UNIT-III

Circuit switching vs. packet switching:

One fundamental way of differentiating networking technologies is on the basis of the method they use to determine the path between devices over which information will flow. In highly simplified terms, there are two approaches: either a path can be set up between the devices in advance, or the data can be sent as individual data elements over a variable path.

Circuit Switching In this networking method, a connection called a *circuit* is set up between two devices, which is used for the whole communication. Information about the nature of the circuit is maintained by the network . The circuit may either be a fixed one that is always present, or it may be a circuit that is created on an as-needed basis. Even if many potential paths through intermediate devices may exist between the two devices communicating, only one will be used for any given dialog. This is illustrated in Figure1

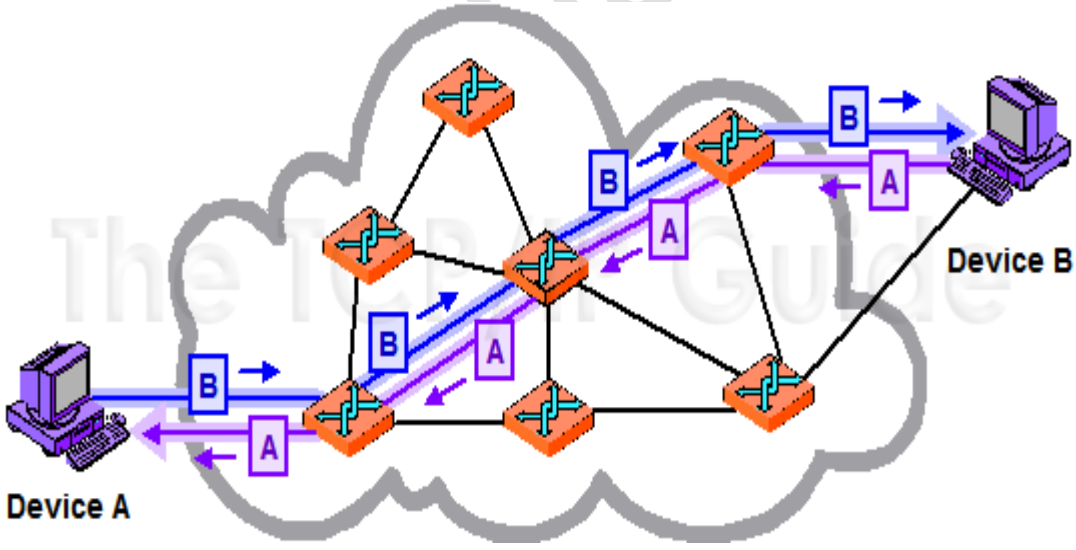


Figure1:Circuit Switching

In a circuit-switched network, before communication can occur between two devices, a circuit is established between them. This is shown as a thick blue line for the conduit of data from Device *A* to Device *B*, and a matching purple line from *B*

back to *A*. Once set up, all communication between these devices takes place over this circuit, even though there are other possible ways that data could conceivably be passed over the network of devices between them. Contrast this diagram to Figure 2

The classic example of a circuit-switched network is the telephone system. When you call someone and they answer, you establish a circuit connection and can pass data between you, in a steady stream if desired. That circuit functions the same way regardless of how many intermediate devices are used to carry your voice. You use it for as long as you need it, and then terminate the circuit. The next time you call, you get a new circuit, which may (probably will) use different hardware than the first circuit did, depending on what's available at that time in the network

Packet Switching

In this network type, no specific path is used for data transfer. Instead, the data is chopped up into small pieces called *packets* and sent over the network. The packets can be routed, combined or fragmented, as required to get them to their eventual destination. On the receiving end, the process is reversed—the data is read from the packets and re-assembled into the form of the original data. A packet-switched network is more analogous to the postal system than it is to the telephone system (though the comparison isn't perfect.) An example is shown in Figure 2.

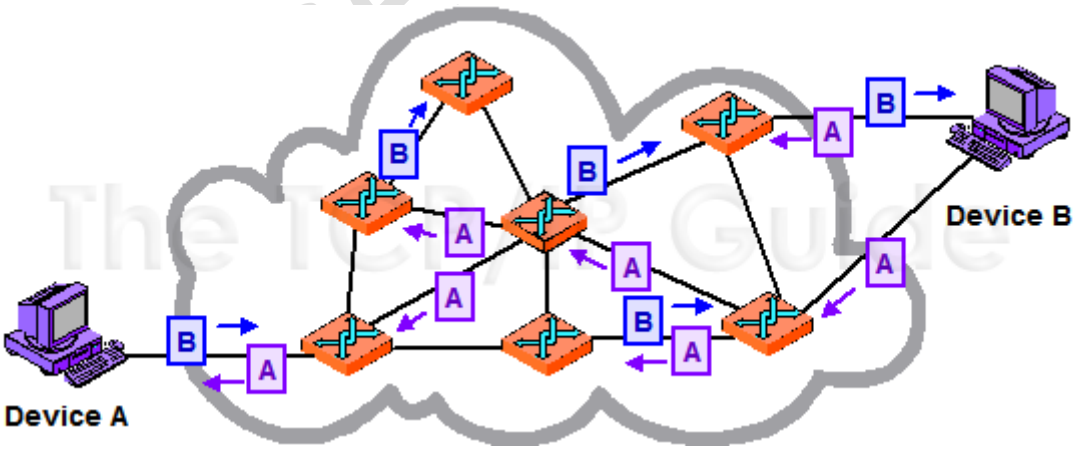


Figure 2: Packet Switching

In a packet-switched network, no circuit is set up prior to sending data between devices. Blocks of data, even from the same file or communication, may take any number of paths as it journeys from one device to another. Compare this to Figure 1

Comparing Circuit Switching and Packet Switching

A common temptation when considering alternatives such as these is to ask which is "better"—and as usually is the case, the answer is "neither". There are places where one is more suited than the other, but if one were clearly superior, both methods wouldn't be used.

One important issue in selecting a switching method is whether the network medium is *shared* or *dedicated*. Your phone line can be used for establishing a circuit because you are the only one who can use it—assuming you can keep that pesky wife/husband/child/sister/brother/father/mother off the phone.

However, this doesn't work well in LANs, which typically use a single shared medium and baseband signaling. If two devices were to establish a connection, they would "lock out" all the other devices for a long period of time. It makes more sense to chop the data into small pieces and send them one at a time. Then, if two other devices want to communicate, *their* packets can be interspersed and everyone can share the network.

