IP Datagram General Format

Data transmitted over an internet using IP is carried in messages called *IP datagrams*. Like all network protocol messages, IP uses a specific format for its datagrams. We are of course looking here at IP version 4 and so we will examine the IPv4 datagram format, which was defined in RFC 791 along with the rest of IPv4.

The IPv4 datagram is conceptually divided into two pieces: the *header* and the *payload*. The header contains addressing and control fields, while the payload carries the actual data to be sent over the internetwork. Unlike some message formats, IP datagrams do not have a footer following the payload.

Even though IP is a relatively simple, connectionless, "unreliable" protocol, the IPv4 header carries a fair bit of information, which makes it rather large. At a minimum, it is 20 bytes long, and with options can be significantly longer. The IP datagram format is described in Table 57 and illustrated in Figure 86.

Table 57: Internet Protocol Version 4 (IPv4) Datagram Format *(Page 1 of 4)*

Field Name	Size (bytes)	Description
Version	1/2 (4 bits)	***Version:*** Identifies the version of IP used to generate the datagram. For IPv4, this is of course the number 4. The purpose of this field is to ensure compatibility between devices that may be running different versions of IP. In general, a device running an older version of IP will reject datagrams created by newer implementations, under the assumption that the older version may not be able to interpret the newer datagram correctly.
IHL	1/2 (4 bits)	***Internet Header Length (IHL):*** Specifies the length of the IP header, in 32-bit words. This includes the length of any options fields and padding. The normal value of this field when no options are used is 5 (5 32-bit words = 5*4 = 20 bytes). Contrast to the longer *Total Length* field below.
TOS	1	***Type Of Service (TOS):*** A field designed to carry information to provide quality of service features, such as prioritized delivery, for IP datagrams. It was never widely used as originally defined, and its meaning has been subsequently redefined for use by a technique called *Differentiated Services (DS)*. See below for more information.
TL	2	***Total Length (TL):*** Specifies the total length of the IP datagram, in bytes. Since this field is 16 bits wide, the maximum length of an IP datagram is 65,535 bytes, though most are much smaller.
Identification	2	***Identification:*** This field contains a 16-bit value that is common to each of the fragments belonging to a particular message; for datagrams originally sent unfragmented it is still filled in, so it can be used if the datagram must be fragmented by a router during delivery. This field is used by the recipient to reassemble messages without accidentally mixing fragments from different messages. This is needed because fragments may arrive from multiple messages mixed together, since IP datagrams can be received out of order from any device. See the discussion of IP message fragmentation.

Table 57: Internet Protocol Version 4 (IPv4) Datagram Format *(Page 2 of 4)*

<table>
<tr><th>Field Name</th><th>Size (bytes)</th><th>Description</th></tr>
<tr><td>Flags</td><td>3/8
(3 bits)</td><td>Flags: Three control flags, two of which are used to manage fragmentation (as described in the topic on fragmentation), and one that is reserved:
<table>
<tr><th>Subfield Name</th><th>Size (bytes)</th><th>Description</th></tr>
<tr><td>Reserved</td><td>1/8 (1 bit)</td><td>Reserved: Not used.</td></tr>
<tr><td>DF</td><td>1/8 (1 bit)</td><td>Don't Fragment: When set to 1, specifies that the datagram should not be fragmented. Since the fragmentation process is generally “invisible” to higher layers, most protocols don't care about this and don't set this flag. It is, however, used for testing the maximum transmission unit (MTU) of a link.</td></tr>
<tr><td>MF</td><td>1/8 (1 bit)</td><td>More Fragments: When set to 0, indicates the last fragment in a message; when set to 1, indicates that more fragments are yet to come in the fragmented message. If no fragmentation is used for a message, then of course there is only one “fragment” (the whole message), and this flag is 0. If fragmentation is used, all fragments but the last set this flag to 1 so the recipient knows when all fragments have been sent.</td></tr>
</table></td></tr>
<tr><td>Fragment Offset</td><td>1 5/8
(13 bits)</td><td>Fragment Offset: When fragmentation of a message occurs, this field specifies the offset, or position, in the overall message where the data in this fragment goes. It is specified in units of 8 bytes (64 bits). The first fragment has an offset of 0. Again, see the discussion of fragmentation for a description of how the field is used.</td></tr>
<tr><td>TTL</td><td>1</td><td>Time To Live (TTL): Short version: Specifies how long the datagram is allowed to “live” on the network, in terms of router hops. Each router decrements the value of the TTL field (reduces it by one) prior to transmitting it. If the TTL field drops to zero, the datagram is assumed to have taken too long a route and is discarded.

