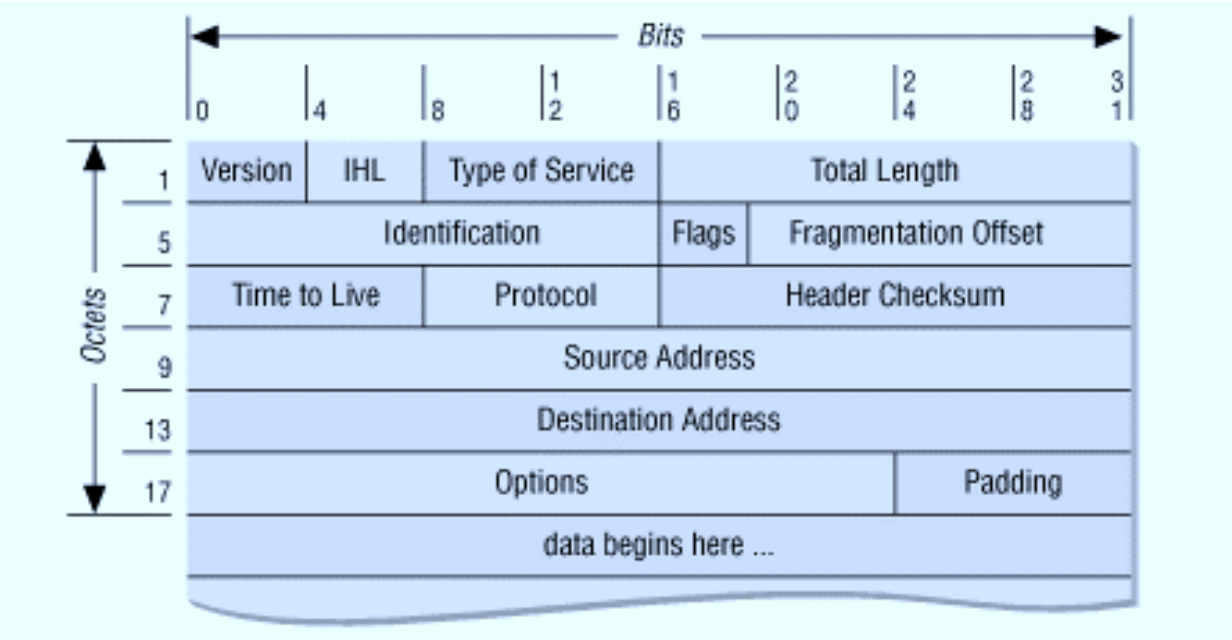
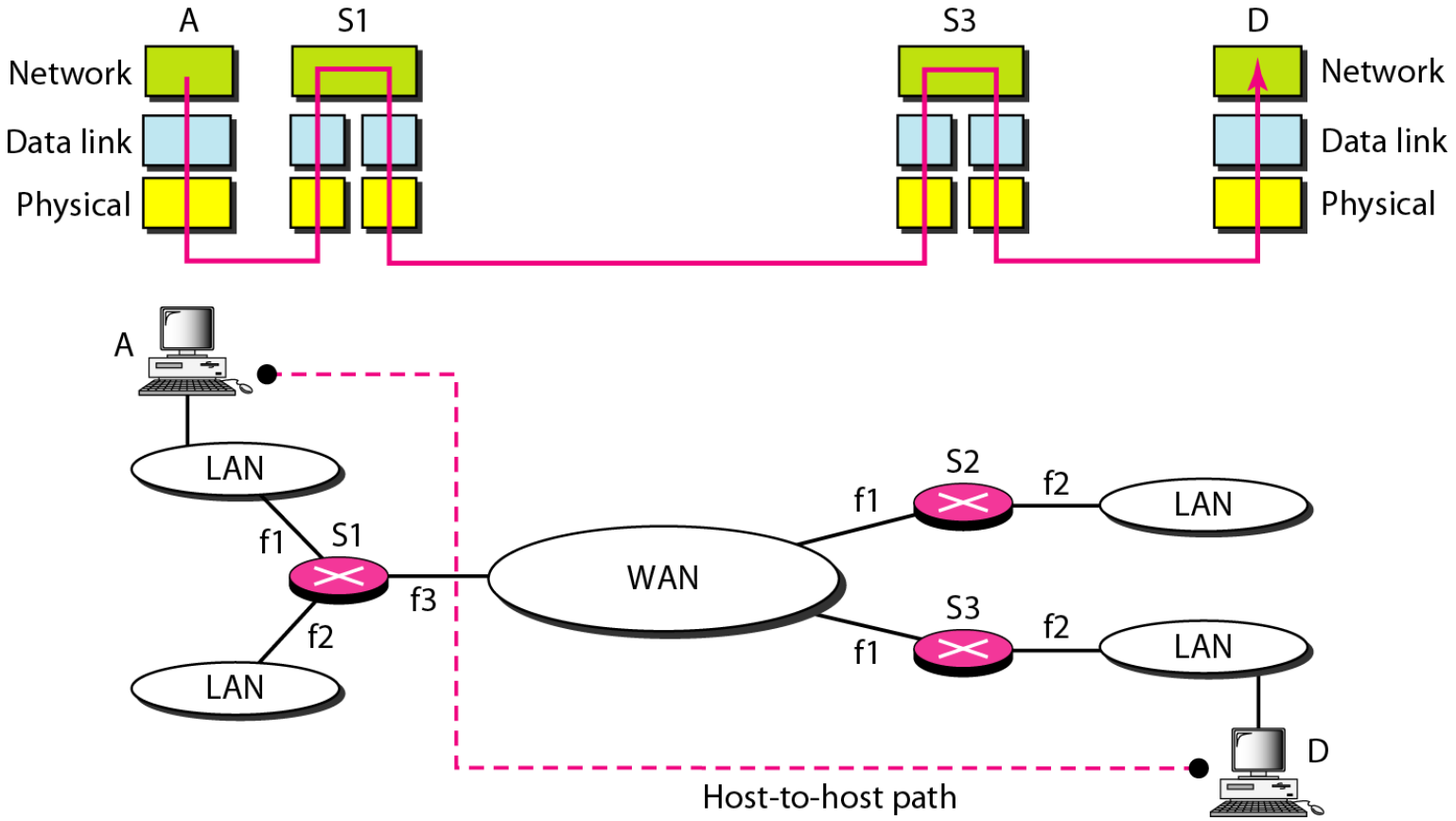
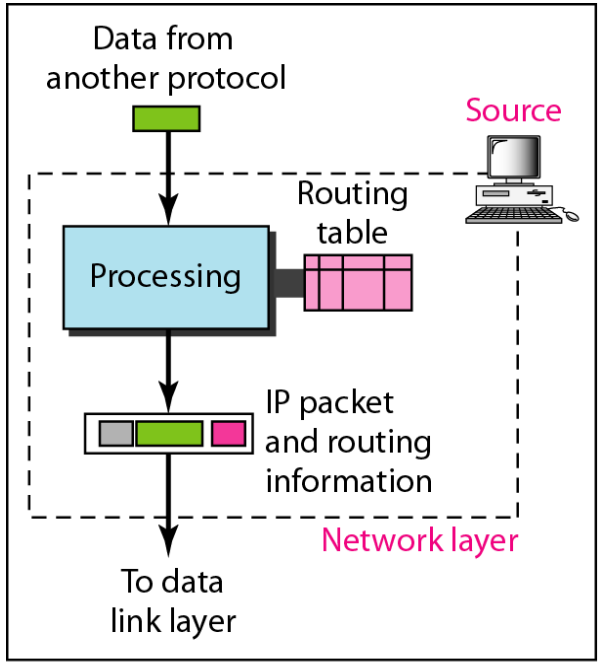


Network Layer
Internet Protocol
&
Routing

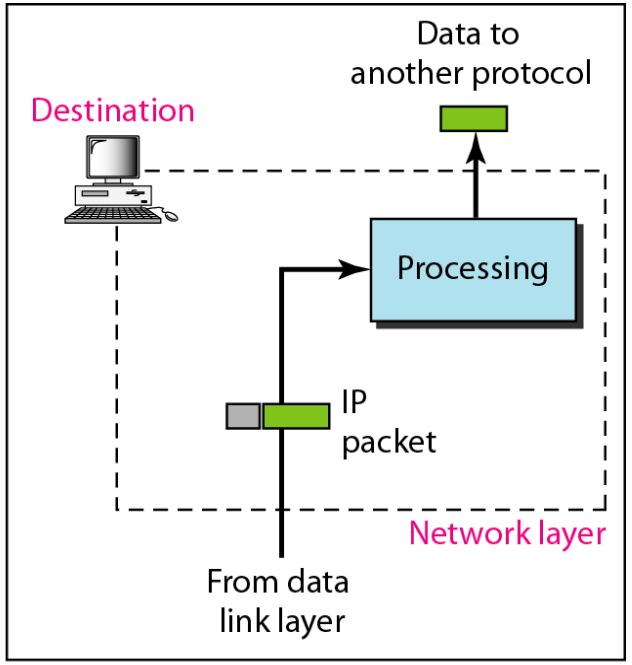
Need of Network Layer



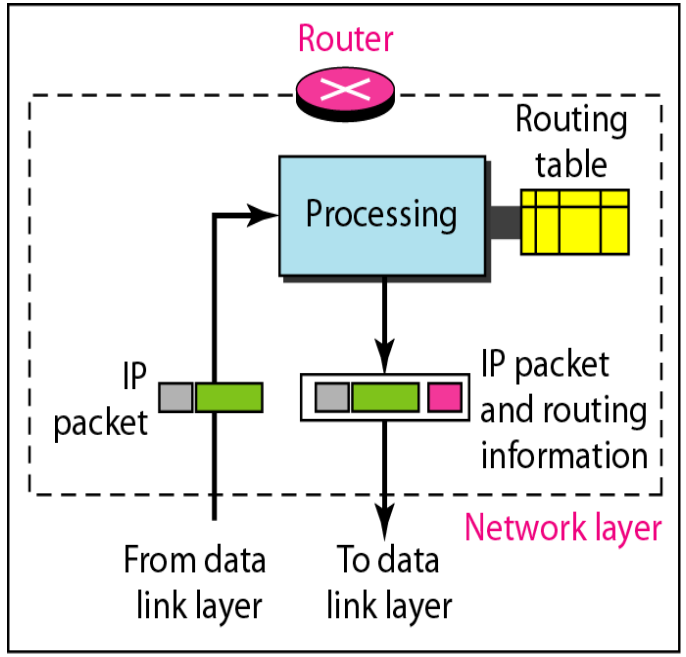
Network Layer at Source, router and destination



a. Network layer at source



b. Network layer at destination



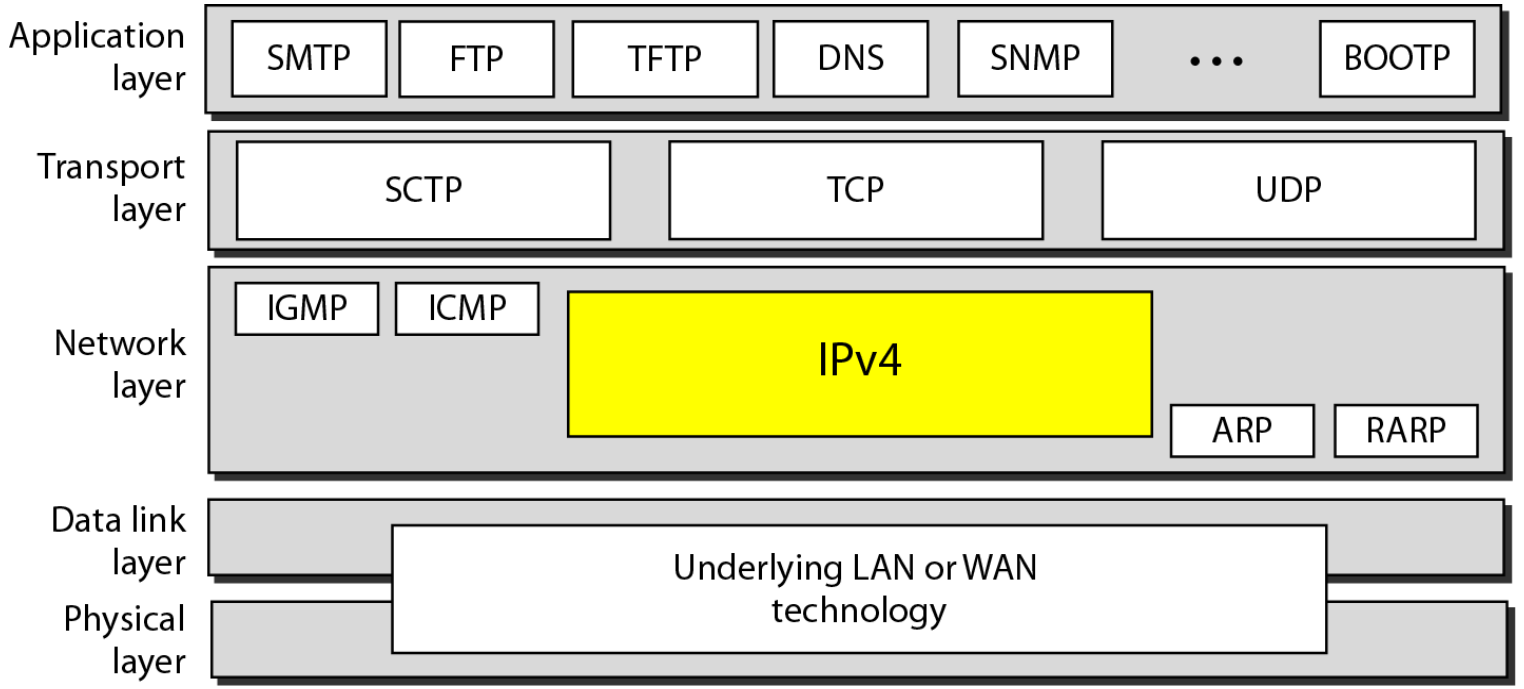
c. Network layer at a router

Internet as a Datagram Network

- The Internet, at the network layer, is a packet-switched network.
- Switching can be divided into three broad categories: circuit switching, packet switching, and message switching.
- Packet switching uses either the virtual circuit approach or the datagram approach.
- The Internet has chosen the datagram approach to switching in the network layer.
- It uses the universal addresses defined in the network layer to route packets from the source to the destination.
- Communication at network layer is connection less.

