Internet Control Message Protocol (ICMP)

- Internet Control Message Protocol (ICMP), documented in RFC 792, is a required protocol tightly integrated with IP.

- ICMP messages, delivered in IP packets, are used for out-of-band messages carrying network error messages and reports other conditions that require the attention of the network software.

- Of course, since ICMP uses IP, ICMP packet delivery is unreliable, so hosts can't count on receiving ICMP packets for any network problem. Some of ICMP's functions are to:

  - Announce network errors, such as a host or entire portion of the network being unreachable, due to some type of failure.

  - A TCP or UDP packet directed at a port number with no receiver attached is also reported via ICMP.

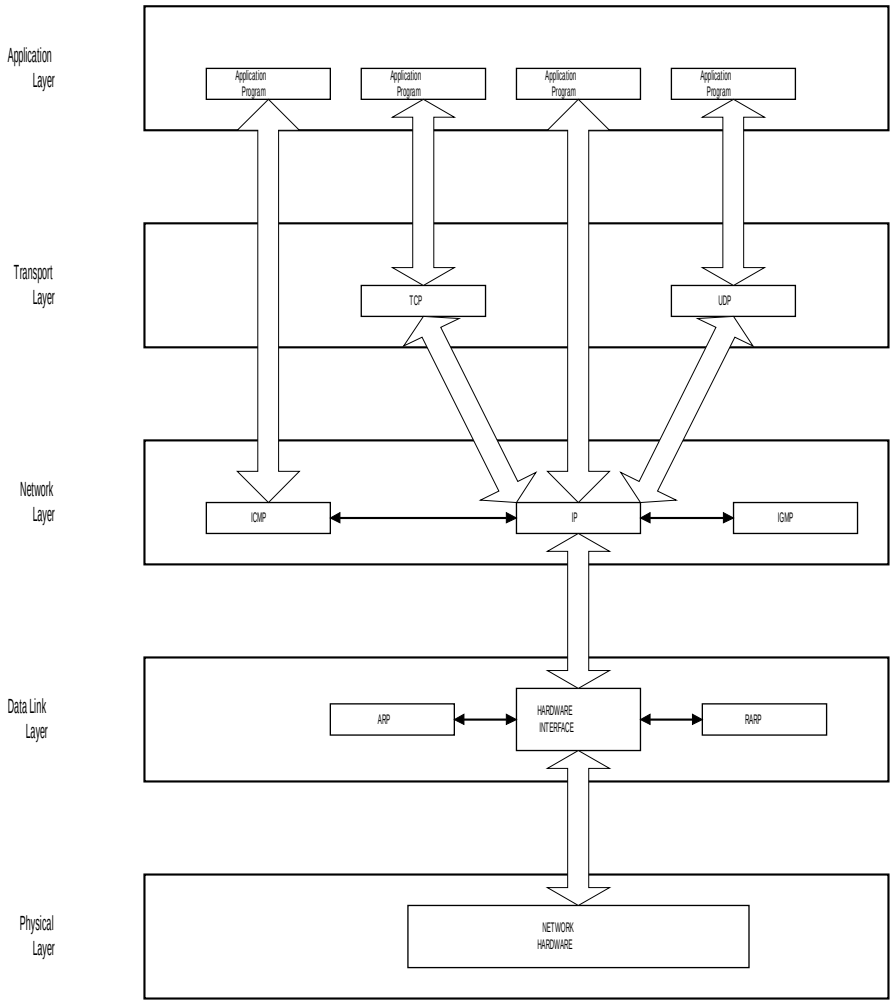
  - Announce network congestion. When a router begins buffering too many packets, due to an inability to transmit them as fast as they are being received, it will generate ICMP Source Quench messages.

  - Directed at the sender, these messages should cause the rate of packet transmission to be slowed. Of course, generating too many Source Quench messages would cause even more network congestion, so they are used sparingly.

  - Assist Troubleshooting. ICMP supports an Echo function, which just sends a packet on a round--trip between two hosts. Ping, a common network management tool, is based on this feature. Ping will transmit a series of packets, measuring average round--trip times and computing loss percentages.

  - Announce Timeouts. If an IP packet's TTL field drops to zero, the router discarding the packet will often generate an ICMP packet announcing this fact. TraceRoute is a tool which maps network routes by sending packets with small TTL values and watching the ICMP timeout announcements.

- The following diagram shows the position of ICMP within the TCP/IP protocol stack.



- Like IP, ICMP is part of the network layer. However, ICMP acts as though it is a higher-level protocol than IP.

- ICMP communicates with and relies upon IP to deliver messages to other hosts on the network in much the same way as TCP and UDP.

- IP does not include any mechanisms for notifying a router or switch that a packet-delivery problem exists. This was done deliberately by the designers to simplify the IP design.

