PRESENTED BY

Group 4

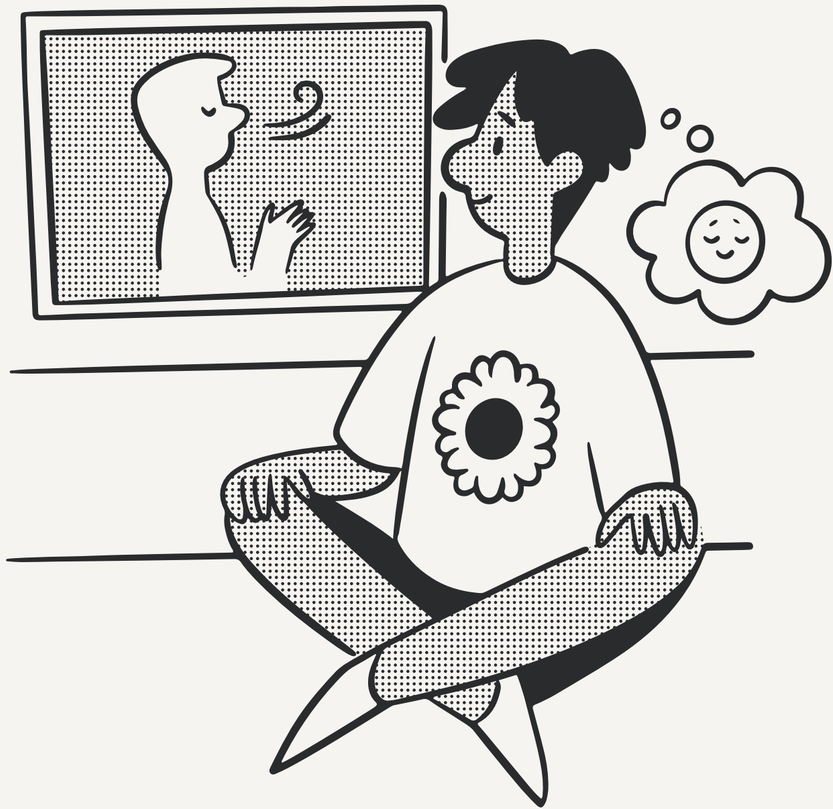
- Barotag
- Bernandino
- Businos
- Briones
- Milan
- Orcullo

NETWORK &
TRANSPORT LAYERS

OBJECTIVES

BY THE END OF THIS CHAPTER, YOU SHOULD:

- Be aware of the TCP/IP protocols
- Be familiar with linking to the application layer, segmenting, and session management
- Be familiar with addressing
- Be familiar with routing
- Understand how TCP/IP works



PRE-TEST!

PLEASE ACCESS THIS WEBSITE:

pretest-icebreaker.netlify.app

INSTRUCTION:

1. Create account by entering your name
2. Tap the ice to break it
    - (NOTE: you have to tap multiple times to break it)
3. Answer the questions
4. Finish the questions before the timer ends

INTRODUCTION

The **NETWORK LAYER** and **TRANSPORT LAYER** are responsible for moving messages from end to end in a network.
They are so closely tied together that they are usually discussed together.

Transport Layer

- **Links** the application layer (where software runs) to the network and is responsible for the end-to-end delivery of the entire message.

The transport layer performs **three functions**:

1. **Linking the application layer to the network** - The transport layer uses port addresses (16-bit numbers) to determine which specific application layer program (e.g., Web browser, email) an incoming message should be delivered to
2. **Segmenting** (breaking long messages into smaller packets for transmission)
   - divides large messages into smaller segments for transmission and reassembles them at the destination.
3. **Session management** (establishing an end-to-end connection between the sender and receiver)
   - establishes, maintains, and terminates communication sessions.

INTRODUCTION

Network Layer

The network layer takes the messages (segments) from the transport layer and performs two primary functions:

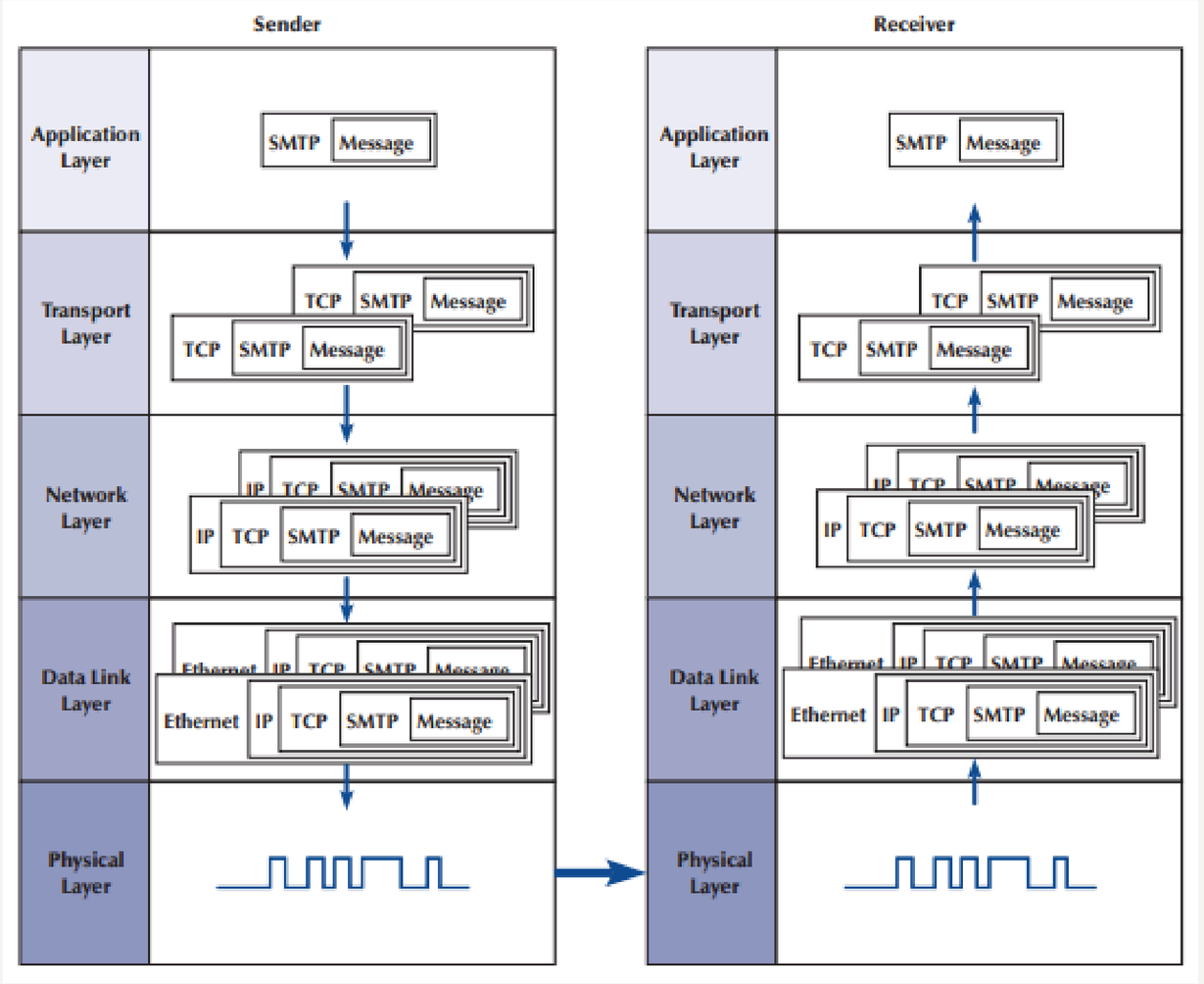
1. **Routing**
   - determines the best path for a message to travel from the sending computer across multiple routers to reach the final destination. This is done by devices called **routers**.