The ability to have many devices communicate simultaneously without dedicated data paths is one reason why packet switching is becoming predominant today. However, there are some disadvantages of packet switching compared to circuit switching. One is that since all data does not take the same, predictable path between devices, it is possible that some pieces of data may get lost in transit, or show up in the incorrect order. In some situations this does not matter, while in others it is very important indeed.

While the theoretical difference between circuit and packet switching is pretty clear-cut, understanding how they are used is a bit more complicated. One of the major issues is that in modern networks, they are often combined. For example, suppose you connect to the Internet using a dial-up modem. You will be using IP datagrams (packets) to carry higher-layer data, but it will be over the circuit-switched telephone network. Yet the data may be sent over the telephone system in digital packetized form. So in some ways, both circuit switching and packet switching are being used concurrently.

The **Internet Protocol** (**IP**) is a protocol used for communicating data across packet switched internetwork using the Internet Protocol Suite, also referred to as TCP/IP.

IP is the primary protocol in the Internet Layer of the Internet Protocol Suite and has the task of delivering distinguished protocol datagrams (packets) from the source host to the destination host solely based on their addresses. For this purpose the Internet Protocol defines addressing methods and structures for datagram encapsulation. The first major version of addressing structure, now referred to as Internet Protocol Version 4 (IPv4) is still the dominant protocol of the Internet, although the successor, Internet Protocol Version 6 (IPv6) is being deployed actively worldwide

IP encapsulation

Data from an upper layer protocol is encapsulated as packets/datagrams (the terms are synonymous in IP). Circuit setup is not needed before a host may send packets to another host that it has previously not communicated with (a characteristic of packet-switched networks), thus IP is a connectionless protocol. This is in contrast to public switched telephone networks that require the setup of a circuit for each phone call (*connection-oriented* protocol).

Services provided by IP

Because of the abstraction provided by encapsulation, IP can be used over a heterogeneous network, i.e., a network connecting computers may consist of a combination of Ethernet, ATM, FDDI, Wi-Fi, token ring, or others. Each link layer implementation may have its own method of addressing (or possibly the complete lack of it), with a corresponding need to resolve IP addresses to data link addresses. This address resolution is handled by the Address Resolution Protocol (ARP) for IPv4 and Neighbor Discovery Protocol (NDP) for IPv6.

Reliability

The design principles of the Internet protocols assume that the network infrastructure is inherently unreliable at any single network element or transmission medium and that it is dynamic in terms of availability of links and nodes. No central monitoring or performance measurement facility exists that tracks or maintains the state of the network. For the benefit of reducing network complexity, the intelligence in the network is purposely mostly located in the end nodes of each data transmission, cf. end-to-end principle. Routers in the transmission path simply forward packets to next known local gateway matching the routing prefix for the destination address.

As a consequence of this design, the Internet Protocol only provides best effort delivery and its service can also be characterized as *unreliable*. In network architectural language it is a *connection-less* protocol, in contrast to so-called connection-oriented modes of transmission. The lack of reliability allows any of the following fault events to occur:

- data corruption
- lost data packets
- duplicate arrival
- out-of-order packet delivery; meaning, if packet 'A' is sent before packet 'B', packet 'B' may arrive before packet 'A'. Since routing is dynamic and there is no memory in the network about the path of prior packets, it is possible that the first packet sent takes a longer path to its destination.

The only assistance that the Internet Protocol provides in Version 4 (IPv4) is to ensure that the IP packet header is error-free through computation of a checksum at the routing nodes. This has the side-effect of discarding packets with bad headers on the spot. In this case no notification is required to be sent to either end node, although a facility exists in the Internet Control Message Protocol (ICMP) to do so.

IPv6, on the other hand, has abandoned the use of IP header checksums for the benefit of rapid forwarding through routing elements in the network.

The resolution or correction of any of these reliability issues is the responsibility of an upper layer protocol. For example, to ensure in-order delivery the upper layer may have to cache data until it can be passed to the application.

In addition to issues of reliability, this dynamic nature and the diversity of the Internet and its components provide no guarantee that any particular path is actually capable of, or suitable for performing the data transmission requested, even if the path is available and reliable. One of the technical constraints is the size of data packets allowed on a given link. An application must assure that it uses proper transmission characteristics. Some of this responsibility lies also in the upper layer protocols between application and IP. Facilities exist to examine the maximum transmission unit (MTU) size of the local link, as well as for the entire projected path to the destination when using IPv6. The IPv4 internetworking layer has the capability to automatically fragment the original datagram into smaller units for transmission. In this case, IP does provide re-ordering of fragments delivered out-of-order.

Transmission Control Protocol (TCP) is an example of a protocol that will adjust its segment size to be smaller than the MTU. User Datagram Protocol (UDP) and Internet

Control Message Protocol (ICMP) disregard MTU size thereby forcing IP to fragment oversized datagrams.

IP addressing and routing

Perhaps the most complex aspects of IP are IP addressing and routing. Addressing refers to how end hosts become assigned IP addresses and how subnetworks of IP host addresses are divided and grouped together. IP routing is performed by all hosts, but most importantly by internetwork routers, which typically use either interior gateway protocols (IGPs) or external gateway protocols (EGPs) to help make IP datagram forwarding decisions across IP connected networks.

Address Resolution Protocol (ARP) &Reverse Address Resolution Protocol(RARP)

RARP (Reverse Address Resolution Protocol) is a protocol by which a physical machine in a local area network can request to learn its IP address from a gateway server's Address Resolution Protocol (ARP) table or cache. A network administrator creates a table in a local area network's gateway router that maps the physical machine (or Media Access Control - MAC address) addresses to corresponding Internet Protocol addresses. When a new machine is set up, its RARP client program requests from the RARP server on the router to be sent its IP address. Assuming that an entry has been set up in the router table, the RARP server will return the IP address to the machine which can store it for future use.

RARP is available for Ethernet, Fiber Distributed-Data Interface, and Token Ring LANs.

The address resolution protocol (arp) is a protocol used by the Internet Protocol (IP) , specifically IPv4, to map IP network addresses to the hardware addresses used by a data link protocol. The protocol operates below the network layer as a part of the interface between the OSI network and OSI link layer. It is used when IPv4 is used over Ethernet.

