

# Task 5 : Capture and Analyze Network Traffic Using Wireshark

**Objective:** Capture live network packets and identify basic protocols and traffic types.

**Tools:** Wireshark (free).

**Deliverables:** A packet capture (.pcap) file and a short report of protocols identified.

Step 1 — Visited a website named wikibook to generate traffic

The screenshot shows a web browser window with the following details:

- Title Bar:** Canadian History/The Geography of the Lands
- Address Bar:** en.wikibooks.org/wiki/Canadian\_History/The\_Geography\_of\_the\_Lands
- Header:** WIKIBOOKS Open books for an open world
- Search Bar:** Search Wikibooks
- Right Side:** Donations, Create account, Log in, [dismiss]
- Content Area:**
  - Left Sidebar (Contents):** Beginning, Geography of the Lands (with sub-links: Coastal Plains, Interior Plains, Canadian Shield, Arctic Islands, Western Mountain Region, Great Lakes and St. Lawrence Lowland, Appalachian Highlands of Canada)
  - Main Content:** Canadian History/The Geography of the Lands
  - Sub-Content:** Geography of the Lands [edit | edit source]
  - Description:** Canada is the second largest country in the world, encompassing 9,970,610 km<sup>2</sup> of land. It is surrounded by three oceans, the Arctic, the Pacific, and the Atlantic. Canada is comprised of ten provinces and three territories, the provinces all lying in the South, and the territories in the North. Canada's climate is fairly moderate, with temperate weather to Arctic conditions in the North. It also has a mild terrain, generally having flatlands, with mountainous areas in the West, and lowlands in the East. There are eight physical regions in Canada, the Appalachian Region, the Coastal Plains, the Great Lakes-St. Lawrence Lowlands, the Interior Plains, the Canadian Shield, the Western Cordillera, the Intermountain Region, and the Arctic. When Europeans came to Canada, they altered the physical environment.
  - Right Sidebar (Appearance):** Text (Small, Standard, Large), Width (Standard, Wide), Color (beta) (Automatic, Light, Dark)

Step 2 :- Wireshark is installed in my system so I started packet capturing

The screenshot shows the Wireshark interface capturing network traffic on the 'Wi-Fi' interface. The packet list pane displays the following information:

No.	Time	Source	Destination	Protocol	Length	Info
213	16.210135	2404:6800:4002:828...:2401:4900:884b:7417...	QUIC	994	Protected Payload (KP0)	
214	16.210135	2404:6800:4002:828...:2401:4900:884b:7417...	QUIC	154	Protected Payload (KP0)	
215	16.211246	2401:4900:884b:7417...	2404:6800:4002:828...: QUIC	169	Protected Payload (KP0), DCID=e09e52e45236560f	
216	16.211417	2401:4900:884b:7417...	2404:6800:4002:828...: QUIC	93	Protected Payload (KP0), DCID=e09e52e45236560f	
217	16.2320840	2404:6800:4002:828...:2401:4900:884b:7417...	QUIC	86	Protected Payload (KP0)	
218	16.234527	2404:6800:4002:828...:2401:4900:884b:7417...	QUIC	182	Protected Payload (KP0)	
219	16.234778	2401:4900:884b:7417...	2404:6800:4002:828...: QUIC	93	Protected Payload (KP0), DCID=e09e52e45236560f	
220	16.260062	2404:6800:4002:828...:2401:4900:884b:7417...	QUIC	88	Protected Payload (KP0)	
221	16.517224	34.174.255.69	192.168.1.7	TLSv1.3	93	Application Data
222	16.518127	192.168.1.7	34.174.255.69	TCP	54	53801 → 443 [FIN, ACK] Seq=766 Ack=5031 Win=254 Len=0
223	16.824347	34.174.255.69	192.168.1.7	TLSv1.3	78	Application Data
224	16.824347	34.174.255.69	192.168.1.7	TCP	54	443 → 53801 [FIN, ACK] Seq=5055 Ack=767 Win=485 Len=0
225	16.824433	192.168.1.7	34.174.255.69	TCP	54	53801 → 443 [RST, ACK] Seq=767 Ack=5055 Win=0 Len=0
226	17.746238	192.168.1.5	239.255.255.250	UDP	77	55508 → 15600 Len=35
227	18.462795	44.242.131.93	192.168.1.7	TCP	66	[TCP Retransmission] 443 → 62685 [SYN, ACK] Seq=0 Ack=1 Win=41496 Len=0 MSS=1258 SACK_PERM ...

