Foundations of Networking Protocols: TCP, UDP, and IP Explained for Newcomers in Cyber Security

Ahsen Beyza Özkul

ahsenbeyza@securedebug.com

Contents

NETWORK PROTOCOLS **3**

WHAT IS IP? **3**

IP ADDRESSING 3
IP ROUTING 3
HOW IP ROUTING WORKS *4*
COMMON IP VULNERABILITIES 5

WHAT IS TCP? **5**

HOW TCP MAKES SURE DATA ARRIVES CORRECTLY 6
THE THREE-WAY HANDSHAKE 6
MANAGING DATA FLOW AND AVOIDING CONGESTION 7
COMMON ISSUES: TCP VULNERABILITIES 7

WHAT IS UDP? **7**

HOW DOES UDP WORK? 8
WHEN IS UDP USED? 8
COMMON UDP-BASED ATTACKS 9

TCP AND UDP DİFFERENCES **9**

REFERENCES **11**

Network Protocols

Networking protocols are like the rules that devices follow to communicate with each other over a network, such as the internet. Just as people need to speak the same language to understand each other, devices need to use the same protocol to exchange information. These rules ensure that data can travel from one device to another accurately and securely, whether you're sending an email, streaming a video, or browsing the web.

In cybersecurity, understanding networking protocols is essential because these rules are the foundation of how data is shared and protected. Knowing how these protocols work helps cybersecurity professionals protect networks from threats, such as hackers trying to steal information. For example, secure protocols like HTTPS keep your data safe when you're shopping online or logging into your bank account.
Without a good grasp of networking protocols, it would be difficult to secure a network or troubleshoot problems. That’s why learning about these protocols is a key part of staying safe online and protecting digital information.

What is IP?

IP stands for Internet Protocol. It’s a set of rules that allows devices like computers, phones, and tablets to communicate with each other over the internet. An IP address is like a digital home address for your device, letting it send and receive information from other devices online.

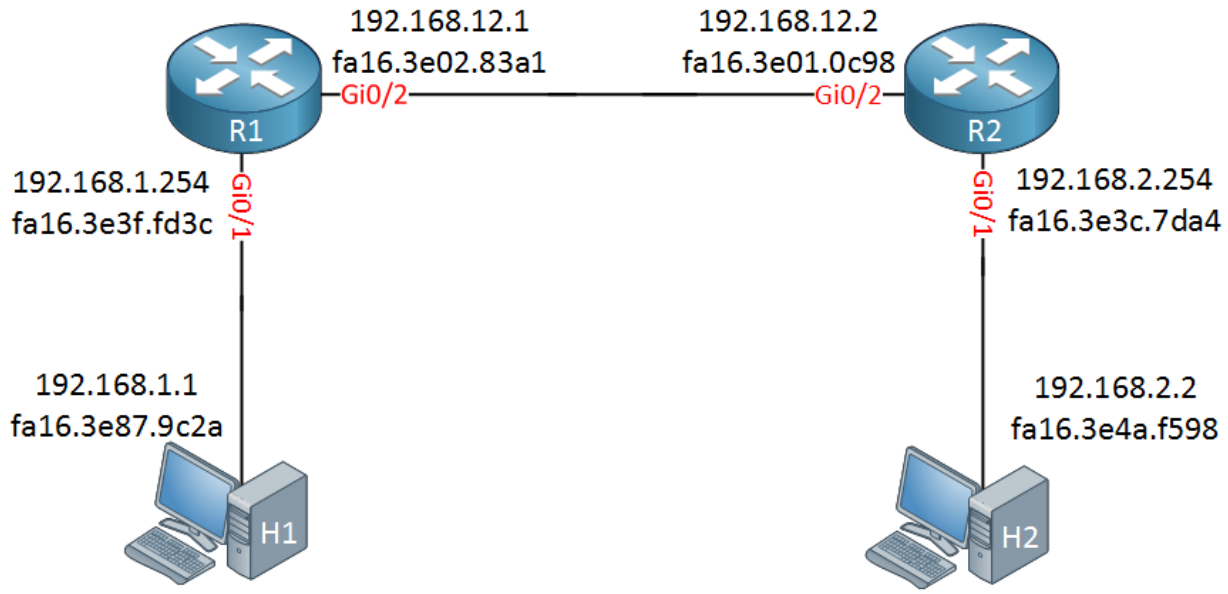
IP Addressing

An IP address is a unique series of numbers that identifies each device on a network, similar to a home address in the real world.

