COMPUTER NETWORKS - II

Subject Code: 10CS64 **I.A. Marks : 25**
Hours/Week : 04 **Exam Hours: 03**
Total Hours : 52 **Exam Marks: 100**

PART - A

UNIT – 1 **6 Hours**
Packet Switching Networks - 1: Network services and internal network operation, Packet network topology, Routing in Packet networks, Shortest path routing: Bellman-Ford algorithm.

UNIT – 2 **6 Hours**
Packet Switching Networks – 2: Shortest path routing (continued), Traffic management at the Packet level, Traffic management at Flow level, Traffic management at flow aggregate level.

UNIT – 3 **7 Hours**
TCP/IP-1: TCP/IP architecture, The Internet Protocol, IPv6, UDP.

UNIT – 4 **7 Hours**
TCP/IP-2: TCP, Internet Routing Protocols, Multicast Routing, DHCP, NATand Mobile IP.

PART – B

UNIT – 5 **6 Hours**
Applications, Network Management, Network Security: Application layer overview, Domain Name System (DNS), Remote Login Protocols, E-mail, File Transfer and FTP, World Wide Web and HTTP, Network management, Overview of network security, Overview of security methods, Secret-key encryption protocols, Public-key encryption protocols, Authentication, Authentication and digital signature, Firewalls.

UNIT – 6 **7 Hours**
QoS, VPNs, Tunneling, Overlay Networks: Overview of QoS, Integrated Services QoS, Differentiated services QoS, Virtual Private Networks, MPLS, Overlay networks.

UNIT – 7 **7 Hours**
Multimedia Networking: Overview of data compression, Digital voice and compression, JPEG, MPEG, Limits of compression with loss, Compression methods without loss, Overview of IP Telephony, VoIP signaling protocols, Real-Time Media Transport Protocols, Stream control Transmission Protocol (SCTP)

UNIT – 8 **6 Hours**
Mobile Ad-Hoc Networks, Wireless sensor Networks: Overview of wireless adhoc networks; Routing in adhoc networks; Routing protocols for adhoc networks; security of adhoc networks. Sensor networks and protocol structures; Communication energy model; Clustering protocols; Routing protocols; Zigbee technology and IEEE 802.15.4

Text Books:

1. Alberto Leon-Garcia and Indra Widjaja: Communication Networks – Fundamental Concepts and Key architectures, 2nd Edition, Tata McGraw-Hill, 2004.
   (Chapters 7, 8, 9, 11, Appendix B)
2. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007.
   (Chapters 12, 16, 17, 18, 19, 20)

TABLE OF CONTENTS

UNIT 1: PACKET SWITCHING NETWORKS

NETWORK SERVICES AND INTERNAL OPERATION

- Transport-layer peer processes
  - → accept messages from their higher layer &
  - → transfer messages by exchanging segments end-to-end across the network (Figure 7.2).
- The network service can be connection-oriented or connectionless.

Connectionless Service

- A connectionless service uses only 2 basic interactions b/w transport-layer & network-layer:
  1) A request(REQ) to the network-layer that it send a packet &
  2) An indication(ACK) from the network-layer that a packet has arrived.
- The user can request transmission of a packet at any time.
- The user does not need to inform network-layer that the user intends to transmit information ahead-of-time.
- A connectionless services puts total responsibility for error-control, sequencing and flow control. on the end-system transport-layer.

Connection-oriented Service

- Transport-layer cannot request transmission of information until a connection between the end systems has been set up.
- Network-layer must be informed about the new flow that is about to be sent to the network.
- During connection setup,
  - → parameters related to usage & quality-of-service(QoS) may be negotiated &
  - → network resources may be allocated.
- A connection-release procedure may also be required to terminate the connection.

- Network-layer can provide following services to the user of network:
  1) Best-effort connectionless service
  2) Low-delay connectionless service
  3) Connection-oriented reliable stream service
  4) Connection-oriented transfer of packets with delay & bandwidth guarantees
- Two interrelated reasons for keeping the set of network services to minimum:
  1) End-to-end argument &
  2) Need for network scalability.

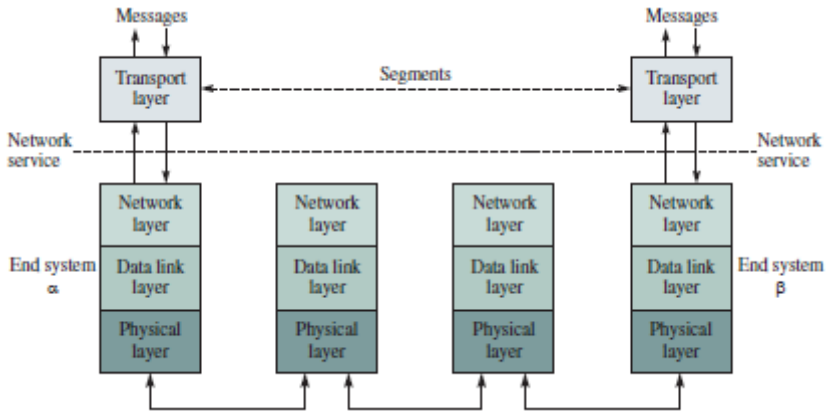

Figure 7.2: Peer to peer protocols operating end to end across the network--protocol stack view

PACKET NETWORK TOPOLOGY

ACCESS NETWORK

• The access multiplexer is used to combine the typically bursty traffic flows from the individual computers into aggregated flows so that the transmission line to the packet network is used more efficiently (Figure 7.4).
• An example of an access multiplexer includes a DSLAM (Digital Subscriber Loop Access Multiplexer) located in a telephone central office.
• The computer in the subscriber home is connected to the DSLAM in the central office via an ADSL modem.

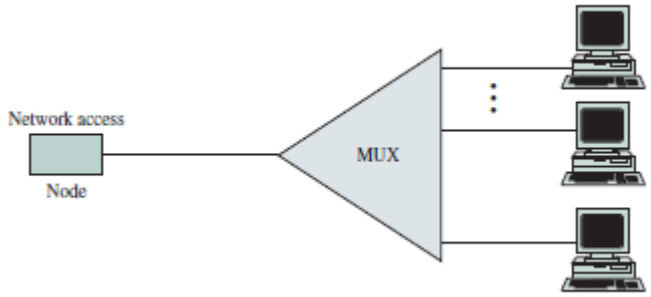

Figure 7.4: Access network

CAMPUS NETWORK

• LANs for a large group of users such as a department are interconnected in an extended LAN through the use of LAN switches(s).
• Resources such as servers & databases that are primarily of use to this department are kept within the subnetwork. Thus, delays in accessing the resources can be reduced (Figure 7.5).
• Each subnetwork has access to the rest of the organization through a router(R) that accesses the campus backbone network.
• A subnetwork also uses the campus backbone to reach the "outside world" such as the Internet.
• Servers containing critical resources that are required by entire organization are usually located in a data center
→ where they can be easily maintained &
→ where security can be enforced
• The routers in the campus network are interconnected to form the campus backbone network(S).
• Routers are interconnected by very high speed LANs. For example: Gigabit Ethernet or an ATM network.
• Routers use IP which enables them to operate over various data link and network technologies.
• Routers exchange information about the state of their links to dynamically calculate routing tables.

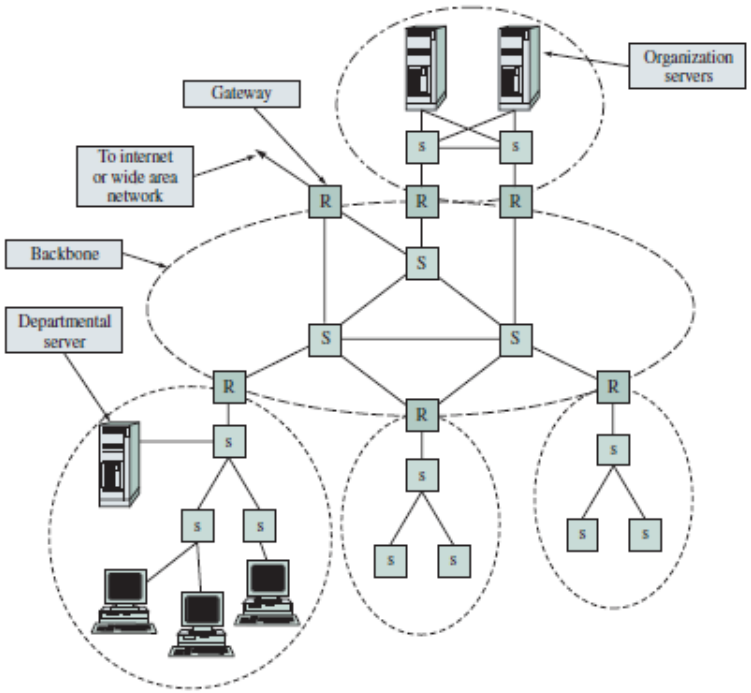

Figure 7.5: Campus network

DATAGRAMS AND VIRTUAL CIRCUITS

CONNECTIONLESS PACKET SWITCHING (DATAGRAM)

- This is analogous to postal system (Figure 7.11).
- Each packet is routed independently through the network.
- Each packet has a header attached to it to provide source and destination addresses.
- Each switch examines the header to determine the next hop in the path to the destination.
- If the transmission line is busy
  then the packet is placed in the queue until the line becomes free.
- Advantage: High utilization of transmission-line can be achieved by sharing among multiple packets.
- Disadvantage: 1) Packets may arrive out-of-order, and re-sequencing may be required at the destination
  2) Loss of packets may occur when a switch has insufficient buffer
- If the path followed by a sequence of packets consists of L hops with identical propagation delays P and transmission speeds (Figure 7.12), then the delay incurred by a message that consists of k packets is given by

$$Lp + LP + (k-1)P$$

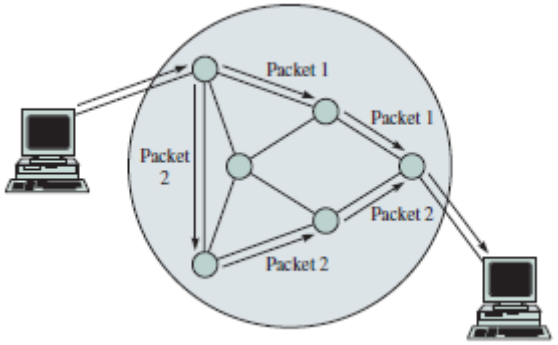

Figure 7.11:Datagram packet switching

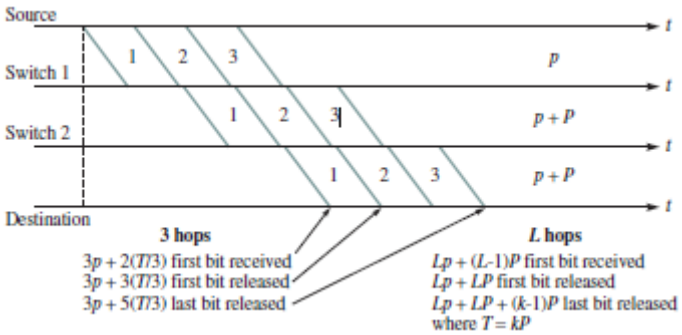

Figure 7.12:Delays in datagrams packet switching

CONNECTION-ORIENTED PACKET SWITCHING (VIRTUAL CIRCUIT)

- This is similar to telephone system (Figure 7.14).
- Fixed path(connection) is established between a source and a destination prior to the transfer of packets.
- The virtual-circuit setup procedure
  - → first determines a path through the network &
  - → sets parameters in the switches by exchanging connect-request & connect-confirm messages
- If a switch does not have enough resources to set up a virtual circuit, the switch responds with a connect-reject message and the setup procedure fails (Figure 7.15).
- A connection-release procedure may also be required to terminate the connection.
- Advantage: 1) Packets from many flows can share the same transmission line.
  2) During connection setup, parameters related to usage & QoS may be negotiated and network resources may be allocated
- Disadvantages: 1) When a fault occurs in the network, all affected connections must be set up again.
  2) The switches in the network need to maintain information about the flows that pass the switches. Thus, the amount of required "state" information grows very quickly with no. of flows.
- Buffer and transmission resources are not dedicated explicitly for the use of the virtual circuit
- The packets are delivered to the receiver in the same order as transmitted by sender (Figure 7.16).

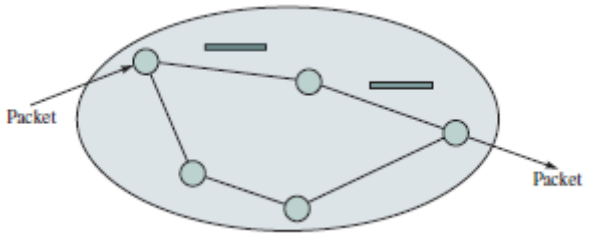

Figure 7.14: Virtual Circuit packet switching

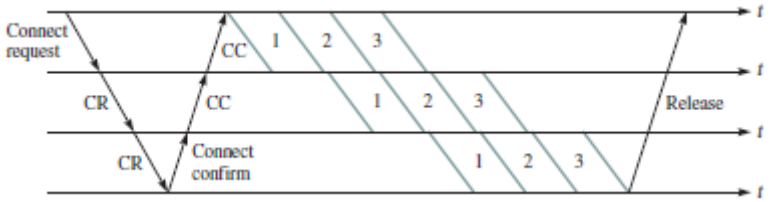

Figure 7.15: Delays in virtual circuit packet switching

Figure 7.16: Signaling message exchange in call set up

Virtual Circuit	***Datagram***
Circuit setup is required	Circuit setup is not required
Each packet contains a short VC number as address	Each packet contains the full source and destination address.
Route chosen when VC is setup and all packets follow this route.	Each packet is routed independently
In case of router failure, all VC that passed through the router are terminated.	Only crashed packets lost.
Congestion control is easy using buffers.	Difficult congestion control.
Complexity in the network layer	Complexity in transport layer.

STRUCTURE OF A PACKET-SWITCH

- A packet-switch performs 2 main functions:
  1) Routing function uses algorithms to find a path to each destination and store the result in a routing table.
  2) Forwarding function processes each incoming packet from an input port and forwards the packet to the appropriate output port (based on the information stored in the routing table).
- A line card contains several input/output ports (Figure 7.19).
  The line card also contains some buffers and the associated scheduling algorithms.
  The line card is concerned with symbol timing, line coding, framing, physical layer addressing & error checking.
- Programmable network processor performs packet-related tasks such as table lookup and packet scheduling.
- A controller contains a general-purpose processor which is used for control & management functions depending on the type of packet switching.
  The controller acts as a central coordinator, as it communicates with each line card and the interconnection fabric.
- Interconnection fabric is used to transfer packets between the line cards.
- Problem: Interconnection fabric is likely to be the bottleneck if there are many high-speed line cards, since all traffic from the input line cards have to go through to the interconnection fabric (Figure 7.20).
- Buffers need to be added to the crossbar interconnection fabric to accommodate packet contention
- How to eliminate head-of-line(HOL) blocking?
  Solution: Provide N separate input buffers at each input port so that each input buffer is dedicated to a particular output. Such an input buffer is called a virtual output buffer.

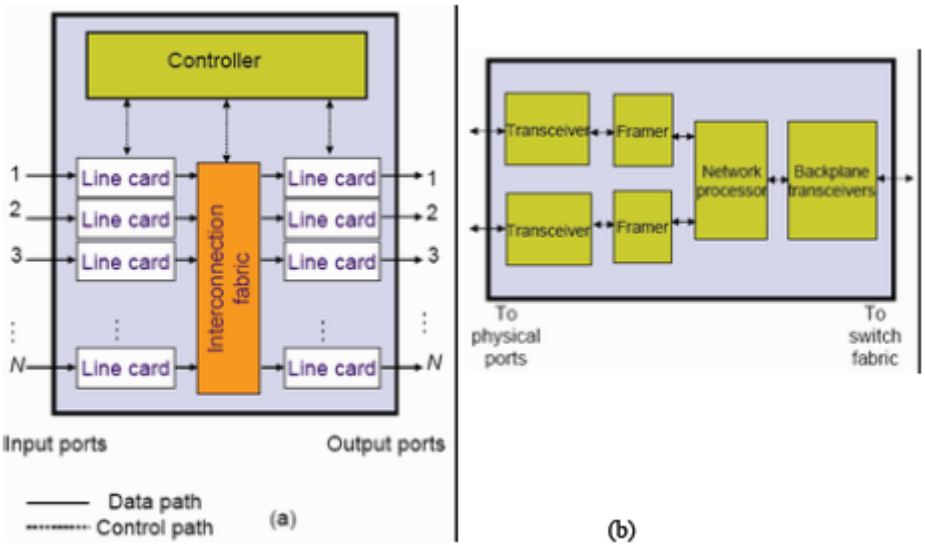

Figure 7.19:a)components of a generic packet switch b)organization of a line card

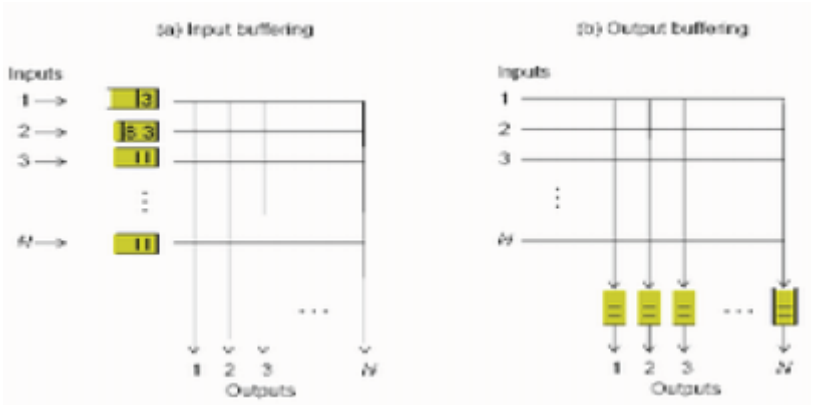

Figure 7.20: Input buffering versus output buffering

ROUTING IN PACKET NETWORKS

• Routing means determining feasible paths for packets to follow from each source to each destination.

ROUTING ALGORITHM SHOULD SEEK ONE OR MORE OF THE FOLLOWING GOALS

1)Rapid and accurate delivery of packets
2) Adaptability to changes in network topology resulting from node or link failures
3) Adaptability to varying source-destination traffic loads
4) Ability to route packets away from temporarily congested links
5) Ability to determine the connectivity of the network
6) Ability to avoid routing loops
7) Low overhead: The control messages represent an overhead on bandwidth usage that should be minimized.

ROUTING ALGORITHM CLASSIFICATION

Static (Non-adaptive) Routing

• Paths are precomputed based on the network topology, link capacities and other information.
• A dedicated host is used to perform computation offline.
• The precomputed paths
→ are loaded to the routing-table &
→ remain fixed for a relatively long period of time
• Static routing is suitable if
→ network size is small
→ network topology is relatively fixed
• Disadvantage: 1) Inability to react rapidly to network failures.
2) Static routing may become cumbersome, as the network size increases.

Dynamic (Adaptive) Routing

• Each node continuously learns state of the network by communicating with its neighbors. Thus, a change in a network-topology is eventually propagated to all nodes
• Based on the information collected, each node computes the best path to the destination.
• Disadvantage: Added complexity in the node.

Centralized Routing

• A network control center
→ computes all paths &
→ then uploads this information to the nodes in the network.

Distributed Routing

• Nodes cooperate by means of message exchanges and perform their own routing computations.
• Advantage: These algorithms scales better than centralized algorithm
Disadvantage: More likely to produce inconsistent results.

ROUTING TABLES

VIRTUAL CIRCUIT

• Once the routing decision is made, the path-information is stored in the routing-table so that the node knows how to forward packets

• Abbreviated headers can be used. At the input to every switch, the virtual circuit is identified by a VCI(virtual circuit identifier).

• The routing table

→ translates each incoming VCI to an outgoing VCI (Figure 7.24) &

→ identifies output port to which to forward a packet based on incoming VCI of the packet

• How to Route Packet from host A to host B(Figure 7.23)?

1) Firstly, packet will be sent from host A to node 1 with outgoing VCI 1
2) After arriving at node 1, packet receives the outgoing VCI 2 and is then forwarded to node 3
3) After arriving at node 3, packet receives the outgoing VCI 7 and is then forwarded to node 6
4) After arriving at node 6, packet receives the outgoing VCI 8 and is finally delivered to host B

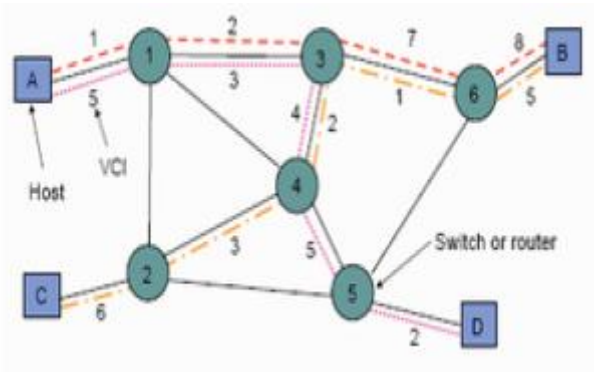

Figure 7.23: Virtual circuit identifier determines the destination

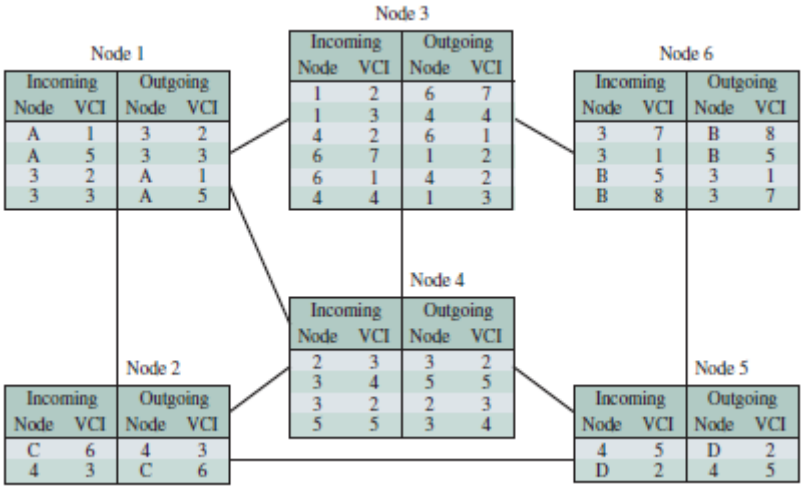

Figure 7.24: Routing tables for the packet switching network in figure 7.23

DATAGRAM PACKET SWITCHING

• The routing table identifies the next hop to which to forward a packet based on the destination address of the packet (Figure 7.25).

• How to route from node 1 to node 6(Figure 7.22)?

1) The packet is first forwarded to node 3 based on the corresponding entry in the routing table at node 1.

2) Node 3 then forwards the packet to node 6.