See below for the longer explanation of TTL.</td></tr>
</table>

Table 57: Internet Protocol Version 4 (IPv4) Datagram Format *(Page 3 of 4)*

<table>
<tr><th>Field Name</th><th>Size (bytes)</th><th>Description</th></tr>
<tr><td>Protocol</td><td>1</td><td>Protocol: Identifies the higher-layer protocol (generally either a transport layer protocol or encapsulated network layer protocol) carried in the datagram. The values of this field were originally defined by the IETF “Assigned Numbers” standard, RFC 1700, and are now maintained by the Internet Assigned Numbers Authority (IANA):
<table>
<tr><th>Value (Hexadecimal)</th><th>Value (Decimal)</th><th>Protocol</th></tr>
<tr><td>00</td><td>0</td><td>Reserved</td></tr>
<tr><td>01</td><td>1</td><td>ICMP</td></tr>
<tr><td>02</td><td>2</td><td>IGMP</td></tr>
<tr><td>03</td><td>3</td><td>GGP</td></tr>
<tr><td>04</td><td>4</td><td>IP-in-IP Encapsulation</td></tr>
<tr><td>06</td><td>6</td><td>TCP</td></tr>
<tr><td>08</td><td>8</td><td>EGP</td></tr>
<tr><td>11</td><td>17</td><td>UDP</td></tr>
<tr><td>32</td><td>50</td><td>Encapsulating Security Payload (ESP) Extension Header</td></tr>
<tr><td>33</td><td>51</td><td>Authentication Header (AH) Extension Header</td></tr>
</table>
Note that the last two entries are used when IPSec inserts additional headers into the datagram: the AH or ESP headers.</td></tr>
<tr><td>Header Checksum</td><td>2</td><td>Header Checksum: A checksum computed over the header to provide basic protection against corruption in transmission. This is not the more complex CRC code typically used by data link layer technologies such as Ethernet; it's just a 16-bit checksum. It is calculated by dividing the header bytes into words (a word is two bytes) and then adding them together. The data is not checksummed, only the header. At each hop the device receiving the datagram does the same checksum calculation and on a mismatch, discards the datagram as damaged.</td></tr>
<tr><td>Source Address</td><td>4</td><td>Source Address: The 32-bit IP address of the originator of the datagram. Note that even though intermediate devices such as routers may handle the datagram, they do not normally put their address into this field—it is always the device that originally sent the datagram.</td></tr>
<tr><td>Destination Address</td><td>4</td><td>Destination Address: The 32-bit IP address of the intended recipient of the datagram. Again, even though devices such as routers may be the intermediate targets of the datagram, this field is always for the ultimate destination.</td></tr>
<tr><td>Options</td><td>Variable</td><td>Options: One or more of several types of options may be included after the standard headers in certain IP datagrams. I discuss them in the topic that follows this one.</td></tr>
</table>

Table 57: Internet Protocol Version 4 (IPv4) Datagram Format *(Page 4 of 4)*

Field Name	Size (bytes)	Description
Padding	Variable	***Padding:*** If one or more options are included, and the number of bits used for them is not a multiple of 32, enough zero bits are added to "pad out" the header to a multiple of 32 bits (4 bytes).
Data	Variable	***Data:*** The data to be transmitted in the datagram, either an entire higher-layer message or a fragment of one.