2. **Addressing**
   - It uses network addresses (like IP addresses) to identify the next computer or router in the path that the message should be sent to.

INTRODUCTION

1. The message starts at the **application layer** (e.g., SMTP for email).
2. The **transport layer** (TCP) breaks it into segments
3. The network layer adds IP information
4. The data link layer adds Ethernet frame headers and trailers (like MAC addresses)
5. The physical layer transmits signals through cables or wireless.

At the receiver side, the process is reversed until the complete message is delivered to the application.



INTRODUCTION

The process is like a set of ***matryoshka (*nested Russian dolls**), with each layer encapsulating the data from the layer above it

- As the data moves down the layers, it gets bigger because each layer adds its own control information. When the data arrives at its destination, the process is reversed: the dolls are "unpacked" one by one (the headers are removed) until the original message is revealed at the very center

TRANSPORT AND NETWORK LAYER PROTOCOLS

- There are different transport/network layer protocols, but one family of protocols, **the Internet Protocol Suite**, dominates

TCP/IP (TRANSMISSION CONTROL PROTOCOL/INTERNET PROTOCOL)

- Developed for the **ARPANET** by **Vinton Cerf and Bob Kahn** in 1974,
- It is the world's most popular protocol set, used by almost all backbone networks (BNs) and WANs
- TCP/IP is known for being efficient and reliable due to its error-checking capabilities.
    - it can send large files across sometimes unreliable networks with great assurance that the data will arrive uncorrupted
- It's also compatible with various data link protocols, which contributes to its popularity.

TCP/IP (TRANSMISSION CONTROL PROTOCOL/INTERNET PROTOCOL)

TCP/IP has two parts.

1. **Transmission Control Protocol (TCP)**
2. **Internet Protocol (IP)**

TCP is the **transport layer protocol** that **links** the application layer to the network layer.

- It performs **segmenting**: TCP breaks long outgoing messages from the application layer into smaller Protocol Data Units (PDUs) called segments. These segments are numbered so the receiving TCP software can reassemble them into the correct order and confirm that no segments have been lost

IP is the **network layer protocol** and performs **addressing** and **routing**.

- it is IP that routes the message to the final destination.

The TCP software needs to be active only at the sender and the receiver, because TCP is involved only when data comes from or goes to the application layer.

TCP (TRANSMISSION CONTROL PROTOCOL)

A typical TCP segment has a 192-bit header (24 bytes) of control information

- it contains the source and destination port identifier

    - **Destination Port**: tells the TCP software at the destination to which application layer program the application layer packet should be sent
    - **Source Port:** tells the receiver which application layer program the packet is from
    - **Sequence Number**: Used to reassemble data in the correct order and check for missing segments.
    - **Header Length**: Indicates the length of the TCP header

Typically 20 bytes long (160 bits). The full header is 24 bytes (192 bits) but the Options field is rarely used

Source port	Destination port	Sequence number	ACK number	Header length	Unused	Flags	Flow control	CRC–16	Urgent pointer	Options	User data
16 bits	16 bits	32 bits	32 bits	4 bits	3 bits	9 bits	16 bits	16 bits	16 bits	32 bits	Varies

TCP (TRANSMISSION CONTROL PROTOCOL)

The Internet Protocol Suite has a second type of transport layer protocol called User Datagram Protocol (UDP).

- UDP PDUs are called **datagrams**.

A **UDP datagram** has only **four fields (8 bytes of overhead)** plus the application layer packet: source port, destination port, length, and a CRC-16.

- Unlike TCP, UDP does not check for lost messages, so occasionally a UDP datagram is lost and the message must be resent. Interestingly, it is not the transport layer that decides whether TCP or UDP is going to be used. This decision is left to the engineer who is writing the application

INTERNET PROTOCOL (IP)

The Internet Protocol (IP) is the network layer protocol in the TCP/IP model.

- Network layer PDUs are called **packets**.

Two forms of IP are currently in use:

1. IPv4 (Internet Protocol Version 4)

- Older form of IP, still widely used.
- Header size: 192 bits (24 bytes); usually 20 bytes because the Options field is rarely used.

Header contains:

- Source & destination addresses
- Packet length
- Packet number

IPv4 addresses are **4 bytes (32 bits)**.
Theoretical maximum: 4.2 billion addresses (but about 500 million reserved).
Address format: Decimal notation (e.g., 128.192.55.72).

Why IPv4 is Limited

- Growth of Internet users + mobile devices = IPv4 address depletion.
- Projected exhaustion of IPv4 addresses: around **2011**

INTERNET PROTOCOL (IP)

2. IPv6

- **IPv6** has a 320-bit header (40 bytes)
- IPv6's **simpler packet structure** makes it easier to perform routing and supports a variety of new approaches to addressing and routing.

- Internet Protocol version 6 uses a 16-byte-long address, which provides a theoretical maximum of 3.4 × 10^38 addresses—more than enough for the foreseeable future

IPv6 Features

- Simpler packet structure → Easier routing.
- Supports new addressing & routing approaches.
- Hexadecimal notation (base 16), like Ethernet.
  - Example: 2001:0890:0600:00d1:0000:0000:abcd:f010
  - Can use compressed notation (removing leading zeros & all-zero blocks):
    - 2001:890:600:d1::abcd:f010

Version number	Header length	Type of service	Total length	Identifiers	Flags	Packet offset	Hop limit	Protocol	CRC 16	Source address	Destination address	Options	User data
4 bits	4 bits	8 bits	16 bits	16 bits	3 bits	13 bits	8 bits	8 bits	16 bits	32 bits	32 bits	32 bits	Varies

FIGURE 5-3 Internet Protocol (IP) packet (version 4). CRC = Cyclical Redundancy Check

Version number	Priority	Flow name	Total length	Next header	Hop limit	Source address	Destination address	User data
4 bits	4 bits	24 bits	16 bits	8 bits	8 bits	128 bits	128 bits	Varies

FIGURE 5-4 Internet Protocol (IP) packet (version 6)

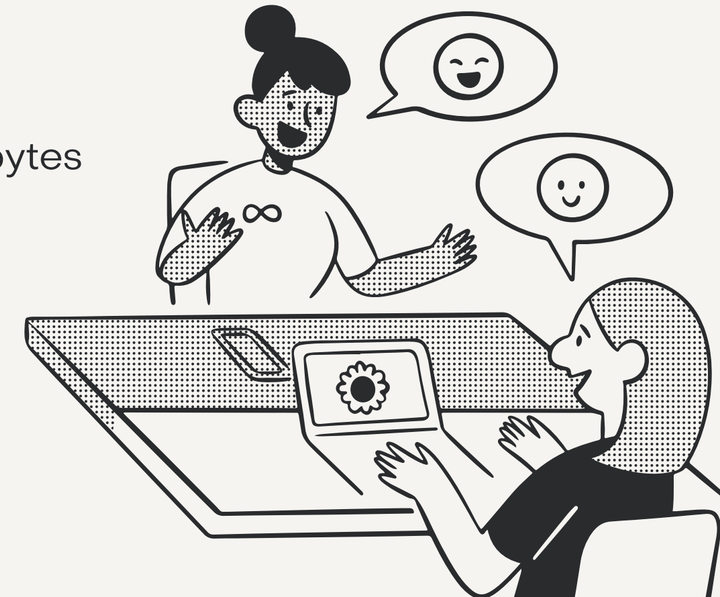
INTERNET PROTOCOL (IP)

Adoption Challenges

- Adoption of IPv6 has been slow. Most organizations have not felt the need to change because IPv6 provides few benefits other than the larger address space and requires their staff to learn a whole new protocol
- Not backward-compatible with IPv4
- Conversion costs + staff retraining → slow adoption (“IPv6 mess’).
- Most organizations run dual stack (IPv4 + IPv6).
- U.S. government mandated IPv6 on WANs/BNs by June 2008 (not completed on time).

