Alan Palayil
ECE 407

ECE 407

Introduction to Computer Networks Laboratory

Practice 3 – PDU Information and the ARP Protocol

Objectives

The goal of this experiment is to:

1. Familiarize students with PDU information and the role of networking devices through a Cisco Packet Tracer simulation.
2. Familiarize students with the Address Resolution Protocol (ARP).

Background

Address Resolution Protocol (ARP) is used by TCP/IP to map a Layer 3 IPv4 address to a Layer 2 MAC address. ARP is a low-level network protocol, operating at Layer 2 of the OSI model. ARP usually is implemented in the device drivers of network operating systems.

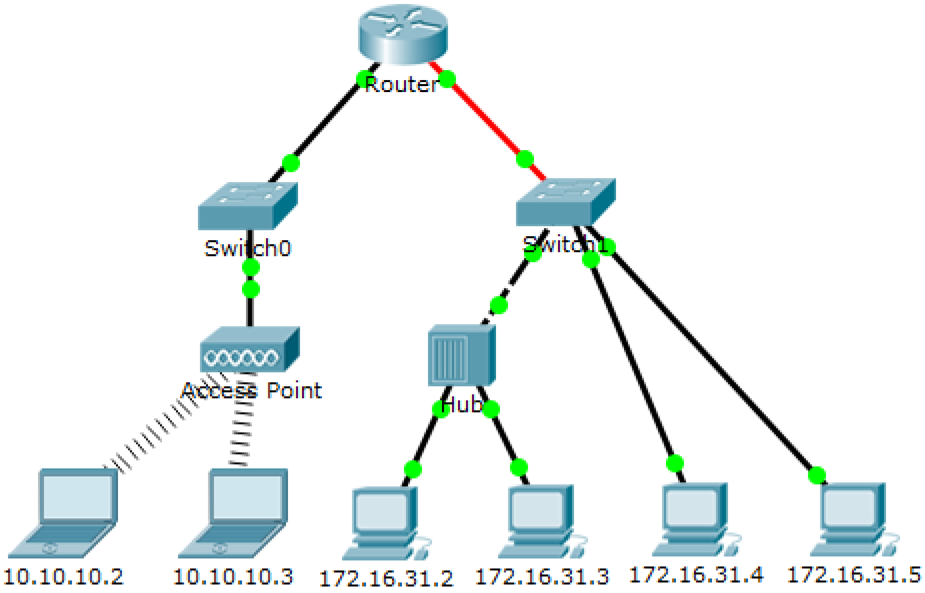
When an Ethernet frame is transmitted on the network, it must have a destination MAC address. To dynamically discover the MAC address of a known destination, the source device broadcasts an ARP request on the local network. The device that is configured with the destination IPv4 address responds to the request with an ARP reply and the MAC address is recorded in the ARP cache. Every device on the LAN maintains its own ARP cache. The ARP cache is a small area in RAM that holds the ARP responses. Viewing an ARP cache on a PC displays the IPv4 address and the MAC address of each device on the LAN with which the PC has exchanged ARP messages. An ARP cache timer removes ARP entries that have not been used for a certain period of time.

ARP is an excellent example of performance tradeoff. With no cache, ARP must continually request address translations each time a frame is placed on the network. This adds latency to the communication and could congest the LAN. Conversely, unlimited hold times could cause errors with devices that leave the network or change the Layer 3 address.

A network administrator should be aware of ARP, but may not interact with the protocol on a regular basis. ARP is a protocol that enables network devices to communicate with the TCP/IP protocol. Without ARP, there is no efficient method to build the datagram Layer 2 destination address. Also, ARP is a potential security risk. ARP spoofing, or ARP poisoning, is a technique used by an attacker to inject the wrong MAC address association in a network. An attacker forges the MAC address of a device, and frames are sent to the wrong destination. Manually configuring static ARP associations is one way to prevent ARP spoofing. Finally, an authorized MAC address list may be configured on Cisco devices to restrict network access to only approved devices.

Exercise 1: Exploring PDU information and device operation (5.3.1.3)

Topology



A. Gather PDU Information

Open the packet tracer activity file "5.3.1.3 Packet Tracer - Identify MAC and IP Addresses.pka" on the local desktop. This activity is optimized for viewing PDUs. The devices are already configured. You will gather PDU information in simulation mode and answer a series of questions about the data you collect. Review the Reflection Questions in Part B before proceeding with Part 1. It will give you an idea of the types of information you will need to gather.

Step 1: Gather PDU information as a packet travels from 172.16.31.2 to 10.10.10.3.

a. Click **172.16.31.2** and open the **Command Prompt**.

b. Enter the **ping 10.10.10.3** command.

c. Switch to simulation mode and repeat the **ping 10.10.10.3** command. A PDU appears next to **172.16.31.2**.

d. Click the PDU and note the following information from the **Outbound PDU Layer** tab:

- Destination MAC Address: 00D0:BA8E:741A
- Source MAC Address: 000C:85CC:1DA7
- Source IP Address: 172.16.31.2
- Destination IP Address: 10.10.10.3
- At Device: Computer

e. Click **Capture / Forward** to move the PDU to the next device. Gather the same information from Step 1d. Repeat this process until the PDU reaches its destination. Record the PDU information you gathered into a spreadsheet using a format like the table shown below:

Example Spreadsheet Format

Test	At Device	Dest. MAC	Src MAC	Src IPv4	Dest IPv4
Ping from 172.16.31.2 to 10.10.10.3	172.16.31.2	00D0:BA8E:741A	000C:85CC:1DA7	172.16.31.2	10.10.10.3
	Hub	--	--	--	--
	Switch1	00D0:BA8E:741A	000C:85CC:1DA7	--	--
	Router	0060:4706:572B	00D0:588C:2401	172.16.31.2	10.10.10.3
	Switch0	0060:4706:572B	00D0:588C:2401	--	--
	Access Point	--	--	--	--
	10.10.10.3	0060:4706:572B	00D0:588C:2401	172.16.31.2	10.10.10.3

Step 2: Gather additional PDU information from other pings.

Repeat the process in Step 1 and gather the information for the following tests:

- Ping 10.10.10.2 from 10.10.10.3.
- Ping 172.16.31.2 from 172.16.31.3.
- Ping 172.16.31.4 from 172.16.31.5.
- Ping 172.16.31.4 from 10.10.10.2.
- Ping 172.16.31.3 from 10.10.10.2.

B. Reflection Questions

Answer the following questions regarding the captured data:

1. Were there different types of wires used to connect devices?

   Yes there is, copper and fiber.

2. Did the wires change the handling of the PDU in any way?

   No

3. Did the **Hub** lose any of the information given to it?

   No

4. What does the **Hub** do with MAC addresses and IP addresses?

   It does nothing

5. Did the wireless **Access Point** do anything with the information given to it?

Yes, this information was repackaged as wireless 802.11

6. Was any MAC or IP address lost during the wireless transfer?

   None were lost.

