IP: The Internet Protocol

- Provides connectionless, best effort (unreliable) data delivery service to TCP, UDP, ICMP, and IGMP. On an exception (e.g., a router running out of buffers), IP simply throws away the datagram and sends an ICMP message to the source.

- Topics covered
  - Encapsulation
  - Routing
  - Fragmentation and Reassembly
  - Useful Commands

Encapsulation (IP Header)

<table>
<thead>
<tr>
<th>4-bit Version</th>
<th>4-bit Header Length</th>
<th>8-bit Type of Service (TOS)</th>
<th>16-bit Total Length (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16-bit Datagram ID</td>
<td>3-bit Flags</td>
<td>13-bit Fragment Offset</td>
<td></td>
</tr>
<tr>
<td>8-bit Time To Live (TTL)</td>
<td>8-bit Protocol</td>
<td>16-bit Header Checksum</td>
<td></td>
</tr>
<tr>
<td>32-bit Source IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32-bit Destination IP Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Options (if used)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data (from TCP or UDP)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Normal header size** (without options) = 20 bytes

**Version**

? 4 (for IPv4)

**Header Length**

? Number of 32-bit (4-byte) words in the header
? Normal value = 5
? Maximum header length = 60 bytes

**Type of Service (TOS)**

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>Delay</th>
<th>Throughput</th>
<th>Reliability</th>
<th>Cost</th>
<th>0</th>
</tr>
</thead>
</table>

? 3-bit precedence field
? Ignored today
? Can be used to indicate the importance of a datagram (0 for normal and 7 for network control) – can be used in **content-based routers**

? 4-bit TOS

? Minimize delay, maximize throughput, maximize reliability, and minimize monetary cost
? Only one bit can be turned on; all bits are turned off (set to 0) for normal service
? RFC340 specifies how these bits should be set by the standard applications

<table>
<thead>
<tr>
<th>Application</th>
<th>8-bit TOS Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TELNET/RLOGIN</td>
<td>0x10</td>
</tr>
<tr>
<td>FTP</td>
<td>0x10 (control)</td>
</tr>
<tr>
<td></td>
<td>0x08 (data)</td>
</tr>
<tr>
<td>SMTP</td>
<td>0x10 (command)</td>
</tr>
<tr>
<td></td>
<td>0x08 (data)</td>
</tr>
<tr>
<td>DNS</td>
<td>0x10 (UDP query)</td>
</tr>
<tr>
<td></td>
<td>0x00 (TCP query)</td>
</tr>
<tr>
<td>ICMP</td>
<td>0x00</td>
</tr>
<tr>
<td>SNMP</td>
<td>0x04</td>
</tr>
<tr>
<td>BOOTP</td>
<td>0x00</td>
</tr>
</tbody>
</table>

? 1-bit Unused and always set to 0
**Total Length**

- Total length of the IP datagram—needed since an IP datagram may be less than the minimum size needed by a data link layer (e.g., 46 bytes for Ethernet). By using this, IP can figure out how many bytes are IP datagram and how many are *pad* bytes.

- By using this field’s value we find out where the IP data starts

- Maximum datagram size = $2^{16}$ bytes = 65,535 bytes. Most link layer protocols require *fragmentation* of such large datagrams at a router. Fragments are *reassembled* at the destination host (see Fragmentation and Reassembly below).

- Most implementations (especially those that use Network File System, NFS), use 8K bytes (8,192 bytes) datagrams.

**Identification**

- 16-bit integer

- Unique identification of a datagram (or a fragment, which is a datagram itself)

**Flags**

- 3 bits

- One bit is set to tell routers to not fragment (at bootup time, some machines need complete boot images). Routers discard datagrams with this bit set if they need to fragment them and an ICMP message is sent to the sender

- One bit indicates more fragments (yes or no). This bit is set except for the last fragment of a datagram.

**Fragment Offset**

- Offset of the fragment from the beginning of the original datagram.

- 16-bit chunks are added by using 1’s complement arithmetic and 1’s complement of the resultant number is stored as checksum

- Only the data field is fragmented. Each fragment gets its own header and is routed independently of other datagrams.

- The header for a fragment is mostly a copy of the header of the original datagram but has new values for some fields such as total length, “more fragments bit”, and fragment offset.
**Time To Live (TTL)**

- Set by the sender host, it puts an upper limit on the number of routers this datagram can pass through; typical value is 32 or 64
- Decremented by one by every router; datagram thrown away when TTL reaches a value of 0 and an **ICMP** message is sent to the sender
- Prevents datagrams to be caught up in routing loops (*ping ponging*)

**Protocol**

- Used by IP to demultiplex incoming data (1 for **ICMP**, 2 for **IGMP**, 6 for **TCP**, and 17 for **UDP**)

**Header Checksum**

- For IP header only
- **ICMP**, **IGMP**, **UDP**, and **TCP** all use their own checksums to cover their header and data
- First set to 0. Then 16-bit chunks of the header are added by using 1’s complement arithmetic and 1’s complement of the resultant sum is stored in this field.
- On the receiver side, if checksum of the received header is computed and is all 1’s if nothing in the header was modified. Otherwise, there is a checksum error and the datagram is discarded. No error message is sent to the sender side—upper layers handle missing datagrams.
- After decrementing the TTL field, the router recomputes checksum and forwards the datagram to the next hop router.

