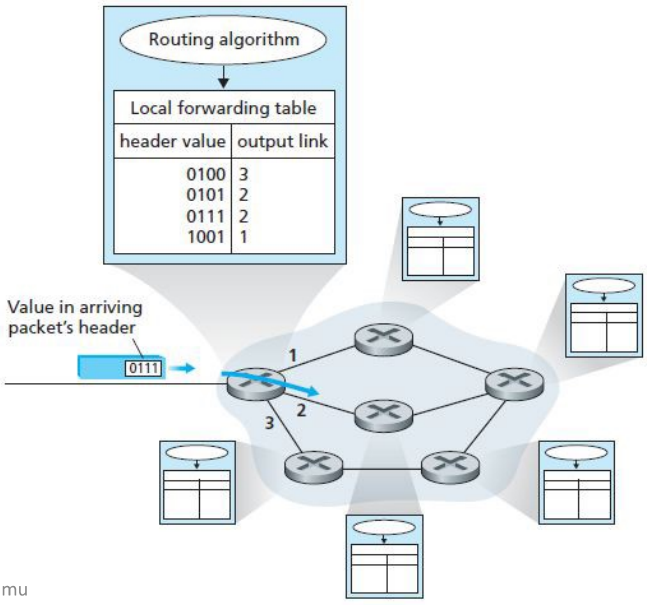
Network Layer: Connection/Connectionless Packet Switching

Network Layer also provide connection oriented and connectionless services in a way similar to the Transport Layer

Computer networks that provide only a connection service at the network layer are called **virtual-circuit (VC) networks**; computer networks that provide only a connectionless service at the network layer are called **datagram networks**



Dr. Gaurav Varshney IIT Jammu

Network Layer: Virtual Circuit Networks

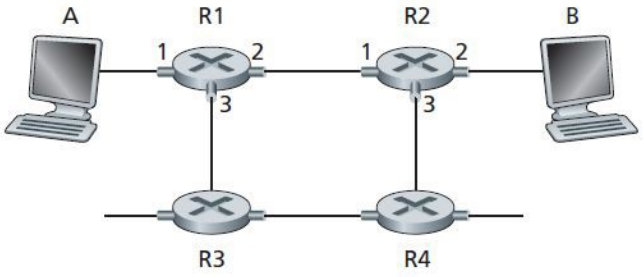
There are three identifiable phases in a virtual circuit:

VC setup. During the setup phase, the sending transport layer contacts the network layer, specifies the receiver's address, and waits for the network to set up the VC. The network layer determines the path between sender and receiver, that is, the series of links and routers through which all packets of the VC will travel.

Data Transfer

VC Teardown

There are global and local addresses[VC Identifier]

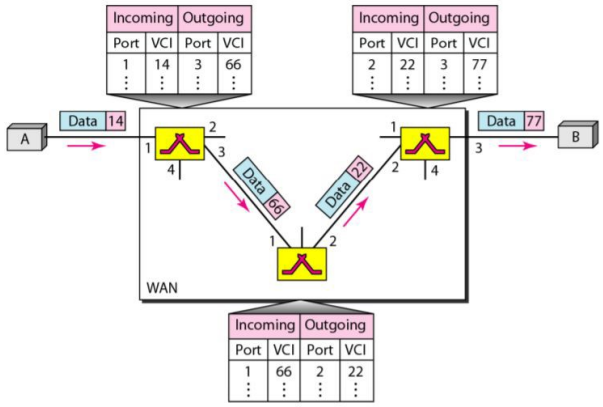


Incoming Interface	Incoming VC #	Outgoing Interface	Outgoing VC #
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87
...	...	...	...

Dr. Gaurav Varshney IIT Jammu

Virtual Circuit Networks

Figure 8.13 *Source-to-destination data transfer in a virtual-circuit network*



Note

In virtual-circuit switching, all packets belonging to the same source and destination travel the same path; but the packets may arrive at the destination with different delays if resource allocation is on demand.

Figure 8.14 *Setup request in a virtual-circuit network*

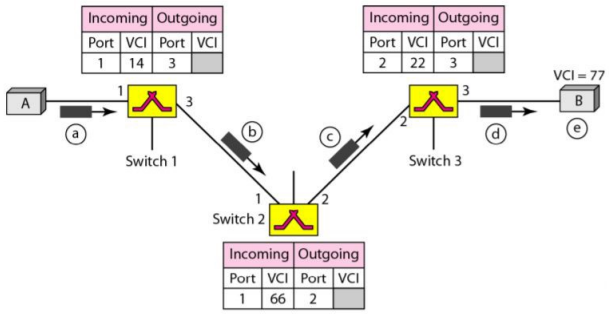
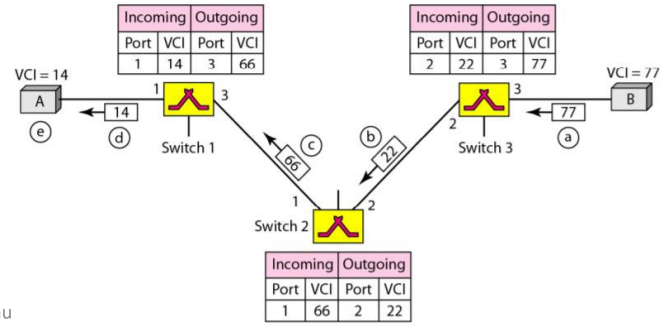


Figure 8.15 *Setup acknowledgment in a virtual-circuit network*



https://www.cpe.ku.ac.th/~plw/dccn/presentation/ch08.pdf

Datagram Networks

In a **datagram network**, each time an end system wants to send a packet, it stamps the packet with the address of the destination end system and then pops the packet into the network.

Specifically, each router has a forwarding table that maps destination addresses to link interfaces; when a packet arrives at the router, the router uses the packet's destination address to look up the appropriate output link interface in the forwarding table.

Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

longest prefix matching rule

Routers in datagram networks maintain no connection state information, they nevertheless maintain forwarding state information in their forwarding tables.

Internet runs as a Datagram Network

Comparison

ISSUE	VIRTUAL CIRCUIT	DATAGRAM
Addressing	Each packet contains a short VC number	Each packet contains the source and the destination address
State Information	State information about each VC is maintained	Does not hold packet level state information
Routing	Route is chosen when VC is setup. All packets follow this route	Each packet is routed independently
Congestion control	Easy if enough buffers can be allocated in advance	Difficult
Resource failure	All VCs passing through the failed resource are terminated	Packets are lost only during resource failure
Suitability	Connection-oriented service	Connection-oriented and connectionless service

Dr. Gaurav Varshney IIT Jammu

Router

An input port performs several key functions

port here—
referring to the physical input and output router interfaces

Control packets (for example, packets carrying routing protocol information) are forwarded from an input port to the routing processor.

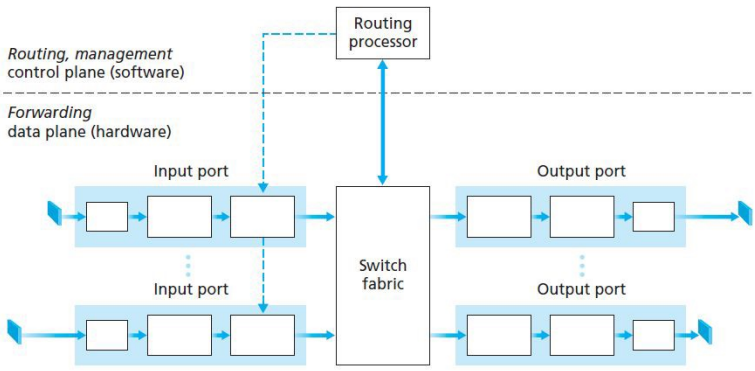


Figure 4.6 ♦ Router architecture

The switching fabric connects the router's input ports to its output ports.

An output port stores packets received from the switching fabric and transmits these packets on the outgoing link by performing the necessary link-layer and physical-layer functions

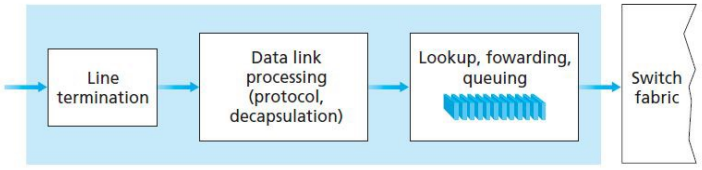


Figure 4.7 ♦ Input port processing

The routing processor executes the routing protocols maintains routing tables and attached link state information, and computes the forwarding table for the router.

Dr. Gaurav Varshney IIT Jammu

Router: Forwarding/Control Plane/Input Processing

A router's input ports, output ports, and switching fabric together implement the forwarding function and are almost always implemented in hardware,

These forwarding functions are sometimes collectively referred to as the **router forwarding plane**

While the forwarding plane operates at the nanosecond time scale, a router's control functions—executing the routing protocols operate at the millisecond or second timescale.

These **router control plane** functions are usually implemented in software and execute on the routing processor

The forwarding table is computed and updated by the routing processor, with a shadow copy typically stored at each input port.

With a shadow copy, forwarding decisions can be made locally, at each input port, without invoking the centralized routing processor on a per-packet basis and thus avoiding a centralized processing bottleneck.

With a TCAM [Ternary Content Addressable Memory], a 32-bit IP address is presented to the memory, which returns the content of the forwarding table entry for that address in essentially constant time.

"match plus action" abstraction

Switching

The switching fabric is at the very heart of a router, as it is through this fabric that the packets are actually switched (that is, forwarded) from an input port to an output port.

Memory: Switching between input and output ports being done under direct control of the CPU (routing processor).