The term address resolution refers to the process of finding an address of a computer in a network. The address is "resolved" using a protocol in which a piece of information is sent by a client process executing on the local computer to a server process executing on a remote computer. The information received by the server allows the server to uniquely identify the network system for which the address was required and therefore to provide the required address. The address resolution

procedure is completed when the client receives a response from the server containing the required address.

An Ethernet network uses two hardware addresses which identify the source and destination of each frame sent by the Ethernet. The destination address (all 1's) may also identify a broadcast packet (to be sent to all connected computers). The hardware address is also known as the Medium Access Control (MAC) address, in reference to the standards which define Ethernet. Each computer network interface card is allocated a globally unique 6 byte link address when the factory manufactures the card (stored in a PROM). This is the normal link source address used by an interface. A computer sends all packets which it creates with its own hardware source link address, and receives all packets which match the same hardware address in the destination field or one (or more) pre-selected broadcast/multicast addresses.

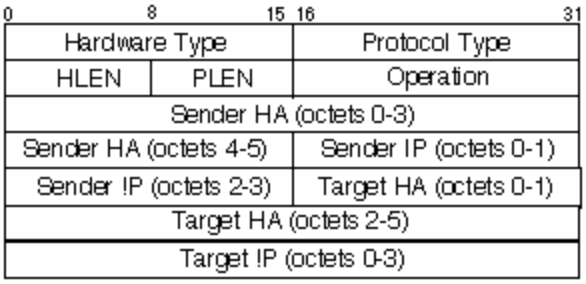
The Ethernet address is a link layer address and is dependent on the interface card which is used. IP operates at the network layer and is not concerned with the link addresses of individual nodes which are to be used.The address resolution protocol (arp) is therefore used to translate between the two types of address. The arp client and server processes operate on all computers using IP over Ethernet. The processes are normally implemented as part of the software driver that drives the network interface card.

There are four types of arp messages that may be sent by the arp protocol. These are identified by four values in the "operation" field of an arp message. The types of message are:

1. ARP request
2. ARP reply
3. RARP request
4. RARP reply

.

The format of an arp message is shown below:



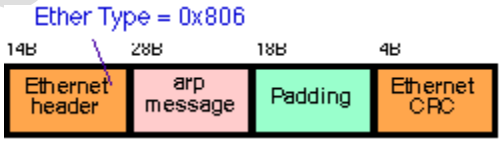
Format of an arp message used to resolve the remote MAC Hardware Address (HA)

To reduce the number of address resolution requests, a client normally **caches** resolved addresses for a (short) period of time. The arp cache is of a finite size, and would become full of incomplete and obsolete entries for computers that are not in use if it was allowed to grow without check. The arp cache is therefore periodically flushed of all entries. This deletes unused entries and frees space in the cache. It also removes any unsuccessful attempts to contact computers which are not currently running.

If a host changes the MAC address it is using, this can be detected by other hosts when the cache entry is deleted and a fresh arp message is sent to establish the new association. The use of gratuitous arp (e.g. triggered when the new NIC interface is enabled with an IP address) provides a more rapid update of this information.

Example of use of the Address Resolution Protocol (arp)

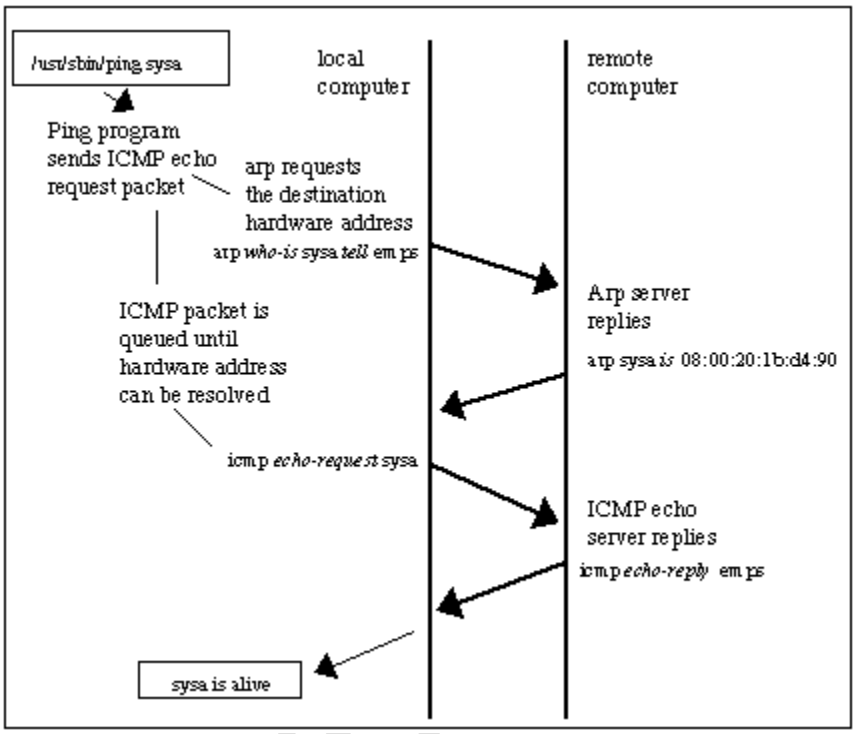
The figure below shows the use of arp when a computer tries to contact a remote computer on the same LAN (known as "sysa") using the "ping" program. It is assumed that no previous IP datagrams have been received form this computer, and therefore arp must first be used to identify the MAC address of the remote computer.



The arp request message ("who is X.X.X.X tell Y.Y.Y.Y", where X.X.X.X and Y.Y.Y.Y are IP addresses) is sent using the Ethernet broadcast address, and an Ethernet protocol type of value 0x806. Since it is broadcast, it is received by all systems in the same collision domain (LAN). This is ensures that is the target of the query is connected to the network, it will receive a copy of the query. Only this system responds. The other systems discard the packet silently.

The target system forms an arp response ("X.X.X.X is hh:hh:hh:hh:hh:hh", where

hh:hh:hh:hh:hh:hh is the Ethernet source address of the computer with the IP address of X.X.X.X). This packet is unicast to the address of the computer sending the query (in this case Y.Y.Y.Y). Since the original request also included the hardware address (Ethernet source address) of the requesting computer, this is already known, and doesn't require another arp message to find this out.