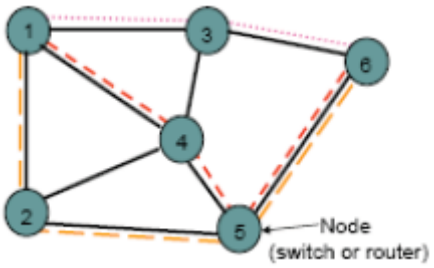

Figure 7.22: Multiple paths in a packet switching network

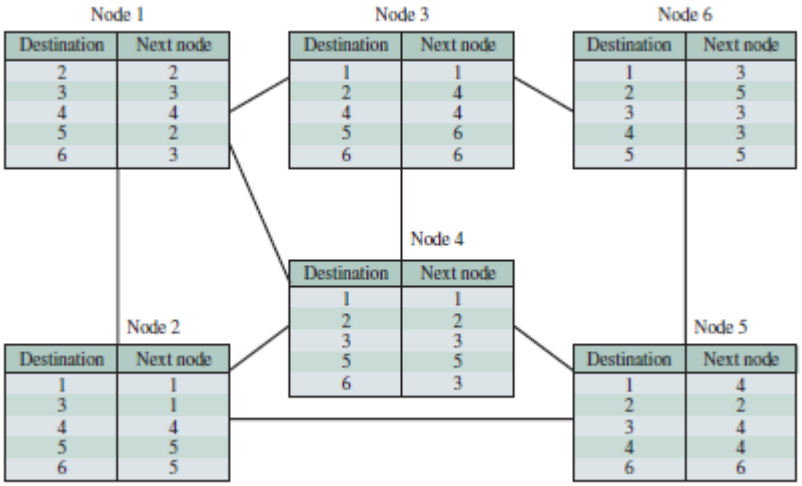

Figure 7.25: Routing tables for datagram network in figure 7.22

HIERARCHICAL ROUTING

• Hierarchical approach can be used to reduce size of routing table in the routers (Figure 7.26).

• Hosts that are near each other should have addresses that have common prefixes.

• In this way, routers need to examine only part of address in order to decide how a packet should be routed.

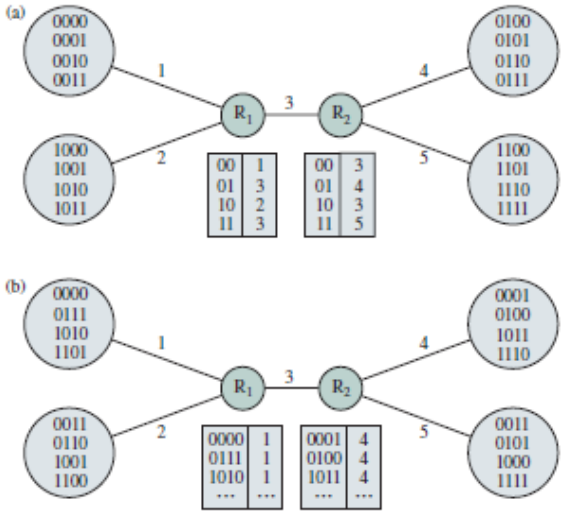

Figure 7.26: Address assignment a) Hierarchical b)flat

SPECIALIZED ROUTING

FLOODING

- Switch forwards an incoming-packet to all ports except the one packet was received from (Fig 7.27).
- Flooding is an effective routing approach
  - → when the information in the routing tables is not available such as during system-startup
  - → when the source needs to broadcast a packet to all nodes in the network
- Problem: Flooding generates large number of duplicate packets, which may easily swamp the network.
- Solution: To reduce resource consumption in the network, following mechanisms can be used:

  1) Use a TTL(time-to-live) field in each packet: When source sends a packet, TTL is initially set to some number. Each node decrements TTL value by 1 before flooding the packet. If TTL reaches zero, the node discards the packet.

  2) Add an identifier before flooding: Each node adds its identifier(ID) to the header before flooding the packet. If the node receives a packet that contains it's own ID, it will discard the packet

  3) Have a unique sequence number: Each packet is assigned a unique sequence number. When a node receives a packet, the node records source address and sequence number of the packet. If the node discovers that the packet has already visited the node, it will discard the packet.

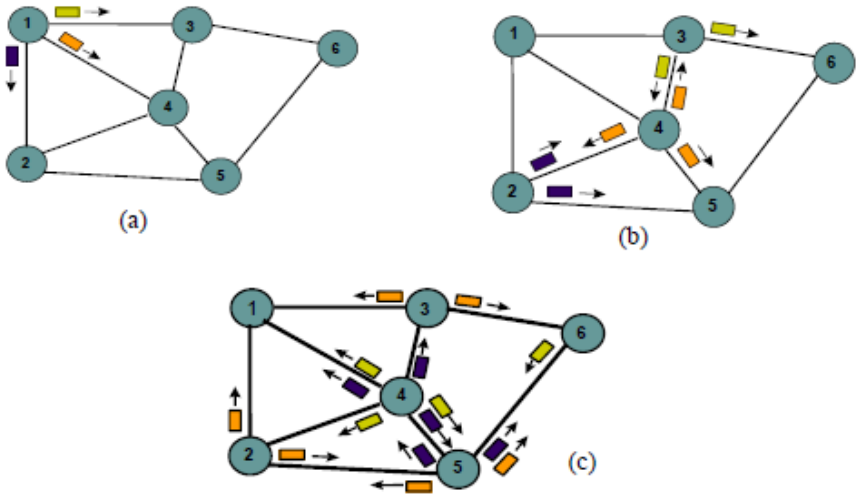

Figure 7.27: Flooding is initiated from node 1: a)hop-1 transmission b)hop-2 transmission and c)hop-3 transmission

DEFLECTION ROUTING

• Multiple paths for each source-destination pair is provided (Figure 7.29).
• Each node first tries to forward a packet to the preferred port.
• If the preferred port is busy or congested, the packet is deflected to another port.
• If node (0,2) wants to send a packet to node (1,0) , the packet can go 2 left and 1 down. However, if the left port of node (0,1) is busy ,the packet will be deflected to node (3,1). Then, it can go through nodes (2,1),(1,1),(1,2),(1,3) and eventually reach the destination node (1,0).
• Advantage: Node can be bufferless (since packets do not have to wait for a specific port to become available)
Disadvantage: Cannot guarantee in-sequence delivery of packets
• Deflection routing can be used
→ in optical networks where optical buffers are currently expensive and difficult to build.
→ to implement high-speed packet switches.

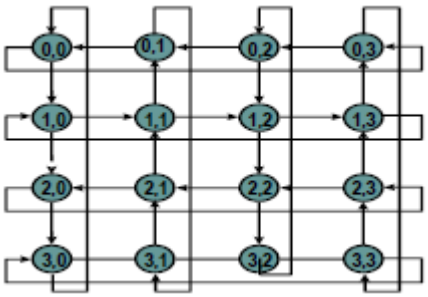

Figure 7.28: Manhattan street network

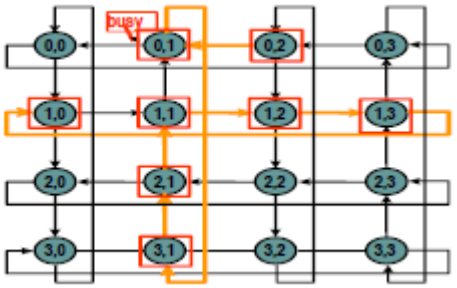

Figure 7.29: Deflection routing in Manhattan street network

SHORTEST PATH ROUTING

- The shortest path from node 2 to node 6 is 2-4-3-6, and the path cost is 4 (Figure 7.30).
- Following metrics can be used to assign a cost to each link:

1) Cost~1/capacity: Here, one assigns higher costs to lower capacity links. The objective is to send a packet through a path with the highest capacity.
2) Cost~delay: Delay includes queuing-delay in switch-buffer and propagation-delay in the link. Shortest path represents the fastest path to reach the destination.
3) Cost~congestion: The shortest path tries to avoid congested links.

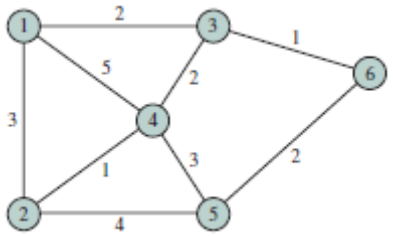

Figure 7.30: A network with associated link costs

BELLMAN-FORD ALGORITHM

- If each neighbor of node A knows the shortest path to node Z, then node A can determine its shortest path to node Z by calculating the cost to node Z through each of its neighbors and picking the minimum.
- Let D_j = current estimate of the minimum cost from node j to the destination

Let C_{ij} = link cost from node i to node j. (For example $C_{13}=C_{31}=2$)

The link cost from node i to itself is defined to be zero ($C_{ii}=0$).

The link cost between node i & node k is infinite if node i & node k are not directly connected.(for example $C_{15}=C_{23}=\sim$)

- If the destination node is node 6, then the minimum cost from node 2 to the destination node 6 can be calculated in terms of distances through node 1, node 4 or node 5(Figure 7.31):

$D_2 = \min\{C_{21}+D_1, C_{24}+D_4, C_{25}+D_5\}$
$= \min\{3+3, 1+3, 4+2\}$
$= 4$

1. Initialization

$$D_i = \infty, \forall i \neq d \quad (4)$$
$$D_d = 0$$

2. Updating: For each $i \neq d$,

$$D_i = \min_j\{C_{ij} + D_j\}, \forall j \neq i \quad (5)$$

Repeat step 2 until no more changes occur in the iteration.

Iteration	Node 1	Node 2	Node 3	Node 4	Node 5
Initial	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$	$(-1, \infty)$
1	$(-1, \infty)$	$(-1, \infty)$	(6, 1)	(3, 3)	(6, 2)
2	(3, 3)	(4, 4)	(6, 1)	(3, 3)	(6, 2)
3	(3, 3)	(4, 4)	(6, 1)	(3, 3)	(6, 2)

Table 7.1: simple processing of Bellman Ford algorithm. Each entry for node represents the next node and cost of the current shortest path to destination 6

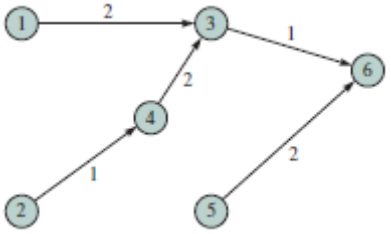

Figure 7.31: shortest path to node 6

UNIT 2: PACKET SWITCHING NETWORKS(CONT.)

DIJKSTRA'S ALGORITHM

- This is used to find the shortest paths from a source node to all other nodes in a network (Figure 7.32).
- Main idea: Progressively identify the closest nodes from the source node in order of increasing path cost.

1. Initialization:

$$N = \{s\}$$
$$D_j = C_{sj}, \forall\, j \neq s \qquad (7)$$
$$D_s = 0$$

2. Finding the next closest node: Find node $i \notin N$ such that

$$D_i = \min_{j \notin N} D_j \qquad (8)$$

Add i to N.
If N contains all the nodes, stop.

3. Updating minimum costs: For each node $j \notin N$

$$D_j = \min\{D_j, D_i + C_{ij}\} \qquad (9)$$

Go to step 2.

Iteration	N	D_2	D_3	D_4	D_5	D_6
Initial	{1}	3	2	5	∞	∞
1	{1,3}	3	2	4	∞	3
2	{1,2,3}	3	2	4	7	3
3	{1,2,3,6}	3	2	4	5	3
4	{1,2,3,4,6}	3	2	4	5	3
5	{1,2,3,4,5,6}	3	2	4	5	3

Table 7.5: execution of Dijkstra's algorithm

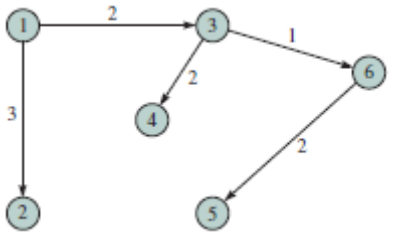

Figure 7.32: shortest path tree from node 1 to other nodes

Destination	Next node	Cost
2	2	3
3	3	2
4	3	4
5	3	5
6	3	3

Table 7.6: routing table for fig 7.32

SOURCE ROUTING VERSUS HOP-BY-HOP ROUTING

Hop-by-hop Routing

- Each node is responsible for determining the next-hop along the shortest path.
- This is used in datagram network.

Source Routing

- Source is responsible for determining the shortest path to the destination (Figure 7.35).
- Source includes path-information in the header.
- Path-information contains sequence of nodes to be traversed.
- There are mainly 2 types:
  1) Strict source routing:
     Source specifies address of each node along the path to the destination
  2) Loose source routing:
     Source only knows partial information of the network topology (i.e. only a subset of the nodes along the path is specified).

Explicit Routing

- This is another variation of source routing.
- This allows a particular node to determine the path.

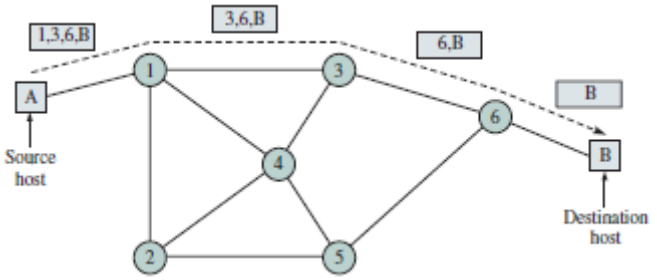

Figure 7.35: Example of source routing

LINK-STATE ROUTING VERSUS DISTANCE-VECTOR ROUTING

Funtionality	***Distance vector***	***Link State***
Primary principle	Sends larger updates , about the complete network information only to neighboring routers	Sends smaller updates, about the link state of neighbors, to all routers
Learning about network	Learns about network only from neighbors	Learn about network only from all routers
Building the routing table	Based on inputs from only neighbors	Based on complete database collected from all routers
Advertisement of updates	Periodically (e.g. every 30 seconds)	Triggered updates, only when there is a change
Routing loops	More prone ; suffer from problems like count-to-infinity	Less prone to routing loops
Convergence (stabilisation)	Slow	Fast
Resources	Less CPU power and memory	More CPU power and memory required
Cost	Less cost	More than Distance vector
Scalability	Less scalable	More scalable than distance vector

TRAFFIC MANAGEMENT AT THE PACKET LEVEL

• Traffic management is concerned
→ with delivery of QoS to the end user &
→ with efficient use of network resources.
• Based on traffic granularity, we can classify traffic management into three levels: packet level, flow level and flow-aggregated level.
• Packet level is concerned with packet queueing and packet scheduling at switches, routers & multiplexers to provide differentiated treatment for packets belonging to different QoS classes.

END-TO-END DELAY

• This is the sum of the individual delays experienced at each system.
• If we can guarantee that the delay at each system can be kept below some upper bound, then the end-to-end delay can be kept below the sum of the upper bounds.

JITTER

• This measures the variability in the packet delays.
• This is typically measured in terms of the difference of the minimum delay and maximum delay.

QUEUE SCHEDULING

• This is concerned with strategies for controlling the transmission bit rates that are provided to the various information flows.

FIFO QUEUEING

• Packets are transmitted in order of their arrival (Figure 7.41a).
• Packets are discarded when they arrive at a full buffer.
• Packet-delay depends on the packet inter-arrival time.
Packet-loss depends on the packet lengths.
• Disadvantage: 1) This is not possible to provide different information flows with different QoS.
2) hogging occurs when a user sends packets at a high rate and fills the buffers in the system,thus depriving other users of access to the buffer.

FIFO QUEUEING WITH DISCARD POLICY

• Provide different characteristics of packet-loss performance to different classes of traffic.
• Higher priority packets(Class 1) are discarded when they arrive at a full buffer(Figure 7.41b).
• Lower priority packets(Class 2) are discarded when buffer reaches a certain threshold.
• Disadvantage: Lower priority packets will experience a higher packet-loss probability.

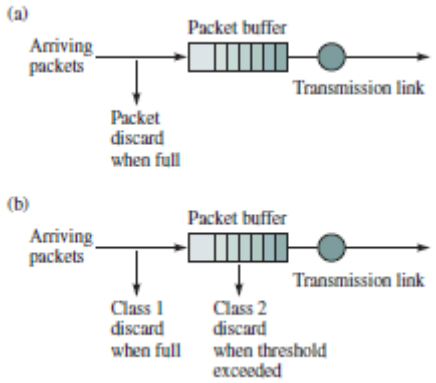

Figure 7.41:a)FIFO queueing b)FIFO queueing with discard policy

HEAD-OF-LINE(HOL) PRIORITY QUEUEING

• Number of priority classes are defined (Figure 7.42).
• A separate buffer is maintained for each priority class.
• Each time the transmission link becomes available, the next packet for transmission is selected from the head of the line(HOL) of the highest priority queue that is not empty.
• The size of the buffers for the different priority classes can be selected to meet different loss probability requirements.
• Disadvantage: 1) This does not discriminate among users of the same priority
2) This does not allow for providing some degree of guaranteed access to transmission bandwidth to the lower priority classes.
3) Fairness problem arises when a certain user hogs the bandwidth by sending an excessive number of packets.

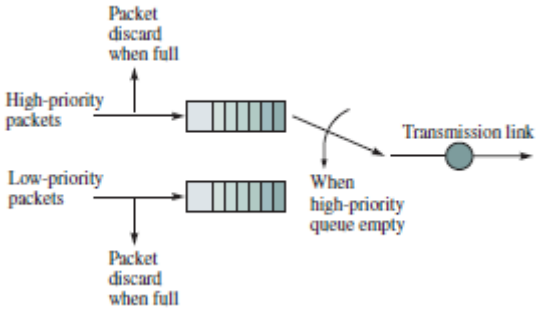

Figure 7.42: HOL Priority Queueing

SORTING PACKETS BASED ON PRIORITY TAG

• Packets are sorted in the buffer based on priority tag (Figure 7.43).
• Priority tag reflects the urgency with which each packet needs to be transmitted.
• Advantage: This system is very flexible because the method for defining priority is open and can even be defined dynamically.

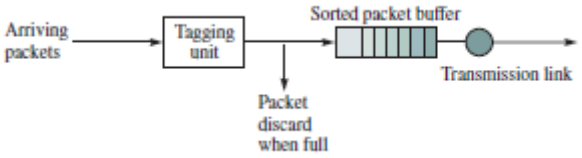

Figure 7.43:Sorting packets according to priority tag

FAIR QUEUEING

- The transmission bandwidth is divided equally among the buffers (that have packets to transmit).
- Each user flow has its own logical buffer (Figure 7.44).
- The size of the buffer for each user flow can be selected to meet specific loss probability requirements.
- Round-robin scheduling is used to service each nonempty buffer one bit at a time.
- Disadvantage: Extensive processing at the destination.

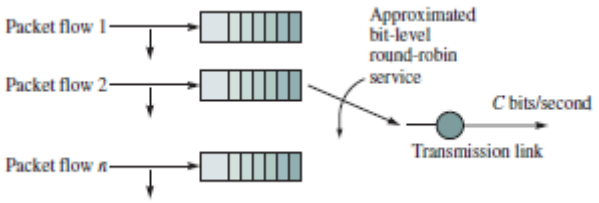

Figure 7.44:Fair queueing

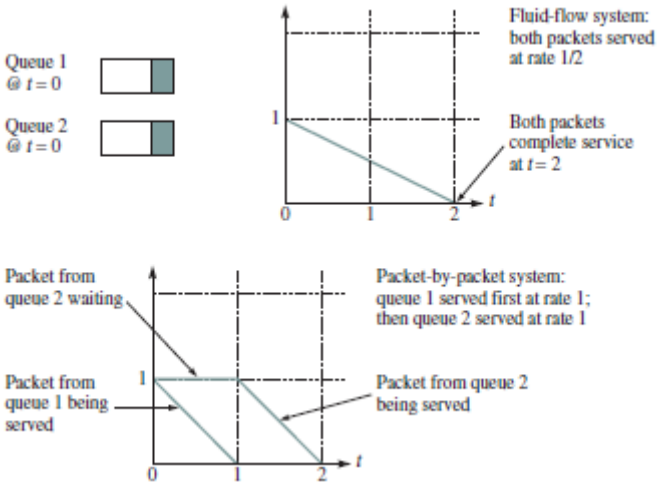

Figure 7.45: fluid flow and packet-to-packet fair queuing (two packets of equal length)

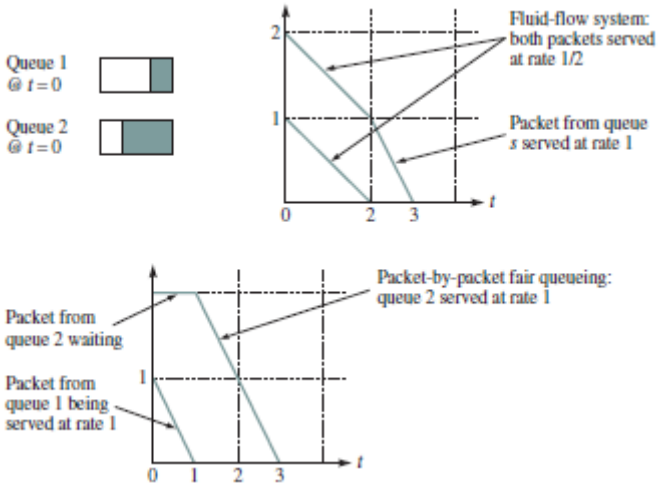

Figure 7.46: fluid flow and packet-to-packet fair queuing(two packets of different length)

WEIGHTED FAIR QUEUEING

• Each user flow has its own buffer, but each user flow also has a weight that determines its relative share of the bandwidth (Figure 7.47).
• If buffer 1 has weight 1 and buffer 2 has weight 3,then buffer 1 will receive 1/4 of the bandwidth and buffer 2 will receive 3/4 of the bandwidth.
• This is also easily approximated in ATM. In each round, each non-empty buffer would transmit a number of packets proportional to its weight.
• Packet by packet weighted fair queueing is also easily generalized from fair queueing.
• Weighted fair-queueing systems are means for providing QoS guarantees.

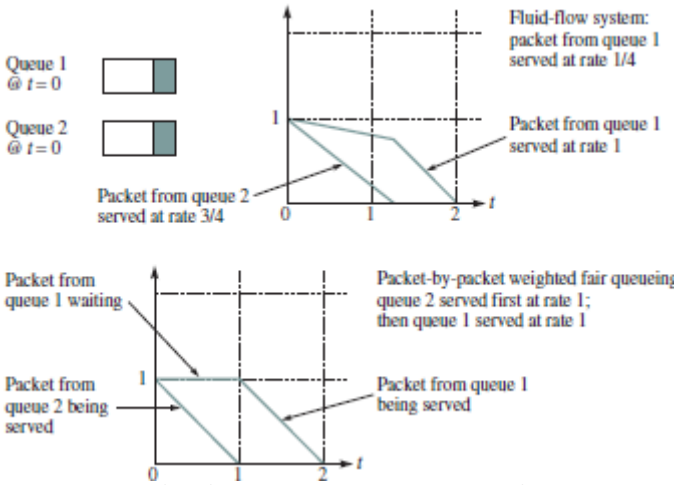

Figure 7.47: fluid flow and packet-to-packet weighted fair queuing

RANDOM EARLY DETECTION

• This is a buffer management technique that attempts to provide equitable access to a FIFO system by randomly dropping arriving packets before the buffer overflows (Figure 7.48).
• A dropped packet provides feedback information to the source and informs the source to reduce its transmission rate.
• This algorithm uses an average queue length rather than instantaneous queue length to decide how to drop packets.
• Specifically, two thresholds are defined: min_{th} and max_{th}.
• When the average queue length is below min_{th}, RED does not drop any arriving packets.
• When the average queue length exceeds max_{th}, RED drops any arriving packets.
• When the average queue length is between min_{th} and max_{th}, RED drops an arriving packet with an increasing probability as the average queue length increases.