- Private IP Address: This is the address your device uses within your home or office network. For example, your laptop, phone, and printer each have a private IP address so they can connect and communicate with each other.
- Public IP Address: This is the address your entire network (like your home Wi-Fi) uses when connecting to the internet. When you visit a website, this public IP address is what the website sees.

IP Routing

IP routing is how data travels across the internet from one device to another. When you send a message or visit a website, the data is broken down into small pieces called packets. Each packet has your IP address and the destination IP address on it. Routers, like traffic managers on the internet, read these addresses and guide the packets through the best path across different networks until they reach their destination. Once they arrive, the packets are put back together into the original message or webpage.



How IP Routing Works

IP routing is like a mail delivery system for the internet. Here's how it works, step by step:

1. Your Device's Decision

When your device wants to send data (like loading a website or sending an email), it first needs to decide where to send it:

- Same Network:
  - If the destination is on the same network (like another device in your home), your device checks a list (called the ARP table) to see if it already knows the MAC address of the destination.
  - If it finds the MAC address, it sends the data directly to that device.

- Different Network:
  - If the destination is outside your local network (like a website on the internet), your device sends the data to your router (the gateway).
  - It checks the ARP table for your router's MAC address and sends the data there.

2. What the Router Does

Once your router gets the data, it has to figure out where to send it next. Here's what happens:

- Check for Errors:

    - The router first checks to make sure the data isn't corrupted. If it is, the router drops the data (doesn't send it).

- Check the Address:
    - The router looks at the destination address to see if the data is meant for it, or if it should be passed along to another device.

- Find the Best Path:
    - The router looks at its own map (called a routing table) to find the best route to the destination. It decides which path the data should take next.

- Prepare the Data:
    - The router slightly modifies the data to prepare it for the next step of its journey. It decreases the Time to Live (TTL), which is like an expiration date for data. If the TTL runs out, the data won't be sent further.

- Send the Data:
    - Finally, the router sends the data along the next path, whether it's directly to the destination or to another router along the way.

This process happens again and again as the data passes through different routers until it finally reaches its destination. Each router helps move the data closer to where it needs to go, just like different mail carriers help deliver a letter to the right address.

Common IP Vulnerabilities

While IP is essential for internet communication, there are some risks:

- IP Spoofing: This is when a hacker pretends to be someone else by using a fake IP address, which can trick systems into accepting harmful data.
- IP Address Tracking: Your IP address can be used to monitor your online activity or even figure out your location, which could lead to unwanted attention or attacks.
- DDoS Attacks: This stands for Distributed Denial of Service, where hackers flood a network with too much data, causing it to crash. They can do this by targeting the IP addresses of vulnerable devices.

What is TCP?

The Transmission Control Protocol (TCP) is like the traffic manager for data on the internet. Imagine you're sending a letter through the mail. TCP is like the postal service that ensures your letter gets to the right place, safely and in the right order.

How TCP Makes Sure Data Arrives Correctly

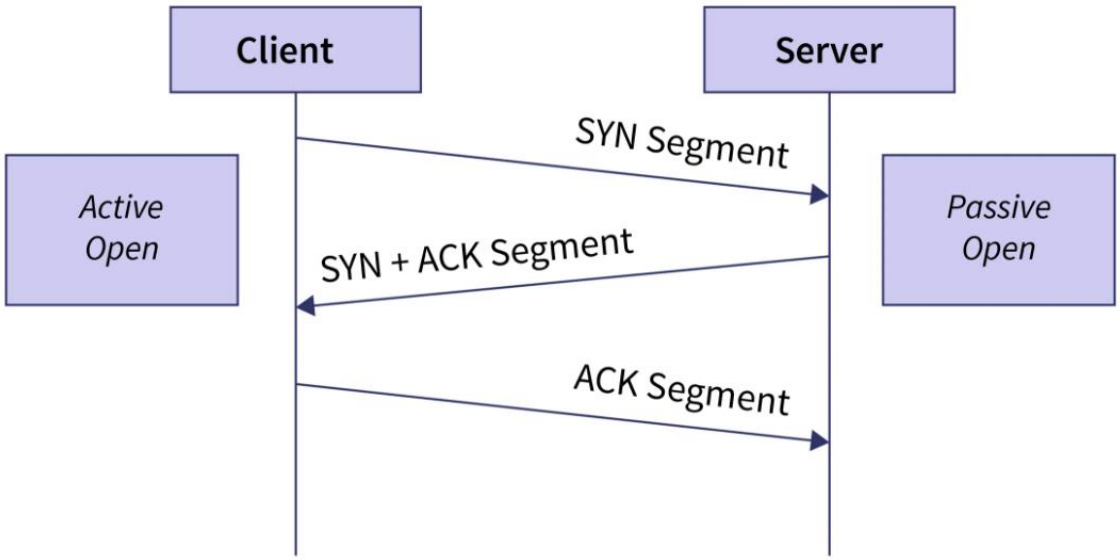
Like we said when you send data over the internet, it's broken up into small pieces called packets. TCP's job is to make sure these packets are delivered correctly:

