $a = 2$:

Index	0	1	2	3	4	5
Number	1	2	4	8	7	5

To obtain the index of a given number, we rearrange the table to make the remainders relatively prime to 9 the primary entry:

Number	1	2	4	5	7	8
Index	0	1	2	5	4	3

Now consider

$$x = a^{\text{ind}_{a,p}(x)} \bmod p \qquad y = a^{\text{ind}_{a,p}(y)} \bmod p$$

$$xy = a^{\text{ind}_{a,p}(xy)} \bmod p$$

Using the rules of modular multiplication,

$$xy \bmod p = (x \bmod p)(y \bmod p)$$

$$a^{\text{ind}_{a,p}(xy)} \bmod p = (a^{\text{ind}_{a,p}(x)} \bmod p)(a^{\text{ind}_{a,p}(y)} \bmod p)$$

$$= (a^{\text{ind}_{a,p}(x) + \text{ind}_{a,p}(y)}) \bmod p$$

But now consider Euler's theorem, which states that, for every a and n that are relatively prime,

$$a^{\phi(n)} \equiv 1 \bmod n$$

Any positive integer z can be expressed in the form $z = q + k\phi(n)$, with $0 \leq q < \phi(n)$. Therefore, by Euler's theorem,

$$a^z \equiv a^q \bmod n \qquad \text{if } z = q \bmod \phi(n)$$

Applying this to the foregoing equality, we have

$$\text{ind}_{a,p}(xy) \equiv [\text{ind}_{a,p}(x) + \text{ind}_{a,p}(y)] \bmod \phi(p)$$

and generalizing,

$$\text{ind}_{a,p}(y^r) \equiv [r \times \text{ind}_{a,p}(y)] \bmod \phi(p)$$

This demonstrates the analogy between true logarithms and indices. For this reason, the latter are often referred to as discrete logarithms.

Keep in mind that unique discrete logarithms mod m to some base a exist only if a is a primitive root of m.

Table 8.4, which is directly derived from Table 7.6, shows the sets of discrete logarithms that can be defined for modulus 19.

Calculation of Discrete Logarithms

Consider the equation

$$y = g^x \bmod p$$

Table 8.4 Tables of Discrete Logarithms, Modulo 19

(a) Discrete logarithms to the base 2, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\text{Ind}_{2,19}(a)$	18	1	13	2	16	14	6	3	8	17	12	15	5	7	11	4	10	9

(b) Discrete logarithms to the base 3, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\text{Ind}_{3,19}(a)$	18	7	1	14	4	8	6	3	2	11	12	15	17	13	5	10	16	9

(c) Discrete logarithms to the base 10, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\text{Ind}_{10,19}(a)$	18	17	5	16	2	4	12	15	10	1	6	3	13	11	7	14	8	9

(d) Discrete logarithms to the base 13, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\text{Ind}_{13,19}(a)$	18	11	17	4	14	10	12	15	16	7	6	3	1	5	13	8	2	9

(e) Discrete logarithms to the base 14, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\text{Ind}_{14,19}(a)$	18	13	7	8	10	2	6	3	14	5	12	15	11	1	17	16	14	9

(f) Discrete logarithms to the base 15, modulo 19

a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\text{Ind}_{15,19}(a)$	18	5	11	10	8	16	12	15	4	13	6	3	7	17	1	2	12	9

Given g, x, and p, it is a straightforward matter to calculate y. At the worst, we must perform x repeated multiplications, and algorithms exist for achieving greater efficiency.

However, given y, g, and p, it is, in general, very difficult to calculate x (take the discrete logarithm). The difficulty seems to be on the same order of magnitude as that of factoring primes required for RSA. At the time of this writing, the asymptotically fastest known algorithm for taking discrete logarithms modulo a prime number is on the order of [BETH91]:

$$e^{((\ln p)^{1/3}(\ln(\ln p))^{2/3})}$$

which is not feasible for large primes.

8.6 RECOMMENDED READING AND WEB SITE

There are many basic texts on the subject of number theory that provide far more detail than most readers of this book will desire. An elementary but nevertheless useful short introduc

attack is infeasible for a particular public-key algorithm. Thus, any given algorithm, including the widely used RSA algorithm, is suspect. The history of cryptanalysis shows that a problem that seems insoluble from one perspective can be found to have a solution if looked at in an entirely different way.

