Module IV

Network Layer: Address Mapping, Error Reporting, and Multicasting

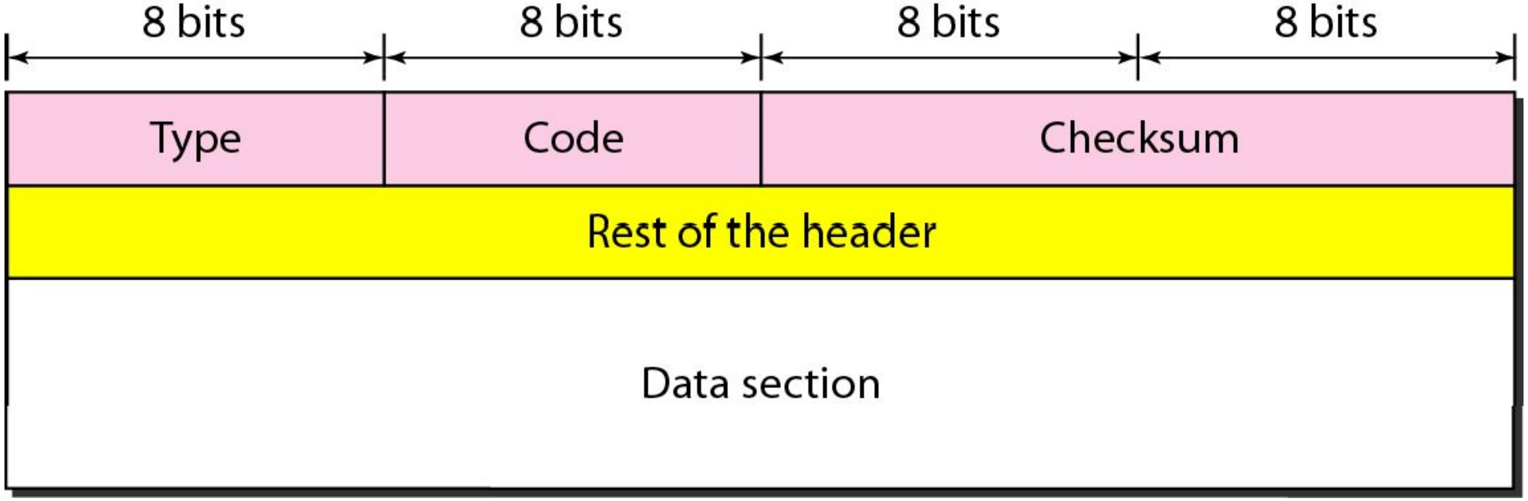
21-2 ICMP

The IP protocol has no error-reporting or error-correcting mechanism. The IP protocol also lacks a mechanism for host and management queries.. The Internet Control Message Protocol (ICMP) has been designed to compensate for the above two deficiencies.. It is a companion to the IP protocol.

Topics discussed in this section:

