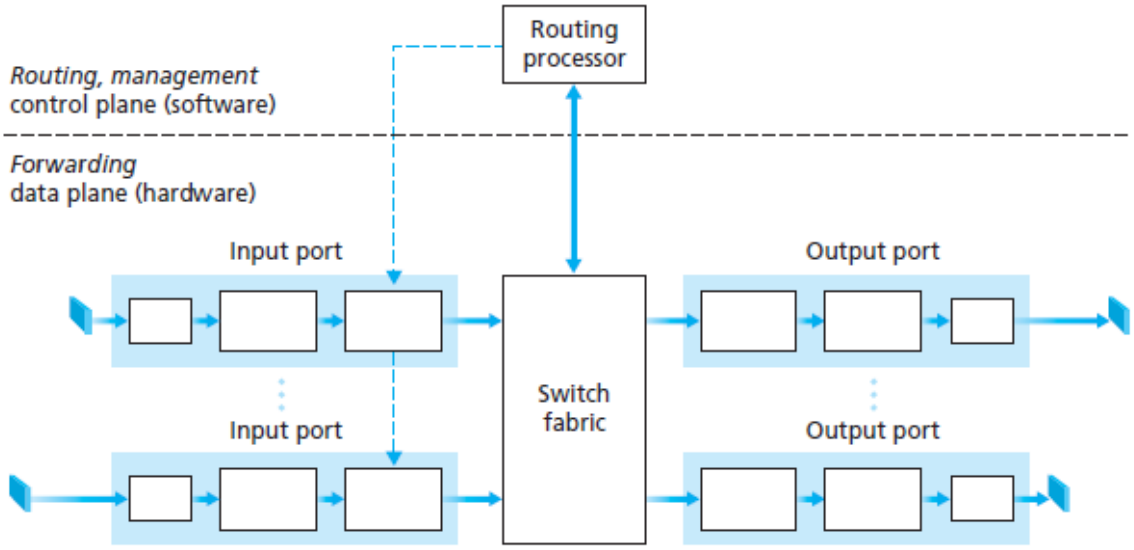
Module – 3

NETWORK LAYER

Structure of a Router

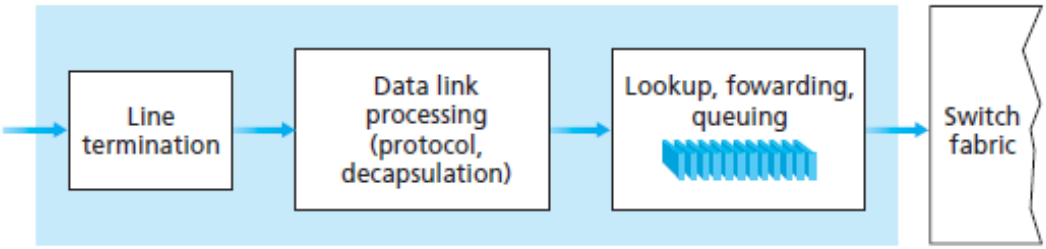
A high-level view of a generic router architecture is shown below. Four router components can be identified:



- **Input ports:** An input port performs several key functions.
  It performs the physical layer function of terminating an incoming physical link at a router.
  An input port also performs link-layer functions needed to interoperate with the link layer at the other side of the incoming link.
  The lookup function is also performed at the input port.
  Forwarding table is consulted to determine the router output port to which an arriving packet will be forwarded via the switching fabric.
- **Switching fabric:** The switching fabric connects the router's input ports to its output ports.
- **Output port:**. An output port stores packets received from the switching fabricand transmits these packets on the outgoing link by performing the necessary link-layer and physical-layer functions.
- **Routing processor:** The routing processor executes the routing protocols maintains routing tables and attached link state information, and computes the forwarding table for the router. It also performs the network management functions.

- 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**.
- **Router control plane** functions are usually implemented in software and execute on the routing processor

Input Processing

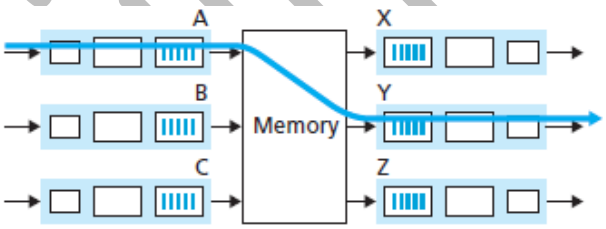


- The input port's line termination function and link-layer processing implement the physical and link layers for that individual input link.
- The lookup performed in the input port is central to the router's operation—it is here that the router uses the forwarding table to look up the output port to which an arriving packet will be forwarded via the switching fabric.
- The forwarding table is computed and updated by the routing processor, with a shadow copy typically stored at each input port. The forwarding table is copied from the routing processor to the line cards over a separate bus.
- Once a packet's output port has been determined via the lookup, the packet can be sent into the switching fabric. In some designs, a packet may be temporarily blocked from entering the switching fabric if packets from other input ports are currently using the fabric. A blocked packet will be queued at the input port and then scheduled to cross the fabric at a later point in time.

