

Advance Computer Network (315321)

Model Question Paper — Set A · Model Answers

Q.1

a) Roles of ICANN

2 M

ICANN (Internet Corporation for Assigned Names and Numbers) is a private, non-profit organization (formed in 1998) that coordinates the unique identifiers of the Internet.

1. Allocates blocks of IP addresses (IPv4 / IPv6) to Regional Internet Registries such as APNIC (IANA function).
2. Manages the DNS root zone and coordinates the root name servers.
3. Creates and manages top-level domains (.com, .org, new gTLDs, country-code TLDs) and accredits domain-name registrars.
4. Maintains protocol parameters such as well-known port numbers and protocol numbers; frames policies and resolves domain-name disputes.

(Any four roles, ½ M each)

b) Routing table and its fields

2 M

Routing table: A table stored in every router (and host) that holds, for each destination network, the information needed to forward a packet — mainly the next hop and the outgoing interface. The router looks up the destination address of each packet in this table.

Fields: Mask, Network (destination) address, Next-hop address, Interface; optional fields — Metric (cost), Flags, Reference count, Use.

(Definition 1 M, any two fields 1 M)

c) Applications of UDP

2 M

1. DNS queries (port 53) — short request/response.
2. SNMP network management (ports 161/162).
3. TFTP — simple file transfer (port 69); DHCP/BOOTP (67/68).
4. Routing updates of RIP (port 520).
5. Real-time multimedia: VoIP, live video streaming, online games (speed more important than retransmission).
6. Multicasting and broadcasting (TCP cannot do this).

(Any four, ½ M each)

d) Domain name space; generic domains

2 M

Domain name space: The hierarchical naming structure of DNS, organised as an **inverted tree** with the unnamed root at the top and at most 128 levels. Each node has a label (up to 63 characters); the full domain name is the sequence of labels from the node up to the root, separated by dots (e.g. *www.google.com.*).

Generic domains: com (commercial), edu (education), org (non-profit), gov (government), net (network providers), mil (military).

(Definition 1 M, two examples 1 M)

e) Need of VoIP

2 M

1. Low call cost — voice travels over the existing IP data network instead of the costly circuit-switched telephone network (cheap long-distance / international calls).
2. One converged network for voice, video and data reduces infrastructure and maintenance cost.
3. Extra features: video calling, conferencing, call forwarding, voicemail to e-mail.
4. Mobility — calls can be made from anywhere using Internet (softphone, mobile app).

(Any two needs, 1 M each)

f) Fragmentation in IPv4

2 M

Each physical network has a **Maximum Transfer Unit (MTU)** — e.g. Ethernet 1500 bytes. When an IP datagram is larger than the MTU of the next network, it is divided into smaller pieces called **fragments**; each fragment gets its own header and is reassembled only at the final destination.

Fields used: Identification (same value in all fragments), Flags (D = do not fragment, M = more fragments) and Fragment offset (position of data in units of 8 bytes).

(Need 1 M, fields 1 M)

g) Connectionless vs connection-oriented service

2 M

Sr.	Parameter	Connectionless service	Connection-oriented service
1	Connection	No connection set-up or release.	Three phases: connection establishment, data transfer, termination.
2	Order / reliability	Packets are independent, may be lost or arrive out of order; no acknowledgement.	Data arrives in order; acknowledgement and retransmission make it reliable.
3	Example	UDP, IP.	TCP, SCTP.

(Any two points, 1 M each)

Q.2

a) ARP address mapping process

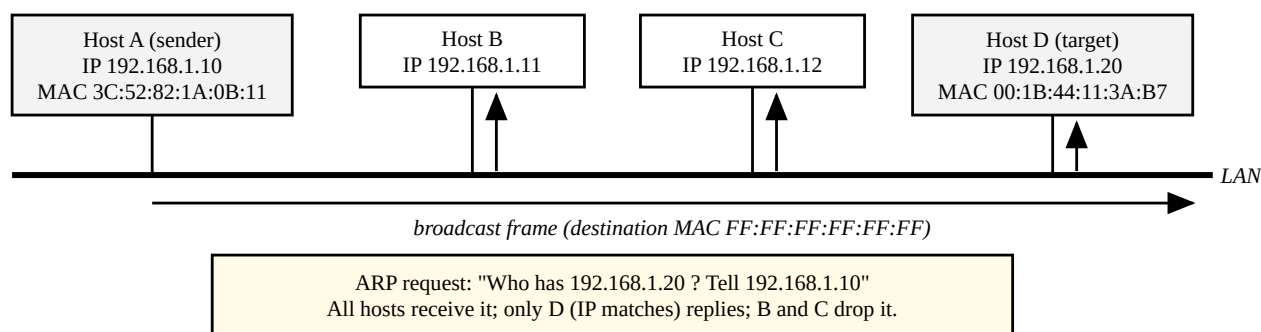
4 M

ARP (Address Resolution Protocol) maps a **logical (IP) address** to a **physical (MAC) address**. A host knows the IP address of the next hop but needs its MAC address to build the frame.

1. The sender first checks its **ARP cache**; if the mapping is present, it is used directly.
2. Otherwise it builds an **ARP request** containing sender MAC, sender IP, target IP and target MAC = all 0s.
3. The request is **broadcast** on the LAN (destination MAC FF:FF:FF:FF:FF:FF; Ethernet type 0x0806).
4. Every host receives it, but only the host whose IP matches the target IP processes it.

5. The target sends an **ARP reply** containing its MAC address, as a **unicast** frame to the sender.
6. The sender stores the mapping in its cache and sends the IP datagram in a frame to that MAC address.

a) ARP request (broadcast)



b) ARP reply (unicast)

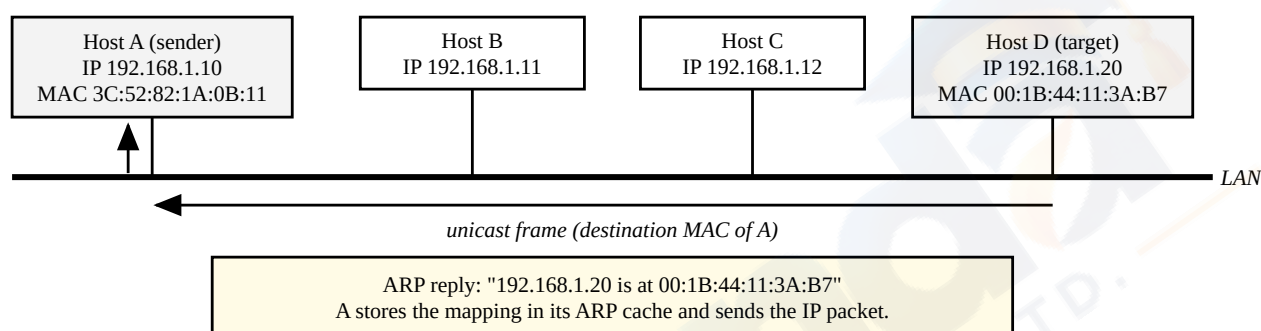


Fig.: ARP operation — request is broadcast to all, reply is unicast to the requester

