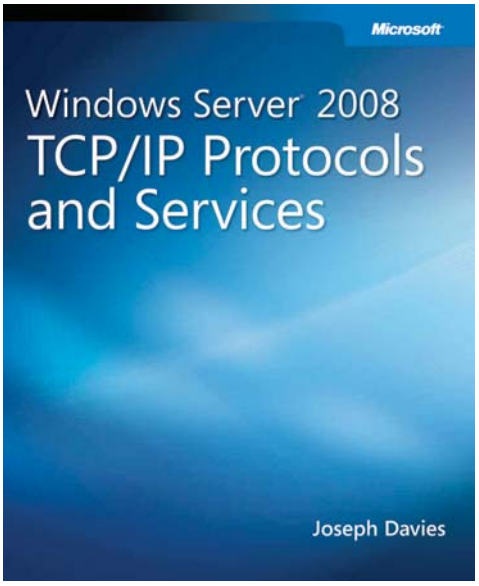


Windows Server® 2008 TCP/IP Protocols and Services

Joseph Davies

To learn more about this book, visit Microsoft Learning at
http://www.microsoft.com/MSPress/books/11630.aspx

Microsoft® Press

978-0-7356-2447-4

Table of Contents

Chapter 5
Internet Protocol (IP)

IP is the internetworking building block of all the other protocols at the Internet Layer and above. IP is a datagram protocol primarily responsible for addressing and routing packets between hosts. This chapter describes the details of the fields in the IP header and their role in IP packet delivery.

> **Note** This chapter uses the term *IP* to refer to version 4 of IP (IPv4), which is in widespread use today. IP version 6 is denoted as IPv6.

Introduction to IP

IP is the primary protocol for the Internet Layer of the Department of Defense (DoD) Advanced Research Projects Agency (DARPA) model and provides the internetworking functionality that makes large-scale internetworks such as the Internet possible. IP has lasted since it was formalized in 1981 with RFC 791, and will continue to be used on the Internet for years to come. Only relatively recently have IP's shortcomings been addressed in a new version known as IPv6. For more information about IPv6, see Chapter 8, "Internet Protocol Version 6 (IPv6)." IP's amazing longevity is a tribute to its original design.

> **MoreInfo** All of the RFCs referenced in this chapter can be found in the \Standards\Chap05_IP folder on the companion CD-ROM.

IP Services

IP offers the following services to upper layer protocols:

Internetworking protocol

- IP is an internetworking protocol, also known as a routable protocol. The IP header contains information necessary for routing the packet, including source and destination IP addresses. An IP address is composed of two components: a network address and a node address. Internetwork delivery, or routing, is possible because of the existence of a destination network address. IP allows the creation of an IP internetwork, which consists of two or more networks interconnected by IP router(s). The IP header also contains a link count, which is used to limit the number of links on which the packet can travel before being discarded.

Multiple client protocols

- IP is an internetwork carrier for upper layer protocols. IP can carry several different upper layer protocols, but each IP packet can contain data from only one upper layer protocol at a time. Because each packet can carry one of several protocols, there must be a way to indicate the upper layer protocol of the packet payload so that it can be forwarded to the appropriate upper layer protocol at the destination. Both the client and the server always use the same protocol for a given exchange of data. Therefore, the packet does not need to indicate separate source and destination protocols.

Examples of upper layer protocols include other Internet Layer protocols such as Internet Control Message Protocol (ICMP) and Internet Group Management Protocol (IGMP) and Transport Layer protocols such as Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

Datagram delivery

- IP is a datagram protocol that provides a connectionless, unreliable delivery service for upper layer protocols. Connectionless means that no handshaking occurs between IP nodes prior to sending data, and no logical connection is created or maintained at the Internet Layer. Unreliable means that IP sends a packet without sequencing and without an acknowledgment that the destination was reached. IP makes a best effort to deliver packets to the next hop or the final destination. End-to-end reliability is the responsibility of upper layer protocols such as TCP.