Types of Messages
Message Format
Error Reporting and Query
Debugging Tools

Figure 21.8 *General format of ICMP*



8 bits | 8 bits | 8 bits | 8 bits

Type | Code | Checksum

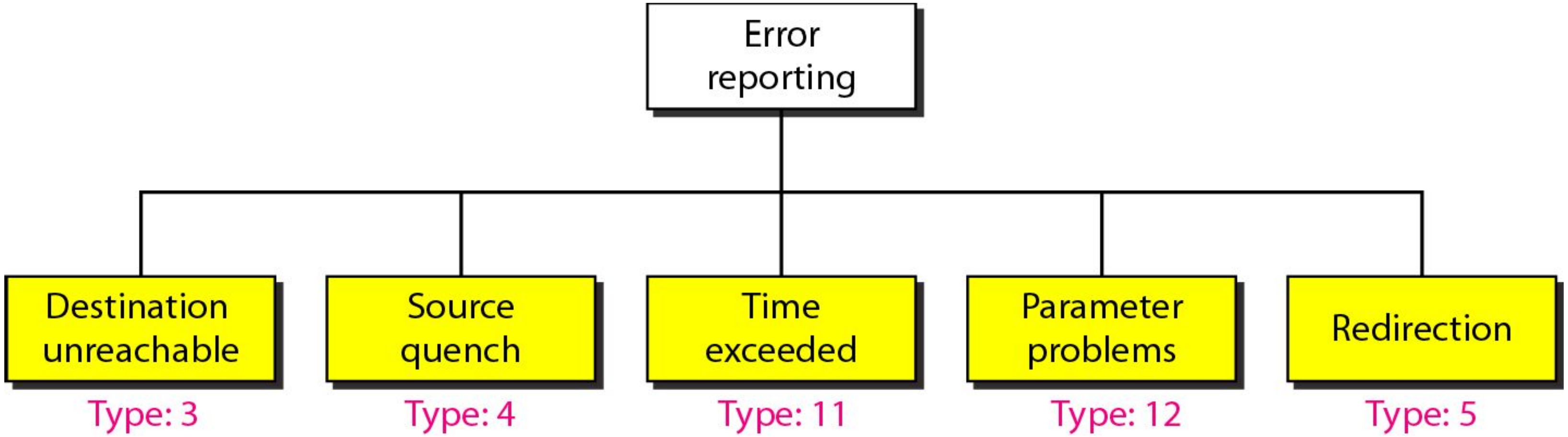
Rest of the header

Data section

Note

ICMP always reports error messages to the original source.

Figure 21.9 *Error-reporting messages*



Note

Important points about ICMP error messages:

- ❑ No ICMP error message will be generated in response to a datagram carrying an ICMP error message.
- ❑ No ICMP error message will be generated for a fragmented datagram that is not the first fragment.
- ❑ No ICMP error message will be generated for a datagram having a multicast address.
- ❑ No ICMP error message will be generated for a datagram having a special address such as 127.0.0.0 or 0.0.0.0.