- ICMP adds this capability to the network layer in TCP/IP networks. The original purpose was to let routers and gateways report the cause of delivery errors to the transmitting host's network layer, which in turn, would decide how to respond to the errors.

- The protocol design however does not limit the use of ICMP to network routers - any TCP/IP host can use ICMP to transmit network error, control and information messages to another host.

- The ICMP messages are encapsulated in the data portion of IP datagrams.

- ICMP is used by a device, often a router, to report and acquire a wide range of communications-related information. Consider the following reasons why a device would have to report or acquire information:

    - A router can not find a host computer on a destination network or when a frame has circulated in a routing loop until its Time To Live counter has been decremented to zero then the router discards the message.

    - A user sends a message to connect to a non-existent program in a host computer (Telnet to a file-server that doesn't support Telnet, for example) then the host discards the message.

    - A host tries to contact a remote destination but it happens to use the wrong router to get to that remote destination; the router that was contacted must inform the host of the correct path.

- In each of these cases, the ICMP protocol has a specific message type that is used to report the error condition back to the originator.

- The ICMP header consists of three fields. An 8-bit **Type** value, an 8-bit **Code** value and a 16-bit **checksum**.

- The Type field specifies the message type and the Code indicates the reason for a particular error message.

- The 16 different message types are listed below.

Type	Query/Error	Description
0	QUERY	Echo reply
3	ERROR	Destination Unreachable
4	ERROR	Source quench
5	ERROR	Redirect
8	QUERY	Echo request
9	QUERY	Router advertisement
10	QUERY	Router solicitation
11	ERROR	Time exceeded
12	ERROR	Parameter problem
13	QUERY	Timestamp request
14	QUERY	Timestamp reply
15	QUERY	Information request (obsolete)
16	QUERY	Information reply (obsolete)
17	QUERY	Address Mask request
18	QUERY	Address Mask reply

- ICMP messages are completely defined in RFC 792 but below is a brief overview of the message types.

Echo and Echo Reply

- Used to implement the PING command. When an ICMP Echo is received it is responded to with an Echo Reply message.
- The reply packet tells the echo-request sender that the host computer that received the echo request is online and responding to network messages.

Destination Unreachable

- There are several reasons why an IP destination may be unreachable and the ICMP message reports the reason.
- This often indicates a configuration or other communications problem.
- The destination-unreachable type messages include more error codes than any other ICMP error message type.

Code	Description
0	Network unreachable
1	Host unreachable
2	Protocol unreachable
3	Port unreachable
4	Fragmentation require but don't fragment bit is set
5	Source route failed
6	Destination network unknown
7	Destination host unknown
8	Source host isolated (obsolete)
9	Destination network administratively prohibited
10	Destination host administratively prohibited
11	Network unreachable for TOS reasons
12	Host unreachable for TOS reasons
13	Communication administratively prohibited by filtering
14	Host precedence violation
15	Precedence cutoff in effect

Source Quench

- When a router is being overloaded by a high-powered host, the router can send a Source Quench message and cause the host to stop transmissions for a short period of time during which the router can catch up.

Redirect

- Recall that a TCP/IP routing table includes entries to reach any network on the Internet.
- However, each router only knows the next stop along the route. In many cases, a router may know more than one route to a destination network.
- Routers periodically exchange routing information with each other to help keep routing information current.
- A router can tell a host to use a different target address when sending frames to some remote location. This is a normal action observed at the beginning of a new conversation.
- The redirect message header contains a Router IP Address that tells the receiving host computer which router to use for future packet delivery.
- The ICMP module uses this information to update the host's routing table.
- A redirect error can be generated for the following reasons:

Code	Description
0	Redirect for network
1	Redirect for host
2	Redirect for TOS and network
3	Redirect for TOS and host

Router Advertisement/Solicitation

- Routers broadcast ICMP router-advertisement messages at random intervals. Typically between 450 and 600 seconds apart.
- Routers can notify other hosts that the router is going offline with router-advertisement messages.
- Typically, a host transmits three router-solicitation messages three seconds apart at boot time.
- The host stops sending the router-solicitation messages as soon it receives a router-advertisement message.
- These messages provide a much more sophisticated and reliable way to initialize host routing tables.