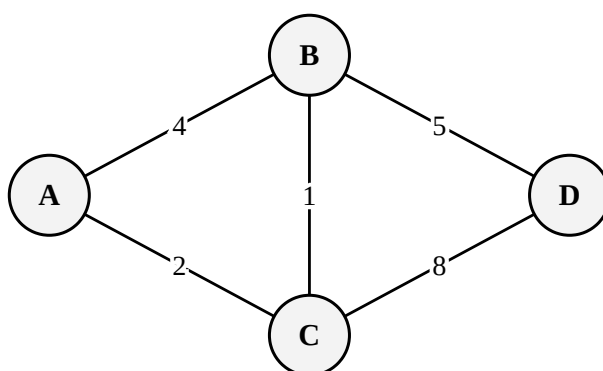
(Explanation 2 M, diagram 2 M)

b) Distance vector routing with example

4 M

In **distance vector routing** each router keeps a table (vector) of the least cost to every destination and periodically shares it **only with its immediate neighbours**. It uses the **Bellman-Ford** equation: $D_x(y) = \min \text{ over neighbours } v \{ c(x, v) + D_v(y) \}$.

1. **Initialization:** a router knows only the cost to its directly connected neighbours; others are ∞ .
2. **Sharing:** each router sends its vector to its neighbours (RIP: every 30 s).
3. **Updating:** the receiver adds the link cost to every received entry; it replaces its entry if the new cost is smaller (or if the update comes from the same next hop).



Given network

Initial tables (each router knows only neighbours):

To	A: cost, next	B: cost, next	C: cost, next	D: cost, next
A	0, —	4, A	2, A	∞ , —
B	4, B	0, —	1, B	5, B
C	2, C	1, C	0, —	8, C
D	∞ , —	5, D	8, D	0, —

A receives C's vector (B = 1, D = 8): via C, B costs $2 + 1 = 3 < 4 \rightarrow$ update B: 3 via C; D costs $2 + 8 = 10$. A receives B's vector (D = 5): via B, D = $4 + 5 = 9 < 10$. Meanwhile C has learnt D = $1 + 5 = 6$ via B; in the next round C's new vector gives A $\rightarrow$ D via C = $2 + 6 = 8 < 9 \rightarrow$ update D: 8 via C.

Final routing table of A (converged after 2 exchange rounds):

Destination	Cost	Next hop
A	0	—
B	3	C
C	2	C
D	8	C

(Explanation 2 M, example 2 M)

c) Features of SCTP

4 M

SCTP (Stream Control Transmission Protocol) is a reliable, message-oriented transport-layer protocol that combines the best features of UDP (message boundaries) and TCP (reliability, congestion control).

- Multiple streams:** one association can carry many streams; loss in one stream does not block others (no head-of-line blocking) — useful for multimedia.
- Multihoming:** each end can have several IP addresses; if the primary path fails, data moves to an alternate path (fault tolerance).
- Full-duplex, connection-oriented:** a connection is called an **association**, set up by a four-way handshake with a **cookie** that protects against SYN-flooding attacks.
- Reliable and message-oriented:** uses TSN (transmission sequence number), SACK acknowledgements and retransmission; message boundaries are preserved; flow and congestion control are similar to TCP.

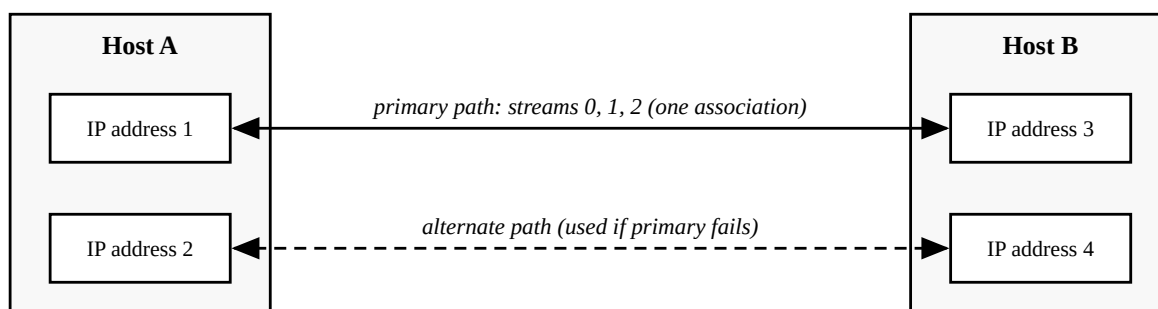


Fig.: SCTP association with multihoming (two paths) and multiple streams

(Any four features, 1 M each)

d) Working of POP3

4 M

POP3 (Post Office Protocol version 3) is a **message access agent (MAA)** protocol used by a receiver to **pull** mail from the mailbox on the mail server to his computer. The client runs on the user's PC, the server on the mail server; it uses **TCP port 110**.

1. Client opens a TCP connection to the server on port 110.
2. Client sends user name and password (USER / PASS) for authentication.
3. Client lists the messages (LIST) and retrieves them one by one (RETR).
4. **Delete mode:** the mail is deleted from the mailbox after retrieval (DELE). **Keep mode:** the mail remains in the mailbox (used when reading from many computers).
5. QUIT ends the session.