Gratuitous ARP

Gratuitous ARP is used when a node (end system) has selected an IP address and then wishes to defend its chosen address on the local area network (i.e. to check no other node is using the same IP address). It can also be used to force a common view of the node's IP address (e.g. after the IP address has changed).

Use of this is common when an interface is first configured, as the node attempts to clear out any stale caches that might be present on other hosts. The node simply sends an arp request for itself.

Proxy ARP

Proxy ARP is the name given when a node responds to an arp request on behalf of another node. This is commonly used to redirect traffic sent to one IP address to another system.Proxy ARP can also be used to subvert traffic away from the intended recipient. By responding instead of the intended recipient, a node can pretend to be a different node in a

network, and therefore force traffic directed to the node to be redirected to itself. The node can then view the traffic (e.g. before forwarding this to the originally intended node) or could modify the traffic. Improper use of Proxy ARP is therefore a significant security vulnerability and some networks therefore implement systems to detect this. Gratuitous ARP can also help defend the correct IP to MAC bindings.

The **Dynamic Host Configuration Protocol** (**DHCP**) is a computer networking protocol used by hosts (*DHCP clients*) to retrieve IP address assignments and other configuration information.

DHCP uses a client-server architecture. The client sends a broadcast request for configuration information. The *DHCP server* receives the request and responds with configuration information from its configuration database.In the absence of DHCP, all hosts on a network must be manually configured individually - a time-consuming and often error-prone undertaking.DHCP is a popular with ISP's because it allows a host to obtain a temporary IP addressWhen a DHCP-configured client (a computer or any other network-aware device) connects to a network, the DHCP client sends a broadcast query requesting necessary information from a DHCP server. The DHCP server manages a pool of IP addresses and information about client configuration parameters such as default gateway, domain name, the name servers, other servers such as time servers, and so forth. On receiving a valid request, the server assigns the computer an IP address, a lease (length of time the allocation is valid), and other IP configuration parameters, such as the subnet mask and the default gateway. The query is typically initiated immediately after booting, and must complete before the client can initiate IP-based communication with other hosts.

Depending on implementation, the DHCP server may have three methods of allocating IP-addresses:

- *dynamic allocation*: A network administrator assigns a range of IP addresses to DHCP, and each client computer on the LAN has its IP software configured to request an IP address from the DHCP server during network initialization. The request-and-grant process uses a lease concept with a controllable time period, allowing the DHCP server to reclaim (and then reallocate) IP addresses that are not renewed (dynamic re-use of IP addresses).
- *automatic allocation*: The DHCP server permanently assigns a free IP address to a requesting client from the range defined by the administrator. This is like dynamic allocation, but the DHCP server keeps a table of past IP address assignments, so that it can preferentially assign to a client the same IP address that the client previously had.
- *static allocation*: The DHCP server allocates an IP address based on a table with MAC address/IP address pairs, which are manually filled in (perhaps by a network

administrator). Only requesting clients with a MAC address listed in this table will be allocated an IP address. This feature (which is not supported by all devices) is variously called *Static DHCP Assignment* (by DD-WRT), *fixed-address* (by the dhcpd documentation), *DHCP reservation* or *Static DHCP* (by Cisco/Linksys), and *IP reservation* or *MAC/IP binding* (by various other router manufacturers).

DHCP discovery

The client broadcasts messages on the physical subnet to discover available DHCP servers. Network administrators can configure a local router to forward DHCP packets to a DHCP server from a different subnet. This client-implementation creates a User Datagram Protocol (UDP) packet with the broadcast destination of 255.255.255.255 or the specific subnet broadcast address.

A DHCP client can also request its last-known IP address (in the example below, 192.168.1.100). If the client remains connected to a network for which this IP is valid, the server might grant the request. Otherwise, it depends whether the server is set up as authoritative or not. An authoritative server will deny the request, making the client ask for a new IP immediately. A non-authoritative server simply ignores the request, leading to an implementation-dependent timeout for the client to give up on the request and ask for a new IP address.

DHCP offer

When a DHCP server receives an IP lease request from a client, it reserves an IP address for the client and extends an IP lease offer by sending a DHCPOFFER message to the client. This message contains the client's MAC address, the IP address that the server is offering, the subnet mask, the lease duration, and the IP address of the DHCP server making the offer.

The server determines the configuration based on the client's hardware address as specified in the CHADDR (Client Hardware Address) field. Here the server, 192.168.1.1, specifies the IP address in the YIADDR (Your IP Address) field.

DHCP request

A client can receive DHCP offers from multiple servers, but it will accept only one DHCP offer and broadcast a DHCP request message. Based on the Transaction ID field in the request, servers are informed whose offer the client has accepted. When other DHCP servers receive this message, they withdraw any offers that they might

have made to the client and return the offered address to the pool of available addresses.

DHCP acknowledgement

When the DHCP server receives the DHCPREQUEST message from the client, the configuration processes enters its final phase. The acknowledgement phase involves sending a DHCPACK packet to the client. This packet includes the lease duration and any other configuration information that the client might have requested. At this point, the IP configuration process is completed.

The protocol expects the DHCP client to configure its network interface with the negotiated parameters.

Internet Control Message Protocol (**ICMP**) is one of the core protocols of the Internet Protocol Suite. It is chiefly used by networked computers' operating systems to send error messages—indicating, for instance, that a requested service is not available or that a host or router could not be reached.

ICMP[1] relies on IP to perform its tasks, and it is an integral part of IP. It differs in purpose from transport protocols such as TCP and UDP in that it is typically not used to send and receive data between end systems. It is usually not used directly by user network applications, with some notable exceptions being the ping tool and traceroute.

ICMP for Internet Protocol version 4 (IPv4) is also known as ICMPv4. IPv6 has a similar protocol, ICMPv6.

Internet Control Message Protocol is part of the Internet Protocol Suite as defined in RFC 792. ICMP messages are typically generated in response to errors in IP datagrams (as specified in RFC 1122) or for diagnostic or routing purposes.

ICMP messages are constructed at the IP layer, usually from a normal IP datagram that has generated an ICMP response. IP encapsulates the appropriate ICMP message with a new IP header (to get the ICMP message back to the original sending host) and transmits the resulting datagram in the usual manner.

For example, every machine (such as an intermediate router) that forwards an IP datagram has to decrement the time to live (TTL) field of the IP header by one; if the TTL reaches 0, an ICMP Time to live exceeded in transit message is sent to the source of the datagram.

