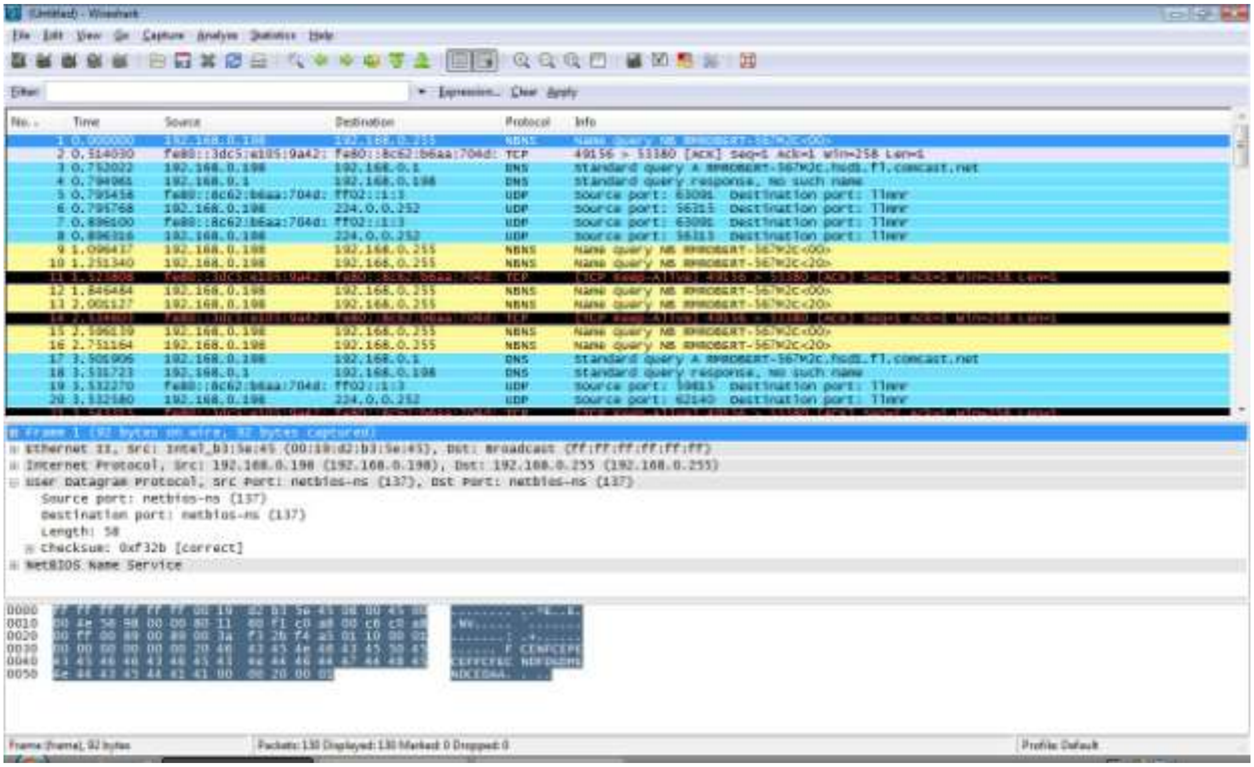
Wireshark - A $1,000.00 protocol analyzer for FREE!

Wireshark is a network packet analyzer. A network packet analyzer captures network packets and then displays details about the contents. Look at the screen capture in Figure 1. To see a sample of Wireshark as it captures a series of packets running across a network system.