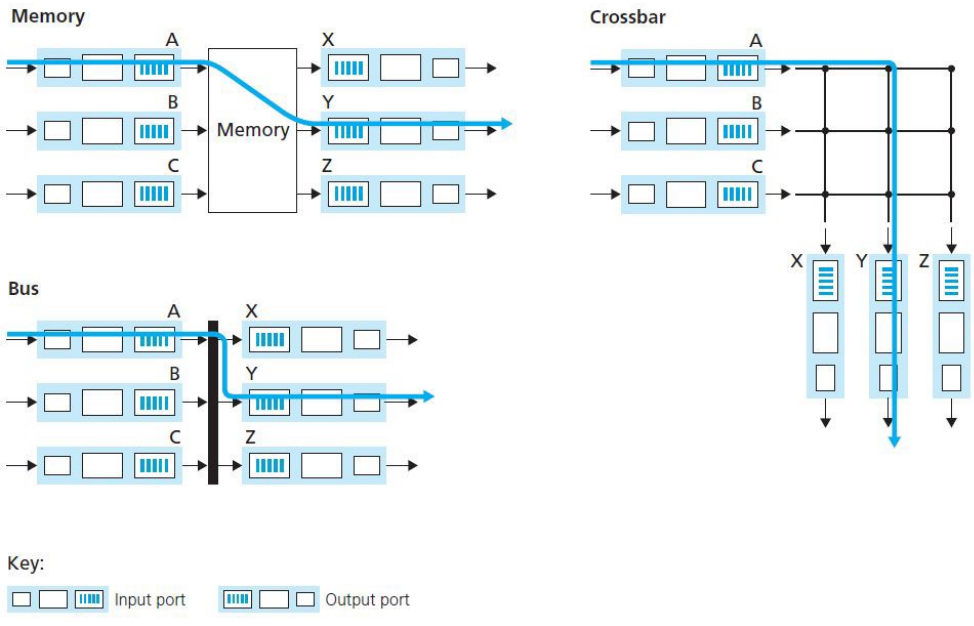


Figure 4.8 ♦ Three switching techniques

if the memory bandwidth is such that B packets per second can be written into, or read from, memory, then the overall forwarding throughput (the total rate at which packets are transferred from input ports to output ports) must be less than $B/2$

Switching

Switching via a bus. In this approach, an input port transfers a packet directly to the output port over a shared bus

This is typically done by having the input port pre-pend a switch-internal label (header) to the packet indicating the local output port to which this packet is being transferred and transmitting the packet onto the bus.

The packet is received by all output ports, but only the port that matches the label will keep the packet.

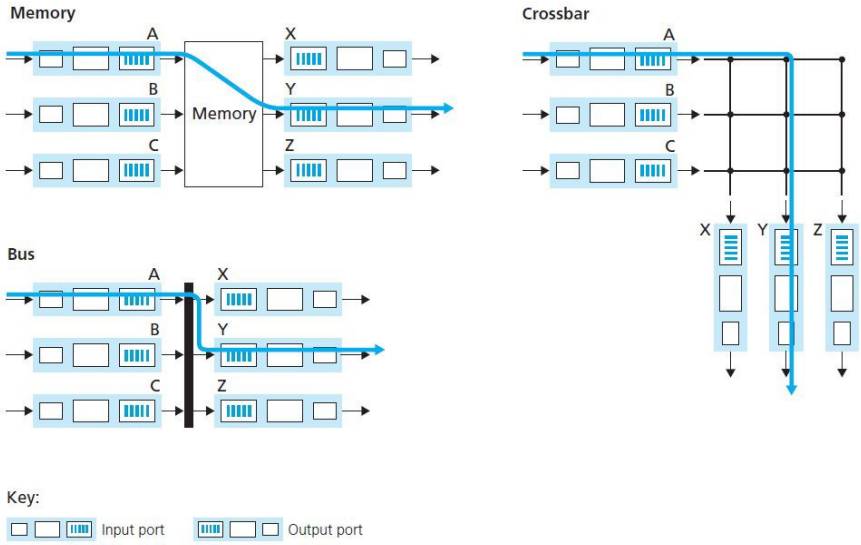


Figure 4.8 ♦ Three switching techniques

If multiple packets arrive to the router at the same time, each at a different input port, all but one must wait since only one packet can cross the bus at a time.

Because every packet must cross the single bus, the switching speed of the router is limited to the bus speed;

Dr. Gaurav Varshney IIT Jammu

Switching

A crossbar switch is an interconnection network consisting of 2*N* buses that connect *N* input ports to *N* output ports

Each vertical bus intersects each horizontal bus at a cross point, which can be opened or closed at any time by the switch fabric controller

When a packet arrives from port A and needs to be forwarded to port Y, the switch controller closes the crosspoint at the intersection of busses A and Y, and port Athen sends the packet onto its bus, which is picked up (only) by bus Y.

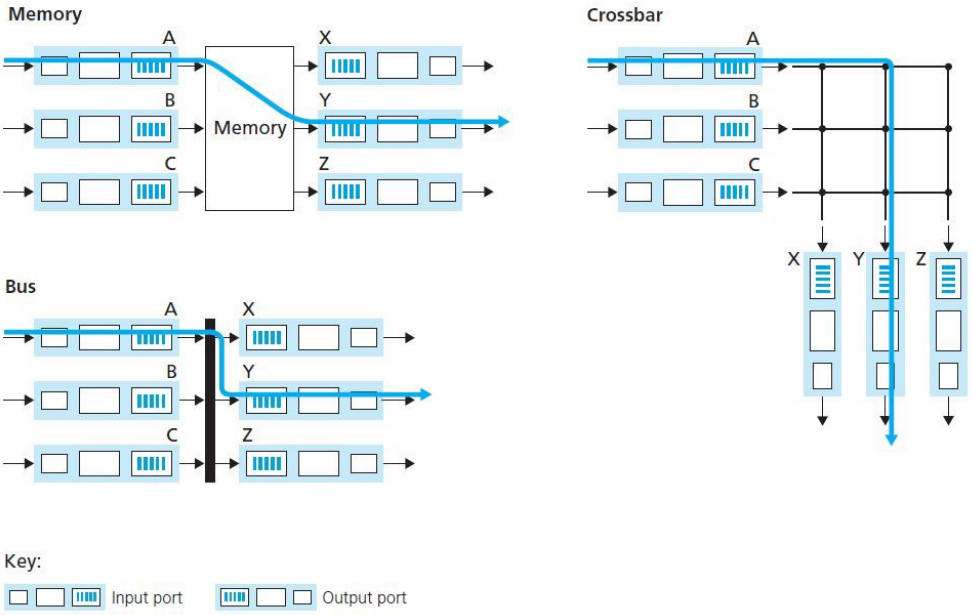


Figure 4.8 ♦ Three switching techniques

Dr. Gaurav Varshney IIT Jammu

Output Processing

packet queues may form at both the input ports *and* the output ports

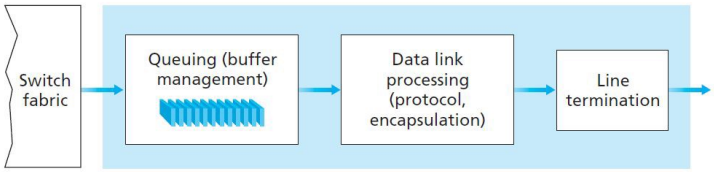


Figure 4.9 ♦ Output port processing

The location and extent of queueing (either at the input port queues or the output port queues) will depend on the traffic load, the relative speed of the switching fabric, and the line speed

A consequence of output port queuing is that a **packet scheduler** at the output port must choose one packet among those queued for transmission

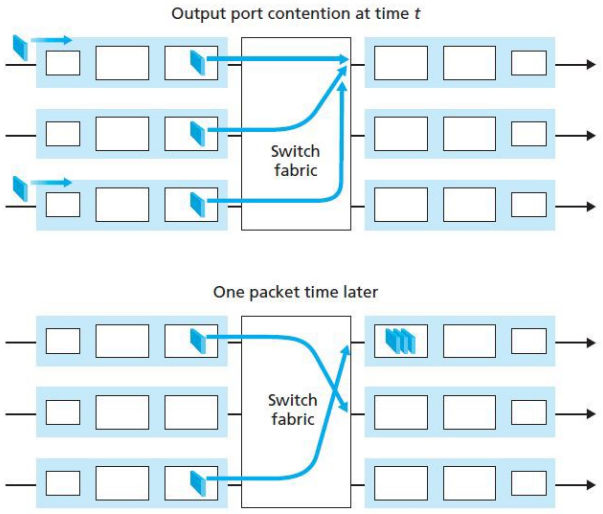


Figure 4.10 ♦ Output port queuing

Switching: Head Of Line (HOL) Blocking

input-queued switch—a queued packet in an input queue must wait for transfer through the fabric (even though its output port is free) because it is blocked by another packet at the head of the line.

Solution: Virtual Output Queue.