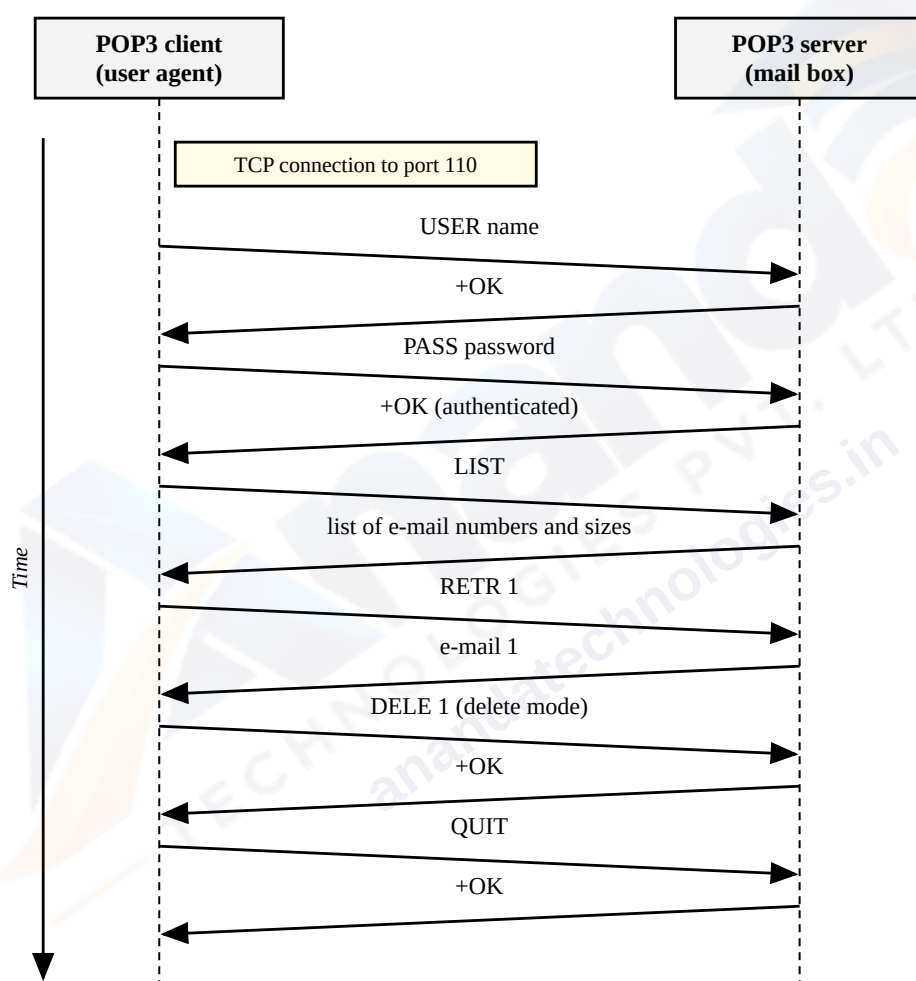


Fig.: POP3 — message access between user agent and mail server

(Working 2 M, diagram 2 M)

Q.3

a) UDP header format

4 M

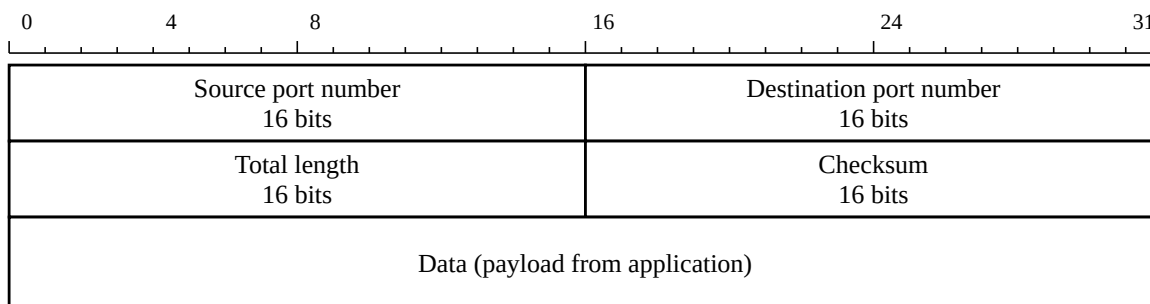


Fig.: UDP user datagram header (8 bytes)

UDP packets are called **user datagrams**; the header is fixed at **8 bytes** (four 16-bit fields).

- Source port number (16 bits):** port of the sending process (0 – 65,535); an ephemeral port if the sender is a client.
- Destination port number (16 bits):** port of the receiving process, e.g. 53 for DNS.
- Total length (16 bits):** length of header + data in bytes (minimum 8; maximum 65,535, practically less because the datagram must fit in an IP datagram).
- Checksum (16 bits):** detects errors in the whole user datagram (pseudo-header + UDP header + data). It is optional in IPv4 (all 0s if not used) and mandatory in IPv6.

(Diagram 2 M, fields 2 M)

b) Open-loop vs closed-loop congestion control

4 M

Sr.	Parameter	Open-loop congestion control	Closed-loop congestion control
1	Aim	Prevents congestion before it happens.	Removes / reduces congestion after it has happened.
2	Feedback	No feedback from the network is used.	Uses feedback (signals) from the network / receiver.
3	Decision	Policies are applied at source or destination in advance.	Action is taken when congestion is detected.
4	Methods	Retransmission policy, window policy (selective repeat), acknowledgement policy, discarding policy, admission policy.	Back-pressure, choke packet, implicit signalling, explicit (forward / backward) signalling.
5	Example	Router refuses a new virtual circuit if it would cause congestion (admission policy).	TCP reduces its window on time-out or 3 duplicate ACKs (implicit signalling).

(Any four points, 1 M each)

c) RIPv2 message format

4 M

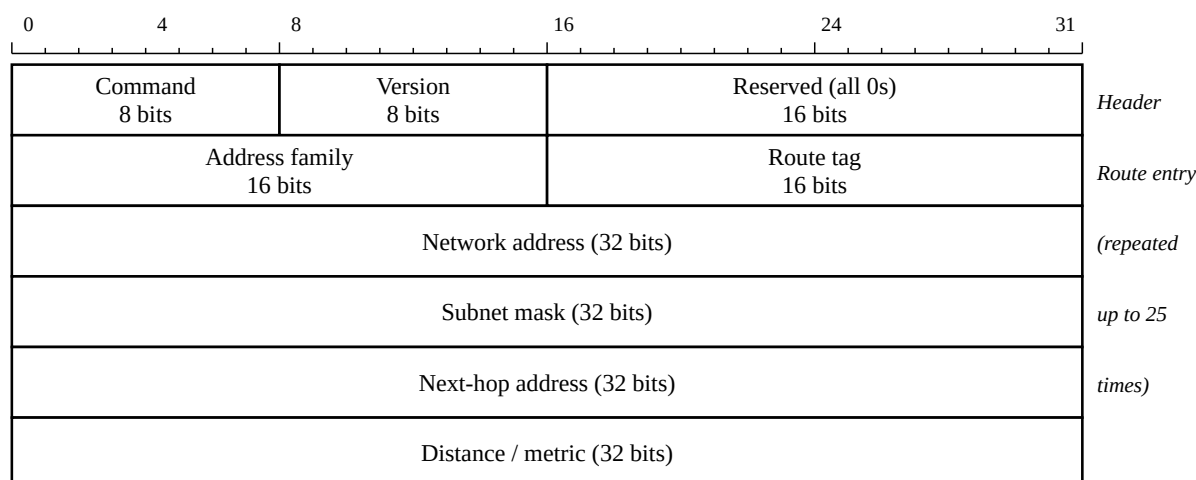


Fig.: RIPv2 message format

- **Command (8 bits):** 1 = request, 2 = response (update).
- **Version (8 bits):** 2 for RIPv2.
- **Address family (16 bits):** 2 for IP (FFFF when the entry carries authentication data).
- **Route tag (16 bits):** carries information such as the AS number of an external route.
- **Network address and subnet mask (32 bits each):** the destination; the mask makes RIPv2 classless (supports subnetting / CIDR).
- **Next-hop address (32 bits):** address of the next router on the route.
- **Distance (32 bits):** hop count 1 – 15; 16 means infinity (unreachable).

