NETWORK LAYER (UNIT-04)

* Store-and-Forwarding Packet Switching:

→ recieve messages in full and send to the next node.

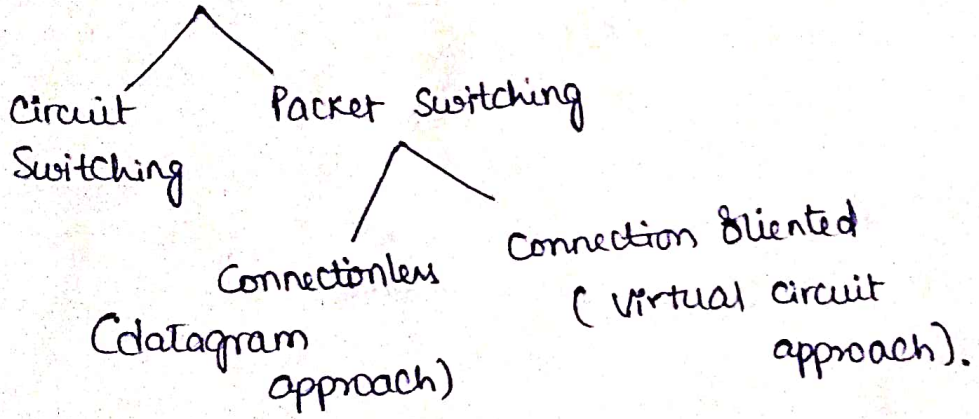
* Cut through:

→ Send the msg to next node before completly receiving the data.

Services Provided to Transport Layer:

→ Services are independent of router technology.
→ Shielded from the number, type and topology of routers present.
→ network address made available use a uniform numbering plan.

Packet Switching: Proces of forwarding data.



Circuit Switching: concept of wires are used.

→ Path is predefined. S ——— D

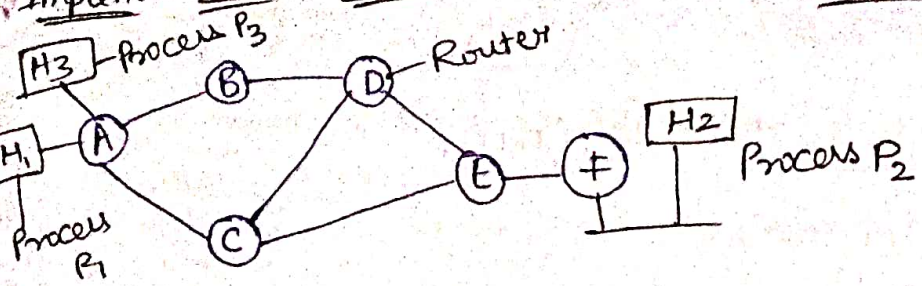
Connectionless: (Datagram).

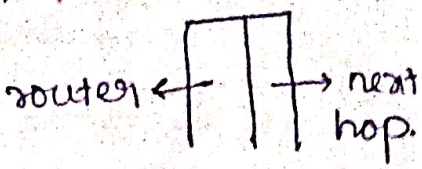
→ NO concept of wires.
→ There are multiple paths.
→ Path is Choosen Randomly.
→ Every Packet is sent Independently.

Connection Oriented (virtual):

→ NO concept of wires
→ Predefined path.

Implementation of Connectionless Service (Datagram):





A's table

A	-
B	B
C	C
D	B
E	C
F	C

Connection oriented:

H_4	1
H_3	1

C	1
C	2

A's table.

1: H_1 taken as
2: H_3 Identifier

NETWORK LAYER SERVICES:

→ Packetizing (Payload – data recieved from upper layer).

→ Routing
↓
best path from source to destination

→ Forwarding
↓
action applied by each router when Packet arrives at one of its interface using routing.

Performance:

→ Delay → Throughput → Packet loss.

Transmission delay – Time taken to send complete data over network.

Propagation delay – time taken from source to des to recieve data.

Processing delay – time required to Process a Packet in a router or a destination host.

Queuing delay – The time a Packet wait in input & output queues.

TOTAL DELAY:

$$\text{total delay} = (n+1)(\text{delay}_{tr} + \text{Delay}_{Pg} + \text{Delay}_{Pr}) + (n)$$

n: no. of

Delay tr = (Packet length) / transmission rate Delay Pg = Distance / Propagation Speed.

n = routes & 1 = destination in (n+1).