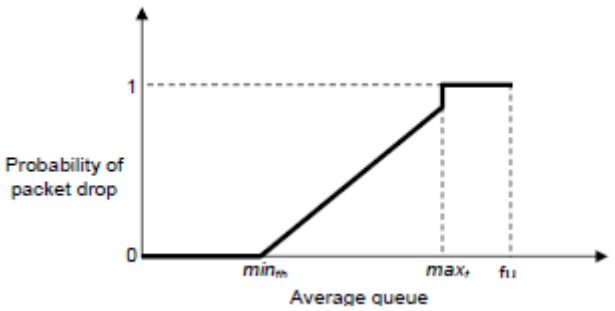

Figure 7.48: Packet drop profile in RED

TRAFFIC MANAGEMENT AT THE FLOW LEVEL

• At flow level, traffic-management is concerned with managing individual flow to ensure that QoS requested by the user is satisfied.

• Purpose of traffic-management: 1) To control flows of traffic &
2) To maintain performance even in the presence of traffic-overload.

• When too many packets compete for same resource, the network-performance degrades and this situation is called as congestion.

• For example, consider the packet-switching network(Figure 7.50). Suppose that nodes 1, 2 and 5 continuously transmit packets to their respective destinations via node 4. If the aggregate incoming-rate of the packet flows to node 4 is greater than the rate the packets can be transmitted out, the buffer in node 4 will build up. If this situation persists, the buffer eventually becomes full and starts discarding packets. The net result is that the throughput at the destination will be very low.

• The process of managing traffic-flow in order to control congestion is called congestion-control.

• Congestion control algorithms can be classified into two types: open-loop control and closed-loop control.

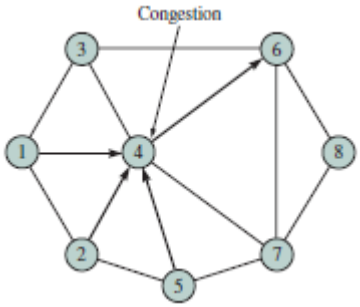

Figure 7.50: Congestion arises when incoming rate exceeds outgoing rate

OPEN-LOOP CONTROL

• This prevents congestion from occurring by making sure that the flow generated by source will not degrade network-performance to a level below the specified QoS.

• If QoS cannot be guaranteed, network rejects flow before it enters the network.

• The function that makes the decision to accept or reject a new traffic-flow is called an admission-control.

Closed-Loop Control

• This reacts to congestion when it
→ is already happening or
→ is about to happen.

• This regulates traffic-flow according to state of network.

• This does not use any reservation.

OPEN-LOOP CONTROL

• This relies on 3 mechanisms to guarantee network-performance during lifetime of admitted flows:
1) Admission control, 2) Policing and 3) Traffic shaping.

ADMISSION CONTROL

• The function that makes the decision to accept or reject a new traffic-flow is called an admission-control.
• This computes resource requirements of a new flow (typically buffer & bandwidth).
• This determines whether resources along path to be followed by new flow are available.
• Before initiating a new flow, a source must first obtain permission from an admission-control entity.
• QoS of new flow can be satisfied without violating QoS of existing flows, the flow is accepted;
otherwise, the flow is rejected.
• To determine whether the QoS of the flow can be satisfied, the admission control entity has to know the traffic parameters and the QoS requirements of the flow.
• Typical traffic parameters include peak rate, average rate and maximum burst size.
• The peak rate defines the maximum rate that the source will generate its packets (Figure 7.51).
The maximum burst size determines maximum length of time traffic can be generated at peak rate.
The average rate defines the average rate that source will generate its packets.
• The amount of bandwidth generally lies between the average rate and the peak rate and is called the effective bandwidth of the flow.

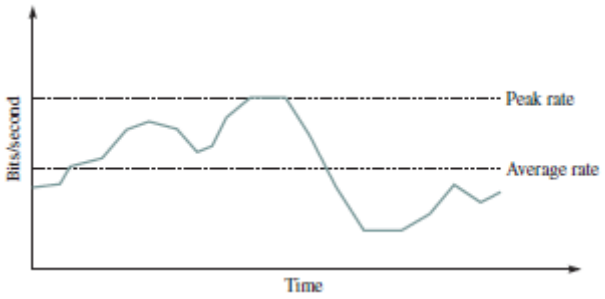

Figure 7.51: Example of a traffic flow

POLICING

• The process of monitoring & enforcing the traffic-flow is called the policing.
• When traffic-flow violates agreed-upon contract, the network may choose to tag (or discard) the nonconforming traffic.
• Tagging essentially lowers priority of nonconforming traffic.
• When network resources are exhausted, tagged traffic is the first to be discarded.
• Policing-device can be implemented based on the concept of a leaky bucket.
• Imagine the traffic-flow to a policing-device as water being poured into a bucket that has a hole at the bottom.
• Bucket leaks at a constant rate (Figure 7.52).
• When bucket is full, a new portion of water is said to be nonconforming and the water can be discarded.
When water is poured into bucket & overflow does not occur, a new portion of water (i.e. packet) is said to be conforming .
• The hole ensures that bucket will never overflow as long as drain-rate is higher than rate water is being poured in.

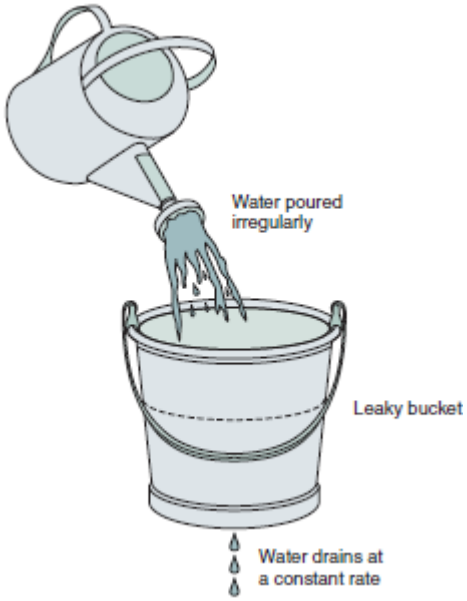

Figure 7.52: leaky bucket

LEAKY BUCKET ALGORITHM

• A counter records content of leaky-bucket (Figure 7.53).
• When a packet arrives, counter is incremented by some value I. Where I indicates nominal interarrival-time of packet that is being policed.
• If bucket-content does not exceed a certain limit, packet is declared to be conforming.
If bucket-content exceeds the limit, counter remains unchanged and packet is declared to be nonconforming.
• As long as the bucket is not empty, the bucket will drain at a continuous rate of 1 unit per packet time.
• The inverse of I is called the sustainable rate, which is the long-term average rate allowed for the conforming traffic.
• Suppose the peak rate of a given traffic flow is denoted by R and its inverse is T, that is, T=1/R. Then, the maximum burst size is given by

$$MBS = 1 + \left[\frac{L}{1-T}\right]$$

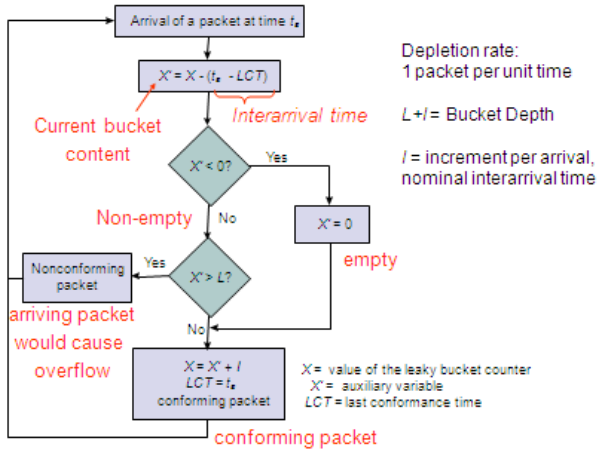

Figure 7.53: Leaky bucket algorithm used for policing

DUAL LEAKY BUCKET

• Dual leaky bucket is use to police multiple traffic parameters like PCR, SCR, and MBS (Figure 7.54).
• Traffic is first checked for SCR at first leaky bucket.
• Nonconforming packets at first bucket are dropped or tagged.
• Conforming (untagged) packets from first bucket are then checked for PCR at second bucket.
• Nonconforming packets at second bucket are dropped or tagged.

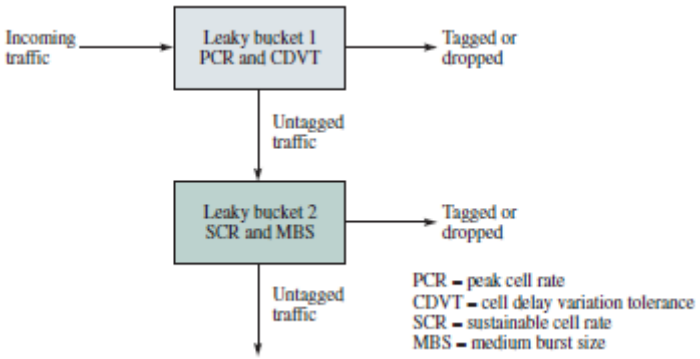

Figure 7.54: A dual Leaky bucket configuration

TRAFFIC SHAPING

• This refers to process of altering a traffic-flow to ensure conformance.
• This can be implemented using 1) Leaky-Bucket 2)Token Bucket

LEAKY-BUCKET TRAFFIC-SHAPER

• Packets are served periodically so that the stream of packets at the output is smooth (Figure 7.57a).
• Buffer is used to store momentary bursts of packets.
• Buffer-size defines the maximum burst that can be accommodated.
• If buffer is full, incoming packets are discarded.
• A policing-device checks and passes each packet on the fly (Figure 7.57b).
• A traffic shaping device needs to introduce certain delays for packets that arrive earlier than their scheduled departures.
• Drawback: The leaky-bucket traffic-shaper is very restricted, since the output rate is constant. Many applications produce variable rate traffic. If the traffic flows from such applications have to go through the traffic-shaper, the delay through the buffer can be unnecessarily long.

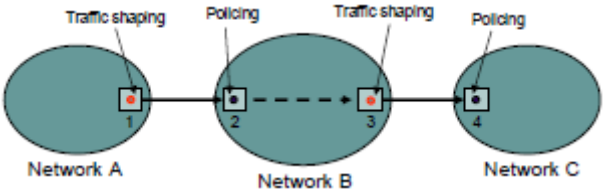

Figure 7.57a: Typical locations of policing and traffic shaping devices

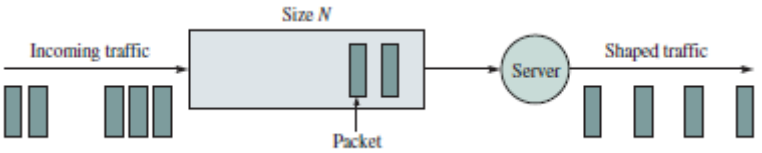

Figure 7.57b:A leaky bucket traffic shaper

TOKEN BUCKET TRAFFIC SHAPER

- This regulates only the packets that are not conforming (Figure 7.58).
- Packets that are deemed conforming are passed through without further delay.
- Tokens are generated periodically at a constant rate.
- Tokens are stored in a bucket.
- If the bucket is full, arriving tokens are discarded.
- A packet from the buffer can be taken out only if a token in the bucket can be drawn.
- If bucket is empty, arriving packets have to wait in the buffer.
- The backlogged packets have to wait for new tokens to be generated before they can be transmitted out.

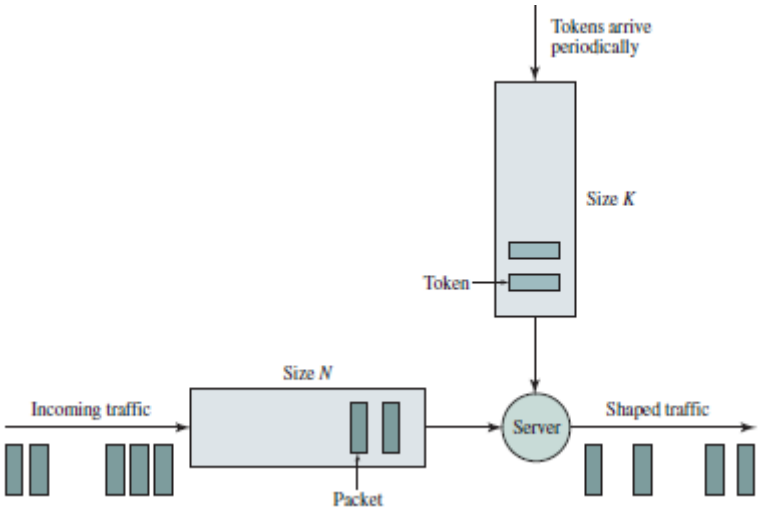

Figure 7.59: token bucket traffic shaper

Leaky-bucket	***Token bucket***
Discards packets	Discards tokens
A packet can be transmitted if the bucket is not full	A packet can only be transmitted if there are enough tokens to cover its length in bytes
Sends the packets at an average rate	Allows for large bursts to be sent faster by speeding up the output
Does not allow saving, a constant rate is maintained	Allows saving up tokens(permissions) to send large bursts

CLOSED-LOOP CONTROL

• This reacts to congestion when it
→ is already happening or
→ is about to happen

• This relies on feedback information to regulate a packet flow-rate according to feedback information about the state of the network (which may be based on buffer content or link utilization).

• The recipient of the feedback information usually depends on the communication layer that is responsible for congestion control.

End-to-End Closed Loop Control

• The feedback information about state of the network is propagated back to the source that can regulate the packet flow-rate (Figure 7.62a).

• The feedback information
→ may be forwarded directly by a node that detects congestion or
→ may be forwarded to the destination first which then relays the information to the source

• The information may not be accurate when the source receives such information.

Hop by Hop Congestion Control

• This can react much faster than the end-to-end counterpart due to shorter propagation delay.

• The state of the network is propagated to the upstream node (Figure 7.62b).

• When a node detects congestion on its outgoing link, it will tell its upstream neighbor to slow down its transmission-rate.

• As a result, the upstream neighbor may also experience congestion some time later if the incoming rate exceeds the outgoing transmission rate.

• This "back-pressure" process from one downstream node to another node upstream may continue all the way to the source.

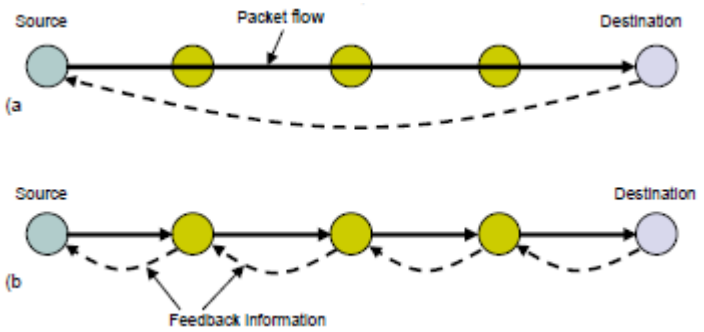

Figure 7.62:closed loop control a)end to end b)hop by hop

Explicit feedback

• A node which detects congestion transmits an explicit message to the source notifying congestion in the network.

• The explicit message can be transmitted as a separate packet (called choke packet) or piggybacked on a data packet.

• For example, Closed loop control in ATM network. Here, each source continuously adjusts its sending rate according to explicit feedback information, which is recorded in a bit(called the EFCI) of the ATM cell header.

Implicit feedback

• The source has to rely on some surrogate information to deduce congestion.

• One example is to use a tin-out based on missing acknowledgments from a destination to decide whether congestion has been encountered in the network.

• For example, TCP congestion control. This regulates the transmission rate using the implicit feedback information derived from a missing acknowledgment.

• When the source performs the time-out, it decreases the transmission rate by reducing it's transmit window.

TRAFFIC MANAGEMENT AT THE FLOW-AGGREGATED LEVEL

• Traffic management at the flow-aggregated level is called traffic engineering (Figure 7.63).
• The main objective of traffic engineering is to map aggregated flows onto the network so that resources are efficiently utilized.
• The shortest-path routing allows traffic to be forwarded to a destination following the shortest path. Unfortunately, mapping the traffic according to shortest paths may not result in overall network efficiency.
• Constraint shortest-path routing is suitable for connection-oriented packet switching networks.
• Suppose that the traffic demand of bandwidth B between a given source and destination pair is to be routed.
• First, the algorithm prunes any link in the network that has available bandwidth less than B. Then, the algorithm runs the shortest path routing to find the paths between the given source and destination pair.
• Suppose that we wish to set up three paths in the following order:
node 1 to node 8(path 1),
node 2 to node 8(path 2), and
node 3 to node 8(path 3).
• Initially, path 1 follows the shortest path 1->4->8. Link (1,4) and link (4,8) are then pruned.
Next path 2 follows the shortest path 2->4->5->8 using the pruned network topology. Now, links (2,4),(4,5) and (5,8) are also pruned.
Path 3 uses the revised pruned topology, which givens 3->6->7->8 as the shortest path.
• Constraint shortest-path routing does not always yield a desired result. Consider the case where the paths to be set up are given in following order: path 2, path 1 and path 3.Here,path 3 cannot be successfully established.
• The order of path setup plays an important role in the eventual path layout.

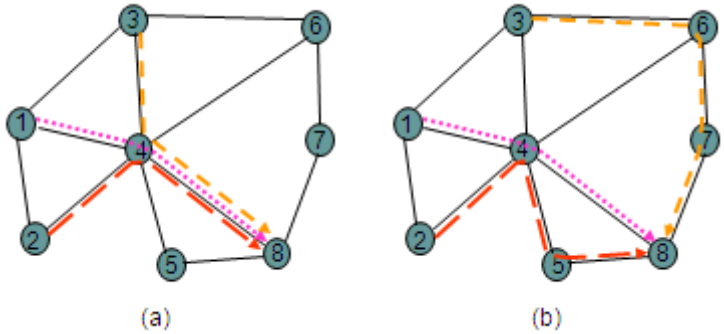

Figure 7.63:Mapping traffic onto the network topology

UNIT 3: TCP/IP

TCP/IP ARCITECTURE

• PDUs exchanged by peer TCP protocols are called *segments*.
PDUs exchanged by UDP protocols are called *datagrams*.
PDUs exchanged by IP protocols are called *packets*.
• IP multiplexes segments & datagrams and performs fragmentation if necessary.
• Packets are sent to the network-interface for delivery across the physical network.
• At the destination, packets are demultiplexed to the appropriate protocol (IP, ARP or RARP).
• The receiving IP entity determines whether a packet should be sent to TCP or UDP.
• Finally, TCP(or UDP) sends each segment(datagram) to the appropriate application based on the port number.

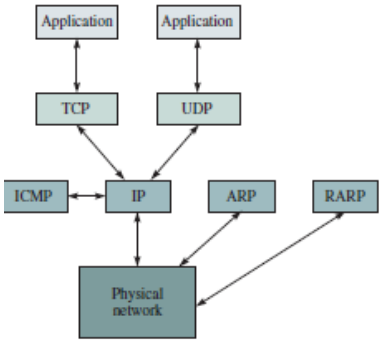

Figure 8.1: TCP/IP protocol suite

INTERNET & NETWORK INTERFACE LAYERS

• An HTTP GET command is passed to TCP layer, which encapsulates message into a TCP segment
• The segment header contains an ephemeral port number for the client process and
well-known port 80 for the HTTP server process (Figure 8.2).
• The TCP segment in turn is passed to the IP layer where it is encapsulated in an IP packet.
• The packet header contains a network address for the sender and a network address for the destination.
• The IP packet is then passed through the network interface and encapsulated into an Ethernet frame.
• The frame header contains physical addresses that identify physical endpoints for Ethernet sender and receiver.
• The logical IP addresses need to be converted into specific physical addresses to carry out the transfer of bits from one device to the other. This conversion is done by an address resolution protocol (ARP).
• Each host in the Internet is identified by a globally unique IP address. An IP address is divided into 2 parts: 1) a network ID and 2) a host ID.
• The Internet layer provides for the transfer of information across multiple networks through the use of routers.
• IP packets are exchanged between routers without a connection setup. They are routed independently and may traverse different paths.
• Network interface layer is concerned with the protocols that are used to access the intermediate networks.

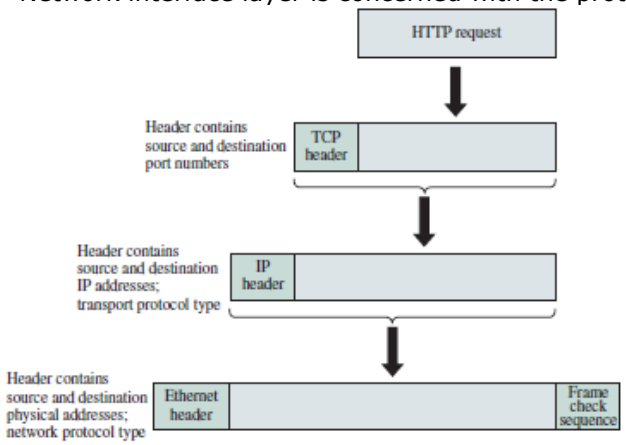

Figure 8.2: Encapsulation of PDUs

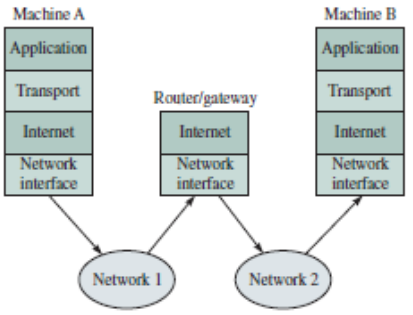

Figure 8.3: Internet & Network interface layer

INTERNET PROTOCOL

IP PACKET

- This contains a header part and a data part. Various fields in header are as follows:

1) Version:

This indicates version number used by the packet. Current version=4 (Figure 8.4).
Version 5 is used for a real-time stream protocol called ST2 and version 6 is used for IPv6.

2) Internet-Header-Length(IHL):

This specifies length of header. Without options field, IHL=5.

3) Type of Service:

This specifies priority of packet based on delay, throughput, reliability & cost requirements. 3 bits are assigned for priority levels and 4 bits for the specific requirement(i.e. delay, throughput, reliability & cost).

4) Total length:

This specifies number of bytes in the packet (including header and data).
Maximum length=65535 bytes.

5) Identification, flags and fragment offset:

These are used for fragmentation and reassembly.

6) Time-to-live(TTL):

This indicates amount of time(in seconds), the packet is allowed to remain in the network. If TTL becomes 0 before packet reaches destination, router discards packet and sends an error message back to the source.

7) Protocol:

This specifies upper-layer protocol that is to receive the packet at the destination-host. Examples of protocols include TCP(protocol=6) and UDP(protocol=17).

8) Header checksum:

This is used to verify integrity of header only. If the verification process fails, packet is discarded.

9) Source IP address and destination IP address:

These contain the addresses of source and destination hosts.

10) Options:

This is of variable length. This allows the packet to request special features such as security level, route to be taken by packet and timestamp at each router.

11>Padding:

This is used to make the header a multiple of 32-bit words.

