Computer Networks

Complete Revision Notes — All Topics Covered

Nishant IITH

October 11, 2025

Contents

1 Network Fundamentals

1.1 What is a Computer Network?

Key Point

A **Computer Network (CN)** is a system of interconnected devices that communicate and share resources using standardized protocols. Data is broken into **packets** for transmission.

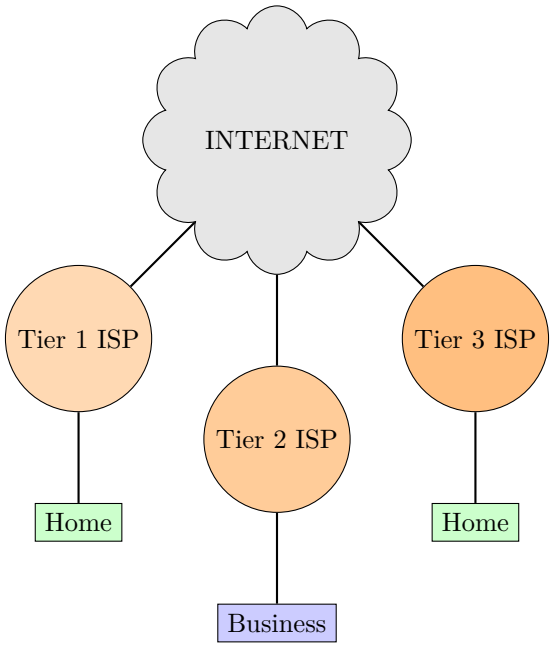
Essential Components:

- **Nodes:** Computers, servers, routers, switches, IoT devices
- **Links:** Physical (cables, fiber) or wireless (WiFi, cellular)
- **Protocols:** Rules for communication (TCP, UDP, HTTP, IP)
- **Data:** Information transferred as packets

1.2 The Internet

Key Point

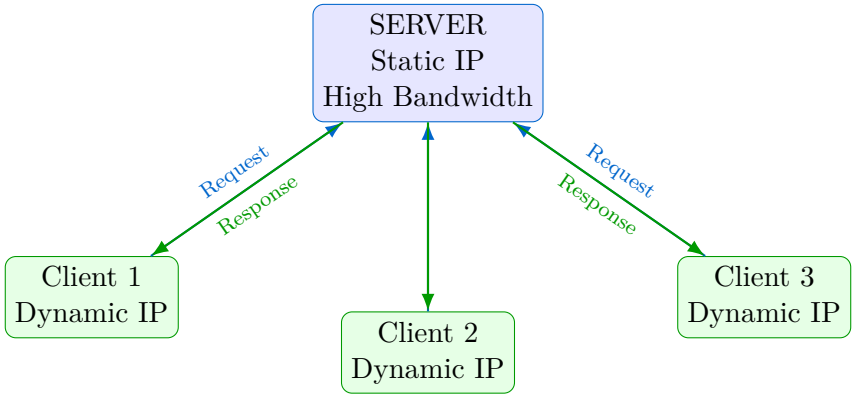
The **Internet** is the global network of networks—the largest WAN connecting billions of devices using TCP/IP protocol suite.



1.3 Client-Server Architecture

Characteristics:

- **Server:** Always-on host with static IP, high bandwidth
- **Clients:** Dynamically connected, may have dynamic IPs
- **Data Centers:** Large server farms owned by organizations
- **Centralized:** Single point of control and data storage



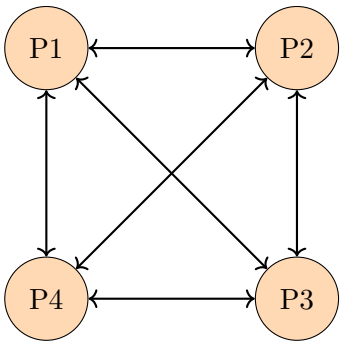
Examples:

- Web servers (Apache, Nginx)
- Email servers (Gmail, Outlook)
- Database servers (MySQL, PostgreSQL)
- File servers (FTP)

1.4 Peer-to-Peer (P2P) Architecture

Key Point

In P2P, every node acts as both client and server. No centralized control—highly scalable and fault-tolerant.



Every peer = Client + Server

Advantages:

- Decentralized—no single point of failure
- Scales automatically as more peers join
- Cost-effective (no dedicated servers)

Disadvantages:

- Security challenges
- Inconsistent performance
- Complex management

Examples:

- **BitTorrent:** File sharing
- **Blockchain:** Distributed ledger
- **Skype (early):** VoIP calls

1.5 Localhost

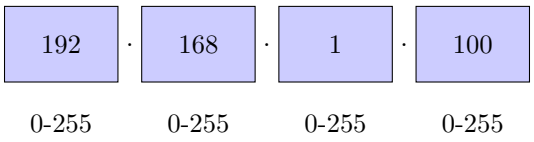
Key Point

Localhost refers to the current machine. The same device acts as both client and server for testing/development.
IPv4: 127.0.0.1 **IPv6:** ::1

2 Network Addressing

2.1 IP Address Structure

IPv4: X.X.X.X (32 bits)



IP Types:

- **Global/Public IP:** ISP-assigned, visible on Internet (e.g., 203.0.113.5)
- **Local/Private IP:** Used within LAN (e.g., 192.168.1.10)
- **IPv4:** 32 bits = 4 octets $\rightarrow \approx 4.3$ billion addresses
- **IPv6:** 128 bits $\rightarrow \approx 3.4 \times 10^{38}$ addresses

Example

Check your public IP:
`$ curl ifconfig.me -s`
or visit: https://whatismyipaddress.com

2.2 IPv4 Address Classes (Legacy)

Class	Start	End	Default Mask	Use
A	0.0.0.0	127.255.255.255	255.0.0.0	Large networks
B	128.0.0.0	191.255.255.255	255.255.0.0	Medium networks
C	192.0.0.0	223.255.255.255	255.255.255.0	Small networks
D	224.0.0.0	239.255.255.255	—	Multicast
E	240.0.0.0	255.255.255.255	—	Experimental

Private IP Ranges (RFC 1918):

- Class A: 10.0.0.0 to 10.255.255.255
- Class B: 172.16.0.0 to 172.31.255.255
- Class C: 192.168.0.0 to 192.168.255.255

2.3 Subnetting and CIDR

Key Point

Subnet mask separates network bits from host bits.
CIDR notation: 192.168.1.0/24 means 24 bits for network, 8 bits for hosts.

Subnet Calculations:

- **/24:** 256 addresses (254 usable: 2-254)
- **/25:** 128 addresses (126 usable)

- **/26:** 64 addresses (62 usable)
- Network address: First IP (e.g., 192.168.1.0)
- Broadcast address: Last IP (e.g., 192.168.1.255)

2.4 IPv6

Key Point

IPv6 uses 128-bit addresses written in hexadecimal, separated by colons.
Example: `2001:0db8:85a3:0000:0000:8a2e:0370:7334`
Shortened: `2001:db8:85a3::8a2e:370:7334`

Why IPv6?

- IPv4 address exhaustion
- Enormous address space
- Simplified header format
- Better security (IPsec built-in)
- No need for NAT

2.5 Port Numbers

Key Point

Ports identify specific applications/services. Combined with IP, they form a **socket**.