7. What was the highest OSI layer that the **Hub** and **Access Point** used?

   The first Layer

8. Did the **Hub** or **Access Point** ever replicate a PDU that was rejected with a red “X”?

   Yes

9. When examining the **PDU Details** tab, which MAC address appeared first, the source or the destination?

   Destination

10. Why would the MAC addresses appear in this order?

    This is because the switch may have sent the frame to a MAC faster if it knows where to go, meaning the destination is first.

11. Was there a pattern to the MAC addressing in the simulation?

    No

12. Did the switches ever replicate a PDU that was rejected with a red “X”?

    No

13. Every time that the PDU was sent between the 10 network and the 172 network, there was a point where the MAC addresses suddenly changed. Where did that occur?

    Router

14. Which device uses MAC addresses starting with 00D0?

    Router

15. To what devices did the other MAC addresses belong?

    Sender and Receiver

16. Did the sending and receiving IPv4 addresses switch in any of the PDUs?

    No

17. If you follow the reply to a ping, sometimes called a *pong*, do the sending and receiving IPv4 addresses switch?

    Yes

18. What is the pattern to the IPv4 addressing in this simulation?

    In all the ports the router requires addresses with no overlaps.

19. Why do different IP networks need to be assigned to different ports of a router?

    This is because it intertwines different IP networks.

20. If this simulation was configured with IPv6 instead of IPv4, what would be different?

    Only the addresses would change from IPv4 to IPv6.

Exercise 2: Address Resolution Protocol (ARP) (3.4.3.5)

A. Capture and Analyze Local ARP Data in Wireshark

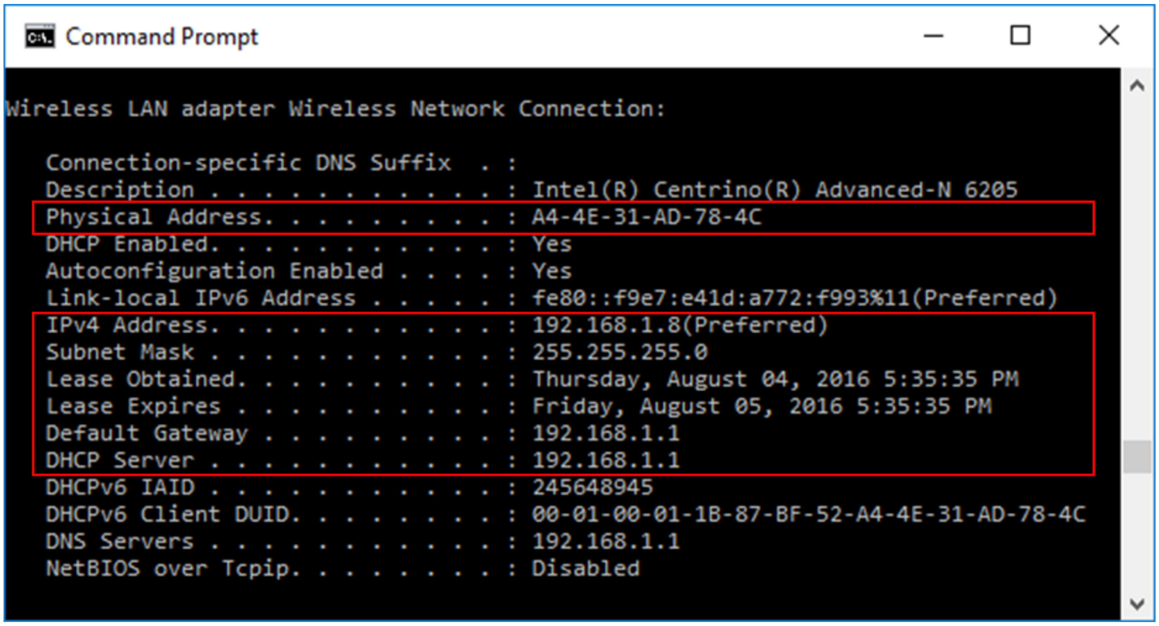
In this part of this exercise, you will ping another PC on the LAN and capture ARP requests and replies in Wireshark. You will also look inside the frames captured for specific information. This analysis should help to clarify how packet headers are used to transport data to their destination.

Step 1: Retrieve your PC's interface addresses.

For this exercise, you will need to retrieve your PC's IPv4 address and the MAC address.

a. Open a command window, type **ipconfig /all**, and then press Enter.

b. Note which network adapter that the PC is using to access the network. Record your PC interface's IPv4 address and MAC address (Physical Address).



```
Wireless LAN adapter Wireless Network Connection:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) Centrino(R) Advanced-N 6205
   Physical Address. . . . . . . . . : A4-4E-31-AD-78-4C
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::f9e7:e41d:a772:f993%11(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.1.8(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Thursday, August 04, 2016 5:35:35 PM
   Lease Expires . . . . . . . . . . : Friday, August 05, 2016 5:35:35 PM
   Default Gateway . . . . . . . . . : 192.168.1.1
   DHCP Server . . . . . . . . . . . : 192.168.1.1
   DHCPv6 IAID . . . . . . . . . . . : 245648945
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-1B-87-BF-52-A4-4E-31-AD-78-4C
   DNS Servers . . . . . . . . . . . : 192.168.1.1
   NetBIOS over Tcpip. . . . . . . . : Disabled
```

c. Ask a team member for their PC's IPv4 address and give your PC's IPv4 address to them. Do not provide them with your MAC address at this time.

   Record the IPv4 addresses of the default gateway and the other PCs on the LAN.