Procesing delay = (n+1) Pr delay.

Pacet loss:

⇒ Buffer have limited size.
  L Storage space for a Packet in queue.

⇒ When buffer is full, the next Packet needs to be dropped.

⇒ Then the Packet needs to be resent.

IPV4 Address:

⇒ Size of IP address = 32 bits.

⇒ represented in { decimal ⇒ Each octet is represented in 8 bits. (only 4 octets)
                  { Binary

⇒ In Classful addressing, the address is divided into five Classes: A, B, C and d, e.

decimal notation	Binary	Subnet mask
0-127	0	— Class A — 8 network bits - 24 ho
128-191	10	— Class B — 16 network — 16 host
192-223	110	— Class C - 24 network - 8 host.
224-239	1110	— Class D
240-255	1111	— Class E

(Classful addressing)

Yellow part defines - network id - represented by '1'.

White part defines host id - represented by '0'.

Class	Start	Finish
A	10.0.0.0	10.255.255.255.
B	172.16.0.0	172.31.255.255
C	192.168.0.0	192.168.255.255

⇒ maximum no.of data in class A = 2^n

Network = 8 bits = 1 bit fixed = 7 bits

No.of data = $2^7 = 128$.

⇒ A,B,C are uni-cast / D-multicast / E - Reserved.

* In Classful addressing, a large part of the available address were wasted.

Classless IP address:

→ According to Requirement Ip address is given.

CIDR - Classless Inter domain Range.

→ Classfull is replaced by classless.

*

In IPV4 addressing, a block of address can be defined as

x.y.z.t/n → defines the mask.

↳ address

* Formula for first address is found by,

Setting right most 32-n bits to 0's.

First address (or) Network address.

* For last address

Setting right most 32-n bits to 1's.

last address (or) broadcast address.

* Number of address in the block can be found by using the formula (Hosts) 2^{32-n}

⇒ The total usable IP's in 16 are 14

Since 1st & last will be not used.

Another way to find the first and last

Eg:

FLSM: - fixed length Subnet Mask.

→ 5 Networks & 32 hosts. (Equal no. of hosts).

↳ max = 8.

VLSM: Variable length Subnet Mask.

→ different no. of hosts.

1
2
4
8
16
32
64
128
256
512

Eg:

190.100.0.0/16

32-n ⇒ 32-16 = 16 ⇒ 255.255.0.0.

(a) 256 ⇒ 2^8 = 256. (8 → host bits.)

11111111.11111111.00000000.00000000. → 255.255.00

For 8 host bits, we update.

11111111.11111111.11111111.00000000.

255.255.255.0.

⇒ If the third octet is changed, we vary the range in 3 octet only.

i.e. first address ↔

190.100.0.0/24 → Cannot be assigned as it is

190.100.1.0/24

⋮

190.100.63.0/24

190.100.0.255/24

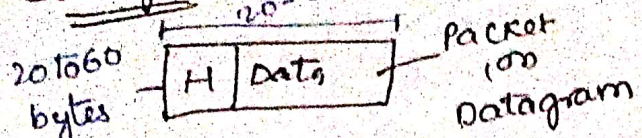
190.100.1.255/24

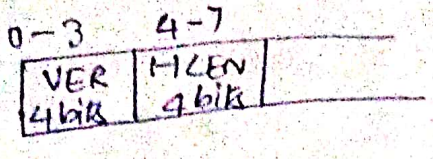
⋮

190.100.63.255/24

0.1-254 → host address

Datagram: 20 - 65,535 bytes





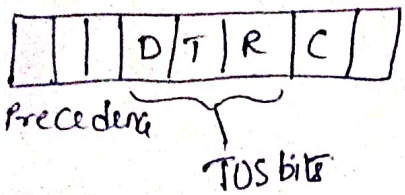
→ If the HLEN have less than 4 bytes that packet is not accepted

→ HLEN should vary from 5-15

4 types of Services:

D: delay, R: Reliable

T: Throughput, C: minimize Cost.

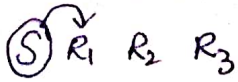


Total length:

length of the datagram + Header.

TTL: Time to Live: 8 bits.

Packet is transferred by how many intermediate hosts.



If TTL = 6

If S sends one packet to R1

TTL = 5

⇒ If TTL becomes zero in the middle then the packet will be dropped.

⇒ If the router is failed then the Packet is sent to source in backward and loop will be created.