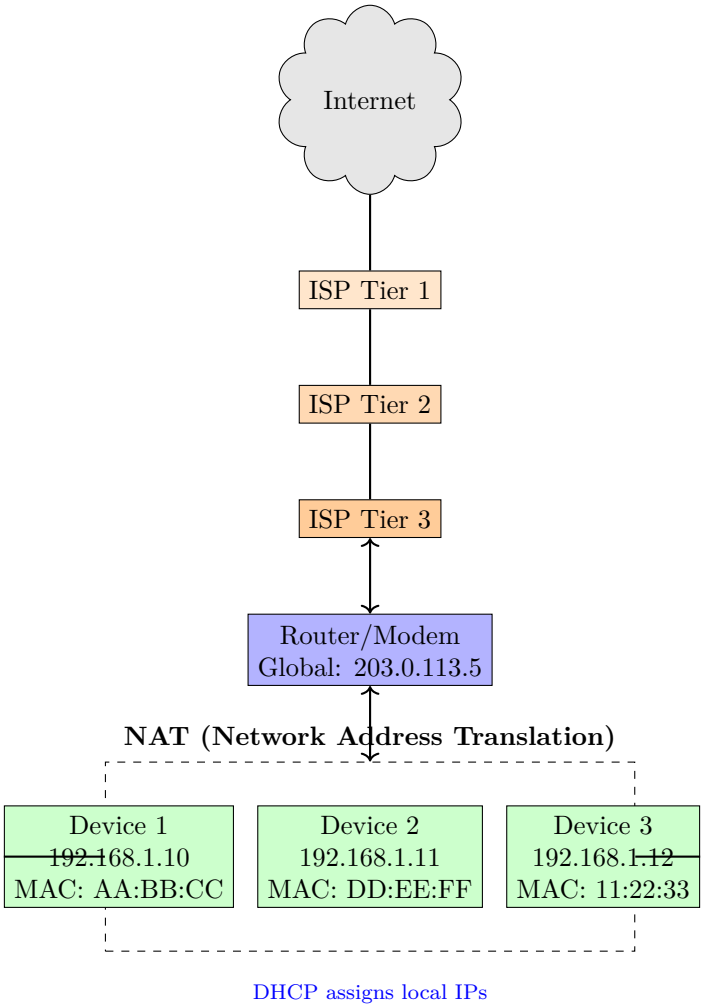
Range	Type	Examples
0–1023	Well-known	HTTP(80), HTTPS(443), SSH(22), FTP(21)
1024–49151	Registered	MongoDB(27017), MySQL(3306)
49152–65535	Dynamic/Ephemeral	Temporary client ports

Example

Complete addressing:
- **Which network?** Global IP (203.0.113.5)
- **Which device?** Local IP (192.168.1.10)
- **Which application?** Port (443 for HTTPS)

Full socket: 192.168.1.10:443

2.6 Complete Network Flow with ISPs



3 Network Hardware & Devices

3.1 Complete Device Comparison

Device	Layer	Forwards by	Function
Repeater	L1	Bits (regenerate)	Amplify/regenerate signals; extend cable reach
Hub	L1	Broadcast all	Connect devices in star; one collision domain (obsolete)
Bridge	L2	MAC table	Segment collision domains; filter by MAC
Switch	L2	MAC table	Modern LAN connectivity; one collision domain per port
Router	L3	IP routing table	Connect different networks; break broadcast domains
Gateway	L7	Protocol/App	Protocol translation; proxy services
Brouter	L2/L3	MAC + IP	Route IP, bridge non-IP protocols

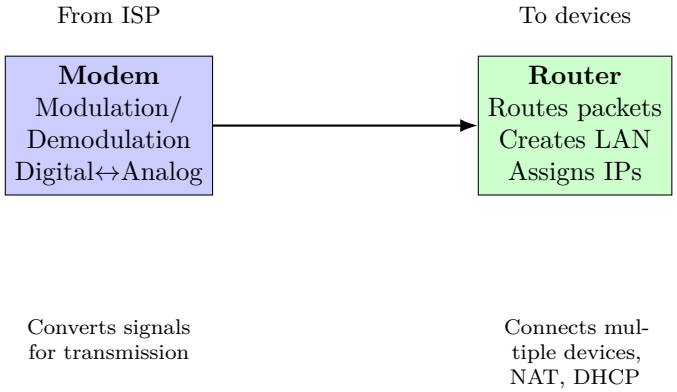
3.2 Collision vs Broadcast Domains

Key Point

Collision Domain: Network segment where packets can collide
Broadcast Domain: Network segment where broadcasts reach all devices

Device	Collision Domains	Broadcast Domains
Hub	1 (all ports)	1
Bridge	Splits collision	Same broadcast
Switch	1 per port	1 (or per VLAN)
Router	Splits collision	Splits broadcast

3.3 Modem vs Router



Note: Modern home routers often integrate modem functionality (combo units).

3.4 Communication Media

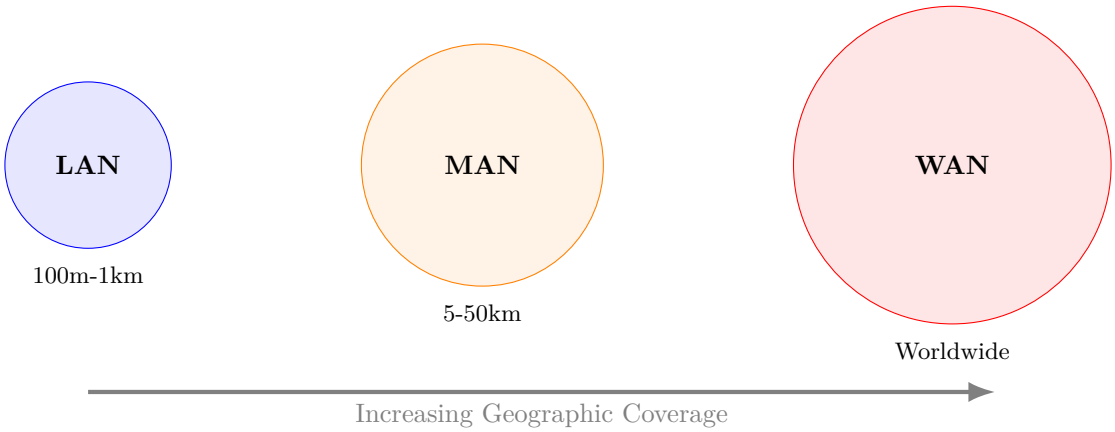
Wired Media:

- **Optical Fiber:**
  - Fastest speeds (up to 100+ Gbps)
  - Long distance (100+ km without repeaters)
  - Immune to electromagnetic interference
  - Single-mode (long distance) vs Multi-mode (shorter)
- **Ethernet Cables:**
  - Cat5e: 1 Gbps, 100m max
  - Cat6: 10 Gbps, 55m max
  - Cat6a: 10 Gbps, 100m max
- **Coaxial Cable:** Cable TV, older networks

Wireless Media:

- **WiFi (802.11):**
  - 802.11n: 600 Mbps
  - 802.11ac: 1-7 Gbps
  - 802.11ax (WiFi 6): 9.6 Gbps
- **Bluetooth:** Short-range (10-100m), low power
- **Cellular:** 4G LTE (100 Mbps), 5G (1-10 Gbps)
- **Satellite:** Wide coverage, high latency

4 Network Types by Scale



Type	Coverage	Speed	Examples
PAN	Personal (10m)	High	Bluetooth, USB
LAN	Building/Campus	Very High	Home, Office WiFi
MAN	City	Medium-High	City fiber networks
WAN	Country/World	Medium	Internet, corporate WANs

Interview Question

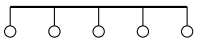
Q: Difference between LAN and WAN?