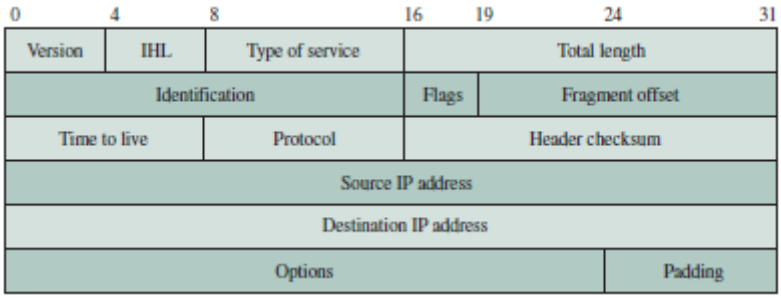

Figure 8.4:IP version 4 header

IP ADDRESS

- This is a numeric identifier assigned to each machine on an IP network.
- This consists of network ID(NID) and host ID(HID).
- HID identifies the network-connection to the host rather than the actual host.
  NID identifies the network to which the host is connected. All the hosts connected to the same network, have the same NID.
- HID is assigned by the network administrator at the local site.
  NID for an organization may be assigned by the ISP(Internet Service Provider).
- Class D addresses are used for multicast services (Figure 8.6).
- Multicast means a host sends message to a group of hosts simultaneously.
- IP addresses are usually written in dotted-decimal notation. The address is broken into four bytes.
  For example, an IP address of
  10000000 10000111 01000100 00000101
  is written as
  128.135.68.5

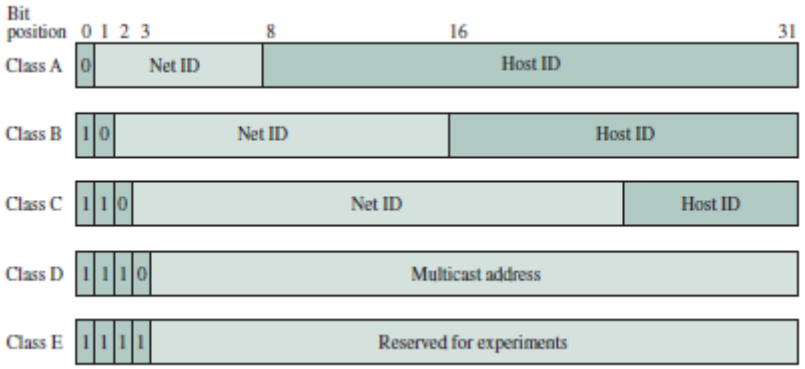

Figure 8.6: The five classes of IP addresses

SPECIAL PURPOSE ADDRESSES

- If HID contains all 1s, the packet is broadcast to all hosts on the network (Figure 8.6).
  If the NID contains all 1s, the packet is broadcast on the local network.
- A host ID that contains all 0s refers to the network specified by the NID rather than to a host.
- 127.X.Y.Y is used for loopback. When a host sends a packet with this address, the packet is returned to the host by the IP protocol software without transmitting it the physical network.
- A set of specific ranges of IP addresses have been set aside for use in private networks. These addresses are considered unregistered and routers in the Internet must discard packets with these addresses.
  Range 1: 10.0.0.0 to 10.255.255.255
  Range 2: 172.16.0.0 to 172.31.255.255
  Range 3: 192.168.0.0 to 192.168.255.255

SUBNET ADDRESSING

• To allow a single network address to span multiple physical networks is called *subnet addressing*
• The beauty of the subnet addressing scheme is that it is oblivious to the network outside the organization (Figure 8.7).
• Inside the organization the local network administrator is free to choose any combination of lengths for the subnet and host ID fields.
• For the subnet address scheme to work, every machine on the network must know which part of the host address will be used as the subnet address. This is accomplished by assigning each machine a subnet mask.
• The 1's in the subnet mask represent the positions that refer to the network or subnet addresses. The 0's represent the positions that refer to the host part of the address.

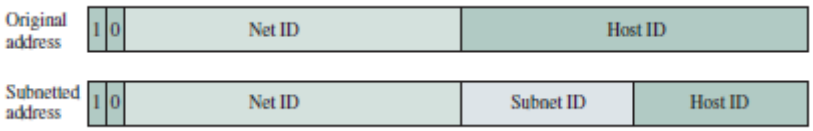

Figure 8.7: Introducing another hierarchical level through subnet addressing

Question: If a packet with a destination IP address of 150.100.12.176 arrives at site from the outside network, which subnet should a router forward this packet to? Assume subnet mask is 255.255.255.128(Fig 8.8).

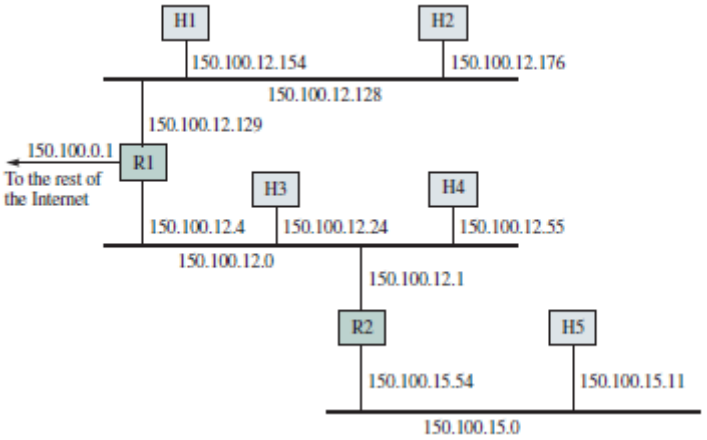

Figure 8.8: Example of address assignment with subnetting

Solution: The router can determine the subnet number by performing a binary AND between the subnet mask and the IP address.

IP address:	10010110	01100100	00001100	10110000	(150.100.12.176)
Subnet mask:	11111111	11111111	11111111	10000000	(255.255.255.128)
Subnet number:	10010110	01100100	00001100	10000000	(150.100.12.128)

This number(150.100.12.128) is used to forward the packet to the correct subnet work inside the organization.

IP ROUTING

• IP layer in the end-system hosts and in the routers work together to route packets from sources to destinations.
• IP layer in each host and router maintains a routing table.
• IP layer uses routing table to determine how to handle each packet.
• Consider the action of the originating host.
If its routing table indicates that the destination host is directly connected to the originating host by a link, then the packet is sent directly to the destination host using the appropriate network interface.
Otherwise, the routing table typically specifies that the packet is to be sent to a default router that is directly connected to the originating host.
• Consider the action of a router.
When a router receives a packet from one of the network interfaces, the router examines its routing table to see whether the packet is destined to itself, and if so, delivers the packet to the appropriate higher layer protocol.
If the destination IP address is not the router's own address, then the router determines the next-hop router and the associated network interface, and then forwards the packet.

CLASSLESS INTER DOMAIN ROUTING(CIDR)

• An arbitrary prefix length is used to indicate the network number known as CIDR.
• Using a CIDR notation, a prefix 205.100.0.0 of length 22 is written as 205.100.0.0/22. The /22 notation indicates that the network mask is 22 bits or 255.255.252.0
• The entries in a CIDR routing table contain a 32-bit IP address and a 32-bit mask.
• CIDR enables technique called supernetting to allow a single routing entry to cover a block of classful addresses.
• For example, instead of having four entries for a contiguous set of Class C addresses(e.g. 205.100.0.0, 205.100.1.0, 205.100.2.0 & 205.100.3.0), CIDR allows a single routing entry 205.100.0.0/22, which includes all IP addresses from 205.100.0.0 205.100.3.255
• To see the route aggregation process in more detail, we note that the original four Class C entries.

Class C address 205.100.0.0=11001101 01100100 00000000 00000000
Class C address 205.100.1.0=11001101 01100100 00000001 00000000
Class C address 205.100.2.0=11001101 01100100 00000010 00000000
Class C address 205.100.3.0=11001101 01100100 00000011 00000000

becomes

Mask 255.255.252.0 = 11111111 11111111 11111100 00000000
Supernet address 205.100.0.0 = 11001101 01100100 00000000 00000000

• The use of variable-length prefixes requires that the routing tables be searched to find the longest prefix match.
• For example, a routing table may contain entries for the above supernet 205.100.0.0/22 as well as for the even larger supernet 205.100.0.0/20. A packet with destination address 205.100.1.1 will match both of these entries, so the algorithm must select the match with the longest prefix.

ADDRESS RESOLUTION PROTOCOL(ARP)

• How does the host map the IP address to the MAC address?

Solution: Use ARP(Address Resolution Protocol) to find the destination MAC address for given IP address.

• Suppose H1 wants to send an IP packet to H3 but does not know the MAC address of H3 (Figure 8.9).

• H1 first broadcasts an ARP request packet asking the destination host(which is identified by H3's IP address) to reply.

• All host in the network receive the packet, but only the intended host(which is H3) responds to H1.

• The ARP response packet contains H3's MAC and IP addresses.

• For future use, H1 caches H3's IP and MAC addresses in its ARP table.

REVERSE ADDRESS RESOLUTION PROTOCOL(RARP)

• In some situations, a host may know its MAC address but not its IP address.

• For example, when a diskless computer is being bootstrapped, it can read the MAC address from its Ethernet card. However, its IP address is usually kept separately in a disk at the server.

• To obtain its IP address, the host first broadcasts an RARP request packet containing its MAC address on the network.

• All hosts on the network receive the packet, but only the server replies to the host by sending an RARP response packet containing the host's MAC and IP addresses.

• Limitation: The server must be located on the same physical network as the host.

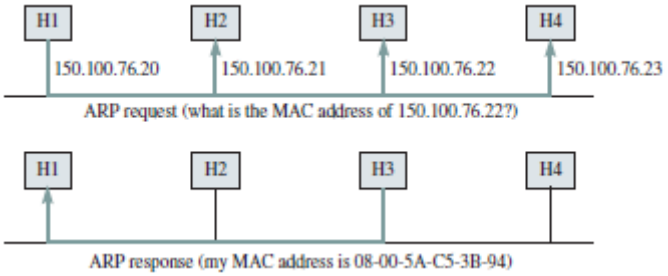

Figure 8.9: Address Resolution Protocol

FRAGMENTATION AND REASSEMBLY

• Fragmentation means the division of a packet into smaller units to accommodate a protocols MTU.
• Each network imposes a certain packet-size limitation on the packets that can be carried, called the maximum transmission unit(MTU). For example, MTU for Ethernet=1500 bytes and MTU for FDDI=4464 bytes (Figure 8.11).
• When IP wants send a packet that is larger than MTU of physical-network, IP breaks packet into smaller fragments.
• Each fragment is sent independently to the destination.
• Destination IP is responsible for reassembling the fragments into the original packet.
• To reassemble the fragments, the destination will wait until it has received all the fragments belonging to the same packet.
• Drawbacks: 1) Total overhead increases because each fragment must have a header.
2) Performance penalty: If one of the fragments is lost,
→ packet cannot be reassembled at the destination &
→ rest of the fragments have to be discarded. This process wastes transmission bandwidth.

IDENTIFICATION, FLAGS AND FRAGMENT-OFFSET

• Three fields in the IP header(identification, flags and fragment offset) are used to manage fragmentation and reassembly.

1) Identification:

This is used to identify to which packet a particular fragment belongs to (so that fragments for different packets do not get mixed up).

To have a safe operation, the source host must not repeat the identification value of the packet destined to the same host until a sufficiently long period of time has passed.

2) Flag:

This has three bits. One unused bit, one "don't fragment(DF)" bit and one "more fragment(MF)" bit.

If DF=1, router should not fragment the packet.

If MF=1, there are some more fragments to come. If MF=0, this is last fragment.

3) Fragment offset:

This identifies location of a fragment in a packet.

The value measures the offset(in units of eight bytes) between the beginning of the packet to be fragmented and beginning of fragment. Thus, first fragment of a packet has an offset value of 0.

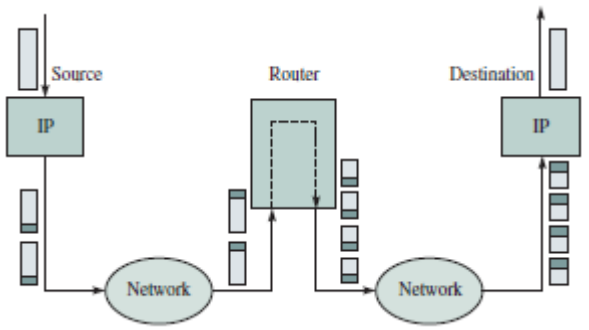

Figure 8.11: Packet Fragmentation

ICMP: ERROR AND CONTROL MESSAGES

This is used to handle error and other control messages (Figure 8.12). Various fields in header are as follows

1) Type:

This identifies the type of message.

2) Code:

This describes the purpose of the message.

For example, Type 3 = problem reaching the destinations Possible values for code field are

0=network unreachable
1=host unreachable
2=protocol unreachable
3= port unreachable
4=fragmentation needed and DF set

Type 11 = time exceeded problem. Possible values for code field are

0 = TTL value has been exceeded.
1 = fragment reassembly time has been exceeded.

3) Checksum:

This is used to detect errors in the ICMP message.

4) IP header plus original datagram:

This can be used for diagnostic purposes by matching the information in the ICMP message with the original data in the IP packet.

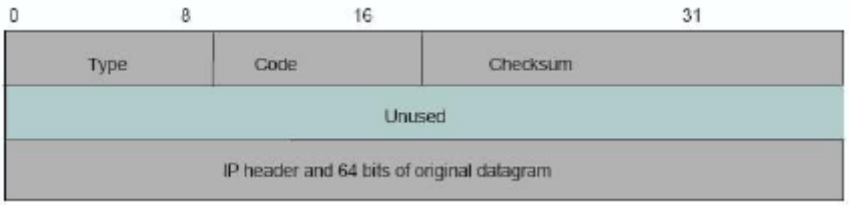

Figure 8.12: ICMP basic error message format

ECHO REQUEST AND ECHO REPLY

• When destination receives an echo request message from a source, the destination simply replies with a corresponding echo reply message back to the source (Figure 8.13).
• The echo request and echo reply messages are used in the PING program and are often used to determine whether a remote host is alive.
• Type=8 is used for echo request while type=0 for echo reply. Code =0 for both types.
• Sequence number: This is used to match the echo reply message with the corresponding echo request message.
• Identifier: This is used to differentiate different sessions using the echo services.
• Data: This is of variable length.

Figure 8.13: echo request and echo reply message format

IPv6

CHANGES FROM IPv4 TO IPv6

1) Longer address fields:
Length of address field is extended from 32 bits to 128 bits. The address space can support up to 3.4*1038 hosts.

2) Simplified header format:
Some of the header fields in IPv4 such as identification, flags and fragment offset do not appear in the IPv6 header.

3) Flexible support for options:
The options in IPv6 appear in optional extension headers that are encoded in a more efficient and flexible fashion than they are in IPv4.

4) Flow label capability:
IPv6 adds a "flow label" to identify a certain packet "flow" that requires a certain QoS.

5) Security:
IPv6 supports build-in authentication and confidentiality.

6) Large packets:
IPv6 supports payloads that are longer than 65 Kbytes called jumbo payloads.

7) Fragmentation at source only:
Routers do not perform packet fragmentation. If a packet needs to be fragmented, the source should check the minimum MTU along the path and perform the necessary fragmentation.

8) No checksum fields:
The checksum field has been removed to reduce packet processing time in a router. Packets carried by the physical network(such as Ethernet, token shop) are typically already checked. Furthermore, higher-layer protocols(such as TCP and UDP) also perform their own verification.

HEADER FORMAT

1) Version:
This specifies version number of protocol. For IPv6,version=6.

2) Traffic class:
This specifies priority of packet. This is used to support differential service (Figure 8.16).

3) Flow label:
This is used to identify QoS requested by packet. A flow is defined as "a sequence of packets sent from a particular source to a particular destination for which the source desires special handling by the interventing routers.

4) Payload length:
This indicates length of data (excluding header). Maximum length=65535 bytes.

5) Next header:
This identifies type of extension header that follows the basic header.

6) Hop limit:
This specifies number of hops the packet can travel before being dropped by a router.

7) Source address & destination address:
These identify source host and destination host respectively.

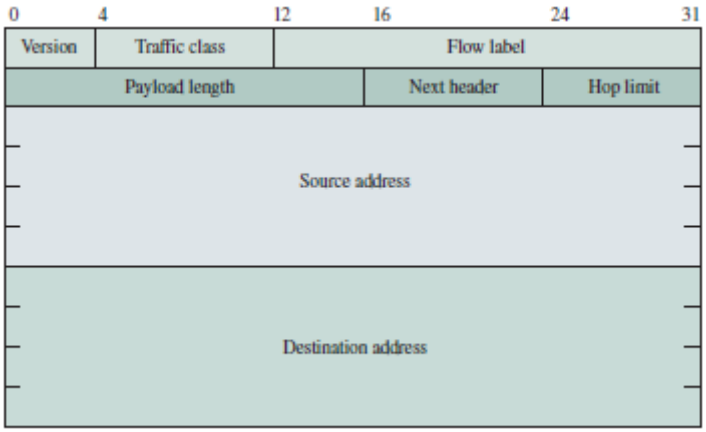

Figure 8.16:IPv6 basic header

NETWORK ADDRESSING

• IPv6 addresses are divided into three categories:

1) Unicast addresses: These identify a single network interface.
2) Multicast addresses: These identify a group of network interfaces, typically at different location. A packet will be sent to all network interfaces in the group.
3) Anycast addresses: These also identify a group of network interfaces. However, a packet will be sent to only one network interface in the group, usually nearest one.

SPECIAL PURPOSE ADDRESSES

• The address 0::0 is called the unspecified address and is never used as a destination address.
• The address ::1 is used for a loopback.
• IPv4 compatible addresses are needed during the transition period where an IPv6 packet needs to be "tunneled" across an IPv4 network.
• IP mapped addresses are used to indicate IPv4 hosts and routers that do not support IPv6.
• Provides based unicast addresses are identified by the prefix 010. These addresses will be mainly used by the Internet service providers to assign addresses to their subscribers. (Figure 8.18).

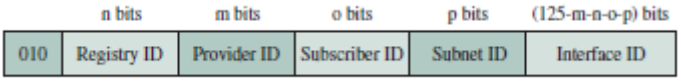

Figure 8.18: Provider based address format

EXTENSION HEADERS

• To support extra functionalities, IPv6 allows an arbitrary number of extension headers to be placed between the basic header and the payload (Figure 8.19).
• Extension headers act like options in IPv4 except former are encoded more efficiently and flexible.

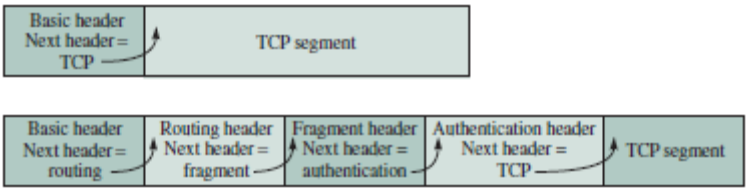

Figure 8.19:Daisy chain extension headers

LARGE PACKET

• Next header: This identifies type of header immediately following this header(Table 8.3).
• The value 194 defines a jumbo payload option (Figure 8.20).
• Payload length: This must be set to 0.
• Option length: This specifies size of jumbo payload length field (in bytes).
• Jumbo payload length: This specifies payload size. Maximum payload size=232 bytes.

Header code	Header type
0	Hop-by-hop options header
43	Routing header
44	Fragment header
51	Authentication header
52	Encapsulating security payload header
60	Destination options header

Table 8.3: Extension headers

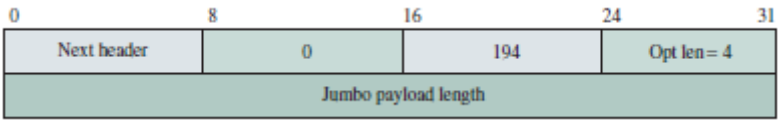

Figure 8.20:Extension header for jumbo packet

FRAGMENTATION

• A source can find the minimum MTU along the path from the source to the destination by performing a "path MTU discovery" procedure (Figure 8.21).
• Advantage of doing fragmentation at the source only is that routers can process packets faster.
• Disadvantage: Path between a source and a destination must remain reasonably static.

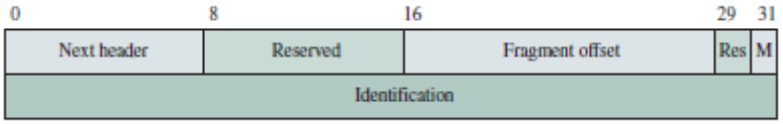

Figure 8.21: Fragment extension header

SOURCE ROUTING

• Header length: This specifies the length of the routing extension header (in units of 64 bits), excluding first 64 bits (Figure 8.22).
• Currently, only type=0 is specified.
• Segment left: This identifies the number of route segments remaining before the destination is reached. Maximum value=23.
• Each bit in the strict/loose bit mask indicates whether the next destination address must be followed strictly(if the bit is set to 1) or loosely(if the bit is set to 0).

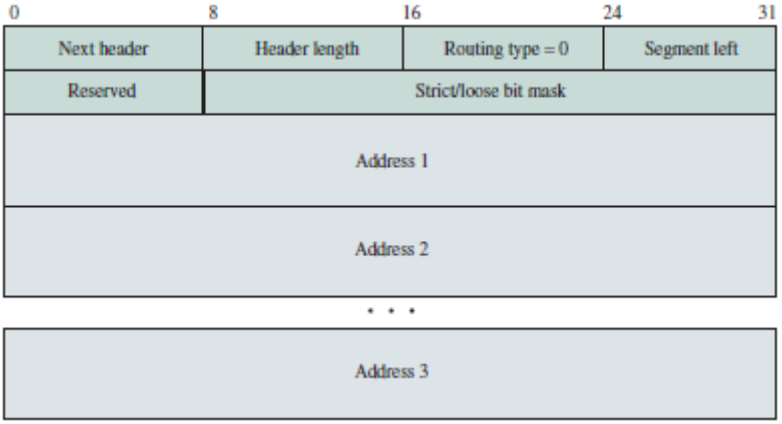

Figure 8.22:Routing extension header

MIGRATION ISSUES FROM IPV4 TO IPV6

• A tunnel is a path created between two nodes so that the tunnel appears as a single link to the user(Figure 8.23).
• An IPv4 tunnel allows IPv6 packets to be forwarded across an IPv4 network without the IPv6 user having to worry about how packets are actually forwarded in the IPv4 network.
• A tunnel is typically realized by encapsulating each user packet in another packet that can be forwarded along the tunnel.

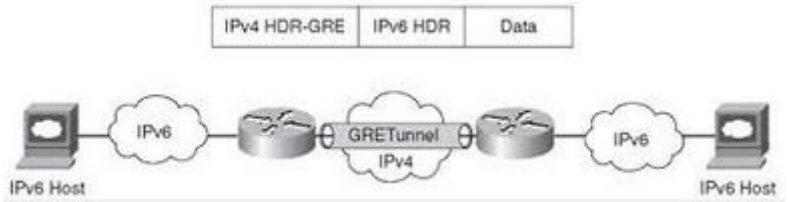

Figure 8.23: Tunneling :a)IPv6 over IPv6 tunnel; b)IPv6 virtual topology

USER DATAGRAM PROTOCOL (UDP)

- This is an unreliable, connectionless transport layer protocol.
- This provides only two additional services beyond IP: demultiplexing and error checking on data.
- This can optionally check the integrity of the entire datagram.
- Applications that do not require zero packet loss such as in packet voice systems are well suited to UDP.
- Applications that use UDP include DNS, SNMP, RTP & TFTP.