Eg: 01000010

⇒ After 64 bits ttl field comes. ⇒ $\frac{64}{4} = 16$

⇒ Predifined code for every protocol.

1 - ICMP	17 - UDP
2 - IGMP	
6 - TCP	89 - OSPF

Identification: Allows the destination host to determine which datagram a newly arrived fragments belongs to.

Fragmentation:

→ If a packet is of 200 bytes & network have 100 bytes. Then the 200 byte packet is framented / divided inorder to meet the size of the Network.

⇒ They will have the same Identification number.

MTU - Capacity of my channel - Maximum transfer unit

**

* Only the data in datagram is fragmented not the header.

**:

If the Packet is divided or fragmented then the fragmented Packets will also have the same identification number as of the original Packet. So, the destination can identify them as fragments of which data.

Fragmentation offset:

Formula ⇒ offset = 0000/8 (or) first by 8.

↳ $\frac{\text{First byte}}{8}$ = fragmentation offset.

0, 175, 350 are fragmentation offset values

⇒ tells the sequence of the fragmented bits.

FLAG:

⇒ 3 bits

X	D	M

D: Donot fragment

M: more fragment.

→ The first bit is reserved.

D:

→ If 'D' bit is 1 then no fragments

0 then there are fragments.

M:

→ If 'M' bit is 1 there will be one more fragment after 1

0 Then there will be no more fragments its the end.

200×8 =

1600 = F.B

8×4 = 32

1600 = 32

200 = 320 + Data

200 / 32 = 268

200 / 32 = 168

1600

267

1867

* OPTIONS AND PADDING: System testing and debugging (0 To 40 bytes)

OPTIONS:

→ not required for datagram.

→ option Processing is required of the IP software.

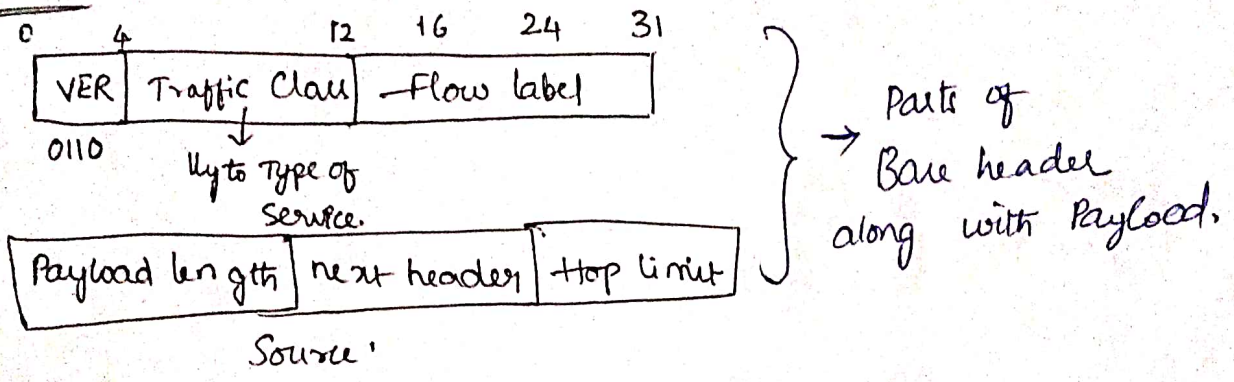
Strict Source route : route is Pre defined

Time Stamp: Tells the time taken by routers to Process the Packet.

→ To check efficiency of routers.

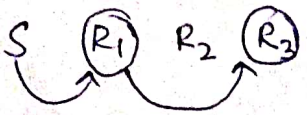
CHECK SUM :

IPV6 Header:



*** Flow Labels:**

→ Routers maintain a flow label



⇒ It is labelled as 1 and 3 So packets move from R_1 and R_3.

*** Payload :**

→ will have only data there will be no header.

*** Hop limit :**

If TTL / HL = 6, It Tells the no. of intermediate hops.

→ Passes through 6 routers.

* In IPv6 there will be two headers,
  → Base header (40 bytes) and → Extension header (optional).
  → Fixed
* So there will be no concept of HLEN.

* If HLEN = 2 then it cannot be determined since the HLEN varies from 20 to 60. So, HLEN = 5-15 vary.

Extension Header:

→ Next header acts as a pointer of the next address.
→ Next header will be shown only after fragmentation is done.

Padding:

Jumbo Payload: If the limit of data part 65,535, we add the additional data to Jumbo Payload field.

