CNL

3.1.17

Assignment No.: 7

Title: Packet Sniffing

Problem statement:

Study of wireshark & header format of ethernet, IP, TCP & UDP and write a program to analyze.

Objectives:

1) To learn of understand header format of ethernet, IP, TCP & UDP
2) To learn concept of wireshark.

Theory:

1) Wireshark:

We use a packet sniffer called wireshark. It is (formely known as ETHEREAL) is a free packet sniffer/analyzer which is available for both UNIX-like (UNIX, Linux, MacOS, BSD & Solaris) & Windows operating system.

It captures packets from a network interface of displays them with detailed protocol information. Wireshark, however, is a passive analyzer. It only captures packet without manipulate them. It neither sends packets to the network nor does other active operations.

Wireshark is a valuable tool for protocol developers who may use it to debug protocol implementations. It is also great educational tool for computer network. Students who can use it is to see details of operations in real time

Main Window:

Wireshark window is made of seven sections Title bar, menubar, filter bar, packet list part packet detail pane, packet byte pane & status bar

IP Protocol Header Format.

Internet Protocol being a layer 3 protocol (OSI) takes tcb segments from layer 4 & divides it into packets. IP packet encapsulates data unit received from above layer & add to its own header information.

IP Header	Layer 4 Data

IP header includes many relevant information including version number which in the context, is 4.

1) Version: Version no: of internet protocol used IPv4.
2) IHL: Internet Header length: Length of entire IP header.

3) DSCP: Differentiated services code point, this is a type of service.
4) ECN: Explicit congestion Notification; It carries information about congestion seen in the route.
5) Total Length: Length of entire IP packet & including IP header & IP payload.
6) Identification: If IP packet is fragmented during the transmission all the fragments contain same identification number to identify original IP packet they belong to.
7) Flags: As required by network resources if IP packet is too large to handle, these flags tell if they can be fragmented or not. In 3-bit flag, MSB is always set to '0'.
8) Fragment offset: Tells exact position of packet of fragment in IP.
9) Time to live: At each hop, its value decremented by one.
10) Protocol: Tells network layer at distinct host, to which protocol this packet belongs to ie next level protocol.
11) Header checksum: used to check if packet error free.
12) Source address: 32 bit address of sender (or source) of packet
13) Destination Address: 32 bit address of receiver of packets

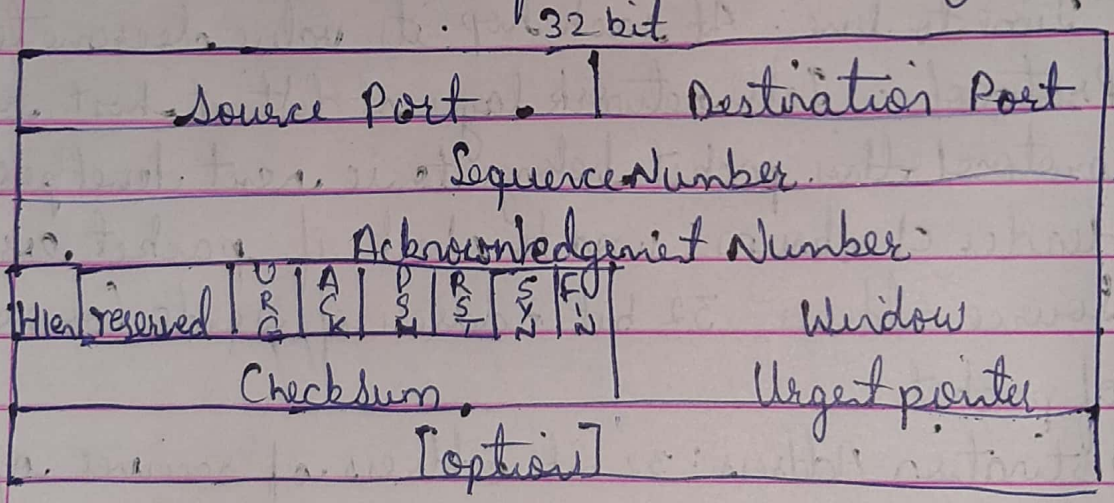
TCP Header format

Each TCP header has ten required fields, totaling 20 bytes (160 bits) in size. They can also optionally include an additional data section upto 40 bytes in size.

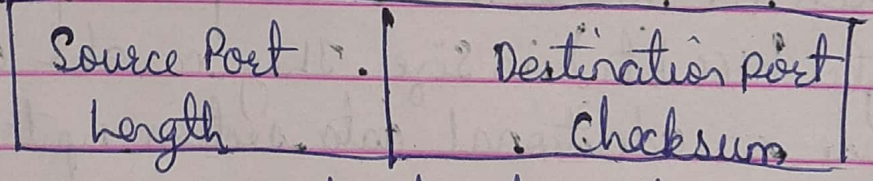
This is layout of TCP header.