A: LAN covers small areas (building/campus) with high speed and low latency, typically owned by one organization. WAN covers large geographic areas (countries/continents) with relatively lower speed and higher latency. The Internet is the largest WAN, connecting millions of LANs worldwide.

5 Network Topologies

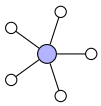
5.1 Common Topologies with Analysis

Bus Topology



✓ Simple, cheap
✓ Easy to extend
× Single point of failure
× Performance degrades with load

Star Topology

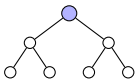


✓ Easy to add/remove nodes
✓ Fault isolation
× Hub dependency
× More cable required

Ring Topology

✓ Equal access time
✓ No collisions
× Single break affects all
× Unidirectional data flow

Tree (Hierarchical)



✓ Hierarchical organization
✓ Scalable
× Root node critical
× Complex cabling

Mesh Topology

✓ High redundancy
✓ Fault tolerant
× Expensive (many cables)
× Complex installation

5.2 Topology Selection Criteria

Topology	Cost	Reliability	Scalability	Best Use
Bus	Low	Low	Poor	Small, temporary networks
Star	Medium	Medium-High	Good	Modern LANs, offices
Ring	Medium	Medium	Medium	Token Ring (legacy)
Tree	Medium-High	Medium	Excellent	Large organizations
Mesh	Very High	Excellent	Limited	Critical infrastructure

6 The OSI Model — Complete Reference

Memory Aid

OSI = Open Systems Interconnection

Top to Bottom: "All People Seem To Need Data Processing"
(Application, Presentation, Session, Transport, Network, Data Link, Physical)

Bottom to Top: "Please Do Not Throw Sausage Pizza Away"
(Physical, Data Link, Network, Transport, Session, Presentation, Application)

6.1 Complete OSI Layer Breakdown

Layer 7: Application — User interface, network services
Protocols: HTTP, HTTPS, FTP, SMTP, DNS, DHCP, SSH, Telnet
Examples: Web browsers, Email clients, Skype, WhatsApp

Layer 6: Presentation — Data translation, encryption, compression
Functions: SSL/TLS, ASCII/EBCDIC conversion, JPEG, MPEG
Data format: Converts to machine-readable format

Layer 5: Session — Connection management
Functions: Setup, maintain, terminate sessions
Authentication, authorization, session recovery

Layer 4: Transport — End-to-end delivery (Segments)
Protocols: TCP (reliable), UDP (fast)
Functions: Segmentation, flow control, error control, ports

Layer 3: Network — Logical addressing, routing (Packets)
Protocols: IP (IPv4/IPv6), ICMP, ARP, OSPF, BGP
Devices: Routers, Layer 3 switches

Layer 2: Data Link — Physical addressing (Frames)
Protocols: Ethernet, WiFi (802.11), PPP
Functions: MAC addressing, error detection (CRC), media access
Devices: Switches, Bridges, NICs

Layer 1: Physical — Bit transmission
Components: Cables, hubs, repeaters, connectors
Specifications: Voltage levels, cable types, frequencies

6.2 Detailed Layer Functions

Layer 7 — Application Layer:

- Direct user interaction with network
- Implemented in software applications
- Provides network services to applications
- **Key Protocols:**
  - HTTP/HTTPS: Web browsing
  - FTP: File transfer
  - SMTP/POP3/IMAP: Email
  - DNS: Domain name resolution
  - DHCP: Dynamic IP assignment
  - SSH: Secure remote login

Layer 6 — Presentation Layer:

- **Translation:** ASCII to EBCDIC, character encoding

- **Encryption/Decryption:** SSL/TLS protocols
- **Compression:** Reduce data size (gzip, JPEG, MPEG)
- **Data Formatting:** JSON, XML, ASN.1
- Ensures data is readable by receiving system

Layer 5 — Session Layer:

- **Session Management:** Setup, maintain, terminate
- **Dialog Control:** Half-duplex or full-duplex
- **Synchronization:** Checkpoints for recovery
- **Authentication:** User verification
- **Authorization:** Access control
- Examples: NetBIOS, RPC (Remote Procedure Call)

Layer 4 — Transport Layer:

- **Segmentation:** Divide data into manageable segments
- **Port Numbers:** Identify applications (0-65535)
- **Flow Control:** Match sender/receiver speeds (sliding window)
- **Error Control:** Checksums, retransmission
- **Multiplexing/Demultiplexing:** Multiple apps on same host
- **TCP:** Connection-oriented, reliable
- **UDP:** Connectionless, fast, best-effort

Example

Flow Control Scenario:
Client sends at 50 Mbps, but server can only receive at 20 Mbps.
Transport layer adjusts transmission rate to 20 Mbps to prevent buffer overflow.

Layer 3 — Network Layer:

- **Logical Addressing:** IP addresses (IPv4: 32-bit, IPv6: 128-bit)
- **Routing:** Path determination using routing tables
- **Packet Forwarding:** Move packets between networks
- **Load Balancing:** Distribute traffic across paths
- **Fragmentation:** Split packets for different MTUs
- **Protocols:** IP, ICMP (ping), ARP, OSPF, BGP
- **Devices:** Routers, Layer 3 switches

Layer 2 — Data Link Layer:

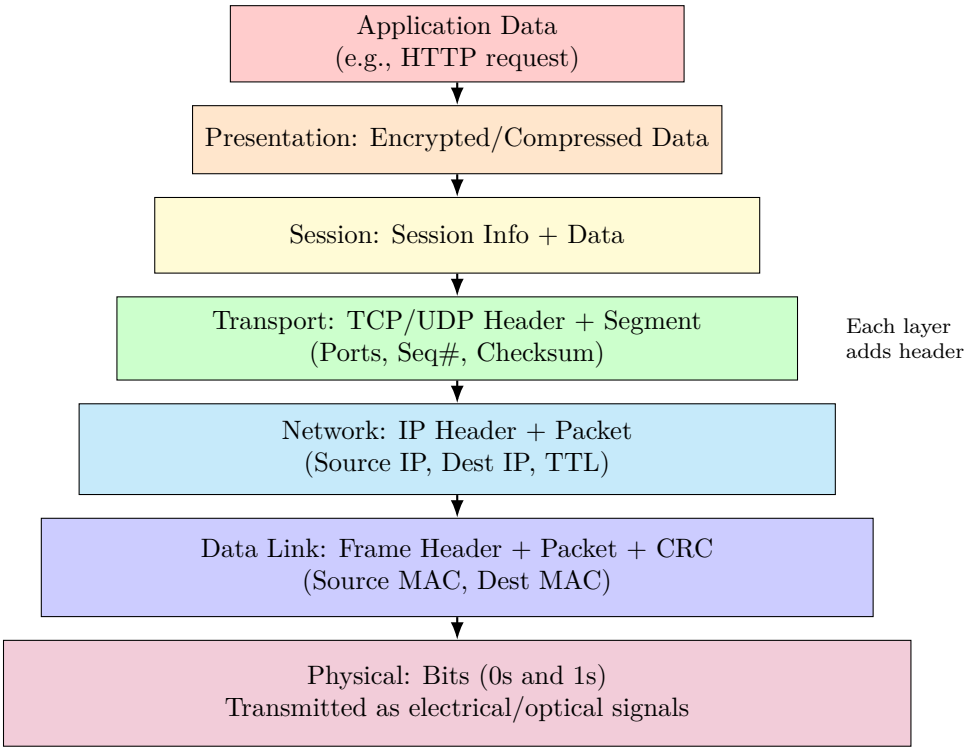
- **Physical Addressing:** MAC addresses (48-bit)
- **Framing:** Encapsulate packets into frames
- **Error Detection:** CRC (Cyclic Redundancy Check)
- **Media Access Control:** CSMA/CD (Ethernet), CSMA/CA (WiFi)
- **Flow Control:** Prevent receiver overflow
- **Protocols:** Ethernet, WiFi (802.11), PPP
- **Devices:** Switches, Bridges, NICs
- **Note:** Device can have multiple MACs (Ethernet, WiFi, Bluetooth)

