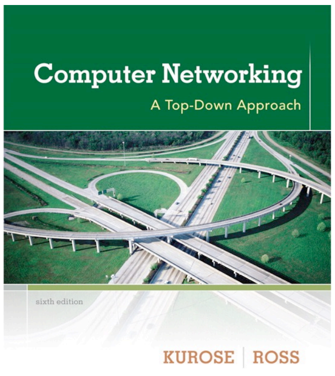
COMP 9121 Week 6
Wireshark Experiment on Ethernet Protocol and ARP

Supplement to *Computer Networking: A Top-Down Approach, 6th ed.,* J.F. Kurose and K.W. Ross

http://www-net.cs.umass.edu/wireshark-labs/



In this lab, we'll investigate the Ethernet protocol and the ARP protocol by Wireshark, a Free and open source packet analyzer. Before the lab, please read the brief introduction of Wireshark at https://en.wikipedia.org/wiki/Wireshark.

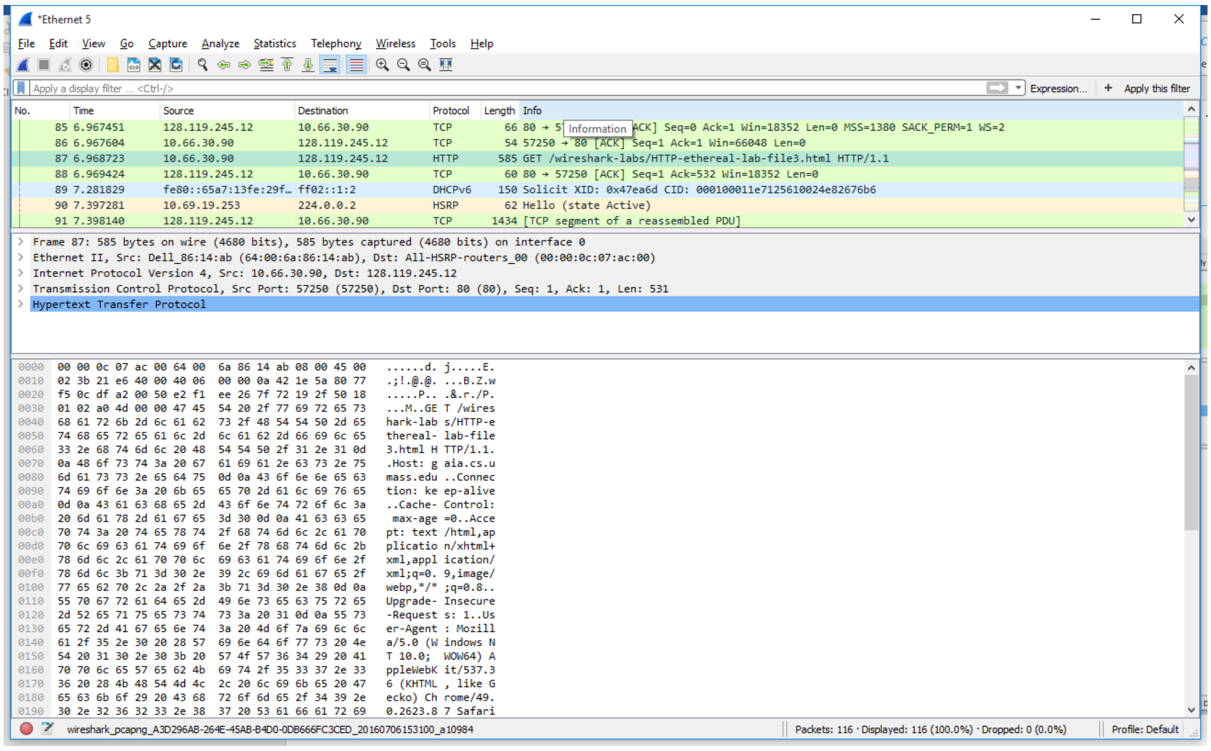
Let's begin by capturing a set of Ethernet frames to study.

First, make sure your browser's cache is empty. To do this under Mozilla Firefox, select *Tools->Clear Recent History* and check the box for Cache. For Internet Explorer, select *Tools->Internet Options->Delete Files.* Start up the Wireshark packet sniffer

Enter the following URL into your browser http://gaia.cs.umass.edu/wireshark-labs/HTTP-ethereal-lab-file3.html Your browser should display the rather lengthy US Bill of Rights.

Stop Wireshark packet capture. First, find the packet numbers (the leftmost column in the upper Wireshark window) of the HTTP GET message that was sent from your computer to gaia.cs.umass.edu, as well as the beginning of the HTTP response message sent to your computer by gaia.cs.umass.edu. You should see a screen that looks something like this (where packet 87 in the screen shot below contains the HTTP GET message).