Each ICMP message is encapsulated directly within a single IP datagram, and thus, like UDP, ICMP is unreliable.

Although ICMP messages are contained within standard IP datagrams, ICMP messages are usually processed as a special case, distinguished from normal IP processing, rather than processed as a normal sub-protocol of IP. In many cases, it is necessary to inspect the contents of the ICMP message and deliver the appropriate error message to the application that generated the original IP packet, the one that prompted the sending of the ICMP message.

Many commonly-used network utilities are based on ICMP messages. The trace route command is implemented by transmitting UDP datagrams with specially set IP TTL header fields, and looking for ICMP Time to live exceeded in transit (above) and "Destination unreachable" messages generated in response. The related ping utility is implemented using the ICMP "Echo request" and "Echo reply" messages.

ICMP segment structure

Header

The ICMP header starts after the IPv4 header.

Bits	**0-7**	**8-15**	**16-23**	**24-31**
0	Type	Code	Checksum	
32	ID		Sequence	

- **Type** - ICMP type as specified below.
- **Code** - further specification of the ICMP type; e.g. : an ICMP Destination Unreachable might have this field set to 1 through 15 each bearing different meaning.
- **Checksum** - This field contains error checking data calculated from the ICMP header+data, with value 0 for this field. The algorithm is the same as the header checksum for IPv4.
- **ID** - This field contains an ID value, should be returned in case of ECHO REPLY.
- **Sequence** - This field contains a sequence value, should be returned in case of ECHO REPLY.

Padding data

Padding data follows the ICMP header (in octets):

- The Linux "ping" utility pads ICMP to a total size of 56 bytes in addition to the 8 octet header.
- Windows "ping.exe" pads to a total size of 32 bytes in addition to the 8 octet header.

Queuing Discipline:

A network of *m* interconnected queues is known as a **BCMP network** if each of the queues is of one of the following four types:

1. FCFS discipline where all customers have the same negative exponential service time distribution. The service rate can be state dependent, so write for the service rate when the queue length is *j*.
2. Processor sharing queues
3. Infinite server queues
4. LCFS with pre-emptive resume (work is not lost)

In the final three cases, service time distributions must have rational Laplace transforms. This means the Laplace transform must be of the form

L(s)=N(s)/D(s)

Also, the following conditions must be met.

1. external arrivals to node *i* (if any) form a Poisson process,
2. a customer completing service at queue *i* will either move to some new queue *j* with (fixed) probability P_{ij} or leave the system with probability , which is non-zero for some subset of the queues.

Routing algorithm:

Routing Information Protocol
Routing Information Protocol (RIP) is a simple routing protocol, originally defined in 1988 as RFC 1058 and more recently as RFC 1723, based upon the original ARPANET routing algorithm. RIP involves a router calculating the best route to all other routers in a network using a **shortest path** algorithm attributable to Bellman (1957) and F ord and Fulkerson (1962). The shortest path in this case is the one that passes through the least number of routers. Each router traversed is known as a **hop**. Therefore the shortest path is described by a **hop count**, or **distance vector**. This is a crude measure of distance or cost to reach a destination. It takes no account of other factors such as propagation delay or

available bandwidth. RIP then builds a routing database that contain stables of the best routes to all the other routers. Each router then advertises its own routing tables to all other routers. Although RIP is simple to implement it is onlyefficient in small networks since, as the size of a network grows, RIP datagrams can become very long, thus consuming substantial amounts of bandwidth.

Distance-vector routing protocol

In computer communication theory relating to packet-switched networks, a **distance-vector routing protocol** is one of the two major classes of routing protocols, the other major class being the link-state protocol. A distance-vector routing protocol uses the Bellman-Ford algorithm to calculate paths.

A distance-vector routing protocol requires that a router informs its neighbors of topology changes periodically and, in some cases, when a change is detected in the topology of a network. Compared to link-state protocols, which require a router to inform all the nodes in a network of topology changes, distance-vector routing protocols have less computational complexity and message overhead

Distance Vector means that Routers are advertised as vector of distance and direction. 'Direction' is represented by next hop address and exit interface, whereas 'Distance' uses metrics such as hop count.

Routers using distance vector protocol do not have knowledge of the entire path to a destination. Instead DV uses two methods:

1. Direction in which or interface to which a packet should be forwarded.
2. Distance from its destination.

Examples of distance-vector routing protocols include RIPv1 and 2 and IGRP. EGP and BGP are not pure distance-vector routing protocols because a distance-vector protocol calculates routes based only on link costs whereas in BGP, for example, the local route preference value takes priority over the link cost.

The methods used to calculate the best path for a network are different between different routing protocols but the fundamental features of distance-vector algorithms are the same across all DV based protocols.

Distance Vector means that Routers are advertised as vector of distance and Direction. Direction is simply next hop address and exit interface and Distance means such as hop count.

Routers using distance vector protocol do not have knowledge of the entire path to a destination. Instead DV uses two methods:

1. Direction in which or interface to which a packet should be forwarded.
2. Distance from its destination.

As the name suggests the DV protocol is based on calculating the direction and distance to any link in a network. The cost of reaching a destination is calculated using various route metrics. RIP uses the hop count of the destination whereas IGRP takes into account other information such as node delay and available bandwidth.

Updates are performed periodically in a distance-vector protocol where all or part of a router's routing table is sent to all its neighbors that are configured to use the same distance-vector routing protocol. RIP supports cross-platform distance vector routing whereas IGRP is a Cisco Systems proprietary distance vector routing protocol. Once a router has this information it is able to amend its own routing table to reflect the changes and then inform its neighbors of the changes. This process has been described as 'routing by rumor' because routers are relying on the information they receive from other routers and cannot determine if the information is actually valid and true. There are a number of features which can be used to help with instability and inaccurate routing information.

