**TCP/UDP Vulnerabilities and Benefits**

Adam Livingston

University of Arizona

CYBV 326: Introductory Methods of Network Analysis

Professor Jonathan Martinez

14 November 2021

**TCP/UDP Vulnerabilities and Benefits**

According to Kurose & Ross, “the internet makes two distinct transport-layer protocols available to the application layer” (p.191). Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) have certain features that require consideration from application developers when specifying which protocol they wish to use in their software. Generally, TCP “provides a reliable connection-oriented service to the invoking application”, while UDP “provides an unreliable, connectionless service to the invoking application.” (p.191). At a glance of these generalities, reliability concerns come to mind and as almost everyone buying a car would state, reliability is the most critical consideration. For others, speed is most important. The application of these protocols in internet-based software is critical in the purpose of the software. Reliability is most important in email, file transfer, and web browsing whereas UDP is most applicable to Zoom meetings, video streaming services, or online gaming (vpnMentor).

As with any protocol used, there are benefits that must be weighed against inherent vulnerabilities. This paper will explore those vulnerabilities, what can be done to mitigate or prevent them, benefits of TCP versus UDP, Internet Control Message Protocol attacks, Ping Commands, Address Resolution Protocol (ARP), and changing a Media Access Control Address (MAC).

**TCP and UDP Vulnerabilities**

Both TCP and UDP protocols are vulnerable to exploitation. In some instances, some of the perceived strengths of the services that either provide are subject to exploit by a determined attacker. Cybersecurity professionals must take necessary steps to either mitigate or prevent altogether the method, opportunity, or motive of attack. Below are just three of many known vulnerabilities associated with TCP or UDP protocols and prevention/mitigation strategies that can be implemented with each.

**Vulnerability 1:** **Exploiting the TCP Connection Mechanism.** Since TCP requires a connection in order to provide service, it requires that the client and host go through a process known as a “three-way handshake” to establish the connection. The client requests a connection via a small TCP segment (synchronize, aka. SYN), the host acknowledges and responds with a small TCP segment (synchronize-acknowledge, SYNACK), and the client confirms (acknowledge, ACK) (Kurose & Ross. p.97). A SYN Flood attack exploits this establishment mechanism. The attacker sends massive amounts of SYN packets to every port on the target which in turn causes the host server, thinking these are legitimate connection requests, to respond with SYNACK packets to establish the connection (Imperva, TCP SYN Flood). The attacker doesn’t send the final ACK and never confirms the connection. Before the connection times out from the host’s port, another SYN packet will be sent and overload the system, causing malfunctions, crashes, and/or denying services to legitimate clients. This is a type of Distributed Denial of Service (DDoS) attack that targets the Availability principle of the CIA triad (Pfleeger, p.11).

Security professionals have a number of methods to mitigate such an attack, but still rely on the “network’s ability to handle large-scale volumetric DDoS attacks” (Imperva, TCP SYN Flood). One such method is using RST (reset) cookies. A server can initially and intentionally send an invalid SYNACK that results in the client IP Address generating an RST packet. If the server receives this from the client, then it knows the request is legitimate. This method acts as a secondary screening measure that filters spoofed IP addresses and other botnets attempting the SYN Flood Attack.

**Vulnerability 2: Exploiting UDP Connectionless Property.** Unlike TCP connections between hosts and clients, UDP does not require a three-way handshake. This allows UDP protocol to run with relatively low overhead, which is preferential for, but not limited to streaming video, Voice over IP (VoIP), and gaming applications (Imperva, UDP Flood). Without the requirement of a handshake “to establish a valid connection, a high volume of ‘best effort’ traffic can be sent over UDP channels to any host” (Imperva, UDP Flood). Unfortunately, the nature of UDP makes it highly susceptible to what is known as a UDP Flood Attack. Like a SYN Flood Attack, a UDP Flood Attack is a type of DDoS, an attack that targets the Availability principle of the CIA triad (Pfleeger, p.11). A UDP Flood is characterized by the attacker flooding the host with invalid UDP packets looking for closed ports. When a port is closed, it will respond with an “[Internet Control Message Protocol (ICMP)] port unreachable” message (CCNA). When all ports are closed or there is too much traffic for the host to handle, most of the bandwidth is used up and results in a DDoS or DoS. What makes UDP Flood Attacks so malicious is that they are easily executed by Script Kiddies with open-source software, like the Low Orbit Ion Cannon (LOIC) (CCNA).

The most basic way to mitigate the enormous use of bandwidth caused by the sending “ICMP Port Unreachable” messages is by limiting the rate of ICMP responses from the host (Imperva, UDP Flood). If the host is capped on the number of ICMP responses, then it can be deduced that the bandwidth required to send those responses is lowered and therefore maintains the Availability principle. On the other hand, “such indiscriminate [sic] filtering will have an impact on legitimate traffic” (Imperva, UDP Flood), so cybersecurity professionals must find an appropriate balance of security versus availability through such measures.

