MASINDE MULIRO UNIVERSITY OF SCIENCE & TECHNOLOGY

LAB-02: USING WIRESHARK & ADDRESS RESOLUTION PROTOCOL

1.0 Objectives

The objectives of this Lab are:

(a) To master how to download and install Wireshark
(b) To understand the use of arp command
(c) To capture ping and arp requests and responses on a LAN using Wireshark
(d) To explain the different fields of a packet trace obtained using Wireshark

2.0 Background

Wireshark software tool is used to capture and examine a packet trace. A packet trace is a record of traffic at a location on the network, as if a snapshot was taken of all the bits that passed across a particular wire. The packet trace records a timestamp for each packet, along with the bits that make up the packet, from the lower-layer headers to the higher-layer contents.

Wireshark runs on most operating systems, including Windows, Mac and Linux. It provides a graphical User Interface (UI) that shows the sequence of packets and the meaning of the bits when interpreted as protocol headers and data. It color-codes packets by their type, and has various ways to filter and analyze packets to let you investigate the behavior of network protocols.

Wireshark is widely used to troubleshoot networks. You can download it from http://www.wireshark.org if it is not already installed on your computer. We highly recommend that start by reading the Introduction to Wireshark material available at the following site.

https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html

3.0 Requirements

This lab uses:

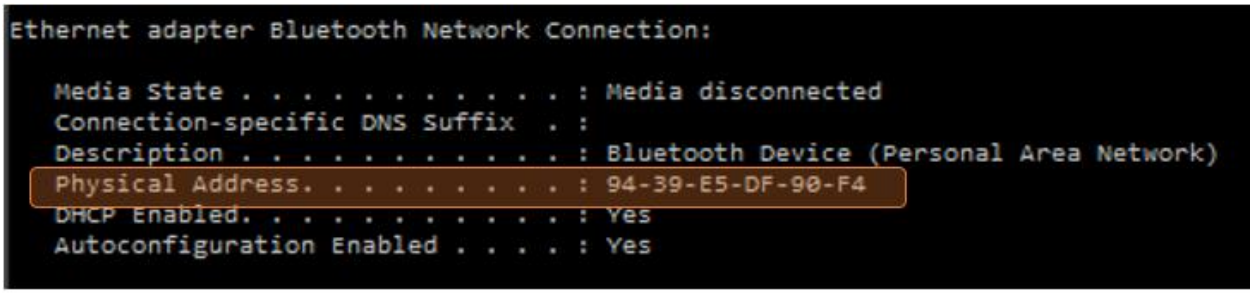
1. **Wireshark software** tool to capture and examine a packet trace.
2. **arp:** This lab uses the "arp" command-line utility to inspect and clear the cache used by the ARP protocol on your computer. arp is installed as part of the operating system on Windows, Linux, and Mac computers, but uses different arguments. It requires administrator privileges to clear the cache.

3. **ifconfig / ipconfig:** This lab uses the "ipconfig" (Windows) command-line utility to inspect the state of your computer's network interface. ipconfig is installed as part of the operating system on Windows computers.
4. **route / netstat:** This lab uses the "route" or "netstat" command-line utility to inspect the routes used by your computer. A key route is the default route (or route to prefix 0.0.0.0) that uses the default gateway to reach remote parts of the Internet. Both "route" and "netstat" are installed as part of the operating system across Windows and Mac/Linux, but there are many variations on the commandline parameters that must be used.
5. **Browser:** This lab uses a web browser to find or fetch pages as a workload. Any web browser will do.