Message Size Impact

- Size of the message field depends on the data link layer protocol.
- Example with Ethernet (max packet = 1,492 bytes):
  - Max TCP message with IPv4 = 1,492 − 24 (TCP header) − 24 (IPv4 header) = 1,444 bytes



PRESENTED BY
Ace

TRANSPORT LAYER FUNCTIONS

- LINKING TO THE APPLICATION LAYER
- SEGMENTING
- SESSION MANAGEMENT

CORE FUNCTIONS OF THE TRANSPORT LAYER

The Transport Layer acts as the bridge between the Application Layer and the Network Layer, performing three critical tasks:

1. **Linking to the Application Layer:** Directing data to the correct application on a computer.
2. **Segmenting:** Breaking large messages into smaller, manageable pieces for transmission.
3. **Session Management:** Establishing, maintaining, and tearing down communication "conversations" between computers.

CORE FUNCTIONS OF THE TRANSPORT LAYER

1. Linking to the Application Layer

How does a computer with a web browser, an email client, and a music app all running at once send and receive data correctly?

Answer: **Port Addresses**

- Every application is assigned a unique port address.
- Well-Known Ports: Standardized ports for common services (e.g., HTTP Web traffic uses port 80, SMTP email uses port 25).
- Temporary Ports: Client applications are assigned temporary port numbers for a session.

This system ensures that a web page request goes to the browser and not the email client.

CORE FUNCTIONS OF THE TRANSPORT LAYER

2. Segmenting

Application messages (like web pages or images) are often too large to be sent in a single piece across the network.

The Solution:

- **Segmenting (Sender):** The transport layer breaks the large message into a set of smaller packets called segments. Each segment is numbered sequentially.
- **Reassembly (Receiver):** The transport layer receives the individual segments, uses the sequence numbers to put them back in the correct order, and reassembles the original message for the application layer.

This process allows for efficient transmission and error handling.

CORE FUNCTIONS OF THE TRANSPORT LAYER

3. Session Management

A session is like a conversation between two computers. The transport layer manages how these conversations start, proceed, and end.

There are two primary approaches:

- **Connection-Oriented Messaging (Formal Conversation):** Establishes a dedicated connection before sending data. It's reliable and ensures all data arrives in order.
- **Connectionless Messaging (Quick Note):** Sends data without establishing a connection first. It's faster but less reliable.

ERROR CONNECTION: AUTOMATIC REPEAT REQUEST (ARQ)

Connection-oriented messaging guarantees delivery using ARQ.

- **Stop-and-Wait ARQ:** The sender sends one packet, stops, and waits for an ACK before sending the next. It's simple but slow.
- **Continuous ARQ (Sliding Window):** The sender can transmit multiple packets before receiving an ACK. This is much more efficient and allows for flow control, preventing the sender from overwhelming the receiver.

CONNECTIONLESS MESSAGING (UDP)

Used when speed is more important than reliability and the message is small enough to fit in one packet.

- **How it works**: The sender just sends the packet (called a datagram) and moves on. No session is established.
- **Common Uses**: DNS requests, some streaming media, and network management messages.
- **Trade-off**: It's very fast due to low overhead, but there is no guarantee the packet will arrive, and no error correction is performed.

PRESENTED BY
Harjii

ADDRESING

- ASSIGNING ADDRESSES
- ADDRESS RESOLUTION

ADDRESSING

- Addressing in computer networks is the method of assigning unique identifiers to devices and applications across different layers of a network so that data can be correctly delivered from a source to a destination. It provides a way to locate and identify endpoints within and across networks.

TYPES OF ADDRESSES

- **Application Layer Address:**

Example - Internet Browser (e.g., www.indiana.edu).

An application layer address is a human-readable name, such as a website URL (www.google.com). It is designed for people to easily identify resources on the Internet. This type of address must be translated into a network layer address before data can be transmitted.

TYPES OF ADDRESSES

- **Network Layer Address:**

Example – Internet Protocol (IP, e.g., 129.79.127.4).

A network layer address is the logical address used by the Internet Protocol (IP). It identifies each device on a network so that data can be routed from the sender to the receiver across different networks. Examples include IPv4 addresses (e.g., 129.79.127.4) and IPv6 addresses (e.g., 2001:db8::1).

TYPES OF ADDRESSES

- **Data Link Layer Address**

Example - Ethernet (MAC Address, 00-0C-00-F5-03-5A).

A data link layer address, also called a physical or MAC (Media Access Control) address, is a unique identifier embedded into a network card by the manufacturer. It is used within a local area network to deliver data to the correct device.

ASSIGNING ADDRESSES

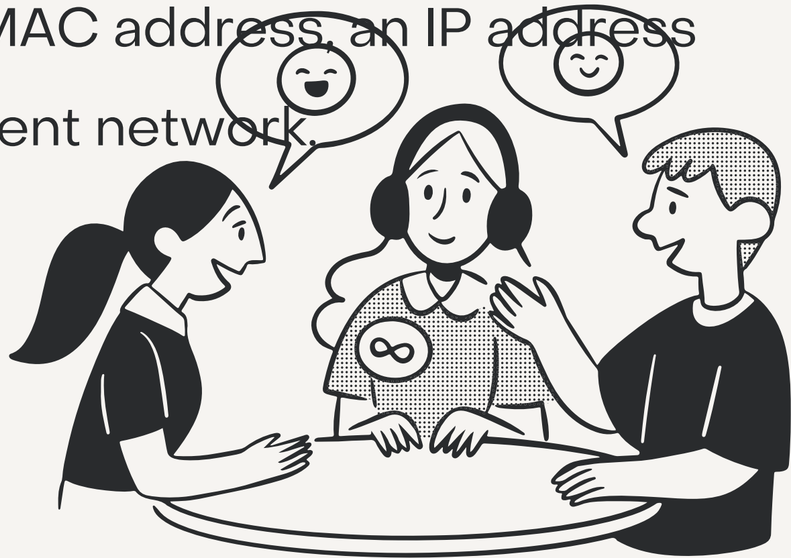
- **Application Layer Address (Domain Name or Device Name):**

The application layer address is a human-friendly identifier, such as an Internet domain name or a device name in an operating system. These addresses are also assigned through software. While client computers often do not require application layer addresses, servers usually have them so that users can easily locate and connect to their resources. Domain names must be approved by standards authorities, such as ICANN, to ensure global uniqueness and avoid conflicts.

ASSIGNING ADDRESSES

- **Network Layer Address (IP Address):**

The network layer address is a logical identifier that is assigned by software rather than hardware. It is usually specified in configuration files managed by network administrators. These addresses must be unique within a network so that messages can be delivered correctly. To prevent duplication across organizations, standards groups define and regulate which network layer addresses can be used. Unlike a MAC address, an IP address can be changed or reassigned when a device moves to a different network.