Layer 1 — Physical Layer:

- **Bit Transmission:** Convert frames to electrical/optical signals
- **Physical Specifications:** Voltage, frequency, cable types
- **Topology:** Physical network layout
- **Transmission Mode:** Simplex, half-duplex, full-duplex

- **Components:** Cables, hubs, repeaters, connectors, NICs
- **Media:** Copper, fiber optic, wireless (radio waves)

6.3 Data Encapsulation Process



Decapsulation: Reverse process at receiving end—each layer removes its header.

6.4 TCP/IP Model (Practical Implementation)

Key Point

The Internet actually uses the **TCP/IP model** (4-5 layers), which is more practical than the theoretical OSI model.

Application Layer: HTTP, DNS, SMTP, FTP, SSH
(Combines OSI Layers 5, 6, 7)

Transport Layer: TCP, UDP, QUIC
(Same as OSI Layer 4)

Internet Layer: IP, ICMP, ARP
(Same as OSI Layer 3)

Link/Network Access Layer: Ethernet, WiFi
(Combines OSI Layers 1, 2)

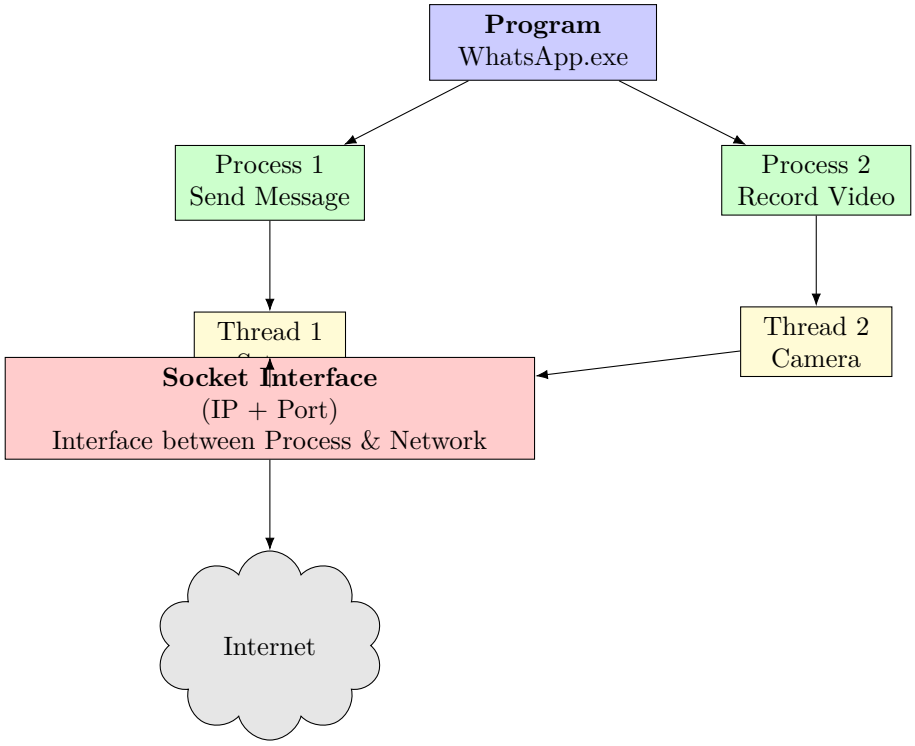
Physical Layer: Hardware, cables, signals
(Sometimes included in Link layer)

6.5 OSI vs TCP/IP Mapping

OSI Model (7 Layers)	TCP/IP Model (4-5 Layers)
7. Application 6. Presentation 5. Session	Application
4. Transport	Transport
3. Network	Internet
2. Data Link 1. Physical	Link/Network Access

7 Application Layer Protocols — Deep Dive

7.1 Programs, Processes, Threads, and Sockets



Definitions:

- **Program:** Executable file on disk (static code)
- **Process:** Running instance of program in memory
- **Thread:** Lightweight unit of execution within a process
- **Socket:** Communication endpoint = IP address + Port number

Key Point

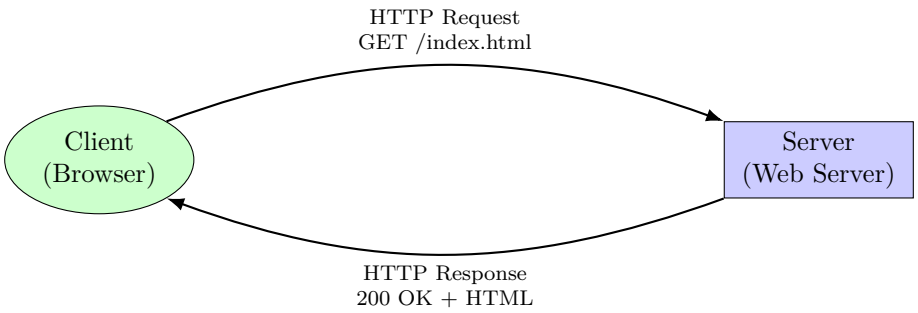
Socket Address: IP:Port (e.g., 192.168.1.10:52413)
Ephemeral Ports: Temporary ports (49152-65535) assigned to clients during communication

7.2 HTTP — HyperText Transfer Protocol

Key Point

HTTP is the foundation of the World Wide Web—a **stateless**, **application-layer** protocol that runs over TCP.

HTTP Request-Response Cycle:



HTTP Methods (Verbs):

- **GET:** Retrieve resource from server (idempotent)
- **POST:** Submit data to server (create resource)
- **PUT:** Update/replace resource (idempotent)
- **DELETE:** Remove resource (idempotent)
- **HEAD:** Get headers only (no body)
- **PATCH:** Partial update
- **OPTIONS:** Get allowed methods

HTTP Status Codes:

Code Range	Meaning	Examples
1xx	Informational	100 Continue, 101 Switching Protocols
2xx	Success	200 OK, 201 Created, 204 No Content
3xx	Redirection	301 Moved Permanently, 302 Found, 304 Not Modified
4xx	Client Error	400 Bad Request, 401 Unauthorized, 403 Forbidden, 404 Not Found
5xx	Server Error	500 Internal Server Error, 502 Bad Gateway, 503 Service Unavailable

HTTP vs HTTPS:

- **HTTP:** Port 80, unencrypted, plain text
- **HTTPS:** Port 443, encrypted with TLS/SSL, secure

7.3 Cookies — Maintaining State

Key Point

HTTP is **stateless**—each request is independent. **Cookies** store unique identifiers to maintain session state.