Up to 25 route entries are carried in one message; RIP messages are sent in UDP on port 520.

(Diagram 2 M, fields 2 M)

d) TCP three-way handshake

4 M

TCP is connection-oriented. The server first does a **passive open** (ready to accept); the client does an **active open**. The connection is set up in three steps:

1. **SYN:** client sends a segment with SYN flag set and its initial sequence number (ISN), e.g. seq = 8000. It carries no data but consumes one sequence number.
2. **SYN + ACK:** server replies with SYN and ACK flags set, its own ISN (seq = 15000) and ack = 8001.
3. **ACK:** client acknowledges with ack = 15001 (seq = 8001). The connection is now established in both directions (full-duplex).

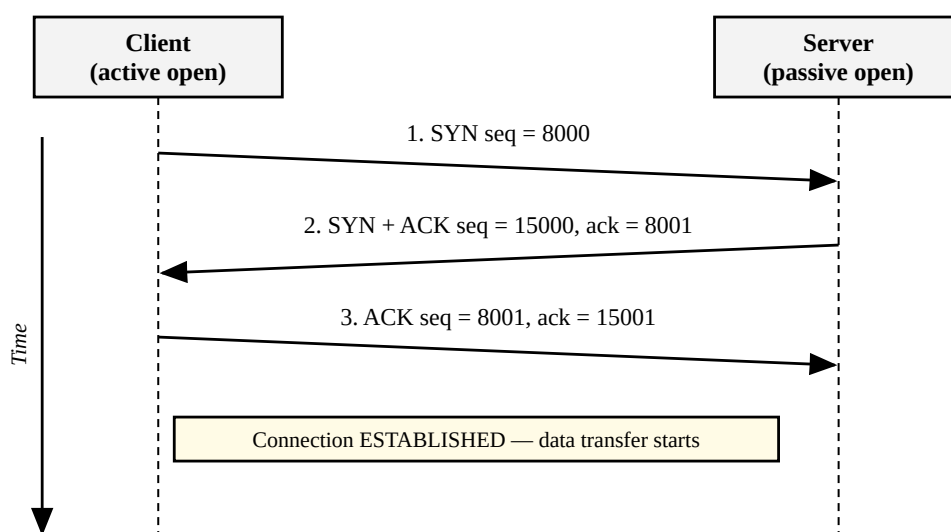


Fig.: TCP connection establishment — three-way handshake

(Steps 2 M, diagram 2 M)

Q.4

a) IPv4 vs IPv6

4 M

Sr.	Parameter	IPv4	IPv6
1	Address size	32 bits (about 4.3 billion addresses).	128 bits (about 3.4×10^{38} addresses).
2	Notation	Dotted decimal, e.g. 192.168.1.1	Hexadecimal colon, e.g. 2001:db8::1
3	Header	Variable 20 – 60 bytes, has options.	Fixed 40-byte base header; options in extension headers.
4	Checksum	Header checksum present.	No header checksum (faster processing).
5	Fragmentation	By source and routers.	Only by the source.
6	Address types	Unicast, multicast, broadcast.	Unicast, multicast, anycast (no broadcast).
7	Configuration / security	Manual or DHCP; IPsec optional.	Auto-configuration (SLAAC); IPsec support built in.

(Any four points, 1 M each)

b) Static, dynamic and active web documents

4 M

- **Static document:** a fixed-content file created and stored on the server. Its content is decided at creation time; every client gets the same copy (e.g. an HTML page of college information). Fast, simple, but must be edited manually to change.
- **Dynamic document:** created by the web server **when a request arrives**. The server runs a program/script (CGI, PHP, JSP, ASP) and returns its output, so contents can differ for each request (e.g. exam result page, current time, train availability).
- **Active document:** a program or script sent by the server and **run at the client (browser)** (e.g. JavaScript, Java applet) — used for animation or interaction with the user.

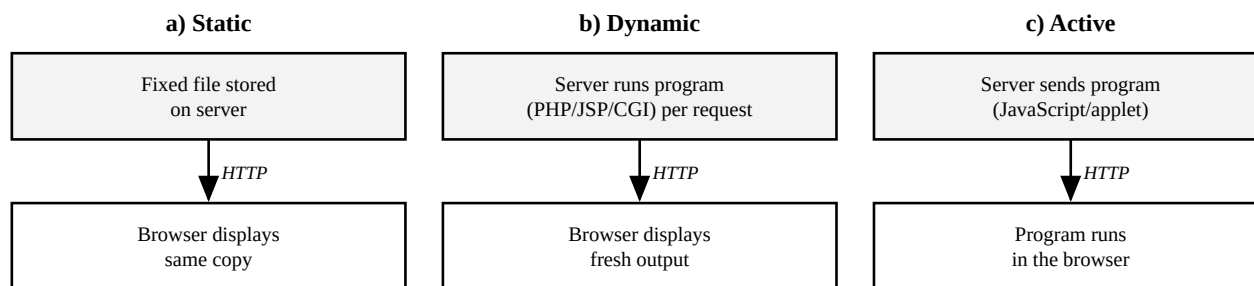


Fig.: Static, dynamic and active documents

(Each type with example 1 M, diagram 1 M)

c) Evolution 3G → 4G → 5G

4 M

Feature	3G	4G	5G
Introduced (approx.)	2001	2009 (LTE)	2019 (India: 2022)
Technology	UMTS / WCDMA, CDMA2000, HSPA	LTE, LTE-Advanced (OFDMA, MIMO)	5G NR (mmWave, massive MIMO, beamforming)
Peak data rate	about 2 Mbps (HSPA up to tens of Mbps)	100 Mbps – 1 Gbps	up to 10 – 20 Gbps
Latency	about 100 ms or more	about 30 – 50 ms	about 1 ms (target)
Core / switching	Circuit (voice) + packet (data)	All-IP packet core (EPC); voice as VoLTE	Cloud-native core with SDN / NFV, network slicing
Typical use	Mobile Internet, video call	HD video streaming, VoLTE	IoT, autonomous vehicles, AR / VR, URLLC

Each generation increased speed and capacity and reduced latency: 3G brought mobile data, 4G made the network all-IP, and 5G adds very low latency, massive device connectivity and network slicing.