**Vulnerability 3: TCP Session Hijacking.** “TCP session hijacking is a security attack on a user session over a protected network” (GeeksforGeeks, Session Hijacking). One such method to perform an attack is known as IP Spoofing, where an attacker disguises themselves as an already authenticated user. “In implementing this technique, [the] attacker has to obtain the IP address of the client and inject his own packets spoofed with the IP address of client into the TCP session” (GeeksforGeeks, Session Hijacking). Obtaining the IP address and authentication data can be done with a packet sniffer. With this authentication camouflage a malicious attacker can gain otherwise unauthorized access to a server to perform malfeasance, making it look like an already authenticated and legitimate user did the attack.

To prevent or mitigate session hijacking, cybersecurity professionals should ensure that packets are ciphered in order to prevent the obtainment of any information that would aid in IP spoofing. “This encryption can be provided by using protocols such as IPSEC, SSL, SSH etc. Internet security protocol (IPSEC) has the ability to encrypt the packet on some shared key between the two parties involved in communication” (GeeksforGeeks. Session Hijacking).

**Benefits of TCP Over UDP**

As discussed above, both TCP and UDP have key differences that will influence their application when developers are choosing which protocol to use. This means that the benefits of TCP are subjective when compared to UDP and is all dependent on the desired functionality of the program.

If the application being developed needs guaranteed delivery of data, a connection-based protocol, extensive error checking mechanisms, and connection acknowledgment through the three-way handshake, then TCP is beneficial. (GeeksforGeeks, TCP vs UDP). Using TCP for sending files and other data would be one application.

However, using UDP is comparatively faster than TCP and is considered “lightweight”. No connection (handshake) is needed and faster, simpler, and more efficient than TCP (GeeksforGeeks, TCP vs UDP). UDP is ideal for broadcasting, such as Zoom calls, video streaming from YouTube or Netflix, as well as online gaming. Some packets may be lost in the transmission, but the high rate of packet flow from UDP makes the packet loss negligible, if not unnoticeable from the client-side experience.

**How a TCP Connection Ends**

Since TCP requires a handshake to connect two systems together, to end the connection either the host or the client must signal that they wish to end it. The graceful way to end a TCP connection is through the FIN-ACK process. In the graphic below borrowed from University of Arizona CYBV 326 class week 5 slides, the entity wishing to close the connection sends a segment with the TCP flag of FIN, indicating that they wish to terminate the connection. The receiving entity acknowledges it and sends a FIN flag of their own, which in turn is acknowledged by the initiator. Once this process is complete, the connection is closed.

A close up of text on a white background

Description generated with high confidence

Another way that a TCP connection can close is through an abrupt shutdown. As discussed above in the section about SYN Flood Attacks, an RST flag can be sent by the receiving entity to screen out potential attackers. This RST flag can also be sent to suddenly close a TCP connection due to lack of resources from the host or if the host is unreachable and has stopped responding (GeeksforGeeks, TCP Connection).

**ICMP Attack**

An Internet Control Message Protocol (ICMP) attack (also known as a Ping Flood Attack) is a type of DDoS that occurs at the network layer (Kurose and Ross p.419) by overwhelming the *R* access rate of the server. According to Kurose and Ross, “if the server has and access rate of *R* bits per second, then the attacker will need to send traffic at a rate of approximately *R* bps to cause damage.” Specifically, the attacker will attempt to overwhelm a device or server with ICMP echo-requests (pings). “By flooding the target with request packets, the network is forced to respond with an equal number of reply packets. This causes the target to become inaccessible to normal traffic.” (Netscout). Typically, DDoS attacks are carried out by botnets (Pfleeger, p.426).

This attack hopes to violate the Availability principle of the CIA Triad by overwhelming: the capacity to meet the service’s needs, timely response to requests, and completion time for services (Pfleeger p.11).

An example of such a Ping Flood Attack is given in Kurose and Ross on p.139 where a botnet was used to target DNS servers. Fortunately, the attack in question caused minimal damage due to in part packet filters that blocked ICMP ping messages at the root servers. It is recommended that to protect or mitigate damage from such an attack would be to have this packet filtering technology installed to block ICMP ping messages in order to prevent attacks such as this, or to disable “the ICMP functionality of the targeted router, computer or other device.” (Netscout) Another method would be to have firewall filters block such request from outside the organization in order to maintain internally-conducted ping requests (Netscout).

**Ping Command Not Working**

There are multiple reasons why a ping command might not work when trying to reach a destination. Since the ping command uses ICMP, as described above one such reason why a ping command might not work is that the ping destination has filters in place that blocks such requests or that the functionality has been disabled. Also mentioned above, a ping command might not work outside of an intranet or VLAN is that the firewall in place of the destination address blocks such commands.

A second reason derives from the error message “Destination Host Unreachable.” “This usually indicates a problem with routing. The local computer may not be configured with the correct default gateway, the remote computer may not be configured with the correct default gateway, or a router between the two may be misconfigured or faulty.” (sourceDaddy).

A third reason derives from another error message, “TTL Expired in Transit.” “If the Time to Live (TTL) value is lower than the number of routers the ping must pass through to reach its destination, the ping packet is discarded.” (sourceDaddy).

