

*a. Functions and responsibilities of the Physical Layer:*

*Examine the types of signals being transmitted, such as Ethernet or Wi-Fi, by looking at the protocol information in the captured packets.*

Frame 151607: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface en0, id 0

```

Section number: 1
> Interface id: 0 (en0)
Encapsulation type: Ethernet (1)
Arrival Time: Jul 27, 2023 18:10:25.627129000 EDT
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1690495825.627129000 seconds
[Time delta from previous captured frame: 0.062976000 seconds]
[Time delta from previous displayed frame: 0.062976000 seconds]
[Time since reference or first frame: 542.455618000 seconds]
Frame Number: 151607
Frame Length: 66 bytes (528 bits)
Capture Length: 66 bytes (528 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ethertype:ip:tcp]
[Coloring Rule Name: Bad TCP]
[Coloring Rule String: tcp.analysis.flags && !tcp.analysis.window_update && !tcp.analysis.keep_alive &..

```

Time	Source	Destination	Protocol	Length	Info
151... 2023-07-27 18:10:25.551911	2607:f8b0:4006:8d2...	2601:19c:4e81:5d50::	QUIC	88	Protected Payload (KP#)
151... 2023-07-27 18:10:25.557498	2607:f8b0:4006:8d2...	2601:19c:4e81:5d50::	QUIC	139	Protected Payload (KP#)
151... 2023-07-27 18:10:25.561369	2607:f8b0:4006:8d2...	2601:19c:4e81:5d50::	QUIC	139	Protected Payload (KP#)
151... 2023-07-27 18:10:25.561369	2601:19c:4e81:5d50::	2607:f8b0:4006:8d2...	QUIC	96	Protected Payload (KP#), DCID=df8f4537bc6d0742
151... 2023-07-27 18:10:25.561830	169.254.215.141	239.255.255.250	SSDP	223	M-SEARCH * HTTP/1.1
151... 2023-07-27 18:10:25.562764	169.254.215.141	239.255.255.250	SSDP	227	M-SEARCH * HTTP/1.1
151... 2023-07-27 18:10:25.563171	169.254.215.141	239.255.255.250	SSDP	231	M-SEARCH * HTTP/1.1
151... 2023-07-27 18:10:25.564153	169.254.215.141	239.255.255.250	SSDP	224	M-SEARCH * HTTP/1.1
151... 2023-07-27 18:10:25.627129	52.22.119.135	10.0.0.43	TCP	66	[TCP Dup ACK 612#1] 443 → 50140 [ACK] Seq=1 Ack=1 Win=8 Len=0 TSval=251673944 TSecr=2986
151... 2023-07-27 18:10:25.627527	10.0.0.43	52.22.119.135	TCP	66	[TCP Dup ACK 613#35] 50140 → 443 [ACK] Seq=1 Ack=2 Win=2048 Len=0 TSval=2986217386 TSecr=2986
151... 2023-07-27 18:10:25.863837	BROADCAST	BROADCAST	ARP	60	Who has 169.254.100.1? Tell 169.254.227.43
151... 2023-07-27 18:10:25.928966	2601:19c:4e81:5d50::	2607:f8b0:4006:8d2...	UDP	91	59454 → 443 Len=29
151... 2023-07-27 18:10:25.932290	2607:f8b0:4006:8d2...	2601:19c:4e81:5d50::	UDP	205	443 → 59454 Len=143
151... 2023-07-27 18:10:25.941351	2601:19c:4e81:5d50::	2607:f8b0:4006:8d2...	UDP	95	59454 → 443 Len=33
151... 2023-07-27 18:10:25.952125	2607:f8b0:4006:8d2...	2601:19c:4e81:5d50::	UDP	88	443 → 59454 Len=26
151... 2023-07-27 18:10:26.118740	2601:19c:4e81:5d50::	2606:4700:6812:7b7	TLSv1..	578	Application Data
151... 2023-07-27 18:10:26.132934	2606:4700:6812:7b7	2601:19c:4e81:5d50::	TCP	86	443 → 50115 [ACK] Seq=7141 Ack=5020 Win=13 Len=0 TSval=2932221245 TSecr=3771741185
151... 2023-07-27 18:10:26.171847	169.254.215.141	239.255.255.250	SSDP	423	NOTIFY * HTTP/1.1
151... 2023-07-27 18:10:26.223205	2606:4700:6812:7b7	2601:19c:4e81:5d50::	TLSv1..	1047	Application Data
151... 2023-07-27 18:10:26.223326	2601:19c:4e81:5d50::	2606:4700:6812:7b7	TCP	86	50115 → 443 [ACK] Seq=5020 Ack=8102 Win=2032 Len=0 TSval=3771741290 TSecr=2932221335

> Frame 151607: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface en0, id 0

