UNDERSTANDING TCP/IP AND UDP PROTOCOLS

VAISHALI SHISHODIA

1. Overview of TCP/IP and UDP

TCP/IP (Transmission Control Protocol / Internet Protocol)

- A suite of **communication protocols** that interconnects network devices on the internet.
- It defines how data should be **packetized**, **addressed**, **transmitted**, **routed**, and **received**.
- Consists of several layers:
    - **Application Layer** (HTTP, FTP, DNS)
    - **Transport Layer** (TCP, UDP)
    - **Internet Layer** (IP, ICMP)
    - **Link Layer** (Ethernet, ARP)

UDP (User Datagram Protocol)

- A **lightweight**, connectionless protocol used for situations where **speed is critical** and occasional packet loss is acceptable.
- Commonly used in **real-time applications** like:
    - Video/audio streaming
    - Online gaming
    - VoIP (Voice over IP)
    - DNS queries

2. How TCP Works

Feature	Description
Connection-Oriented	TCP initiates a connection through a **three-way handshake** before transmitting data.
Reliable	TCP guarantees delivery of packets in the correct order. If a packet is lost, it's retransmitted.
Error Checking	TCP uses **checksum**, **ACKs**, and **sequence numbers** to detect and recover from errors.
Flow Control	TCP employs mechanisms like **Sliding Window Protocol** to ensure the receiver is not overwhelmed.
Congestion Control	Uses algorithms like **TCP Tahoe**, **Reno**, **Cubic** to manage traffic load.

TCP 3-Way Handshake

1. **Client to Server**: Sends SYN (synchronize) packet to initiate a connection.
2. **Server to Client**: Replies with SYN-ACK.
3. **Client to Server**: Sends ACK to confirm the connection is established.

After this handshake, **data transfer begins**. When the session ends, a **4-step termination process** occurs using FIN and ACK flags.

3.How UDP Works

Feature	Description
Connectionless	No handshake. Data is sent immediately without verifying recipient availability.
Lightweight	Header is only 8 bytes (compared to TCP's 20+ bytes). Less overhead.
No Reliability	No retransmission. No ACKs. Ideal for apps where **speed > reliability**.
Multicast Support	UDP supports multicast transmission, which is beneficial for streaming.
Best-Effort Delivery	Delivery is not guaranteed. Some packets may get dropped or arrive out of order.

Use cases: **DNS**, **VoIP**, **TFTP**, **NTP**, **Streaming services**.

4. Common TCP Attacks

Attack Name	Description	Impact
SYN Flood	Attacker sends numerous SYN requests but never completes the handshake. Server waits for ACK, exhausting resources.	Denial of Service (DoS)
TCP Reset (RST) Attack	An attacker sends spoofed TCP RST packets to prematurely terminate a connection.	Session termination, Data loss
Session Hijacking	Intercepting or predicting session tokens or sequence numbers to take control of a session.	Unauthorized access
ACK Storm	Spoofed ACK packets lead to constant ACKs between devices, congesting the network.	Network disruption

Attack Name	Description	Impact
TCP Injection	Injecting malicious data into an existing TCP stream by predicting sequence numbers.	Data corruption, Control manipulation

5. Common UDP Attacks

Attack Name	Description	Impact
UDP Flood	Overwhelms target with high volume of UDP packets to random ports.	Exhausts server resources
DNS Amplification	Exploits DNS servers to reflect and amplify traffic towards a victim. Small requests = huge responses.	Massive DDoS
NTP/SNMP Amplification	Sends spoofed requests to NTP/SNMP servers to amplify attack bandwidth.	Bandwidth exhaustion
UDP Port Scanning	Sends empty UDP packets to various ports to discover open ones based on ICMP error messages.	Reconnaissance
Fraggle Attack	Similar to Smurf attack but uses UDP echo. Spoofs broadcast messages with victim's IP.	Network congestion

6. Mitigation Strategies (SOC Perspective)

TCP Attack Mitigation

Attack	Mitigation Strategies
SYN Flood	Enable **SYN cookies**, configure **firewall rate limiting**, use **reverse proxies**, deploy **load balancers**.
TCP Reset	Use **TLS encryption** to authenticate sessions, configure **ACLs** to drop suspicious RST packets.
Session Hijacking	Use **HTTPS**, implement **IPsec**, monitor for abnormal session behavior.
TCP Injection	Sanitize inputs, monitor for abnormal TCP sequences using IDS like **Snort**, use **TLS** to encrypt traffic.