ASSIGNING ADDRESSES

- **Data Link Layer Address (MAC Address):**

The data link layer address, also called the physical or MAC (Media Access Control) address, is permanently encoded into the hardware of a network card (such as an Ethernet card). Each manufacturer is assigned a specific range of addresses, ensuring that no two devices in the world will share the same MAC address, even if they come from different companies. Because it is built into the hardware, this address does not change and uniquely identifies every device as soon as the network card is installed.

FUN FACT !

- **IPv4 was depleted on September 24, 2015**

There are no more IPv4 addresses left to be assigned. The American Registry for Internet Numbers (ARIN), which is in charge of the IPv4 address space

- IPv4 (Internet Protocol version 4) has about 4.3 billion unique addresses (32-bit).
- As the internet grew (smartphones, IoT, etc.), demand for addresses far exceeded supply.
- The regional internet registries (RIRs, like ARIN, APNIC, RIPE, etc.) officially ran out of unallocated IPv4 blocks between 2011 and 2019.

ADDRESS RESOLUTION

Address Resolution is the process of translating an Application Layer Address (example www.google.com), into a network layer address (example 172.217.19.164), and in turn translate that network layer into a data link layer address (example 00:1A:2B:3C:4D:5E).

In Short

www.google.com → 172.217.19.164 → 00:1A:2B:3C:4D:5E

SERVER NAME RESOLUTION

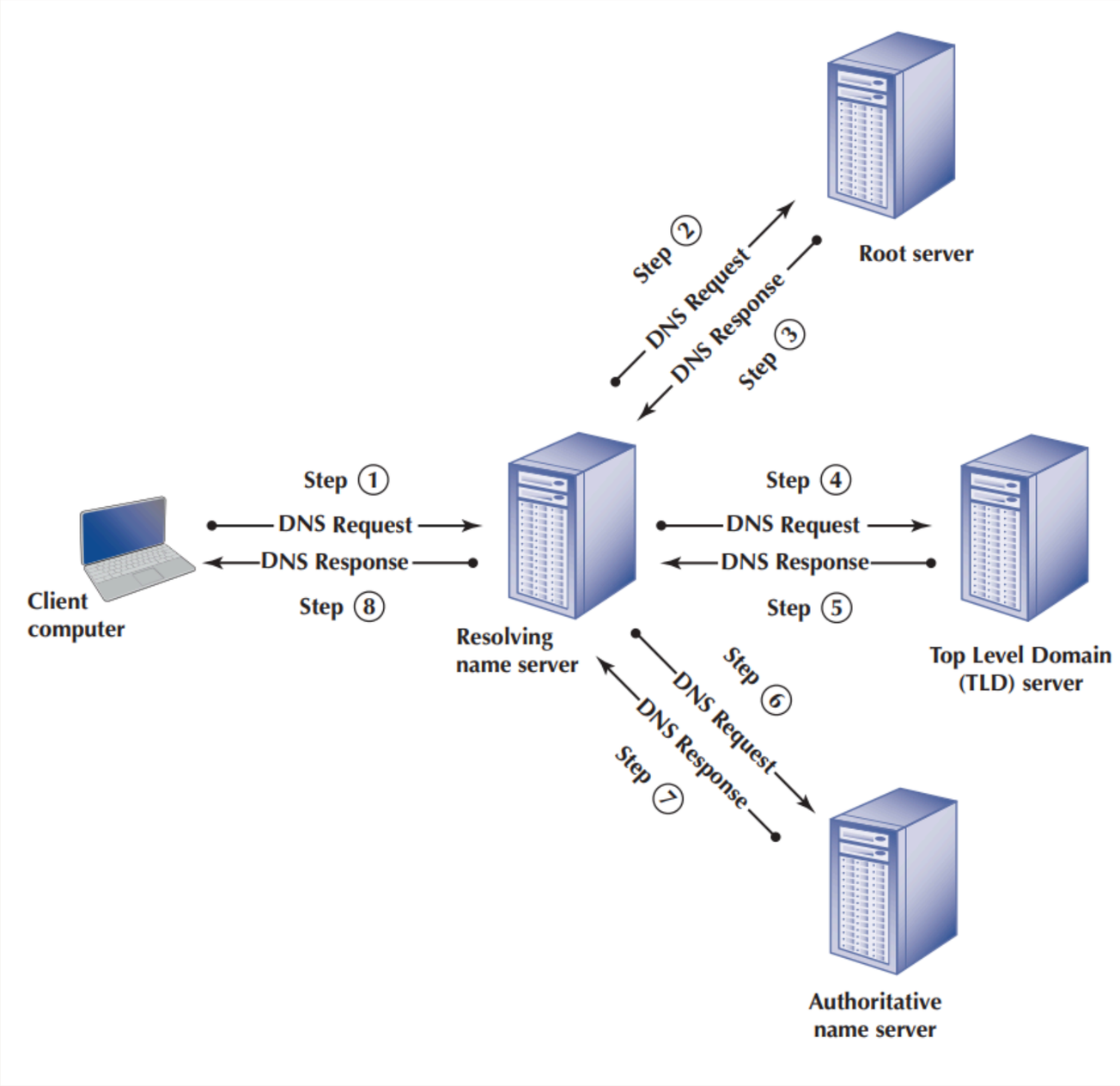
Server name resolution is the translation of application layer addresses into network layer addresses (e.g., translating an Internet address such as www.yahoo.com into an IP address such as 204.71.200.74).This is done using the Domain Name Service (DNS).Throughout the Internet a series of computers called name servers provide DNS services.

DATA LINK LAYER ADDRESS RESOLUTION

Data link layer address resolution is the process of determining the physical MAC address that corresponds to a known IP address, usually by using the Address Resolution Protocol (ARP), so that a computer can accurately deliver data across the local network or to the next device, such as a router.

