Here e indicates encryption and d indicates decryption. Interestingly, their roles can be interchanged, as one can use d in encryption and e for decryption and the result would be same. The following example illustrates how one can use RSA.

Just for demonstration purpose, let us suppose p is set as 11 and q as 5. Its outcome will be as follows:

1. $n = p \times q = 11 \times 5 = 55$
2. $z = (p - 1) \times (q - 1) = 10 \times 4 = 40$
3. We will take d as 7 because 7 and 40 has no common factors.
4. We need e such that (ed mod z) = 1. Thus we should have ($e \times 7$ mod 40) = 1. One such number is 23 as $23 \times 7 = 161$, which when divided by 40, generates 1 as remainder. Also, 23 is a good candidate when we calculate ($23 \times 7 - 1$ mod 40) = (161 – 1 mod 40) = (160 mod 40) = 0. So we take e as 23.
5. Suppose the sender is sending a message 'football' to other end. Assume 'a' is numbered 1, 'b' is numbered as 2, and so on till 'z' is

32 Even an RSA 1024-bit key can be cracked using hardware-based cryptanalysis. Researchers in some cases exploit the vulnerability of the application (not the algorithm) to extract the private key. Hence one should opt for larger keys and install properly patched applications.

33 One may start thinking, what we are doing here? Why we are taking these p, q, z etc? The idea of RSA is based on some mathematical concepts which are beyond the scope of this book. Next session throws some light on that.

Plaintext	Value	P^e	$C = P^e \bmod n$
f	6	789730223053602816	51
o	15	1122274146401882171630859375	20
o	15	1122274146401882171630859375	20
t	20	8388608000000000000000000000	25
b	2	8388608	8
a	1	1	1
l	12	6624737266949237011120128	23
l	12	6624737266949237011120128	23

Ciphertext	C^d	Value = $c^d \bmod 55$	Plaintext
51	897410677851	6	f
20	1280000000	15	o
20	1280000000	15	o
25	6103515625	20	t
8	2097152	2	b
1	1	1	a
23	3404825447	12	l
23	3404825447	12	l

FIGURE 10.26 Encryption and decryption using RSA