Switching

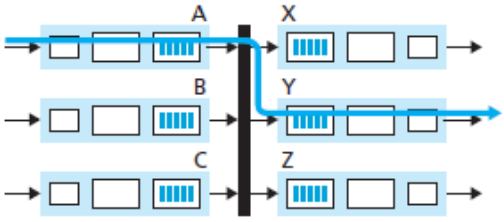
The switching fabric switches the packet from an input port to an output port. Switching can be accomplished in a number of ways:

1. **Switching via memory:**
   - The simplest, earliest routers were traditional computers, with switching between input and output ports being done under direct control of the CPU (routing processor).
   - Input and output ports functioned as traditional I/O devices in a traditional operating system.
   - An input port with an arriving packet first signaled the routing processor via an interrupt.
   - The packet was then copied from the input port into processor memory.
   - The routing processor then extracted the destination address from the header, looked up the appropriate output port in the forwarding table, and copied the packet to the output port's buffers.
   - Here two packets cannot be forwarded at the same time, even if they have different destination ports, since only one memory read/write over the shared system bus can be done at a time.
   - Many modern routers switch via memory. A major difference from early routers is that the lookup of the destination address and the storing of the packet into the appropriate memory location are performed by processing on the input line cards.



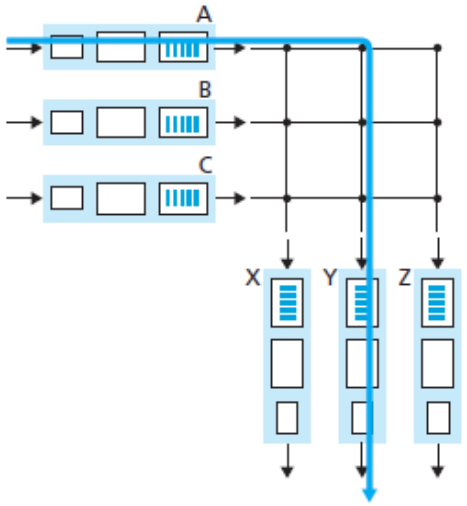
2. **Switching via a bus:**
   - In this approach, an input port transfers a packet directly to the output port over a shared bus, without intervention by the routing processor.
   - This is typically done by having the input port pre-pend a switch-internal label (header) to the packet indicating the 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.
   - The label is then removed at the output port, as this label is only used within the switch to cross the bus.

- 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.



3. Switching via an interconnection network:

- One way to overcome the bandwidth limitation of a single, shared bus is to use a more sophisticated interconnection network, such as those that have been used in the past to interconnect processors in multiprocessor computer architecture.
- A crossbar switch is an interconnection network consisting of 2N buses that connect N input ports to N output ports.
- Each vertical bus intersects each horizontal bus at a crosspoint, which can be opened or closed at any time by the switch fabric.

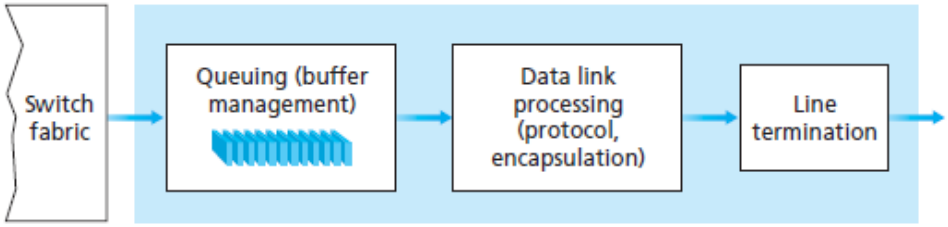


When a packet arrives from port A and needs to be forwarded to port Y, the switch controller closes the crosspoint at the intersection of busses A and Y, and port A then sends the packet onto its bus, which is picked up (only) by bus Y. Note that a packet from port B can be forwarded to port X at the same time, since the A-to-Y and B-to-X packets use different input and output busses. Thus, unlike the previous two switching approaches, crossbar networks are capable of forwarding multiple packets in parallel. However, if two packets

from two different input ports are destined to the same output port, then one will have to wait at the input, since only one packet can be sent over any given bus at a time.