Internet Protocol(IPv4)

- The Internet Protocol version 4 (IPv4) is the delivery mechanism used by the TCP/IP protocols.

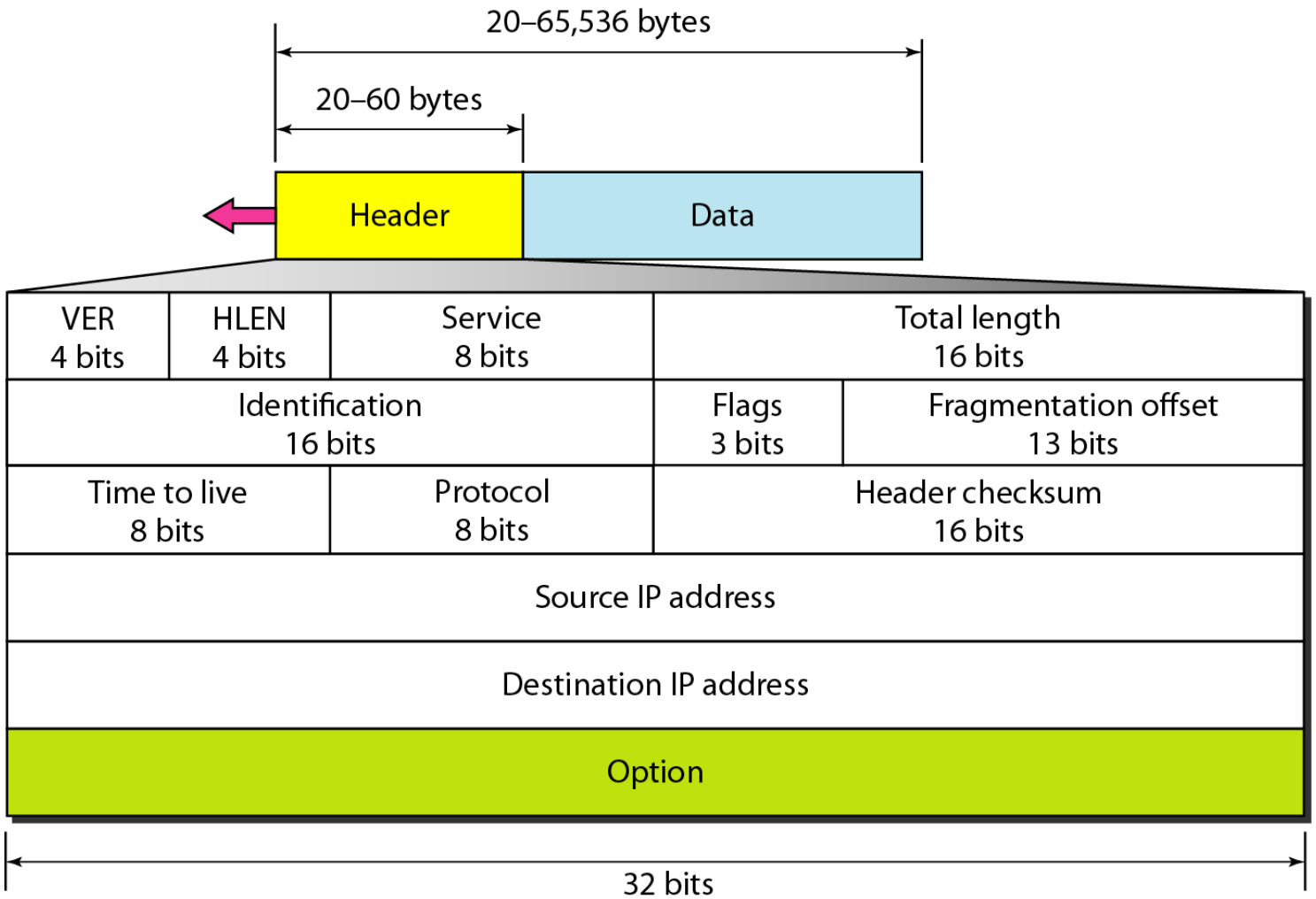


Position of IPv4 in TCP/IP protocol suite

The Internet Protocol(IPv4)

- Best effort delivery
- No error control
- No flow control
- Each datagram is handled independently
- Each datagram can follow different routes
- At destination they could arrive out of order
- If reliability is important, IPv4 must be paired with a reliable protocol such as TCP.
- Packet in the IPv4 layer are called datagrams

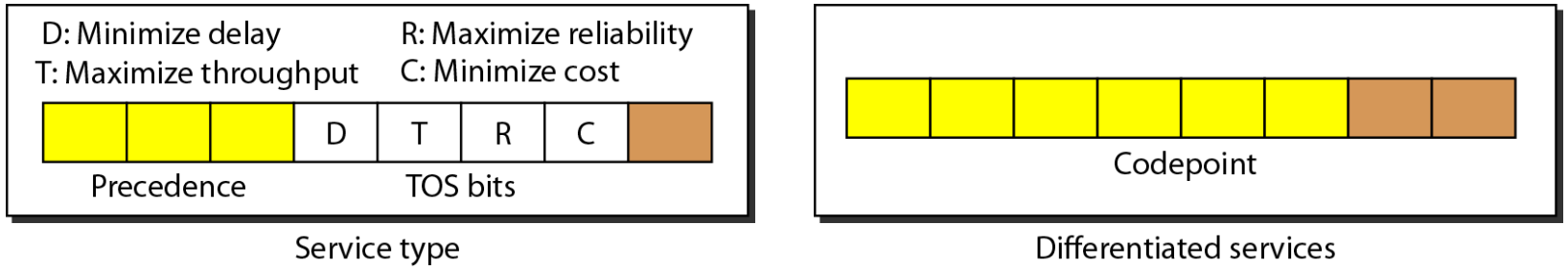
IPv4 datagram format



IPv4 datagram format

- Version (VER).
  - This 4-bit field defines the version of the IPv4 protocol.
  - Currently the version is 4. However, version 6 (or IPng) may totally replace version 4 in the future. This field tells the IPv4 software running in the processing machine that the datagram has the format of version 4.
- Header length (HLEN).
  - This 4-bit field defines the total length of the datagram
  - header in 4-byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes). When there are no options, the header length is 20 bytes, and the value of this field is 5 (5 x 4 = 20). When the option field is at its maximum size, the value of this field is 15 (15 x 4 = 60).
- Services.
  - IETF has changed the interpretation and name of this 8-bit field.
  - This field, previously called service type, is now called differentiated services.

Service type or differentiated services



- Precedence is a 3-bit subfield ranging from 0 (000 in binary) to 7 (111 in binary).
- The precedence defines the priority of the datagram in issues such as congestion.
- If a router is congested and needs to discard some datagrams, those datagrams with lowest precedence are discarded first

Types of Service

TOS Bits	*Description*
0000	Normal (default)
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

Value	*Protocol*
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

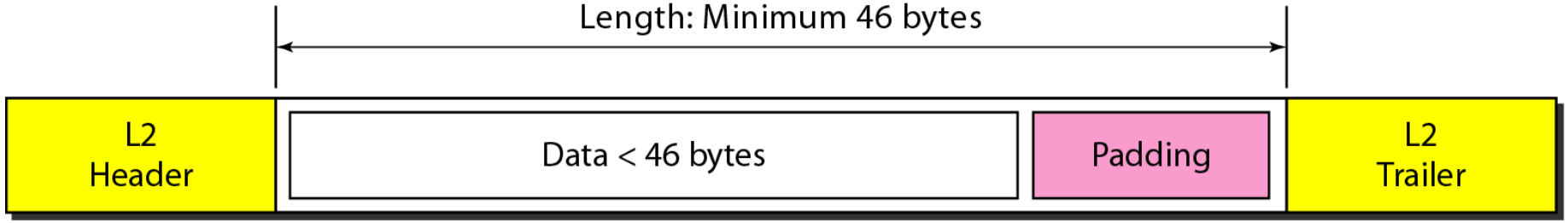
Differentiated services

Protocol	*TOS Bits*	*Description*
ICMP	0000	Normal
BOOTP	0000	Normal
NNTP	0001	Minimize cost
IGP	0010	Maximize reliability
SNMP	0010	Maximize reliability
TELNET	1000	Minimize delay
FTP (data)	0100	Maximize throughput
FTP (control)	1000	Minimize delay
TFTP	1000	Minimize delay
SMTP (command)	1000	Minimize delay
SMTP (data)	0100	Maximize throughput
DNS (UDP query)	1000	Minimize delay
DNS (TCP query)	0000	Normal
DNS (zone)	0100	Maximize throughput

Default types of service

Types of Service

The total length field defines the total length of the datagram including the header.

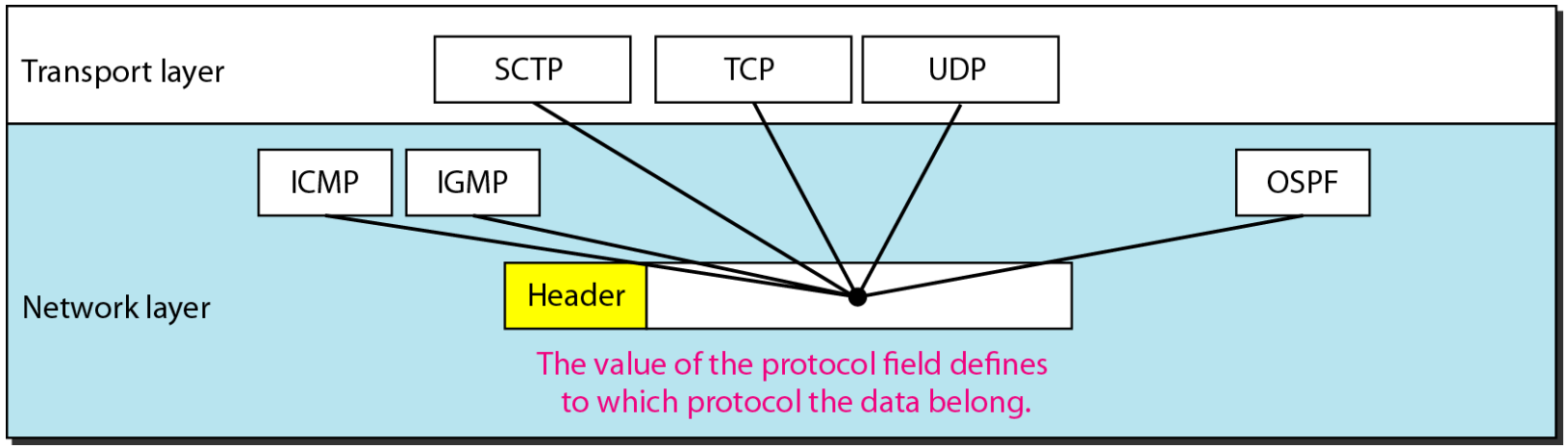


Encapsulation of a small datagram in an Ethernet frame

IPv4 datagram format

- Identification, Flags and Fragmentation offset.
  - All above field is used in fragmentation (discussed in the next section).
- Time to live.
  - A datagram has a limited lifetime in its travel through an internet.
  - This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero.
  - However, for this scheme, all the machines must have synchronized clocks and must know how long it takes for a datagram to go from one machine to another.
- Protocol.
  - This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer.
  - An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP.
  - This field specifies the final destination protocol to which the IPv4 datagram is delivered.

Protocol Field and encapsulated data



Protocol values

Value	*Protocol*
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

IPv4 datagram format

- Checksum.
  - The checksum concept and its calculation are discussed later in this chapter.
- Source address.
  - This 32-bit field defines the IPv4 address of the source.
  - This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.
- Destination address.
  - This 32-bit field defines the IPv4 address of the destination.
  - This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

Example 1

An IPv4 packet has arrived with the first 8 bits as shown:

01000010

The receiver discards the packet or not. Why?

Example 1

An IPv4 packet has arrived with the first 8 bits as shown:

01000010

The receiver discards the packet or not. Why?