Figure 21.10 *Contents of data field for the error messages*

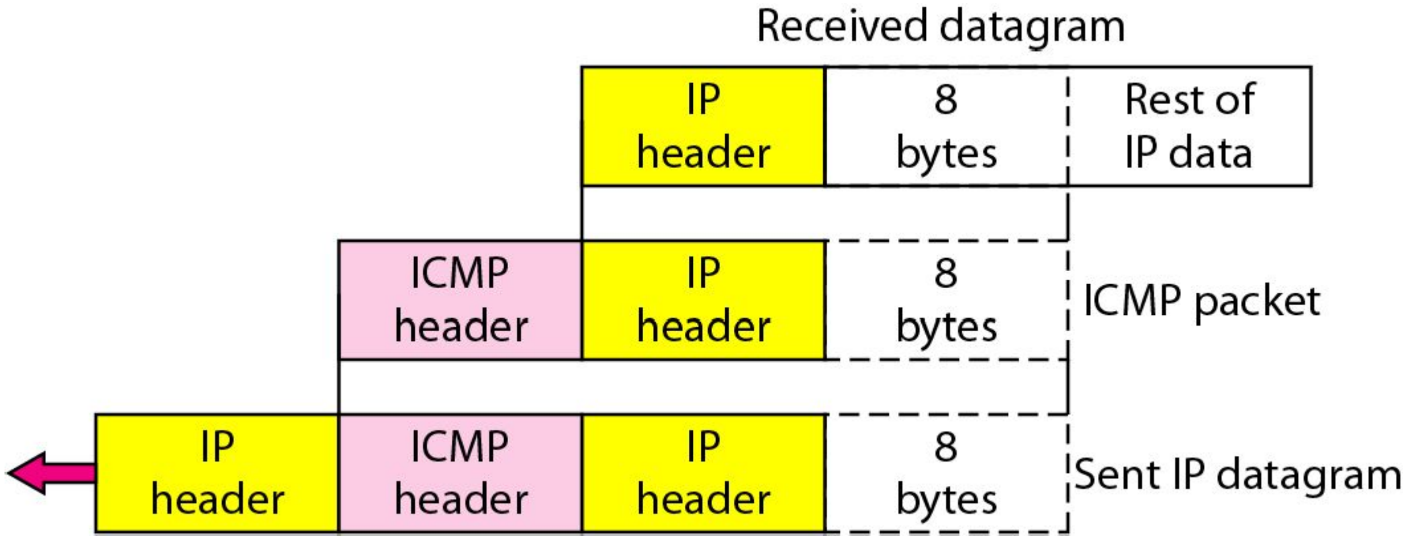


Figure 21.11 *Redirection concept*

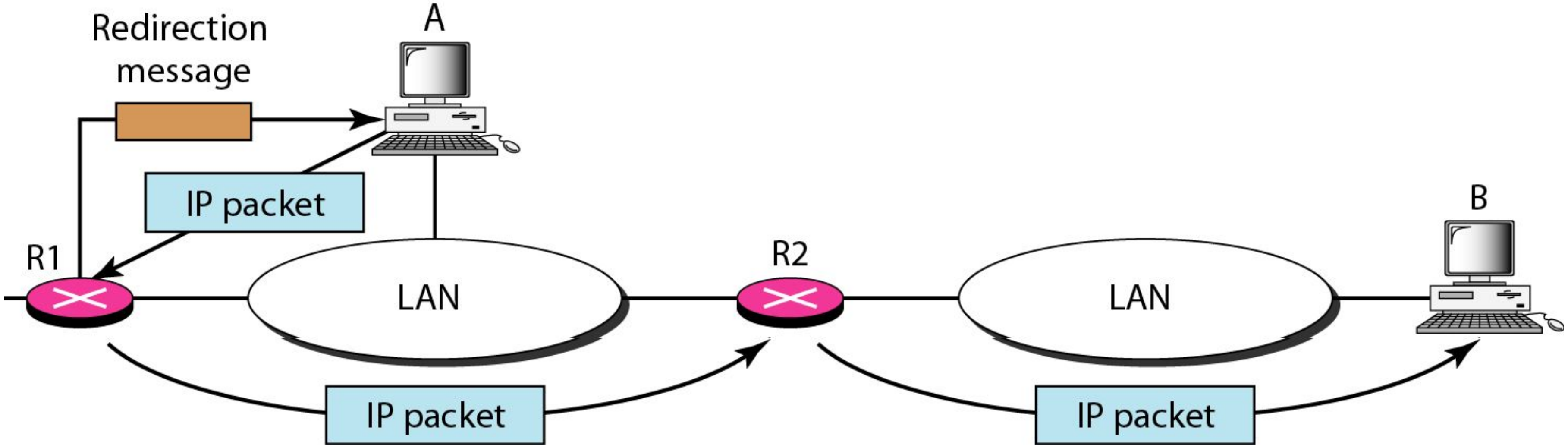


Figure 21.12 *Query messages*

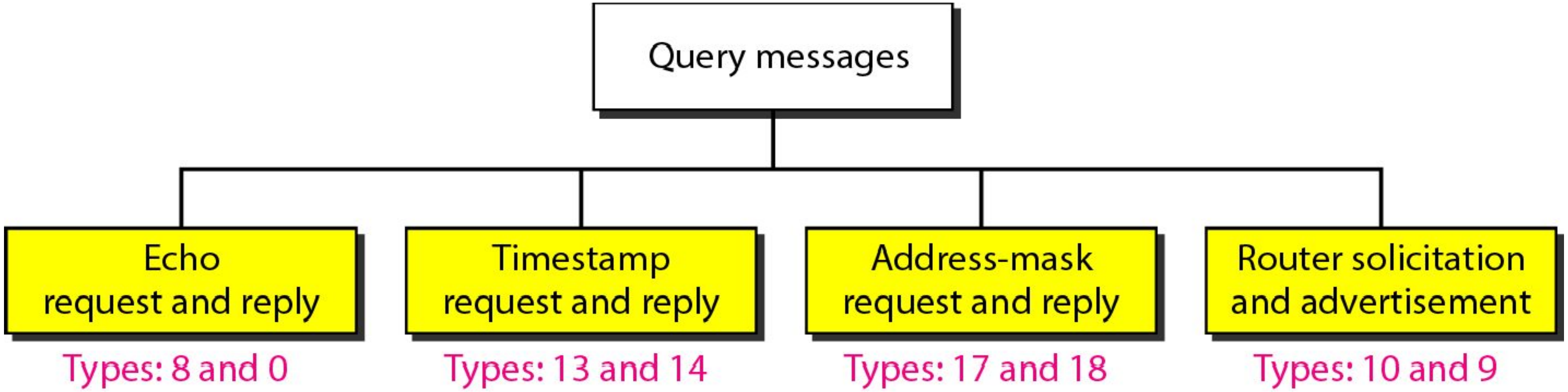
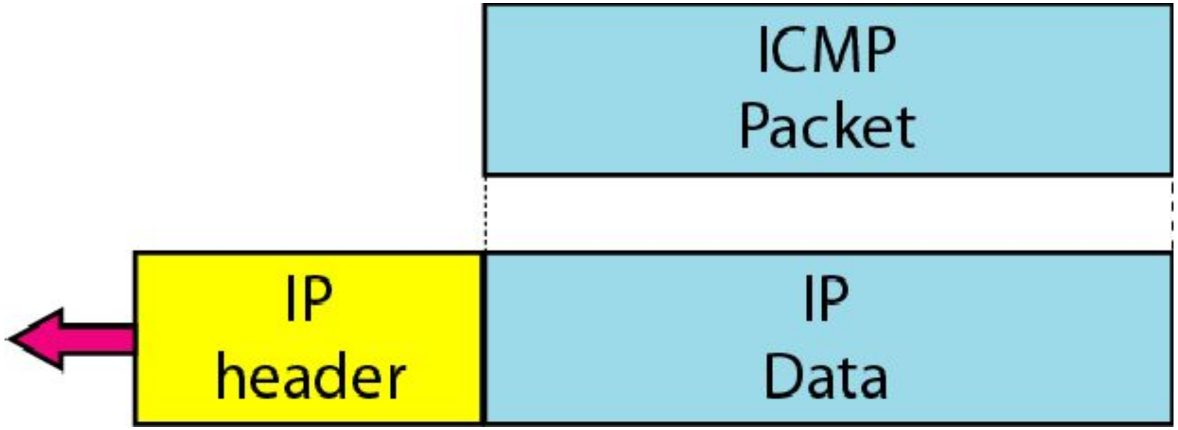


