04. Computer Networks - Comprehensive Textbook & Interview Reference

> **Module Focus**: In-depth coverage of OSI 7-Layer vs TCP/IP Architecture, Classful & CIDR Subnetting Math (with Worked Numerical Examples), TCP 3-Way Handshake & 4-Way Termination, Flow & Congestion Control, DNS, DHCP DORA Process, ARP/RARP Mechanics, and Networking Hardware.

---

1. Computer Network Architecture: OSI 7-Layer vs TCP/IP Models

1.1 The 7-Layer OSI Reference Model

The **Open Systems Interconnection (OSI)** model is a theoretical framework defined by ISO that standardizes network communications into 7 distinct layers. Each layer performs specific functions and communicates directly with the layer immediately above and below it via predefined interfaces and **Protocol Data Units (PDUs)**.

```
+-----------------------------------------------------------------------------------+
|  OSI LAYER               PDU                    PRIMARY FUNCTIONS / PROTOCOLS     |
+-----------------------------------------------------------------------------------+
|  7. Application          Data                   User services, HTTP, DNS, DHCP, FTP  |
|  6. Presentation         Data                   Data format, Encryption, SSL/TLS     |
|  5. Session              Data                   Session management, RPC, Sockets     |
+-----------------------------------------------------------------------------------+
|  4. Transport            Segment (TCP) /        End-to-End Reliability, Port #s,     |
|                          Datagram (UDP)         TCP, UDP, Flow & Congestion Control  |
+-----------------------------------------------------------------------------------+
|  3. Network              Packet                 Logical Routing, IP Addressing,      |
|                                                 IPv4, IPv6, ICMP, ARP, OSPF, BGP     |
+-----------------------------------------------------------------------------------+
|  2. Data Link            Frame                  Node-to-Node Delivery, MAC Addr,     |
|                                                 LLC & MAC Sublayers, Switches, CRC   |
+-----------------------------------------------------------------------------------+
|  1. Physical             Bits                   Bit Transmission over Physical Medium|
|                                                 Hubs, Repeaters, Cables, Modulation  |
+-----------------------------------------------------------------------------------+
```

Layer-by-Layer Detailed Responsibilities:

1. **Physical Layer (Layer 1)**:
2. **Responsibility**: Unstructured bit-stream transmission over physical media (copper wire, fiber optics, radio waves).

3. **Key Features**: Voltage signaling, bit synchronization, physical data rates ($10\text{ Mbps}, 1\text{ Gbps}$), physical topology (Star, Bus, Ring).
4. **Hardware Devices**: Hubs, Repeaters, Transceivers, Network Cables (Cat6, Fiber).
5. **Data Link Layer (Layer 2)**:
6. **Responsibility**: Reliable node-to-node frame transfer across a single physical link.
7. **Sublayers**:
    - **LLC (Logical Link Control)**: Manages frame synchronization, flow control, and error checking.
    - **MAC (Media Access Control)**: Controls hardware access to the medium using 48-bit physical MAC addresses (e.g., `00:1A:2B:3C:4D:5E`).
8. **Mechanisms**: Framing, Error Detection via **CRC (Cyclic Redundancy Check)**, Access control (**CSMA/CD** for Ethernet, **CSMA/CA** for Wi-Fi).
9. **Hardware Devices**: Layer 2 Switches, Network Interface Cards (NICs), Bridges.
10. **Network Layer (Layer 3)**:
11. **Responsibility**: End-to-end packet routing across heterogeneous inter-networks using logical addressing (**IPv4 / IPv6**).
12. **Key Functions**: Logical IP addressing, packet routing, path determination (Routing algorithms), IP fragmentation and reassembly.
13. **Protocols & Hardware**: Routers, Layer 3 Switches. Protocols: IPv4, IPv6, ICMP, IGMP, ARP, RARP, OSPF, BGP, RIP.
14. **Transport Layer (Layer 4)**:
15. **Responsibility**: End-to-end message delivery between process endpoints (Applications) on source and destination hosts.
16. **Key Functions**: Process-to-process communication via **16-bit Port Numbers** (`0–65535`), Segmentation and Reassembly, Connection Control, Flow Control (Sliding Window), and Congestion Control.
17. **Protocols**: TCP (Reliable, Connection-Oriented) and UDP (Unreliable, Connectionless).
18. **Session Layer (Layer 5)**:
19. **Responsibility**: Establishing, maintaining, synchronizing, and terminating communication sessions between applications.
20. **Key Functions**: Dialog control (Simplex, Half-Duplex, Full-Duplex), synchronization checkpoints for long file transfers, RPC (Remote Procedure Call).
21. **Presentation Layer (Layer 6)**:
22. **Responsibility**: Data formatting, translation, syntax matching, data compression, and encryption/decryption.
23. **Key Functions**: Character encoding conversion (ASCII, EBCDIC, Unicode), Data Encryption/Decryption (TLS/SSL), Data Compression (JPEG, MPEG, GZIP).