1. Organizes Data: TCP takes your data and splits it into packets. Each packet is given a number, so TCP knows which order they should be in when they reach the other end.

2. Checks for Errors: TCP includes a special code in each packet called a checksum. This helps check if the packet got damaged during the journey. If a packet arrives with errors, TCP can ask for it to be sent again.

3. Puts Data Back Together: When all the packets arrive at their destination, TCP reassembles them in the right order to recreate your original data.

The Three-Way Handshake



Before data can be sent, TCP needs to establish a connection. Think of it like a handshake before a conversation. Here’s how it works:

1. SYN (Synchronize): The sender says, “Hi, I want to start a conversation,” by sending a special message called a SYN.

2. SYN-ACK (Synchronize-Acknowledge): The receiver responds, “Hi back, I’m ready to talk,” with a message that acknowledges the sender’s request and includes its own SYN.

3. ACK (Acknowledge): Finally, the sender confirms, “Great, let’s start!” by sending an acknowledgment message.

This handshake ensures both sides are ready and agree on how to communicate.

Managing Data Flow and Avoiding Congestion

TCP also manages how quickly data is sent to prevent overload:

1. Windowing: Imagine a conveyor belt moving boxes (data packets). TCP controls the size of the conveyor belt (window size) based on how quickly the receiver can handle the boxes. It adjusts this size to keep things running smoothly.

2. Slow Start: When a connection begins, TCP starts by sending data slowly to avoid overwhelming the network. It then gradually increases the speed as long as everything is working well.

3. Congestion Control: If the network gets crowded, TCP slows down the data flow to prevent jams and collisions, similar to reducing traffic speed in a congested area.

Common Issues: TCP Vulnerabilities

Sometimes, attackers try to disrupt TCP connections. One common attack is:

- SYN Flooding: The attacker sends a flood of SYN messages to a server but never completes the handshake. This overloads the server, making it unable to handle real connections.

To protect against such attacks, various techniques are used, like checking for valid connections and limiting the number of connection attempts.

What is UDP?

The User Datagram Protocol (UDP) is a way to send data across the internet. Think of it like sending a series of postcards instead of a letter. Each postcard (or "datagram") is sent independently, and there's no guarantee they all arrive in order or even at all. This makes UDP fast but less reliable compared to other protocols like TCP.