UDP Format

- The destination port allows the UDP module to demultiplex datagrams to the correct application in a given host.
- The source port identifies the particular application in the source host to receive replies.
- The length field indicates the number of bytes in datagram(including header and eat).
- Checksum field detects errors in the datagram and its use is optional.
- Checksum computation procedure is similar to that in computing IP checksum except for 2 new twists.
    1) First, if the length of the datagram is not a multiple of 16 bits, the datagram will be padded out with 0s to make it a multiple of 16bits.
    2) Second, UDP adds a pseudoheader to the beginning of the datagram when performing the checksum computation.
- The pseudoheader is also created by the source and destination hosts only during the checksum computation and is not transmitted.

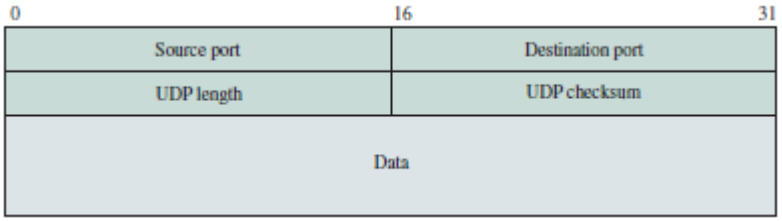

Figure 8.24: UDP datagram

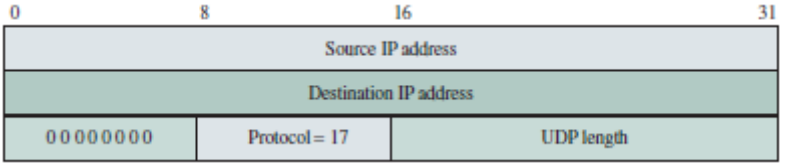

Figure 8.25:UDP pseudoheader

UNIT 5: APPLICATIONS AND NETWORK MANAGEMENT

APPLICATION LAYER

• This provides network services to user applications,
• This provides services such as email, remote access to computers, file transfer, and web etc.
• This has its own software dependencies, i.e. when a new application is developed, its software must be able to run on multiple machines.

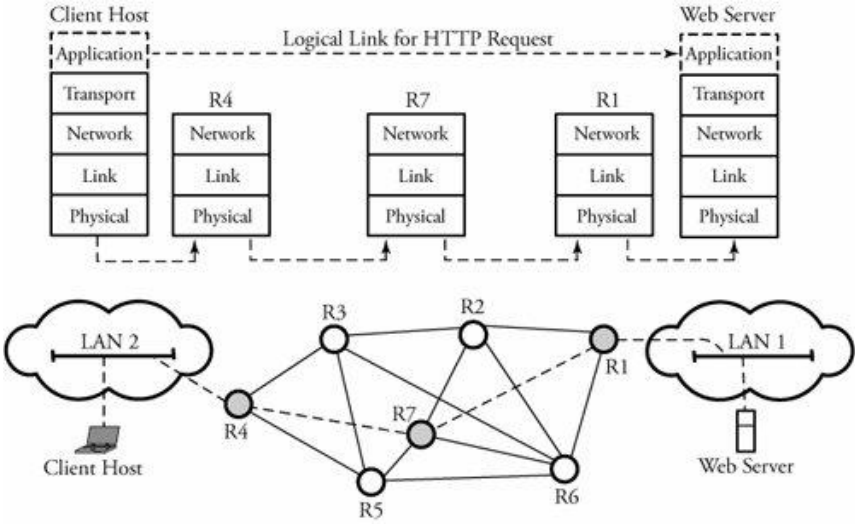

Figure 9.1. Web communication between two end systems

CLIENT & SERVER MODEL

• This provides specific computational services to multiple machines (Figure 9.1).
• A client-host requests services from a server-host.

DNS (DOMAIN NAME SYSTEM)

• This is a distributed hierarchical and global directory that translates domain names into numerical IP address and vice versa.
• This is an application-layer protocol.
• This is a critical infrastructure, and all hosts contact DNS to access servers and start connections.
• This can run over either UDP or TCP. However, running over UDP is preferred, since a fast response is required.
• Functions of a DNS server are
    1) Finding the address of a particular host.
    2) Mapping IP addresses to host names.
    3) Finding an alias for the real name of a host.
    4) Finding the host type and the operating-system information.
    5) Naming a host that processes incoming mail for the designated target.
    6) Delegating a subtree of server names to another server.
    7) Denoting the start of the subtree that contains cache and configuration parameters, and giving corresponding addresses.
• Every ISP(Internet Service Provider) has a DNS server.
• All hosts contact DNS servers when they initiate connections.
• A host sends UDP queries to a DNS server.
• DNS server either replies or directs the queries to smarter servers.

DOMAIN NAME SPACE

- Any entity in the internet is uniquely identified by an IP address.
- An IP address can also be assigned a domain name.
- A domain name is a sequence of labels separated by dots.
- A label is a string of characters.
- Domain names are defined in a tree-based structure with the root at the top (Figure 9.2).
- A tree is a structured with a maximum of 128 levels, starting at level 0(root).
- Each level consists of nodes
- A node on a tree is identified by a label
- A label can be of length up to 63 characters.

The root label has empty string.

The last label of a domain name expresses the type of organization.

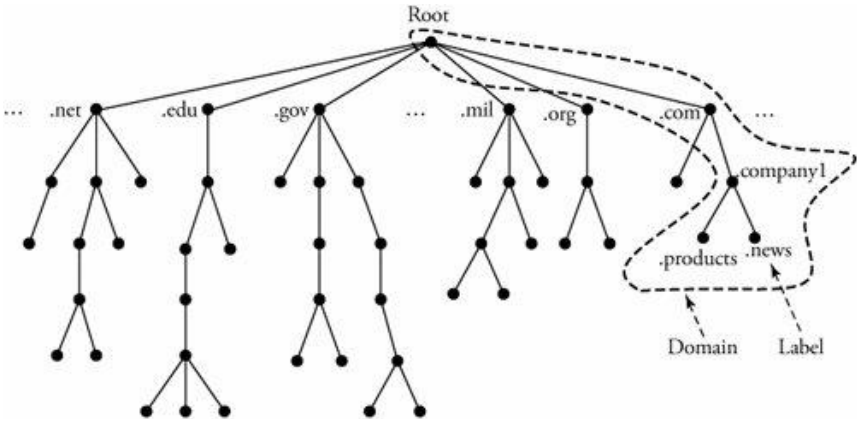

Figure 9.2. Hierarchy of domain name space, labels, and domain names

DOMAIN NAME SERVER (DNS SERVER)

- The domain name space is divided into subdomains, and each subdomain is assigned a domain-name-server(DNS Server).
- A DNS Server has a database consisting of all the information for every node under that domain (Figure 9.3).
- The root-server supervises the entire domain name space.
- A root-server typically keeps references only to servers over which it has authority.

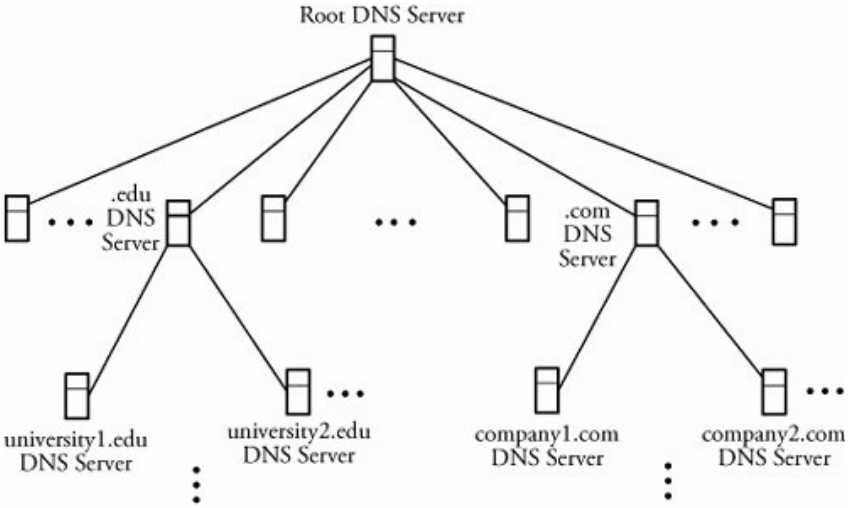

Figure 9.3. Hierarchy of DNS domain name servers

NAME/ADDRESS MAPPING

- DNS operates based on the client/server application.
- Each host sends its request to the closest DNS server.
- The server finds and sends the requested information to the host.
- If the requested information is not found, the server either
  → delegates the request to other servers or
  → asks them to provide the information.
- Mapping can be of either recursive or iterative.

Recursive Mapping

1) Client-host sends its request to the closest DNS server (Figure 9.4).
2) DNS server is responsible for finding the answer recursively.
3) The local DNS server of the requested place is called the authoritative server.

Iterative Mapping

1) Client-host sends its request to the closest DNS server (Figure 9.5).
2) If the DNS server does not have the name to provide, it returns to the client host.
3) The host must then repeat the query to the next DNS server that may be able to provide the name. This continues until the host succeeds in obtaining the name.

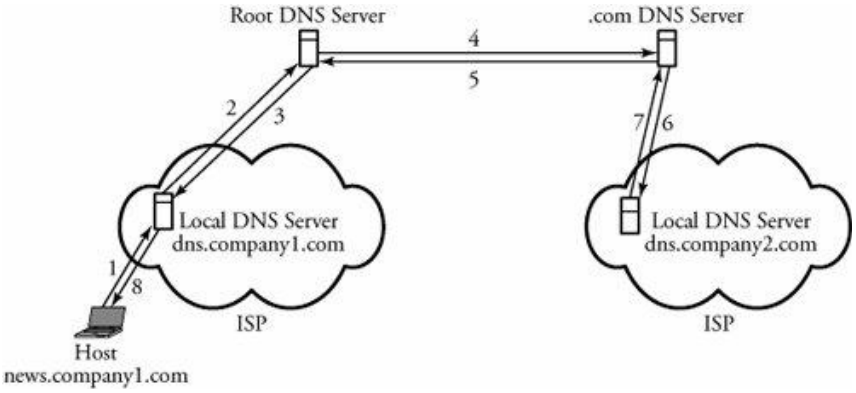

Figure 9.4. Recursive mapping

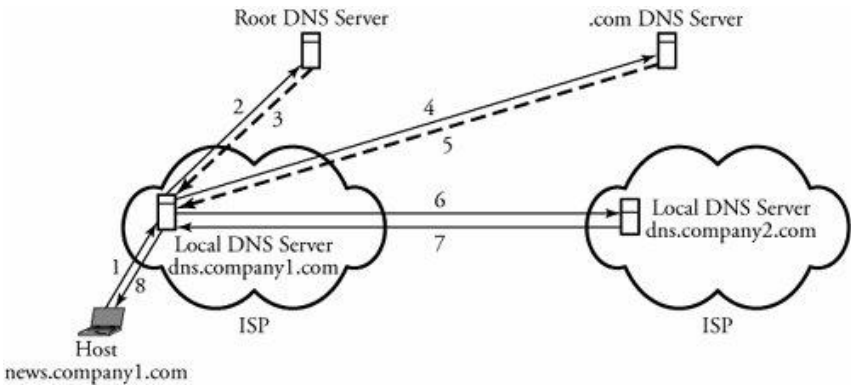

Figure 9.5. Iterative mapping

DNS MESSAGE FORMAT

- DNS communication is made possible through 2 messages: query and reply (Figure 9.6).
- The query message consists of a header and a question message only.
  Whereas the reply message consists of a header and 4 message fields: question, answer, authority and additional information.
- Following are the fields of these messages:

Identification
>This is used to match the reply with the query.

Flags
>This represents the type of the message, such as whether mapping is recursive or iterative.

Number of questions
>This indicates how many queries are in the question portion of the message.

Number of answers
>This shows how many answers are in the answer field.

Number of authoritative records
>This consists of the number of authoritative records in the authority portion of a reply message.

Number of additional records
> These are in the additional information portion of a reply message.

Question
>This contains one or more questions.

Answer
>This consists of one or more replies from a DNS server to the corresponding client.

Authority
> This provides the domain name information about one or more authoritative servers.

Additional information
>This contains other information such as the IP address of the authoritative server.

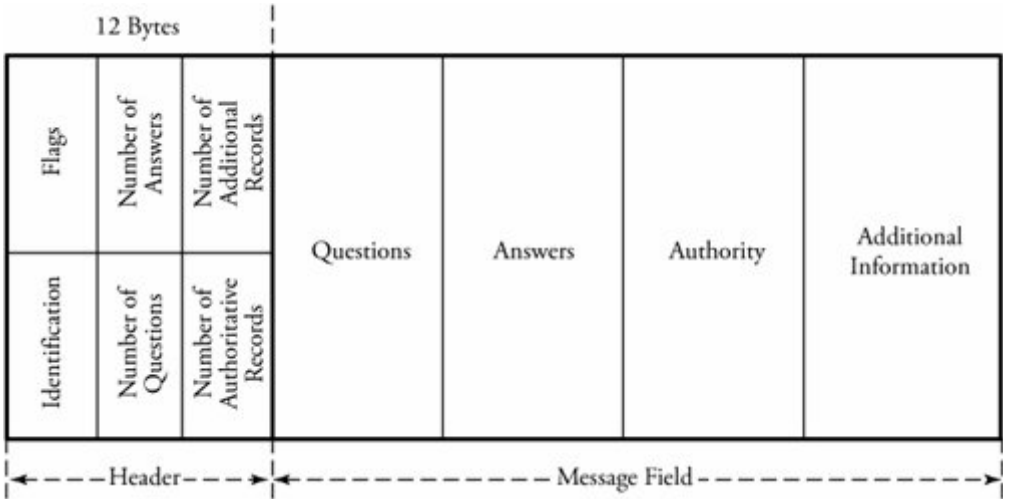

Figure 9.6. DNS message format

REMOTE LOGIN PROTOCOLS

• Using client/server model, a user can establish a session on the remote-machine and then run its applications. This application is known as remote login.
• For example, an employee working at home can log into his work-server to access application programs for doing a project.
• Two remote login protocols are TELNET & SSH.

TELNET

• This is a TCP/IP standard for establishing a connection to a remote-machine.
• This allows a user to log into a remote-machine across the Internet.
• This makes a TCP connection and then passes the detail of the application from the user to the remote-machine.

Logging to Remote Servers

• TELNET has the following properties:
1) Client-programs are built to use the standard client/server interfaces without knowing the details of server-programs.
2) A client and a server can negotiate data format options.
3) Once a connection is established, both ends of the connection are treated symmetrically.
• When a user logs into a remote-server, the client's terminal-driver accepts the keystrokes and interprets them as characters by its operating-system.
• Characters are typically transformed to a universal character set called NVT (network virtual terminal).
• The client then establishes a TCP connection to the server.
• Texts in the NVT formal are transmitted using a TCP session and are delivered to the operating-system of the remote-server.
• The server converts the characters back from NVT to the local client machine's format.

SECURE SHELL (SSH) PROTOCOL

• This is based on UNIX programs.
• This uses TCP for communications.
• This is more powerful and flexible than TELNET.
• This allows the user to more easily execute a single command on a remote client.
• This has the following advantages over TELNET
1) Security: SSH provides a secure communication by encrypting and authenticating messages. Security is implemented by using public-key encryption.
2) Multiplexing: SSH provides several additional data transfers over the same connection by multiplexing multiple channels.
• SSH packets contains following fields (Figure 9.7).
1) Length: This indicates the size of the packet
2) Padding: This causes an intrusion to be more difficult.
3) Type: This identifies the type of message.
3) CRC: This is an error detection field.

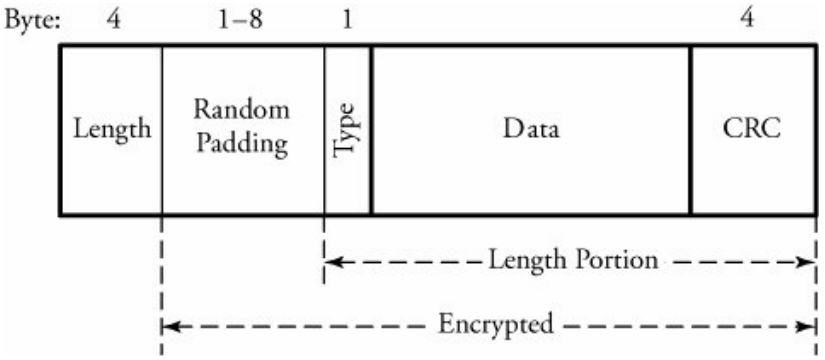

Figure 9.7. SSH packet format

SMTP (SIMPLE MAIL TRANSFER PROTOCOL) & EMAIL (ELECTRONIC MAIL)

- SMTP transfers email from the mail-server of a source to the mail-servers of destinations.
- A user-mailbox is a space in the mail-server allocated to the user to keep its email.

Begin SMTP between two users

1) User-1 provides user-2's email address(user2@organization.com) and composes its message(Figure 9.8).
2) User-1 sends the message to its mail-server(isp.com)
3) Server isp.com places the message in its queue.
4) SMTP on user 1's mail-server
   → notices the message in the queue and
   →opens a TCP connection with the organization mail-server (organization.com)
5) Initial SMTP handshaking takes place between the two servers.
6) The message is sent to organization.com's mail-server.
7) User-2's mail-server receives the message and then puts it in user-2's mailbox.

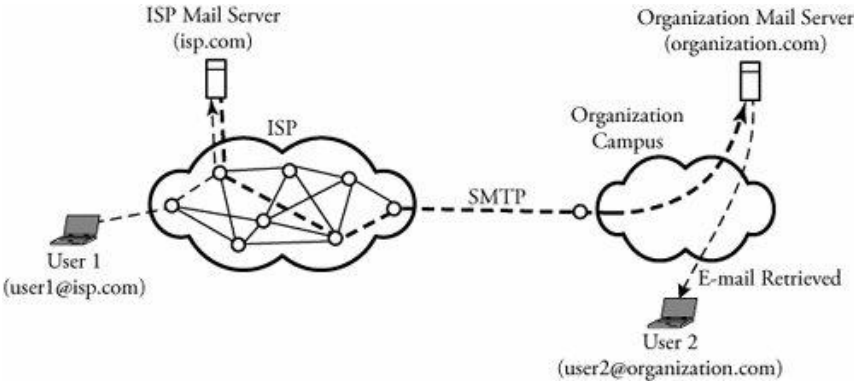

Figure 9.8. Two users exchanging e-mail through SMTP

FILE TRANSFER PROTOCOLS

FTP (File Transfer Protocol)

- FTP is used to transfer files from one host to another host over the internet.
- TELNET provides broader access to a user, whereas FTP allows access only to certain files.

Begin File Transfer Protocol

1) A user requests a connection to a remote-server.
2) The user waits for an acknowledgement.
3) Once connected, the user must enter a user ID, followed by a password.
4) The connection is established over a TCP session.
5) The desired file is transferred.
6) The user closes the FTP connection.

SCP (Secure Copy Protocol)

- This is similar to TELNET but is secure.
- A number of encryption and authentication features are incorporated.
- This cannot handle file transfer between machines of significantly different architectures.

WORLD WIDE WEB (WWW) AND HTTP (HYPER TEXT TRANSFER PROTOCOL)

- Web is a global network of servers.
- The servers are linked by a common protocol such as http, ftp & so on.
- When a client-host requests an object, a web-server responds by sending the requested-object.
- A browser is a user-agent which displays the requested web-page.
- HTTP transfers web-page at the application layer.
- HTTP uses TCP rather than UDP, since reliable delivery of web-pages with text is important.
- The TCP connection-establishment delay is one of the main contributing delay factors associated with downloading web-documents.
- Web-page consists of HTML(Hypertext Markup Language) files that can be addressed by a single URL(uniform resource locator).
- A URL is a global address of an HTML document and has two parts.
    1) First part indicates what protocol is used, and
    2) Second part determines the IP address of the associated resource.

WEB CACHING (PROXY SERVER)

- An HTTP request from a user is first directed to the web-cache.
- The web-cache must contain updated-copies of all objects in its defined proximity.
- Two reasons for web caching:
    1) To reduce the response-time for a user-request.
    2) To reduce traffic on an organization's access link to the Internet

Begin Web Caching Algorithm

1) The user-browser makes a TCP connection to the web-cache (Figure 9.9).
2) The user-browser transmits its HTTP request to the web-cache.
3) If web-cache has a copy of the requested-object,
    i) web-cache forwards the object to the user-browser
    Otherwise,
    ii) web-cache establishes a TCP connection to the requested-server and asks for the object. Once it receives the object, the web-cache stores a copy of it and forwards another copy to the user-browser.

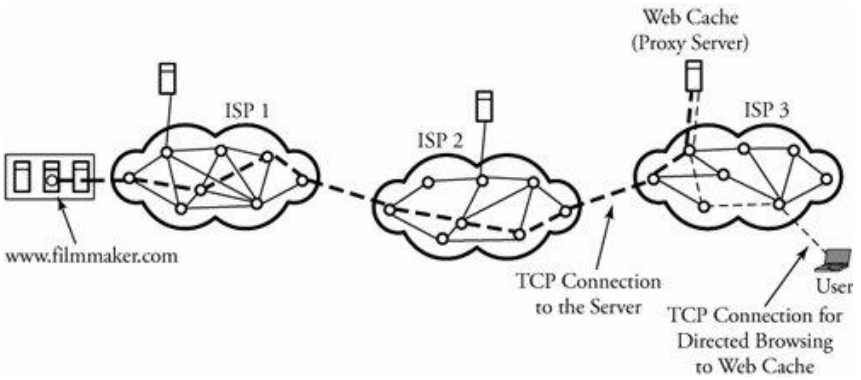

Figure 9.9. A user's browser requesting an object through the Web cache

NETWORK MANAGEMENT

• The purpose of network management is
→ to monitor, test and analyze the hardware, software and human elements of a network and
→ then to configure & control those elements to meet the operational performance requirements of the network(Figure 9..10).