Solution

There is an error in this packet.
The 4 leftmost bits (0100) show the version, which is correct.
The next 4 bits (0010) show an invalid header length ($2 \times 4 = 8$).
The minimum number of bytes in the header must be 20.
The packet has been corrupted in transmission.

Example 2

In an IPv4 packet, the value of HLEN is 1000 in binary. How many bytes of options are being carried by this packet?

Example 2

In an IPv4 packet, the value of HLEN is 1000 in binary. How many bytes of options are being carried by this packet?

Solution

The HLEN value is 8, which means the total number of bytes in the header is 8 × 4, or 32 bytes. The first 20 bytes are the base header, the next 12 bytes are the options.

Example 3

In an IPv4 packet, the value of HLEN is 5, and the value of the total length field is 0x0028. How many bytes of data are being carried by this packet?

Example 3

In an IPv4 packet, the value of HLEN is 5, and the value of the total length field is 0x0028. How many bytes of data are being carried by this packet?

Solution

The HLEN value is 5, which means the total number of bytes in the header is 5 × 4, or 20 bytes (no options). The total length is 40 bytes, which means the packet is carrying 20 bytes of data (40 − 20).

Example 4

An IPv4 packet has arrived with the first few hexadecimal digits as shown.

0x45000028000100000102 . . .

How many hops can this packet travel before being dropped? The data belong to what upper-layer protocol?

Example 4

An IPv4 packet has arrived with the first few hexadecimal digits as shown.

0x45000028000100000102 . . .

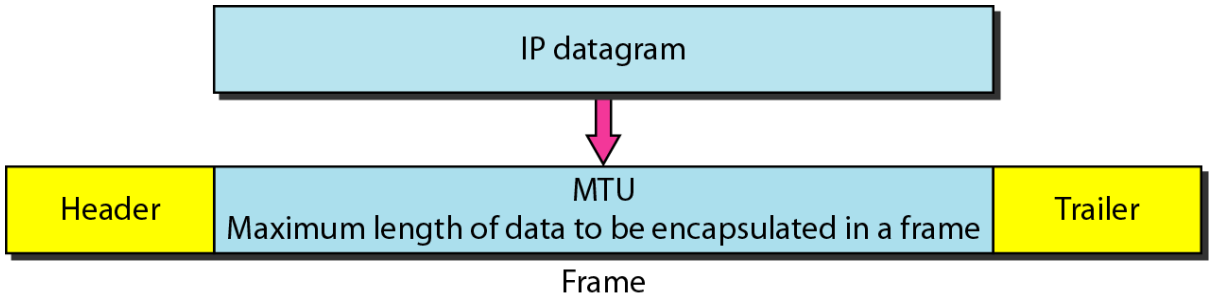
How many hops can this packet travel before being dropped? The data belong to what upper-layer protocol?

Solution

To find the time-to-live field, we skip 8 bytes. The time-to-live field is the ninth byte, which is 01. This means the packet can travel only one hop. The protocol field is the next byte (02), which means that the upper-layer protocol is IGMP.

Fragmentation

- A datagram can travel through different networks. Each router decapsulates the IPv4 datagram from the frame it receives, processes it, and then encapsulates it in another frame.
- The format and size of the received frame depend on the protocol used by the physical network through which the frame has just traveled.
- The format and size of the sent frame depend on the protocol used by the physical network through which the frame is going to travel.
- For example, if a router connects a LAN to a WAN, it receives a frame in the LAN format and sends a frame in the WAN format.



Maximum Transfer Unit (MTU)

MTUs for some networks

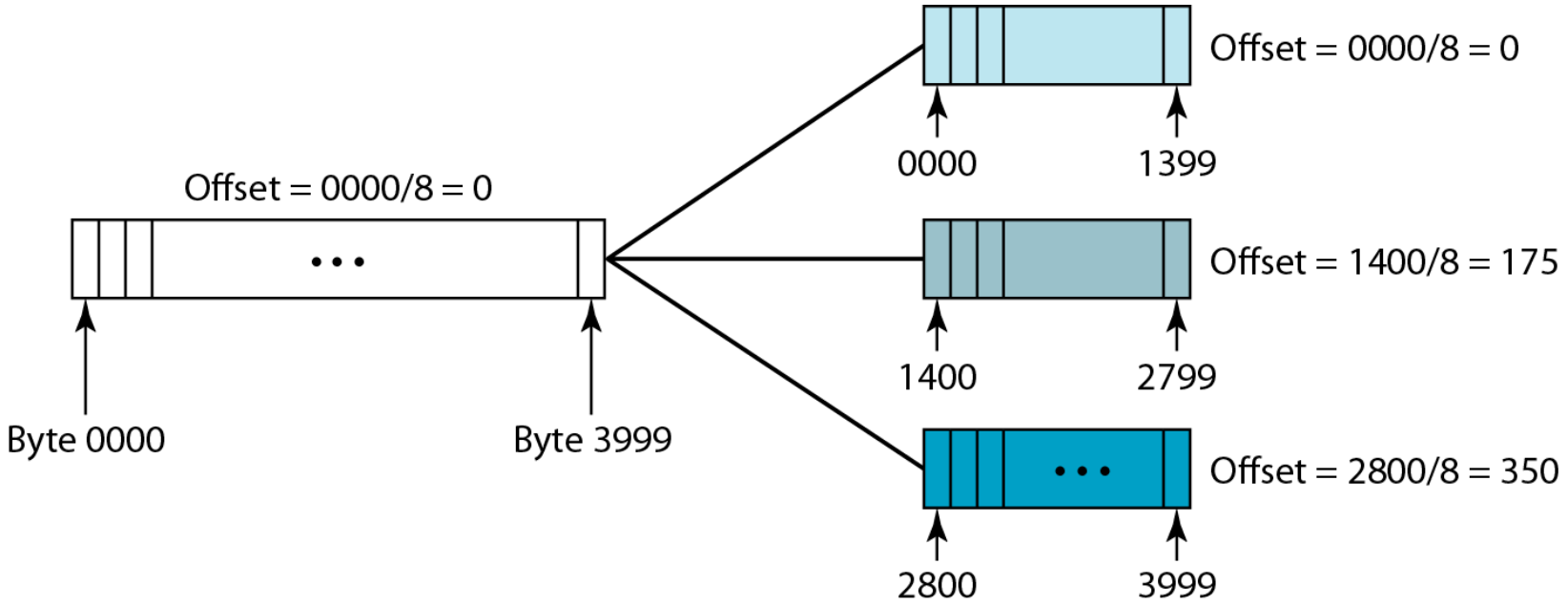
Protocol	*MTU*
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

Fragmentation Fields

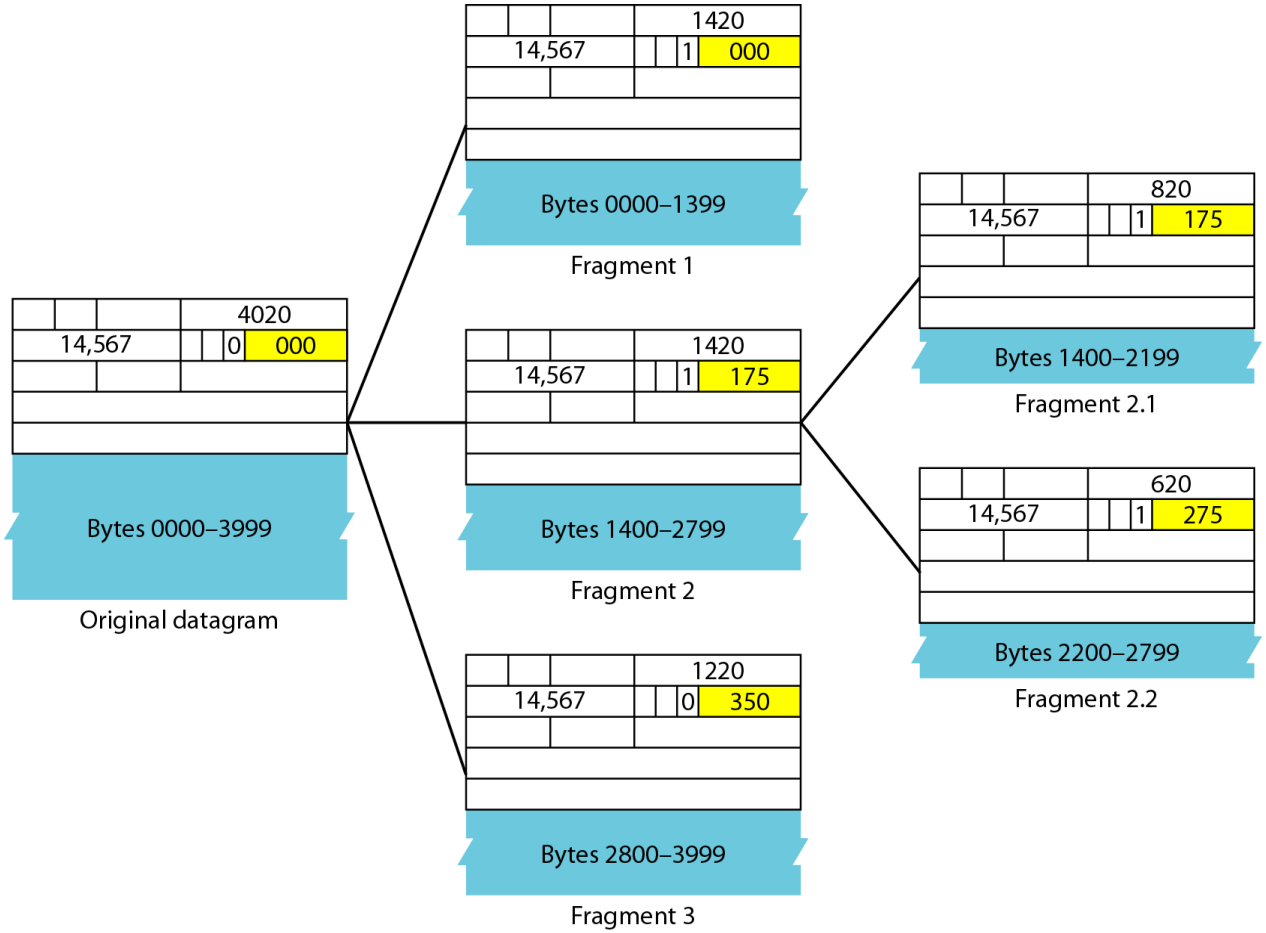
- Identification.
  - This 16-bit field identifies a datagram originating from the source host.
  - The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host
- Flags.
  - This is a 3-bit field. The first bit is reserved.
  - The second bit is called the do not fragment bit.
  - If its value is 1, the machine must not fragment the datagram.
  - If its value is 0, the datagram can be fragmented if necessary.
  - The third bit is called the more fragment bit. If its value is 1, it means the datagram is not the last fragment; there are more fragments after this one.
  - If its value is 0, it means this is the last or only fragment

D: Do not fragment
M: More fragments

Fragmentation example



Fragmentation example



Fragmentation example 1

- A datagram of 3000 bytes (header+ data) reached at router and must be forwarded to link with MTU of 500 B . How many fragments will be generated and also write MF, offset and total length value for all fragments.

Fragmentation example 1

- A datagram of 3000 bytes (header+ data) reached at router and must be forwarded to link with MTU of 500 B . How many fragments will be generated and also write MF, offset and total length value for all fragments.

Solution

- Datagram size is 3000 = 20 Bytes header + 2980 bytes data.
- MTU is 500, so its allow including header maximum 500 bytes and in each packets we can send maximum 480 data + 20 bytes of headers.
- Datagram divided into 2980/480 = 6.208 = 7 fragments.
- Offset value must be in scale of 8, 480/8 = 60

Fragment No	Total Length	MF	Offset
1	480 + 20 = 500	1	0
2	480 + 20 = 500	1	60
3	480 + 20 = 500	1	120
4	480 + 20 = 500	1	180
5	480 + 20 = 500	1	240
6	480 + 20 = 500	1	300
7	100 + 20 = 120	0	360

Example 2

- A packet has arrived in which the offset value is 100, the value of HLEN is 5, and the value of the total length field is 200. What are the numbers of the first byte and the last byte?

Example 2

- A packet has arrived in which the offset value is 100, the value of HLEN is 5, and the value of the total length field is 200. What are the numbers of the first byte and the last byte?