Finally, there is a form of attack that is peculiar to public-key systems. This is, in essence, a probable-message attack. Suppose, for example, that a message were to be sent that consisted solely of a 56-bit DES key. An opponent could encrypt all possible keys using the public key and could decipher any message by matching the transmitted ciphertext. Thus, no matter how large the key size of the public-key scheme, the attack is reduced to a brute-force attack on a 56-bit key. This attack can be thwarted by appending some random bits to such simple messages.

9.2 THE RSA ALGORITHM

The pioneering paper by Diffie and Hellman [DIFF76b] introduced a new approach to cryptography and, in effect, challenged cryptologists to come up with a cryptographic algorithm that met the requirements for public-key systems. One of the first of the responses to the challenge was developed in 1977 by Ron Rivest, Adi Shamir, and Len Adleman at MIT and first published in 1978 [RIVE78].[4] The Rivest-Shamir-Adleman (RSA) scheme has since that time reigned supreme as the most widely accepted and implemented general-purpose approach to public-key encryption.

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and $n - 1$ for some n. A typical size for n is 1024 bits, or 309 decimal digits. We examine RSA in this section in some detail, beginning with an explanation of the algorithm. Then we examine some of the computational and cryptanalytical implications of RSA.

Description of the Algorithm

The scheme developed by Rivest, Shamir, and Adleman makes use of an expression with exponentials. Plaintext is encrypted in blocks, with each block having a binary value less than some number n. That is, the block size must be less than or equal to $\log_2(n)$; in practice, the block size is k bits, where $2^k < n \leq 2^{k+1}$. Encryption and decryption are of the following form, for some plaintext block M and ciphertext block C:

$$C = M^e \bmod n$$
$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Both sender and receiver must know the value of n. The sender knows the value of e, and only the receiver knows the value of d. Thus, this is a public-key encryption

[4] Apparently, the first workable public-key system for encryption/decryption was put forward by Clifford Cocks of Britain's CESG in 1973 [COCK73]; Cocks's method is virtually identical to RSA.

algorithm with a public key of $KU = \{e, n\}$ and a private key of $KR = \{d, n\}$. For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

1. It is possible to find values of e, d, n such that $M^{ed} = M \bmod n$ for all $M < n$.
2. It is relatively easy to calculate M^e and C^d for all values of $M < n$.
3. It is infeasible to determine d given e and n.

For now, we focus on the first requirement and consider the other questions later. We need to find a relationship of the form

$$M^{ed} = M \bmod n$$

A corollary to Euler's theorem, presented in Chapter 8 [Equation (8.6)], fits the bill: Given two prime numbers, p and q, and two integers, n and m, such that $n = pq$ and $0 < m < n$, and arbitrary integer k, the following relationship holds:

$$m^{k\phi(n)+1} = m^{k(p-1)(q-1)+1} \equiv m \bmod n$$

where $\phi(n)$ is the Euler totient function, which is the number of positive integers less than n and relatively prime to n. It is shown in Chapter 8 that for p, q prime, $\phi(pq) = (p - 1)(q - 1)$. Thus, we can achieve the desired relationship if

$$ed = k\phi(n) + 1$$

This is equivalent to saying:

$$\begin{aligned} ed &\equiv 1 \bmod \phi(n) \\ d &\equiv e^{-1} \bmod \phi(n) \end{aligned}$$

That is, e and d are multiplicative inverses mod $\phi(n)$. Note that, according to the rules of modular arithmetic, this is true only if d (and therefore e) is relatively prime to $\phi(n)$. Equivalently, $\gcd(\phi(n), d) = 1$.

We are now ready to state the RSA scheme. The ingredients are the following:

p, q, two prime numbers	(private, chosen)
$n = pq$	(public, calculated)
e, with $\gcd(\phi(n), e) = 1; 1 < e < \phi(n)$	(public, chosen)
$d \equiv e^{-1} \bmod \phi(n)$	(private, calculated)

The private key consists of $\{d, n\}$ and the public key consists of $\{e, n\}$. Suppose that user A has published its public key and that user B wishes to send the message M to A. Then B calculates $C = M^e \pmod n$ and transmits C. On receipt of this ciphertext, user A decrypts by calculating $M = C^d \pmod n$.

