Networking

Before beginning please take note of the following:

- **Chat GPT is used**
- **Read cautiously as details are not verified**
- **This is only a brief introduction so please continue researching**

Start of Notes

NETWORKING GUIDE

1. Introduction to Networking

- Networking connects devices to **share data and resources**.

- **Internet** = global network
- **Web (WWW)** = collection of webpages
- **Origin**: ARPANET → TCP/IP → modern Internet
- **Protocols**: TCP/IP
- **Key concepts**: HTTP/HTTPS, URLs, hyperlinks, search engines

Cross-reference: Diagram 1 shows **Client-Server & HTTP flow (see next page!)**

GET:

```
[Client Browser]
      |
      |  GET /index.html
      v
   [Router/NAT]
      |
      |  Forward to Server IP
      v
   [Server]
      |
      |  Response: HTML page
      v
[Router/NAT]
      |
      v
[Client Browser]
      |
   Displays page
```

POST:

```
 [Client Browser]
       |
       |  POST /form-data
       v
    [Router/NAT]
       |
       v
    [Server]
       |
       |  Response: Success/Failure
       v
 [Router/NAT]
       |
       v
 [Client Browser]
```

2. Client–Server Architecture

- Client = requests data
- Server = provides data

Flow Example (See Diagram 1: Client-Server / HTTP Flow(right above this page):

1. Browser requests google.com
2. Server receives request → processes it
3. Server sends back webpage
4. Browser displays it

Key idea: request → process → response

3. Protocols

- **TCP(Transmission Control Protocol)**: reliable, ordered (web browsing, emails)
- **UDP(User Datagram Protocol)**: fast, no guarantee (gaming, streaming)
- **HTTP / HTTPS(HyperText Transfer Protocol and/or Secure)**: transfer webpages (HTTPS secure)

Protocol	Port	Use
HTTP	80	Webpages
HTTPS	443	Secure webpages
DNS	53	Domain name resolution
FTP	21	File transfer
SSH	22	Secure remote login

4. Data Transmission & Packets

- Data is split into **packets**
- Each packet contains: Source IP, Destination IP, Ports, Sequence Number, Payload
- Routers forward packets; destination reassembles

See Diagram 2: Packet Structure

```
+-------------------------+
| Source IP               |
+-------------------------+
| Destination IP          |
+-------------------------+
| Source Port             |
+-------------------------+
| Destination Port        |
+-------------------------+
| Sequence Number         |
+-------------------------+
| Payload (Actual Data)   |
+-------------------------+
```

5. IP Addressing

- **Global IP** = assigned by ISP
- **Local IP** = assigned by router (DHCP), e.g., 192.168.x.x
- **NAT(Network Address Translation)**:maps multiple local IPs →single constant public IP

See Diagram 3: NAT + DHCP Flow

```
[PC1]        [PC2]       [PC3]
  |            |           |
  +------------+-----------+
               |
           [Home Router]
       DHCP assigns local IPs
           NAT maps to
        Public IP (ISP)
               |
           [Internet]
```

6. Ports

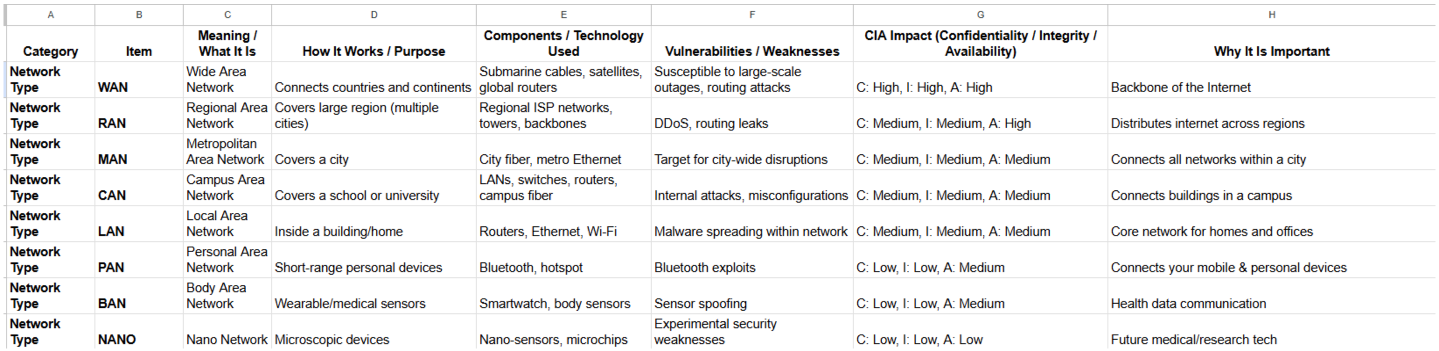
- Ports = “doors” for applications
- It defines the application to where the data would be sent to

- **Ranges:**
  - 0–1023: well-known (HTTP 80, HTTPS 443)
  - 1024–49152: registered
  - 49152–65535: private

7. Internet Speeds

- 1 Mbps = 1 million bits/sec
- 1 Gbps = 1 billion bits/sec
- Download = receive, Upload = send

8.Network Types



A	B	C	D	E	F	G	H
Category	**Item**	**Meaning / What It Is**	**How It Works / Purpose**	**Components / Technology Used**	**Vulnerabilities / Weaknesses**	**CIA Impact (Confidentiality / Integrity / Availability)**	**Why It Is Important**
Network Type	**WAN**	Wide Area Network	Connects countries and continents	Submarine cables, satellites, global routers	Susceptible to large-scale outages, routing attacks	C: High, I: High, A: High	Backbone of the Internet
Network Type	**RAN**	Regional Area Network	Covers large region (multiple cities)	Regional ISP networks, towers, backbones	DDoS, routing leaks	C: Medium, I: Medium, A: High	Distributes internet across regions
Network Type	**MAN**	Metropolitan Area Network	Covers a city	City fiber, metro Ethernet	Target for city-wide disruptions	C: Medium, I: Medium, A: Medium	Connects all networks within a city
Network Type	**CAN**	Campus Area Network	Covers a school or university	LANs, switches, routers, campus fiber	Internal attacks, misconfigurations	C: Medium, I: Medium, A: Medium	Connects buildings in a campus
Network Type	**LAN**	Local Area Network	Inside a building/home	Routers, Ethernet, Wi-Fi	Malware spreading within network	C: Medium, I: Medium, A: Medium	Core network for homes and offices
Network Type	**PAN**	Personal Area Network	Short-range personal devices	Bluetooth, hotspot	Bluetooth exploits	C: Low, I: Low, A: Medium	Connects your mobile & personal devices
Network Type	**BAN**	Body Area Network	Wearable/medical sensors	Smartwatch, body sensors	Sensor spoofing	C: Low, I: Low, A: Medium	Health data communication
Network Type	**NANO**	Nano Network	Microscopic devices	Nano-sensors, microchips	Experimental security weaknesses	C: Low, I: Low, A: Low	Future medical/research tech

Link to table: Tables for networking

Order of Network types (size):

```
+-------------------------------------------------------------+
|                           WAN                               |
|   +-----------------------------------------------------+   |
|   |                        RAN                          |   |
|   |   +---------------------------------------------+   |   |
|   |   |                    MAN                      |   |   |
|   |   |   +-------------------------------------+   |   |   |
|   |   |   |                 CAN                  |   |   |   |
|   |   |   |   +------------------------------+    |   |   |   |
|   |   |   |   |             LAN              |    |   |   |   |
|   |   |   |   |   +-----------------------+  |    |   |   |   |
|   |   |   |   |   |         PAN           |  |    |   |   |   |
|   |   |   |   |   |   +----------------+  |  |    |   |   |   |
|   |   |   |   |   |   |      BAN       |  |  |    |   |   |   |
|   |   |   |   |   |   |   +----------+ |  |  |    |   |   |   |
|   |   |   |   |   |   |   |  NANO    | |  |  |    |   |   |   |
+-------------------------------------------------------------+
```

9. Transmission Media

Wired: Ethernet, Coaxial, Fibre
Wireless: Wi-Fi, Bluetooth, Cellular (3G/4G/5G), Satellite

How the Ethernet works/RJ45:

- Physical connection: Devices are physically connected to each other or to a central device like a switch or router using an Ethernet cable.
- Data transmission: Data is sent as electrical pulses through the wires in the cable.
- Protocol: A set of rules called a protocol (governed by the IEEE 802.3 standard) ensures that devices can format and transmit data in a way that other devices can understand.
- Collision detection: A system known as Carrier Sense Multiple Access with Collision Detection (CSMA/CD) manages the data flow and ensures that if two devices try to send data at the same time, they stop, wait a random amount of time, and try again.

An RJ45 (Registered Jack 45) is the standard, 8-position, 8-conductor modular connector used for Ethernet networking, connecting devices like computers, routers, and switches via Ethernet cables

10. Global Internet Connectivity

- Submarine cables connect continents
- Data flow: Device → Router → ISP → Internet → Server → Back

11. Modem vs Router

Modem: digital converting to/from analog signals

Router: routes packets, assigns local IPs (DHCP), NAT

Further Explanations: DHCP (Dynamic Host Configuration Protocol) is a network protocol that automatically assigns IP addresses and other crucial network settings to devices (hosts) on a network

This graph explains the Router better:

```
    [Internet / Server]
              ^
              |  Response Packets (Public IP)
              |
       +-------------+
       |   Router    |
       |-------------|
       | NAT Table   |  <- Maps Public IP + Port → Local IP + Port
       | DHCP Server |  <- Assigns local IPs to devices
       +-------------+
      /       |       \
     /        |        \
 [PC1]      [PC2]      [PC3]
192.168.1.2 192.168.1.3 192.168.1.4
  |            |            |
  |  Packet    |  Packet    |
  +------------+------------+

         Internal LAN
```

12. Network Topologies

- Various layout of how devices are connected

Diagram 4: Topologies

Bus Topology:

```
[PC1]---[PC2]---[PC3]---[PC4]
       (single backbone)
```

Ring Topology:

```
[PC1]---[PC2]
 |        |
[PC4]---[PC3]
```

Star Topology:

```
     [Switch/Hub]
       /   |   |   \
  [PC1] [PC2] [PC3] [PC4]
```

Tree Topology:

```
        [Main Switch]
       /      |      \
   [Switch1][Switch2][Switch3]
    / \       |       / \
 [PC1][PC2] [PC3] [PC4][PC5]
```

Note(All of the Topology listed above): All of the above layout is easy to establish but once main connection is corrupted, the whole connection is shut down

Mesh Topology:

```
[PC1]----[PC2]
 |  \   /  |
 |   [PC3] |
 |  /   \  |
[PC4]----[PC5]
```

Note(Mesh Topology Only):All are connected preventing the whole connection from shutting down in this layout , but can only be done on a smaller scale due to the high cost and feasibility of everyone to be in this Topology

13. OSI Model — 7 Layers

Diagram 5: OSI Stack

```
+---------------------+  Layer 7: Application
+---------------------+  Layer 6: Presentation
+---------------------+  Layer 5: Session
+---------------------+  Layer 4: Transport (TCP/UDP)
+---------------------+  Layer 3: Network (IP, Routing)
+---------------------+  Layer 2: Data Link (MAC, Switches)
+---------------------+  Layer 1: Physical (Cables, Signals)
```

How it works:

7. Application Layer

What it does:
Provides services directly to users and software applications (e.g., browsers, email clients).