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply

```
No. .  Time      Source                 Destination            Protocol  Info
  1 0.000000   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<00>
  2 0.514030   fe80::3dc5:e105:9a42:  fe80::8c62:b6aa:704d:  TCP       49156 > 53180 [ACK] Seq=1 Ack=1 Win=258 Len=1
  3 0.752022   192.168.0.198          192.168.0.1            DNS       Standard query A RMROBERT-567M2C.hsd1.fl.comcast.net
  4 0.794061   192.168.0.1            192.168.0.198          DNS       Standard query response, No such name
  5 0.795458   fe80::8c62:b6aa:704d:  ff02::1:3              UDP       Source port: 63091  Destination port: llmnr
  6 0.795768   192.168.0.198          224.0.0.252            UDP       Source port: 56315  Destination port: llmnr
  7 0.896100   fe80::8c62:b6aa:704d:  ff02::1:3              UDP       Source port: 63091  Destination port: llmnr
  8 0.896316   192.168.0.198          224.0.0.252            UDP       Source port: 56315  Destination port: llmnr
  9 1.096437   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<00>
 10 1.251340   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<20>
 11 1.323808   fe80::3dc5:e105:9a42:  fe80::8c62:b6aa:704d:  TCP       [TCP Keep-Alive] 49156 > 53180 [ACK] Seq=1 Ack=1 Win=258 Len=1
 12 1.846484   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<00>
 13 2.001127   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<20>
 14 2.[illegible] fe80::3dc5:e105:9a42:  fe80::8c62:b6aa:704d:  TCP       [TCP Keep-Alive] 49156 > 53180 [ACK] Seq=1 Ack=1 Win=258 Len=1
 15 2.596139   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<00>
 16 2.751164   192.168.0.198          192.168.0.255          NBNS      Name query NB RMROBERT-567M2C<20>
 17 3.505906   192.168.0.198          192.168.0.1            DNS       Standard query A RMROBERT-567M2C.hsd1.fl.comcast.net
 18 3.531723   192.168.0.1            192.168.0.198          DNS       Standard query response, No such name
 19 3.532270   fe80::8c62:b6aa:704d:  ff02::1:3              UDP       Source port: 59815  Destination port: llmnr
 20 3.532580   192.168.0.198          224.0.0.252            UDP       Source port: 62140  Destination port: llmnr
```

```
⊞ Frame 1 (92 bytes on wire, 92 bytes captured)
⊞ Ethernet II, Src: Intel_b3:5e:45 (00:19:d2:b3:5e:45), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊞ Internet Protocol, Src: 192.168.0.198 (192.168.0.198), Dst: 192.168.0.255 (192.168.0.255)
⊟ User Datagram Protocol, Src Port: netbios-ns (137), Dst Port: netbios-ns (137)
    Source port: netbios-ns (137)
    Destination port: netbios-ns (137)
    Length: 58
  ⊞ Checksum: 0xf32b [correct]
⊞ NetBIOS Name Service
```

```
0000  ff ff ff ff ff ff 00 19  d2 b3 5e 45 08 00 45 00   ..........^E..E.
0010  00 4e 58 98 00 00 80 11  00 f1 c0 a8 00 c6 c0 a8   .NX.............
0020  00 ff 00 89 00 89 00 3a  f3 2b f4 a5 01 10 00 01   .......:.+......
0030  00 00 00 00 00 00 20 46  43 45 4e 46 43 45 50 45   ...... F CENFCEPE
0040  43 45 46 46 43 46 45 43  4e 44 46 44 47 44 48 43   CEFFCFEC NDFDGDHC
0050  4e 44 43 45 44 41 41 00  00 20 00 01               NDCEDAA. . .
```

Figure 1 - Wireshark network packets capture displayed.

For a copy of Wireshark, simply go to the following link http://www.wireshark.org/ and download the latest version for Windows or Linux   Wireshark is free and worth over a $1000.00 dollars when you compare it to other protocol analyzers on the market today. Wireshark is free because it was developed originally by Gerald Combs while he was a graduate student at the University of Missouri-Kansas City. Originally named Ethereal it was released to the public under the GNU General Public License as open source software, free to copy and distribute. Many people have contributed to the software program and in 2006 Ethereal was renamed Wireshark. It is still open source and most likely one of the most valuable open source software programs available today.

The Wireshark protocol analyzer works just like the original Ethernet protocol analyzer with some enhancements.

After downloading installing Wireshark, you will need to identify which network adapter you will be using for your captures. You simply select “Capture “from the menu bar and then select “interfaces” to select the network adapter you will be using for the captures.

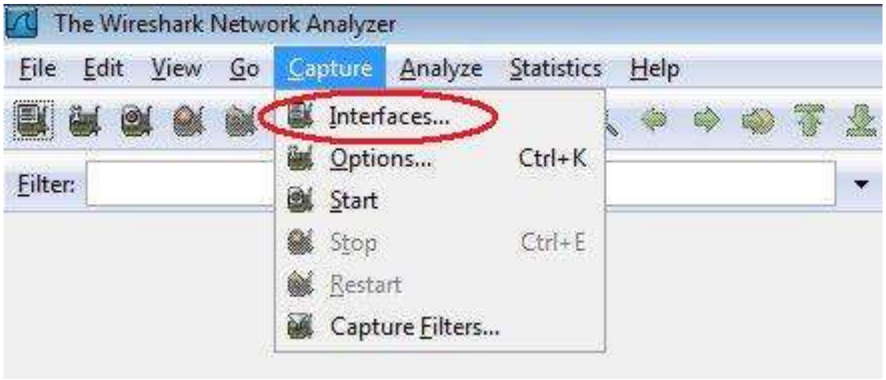
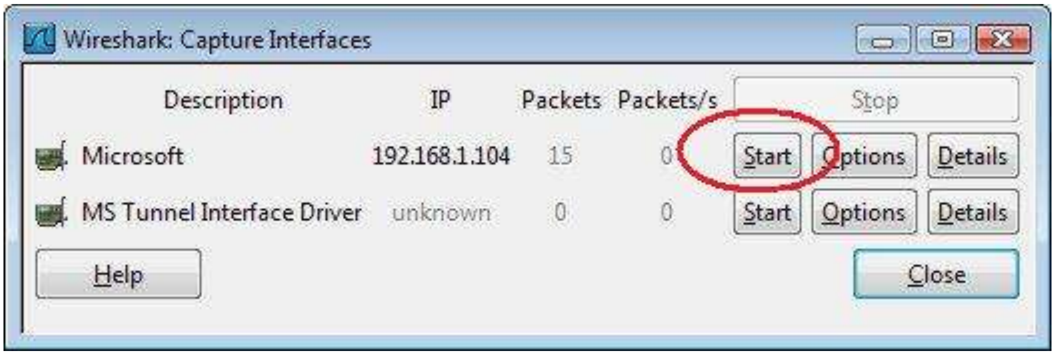