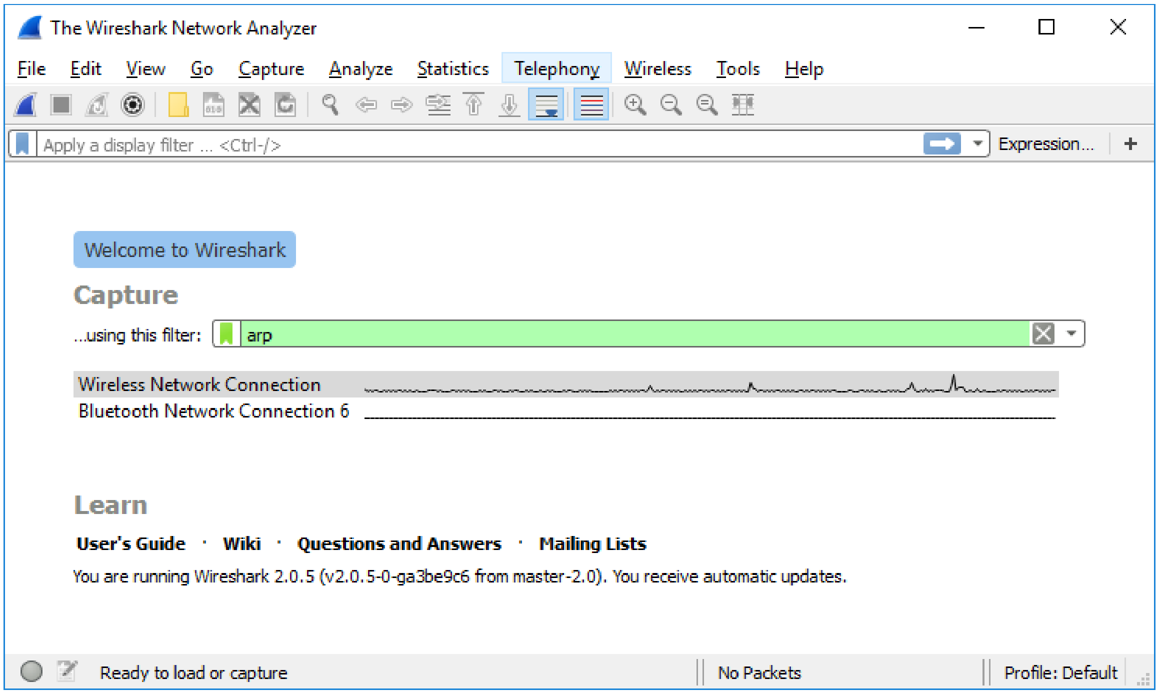
   1. 198.37.25.119
   2. 198.37.24.2

Step 2: Start Wireshark and begin capturing data.

a. On your PC, click **Start** and type **Wireshark**. Click **Wireshark Desktop App** when it appears in the search results window.

Note: Alternatively, your installation of Wireshark may also provide a Wireshark Legacy option. This displays Wireshark in the older but widely recognized GUI. The remainder of this exercise was completed using the newer Desktop App GUI.

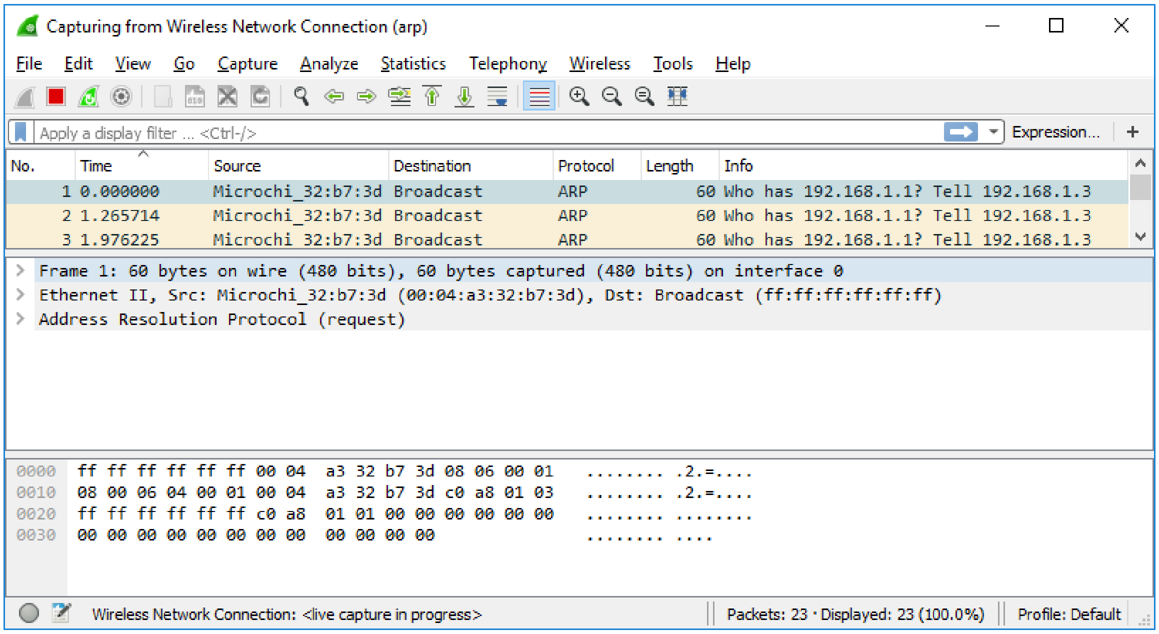
b. After Wireshark starts, select the network interface that you identified with the **ipconfig** command. Enter **arp** in the filter box. This selection configures Wireshark to only display packets that are part of the ARP exchanges between the devices on the local network.


The Wireshark Network Analyzer
File
Edit
View
Go
Capture
Analyze
Statistics
Telephony
Wireless
Tools
Help
Apply a display filter ... <Ctrl-/>
Expression...
+
Welcome to Wireshark
Capture
Wireless Network Connection
Bluetooth Network Connection 6
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
Ready to load or capture
No Packets
Profile: Default

...using this filter: arp

You are running Wireshark 2.0.5 (v2.0.5-0-ga3be9c6 from master-2.0). You receive automatic updates.

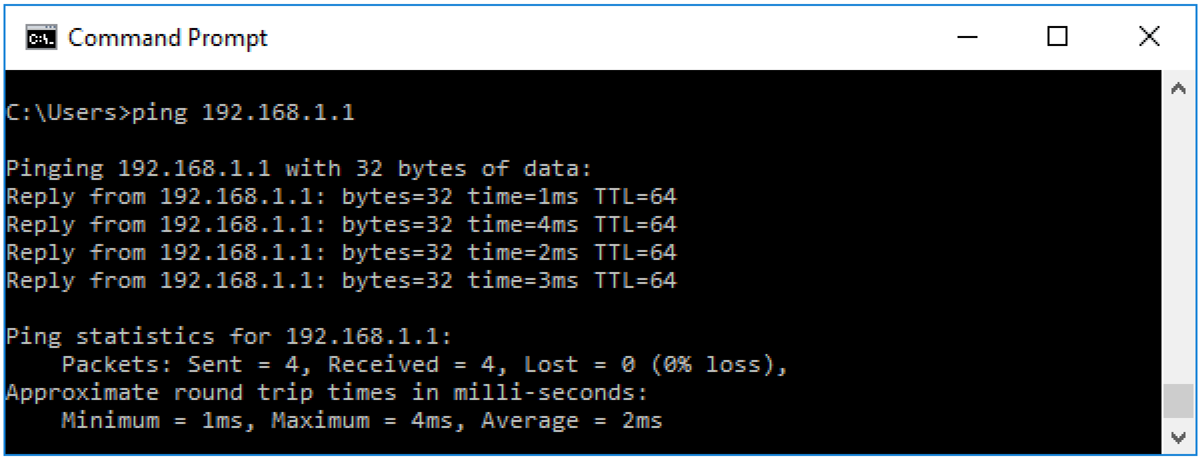
c. After you have selected the correct interface and entered the filter information, click **Start** ( ) to begin the data capture. Information will start scrolling down the top section in Wireshark. Each line represents a message being sent between a source and destination device on the network.