Examples:
HTTP, HTTPS, FTP, SMTP, DNS

Keyword: User interaction

---

6. Presentation Layer

What it does:
Prepares and transforms data so applications can understand it.

Functions:

- **Encryption** (e.g., SSL/TLS)
- **Compression**
- **Translation** (text ↔ binary formats)

Keyword: Data formatting

5. Session Layer

What it does:
Creates, maintains, and ends sessions between two devices.

Functions:

- Session **setup**
- Session **maintenance**
- Session **termination**

Keyword: Connection management

4. Transport Layer

What it does:
Controls end-to-end delivery of data.

Functions:

- **Segmentation** (break data into chunks)
- **Flow control**
- **Error checking**
- Reliability (TCP) or speed (UDP)

Protocols:

- **TCP** – reliable
- **UDP** – fast, no confirmation

Keyword: Delivery quality

3. Network Layer

What it does:
Handles routing between **different networks**.

Functions:

- **Routing**
- **IP addressing** (logical addresses)
- Packet forwarding

Devices/Protocols: IP, **Routers**

Keyword: Path selection

2. Data Link Layer

What it does:
Moves data within the **same local network**.

Functions:

- **Framing**
- **MAC addressing**
- Basic error detection
- Switch-to-switch communication

Devices: Switches

Keyword: Local delivery

---

1. Physical Layer

What it does:
Transmits **raw bits** (0s and 1s) through physical signals.

Examples:
Cables, Wi-Fi radio waves, fiber-optic light pulses, voltages

Keyword: Bits & wires

🔄 How Data Flows (Very Important)

When sending data (your device → network):

It goes **DOWN** the OSI layers:
Application → Presentation → Session → Transport → Network → Data Link → Physical

Physical Layer sends electrical/light/radio signals out.

When receiving data (network → your device):

It goes **UP** the OSI layers in **reverse order**:
Physical → Data Link → Network → Transport → Session → Presentation → Application

This is how raw bits eventually become the web page, video, or text the user sees.

TCP/IP Model:

- The TCP/IP model is a simplified version of the OSI model used in real-world networking. It describes how data travels across networks like the Internet.

1. Application Layer

What it does

- Provides services directly to users and applications.
- Handles protocols that allow software to communicate over the network.

Examples

- **HTTP/HTTPS** – web browsing
- **FTP** – file transfer
- **DNS** – domain name lookup
- **SMTP** – email

---

2. Transport Layer

What it does

- Ensures reliable or fast-but-unreliable delivery between devices.
- Breaks data into segments and reassembles them.

Main Protocols

TCP (Transmission Control Protocol)

- Reliable
- Connection-oriented
- Error checking
- Guarantees delivery

UDP (User Datagram Protocol)

- Fast
- Connectionless
- No delivery guarantee
- Used for games, live streaming, VoIP

---

3. Internet Layer

What it does

- Moves packets across networks.
- Finds the best path for data.
- Handles IP addressing.

Main Protocol

- **IP (Internet Protocol)**
  - IPv4 / IPv6
  - Routing

Other Protocols

- **ICMP** – ping / error reporting
- **ARP** – address resolution

---

4. Network Access Layer

(Also called Link Layer / Network Interface Layer)

What it does

- Deals with physical network hardware.
- Sends and receives frames on a local network.
- Converts data into signals (wired or wireless).

Examples

- Ethernet
- Wi-Fi
- MAC addresses
- Physical cables / radio

14. Cookies, HTTP GET/POST, Pull & Push

- **Cookies: browser data (login, preferences)**
- **GET: request data**
- **POST: send data**
- **Pull: client requests updates**
- **Push: server sends updates**

Diagram 6:

HTTP GET Flow

```
(Client)                 (Server)
   |      GET /page       |
   |--------------------->|
   |      Response        |
   |<---------------------|
```

HTTP Post Flow

```
(Client)                 (Server)
   |      POST /form      |
   |--------------------->|
   |      Response        |
   |<---------------------|
```

15. Subnets & VLANs

- **Subnet: divide network**
- **VLAN(Virtual Local Area Network): logical separation of devices**

How subnetting works

- ☐ **A subnet mask is used to identify which part of an IP address represents the network or subnet and which part represents the host device.**
- ☐ **By using different subnet masks, a single large network can be divided into multiple smaller subnets. This is useful for both managing IP addresses and controlling traffic flow.**
- ☐ **For example, a network administrator could create a subnet for the marketing department and another for the sales department. This would prevent traffic from one department from unnecessarily congesting the network of the other**

How VLAN works

- ☐ **Logical Grouping: Devices (computers, printers, servers) are assigned to VLANs (e.g., VLAN 10 for Sales, VLAN 20 for Engineering) through switch port configuration.**
- ☐ **Frame Tagging: When data travels between switches, a small tag (VLAN ID) is added to the Ethernet frame, identifying its VLAN.**
- ☐ **Isolation: Switches read the tag and forward the frame only to ports belonging to the same VLAN, keeping traffic separate from other VLANs.**

□ **Inter-VLAN Communication: Devices in different VLANs need a router or Layer 3 switch to communicate, adding a layer of control**

16. Routing Basics

- **Routers forward packets using IP addresses**
- **Protocols: RIP, OSPF, BGP**

Further information for Routers and routing:

Router packets refer to how routers (traffic cops of the internet) manage and forward small chunks of data called packets, using info in the packet's header (like IP addresses) and routing tables to send them efficiently across networks to their destination, making internet communication fast and reliable by breaking large data into manageable pieces.

How Router Packets Work:

1. **Data Splitting: A large file (like a webpage or video) is broken into many small packets.**
2. **Packet Structure: Each packet has:**
   - **Header: Contains source/destination IP addresses, sequence numbers, and protocol info (like a mailing label).**
   - **Payload: The actual piece of data.**
   - **Trailer: Sometimes added for error checking.**
3. **Routing:**
   - **When you send data, your device sends packets to your local router.**
   - **The router reads the destination IP in the header and checks its routing table (a map of network paths).**
   - **It forwards the packet to the next best router on the path.**

    - **This process repeats across many routers until the packet reaches its final destination.**

4. **Reassembly: The destination device receives all the packets and reassembles them in the correct order to reconstruct the original data**

Routing Protocols – Quick Reference

1. RIP – Routing Information Protocol

- **Type: Distance-vector protocol**
- **How it works: Routers share their routing tables with neighbors regularly.**
- **Metric: Number of hops (routers to reach destination)**
- **Limitations: Maximum of 15 hops → not suitable for large networks**
- **Use case: Small networks, labs, simple setups**

---

2. OSPF – Open Shortest Path First

- **Type: Link-state protocol**
- **How it works:**
    - **Routers map the network topology (who is connected to whom)**
    - **Calculates shortest path using Dijkstra's algorithm**
- **Advantages:**
    - **Fast convergence (adapts quickly to changes)**
    - **Scales well for large networks**
- **Use case: Enterprise networks, campus networks, ISPs**

---

3. BGP – Border Gateway Protocol

- **Type: Path-vector protocol**
- **How it works:**
    - **Routes data between autonomous systems (AS), e.g., different ISPs or large networks**
    - **Chooses paths based on policies, not just shortest distance**
- **Advantages:**
    - **Scalable for the Internet**
    - **Supports policy-based routing**
- **Use case: Internet backbone, large-scale ISPs**

Protocol	Type	Metric / Basis	Use Case

RIP	**Distance-vector**	**Hop count**	**Small networks, labs**
OSPF	**Link-state**	**Shortest path (cost)**	**Enterprise/campus networks**
BGP	**Path-vector**	**Policies / AS path**	**Internet backbone, ISPs**

Routing Protocols Visual Map (RIP, OSPF, BGP)

```
                +-----------------------+
                |     INTERNET          |
                |   (Global Routing)    |
                |        BGP            |
                +----------+-----------+
                           |
                           |
                 BGP Peering Between ISPs
                           |
      ------------------------------------------------
      |                                              |
+-----v-----+                                 +------v------+
|  ISP / AS |                                 |  ISP / AS   |
| BGP Edge  |                                 |  BGP Edge   |
+-----+-----+                                 +------+------+
      |
      |  Internal Routing (Inside ISP / Org)
      |  Uses OSPF
      |
   +--+------------------+-------------------+
   |                     |                   |
+--v--+             +----v----+        +-----v-----+
|Rtr 1|             | Rtr 2   |        |  Rtr 3    |
|OSPF |<--OSPF LS-->| OSPF    |<--OSPF>|  OSPF     |
+-----+             +---------+        +-----------+
      |
      |  Small LANs / Branch Sites
      |  may use RIP
      |
   +--v--+       +--v--+
   |RIP R| <---> |RIP R|
   +-----+       +-----+
```

17. Switches vs Hubs

Diagram 7: Hub vs Switch

```
Hub (broadcast):

[PC1]--\

[PC2]---[Hub]---[PC3]

[PC4]--/
```

```
Switch (direct delivery):

[PC1]--\

[PC2]---[Switch]---[PC3]

[PC4]--/
```

Summary Table — Hub vs Switch

Feature	Hub	Switch
Sends data to	Everyone	Only the target device
Uses MAC table?	❌ No	✔ Yes
Collisions	✔ Many	❌ None

Speed	Slow	Fast
Security	Very low	Higher
Traffic efficiency	Poor	Efficient

18. Wireless Standards

- **802.11 a/b/g/n/ac/ax**
- **Wi-Fi 6 = 802.11ax**

Diagram 8: Wi-Fi Flow

```
[PC/Phone] --Wi-Fi--> [Access Point / Router] --Internet--> [Server]
```

19. Security Basics

- **TLS/SSL: encrypts web data**
- **VPN: secure data over public networks**
- **Firewall: blocks unauthorized access**
- **Proxy/CDN: caching, performance, security**

Diagram 9: VPN Flow

```
[PC] --Encrypted Tunnel--> [VPN Server] --> Internet
```

20. Packet Flow Across Network

Diagram 10: LAN → Internet:

```
[PC1]        [PC2]
   |            |
   v            v
  [Switch] (local traffic handled internally)
   |
   v
 [Router/NAT]
   | (internet-bound traffic)
   v
   [ISP]
   |
   v
[Internet Backbone]
   |
   v
[Web Server / Service]
```

END OF Notes

Generations

Disruptive innovation:

- **Disruption happens when a technology changes how people live, work, or interact**, often creating **entirely new markets** or ways of doing things.