(Any four correct comparison points, 1 M each)

d) NFV architecture

4 M

Network Function Virtualization (NFV) replaces dedicated hardware appliances (router, firewall, load balancer) by **software functions running on standard servers** using virtualization. Example: a telecom operator runs a virtual firewall and virtual EPC as VMs instead of buying separate boxes.

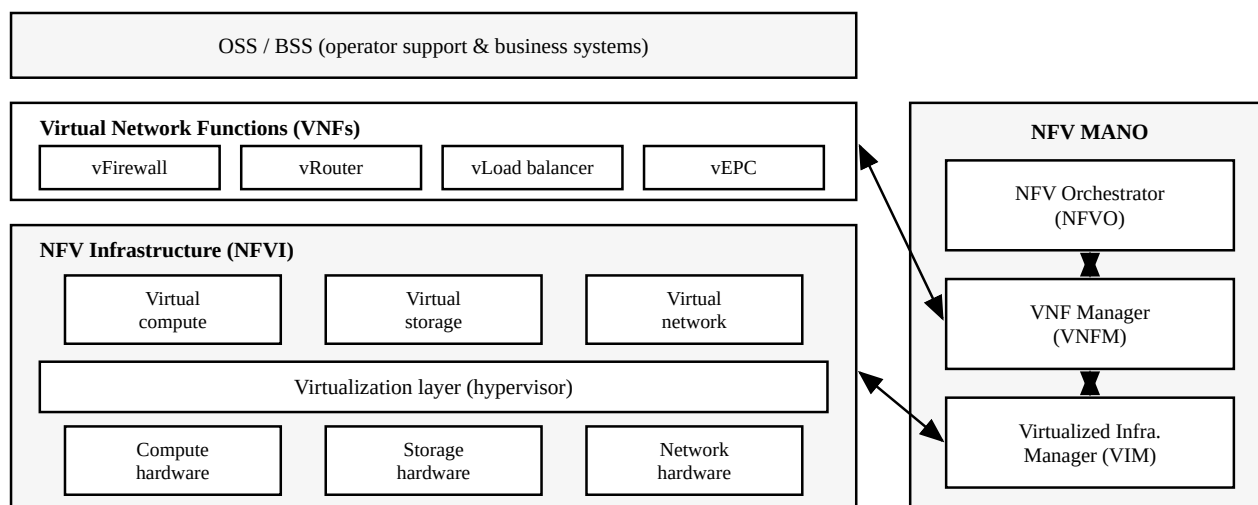


Fig.: NFV reference architecture (ETSI)

- **VNF (Virtual Network Function)**: software implementation of a network function running in a VM / container.
- **NFVI (NFV Infrastructure)**: physical compute, storage and network hardware plus the virtualization layer (hypervisor) that offers virtual resources.
- **NFV MANO**: *NFV Orchestrator* (creates and manages network services), *VNF Manager* (life-cycle of VNFs: start, scale, stop) and *VIM* (controls NFVI resources, e.g. OpenStack).

(Diagram 2 M, explanation 2 M)

e) PGP security parameters and algorithms

4 M

E-mail is a one-way, single-message transaction — sender and receiver do not create a session. Therefore, in PGP the sender must **include the security parameters in the message itself** so the receiver can process it:

- Identifiers of the algorithms used (hash, public-key, symmetric-key and compression).
- The one-time **session key**, encrypted with the receiver's public key.
- **Key IDs** of the public keys used, so the receiver can pick the right key from its key ring.
- Timestamp and the digital signature (encrypted message digest).

Algorithms used by PGP:

Purpose	Algorithms
Public key (signature, session-key encryption)	RSA, DSS, ElGamal
Symmetric key (message encryption)	IDEA, Triple DES, CAST-128
Hash (message digest)	MD5, SHA-1, RIPEMD-160
Compression	ZIP
E-mail compatibility	Radix-64 encoding

(Security parameters 2 M, algorithms 2 M)

Q.5

a) IPv4 header format and fields

6 M

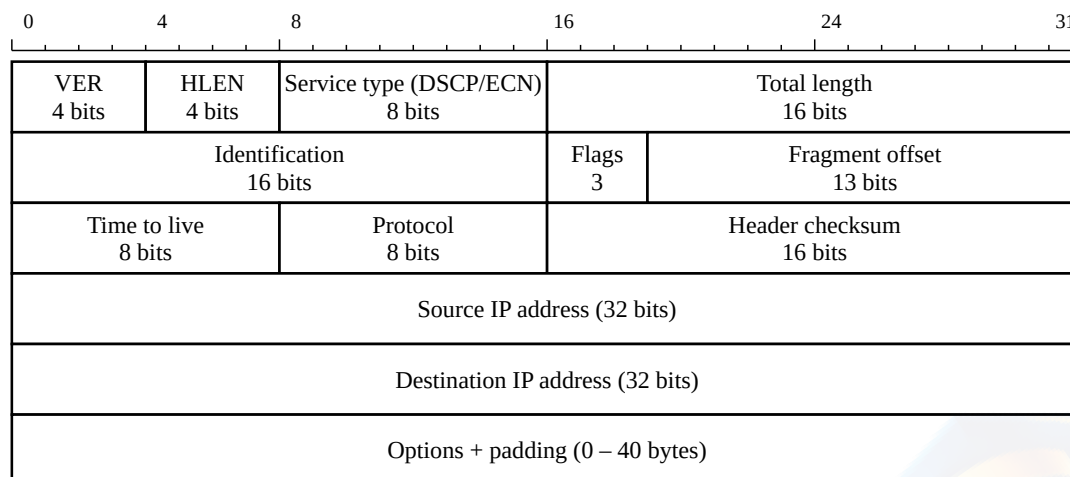


Fig.: IPv4 datagram header format

- VER (4 bits):** IP version, 4 for IPv4. **HLEN (4 bits):** header length in 4-byte words; 5 (20 bytes, no options) to 15 (60 bytes).
- Total length (16 bits):** length of the whole datagram (header + data) in bytes; maximum 65,535.
- Flags (3 bits):** reserved bit, **D** (do not fragment) and **M** (more fragments). **Fragment offset (13 bits):** position of this fragment's data in the original datagram, in units of 8 bytes. With Identification these allow reassembly.
- TTL (8 bits):** limits the lifetime (hops) of a datagram; every router decrements it, and when it becomes 0 the datagram is discarded and an ICMP time-exceeded message is sent — prevents looping.
- Protocol (8 bits):** higher-level protocol carried in the data: 1 = ICMP, 2 = IGMP, 6 = TCP, 17 = UDP, 89 = OSPF.
- Header checksum (16 bits):** error check on the header only; recalculated at every router because TTL changes.

