Hands-On Project: Using Wireshark to Capture IPv4 Pings

Time Required: 20 minutes

Objective: Capture and begin to analyze network traffic.

Description: To get a better idea of what TCP/IP packet headers look like, you can use a network traffic analyzer to capture packets as they enter or leave your network. In this activity, you capture IPv4 ping packets with Wireshark and begin to analyze how ICMP and ARP traffic function.

1. Log on to Windows 7 with an administrative account.

2. On the system, access a command prompt and enter **ipconfig** to verify the systems' IP addresses. Verify that you have connectivity by pinging the IPv4 address of your neighbor from Windows 7. If this ping is not successful, you need to troubleshoot connectivity before continuing with the project.

3. On both systems, open Wireshark. Click **Interface List** in the upper-left corner to display the interfaces detected by Wireshark. If multiple interfaces are displayed, use the one that shows the most activity in the Packets column.

4. Before you start to capture network traffic, open a command prompt and enter a ping to the IPv4 address of your partner's system, but do not press Enter yet. Arrange your desktop so that you can see both the command prompt window and the Wireshark Capture Interfaces window. You should perform the next few steps quickly so that you do not capture excess traffic. Click the Start button next to the interface with the most activity. The Capture Interfaces window closes and the Wireshark window opens. Immediately click in the command prompt window and press Enter to begin the ping. When the ping is complete, immediately click the Capture menu in Wireshark and click Stop. Each partner should perform these steps.

5. Figure 2-14 shows a typical capture of the preceding steps. Notice the entries in the Protocol column; the ARP and ICMP protocols are being used. Each line in the upper frame of the window represents one packet. ARP is used to resolve IP addresses to MAC addresses. Notice the Source and Destination columns. The first frame came from a MAC address that

ended in 17:25:16 and was sent as a broadcast to every host on the network. In the Info column, you can see that the sender of the packet wanted to know who has an IP address of 192.168.1.110. The sender wants that system to tell 192.168.1.22 its MAC address. In the second packet, you can see that the source and destination addresses are also MAC addresses, but that no broadcast was needed. In the Info column, you can see that the computer answered and provided its MAC address.

6. In the third frame, the protocol changes to ICMP, the protocol that ping uses. The Info section indicates that the frame is an Echo request, the first step of a ping. The Source column indicates that the frame came from 192.168.1.22. But why do more ARPs follow the third packet? Why doesn't the neighbor's computer simply send an Echo Reply frame? In fact, the neighbor's computer does not know the MAC address of the Windows 7 computer, and the ARP processes must be reversed so that the Windows Server 2008 computer can learn which MAC address to use to send the Echo Reply.

7. In subsequent packets, you can see that the four Echo Requests and Echo Replies are completed.