Mobile Network Generations: 1G to 5G

1G — First Generation (Analog)

Era: 1980s
Technology: Analog (AMPS, TACS)
Frequency: 800–900 MHz

Capabilities

- Voice calls only
- No data transmission
- Basic mobile coverage

Problems It Solved

- Allowed mobile communication for the first time
- Enabled people to make calls on the move

Limitations

- **Poor voice quality** due to analog interference
- **No security** – calls could be easily intercepted
- **No data** – cannot send texts, multimedia, or access the internet
- Limited coverage and capacity

Impact

- Mobile phones became widely available
- First step toward digital communication

2G — Second Generation (Digital)

Era: 1990s
Technology: Digital (GSM, CDMA, IS-95)
Frequency: 900–1800 MHz

Capabilities

- Digital voice calls (better quality than 1G)
- SMS (text messaging)
- Basic data (GPRS, EDGE – very slow internet)

Problems It Solved

- **Security:** Encryption protects calls
- **Texting:** Enabled SMS and later MMS
- **Better coverage** and more simultaneous calls

Limitations

- **Slow internet speeds** (tens of kbps)
- Limited multimedia support
- Cannot support modern smartphones or apps

Impact

- Birth of texting culture
- Digital communication set the foundation for mobile internet

3G — Third Generation (Mobile Internet)

Era: 2000s
Technology: UMTS, HSPA
Frequency: 1.8–2.5 GHz

Capabilities

- High-speed data (hundreds of kbps → several Mbps)
- Video calls
- Mobile internet browsing
- Multimedia messaging (MMS)
- Early smartphone support

Problems It Solved

- Enabled **mobile internet** on the go
- Better voice + data integration
- Early mobile apps possible

Limitations

- **Network congestion** with high traffic
- Speeds insufficient for HD video streaming
- Latency higher (hundreds of milliseconds)

Impact

- Popularized smartphones
- Supported app stores and early mobile social media
- Early foundation for streaming and online games

4G — Fourth Generation (Broadband Mobile)

Era: 2010s
Technology: LTE (Long Term Evolution)
Frequency: 2–8 GHz

Capabilities

- High-speed internet: 100 Mbps → 1 Gbps
- HD video streaming
- Real-time online gaming
- Video conferencing
- Mobile hotspots
- GPS-enabled services
- Cloud applications

Problems It Solved

- Slow 3G internet speeds
- Poor multimedia support
- Latency issues (now reduced to ~50 ms)

Limitations

- Cannot handle **massive IoT devices** efficiently
- Latency still too high for real-time autonomous control or VR/AR

Impact

- Mobile apps explode in popularity
- Streaming, video calls, and cloud-based apps become mainstream
- Smartphones become central to daily life

5G — Fifth Generation (Ultra-Fast, Low Latency, Massive IoT)

Era: 2020s
Technology: 5G NR (New Radio)
Frequency: Sub-6 GHz + mmWave (24–100 GHz)

Capabilities

- **Speeds:** 1–10 Gbps (up to 100x faster than 4G)
- **Latency:** ~1 ms (near real-time)
- **Connection density:** Millions of devices per km²
- **Reliability:** Ultra-stable networks for critical applications
- Supports advanced technologies:
  - Autonomous vehicles
  - Remote surgery
  - Industrial automation
  - Smart cities
  - AR/VR, cloud gaming

Problems It Solved

- Network congestion in urban areas
- Slow speeds for high-data applications
- Latency too high for real-time automation
- Insufficient support for massive IoT devices

Limitations

- mmWave signals have **short range**, require small cell networks
- High deployment cost
- Not yet available everywhere
- Encryption Gaps
- Inconsistent Speeds

Impact

- Enables **Industry 4.0** and smart infrastructure
- Supports immersive technologies (VR/AR)
- Foundation for autonomous transport and telemedicine

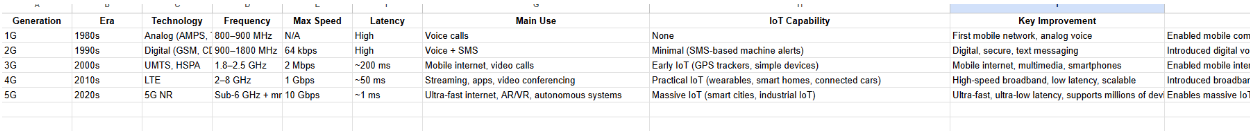
Progression of IOT in the Gs

IoT Progression

- **2G:** Minimal, SMS-based machine alerts
- **3G:** Early IoT, low-speed internet for GPS trackers and simple devices
- **4G:** Practical IoT, supports wearables, smart homes, and connected cars
- **5G:** Massive IoT, enables smart cities, industrial automation, autonomous vehicles, and mission-critical applications

Comparison:

1. Comparison of All Mobile Generations (1G → 5G)



Generation	Era	Technology	Frequency	Max Speed	Latency	Main Use	IoT Capability	Key Improvement	
1G	1980s	Analog (AMPS, T	800–900 MHz	N/A	High	Voice calls	None	First mobile network, analog voice	Enabled mobile com
2G	1990s	Digital (GSM, CD	900–1800 MHz	64 kbps	High	Voice + SMS	Minimal (SMS-based machine alerts)	Digital, secure, text messaging	Introduced digital vo
3G	2000s	UMTS, HSPA	1.8–2.5 GHz	2 Mbps	~200 ms	Mobile internet, video calls	Early IoT (GPS trackers, simple devices)	Mobile internet, multimedia, smartphones	Enabled mobile inter
4G	2010s	LTE	2–8 GHz	1 Gbps	~50 ms	Streaming, apps, video conferencing	Practical IoT (wearables, smart homes, connected cars)	High-speed broadband, low latency, scalable	Introduced broadbar
5G	2020s	5G NR	Sub-6 GHz + mr	10 Gbps	~1 ms	Ultra-fast internet, AR/VR, autonomous systems	Massive IoT (smart cities, industrial IoT)	Ultra-fast, ultra-low latency, supports millions of devi	Enables massive IoT

Link to table: 🟩 Tables for networking

2. Comparison of 4G and 5G

Feature	4G LTE(Long Term Evolution)	5G NR(New Radio)
Max Speed	1 Gbps	10 Gbps
Latency	~50 ms	~1 ms
Frequency	2–8 GHz	Sub-6 GHz + mmWave (24–100 GHz)
Network Type	Broadband, IP-based	Ultra-broadband, low-latency, IP-based
IoT Capability	Practical IoT (wearables, smart homes, connected cars)	Massive IoT (smart cities, industrial automation, mission-critical devices)
Use Cases	HD video streaming, online gaming, video conferencing, social media apps	Autonomous vehicles, smart factories, AR/VR, remote surgery, real-time automation
Disruptive Potential	Disrupted media, entertainment, cloud computing	Disrupting industries like healthcare, manufacturing, urban infrastructure, transportation
Scalability	Supports millions of users	Supports millions of devices per km² (massive IoT)
Network Efficiency	Good, shared IP network	Ultra-efficient with dynamic spectrum and low-latency routing

Cell Towers:

1G Cell Towers

- **Technology: Analog, circuit-switched**
- **Tower Type: Large macro towers**
- **Coverage: Very wide (~30–50 km per tower in rural areas)**
- **Capacity: Low (few simultaneous calls)**
- **Antenna Type: Omnidirectional**
- **Characteristics:**
    - **Only supports voice**
    - **Simple infrastructure**
    - **Poor efficiency for many users**

2G Cell Towers

- **Technology: Digital (GSM/CDMA)**
- **Tower Type: Macro towers (like 1G), upgraded with digital equipment**
- **Coverage: ~2–35 km per tower depending on environment**
- **Capacity: Higher than 1G (more simultaneous users via TDMA/CDMA)**
- **Antenna Type: Omnidirectional, some sectorized antennas**
- **Characteristics:**
    - **Supports SMS and very low-speed data**
    - **Digital encoding allows more efficient use of spectrum**

3G Cell Towers

- **Technology: UMTS/HSPA**
- **Tower Type: Macro towers + small base stations in dense areas**
- **Coverage: ~1–2 km in cities, longer in rural areas**
- **Capacity: Moderate (more simultaneous data and voice users)**
- **Antenna Type: Sectorized, multiple antennas per tower**
- **Characteristics:**
  - **Supports mobile internet**
  - **Higher frequencies → smaller coverage → need more towers in cities**

4G Cell Towers

- **Technology: LTE**
- **Tower Type: Macro towers + small cells + distributed antenna systems (DAS) in dense urban areas**
- **Coverage: 1–5 km per tower in cities, 5–15 km in rural areas**
- **Capacity: High (broadband, high-speed streaming)**
- **Antenna Type: Sectorized with MIMO (Multiple Input Multiple Output) antennas**
- **Characteristics:**
  - **Supports HD video, apps, mobile hotspots**
  - **Uses IP-based data → highly efficient spectrum use**
  - **Requires more towers in urban areas due to higher frequencies**

5G Cell Towers

- **Technology: 5G NR (Sub-6 GHz + mmWave)**
- **Tower Type: Macro towers + massive small cells + beamforming antennas**
- **Coverage:**
  - **Sub-6 GHz: similar to 4G (1–5 km)**
  - **mmWave: very short (~200–500 m), requires dense small cells**
- **Capacity: Extremely high (millions of devices per km²)**
- **Antenna Type:**
  - **Massive MIMO (dozens/hundreds of antennas per tower)**
  - **Beamforming to direct signals precisely**
- **Characteristics:**
  - **Ultra-low latency (~1 ms)**
  - **Supports massive IoT, AR/VR, autonomous vehicles**
  - **Dense network deployment in cities → hundreds of small cells per square km**

Note:

1. **Tower density increases with each generation, especially for 5G mmWave.**
2. **Antenna technology improves: Omnidirectional → sectorized → MIMO → massive MIMO with beamforming.**
3. **Coverage vs capacity trade-off: higher frequencies → shorter range → more towers, but much higher capacity and speed.**

Extra information on Cell Towers:

1. Omnidirectional Antenna

- **What it is: An antenna that radiates signals equally in all directions (360° around the tower).**
- **Used in: 1G, early 2G towers**
- **Advantages:**
  - **Simple, cheap**
  - **Covers a wide area**
- **Limitations:**
  - **Low efficiency for multiple users**
  - **Signal power wasted in directions where no users are present**
- **Analogy: Like shouting in all directions instead of talking directly to someone**

---

2. Sectorized Antenna

- **What it is: The antenna divides coverage into sectors, usually 3–6 per tower.**
- **Used in: 2G/3G/4G towers**
- **Advantages:**
  - **Focused signal in specific directions → better capacity and coverage**
  - **Can serve more users simultaneously**
- **Limitations:**
  - **Requires careful alignment**

- **Analogy: Like dividing a classroom into sections and talking to each group individually**

---

3. MIMO (Multiple Input, Multiple Output)

- **What it is: Uses multiple antennas at both the tower and the device to send and receive multiple data streams simultaneously.**
- **Used in: 4G LTE**
- **Advantages:**
  - **Increases data throughput without needing more spectrum**
  - **Reduces interference**
  - **Supports more simultaneous users**
- **Limitations:**
  - **More complex hardware**
- **Analogy: Like having multiple people carrying multiple packages at the same time instead of one person with one package**

---

4. Massive MIMO with Beamforming

- **What it is:**
  - **Massive MIMO: Uses dozens or hundreds of antennas at the tower**
  - **Beamforming: Focuses the signal directly to each user, rather than broadcasting in all directions**
- **Used in: 5G**

- **Advantages:**
    - **Extremely high capacity → millions of devices in the area**
    - **Ultra-low latency → fast real-time communication**
    - **Energy efficient → only sends signal where needed**
    - **Reduces interference between users**
- **Analogy: Like using a laser pointer to shine light exactly where someone is standing, instead of turning on a room light for everyone**