Capturing from Wireless Network Connection (arp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

```
No.  Time        Source             Destination  Protocol  Length  Info
   1 0.000000    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   2 1.265714    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   3 1.976225    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: Microchi_32:b7:3d (00:04:a3:32:b7:3d), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)

0000  ff ff ff ff ff ff 00 04  a3 32 b7 3d 08 06 00 01   ........ .2.=....
0010  08 00 06 04 00 01 00 04  a3 32 b7 3d c0 a8 01 03   ........ .2.=....
0020  ff ff ff ff ff ff c0 a8  01 01 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

Wireless Network Connection: <live capture in progress>
Packets: 23 · Displayed: 23 (100.0%)
Profile: Default

d. Open a command prompt window. Use the **ping** command to test connectivity to the default gateway address that you identified in Part 2, step 1c.



```
C:\Users>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time=1ms TTL=64
Reply from 192.168.1.1: bytes=32 time=4ms TTL=64
Reply from 192.168.1.1: bytes=32 time=2ms TTL=64
Reply from 192.168.1.1: bytes=32 time=3ms TTL=64

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 4ms, Average = 2ms
```

e. Ping the IPv4 addresses of other PCs on the LAN that were provided to you by your team members.

Note: If your team member's PC does not reply to your pings, this may be because their PC firewall is blocking these requests. Ask your instructor for assistance if necessary to disable the PC firewall.

f. Stop capturing data by clicking **Stop Capture (** ■ **)** on the toolbar.

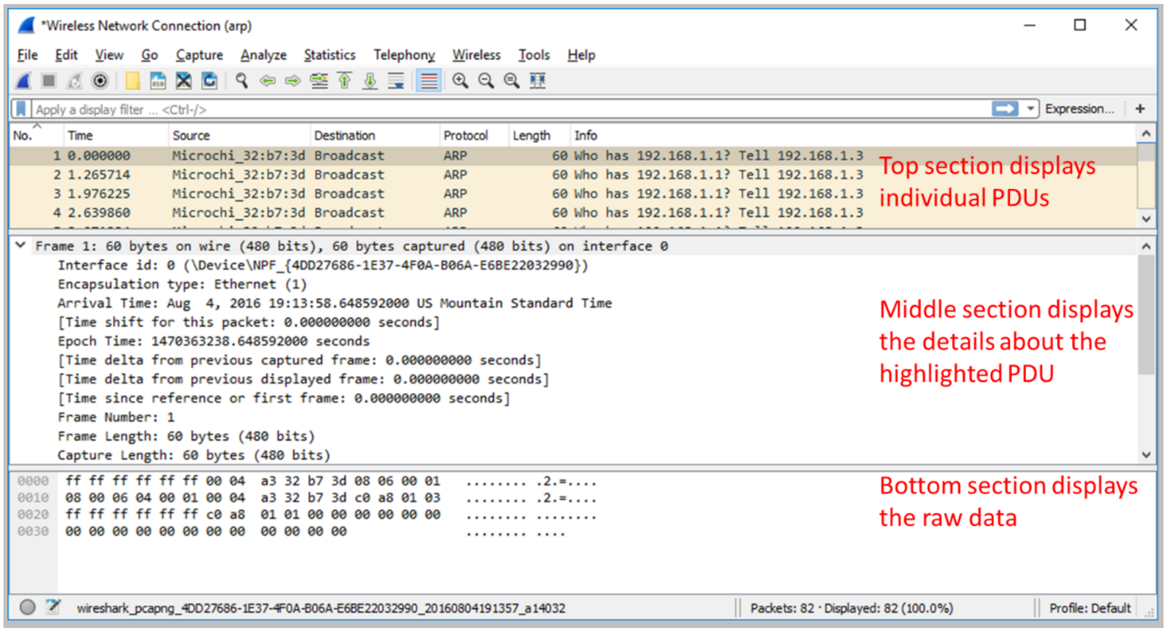
Step 3: Examine the captured data.

In Step 3, examine the data that was generated by the **ping** requests of your team member's PC. Wireshark data is displayed in three sections:

1) The top section displays the list of PDU frames captured with a summary of the IPv4 packet information listed.

2) The middle section lists PDU information for the frame selected in the top part of the screen and separates a captured PDU frame by its protocol layers.

3) The bottom section displays the raw data of each layer. The raw data is displayed in both hexadecimal and decimal form.



```
No.  Time        Source             Destination  Protocol  Length  Info
   1 0.000000    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   2 1.265714    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   3 1.976225    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   4 2.639860    Microchi_32:b7:3d  Broadcast    ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
```

Top section displays individual PDUs

```
v Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
    Interface id: 0 (\Device\NPF_{4DD27686-1E37-4F0A-B06A-E6BE22032990})
    Encapsulation type: Ethernet (1)
    Arrival Time: Aug  4, 2016 19:13:58.648592000 US Mountain Standard Time
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1470363238.648592000 seconds
    [Time delta from previous captured frame: 0.000000000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 0.000000000 seconds]
    Frame Number: 1
    Frame Length: 60 bytes (480 bits)
    Capture Length: 60 bytes (480 bits)
```

Middle section displays the details about the highlighted PDU

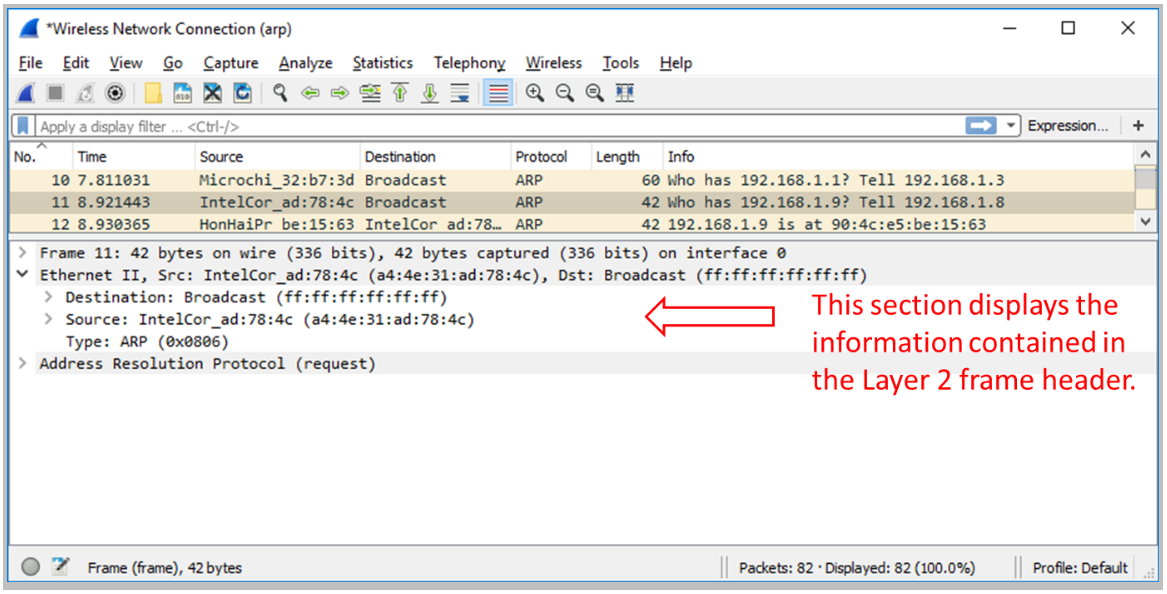
```
0000  ff ff ff ff ff ff 00 04  a3 32 b7 3d 08 06 00 01   ........ .2.=....
0010  08 00 06 04 00 01 00 04  a3 32 b7 3d c0 a8 01 03   ........ .2.=....
0020  ff ff ff ff ff ff c0 a8  01 01 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