Open Shortest Path First

A more powerful routing protocol developed subsequent to RIP, defined originally as RFC 1131 and more recently as RFC 2178, is called **Open Shortest Path First** (OSPF). It is the preferred routing protocol for medium or large networks which, in OSPF, are referred to as **autonomous systems** (ASs). OSPF endeavours to establish a least-cost shortest route within an autonomous system. Cost does not necessarily involve monetary considerations, but means that parameters are used that are of particular importance to the network operator. They may be financial or could be based on delay or transmission rate. Such parameters are known as **metrics**. Whereas RIP is a distance-vector-based protocol, OSPF is described as a **link state** routing protocol. This is because it only advertises the changes in the state of its routing tables to other routers using **link state advertisements** rather than the full tables. Link state advertisements that are exchanged between routers produce much less traffic than is generated by RIP datagrams. Each router holds a database, each containing the same information, as a result of the exchange of link state update messages. It is worth noting that, unlike RIP, OSPF only exchanges changes in a network rather than complete topologies. This is a major advantage over RIP and results in much less information being exchanged. Cost metrics are indicated at the output ports of each router and may be deduced by router software or configured by a network administrator.

Since autonomous systems can be large, OSPF allows for them to be divided into

numbered areas such that topology information is largely contained within a single area. Area 0 is a special case, termed the backbone area, and is arranged so that all other areas can be interconnected through it. Routers operating in an OSPF environment can be categorized by their connectivity with other routers and the type of traffic that they carry. A **stub router** has only one entry/exit point to the
router and all traffic passes through this one point, whereas **multihomed routers** have more than one connection to other routers.

OSPF, in common with certain other routing protocols, can also use equal-cost multipath routing to avoid some parts of the network becoming congested while other parts are not fully utilized. Such procedures are not part of the OSPF protocol and an equal-cost multipath algorithm is analysed in RFC 2992. Equal-cost multipath routing ,as the name implies, is a technique for routing datagrams along multiple paths of equal cost. The forwarding algorithm identifies paths by next-hop and the router must then decide which next-hop (path) to use when forwarding a datagram. For example,a round-robin technique might be used whereby each eligible path is used in turn. However, such an approach is not suitable for TCP sessions, which perform better if the path they flow along does not change while the stream is connected. A more useful method for determining which next-hop to use is known as a **hash-threshold**. The router first selects a **key** by performing a cyclic redundancy check (known as a **hash**) over the datagram header fields that identify a flow (typically the source and destination IP addresses). With the very simplest implementation of the algorithm, the combined source and destination addresses are divided by the number of equal-cost routes and the remainder of this division is the key. The router then uses the key to determine which next-hop to use. This should result in a more balanced use of the available paths and is known as **load balancing**.

Subnetting: A subnetwork, or **subnet**, is a logically visible, distinctly addressed part of a single Internet Protocol network.[1][2] The process of **subnetting** is the division of a computer network into groups of computers that have a common, designated IP address routing prefix.

Subnetting breaks a network into smaller realms that may use existing address space more efficiently, and, when physically separated, may prevent excessive rates of Ethernet packet collision in a larger network. The subnets may be arranged logically in a hierarchical architecture, partitioning the organization's network address space (see also Autonomous System) into a tree-like routing structure. Routers are used to interchange traffic between subnetworks and constitute logical or physical borders between the subnets. They manage traffic between subnets based on the high-order bit sequence (routing prefix) of the addresses.

A routing prefix is the sequence of leading (most-significant) bits of an IP address that precede the portion of the address used as host identifier and, if applicable, the set of bits

that designate the subnet number. Routing prefixes are expressed in CIDR notation, which uses the first address of a network followed by the bit-length of the prefix, separated by a slash (*/*) character. For example, 192.168.1.0/24 is the prefix of the IPv4 network starting at the given address, having 24 bits allocated for the network number, and the rest (8 bits) reserved for host addressing. The IPv6 address specification 2001:db8::/32 is a large network for 2^{96} hosts, having a 32-bit routing prefix.

In IPv4 networks, the routing prefix is traditionally expressed as a *subnet mask*, which is the prefix bit mask expressed in quad-dotted decimal representation. For example, 255.255.255.0 is the subnet mask for the 192.168.1.0/24 prefix.

All hosts within a subnet can be reached in one routing hop, implying that all hosts in a subnet are connected to the same link.

A typical subnet is a physical network served by one router, for instance an Ethernet network, possibly consisting of one or several Ethernet segments or local area networks, interconnected by network switches and network bridges or a Virtual Local Area Network (VLAN). However, subnetting allows the network to be logically divided regardless of the physical layout of a network, since it is possible to divide a physical network into several subnets by configuring different host computers to use different routers.

While improving network performance, subnetting increases routing complexity, since each locally connected subnet must be represented by a separate entry in the routing tables of each connected router. However, by careful design of the network, routes to collections of more distant subnets within the branches of a tree-hierarchy can be aggregated by single routes. Existing subnetting functionality in routers made the introduction of Classless Inter-Domain Routing seamless.

Classless Inter-Domain Routing (**CIDR**) is a methodology of allocating IP addresses and routing Internet Protocol packets. It was introduced in 1993 to replace the prior addressing architecture of classful network design in the Internet with the goal to slow the growth of routing tables on routers across the Internet, and to help slow the rapid exhaustion of IPv4 addresses.[1][2]

IP addresses are described as consisting of two groups of bits in the address: the most significant part is the *network address* which identifies a whole network or subnet and the least significant portion is the *host identifier*, which specifies a particular host interface on that network. This division is used as the basis of traffic routing between IP networks and for address allocation policies. Classful network design for IPv4 sized the network address as one or more 8-bit groups, resulting in the blocks of Class A, B, or C addresses. Classless Inter-Domain Routing allocates address space to Internet service providers and end users on

any address bit boundary, instead of on 8-bit segments. In IPv6, however, the host identifier has a fixed size of 64 bits by convention, and smaller subnets are never allocated to end users.

CIDR notation uses a syntax of specifying IP addresses for IPv4 and IPv6, using the base address of the network followed by a slash and the size of the routing prefix, e.g., 192.168.0.0/16 (IPv4), and 2001:db8::/32 (IPv6).

CIDR is principally a bitwise, prefix-based standard for the interpretation of IP addresses. It facilitates routing by allowing blocks of addresses to be grouped together into single routing table entries. These groups, commonly called *CIDR blocks*, share an initial sequence of bits in the binary representation of their IP addresses. IPv4 CIDR blocks are identified using a syntax similar to that of IPv4 addresses: a four-part dotted-decimal address, followed by a slash, then a number from 0 to 32: *A.B.C.D/N*. The dotted decimal portion is interpreted, like an IPv4 address, as a 32-bit binary number that has been broken into four octets. The number following the slash is the prefix length, the number of shared initial bits, counting from the most significant bit of the address. When emphasizing only the size of a network, terms like */20* are used, which is a CIDR block with an unspecified 20-bit prefix.