If you are unable to run Wireshark live on a computer, you can download trace file lab-trace-1 in the Canvas.


*Ethernet 5
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+ Apply this filter
Information

```
No.  Time      Source                 Destination      Protocol Length Info
  85 6.967451  128.119.245.12         10.66.30.90      TCP        66 80 → 5  ACK] Seq=0 Ack=1 Win=18352 Len=0 MSS=1380 SACK_PERM=1 WS=2
  86 6.967604  10.66.30.90            128.119.245.12   TCP        54 57250 → 80 [ACK] Seq=1 Ack=1 Win=66048 Len=0
  87 6.968723  10.66.30.90            128.119.245.12   HTTP      585 GET /wireshark-labs/HTTP-ethereal-lab-file3.html HTTP/1.1
  88 6.969424  128.119.245.12         10.66.30.90      TCP        60 80 → 57250 [ACK] Seq=1 Ack=532 Win=18352 Len=0
  89 7.281829  fe80::65a7:13fe:29f…   ff02::1:2        DHCPv6    150 Solicit XID: 0x47ea6d CID: 000100011e7125610024e82676b6
  90 7.397281  10.69.19.253           224.0.0.2        HSRP       62 Hello (state Active)
  91 7.398140  128.119.245.12         10.66.30.90      TCP      1434 [TCP segment of a reassembled PDU]
```

```
> Frame 87: 585 bytes on wire (4680 bits), 585 bytes captured (4680 bits) on interface 0
> Ethernet II, Src: Dell_86:14:ab (64:00:6a:86:14:ab), Dst: All-HSRP-routers_00 (00:00:0c:07:ac:00)
> Internet Protocol Version 4, Src: 10.66.30.90, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 57250 (57250), Dst Port: 80 (80), Seq: 1, Ack: 1, Len: 531
> Hypertext Transfer Protocol
```

```
0000  00 00 0c 07 ac 00 64 00  6a 86 14 ab 08 00 45 00   ......d. j.....E.
0010  02 3b 21 e6 40 00 40 06  00 00 0a 42 1e 5a 80 77   .;!.@.@. ...B.Z.w
0020  f5 0c df a2 00 50 e2 f1  ee 26 7f 72 19 2f 50 18   .....P.. .&.r./P.
0030  01 02 a0 4d 00 00 47 45  54 20 2f 77 69 72 65 73   ...M..GE T /wires
0040  68 61 72 6b 2d 6c 61 62  73 2f 48 54 54 50 2d 65   hark-lab s/HTTP-e
0050  74 68 65 72 65 61 6c 2d  6c 61 62 2d 66 69 6c 65   thereal- lab-file
0060  33 2e 68 74 6d 6c 20 48  54 54 50 2f 31 2e 31 0d   3.html H TTP/1.1.
0070  0a 48 6f 73 74 3a 20 67  61 69 61 2e 63 73 2e 75   .Host: g aia.cs.u
0080  6d 61 73 73 2e 65 64 75  0d 0a 43 6f 6e 6e 65 63   mass.edu ..Connec
0090  74 69 6f 6e 3a 20 6b 65  65 70 2d 61 6c 69 76 65   tion: ke ep-alive
00a0  0d 0a 43 61 63 68 65 2d  43 6f 6e 74 72 6f 6c 3a   ..Cache- Control:
00b0  20 6d 61 78 2d 61 67 65  3d 30 0d 0a 41 63 63 65    max-age =0..Acce
00c0  70 74 3a 20 74 65 78 74  2f 68 74 6d 6c 2c 61 70   pt: text /html,ap
00d0  70 6c 69 63 61 74 69 6f  6e 2f 78 68 74 6d 6c 2b   plicatio n/xhtml+
00e0  78 6d 6c 2c 61 70 70 6c  69 63 61 74 69 6f 6e 2f   xml,appl ication/
00f0  78 6d 6c 3b 71 3d 30 2e  39 2c 69 6d 61 67 65 2f   xml;q=0. 9,image/
0100  77 65 62 70 2c 2a 2f 2a  3b 71 3d 30 2e 38 0d 0a   webp,*/* ;q=0.8..
0110  55 70 67 72 61 64 65 2d  49 6e 73 65 63 75 72 65   Upgrade- Insecure
0120  2d 52 65 71 75 65 73 74  73 3a 20 31 0d 0a 55 73   -Request s: 1..Us
0130  65 72 2d 41 67 65 6e 74  3a 20 4d 6f 7a 69 6c 6c   er-Agent : Mozill
0140  61 2f 35 2e 30 20 28 57  69 6e 64 6f 77 73 20 4e   a/5.0 (W indows N
0150  54 20 31 30 2e 30 3b 20  57 4f 57 36 34 29 20 41   T 10.0;  WOW64) A
0160  70 70 6c 65 57 65 62 4b  69 74 2f 35 33 37 2e 33   ppleWebK it/537.3
0170  36 20 28 4b 48 54 4d 4c  2c 20 6c 69 6b 65 20 47   6 (KHTML , like G
0180  65 63 6b 6f 29 20 43 68  72 6f 6d 65 2f 34 39 2e   ecko) Ch rome/49.
0190  30 2e 32 36 32 33 2e 38  37 20 53 61 66 61 72 69   0.2623.8 7 Safari
```