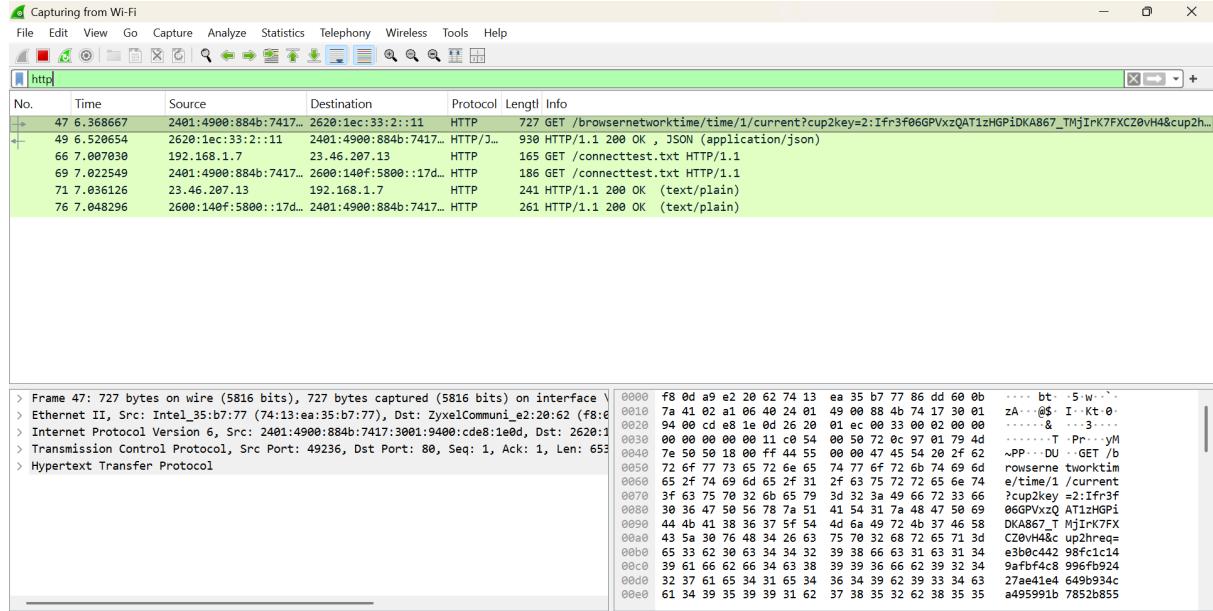
The packet details and bytes panes show the raw hex and ASCII data for the selected packet (No. 225). The status bar at the bottom indicates:

```
> Frame 1: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface \Device\NPF_{...}
```

Three different protocols have been observed in this generic packet capture:

- TCP
- UDP
- QUIC

After applying HTTP filter ■



## 1. HTTP Traffic Findings

The HTTP-filtered packets clearly show **unencrypted HTTP communication**, allowing full visibility into requests and responses.

### Observed details:

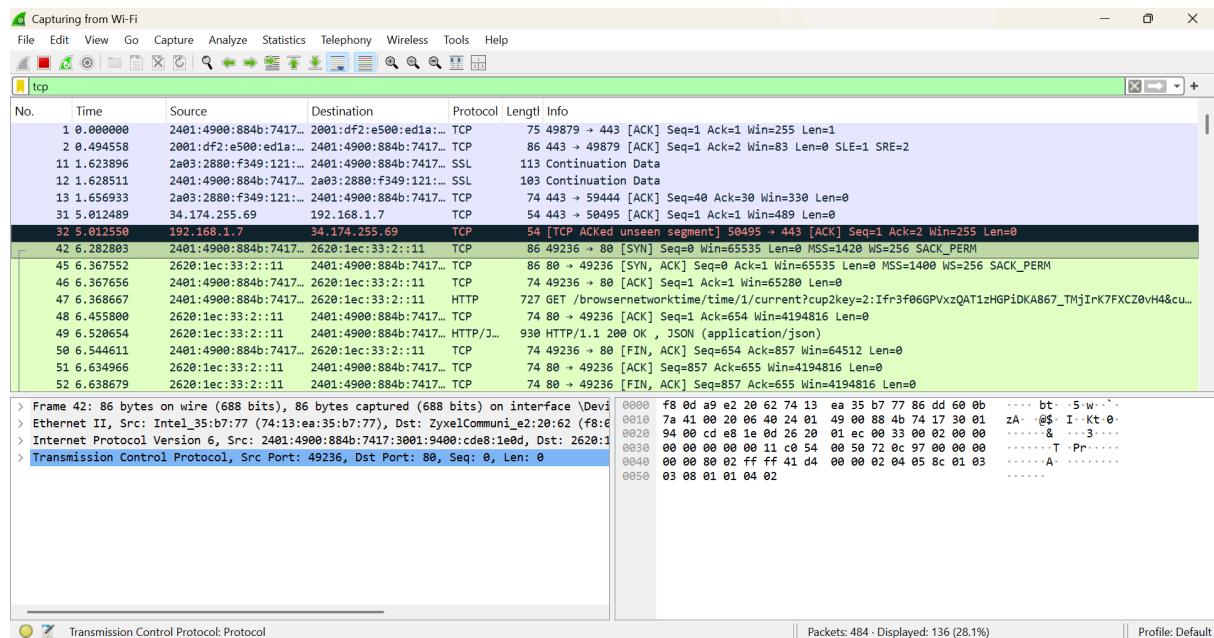
- HTTP **GET requests** were observed, including:
  - Requests to `/connecttest.txt`
  - Requests to `/browsernetworktime/time/1/current`
- HTTP responses included:
  - **HTTP/1.1 200 OK**
  - Content types such as `text/plain` and `application/json`
- HTTP traffic was exchanged over **TCP port 80**