• Network management tasks can be characterized as follows

i) QoS and performance management: A network-administrator periodically
→ monitors & analyzes routers, hosts and utilization of links and
→ then redirect traffic-flow to avoid any overloaded spots.

ii) Network failure management: Any fault in a network, such as link, host or router hardware or software outages, must be detected, located and responded to by the network.

iii) Configuration management: This task involves
→ tracking all the devices under management and
→ ensuring that all devices are connected and operate properly.

iv) Security management: This task is handled through firewall which can monitor & control access points.

v) Billing & accounting management: The network administrator
→ issues all billing & charges to users and
→ specifies user access or restrictions to network resources

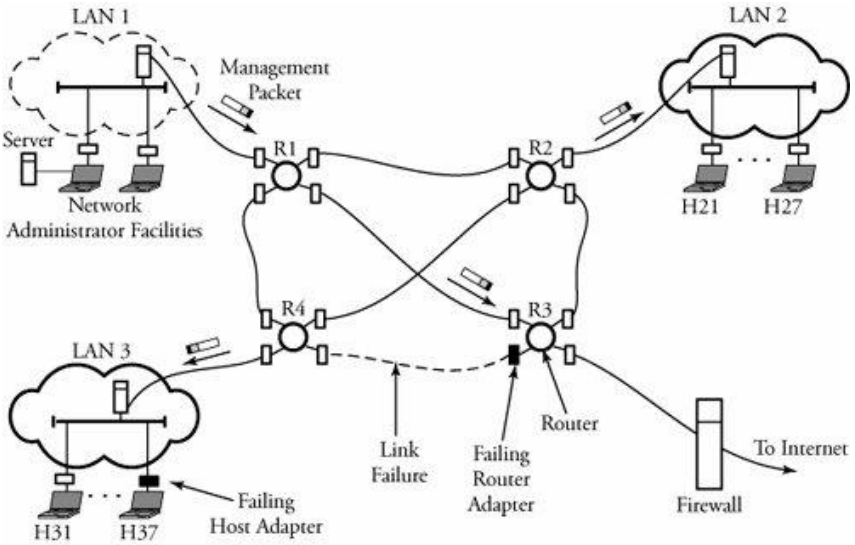

Figure 9.10. Simple network management in a scenario of LANs connecting to the Internet

ELEMENTS OF NETWORK MANAGEMENT

• Network management has three main components:
i) Managing-center consists of the network-administrator and his facilities.
ii) Managed-device is the network-equipment that is controlled by the managing-center.
The managed-device includes hub, bridge, router, server, printer or modem.
iii) Network-management-protocol is a policy between the managing-center and the managed-devices.

• An agent is a managed-device such as router, hub or bridge.
Whereas a manager is a network administrative device, such as a management host.

STRUCTURE OF MANAGEMENT INFORMATION (SMI)

- This is used
  - → to define the rules for naming objects and
  - → to encode objects in a managed network center
- For ex, Integer32 means a 32-bit integer with a value between -231 and -231-1.
- This also provides higher-level language constructs. These constructs typically specify the data type, status and semantics of managed-objects containing the management data.
- For ex, the STATUS clause specifies whether the object definition is current or obsolete.

MANAGEMENT INFORMATION BASE (MIB)

- This is an information storage medium.
- This contains managed-objects which reflects the current status of the network.
- This also shows relationships among managed-objects.
- Objects
  - → are organized in a hierarchical manner and
  - → are identified by the ASN.1 object definition language (ASN.1=Abstract Syntax Notation One).
- The hierarchy of object names(known as ASN.1 object identifier) is an object identifier tree in which each branch has both a name and a number(Figure 9.11).
- Network management can then identify an object by a sequence of names or numbers from the root to that object.
- On the root of the object identifier hierarchy are three entries: ISO(International Standardization Organization), ITU-T(International Telecommunication Union Telecommunication) & ISO-ITU-T.
- For ex, the organization(3) branch is labeled sequentially from the root as 1.3

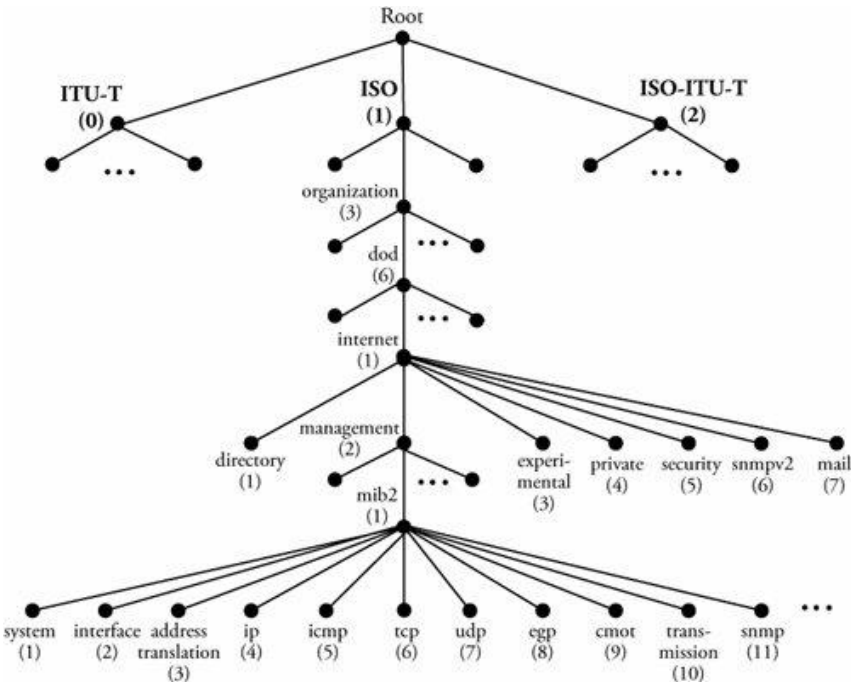

Figure 9.11. ASN.1 object identifier organized hierarchically

SNMP (SIMPLE NETWORK MANAGEMENT PROTOCOL)

• The purpose of network management is
→ to monitor, test and analyze the hardware, software and human elements of a network and
→ then to configure & control those elements to meet operational performance requirements of network

• This runs on top of UDP and uses client/server configuration.

• PDUs(Protocol Data Unit) are carried in the payload of a UDP datagram, and so its delivery to a destination is not guaranteed.

• Managed-devices (such as routers and hosts) are objects, and each object has a formal ASN.1 definition.

• The task of SNMP is to transport MIB information among managing-centers and agents executing on behalf of managing-centers.

• For each managed MIB object, an SNMP request is used to retrieve (or change) it's associated value.

• If an unsolicited message is received by an agent(or when an interface/device goes down),the protocol can also inform the managing-center.

• SNMPv2 has seven PDU's(or messages) as follows.
1) GetRequest: This is used to obtain the value of a MIB object.
2) GetNextRequest: This is used to obtain the next value of a MIB object.
3) GetBulkRequest : This is used to get multiple values, equivalent to multiple GetRequests but without using multiple overheads.
4) InformRequest: This is a manager-to-manager message that two communicating management centers are remote to each other.
5) SetRequest : This is used to set the value of a MIB object.
6) Response: is a reply message to a request-type PDU.
7) Trap: This notifies a managing-center that an unexpected event has occurred.

• Get or Set PDU format has following fields (Figure 9.12).
1) PDU type: This indicates one of the seven PDU types.
2) Request ID: This is used to verify the response of a request.
3) Error status: This indicates types of errors reported by an agent.
4) Error index: This indicates to a network administrator which name has caused an error.

• Trap PDU format has following fields
1) Enterprise: This is for use in multiple networks
2) Timestamp: This is used for measuring up time.
3) Agent address: This indicates address of the managed agent is included in the PDU header.

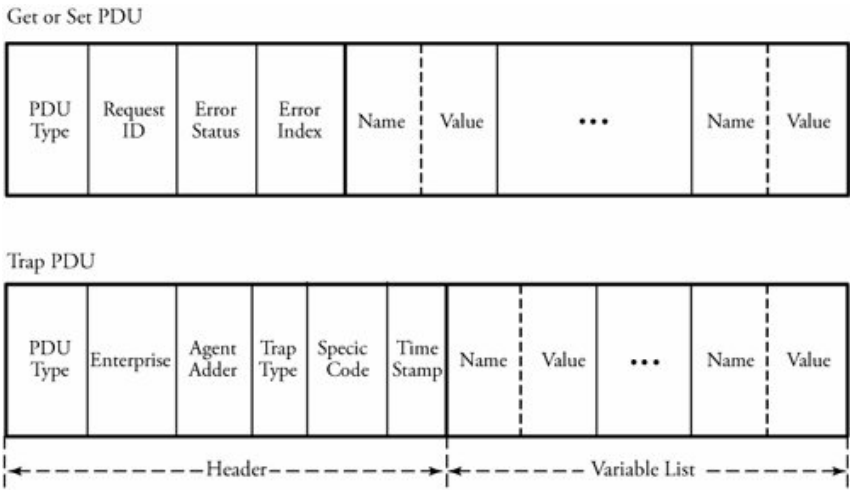

Figure 9.12. SNMP PDU format

UNIT 5(CONT.): NETWORK SECURITY

OVERVIEW OF NETWORK SECURITY

• Network security is required by the users to communicate on the network.
• If medium is insecure then an intruder may intercept, read and modify the transmitted-data from sender to receiver.

ELEMENTS OF NETWORK SECURITY

1) Confidentiality: Information should be available only to those who have rightful access to it
2) Authenticity and integrity: The sender of a message and the message itself should be verified at the receiving-point (Fig10.1).

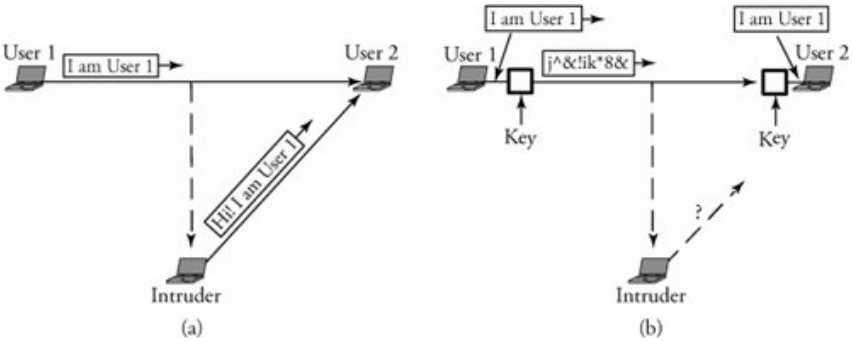

Figure 10.1. (a) Message content and sender identity falsified by intruder; (b) a method of applied security

• In figure 10.1a, user 1 sends a message ("i am user 1") to user 2.
Since the network lacks any security system, an intruder can receive the message and change its content to a different message ("hi i am user 1") and send it to user 2.
User 2 may not know that this falsified message is really from user 1(authentication).
• In figure 10.1b, a security block is added to each side of the communication, and a secret key that only users 1 and 2 would know about is included.
Therefore, the message is changed to a form that cannot be altered by the intruder.

THREATS TO NETWORK SECURITY

• Internet infrastructure attacks are broadly classified into 4 categories

1) DNS hacking 2) Routing table poisoning
3) Packet mistreatment 4) Denial of Service (DOS)

DNS HACKING ATTACKS

• DNS server is a distributed hierarchical and global directory that translates domain names into numerical IP address.
• DNS is a critical infrastructure, and all hosts contact DNS to access servers and start connections.
• Name-resolution services in the modern Internet environment are essential for email transmission, navigation to web sites, or data transfer. Thus, an attack on DNS can potentially affect a large portion of the Internet.
• A DNS hacking attack can appear in any of the following forms

Masquerading Attack

• The attacker poses as a trusted entity and obtains all the secret information.
• The attacker
→ can stop any message from being transmitted further or
→ can change the content or redirect the packet to bogus servers. This action is also known as a middle-man attack.

Domain Highjacking Attack

• Whenever a user enters a domain address, he is forced to enter into the attacker's Web site.

Information Leakage Attack

• The attacker
→ sends a query to all hosts
→ identifies which IP addresses are not used
→ uses those IP address to make other types of attacks

Information-Level Attack(Cache Poisoning)

• This forces a server to correspond with other than the correct answer.
• The hacker
→ tricks a remote name-servers into caching the answer for a third-party domain by providing malicious information.and
→ redirects traffic to a preselected site.

ROUTING TABLE POISONING

• This is the undesired modification of routing tables.
• This results in a lower throughput of the network.
• Two types of attacks are: i)link attack and ii)router attack.

Link Attack

• This occurs when a hacker gets access to a link and thereby intercepts, interrupts or modifies routing messages.
• This act similarly on both the link-state and the distance-vector protocols.
• If an attacker succeeds in placing an attack in a link-state routing protocol, a router may
→ send incorrect updates about its neighbors or
→ remain silent even if the link state of its neighbor has changed

Router Attack

• This may affect the link-state protocol or even the distance-vector protocol.
• In link-state protocol, if routers are attacked, they become malicious. As a result, routers may
→ add a nonexisting link to a routing table
→ delete an existing link or
→ change the cost of a link.
• In the distance-vector protocol, an attacker may cause routers to send wrong updates about any node in the network, thereby misleading a router and resulting in network problems.

DOS ATTACKS (DENIAL OF SERVICE)

• This is a type of security breach that prohibits a user from accessing normally provided services.
• This can cost the target person a large amount of time and money.
• This affects the destination rather than a data-packet or router.
• They take important servers out of action for few hours, thereby denying service to all users.
• Two types of attacks are:

1) *Single-source:* An attacker sends a large number of packets to a target system to overwhelm & disable it
2) *Distributed:* A large number of hosts are used to flood unwanted traffic to a single target. The target cannot then be accessible to other users in the network.

PACKET MISTREATMENT ATTACKS

• This can occur during any data transmission.
• A hacker may capture certain data packets and mistreat them.
• The attack may result in
→ congestion
→ lowering throughput &
→ DOS attacks
• Link-attack causes interruption, modification or replication of data packets.
Whereas, a router-attack can misroute all packets and may result in congestion or DOS
• Following are some examples:

Interruption

• If an attacker intercepts packets, they may not be allowed to be propagated to their destinations.

Modification

• Attackers may succeed in accessing the content of a packet. They can then
→ change the address of the packet or
→ change the data of the packet
• This kind of attack can be detected by digital signature mechanism.

Replication

• An attacker may trap a packet and replay it.
• This kind of attack can be detected by using the sequence number for each packet.

Malicious Misrouting of Packets

• A hacker may attack a router and change its routing table, resulting in misrouting of data packets.

Ping of death

• An attacker may send a ping message, which is large and therefore must be fragmented for transport.
• The receiver then starts to reassemble the fragments as the ping fragments arrive.
• The total packet length becomes too large and might cause a system crash.

OVERVIEW OF SECURITY METHODS

• Common solutions that can protect computer communication networks from attacks are classified are cryptographic techniques or authentication techniques(verification).

CRYPTOGRAPHIC TECHNIQUES

• Cryptography is the process of transforming a piece of information or message shared by two parties into some sort of code.
• The message is scrambled before transmission so that it is undetectable by outside watchers.
• The scrambled-message needs to be decoded at the receiving-end before any further processing.
• The main tool used to encrypt a message M is a secret-key K.
The fundamental operation used to encrypt a message is the exclusive-OR($\oplus$).
• Assume that we have one-bit M and a secret-bit K. A simple encryption is carried out using $M \oplus K$.
• To decrypt this message, the second party can detect M by performing the following operation:
$(M \oplus K) \oplus K = M$
• In *end-to-end encryption*, secret coding is carried out at both end systems (Figure 10.2).
In *link encryption*, all the traffic passing over that link is secured.
• Two types of encryption techniques are secret-key & public-key encryption
1) In *secret-key model*, both sender & receiver conventionally use same key for an encryption process.
In *public-key model*, a sender and a receiver each use a different key.
2) The public-key system
→ is more powerful than the secret key system &
→ provides better security and message privacy.
3) Drawbacks of public-key system:
→ slow speed
→ more complex computationally

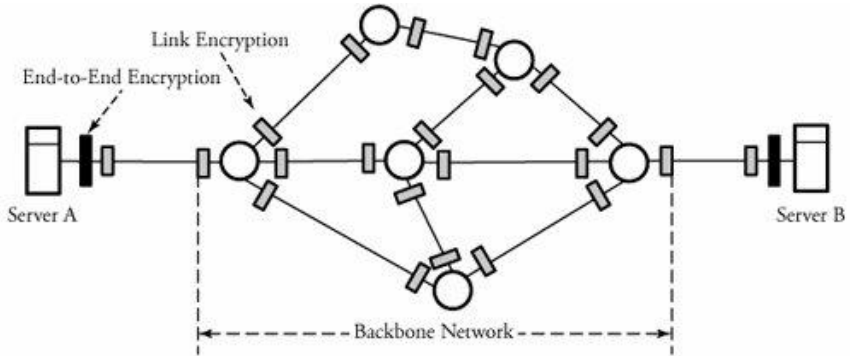

Figure 10.2. Overview of encryption points in a communication network

AUTHENTICATION TECHNIQUES

• Encryption methods offer the assurance of message confidentiality.
• A networking-system must be able to verify the authenticity of the message and the sender of the message. These forms of security techniques are known as authentication techniques.
• Authentication techniques are categorized as i) authentication with message digest and
ii) authentication with digital signature.

SECRET KEY ENCRYPTION PROTOCOLS

- This is also called as symmetric encryption or single-key encryption.
- Sender and receiver conventionally use the same key for an encryption process.
- This consist of
  - → an encryption-algorithm
  - → a key and
  - → a decryption-algorithm
- The encrypted-message is called ciphertext.
- Two popular protocols are: 1) DES (Data Encryption Standard)
  2) AES (Advanced Encryption Standard)
- A shared secret-key between a transmitter and a receiver is assigned at the transmitter and receiver points.
- At the receiving end, the encrypted information can be transformed back to the original data by using
  - → decryption algorithm and
  - → secret key

DES

- Plaintext messages are converted into 64-bit blocks & each block is encrypted using a key.
- The key length is 56 bits.
- This consists of 16 identical rounds of an operation (Figure 10.3).

Begin DES Algorithm

1) Initialize. Before round 1 begins, all 64 bits of the message and all 56 bits of the secret key are separately permuted(shuffled).

2) Each incoming 64-bit message is broken into two 32-bit halves denoted by L_i and R_i respectively.

3) The 56 bits of the key are also broken into two 28-halves, and each half is rotated one or two bit positions, depending on the round.

4) All 56 bits of the key are permuted, producing version k_i of the key on round i.

5) L_i and R_i are determined by

$L_i = R_{i-1}$

and

$R_i = L_{i-1} \oplus F(R_{i-1}, k_i)$

6) All 64 bits of a message are permuted.

Operation of function F()

- Out of 56 bits of k_i, function F( ) chooses 48 bits.
- The 32-bit R_{i-1} is expanded from 32 bits to 48 bits so that it can be combined with 48 bit k_i.
- F( ) also partitions the 48 bits of k_i into eight 6-bit chunks.
- The corresponding eight chunks of R_{i-1} and eight chunks of k_i are combined as follows

$R_{i=1} = R_{i-1} \oplus k_i$

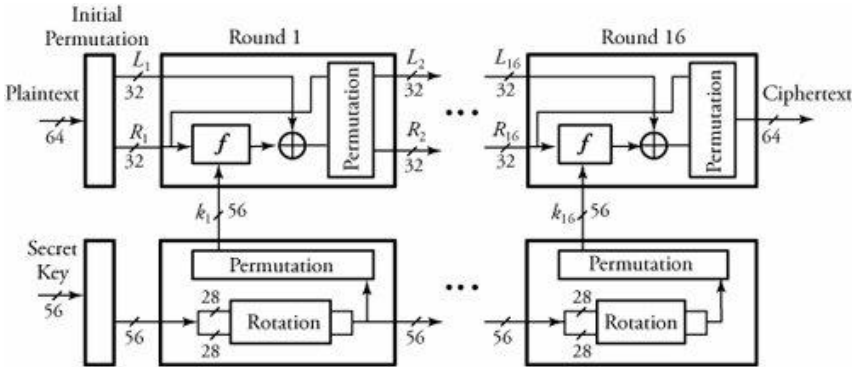

Figure 10.3. The Data Encryption Standard (DES)

AES

• This has a better security strength than DES (Figure 10.4).
• Message size=128-bit block
Key size=128,192 or 256 bit
Number of rounds= 10 to 14
• The plaintext is formed as 16 bytes m_0 through m_{15} and is fed into round 1 after an initialization stage.
• In this round, substitute-units(S) perform a byte-by-byte substitution of blocks.
• The ciphers move through a permutation-stage to shift rows to mix-columns.
• At the end of this round, all 16 blocks of ciphers are Exclusive-ORed with the 16 bytes of round 1 key $k_0(1)$ through $k_{15}(1)$.

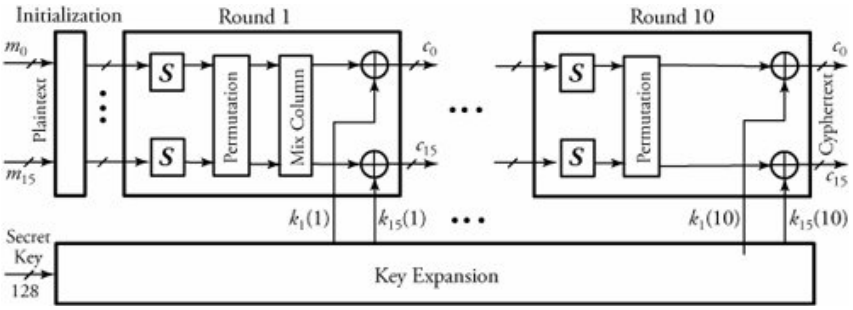

Figure 10.4. Overview of Advanced Encryption Standard (AES) protocol

PUBLIC KEY ENCRYPTION PROTOCOLS

• This is also called as asymmetric or two key encryption.
• A sender/receiver pair use different keys.
• This is based on mathematical functions rather than on substitution or permutation.
• Two popular protocols are: i)RSA protocol ii)Diffie-Hillman key-exchange protocol.
• Either of the two related keys can be used for encryption; the other one for decryption.
• Each system publishes its encryption key by placing it in a public-register & sorts out key as public one.
The companion key is kept private.
• If A wishes to send a message to B, A encrypts the message by using B's public key.
At receiving end, B decrypts the message by using its private key.
No other recipients can decrypt the message, since only B knows its private key.
• The public-key system
→ is more powerful than the secret key system &
→ provides better security and message privacy.
• Drawbacks of public-key system:
→ slow speed
→ more complex computationally

RSA ALGORITHM

- Assume that a plaintext m must be encrypted to a ciphertext c.
- This has three phases: key generation, encryption and decryption.

Key Generation Algorithm

1) Choose two prime numbers a and b and compute n=a.b
2) *Find x*. Select encryption-key x such that x and (a-1)(b-1) are relatively prime.
3) *Find y.* Calculate decryption-key y.

$$x\ y \bmod (a-1)(b-1) = 1$$

4) At this point, a and b can be discarded.
5) The public key = {x, n}
6) The private key = {y, n}

Encryption

1) Both sender and receiver must know the value of n.
2) The sender knows the value of x and only the receiver knows the value of y.
3) Ciphertext c is constructed by

$c=m^x \bmod n$

Decryption

1) Given the ciphertext c, the plaintext m is extracted by

$m=c^y \bmod n$.

DIFFIE-HILLMAN KEY-EXCHANGE PROTOCOL

- Two end users can agree on a shared secret-code without any information shared in advance.
- This protocol is normally used for VPN(virtual private network).
- Assume that user-1 wishes to communicate with user-2.