Circuit-Switched vs Packet-Switched Networks

1. Circuit-Switched (1G)

- **How it worked:**
  - **When you make a call, the network reserves a dedicated physical channel (circuit) between you and the receiver for the entire call.**
  - **The channel is exclusive, even if no one is speaking.**
- **Problems:**
  - **Inefficient: Many channels are idle during pauses in conversation.**
  - **Low capacity: Few simultaneous calls possible.**
  - **Only supports voice, not data.**
- **Example: 1G analog calls (AMPS, TACS)**

---

2. Packet-Switched / Digital (2G and beyond)

- **How it works:**
  - **Voice and data are converted to digital signals, then split into small packets.**
  - **Each packet is sent independently over the network.**
  - **Packets are reassembled at the receiver.**
- **Advantages over circuit-switched:**
  - **Efficiency: Multiple users share the same network paths; bandwidth is not wasted during silence.**
  - **Data support: Internet, text messages, multimedia can be sent alongside voice.**

  - **Scalability: Supports more simultaneous users.**
  - **Security: Digital signals can be encrypted easily.**
  - **Flexibility: Routing can dynamically adapt to network congestion.**
- **Example:**
  - **2G GSM uses Time Division Multiple Access (TDMA) or Code Division Multiple Access (CDMA) to send packets efficiently.**
  - **3G/4G/5G use all-IP networks, where everything (voice, video, apps) is sent as IP packets.**

Impact of the Transition

Aspect	**Circuit-Switched (1G)**	**Packet-Switched / Digital (2G+)**
Signal Type	**Analog**	**Digital**
Efficiency	**Low, dedicated channels**	**High, shared network**
Data Support	**None**	**SMS, Internet, multimedia**
Security	**Minimal**	**Encryption possible**
Scalability	**Limited**	**Supports millions of users**
Flexibility	**Fixed path**	**Dynamic routing**

Summary

- **1G: Circuit-switched → voice only, inefficient**
- **2G: Digital → packet-switched → SMS + basic data**
- **3G+: Fully digital, IP-based → mobile internet, video calls**
- **4G/5G: All-IP packet-switched networks → broadband, ultra-fast data, IoT, VR/AR**

The key change was that the network no longer had to dedicate a single path for a call. Digital packets allowed data to travel more efficiently, securely, and flexibly, which is the foundation for modern smartphones and mobile internet.

MMS:

MMS stands for **Multimedia Messaging Service**. It's basically an upgrade from **SMS (Short Message Service)**. Here's a detailed breakdown:

---

1. What MMS Is

- A service that allows you to **send multimedia content via mobile networks**, not just text.
- Can include:
  - **Images** (photos)
  - **Audio** (voice clips, music)
  - **Video clips**
  - **Rich text** (longer messages, formatted text)

---

2. How It Works

- Uses **cellular data networks** (2G/3G and above) to transfer content.
- Larger message sizes than SMS (up to several MBs depending on carrier).
- Each MMS message is **packet-switched**, unlike SMS which is sent over signaling channels.

---

3. Why MMS Was Important

- Enabled users to **share media on mobile phones**, not just text.
- Allowed early forms of mobile content sharing (before smartphones and apps).
- Helped mobile networks move toward **digital packet-based communication**, paving the way for **mobile internet and apps**.

---

4. Limitations

- Slower than modern messaging apps (like WhatsApp, Telegram)
- Often costs more per message
- Depends on network coverage and size limitations

When it was used:

Generation	Service	Description
2G	**SMS**	**Text-only messages, small size**
2.5G / 3G	**MMS**	**Images, audio, video, longer messages**
4G/5G	**Rich messaging apps**	**WhatsApp, Messenger, streaming → much larger, faster, multimedia-rich**

OSI(Detailed)

OSI 7 Layers Detailed:

7. Application Layer — “Where humans interact with the network”

This layer is where applications like Google Chrome, Outlook, WhatsApp, and games interface with the network.
 It doesn’t move data itself — instead, it uses many **application protocols**.
 Here are the important ones **with full names**:

HyperText Transfer Protocol (HTTP)

This is the protocol used to load websites.
 Your browser asks a web server for things like HTML pages or images, and the server sends them back.
 HTTP itself is **not encrypted**, so people could theoretically see what is being transferred if they intercept the data.

HyperText Transfer Protocol Secure (HTTPS)

This is HTTP but combined with encryption using a method called **Transport Layer Security**.
 This means data is scrambled in a way that only your computer and the website can understand — anyone in the middle sees only gibberish.
 This protects passwords, bank details, and personal info.

File Transfer Protocol (FTP)

This protocol is used to upload and download files between computers.
 It works by creating a control connection (to send commands) and a data connection (to send actual files).
 It is mostly used by developers or servers to move large files.

Simple Mail Transfer Protocol (SMTP)

This is the main method the internet uses to **send emails**.
When you hit “send”, your email client connects to an SMTP server, hands over the email, and the server relays it across the network to another server.
SMTP only sends emails, not receive them.

Domain Name System (DNS)

DNS turns a name like “google.com” into an Internet Protocol address like “142.250.190.78”.
It works like a phonebook:

1. Your computer asks: “What is the address of google.com?”

2. DNS servers reply with the IP.
   Your computer then knows where to send the request.

What the Application Layer actually does:

- Lets software use the network

- Provides web browsing, email, file downloads, messaging

- Converts user requests into network actions

---

6. Presentation Layer — “Formats and protects the data”

This layer prepares data so the application layer can understand it.
You can imagine it as the “translator” and “security guard” of the network.

Encryption (example: Transport Layer Security)

This layer handles turning readable data into unreadable data using cryptographic algorithms.
Only the intended receiver can decrypt it.
This protects your login information, credit card numbers, chats, etc.

Compression

This reduces file size so that transfers happen faster.
For example:

- A photo might be compressed using JPEG
- A file might be compressed using ZIP
  This layer handles these conversions.

Data translation

Computers don’t store all data in the same format.
This layer translates things like:

- Text into binary
- Images into formats like PNG or JPEG
- Video into formats like MP4

This ensures different systems can communicate smoothly.

5. Session Layer — “Manages connections between two devices”

This layer creates, maintains, and ends sessions — ongoing communication between two machines.

Example of a session:
You open Google Docs and stay connected to save your work.
You’re in a session with Google’s server.

How this layer works:

1. **Session setup:**
   Establishes a connection between two devices.
2. **Session maintenance:**
   Keeps the communication alive by sending “keep-alive” messages.
3. **Session termination:**
   Gracefully closes the connection so no data is lost.

Real-life examples:

- Logging into a website
- A video call
- Multiplayer game connection
- Staying logged into Gmail

It makes sure actions happen in the correct order.

4. Transport Layer — “Delivers data reliably or quickly”

This layer is about **end-to-end delivery** between two devices.
It breaks data into chunks, sends them, and reassembles them.

There are two major protocols here:

Transmission Control Protocol (TCP)

TCP is **reliable**.
It guarantees that data arrives exactly as sent.
It does this by:

- numbering each packet
- checking if packets arrived
- resending lost packets
- rearranging packets in the correct order

Used for:

- web pages
- emails
- file downloads
- anything where accuracy matters

User Datagram Protocol (UDP)

UDP is **fast but not guaranteed**.
It does not check for lost packets or errors.
If something is missing, it is simply skipped.

Used for:

- live streaming
- voice calls
- gaming
  Because speed is more important than perfection.

What the Transport Layer does:

- Breaks big data into segments
- Controls speed so devices don’t overload
- Checks for errors (TCP)
- Ensures data arrives (TCP) or arrives fast (UDP)

---

3. Network Layer — “Finds the best path for data to travel”

This layer handles **routing**, meaning it decides how data moves between networks.

Internet Protocol (IP)

IP gives every device a logical address, like:
“192.168.1.10” (IPv4)
or
“2406:3003::18” (IPv6).

IP also decides the route packets take, like GPS for data.

Routers

These are devices that forward data between networks.
They look at IP addresses and decide the next best hop to reach the destination.

How it works:

1. Your packet gets an IP address of the destination.
2. Routers read the IP address.
3. Routers forward the packet to the next router.
4. Eventually, it reaches the destination.

This is how data travels worldwide.

---

2. Data Link Layer — “Moves data inside the local network”

This layer works within a **single** local area network (LAN).

Media Access Control address (MAC address)

Every network card in the world has a unique ID like:
“A4-6B-12-9A-01-FE”.
This is the hardware address used inside your LAN.

Switches

These are devices that read MAC addresses and forward frames to the correct device inside the network.

Frames

This layer doesn’t send packets yet; it sends **frames** (local data packages).
Frames include:

- the sender’s MAC address
- the receiver’s MAC address
- error detection bits

How it works:

If a laptop sends a message to a printer:

1. Laptop uses printer’s MAC address

2. Switch reads the MAC
3. Switch forwards the frame to the correct port
4. Printer receives it

No routers involved — this is purely local.

---

1. Physical Layer — “Transmits raw electrical or radio signals”

This layer transfers bits (zeros and ones) through a physical medium.

How it works:

It uses:

- voltage changes in copper cables
- light pulses in fiber optics
- radio waves in Wi-Fi

This layer does not care what the bits mean.
Its job is only to **send** and **receive** them.

Examples of physical media:

- Ethernet cables
- Fiber-optic cables
- Wi-Fi radio frequencies
- Bluetooth signals

Note:

After the above, the data will go through the model reversed and sent to the Client through Application

OSI Diagram Model

```
┌───────────────────────────────────────────────────────────────┐
│                     7. APPLICATION LAYER                      │
├───────────────────────────────────────────────────────────────┤
│ • Where user-facing software interacts with the network       │
│ • Provides web browsing, emailing, file transfer, DNS lookup  │
│                                                               │
│   Examples:                                                   │
│   - HTTP  (HyperText Transfer Protocol)                       │
│   - HTTPS (HTTP + encryption using TLS)                       │
│   - FTP   (File Transfer Protocol — upload/download files)    │
│   - SMTP  (Simple Mail Transfer Protocol — send emails)       │
│   - DNS   (Domain Name System — name → IP conversion)         │
└───────────────────────────────────────────────────────────────┘

┌───────────────────────────────────────────────────────────────┐
│                     6. PRESENTATION LAYER                     │
├───────────────────────────────────────────────────────────────┤
│ • Formats, encrypts, and compresses data                      │
│ • Ensures data is readable by different systems               │
│                                                               │
│   Functions:                                                  │
│   - Encryption (e.g., TLS protecting HTTPS)                   │
│   - Compression (reduces file sizes)                          │
│   - Translation (text ↔ binary, media format conversions)     │
└───────────────────────────────────────────────────────────────┘

┌───────────────────────────────────────────────────────────────┐
│                       5. SESSION LAYER                        │
├───────────────────────────────────────────────────────────────┤
│ • Controls communication sessions between devices             │
│ • Opens, maintains, and closes connections                    │
│                                                               │
│   Examples:                                                   │
│   - Logging into Google accounts                              │
│   - A continuous video call                                   │
│   - Multiplayer game rooms                                    │
└───────────────────────────────────────────────────────────────┘
```

4. TRANSPORT LAYER

- Responsible for end-to-end delivery
- Splits data into segments and reassembles them

Protocols:

- TCP (Transmission Control Protocol — reliable)
    - * Checks for errors
    - * Resends lost packets
    - * Ensures correct order

- UDP (User Datagram Protocol — fast)
    - * No resending, no guarantee
    - * Used in gaming, livestreams, VoIP