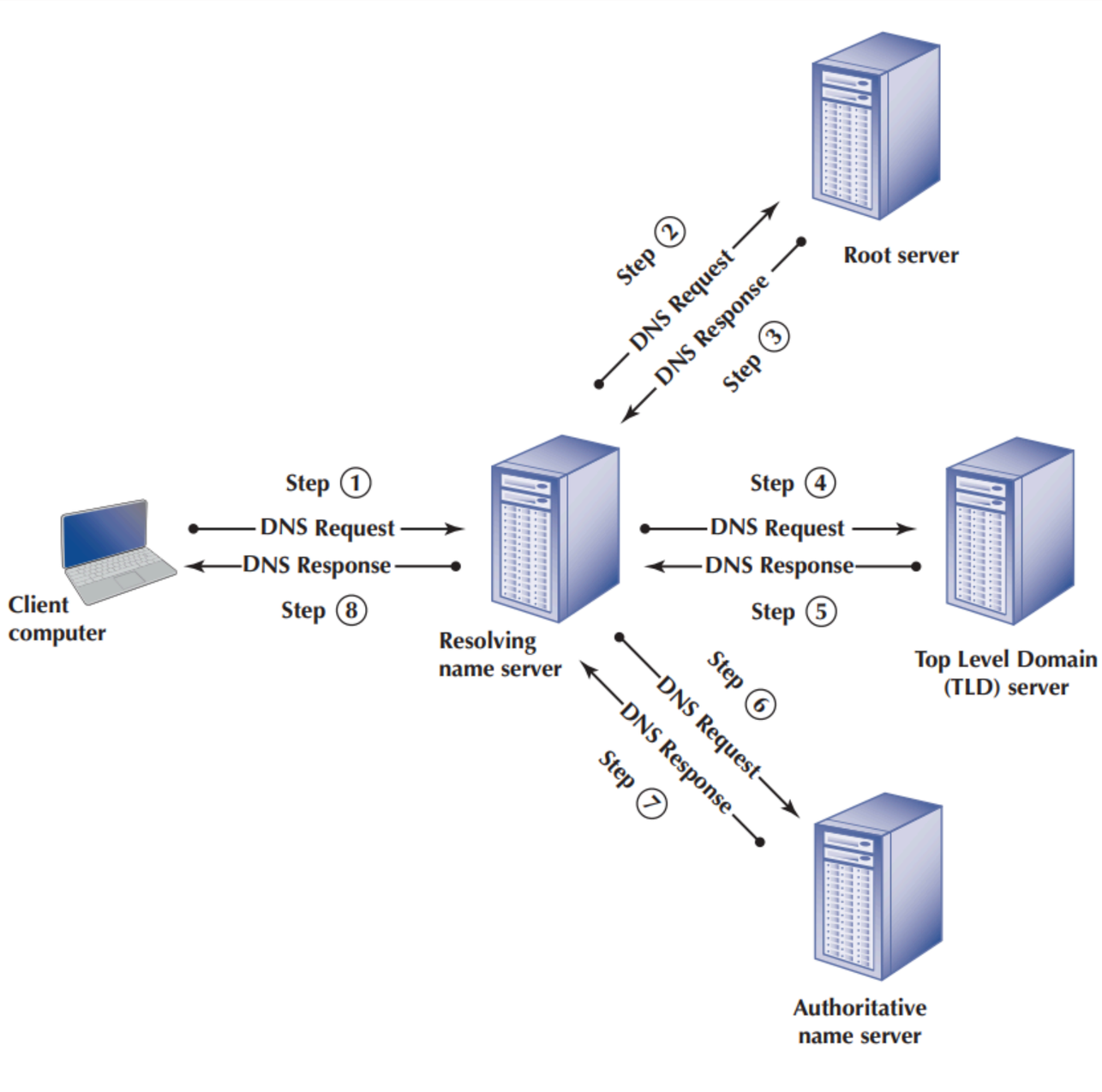
HOW DNS SYSTEM WORKS

- Your computer asks its local DNS server (called the resolving name server) for the IP address of the website. For example, the University of Toronto computer asks the University of Toronto's DNS server.
- If the local DNS server already has the answer stored (in its cache), it immediately replies. If not, it has to go find the answer.
- The local DNS server first asks a root server. Root servers are like the "directory assistance' of the Internet. They don't know the exact IP of the website, but they know where to point you next.
- The root server replies: "I don't know the address of www.kelley.indiana.edu, but I know which server handles all .edu websites.' That server is called the Top-Level Domain (TLD) server for .edu.k to your computer, and your browser can now connect directly to the Indiana University server.



HOW DNS SYSTEM WORKS

- The local DNS server then asks the .edu TLD server.
- The TLD server replies: “I don’t know the exact IP either, but I know the authoritative name server for indiana.edu.’ The authoritative server is the one that has the final, official answers for that domain.
- The local DNS server then asks the authoritative server for indiana.edu: “What is the IP of *www.kelley.indiana.edu*?’
- The authoritative server responds with the correct IP address.
- Finally, the local DNS server passes that IP address back to your computer, and your browser can now connect directly to the Indiana University server.



PRESENTED BY
JB

ROUTING

- TYPES OF ROUTING
- ROUTING PROTOCOLS
- MULTICASTING
- THE ANATOMY OF A ROUTER

ROUTING

WHAT IS ROUTING?

The process of determining the path that a message will travel from a sending computer to a receiving computer.

- Routers are the key devices. They connect different networks (subnets) and make decisions about where to forward messages.
- Each router maintains a routing table to help it make these decisions.

ROUTING TABLE

The process of determining the path that a message will travel from a sending computer to a receiving computer.

Router R1's Routing Table

Network Address	Interface
10.10.51.0–10.10.51.255	1
10.10.52.0–10.10.52.255	2
10.10.53.0–10.10.53.255	3
10.10.20.0–10.10.20.255	3
10.10.43.0–10.10.43.255	3
10.10.1.2	3
All other addresses	0

Router R2's Routing Table

Network Address	Interface
10.10.1.1	0
10.10.53.0–10.10.53.255	1
10.10.20.0–10.10.20.255	2
10.10.43.0–10.10.43.255	3
All other addresses	0

ROUTING TABLE

WHEN A PACKET ARRIVES, THE ROUTER...

- Reads the destination IP address.
- Finds a matching network in its routing table.
- Forwards the packet out of the corresponding interface.

APPROACHES TO ROUTING

THERE ARE THREE FUNDAMENTAL WAYS ROUTING DECISIONS CAN BE MADE:

- **Centralized Routing**: A single, central computer makes all routing decisions.
- **Static Routing**: Each router has a fixed routing table, manually configured by a network administrator.
- **Dynamic Routing**: Routers automatically exchange information with other routers to build and update their own routing tables.

STATIC ROUTING

HOW DOES IT WORK?

Paths are fixed and manually entered. They only change when an administrator updates them.

WHAT IS IT BEST FOR?

Simple, stable networks where traffic paths rarely change.

PROS:

Predictable, secure, and uses minimal router resources.

CONS:

Not scalable and does not adapt to network outages or changes.

DYNAMIC ROUTING

HOW DOES IT WORK?

Routers use special protocols to "talk" to each other and learn about the best available paths.

WHAT IS IT BEST FOR?

Complex, large networks where conditions are always changing (like the Internet).

PROS:

Adapts to network changes automatically, finds the fastest route.

CONS:

Requires more processing power and generates additional network traffic.

HOW DYNAMIC ROUTERS TALK: ROUTING PROTOCOLS

A routing protocol is the set of rules or language that routers use to exchange information and build their routing tables.

Autonomous Systems (AS)

To manage the complexity of the global internet, the network is divided into Autonomous Systems. An AS is a large network operated by a single organization (e.g., an ISP like Comcast, a tech company like Google, or a university).

- **Interior Routing Protocols (IGP)**: Used for routing within an Autonomous System.
- **Exterior Routing Protocols (EGP)**: Used for routing between different Autonomous Systems.

COMMON ROUTING PROTOCOLS

A routing protocol is the set of rules or language that routers use to exchange information and build their routing tables.

Interior Gateway Protocols (IGP)

- **Routing Information Protocol (RIP):** A simple distance-vector protocol that counts "hops" (number of routers) to find a path. Good for small networks.
- **Open Shortest Path First (OSPF):** A more advanced link-state protocol that considers factors like circuit speed and network traffic to find the best path. Preferred for most large corporate networks.

Exterior Gateway Protocol (EGP)

- **Border Gateway Protocol (BGP):** The core routing protocol of the Internet. It is used to exchange routing information between the massive Autonomous Systems that make up the internet.

MULTICASTING: EFFICIENT ONE-TO-MANY COMMUNICATION

Beyond sending a message to just one computer, there are other ways to address packets:

- **Unicast:** One sender to one receiver. (Standard communication)
- **Broadcast:** One sender to all other computers on a subnet.
- **Multicast:** One sender to a specific group of interested receivers.