An IP address is part of a CIDR block, and is said to match the CIDR prefix if the initial N bits of the address and the CIDR prefix are the same. Thus, understanding CIDR requires that IP address be visualized in binary. Since the length of an IPv4 address has 32 bits, an N-bit CIDR prefix leaves 32-N bits unmatched, meaning that 2^{32-N} IPv4 addresses match a given N-bit CIDR prefix. *Shorter* CIDR prefixes match more addresses, while *longer* CIDR prefixes match fewer. An address can match multiple CIDR prefixes of different lengths.

CIDR is also used with IPv6 addresses and the syntax semantic is identical. A prefix length can range from 0 to 128, due to the larger number of bits in the address, however, by convention a subnet on broadcast MAC layer networks always has 64-bit host identifiers. Larger prefixes are rarely used even on point-to-point links.

Assignment of CIDR blocks

The Internet Assigned Numbers Authority (IANA) issues to Regional Internet Registries (RIRs) large, short-prefix (typically /8) CIDR blocks. For example, 62.0.0.0/8, with over sixteen million addresses, is administered by RIPE NCC, the European RIR. The RIRs, each responsible for a single, large, geographic area (such as Europe or North America), then subdivide these allocations into smaller blocks and issue them to local Internet registries. This subdividing process can be repeated several times at different levels of delegation. End user networks receive subnets sized according to the size of their network and projected short term need. Networks served by a single ISP are encouraged by IETF

recommendations to obtain IP address space directly from their ISP. Networks served by multiple ISPs, on the other hand, may often obtain independent CIDR blocks directly from the appropriate RIR.

For example, in the late 1990s, the IP address 208.130.29.33 (since reassigned) was used by www.freesoft.org. An analysis of this address identified three CIDR prefixes. 208.128.0.0/11, a large CIDR block containing over 2 million addresses, had been assigned by ARIN (the North American RIR) to MCI. Automation Research Systems, a Virginia VAR, leased an Internet connection from MCI and was assigned the 208.130.28.0/22 block, capable of addressing just over 1000 devices. ARS used a /24 block for its publicly accessible servers, of which 208.130.29.33 was one.

All of these CIDR prefixes would be used, at different locations in the network. Outside of MCI's network, the 208.128.0.0/11 prefix would be used to direct to MCI traffic bound not only for 208.130.29.33, but also for any of the roughly two million IP addresses with the same initial 11 bits. Within MCI's network, 208.130.28.0/22 would become visible, directing traffic to the leased line serving ARS. Only within the ARS corporate network would the 208.130.29.0/24 prefix have been used.

Subnet masks

A subnet mask is a bitmask that encodes the prefix length in quad-dotted notation: 32 bits, starting with a number of 1 bits equal to the prefix length, ending with 0 bits, and encoded in four-part dotted-decimal format. A subnet mask encodes the same information as a prefix length, but predates the advent of CIDR. However, in CIDR notation, the prefix bits are always contiguous, whereas subnet masks may specify non-contiguous bits. However, this has no practical advantage for increasing efficiency.

Interdomain routing -BGP

The **Border Gateway Protocol** (**BGP**) is the core routing protocol of the Internet. It maintains a table of IP networks or 'prefixes' which designate network reachability among autonomous systems (AS). It is described as a path vector protocol. BGP does not use traditional Interior Gateway Protocol (**IGP**) metrics, but makes routing decisions based on path, network policies and/or rulesets.

BGP was created to replace the Exterior Gateway Protocol (**EGP**) routing protocol to allow fully decentralized routing in order to allow the removal of the NSFNet Internet

backbone network. This allowed the Internet to become a truly decentralized system. Since 1994, version four of the BGP has been in use on the Internet. All previous versions are now obsolete. The major enhancement in version 4 was support of Classless Inter-Domain Routing and use of route aggregation to decrease the size of routing tables. Since January 2006, version 4 is codified in RFC 4271, which went through well over 20 drafts based on the earlier RFC 1771 version 4. The RFC 4271 version corrected a number of errors, clarified ambiguities, and also brought the RFC much closer to industry practices.

Most Internet users do not use BGP directly. However, since most Internet service providers must use BGP to establish routing between one another (especially if they are multihomed), it is one of the most important protocols of the Internet. Compare this with Signaling System 7 (SS7), which is the inter-provider core call setup protocol on the PSTN. Very large private IP networks use BGP internally. An example would be the joining of a number of large Open Shortest Path First (OSPF) networks where OSPF by itself would not scale to size. Another reason to use BGP is multihoming a network for better redundancy either to multiple access points of a single ISP (RFC 1998) or to multiple ISPs.

Operation

BGP neighbors, or peers, are established by manual configuration between routers to create a TCP session on port 179. A BGP speaker will periodically send 19-byte keep-alive messages to maintain the connection (every 60 seconds by default). Among routing protocols, BGP is unique in using TCP as its transport protocol.

When BGP is running inside an autonomous system (AS), it is referred to as *Internal BGP* (*IBGP* or *Interior Border Gateway Protocol*). When it runs between autonomous systems, it is called *External BGP* (*EBGP* or *Exterior Border Gateway Protocol*). Routers on the boundary of one AS, exchanging information with another AS, are called border or edge routers. In the Cisco operating system, IBGP routes have an administrative distance of 200, which is less preferred than either external BGP or any interior routing protocol. Other router implementations also prefer EBGP to IGPs, and IGPs to IBGP.