Bottom section displays the raw data

```
wireshark_pcapng_4DD27686-1E37-4F0A-B06A-E6BE22032990_20160804191357_a14032 || Packets: 82 · Displayed: 82 (100.0%) || Profile: Default
```

a. Click one of the ARP frames in the top section that has your PC MAC address as the source address in the frame and "broadcast" as the destination of the frame.

b. With this PDU frame still selected in the top section, navigate to the middle section. Click the arrow to the left of the Ethernet II row to view the Destination and Source MAC addresses.


*Wireless Network Connection (arp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Frame (frame), 42 bytes
Packets: 82 · Displayed: 82 (100.0%)
Profile: Default

```
No.   Time       Source            Destination      Protocol  Length  Info
   10 7.811031   Microchi_32:b7:3d Broadcast        ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   11 8.921443   IntelCor_ad:78:4c Broadcast        ARP           42  Who has 192.168.1.9? Tell 192.168.1.8
   12 8.930365   HonHaiPr_be:15:63 IntelCor_ad:78…  ARP           42  192.168.1.9 is at 90:4c:e5:be:15:63
```

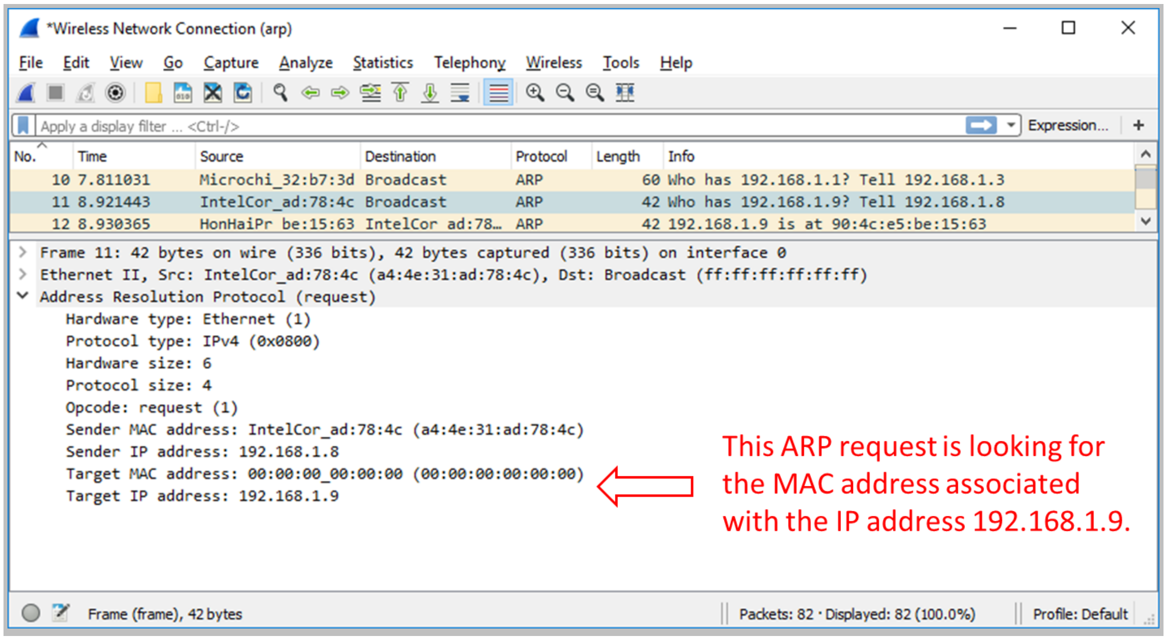
```
> Frame 11: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
v Ethernet II, Src: IntelCor_ad:78:4c (a4:4e:31:ad:78:4c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  > Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  > Source: IntelCor_ad:78:4c (a4:4e:31:ad:78:4c)
    Type: ARP (0x0806)
> Address Resolution Protocol (request)
```

This section displays the information contained in the Layer 2 frame header.

Does the Source MAC address match your PC's interface?

Yes they match.

g. Click the arrow to the left of the Address Resolution Protocol (request) row to view the content of the ARP request.


*Wireless Network Connection (arp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
Frame (frame), 42 bytes
Packets: 82 · Displayed: 82 (100.0%)
Profile: Default

```
No.   Time       Source            Destination      Protocol  Length  Info
   10 7.811031   Microchi_32:b7:3d Broadcast        ARP           60  Who has 192.168.1.1? Tell 192.168.1.3
   11 8.921443   IntelCor_ad:78:4c Broadcast        ARP           42  Who has 192.168.1.9? Tell 192.168.1.8
   12 8.930365   HonHaiPr_be:15:63 IntelCor_ad:78…  ARP           42  192.168.1.9 is at 90:4c:e5:be:15:63
```

```
> Frame 11: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
> Ethernet II, Src: IntelCor_ad:78:4c (a4:4e:31:ad:78:4c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
v Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: IntelCor_ad:78:4c (a4:4e:31:ad:78:4c)
    Sender IP address: 192.168.1.8
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.1.9
```

This ARP request is looking for the MAC address associated with the IP address 192.168.1.9.

Step 4: Locate the ARP response frame that corresponds to the ARP request that you highlighted.

a. Using the Target IPv4 address in the ARP request, locate the ARP response frame in the upper section of the Wireshark capture screen.