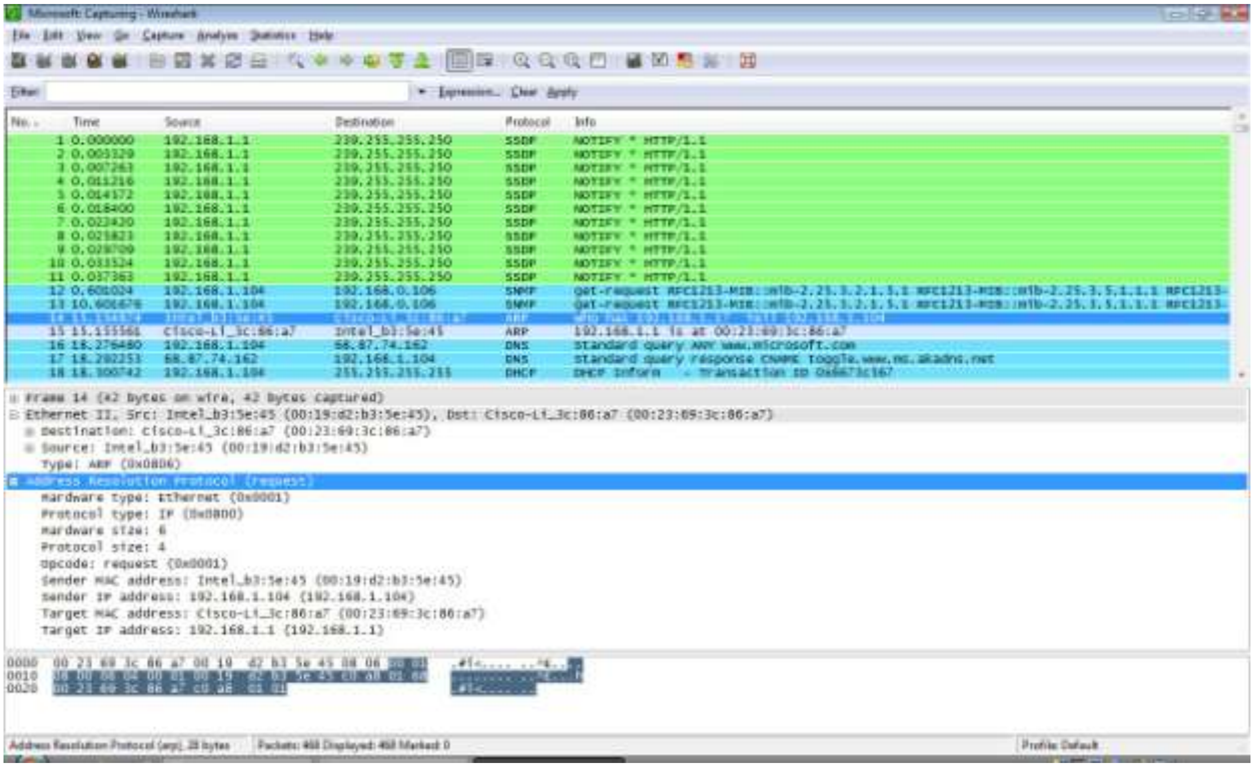
Figure 2 - The Capture menu list, select Interfaces.



Description	IP	Packets	Packets/s
Microsoft	192.168.1.104	15	0
MS Tunnel Interface Driver	unknown	0	0

Figure 3 - Select the Start button associated with the desired network adapter.

Now Wireshark should be running and capturing a series of packets also known as frames. You will see the capture of packets displayed in real time as in the screen capture below in Figure 4.