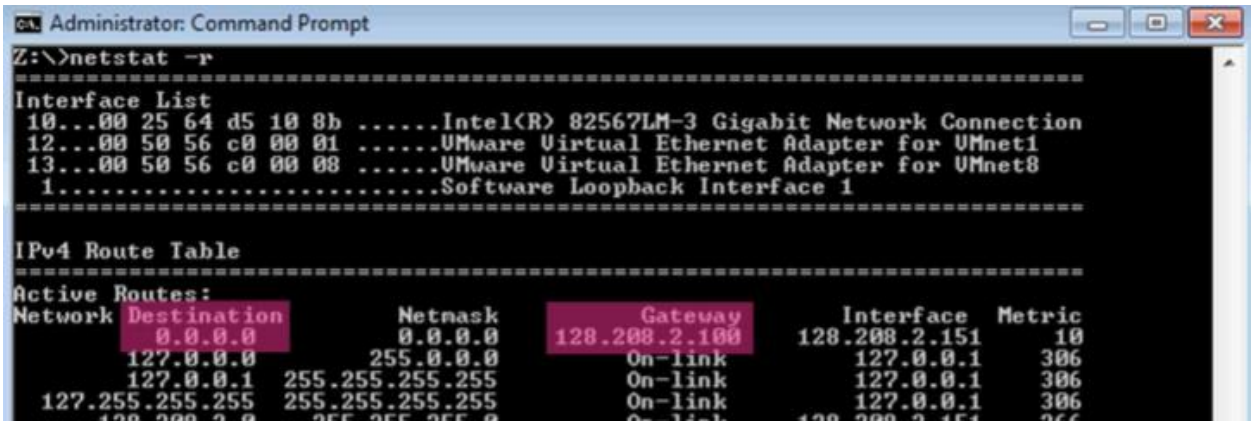
EXERCISE 1.

1. Find and record the Ethernet address of the main network interface of your computer with the ifconfig / ipconfig command. You will want to know this address for later analysis.

   On Windows, bring up a command-line shell and type "ipconfig /all". Among the output will be a section for the main interface of the computer (likely an Ethernet interface) and its Ethernet address. Common names for the interface are "eth0", "en0", or "Ethernet adapter". An example is shown below.



2. Find and record the IP address of the local router or default gateway that your computer uses to reach the rest of the Internet using the netstat / route command. You should be able to use the netstat command ("netstat –r" on Windows, Mac and Linux, may require ctrl-C to stop). Alternatively, you can use the route command ("route print" on Windows, "route" on Linux, "route –n get default" on Mac). In either case you are looking for the gateway IP address that corresponds to the destination of default or 0.0.0.0. An example is shown in below.



3. Launch Wireshark and start a capture with a filter of "arp". Your capture window should be similar to the one pictured below, other than our highlighting. Select the interface from which to capture as the main wired or wireless interface used by your computer to connect to the Internet. If unsure, guess and revisit this step later if your capture is not successful. Uncheck "capture packets in promiscuous mode". This mode is useful to overhear packets sent to/from other computers on broadcast networks. We only want to record packets sent to/from your computer. Leave other options at their default values. The capture filter, if present, is used to prevent the capture of other traffic your computer may send or receive. On Wireshark 1.8, the capture filter box is present directly on the options screen, but on Wireshark 1.9, you set a capture filter by double-clicking on the interface.

EXERCISE TWO

Assume that we desire to ping another computer on your LAN. We know that ping command works using **ICMP**. **ICMP** is encapsulated inside **IP datagram** and **IP datagram** is encapsulated within **Ethernet Frame**. We need Source **IP Address** (e.g, 192.168.0.84), Destination **IP Address** (e.g. 192.168.0.122), Source **MAC Address** (my **MAC Address** e.g. 08:00:27:58:58:98) and Destination **MAC Address** to make the Ethernet Frame for ICMP message.

To ping the other machine, the source **IP Address**, Destination **IP Address**, Source MAC Address must be known. However, the Destination **MAC Address** is unknown.

1. Start wireshark

2. Check and write down the IP address of your computer.

3. Move to the next computer and write down its IP address.

4. Enter the command mode on the first workstation and Type:

>ARP

Your screen should resemble the one shown below (Not obtained from your Lab!).

```
C:\Users\Ambani>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.
```

3. View the ARP cache by entering the command:

> arp -a

By setting various options, listed, discuss the applications of the command ARP for network administration.

4. Clear the ARP cache. Request for administrative rights if necessary.

5. To ping the other workstation, your computer has to resolve the destination MAC address using ARP. It will prepare an ARP Request message and send it with a Destination **MAC Address** as FF:FF:FF:FF:FF:FF (Broadcast **MAC Address**) to LAN Switch.