24. **Application Layer (Layer 7)**:
25. **Responsibility**: Directly provides interface services to end-user software applications (Web browsers, email clients).
26. **Key Protocols**: HTTP, HTTPS, FTP, SFTP, SMTP, POP3, IMAP, DNS, DHCP, Telnet, SSH.

1.2 The TCP/IP Architecture Model

The **TCP/IP Model** (Internet Architecture) is a practical 4-layer (or 5-layer update) standard designed by DARPA that forms the backbone of modern Internet communications.

Layer Mapping: OSI vs TCP/IP (4-Layer & 5-Layer Models)

OSI 7-Layer Model	TCP/IP 5-Layer Model	TCP/IP 4-Layer Model	PDU	Core Protocols / Technologies
7. Application	**5. Application**	**4. Application**	Data	HTTP, HTTPS, DNS, DHCP, FTP, SMTP, SSH, Telnet
6. Presentation			Data	SSL/TLS, ASCII, JPEG
5. Session			Data	RPC, NetBIOS, Sockets
4. Transport	**4. Transport**	**3. Transport**	Segment / Datagram	TCP, UDP
3. Network	**3. Network**	**2. Internet**	Packet	IPv4, IPv6, ICMP, ARP, RARP, OSPF, BGP
2. Data Link	**2. Data Link**	**1. Network Access / Link**	Frame	Ethernet (802.3), Wi-Fi (802.11), PPP, Switches
1. Physical	**1. Physical**		Bits	Cables, Hubs, Signal Encoding, Bitstreams

Key Conceptual Differences:

- **Theoretical vs Practical**: OSI is a strict theoretical model designed before protocol implementation. TCP/IP was created based on practical working protocols.
- **Upper Layers Combined**: In the 4-layer TCP/IP model, OSI Application, Presentation, and Session layers are merged into a single **Application Layer**.
- **Connection Services**:
- OSI supports both connection-oriented and connectionless service at the Network layer.
- TCP/IP supports *only connectionless* service (IP protocol) at the Network layer; connection-orientation is handled strictly at the Transport layer (TCP).

---

2. IP Addressing & Subnetting Mechanics

2.1 IPv4 Address Fundamentals

An IPv4 address is a **32-bit logical address** represented in **dotted-decimal notation** (4 octets of 8 bits each separated by dots), e.g., `192.168.10.1`.

$$\text{Total IPv4 Address Space} = 2^{32} = 4,294,967,296 \text{ addresses}$$

Each octet ranges from 0 (`00000000`) to 255 (`11111111`).

2.2 Classful IP Address Ranges

Historically, IPv4 addresses were divided into 5 classes based on the leading bits of the first octet:

Class	1st Octet Binary Prefix	1st Octet Range	Default Subnet Mask	Default CIDR	Network Bits	Host Bits	Max Usable Hosts per Network	Primary Purpose
Class A	`0...`	`1.0.0.0` – `126.255.255.255`	`255.0.0.0`	`/8`	8	24	$2^{24} - 2 = 16,777,214$	Large Enterprise / ISPs
Class B	`10..`	`128.0.0.0` – `191.255.255.255`	`255.255.0.0`	`/16`	16	16	$2^{16} - 2 = 65,534$	Medium-sized Organizations
Class C	`110.`	`192.0.0.0` – `223.255.255.255`	`255.255.255.0`	`/24`	24	8	$2^8 - 2 = 254$	Small Local LANs
Class D	`1110`	`224.0.0.0` – `239.255.255.255`	Reserved	N/A	N/A	N/A	N/A	Multicasting Groups
Class E	`1111`	`240.0.0.0` – `255.255.255.255`	Reserved	N/A	N/A	N/A	N/A	Research & Experimental

> **Note**: `127.0.0.0/8` (specifically `127.0.0.1`) is omitted from Class A assignment as it is globally reserved for **Loopback Diagnostic Testing**.

2.3 Reserved & Private IP Ranges

RFC 1918 Private IP Address Ranges (Non-routable on Public Internet):

- **Class A Private**: `10.0.0.0` to `10.255.255.255` (`10.0.0.0/8`)
- **Class B Private**: `172.16.0.0` to `172.31.255.255` (`172.16.0.0/12`)
- **Class C Private**: `192.168.0.0` to `192.168.255.255` (`192.168.0.0/16`)

Special Purpose Ranges:

- **APIPA (Automatic Private IP Addressing)**: `169.254.0.0/16` (`169.254.0.1` to `169.254.255.254`). Automatically assigned by Windows/DHCP clients when a DHCP server is unreachable.
- **Limited Broadcast Address**: `255.255.255.255` (Broadcasts to all hosts on the local network segment; not forwarded by routers).
- **Default / Unspecified Route**: `0.0.0.0/0`.