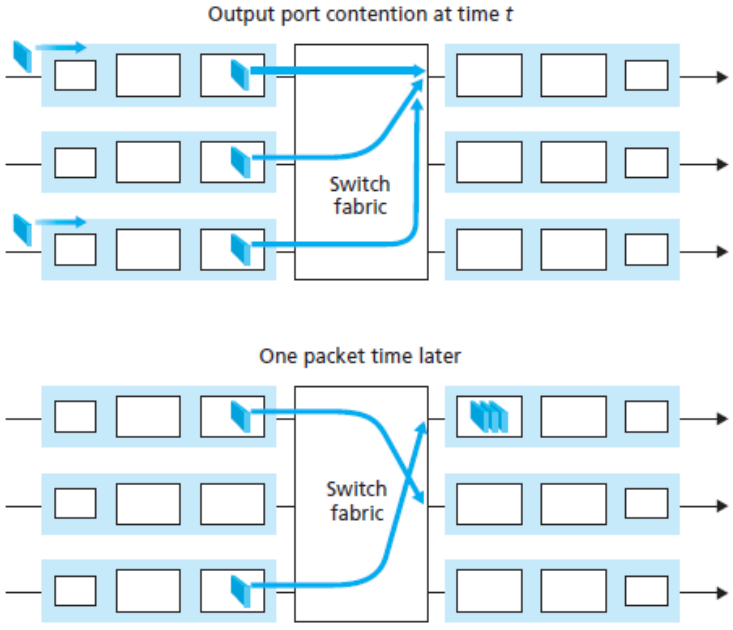
Output Processing

Output port processing takes packets that have been stored in the output port's memory and transmits them over the output link. This includes selecting and de-queuing packets for transmission, and performing the needed link layer and physical-layer transmission functions.



Queuing

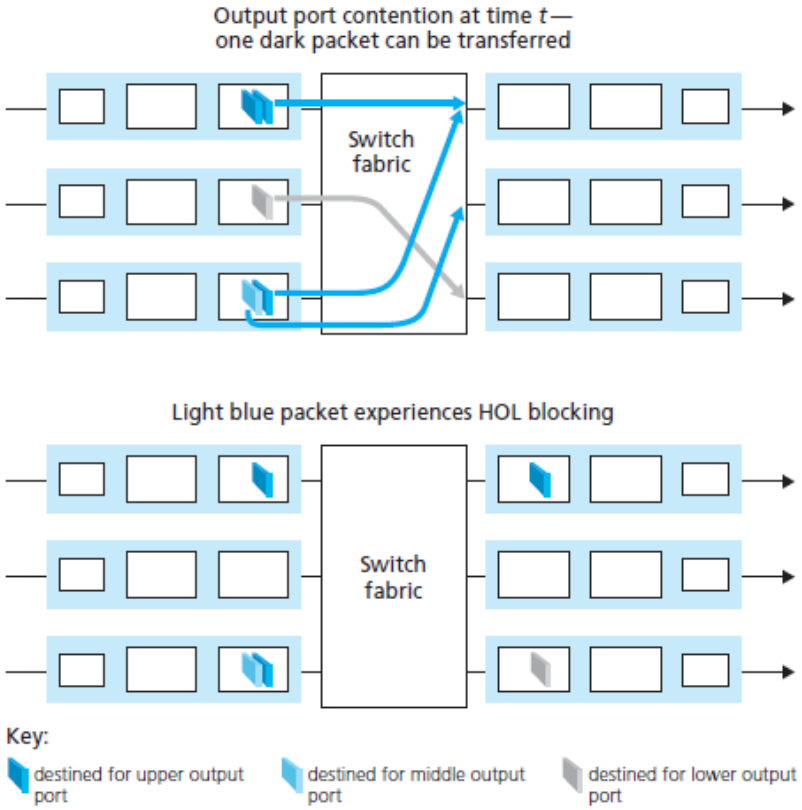
- Packet queues may form at both the input ports and the output ports.
- The location and extent of queuing will depend on the traffic load, the relative speed of the switching fabric, and the line speed.
- When many packets arrive from same source at a faster rate than switching rate queuing at input port occurs.
- When many packets are destined towards same output port queuing at output port occurs.



- A consequence of output port queuing is that a **packet scheduler** at the output port must choose one packet among those queued for transmission.
- Many packet scheduling algorithms like first-come-first-served (FCFS) scheduling, priority queuing, fair queuing or a more sophisticated scheduling discipline such as weighted fair queuing (WFQ) is available.
- Packet scheduling plays a crucial role in providing quality-of-service guarantees.
- Similarly, if there is not enough memory to buffer an incoming packet, a decision must be made to either drop the arriving packet (a policy known as **drop-tail**) or remove one or more already-queued packets to make room for the newly arrived packet.
- In some cases, it may be advantageous to drop (or mark the header of) a packet before the buffer is full in order to provide a congestion signal to the sender.
- A number of packet-dropping and -marking policies (which collectively have become known as **active queue management (AQM)** algorithms) have been proposed and analyzed.
- One of the most widely studied and implemented AQM algorithms is the **Random Early Detection (RED)** algorithm. Under RED, a weighted average is maintained for the length of the output queue.
- If the average queue length is less than a minimum threshold, min_{th}, when a packet arrives, the packet is admitted to the queue.
- Conversely, if the queue is full or the average queue length is greater than a maximum threshold, max_{th}, when a packet arrives, the packet is marked or dropped.
- Finally, if the packet arrives to find an average queue length in the interval [min_{th}, max_{th}], the packet is marked or dropped with a probability that is typically some function of the average queue length, min_{th}, and max_{th}.

Consider the following scenario:

Suppose that in the below figure the switch fabric chooses to transfer the packet from the front of the upper-left queue. In this case, the darkly shaded packet in the lower-left queue must wait. But not only must this darkly shaded packet wait, so too must the lightly shaded packet that is queued behind that packet in the lower-left queue, even though there is no contention for the middle-right output port (the destination for the lightly shaded packet). This phenomenon is known as **head-of-the-line (HOL)** blocking.