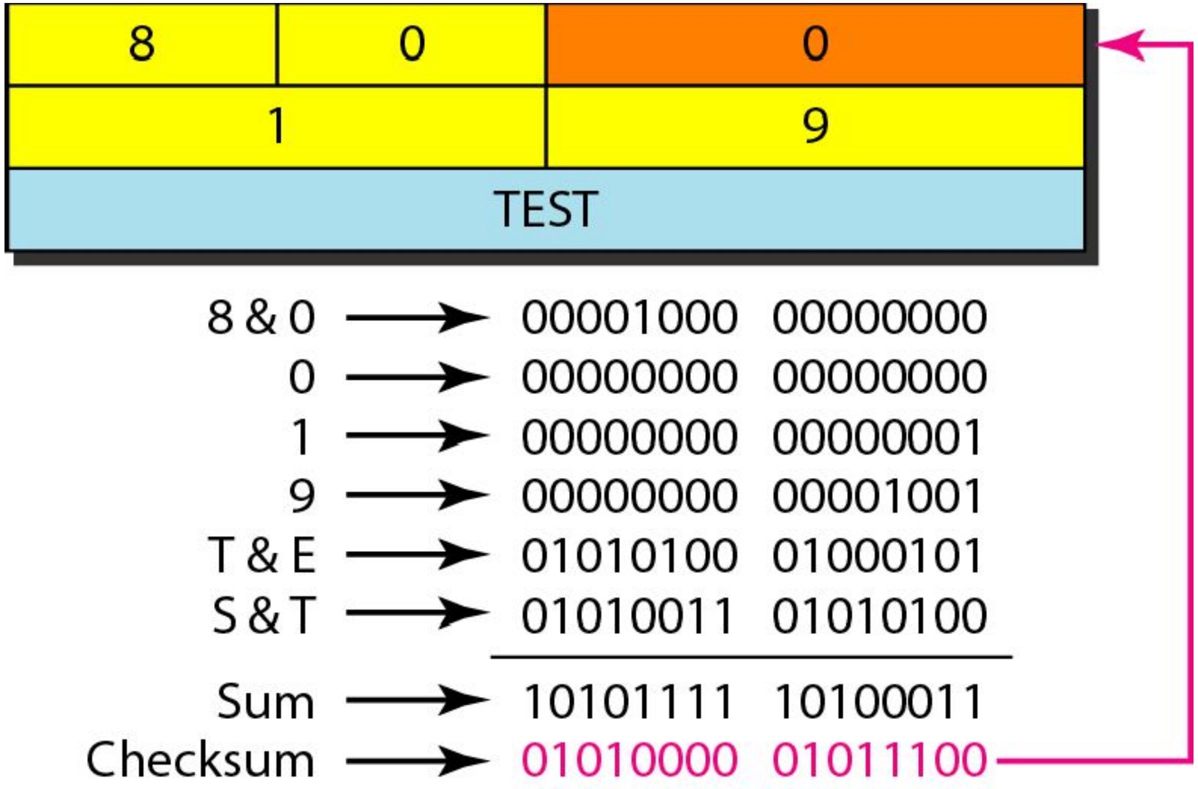
Figure 21.13 *Encapsulation of ICMP query messages*