2.4 CIDR Subnetting Formulas & Mathematics

Classless Inter-Domain Routing (CIDR) replaced classful addressing by allowing arbitrary network prefix lengths denoted by `/N`.

Core Subnetting Equations:

1. **Given CIDR Prefix**: $/N$ (where N is the number of network bits).
2. **Host Bits (H)**: $$H = 32 - N$$
3. **Total IP Addresses per Subnet**: $$\text{Total IPs} = 2^H$$
4. **Usable Host Addresses per Subnet**: $$\text{Usable Hosts} = 2^H - 2$$ *(Subtract 2: one for Network ID where host bits are all 0s, and one for Broadcast ID where host bits are all 1s).*
5. **Number of Created Subnets** (when borrowing b bits from classful host portion): $$\text{Subnets} = 2^b$$
6. **Block Size (Magic Number Method)**: $$\text{Block Size} = 256 - \text{Interesting Octet Subnet Mask Decimal Value} = 2^{(8 - \text{bits in interesting octet})}$$

2.5 Worked Numerical Subnetting Examples

Worked Example 1: Class C CIDR Subnetting

Problem: Given the IP address `192.168.1.140/27`, calculate: 1. Subnet Mask 2. Network ID 3. First Usable Host IP 4. Last Usable Host IP 5. Broadcast ID 6. Total Subnets & Usable Hosts per Subnet

Step-by-Step Solution: 1. **Prefix Length**: $N = 27$. Host bits $H = 32 - 27 = 5$. 2. **Subnet Mask**: - Binary: `11111111.11111111.11111111.11100000` - Decimal: `255.255.255.224` (Since $128 + 64 + 32 = 224$). 3. **Block Size**: $256 - 224 = 32$ (or $2^5 = 32$). 4. **Subnet Increments in 4th Octet**: Multiples of 32 ($0, 32, 64, 96, 128, 160, 192, 224$). 5. **Locate Target IP (`.140`)**: - Lies between 128 and 159. - **Network ID**: `192.168.1.128` - **First Usable Host IP**: `192.168.1.129` - **Last Usable Host IP**: `192.168.1.158` - **Broadcast ID**: `192.168.1.159` 6. **Host Capacity**: - Total IPs per subnet $= 2^5 = 32$. - Usable Hosts per subnet $= 32 - 2 = 30$. - Total Subnets created from default $/24$ $= 2^{27 - 24} = 2^3 = 8$ subnets.

Worked Example 2: Class B Subnetting

Problem: Given the IP address `172.16.45.100/22`, determine the network details.

Step-by-Step Solution: 1. **Prefix Length**: $N = 22$. Host bits $H = 32 - 22 = 10$. 2. **Subnet Mask**: - Binary: `11111111.11111111.11111100.00000000` - Decimal: `255.255.252.0` 3. **Interesting Octet**: 3rd octet (252). - Block Size in 3rd Octet $= 256 - 252 = 4$ (or $2^{8 - 6} = 4$). 4. **3rd Octet Subnet Multiples**: $0, 4, 8, 12, \dots, 40, 44, 48, \dots$ 5. **Locate Target Octet value (`45`)**: - Lies in the block starting at 44 up to 47. - **Network ID**: `172.16.44.0` - **First Usable Host IP**: `172.16.44.1` - **Last Usable Host IP**: `172.16.47.254` - **Broadcast ID**: `172.16.47.255` 6. **Host Capacity**: - Total IPs $= 2^{10} = 1024$. - Usable Hosts $= 1024 - 2 = 1022$.

Worked Example 3: Class A Subnetting

Problem: Find the Network ID and Broadcast address for `10.150.32.15/19`.

Step-by-Step Solution: 1. **Prefix Length**: $N = 19$. Host bits $H = 32 - 19 = 13$. 2. **Subnet Mask**: `255.255.224.0` (Binary 3rd octet: `11100000`). 3. **Block Size in 3rd Octet**: $256 - 224 = 32$. 4. **3rd Octet Multiples**: $0, 32, 64, 96, 128, 160, \dots$ 5. **Target 3rd Octet (`32`)**: - Exact match at multiple 32! Range: 32.0 to 63.255. - **Network ID**: `10.150.32.0` - **First Usable Host IP**: `10.150.32.1` - **Last Usable Host IP**: `10.150.63.254` - **Broadcast ID**: `10.150.63.255` - **Usable Hosts**: $2^{13} - 2 = 8192 - 2 = 8190$.