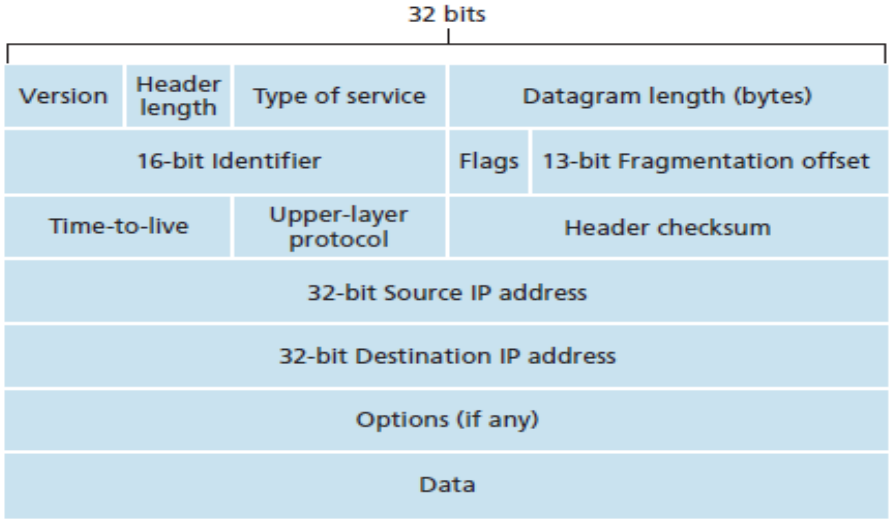


The Internet Protocol (IP)

Internet addressing and forwarding are important components of the Internet Protocol (IP).

There are two versions of IP in use today: IPv4, IPv6.

Datagram Format



- **Version number:** These 4 bits specify the IP protocol version of the datagram.
- **Header length:** Because an IPv4 datagram can contain a variable number of options these 4 bits specify the total header bytes.
- **Type of service:** The type of service (TOS) bits were included in the IPv4 header to allow different types of IP datagrams (for example, datagrams particularly requiring low delay, high throughput, or reliability) to be distinguished from each other.
- **Datagram length:** This is the total length of the IP datagram (header plus data), measured in bytes.
- **Identification:** this field represents identification number assigned to related fragments.
- **Flag:** there are three flags, first bit is unused, second bit is do not fragment bit, If this bit is set intermediate nodes should not perform fragmentation. Last bit is more fragment bit. More fragment bit represents more fragments to follow after this.
- **Fragmentation offset:** Starting byte of a fragment (In multiples of 8 bytes).
- **Time-to-live:** The time-to-live (TTL) field is included to ensure that datagrams do not circulate forever in the network. It represents hop limit.
- **Protocol:** Represents the upper layer protocol, 6 for TCP, 17 for UDP.
- **Header checksum:** field is used for error detection.
- **Source and destination IP addresses:** represents 32 bit source and destination IP address.
- **Options:** The options fields allow an IP header to be extended. It allows to include additional functionalities.
- **Data (payload):** represents the data that has to be transmitted.

IP Datagram Fragmentation

- The maximum amount of data that a link-layer frame can carry is called the maximum transmission unit (MTU). Because each IP datagram is encapsulated within the link-layer frame for transport from one router to the next router, the MTU of the link-layer protocol places a hard limit on the length of an IP datagram.
- IP datagram is divided into smaller packets according to MTU. These smaller packets are called fragments and process is called fragmentation.
- Fragments need to be reassembled before they reach the transport layer at the destination.

- Receiver needs to reassemble all the fragments belong to same original IP datagram. In order to identify all the related fragments **Identification** field is used.
- There are three flags, first bit is unused, second bit is do not fragment bit, If this bit is set intermediate nodes should not perform fragmentation. Last bit is more fragment bit. More fragment bit represents more fragments to follow after this.
- In order for the destination host to determine whether a fragment is missing the offset field is used to specify where the fragment fits within the original IP datagram.

Example:

A datagram of 4,000 bytes (20 bytes of IP header plus 3,980 bytes of IP payload) arrives at a router and must be forwarded to a link with an MTU of 1,500 bytes. This implies that the 3,980 data bytes in the original datagram must be allocated to three separate fragments. Suppose that the original datagram is stamped with an identification number of 777. Following table shows the fragmentation.

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)

At the destination, the payload of the datagram is passed to the transport layer only after the IP layer has fully reconstructed the original IP datagram. If one or more of the fragments does not arrive at the destination, the incomplete datagram is discarded and not passed to the transport layer.