Key Generation Algorithm

1) User-1
   → selects a prime number 'a', random integer number 'x_1',and a generator 'g'
   → creates 'y_1' such that

$$y_1 = g^{x1} \bmod a$$

2) User-2
   → performs the same function and
   → creates y_2 such that

$$y_2 = g^{x2} \bmod a$$

3) User-1 then sends y1 to user-2. Now, user-1 forms its key k_1 using the information its partner sent as

$$k_1 = y_2^{x1} \bmod a$$

4) User-2 forms its key ka using the information its partner send it as

$$k_2 = y_1^{x2} \bmod a$$

5) The two keys k_1 and k_2 are equal. The two users can now encrypt their messages, each using its own key

AUTHENTICATION

- Message-authentication verifies the authenticity of both the message-sender and the message-content.
- Message-sender is authenticated through implementation of a digital signature.
  Message-content is authenticated through implementation of a hash function and encryption of the resulting message-digest.
- Hash-function is used to produce a "fingerprint" of a message.
- The hash-value is added at the end of message before transmission.
- The receiver re-computes the hash-value from the received message and compares it to the received hash-value.
- If the two hash-values are the same, the message was not altered during transmission.
- Once a hash-function is applied on a message m, the result is known as a message-digest h(m).
- The hash-function has the following properties
  1) Unlike the encryption-algorithm, the authentication algorithm is not required to be reversible.
  2) Given a message-digest h(m),it is computationally infeasible to find m.
  3) This is computationally infeasible to find two different messages m_1 and m_2 such that $h(m_1)=h(m_2)$.
- Message-authentication can be implemented by two methods (Figure 10.5):
  1) In first method, a hash-function is applied on a message and then a process of encryption is implemented. At the receiver site, the received message-digest is decrypted and the comparison is made between the decrypted h(m) and the message-digest made locally from the received message. compare it with the one made locally at its site for any judgments on the integrity of the message.
  2) In second method, no encryption is involved. The two parties share a secret key. Hence, at the receiving site, the comparison is made between the received h(m) and the message-digest made locally from the received message.

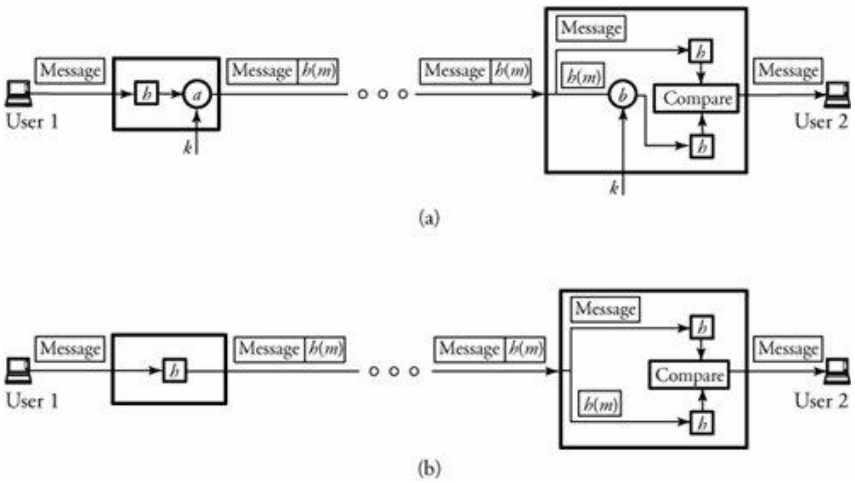

Figure 10.5. Message authentication: (a) combined with encryption; (b) use of the hash function

AUTHENTICATION AND DIGITAL SIGNATURE

- A digital signature on a message is required for the authentication and identification of the right sender.
- RSA algorithm can be used to implement digital signature.
- The message is encrypted with the sender's private key. Thus, the entire encrypted message serves as a digital signature.
- At the receiving end, the receiver can decrypt the message using the public key. This authenticates that the packet comes from the right user.

IPSEC (IP SECURITY)

- This is a set of protocols to support the secure communication at the IP layer(Figure 10.6).
- An IPSec authentication header has the following fields:

Security parameter index
This expresses a one-way relationship between two communicating users.
Sequence number
This is an increasing counter number.
Payload data
This is an encryption protected upper-layer segment.
Padding
This is used to make plaintext a certain multiple of 1byte.
Pad length
This specifies the number of bytes for the padding.
Next header
This indicates the type of next-header attached.
Authentication data
This provides the integrity check value.

- IPsec has two encryption modes:

1) Tunnel mode encrypts the header and the payload of each packet
2) Transport mode encrypts the payload.

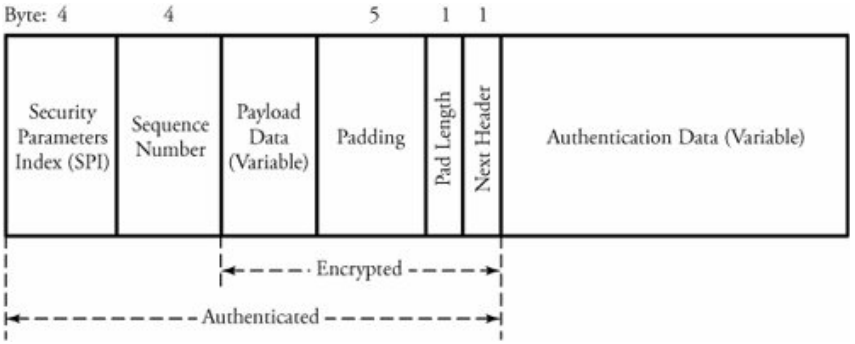

Figure 10.6. IPsec authentication header format

SECURITY OF WIRELESS NETWORKS (IEEE802.11A AND B)

• Wireless networks are particularly vulnerable because of their non-wired infrastructure.
• Security mechanisms for the wireless 802.11 standards are known as wired equivalent privacy(WEP).
• WEP is a standard for security for IEEE802.11a and b.
• WEP offers authentication and data encryption between a host and a wireless base-station, using a secret shared key.
• Communication between a host and a base station happens as follows:
1) The host requests authentication from the base-station.
2) The base-station responds.
3) The host encrypts data by using secret-key encryption.
4) The base-station decrypts the received data.
If the decrypted data matches the original one, the host is authenticated by the base-station.

Message Format

• A 40-bit secret key k known by both the host and the base-station is created (Figure 10.7).
• A 24-bit initialization field to be used to encrypt a single frame
• A 4-byte CRC field is computed for the data payload.
• The encryption is done by

$c_i = m_i \oplus k_i$ where c_i = ith byte of the ciphertext, m_i = ith byte of the plaintext, k_i = ith key

• The decryption is done by

$m_i = c_i \oplus k_i$.

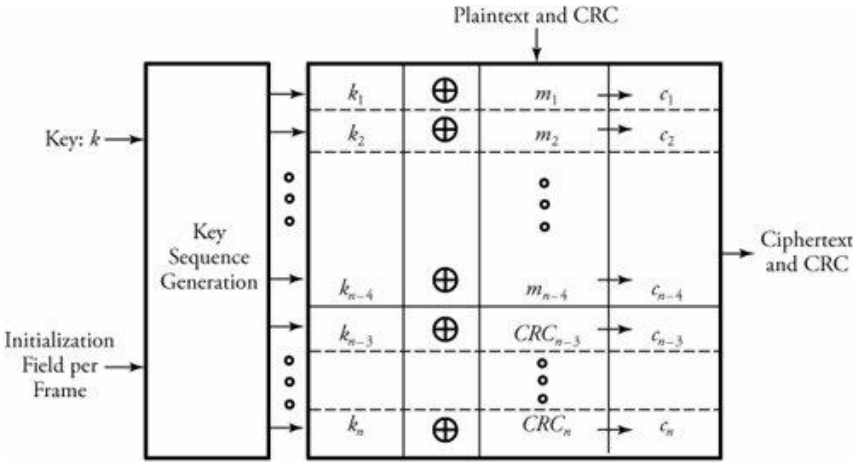

Figure 10.7. Security implementation in wireless IEEE 802.11

SECURITY OF IEEE 802.11i

• This standard specifies an authentication-server for the base-station communication.
• The separation of the authentication-server from the base-station means that the authentication-server can serve many base stations.
• Extensible Authentication Protocol(EAP) specifies the interaction between a user and an authentication server(IEEE802.11i)

To summarize the IEEE802.11i security mechanism:

• A base-station first announces its presence and types of security services it can provide to wireless users.
• EAP frames are encapsulated and sent over the wireless link.
• After decapsulation at the base station, the frames are encapsulated again, this time using a protocol called RADIUS for transmission over UDP to the authentication-server.
• With EAP, public-key encryption is used.

FIREWALLS

- This is placed between hosts of a certain network and the outside world (Figure 10.8).
- This is used to protect the network from unwanted web sites and potential hackers.
- The main objective is to monitor and filter packets coming from unknown sources.
- This can also be used to control data traffic.
- This can be a software program or a hardware device.
  1) Software firewalls can be installed in home computers by using an Internet connection with gateways.
  2) Hardware firewalls
     → are more secure than software firewalls
     → are not expensive.
- A firewall controls the flow of traffic by one of the following three methods:
  1) Packet filtering: A firewall filters those packets that pass through.
  If packets can get through the filter, they reach their destinations: otherwise, they are discarded
  2) A firewall filters packets based on the source IP address. This filtering is helpful when a host has to be protected from any unwanted external packets.
  3) Denial of Service(DOS). This method controls the number of packets entering a network.

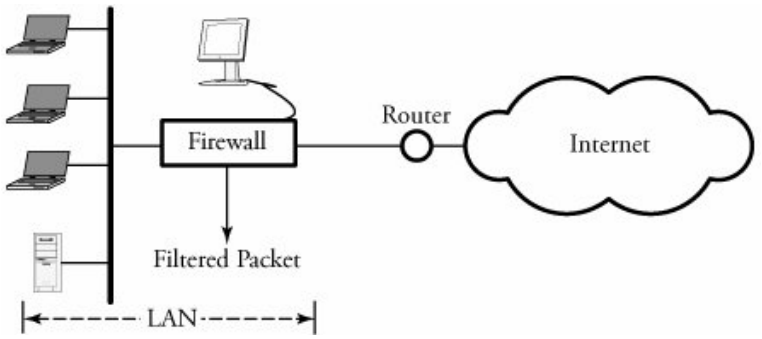

Figure 10.8. A simple configuration of a secured network using

UNIT 6: QUALITY OF SERVICE AND RESOURCE ALLOCATION

OVERVIEW OF QOS

• In a data-network port-processor, QoS unit is used to control access
→ to available bandwidth and
→ to regulate traffic.
• The provision of QoS to a network either does or does not come with a guarantee.
1) Non-guaranteed QoS(Real time application): This is based on the best-effort model, whereby a network provides no guarantees on delivery of packets but makes best effort to do so.
2) Guaranteed QoS(Non-real time application): A network can use the retransmit strategy for successful data delivery.
• Real-time packets include video and audio data which have a low delay requirement.
Non-real time packets include normal data packets, which do not have a delay requirement.
• Two broad approaches to QoS are integrated services and differentiated service.
1) Integrated-services provide QoS to individual applications and flow records.
2) Differentiated-service provide QoS support to a broad class of applications.

INTEGRATED SERVICES

• This provides QoS to individual applications and flow records.
• This consists of two service classes
1) *Guaranteed service class* is used for applications that cannot tolerate a delay beyond a particular value. This can be used for real-time applications such as video communications.
2) *Controlled-load service class* is used for applications that can tolerate some delay and loss. This is designed such that applications run very well when the network is not heavily loaded or congested.
• Four common categories of processes providing quality of service are
1) *Traffic-shaping* regulates turbulent traffic (Figure 12.1).
2) *Admission-control* governs whether the network can admit or reject the flow based on given information about an application's flow,.
3) *Resource allocation* allows network-users to reserve bandwidth on neighboring routers.
4) *Packet-scheduling* sets the timetable for the transmission of packet flows. Any involving router needs to queue and transmit packets for each flow appropriately.
• This approach has been deterred owning to scalability issues. (As the network size increases, routers need to handle larger routing tables and switch larger numbers of bits per second. In such situations, routers need to refresh information periodically).

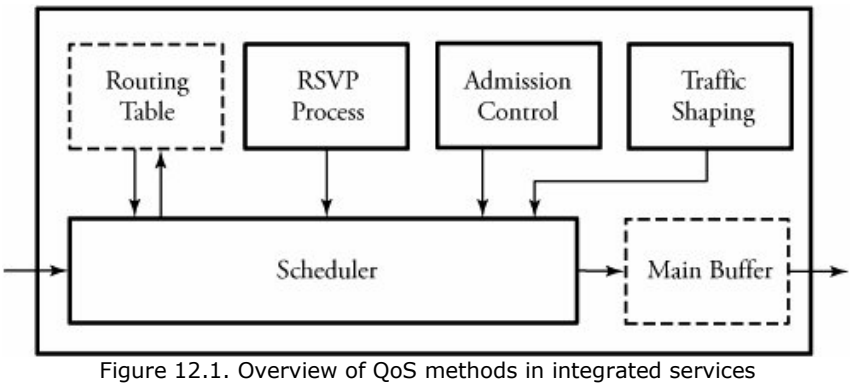

Figure 12.1. Overview of QoS methods in integrated services

TRAFFIC SHAPING

- The goal of traffic-shaping is
  → to control access to available bandwidth to regulate incoming data to avoid congestion(Fig 12.2).
  → to control the delay incurred by packets.
- Monitoring the traffic flow is called *traffic-policing.*
- Two of the most popular traffic-shaping algorithms are leaky bucket and token bucket.

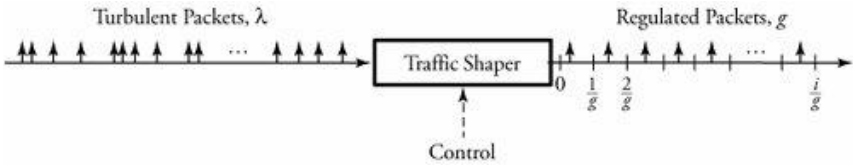

Figure 12.2. Traffic shaping to regulate any incoming turbulent traffic

LEAKY BUCKET TRAFFIC SHAPING

- This converts any turbulent incoming traffic into a smooth, regular stream of packets (Figure 12.3).
- No matter at what rate packets enter the traffic shaper, the outflow is regulated at a constant rate.
- When a packet arrives, the interface decides whether that packet should be queued or discarded, depending on the buffer-capacity.
- Incoming packets are discarded once the bucket becomes full.
- Packets are transmitted as either fixed-size packets or variable-size packets.
  In the fixed-size packet environment, a packet is transmitted at each clock tick.
  In the variable-size packet environment, a fixed sized block of a packet is transmitted.
- The leaky bucket scheme is modeled by 2 main buffers (Figure 12.4).
  One buffer forms a queue of incoming packets, and another buffer receives authorizations.

  Begin Leaky Bucket Algorithm

  1) Define for the algorithm:
     λ = rate at which packets with irregular rate arrives at the main buffer
     g = rate at which authorization grants arrive at the grant buffer
     w = size of the grant buffer and can be dynamically adjusted.
  2) Every 1/g seconds, a grant arrives
  3) Over each period of i/g seconds, i grants can be assigned to the first i incoming packets, where t<=w, and packets exit from the queue one at a time every 1/g seconds, totaling i/g seconds.
  4) If more than w packets are in the main buffer, only the first w packets are assigned grants at each window time of 1/g, and the rest remain in the main queue to be examined in the next 1/g interval.
  5) If no grant is in the grant buffer, packets start to be queued.

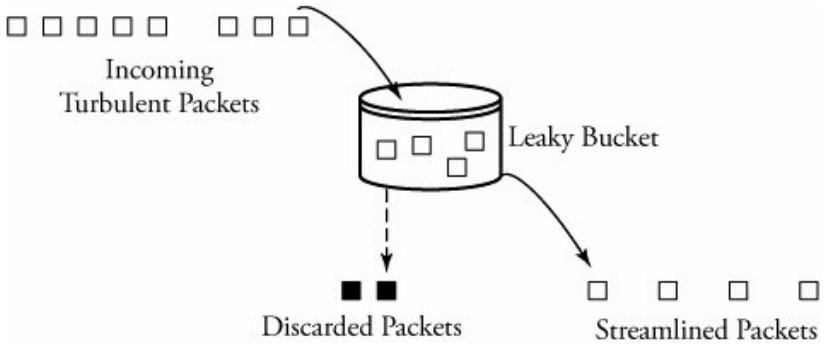

Figure 12.3. The leaky-bucket traffic-shaping algorithm

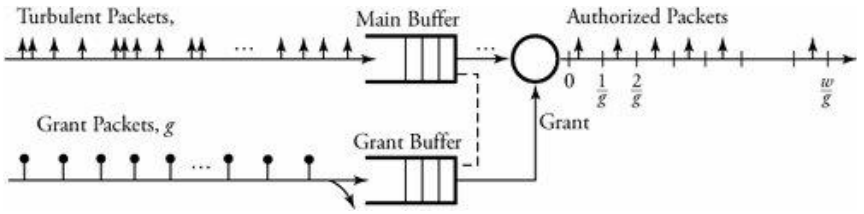

Figure 12.4. Queueing model of leaky-bucket traffic-shaping algorithm

TOKEN BUCKET TRAFFIC SHAPING

• Two parameters describe the operation of the token bucket traffic shaper (Figure 12.6).
  1) Token arrival-rate v: is normally set to the average traffic rate of the source.
  2) Bucket-depth b: is a measure of maximum amount of traffic that a sender can send in a burst.
• This consists of a buffer that accepts fixed-size tokens of data generated by a token-generator at a constant rate every clock cycle.
• According to this protocol, a sufficient number of tokens are attached to each incoming packet, depending on its size, to enter the network.
• If the bucket is full of tokens, additional tokens are discarded.
  If the bucket is empty, incoming packets are delayed (buffered) until a sufficient number of tokens is generated.

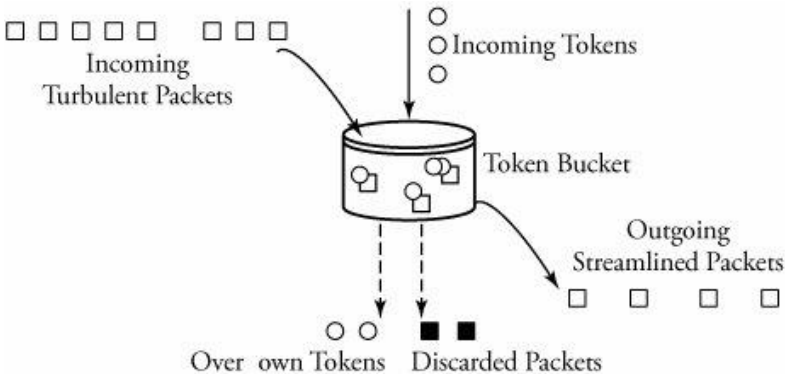

Figure 12.6. The token-bucket traffic-shaping method

COMPARISON OF LEAKY BUCKET & TOKEN BUCKET APPROACHES

• The token bucket algorithm enforces a more flexible output pattern at the average rate, no matter how irregular the incoming traffic is.
  Whereas leaky bucket method enforces a more rigid pattern
• The token bucket is known as a more flexible traffic shaping method but has greater system complexity.
  Whereas in leaky bucket approach, no virtual tokens are seen, greatly increasing to the speed of operation and enhancing the performance of the communication-node.

ADMISSION CONTROL

• The admission control process decides whether to accept traffic flow by looking at two factors
  1) r_s =type of service requested
  2) t_s =required bandwidth information about the flow
• For controlled load services, no additional parameters are required.
  However, for guaranteed services, the maximum amount of delay must also be specified.
• Any admission control scheme must be aided by a policing-scheme.
• Once a flow is admitted, the policing-scheme must make sure that the flow confirms to the specified t_s.
  If not, packets from there flows become obvious candidates for packet drops during a congestion event.

RESOURCE RESERVATION PROTOCOL (RSVP)

• This is typically used to provide real time services over a connectionless network.
• This is a soft-state protocol so that
  → This can handle link-failure efficiently
  → This can adapt to router-failures by using an alternative path
• This adopts a receiver-oriented approach to resource reservations. This process is required to provide the desired QoS for an application.
• This supports both unicast and multicast flows.
• The router can accept or deny the reservation based on its available resources.
• Reservation messages from multiple receivers can be merged if their delay requirements are similar.

PACKET SCHEDULING

- This involves managing packets in queues to provide the QoS associated with the packets (Fig 12.7).
- A packet-classifier is the heart of scheduling scheme.
- Packet-classifying involves
  - → identifying each packet with its reservation and
  - → ensuring that the packet is handled correctly
- Packets are classified according to the following parameters
  1) Source/destination IP address
  2) Source/destination port
  3) Packet flow priority
  4) Protocol type
  5) Delay sensitivity

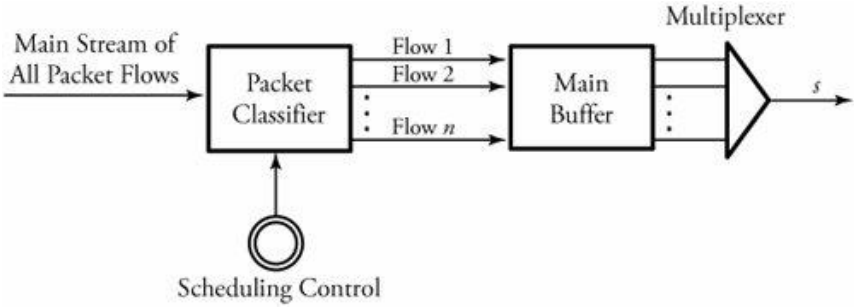

Figure 12.7. Overview of a packet scheduler

FIFO SCHEDULER

- This is the most commonly implemented scheduling policy because of its simplicity(Figure 12.8).
- Incoming packets are served in the order in which they arrive.
- A pure FIFO scheme provides no fair treatment to packets.
- A higher speed user can
  - → take up more space in the buffer and
  - → consume more than its fair share of bandwidth.
- With smart buffer management schemes, it is possible to control bandwidth sharing among different classes and traffic.
- Delay bound T_q is calculated based on the queueing buffer size as follows:

  $T_q <= (K/s)$ where K=maximum buffer size
  s=outgoing link speed.

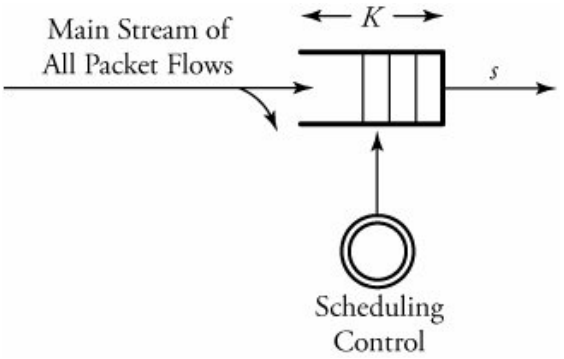

Figure 12.8. A typical FIFO queueing scheduler

PRIORITY QUEUEING(PQ) SCHEDULER

- This combines the simplicity of FIFO schedulers with the ability to provide service classes(Figure 12.9).
- Packets are classified based on the priority-of-service indicated in the packet-headers.
- Lower-priority queues are serviced only after all packets from the higher-priority queues are serviced.
- Higher-priority queues have a delay bound similar to that of FIFO.
  Lower-priority queues have a delay bound that includes the delays incurred by higher-priority queues.
- Disadvantage: Queues with lower priorities are subject to bandwidth starvation if the traffic rates for higher-priority queues are not controlled.
- Priority queueing is either non-preemptive or preemptive.
  1) Non preemptive priority queues: The process of lower-priority packets cannot be interrupted under any condition.
  2) Preemptive priority queues: The process of lower-priority packet can be interrupted by incoming higher-priority packets.