Worked Example 4: VLSM (Variable Length Subnet Masking) Design

Problem: An organization is assigned block `192.168.10.0/24`. Allocate subnets efficiently for: - Department A: 60 hosts - Department B: 28 hosts - Department C: 12 hosts - WAN Router Link: 2 hosts

Step-by-Step Solution (Sort requirements largest to smallest):

1. **Dept A (60 hosts)**:
2. Needed IPs $= 60 + 2 = 62$. Smallest power of 2 is $2^6 = 64 \implies H = 6$.
3. Prefix length $= 32 - 6 = /26$.
4. Subnet Mask: `255.255.255.192`.
5. **Network ID**: `192.168.10.0/26` (Range: `.0` to `.63`).
6. Usable Range: `192.168.10.1` – `192.168.10.62`, Broadcast: `192.168.10.63`.
7. **Dept B (28 hosts)**:
8. Next available IP starts at `192.168.10.64`.
9. Needed IPs $= 28 + 2 = 30$. Smallest power of 2 is $2^5 = 32 \implies H = 5$.
10. Prefix length $= 32 - 5 = /27$.
11. Subnet Mask: `255.255.255.224`.
12. **Network ID**: `192.168.10.64/27` (Range: `.64` to `.95`).
13. Usable Range: `192.168.10.65` – `192.168.10.94`, Broadcast: `192.168.10.95`.
14. **Dept C (12 hosts)**:
15. Next available IP starts at `192.168.10.96`.
16. Needed IPs $= 12 + 2 = 14$. Smallest power of 2 is $2^4 = 16 \implies H = 4$.

17. Prefix length $= 32 - 4 = /28$.
18. Subnet Mask: `255.255.255.240`.
19. **Network ID**: `192.168.10.96/28` (Range: `.96` to `.111`).
20. Usable Range: `192.168.10.97` – `192.168.10.110`, Broadcast: `192.168.10.111`.
21. **WAN Link (2 hosts)**:
22. Next available IP starts at `192.168.10.112`.
23. Needed IPs $= 2 + 2 = 4$. Power of 2 is $2^2 = 4 \implies H = 2$.
24. Prefix length $= 32 - 2 = /30$.
25. Subnet Mask: `255.255.255.252`.
26. **Network ID**: `192.168.10.112/30` (Range: `.112` to `.115`).
27. Usable Range: `192.168.10.113` – `192.168.10.114`, Broadcast: `192.168.10.115`.

---

3. Transport Layer Mechanics: TCP vs UDP

3.1 TCP 3-Way Handshake & Connection State Diagram

TCP is a **connection-oriented**, **reliable**, **full-duplex**, **byte-stream** transport protocol. Before data transmission begins, a 3-Way Handshake establishes initial sequence numbers (ISN) and buffer parameters.