6. Get the screenshot (use snipping tool) of the of the resulting broadcast and reply.

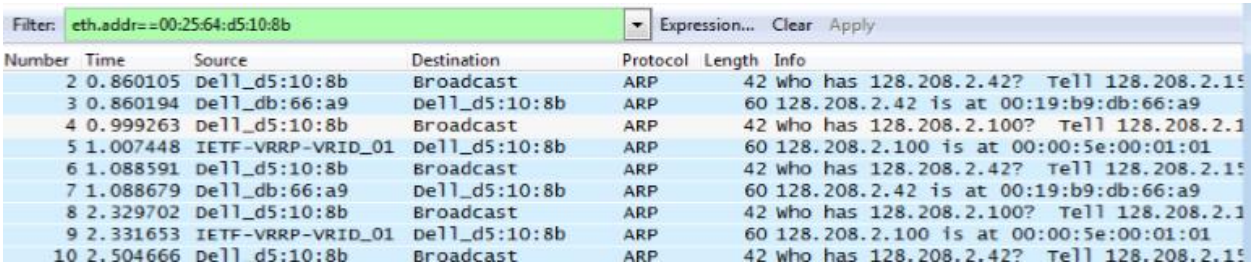
7. Describe the various fields of the screen shot.

EXERCISE THREE – ARP EXCHANGE

3.1 EXPERIMENTS

1. Since there may be many ARP packets in your trace, we'll first narrow our view to only the ARP packets that are sent directly from or to your computer.

   Set a display filter for packets with the Ethernet address of your computer. You can do this by entering an expression in the blank "Filter:" box near the top of the Wireshark window and clicking "Apply". The filter to enter depends on your Ethernet address. For example, if your Ethernet address is 01:02:03:04:05:06 then enter a filter expression of "eth.addr==01:02:03:04:05:06". Note the double equal sign. If you are using the supplied trace, it comes with an additional text file giving the Ethernet address and default gateway IP address. After applying this filter your capture should look something like the figure below, in which we have expanded the ARP protocol details.



2. Find and select an ARP request for the default gateway and examine its fields. There are two kinds of ARP packets, a request and a reply, and we will look at each one in turn. The Info line for the request will start with "Who has …". You want to look for one of these packets that asks for the MAC address of the default gateway, e.g., "Who has xx.xx.xx.xx …" where xx.xx.xx.xx is your default gateway. You can click on the + expander or icon for the Address Resolution Protocol block to view the fields:
   - **Hardware and Protocol** type are set to constants that tell us the hardware is Ethernet and the protocol is IP. This matches the ARP translation from IP to Ethernet address.
   - **Hardware and Protocol size** are set to 6 and 4, respectively. These are the sizes of Ethernet and IP addresses in bytes.
   - **The opcode field** tells us that this is a request.
   - **the sender MAC (Ethernet) and IP and the target MAC (Ethernet) and IP**. These fields are filled in as much as possible. For a request, the sender knows their MAC and IP address and fills them in.
   - **The sender also knows the target IP address** – it is the IP address for which an Ethernet address is wanted. But the sender does not know the target MAC address, so it does not fill it in.

3. Next, select an ARP reply and examine its fields. The reply will answer a request and have an Info line of the form “xx.xx.xx.xx is at yy:yy:yy:yy:yy:yy”:
    - The Hardware and Protocol type and sizes are as set as before
    - The opcode field has a different value that tells us that this is a reply.

    - Next come the four key fields, the sender MAC (Ethernet) and IP and the target MAC (Ethernet) and IP just as before. These fields are reversed from the corresponding request, since the old target is the new sender (and vice versa). The fields should now be all filled in since both computers have supplied their addresses.

3.2 REPORT WRITING

ARP packets are carried in Ethernet frames, and the values of the Ethernet header fields are chosen to support ARP. For instance, you may wonder how an ARP request packet is delivered to the target computer so that it can reply and tell the requestor its MAC address. The answer is that the ARP request is (normally) broadcast at the Ethernet layer so that it is received by all computers on the local network including the target. Look specifically at the destination Ethernet address of a request: it is set to ff:ff:ff:ff:ff:ff, the broadcast address. So the target receives the request and recognizes that it is the intended recipient of the message; other computers that receive the request know that it is not meant for them. Only the target responds with a reply. However, anyone who receives an ARP packet can learn a mapping from it: the sender MAC and sender IP pair.

To look at further details of ARP, examine an ARP request and ARP reply to answer these questions in your report.

1. What opcode is used to indicate a request? What about a reply?
2. How large is the ARP header for a request? What about for a reply?
3. What value is carried on a request for the unknown target MAC address?
4. What Ethernet Type value which indicates that ARP is the higher layer protocol?
5. Is the ARP reply broadcast (like the ARP request) or not?