Solution

- Fragmentation offset : 100 , HLEN = 5 and Total length field = 200.
- Header length = HLEN x 4 = 20 Bytes.
- Bytes ahead of this fragment: 100 x 8 = 800 bytes.
- Total length is 200 ; data part will be 200 – 20 = 180 bytes.
- So last byte number will be 800 + 180 = 980

Example 3

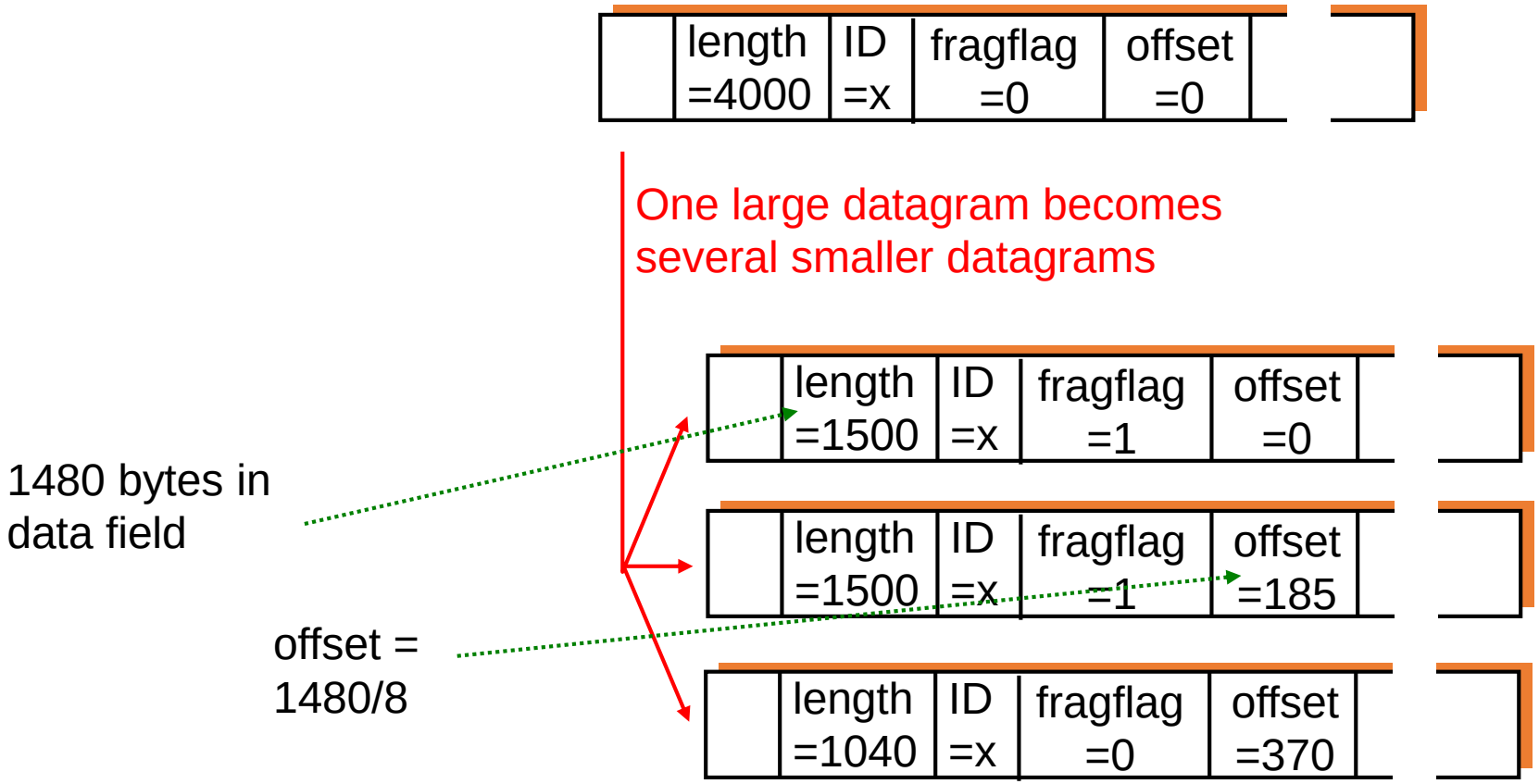
- A datagram of 4000 bytes (header+ data) reached at router and must be forwarded to link with MTU of 1500 bytes . How many fragments will be generated and also write MF, offset and total length value for all fragments.

Solution

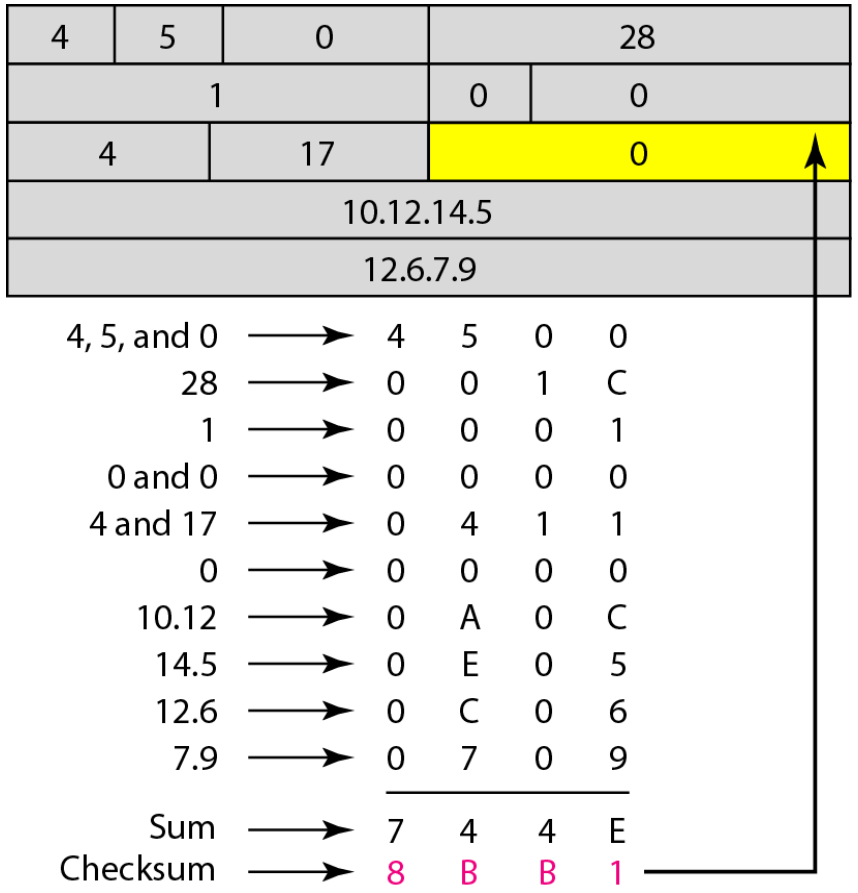
Example 3

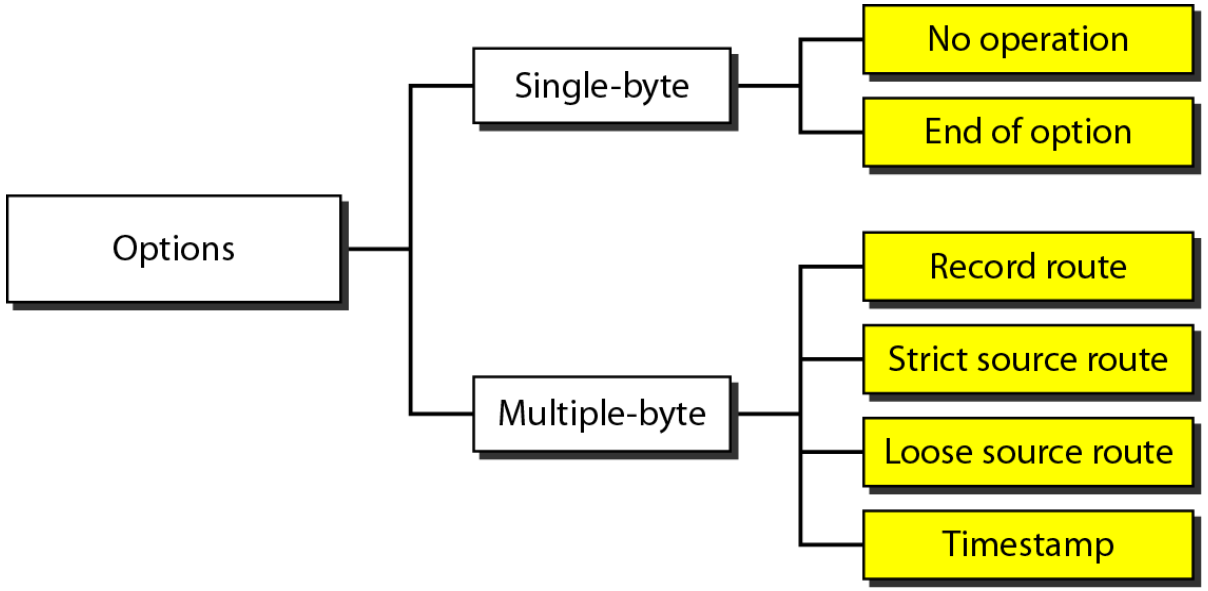
- A datagram of 4000 bytes (header+ data) reached at router and must be forwarded to link with MTU of 1500 bytes . How many fragments will be generated and also write MF, offset and total length value for all fragments.

Solution



Fragmentation – Checksum & Options



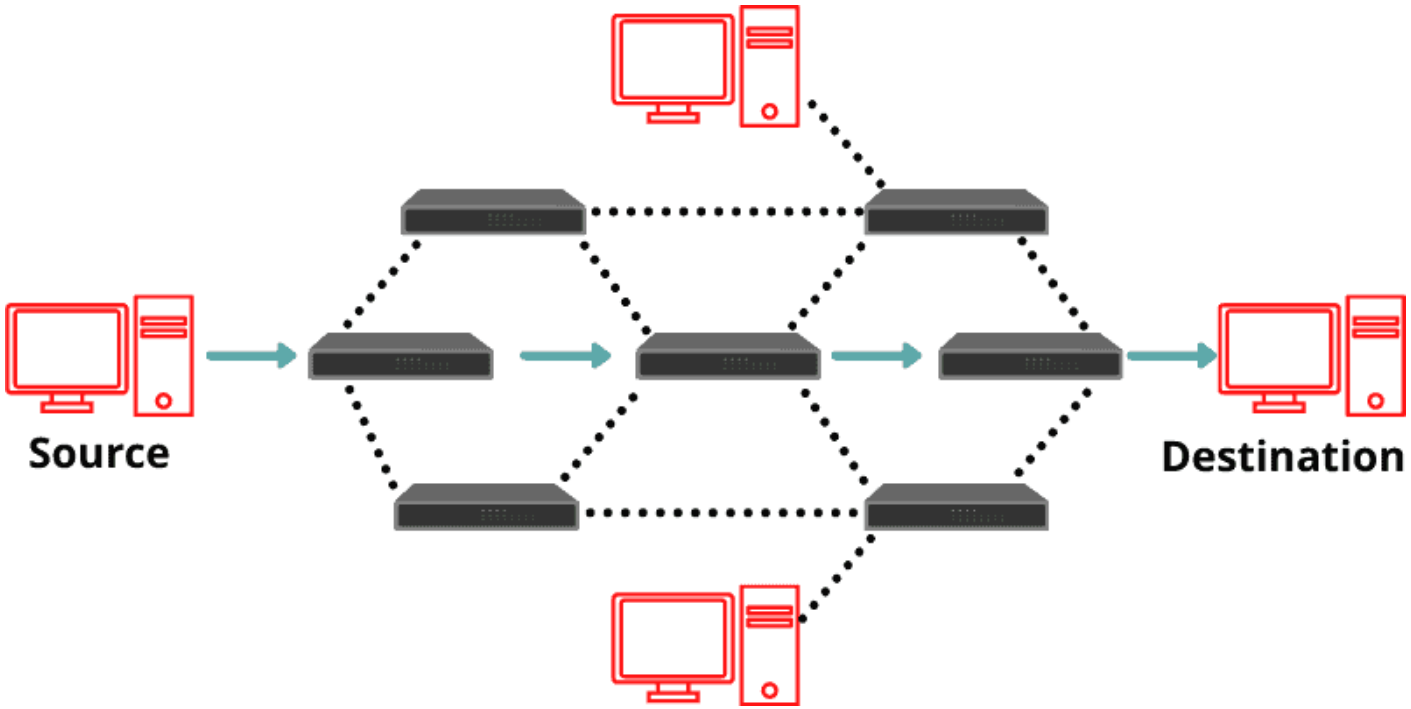


Option and Padding in IPv4

- Record Route option
  - Allow to add intermediate router IP addresse in datagram
  - Maximum 10 routers IP
- Source Routing
  - Strict Source Routing
    - Source system predefine the fix route towards to the destination and added with datagram
  - Loose Source Routing
    - Add only some of the intermediate router IP
- Timestamp
  - Measure the delay (which identification of cost for construction of routing table)

Routing

- In packet-switching networks, such as the Internet, routing selects the paths for Internet Protocol (IP) packets to travel from their origin to their destination. These Internet routing decisions are made by specialized pieces of network hardware called routers.



Routing Protocol Classification

Global or decentralized information?

Global:

- all routers have complete topology, link cost info
- "link state" algorithms

Decentralized:

- router knows physically-connected neighbors, link costs to neighbors
- iterative process of computation, exchange of info with neighbors
- "distance vector" algorithms

Static or dynamic?