Authentication: Checking whether the user is authorized or not.

Encrypted security Payload:
→ using code of encryption (50) to the next header.

Transition from IPV4 to IPV6:

① Dual Stack: Machine able to support IPV4 and IPV6.

② Tunneling: Both the source & destination supports IPV6. But the routers supports IPV4.
  → IPV6 will be encapsulated into IPV4.

③ Header Translation strategy:
→ source of IPV6 and destination by IPV4. The Header of IPV6 will be changed to IPV4 (Header translation)

IPV6:

⇒ IPV6 address is 128 bits long.

→ represented by 8 octets. and in hexadecimal.

f : 4
D : 4
E : 4
C : 4

Original:

→ Where there are 4 zero's convert it to one zero.

→ Preceded zero's exclude these zero's.

or: FDEC : 0074 : 0000 : 0000 : 0000 : BOFF : 0000 : FFFO

ab: FDEC : 74 : 0 : 0 : 0 : BOFF : 0 : FFO .

⇒ Continuous zero's are ~~repreplaced~~ replaced by two colons '::'.

more ab : FDEC : 74 : : BOFF : 0 : FFFO.

Eg: 0:15::1:12:1213

⇒ 0000 : 0015 : 0000 : 0000 : 0000 : 0001 : 0012 : 1213

⇝ 0000 : 0015 : 0000 : 0000 : 0000 : 0001 : 0012 : 1213,

UNIT-05 : ROUTING

→ Routing is used to decide the best path to transfer data.
→ It can be static or dynamic.
→ Static : configuring manually
→ dynamic : maintained by the router.

Autonomous System :

→ Every network will have the controlling entity. and that entity shares information from one network to other.
→ When it recieves a packet, the next path to share the data is based on optimization.

Metrics to choose a path :

→ One approach is to assign a cost for paving through a network. Called Cost a metric.
→ RIP - Routing Information Protocol - hop count based.

Routing Protocols

→ Intradomain - Packet sent within the network. < Distance vector (RIP) / Link state (OSPF)
→ Inter domain - other Networks.
  / Path vector (BGP)

Distance vector : the least-cost route b/w any two nodes is the route with min. distance.

A's table

		Next hop
A	0	-
B	5	-
C	2	-
D	3	-
E	∞	

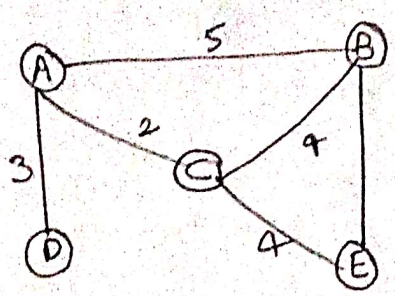


fig: 1

B's table

A	5	-
B	0	-
C	4	-
D	∞	
E	3	-

C's table

A	2	-
B	4	-
C	0	-
D	∞	
E	4	-

D's table

A	3	-
B	∞	
C	∞	
D	0	-
E	∞	

⇒ Each node shares its routing table with its immediate neighbours periodically & when there is a change. - distance vector routing. } Periodic update

⇒ Even when the failure is done / changes → triggered update.

⇒ To maintain consistency it is done.

⇒ In fig: 1 the table A is going to share with 'C' since it has the least distance. and the first two columns are shared.

→ 'C' will update it's table according to 'A'.

C

A	2	+2
B	4	+2
C	0	+2
D	∞	+2
E	4	+2

A from 'c'.

A	4
B	6
C	2
D	∞
E	6

⇒

A's new table after comparing with old 'A'.

A's

A	0	-
B	5	-
C	2	-
D	3	-
E	6	C

* Problems in Distance Vector Routing:

⇒ The table who lost the link doesnot share its table with other but other table shares it - Two-node instability and the loop will be created till ∞.

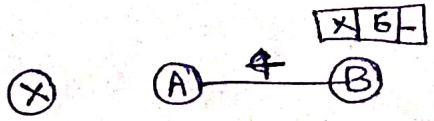
Solution:

Case-i ⇒ redefine infinity value to smaller value. (16 Predefined)

Case-ii ⇒ Split Horizon: 'B' shouldn't send the same information to 'A' again because 'A' is already updated according to 'B'.