1) Source TCP port number (2 bytes)
2) Destination TCP port number (2 bytes)
3) Sequence number (4 bytes)
4) Acknowledgement number (4 bytes)
5) TCP data offset (4 bits)
6) Reserved data (3 bits)
7) Control flags (upto 9 bits)
8) Window Size (2 bytes)
9) TCP checksum (2 bytes)
10) Urgent pointer (2 bytes)
11) TCP optional data (0-40 bytes)

TCP inserts header fields into message stream



UDP header Format:



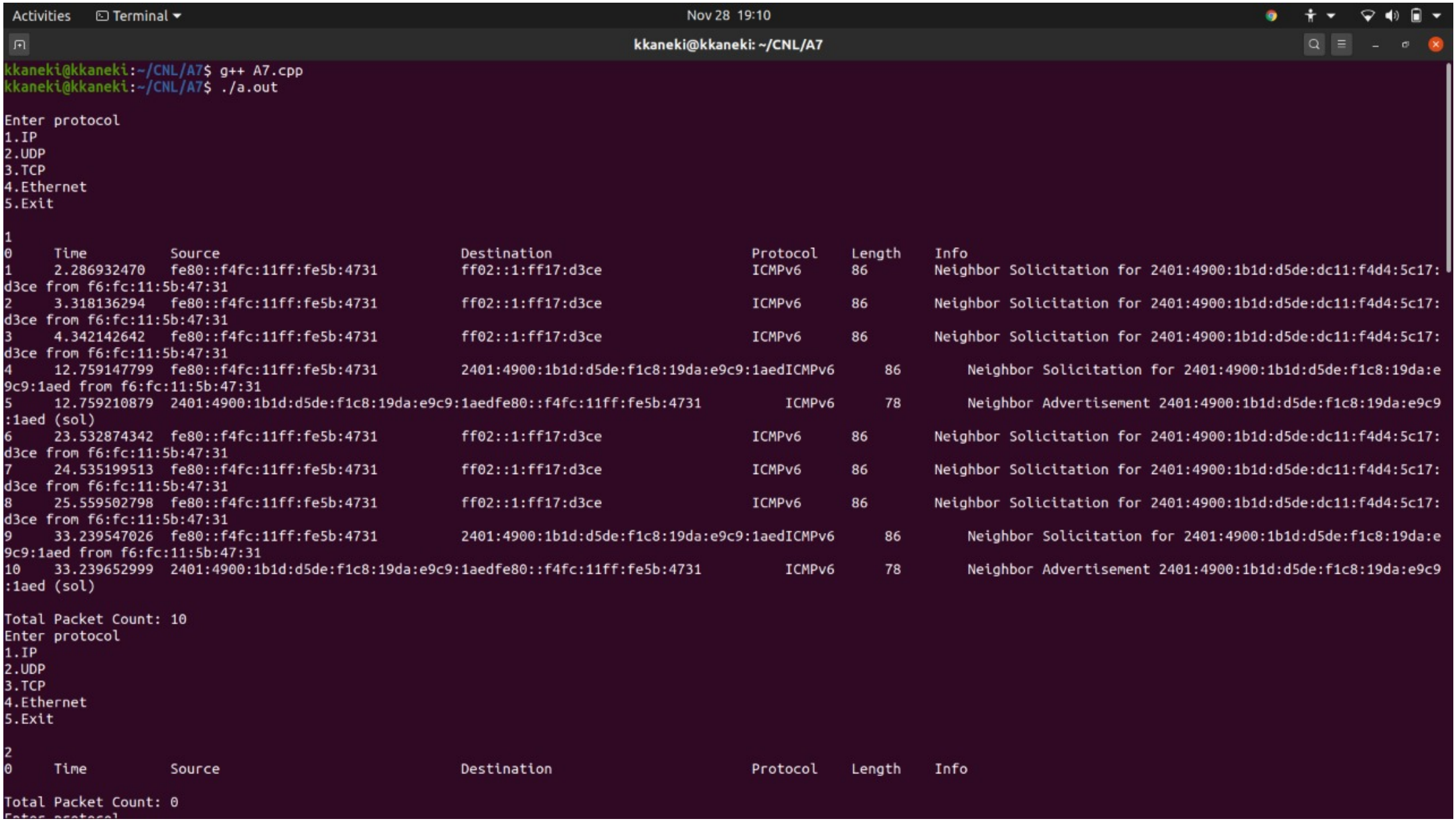
UDP header format.