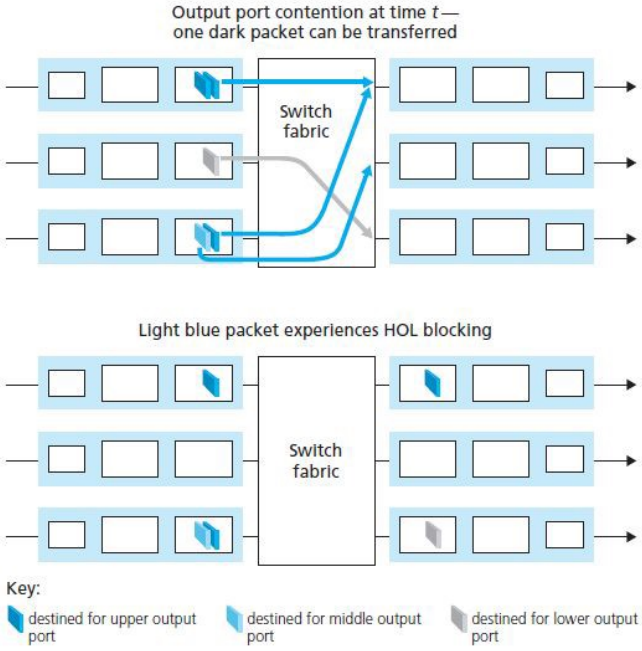


Figure 4.11 ♦ HOL blocking at an input queued switch

IP: Forwarding and Addressing

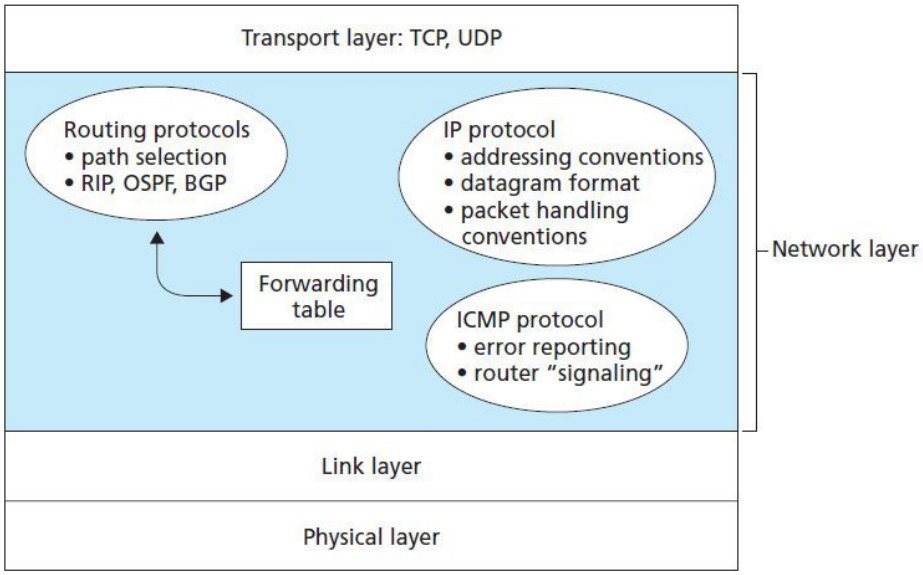


Figure 4.12 ♦ A look inside the Internet's network layer

IPV4

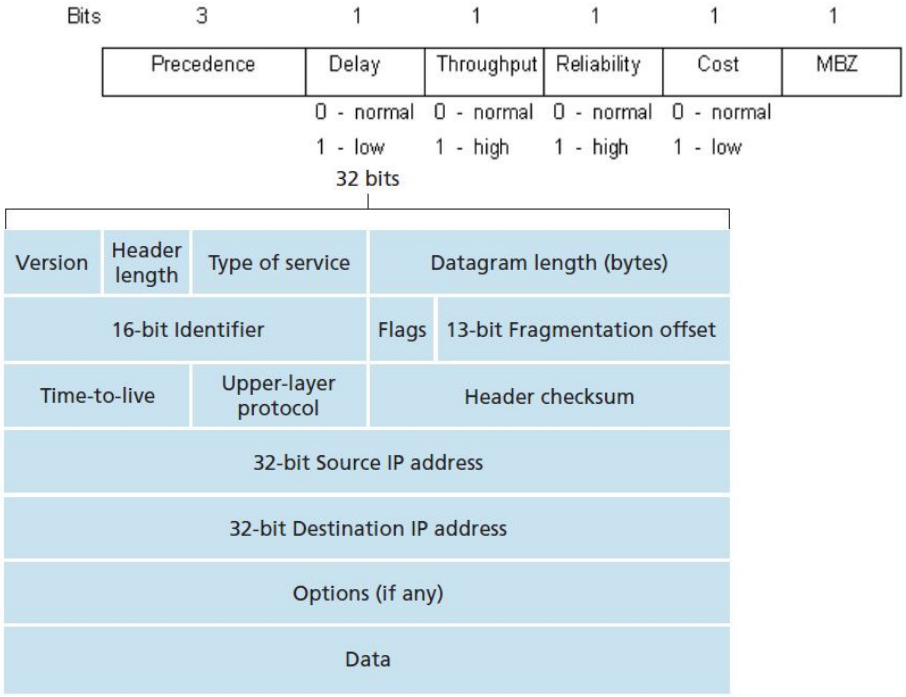
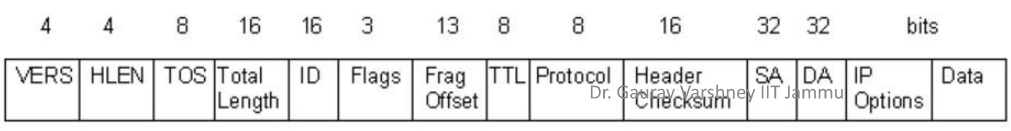
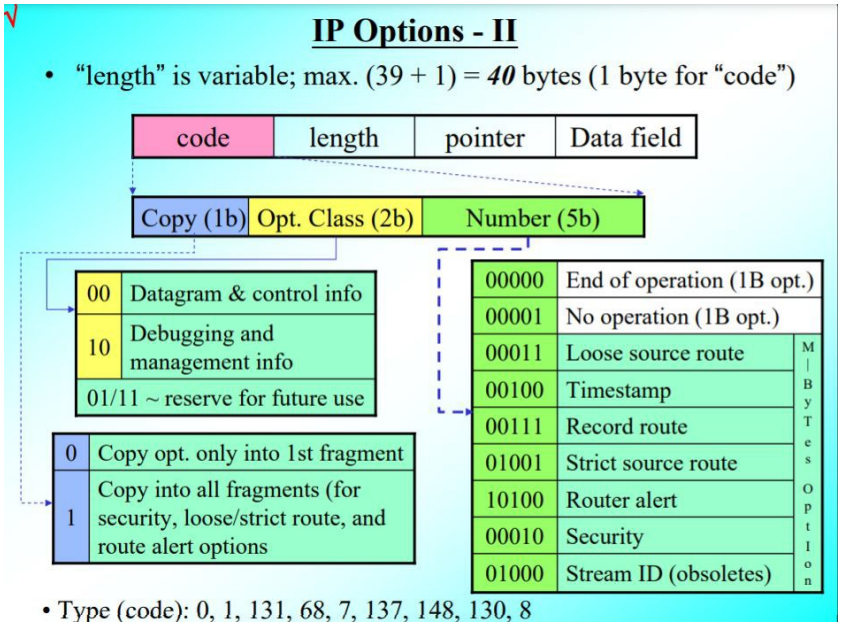


Figure 4.13 ♦ IPv4 datagram format

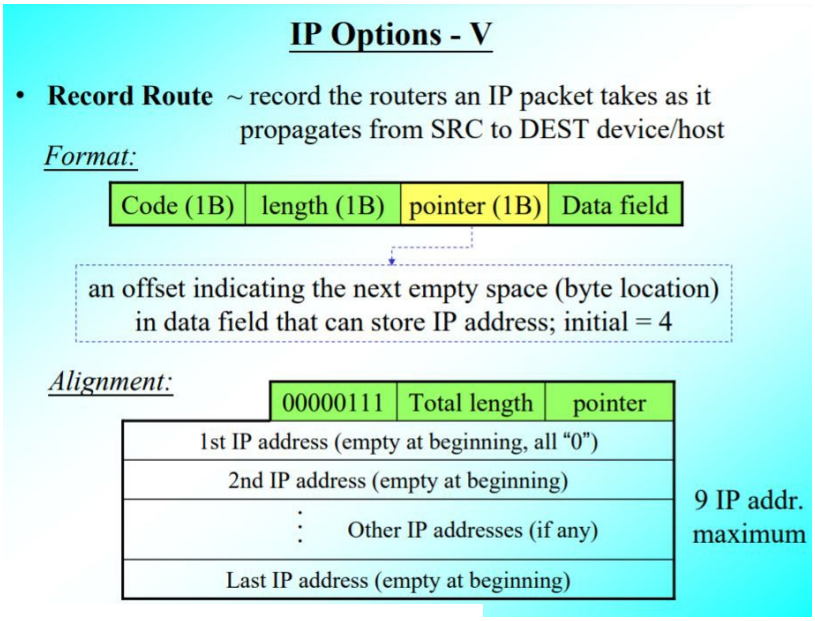


https://www.eit.lth.se/ppplab/IPHeader.htm

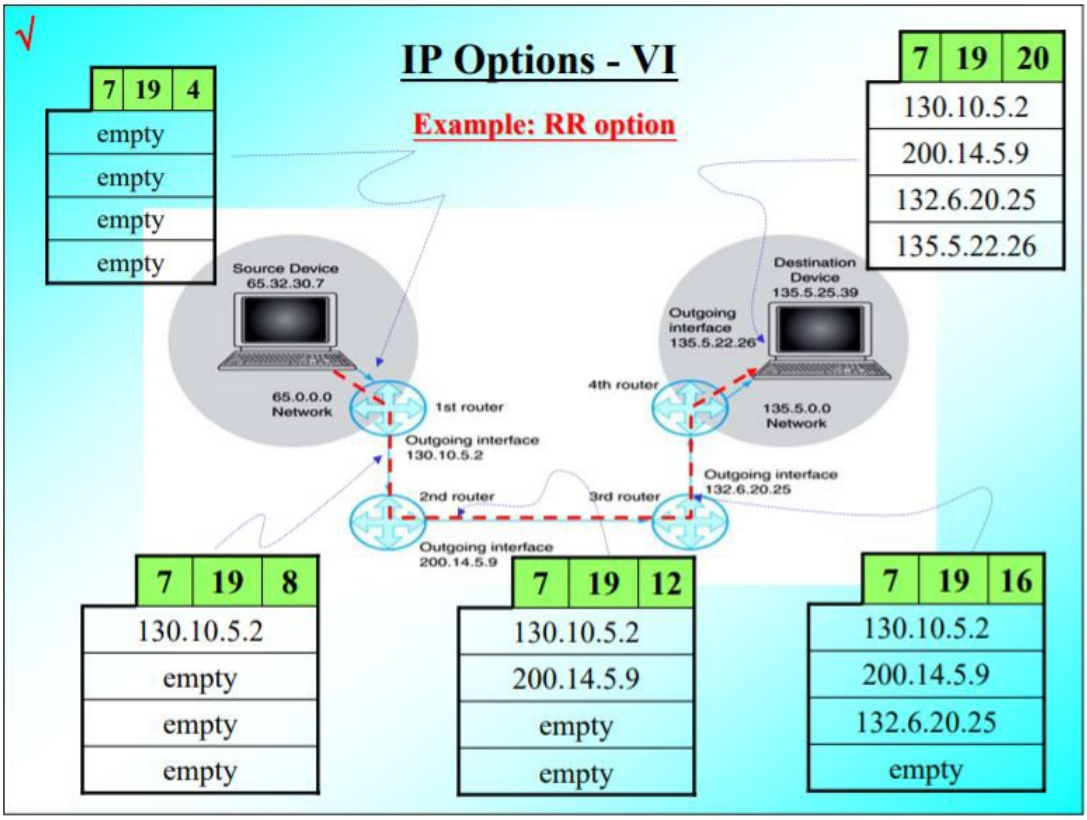
Decimal	Keyword	Protocol
0	HOPOPT	IPv6 Hop-by-Hop Option
1	ICMP	Internet Control Message
2	IGMP	Internet Group Management
3	GGP	Gateway-to-Gateway
4	IPv4	IPv4 encapsulation
5	ST	Stream
6	TCP	Transmission Control
7	CBT	CBT
8	EGP	Exterior Gateway Protocol
9	IGP	any private interior gateway (used by Cisco for their IGRP)
10	BBN-RCC-MON	BBN RCC Monitoring
11	NVP-II	Network Voice Protocol
12	PUP	PUP
13	ARGUS (deprecated)	ARGUS
14	EMCON	EMCON
15	XNET	Cross Net Debugger
16	CHAOS	Chaos
17	UDP	User Datagram