Example 21.2

Figure 21.14 shows an example of checksum calculation for a simple echo-request message. We randomly chose the identifier to be 1 and the sequence number to be 9. The message is divided into 16-bit (2-byte) words. The words are added and the sum is complemented. Now the sender can put this value in the checksum field.

Figure 21.14 *Example of checksum calculation*

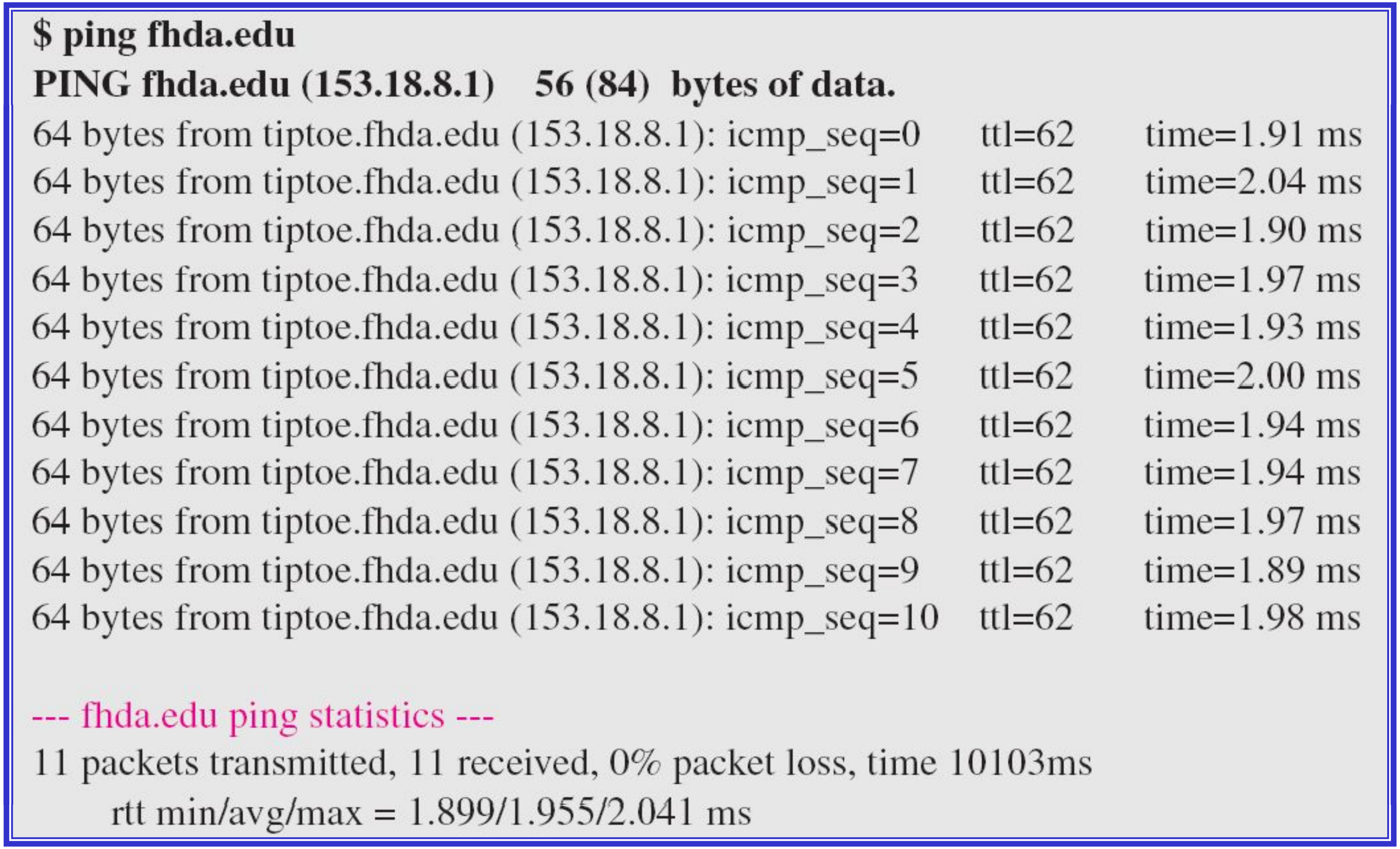


8 & 0	00001000	00000000
0	00000000	00000000
1	00000000	00000001
9	00000000	00001001
T & E	01010100	01000101
S & T	01010011	01010100
Sum	10101111	10100011
Checksum	01010000	01011100