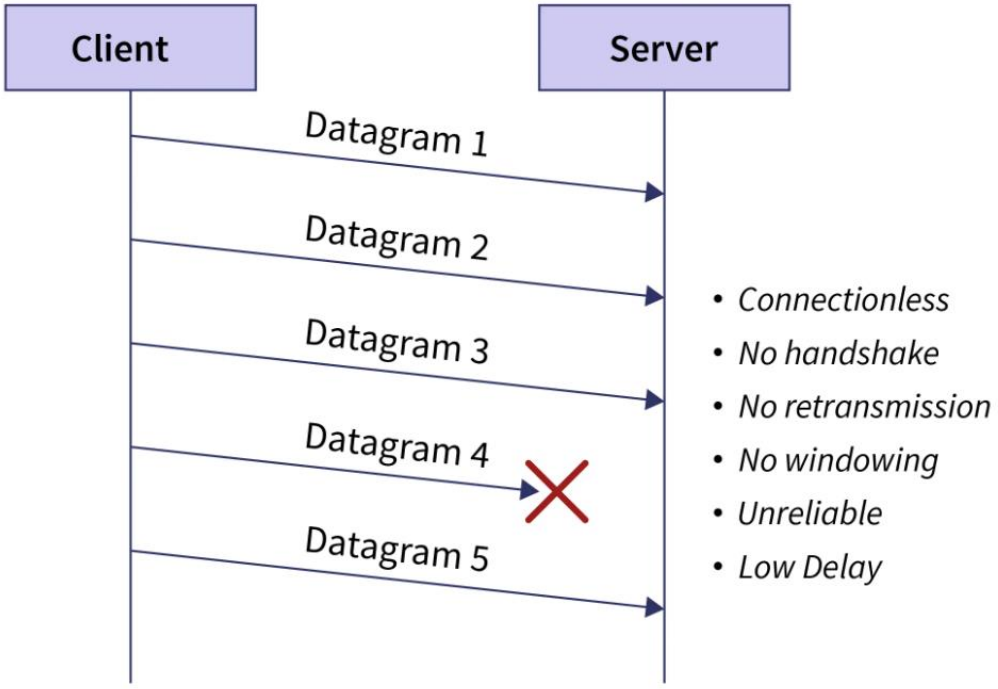
How Does UDP Work?

UDP is designed for speed and simplicity:

1. No Connection Setup: Unlike TCP, UDP doesn't establish a connection before sending data. It just sends packets directly to the destination without a handshake.

2. No Order or Error Checking: UDP packets don’t have to arrive in order, and UDP doesn’t check if packets arrived correctly. There’s no built-in acknowledgment or retransmission if packets get lost.

3. Packets Are Independent: Each packet (or datagram) is sent individually, so packets can take different routes and may arrive out of order or get lost.



When is UDP Used?

UDP is great for applications where speed is more important than perfect reliability:

1. Video Streaming: In video streaming, such as watching a movie online, it's better to have a few lost packets than to wait for them to be retransmitted. The video will play smoother with a few glitches than if it had to wait for every packet to be resent.

2. Online Gaming: For online games, real-time interaction is crucial. A small delay caused by waiting for packet acknowledgment can ruin the gaming experience. UDP allows the game to continue running smoothly even if some packets are lost.

3. Voice Over IP (VoIP): VoIP services like internet phone calls use UDP because a slight delay or lost packet is less disruptive than having the call lag or stutter.

4. DNS Lookups: When you look up a website, DNS servers use UDP to quickly respond to queries. The speed of UDP is beneficial here, and missing a few DNS queries doesn't usually cause major issues.

Common UDP-Based Attacks

UDP's speed can be exploited in several ways:

1. UDP Flood Attack: Attackers overwhelm a server with numerous UDP packets, causing it to respond with error messages and potentially crash or slow down.

2. DNS Amplification Attack: Attackers send small UDP requests to DNS servers with a spoofed victim's IP address, leading to large responses that flood and overwhelm the victim.

3. UDP Port Scan: Attackers send UDP packets to various ports to discover which ones are open. Open ports can then be targeted for further attacks.

TCP and UDP Differences

1. Connection:
   - TCP (Transmission Control Protocol): Connection-oriented. Establishes a connection between sender and receiver before data transfer begins.
   - UDP (User Datagram Protocol): Connectionless. Sends data without establishing a connection or ensuring the receiver is ready.

2. Reliability:
   - TCP: Reliable. Ensures that data packets are delivered in order, without duplication or loss, using acknowledgments and retransmissions.
   - UDP: Unreliable. Does not guarantee delivery, order, or error-checking. Packets can be lost or arrive out of sequence.

3. Speed:
   - TCP: Slower due to connection setup, error checking, and retransmissions.
   - UDP: Faster since it skips the connection setup and error-checking processes.