https://myweb.ntut.edu.tw/~kwke/DC2006/ipo.pdf




```
Options: (24 bytes), Record Route
  IP Option - Record Route (23 bytes)
    Type: 7
    Length: 23
    Pointer: 4
    Empty Route: 0.0.0.0 <- (next)
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
    Empty Route: 0.0.0.0
```



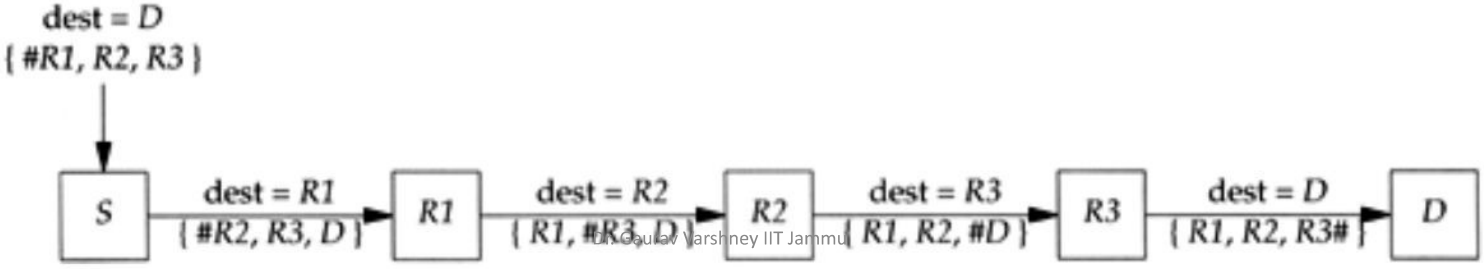
- **Strict source route** (code = 137)
  - The SRC provides a list of hops (router interfaces) to specify the exact path the packet must take to reach its DEST host
- **Loose source route** (code = 131)
  - Similar to SSR, but more forgiving

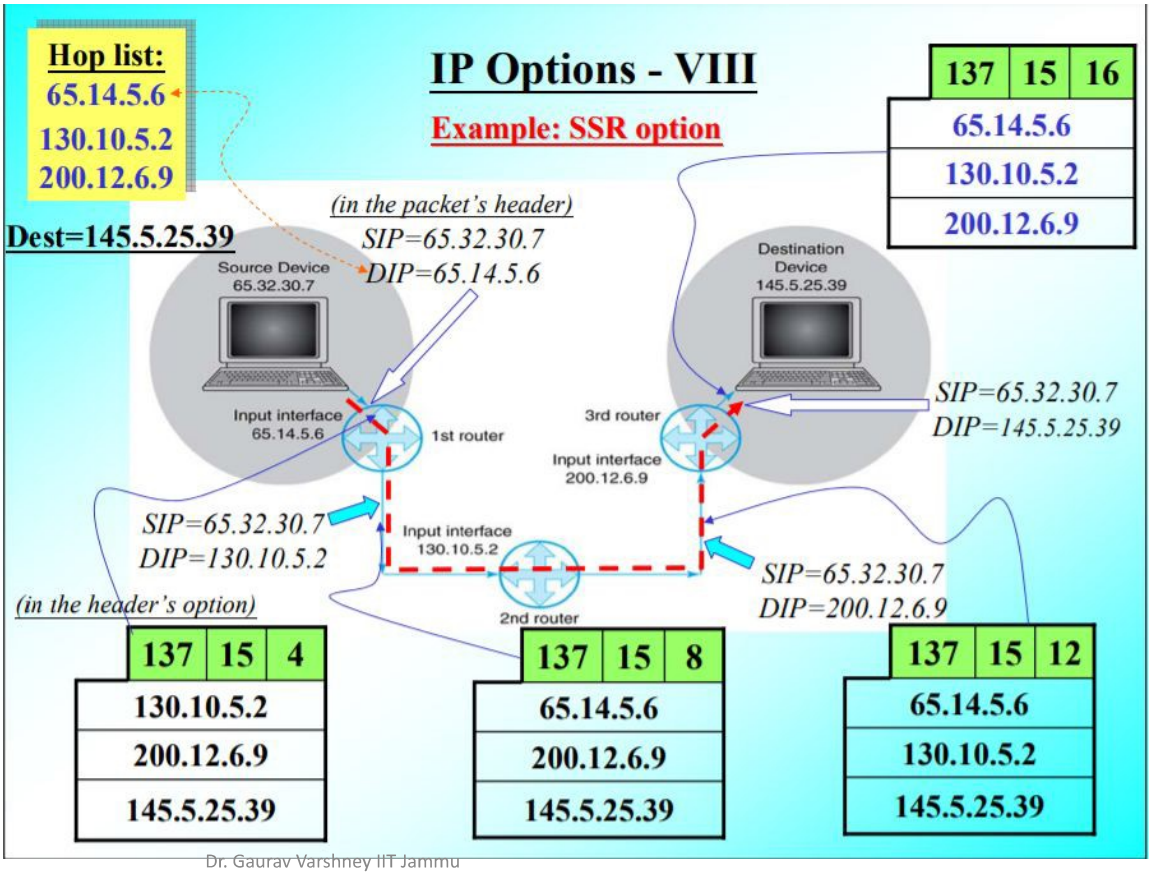
SSR/LSR

137/131	Total length	pointer
1st Dest IP address = DIP in IP header		
2nd Dest IP address		
Other Dest IP addresses		
Last Dest IP address (the Dest host)		

RA

Code (1B)	length (1B)	Data field (2B)

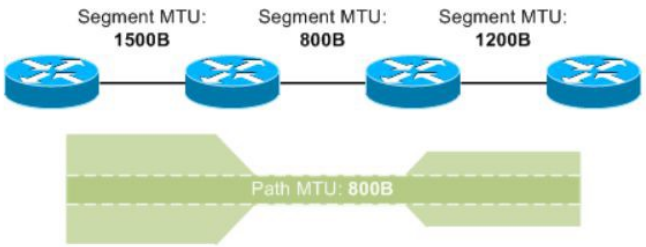


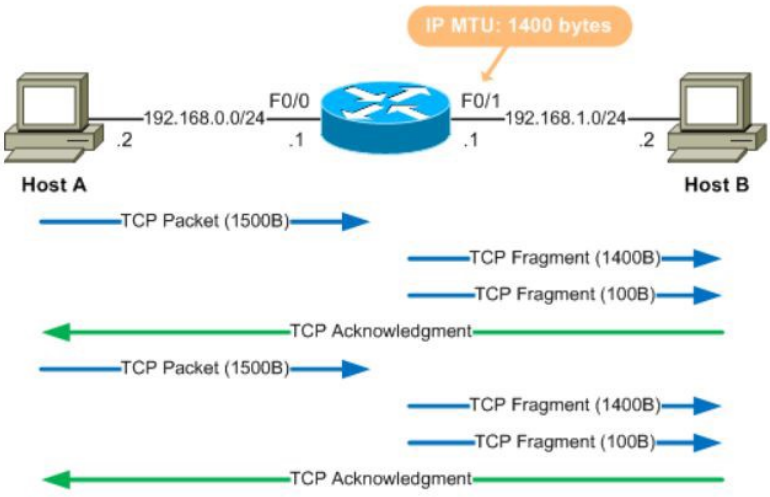



```
Internet Protocol Version 4, Src: 192.168.79.209, Dst: 14.139.53.140, Via: 192.168.0.1
    0100 .... = Version: 4
    .... 1001 = Header Length: 36 bytes (9)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 76
    Identification: 0xf7a7 (63399)
  Flags: 0x0000
      0... .... .... .... = Reserved bit: Not set
      .0.. .... .... .... = Don't fragment: Not set
      ..0. .... .... .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 128
    Protocol: ICMP (1)
    Header checksum: 0xc68c [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.79.209
    Current Route: 192.168.0.1
  Options: (16 bytes), Strict Source Route
    IP Option - Strict Source Route (15 bytes)
      Type: 137
      Length: 15
      Pointer: 4
      Source Route: 127.0.0.1 <- (next)
      Source Route: 135.0.0.1
      Destination: 14.139.53.140
    IP Option - End of Options List (EOL)
      Type: 0
```