- Ethernet II, Src: Vantivall\_86:73:31 (98:9d:5d:86:73:31), Dst: Apple\_30:ca:fe (f4:0f:24:30:ca:fe)
  - Destination: Apple\_30:ca:fe (f4:0f:24:30:ca:fe)
  - Source: Vantivall\_86:73:31 (98:9d:5d:86:73:31)
  - Type: IPv4 (0x0800)
- Internet Protocol Version 4, Src: 52.22.119.135, Dst: 10.0.0.43
  - 0100 .... = Version: 4
  - .... 0101 = Header Length: 20 bytes (5)
  - Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  - Total Length: 52
  - Identification: 0x025e (606)
  - 010. .... = Flags: 0x2, Don't fragment
  - ...0 0000 0000 0000 = Fragment Offset: 0
  - Time to Live: 44
  - Protocol: TCP (6)
  - Header Checksum: 0x969e [validation disabled]
  - [Header checksum status: Unverified]
  - Source Address: 52.22.119.135
  - Destination Address: 10.0.0.43
- Transmission Control Protocol, Src Port: 443, Dst Port: 50140, Seq: 1, Ack: 1, Len: 0

Internet Protocol Version 4

Packets: 151819 · Displayed: 151819 (100.0%) · Dropped: 247 (0.2%) Profile: Default

*b. Transmission media:*

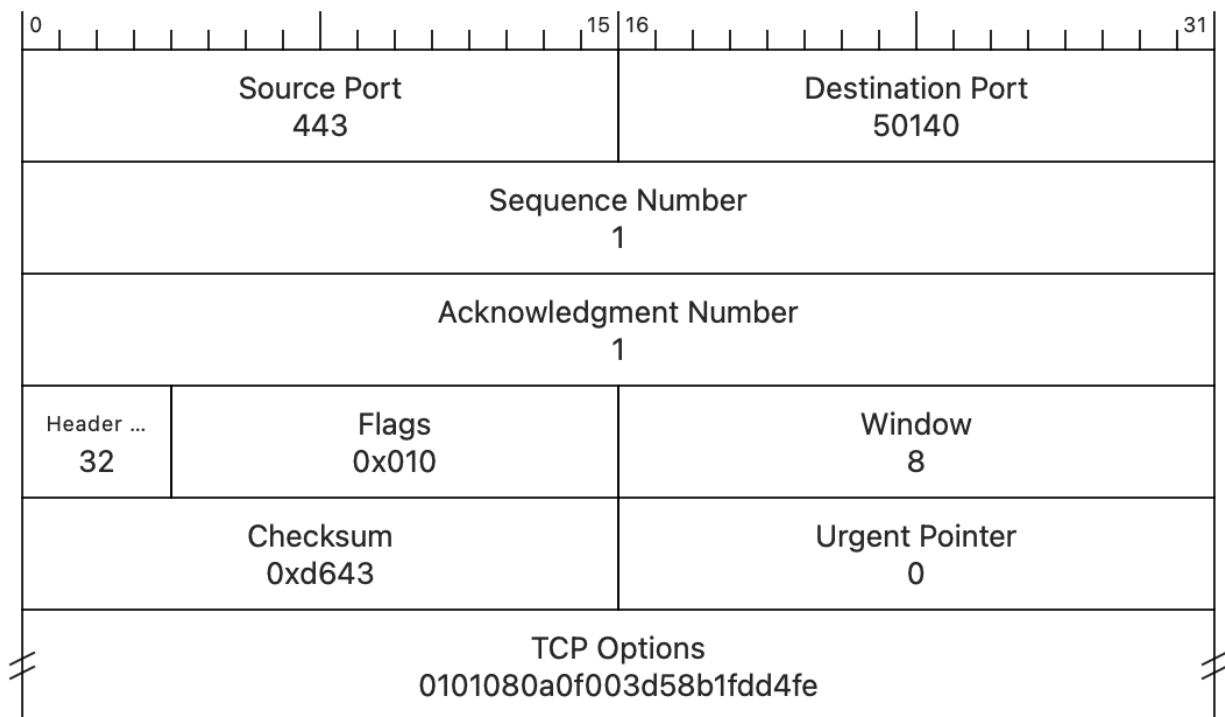
Observe the types of transmission media being used, such as wired (copper or fiber optic) or wireless (radio or infrared), by examining the network interface used for capturing the packet's protocol information.

Frame > Interface ID > Indicated On Mac OS, "en0", "en1", ...: Ethernet

*c. Error detection and correction methods:*

*Investigate the captured packets for evidence of error detection and correction methods, such as checksums or parity bits, by examining the packet details and researching the specific methods used by the identified protocols.*

## Transmission Control Protocol



Checksum above

- a. A brief introduction to the Physical Layer and its functions and responsibilities
- b. Analysis of the transmission media observed in the captured traffic
- c. A discussion of error detection and correction methods observed in the captured traffic

—

The Physical Layer is responsible for the transmission of raw bits over a communication channel. By examining the captured packets, we can identify the types of signals being transmitted. For instance, Ethernet or Wi-Fi protocols indicate the presence of wired or wireless transmission within the network. The Physical Layer ensures that the transmitted signals adhere to specified standards and regulates physical characteristics such as voltage levels, signal timing, and data encoding.

Examining the captured packets allowed me to explore the implementation of error detection and correction methods. Error detection mechanisms, such as checksums or parity bits, are used to identify transmission errors. These methods enhance the reliability of data transmission by automatically recovering from detected errors or requesting retransmission of corrupted packets.