How Cookies Work:

1. Server sends `Set-Cookie` header in response
2. Browser stores cookie (key-value pair + metadata)
3. Browser sends cookie with subsequent requests
4. Server uses cookie to identify user/session

Cookie Types:

- **Session Cookies:** Temporary, deleted when browser closes
- **Persistent Cookies:** Have expiration date, stored on disk
- **First-party Cookies:** Set by visited website
- **Third-party Cookies:** Set by external domains (ads, analytics)

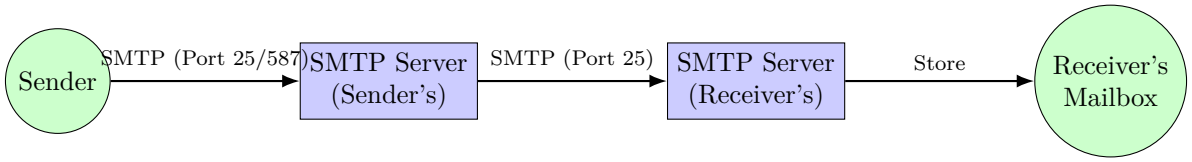
Example

When you log into Amazon:

1. You enter credentials
2. Server validates and creates session
3. Server sends cookie with session ID
4. Browser includes cookie in future requests
5. You remain logged in until cookie expires

7.4 Email Protocols — Complete Flow

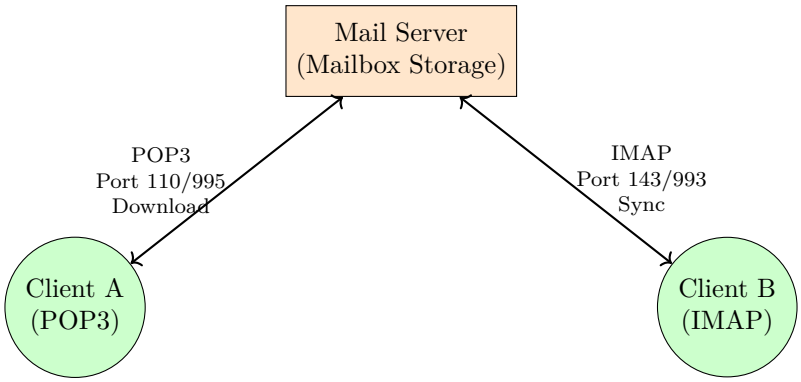
Sending Email (SMTP):



SMTP (Simple Mail Transfer Protocol):

- **Purpose:** Sending/relaying emails
- **Ports:** 25 (server-to-server), 587 (client submission), 465 (SMTPS)
- **Transport:** TCP (reliable delivery required)
- **Push Protocol:** Sender pushes to receiver's server

Receiving Email:



POP3 (Post Office Protocol v3):

- **Model:** Download and delete from server
- **Ports:** 110 (plain), 995 (POP3S with SSL/TLS)
- **Use Case:** Single device, limited server storage
- **Modes:** Delete (default) or Keep copy on server

IMAP (Internet Message Access Protocol):

- **Model:** Synchronize across multiple devices
- **Ports:** 143 (plain), 993 (IMAPS with SSL/TLS)
- **Features:** Folder management, partial download, search on server
- **Use Case:** Multiple devices (phone, laptop, tablet)

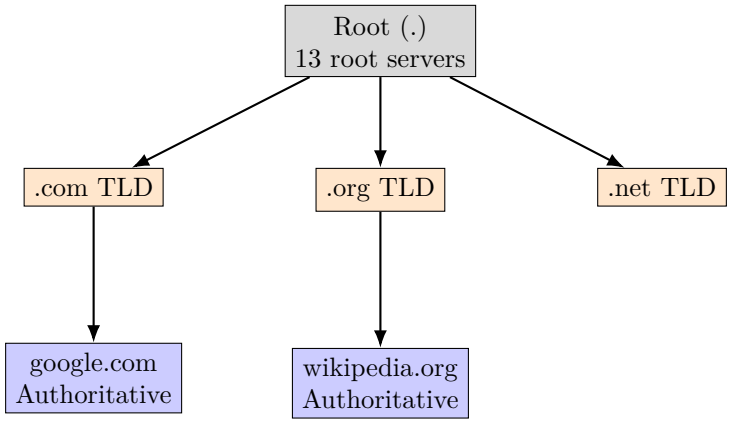
Feature	POP3	IMAP
Storage	Local (downloaded)	Server (synchronized)
Multiple Devices	Poor support	Excellent support
Server Space	Conserves space	Requires more space
Folders	Local only	Server-side folders
Speed	Faster initial download	Faster access (partial fetch)

7.5 DNS — Domain Name System

Key Point

DNS translates human-readable domain names (www.google.com) into machine-readable IP addresses (142.250.77.206).

DNS Hierarchy:



DNS Resolution Process:

1. **Browser Cache:** Check if IP is cached
2. **OS Cache:** Check operating system's DNS cache
3. **Hosts File:** Check /etc/hosts (Unix) or C:\Windows\System32\drivers\etc\hosts
4. **Recursive Resolver:** Query ISP or public DNS (8.8.8.8, 1.1.1.1)
5. **Root Server:** Points to appropriate TLD server
6. **TLD Server:** Points to authoritative nameserver
7. **Authoritative Server:** Returns IP address
8. **Cache Result:** Store for TTL (Time To Live) period

DNS Record Types:

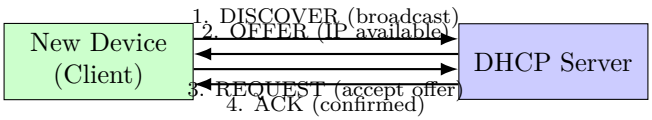
- **A:** IPv4 address (e.g., google.com → 142.250.77.206)
- **AAAA:** IPv6 address
- **CNAME:** Canonical name (alias)
- **MX:** Mail exchange server
- **NS:** Nameserver
- **TXT:** Text records (SPF, DKIM)
- **PTR:** Reverse DNS lookup

7.6 DHCP — Dynamic Host Configuration Protocol

Key Point

DHCP automatically assigns IP addresses and network configuration to devices joining a network.

DORA Process:



DHCP Lease Information Includes:

- **IP Address:** Assigned to the device
- **Subnet Mask:** Network/host separation

- **Default Gateway:** Router IP for external communication
- **DNS Servers:** For name resolution
- **Lease Time:** How long IP is valid (typically hours/days)

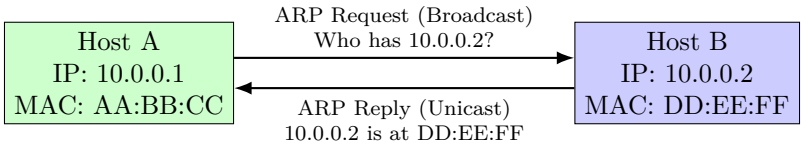
Ports: UDP 67 (server), UDP 68 (client)

7.7 ARP — Address Resolution Protocol

Key Point

ARP maps IP addresses to MAC addresses within the same local network (subnet).

ARP Process:



ARP Cache/Table:

- Stores recent IP-to-MAC mappings
- Reduces ARP broadcasts
- Entries expire after timeout (typically 2-20 minutes)
- View with: `arp -a` (Windows/Linux)

7.8 FTP — File Transfer Protocol

Key Point

FTP transfers files between client and server using two separate channels.