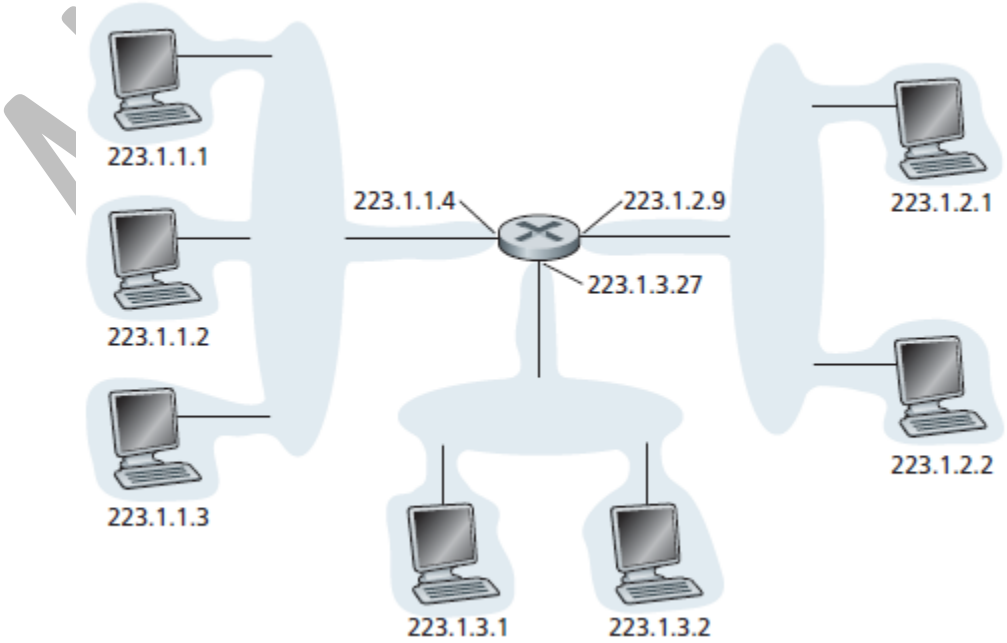
IPv4 Addressing

- Each IP address is 32 bits long (equivalently, 4 bytes), and there are thus a total of 232 possible IP addresses. Approximately there are about 4 billion possible IP addresses.

- IP 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
- The address 193.32.216.9 in binary notation is 11000001 00100000 11011000 00001001
- Each interface on every host and router in the global Internet must have an IP address that is globally unique.
- IP address has 2 parts: Network ID and Host ID. Network ID is used to identify the network and Host ID is used to identify host in the network.
- A network can be divided into smaller sub networks called subnets.
- Initially IP address was divided into 5 classes. We call this as classful addressing. It leads to shortage of IP Address. Now Classless Inter Domain Routing (CIDR) addressing is used. In CIDR IP address is represented as a.b.c.d/x where x represents number of bits used for Network ID.
- Example.

  If 256 hosts are there last 8 bit is allocated to host ID remaining 24 bit is allocated to network ID. Here /x value is /24.

Below figure shows sub netting.



Dynamic Host Configuration Protocol

- Once an organization has obtained a block of addresses, it can assign individual IP addresses to the host and router interfaces in its organization.
- A system administrator will typically manually configure the IP addresses into the.
- Host addresses can also be configured manually, but more often this task is now done using the Dynamic Host Configuration Protocol (DHCP).
- DHCP allows a host to obtain (be allocated) an IP address automatically.
- DHCP works over UDP with port number 67.

DHCP involves four steps:

1) DHCP server discovery

Host broadcast DHCP discovery message with source address 0.0.0.0 and destination address 255.255.255.255

2) DHCP server offer(s)

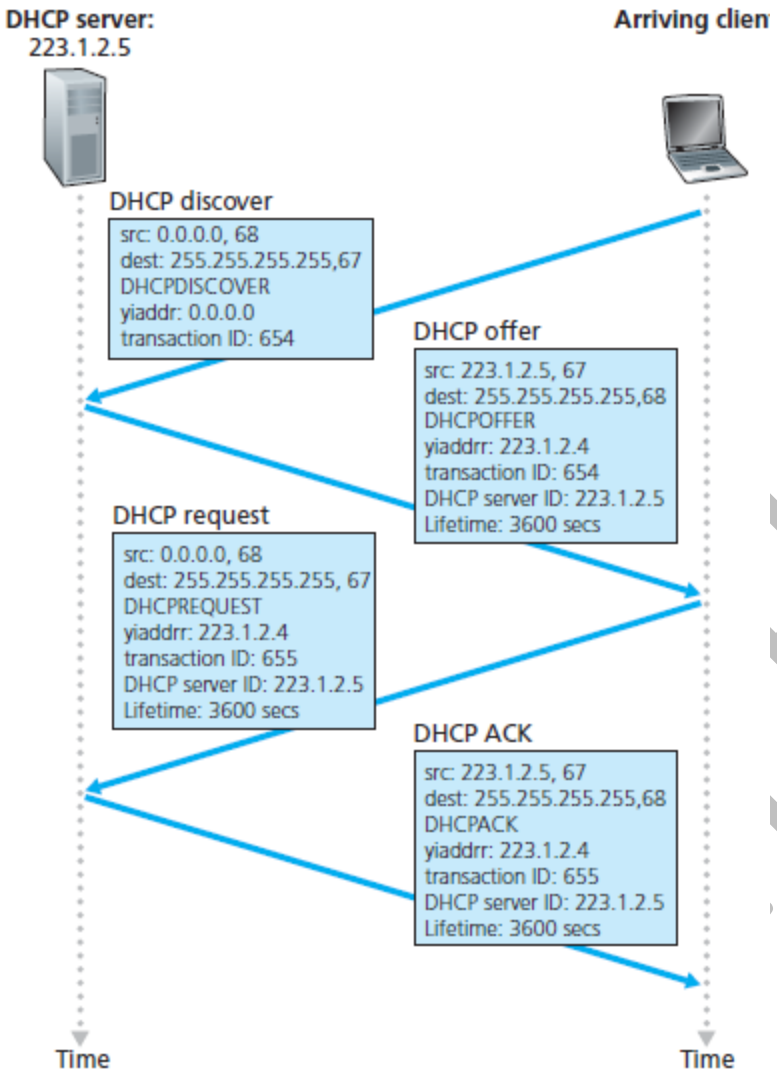
A DHCP server receiving a DHCP discover message responds to the client with a DHCP offer message that is broadcast to all nodes on the subnet, again using the IP broadcast address of 255.255.255.255.

