Understanding Binary

Before you can understand the details of how TCP/IP — in particular, IP — addressing works, you need to understand how the binary numbering system works because binary is the basis of IP addressing. If you already understand binary, please skip right over this section to the next main section, "Introducing IP Addresses." I don't want to bore you with stuff that's too basic.

Counting by ones

The *binary* counting system uses only two numerals: 0 and 1. In the decimal system to which most people are accustomed, you use ten numerals: 0 through 9. In an ordinary decimal number, such as 3,482, the rightmost digit represents ones; the next digit to the left, tens; the next, hundreds; the next, thousands; and so on. These digits represent powers of ten: first 10^0 (which is 1); next, 10^1 (10); then 10^2 (100); then 10^3 (1,000); and so on.

In binary, you have only two numerals rather than ten, which is why binary numbers look somewhat monotonous, as in 110011, 101111, and 100001.

The positions in a binary number (called *bits* rather than *digits*) represent powers of two rather than powers of ten — working from right to left, each bit represents the decimal values 1, 2, 4, 8, 16, 32, and so on. To figure the decimal value of a binary number, you multiply each bit by its corresponding power of two and then add the results. The decimal value of binary 10111, for example, is calculated as follows:

```
1 × 2⁰ = 1 × 1 = 1
+ 1 × 2¹ = 1 × 2 = 2
+ 1 × 2² = 1 × 4 = 4
+ 0 × 2³ = 0 × 8 = 0
+ 1 × 2⁴ = 1 × 16 = 16
         23
```

Fortunately, a computer is good at converting a number between binary and decimal — so good, in fact, that you're unlikely ever to need to do any conversions yourself. The point of knowing binary isn't to be able to look at a number, such as 1110110110110, and say instantly, "Ah! Decimal 7,606!" (If you could do that, you would probably be interviewed on the *Today* show, and they would even make a movie about you.)