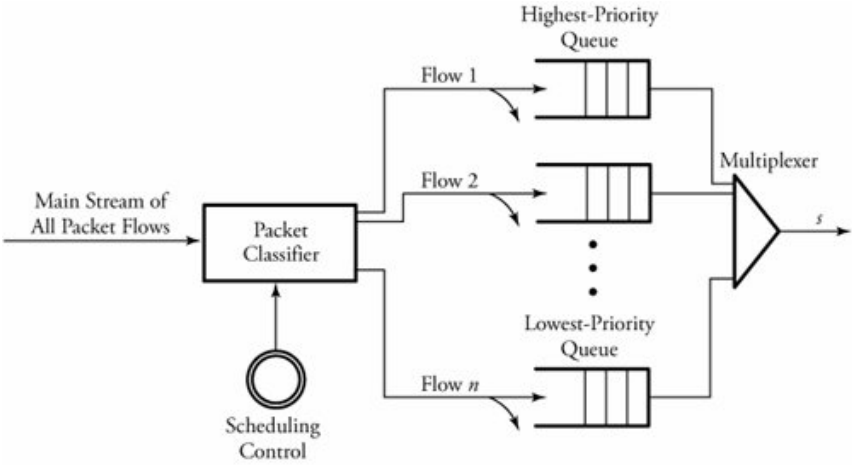

Figure 12.9: A typical priority-queueing scheduler

FAIR QUEUEING(FQ) SCHEDULER

- This is designed to better and more fairly treat servicing packets (Figure 12.11).
- This eliminates the process of packet priority sorting.
- This improves the performance and speed of the scheduler significantly.
- Each flow is assigned a separate queue.
- Each flow is guaranteed a minimum fair share of s/n bits per second on the output.
  where s=transmission bandwidth
  n=number of flows(or number of queues)
- Since all the inputs(flows) may not necessarily have packets at the same time, the individual queue's bandwidth share will be higher than s/n.
- The virtual clock count cj needed to transmit the packet is given by

$c_j = f_j - s_j$

$c_j = f_j - \max(f_{j-1}, a_j)$ where s_j = start time f_j = ending time
a_j = arriving time of packet j of a flow

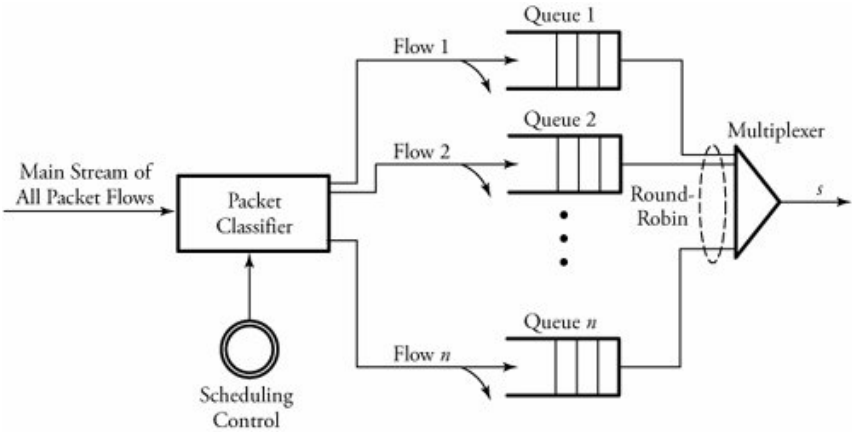

Figure 12.11. Overview of fair-queueing scheduler

WFQ SCHEDULER (WEIGHED FAIR QUEUEING)

- For an n-queue system, queue i € {1 . . . n} is assigned a weight w_i (Figure 12.12).
- Each weight w_i specifies the fraction of the total output port bandwidth dedicated to flow i.
- The outgoing link capacity s is shared among the flows with respect to their allocated weights.
- Each flow i is guaranteed to have a service rate of at least

$$r_i = s(w_i / \Sigma^{n}_{j-1}.w_j)$$

- If a queue is empty at a given time,
  unused portion of its bandwidth is shared among other active queues according to their respective weights.
- This provides a significantly effective solution for servicing real-time packets and non-real time packets, owing to its fairness and ability to satisfy real-time constraints

WRR Scheduler (Weighted Round-Robin)

- This is a version of WFQ scheduler.
- This was proposed for asynchronous transfer mode.
- Each flow is served in a round-robin fashion with respect to a weight assigned for each flow i without considering packet length.

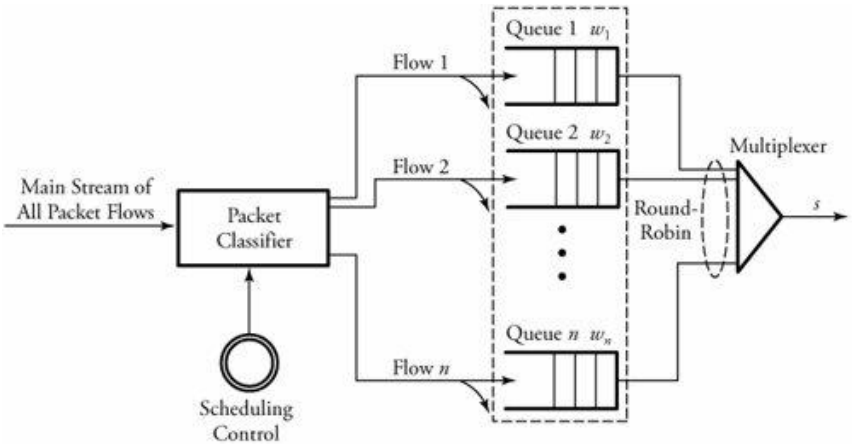

Figure 12.12. Overview of weighted fair-queueing scheduler

DIFFERENTIATED SERVICES QOS

• This provides QoS support to a broad class of applications.
• This provides a simpler and more scalable QoS .
• This minimizes the amount of storage needed in a router by processing traffic flows in an aggregate manner.
• The *traffic-classifier* routes packets to specific outputs, based on the values found inside multiple fields of a packet header (Figure 12.14).
• The *traffic-conditioner* detects and responds if any packet has violated any of the rules specified in the TCA(Traffic-Conditioning Agreement).
• The traffic-conditioner has 4 major components:

Meter
This measures the traffic to make sure that packets do not exceed their traffic profiles
Marker
This marks or unmarks packets in order to keep track of their situations in the DS node
Shaper
This delays any packet that is not complaint with the traffic profile
Dropper
This discards any packet that violates its traffic profile

• A **bandwidth-broker** is needed to allocate and control the available bandwidth within the DS domain,
• In order to process traffic flows in an aggregate manner, a packet must go through SLA(Service Level Agreement) that includes a TCA
• An *SLA* indicates the type of forwarding service, and
a *TCA* presents all the detailed parameters that a customer receives.
• An SLA can be either static or dynamic
1) Static SLA is a long-term agreement.
2) Dynamic SLA uses the bandwidth-broker that allows users to make changes more frequently.

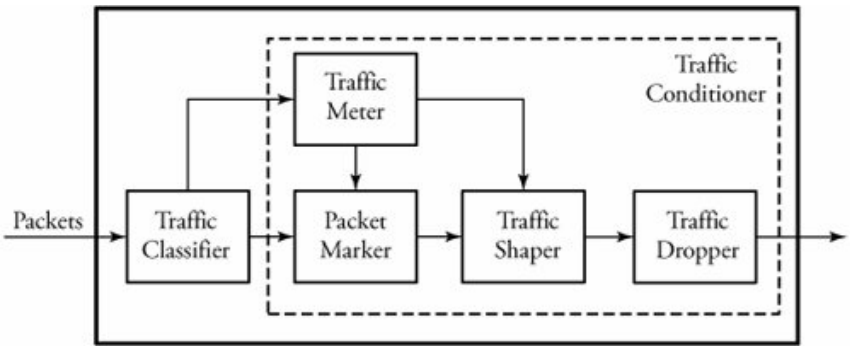

Figure 12.14. Overview of DiffServ operation

PHB (PER HOP BEHAVIOR)

• Two types of PHB are: expedited forwarding and assured forwarding.

Expedited forwarding PHB
• This provides low-loss, low latency, low jitter, ensured bandwidth and end-to-end services.
• The aggregate arrival-rate should be less than the aggregate minimum departure-rate.

Ensured forwarding PHB
• This delivers packets with high assurance and high throughput, as long as the aggregated traffic does not exceed TCA.
• This does not provide low latency and low jitter application.
• This group can be classified into three service types: good, average and poor.

UNIT 6(CONT.): VPNs, TUNNELING, AND OVERLAY NETWORKS

VPN (VIRTUAL PRIVATE NETWORKS)

• This is a networking infrastructure whereby a private-network makes use of the public-network.
• VPN part of public network is set up "virtually" by a private-sector entity to provide public networking services to small entities.
• This maintains privacy by using tunneling protocols and security procedures.
• Benefits to an organization by using VPN
    1) Extended geographical communication
    2) Reduced operational cost
    3) Enhanced organizational management
    4) Enhanced network management with simplified local area networks
    5) Improved productivity and globalization.
• There are two types of VPNs: remote access and site-to-site (Figure 16.2).

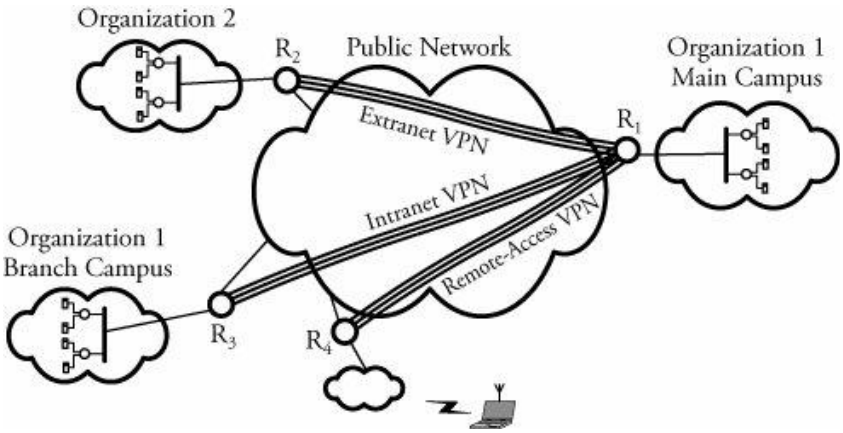

Figure 16.2. Three types of VPNs to and from a headquarter organization

REMOTE ACCESS VPN

• This is a user-to-LAN connection.
• An organization uses VPN to connect its users to a private network from various remote locations.
• This allows encrypted connections between an organization's private network and remote-users through a third-party service provider.
• Tunneling uses mainly the Point-to-point protocol(PPP).
• PPP is the carrier for other Internet protocols when communicating over the network between a host computer and a remote point.
• Besides IPsec, other types of protocols associated with PPP are
    1) L2F(layer 2 forwarding) protocol uses the authentication scheme supported by PPP.
    2) Point to Point Tunneling Protocol(PPTP) supports 40-bit and 128-bit encryption and uses the authentication scheme supported by PPP.
    3) L2TP(Layer 2 tunneling protocol) combines features of both PPTP & L2F.

SITE TO SITE VPN

• An organization uses VPN to connect multiple fixed sites over a public-network.
• VPNs can be classified as either intranets or extranets.
    1) Intranet VPNs: connect an organization's remote site LANs into a single private network.
    2) Extranet VPNs: allow two organizations to work in a shared environment through a tunnel built to connect their LANs.
• The main benefit of using a VPN is scalability with a reasonable cost.
• GRE (Generic Routing Encapsulation) is normally the encapsulating protocol. It provides the framework for the encapsulation over an IP-based protocol.
• L2TP can also be used. This fully supports IPSec regulations and can be used as a tunneling protocol for remote access VPNs.

TUNNELING AND PPP (POINT-TO-POINT PROTOCOL)

• A tunnel is a connection that forms a virtual-network on top of a physical-network(Figure 16.3).
• Tunneling is a process of encapsulating packets and sending them over the public-network.
• A tunnel is a relatively inexpensive connection, since it uses the Internet as its primary form of communication.
• Tunneling requires two other types of protocols:
    1) Carrier protocols are used for carrying information over the public network..
    2) Encapsulating protocols are used for wrapping, encapsulating and securing data.
• VPNs can put a packet that uses a non-routable IP address inside a packet to extend a private-network over the Internet.

PPP

• This is a Tunneling protocol.
• This allows an organization to establish secure connection from one point to another while using public resources.
• A PPP connection is a serial connection between a user and an ISP.

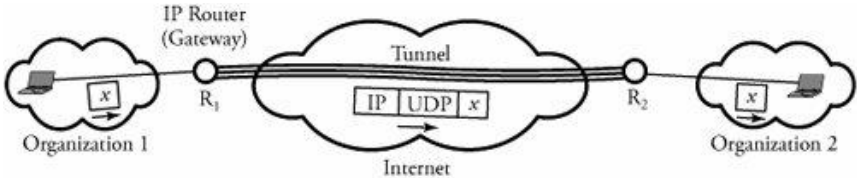

Figure 16.3. A customized protocol packet tunneling through the Internet

Security in VPNs

• Firewalls provides an effective barrier between a private network and the Internet.
• This can be set up to restrict the number of open ports to monitor what types of packets are passed through and which protocols are allowed through.

MPLS (MULTI PROTOCOL LABEL SWITCHING)

• MPLS transmission is a special case of tunneling.
• Features of MPLS:
→ connection-oriented forwarding mechanism
→ has layer 2 label-based lookups
→ enables traffic engineering to implement peer-to-peer VPNs effectively
→ supports other applications, such as IP multicast routing and QoS extension.
• This uses a small label appended to packets and typically makes efficient routing decisions.
• Main benefit: flexibility in merging IP-based networks with fast-switching capabilities.
• This technology adds new capabilities to IP-based networks
→ VPN support
→ Connection oriented QoS support
→ Traffic engineering: for efficient link bandwidth assignments
→ Multiprotocol support: Multiple labels can be combined in a packet to form a header for efficient tunneling

MPLS OPERATION

• MPLS network consists of nodes called label-switch-routers(LSR).
• An LSR switches labeled packets according to particular switching tables (Figure 16.5).
• An LSR has two distinct functional components: a control component and a forwarding component.
1) The control component uses routing protocols such as OSPF and BGP.
2) The control component also facilitates the exchange of information with other LSRs to build and maintain the forwarding table.
3) A label is a header used by an LSR to forward packets.
4) When a packet arrives, the forwarding component uses the label of the packet as an index to search the forwarding table for a match.
5) The forwarding component then directs the packets from the input interface to the output interface through the switching fabric.
• Key to scalability of MPLS: labels have only local significance between two devices that communicate.

MPSL Packet Format

• Label value: This is significant only locally (Figure 16.6).
• Exp: This is reserved for future experimental use.
• S is set to 1 for the oldest entry in the stack and to 0 for all other entries.
• TTL: This is used to encode a hop-count value to prevent packets from looping forever in the network.

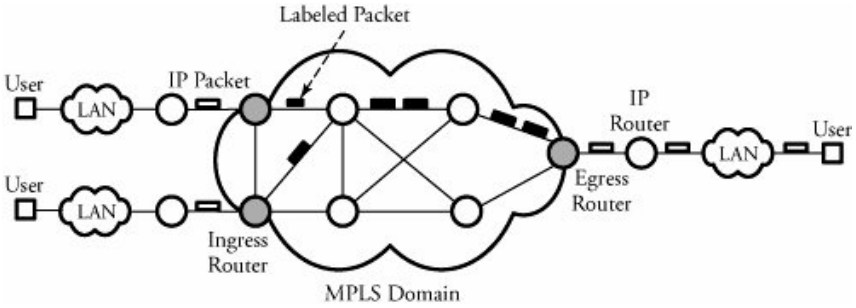

Figure 16.5. An MPLS network

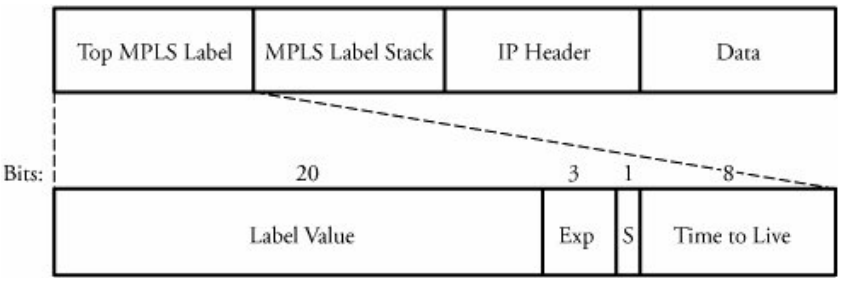

Figure 16.6. MPLS header encapsulation for an IP packet

ROUTING IN MPLS DOMAINS

• An MPLS domain has three label manipulation instructions:

1) ***Ingress LSR*** is an edge device which performs initial packet processing and classification and applies the first label. This creates a new label (Figure 16.7).

2) ***Core LSR*** swaps the incoming label with a corresponding next-hop label found from a forwarding table.

3) ***Egress LSR*** is an edge router which pops the label from the packets.

• Label stacking enables multilevel hierarchical routing.

• Once an IP packet enters an MPLS domain, the ingress LSR processes its header information and maps that packet to a FEC(Forward Equivalence Class).

• At this point, a label-switch-path(LSP) through the network must be defined, and the QoS parameters along that path must be established.

• The QoS parameters defines how many resources are to be used for that path and what queueing and discarding policy are to be used.

• An intra-domain routing protocol such as OSPF is used to exchange routing information and the LDP(label distribution protocol) is used to assign labels.

• At the end of the process, the router appends an appropriate label for FEC purposes and forwards the packet through.

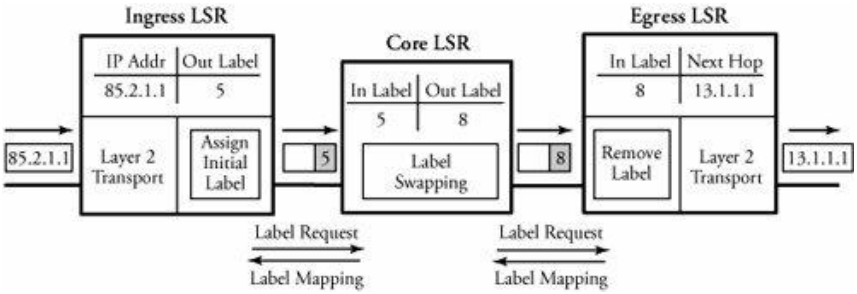

Figure 16.7. Multiple layer 2 switching example in MPLS

TUNNELING AND USE OF FEC

• Any traffic is grouped into FECs (Figure 16.8).

• FEC implies that a group of IP packets are forwarded in the same manner (for example, over the same path or with the same forwarding treatment).

• A packet can be mapped to a particular FEC, based on the following criteria

→ source and/or destination IP address
→ TCP/UDP port numbers
→ class of service
→ Applications

• Route selection can be done either hop by hop or by explicit routing.

1) With ***hop by hop routing***, each LSR can independently choose the next hop for each FEC. This does not support traffic engineering because of limited available resources.

2) With ***explicit routing***, a single LSR determines the LSP for a given FEC. This can provide all the benefits of traffic engineering.

LDP (Label Distribution Protocol)

• This is a set of rules by which LSRs exchange information effectively.

• This enables two LSRs to understand each other's MPLS capabilities.

• LSP schemes are either downstream on demand or downstream unsolicited.

1) ***Downstream on demand scheme:*** An upstream-node explicitly requests a label from a downstream-node and the downstream-node forms the requested label.

2) ***Downstream unsolicited scheme:*** A downstream-node advertiser a label mapping even without receiving any advance requests.

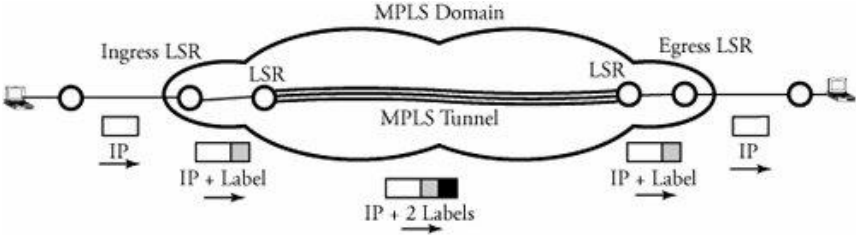

Figure 16.8. An IP packet labeled in an MPLS domain and tunneled to reach the other end of the domain

TRAFFIC ENGINEERING

• This enables an ISP to route high quality traffic to offer the best service to users in terms of throughput and delay.
• This substitutes the need to manually configure network devices to set up explicit routes (Figure 16.9).
• This is an automated scheme for control signaling and link bandwidth assignment and has a dynamic adaptation mechanism.
• This can be either traffic oriented or resource oriented.

1) ***Traffic-oriented:*** This relates to the optimization of traffic performance parameters such as
-the minimization of packet loss and delay and
-quick fault recovery when a node or a link fails.
2) ***Resource-oriented:*** This relates to the optimization of network resource utilization.

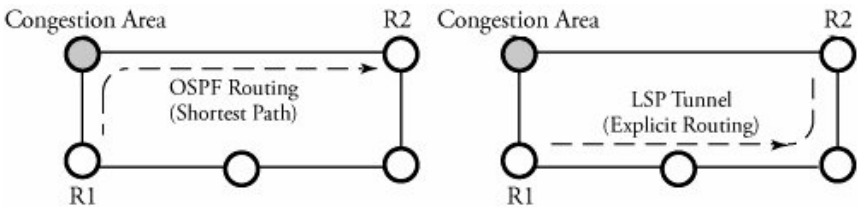

Figure 16.9. A traffic engineering scenario

OVERLAY NETWORK

• This is an application specific computer network built on top of another network(Figure 16.11).
• This creates a virtual topology on top of the physical topology of the public network.
• This type of network is created to protect the existing network structure from new protocols whose testing phases requires Internet use.
• These have no control over how packets are routed in the underlying network between a pair of overlay source/destination nodes.
• However, these can control a sequence of overlay nodes through a message passing function before reaching destination.
• These are self-organized. When a node fails, the overlay-network algorithm should provide solutions that let the network recover and recreate an appropriate network structure.
• These permit routing messages to destinations when the IP address is not known in advance.

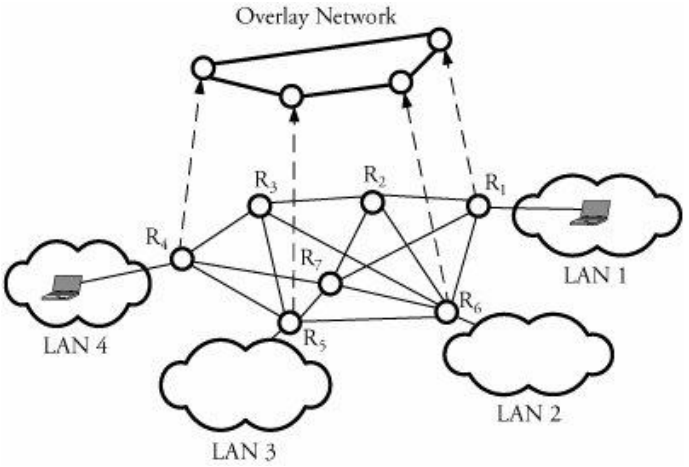

Figure 16.11. An overlay network for connections between two LANs associated with routers R1 and R4