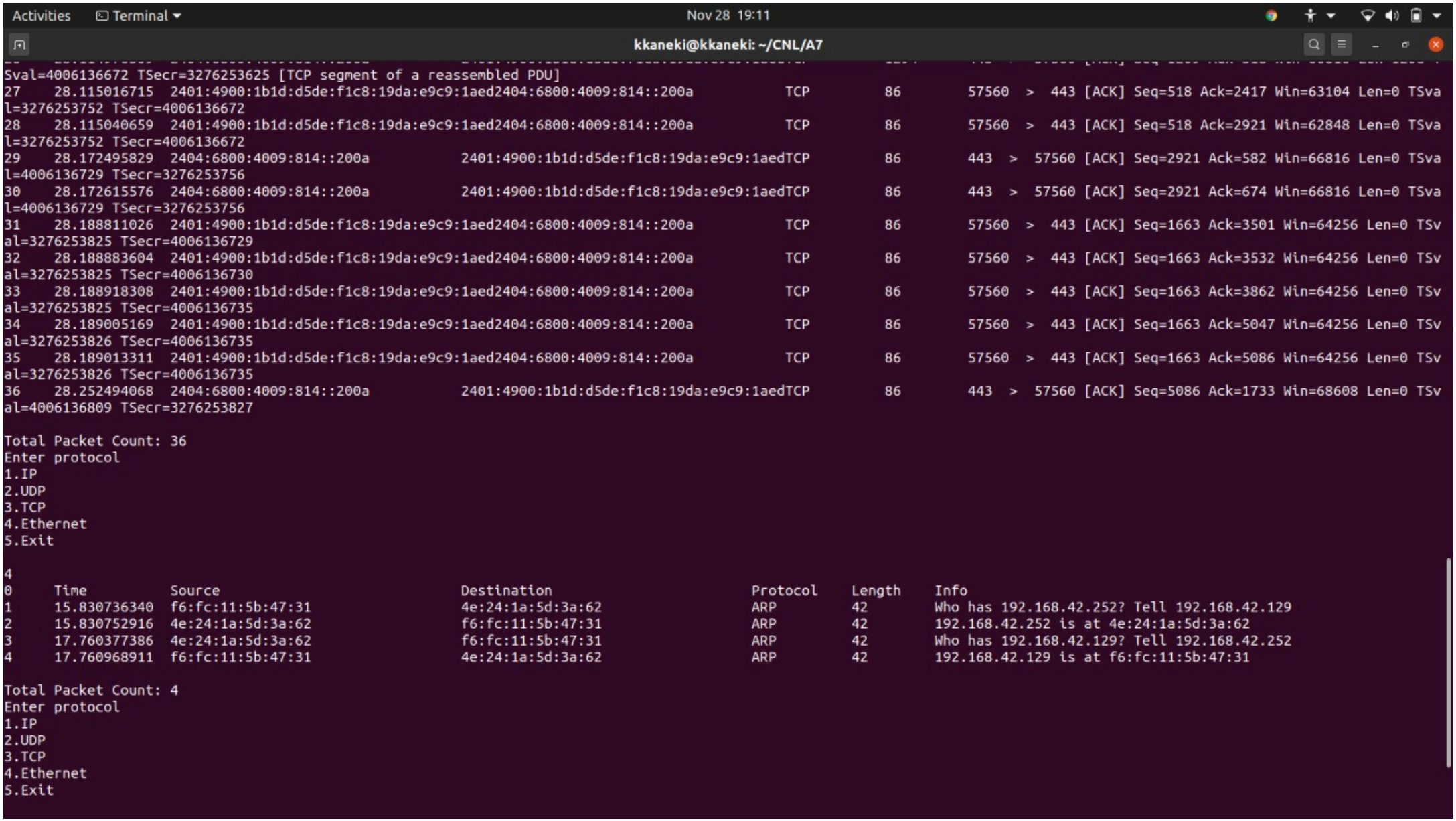
Because UDP is significantly more limited in compatability than TCP, its headers are much smaller. A UDP header contains 8 byte, divided into the following four required fields

1) Source Port number (2 bytes)
2) Destination port number (2 byte).
3) Length of data (2 bytes).
4) UDP checksum (2 bytes).

Conclusion:

We studied & implemented program to analyze following packet format captured the wireshark. 1. Ethernet 2. IP 3. TCP 4. UDP.