Static:

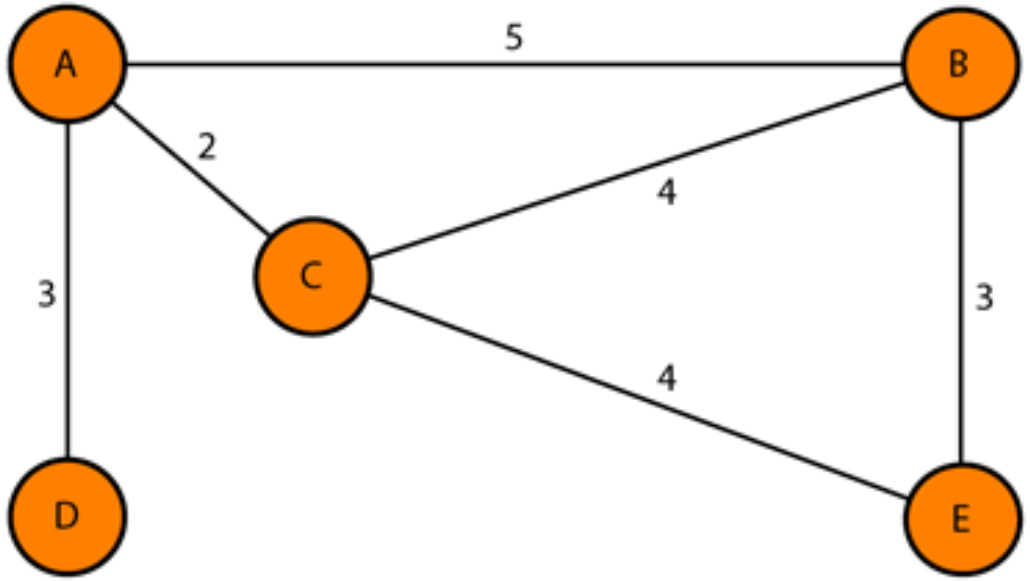
- routes change slowly over time

Dynamic:

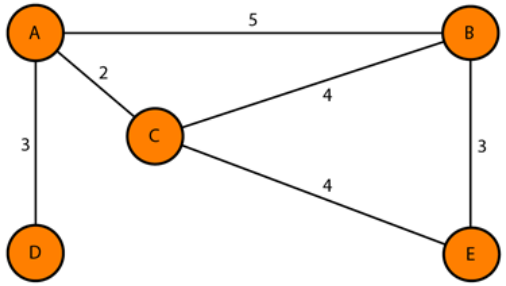
- routes change more quickly
  - periodic update
  - in response to link cost changes

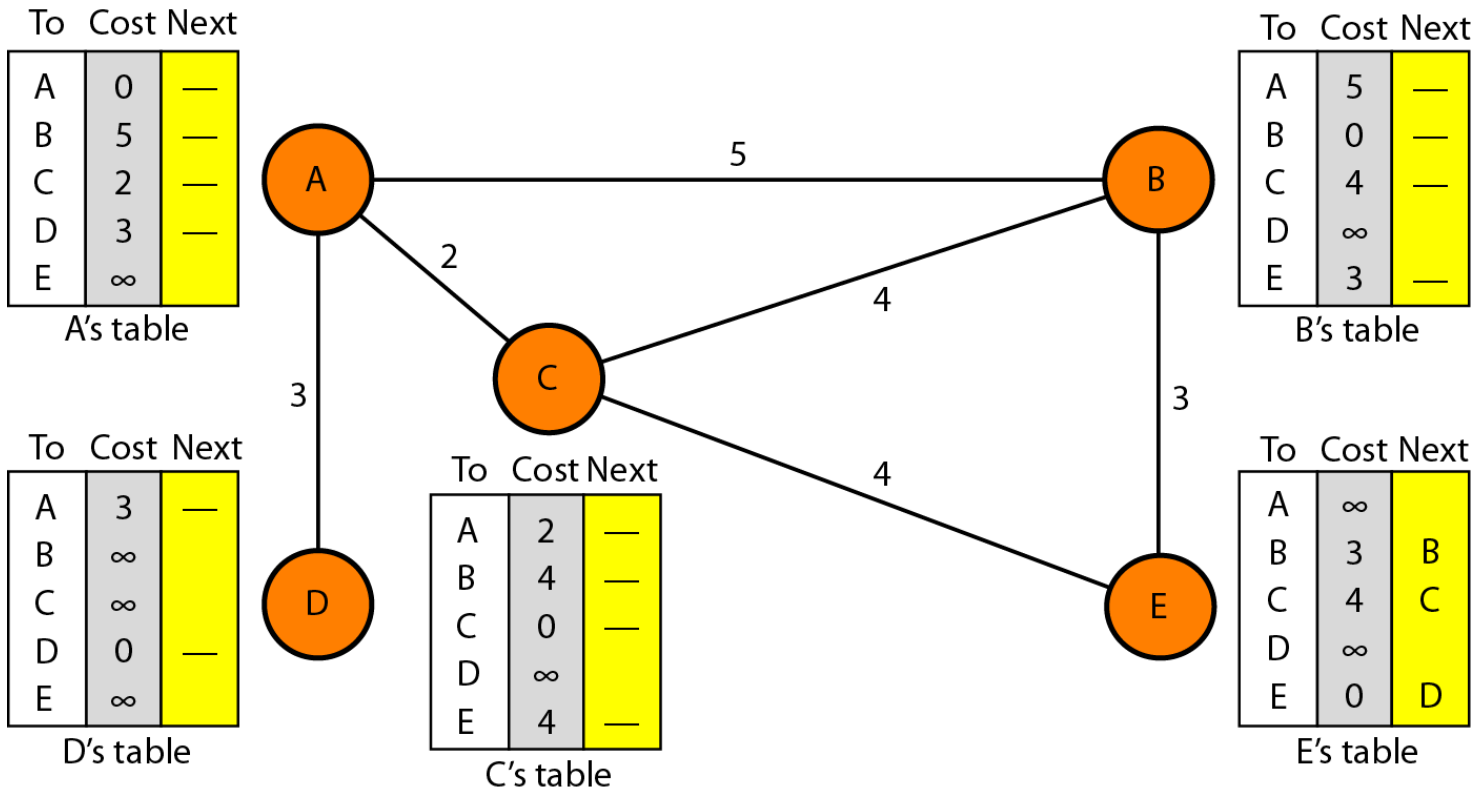
Distance Vector Routing

- The distance vector routing algorithm works by having each router maintain a routing table, giving the best-known distance from source to destination and which route is used to get there. These tables are updated by exchanging the information with the neighbour having a direct link



Distance Vector Routing

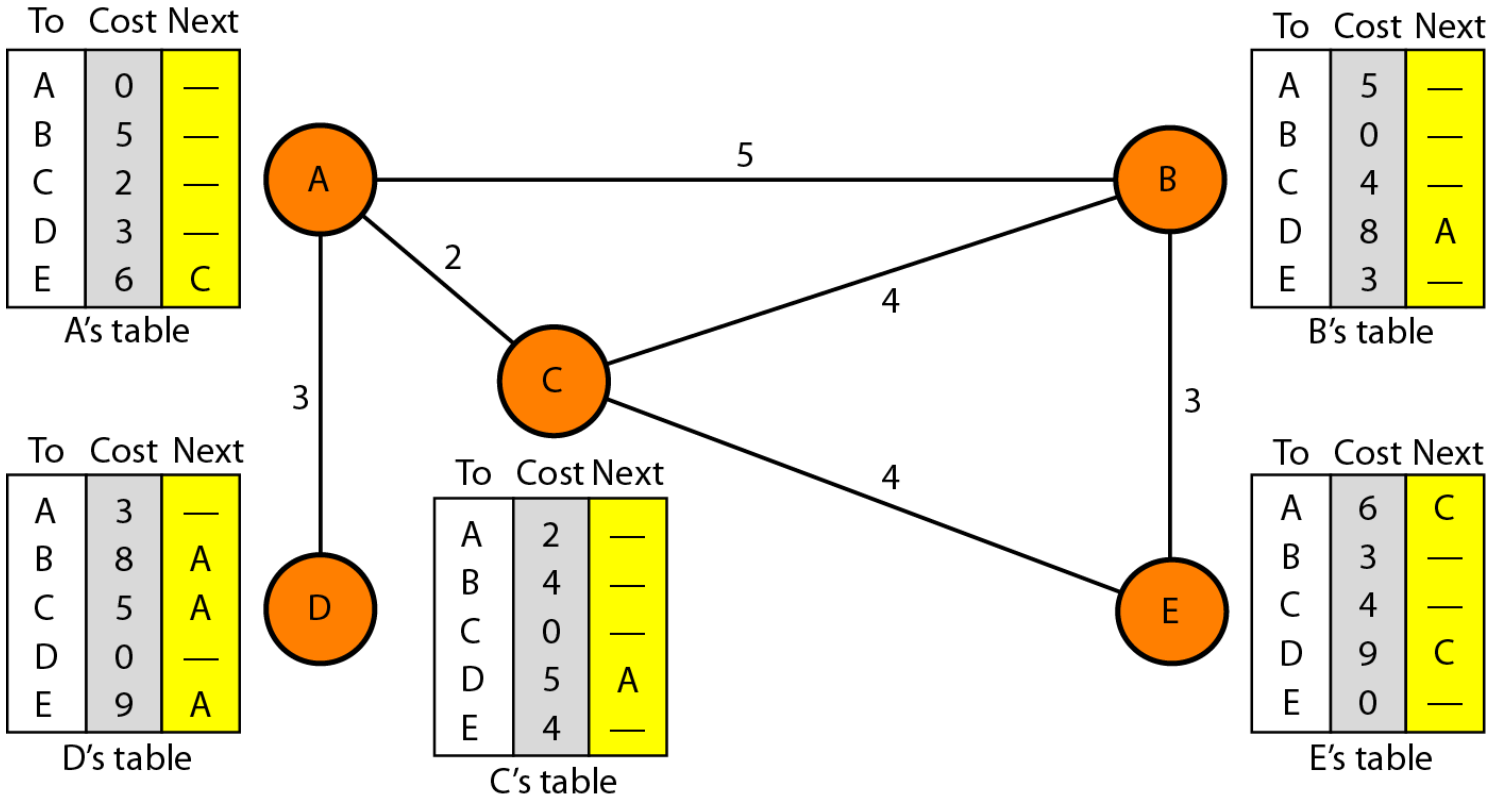




Initialization of tables in distance vector routing

Distance Vector Routing

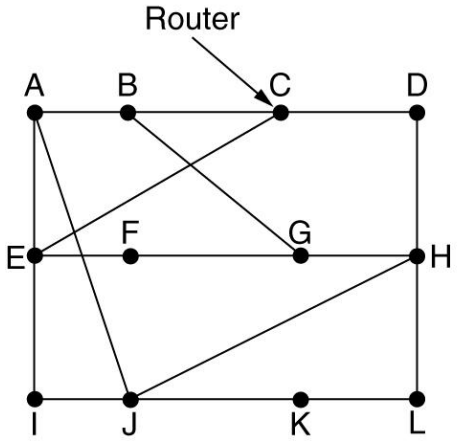
- In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.



Final Routing tables in distance vector routing

Distance Vector Routing

- (a) A subnet. (b) Input from A, I, H, K, and the new routing table for J.



(a)

To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K

JA delay is 8 | JI delay is 10 | JH delay is 12 | JK delay is 6

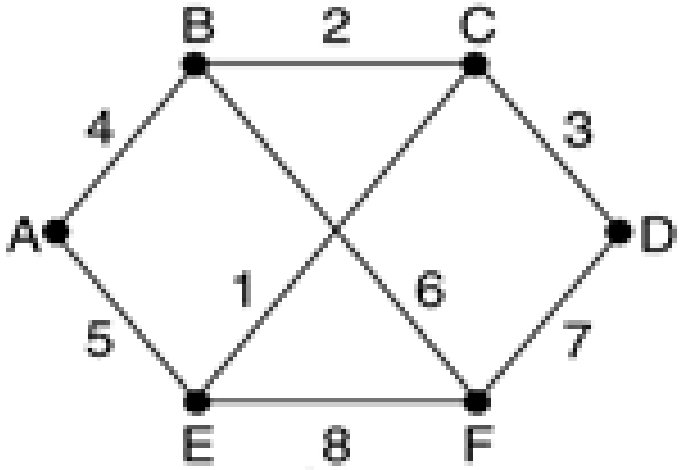
Vectors received from J's four neighbors