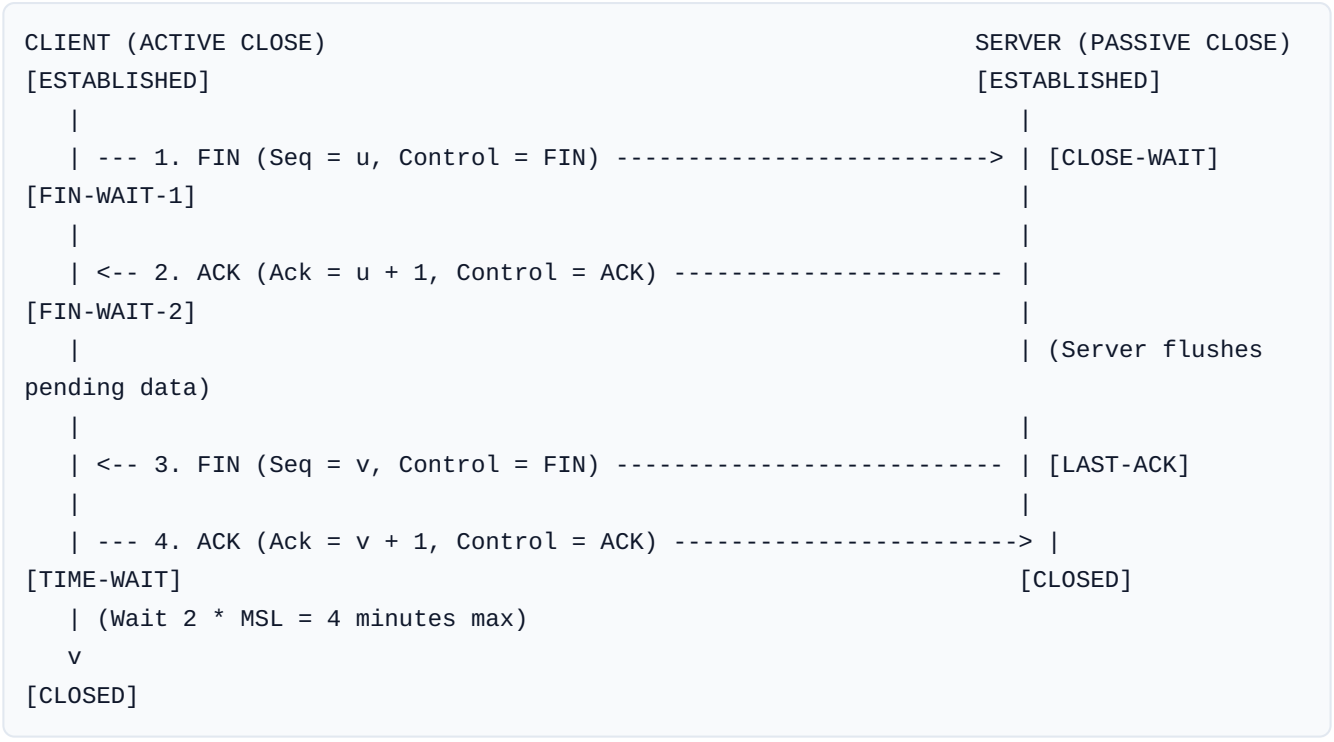
```
CLIENT (ACTIVE OPEN)                                                SERVER (PASSIVE OPEN)
[CLOSED]                                                            [LISTEN]
   |                                                                   |
   | --- 1. SYN (Seq = x, Control = SYN) -------------------------> | [SYN-RECEIVED]
   |                                                                   |
   | <-- 2. SYN-ACK (Seq = y, Ack = x + 1, Control = SYN, ACK) ------ |
[SYN-SENT]                                                          |
   |                                                                   |
   | --- 3. ACK (Seq = x + 1, Ack = y + 1, Control = ACK) ---------> |
[ESTABLISHED]                                                       [ESTABLISHED]
   |                                                                   |
   +================== BIDIRECTIONAL DATA TRANSFER ==================+
```

Detailed Step Mechanics:

1. **Step 1 (SYN)**: Client selects a random **Initial Sequence Number** x and sends a segment with control flag `SYN = 1`. Client state transitions to `SYN_SENT`.
2. **Step 2 (SYN-ACK)**: Server responds with its own random sequence number y, sets `SYN = 1`, and acknowledges client's sequence number by setting `ACK = 1` and `Ack Number = x + 1`. Server state transitions to `SYN_RCVD`.
3. **Step 3 (ACK)**: Client acknowledges server's sequence number by sending `ACK = 1`, `Seq Number = x + 1`, and `Ack Number = y + 1`. Both hosts enter `ESTABLISHED` state.

3.2 TCP 4-Way Connection Termination

Closing a TCP connection requires a 4-step exchange because TCP is full-duplex (each direction must be closed independently).



Critical Concept: Why `TIME_WAIT` State Exists

The client stays in `TIME_WAIT` for $2 \times \text{MSL}$ (Maximum Segment Lifetime, typically 30s to 2 minutes) to: 1. **Ensure Final ACK Delivery**: If the final ACK is lost in transit, the server retransmits the `FIN` frame. `TIME_WAIT` allows the client to resend the ACK. 2. **Prevent Stale Segment Corruption**: Ensures old duplicate packets belonging to the closed connection expire before a new connection reuses the same IP address and port numbers.

3.3 TCP Control Flags Summary

A standard 20-byte TCP header contains 6 primary 1-bit control flags:

Flag	Name	Function
SYN	Synchronize	Initiates a connection and synchronizes initial sequence numbers.
ACK	Acknowledge	Indicates that the Acknowledgment number field is valid.
FIN	Finish	Initiates connection termination (no more data from sender).
RST	Reset	Abruptly aborts/resets a broken or invalid connection.
PSH	Push	Forces immediate data push to the application layer without waiting for buffer to fill.
URG	Urgent	Marks urgent data processed out-of-band via Urgent Pointer.