- Packet payloads were readable, highlighting the lack of encryption in HTTP traffic.

### Inference:

HTTP traffic confirms successful application-layer communication and demonstrates why HTTP is considered insecure compared to HTTPS.

After applying TCP filter ■



## 2. TCP Traffic Findings

The TCP-filtered packets reveal both **HTTP and HTTPS-related TCP sessions**, showing reliable, connection-oriented communication.

### Observed details:

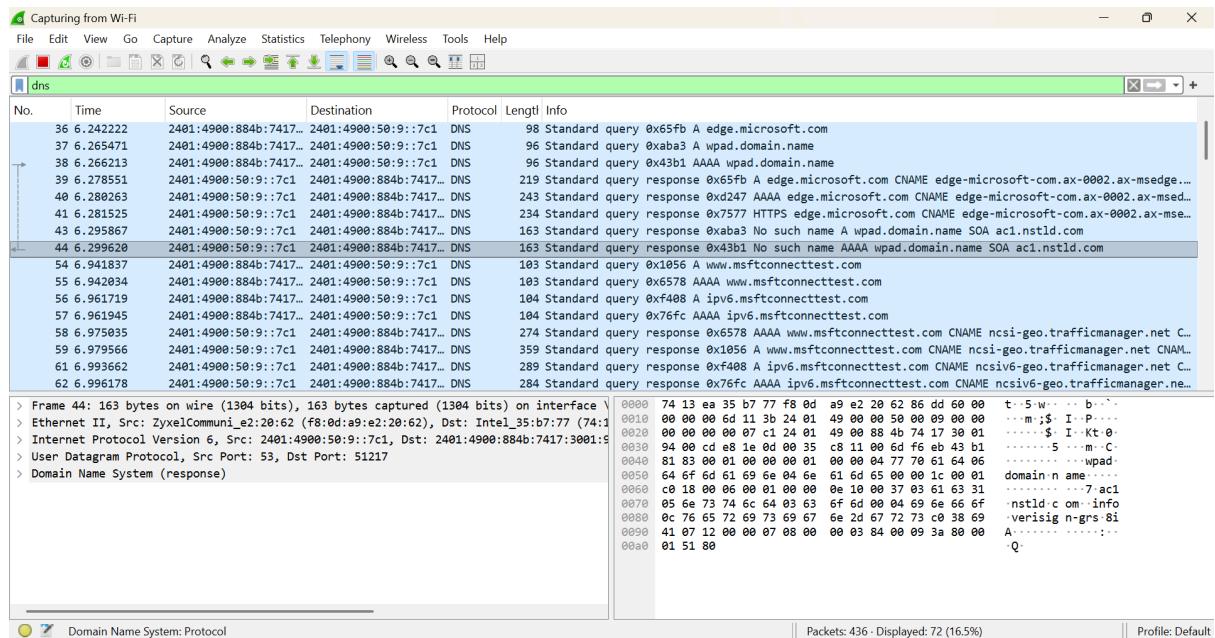
- TCP **three-way handshake** was clearly visible:
  - **SYN → SYN, ACK → ACK**
- TCP connections were established on:
  - **Port 80** for HTTP traffic
  - **Port 443** for encrypted (TLS/SSL) traffic

- TCP session lifecycle was observed:
  - Connection establishment
  - Data transfer
  - Graceful connection termination using FIN, ACK
- An instance of “TCP ACKed unseen segment” was observed, which can occur due to:
  - Packet loss
  - Capture starting mid-session
  - Out-of-order packet arrival

#### Inference:

TCP ensured reliable data delivery and session management for application-layer protocols such as HTTP and HTTPS.