It is worthwhile to summarize the justification for this algorithm. We have chosen e and d such that

$$d \equiv e^{-1} \bmod \phi(n)$$

Therefore,

$$ed \equiv 1 \bmod \phi(n)$$

Therefore, ed is of the form $k\phi(n) + 1$. But by the corollary to Euler's theorem, provided in Chapter 8, given two prime numbers, p and q, and integers $n = pq$ and M with $0 < M < n$:

$$M^{k\phi(n)+1} = M^{k(p-1)(q-1)+1} \equiv M \bmod n$$

So $M^{ed} \equiv M \bmod n$. Now

$$C = M^e \bmod n$$
$$M = C^d \bmod n \equiv (M^e)^d \bmod n \equiv M^{ed} \bmod n \equiv M \bmod n$$

Figure 9.5 summarizes the RSA algorithm. An example, from [SING99], is shown in Figure 9.6. For this example, the keys were generated as follows:

1. Select two prime numbers, $p = 17$ and $q = 11$.
2. Calculate $n = pq = 17 \times 11 = 187$.
3. Calculate $\phi(n) = (p - 1)(q - 1) = 16 \times 10 = 160$.

Key Generation

Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$d \equiv e^{-1} \bmod \phi(n)$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

Encryption

Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod n$

Decryption

Ciphertext:	C
Plaintext:	$M = C^d \pmod n$

Figure 9.5 The RSA Algorithm

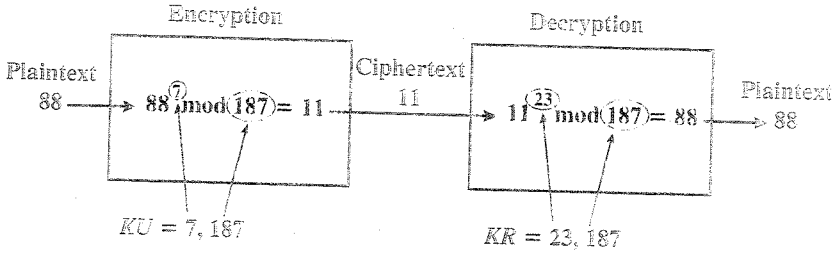


Figure 9.6 Example of RSA Algorithm

4. Select e such that e is relatively prime to $\phi(n) = 160$ and less than $\phi(n)$; we choose $e = 7$.
5. Determine d such that $de \equiv 1 \bmod 160$ and $d < 160$. The correct value is $d = 23$, because $23 \times 7 = 161 = 10 \times 160 + 1$; d can be calculated using the extended Euclid's algorithm (Chapter 4).

The resulting keys are public key $KU = \{7, 187\}$ and private key $KR = \{23, 187\}$. The example shows the use of these keys for a plaintext input of $M = 88$. For encryption, we need to calculate $C = 88^7 \bmod 187$. Exploiting the properties of modular arithmetic, we can do this is follows:

$$88^7 \bmod 187 = [(88^4 \bmod 187) \times (88^2 \bmod 187) \times (88^1 \bmod 187)] \bmod 187$$
$$88^1 \bmod 187 = 88$$
$$88^2 \bmod 187 = 7744 \bmod 187 = 77$$
$$88^4 \bmod 187 = 59{,}969{,}536 \bmod 187 = 132$$
$$88^7 \bmod 187 = (88 \times 77 \times 132) \bmod 187 = 894{,}432 \bmod 187 = 11$$

For decryption, we calculate $M = 11^{23} \bmod 187$:

$$11^{23} \bmod 187 = [(11^1 \bmod 187) \times (11^2 \bmod 187) \times (11^4 \bmod 187) \times (11^8 \bmod 187) \times (11^8 \bmod 187)] \bmod 187$$
$$11^1 \bmod 187 = 11$$
$$11^2 \bmod 187 = 121$$
$$11^4 \bmod 187 = 14{,}641 \bmod 187 = 55$$
$$11^8 \bmod 187 = 214{,}358{,}881 \bmod 187 = 33$$
$$11^{23} \bmod 187 = (11 \times 121 \times 55 \times 33 \times 33) \bmod 187 = 79{,}720{,}245 \bmod 187 = 88$$

Computational Aspects

We now turn to the issue of the complexity of the computation required to use RSA. There are actually two issues to consider: key generation and encryption/decryption. Let us look first at the process of encryption and decryption and then return to the issue of key generation.