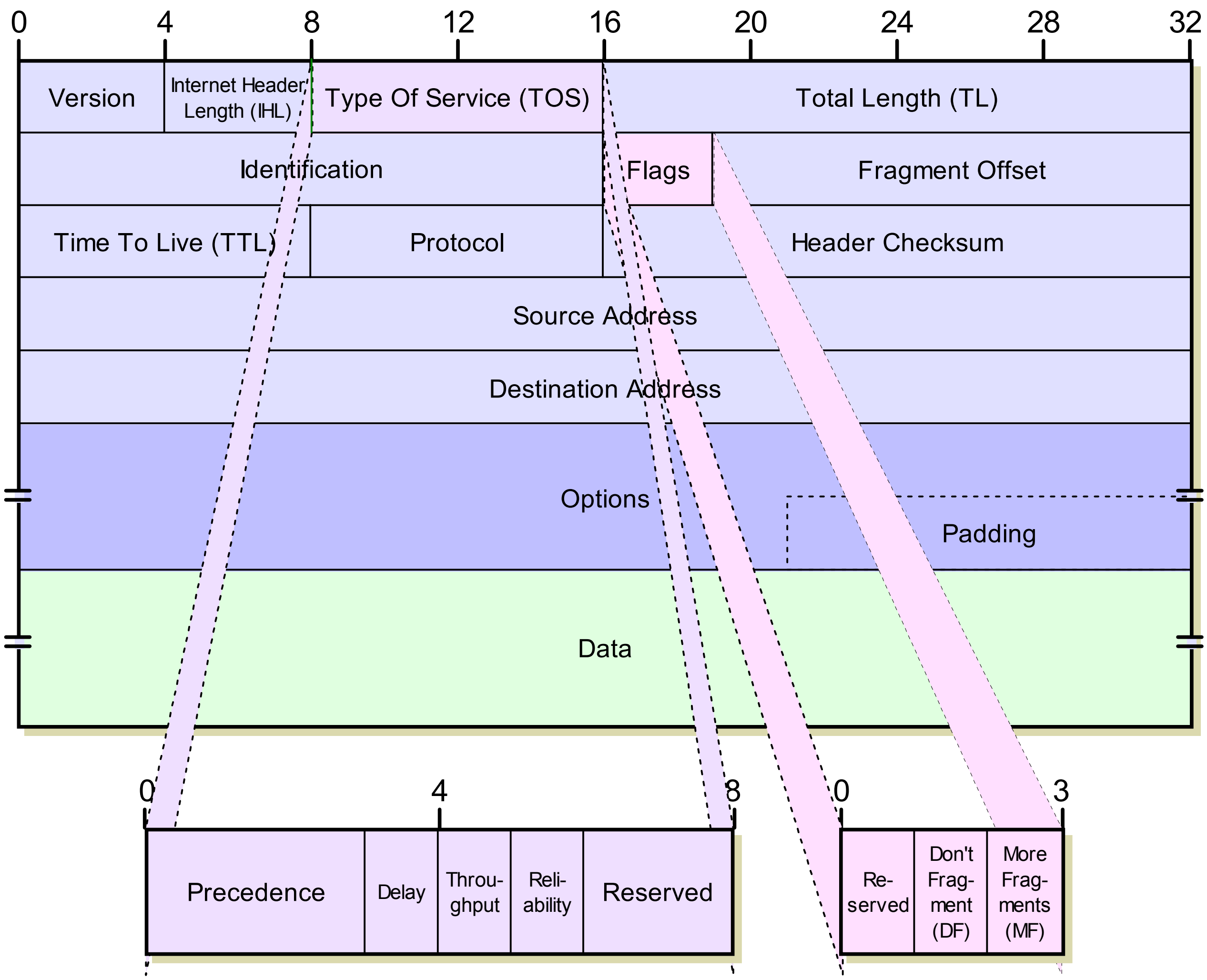

Figure 86: Internet Protocol Version 4 (IPv4) Datagram Format

This diagram shows graphically the all-important IPv4 datagram format. The first 20 bytes are the fixed IP header, followed by an optional *Options* section, and a variable-length *Data* area. Note that the *Type Of Service* field is shown as originally defined in the IPv4 standard.

That's a pretty big table, because the IP datagram format is pretty important and has a lot of fields that need explaining. To keep it from being even longer, I decided to move a couple of the more complex descriptions out of the table.