   What is the IPv4 address of the Target device in your ARP request? ________________________

   198.37.25.120

b. Highlight the response frame in the upper section of the Wireshark output. You may have to scroll the window to find the response frame that matches the Target IPv4 address identified in the previous step. Expand the Ethernet II and Address Resolution Protocol (response) rows in the middle section of the screen.

   Is the ARP response frame a broadcast frame? ______________________________

   No

   What is the destination MAC address of the frame? ______________________________________

   a8:a1:59:53:97:34

   Is this the MAC address of your PC?

   Yes

   What MAC address is the source of the frame? ____

   54:bf:64:04:d6:32

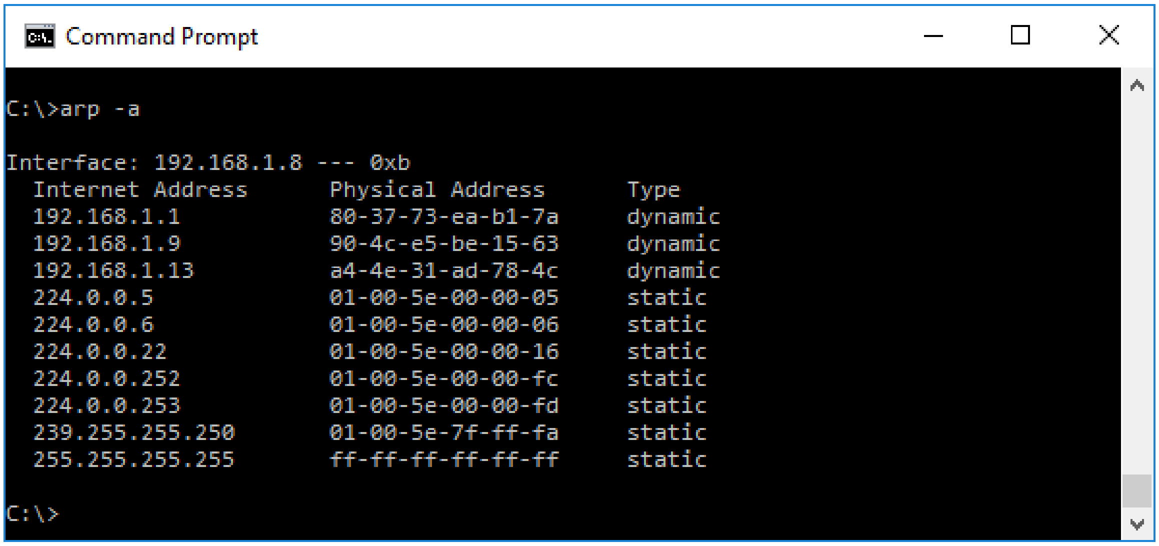
c. Verify with your team member that the MAC address matches the MAC address of their PC.

B. Examine the ARP cache entries on the PC.

After the ARP reply is received by the PC, the MAC Address to IPv4 address association is stored in cache memory on the PC. These entries will stay in memory for a short period of time (from 15 to 45 seconds), then, if they are not used within that time, they will be removed from cache.

Step 1: View ARP cache entries on a Windows PC.

a. Open a command prompt window on the PC. At the prompt, enter **arp –a** and press enter.



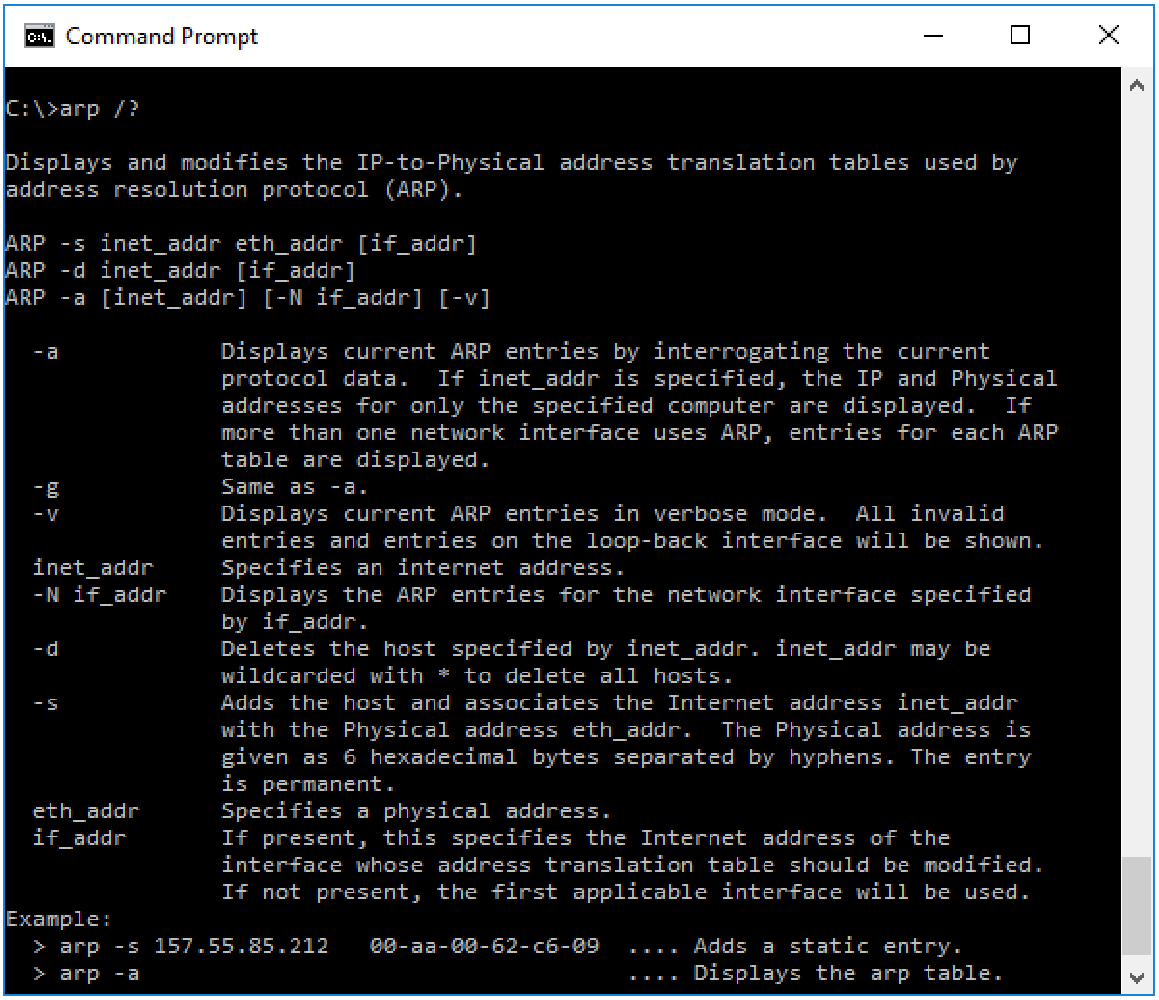
```
C:\>arp -a

Interface: 192.168.1.8 --- 0xb
  Internet Address      Physical Address      Type
  192.168.1.1           80-37-73-ea-b1-7a     dynamic
  192.168.1.9           90-4c-e5-be-15-63     dynamic
  192.168.1.13          a4-4e-31-ad-78-4c     dynamic
  224.0.0.5             01-00-5e-00-00-05     static
  224.0.0.6             01-00-5e-00-00-06     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  224.0.0.253           01-00-5e-00-00-fd     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\>
```

The output of the **arp –a** command displays the entries that are in the cache on the PC. In the example, the PC has entries for the default gateway (192.168.1.1) and for two PCs that are located on the same LAN (192.168.1.9 and 192.168.1.13).