3) DHCP request

The newly arriving client will choose from among one or more server offers and respond to its selected offer with a DHCP request message.

4) DHCP ACK

The server responds to the DHCP request message with a DHCP ACK message, confirming the requested parameters.



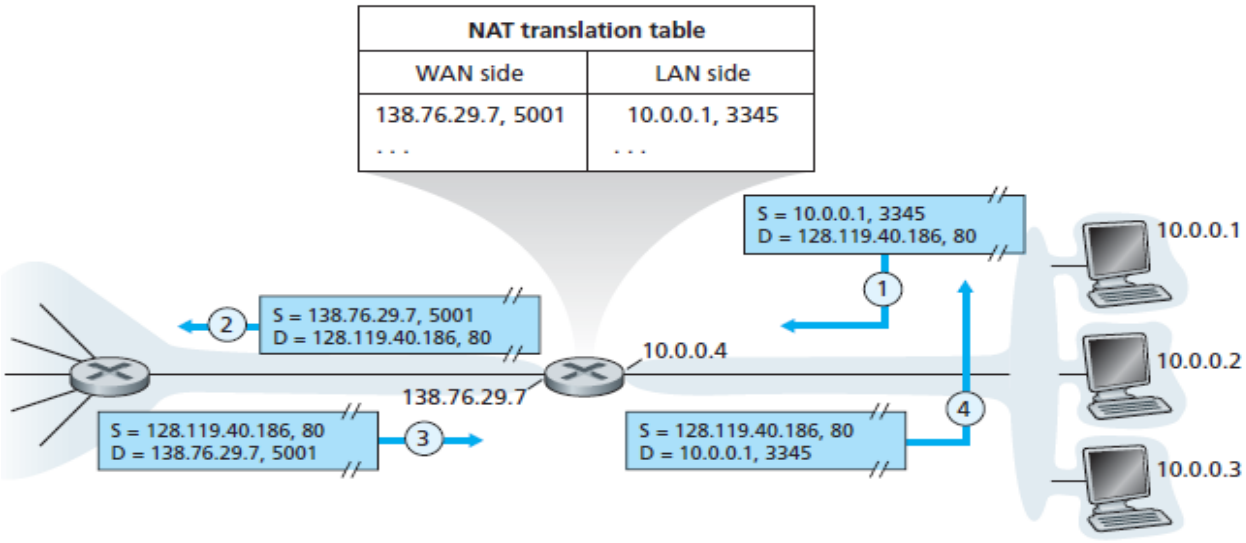
Network Address Translation (NAT)

- Private IP addresses shown below are used inside company/campus/organization or home network.

  10.0.0.0 to 10.255.255.255

  171.16.0.0 to 171.31.255.255

  192.168.0.0 to 192.168.255.255
- But company/campus/organization or home network connect to internet through global IP address.
- To convert from private IP address to global IP address and vice-versa NAT is used.
- NAT can be illustrated with the following diagram.



- Here the host with IP 10.0.0.1 sends the IP datagram with source address, port number 10.0.0.1, 3345 and destination address, port number 128.119.40.186, 80.
- NAT router maintains a NAT table as shown above
- NAT router make an entry in its table and replaces the source IP address with global IP address 138.76.29.7 and port number 5001 and send it to destination.
- When the response comes from destination, the global IP address will be replaced by private IP address according to entry available in NAT table. Then the message is delivered to appropriate host.

UPnP

- NAT traversal is increasingly provided by Universal Plug and Play (UPnP), which is a protocol that allows a host to discover and configure a nearby NAT.
- UPnP requires that both the host and the NAT be UPnP compatible.
- With UPnP, an application running in a host can request a NAT mapping between its (private IP address, private port number) and the (public IP address, public port number) for some requested public port number.
- If the NAT accepts the request and creates the mapping, then nodes from the outside can initiate TCP connections to (public IP address, public port number).
- Furthermore, UPnP lets the application know the value of (public IP address, public port number), so that the application can advertise it to the outside world.