This is very efficient for applications like video conferencing or stock market data feeds, as the sender only has to transmit the packet once.

ANATOMY OF A ROUTER

A router is a special-purpose computer designed to move data between networks.

Core Components:

- **Hardware:** CPU, Memory (RAM and ROM), and multiple network interfaces (ports).
- **Operating System:** Most routers run a command-line OS, like Cisco IOS.

Configuration & Management:

- A config file stores the router's settings (IP addresses, routing protocols).
- An Access Control List (ACL) acts as a firewall, defining what traffic is allowed or denied.
- Admins connect via a console port for initial setup, then a network interface for remote management.

PRESENTED BY
Ed

TCP/IP EXAMPLE

- KNOWN ADDRESSES
- UNKNOWN ADDRESSES
- TCP CONNECTIONS
- TCP/IP AND NETWORK LAYERS

KNOWN ADDRESSES

PRESENTED BY
Ed

WHEN WE TALK ABOUT KNOWN ADDRESSES, WE MEAN THAT THE DEVICE INITIATING COMMUNICATION ALREADY HAS THE INFORMATION NEEDED TO CONTACT THE DESTINATION. THIS USUALLY INCLUDES THE IP ADDRESS OF THE DESTINATION HOST AND THE PORT NUMBER OF THE SERVICE THAT IT WANTS TO USE.

FOR EXAMPLE, IF AN ADMINISTRATOR WANTS TO CONNECT TO A SERVER USING THE SSH PROTOCOL, AND THE SERVER'S IP ADDRESS IS 203.0.113.5, THE CLIENT COMPUTER ALREADY KNOWS THE DESTINATION: IP 203.0.113.5 AND PORT 22 (THE DEFAULT SSH PORT). IN THIS CASE, THE CLIENT CAN IMMEDIATELY BEGIN THE PROCESS OF ESTABLISHING A CONNECTION.

KNOWN ADDRESSES

PRESENTED BY
Ed

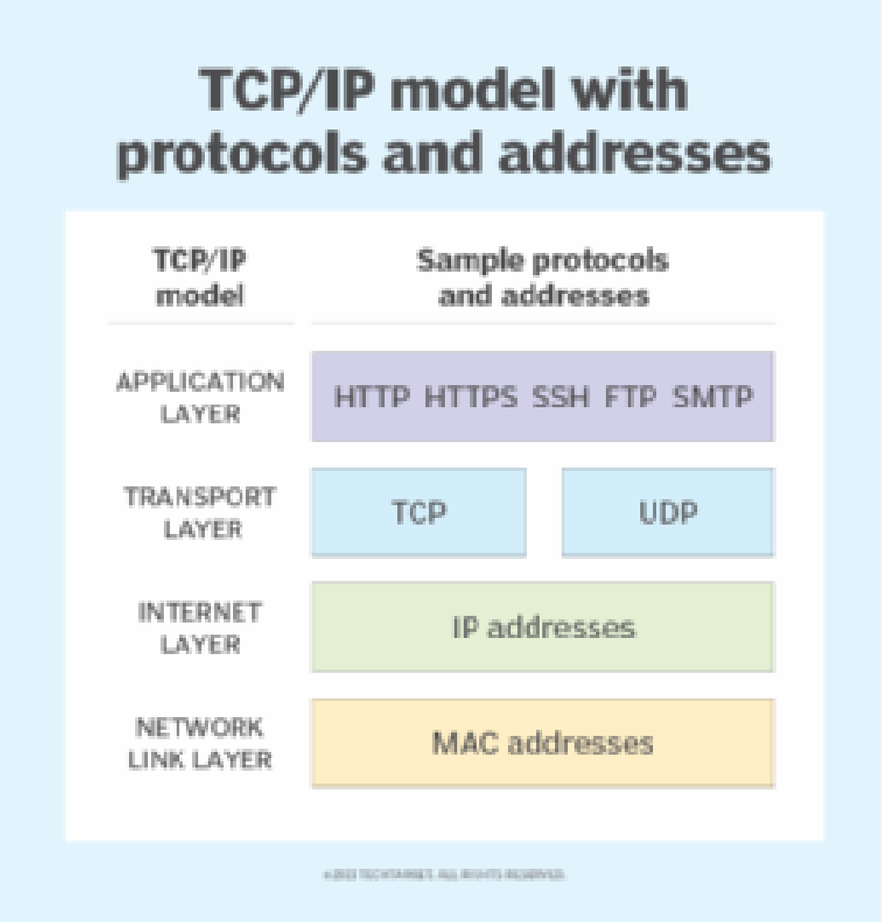
HERE IS WHAT HAPPENS STEP BY STEP:

1. IF THE DESTINATION HOST IS ON THE SAME LOCAL NETWORK, THE CLIENT MUST STILL RESOLVE THE HARDWARE ADDRESS (MAC ADDRESS) OF THE DESTINATION MACHINE. THIS IS DONE USING THE ADDRESS RESOLUTION PROTOCOL (ARP) IN IPV4 OR THE NEIGHBOR DISCOVERY PROTOCOL (NDP) IN IPV6. ONCE THE MAC ADDRESS IS KNOWN, THE DATA CAN BE SENT DIRECTLY ACROSS THE LOCAL LINK.
2. IF THE DESTINATION IS OUTSIDE THE LOCAL NETWORK, THE CLIENT DOES NOT SEND PACKETS DIRECTLY TO THE FINAL DESTINATION. INSTEAD, IT SENDS THE PACKETS TO A DEFAULT GATEWAY, WHICH IS TYPICALLY A ROUTER. THE GATEWAY'S MAC ADDRESS IS RESOLVED WITH ARP, AND THE PACKETS ARE FORWARDED TOWARD THE FINAL DESTINATION ACROSS DIFFERENT NETWORKS.
3. ONCE THE PATH TO THE DESTINATION IS SET, THE TCP HANDSHAKE PROCESS BEGINS BETWEEN THE CLIENT AND SERVER TO ESTABLISH A RELIABLE CONNECTION.

KNOWN ADDRESSES

PRESENTED BY
Ed

IN SHORT, KNOWN ADDRESSES MEAN NO DISCOVERY OR LOOKUP IS REQUIRED BEYOND LOCAL MAC RESOLUTION. COMMUNICATION CAN PROCEED IMMEDIATELY BECAUSE THE CLIENT HAS THE NECESSARY ADDRESSING INFORMATION.



UNKNOWN ADDRESSES

PRESENTED BY
Ed

IN COMPUTER NETWORKS, COMMUNICATION REQUIRES KNOWING WHERE TO SEND THE DATA. NORMALLY, THIS IS DONE USING IP ADDRESSES (FOR LOGICAL ADDRESSING ACROSS NETWORKS) AND MAC ADDRESSES (FOR HARDWARE-LEVEL DELIVERY ON THE LOCAL NETWORK). HOWEVER, THERE ARE MANY SITUATIONS WHERE THE INITIATING DEVICE DOES NOT KNOW THE DESTINATION ADDRESS. INSTEAD, IT ONLY HAS PARTIAL INFORMATION, OR IN SOME CASES, NO INFORMATION AT ALL. TO SOLVE THIS, THE SYSTEM RELIES ON ADDRESS RESOLUTION AND DISCOVERY PROTOCOLS. THESE PROTOCOLS ACT LIKE “TRANSLATORS’ OR “HELPERS’ THAT FILL IN THE MISSING INFORMATION SO COMMUNICATION CAN PROCEED.