**Source and Destination IP Addresses**

- 32-bit in IPv4

**Options**

- Security restrictions
- Record route: have each router record its IP address
- Time stamp: have each router record its local time
- Always ends on a 32-bit boundary; padded (with null bytes) if necessary
Fragmentation and Reassembly of IP Datagrams

Fragmentation Performed by IP at a Router

Fragmentation is performed by a router when it receives an IP datagram whose size is greater than the MTU of the underlying link layer protocol. IP queries the link layer to get this information.

Each fragment becomes a new datagram, has its own header, and is routed independently of other datagrams. The header contains the ID of the original datagram. The “more fragments” bit is turned on for all, except the last fragment of a datagram. Fragment offset field contains the offset of this fragment’s data with respect to the beginning of the data of the original datagram. The total length field is also changed to indicate the length of the datagram that contains the fragment.

Fragments can be fragmented as well, possibly more than once.

Fragmentation is transparent to TCP and UDP but there is a little performance degradation.

The term datagram is used for the unit of data the flows between end-to-end IP layers. The term packet is used for the unit of data passed between the IP layer and link layer.

Re-assembly is performed by the destination IP. The receiver IP starts a re-assembly timer when it receives a fragment. If a fragment is lost, the entire original datagram is discarded and no error message is sent to the sender IP.

Re-assembly at the destination host reduces the cost of state information maintained by routers and allows routes to change dynamically, i.e., multiple routers may get fragments. Also, if routers do re-assembly, they need to send messages to the sender host – too much overhead for routers!
Example

- Data size in the IP datagram = 272 bytes
- Link layer MTU = 148 bytes
- Maximum packet size for the link layer = 148 bytes

= 20-byte header + 128-byte data

272-byte Data

IP Header

128-byte Data
Offset = 0

IP Header

128-byte Data
Offset = 128

IP Header

16-byte Data
Offset = 256
ICMP: Internet Control Message Protocol

Originally designed so that a source (sender host) could find out why a datagram could not be delivered.

IP provides “best effort” delivery service but it does not mean that it is a “careless” service--it attempts to prevent errors and reports when they do occur.

IP uses ICMP to report error messages and ICMP used IP to transport error messages ... they are co-dependent.

ICMP defines

- 5 error messages
  - Source Quench  Sent by a router to the source of a datagram which is discarded as the router runs out of buffer space.
  - Time Exceeded  TTL expired.
  - Destination Unreachable  When either the next hop router is down or the destination host is unreachable (e.g., bad host ID or host is down).
  - Redirect  When a router determines that the sender has sent datagram to the router by mistake, telling the host which router to send the datagram to. This situation usually occurs due to a change in the network routing table.
  - Fragmentation Required  No fragmentation bit was set by the host but a router determines that the link layer for the next network has a smaller MTU than the size of the IP datagram.

- 4 information messages
  - Echo Request/Reply  ICMP to ICMP echo request/reply.
  - Address Mask Request/Reply  Sent by a host at boot time; routers that receive this message respond with the correct 32-bit subnet mask for the network, e.g., 255.255.255.0 for a class C network.

Encapsulation

<table>
<thead>
<tr>
<th>ICMP Header</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Header</td>
<td>ICMP Header</td>
</tr>
</tbody>
</table>
Two Commonly Used Tools That Use ICMP

? **Ping**
  ? Uses ICMP **echo** message

? **Traceroute**
  ? Uses ICMP **time exceeded** messages to determine a list of the routers along a path to a given destination
  ? 1st datagram has a TTL = 1
  ? 2nd datagram has TTL = 2
  ...
  ? Allows a user to specify how long to wait before assuming lost datagram / bad checksum error
  ? Uses UDP datagrams to non-existing applications
  ? Either receives TTL expired message or non-existing application (unreachable destination) message from the ultimate destination.

? **Path MTU discovery**
  ? Datagram with don’t fragment bit set
  ? ICMP fragmentation required message send back if fragmentation is required.
  ? Host software (or hardware) sends a sequence of probes to determine the path MTU. This is done by IPv6 sender host in order to perform fragmentation. (Fragmentation and reassembly in IPv6 is different than in IPv4.)