3. NETWORK LAYER

- Chooses the best path across networks (routing)
- Uses logical addressing (IP addresses)

Examples:

- IP (Internet Protocol — addressing & routing)
- Routers (forward packets between networks)

2. DATA LINK LAYER

- Handles communication inside the local network (LAN)
- Uses hardware addresses (MAC addresses)

Examples:

- Frames (local data units)
- Switches (forward frames using MAC addresses)
- MAC (Media Access Control — device hardware IDs)

```
┌───────────────────────────────────────────────────────────────┐
│                     1. PHYSICAL LAYER                         │
├───────────────────────────────────────────────────────────────┤
│ • Moves raw bits (0s and 1s) using physical signals           │
│ • No understanding of meaning — just electrical/light/radio   │
│                                                               │
│   Examples:                                                   │
│   - Ethernet cables (electric voltage)                        │
│   - Fiber optics (light pulses)                               │
│   - Wi-Fi & Bluetooth (radio waves)                           │
│   - Hubs
│
└───────────────────────────────────────────────────────────────┘
```

TCP/IP model

```
                 TCP/IP MODEL (4 Layers)
───────────────────────────────────────────────────────────────
```

```
┌───────────────────────────────────────────────────────────────┐
│             4. APPLICATION LAYER (TCP/IP)                     │
├───────────────────────────────────────────────────────────────┤
│ • Combines OSI layers 5, 6, and 7                             │
│ • Provides all user-level network services                    │
│                                                               │
│   Includes:                                                   │
│   - HTTP / HTTPS                                              │
│   - FTP (File Transfer Protocol)                              │
│   - SMTP (email sending)                                      │
│   - DNS (name → IP lookup)                                    │
│   - SSH (remote secure login)                                 │
│   - DHCP (automatic IP assignment)                            │
│                                                               │
│   OSI Equivalent:  Application + Presentation + Session      │
└──────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────┐
│             3. TRANSPORT LAYER (TCP/IP)                       │
├──────────────────────────────────────────────────────────────┤
│ • Provides end-to-end delivery                                │
│                                                               │
│   Protocols:                                                  │
│   - TCP (Transmission Control Protocol)                       │
│       * Reliable, slow, ensures no data loss                  │
│                                                               │
│   - UDP (User Datagram Protocol)                              │
│       * Fast, used in streaming and gaming                    │
│                                                               │
│   OSI Equivalent: Transport Layer                             │
└──────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────┐
│             2. INTERNET LAYER (TCP/IP)                        │
├──────────────────────────────────────────────────────────────┤
│ • Responsible for choosing paths across networks              │
│ • Uses IP addressing                                          │
│                                                               │
│   Protocols & Devices:                                        │
│   - IP (Internet Protocol — IPv4/IPv6)                        │
│   - ICMP (Internet Control Message Protocol — ping)           │
│   - Routers (forward packets to next network)                 │
│                                                               │
│   OSI Equivalent: Network Layer                               │
└──────────────────────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────────┐
│             1. NETWORK ACCESS LAYER (TCP/IP)                  │
├──────────────────────────────────────────────────────────────┤
│ • Combines OSI Layers 1 and 2                                 │
│ • Moves frames and physical bits inside the local network     │
│                                                               │
│   Contains:                                                   │
│   - MAC addressing                                            │
│   - Ethernet, Wi-Fi                                           │
│   - Switches, network cards                                   │
│   - Electrical signals, radio waves, light pulses             │
│                                                               │
│   OSI Equivalent: Data Link + Physical Layers                 │
```

WIFI

Complete Wi-Fi Guide

1️⃣ What Wi-Fi Is

Wi-Fi = **Wireless Local Area Network (WLAN)** technology allowing devices to communicate over radio waves using **IEEE 802.11 standards** (Institute of Electrical and Electronics Engineers).

2️⃣ Access Point (AP)

Access Point (AP) → device that provides network access.

Functions:

- Broadcasts **Service Set Identifier (SSID)**
- Handles authentication (**Wi-Fi Protected Access 2 / Wi-Fi Protected Access 3**, WPA2/WPA3)
- Assigns Internet Protocol (IP) addresses (or passes to router)
- Connects endpoints to the **Local Area Network (LAN)**
- Manages radio frequency channels (2.4 GHz / 5 GHz)

AP ≠ Only for Wi-Fi:

- Cell tower → AP for mobile network
- Bluetooth hub → AP for Bluetooth (BT) devices
- Zigbee hub → AP for Internet of Things (IoT) devices

3️⃣ Endpoint

Endpoint = device that connects to the network

Examples: Laptop, Phone, Tablet, Smart TV, IoT sensors, Printer

Analogy: AP = provider, Endpoint = client

4️⃣ Service Set Identifier (SSID)

SSID = name of the Wi-Fi network.

- Can be broadcasted or hidden
- Devices scan for SSIDs to connect
- AP can broadcast multiple SSIDs (e.g., Guest Wi-Fi)

5️⃣ Wi-Fi Frequency Bands

Band	Range	Speed	Notes
2.4 GHz	Long	Slower	More interference, only 3 non-overlapping channels (1/6/11)
5 GHz	Short	Fast	Less interference, many channels, good for gaming/streaming
6 GHz	Short	Very fast	Wi-Fi 6E (Extended), newest devices only

---

6️⃣ Wi-Fi Generations (IEEE 802.11a/b/g/n/ac/ax)

Generation	Year	Band	Max Speed	Notes
802.11a	1999	5 GHz	54 Megabits per second (Mbps)	Short range, less interference
802.11b	1999	2.4 GHz	11 Mbps	Longer range, slower, interference prone
802.11g	2003	2.4 GHz	54 Mbps	Backward compatible with 802.11b
802.11n	2009	2.4/5 GHz	600 Mbps	Introduced **Multiple Input Multiple Output (MIMO)**
802.11ac	2014	5 GHz	1.3 Gigabits per second (Gbps)	MU-MIMO (Multi-User MIMO), wider channels
802.11ax	2019	2.4/5/6 GHz	10 Gbps	Wi-Fi 6: **Orthogonal Frequency Division Multiple Access (OFDMA)** + MU-MIMO + Target Wake Time (TWT) + BSS Coloring

✅ **Tip:** Wi-Fi 6 (802.11ax) combines speed, efficiency, and multiple device support.

7️⃣ Orthogonal Frequency Division Multiple Access (OFDMA)

- Splits a Wi-Fi channel into smaller **Resource Units (RUs)**
- Multiple devices transmit **simultaneously**, reducing latency

```
Wi-Fi Channel (20 Megahertz)
+------------------------------------------------+
| RU1 | RU2 | RU3 | RU4 | RU5 | RU6 | RU7 | RU8 |
+------------------------------------------------+
  ↑     ↑     ↑     ↑     ↑     ↑     ↑     ↑
Device1 Device2 Device3 Device4 Device5 Device6 Device7 Device8
```

8️⃣ Multi-User Multiple Input Multiple Output (MU-MIMO)

- Access Point with multiple antennas sends data **to multiple devices at once**

```
Access Point (4 antennas)
   Antenna1 —> Device A
   Antenna2 —> Device B
   Antenna3 —> Device C
   Antenna4 —> Device D
```

9️⃣ Dynamic Host Configuration Protocol (DHCP)

- Automatically assigns **Internet Protocol (IP) addresses** to devices

```
Device                 DHCP Server
  | ---- DHCP Discover ---> |
  | <--- DHCP Offer ------- |
  | ---- DHCP Request ---> |
  | <--- DHCP Acknowledgment (ACK) --------- |
```

🔟 Network Address Translation (NAT)

- Lets multiple devices share a single **public IP address**

```
Private Network:
Device A (192.168.1.2)
Device B (192.168.1.3)
Device C (192.168.1.4)
          |
          v
        Router (NAT)
          | Public IP: 203.0.113.10
          v
       Internet
```

1️⃣1️⃣ Address Resolution Protocol (ARP)

- Maps **IP addresses → Media Access Control (MAC) addresses** in the local network

```
Device A (192.168.1.2) ---> Broadcast: Who has 192.168.1.5? --->
Network
                                                               |
Device B (192.168.1.5) <--- ARP Reply: MAC=AA:BB:CC:DD:EE:FF <---
```

1️⃣2️⃣ Wi-Fi Security Standards

- **Wired Equivalent Privacy (WEP):** Weak, old
- **Wi-Fi Protected Access (WPA):** TKIP encryption, temporary fix
- **Wi-Fi Protected Access 2 (WPA2):** AES-CCMP encryption, secure
- **Wi-Fi Protected Access 3 (WPA3):** SAE handshake, forward secrecy, strongest

Authentication Flow:

```
Device ---> Access Point
   |         |
   | "I want to join SSID"
   |------------------------>
   |         |
   | Access Point sends security requirements
   |<-------------------------
   |         |
   | Device proves key (WPA2/WPA3)
   |------------------------->
   |         |
   | Access Point accepts and gives IP address
   |<-------------------------
```

1️⃣3️⃣ 4-Way Handshake (WPA2)

- Ensures AP and device **share correct password** and **create encryption keys**

```
CLIENT                                 ACCESS POINT
   |                                           |
   |---------(1) ANonce ---------------------->|  AP sends
**Authenticator Nonce**
   |                                           |
   |-(2) SNonce + MIC ------------------------>|  Device sends
**Supplicant Nonce + Message Integrity Code**
   |                                           |
   |<--------(3) Install PTK + GTK ------------|  AP sends **Pairwise
Transient Key + Group Temporal Key**
   |                                           |
   |--------(4) Confirmation ------------------>|  Device confirms
   |                                           |
Connection secured.
```

Analogy: Step 1+2 = make secret handshake, Step 3+4 = handshake confirmed → encrypted communication.

1️⃣4️⃣ Combined Wi-Fi Visual (Mega ASCII)

```
                       [ INTERNET ]
                             |
                    +----------------+
                    |  MAIN ROUTER   |
                    | NAT / DHCP     |
                    +----------------+
                             |
                  +----------------------+
                  | Wi-Fi Access Point   |
                  | SSID: Home_WiFi      |
                  | OFDMA + MU-MIMO      |
                  +----------------------+
                   /                   \
          2.4 GHz (long)             5 GHz (fast)
                 |                       |
      ------------------------------------------
      |          |           |         |       |
   [Laptop]   [Phone]    [Smart TV]  [IoT Devices]
    Endpoint   Endpoint   Endpoint    Endpoint
         \         |         /          /
             ------ DHCP / ARP ------
              /                     \
Security: WPA2 / WPA3           OFDMA Subcarriers
```

DATA FLOW

NOTE: SHORT DEFINITIONS IS ATTACHED AT PAGE 1. ALL UNSURE ACRONYMS IN THIS TAB CAN BE FOUND IN THIS TABLE

Short Definitions

Component	Layer	What it Does
DNS	Application	Converts domain names → IP addresses so computers can locate each other
TLD	DNS hierarchy	Categorizes domains (e.g., .com, .org, .sg)
SLD	DNS hierarchy	Main registered domain (e.g., google in google.com)
Subdomain	DNS hierarchy	Organizes services (e.g., mail.google.com, api.discord.com)
A/AAAA Record	DNS	Maps domain names → IPv4/IPv6 addresses
MAC Address	Data Link	Hardware identifier of network card, used inside local networks (LAN)
ARP	Data Link	Resolves IP → MAC to send frames locally
DHCP	Application	Automatically assigns IP configuration: IP, subnet, gateway, DNS, lease time
IP	Network	Logical addressing & routing; identifies devices globally
Router	Network	Forwards packets between networks based on IP addresses
Switch	Data Link	Forwards frames inside LAN using MAC addresses
Gateway	Network	Exit point of LAN to Internet; connects local network to outside networks
Packet	Network	Unit of data at the Network Layer; contains source & destination IP addresses + payload. Sent across routers & the Internet. Encapsulated inside frames at Data Link Layer.
Frame	Data Link	Unit of data at the Data Link Layer; contains source & destination MAC addresses + payload (which is a packet). Used to travel inside local networks (LAN).
Segment	Transport	Unit of data at Transport Layer; contains TCP/UDP headers and payload (part of the packet)

1 — DOMAIN NAME SYSTEM (DNS)

DNS = **The Internet’s phonebook.**
It translates **domain names** → **IP addresses** so computers know where to connect.