⇒ If there is no answer from neighbour node then we assume that node is failed. (ii)

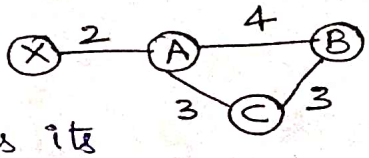
for (i)



Sol: Poison Reverse:

⇒ 'B' node sends the information but changes its table to '∞'.

Three-node-instability:



⇒ If the 'x-A' link is failed 'B' updates its table but 'C' doesn't. So, 'B' shares its table with 'C' as it thinks 'C' is the another path to reach 'x'.

⇒ ROUTING INFORMATION PROTOCOL (RIP):

⇒ metric will be hop count based. It should be min.

→ If HC=15, routers are 15.

⇒ Link State Routing:

→ each node in domain has the entire topology of the domain. LSP - Link state Packet

Build Routing table:

→ Every node creates LSP Packets and into of sending node.

→ Sending Packet to every node called flooding, efficient & reliable way

→ use shortest Path

→ calculate routing table.

- node identity - list of link - sequence number (for every LSP) and info about latest LSP and greatest LSP value gives the latest LSP) - age (every node germinate so many LSP, so we add age factor like 10min and after 10min, it will be discarded)
→ Step-01.

LSP's are generated when -
→ change in topology → on periodic basis.

OSPF - Open Shortest Path First → It is an Intradomain routing protocol based on Link State routing.

Metric in OSPF:
→ allows administrator to assign a cost called metric.
→ It can be based on type of service - minimum delay & max throughput & so on.
→ can have multiple routing-tables based on diff type of service.

* Point-Point : two router with point connection.
* Transient : Bus topology
* unrelistic : mesh topology

PATH VECTOR ROUTING:
→ There is atleast one node called speaker node.
* Initialization * Sharing * updating. AS - Autonomous system

BGP (Border Gateway Protocol): (Path vector Routing)
→ appeared in 1989 & have 4 versions.

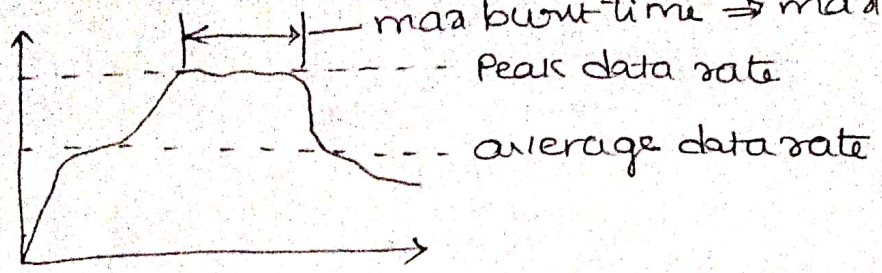
Path attributes:
well known: * Mandatory * Discretionary (well known but not mandatory)
Eg: Header
optional: * Transitive * non-Transitive
↓ ↓
Changes are performed to all router → Irreversible

BGP sessions:

→ reliable connections are made.

CONGESTION CONTROL:

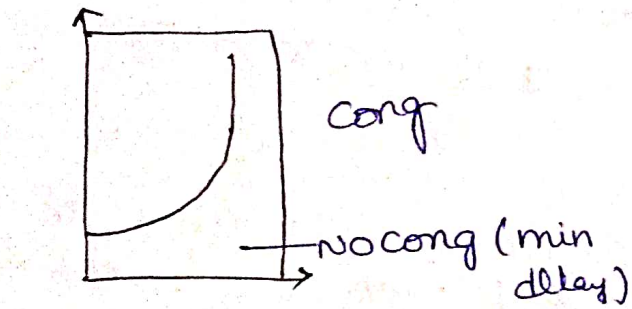
* Data Traffic:



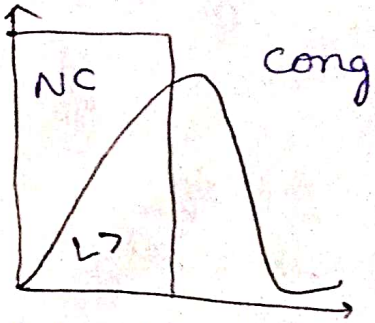
Congestion: It may occur if the load of on the network is more than its capacity.

→ Every router maintain a queue.

Network Performance:



Delay as fnc of load



~~Goto~~ through as a fnc of load.

* open-loop congestion control (Prevention).
* closed-loop congestion control (removal).

* Retransmission Policy:

When to transmit

→ lost frame

→ lost acknowledgement

→ Damaged frame.