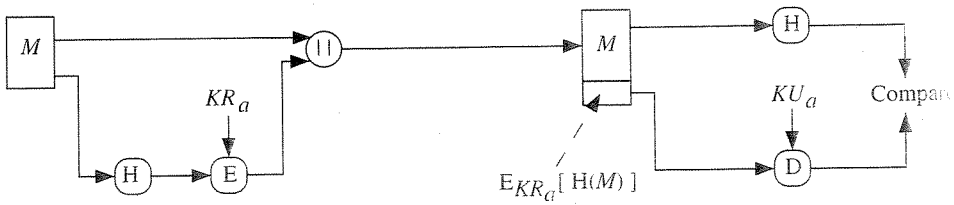
13.3 DIGITAL SIGNATURE STANDARD

The National Institute of Standards and Technology (NIST) has published Federal Information Processing Standard FIPS 186, known as the Digital Signature Standard (DSS). The DSS makes use of the Secure Hash Algorithm (SHA) described in Chapter 12 and presents a new digital signature technique, the Digital Signature Algorithm (DSA). The DSS was originally proposed in 1991 and revised in 1993 in response to public feedback concerning the security of the scheme. There was a further minor revision in 1996. In 2000, an expanded version of the standard was issued as FIP 186-2. This latest version also incorporates digital signature algorithms based on RSA and on elliptic curve cryptography. In this section, we discuss the original DSS algorithm.

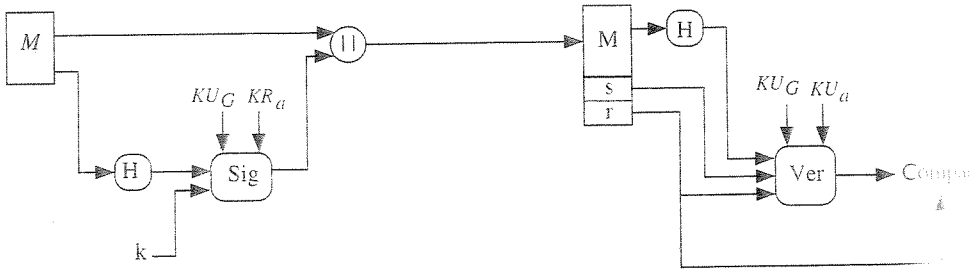
The DSS Approach

The DSS uses an algorithm that is designed to provide only the digital signature function. Unlike RSA, it cannot be used for encryption or key exchange. Nevertheless, it is a public-key technique.

Figure 13.1 contrasts the DSS approach for generating digital signatures to that used with RSA. In the RSA approach, the message to be signed is input to a hash function that produces a secure hash code of fixed length. This hash code is then encrypted using the sender's private key to form the signature. Both the message and the signature are then transmitted. The recipient takes the message and



(a) RSA approach



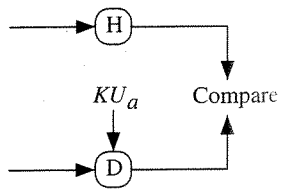
(b) DSS approach

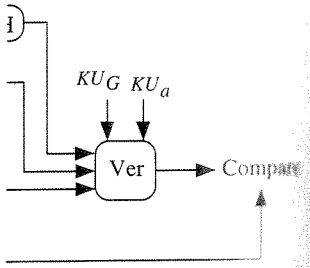
Figure 13.1 Two Approaches to Digital Signatures

ST) has published Federal
Digital Signature Standard
ithm (SHA) described in
lue, the Digital Signature
991 and revised in 1993 in
e scheme. There was a fur-
of the standard was issues
ignature algorithms based
on, we discuss the original

only the digital signature
or key exchange. Never-

ating digital signatures to
to be signed is input to a
length. This hash code is
e signature. Both the mes-
ent takes the message and





produces a hash code. The recipient also decrypts the signature using the sender's public key. If the calculated hash code matches the decrypted signature, the signature is accepted as valid. Because only the sender knows the private key, only the sender could have produced a valid signature.