UDP Attack Mitigation

Attack	Mitigation Strategies
UDP Flood	Block unnecessary UDP services, implement **rate limiting**, configure **anti-DDoS protections**.
DNS/NTP Amplification	Disable open recursive DNS, configure NTP with **noquery** or restrict default settings, implement **BCP38** (ingress filtering).
SNMP Amplification	Use **SNMPv3**, restrict SNMP access, block UDP port 161 from external sources.
Port Scanning	Use **port knocking**, configure **IPS/IDS**, block ICMP responses where possible.

7. SOC Analyst's Role in TCP/UDP Monitoring

Task	Actions/Tools
Log Analysis	Review firewall logs, system logs, and application logs for anomalies. Use **SIEM** platforms like **Splunk**, **ELK Stack**, or **QRadar**.
Anomaly Detection	Deploy **UEBA** (User and Entity Behavior Analytics), look for traffic spikes, unusual port activity.
Packet Analysis	Use **Wireshark**, **tcpdump** to capture and inspect network packets in detail. Look for malformed packets, spoofed IPs, etc.
Threat Hunting	Use **threat intelligence feeds**, search for known IOCs, investigate abnormal flows or sessions.
Incident Response	Identify source IPs, isolate affected hosts, create rules to block malicious traffic, inform stakeholders, document incident.

8. Tools for Analyzing TCP/UDP Traffic

Tool	Purpose
Wireshark	Deep packet inspection with filtering capabilities.
tcpdump	CLI-based tool for quick capture and filtering.
Nmap	Network scanner to identify open ports, services, and vulnerabilities.
Zeek (Bro)	Network traffic analyzer for security monitoring and logging.
NetFlow/sFlow	Provides metadata about IP traffic, helpful in traffic pattern analysis.

Tool	Purpose
Suricata/Snort	IDS/IPS tools that can detect known attack signatures in TCP/UDP traffic.

Conclusion

- TCP ensures **reliable and ordered** delivery but is **slower** due to its overhead.
- UDP provides **speed and efficiency** but lacks reliability and ordering.
- Each protocol has specific use cases and **security risks**.
- As a SOC Analyst, your job is to **detect**, **analyze**, and **respond** to attacks using TCP/UDP traffic analysis, logs, and threat intelligence tools.

9. Scenario-Based Interview Questions and Answers

Q1: A server is facing intermittent disconnections. Packet captures show many RST packets. What might be happening?

A: This could indicate a TCP Reset (RST) attack. The attacker may be sending spoofed RST packets to disrupt ongoing connections. Investigate packet source IPs and analyze sequence numbers. Use TLS or firewall rules to drop suspicious RSTs.

Q2: You see a spike in DNS traffic and outbound traffic to random IPs. What could this be?

A: Likely a DNS Amplification Attack or malware using DNS for data exfiltration. Check for large DNS responses and sources sending abnormal amounts of small queries. Mitigation includes disabling open recursion and enabling rate limiting.

Q3: How would you identify a SYN flood attack using logs or packet capture?

A: Look for excessive SYN packets without corresponding ACKs. High number of half-open connections. Use tools like Wireshark or review firewall logs. Employ SYN cookies or rate-limiting rules.

Q4: A system is running slow, and UDP packets are coming from many IPs to port 80. What's your next step?

A: Investigate for UDP flood or reflection attack. Port 80 (HTTP) usually uses TCP, so receiving UDP is suspicious. Check for amplification patterns and configure firewall to drop unwanted UDP.

Q5: A user complains of slow video streaming. How do you check whether it’s a TCP or UDP issue?

A: Identify the application protocol (e.g., RTP usually uses UDP). Use packet captures to check for packet loss or out-of-order packets. If it’s TCP, check retransmissions and congestion. If UDP, check jitter and loss.

Q6: How do you differentiate between a normal UDP port scan and malicious intent?

A: A normal scan might be from a vulnerability scanner. Malicious scans are stealthy, repetitive over time, or targeting sensitive ports (e.g., SNMP, TFTP). Review historical logs and use IDS alerts to correlate behavior.