Internet Control Message Protocol (ICMP)

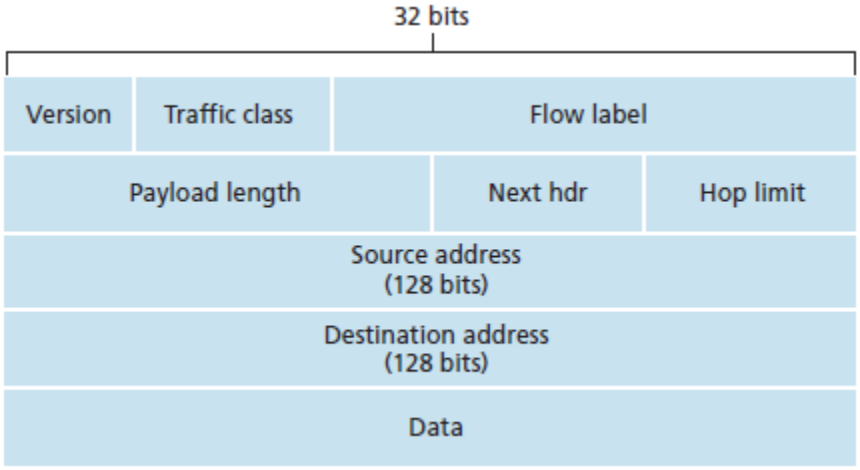
- ICMP is used by hosts and routers to communicate network- layer information to each other. The most typical use of ICMP is for error reporting.
- ICMP is part of IP but architecturally it lies just above IP hence ICMP messages are carried inside IP datagrams. That is, ICMP messages are carried as IP payload, just as TCP or UDP segments are carried as IP payload.
- ICMP messages have a type and a code field, and contain the header and the first 8 bytes of the IP datagram that caused the ICMP message to be generated in the first place.
- Popular ICMP messages are listed below

ICMP Type	Code	Description
0	0	echo reply (to ping)
3	0	destination network unreachable
3	1	destination host unreachable
3	2	destination protocol unreachable
3	3	destination port unreachable
3	6	destination network unknown
3	7	destination host unknown
4	0	source quench (congestion control)
8	0	echo request
9	0	router advertisement
10	0	router discovery
11	0	TTL expired
12	0	IP header bad

- The well-known ping program sends an ICMP type 8 code 0 message to the specified host. The destination host, seeing the echo request, sends back a type 0 code 0 ICMP echo reply.
- Congested router send an ICMP source quench message to a host toforce that host to reduce its transmission rate.

IPv6

IPv6 Datagram Format



The most important changes introduced in IPv6 are:

- **Expanded addressing capabilities:** IPv6 increases the size of the IP address from 32 to 128 bits.
- **A streamlined 40-byte header:** 40 bytes of mandatory header is used in IPv6 whereas IPv4 uses 20 bytes of mandatory header.
- **Flow labeling and priority:** Flow label refers to labeling of packets belonging to particular flows for which the sender requests special handling, such as a non default quality of service or real-time service. The IPv6 header also has an 8-bit traffic class field. This field, like the TOS field in IPv4, can be used to give priority to certain datagrams within a flow.

The following fields are defined in IPv6:

- **Version:** This 4-bit field identifies the IP version number.
- Traffic class: This 8-bit field specify priority.
- **Flow label:** this 20-bit field is used to identify a flow of datagrams.
- **Payload length:** This 16-bit value is treated as an unsigned integer giving the number of bytes in the IPv6 datagram following the fixed-length, 40-byte datagram header.
- **Next header:** This field identifies the next following header.
- **Hop limit:** The contents of this field are decremented by one by each router that forwards the datagram. If the hop limit count reaches zero, the datagram is discarded.
- **Source and destination addresses:** 128 bit IPv6 address.

- **Data:** This is the payload portion of the IPv6 datagram.

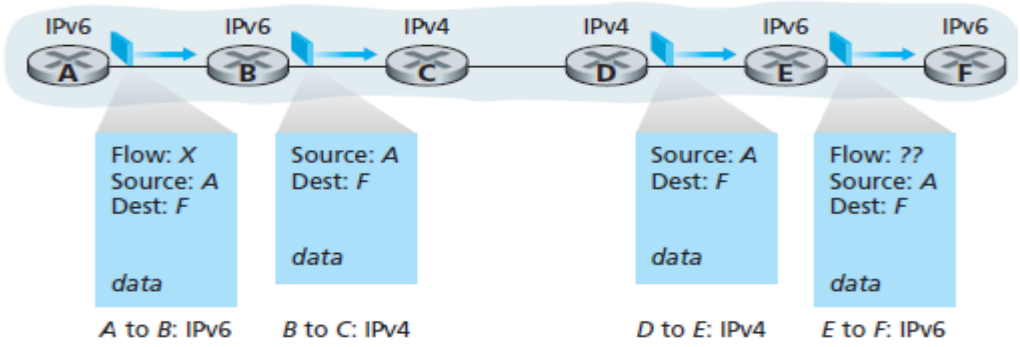
Following fields appearing in the IPv4 datagram are no longer present in the IPv6 datagram:

- **Fragmentation/Reassembly:** IPv6 does not allow for fragmentation and reassembly at intermediate routers; these operations can be performed only by the source and destination. If an IPv6 datagram received by a router is too large to be forwarded over the outgoing link, the router simply drops the datagram and sends a "Packet Too Big" ICMP error message (see below) back to the sender. The sender can then resend the data, using a smaller IP datagram size.
- **Header checksum:** Because the transport-layer and link-layer protocols in the Internet layers perform check-summing, the designers of IP probably felt that this functionality was sufficiently redundant in the network layer that it could be removed.
- **Options:** An options field is no longer a part of the standard IP header. Instead of option field extension headers are used.