Example 21.3

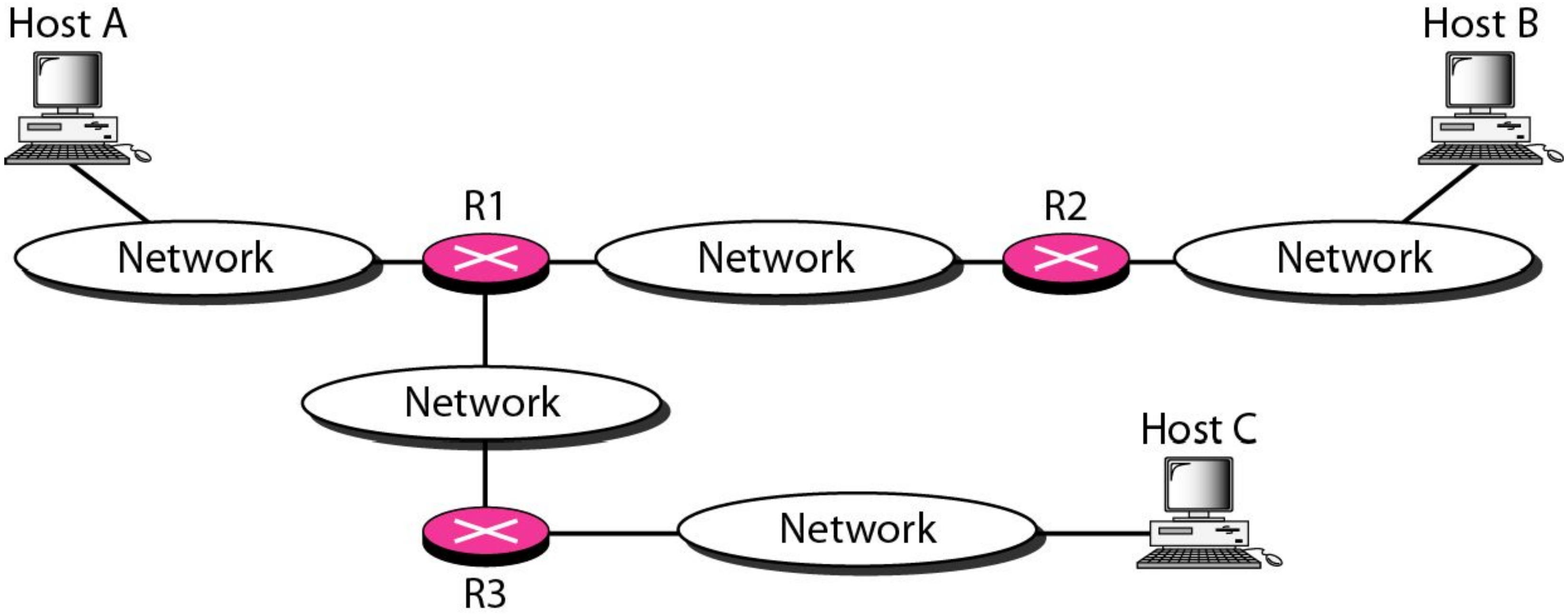
We use the ping program to test the server fhda.edu. The result is shown on the next slide. The ping program sends messages with sequence numbers starting from 0. For each probe it gives us the RTT time. The TTL (time to live) field in the IP datagram that encapsulates an ICMP message has been set to 62. At the beginning, ping defines the number of data bytes as 56 and the total number of bytes as 84. It is obvious that if we add 8 bytes of ICMP header and 20 bytes of IP header to 56, the result is 84. However, note that in each probe ping defines the number of bytes as 64. This is the total number of bytes in

Example 21.3 (continued)