UNKNOWN ADDRESSES

PRESENTED BY
Ed

SOME COMMON SCENARIOS INCLUDE:

A USER ENTERS A WEB ADDRESS SUCH AS WWW.EXAMPLE.COM. THE CLIENT COMPUTER KNOWS THE HOSTNAME BUT NOT THE IP ADDRESS. TO SOLVE THIS, IT SENDS A QUERY TO THE DOMAIN NAME SYSTEM (DNS). DNS SERVERS THEN RETURN THE CORRESPONDING IP ADDRESS, SUCH AS 198.51.100.7.

IF THE DEVICE ONLY KNOWS THE IP ADDRESS BUT NEEDS THE HARDWARE (MAC) ADDRESS ON THE LOCAL NETWORK, IT BROADCASTS AN ARP REQUEST IN IPV4, OR AN NDP REQUEST IN IPV6. THE HOST THAT OWNS THAT IP REPLIES WITH ITS MAC ADDRESS.

WHEN A DEVICE JOINS A NETWORK FOR THE FIRST TIME AND DOES NOT HAVE AN IP ADDRESS AT ALL, IT USES THE DYNAMIC HOST CONFIGURATION PROTOCOL (DHCP). THE DEVICE BROADCASTS A REQUEST, AND THE DHCP SERVER RESPONDS BY ASSIGNING AN IP ADDRESS AND OTHER CONFIGURATION SETTINGS SUCH AS THE GATEWAY AND DNS SERVER.

IN LOCAL NETWORKS WITHOUT CENTRALIZED DNS, PROTOCOLS LIKE MULTICAST DNS (MDNS) OR UNIVERSAL PLUG AND PLAY (UPNP) ALLOW DEVICES TO DISCOVER EACH OTHER AND ADVERTISE SERVICES. FOR EXAMPLE, A NETWORK PRINTER MAY ANNOUNCE ITSELF SO THAT CLIENT COMPUTERS CAN FIND IT WITHOUT KNOWING ITS IP ADDRESS.

TCP CONNECTIONS

PRESENTED BY
Ed

THE TRANSMISSION CONTROL PROTOCOL (TCP) IS ONE OF THE MOST IMPORTANT PARTS OF THE TCP/IP SUITE. IT IS A RELIABLE, CONNECTION-ORIENTED PROTOCOL USED TO DELIVER DATA ACCURATELY AND IN ORDER. UNLIKE UDP, WHICH SIMPLY SENDS PACKETS WITHOUT GUARANTEEING DELIVERY, TCP ENSURES THAT EVERY PIECE OF DATA REACHES ITS DESTINATION IN THE CORRECT SEQUENCE.

A TCP CONNECTION IS ESTABLISHED USING A PROCESS CALLED THE THREE-WAY HANDSHAKE. HERE IS HOW IT WORKS IN DETAIL:

1. THE CLIENT BEGINS BY SENDING A SYN (SYNCHRONIZE) MESSAGE TO THE SERVER. THIS IS ESSENTIALLY A REQUEST TO START COMMUNICATION AND CONTAINS AN INITIAL SEQUENCE NUMBER.
2. THE SERVER RESPONDS WITH A SYN-ACK (SYNCHRONIZE-ACKNOWLEDGE) MESSAGE. THIS MEANS THE SERVER ACCEPTS THE CLIENT'S REQUEST AND ALSO SENDS ITS OWN SEQUENCE NUMBER.
3. THE CLIENT REPLIES WITH AN ACK (ACKNOWLEDGE) MESSAGE TO CONFIRM RECEIPT OF THE SERVER'S SEQUENCE NUMBER.

AFTER THIS THREE-STEP PROCESS, THE CONNECTION IS FULLY ESTABLISHED, AND BOTH SIDES CAN BEGIN TRANSMITTING DATA.

TCP CONNECTIONS

PRESENTED BY
Ed

OTHER IMPORTANT FEATURES OF TCP INCLUDE:

- SEQUENCING AND ACKNOWLEDGMENT: EVERY BYTE OF DATA IS NUMBERED, AND ACKNOWLEDGMENTS CONFIRM RECEIPT. IF DATA IS LOST, IT WILL BE RETRANSMITTED.

- FLOW CONTROL: TCP ENSURES THAT A SENDER DOES NOT OVERWHELM A RECEIVER BY CHECKING THE RECEIVER'S AVAILABLE BUFFER SPACE.

- CONGESTION CONTROL: TCP DETECTS NETWORK CONGESTION AND REDUCES ITS SENDING RATE TO AVOID PACKET LOSS.

- GRACEFUL TERMINATION: WHEN COMMUNICATION ENDS, BOTH SIDES EXCHANGE FIN (FINISH) AND ACK MESSAGES TO PROPERLY CLOSE THE CONNECTION. A STATE CALLED TIME-WAIT ENSURES DELAYED PACKETS DO NOT INTERFERE WITH FUTURE CONNECTIONS.

IN SHORT, TCP PROVIDES RELIABILITY, ORDERING, AND CONTROL OVER DATA TRANSMISSION, MAKING IT ESSENTIAL FOR APPLICATIONS SUCH AS WEB BROWSING, EMAIL, AND FILE TRANSFERS.

TCP/IP AND NETWORK LAYERS

PRESENTED BY
Ed

1. APPLICATION LAYER

THIS IS THE TOPMOST LAYER, WHERE NETWORK SERVICES AND APPLICATIONS DIRECTLY INTERACT WITH USERS OR WITH OTHER SOFTWARE.

- RESPONSIBILITIES:
  - PROVIDES SERVICES TO END-USERS OR APPLICATIONS.
  - DEFINES HOW DATA IS PRESENTED, STRUCTURED, AND INTERPRETED.
  - ENSURES COMPATIBILITY BETWEEN DIFFERENT SYSTEMS.
- EXAMPLES OF PROTOCOLS:
  - HTTP/HTTPS – USED BY WEB BROWSERS FOR ACCESSING WEBSITES.
  - SMTP, IMAP, POP3 – USED FOR SENDING AND RECEIVING EMAIL.
  - FTP, SFTP – USED FOR FILE TRANSFERS.
  - DNS – RESOLVES HOSTNAMES TO IP ADDRESSES.
- REAL-WORLD ANALOGY: THINK OF THE APPLICATION LAYER AS THE “USER INTERFACE’ OF A POST OFFICE. IT’S WHERE YOU FILL OUT FORMS, WRITE LETTERS, OR PACKAGE BOXES BEFORE MAILING.

TCP/IP AND NETWORK LAYERS

PRESENTED BY
Ed

2. TRANSPORT LAYER

THE TRANSPORT LAYER ENSURES THAT COMMUNICATION BETWEEN APPLICATIONS ON DIFFERENT DEVICES IS RELIABLE, ORDERED, AND EFFICIENT.