Transitioning from IPv4 to IPv6

1) Dual-stack:

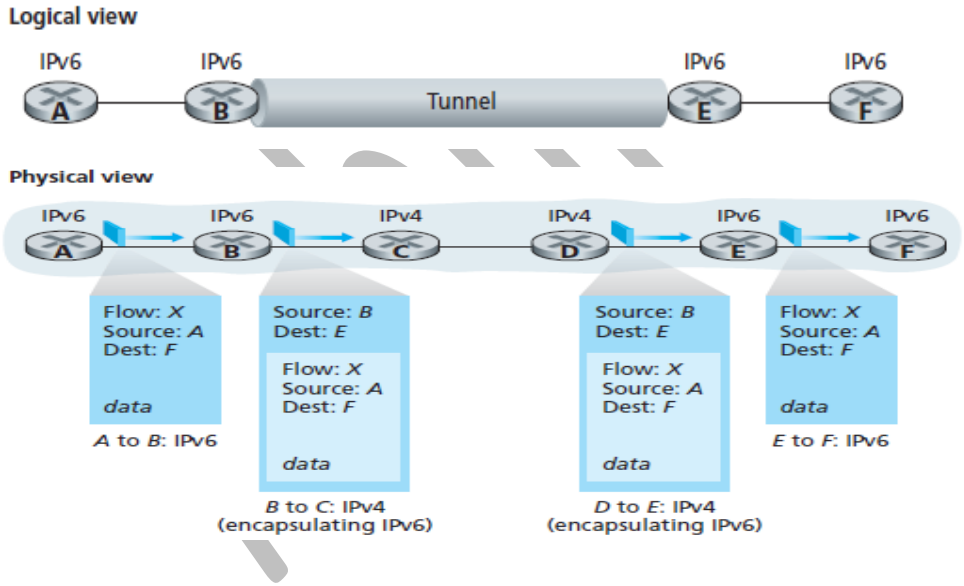
- Here IPv6 nodes also have a complete IPv4 implementation. Such a node, has the ability to send and receive both IPv4 and IPv6 datagrams. When interoperating with an IPv4 node, an IPv6/IPv4 node can use IPv4 datagrams; when interoperating with an IPv6 node, it can speak IPv6. IPv6/IPv4 nodes must have both IPv6 and IPv4 addresses. They must furthermore be able to determine whether another node is IPv6-capable or IPv4-only.
- In the dual-stack approach, if either the sender or the receiver is only IPv4- capable, an IPv4 datagram must be used. As a result, it is possible that two IPv6- capable nodes can end up, in essence, sending IPv4 datagrams to each other.



Example: Suppose Node A is IPv6-capable and wants to send an IP datagram to Node F, which is also IPv6-capable. Nodes A and B can exchange an IPv6 datagram. However, Node B must create an IPv4 datagram to send to C. Certainly, the data field of the IPv6 datagram can be copied into the data field of the IPv4 datagram and appropriate address mapping can be done. However, in performing the conversion from IPv6 to IPv4, there will be IPv6-specific fields in the IPv6 datagram that have no counterpart in IPv4. The information in these fields will be lost. Thus, even though E and F can exchange IPv6 datagrams, the arriving IPv4 datagrams at E from D do not contain all of the fields that were in the original IPv6 datagram sent from A.

2) Tunneling

Tunneling can solve the problem noted above. The basic idea behind tunneling is the following. Suppose two IPv6 nodes want to interoperate using IPv6 datagrams but are connected to each other by intervening IPv4 routers. We refer to the intervening set of IPv4 routers between two IPv6 routers as a tunnel. With tunneling, the IPv6 node on the sending side of the tunnel takes the entire IPv6 datagram and puts it in the data (payload) field of an IPv4 datagram.



IP Security

IPsec is the security protocol used for IP security. The services provided by an IPsec session include:

- **Cryptographic agreement:** Mechanisms that allow the two communicating hosts to agree on cryptographic algorithms and keys.

- **Encryption of IP datagram payloads:** When the sending host receives a segment from the transport layer, IPsec encrypts the payload. The payload can only bedecrypted by IPsec in the receiving host.
- **Data integrity:** IPsec allows the receiving host to verify that the datagram's header fields and encrypted payload were not modified while the datagram was en route from source to destination.
- **Origin authentication:** When a host receives an IPsec datagram from a trusted source, the host is assured that the source IP address in the datagram is the actual source of the datagram.