Extensions negotiation

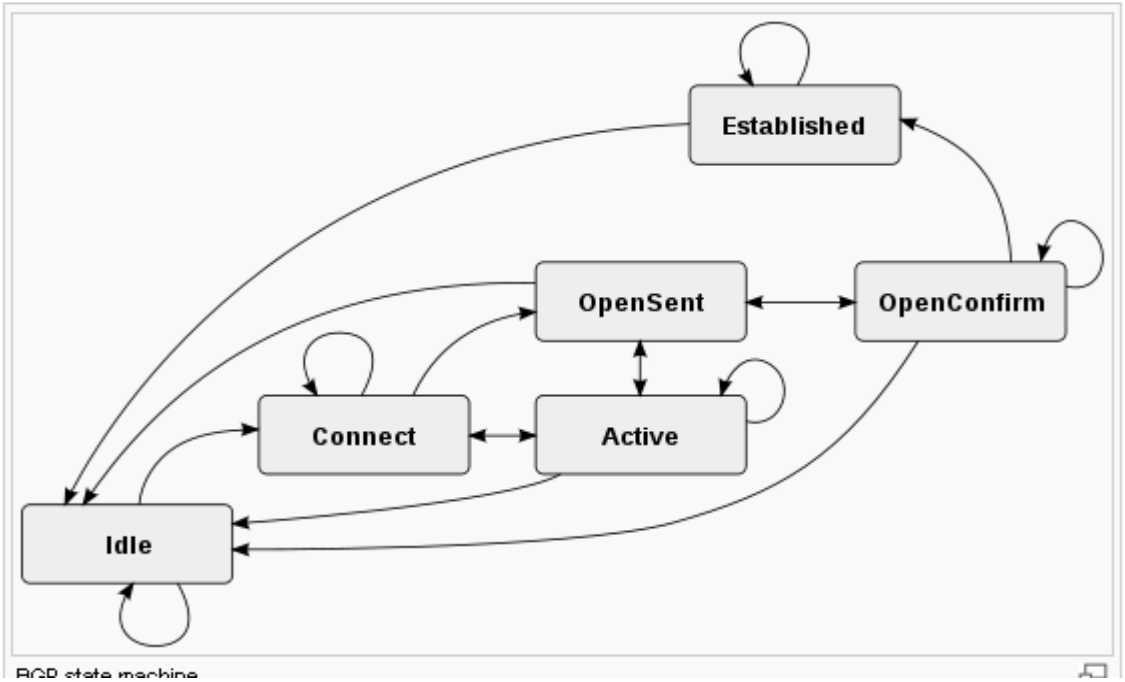
During the OPEN BGP speakers can negotiate optional capabilities of the session, including multiprotocol extensions and various recovery modes. If the multiprotocol extensions to BGP are negotiated at the time of creation, the BGP speaker can prefix the Network Layer Reachability Information (NLRI) it advertises with an address family prefix. These families include the default IPv4, but also IPv6, IPv4 and IPv6

Virtual Private Networks, and multicast BGP. Increasingly, BGP is used as a generalized signaling protocol to carry information about routes that may not be part of the global Internet, such as VPNs.

Finite state machine

BGP state machine

In order to make decisions in its operations with other BGP peers, a BGP peer uses a simple finite state machine (FSM) that consists of six states: Idle, Connect, Active, OpenSent, OpenConfirm, and Established. For each peer-to-peer session, a BGP implementation maintains a state variable that tracks which of these six states the session is in. The BGP protocol defines the messages that each peer should exchange in order to change the session from one state to another. The first mode is the “Idle” mode. In this mode BGP initializes all resources, refuses all inbound BGP connection attempts, and initiates a TCP connection to the peer. The second state is “Connect”. In this state the router waits for the TCP connection to complete, transitioning to the "OpenSent" state if successful. If not, it resets the ConnectRetry timer and transitions to the "Active" state upon expiration. In the "Active" state, the router resets the ConnectRetry timer to zero, and returns to the "Connect" state. After "OpenSent," the router sends an Open message, and waits for one in return. Keepalive messages are exchanged next, and upon successful receipt, the router is placed in the “Established” state. Once established the router can now send/receive Keepalive, Update, and Notification messages to/from its peer



BGP State Machine

Internet Protocol version 6 (**IPv6**) is the next-generation Internet Protocol version designated as the successor to IPv4, the first implementation used in the Internet that is still in dominant use currently. It is an Internet Layer protocol for packet-switched internetworks. The main driving force for the redesign of Internet Protocol is the foreseeable IPv4 address exhaustion. IPv6 was defined in December 1998 by the Internet Engineering Task Force (IETF) with the publication of an Internet standard specification, RFC 2460.

IPv6 has a vastly larger address space than IPv4. This results from the use of a 128-bit address, whereas IPv4 uses only 32 bits. The new address space thus supports 2^{128} (about 3.4×10^{38}) addresses. This expansion provides flexibility in allocating addresses and routing traffic and eliminates the primary need for network address translation (NAT), which gained widespread deployment as an effort to alleviate IPv4 address exhaustion.

IPv6 also implements new features that simplify aspects of address assignment (stateless address autoconfiguration) and network renumbering (prefix and router announcements) when changing Internet connectivity providers. The IPv6 subnet size has been standardized by fixing the size of the host identifier portion of an address to 64 bits to facilitate an automatic mechanism for forming the host identifier from Link Layer media addressing information (MAC address).

Network security is integrated into the design of the IPv6 architecture. Internet Protocol Security (IPsec) was originally developed for IPv6, but found widespread optional deployment first in IPv4 (into which it was back-engineered). The IPv6 specifications mandate IPsec implementation as a fundamental interoperability requirement.

Congestion avoidance in n/w Layer:

Congestion control refers to technique and mecchanism that can either prevent congestion,before it happens or remove congestion,after it ahs happened.It is divided into Open-Loop Congestion Control and Closed loop congestion control

Open loop Congestion control:

In this type,policies are applied to prevent congestion before it happen.

Retransmission Policy:

A good retransmission policy can prevent congestion .The retransmission policy and the retransmission timers must be designed to optimize
Effiency and at the same time prevent congestion .

Window Policy:

The type of the window at the sender may also affect congestion.The selective Repeat window is better than the Go-Back-N window for Congestion control.

Acknowledgment :

The acknowledgment policy imposed by the receiver may affect congestion .If the receiver does not acknowledge every packet it receives ,it may slow down the sender and help prevent congestion.

Discarding Policy:

A good discarding policy by the routers may prevnt congestion and at the same time may not harm the ingrity of the transmission .

Admission policy:

An admission Policy, which is a quality of service mechanism, can also prevent congestion in virtual circuit networks.

Closed –Loop Congestion Control:

It is mechanism try to alleviate congestion after it happens.Several mechanisms have been used by different protocols

Back Pressure:

When a router is congested ,it can inform the previous upstream router to reduce the rate of outgoing packet.

Choke Point:

A Choke point is a packet sent by a router to the source inform it of congestion.This type of control is similar to ICMP' s source quench packet.

Implicit signaling :

The source can detect an implicit signal concering congestion and slow down its sender rate.

Explicit signaling :

The router that experience congestion can send an explicit signal ,the setting of a bit in a packet .