FTP Channels:

- **Control Channel:** Port 21 (commands, responses)
- **Data Channel:** Port 20 (actual file transfer)

FTP Modes:

- **Active Mode:** Server initiates data connection
- **Passive Mode:** Client initiates data connection (firewall-friendly)

Secure Alternatives:

- **SFTP:** SSH File Transfer Protocol (Port 22)
- **FTPS:** FTP over SSL/TLS (Ports 989/990)

7.9 SSH — Secure Shell

Key Point

SSH provides secure remote login and command execution with encryption.

Features:

- **Port:** 22 (TCP)
- **Encryption:** All traffic encrypted
- **Authentication:** Password or public key
- **Uses:** Remote terminal, file transfer (SCP, SFTP), tunneling

7.10 Telnet (Legacy)

- **Port:** 23 (TCP)

- **Security:** None—plain text transmission
- **Status:** Obsolete, replaced by SSH
- **Use Today:** Testing/debugging only

8 Transport Layer — Complete Reference

8.1 Transport Layer Overview

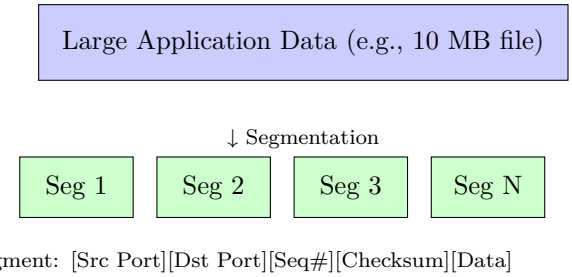
Key Point

Provides **process-to-process** delivery using port numbers. Ensures data reaches the correct application.

Key Functions:

- **Segmentation:** Divide application data into segments
- **Multiplexing:** Multiple apps share network connection
- **Demultiplexing:** Deliver segments to correct application (by port)
- **Error Detection:** Checksums verify data integrity
- **Flow Control:** Match sender/receiver speeds
- **Congestion Control:** Prevent network overload

8.2 Segmentation Process



Why Segmentation?

- Networks have MTU (Maximum Transmission Unit) limits
- Enables error recovery (retransmit only failed segments)
- Allows multiplexing of multiple streams
- Improves efficiency and reliability

8.3 TCP — Transmission Control Protocol

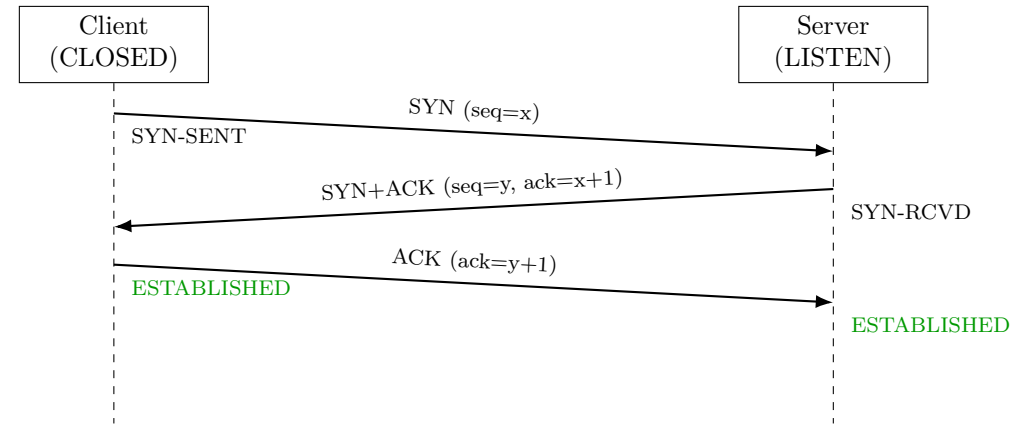
Key Point

TCP is **connection-oriented** and provides **reliable**, **ordered** delivery with **flow** and **congestion control**.

TCP Characteristics:

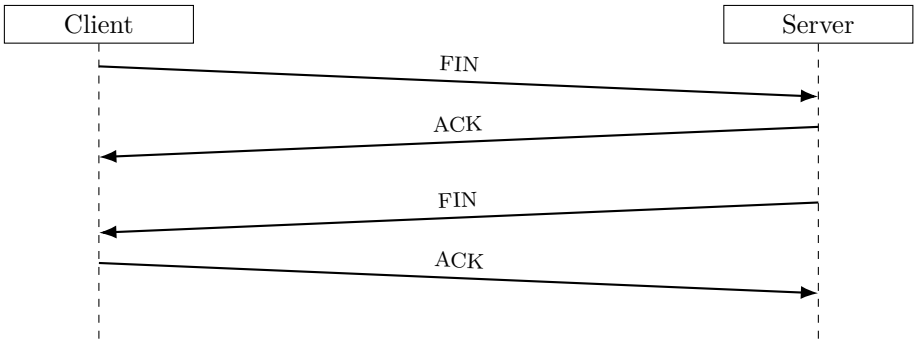
- **Connection-oriented:** 3-way handshake before data transfer
- **Reliable:** Acknowledgments and retransmissions
- **Ordered:** Sequence numbers ensure correct order
- **Full-duplex:** Bidirectional communication
- **Flow Control:** Sliding window protocol
- **Congestion Control:** CUBIC, BBR algorithms
- **Error Detection:** Checksums

TCP Three-Way Handshake:



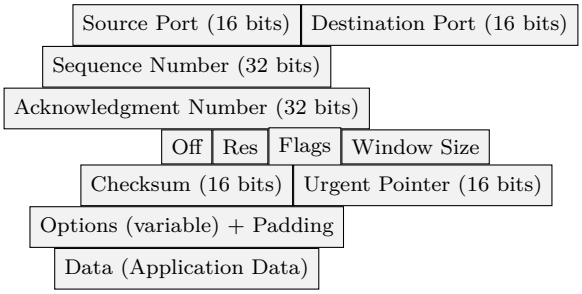
Connection Ready!

TCP Connection Termination (4-way handshake):



Connection Closed

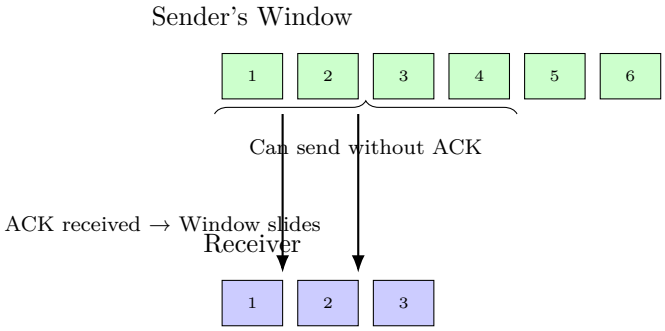
TCP Header Structure:



TCP Flags:

- **SYN:** Synchronize sequence numbers (connection setup)
- **ACK:** Acknowledgment field valid
- **FIN:** Finish, no more data (connection teardown)
- **RST:** Reset connection (abort)
- **PSH:** Push data immediately to application
- **URG:** Urgent pointer field valid

Sliding Window (Flow Control):



Congestion Control:

- **Slow Start:** Exponentially increase window size
- **Congestion Avoidance:** Linear increase after threshold
- **Fast Retransmit:** Retransmit on 3 duplicate ACKs
- **Fast Recovery:** Halve window on packet loss
- **Algorithms:** CUBIC (default), BBR (Google)

8.4 UDP — User Datagram Protocol

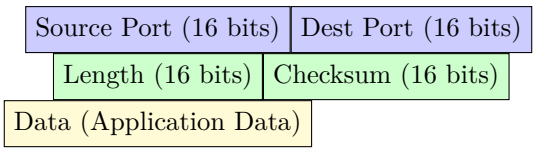
Key Point

UDP is **connectionless**, **unreliable**, but **fast**. No handshakes, no acknowledgments, minimal overhead.