After applying dns filter —



### 3. DNS Traffic Findings

The DNS-filtered view shows multiple **standard DNS queries and responses**, primarily related to Microsoft connectivity and service checks.

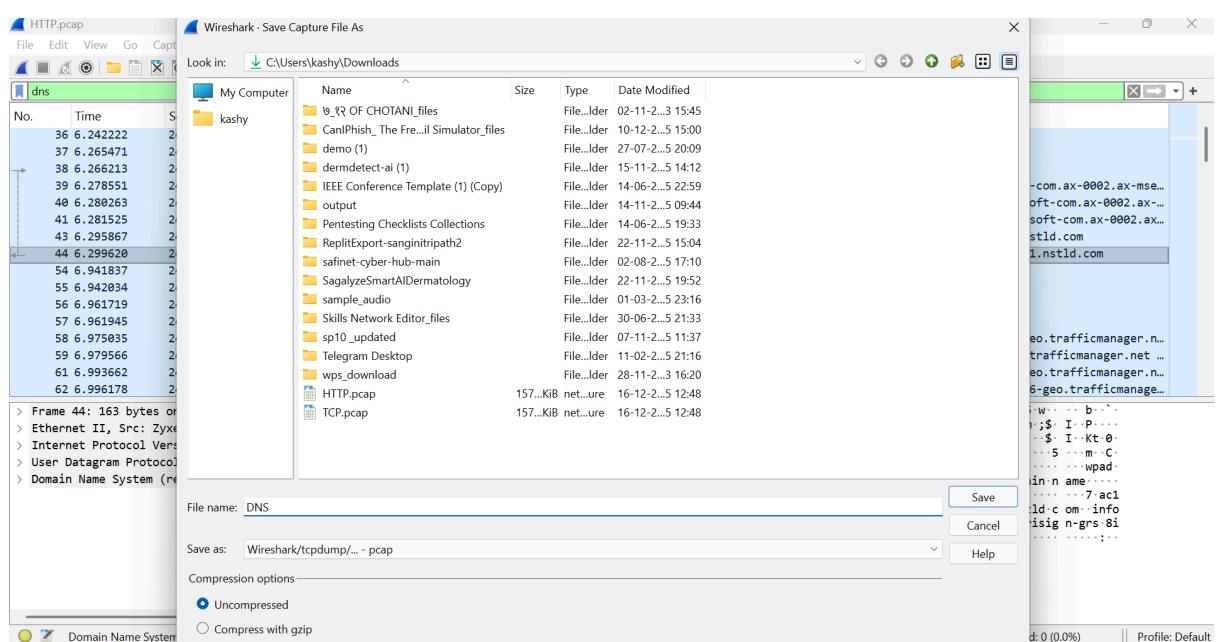
## Observed details:

- DNS queries for domains such as:
  - `edge.microsoft.com`
  - `www.msftconnecttest.com`
  - `ipv6.msftconnecttest.com`
  - `wpad.domain.name`
- Both **A (IPv4)** and **AAAA (IPv6)** record queries were observed.
- DNS responses include:
  - Successful resolutions using **CNAME records** (e.g., traffic manager domains)
  - **No such name** responses for `wpad.domain.name`, which is a common and normal behavior when WPAD is not configured.
- DNS communication occurred over **UDP port 53**, confirming standard DNS resolution behavior.

### Inference:

DNS traffic confirms that the system was actively resolving domain names before initiating HTTP/TCP connections, which is the first step in web communication

And Lastly all the files were saved as .pcap file.



## Summary

### Overall Analysis and Conclusion

- The captured traffic accurately demonstrates the **end-to-end flow of network communication**:
  1. **DNS** resolves domain names to IP addresses
  2. **TCP** establishes reliable connections
  3. **HTTP** transfers application data
- Both **IPv4 and IPv6 traffic** were observed, indicating a dual-stack network environment.
- The traffic captured represents **normal, legitimate system and browser activity**, including connectivity checks and web requests.
- Wireshark filtering proved effective in isolating protocols and analyzing packet behavior.