- RESPONSIBILITIES:
  - BREAKS DATA INTO SMALLER CHUNKS (CALLED SEGMENTS).
  - ENSURES RELIABLE DELIVERY WITH ERROR DETECTION AND CORRECTION.
  - MANAGES FLOW CONTROL TO PREVENT OVERWHELMING THE RECEIVER.
  - USES PORTS (LIKE DOOR NUMBERS) TO IDENTIFY WHICH APPLICATION DATA BELONGS TO.
- KEY PROTOCOLS:
  - TCP (TRANSMISSION CONTROL PROTOCOL):
    - RELIABLE, CONNECTION-ORIENTED, GUARANTEES DELIVERY.
    - USED IN WEB BROWSING, EMAIL, FILE TRANSFERS.
  - UDP (USER DATAGRAM PROTOCOL):
    - FASTER, CONNECTIONLESS, NO DELIVERY GUARANTEE.
    - USED IN ONLINE GAMES, LIVE STREAMING, VOIP CALLS.
- REAL-WORLD ANALOGY: IF DATA IS LIKE PACKAGES BEING SHIPPED, TCP IS LIKE A COURIER SERVICE THAT PROVIDES TRACKING, REQUIRES SIGNATURES, AND ENSURES EVERY PACKAGE ARRIVES IN ORDER. UDP IS LIKE REGULAR MAIL — FASTER, BUT LESS RELIABLE.

TCP/IP AND NETWORK LAYERS

PRESENTED BY
Ed

3. INTERNET LAYER

THE INTERNET LAYER IS RESPONSIBLE FOR ROUTING DATA ACROSS NETWORKS, ENSURING THAT PACKETS TRAVEL FROM THE SOURCE DEVICE TO THE DESTINATION DEVICE, EVEN IF THEY ARE ON DIFFERENT NETWORKS.

- RESPONSIBILITIES:
  - LOGICAL ADDRESSING (IP ADDRESSES) TO IDENTIFY DEVICES GLOBALLY.
  - ROUTING PACKETS ACROSS MULTIPLE NETWORKS.
  - HANDLING ERRORS AND PROVIDING CONTROL MESSAGES.
- KEY PROTOCOLS:
  - IP (INTERNET PROTOCOL):
    - IPV4 (32-BIT ADDRESSES, E.G., 192.168.1.1).
    - IPV6 (128-BIT ADDRESSES, E.G., 2001:DB8::1).
  - ICMP (INTERNET CONTROL MESSAGE PROTOCOL): USED FOR ERROR REPORTING AND DIAGNOSTICS (E.G., PING).
  - ARP/NDP: USED TO RESOLVE IP ADDRESSES TO MAC ADDRESSES.
- REAL-WORLD ANALOGY: THINK OF THE INTERNET LAYER AS THE POSTAL SYSTEM. IT DOESN'T CARE WHAT'S INSIDE THE PACKAGE; IT ONLY LOOKS AT THE DESTINATION ADDRESS AND FIGURES OUT THE BEST ROUTE TO DELIVER IT.

TCP/IP AND NETWORK LAYERS

PRESENTED BY
Ed

4. LINK LAYER (NETWORK ACCESS LAYER)

THE LINK LAYER DEALS WITH THE PHYSICAL DELIVERY OF DATA WITHIN A LOCAL NETWORK SEGMENT.

- RESPONSIBILITIES:
  - WORKS WITH MAC ADDRESSES FOR LOCAL DEVICE IDENTIFICATION.
  - DEFINES HOW DATA IS FRAMED AND TRANSMITTED OVER PHYSICAL MEDIA (CABLES, RADIO WAVES, ETC.).
  - PROVIDES ERROR DETECTION USING CHECKSUMS OR CRC (CYCLIC REDUNDANCY CHECK).
  - DETERMINES HOW DEVICES SHARE THE MEDIUM (ETHERNET, WI-FI ACCESS RULES).
- EXAMPLES OF TECHNOLOGIES:
  - ETHERNET (WIRED LANS).
  - WI-FI (WIRELESS LANS).
  - PPP, FRAME RELAY, ATM IN OLDER SYSTEMS.
- REAL-WORLD ANALOGY: IF THE INTERNET LAYER IS LIKE WRITING THE ADDRESS ON A LETTER, THE LINK LAYER IS LIKE THE DELIVERY TRUCK OR AIRPLANE THAT PHYSICALLY CARRIES THE LETTER TO THE NEXT POST OFFICE.

TCP/IP AND NETWORK LAYERS

PRESENTED BY
Ed

5. ENCAPSULATION PROCESS

WHEN DATA MOVES FROM THE APPLICATION DOWN THE STACK, EACH LAYER ADDS ITS OWN HEADER BEFORE PASSING IT TO THE NEXT LAYER. FOR EXAMPLE, AN HTTP REQUEST BECOMES A TCP SEGMENT, THEN AN IP PACKET, AND FINALLY AN ETHERNET FRAME ON THE WIRE. WHEN THE RECEIVING COMPUTER GETS THE FRAME, IT REMOVES THE HEADERS IN REVERSE ORDER UNTIL THE APPLICATION DATA IS RECOVERED.

THIS LAYERED APPROACH ENSURES THAT EACH LAYER HAS CLEAR RESPONSIBILITIES, AND DIFFERENT TECHNOLOGIES CAN WORK TOGETHER SMOOTHLY. FOR EXAMPLE, TCP/IP WORKS ON ETHERNET, WI-FI, OR EVEN CELLULAR NETWORKS BECAUSE THE LINK LAYER CAN CHANGE WHILE HIGHER LAYERS REMAIN THE SAME.

- STEP-BY-STEP EXAMPLE:
  A. APPLICATION LAYER: CREATES DATA (E.G., HTTP REQUEST).
  B. TRANSPORT LAYER: WRAPS IT IN A TCP SEGMENT (ADDS SEQUENCE AND PORT NUMBERS).
  C. INTERNET LAYER: ENCAPSULATES IT IN AN IP PACKET (ADDS SOURCE AND DESTINATION IP).
  D. LINK LAYER: PLACES IT INTO A FRAME (ADDS MAC ADDRESSES AND ERROR-CHECKING INFO).
- AT THE RECEIVING END: THE PROCESS IS REVERSED — THE HEADERS ARE STRIPPED OFF LAYER BY LAYER UNTIL THE ORIGINAL APPLICATION DATA IS RECOVERED.

EXAMPLE (WEB BROWSING):

- USER TYPES WWW.GOOGLE.COM → DNS RESOLVES IP.
- BROWSER SENDS AN HTTP REQUEST (APPLICATION LAYER).
- REQUEST BECOMES A TCP SEGMENT (TRANSPORT LAYER).
- THEN WRAPPED AS AN IP PACKET (INTERNET LAYER).
- FINALLY, SENT AS AN ETHERNET/WI-FI FRAME (LINK LAYER).

PRESENTED BY
Dan

IMPLICATIONS FOR CYBER SECURITY

& SUMMARY

- THE INTERNET AND TCP/IP WERE ORIGINALLY DESIGNED FOR RESEARCHERS AND THE MILITARY.
- SECURITY WAS NOT A MAJOR FOCUS THEN.
- WITH BILLIONS OF USERS TODAY, THIS DESIGN FLAW CREATES SIGNIFICANT CYBER SECURITY CHALLENGES.

IMPLICATIONS

- TCP three-way handshake can be exploited by hackers using fake requests..
- Attacks can overload servers, preventing them from responding to legitimate users.
- IP addresses can reveal approximate geographical locations.
- Collected data is often used for advertising or sold to third parties.

SUMMARY

TRANSPORT AND LAYER PROTOCOL

TCP/IP are the standard transport and network protocol that perform addressing, routing and segmenting.

TRANSPORT LAYER

Uses source and destination port address to link the application layer software to network.

ADDRESSING

Three kind of adressess: application layer address, network layer address and data link layer address.

ADDRESS RESOLUTION

Is the process of translating an application layer address into a network layer address or translating network layer address into a data link layer address.

ROUTING

Is the process of selecting the route or path through the network that a massage will travel from the sending computer to receiving computer.