UDP Characteristics:

- **Connectionless:** No setup required
- **Unreliable:** No acknowledgments or retransmissions
- **No ordering:** Datagrams may arrive out of order
- **Lightweight:** 8-byte header (vs TCP's 20+ bytes)
- **Fast:** Low latency, no connection overhead
- **Broadcast/Multicast:** Supports one-to-many

UDP Header (8 bytes):



UDP Use Cases:

- **DNS:** Quick queries (one request, one response)
- **Streaming:** Video/Audio (RTP) - late packets useless
- **Gaming:** Real-time position updates
- **VoIP:** Voice over IP (Skype, Zoom audio)
- **IoT:** Sensor data, telemetry
- **DHCP:** IP address assignment
- **TFTP:** Trivial File Transfer Protocol

8.5 TCP vs UDP — Complete Comparison

Feature	TCP	UDP
Connection	Connection-oriented (handshake)	Connectionless
Reliability	Guaranteed delivery, retransmission	Best-effort, no guarantees
Ordering	In-order delivery via sequence numbers	No ordering
Speed	Slower (overhead)	Faster (minimal overhead)
Header Size	20-60 bytes	8 bytes
Flow Control	Yes (sliding window, rwnd)	No
Congestion Control	Yes (CUBIC, BBR)	No
Error Checking	Yes (checksum + retransmit)	Checksum only
Broadcasting	No	Yes
Use Cases	Web (HTTP), Email (SMTP), File Transfer (FTP), SSH	Streaming, Gaming, DNS, VoIP, DHCP
When to Use	Reliability > Speed	Speed > Reliability

Interview Question

Q: When would you choose UDP over TCP?

A: Choose UDP when:

- Low latency is critical (gaming, VoIP)
- Real-time data where old data is useless (video streaming)
- Small transactions (DNS queries)
- Broadcast/multicast needed
- Application implements its own reliability (QUIC)

Choose TCP when data integrity and order matter (web, email, file transfer).

9 Network Layer — Routing & IP

9.1 Network Layer Functions

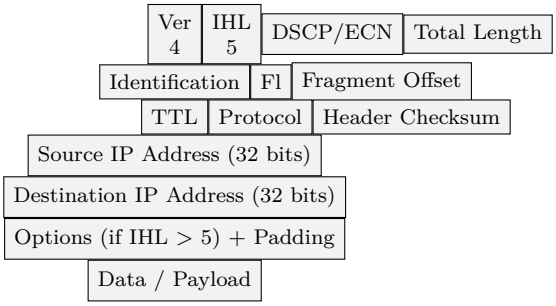
Key Point

The Network layer handles **end-to-end** packet delivery across multiple networks using **logical addressing** (IP) and **routing**.

Key Responsibilities:

- **Logical Addressing:** IP addresses identify hosts globally
- **Routing:** Path selection using routing tables
- **Packet Forwarding:** Move packets hop-by-hop
- **Fragmentation:** Split packets for different MTUs
- **Error Reporting:** ICMP messages
- **Quality of Service:** Traffic prioritization

9.2 IPv4 Packet Structure



Important IPv4 Header Fields:

- **Version:** 4 for IPv4, 6 for IPv6
- **IHL:** Internet Header Length (5 = 20 bytes minimum)
- **Total Length:** Entire packet size (max 65,535 bytes)
- **TTL:** Time To Live—decremented at each hop, prevents loops
- **Protocol:** Upper layer protocol (6=TCP, 17=UDP, 1=ICMP)
- **Header Checksum:** Error detection for header only
- **Source/Dest IP:** 32-bit addresses

9.3 Routing — Path Selection

Routing Process:

1. Router receives packet
2. Examines destination IP address
3. Looks up routing table for best match
4. Decrements TTL (discard if TTL=0)
5. Forwards to next hop
6. Updates packet headers (new MAC addresses at L2)

Routing Table Example:

Destination Network	Subnet Mask	Next Hop	Interface
192.168.1.0	255.255.255.0	192.168.1.1	eth0
10.0.0.0	255.0.0.0	10.0.0.1	eth1
0.0.0.0	0.0.0.0	203.0.113.1	eth2 (default)

Routing Types:

1. Static Routing:

- Manually configured by administrator
- Simple, predictable, secure
- Best for small networks
- No automatic updates if topology changes

2. Dynamic Routing:

- Automatically discovers and updates routes
- Adapts to network changes
- **Protocols:**
  - **RIP:** Distance vector, max 15 hops
  - **OSPF:** Link state, fast convergence
  - **BGP:** Path vector, Internet backbone

9.4 Control Plane vs Data Plane

Key Point

Control Plane: Builds routing tables (routing protocols)
Data Plane: Forwards packets using those tables (fast path)

9.5 ICMP — Internet Control Message Protocol

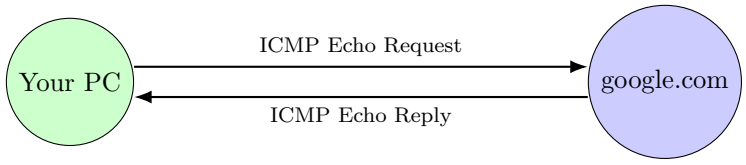
Purpose: Error reporting and diagnostics

Common ICMP Messages:

- **Echo Request/Reply (Type 8/0):** Used by `ping`
- **Destination Unreachable (Type 3):** Host/network/port unreachable
- **Time Exceeded (Type 11):** TTL expired (used by `traceroute`)

- **Redirect (Type 5):** Better route available

Ping — Round Trip Time Measurement:



Measure RTT (Round Trip Time)

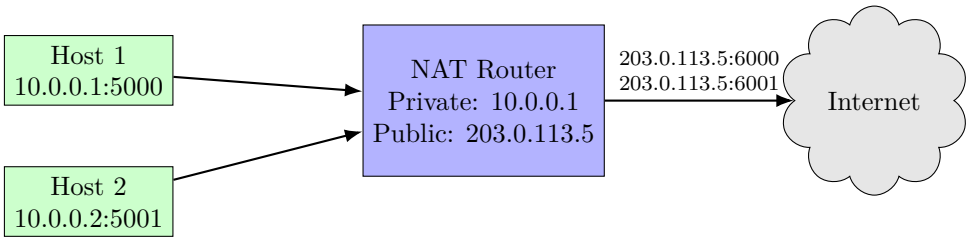
Traceroute — Path Discovery:

- Sends packets with increasing TTL (1, 2, 3, ...)
- Each router decrements TTL and sends ICMP Time Exceeded
- Maps the path to destination

9.6 NAT — Network Address Translation

Key Point

NAT maps private IP addresses to public IP addresses, allowing multiple devices to share one global IP.