New routing table for J

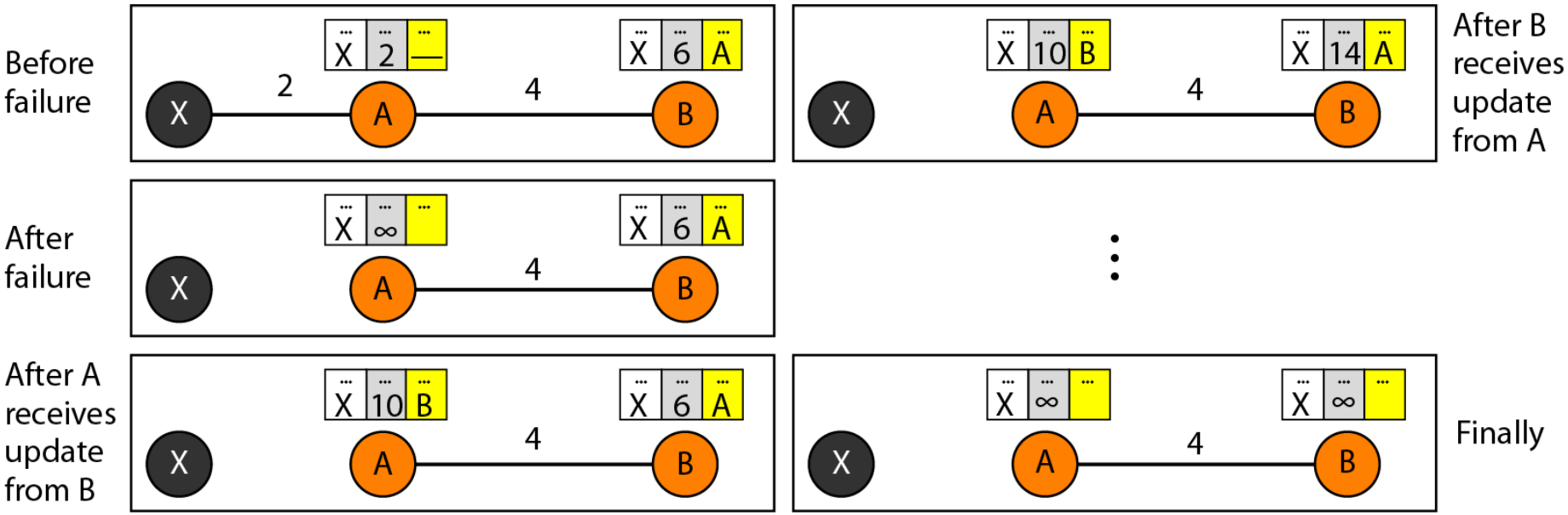
(b)

Distance Vector Routing - Example

- Consider the subnet of in Figure . Distance vector routing is used, and the following vectors have just come in to router C: from B:(5,0,8,12,6,2) from D: (16,12,6,0,9,10) and from E: (7,6,3,9,0,4). The measured delays to B, D and E are 5,4 and 6 respectively. What is C's new routing table? Give both the outgoing line to use and the expected delay



Distance Vector Routing – Count to Infinity Problem



Solution : Split Horizon & poison Reverse

Count to Infinity Problem - Solutions

Defining Infinity:

- Set Maximum value is 16 (limit up to 16 , so can not be used for large system)
- Size should be 15 in each direction

Split Horizon :

- Instead of flooding the table through each interface, each node sends only part of its table through each interface
- E.g. node B thinks that the optimum route to reach X is via A, it does not need to advertise this piece of information to A.

Problem:

- Normally, the DV protocol uses a timer and if there is no news about a route, the node deletes the route from its table
- In the previous e.g., node A cannot guess that this is due to split horizon or because B has not received any news about X recently

Poison Reverse:

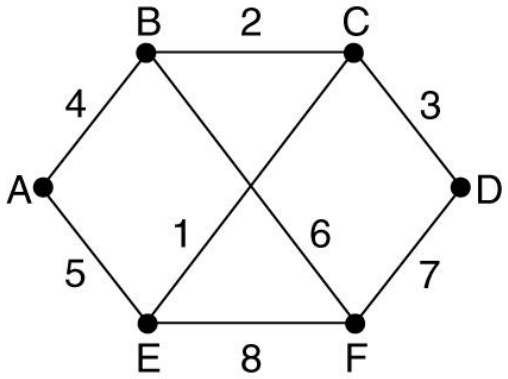
- Node B can still advertise the value for X, but is the source of information is A, it can replace the distance with infinity as a warning

Link State Routing

Each router must do the following five step:

- Discover its neighbors, learn their network address.
- Measure the delay or cost to each of its neighbors.
- Construct a packet telling all it has just learned.
- Send this packet to all other routers.
- Compute the shortest path to every other router.

Link State Routing- Building link state packet



(a)

Link State Packets

A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(b)

(a) A subnet. (b) The link state packets for this subnet.

Packet containing:

Identity of sender

Sequence number + age (TTL)

For each neighbour: name + distance

Link State Routing- Building link state packet

- Sequence packet : each packet contains a sequence no, this keep tracks how many times this packet has been sent
- Age Packet – shows the life of a packet in Network after passing by one Hop it is decremented by one.
- When to build Link State Packets?
  - Build them periodically at regular intervals.
  - Build them when some significant event occurs.

Link State Routing- Distributing link state packet

- Flooding is used to distribute the link state packets to all routers.
- To keep the flood in check, each packet contains a sequence number that is incremented for each new packet sent.
- Routers keep the track of all the (source router, sequence) pairs they see.
- When a new link state packet comes in , it is checked against the list of packets already seen.
- If it is new forwarded on all lines except the one it arrived on. If it is a duplicate, it is discarded.
- When age hits zero, the information from that router is discarded

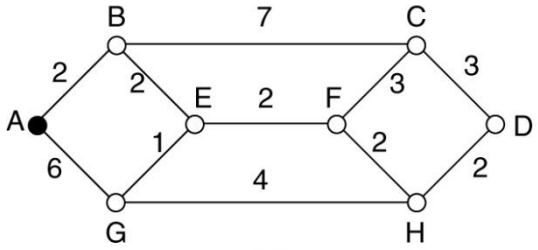
The packet buffer for router B

			Send flags			ACK flags			
Source	Seq.	Age	A	C	F	A	C	F	Data
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

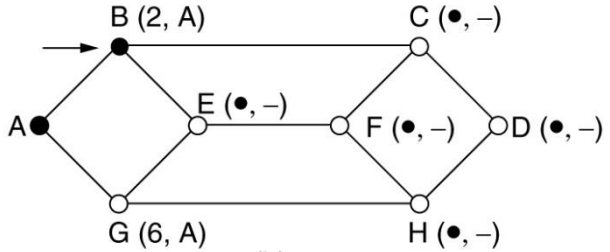
Link State Routing- Computing the New Routes

- Dijkstra's algorithm can be run locally to construct the shortest paths to all possible destinations.
- This algorithm tell the router which link to use to reach each destination. This information is installed in the routing tables.
- Compared to distance vector routing, link state routing require more memory and computation.

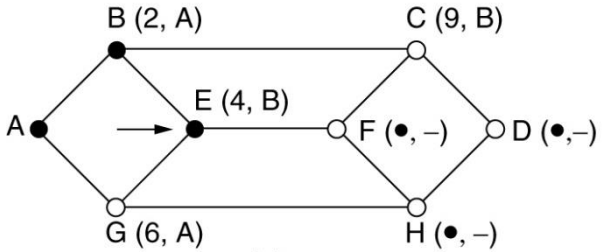
Shortest Path Routing Example



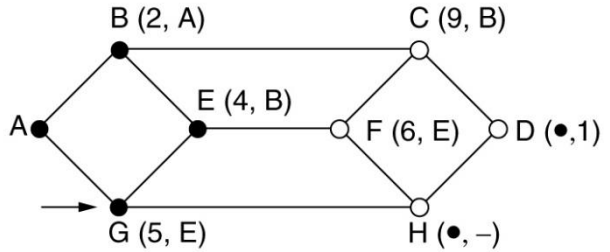
(a)



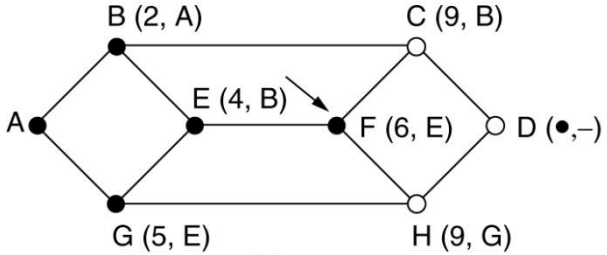
(b)



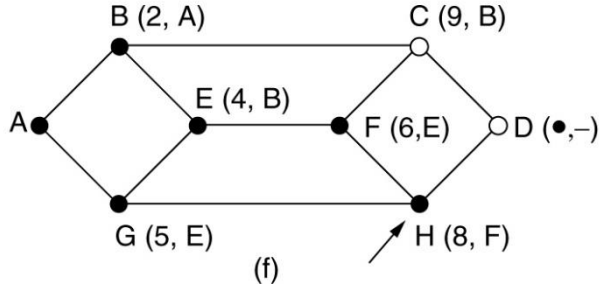
(c)



(d)



(e)



(f)

Hierarchical Routing

Need?

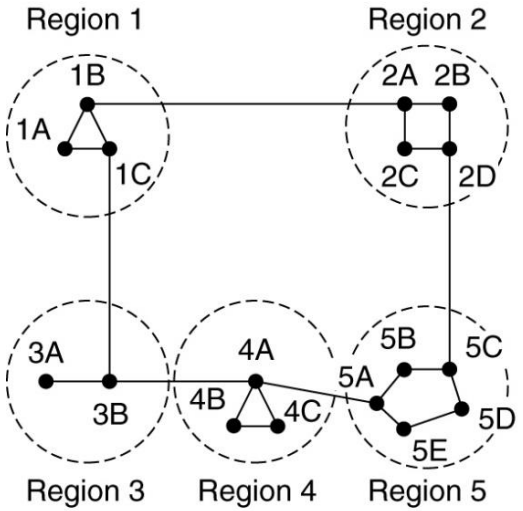
- As networks grow in size, the router routing tables grow proportionally
- Not only is router memory consumed by ever-increasing tables, but more CPU time is needed to scan them and more bandwidth is needed to send status reports about them.
- The network may grow to the point where it is no longer feasible for every router to have entry for every other router

Concept

- In a network, The routers are divided into regions.
- Each router knows all the details about how to route packets to destinations within its own region but knows nothing about the internal structure of other regions .
- For huge networks, a two-level hierarchy may be insufficient; it may be necessary to group into clusters, the clusters into zones, the zones into groups, and so on

Hierarchical Routing

- Lets' Assume source is 1A



(a)

Full table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

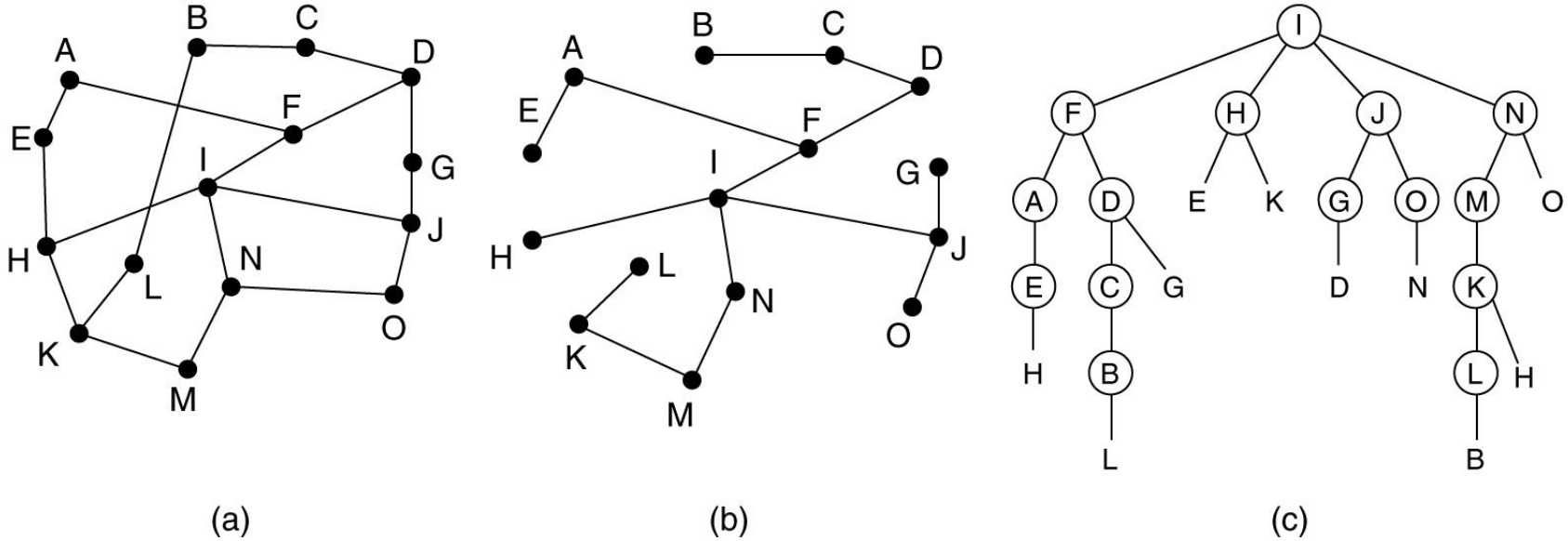
Broadcast Routing

Applications: Stock market updates, distributing weather reports, live radio/tv programs

Methods:

- Source sends packets to each destination (Distinct point to point routing)
- Flooding
- Multi destination routing
- Spanning tree: Subset of a subnet that includes all the routers but contains no loops
- Reverse path forwarding