wireshark_pcapng_A3D296AB-264E-45AB-B4D0-0DB666FC3CED_20160706153100_a10984 | Packets: 116 · Displayed: 116 (100.0%) · Dropped: 0 (0.0%) | Profile: Default

In order to answer the following questions, you'll need to look into the packet details and packet contents windows (the middle and lower display windows in Wireshark).

Select the Ethernet frame containing the HTTP GET message. (HTTP GET message is carried inside of a TCP segment, which is carried inside of an IP datagram, which is carried inside of an Ethernet frame). Expand the Ethernet II information in the packet details window. Note that the contents of the Ethernet frame (header as well as payload) are displayed in the packet contents window.

Answer the following questions, based on the contents of the Ethernet frame containing the HTTP GET message.

1. What is the 48-bit Ethernet address of your computer? Src [illegible] DST [illegible]
2. What is the 48-bit destination address in the Ethernet frame? Is this the Ethernet address of gaia.cs.umass.edu? (Hint: the answer is *no*). What device has this as its Ethernet address? 1st Router
3. Give the hexadecimal value for the two-byte Frame type field. What upper layer protocol does this correspond to?
IPv4 0x0800

Go to https://aruljohn.com/mac.pl, check the Vendor of the network interface card of your computer.

ARP Caching

Recall that the ARP protocol typically maintains a cache of IP-to-Ethernet address translation pairs on your computer. The *arp* command (in both MSDOS and Linux/Unix) is used to view and manipulate the contents of this cache. Since the *arp* command and the ARP protocol have the same name, it's understandably easy to confuse them. But keep in mind that they are different - the *arp* command is used to view and manipulate the ARP cache contents, while the ARP protocol defines the format and meaning of the messages sent and received, and defines the actions taken on message transmission and receipt.

Let's take a look at the contents of the ARP cache on your computer:
- **MS-DOS.** The *arp* command is in c:\windows\system32, so type either "*arp*" or "*arp -a"* in the MS-DOS command line (without quotation marks).
- **Linux/Unix/MacOS.** The executable for the *arp* command can be in various places. Popular locations are /sbin/arp (for linux) and /usr/etc/arp (for some Unix variants).

The Windows *arp* command with no arguments will display the contents of the ARP cache on your computer. Run the *arp* command.

In order to observe your computer sending and receiving ARP messages, we'll need to clear the ARP cache, since otherwise your computer is likely to find a needed IP-Ethernet address translation pair in its cache and consequently not need to send out an ARP message.

- **MS-DOS.** The MS-DOS *arp –d* * command will clear your ARP cache. The *–d* flag indicates a deletion operation, and the * is the wildcard that says to delete all table entries.

- **Linux/Unix/MacOS.** The *arp –d* * will clear your ARP cache. In order to run this command you'll need root privileges. If you don't have root privileges and can't run Wireshark on a Windows machine, you can skip the trace collection part of this lab and just use the trace discussed in the earlier footnote.