Independence from Network Interface Layer

- At the Internet Layer, IP is designed to be independent of the network technology present at the Network Interface Layer of the DARPA model, which encompasses the Open Systems Interconnection (OSI) Physical and Data Link Layers. IP is independent of OSI Physical Layer attributes such as cabling, signaling, and bit rate. It also is independent of OSI Data Link Layer attributes such as media access control (MAC) scheme, addressing, and maximum frame size. IP uses a 32-bit address that is independent of the addressing scheme used at the Network Interface Layer.

Fragmentation and reassembly

- To support the maximum frame sizes of different Network Interface Layer technologies, IP allows for the fragmentation of a payload when forwarding onto a link that has a lower maximum transmission unit (MTU) than the IP datagram size. Routers or sending hosts fragment an IP payload, and fragmentation can occur multiple times. The destination host then reassembles the fragments into the originally sent IP payload. More information on fragmentation and reassembly are provided later in this chapter in the section entitled "Fragmentation."

Extensible through IP options

- When features are required that are not available using the standard IP header, IP options can be used. IP options are appended to the standard IP header and provide custom functionality, such as the ability to specify a path that an IP datagram follows through the IP internetwork.

Datagram packet-switching technology

- IP is an example of a datagram packet-switching technology: Each packet is a datagram, an unacknowledged and nonsequenced message that is forwarded by the switches of the switching network using a globally significant address. In the case of IP, each switch in the switching network is an IP router, and the globally significant address is the destination IP address. This address is examined at each router, which makes an independent routing decision and forwards the packet. Because each router decides independently where to forward a packet, a packet's path from Node 1 to Node 2 is not necessarily a packet's path from Node 2 to Node 1. Because each packet is separately switched, each can take a different path between the source and destination. Because of various transit delays, each packet can arrive in a different order from which it was sent. Additionally, packets can be duplicated by intermediate routers.

Note The term *switch* is used here for a generalized forwarding device and is not meant to imply a Layer 2 switch. A Layer 2 switch is typically used in Ethernet environments to segment traffic.

IP MTU

Each Network Interface Layer technology imposes a maximum-sized frame that can be sent. This frame typically consists of the framing header and trailer and a payload. The maximum size of a frame for a given Network Interface Layer technology is called the MTU. For an IP packet, the Network Interface Layer payload is an IP datagram. Therefore, the maximum-sized payload becomes the maximum-sized IP datagram. This is known as the IP MTU.

Table 5-1 lists the IP MTUs for the various Network Interface Layer technologies that are described in Chapter 1, "Local Area Network (LAN) Technologies," and Chapter 2, "Wide Area Network (WAN) Technologies."

Table 5-1. IP MTUs for Common Network Interface Layer Technologies

Network Interface Layer Technology	IP MTU
Ethernet (Ethernet II encapsulation)	1500
Ethernet (IEEE 802.3 Sub-Network Access Protocol [SNAP] encapsulation)	1492
Token Ring (4 and 16 Mbps)	Varies based on token holding time
Fiber Distributed Data Interface (FDDI)	4352
X.25	1600
Frame relay	1592 (with a 2-byte Address field in the Frame Relay header)
Asynchronous Transfer Mode (ATM; Classical IP over ATM)	9180

In an environment with mixed Network Interface Layer protocols, fragmentation can occur when crossing a router from a link with a higher IP MTU to a link with a lower IP MTU. IP fragmentation is discussed in more detail later in this chapter in the section entitled "Fragmentation."

In Windows Server 2008 and Windows Vista, it is possible to override the MTU as reported to the Network Driver Interface Specification (NDIS) interface by the network adapter driver with the following command:

netsh interface ipv4 set interface *InterfaceNameOrIndex* **mtu=***MtuSize*

InterfaceNameOrIndex is the name of the interface from the Network Connections folder or its interface index. *MtuSize* is the IP MTU.

You can also use the following registry value:

MTU

```
Key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\
Parameters\Interfaces\InterfaceGUID
Data type: REG_DWORD
Valid range: 576 - <the MTU reported by the network adapter>
Default: 0xFFFFFFFF (the MTU reported by the network adapter)
Present by default: No
```

When TCP/IP initializes, it queries its bound NDIS network adapter driver and receives the MTU. The MTU registry value is used to set an MTU that is lower than the default MTU, as reported by the NDIS driver, and greater than the minimum value of 576. Values in the MTU registry value that are greater than the default MTU are ignored. If the MTU registry value is set to a value less than 576, 576 is used.