3.4 TCP Flow & Congestion Control

1. Flow Control (Sliding Window Protocol):

- **Objective**: Prevents a fast sender from overwhelming a slow receiver's buffer.
- **Mechanism**: Receiver advertises a **Receive Window (`rwnd`)** in every TCP header. Sender ensures: $$ \text{Unacknowledged In-Flight Bytes} \le \text{rwnd}$$

2. Congestion Control:

- **Objective**: Prevents the network core routers from collapsing due to excessive traffic.
- **Mechanism**: Sender calculates a **Congestion Window (`cwnd`)**. Actual Transmission Limit: $$ \text{Effective Window} = \min(\text{rwnd}, \text{cwnd})$$
- **4 Congestion Control Phases**:
- **Slow Start**: `cwnd` starts at 1 MSS (Maximum Segment Size) and doubles exponentially every RTT ($1 \to 2 \to 4 \to 8$).
- **Congestion Avoidance**: When `cwnd` reaches threshold `ssthresh`, growth becomes linear ($+1 \text{ MSS}$ per RTT).
- **Fast Retransmit**: Triggered when sender receives **3 Duplicate ACKs**. Immediately retransmits missing segment without waiting for Retransmission Timeout (RTO).
- **Fast Recovery**: Sets `ssthresh = cwnd / 2`, sets `cwnd = ssthresh`, and resumes linear growth (TCP Reno).

3.5 Detailed Comparison: TCP vs UDP

Architectural Feature	TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
Connection Type	Connection-Oriented (3-way handshake required)	Connectionless (No handshake)
Reliability	**Guaranteed Delivery**: ACKs, retransmissions, checksums	**Best-Effort Delivery**: Packets may be lost, duplicated, out of order
Data Ordering	In-order delivery guaranteed (resequences packets)	No ordering guarantees (handled by application)
Header Overhead	Minimum 20 Bytes (up to 60 Bytes with Options)	Fixed 8 Bytes (Source Port, Dest Port, Length, Checksum)
Flow & Congestion Control	Yes (Sliding Window, Slow Start, Fast Recovery)	None
Data Framing	Byte-stream (continuous un-segmented stream)	Message datagrams (preserves packet boundaries)
Transmission Speed	Moderate (Overhead from ordering, ACKs, sliding window)	Ultra-Fast / Low Latency
Primary Protocols	HTTP, HTTPS, FTP, SSH, SMTP, Telnet	

Architectural Feature	TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
		DNS queries, DHCP, VoIP, Video Streaming, TFTP, SNMP

---

4. Core Network Protocols & Port Numbers

4.1 DNS (Domain Name System - Port 53)

DNS is a distributed hierarchical database that resolves human-readable hostnames (`www.example.com`) to IP addresses (`193.0.2.1`).

1. DNS Resolution Architecture:

```
CLIENT -> LOCAL DNS SERVER -> ROOT SERVER (.) -> TLD SERVER (.com) -> AUTHORITATIVE SERVER (example.com)
```

- **Recursive Query**: Client asks Local DNS Server to fully resolve the address. Local DNS server bears complete responsibility.
- **Iterative Query**: Local DNS Server queries upstream servers step-by-step; each server returns a referral point to the next hierarchy level.

2. Common DNS Record Types:

Record Type	Meaning / Purpose	Example Record Format
A	Maps IPv4 hostname to IP address	`example.com. IN A 93.184.216.34`
AAAA	Maps IPv6 hostname to IP address	`example.com. IN AAAA 2606:2800:220:1:248:1893:25c8:1946`
CNAME	Canonical Name (Alias mapping to another hostname)	`www.example.com. IN CNAME example.com.`
MX	Mail Exchange (Specifies mail server for domain)	`example.com. IN MX 10 mail.example.com.`
NS	Name Server (Delegates zone authority)	`example.com. IN NS ns1.example.com.`
PTR	Pointer Record (Reverse DNS lookup: IP to domain)	`34.216.184.93.in-addr.arpa. IN PTR example.com.`
TXT	Text record (Holds SPF, DKIM, verification text)	`example.com. IN TXT "v=spf1 include:_spf.google.com ~all"`

> **Transport Protocol Usage**: - **UDP Port 53**: Used for standard client DNS query requests (≤ 512 bytes payload). - **TCP Port 53**: Used for **Zone Transfers** between DNS servers or when responses exceed 512 bytes (or DNSSEC).

4.2 DHCP (Dynamic Host Configuration Protocol - Ports 67/68)

DHCP automatically configures IP addresses, subnet masks, default gateways, and DNS servers for clients on a network.