Time To Live (TTL) Field

Since IP datagrams are sent from router to router as they travel across an internetwork, it is possible that a situation could result where a datagram gets passed from router A to router B to router C and then back to router A. Router loops are not supposed to happen, and rarely do, but are possible.

To ensure that datagrams don't circle around endlessly, the *TTL* field was intended to be filled in with a time value (in seconds) when a datagram was originally sent. Routers would decrease the time value periodically, and if it ever hit zero, the datagram would be destroyed. This was also intended to be used to ensure that time-critical datagrams wouldn't linger past the point where they would be "stale".

In practice, this field is not used in exactly this manner. Routers today are fast and usually take far less than a second to forward a datagram; measuring the time that a datagram "lives" would be impractical. Instead, this field is used as a "maximum hop count" for the datagram. Each time a router processes a datagram, it reduces the value of the *TTL* field by one. If doing this results in the field being zero, the datagram is said to have expired. It is dropped, and usually an ICMP *Time Exceeded* message is sent to inform the originator of the message that this happened.

The *TTL* field is one of the primary mechanisms by which networks are protected from router loops (see the description of ICMP Time Exceeded messages for more on how *TTL* helps IP handle router loops.)

Type Of Service (TOS) Field

This one-byte field was originally intended to provide certain quality of service features for IP datagram delivery. It allowed IP datagrams to be tagged with information indicating not only their precedence, but the preferred manner in which they should be delivered. It was divided into a number of subfields, as shown in Table 58 (and Figure 86).

The lack of quality of service features has been considered a weakness of IP for a long time. But as we can see in Table 58, these features were built into IP from the start. What's going on here? The answer is that even though this field was defined in the standard back in the early 1980s, it was not widely used by hardware and software. For years, it was just passed around with all zeroes in the bits and mostly ignored.

Table 58: Original Definition Of IPv4 *Type Of Service (TOS)* Field

<table>
<tr><th>Subfield Name</th><th>Size (bytes)</th><th>Description</th></tr>
<tr><td>Precedence</td><td>3/8
(3 bits)</td><td>Precedence: A field indicating the priority of the datagram. There were eight defined values, from lowest to highest priority:
<table>
<tr><th>Precedence Value</th><th>Priority Level</th></tr>
<tr><td>000</td><td>Routine</td></tr>
<tr><td>001</td><td>Priority</td></tr>
<tr><td>010</td><td>Immediate</td></tr>
<tr><td>011</td><td>Flash</td></tr>
<tr><td>100</td><td>Flash Override</td></tr>
<tr><td>101</td><td>CRITIC/ECP</td></tr>
<tr><td>110</td><td>Internetwork Control</td></tr>
<tr><td>111</td><td>Network Control</td></tr>
</table></td></tr>
<tr><td>D</td><td>1/8
(1 bit)</td><td>Delay: Set to 0 to request “normal” delay in delivery; set to 1 if low delay delivery is requested.</td></tr>
<tr><td>T</td><td>1/8
(1 bit)</td><td>Throughput: Set to 0 to request “normal” delivery throughput; set to 1 if higher throughput delivery is requested.</td></tr>
<tr><td>R</td><td>1/8
(1 bit)</td><td>Reliability: Set to 0 to request “normal” reliability in delivery; set to 1 if higher reliability delivery is requested.</td></tr>
<tr><td>Reserved</td><td>2/8
(2 bits)</td><td>Reserved: Not used.</td></tr>
</table>

The IETF, seeing the field unused, attempted to revive its use. In 1998, RFC 2474 redefines the first six bits of the *TOS* field to support a technique called *Differentiated Services (DS)*. Under DS, the values in the *TOS* field are called *codepoints* and are associated with different service levels. This starts to get rather complicated, so refer to RFC 2474 if you want all the details.

Understanding the IP datagram format is an important part of troubleshooting IP networks. Be sure to see the following topic on options for more information on how IP options are used in datagrams, and the topic on fragmenting for some more context on the use of fragmentation-related fields such as *Identification*, *Fragment Offset*, and *More Fragments*.