Activities
Terminal
Nov 28 19:10
kkaneki@kkaneki: ~/CNL/A7
kkaneki@kkaneki:~/CNL/A7$ g++ A7.cpp
kkaneki@kkaneki:~/CNL/A7$ ./a.out
Enter protocol
1.IP
2.UDP
3.TCP
4.Ethernet
5.Exit
1
0 Time Source Destination Protocol Length Info
1 2.286932470 fe80::f4fc:11ff:fe5b:4731 ff02::1:ff17:d3ce ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:dc11:f4d4:5c17:d3ce from f6:fc:11:5b:47:31
2 3.318136294 fe80::f4fc:11ff:fe5b:4731 ff02::1:ff17:d3ce ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:dc11:f4d4:5c17:d3ce from f6:fc:11:5b:47:31
3 4.342142642 fe80::f4fc:11ff:fe5b:4731 ff02::1:ff17:d3ce ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:dc11:f4d4:5c17:d3ce from f6:fc:11:5b:47:31
4 12.759147799 fe80::f4fc:11ff:fe5b:4731 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed from f6:fc:11:5b:47:31
5 12.759210879 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed fe80::f4fc:11ff:fe5b:4731 ICMPv6 78 Neighbor Advertisement 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed (sol)
6 23.532874342 fe80::f4fc:11ff:fe5b:4731 ff02::1:ff17:d3ce ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:dc11:f4d4:5c17:d3ce from f6:fc:11:5b:47:31
7 24.535199513 fe80::f4fc:11ff:fe5b:4731 ff02::1:ff17:d3ce ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:dc11:f4d4:5c17:d3ce from f6:fc:11:5b:47:31
8 25.559502798 fe80::f4fc:11ff:fe5b:4731 ff02::1:ff17:d3ce ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:dc11:f4d4:5c17:d3ce from f6:fc:11:5b:47:31
9 33.239547026 fe80::f4fc:11ff:fe5b:4731 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed ICMPv6 86 Neighbor Solicitation for 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed from f6:fc:11:5b:47:31
10 33.239652999 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed fe80::f4fc:11ff:fe5b:4731 ICMPv6 78 Neighbor Advertisement 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed (sol)
Total Packet Count: 10
Enter protocol
1.IP
2.UDP
3.TCP
4.Ethernet
5.Exit
2
0 Time Source Destination Protocol Length Info
Total Packet Count: 0


Activities
Terminal
Nov 28 19:11
kkaneki@kkaneki: ~/CNL/A7
Sval=4006136672 TSecr=3276253625 [TCP segment of a reassembled PDU]
27 28.115016715 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=518 Ack=2417 Win=63104 Len=0 TSval=3276253752 TSecr=4006136672
28 28.115040659 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=518 Ack=2921 Win=62848 Len=0 TSval=3276253752 TSecr=4006136672
29 28.172495829 2404:6800:4009:814::200a 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aedTCP 86 443 > 57560 [ACK] Seq=2921 Ack=582 Win=66816 Len=0 TSval=4006136729 TSecr=3276253756
30 28.172615576 2404:6800:4009:814::200a 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aedTCP 86 443 > 57560 [ACK] Seq=2921 Ack=674 Win=66816 Len=0 TSval=4006136729 TSecr=3276253756
31 28.188811026 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=1663 Ack=3501 Win=64256 Len=0 TSval=3276253825 TSecr=4006136729
32 28.188883604 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=1663 Ack=3532 Win=64256 Len=0 TSval=3276253825 TSecr=4006136730
33 28.188918308 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=1663 Ack=3862 Win=64256 Len=0 TSval=3276253825 TSecr=4006136735
34 28.189005169 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=1663 Ack=5047 Win=64256 Len=0 TSval=3276253826 TSecr=4006136735
35 28.189013311 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aed2404:6800:4009:814::200a TCP 86 57560 > 443 [ACK] Seq=1663 Ack=5086 Win=64256 Len=0 TSval=3276253826 TSecr=4006136735
36 28.252494068 2404:6800:4009:814::200a 2401:4900:1b1d:d5de:f1c8:19da:e9c9:1aedTCP 86 443 > 57560 [ACK] Seq=5086 Ack=1733 Win=68608 Len=0 TSval=4006136809 TSecr=3276253827
Total Packet Count: 36
Enter protocol
1.IP
2.UDP
3.TCP
4.Ethernet
5.Exit
4
0 Time Source Destination Protocol Length Info
1 15.830736340 f6:fc:11:5b:47:31 4e:24:1a:5d:3a:62 ARP 42 Who has 192.168.42.252? Tell 192.168.42.129
2 15.830752916 4e:24:1a:5d:3a:62 f6:fc:11:5b:47:31 ARP 42 192.168.42.252 is at 4e:24:1a:5d:3a:62
3 17.760377386 4e:24:1a:5d:3a:62 f6:fc:11:5b:47:31 ARP 42 Who has 192.168.42.129? Tell 192.168.42.252
4 17.760968911 f6:fc:11:5b:47:31 4e:24:1a:5d:3a:62 ARP 42 192.168.42.129 is at f6:fc:11:5b:47:31
Total Packet Count: 4
Enter protocol
1.IP
2.UDP
3.TCP
4.Ethernet
5.Exit