Dr. Gaurav Varshney IIT Jammu

IP Fragmentation





When a host needs to transmit data out an interface, it references the interface's *Maximum Transmission Unit (MTU)* to determine how much data it can put into each packet. Ethernet interfaces, for example, have a default MTU of 1500 bytes, not including the Ethernet header or trailer. This means a host needing to send a TCP data stream would typically use the first 20 of these 1500 bytes for the IP header, the next 20 for the TCP header, and as much of the remaining 1460 bytes as necessary for the data payload. Encapsulating data in maximum-size packets like this allows for the least possible consumption of bandwidth by protocol overhead.

https://packetlife.net/blog/2008/aug/18/path-mtu-discovery/

IP Fragmentation

Ethernet v2	1500[11]
Ethernet with LLC and SNAP	1492[12]
Ethernet jumbo frames	1501 – 9202[13] or more[14]
PPPoE v2	1492[16]

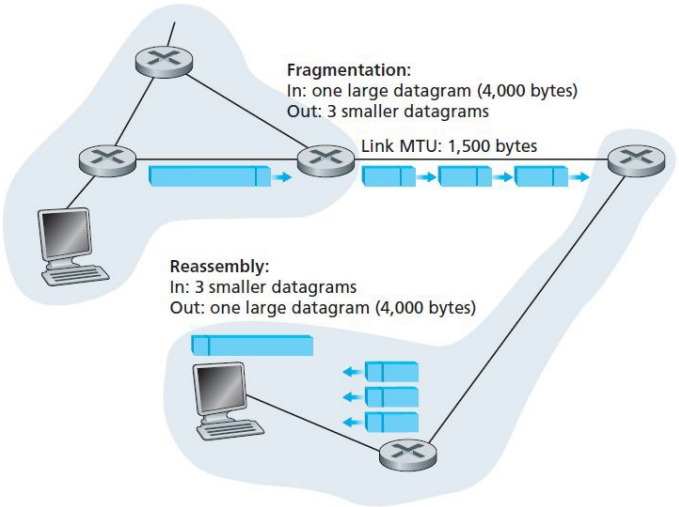
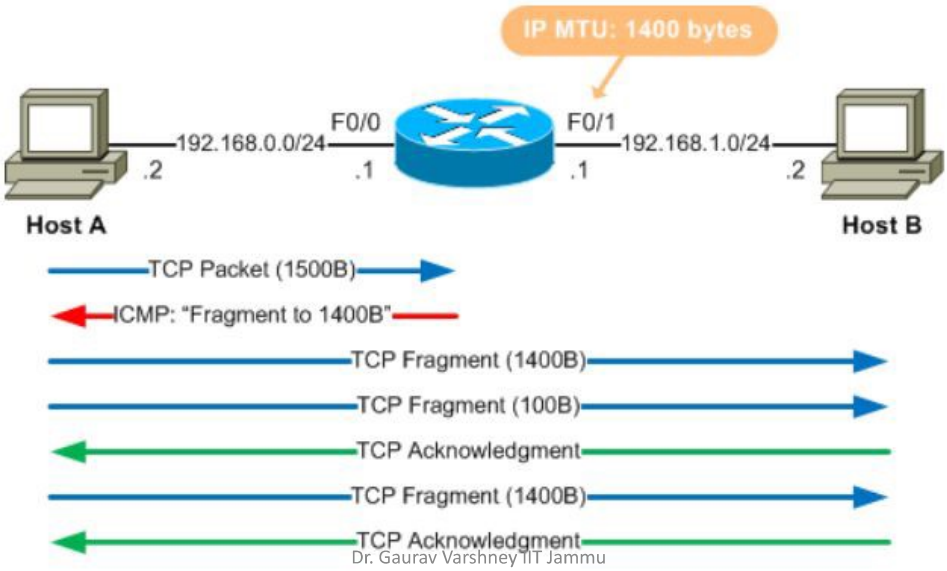


Figure 4.14 ♦ IP fragmentation and reassembly

Fragment	Bytes	ID	Offset	Flag
1st fragment	1,480 bytes in the data field of the IP datagram	identification = 777	offset = 0 (meaning the data should be inserted beginning at byte 0)	flag = 1 (meaning there is more)
2nd fragment	1,480 bytes of data	identification = 777	offset = 185 (meaning the data should be inserted beginning at byte 1,480. Note that 185 · 8 = 1,480)	flag = 1 (meaning there is more)
3rd fragment	1,020 bytes (= 3,980–1,480–1,480) of data	identification = 777	offset = 370 (meaning the data should be inserted beginning at byte 2,960. Note that 370 · 8 = 2,960)	flag = 0 (meaning this is the last fragment)

Path MTU Discovery (**PMTUD**) is a standardized technique in computer networking for determining the maximum transmission unit (MTU)

Setting the DF bit in an IP packet prevents a router from performing fragmentation when it encounters an MTU less than the packet size. Instead, the packet is discarded and an ICMP Fragmentation Needed message is sent to the originating host. Essentially, the router is indicating that it needs to fragment the packet but the DF flag won't allow for it. Conveniently, RFC 1191 expands the Fragmentation Needed message to include the MTU of the link necessitating fragmentation. A Fragmentation Needed message can be seen in packet #6 of this packet capture.



IP Addressing

- An IP address is technically associated with an interface, rather than with the host or router containing that interface.
- The boundary between the host and the physical link is called an **interface**.
- A router has multiple interfaces one for each of its links.
- Each IP address is 32 bits long (equivalently, 4 bytes), and there are thus a total of 2^{32} possible IP addresses
- These addresses are typically written in so-called **dotted-decimal notation**, in which each byte of the address is written in its decimal form and is separated by a period (dot) from other bytes in the address. Ex. 193.32.216.9

IP Addressing: Subnet

- A subnet is a subnetwork inside a network.
- Address of subnet 1 is 223.1.1.0/24
  - /24 denotes the subnet mask as 255.255.255.0 and indicates that the leftmost 24 bits of the 32 bit quantity defin the subnet address.
  - The subnet consists of 223.1.1.1, 223.1.1.2, 223.1.1.3, 223.1.1.4[router interface]
  - There are two more subnets 223.1.2.0/24 and 223.1.3.0/24

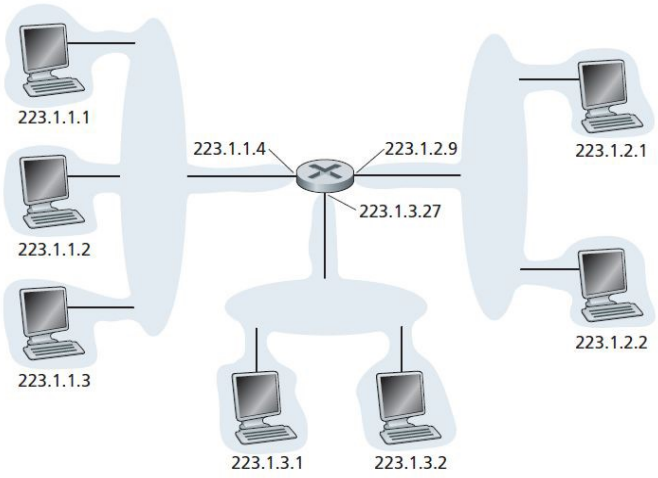


Figure 4.15 ♦ Interface addresses and subnets

The Internet's address assignment strategy is known as **Classless Interdomain Routing** (**CIDR**—pronounced *cider*) [RFC 4632]. CIDR generalizes the notion of subnet addressing. As with subnet addressing, the 32-bit IP address is divided into two parts and again has the dotted-decimal form *a.b.c.d/x*, where *x* indicates the number of bits in the first part of the address

IP Addressing: Subnet

The *x* most significant bits of an address of the form *a.b.c.d/x* constitute the network portion of the IP address, and are often referred to as the **prefix** (or *network prefix*) of the address.

An organization is typically assigned a block of contiguous addresses. A range of addresses with a common prefix.

To determine the subnets, detach each interface from its host or router, creating islands of isolated networks, with interfaces terminating the end points of the isolated networks. Each of these isolated networks is called a **subnet**

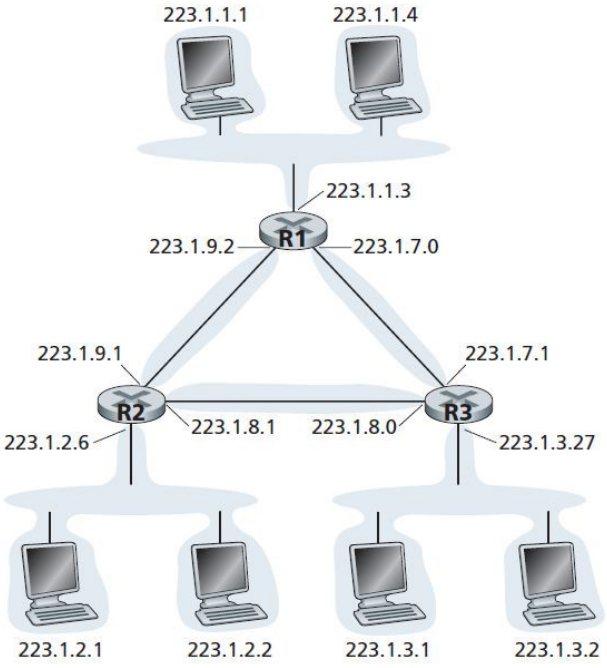


Figure 4.17 ♦ Three routers interconnecting six subnets

https://networklessons.com/subnetting/variable-length-subnet-mask-vlsm

Route Aggregation / Summarization

When a router outside the organization forwards a datagram whose destination address is inside the organization, only the leading *x* bits of the address need be considered. This considerably reduces the size of the forwarding table in these routers, since a *single* entry of the form *a.b.c.d/x* will be sufficient to forward packets to *any* destination within the organization.