1.1 DNS STRUCTURE (Root → TLD → SLD → Subdomain)

Example Domain:

```
mail.google.com.
```

Split parts:

Part	Meaning
`. (root)`	Top of DNS tree, invisible in browsers
`com`	Top-Level Domain (TLD)
`google`	Second-Level Domain (SLD)
`mail`	Subdomain

Hierarchy diagram:

```
Root (.)
   └── com
        └── google
              └── mail
```

1.2 DEFINITIONS

Root

- The absolute top of DNS.
- Managed by ICANN & IANA.
- Has 13 root server *clusters*, not servers.

TLD (Top-Level Domain)

Highest visible category.

Types:

- Generic: `.com`, `.net`, `.org`
- Country: `.sg`, `.jp`, `.uk`
- Restricted: `.gov`, `.edu`

SLD (Second-Level Domain)

- The main organization name.
- Example: `google` in `google.com`.

Subdomain

- Used to organize services.
- Examples:
    - `mail.google.com`
    - `api.discord.com`
    - `en.wikipedia.org`

Unlimited levels allowed.

1.3 DNS RECORD TYPES (FULL TABLE)

Record	Meaning	Example
A	Domain → IPv4	`google.com → 142.x.x.x`
AAAA	Domain → IPv6	Internet future-proofing
CNAME	Alias → real domain	`www → google.com`
MX	Mail exchange server	Email routing
NS	Nameserver for domain	Stores records
TXT	Text data	SPF, DKIM, verification
PTR	Reverse DNS (IP → name)	Spam prevention
SOA	Domain metadata	Refresh, zone info

1.4 FULL DNS RESOLUTION PROCESS

You type:

```
mail.google.com
```

Your computer asks:

1. Browser cache
2. OS DNS cache
3. Router DNS cache
4. ISP DNS resolver

If not found → recursive DNS begins:

```
PC → Resolver → Root:
"Where is .com?"

Root → Resolver:
"Ask the .com TLD servers."

Resolver → TLD (.com):
"Where are google.com servers?"

TLD → Resolver:
"Here are google.com's name servers."

Resolver → Google authoritative server:
"What is the IP of mail.google.com?"

Google → Resolver:
"A = 142.xxx.xxx.xxx"

Resolver → PC
```

Diagram:

```
PC → Resolver → Root → TLD → Authoritative → Resolver → PC
```

1.5 WHY DNS IS RIGHT-TO-LEFT

Because DNS is hierarchical.

Computer resolves:

```
Start: . (root)
Next: com
Next: google
Last: mail
```

Same logic as navigating folders:

```
C:\Folder\Subfolder\File.txt
```

2 — IP AND ROUTING

2.1 IP ADDRESS (Network Layer)

IP = Logical address used to route packets across the Internet.

Types:

- **IPv4**: 32-bit (limited)
- **IPv6**: 128-bit (huge space)

Used to identify devices **globally**, unlike MAC which is local.

3 — MAC(Media Access Control) ADDRESS (Data Link Layer)

MAC = **Physical address** burned into network card.

Example:

```
AA:BB:CC:DD:EE:FF
```

Used ONLY inside local networks (LAN).
Routers use MAC addresses to move frames locally.

---

4 — ARP (Address Resolution Protocol)

ARP = IP → MAC

If you know a device’s **IP**, but not its **MAC**, ARP resolves it.

Process:

```
PC: Who has 192.168.1.1? (broadcast)
Router: That’s me. MAC = AA:BB:CC:DD:EE:FF
```

ARP Table example:

```
192.168.1.1 -> AA:BB:CC:DD:EE:FF
192.168.1.25 -> EE:44:22:11:33:55
```

Used heavily inside LANs.

---

5 — DHCP (Dynamic Host Configuration Protocol)

DHCP automatically gives devices:

- IP address
- Subnet mask
- Default gateway
- DNS server
- Lease time

DHCP DORA Process:

```
Discover → Offer → Request → Acknowledge

PC → DHCP: Discover
DHCP → PC: Offer
PC → DHCP: Request
DHCP → PC: ACK
```

6 — Packets and Data flow

Explanation of Packets, Frames, Segments, and Data Flow

Packet (Network Layer):

- A **packet** is the main unit of data at the **Network Layer (Layer 3)**.
- Data sent over the Internet (like your HTTP request) is split into packets.
- **Structure of a packet:**
  - **Header:** Contains source IP, destination IP, and other routing information.
  - **Payload:** The actual data being sent (e.g., a chunk of a webpage).
- **Purpose:** Packets allow data to be routed across multiple networks and reach the correct destination.

Frame (Data Link Layer):

- A **frame** is the unit of data at the **Data Link Layer (Layer 2)**.
- Each **packet** is encapsulated inside a frame for delivery within a local network (LAN).
- **Structure of a frame:**
  - **Source MAC address (SRC)** – identifies the sending device in the LAN.
  - **Destination MAC address (DST)** – identifies the receiving device in the LAN.
  - **Payload:** The packet from the Network Layer.
- **Purpose:** Ensures data can move between devices on the same local network.

Segment (Transport Layer):

- A **segment** is the unit of data at the **Transport Layer (Layer 4)**.
- Data is split into segments before being placed into packets.

- **TCP segments** include sequence numbers and acknowledgments to ensure reliable delivery.

- **UDP segments** are faster but do not guarantee delivery.

- **Purpose:** Segments provide end-to-end delivery control between your PC and the server.

Data Flow from PC → Server:

1. Browser request → divided into **segments** (Transport Layer).

2. Segments → placed into **packets** (Network Layer).

3. Packets → placed into **frames** (Data Link Layer).

4. Frames travel: **PC → Switch → Router → ISP → Internet Backbone → Google server**.

5. Server responds in reverse: **frames → packets → segments → reassembled by your PC**.

Connection to Your Diagram:

- **Data Link Layer (frames):** Uses MAC addresses to move data inside the LAN.

- **Network Layer (packets):** Uses IP addresses to route data across the Internet.

- **Transport Layer (segments):** Ensures that the data is delivered reliably (TCP) or quickly (UDP).

7 — OSI MODEL (All Concepts Mapped)

```
+---------------------+
| 7. Application      | DNS, DHCP
+---------------------+
| 6. Presentation     | Encryption, SSL/TLS
+---------------------+
| 5. Session          | Sessions
+---------------------+
| 4. Transport        | TCP/UDP
+---------------------+
| 3. Network          | IP, Routing
+---------------------+
| 2. Data Link        | MAC, ARP, Ethernet, Wi-Fi
+---------------------+
| 1. Physical         | Cables, Radio Waves
+---------------------+
```

---

8 — Full End-to-End Internet Request (All Systems Working Together)

Scenario: You type `mail.google.com` in your browser.

8.1 Step 1 — DNS Resolution

- Browser sends a request to resolve the domain name into an IP address.
- **DNS (Domain Name System)** translates `mail.google.com` → IP address.
- This allows your PC to know the destination for the request.

8.2 Step 2 — DHCP (if needed)

- If your PC does not already have an IP address, **DHCP (Dynamic Host Configuration Protocol)** provides:
  - IP address
  - Subnet mask
  - Default gateway
  - DNS server

8.3 Step 3 — ARP (Address Resolution Protocol)

- To send data inside the local network (LAN), your PC must know the router's **MAC address**.
- ARP maps the router's IP → MAC for local delivery.

8.4 Step 4 — Data Link Layer (Frames)

- Data is packaged into **frames** at Layer 2 for LAN delivery.
- **Frame structure:**
  - **SRC MAC:** Your PC
  - **DST MAC:** Router
  - **Payload:** Packet (Network Layer)
- Frames ensure data moves inside your LAN from PC → router.

8.5 Step 5 — Network Layer (Packets)

- Each frame carries a **packet**, the main unit at Layer 3.
- **Packet structure:**
  - **Header:** SRC IP (PC), DST IP (Google server), routing info
  - **Payload:** Segment (Transport Layer)
- Packets allow data to be routed across networks:
  - PC → Router → ISP → Internet Backbone → Google server

8.6 Step 6 — Transport Layer (Segments)

- Data is divided into **segments** at Layer 4 before being placed in packets.
- **TCP segments:** Include sequence numbers and acknowledgments for reliable delivery.
- **UDP segments:** Faster but no delivery guarantees.
- Segments ensure end-to-end communication between your PC and the server.

8.7 Step 7 — Server Response

- Google server receives the request, processes it, and sends a response.
- Response travels back in reverse:
  - Segments → Packets → Frames → Delivered to your PC
- Browser reassembles the segments into the complete webpage.

8.8 Step 8 — Data Flow Summary

- **Segment → Packet → Frame → Transmission → Frame → Packet → Segment**
- **Data Link Layer:** Frames with MAC addresses move data inside LAN.
- **Network Layer:** Packets with IP addresses travel across the Internet.
- **Transport Layer:** Segments ensure correct order, reliability (TCP), or fast delivery (UDP).