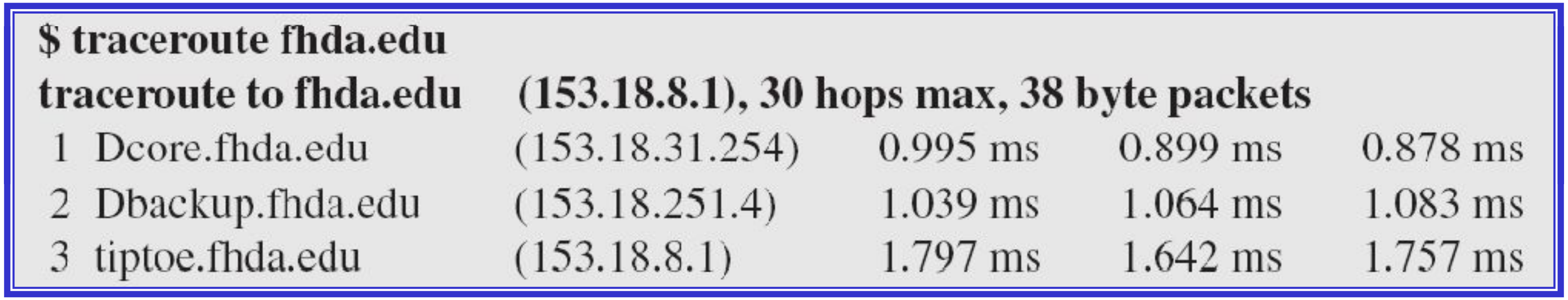
```
$ ping fhda.edu
PING fhda.edu (153.18.8.1)    56 (84)  bytes of data.
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=0     ttl=62     time=1.91 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=1     ttl=62     time=2.04 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=2     ttl=62     time=1.90 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=3     ttl=62     time=1.97 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=4     ttl=62     time=1.93 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=5     ttl=62     time=2.00 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=6     ttl=62     time=1.94 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=7     ttl=62     time=1.94 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=8     ttl=62     time=1.97 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=9     ttl=62     time=1.89 ms
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp_seq=10    ttl=62     time=1.98 ms

--- fhda.edu ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10103ms
     rtt min/avg/max = 1.899/1.955/2.041 ms
```

Figure 21.15 *The traceroute program operation*



Example 21.4

We use the traceroute program to find the route from the computer voyager.deanza.edu to the server fhda.edu. The following shows the result:



```
$ traceroute fhda.edu
traceroute to fhda.edu     (153.18.8.1), 30 hops max, 38 byte packets
 1  Dcore.fhda.edu         (153.18.31.254)    0.995 ms    0.899 ms    0.878 ms
 2  Dbackup.fhda.edu       (153.18.251.4)     1.039 ms    1.064 ms    1.083 ms
 3  tiptoe.fhda.edu        (153.18.8.1)       1.797 ms    1.642 ms    1.757 ms
```

The unnumbered line after the command shows that the destination is 153.18.8.1. The packet contains 38 bytes: 20 bytes of IP header, 8 bytes of UDP header, and 10 bytes of application data. The application data are used by traceroute to keep track of the packets.