DISCARDING POLICY:

* In case of congestion
  → Low priority
  → Newer Packet
  → Discard which is not near to destination.

Closed loop CC:

→ Backpressure → Choke packet → Implicit → Explicit Backpressure

BACK PRESSUREE:

⇒ If congestion is detected by a node & the msg is sent to the Previous node and source slows docon its speed.

Choke Packet:

⇒ The msg of detection of congestion at node is directly sent to source.

Implicit Signaling:

⇒ Source coll be aware about congestion in network.
  * No communication is sent to source from the node.

Explicit Signaling:

Network alerts sender or destination to slow docon the rate of transmission by sending signal that is included in Packets that carry data.

* Backward — warning to Source forward — warning to destination.

Quality of Servic (QOS):

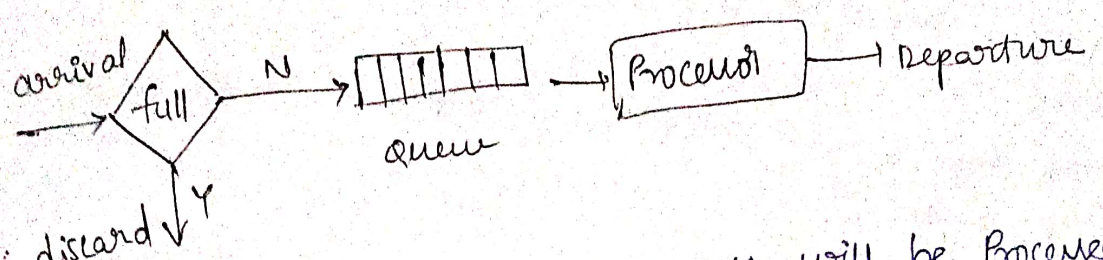
→ Handle any amount of traffic
→ over-provisioning
* send max no.of data even if the capacity is less.

Techniques to improve QoS:

*Resource Resolution:

→ Reserving a source, so there will be no interruption.

FIFO queue:



→ The 1st packet that enters the queue will be Processed first.

⇒ Priority queue:

* The Higher Priority Packets are Processed before Lower-Priority Packets.

* Starvation: Lower Priority Packets do not get the Chance to be Processed.

⇒ It can be removed by weighted fair queue.

* We pick 2-3 Packets comparing the weight, so the lower Priority Packets gets chance for Processing.

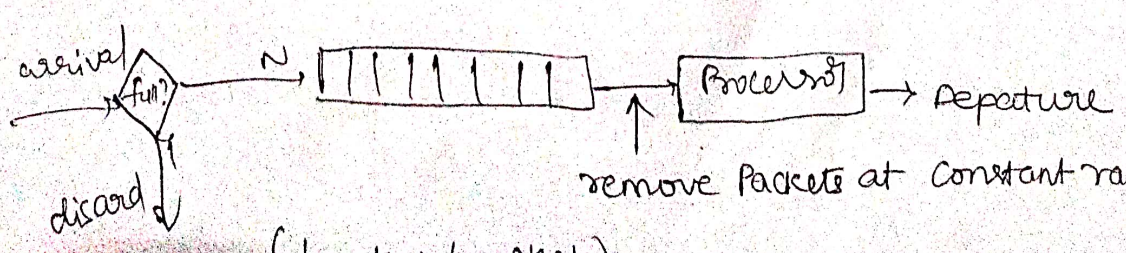
Traffic Shaping:

→ Technique for controlling the average rate and burstiness of a flow of data that enters the network.

*reduces the congestion.

*Monitoring of traffic flow is traffic Policing

* Leaky Bucket ⇒ converting Bursty-flow to fixed flow.



(Leaky bucket)

* Leaky bucket can drop the Packets if the bucket is full.

→ Token Packet

→ Based on the ~~Packet~~ token, Packet will be send.

* Packets can be sent at a time.

→ In Leaky bucket the data is sent at constant rate but in token bucket it isn't constant and Packets are sent at a time.

* The token bucket allows bursty traffic at a regulated maximum rate.

UNIT-06 TRANSPORT &

* Process - Process delivery.
→ local host - Client
→ Remote host - Server

Size of data = 2^{16} / 16 bits

IANA - Internet Assigned Number Authority.