The DSS approach also makes use of a hash function. The hash code is provided as input to a signature function along with a random number k generated for this particular signature. The signature function also depends on the sender's private key (KR_a) and a set of parameters known to a group of communicating principals. We can consider this set to constitute a global public key (KU_G).[4] The result is a signature consisting of two components, labeled s and r.

At the receiving end, the hash code of the incoming message is generated. This plus the signature is input to a verification function. The verification function also depends on the global public key as well as the sender's public key (KU_a), which is paired with the sender's private key. The output of the verification function is a value that is equal to the signature component r if the signature is valid. The signature function is such that only the sender, with knowledge of the private key, could have produced the valid signature.

We turn now to the details of the algorithm.

The Digital Signature Algorithm

The DSA is based on the difficulty of computing discrete logarithms (see Chapter 8) and is based on schemes originally presented by ElGamal [ELGA85] and Schnorr [SCHN91].

Figure 13.2 summarizes the algorithm. There are three parameters that are public and can be common to a group of users. A 160-bit prime number q is chosen. Next, a prime number p is selected with a length between 512 and 1024 bits such that q divides $(p - 1)$. Finally, g is chosen to be of the form $h^{(p-1)/q} \bmod p$, where h is an integer between 1 and $(p - 1)$ with the restriction that g must be greater than 1.[5]

With these numbers in hand, each user selects a private key and generates a public key. The private key x must be a number from 1 to $(q - 1)$ and should be chosen randomly or pseudorandomly. The public key is calculated from the private key as $y = g^x \bmod p$. The calculation of y given x is relatively straightforward. However, given the public key y, it is believed to be computationally infeasible to determine x, which is the discrete logarithm of y to the base g, mod p (see Chapter 8).

To create a signature, a user calculates two quantities, r and s, that are functions of the public key components (p, q, g), the user's private key (x), the hash code of the message, $H(M)$, and an additional integer k that should be generated randomly or pseudorandomly and be unique for each signing.

[4]It is also possible to allow these additional parameters to vary with each user so that they are a part of a user's public key. In practice, it is more likely that a global public key will be used that is separate from each user's public key.

[5]In number-theoretic terms, g is of order q mod p; see Chapter 8.