**Address Resolution Protocol (ARP) Attacks**

One type of ARP attack is known as “ARP Spoofing”. This is where an attacker sends falsified ARP messages and results in the attacker’s MAC address being linked with the legitimate IP address of a computer or server on the attacked network (Veracode) (Note: MAC addresses are the common term for link-layer addresses (Kurose & Ross, p.489)). This allows the attacker to receive any data intended for the legitimate IP address and to do what they please with it.

In an article published on Networkworld.com in 2008, an ARP Spoofing attack is reported to have occurred against the Metasploit Project. Ironically, the Metasploit Project is an open-source penetration testing software (Metasploit) and the hackers used the ARP Spoofing attack to redirect all network traffic to a defaced website. Thankfully for the owner of Metasploit, the original website was left intact which also gave him an idea of the point of attack. The hacktivists had conducted this ARP Spoofing attack by redirecting traffic on the VLAN to another device, which was rectified by the owner by “hardcoding his service provider’s MAC address into his ARP cache.” (Networkworld).

**Changing a MAC Address**

A MAC address is an identification code for real-world, physical devices such as routers, PCs, and other hardware. It can metaphorically be described as the device’s social security number, so changing the address for the average person typically not advised. To change a MAC Address, one can download software tools such as Technitium MAC Address Changer (digitaltrends) or follow these steps:

* On windows, go to Device Manager and look for “Network Adapters”
* Choose the network interface you wish to change
* Right click, select “Properties”, select “Advanced”
* Select “MAC Address” and enter in a value

One way an attacker could use this functionality is to spoof the MAC address of a known and authenticated device on a network they wish to attack. Since MAC addresses are unique and typically are cataloged by competent institutions for security purposes (digitaltrends), it can be deduced that spoofing a MAC address could grant an attacker unfettered access to the target network depending on the permissions granted to the spoofed MAC address. In conjunction with spoofing the IP Address of the corresponding MAC address, an attacker could essentially make a digital clone of a device with access to the network and act within the access controls put in place for that device.

**Conclusion**

TCP and UDP are both highly effective protocols depending on their use within the transport layer of the OSI model and each have their strengths as well as drawbacks. But as stated, those strengths and weaknesses are amplified by their application to a program’s specific purpose as well as mitigation measures put in place to defend against their inherent characteristics. ARP attacks are also a threat in that protocol, and changing MAC addresses in the link layer can allow attackers another avenue of attack on vulnerable networks. There are seemingly limitless possibilities in which malicious attackers can exploit inherent and subtle vulnerabilities in networks, and it is up to those tasked as cybersecurity professionals to stay ahead of the curve.

**References**

Kurose, J. F., & Ross, K. W. (2022). *Computer networking: A top-down approach*. Pearson.

Pfleeger, C. P. (2015). *Security in computing*.

*TCP VS UDP: Understanding the difference*. vpnMentor. (n.d.). Retrieved November 11, 2021, from https://www.vpnmentor.com/blog/tcp-vs-udp/.

*What is a TCP syn flood: Ddos attack glossary: Imperva*. Learning Center. (2020, September 30). Retrieved November 12, 2021, from https://www.imperva.com/learn/ddos/syn-flood/.

*What is a UDP flood: Ddos attack glossary: Imperva*. Learning Center. (2020, September 30). Retrieved November 12, 2021, from https://www.imperva.com/learn/ddos/udp-flood/.

3, C. C. N. A., & read, C. C. N. A. 3·5 min. (2020, September 1). *▷ TCP and UDP vulnerabilities " CCNA 200*. 301. Retrieved November 12, 2021, from https://ccna-200-301.online/tcp-and-udp-vulnerabilities/.

*Session hijacking*. GeeksforGeeks. (2018, February 15). Retrieved November 13, 2021, from https://www.geeksforgeeks.org/session-hijacking/.

*Differences between TCP and UDP*. GeeksforGeeks. (2021, October 25). Retrieved November 13, 2021, from https://www.geeksforgeeks.org/differences-between-tcp-and-udp/.

*TCP connection termination*. GeeksforGeeks. (2021, October 26). Retrieved November 13, 2021, from https://www.geeksforgeeks.org/tcp-connection-termination/.

*What is an ICMP flood ddos attack?* NETSCOUT. (n.d.). Retrieved November 13, 2021, from https://www.netscout.com/what-is-ddos/icmp-flood.

*Troubleshooting Connectivity with Ping*. sourceDaddy. Troubleshooting Connectivity with ping - Networking Tutorial. (n.d.). Retrieved November 14, 2021, from https://sourcedaddy.com/networking/troubleshooting-connectivity-with-ping.html.

*ARP spoofing*. Veracode. (n.d.). Retrieved November 14, 2021, from https://www.veracode.com/security/arp-spoofing.

Jongsma, C. (2008, June 8). *Surprise Arp attack draws attention*. Network World. Retrieved November 14, 2021, from https://www.networkworld.com/article/2280475/surprise-arp-attack-draws-attention.html.

Martin, D. (2021, March 9). *How to change your mac address*. Digital Trends. Retrieved November 14, 2021, from https://www.digitaltrends.com/computing/how-to-change-mac-address/.