|  |  |  |  |
| --- | --- | --- | --- |
| **Ex. No.: 11**  **Date: 14/10/22** | **Working with tools for analyzing protocols (Wireshark)** | | |
| **Objective(s):**    To analyze various protocols such as ARP, DNS, TCP, UDP, HTTP etc. and packet formats of these protocols using Wireshark tool. | | | |
| **Implementation:**  **Comparison Operators**   1. eq, == Equal 2. ne, != Not Equal 3. gt, > Greater Than 4. lt, < Less Than 5. ge, >= Greater than or Equal to 6. le, <= Less than or Equal to   **Logical Operators**   1. not, ! Logical NOT (right-associative) 2. and, && Logical AND (left-associative) 3. or, || Logical OR (left-associative)   **Commands**   1. **ip.src==IP-address and ip.dst==IP-address**   This filter shows packets sent from one computer (ip.src) to another (ip.dst).     1. **tcp.port == PORT\_NUMBER || udp.port == PORT\_NUMBER**   This filter will show you all traffic on port PORT\_NUMBER, which is usually SMTP traffic     1. **icmp**   This filter will show you only ICMP traffic in the capture, most likely they are pings     1. **ip.addr == IP\_ADDRESS**   This filter shows you all traffic to or from the computer with the specified IP address     1. **ip.addr != IP\_ADDRESS**   This filter shows you all traffic except the traffic to or from the computer with the specified IP address     1. **ip.len**   Total Length of the IP address - Unsigned integer (2 bytes) - 1.0.0 to 4.0.0     1. **tcp.window\_size == 0**   When a client (or server – but it is usually the client) advertises a zero value for its window size, this indicates that the TCP receive buffer is full and it cannot receive any more data.     1. **tcp.flags.reset != 1**   It is used to terminate the connection if the RST sender feels something is wrong with the TCP connection or that the conversation should not exist. It can get send from receiver side when packet is send to particular host that was not expecting it.     1. **dns**   DNS filtering is the process of using the Domain Name System to block malicious websites and filter out harmful or inappropriate content.The built-in dns filter in Wireshark shows only DNS protocol traffic. Also, as shown below, DNS traffic is shown in a light blue in Wireshark by default.     1. **udp**   The UDP layer provides datagram based connectionless transport layer (layer 4) functionality in the InternetProtocolFamily. UDP is only a thin layer, and provides not much more than the described UDP port multiplexing.         1. **http**   To view only HTTP traffic, type http (lower case) in the Filter box and press Enter. Select the first HTTP packet labeled GET /. Observe the destination IP address. To view all related traffic for this connection, change the filter to ip.addr == <destination>, where <destination> is the destination address of the HTTP packet.     1. **eth**   Ethernet is the most common local area networking technology, and, with gigabit and 10 gigabit Ethernet, is also being used for metropolitan-area and wide-area networking. Ethernet packets with less than the minimum 64 bytes for an Ethernet packet (header + user data + FCS) are padded to 64 bytes, which means that if there's less than 64-(14+4) = 46 bytes of user data, extra padding data is added to the packet. | | | |
| **Result:**  Thus the various protocols such as ARP, DNS, TCP, UDP, HTTP etc. and packet formats of these protocols are analyzed using Wireshark tool. | | | |
| **Remarks by the Course Instructor:** | | **Date of completion** |  |
| **Marks** |  |
| **Course Instructor’s signature with date** |  |