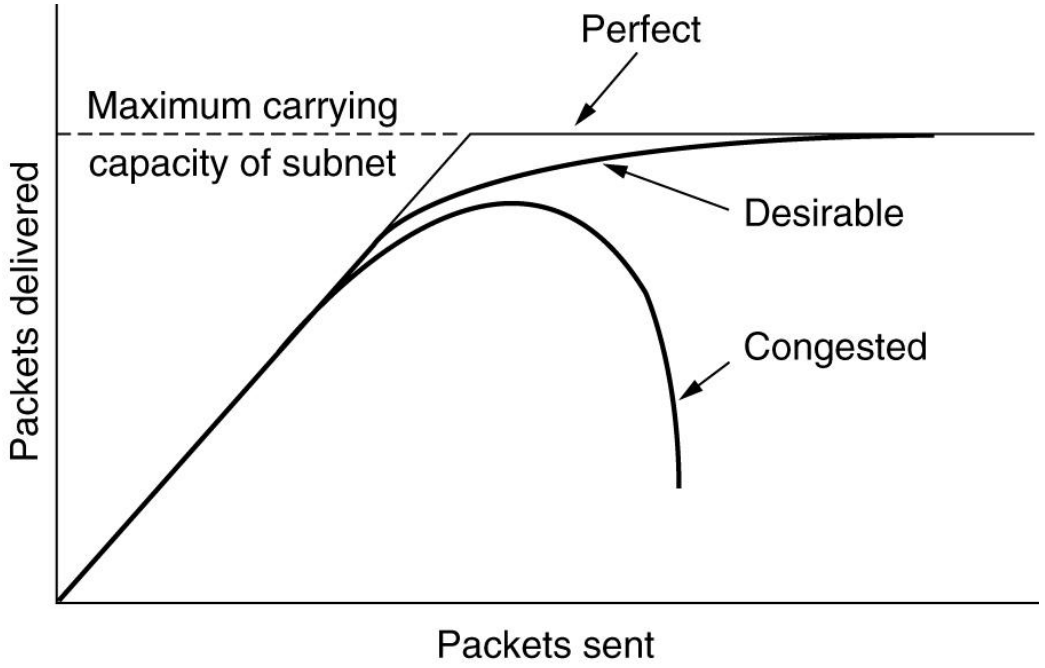
Broadcast Routing



Reverse path forwarding. (a) A subnet. (b) a Sink tree. (c) The tree built by reverse path forwarding.

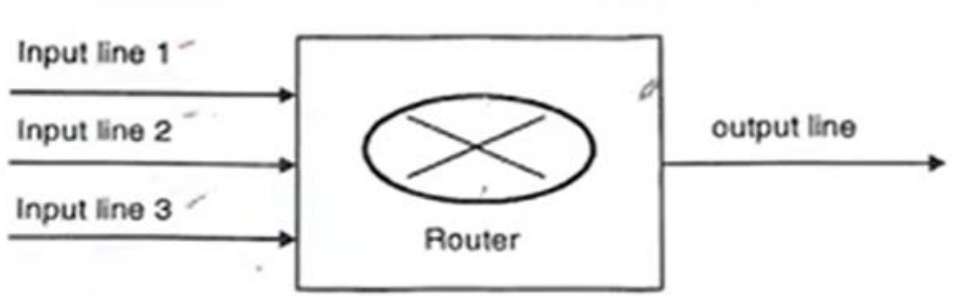
Congestion

- When too much traffic is offered, congestion sets in and performance degrades sharply
- Congestion in a network may occurs when the load on the network is greater than the capacity of the network.

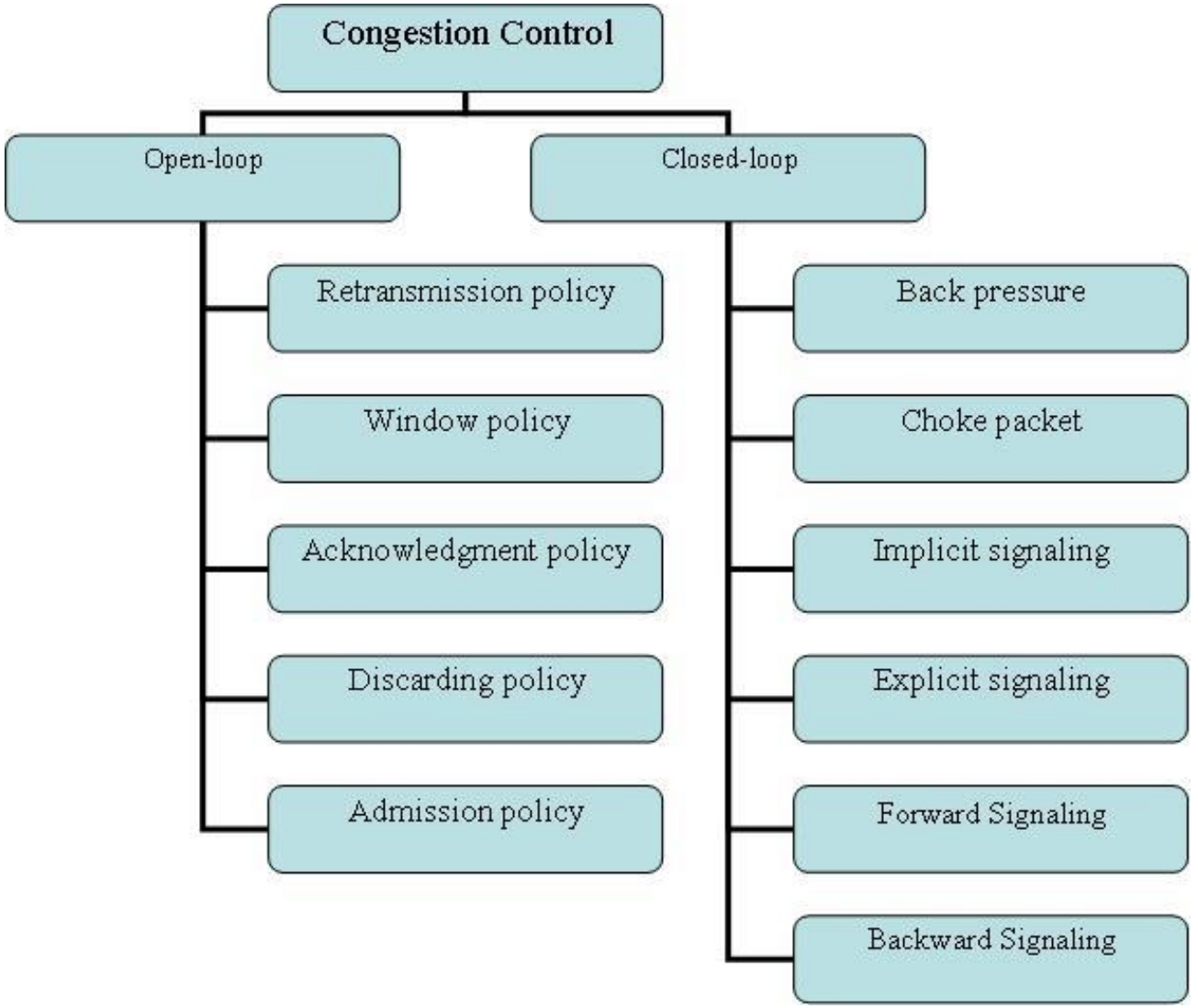


Causes of Network Congestion

- If suddenly, a stream of packet start arriving on three or four inputs and all need the same output lines as shown in figure.
- Congestion in a subnet can occur if the processors are slow.
- Congestion is also caused by slow links.
- Due to retransmission.



Congestion Control Mechanisms



Congestion Control Mechanisms

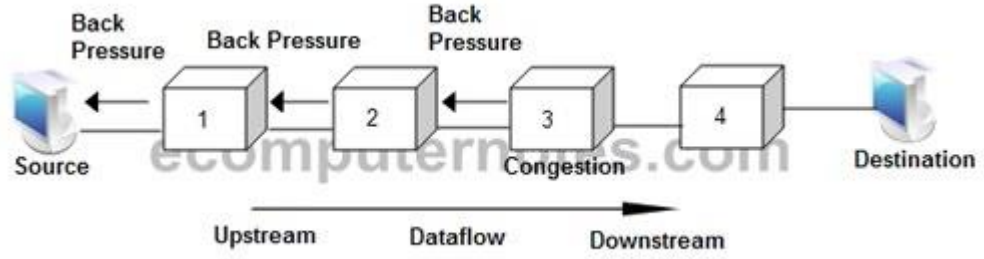
Open loop congestion

- Open-loop congestion control policies are preventing congestion before it happens. Congestion control is handled either by source or by destination host.
- **Re-transmission Policy :** Retransmission is happen either of the sender or receiver feels that data exchanged was corrupted or dropped on the way. Good retransmission policy avoid congestion in the network by increasing the timer for retransmission.
- **Window Policy :**The window used at the sender may also cause congestion. The Selective Repeat window is better than the Go-Back-N window.
- **Acknowledgment Policy:** The acknowledgment policy may also affect congestion. Acknowledgement policies define whether acknowledge each packet or acknowledge a group of packets and so on.
- **Discarding Policy:** A good discard policy also reduce the probability happen congestion. For instance, in a audio transmission, if the strategy is to discard less sensitive packets when congestion is probable to happen, the quality of sound is still preserved and congestion is prevented.
- **Admission Policy :** An admission policy can also prevent congestion. For example, deny access to the channel if there is a chance to congestion.

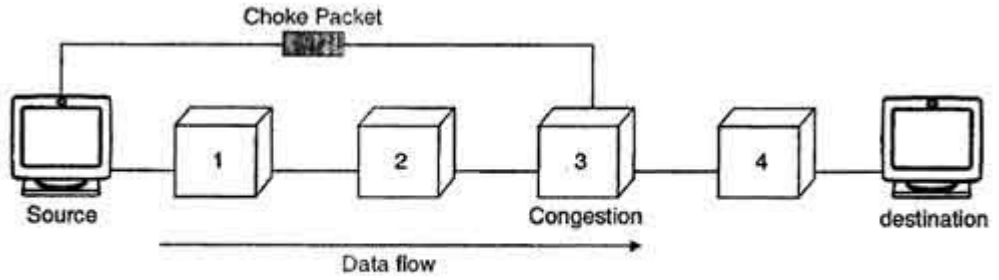
Congestion Control Mechanisms

Close loop congestion

- Closed-loop congestion control methods try to overcome congestion after it happens.
- **Backpressure :** In backpressure method, once a congested then it stops the receiving of packets from its immediate upstream node. This may cause the upstream node to become congested and then it stops receiving of packets from its immediate upstream node. And this process continues up to sender. Sender then stops sending.
- **Choke Packet:** A choke packet is a special packet sent by a congested node to the source to notify congestion. In choke packet method the congested node send chock packet directly to sender it does not notify any intermediate one.



Backpressure



Choke Packet

Congestion Control Mechanisms

Close loop congestion

- **Implicit Signaling :** In this method there is no explicit communication between sender and congested node. The sender determines that there is a congestion in the network implicitly. For example, if there is no acknowledgement for many packets from receiver. Slow down it self.
- **Explicit Signaling:** In this method the congested node explicitly notify sender about congestion. The difference between chock packet and explicit signaling is that, in chock packet the congested node create a special packet to notify congestion but, in explicit signaling the signal is included in the packets that carry data.
- **Forward Signaling:** A bit can be set in a packet moving in the direction of the congestion. This bit notifies the receiver that there is congestion. Then the receiver takes actions to alleviate congestion.
- **Backward Signaling:** A bit can be set in a packet moving in the direction opposite to the congestion. This bit notifies the source that there is a congestion and need to slow down the sending of packets.