The 4-Step DHCP DORA Process:

```
CLIENT (Port 68)                                                SERVER (Port 67)
   |                                                                 |
   | --- 1. DISCOVER (L2: Broadcast, L3: 255.255.255.255) ---------> |
   |                                                                 |
   | <-- 2. OFFER (L2: Unicast/Broadcast, Offered IP) --------------- |
   |                                                                 |
   | --- 3. REQUEST (L2: Broadcast, Accepts Offered IP) -----------> |
   |                                                                 |
   | <-- 4. ACKNOWLEDGMENT (L2: Unicast/Broadcast, Confirms Lease) - |
   v                                                                 v
```

1. **D - Discover**: Client broadcasts a `DHCPDISCOVER` packet (`Src IP: 0.0.0.0` , `Dst IP: 255.255.255.255` , `Src MAC: Client MAC` , `Dst MAC: FF:FF:FF:FF:FF:FF`) to locate available DHCP servers.
2. **O - Offer**: Server responds with a `DHCPOFFER` packet proposing an IP address, Subnet Mask, Default Gateway, DNS Server, and Lease Time.
3. **R - Request**: Client broadcasts a `DHCPREQUEST` message notifying all DHCP servers that it accepts the proposed offer (broadcasting allows other DHCP servers to reclaim unselected offers).
4. **A - ACK (Acknowledgment)**: Server responds with a `DHCPACK` confirming the binding and lease duration.

> **DHCP Relay Agent**: Because DHCP Discover messages are Layer 3 broadcasts (`255.255.255.255`), routers block them by default. A **DHCP Relay Agent** (configured via `ip helper-address`) intercepts broadcast Discovers and forwards them as unicast packets to a DHCP server on another subnet.

4.3 ARP & RARP Protocol Mechanics

1. ARP (Address Resolution Protocol):

- **Purpose**: Maps a known 32-bit **Network IP Address** to an unknown 48-bit **Physical MAC Address** on the same local broadcast segment.

- **Operation**:
- **ARP Request**: Host broadcasts (`Dst MAC: FF:FF:FF:FF:FF:FF` , "Who has IP `192.168.1.1` ? Tell `192.168.1.50` ").
- **ARP Reply**: Destination node responds via **Unicast** (`Dst MAC: Host MAC` , "IP `192.168.1.1` is at MAC `AA:BB:CC:DD:EE:FF` ").
- Results are cached in the host's **ARP Table / Cache** with an expiration timer.

2. Special ARP Variations:

- **Gratuitous ARP**: An unprompted ARP Request/Reply sent by a host advertising its own IP/MAC mapping. Used for:
- **Duplicate IP Detection**: If another host replies, an IP collision exists.
- **Updating Neighbor Caches**: Updating ARP tables on switches/routers during NIC failover or IP changes.
- **Proxy ARP**: A router responds to an ARP request for an IP address that is not on the local network, pretending to be the destination so it can route traffic.

3. RARP (Reverse Address Resolution Protocol):

- **Purpose**: Maps a known 48-bit **Physical MAC Address** to an unknown **Logical IP Address**.
- **Historical Context**: Used by diskless workstations to obtain their IP address upon booting from a centralized RARP server.
- **Evolution**: Obsoleted by **BOOTP**, which was later superseded by **DHCP** (which supplies IP, Gateway, DNS, and subnet mask in one protocol).

---

5. Network Hardware & Domain Mechanics

5.1 Collision Domain vs Broadcast Domain

Domain Type	Definition	Segmented / Bounded By	Hardware Behavior
Collision Domain	Network section where simultaneous frame transmissions cause physical bit collisions.	Switches, Bridges, Routers	**Hubs**: 1 single collision domain for all ports. **Switches**: Each port is an isolated collision domain.
Broadcast Domain	Network section where a broadcast frame (`FF:FF:FF:FF:FF:FF`) is received by all nodes.	Routers, Layer 3 Switches, VLANs	**Hubs & Switches**: Forward broadcasts to all ports (1 broadcast domain). **Routers**: Stop Layer 3/2 broadcasts. Each interface is a separate broadcast domain.

Hardware Comparison Summary:

Feature	Hub (Layer 1)	Switch (Layer 2)	Router (Layer 3)
Operating Layer	Layer 1 (Physical)	Layer 2 (Data Link)	Layer 3 (Network)
Addressing Used	None (Bits)	MAC Addresses	IP Addresses
Collision Domains	1 Shared	1 per Port	1 per Port
Broadcast Domains	1 Shared	1 Shared (unless VLANs)	1 per Interface
Forwarding Logic	Floods all ports	MAC Table lookup	Routing Table lookup