9—EVERYTHING CONNECTED

```
                                            +-------------------------+
                                            |        You: Browser     |
                                            |   mail.google.com      |
                                            +-----------+-------------+
                                                               |
                                                               v
                        +--------------------------------------------------+
                        |                  DNS PROCESS                     |
                        | Root → TLD → SLD → Subdomain → IP Address        |
                        +--------------------------------------------------+
                                                               |
                                                               v
            +------------------------------------------------------------------+
            | DHCP: gives IP address, subnet mask, gateway, DNS, lease time    |
            +------------------------------------------------------------------+
                                                               |
                                                               v
                           +-------------------------+--------------------+
                           | ARP: Find MAC of router (IP → MAC mapping)  |
                           +-------------------------+--------------------+
                                                                |
                                                                v
+-----------------------------------------------------------------------------------+
| DATA LINK LAYER (Layer 2): Frames travel using MAC addresses in LAN               |
|      SRC MAC = your PC                                                            |
|      DST MAC = your router                                                        |
|      Frame contains a **Packet** (Network Layer)                                   |
+-----------------------------------------------------------------------------------+
                                                                |
                                                                v
+-----------------------------------------------------------------------------------+
| NETWORK LAYER (Layer 3): Packets routed across Internet using IP addresses        |
|       SRC IP = your PC                                                            |
|       DST IP = Google Server                                                      |
|       Packet contains a **Segment** (Transport Layer)                             |
|       Path: PC → Router → ISP → Backbone → Google Data Center                     |
+-----------------------------------------------------------------------------------+
                                                                |
                                                                v
+-----------------------------------------------------------------------------------+
| TRANSPORT LAYER (Layer 4): Segments (TCP/UDP)                                     |
|      TCP adds sequence numbers, ACKs for reliability                               |
|      Segment is the payload inside a Packet                                         |
+-----------------------------------------------------------------------------------+
                                                                |
                                                                v
                             +-----------------------------------------------+
                             | Google Server Responds with webpage data      |
                             | Response travels back in:                     |
                              | Segment → Packet → Frame → Reassembled at PC|
                             +-----------------------------------------------+
[KEY / ACRONYMS & TERMS]
SRC = Source (origin of data)
DST = Destination (target of data)
MAC = Media Access Control (hardware address, Layer 2)
IP = Internet Protocol (logical address, Layer 3)
ARP = Address Resolution Protocol (IP → MAC mapping)
DHCP = Dynamic Host Configuration Protocol (automatic IP config)
DNS = Domain Name System (names → IP)
TCP = Transmission Control Protocol (reliable Transport Layer protocol)
UDP = User Datagram Protocol (fast, unreliable Transport Layer protocol)
Packet = Unit of data at Network Layer (contains Segment)
Segment = Unit of data at Transport Layer (payload for packet)
Frame = Unit of data at Data Link Layer (contains packet)
```

Explanation:

Step 1 — DNS Resolution:

- You type `mail.google.com` in your browser.
- **DNS (Domain Name System)** translates the domain name into an **IP address** so your computer knows where to send the request.

Step 2 — DHCP Assignment:

- If your computer does not already have an IP address, **DHCP (Dynamic Host Configuration Protocol)** assigns:
  - IP address
  - Subnet mask
  - Gateway
  - DNS server

Step 3 — ARP Lookup:

- To send data in the local network, your computer uses **ARP (Address Resolution Protocol)** to find the **MAC (Media Access Control) address** of the router.

Step 4 — Data Link Layer (Layer 2):

- Data is packaged into **frames**.
- Each frame contains:
  - **SRC (source) MAC address** – your PC
  - **DST (destination) MAC address** – router
  - Payload: a **packet** (Network Layer data)

Step 5 — Network Layer (Layer 3):

- The **packet** contains:
  - **SRC IP** – your PC
  - **DST IP** – Google server
  - Payload: a **segment** (Transport Layer data)
- Packets are routed across the Internet via routers, ISP, and backbone networks.

Step 6 — Transport Layer (Layer 4):

- Data is divided into **segments**.
- **TCP (Transmission Control Protocol)** adds sequence numbers and acknowledgments for reliability.
- **UDP (User Datagram Protocol)** sends data faster but without guaranteed delivery.

Step 7 — Server Response:

- The Google server receives the request and responds.
- Response travels back: **segments → packets → frames**.
- Your PC reassembles the segments into the webpage, which the browser displays.

Step 8 — How Layers are Linked:

- **Segments** → placed in **packets** → placed in **frames** → sent as electrical/wireless signals.
- **SRC (source) and DST (destination)** addresses at each layer ensure data reaches the correct device.

NETWORK DEVICES

Network Devices

1–15: Core Network Devices

1. Router

- **Function:** Connects different networks and forwards data packets. Chooses the best path for data using IP addresses.

- **OSI Layer:** 3 (Network Layer)

- **Example:** Home or enterprise router connecting LAN to the internet.

- **Extra Info:** Can include NAT, DHCP, firewall functions; acts as a gateway.

2. Switch

- **Function:** Connects devices within a LAN and forwards data only to the intended device using MAC addresses.

- **OSI Layer:** 2 (Data Link Layer), some Layer 3 switches also handle routing.

- **Example:** Office network switch connecting PCs and printers.

- **Extra Info:** Supports VLANs and reduces network collisions.

3. Hub

- **Function:** Connects devices in a LAN but broadcasts all data to every port.

- **OSI Layer:** 1 (Physical Layer)

- **Example:** Old Ethernet hubs.

- **Extra Info:** Simple but inefficient; can cause collisions.

4. Access Point (AP)

- **Function:** Allows wireless devices to connect to a wired network.
- **OSI Layer:** 2 (Data Link Layer)
- **Example:** Wi-Fi access point in schools or homes.
- **Extra Info:** Can be managed by a wireless controller; supports Wi-Fi standards like 802.11ac/ax.

5. Modem

- **Function:** Converts ISP signals (DSL, cable, fiber) to digital signals for the network.
- **OSI Layer:** 1 (Physical Layer)
- **Example:** DSL or cable modem.
- **Extra Info:** Some combine modem + router; essential for internet access.

6. Firewall

- **Function:** Monitors and filters traffic to prevent unauthorized access.
- **OSI Layer:** 3–7 (Network to Application Layer)
- **Example:** Hardware firewall in enterprise networks.
- **Extra Info:** Can block IPs, ports, or applications.

7. Network Interface Card (NIC)

- **Function:** Hardware that enables a device to connect to a network.
- **OSI Layer:** 2 (Data Link Layer)
- **Example:** Ethernet NIC or Wi-Fi adapter in a laptop.
- **Extra Info:** Each NIC has a unique MAC address; modern NICs support high-speed data transfer.

8. Repeater

- **Function:** Amplifies or regenerates signals to extend network range.
- **OSI Layer:** 1 (Physical Layer)
- **Example:** Ethernet repeater extending cable runs.
- **Extra Info:** Only boosts signals; does not filter or route traffic.

9. Gateway

- **Function:** Connects networks using different protocols and translates data as needed.
- **OSI Layer:** 3+ (Network Layer and above)
- **Example:** Enterprise network connecting to cloud services.
- **Extra Info:** Routers act as gateways; can handle protocol conversion.

10. Bridge

- **Function:** Connects two LAN segments and filters traffic.
- **OSI Layer:** 2 (Data Link Layer)
- **Example:** Connecting two office floors while reducing collisions.
- **Extra Info:** Used less today; replaced by switches.

11. Load Balancer

- **Function:** Distributes traffic across multiple servers to prevent overload.
- **OSI Layer:** 4–7 (Transport to Application Layer)
- **Example:** Web servers handling many users.
- **Extra Info:** Can be hardware or software; ensures high availability.

12. Proxy Server

- **Function:** Intermediary between clients and servers; can cache data, filter content, or provide anonymity.
- **OSI Layer:** 7 (Application Layer)
- **Example:** School network caching web pages.
- **Extra Info:** Improves performance and security.

13. VPN Concentrator

- **Function:** Handles multiple secure VPN connections for remote users.
- **OSI Layer:** 3–7 (Network to Application Layer)
- **Example:** Corporate VPN appliance.
- **Extra Info:** Encrypts traffic and authenticates users.

14. Media Converter

- **Function:** Converts signals between media types (fiber ↔ copper).
- **OSI Layer:** 1 (Physical Layer)
- **Example:** Fiber-to-Ethernet converter in a data center.
- **Extra Info:** Useful for integrating older networks with modern fiber.

15. Wireless Controller

- **Function:** Manages multiple access points centrally.
- **OSI Layer:** 2–3 (Data Link / Network Layer)
- **Example:** University Wi-Fi network controller.
- **Extra Info:** Controls SSIDs, security, and firmware updates.

16–22: Specialized Network Devices

16. Patch Panel

- **Function:** Organizes and manages network cables in racks.
- **OSI Layer:** 1 (Physical Layer)
- **Example:** Data center patch panel connecting switches to servers.
- **Extra Info:** Helps maintain order and simplifies troubleshooting.

17. VoIP Gateway

- **Function:** Converts voice signals to digital packets for IP networks.
- **OSI Layer:** 3–7 (Network to Application Layer)
- **Example:** Business phone system connecting to the internet.
- **Extra Info:** Enables IP-based phone systems.

18. Optical Switch

- **Function:** Routes fiber-optic signals between multiple paths.
- **OSI Layer:** 1 (Physical Layer)
- **Example:** Data centers routing optical traffic.
- **Extra Info:** Operates very fast; often used in backbone networks.

19. Content Switch

- **Function:** Routes traffic based on content type or request parameters.
- **OSI Layer:** 4–7 (Transport / Application Layer)
- **Example:** Data centers directing HTTP requests to appropriate servers.
- **Extra Info:** Improves performance for web services.

20. IDS/IPS (Intrusion Detection/Prevention System)

- **Function:** Monitors network traffic for malicious activity and blocks threats.
- **OSI Layer:** 3–7 (Network to Application Layer)
- **Example:** Enterprise network security appliance.
- **Extra Info:** IDS alerts, IPS blocks attacks in real-time.

21. Network Tap

- **Function:** Copies network traffic for monitoring or analysis.
- **OSI Layer:** 1–2 (Physical / Data Link Layer)
- **Example:** Security monitoring in data centers.
- **Extra Info:** Transparent to the network; doesn’t interfere with traffic.

22. Customer Premises Equipment (CPE)

- **Function:** Any device installed at the customer’s side, like modems, routers, or set-top boxes.
- **OSI Layer:** Depends on device type.
- **Example:** Home gateway devices from ISPs.
- **Extra Info:** Provides the interface between ISP and customer network.

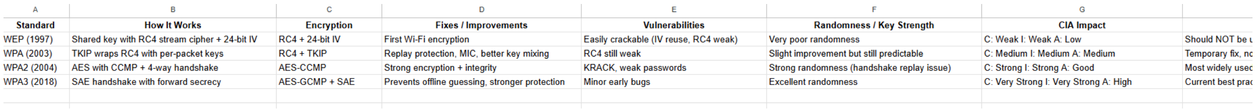
Diagram Example:

```
                      ┌─────────────────┐
                      │    Internet     │
                      └─────────────────┘
                               │
                               │
                           ┌─────────┐
                           │ Modem   │ Layer 1
                           └─────────┘
                               │
                           ┌─────────┐
                           │ Router  │ Layer 3
                           └─────────┘
                               │
                    ┌──────────┴──────────┐
                    │                     │
              ┌───────────┐        ┌────────────┐
              │ Firewall  │ Layer 3-7 │ VPN      │ Layer 3-7
              └───────────┘        │ Concentrator│
                                   └────────────┘
                    │
               ┌─────────┐
               │ Switch  │ Layer 2/3
               └─────────┘
      ┌────────────┴─────────────┐
      │            │             │
 ┌─────────┐   ┌─────────┐   ┌─────────┐
 │ Hub     │   │ Access  │   │ Media   │
 │ Layer 1 │   │ Point   │   │ Converter│
 └─────────┘   │ Layer 2 │   │ Layer 1 │
               └─────────┘   └─────────┘
      │                     │
 ┌─────────┐          ┌─────────┐
 │ NICs    │ Layer 2  │ Wireless│
 │ Layer 2 │          │ Controller │ Layer 2-3
 └─────────┘          └─────────┘
      │                     │
 ┌─────────┐          ┌─────────┐
 │ Repeater│ Layer 1  │ Patch   │ Layer 1
 │ Layer 1 │          │ Panel   │
 └─────────┘          └─────────┘
      │
┌─────────────┐
│ Bridge      │ Layer 2
└─────────────┘
      │
 ┌────────────┐
 │ Load       │ Layer 4-7
 │ Balancer   │
 └────────────┘
      │
 ┌────────────┐
 │ Proxy      │ Layer 7
 │ Server     │
 └────────────┘
      │
 ┌────────────┐
 │ VoIP       │ Layer 3-7
 │ Gateway    │
 └────────────┘
      │
 ┌────────────┐
 │ IDS/IPS    │ Layer 3-7
 └────────────┘
      │
 ┌────────────┐
 │ Content    │ Layer 4-7
 │ Switch     │
 └────────────┘
      │
 ┌────────────┐
 │ Network Tap│ Layer 1-2
 └────────────┘
      │
 ┌────────────┐
 │ CPE        │ Depends on Device
 └────────────┘
```