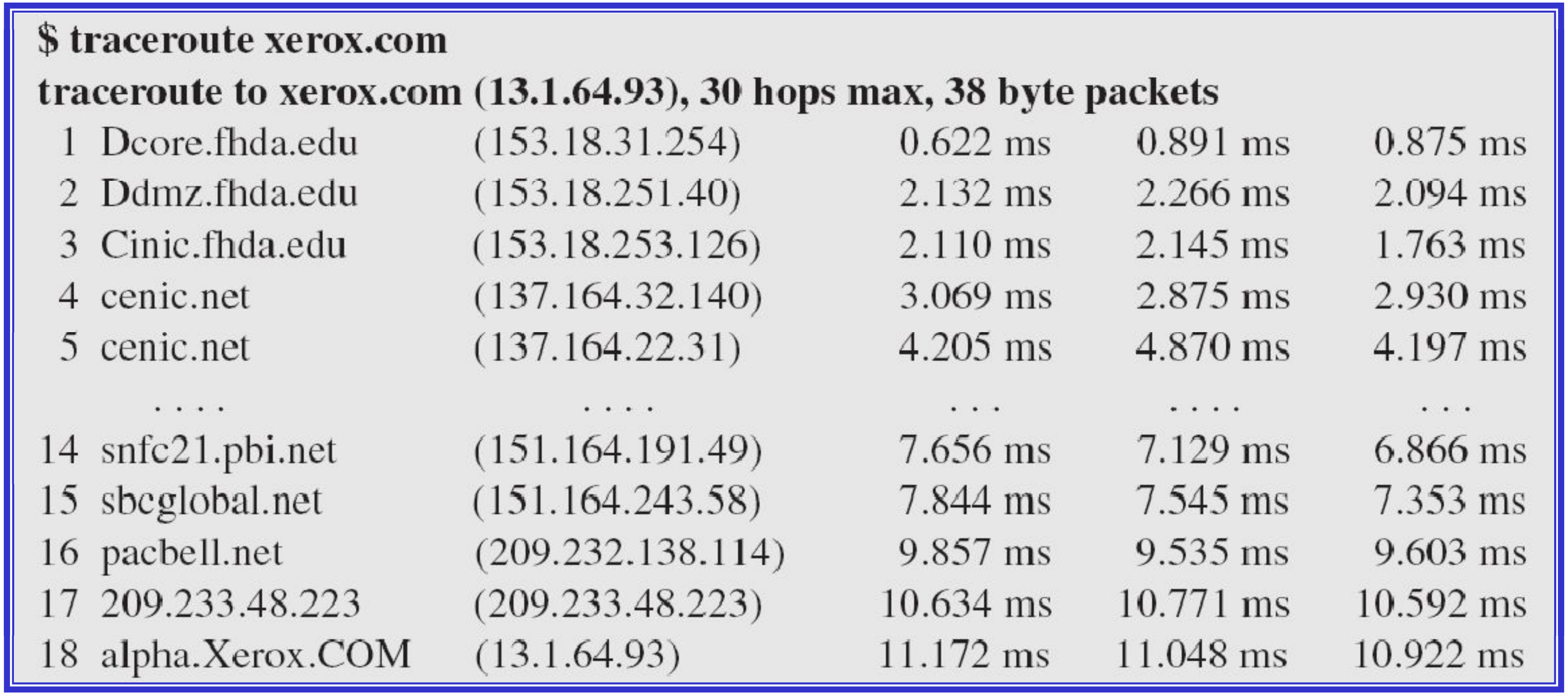
Example 21.4 (continued)

The first line shows the first router visited. The router is named Dcore fhda edu with IP address 153 18 31 254 The first round-trip time was 0.995 ms, the second was 0 899 ms, and the third was 0 878 ms The second line shows the second router visited. The router is named Dbackup fhda edu with IP address 153 18 251 4 The three round-trip times are also shown. The third line shows the destination host We know that this is the destination host because there are no more lines. The destination host is the server fhda edu, but it is named tiptoe.fhda.edu with the IP address 153.18.8.1. The three round-trip times are also shown

Example 21.5

In this example, we trace a longer route, the route to xerox.com (see next slide). Here there are 17 hops between source and destination. Note that some round-trip times look unusual. It could be that a router was too busy to process the packet immediately.

Example 21.5 (continued)



```
$ traceroute xerox.com
traceroute to xerox.com (13.1.64.93), 30 hops max, 38 byte packets
  1  Dcore.fhda.edu        (153.18.31.254)      0.622 ms    0.891 ms    0.875 ms
  2  Ddmz.fhda.edu         (153.18.251.40)      2.132 ms    2.266 ms    2.094 ms
  3  Cinic.fhda.edu        (153.18.253.126)     2.110 ms    2.145 ms    1.763 ms
  4  cenic.net             (137.164.32.140)     3.069 ms    2.875 ms    2.930 ms
  5  cenic.net             (137.164.22.31)      4.205 ms    4.870 ms    4.197 ms
         . . . .                . . . .              . . .       . . . .       . . .
 14  snfc21.pbi.net        (151.164.191.49)     7.656 ms    7.129 ms    6.866 ms
 15  sbcglobal.net         (151.164.243.58)     7.844 ms    7.545 ms    7.353 ms
 16  pacbell.net           (209.232.138.114)    9.857 ms    9.535 ms    9.603 ms
 17  209.233.48.223        (209.233.48.223)    10.634 ms   10.771 ms   10.592 ms
 18  alpha.Xerox.COM       (13.1.64.93)        11.172 ms   11.048 ms   10.922 ms
```