Congestion Control Policies used in Layers

Layer	Policies
Transport	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy • Timeout determination
Network	• Virtual circuits versus datagram inside the subnet • Packet queueing and service policy • Packet discard policy • Routing algorithm • Packet lifetime management
Data link	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy

Quality of Service

- Requirements
- Techniques for Achieving Good Quality of Service
- Integrated Services
- Differentiated Services
- Label Switching and MPLS

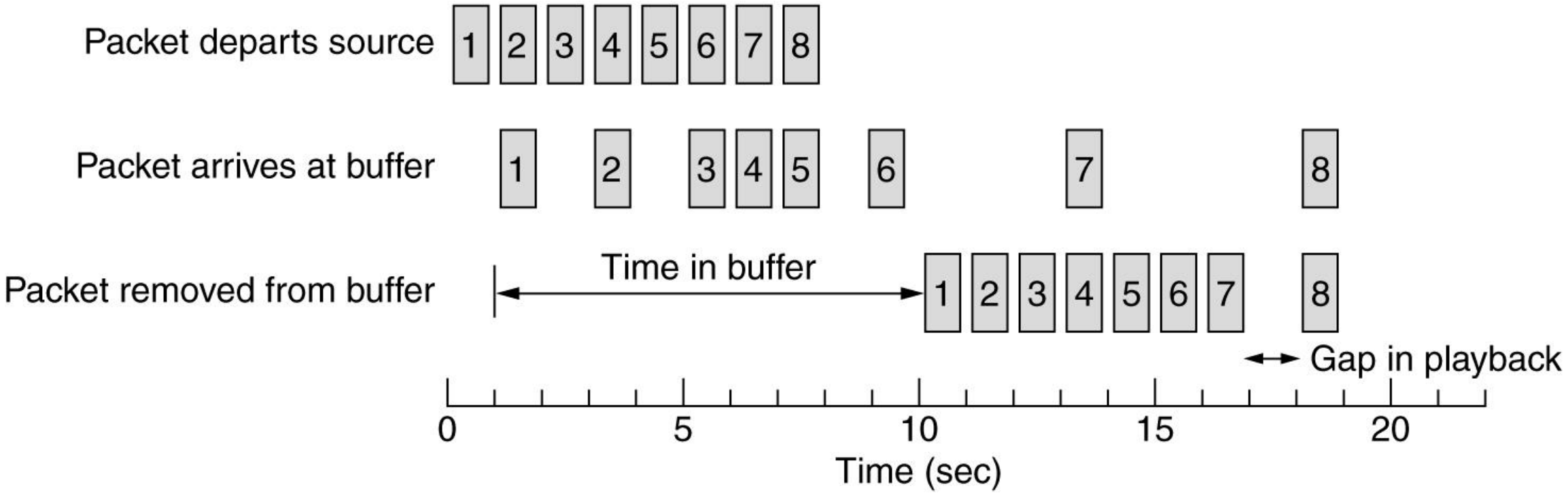
Requirements

Application	Reliability	Delay	Jitter	Bandwidth
E-mail	High	Low	Low	Low
File transfer	High	Low	Low	Medium
Web access	High	Medium	Low	Medium
Remote login	High	Medium	Medium	Low
Audio on demand	Low	Low	High	Medium
Video on demand	Low	Low	High	High
Telephony	Low	High	High	Low
Videoconferencing	Low	High	High	High

Techniques for Achieving Good Quality of Service

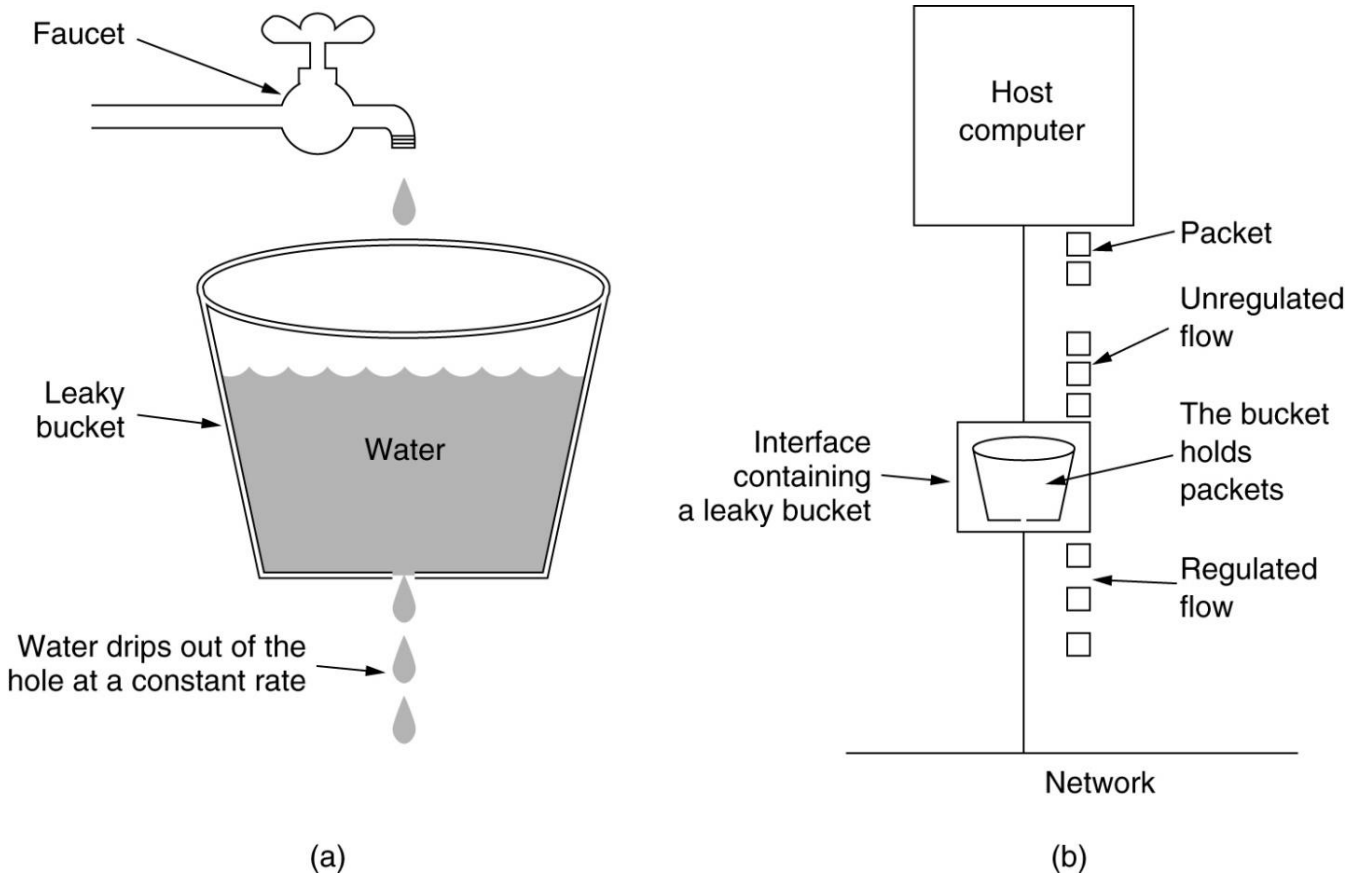
- Overprovisioning
- Buffering
- Traffic Shaping
  - Leaky bucket
  - Token Bucket

Buffering



Smoothing the output stream by buffering packets.

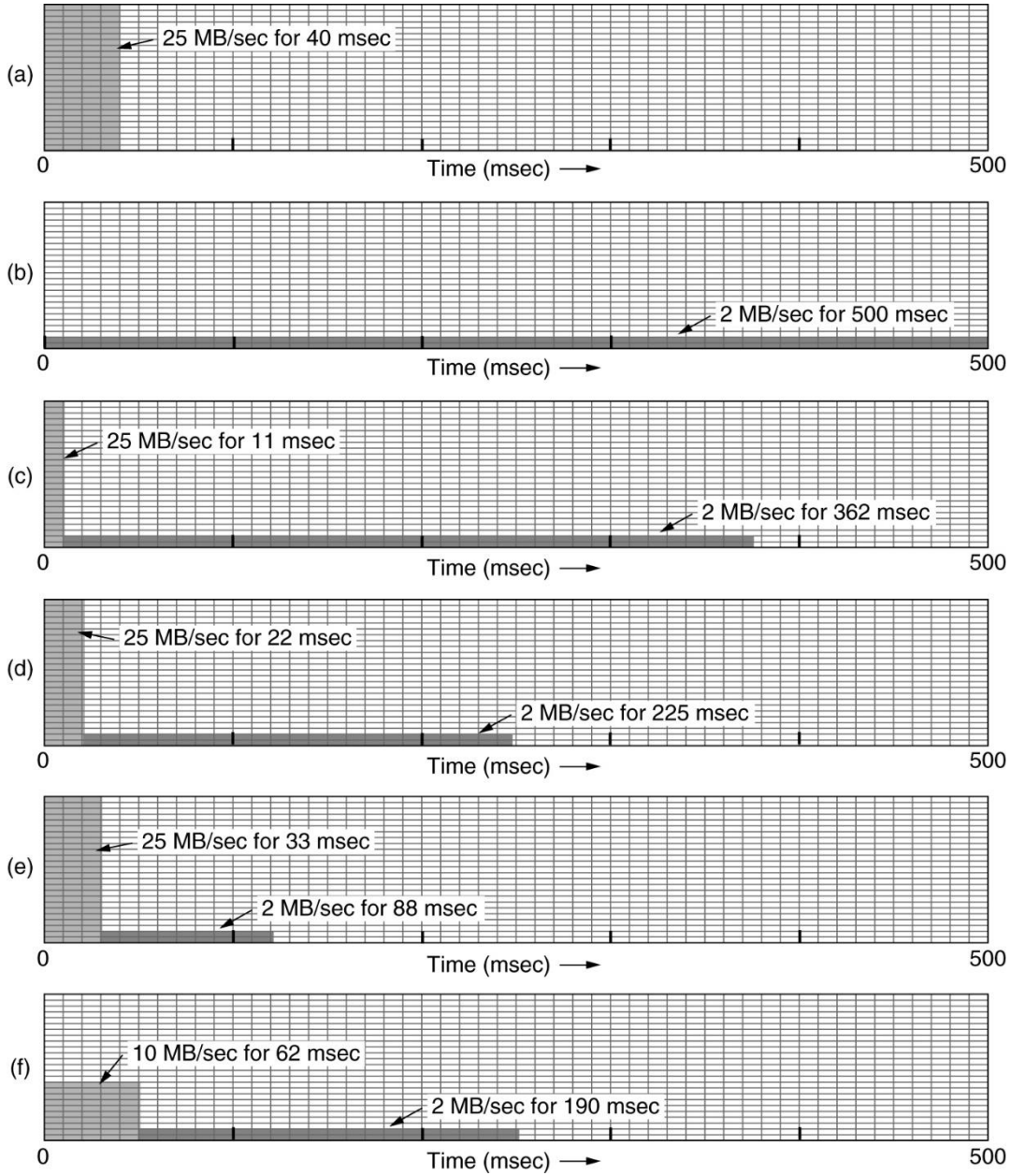
The Leaky Bucket Algorithm



(a) A leaky bucket with water.

(b) A leaky bucket with packets

The Leaky Bucket Algorithm



(a) Input to a leaky bucket.

(b) Output from a leaky bucket. Output from a token bucket with capacities of

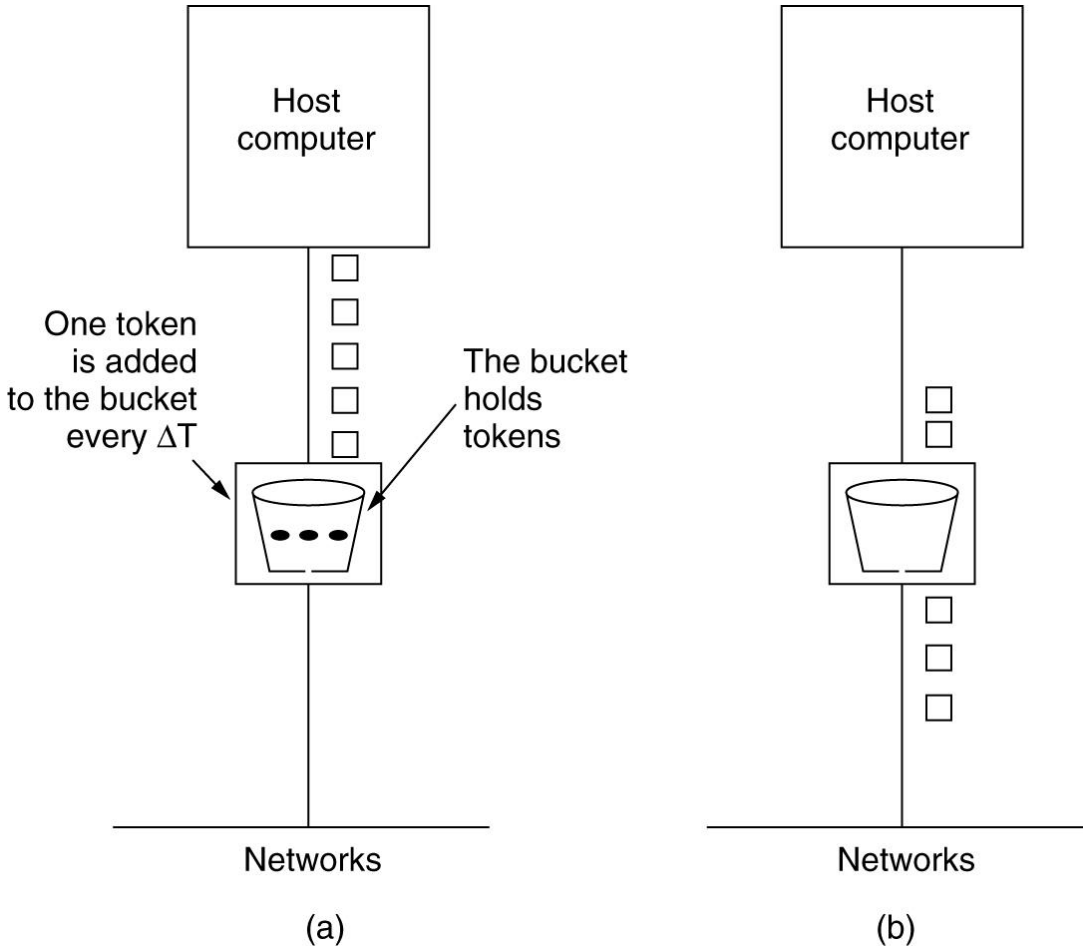
(c) 250 KB,

(d) 500 KB,

(e) 750 KB,

(f) Output from a 500KB token bucket feeding a 10-MB/sec leaky bucket.

The Token Bucket Algorithm



(a) Before. (b) After.