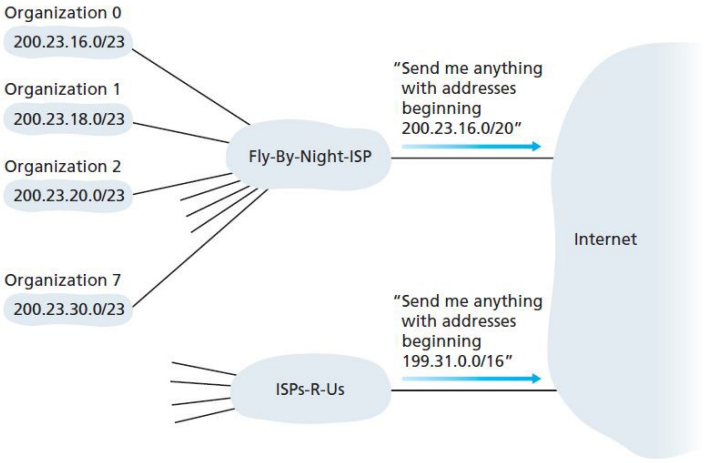


Figure 4.18 ♦ Hierarchical addressing and route aggregation

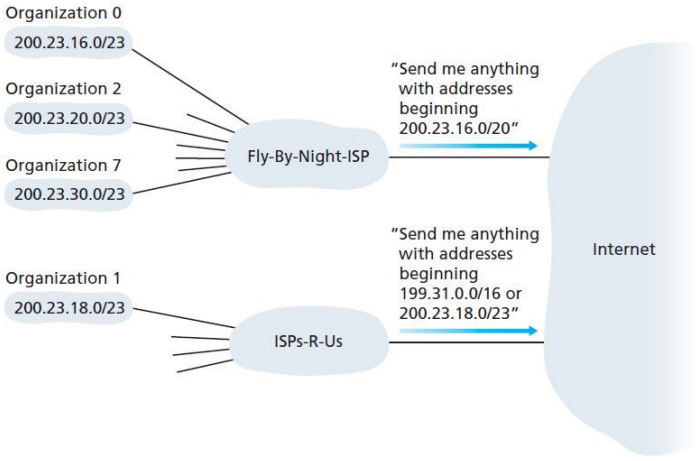
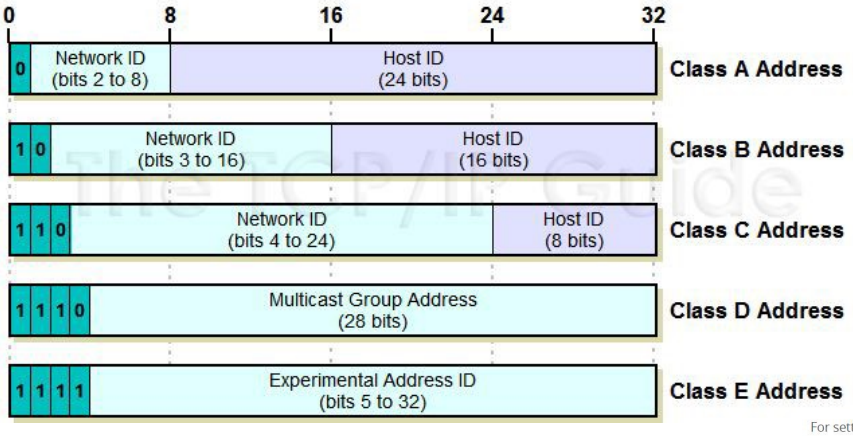


Figure 4.19 ♦ ISPs-R-Us has a more specific route to Organization 1



For setting up private networks, three IP address ranges have been reserved by IANA:

RESERVED IP ADDRESS RANGE	NETWORK CLASS
10.0.0.0 - 10.255.255.255	Class A
172.16.0.0 - 172.31.255.255	Class B
192.168.0.0 - 192.168.255.255	Class C

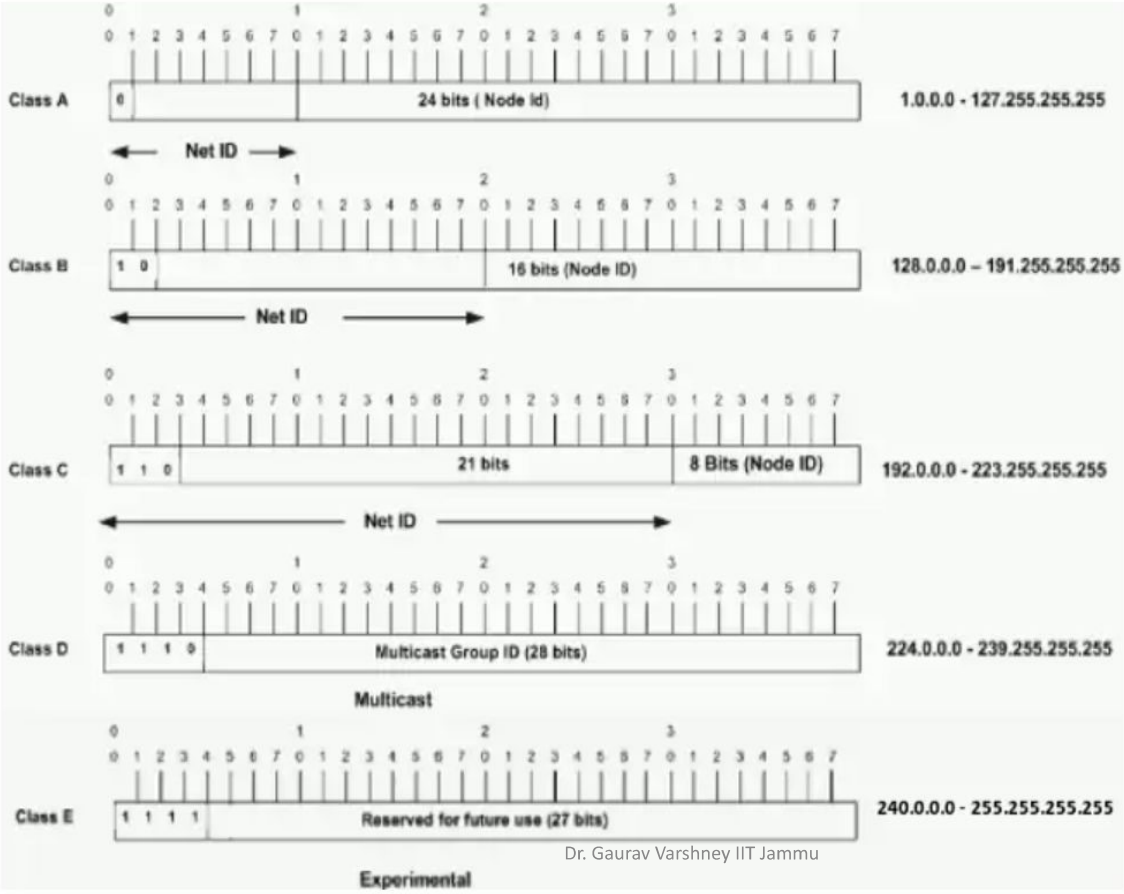
These private IP address ranges have been documented in ↗ RFC 1597 and ↗ RFC 1918.

Private IP address ranges are not routed in the Internet and can be used without registration in any number of private networks.

http://www.tcpipguide.com/free/t_IPClassfulAddressingNetworkandHostIdentificationan-3.htm

https://community.cisco.com/t5/switching/multicast-confusion/td-p/946489

https://www.meridianoutpost.com/resources/articles/IP-classes.php



IP Addressing

An IP address is an address used in order to uniquely identify a device on an IP network. The address is made up of 32 binary bits, which can be divisible into a network portion and host portion with the help of a subnet mask.

The 32 binary bits are broken into four octets (1 octet = 8 bits). Each octet is converted to decimal and separated by a period (dot)

```
  1   1   1   1 1 1 1 1
128  64  32  16 8 4 2 1 (128+64+32+16+8+4+2+1=255)
```

And this sample shows an IP address represented in both binary and decimal.

```
        10.       1.      23.      19 (decimal)
00001010.00000001.00010111.00010011 (binary)
```

Given an IP address, its class can be determined from the three high-order bits (the three leftmost bits in the first octet).

IP Addressing

https://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13788-3.html

A network mask helps you know which portion of the address identifies the network and which portion of the address identifies the node. Class A, B, and C networks have default masks, also known as natural masks,

```
Class A: 255.0.0.0
Class B: 255.255.0.0
Class C: 255.255.255.0
```

```
8.20.15.1 = 00001000.00010100.00001111.00000001
255.0.0.0 = 11111111.00000000.00000000.00000000
            -----------------------------------
             net id |      host id

netid =  00001000 = 8
hostid = 00010100.00001111.00000001 = 20.15.1
```

Subnetting allows you to create multiple logical networks that exist within a single Class A, B, or C network.

Each data link on a network must have a unique network ID, with every node on that link being a member of the same network. If you break a major network (Class A, B, or C) into smaller subnetworks, it allows you to create a network of interconnecting subnetworks. Each data link on this network would then have a unique network/subnetwork ID. Any device, or gateway, that connects *n* networks/subnetworks has *n* distinct IP addresses, one for each network / subnetwork that it interconnects.