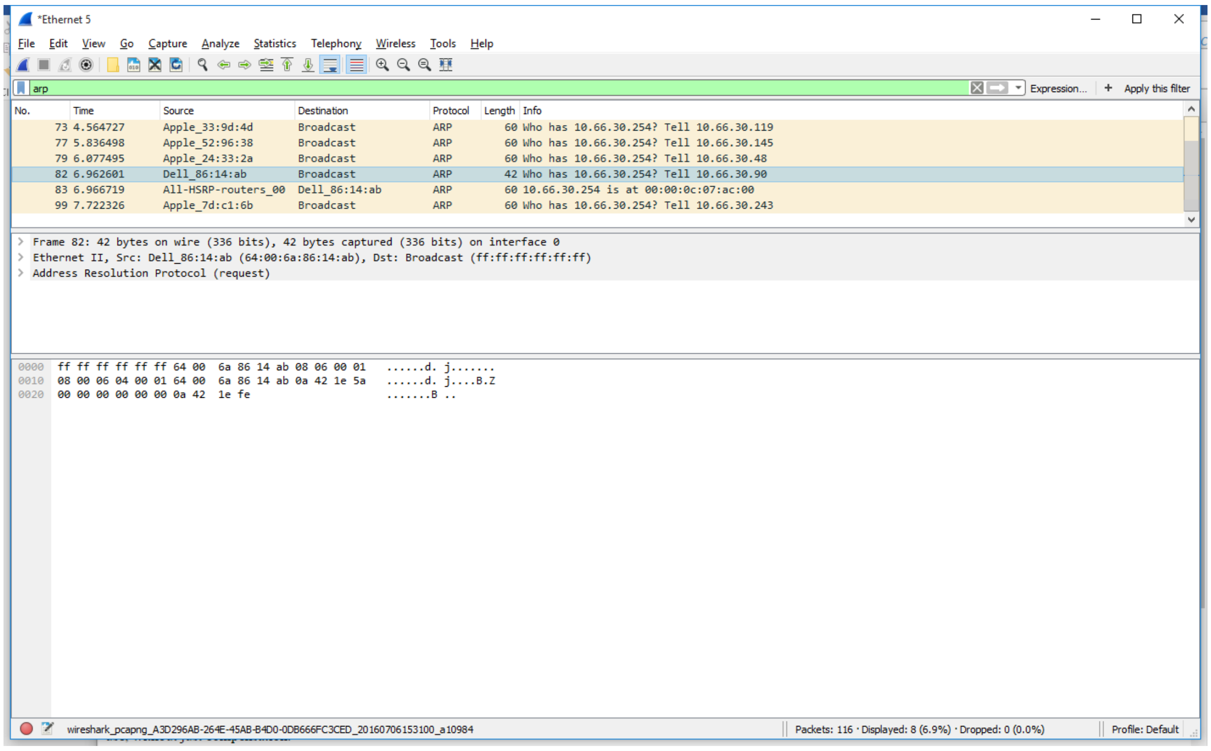
- Some versions of MacOS. *arp -a -d* will clear APR cache.

Clear your ARP cache, as described above.
- Next, make sure your browser's cache is empty.
- Start up the Wireshark packet sniffer
- Enter the following URL into your browser
http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-lab-file3.html
Your browser should again display the rather lengthy US Bill of Rights.

- Stop Wireshark packet capture.

Since this lab is about Ethernet and ARP, so let's change Wireshark's "listing of captured packets" window so that it shows information only about protocols below IP. To have Wireshark do this, input ARP at the display filter.

You should now see a Wireshark window that looks like:


*Ethernet 5
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression... + Apply this filter

arp

```
No.  Time        Source              Destination     Protocol  Length  Info
  73 4.564727    Apple_33:9d:4d      Broadcast       ARP           60  Who has 10.66.30.254? Tell 10.66.30.119
  77 5.836498    Apple_52:96:38      Broadcast       ARP           60  Who has 10.66.30.254? Tell 10.66.30.145
  79 6.077495    Apple_24:33:2a      Broadcast       ARP           60  Who has 10.66.30.254? Tell 10.66.30.48
  82 6.962601    Dell_86:14:ab       Broadcast       ARP           42  Who has 10.66.30.254? Tell 10.66.30.90
  83 6.966719    All-HSRP-routers_00 Dell_86:14:ab   ARP           60  10.66.30.254 is at 00:00:0c:07:ac:00
  99 7.722326    Apple_7d:c1:6b      Broadcast       ARP           60  Who has 10.66.30.254? Tell 10.66.30.243

> Frame 82: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
> Ethernet II, Src: Dell_86:14:ab (64:00:6a:86:14:ab), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)

0000  ff ff ff ff ff ff 64 00  6a 86 14 ab 08 06 00 01   ......d. j.......
0010  08 00 06 04 00 01 64 00  6a 86 14 ab 0a 42 1e 5a   ......d. j....B.Z
0020  00 00 00 00 00 00 0a 42  1e fe                     .......B ..
```

wireshark_pcapng_A3D296AB-264E-45AB-B4D0-0DB666FC3CED_20160706153100_a10984 | Packets: 116 · Displayed: 8 (6.9%) · Dropped: 0 (0.0%) | Profile: Default

4. What are the hexadecimal values for the source and destination addresses in the Ethernet frame containing the ARP request message? Give the hexadecimal value for the two-byte Ethernet Frame type field. What upper layer protocol does this correspond to? Does the ARP message contain the IP address of the sender?

ARP (0x0806) Yes

5. Now find the ARP reply that was sent in response to the ARP request. Where in the ARP message does the "answer" to the earlier ARP request appear – the IP address of the machine having the Ethernet address whose corresponding IP address is being queried?