Time Exceeded

- A router has received a frame that has had its Time To Live field decremented to one. The router now decrements the field to zero and discards the frame.
- The Time Exceeded message is used to inform the originator of the frame that the discard action took place.
- This message type uses the following code types:

Code	Description
0	TTL equals 0 during transit
1	Fragment reassembly time exceeded

Parameter Problem

- Hosts or routers send parameter-problem messages when a routing or delivery failure occurs and the host cannot identify the cause of the problem.
- This is usually due to an error (probably a communications software bug!) in the construction of an IP datagram.
- The message type uses the following two codes:

Code	Description
0	IP header bad (a catchall error)
1	Required option missing

Timestamp Request/Reply

- ICMP can be used by a diagnostic program to time the round-trip delay in a communication path.
- There are three timestamp fields in the ICMP header that are used for this purpose. Each field represents the number of seconds past midnight.
- Before transmitting the packet, the sending host fills in the **Originate** field with the current time.
- The receiver fills in the **Receive** field with as soon as the datagram arrives. The receiver then fills in the **Transmit** field just before transmitting the reply.
- The original sending program can use these fields to estimate the round-trip transit-time between hosts.

Address Mask Request and Reply

- ICMP can be used by a program to read the Address Mask (the subnet mask) value from another machine. This is typical for a "diskless" workstation because it has no way of retrieving its mask.
- The ICMP address-mask reply message contains the subnet mask of for the host that sent the request.
- To use the ICMP address-mask request message a workstation can simply broadcast the query at boot time.
- One or more hosts on the network will respond with an ICMP address-mask reply message that contains the workstation's subnet mask.

Raw Sockets

- A raw socket allows us to bypass the TCP/IP transport layer and access low-level protocols such as ICMP.
- As such, your programs must perform normal transport-layer functions such as data encapsulation, for the IP layer.
- The standard Berkeley Sockets SOCK_RAW socket type, is normally used to create ping (echo request/reply), and sometimes traceroute applications (the original traceroute application from Van Jacobson used UDP, rather than ICMP).
- Microsoft's WinSock version 2 implementations support raw sockets and setsockopt(IP_TTL).
- Microsoft has their own API for an ICMP.DLL that their ping and tracert applications use (which are both non-GUI text-based console applications). Not recommended!
- This is a proprietary API, and all function calls that involve network functions operate in blocking mode.
- To create a raw socket, you must perform significantly more programming than when you create standard sockets.
- You must define your own data structures to store datagram-header information, as well as fill the structures with the correct values.
- This will require a complete understanding of the underlying protocol and its packet structure.

- First create a socket using the standard socket call:

  sd = socket(AF_INET, type, proto)

- Note that you must be root (UNIX), or Administrator (Windows) to create a raw socket.

- This is done to prevent ordinary users from writing their own IP datagrams to the network.

- The type field is set to SOCK_RAW and the proto argument is set to IPPROTO_ICMP.
- We can set the number of hops to traverse by using the IP_TTL in setsockopt. ttl is the time to live (a.k.a. number of hops) for the packet.

  setsockopt(sd, IPPROTO_IP, IP_TTL, (const char*)&ttl, sizeof(ttl)

- Most systems have a socket option IP_HDRINCL which allows you to build your own IP header along with the rest of the packet. If your system doesn't have this option, you may or may not be able to include your own IP header.

- Normally the kernel builds the IP header for datagrams sent on a raw socket, but some applications (traceroute for example) build their own IP header to override values that IP would place onto certain header fields.

- If the IP_HDRINCL option is not set, the starting address of the data for the kernel to write specifies the first byte following the IP header, because the kernel will build the IP header and prepend it to the data from the process.

- If the IP_HDRINCL option is set, the starting address of the data for the kernel to write specifies the first byte of the IP header. The amount of data to write must include the size of the caller's IP header.

- The process builds the entire IP header, except:

  (a). The IPv4 identification field can be set to 0, which tells the kernel to set this
  value.

  (b). The kernel always calculates and stores the IPv4 header checksum.

- If it is available, you can use it as follows:

  ```
  const int on = 1;
  setsockopt (sd, IPPROTO_IP, IP_HDRINCL, &on, sizeof(on));
  ```

- With IPv4 it is the responsibility of the user process to calculate and set any header checksums contained in whatever follows the IP header.

- This means that your program must implement the code to calculate the checksum for the data portion of the packet.
- You then build the packet and use the sendto() system call to transmit it.
- Datagrams are received from a raw socket using the recvfrom() system call.
- Whenever a received datagram is passed to a raw IPv4 socket, the entire datagram, including the IP header, is passed to the process.
- This means that you program must implement the code to process the header and the data.
- A very good example of an application that uses ICMP and raw sockets in the public domain ***ping*** program.
- This is a TCP/IP application that sends an "echo" packet to a specified host via the ICMP protocol. The destination responds to the packet by sending it back to the source host.
- This allows the ping program running on the source host to note when the reply arrives and to display how long it takes to send a packet.
- The operation of ***ping*** is quite simple: an ICMP echo request is sent to a specified IP address and that host responds with an ICMP echo reply.
- Your textbook shows the format of the ICMPv4 and ICMPv6 echo request and echo reply messages (Figure 25.1).
- The code examples provided show the Windows and Linux implementations of the ***ping*** program.