Subnetting

https://www.comparitech.com/net-admin/variable-length-subnet-mask-vlsm-tutorial/

- Class C address 204.17.5.0 has natural subnet mask 255.255.255.0
- By using some bits from host ID you can use it to create subnets from this IP address

```
204.17.5.0 -       11001100.00010001.00000101.00000000
255.255.255.224 -  11111111.11111111.11111111.11100000
                   --------------------------|sub|----
```

```
204.17.5.0 255.255.255.224     host address range 1 to 30
204.17.5.32 255.255.255.224    host address range 33 to 62
204.17.5.64 255.255.255.224    host address range 65 to 94
204.17.5.96 255.255.255.224    host address range 97 to 126
204.17.5.128 255.255.255.224   host address range 129 to 158
204.17.5.160 255.255.255.224   host address range 161 to 190
204.17.5.192 255.255.255.224   host address range 193 to 222
204.17.5.224 255.255.255.224   host address range 225 to 254
```

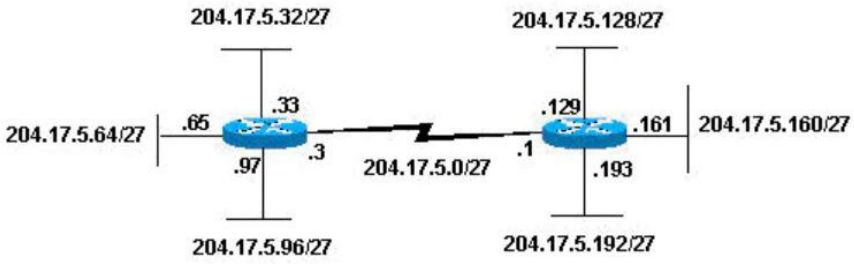
With CIDR method, one of these networks can be described with the notation prefix/length. For example, 204.17.5.32/27 denotes the network 204.17.5.32 255.255.255.224.

each subnet can have up to 32 host addresses, 30 of which can actually be assigned to a device *since host ids of all zeros or all ones are not allowed*

Subnetting

The network subnetting scheme in this section allows for eight subnets, and the network might appear as:

Figure 2



This brings up an interesting point. The more host bits you use for a subnet mask, the more subnets you have available. However, the more subnets available, the less host addresses available per subnet.

```
172.16.0.0   -   10101100.00010000.00000000.00000000
255.255.248.0 - 11111111.11111111.11111000.00000000
                ------------------| sub |-----------
```

You use five bits from the original host bits for subnets. This allows you to have 32 subnets (2^5). After using the five bits for subnetting, you are left with 11 bits for host addresses. This allows each subnet so have 2048 host addresses (2^{11}),

Subnetting

```
DeviceA: 172.16.17.30/20
DeviceB: 172.16.28.15/20
```

Determine the Subnet for DeviceA:

```
172.16.17.30   -   10101100.00010000.00010001.00011110
255.255.240.0  -   11111111.11111111.11110000.00000000
                   ------------------| sub|------------
subnet =           10101100.00010000.00010000.00000000 = 172.16.16.0
```

Looking at the address bits that have a corresponding mask bit set to one, and setting all the other address bits to zero (this is equivalent to performing a logical "AND" between the mask and address), shows you to which subnet this address belongs. In this case, DeviceA belongs to subnet 172.16.16.0.

Determine the Subnet for DeviceB:

```
172.16.28.15   -   10101100.00010000.00011100.00001111
255.255.240.0  -   11111111.11111111.11110000.00000000
                   ------------------| sub|------------
subnet =           10101100.00010000.00010000.00000000 = 172.16.16.0
```

From these determinations, DeviceA and DeviceB have addresses that are part of the same subnet.

VLSM

Variable Length Subnet Masking

No.	Segment	Host requirement	Block size that fulfills the host IP requirement	Valid hosts in block
1	LAN Segment1	29	32	30 (32 -2)
2	LAN Segment 2	21	32	30 (32 -2)
3	LAN Segment 3	12	16	14 (16-2)
4	LAN Segment 4	8	16	14 (16-2)
5	WAN Link 1	2	4	2 (4-2)
6	WAN Link 2	2	4	2 (4-2)
7	WAN Link 3	2	4	2 (4-2)
8	WAN Link 4	2	4	2 (4-2)

Subnet mask (In slash notation)	Subnet mask (In decimal notation)	Network bits	Host bits	Subnets	Block Size or Total Hosts addresses	Valid hosts addresses
/24	255.255.255.0	0	8	1	256	254
/25	255.255.255.128	1	7	2	128	126
/26	255.255.255.192	2	6	4	64	62
/27	255.255.255.224	3	5	8	32	30
/28	255.255.255.240	4	4	16	16	14
/29	255.255.255.248	5	3	32	8	6
/30	255.255.255.252	6	2	64	4	2

Segment	CIDR	Subnet Mask	Network Address	Broad cast Address	Valid host addresses
LAN Segment1	/27	255.255.255.224	192.168.1.0	192.168.1.31	192.168.1.1 to 192.168.1.30
LAN Segment 2	/27	255.255.255.224	192.168.1.32	192.168.1.63	192.168.1.33 to 192.168.1.62
LAN Segment 3	/28	255.255.255.240	192.168.1.64	192.168.1.79	192.168.1.65 to 192.168.1.78
LAN Segment 4	/28	255.255.255.240	192.168.1.80	192.168.1.95	192.168.1.81 to 192.168.1.94
WAN Link 1	/30	255.255.255.252	192.168.1.96	192.168.1.99	192.168.1.97 to 192.168.1.98
WAN Link 2	/30	255.255.255.252	192.168.1.100	192.168.1.103	192.168.1.101 to 192.168.1.102
WAN Link 3	/30	255.255.255.252	192.168.1.104	192.168.1.107	192.168.1.105 to 192.168.1.106
WAN Link 4	/30	255.255.255.252	192.168.1.108	192.168.1.111	192.168.1.107 to 192.168.1.108

Subnetting

IP addresses are managed under the authority of the Internet Corporation for Assigned Names and Numbers (ICANN) [ICANN 2012], based on guidelines set forth in [RFC 2050]

Block	Organization	IANA date	RIR date	Notes
0.0.0.0/8	IANA - Local Identification	1981-09		Originally IANA - Reserved 1981-09. 0.0.0.0/8 re identification.[1]
10.0.0.0/8	IANA - Private Use	1995-06		Reserved for Private Networks.[2] Formerly ARPA
127.0.0.0/8	IANA - Loopback	1981-09		127.0.0.0/8 is reserved for loopback.[1]
224.0.0.0/8–239.0.0.0/8	Multicast	1981-09	1991-05-22	Multicast (formerly "Class D"[4]) registered in [1]
240.0.0.0/8–255.0.0.0/8	Future Use	1981-09		Reserved for future use (formerly "Class E"[5]). 2 reserved for "limited broadcast" destination addr

List of assigned /8 blocks to commercial organisations [edit]

Block	Organization	IANA date	RIR date	Notes
12.0.0.0/8	AT&T Services	1995-06	1983-08-23	Originally AT&T Bell Laboratories, but retained by AT&T w off to Lucent Technologies in 1996. Assignment administer space)
17.0.0.0/8	Apple Inc.	1992-07	1990-04-16	Assignment administered by ARIN (Legacy space)
19.0.0.0/8	Ford Motor Company	1995-05	1988-06-15	Assignment administered by ARIN (Legacy space)

ISP's block	200.23.16.0/20	11001000 00010111 00010000 000
Organization 0	200.23.16.0/23	11001000 00010111 00010000 000
Organization 1	200.23.18.0/23	11001000 00010111 00010010 000
Organization 2	200.23.20.0/23	11001000 00010111 00010100 000
. . .	. . .	. . .
Organization 7	200.23.30.0/23	11001000 00010111 00011110 000

Routing Algorithms: Classification

- Global Routing Algorithms/Link State Routing [OSPF]
  - Each node has knowledge of costs of all links within the Autonomous System(AS).
  - Computes the least cost path between source and destination router using complete and global knowledge of the network. These are also known as link state routing algorithms.
- Distributed/Decentralized Routing Algorithms [RIP, EIGRP]
  - Each node has the knowledge of the costs of its own and directly connected neighbors.
  - Through an iterative process of calculation and exchange of information with its neighboring node a node gradually calculates the least cost path to all the destinations in the AS.
  - Distance Vector [DV] routing algorithm.
- Static routing algorithms change a path slowly and via human intervention while
- Dynamic routing algorithms change the path with network traffic loads or when topology changes.

We refer to the default router of the source host as the **source router** and the default router of the destination host as the **destination router**. The problem of routing a packet from source host to destination host clearly boils down to the problem of routing the packet from source router to destination router.

The purpose of a routing algorithm is then simple: given a set of routers, with links connecting the routers, a routing algorithm finds a "good"(least cost) path from source router to destination router.

A graph is used to formulate routing problems. Recall that a graph G =(N,E) is a set N of nodes and a collection E of edges, where each edge is between a pair of nodes from N.

In the context of network-layer routing, the nodes in the graph(undirected) represent routers—the points at which packet-forwarding decisions are made—and the edges connecting these nodes represent the physical links between these routers.

Typically, an edge's cost may reflect the physical length of the corresponding link (for example, a transoceanic link might have a higher cost than a short-haul terrestrial link), the link speed, or the monetary cost associated with a link.

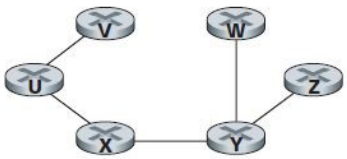
Link State Routing Algorithms

- The network topology and all link costs is available as input to an LS algorithm.
- Each node (router) broadcast link state packets [In OSPF Router LSAs (Link State Advertisements)] to all other nodes in the network which contains identities and costs of its attached links.
- Example is Open Shortest Path First or OSPF that is used in the router as an Intra AS routing protocol. In OSPF Routers perform link state broadcasts [Designated Routers multicast Network LSAs]
- The information received via multicast/broadcasts from all nodes provides the global picture of the network to a node which then runs the LS algorithm to compute least cost path to all destinations.

- **Link State Routing via Dijkstra's**

Link-State (LS) Algorithm for Source Node *u*

```
1  Initialization:
2    N' = {u}
3    for all nodes v
4      if v is a neighbor of u
5        then D(v) = c(u,v)
6      else D(v) = ∞
7
8  Loop
9    find w not in N' such that D(w) is a minimum
10   add w to N'
11   update D(v) for each neighbor v of w and not in N':
12         D(v) = min( D(v), D(w) + c(w,v) )
13   /* new cost to v is either old cost to v or known
14    least path cost to w plus cost from w to v */
15 until N'= N
```



Destination	Link
v	(u, v)
w	(u, x)
x	(u, x)
y	(u, x)
z	(u, x)

Figure 4.28 ♦ Least cost path and forwarding table for nodule u

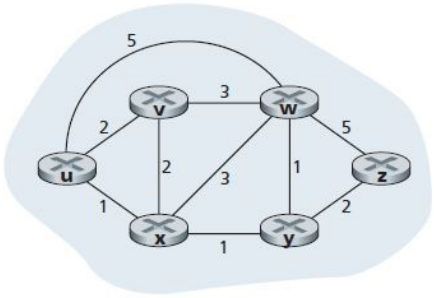


Figure 4.27 ♦ Abstract graph model of a computer network

step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

Table 4.3 ♦ Running the link-state algorithm on the network in Figure 4.27

OSPF: Open Shortest Path First is an example of LS

https://networklessons.com/ospf/ospf-lsa-types-explained

Distance Vector Routing

- The **distance vector** (**DV**) algorithm is iterative, asynchronous, and distributed.
- It is distributed as each node receives some information from its directly connected neighbors, performs its calculations and send it back to its neighbors.
- Iterative as the process continues until no more information needs to be exchanged between directly connected neighbors.
- It is asynchronous as it does not require the nodes to work in lock step manner.