Other fields: Service type / DSCP (priority, QoS), Identification (same for all fragments of a datagram), Source and destination IP addresses (32 bits each), Options (0 – 40 bytes).

(Diagram 2 M, fields 4 M)

b) Software Defined Networking (SDN)

6 M

SDN is a networking approach in which the **control plane** (decision making) is separated from the **data plane** (packet forwarding) and moved to a centralized, software-programmable **controller**.

i) Architecture:

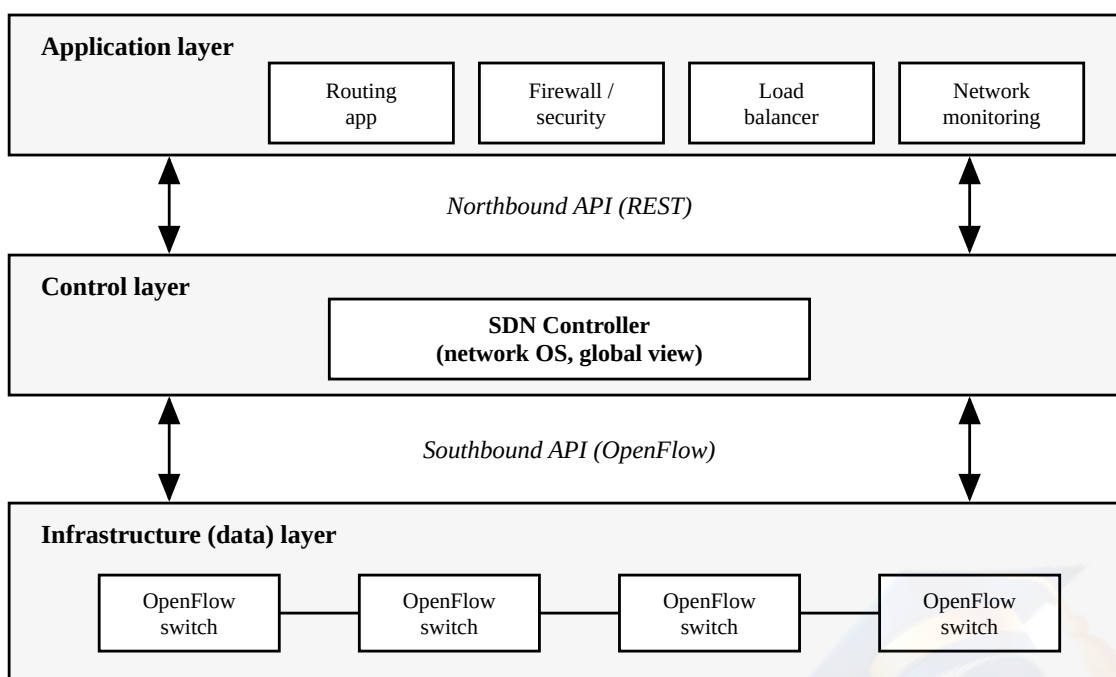


Fig.: SDN architecture — application, control and infrastructure layers

- **Application layer:** network applications (routing, firewall, load balancing, monitoring) that tell the controller what is needed.
- **Control layer:** SDN controller (e.g. OpenDaylight, ONOS) with a global view of the network; talks to applications through the **northbound API** (REST).
- **Infrastructure layer:** simple switches that forward packets according to flow tables; programmed by the controller through the **southbound API** (OpenFlow).

ii) Working:

1. A packet arrives at a switch; the switch matches it with its flow table.
2. If no entry matches, the switch sends the packet header to the controller (packet-in).
3. The controller decides the path using its global view and installs a flow rule in the switches (flow-mod).
4. Later packets of that flow are forwarded directly by the switches at full speed; applications can change the policy any time through the controller.

iii) Applications: data-centre and cloud networks, traffic engineering and load balancing, network security (firewall, DDoS mitigation), network virtualization for multi-tenant clouds, SD-WAN for branch offices, 5G network slicing, campus networks.

(Architecture with diagram 3 M, working 2 M, applications 1 M)

c) OSPF — link types, graphical representation, features

6 M

OSPF (Open Shortest Path First) is an intra-domain **link-state** routing protocol. Each router floods link-state advertisements (LSAs), builds the same link-state database and runs **Dijkstra's algorithm** to get a shortest-path tree.

i) Types of links: **Point-to-point** (two routers directly connected), **Transient** (a network such as a LAN with several routers attached; one designated router represents it), **Stub** (a network connected

to only one router), **Virtual** (created by the administrator when a link is broken, e.g. to reach the backbone area).