Security

1. WIFI Security Standards:

- **WEP (Least secure) →WPA→WPA2→WPA3 (Most secure)**



A	B	C	D	E	F	G	
Standard	**How It Works**	**Encryption**	**Fixes / Improvements**	**Vulnerabilities**	**Randomness / Key Strength**	**CIA Impact**	
WEP (1997)	Shared key with RC4 stream cipher + 24-bit IV	RC4 + 24-bit IV	First Wi-Fi encryption	Easily crackable (IV reuse, RC4 weak)	Very poor randomness	C: Weak I: Weak A: Low	Should NOT be u
WPA (2003)	TKIP wraps RC4 with per-packet keys	RC4 + TKIP	Replay protection, MIC, better key mixing	RC4 still weak	Slight improvement but still predictable	C: Medium I: Medium A: Medium	Temporary fix, nc
WPA2 (2004)	AES with CCMP + 4-way handshake	AES-CCMP	Strong encryption + integrity	KRACK, weak passwords	Strong randomness (handshake replay issue)	C: Strong I: Strong A: Good	Most widely usec
WPA3 (2018)	SAE handshake with forward secrecy	AES-GCMP + SAE	Prevents offline guessing, stronger protection	Minor early bugs	Excellent randomness	C: Very Strong I: Very Strong A: High	Current best prac

Link to table: Tables for networking

2. Redundancy:

In networking, redundancy means having extra components, paths, or systems in place so that if something fails, the network can continue operating without disruption. Think of it like having a backup plan for your network.

Why Redundancy is Important

1. **Reliability: Keeps the network running even if a device, link, or path fails.**
2. **Availability: Ensures users can access network services continuously.**
3. **Fault Tolerance: Reduces the risk of a single point of failure causing total network downtime.**

Types of Redundancy in Networking

1. **Link Redundancy (Path Redundancy)**
    - **Having multiple connections between network devices.**
    - **Example: Two routers connected to the internet via separate ISPs.**

- **Benefit: If one connection goes down, traffic automatically uses the other.**

2. **Device Redundancy (Hardware Redundancy)**
   - **Having duplicate hardware devices like routers, switches, or firewalls.**
   - **Example: Two core switches in a data center, where one is active and the other is standby.**
   - **Benefit: Prevents downtime if a device fails.**
3. **Power Redundancy**
   - **Backup power supplies for networking equipment (UPS or generators).**
   - **Benefit: Network stays on during power outages.**
4. **Data Redundancy**
   - **Storing copies of critical data on multiple servers or storage devices.**
   - **Example: RAID (Redundant Array of Independent Disks) or cloud backups.**
   - **Benefit: Prevents data loss if a storage device fails.**
5. **Path Redundancy Protocols**
   - **Spanning Tree Protocol (STP): Prevents loops but allows backup paths in LANs.**
   - **HSRP (Hot Standby Router Protocol) / VRRP (Virtual Router Redundancy Protocol): Ensures routers failover automatically.**
   - **BGP (Border Gateway Protocol): Provides multiple routes between networks for reliability.**

How Redundancy Works in Practice

- **Scenario: You have a home network with two internet connections. If your main ISP fails, the secondary connection takes over automatically. That’s redundancy at** work.

- In enterprise networks, redundancy is built into core switches, routers, and links, so a single failure doesn't disrupt the network.

Trade-offs

- Redundancy improves reliability but comes with:
    - Cost: More devices, cables, and infrastructure needed.
    - Complexity: More management, configuration, and monitoring required.
    - Potential Loops: Extra paths can cause loops unless protocols like STP are used.

ASCII Visual Example

```
Internet
    |
[Router1]----+
    |        |
[Switch1]   [Switch2]  <-- Redundant path
    |        |
  PCs/Servers
```

- If Switch1 fails, traffic can still go through Switch2.
- The network stays operational thanks to redundancy.

In short, network redundancy is all about backups—backup devices, backup paths, and backup power—to make the network more reliable and fault-tolerant.

CRC

- CRC networking refers to the use of Cyclic Redundancy Check (CRC), an error-detection algorithm, to ensure data integrity in computer networks

Function:

1. Checksum-Based: CRC generates a fixed-size check value, often called a checksum, which is appended to the data block before transmission.
2. Mathematical Basis: The algorithm treats the data as a binary polynomial and uses polynomial division (modulo-2 arithmetic) with a predefined generator polynomial to calculate the remainder, which is the CRC value.

Tab 8

✅ FULL SAFE LEARNING PROCEDURE (Professional Style)

Phase 1 — Lab Isolation (Most Important)

Step 1: VMware Network Setup

1. Open **VMware** → **Edit** → **Virtual Network Editor**
2. Select **VMnet1**
3. Configure:
   - ○ Network type: **Host-only**
   - ○ Subnet: e.g. `192.168.56.0 /24`
   - ○ DHCP: **Enabled**
4. Apply and close

> Purpose: isolate your lab so nothing touches real networks.

Phase 2 — VM Configuration

Step 2: Attach Both VMs to Same Network

For **Kali Linux**:

- Network Adapter → **Host-only (VMnet1)**

For **Windows 7**:

- Network Adapter → **Host-only (VMnet1)**

⚠️ If these differ, nothing works.

Phase 3 — IP Address Verification

Step 3: Check IP Addresses

Expected result:

- Kali: `192.168.56.xxx`
- Windows 7: `192.168.56.yyy`

Rules:

- Same first 3 octets
- Different last octet
- NOT `169.254.x.x`

If IPs are wrong → fix networking **before anything else**.

Phase 4 — Connectivity Validation

Step 4: Basic Network Reachability

Before security testing, confirm:

- Kali can reach Windows
- Windows can reach Kali

If this fails:

> Exploits will *never* work — this is not optional.

Phase 5 — Target Understanding (This Is Where Learners Usually Skip)

Step 5: Identify the Target System

You must confirm:

- Windows version (Windows 7)
- Architecture (32-bit vs 64-bit)
- Patch level
- Services enabled (e.g. file sharing)

💡 **This step determines whether EternalBlue is even possible.**

Most failures happen because this step was skipped.

Phase 6 — Vulnerability Validation (NOT Exploitation)

Step 6: Check for Vulnerability *Safely*

Professionals:

- Check whether the system **would be vulnerable**
- Do **not** jump straight to exploitation

Why?

- Many systems are patched
- Exploits crash machines
- Detection matters more than exploitation

If the system is patched:

Stopping here is the **correct outcome**, not failure.

Phase 7 — Analysis & Documentation

Step 7: Record Findings

A real cybersecurity workflow ends with:

- Network diagram
- OS & service identification
- Vulnerability status
- Reason exploit would / wouldn’t work

Tab 9

BlueKeep (CVE-2019-0708)

What it is

BlueKeep is a **critical vulnerability in Microsoft Remote Desktop (RDP)**.

- Affects:
  Windows 7, Windows Server 2008 (and older)
- Where:
  The RDP service (port 3389)
- Type:
  Pre-authentication, remote code execution
- Year disclosed:
  2019

Why it was scary

- An attacker didn't need a username or password
- It was **"wormable"** (could spread automatically)
- It worked **before login**
- A single exposed machine could infect others

Because of this, Microsoft:

- Issued **emergency patches**
- Even patched **end-of-life Windows XP** (very rare)

Real-world impact

- Fewer mass attacks than expected
- Exploitation was **unstable**

- Still considered extremely dangerous if unpatched

EternalBlue (MS17-010)

What it is

EternalBlue is a vulnerability in **SMB (file sharing)**.

- Affects:
  Older Windows versions (XP → Windows 7)
- Where:
  SMBv1 (port 445)
- Type:
  Remote code execution
- Year disclosed:
  2017

Why it became infamous

EternalBlue was used in:

- **WannaCry ransomware**
- **NotPetya**
- Global outbreaks that:
  - Shut down hospitals
  - Hit shipping companies
  - Caused billions in damage

Unlike BlueKeep:

- EternalBlue exploitation was **reliable**
- Automated attacks spread rapidly worldwide

Key differences (important)

Feature	BlueKeep	EternalBlue
Protocol	RDP (3389)	SMB (445)
Auth required	❌ No	❌ No
Reliability	❌ Unstable	✅ Very reliable
Wormable	✅ Yes	✅ Yes
Used in major outbreaks	❌ (mostly avoided)	✅ (WannaCry)
Patch urgency	Emergency	Emergency

Why they're often mentioned together

Both:

- Are **"Blue" vulnerabilities**
- Are **wormable**
- Target Windows network services
- Forced Microsoft into emergency action
- Changed how organizations treat patching

They're taught together because they represent:

> "What happens when core Windows services are exposed."

Defensive lesson (the real takeaway)

Security professionals remember them not for exploitation, but because they proved:

- Exposed services are dangerous
- Patching matters more than firewalls alone
- Legacy systems are a massive risk
- One bug can cause global damage

1️⃣ SMB (Server Message Block)

What it is

- **SMB** is a network protocol that allows computers to **share files, printers, and other resources** over a network.
- Mostly used by **Windows computers**, but other OSes can support it too.

How it works (conceptually)

- Think of SMB as a **messenger for your files**:
    - You ask a computer: “Can I see this folder?”
    - SMB handles the request, sends the data, and manages permissions.

Common ports

- **445** (modern Windows)
- **139** (older systems)

Why it matters in security

- SMB exposes a lot of internal data if misconfigured
- Vulnerabilities in SMB (like **EternalBlue**) allow attackers to run code remotely without credentials
- Historically caused **WannaCry and NotPetya** outbreaks

2️⃣ RDP (Remote Desktop Protocol)

What it is

- **RDP** is a protocol for **remotely controlling a Windows computer**.
- It lets you **see the desktop** of another computer and operate it as if you were sitting there.

How it works (conceptually)

- You connect to a Windows PC over a network
- The remote computer sends its screen, mouse, and keyboard data
- You can work on it as if local

Common port

- **3389**

Why it matters in security

- Exposed RDP can allow attackers to **log in remotely**
- Vulnerabilities like **BlueKeep** let attackers run code without a username or password
- Often targeted in ransomware attacks

Key difference

Feature	SMB	RDP
Purpose	File/printer sharing	Remote control of a computer
Port	445	3389
User-visible	No	Yes (desktop interface)
Common attack example	EternalBlue	BlueKeep
Access type	Data/service access	Full desktop/session control

Simple analogy

- **SMB** = “I want to borrow your files or printer”
- **RDP** = “I want to sit at your computer and use it myself”