NAME: YANKISO MASOEBE

DATA COM WEEK 10

1. Capture Analysis

Using Wireshark, I analyzed a packet capture to observe real-time network communication through the TCP/IP model. The capture displayed various protocols including TCP, DNS, and HTTP, each corresponding to specific layers of the model. First, I identified a **TCP three-way handshake**, which establishes a reliable connection between two hosts. The process began with a **SYN** packet from the client, followed by a **SYN-ACK** from the server, and finally an **ACK** from the client. This interaction occurs at the **Transport layer**, ensuring a reliable communication channel before data transmission begins.

Next, I observed a **DNS query**, where the client sent a request to resolve a domain name (e.g., www.example.com) into an IP address. The DNS server replied with the corresponding IP address. This process operates primarily at the **Application layer**, while relying on **UDP** at the **Transport layer** for faster, connectionless delivery.

2'TCP vs. UDP Comparison Table

Protocol Type	Application	Reason for Using TCP or UDP
TCP	**Web Browsing (HTTP/HTTPS)**	Uses TCP because web pages require reliable, ordered delivery of data to display correctly.
TCP	**Email (SMTP/IMAP/POP3)**	Emails must be transmitted completely and accurately, making TCP's reliability essential.
TCP	**File Transfer (FTP)**	Ensures all file data is received in the correct order with no loss or corruption.
TCP	**Remote Login (SSH/Telnet)**	Reliable and secure transmission is needed for accurate command input and response.
TCP	**Online Banking Applications**	Requires guaranteed and error-free communication to maintain data integrity and security.
UDP	**Online Gaming**	Uses UDP for low latency; minor packet loss is acceptable to maintain real-time gameplay.
UDP	**Video Streaming (YouTube, Netflix)**	Prioritizes continuous playback and speed over perfect accuracy of each packet.
UDP	**Voice over IP (VoIP)**	Delivers audio in real time; dropped packets are preferable to delays caused by retransmission.
UDP	**DNS Queries**	Fast, connectionless lookups make UDP ideal for short, single-packet exchanges.
UDP	**Live Broadcasting (Twitch, IPTV)**	Real-time data flow is more important than perfect packet delivery, so UDP minimizes del

Scenario Troubleshooting

In this scenario, the user cannot connect to **www.example.com** but can access other websites using their **IP addresses** directly. This indicates that the problem is most likely occurring at the **Application layer** of the **TCP/IP model**, specifically involving the **Domain Name System (DNS)** protocol. DNS is responsible for translating domain names into IP addresses. Since connections work when using IP addresses, the lower layers of the TCP/IP model — such as the **Network (Internet)** and **Transport** layers — are functioning correctly. This means the computer can send and receive packets, and protocols like **TCP** or **UDP** are not the cause of the issue.

The failure most likely lies in **name resolution**, where the DNS server cannot properly respond to the query for www.example.com. Possible causes include an incorrect DNS server configuration, a corrupted DNS cache, or a temporary outage at the DNS provider. To troubleshoot, one could use commands such as **nslookup** or **ping** to test domain resolution, clear the DNS cache, or manually set a reliable DNS server like **8.8.8.8 (Google DNS)**.