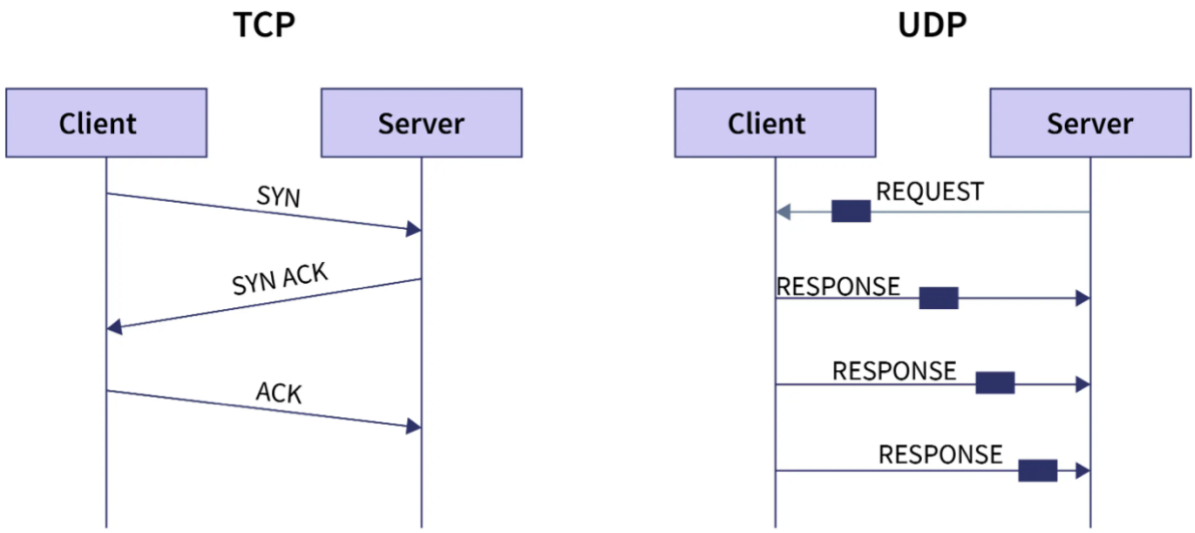
4. Error Handling:

- TCP: Provides error correction with acknowledgments and retransmissions. It ensures data integrity and correct sequence.
- UDP: Provides minimal error checking with checksums. It does not correct errors or reorder packets.

5. Use Cases:
   - TCP: Used for applications where reliability is crucial, such as web browsing (HTTP/HTTPS), email (SMTP/IMAP), and file transfers (FTP).
   - UDP: Used for time-sensitive applications where speed is more critical than reliability, such as video streaming, VoIP, and online gaming.

6. Overhead:
   - TCP: Higher overhead due to connection management, error checking, and retransmissions.
   - UDP: Lower overhead since it lacks connection management and error recovery mechanisms.

7. Data Ordering:
   - TCP: Guarantees that data packets are received in the order they were sent.
   - UDP: Does not guarantee packet order, so packets may arrive out of sequence.



References

- Somani, S. (2023, October 23). *TCP vs UDP: Understanding the differences*. Scaler. https://www.scaler.com/topics/computer-network/tcp-vs-udp/
- Karel. (2023, September 1). *TCP ve UDP arasındaki farklar nedir?* Karel. https://www.karel.com.tr/bilgi/tcp-ve-udp-arasindaki-farklar-nedir
- Cloudflare. (n.d.). *User Datagram Protocol (UDP)*. Retrieved September 2, 2024, from https://www.cloudflare.com/learning/ddos/glossary/user-datagram-protocol-udp/
- JavaTpoint. (n.d.). *UDP protocol*. Retrieved September 2, 2024, from https://www.javatpoint.com/udp-protocol
- HAProxy. (2024, May 13). *What is User Datagram Protocol (UDP)?* HAProxy. https://www.haproxy.com/glossary/what-is-user-datagram-protocol-udp
- Allied Telesis. (n.d.). *What is TCP?* Allied Telesis. Retrieved September 2, 2024, from https://www.alliedtelesis.com/tr/en/foundations/what-is-tcp
- Pramatarov, M. (2023, November 22). *TCP: Transmission Control Protocol—What is it and how does it work?* CloudNS. https://www.cloudns.net/blog/tcp-transmission-control-protocol-what-is-it-and-how-does-it-work/
- Kaspersky. (n.d.). *What is an IP address?* Kaspersky. Retrieved September 2, 2024, from https://www.kaspersky.com/resource-center/definitions/what-is-an-ip-address
- Cloudflare. (n.d.). *Internet Protocol*. Retrieved September 2, 2024, from https://www.cloudflare.com/learning/network-layer/internet-protocol/
- Network Lessons. (n.d.). *IP routing explained*. Retrieved September 2, 2024, from https://networklessons.com/cisco/ccna-routing-switching-icnd1-100-105/ip-routing-explained