What is the result of executing the **arp –a** command on your PC?

It is the list of all MAC known by IPv4 addresses

h. The **arp** command on the Windows PC has another functionality. Enter **arp /?** at the command prompt and press enter. The **arp** command options enable you to view, add and remove ARP table entries if necessary.



```
C:\>arp /?

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.  All invalid
                entries and entries on the loop-back interface will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be modified.
                If not present, the first applicable interface will be used.
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
  > arp -a                                    .... Displays the arp table.
```

Which option deletes an entry from the ARP cache?

arp -d

What would be the result of issuing the **arp –d *** command?

The current address bindings in the ARP cache would be deleted.

Reflection

1. What is a benefit of keeping ARP cache entries in memory on the source computer?

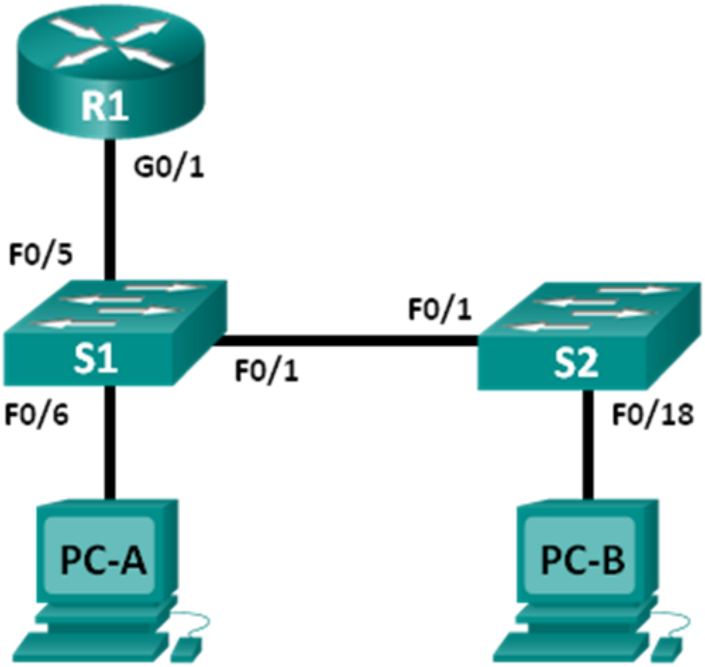
Our computers always check local cache before searching out for information by itself. So if something is known it could save time by checking this list of known address bindings. This also ensures that ARP requests are not constantly made.

2. If the destination IPv4 address is not located on the same network as the source host, what MAC address will be used as the destination target MAC address in the frame?

    The MAC of the gateway.

Exercise 3: Observing ARP in Windows CLI and IOS CLI (App)

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/1	192.168.1.1	255.255.255.0	N/A
S1	VLAN 1	192.168.1.11	255.255.255.0	192.168.1.1
S2	VLAN 1	192.168.1.12	255.255.255.0	192.168.1.1
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1
PC-B	NIC	192.168.1.2	255.255.255.0	192.168.1.1

Note: Use the following table as a guideline to choose the correct cable type between devices.

	Hub	**Switch**	**Router**	**PC**
Hub	Crossover	Crossover	Straight-through	Straight-through
Switch	Crossover	Crossover	Straight-through	Straight-through
Router	Straight-through	Straight-through	Crossover	Crossover
PC	Straight-through	Straight-through	Crossover	Crossover

A. Build and Configure the Network in Packet Tracer

a. Cable the network according to the topology.

b. Configure the IP addresses for the devices according to the Addressing Table. If you do not recall the commands, you may refer to Exp 3.

c. Verify network connectivity by pinging all the devices from PC-B.

B. Use the Windows ARP Command

The **arp** command allows the user to view and modify the ARP cache in Windows. You access this command from the Windows command prompt.

a. Display the ARP cache.

i. Open a command window on PC-A and type **arp**.

```
C:\> arp

Displays and modifies the IP-to-Physical address translation
tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]
```

```
  -a            Displays current ARP entries by interrogating
the current
                protocol data.  If inet_addr is specified, the
IP and Physical
                addresses for only the specified computer are
displayed.  If
                more than one network interface uses ARP,
entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.
All invalid
                entries and entries on the loop-back interface
will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network
interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr.
inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet
address inet_addr
                with the Physical address eth_addr.  The
Physical address is
                given as 6 hexadecimal bytes separated by
hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address
of the
                interface whose address translation table
should be modified.
                If not present, the first applicable interface
will be used.
Example:
   > arp -s 157.55.85.212    00-aa-00-62-c6-09   .... Adds a
static entry.
  > arp -a                                     .... Displays the
arp table.
```

ii. Examine the output.

What command would be used to display all entries in the ARP cache?

arp -a

What command would be used to delete all ARP cache entries (flush ARP cache)?

arp -d *

What command would be used to delete the ARP cache entry for 192.168.1.11?

arp -d 192.168.1.11

iii. Type **arp –a** to display the ARP table.

```
C:\> arp –a

Interface: 192.168.1.3 --- 0x13
  Internet Address      Physical Address      Type
  192.168.1.1           50-3d-e5-aa-c0-a1     dynamic
  224.0.0.22            01-00-5e-00-00-16     static
C:\> arp -a
No ARP Entries Found.
```

iv. Ping from PC-A to PC-B to dynamically add entries in the ARP cache.

```
C:\> ping 192.168.1.2

Interface: 192.168.1.3 --- 0x13
  Internet Address      Physical Address      Type
  192.168.1.1           50-3d-e5-aa-c0-a1     dynamic
  192.168.1.2           00-21-70-cf-3d-cc     dynamic
  224.0.0.22            01-00-5e-00-00-16     static
```

What is the physical address for the host with an IP address of 192.168.1.2?

00-21-70-cf-3d-cc

b. Adjust entries in the ARP cache manually.