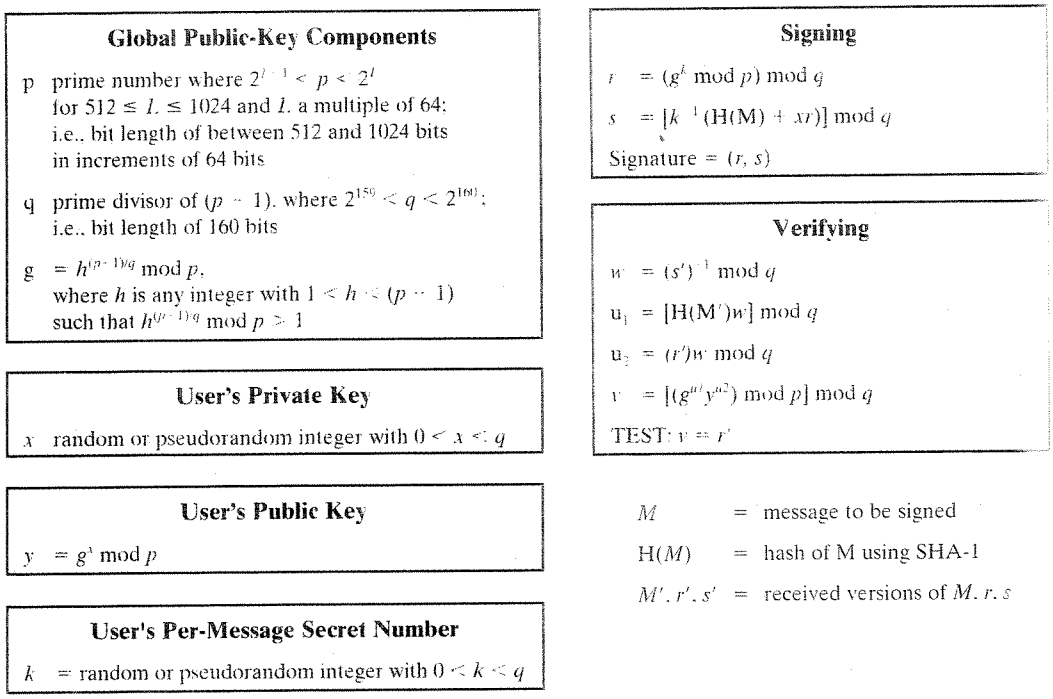


Figure 13.2 The Digital Signature Algorithm (DSA)

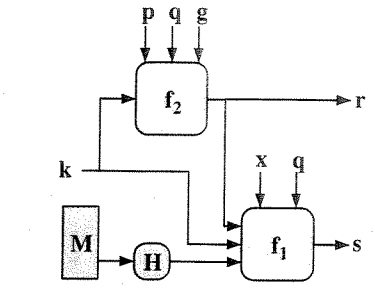
At the receiving end, verification is performed using the formulas shown in ure 13.2. The receiver generates a quantity v that is a function of the public key c ponents, the sender's public key, and the hash code of the incoming message. If quantity matches the r component of the signature, then the signature is valida

Figure 13.3 depicts the functions of signing and verifying.

The structure of the algorithm, as revealed in Figure 13.3, is quite intere Note that the test at the end is on the value r, which does not depend on the sage at all. Instead, r is a function of k and the three global public-key compon The multiplicative inverse of k (mod q) is passed to a function that also has as i the message hash code and the user's private key. The structure of this funct such that the receiver can recover r using the incoming message and signatur public key of the user, and the global public key. It is certainly not obvious fro ure 13.2 or Figure 13.3 that such a scheme would work. A proof is provided a book's Web site.

Given the difficulty of taking discrete logarithms, it is infeasible for an opp nent to recover k from r or to recover x from s.

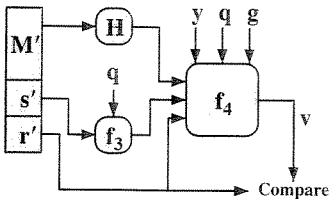
Another point worth noting is that the only computationally demandi in signature generation is the exponential calculation $g^k \bmod p$. Because thi does not depend on the message to be signed, it can be computed ahead Indeed, a user could precalculate a number of values of r to be used to sig ments as needed. The only other somewhat demanding task is the determin a multiplicative inverse, k^{-1}. Again, a number of these values can be preca



$s = f_1(H(M), k, x, r, q) = (k^{-1}(H(M) + xr)) \bmod q$

$r = f_2(k, p, q, g) = (g^k \bmod p) \bmod q$

(a) Signing



$w = f_3(s', q) = (s')^{-1} \bmod q$

$v = f_4(y, q, g, H(M'), w, r')$

$= ((g(H(M')w) \bmod q\ yr'w \bmod q) \bmod p) \bmod q$

(b) Verifying

Figure 13.3 DSS Signing and Verifying

13.4 RECOMMENDED READING

[AKL83] is the classic paper on digital signatures and is still highly relevant. A more recent, and excellent, survey is [MITC92].

AKL83 Akl, S. "Digital Signatures: A Tutorial Survey." *Computer*, February 1983.
MITC92 Mitchell, C.; Piper, F. ; and Wild, P. "Digital Signatures." In [SIMM92a].

13.5 KEY TERMS, REVIEW QUESTIONS, AND PROBLEMS

Key Terms

arbiter	digital signature algorithm (DSA)	nonce
arbitrated digital signature	digital signature standard (DSS)	one-way authentication
direct digital signature	mutual authentication	replay attack
digital signature		suppress-replay attack
		timestamp

Review Questions

13.1 List two disputes that can arise in the context of message authentication.
13.2 What are the properties a digital signature should have?
13.3 What requirements should a digital signature scheme satisfy?
13.4 What is the difference between direct and arbitrated digital signature?
13.5 In what order should the signature function and the confidentiality function be applied to a message, and why?