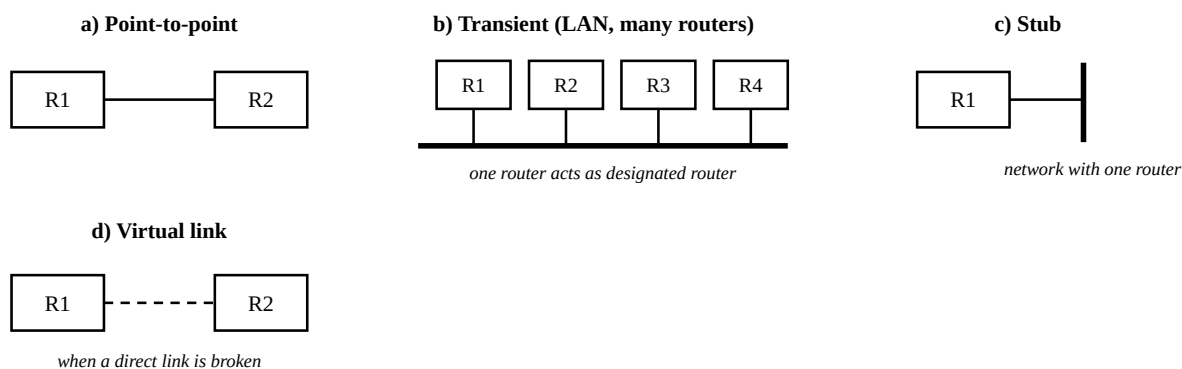
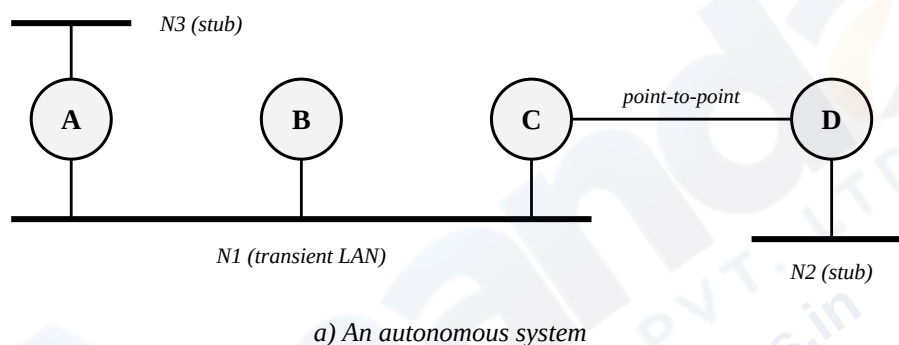
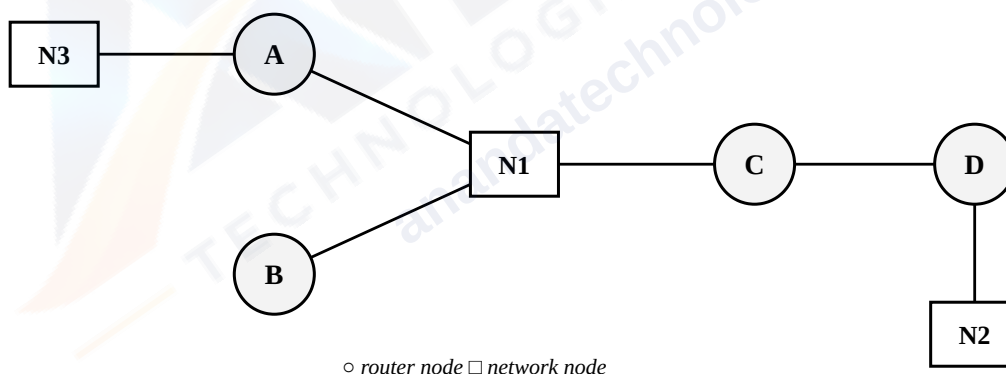


Fig.: Types of links in OSPF

ii) Graphical representation: OSPF represents an AS as a graph — routers and transient networks are nodes; point-to-point links are edges between routers; a stub network is a node connected to only one router.



b) Graphical representation used by OSPF

iii) Features:

- Divides an AS into **areas**; all areas connect to backbone area 0 — reduces flooding.
- Metric is a **cost** (based on bandwidth, delay) set by the administrator, not hop count; no 15-hop limit.
- Fast convergence: LSAs sent only when a change occurs (plus every 30 min refresh); loop-free routes.

- Supports VLSM / CIDR, authentication, equal-cost multipath; runs directly over IP (protocol 89) using multicast 224.0.0.5.

(Link types 2 M, graphical representation 2 M, features 2 M)

Q.6

a) Subnetting 200.15.30.0/24 into 4 subnets

6 M

i) **Class and default mask:** first octet 200 = 11001000_2 (starts with 110) → **Class C**; default mask 255.255.255.0 (/24).

ii) **Bits borrowed:** $2^n \geq 4 \rightarrow n = 2$. New prefix = $24 + 2 = /26$; last octet of mask = $11000000_2 \rightarrow$ subnet mask = 255.255.255.192.

iii) **Hosts per subnet:** host bits = $32 - 26 = 6$; addresses per subnet = $2^6 = 64$; usable hosts = $64 - 2 = 62$.

iv) **Subnet details** (block size $256 - 192 = 64$):

Subnet	Network address	First host	Last host	Broadcast address
1	200.15.30.0/26	200.15.30.1	200.15.30.62	200.15.30.63
2	200.15.30.64/26	200.15.30.65	200.15.30.126	200.15.30.127
3	200.15.30.128/26	200.15.30.129	200.15.30.190	200.15.30.191
4	200.15.30.192/26	200.15.30.193	200.15.30.254	200.15.30.255

(Class & mask 1 M, borrowed bits & new mask 1 M, hosts 1 M, table 3 M)

b) ICMP error-reporting messages

6 M

ICMP (Internet Control Message Protocol) is a companion of IP. Because IP has no error-reporting mechanism, ICMP reports errors **back to the source** of the datagram. ICMP messages are carried inside IP datagrams (protocol = 1). ICMP only reports errors; it does not correct them.

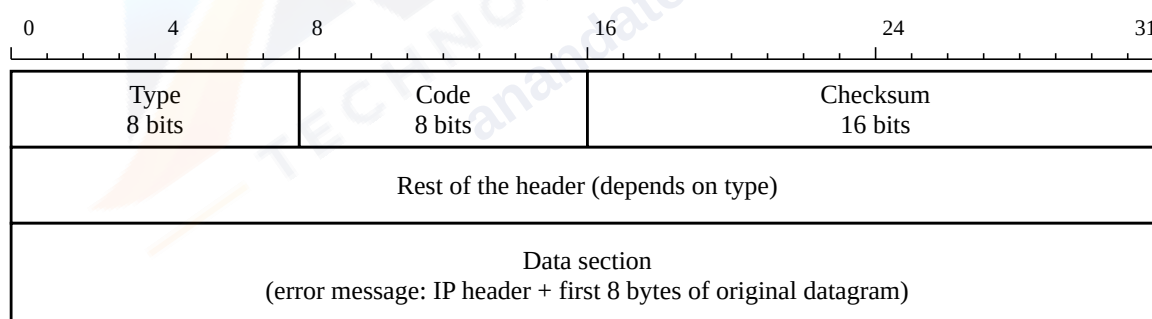


Fig.: General format of ICMP message

i) Error-reporting message types:

Type	Message	Purpose
3	Destination unreachable	Router/host cannot deliver the datagram (network, host, port or protocol unreachable); datagram discarded.
4	Source quench	Router is congested and discarded the datagram; asks source to slow down (now deprecated).

11	Time exceeded	TTL became 0 at a router, or all fragments not received in time at the destination.
12	Parameter problem	Ambiguous or missing value in a header field.
5	Redirection	Router tells a host that a better next-hop router exists for that destination.

No ICMP error message is generated for an ICMP error message, for non-first fragments, for multicast datagrams, or for special addresses such as 127.0.0.0 or 0.0.0.0. The error message carries the IP header and first 8 bytes of the original datagram.

ii) Example: Host A (10.1.1.5) sends a datagram to 30.1.1.9 with TTL = 1. Router R1 decrements TTL to 0, discards it and sends **Time exceeded** (type 11) to 10.1.1.5 — this is how *traceroute* discovers routers. A datagram to 30.1.1.99 (no such host) is returned as **Destination unreachable** (type 3).