Microsoft: Capturing - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression... Clear Apply

```
No. .   Time         Source                Destination         Protocol  Info
      1 0.000000     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      2 0.003329     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      3 0.007263     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      4 0.011216     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      5 0.014172     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      6 0.018400     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      7 0.022420     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      8 0.025823     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
      9 0.029709     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
     10 0.033524     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
     11 0.037363     192.168.1.1           239.255.255.250     SSDP      NOTIFY * HTTP/1.1
     12 0.601024     192.168.1.104         192.168.0.106       SNMP      get-request RFC1213-MIB::mib-2.25.3.2.1.5.1 RFC1213-MIB::mib-2.25.3.5.1.1.1 RFC1213-
     13 10.601676    192.168.1.104         192.168.0.106       SNMP      get-request RFC1213-MIB::mib-2.25.3.2.1.5.1 RFC1213-MIB::mib-2.25.3.5.1.1.1 RFC1213-
     14 [illegible]  Intel_b3:5e:45        Cisco-Li_3c:86:a7   ARP       Who has 192.168.1.1?  Tell 192.168.1.104
     15 15.155561    Cisco-Li_3c:86:a7     Intel_b3:5e:45      ARP       192.168.1.1 is at 00:23:69:3c:86:a7
     16 18.276480    192.168.1.104         68.87.74.162        DNS       Standard query ANY www.microsoft.com
     17 18.292253    68.87.74.162          192.168.1.104       DNS       Standard query response CNAME toggle.www.ms.akadns.net
     18 18.300742    192.168.1.104         255.255.255.255     DHCP      DHCP Inform   - Transaction ID 0x6673c167
```

```
⊞ Frame 14 (42 bytes on wire, 42 bytes captured)
⊟ Ethernet II, Src: Intel_b3:5e:45 (00:19:d2:b3:5e:45), Dst: Cisco-Li_3c:86:a7 (00:23:69:3c:86:a7)
  ⊞ Destination: Cisco-Li_3c:86:a7 (00:23:69:3c:86:a7)
  ⊞ Source: Intel_b3:5e:45 (00:19:d2:b3:5e:45)
    Type: ARP (0x0806)
⊟ Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    Sender MAC address: Intel_b3:5e:45 (00:19:d2:b3:5e:45)
    Sender IP address: 192.168.1.104 (192.168.1.104)
    Target MAC address: Cisco-Li_3c:86:a7 (00:23:69:3c:86:a7)
    Target IP address: 192.168.1.1 (192.168.1.1)
```

```
0000  00 23 69 3c 86 a7 00 19  d2 b3 5e 45 08 06 00 01   .#i<.... ..^E....
0010  08 00 06 04 00 01 00 19  d2 b3 5e 45 c0 a8 01 68   ........ ..^E...h
0020  00 23 69 3c 86 a7 c0 a8  01 01                     .#i<.... ..
```

Address Resolution Protocol (arp), 28 bytes
Packets: 468 Displayed: 468 Marked: 0
Profile: Default

Figure 4 - Packet collection appears on screen while Wireshark is running.

There is a tremendous amount of information displayed. Looking from left to right on the top half of the display you will first see a number assigned to each packet starting with 1. The next column displays the exact amount of time from the start of Wireshark until the packet was captured. The next two columns display the source and destination IP address in IPv4 or IPv6 format, depending on which format is being used by the network operating system. In the fifth column the protocol is identified such as ARP, DNS, DNS, DHCP, etc. Wireshark can identify approximately 700 different protocols. (More than you will ever normally need.) The last column on the right will provide a brief bit of information about each packet captured. Some examples of the information will be things such as a request to logon, who is at a specific IP address, DHCP information, response to another packets and much, much more. For detailed information about a packet, you simply select the packet with the mouse and the bottom half of the display presents you with detailed information and a list of protocols used to encapsulate each packet. In the sample in Figure 4 you can see that frame 14 is an Ethernet type II packet encapsulating an Address Resolution Protocol (ARP) request to match source and destination IP addresses and MAC addresses. This is a very common protocol encountered on Ethernet networks.

You can use Wireshark to study each of the various protocols as you encounter them in your training course. You can either make your own captures or download sample of protocols that are available at the Wireshark organization website. You can also use

Wireshark to conduct experiments such as watching the sequence of packets and protocols during the initial startup of a computer station on a network. You can see how it identifies itself to other devices on the network and how it requests an IP address from the DHCP server or router.

At first Wireshark as any other protocol analyzer will be very intimidating but the more you use it the more comfortable you will be.

Be sure to download a copy of the Wireshark manual available at the Wireshark website.

Look over the summary of features such as filtering which allows you to capture only packets as related to a particular protocol or a particular computer or select number of computers as identified by IP address. By using a filter you can limit the collection of packets to a single or pair of computers on a network that contains numerous computers making it very difficult to identify only those packets associated with a particular computer.

A protocol analyzer is an extremely useful tool when it comes to network security. It will allow you to study how security protocols function and can also be used to identify the source of an attack on a network system.

A User Guide for Wireshark is available at the Wireshark website.

http://www.wireshark.org/docs/wsug_html_chunked/

Based on Wireshark 1.0.7 version.

If you are to become a true networking professional, you need to master how to use a protocol analyzer.