NAT Types:

- **Static NAT:** One-to-one mapping (one private → one public)
- **Dynamic NAT:** Many-to-many from pool
- **PAT (Port Address Translation):** Many-to-one using ports (most common)

Benefits:

- Conserves IPv4 addresses
- Provides basic security (hides internal structure)
- Allows private networks to access Internet

Drawbacks:

- Breaks end-to-end connectivity
- Complicates peer-to-peer applications
- Adds latency and processing overhead

9.7 Firewalls

Types:

- **Stateless:** Inspects each packet independently
- **Stateful:** Tracks connection state (more secure)

Functions:

- Filter traffic based on IP, port, protocol
- Block unauthorized access
- Log security events

10 Data Link Layer

10.1 Data Link Functions

Key Point

Provides **physical addressing** (MAC), **framing**, and **error detection** for local network delivery.

10.2 MAC Address

- **48-bit** hardware address (6 bytes)
- Format: `AA:BB:CC:DD:EE:FF`
- First 3 bytes: Manufacturer ID (OUI)
- Last 3 bytes: Device-specific
- Unique for each network interface

10.3 Ethernet Frame Structure

Fields:

- **Type:** Protocol (0x0800=IPv4, 0x0806=ARP, 0x86DD=IPv6)
- **CRC:** Cyclic Redundancy Check for error detection

10.4 Collision vs Broadcast Domains

Collision Domain:

- Network segment where data packets can collide
- Hubs create one large collision domain
- Switches separate collision domains (one per port)

Broadcast Domain:

- Network segment where broadcasts reach all devices
- Switches keep same broadcast domain
- Routers separate broadcast domains
- VLANs can divide broadcast domains on switches

11 Practical Scenarios

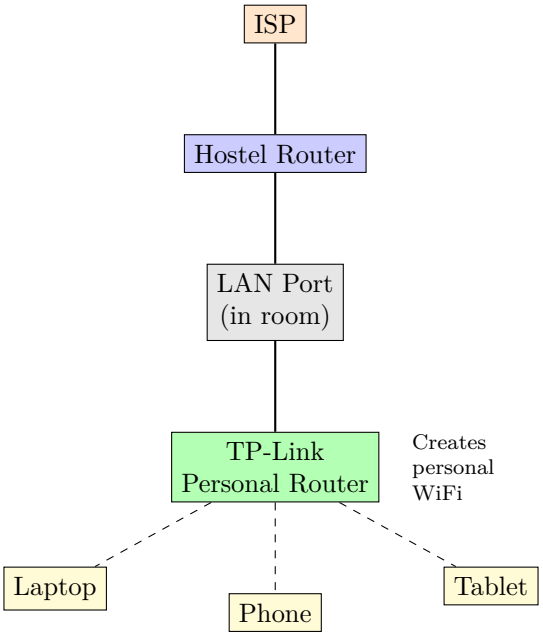
11.1 What Happens When You Type www.google.com?

1. **DNS Resolution:**
   - Browser checks cache
   - Queries recursive resolver
   - Resolver queries Root → TLD (.com) → Authoritative
   - Returns IP address (e.g., 142.250.77.206)
2. **TCP Connection:**
   - Three-way handshake with Google's server
   - Establishes connection on port 443 (HTTPS)
3. **TLS Handshake:**
   - Negotiate encryption algorithms
   - Verify server certificate
   - Establish encrypted session
4. **HTTP Request:**

- Browser sends GET / HTTP/1.1
- Includes headers (User-Agent, Accept, Cookies)

5. **Server Processing:**
   - Server processes request
   - Generates HTML response
   - Sends HTTP 200 OK with content
6. **Rendering:**
   - Browser parses HTML
   - Builds DOM tree
   - Fetches CSS, JavaScript, images
   - Renders page

11.2 Hostel WiFi Setup Example



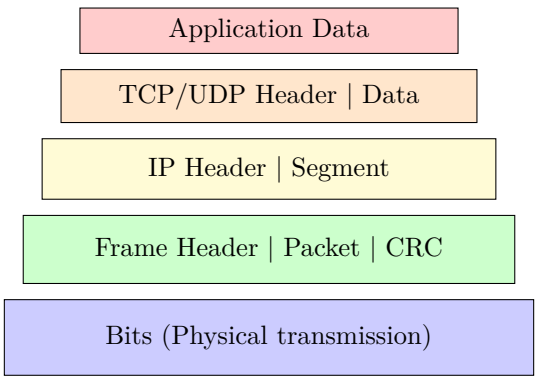
Process:

1. Hostel router assigns IP to TP-Link via DHCP
2. TP-Link acts as DHCP server for your devices
3. Each device gets unique local IP (192.168.x.x)
4. MAC addresses identify devices at hardware level

12 Key Networking Concepts

12.1 Encapsulation

Data moves down the OSI layers, each adding its header:



12.2 Sockets

Key Point

A **socket** is an endpoint for network communication, identified by:

$$\text{Socket} = \text{IP Address} + \text{Port Number}$$

Example:

- Client socket: 192.168.1.10:52413
- Server socket: 142.250.77.206:443
- Together they form a unique connection

12.3 Cookies

Key Point

Cookies maintain state in stateless HTTP protocol.
Store session IDs, preferences, tracking data.

Types:

- **First-party:** Set by visited website
- **Third-party:** Set by external domains (ads, analytics)

12.4 Localhost

- **IP:** 127.0.0.1 (IPv4) or ::1 (IPv6)
- **Purpose:** Loopback address—same machine acts as client and server
- **Use:** Testing, development, local services

13 Protocol Reference Table

Protocol	Layer	Port	Transport	Purpose
HTTP	Application	80	TCP	Web traffic
HTTPS	Application	443	TCP	Secure web
FTP	Application	20/21	TCP	File transfer
SSH	Application	22	TCP	Secure shell
Telnet	Application	23	TCP	Remote terminal
SMTP	Application	25/587	TCP	Send email
DNS	Application	53	UDP/TCP	Name resolution
DHCP	Application	67/68	UDP	IP assignment
POP3	Application	110/995	TCP	Retrieve email
IMAP	Application	143/993	TCP	Sync email
TCP	Transport	—	—	Reliable delivery
UDP	Transport	—	—	Fast delivery
IP	Network	—	—	Logical addressing
ICMP	Network	—	—	Error reporting
ARP	Data Link	—	—	IP to MAC

14 Quick Reference

14.1 OSI vs TCP/IP

OSI Model	TCP/IP Model
7. Application 6. Presentation 5. Session	Application
4. Transport	Transport
3. Network	Internet
2. Data Link 1. Physical	Link/Network Access

14.2 PDU (Protocol Data Unit) Names

- **Application Layer:** Data/Message
- **Transport Layer:** Segment (TCP) / Datagram (UDP)
- **Network Layer:** Packet
- **Data Link Layer:** Frame
- **Physical Layer:** Bits

14.3 Network Commands

Useful networking commands:

- `ping <host>` — Test connectivity and measure RTT
- `traceroute <host>` — Show path packets take
- `nslookup <domain>` — Query DNS records
- `ipconfig` (Windows) / `ifconfig` (Linux) — Show network config
- `netstat` — Show network connections and ports
- `curl ifconfig.me` — Get public IP address
- `arp -a` — Display ARP cache

14.4 Memory Aids

Memory Aid

OSI Layers (Bottom to Top):
"Please Do Not Throw Sausage Pizza Away"
(Physical, Data Link, Network, Transport, Session, Presentation, Application)
OSI Layers (Top to Bottom):
"All People Seem To Need Data Processing"
(Application, Presentation, Session, Transport, Network, Data Link, Physical)