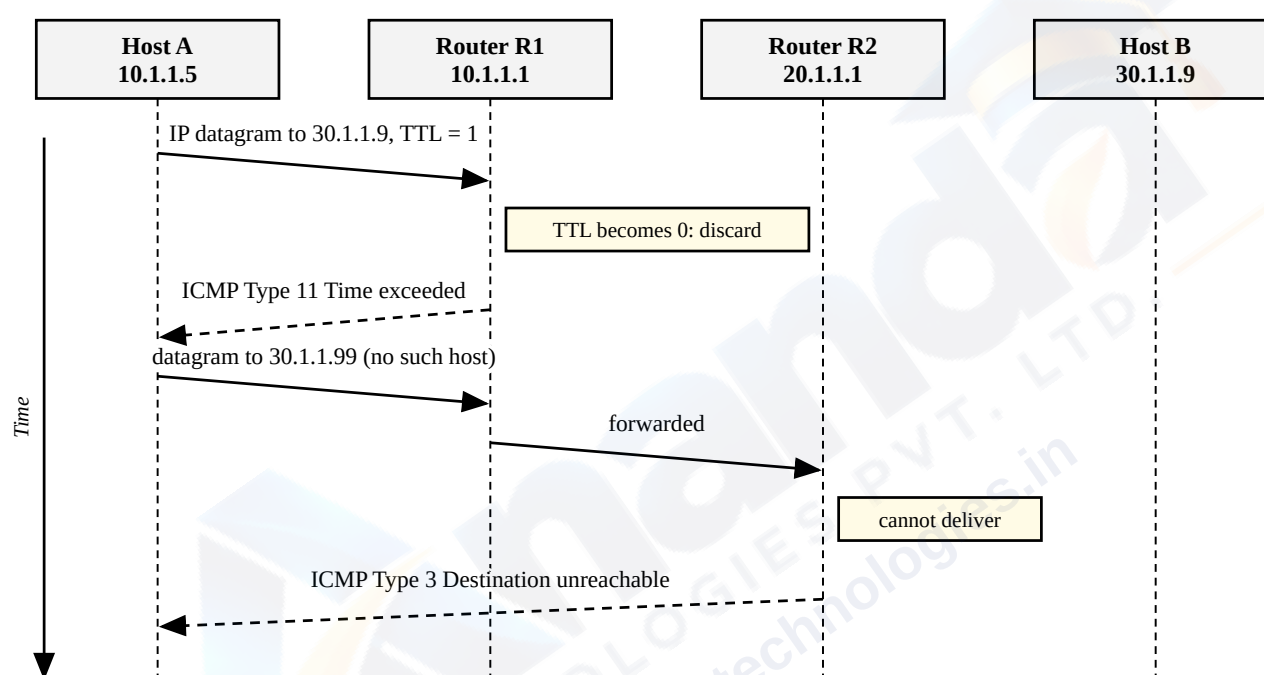


Fig.: ICMP error reporting — time exceeded and destination unreachable

(Types 3 M, example with diagram 3 M)

c) FTP — connections, transmission modes, anonymous FTP

6 M

FTP (File Transfer Protocol) is the standard TCP/IP application for copying files between two hosts. It uses **two TCP connections**.

i) Control and data connections:

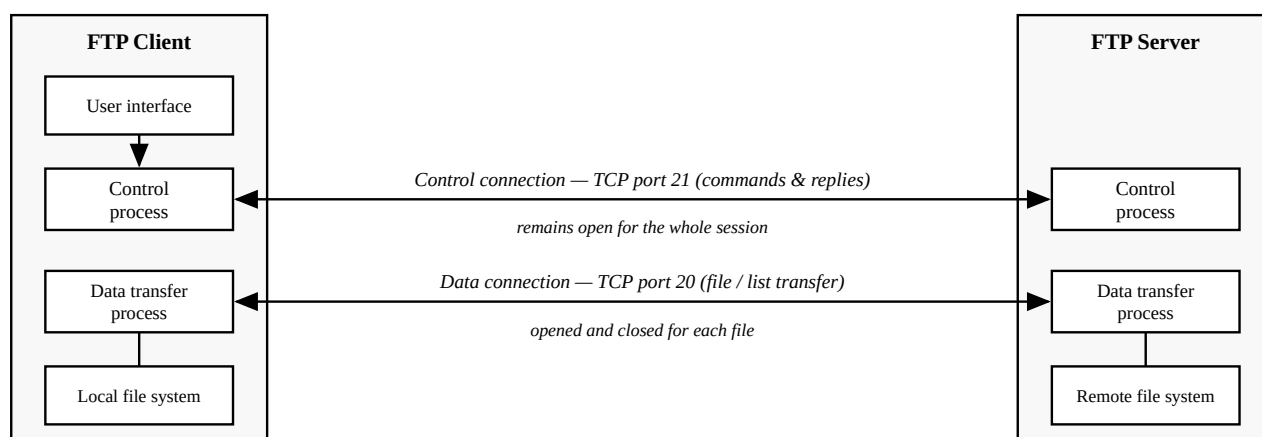


Fig.: FTP model — two TCP connections: control (port 21) and data (port 20)

- **Control connection (port 21):** opened first and kept open for the whole session; carries commands (USER, PASS, RETR, STOR, LIST) and 3-digit replies one line at a time.
- **Data connection (port 20):** opened when a file or directory list must be transferred and closed after each transfer.
- Before transfer, client and server agree on **file type** (ASCII, EBCDIC, image/binary) and **data structure** (file, record, page).

ii) Transmission modes:

- **Stream mode** (default): data is sent as a continuous stream of bytes; end of file is shown by closing the data connection.
- **Block mode:** data is sent in blocks, each with a 3-byte header (1-byte descriptor + 2-byte size).
- **Compressed mode:** repeated bytes are compressed with run-length encoding to save bandwidth.

iii) Anonymous FTP: lets any user access public files on a server without an account; user name is *anonymous* and the password is usually the e-mail address. Access is restricted to a public directory and normally only download (GET) is allowed. Used for distributing free software, drivers and documents.

(Connections with diagram 2 M, modes 2 M, anonymous FTP 2 M)

Note: This is a **model question paper prepared by Ananda Technologies** only for practice and exam preparation, based on the pattern of previous MSBTE question papers and the K-scheme syllabus. It is **not an official MSBTE paper**, and it does **not** mean that these questions will come in the exam. Answers follow standard textbook points; marks shown are indicative. Free for students — anandatechnologies.in