To delete entries in ARP cache, issue the command **arp –d {inet-addr | *}**Addresses can be deleted individually by specifying the IP address, or all entries can be deleted with the wildcard *.

Verify that the ARP cache contains the following entries: the R1 G0/1 default gateway (192.168.1.1), PC-B (192.168.1.2) and both switches (192.168.1.11 and 192.168.1.12).

i. From PC-A, ping all the addresses in the Address Table.

ii. Verify that all the addresses have been added to the ARP cache. If the address is not in ARP cache, ping the destination address and verify that the address was added to the ARP cache.

```
C:\> arp –a

Interface: 192.168.1.3 --- 0x13
  Internet Address      Physical Address      Type
  192.168.1.1           50-3d-e5-aa-c0-a1     dynamic
  192.168.1.2           00-21-70-cf-3d-cc     dynamic
  192.168.1.11          00-09-b7-e6-c0-40     dynamic
  192.168.1.12          00-17-e0-2c-56-c0     dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
```

iii. Type **arp –d ***. This command deletes all the ARP cache entries. Verify that all the ARP cache entries are deleted by typing **arp –a** at the command prompt.

```
C:\windows\system32> arp –d *
C:\windows\system32> arp –a
No ARP Entries Found.
```

iv. Wait a few minutes. The Neighbor Discovery protocol starts to populate the ARP cache again.

```
C:\> arp –a

Interface: 192.168.1.3 --- 0xb
  Internet Address      Physical Address      Type
  192.168.1.255         ff-ff-ff-ff-ff-ff     static
```

v. From PC-A, ping PC-B (192.168.1.2) and the switches (192.168.1.11 and 192.168.1.12) to add the ARP entries. Verify that the ARP entries have been added to the cache.

```
C:\> arp –a
Interface: 192.168.1.3 --- 0xb
  192.168.1.1           50-3d-e5-aa-c0-a1     dynamic
  192.168.1.2           00-21-70-cf-3d-cc     dynamic
  192.168.1.11          00-09-b7-e6-c0-40     dynamic
  192.168.1.12          00-17-e0-2c-56-c0     dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
```

vi. Record the physical address for switch S2.

0c:17:e0:2c:56:c0

vii. Delete a specific ARP cache entry by typing **arp –d *inet-addr***. At the command prompt, type **arp -d 192.168.1.12** to delete the ARP entry for S2.

```
C:\windows\system32> arp –d 192.168.1.12
```

viii. Type **arp –a** to verify that the ARP entry for S2 has been removed from the ARP cache.

```
C:\> arp –a

Interface: 192.168.1.3 --- 0x13
  Internet Address      Physical Address      Type
  192.168.1.1           50-3d-e5-aa-c0-a1     dynamic
  192.168.1.2           00-21-70-cf-3d-cc     dynamic
  192.168.1.11          00-09-b7-e6-c0-40     dynamic
  224.0.0.22            01-00-5e-00-00-16     static
```

C. Use the IOS show arp Command

The Cisco IOS can also display the ARP cache on routers and switches with the **show arp** or **show ip arp** command.

a. Display ARP entries on router R1.

```
R1# show arp
Protocol   Address           Age (min)   Hardware Addr   Type
Interface
Internet   192.168.1.1              -     503d.e5aa.c0a1   ARPA
GigabitEthernet0/1
```

```
Internet   192.168.1.2               29    0021.70cf.3dcc   ARPA
GigabitEthernet0/1

Internet   192.168.1.3               10    0026.b9dd.0091   ARPA
GigabitEthernet0/1

Internet   192.168.1.12              37    0017.e02c.56c0   ARPA
GigabitEthernet0/1

R1#
```

Notice there is no Age (-) for the first entry, router interface G0/1 (the LAN default gateway). The Age is the number of minutes (min) that the entry has been in ARP cache and is incremented for the other entries. The Neighbor Discovery protocol populates the PC-A and PC-B IP and MAC address ARP entries.

b. Add ARP entries on router R1.

You can add ARP entries to the ARP table of the router by pinging other devices.

i. Ping switch S1.

```
R1# ping 192.168.1.11
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.11, timeout is 2
seconds:
..!!!
Success rate is 60 percent (3/5), round-trip min/avg/max =
1/1/1 ms
```

ii. Verify that an ARP entry for switch S1 has been added to the ARP table of R1.

```
R1# show ip arp
Protocol   Address              Age  (min)   Hardware Addr    Type
Interface
Internet   192.168.1.1                  -     503d.e5aa.c0a1   ARPA
GigabitEthernet0/1
Internet   192.168.1.2                 32     0021.70cf.3dcc   ARPA
GigabitEthernet0/1
Internet   192.168.1.3                 13     0026.b9dd.0091   ARPA
GigabitEthernet0/1
Internet   192.168.1.11                 0     0009.b7e6.c040   ARPA
GigabitEthernet0/1
Internet   192.168.1.12                40     0017.e02c.56c0   ARPA
GigabitEthernet0/1
```

c. Display ARP entries on switch S1.

```
S1# show ip arp
Protocol   Address               Age  (min)   Hardware  Addr    Type
Interface
Internet   192.168.1.11                  -     0009.b7e6.c040   ARPA
VLAN1
Internet   192.168.1.12                 42     0017.e02c.56c0   ARPA
VLAN1
Internet   192.168.1.1                   3     503d.e5aa.c0a1   ARPA
VLAN1
Internet   192.168.1.3                  16     0026.b9dd.0091   ARPA
VLAN1
```

d. Add ARP entries on switch S1.

By pinging other devices, ARP entries can also be added to the ARP table of the switch.

i. From switch S1, ping PC-B.

```
S1# ping 192.168.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.2, min/avg/max =
1/201/1002 ms
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max =
1/2/8 ms
```

ii. Verify that the ARP entry for PC-B has been added to the ARP table of S1.

```
S1# show ip arp
Protocol   Address               Age  (min)   Hardware  Addr    Type
Interface
Internet   192.168.1.11                  -     0009.b7e6.c040   ARPA
VLAN1
Internet   192.168.1.12                 44     0017.e02c.56c0   ARPA
VLAN1
Internet   192.168.1.1                   5     503d.e5aa.c0a1   ARPA
VLAN1
Internet   192.168.1.3                  17     0026.b9dd.0091   ARPA
VLAN1
```

```
Internet   192.168.1.2              0    0021.70cf.3dcc   ARPA
VLAN1
```

Reflection

1. How and when are static ARP entries removed?

   Deleted manually using arp -d.

2. Why do you want to add static ARP entries in the cache?

   As it is able to mitigate ARP spoofing or damage within the network.

3. If ARP requests can cause network latency, why is it a bad idea to have unlimited hold times for ARP entries?

It is because it may cause issues with devices that leave the network and it would be unnecessary information withheld. Along with that there may be problems within devices capable of changing the third layer.