* well known ports - 0 to 1023 - assigned
* Registered Ports - 1024 to 49,151 - not assigned
* Dynamic Port - 49,152 to 65,535 - not assigned, not registered
(~~Eff~~ Ephemeral Port)

* well-known ports are used by servers mostly.
Port number - Selects the Process.

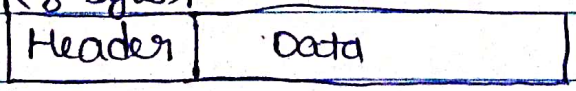
⇒ Socket Address : Combination of IP address & Port number.

Connectionless Service: UDP

disad: Packet loss, Data is delayed, no acknowledgement

UDP (User datagram Protocol):

* Connectionless



Source port 16 bits	destination port 16 bits
total length 16 bits	checksum 16 bits

* UDP Length = IP Length - IP header's length.

UDP Packet = Header + data.

Transport → H + Data

↓

Network → H + UDP.

Eg: If TL = 64, HLEN = 6; UDP Packet length & UDP data Portions are

Sol: TL = 64

HLEN = 6 × 4 = 24

UDP Packet = 64 − 24 ⇒ 40

∴ UDP data = 40 − 8 = 32.

UDP operation:

→ No error control except Checksum
→ data is called as datagram.
→ Encapsulation & Decapsulations.

QUEUING:

→ Incoming & outgoing queue
→ obtain only one Port number.

Uses of UDP:

→ Suitable for multicasting.
→ used for management Process such as SNMP
SNMP – Simple network management Protocol.
→ used for routing update Protocol.

→ TCP :

* Connection Oriented. * data will be called as segments.
* Reliable * Sequence number size } ⇒ 32 bits.

→ Numbering System.

* NO Segment number — we byte number - Sequence number, acknowledgement number

→ $0 - 2^{32} - 1$ * Flow & error control * Congestion.

→ 1st Sequence number tells the first byte in that segment
Acknowledgement - tells about the next expected frame.
Cummulative Ack - one Acknowledgement will be given for the whole data sent.
Cd - Congestion will be less.

TCP header :

Source 16 bits		dest 16 bits
Sequence 32 bits		
Acknowledgement 32 bits		
HLEN 4 bits	reserved 6 bits	Type of segment 6 bits / window size 16 bits
Check Sum 16 bits		urgent Pointer 16 bits
options & Padding		

Type of segment - 6 bits.

URG ACK PSH RST SYN FIN

URG - urgent pointer
- If sending urgent info (failure) - 1

ACK - Valid Acknowledgement - 1

Push - Request is made - 1 Segment having data to send.

RST - Reset Connection - 1 to Reset

SYN - Connection b/w S and R - 1

FIN - To terminate Connection - 1

Urgent Pointer - How many bytes are sent as a urgent data.

* will be saved in the beginning.
* Point to last sequence number of data which carry the Urgent data.

* Three-way Handshaking :

* A SYN Segment cannot carry any data but still have one Sequence Number.

* An ACK Segment, if no data is carried, sequence number will be '0'.

* A SYN+ACK will not carry any data, but still consumes a sequence number. (1 sequence number)

→ Connection establishment. (10 marks) * → TCP / UDP

* Denial of Service : (DOS)

Syn flooding attack:

* Simultaneous open → Cand s send request on same time for connection establishment.

→ FIN Segment consumes one sequence number if it is not carrying data.

Electronic mail :

→ With user agent we are able to compose the mail.

* When the sender & reciever of an email are on the same system ⟹ First Scenario.

Second Scenario :

⇒ It will have 2 user agents.

⇒ And MTA client & MTA user are taken as 1 pair.

Third Scenario :

⇒ 2 user agents and 2 MTA pairs.

UA : user agent

MTA : Message transfer agent

MAA : Message access agent.

fourth Scenario :

→ MAA is used to pull the mail.

→ MTA is used to push the mail.

→ Two UA's, Two MTA's and a pair of MAA.

⇒ command-driven user agents are mail, Pine and elm.

⇒ Gui-based agents are - Eudora, outlook and netscape.

→ E-mail address:

→ Contains Local part @ Domain name
⇓
gmail.com.

Name Space:
DNS - domain Name space. System (contains domain to IP address.
Flat → complete address will not be provided.
Hierarical → complete address will be provided.

SMTP - Simple Mail Transfer Protocol

disadvantage of flat Name space:

→ Can't be used in a large system such as the Internet.

→ Fully qualified Domain Name (FQDN)
* complete domain name will be written.