---

5.2 Summary of Key Protocol Port Numbers

Port Number	Protocol	Transport Protocol	Application Purpose
20 / 21	FTP	TCP	File Transfer Protocol (Data / Control)
22	SSH / SFTP	TCP	Secure Shell / Secure File Transfer
23	Telnet	TCP	Unencrypted Text Remote Terminal
25	SMTP	TCP	Simple Mail Transfer Protocol
53	DNS	UDP / TCP	Domain Name System
67 / 68	DHCP	UDP	Server (67) / Client (68)
80	HTTP	TCP	Hypertext Transfer Protocol
110	POP3	TCP	Post Office Protocol v3
123	NTP	UDP	Network Time Protocol
143	IMAP	TCP	Internet Message Access Protocol
161 / 162	SNMP	UDP	Simple Network Management Protocol
443	HTTPS	TCP	HTTP Secure (SSL/TLS encrypted)

---

6. 🎯 CoCubes Computer Networks Technical MCQ Master Patterns & High-Yield Insights

6.1 TCP Handshake & Sequence Number MCQ Rules

CoCubes questions test precise packet flag states and sequence/acknowledgment numbers during connection setup and teardown:

1. Sequence Number Consumption Rule:

- **SYN** and **FIN** flags consume **1 unit of sequence space** even if payload length is 0 bytes!
- A standard **ACK** packet carrying no payload consumes **0 sequence space**.

2. 3-Way Handshake State Sequence:

$$\begin{array}{|c|c|c|c|c|} \hline \mathbf{Step} & \mathbf{Sender} & \mathbf{Flags\ Set} & \mathbf{Sequence\ Number} & \mathbf{Acknowledgment\ Number} \\ \hline \text{1} & \text{Client} & \text{SYN = 1, ACK = 0} & Seq = x & \text{N/A} \\ \hline \text{2} & \text{Server} & \text{SYN = 1, ACK = 1} & Seq = y & Ack = x + 1 \\ \hline \text{3} & \text{Client} & \text{SYN = 0, ACK = 1} & Seq = x + 1 & Ack = y + 1 \\ \hline \end{array}$$

3. 4-Way Teardown & `TIME_WAIT` Purpose:

- Connection termination requires 4 packets because TCP is **Full-Duplex** (each side closes its sending direction independently).
- Client enters `TIME_WAIT` state for $2 \times \text{MSL}$ (Maximum Segment Lifetime) to ensure the server receives the final `ACK` and to flush orphaned packets from the network.

6.2 Subnetting & CIDR Fast-Calculation Cheat Sheet

In timed CoCubes exams (30 questions in 30 minutes), use this lookup table for instant subnetwork math:

$$\begin{array}{|c|c|c|c|c|} \hline \mathbf{CIDR} & \mathbf{Subnet\ Mask} & \mathbf{Block\ Size} & \mathbf{Host\ Bits\ (H)} & \mathbf{Usable\ Hosts\ (2^H - 2)} \\ \hline \mathbf{/24} & 255.255.255.0 & 256 & 8 & 254 \\ \hline \mathbf{/25} & 255.255.255.128 & 128 & 7 & 126 \\ \hline \mathbf{/26} & 255.255.255.192 & 64 & 6 & 62 \\ \hline \mathbf{/27} & 255.255.255.224 & 32 & 5 & \mathbf{30} \\ \hline \mathbf{/28} & 255.255.255.240 & 16 & 4 & \mathbf{14} \\ \hline \mathbf{/29} & 255.255.255.248 & 8 & 3 & \mathbf{6} \\ \hline \mathbf{/30} & 255.255.255.252 & 4 & 2 & \mathbf{2\ (Router\ Links)} \\ \hline \end{array}$$

> **CoCubes Calculation Shortcut**:
> To find the Network ID for IP `X.Y.Z.W/N`: 1. Determine `Block Size` $= 256 - \text{Subnet Mask Octet}$. 2. Divide `W` by `Block Size` and take floor: $K = \lfloor W / \text{Block Size} \rfloor$. 3. `Network ID` octet $= K \times \text{Block Size}$. 4. `Broadcast ID` octet $= (K + 1) \times \text{Block Size} - 1$.

6.3 Layer Protocol Classification Distractors

Feature / Protocol	Correct OSI Layer	CoCubes Distractor Trap
SSL / TLS / Encryption	Presentation (Layer 6)	Don't confuse with Session or Transport.
DNS / DHCP / HTTP	Application (Layer 7)	DNS uses UDP/TCP Port 53, but is Layer 7.
ARP / RARP		

Feature / Protocol	Correct OSI Layer	CoCubes Distractor Trap
	Network (Layer 3) / Data Link (Layer 2.5)	Operates between L2 and L3; maps IP $ \leftrightarrow$ MAC.
ICMP (Ping/ Traceroute)	Network (Layer 3)	Encapsulated inside IP datagrams, but functions at L3.
TCP / UDP	Transport (Layer 4)	Handles port numbers (`0–65535`) & end-to-end reliability.