Each node x maintains the following routing information:

- For each neighbor v, the cost $c(x,v)$ from x to directly attached neighbor, v
- Node x's distance vector, that is, $\boldsymbol{D}_x = [D_x(y): y \text{ in } N]$, containing x's estimate of its cost to all destinations, y, in N
- The distance vectors of each of its neighbors, that is, $\boldsymbol{D}_v = [D_v(y): y \text{ in } N]$ for each neighbor v of x

Bellman Ford Equation –

$$d_x(y) = \min_v\{c(x,v) + d_v(y)\},$$

Where $d_x(y)$ is the least cost path from node x to node y.

RIP(Routing Information Protocol) is a DV protocol

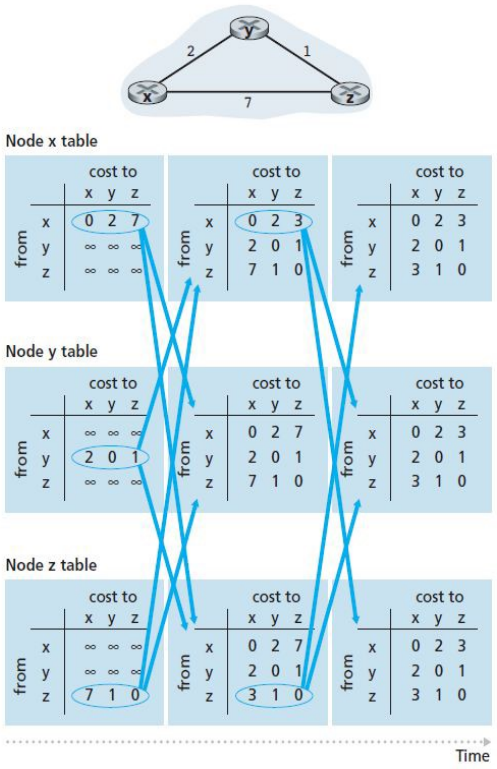


Figure 4.30 ♦ Distance-vector (DV) algorithm

Distance-Vector (DV) Algorithm

At each node, x:

```
1  Initialization:
2     for all destinations y in N:
3        Dx(y) = c(x,y)   /* if y is not a neighbor then c(x,y) = ∞ */
4     for each neighbor w
5        Dw(y) = ? for all destinations y in N
6     for each neighbor w
7        send distance vector Dx = [Dx(y): y in N] to w
8
9  loop
10    wait (until I see a link cost change to some neighbor w or
11          until I receive a distance vector from some neighbor w)
12
13    for each y in N:
14       Dx(y) = minv{c(x,v) + Dv(y)}
15
16    if Dx(y) changed for any destination y
17       send distance vector Dx = [Dx(y): y in N] to all neighbors
18
19 forever
```

$$D_x(x) = 0$$

$$D_x(y) = \min\{c(x,y) + D_y(y),\ c(x,z) + D_z(y)\} = \min\{2 + 0,\ 7 + 1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z),\ c(x,z) + D_z(z)\} = \min\{2 + 1,\ 7 + 0\} = 3$$

Routing Loop, Count to Infinity and Poisoned Reverse

- The decreased cost between x and y propagate smoother (Fig 4.31 (a)) than an increased cost between x and y 4.31 (b).

1. Before the link cost changes, $D_y(x) = 4$, $D_y(z) = 1$, $D_z(y) = 1$, and $D_z(x) = 5$. At time t_0, y detects the link-cost change (the cost has changed from 4 to 60). y computes its new minimum-cost path to x to have a cost of

$$D_y(x) = \min\{c(y,x) + D_x(x),\ c(y,z) + D_z(x)\} = \min\{60 + 0,\ 1 + 5\} = 6$$

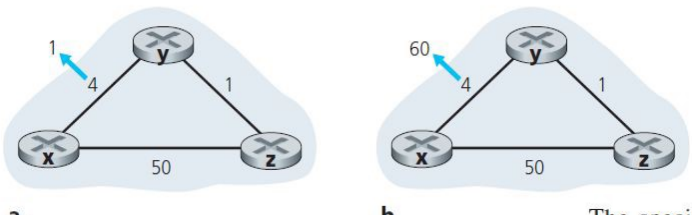


Figure 4.31 ♦ Changes in link cost

The specific looping scenario just described can be avoided using a technique known as *poisoned reverse*. The idea is simple—if z routes through y to get to destination x, then z will advertise to y that its distance to x is infinity, that is, z will advertise to y that $D_z(x) = \infty$ (even though z knows $D_z(x) = 5$ in truth). z will continue telling this little white lie to y as long as it routes to x via y. Since y believes that z has no path to x, y will never attempt to route to x via z, as long as z continues to route to x via y (and lies about doing so).

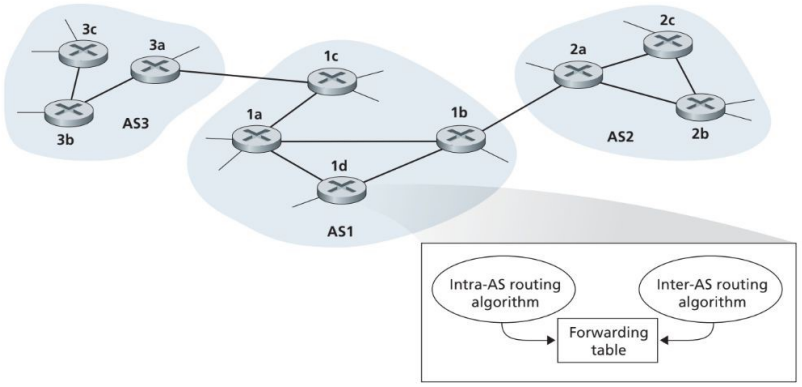
Comparison LS and DV

LS has inherent message complexity. O(|N||E|) message need to be sent.

DV has low speed of convergence as it may end up stabilizing slowly and may cause routing loops.

DV is less robust as a sabotaged update by a router may corrupt the routing tables of many routers. In LS route computation happens at every node increasing the degree of robustness.

Interconnected Autonomous System



ASN	NAME	NUM IPS
AS9829	National Internet Backbone	5,418,752
AS55836	Reliance Jio Infocomm Limited	4,174,848
AS45609	Bharti Airtel Ltd. AS for GPRS Service	3,922,432
AS24560	Bharti Airtel Ltd., Telemedia Services	3,426,304
AS9498	BHARTI Airtel Ltd.	2,016,000
AS4755	TATA Communications formerly VSNL is Leading ISP	1,802,240
AS17813	Mahanagar Telephone Nigam Limited	1,390,592

Figure 4.32 ♦ An example of interconnected autonomous systems

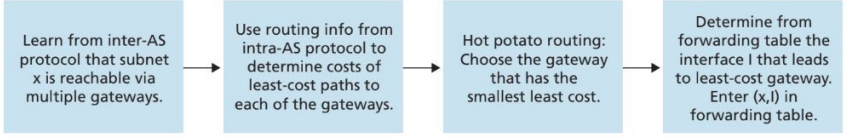
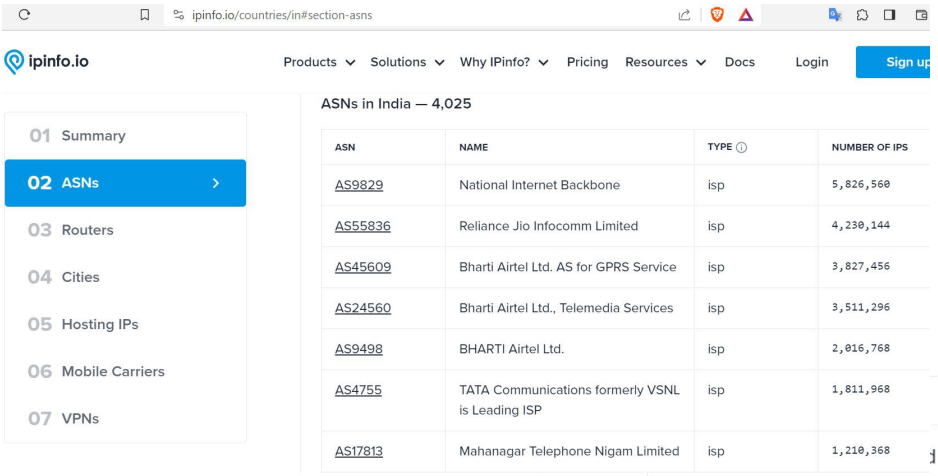


Figure 4.33 ♦ Steps in adding an outside-AS destination in a router's forwarding table



		NUM OF IPS
		1,024
104.112.227.0/24	Akamai Technologies, Inc.	256
104.118.7.0/24	Akamai Technologies, Inc.	256
104.95.190.0/24	Akamai Technologies, Inc.	256
104.95.97.0/24	Akamai Technologies, Inc.	256
117.192.0.0/16	Broadband Multiplay Project, O/o DGM BB, NOC BSNL Bangalore	65,536
117.192.0.0/20	BSNL BBoWIFI services in South by M/s Tikona	4,096
117.192.112.0/20	Broadband Multiplay Project, O/o DGM BB, NOC BSNL Bangalore	4,096