It is useful to change the default MTU size for testing or for solving MTU issues in translational bridge environments.

The IP Datagram

Figure 5-1 shows the structure of an IP datagram.

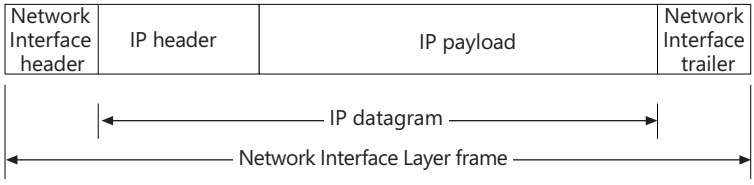


Figure 5-1 The structure of the IP datagram at the Network Interface layer.

The IP datagram consists of the following:

IP header

- The IP header is of variable size, between 20 and 60 bytes, in 4-byte increments. It provides routing support, payload identification, IP header and datagram size indication, fragmentation support, and options.

IP payload

- The IP payload is of variable size, ranging from 0 bytes (a 20-byte IP datagram with a 20-byte IP header) to 65,515 bytes (a 65,535-byte IP datagram with a 20-byte header).

As sent on a link, the IP datagram is wrapped with a Network Interface Layer header and trailer to create a Network Interface Layer frame.

The IP Header

Figure 5-2 shows the IP header's structure. The following sections discuss the fields of the IP header.

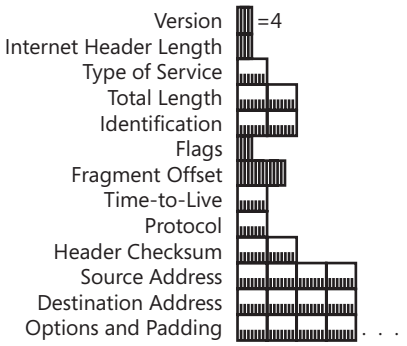


Figure 5-2 The structure of the IP header.

Version

The Version field is 4 bits long and is used to indicate the IP header version. A 4-bit field can have values from 0 through 15. The most prevalent IP version used today on organization intranets and the Internet is version 4, sometimes referred to as IPv4. The next version of IP is IPv6. All other values for the Version field are either undefined or not in use. For the latest list of the defined values of the IP Version field, see *http://www.iana.org/assignments/version-numbers*.

Header Length

The Header Length field is 4 bits long and is used to indicate the IP header size. The maximum number that can be represented with 4 bits is 15. Therefore, the Header Length field cannot possibly be a byte counter. Rather, the Header Length field indicates the number of 32-bit words (4-byte blocks) in the IP header. The typical IP header does not contain any options and is 20 bytes long. The smallest possible Header Length value is 5 (0x5). With the maximum amount of IP options, the largest IP header can be 60 bytes long, indicated with a Header Length value of 15 (0xF).

Using a 4-byte block counter to indicate the IP header size means that the IP header size must always be a multiple of 4. If a set of IP options extend the IP header, they must do so in 4-byte increments. If the set of IP options is not a multiple of 4 bytes long, option padding bytes must be used so that the IP header is always on a 4-byte boundary.

Type Of Service

The Type Of Service (TOS) field is 8 bits long and is used to indicate the quality of service with which this datagram is to be delivered by the internetwork routers. The TOS field has two definitions: the original RFC 791 definition and the newer definition based on RFCs 2474 and 3168. The RFC 791 definition has been deprecated by RFCs 2474 and 3168.

RFC 791 Definition of the TOS Field

As defined in RFC 791, the TOS field contains subfields and flags to indicate desired precedence, delay, throughput, reliability, and cost characteristics.

Within the 8 bits of the TOS field, there are five fields that indicate a different quality of the datagram delivery, as shown in Figure 5-3. The TOS field is set by the sending host and is not modified by routers. All IP fragments contain the same TOS setting as the original IP datagram.

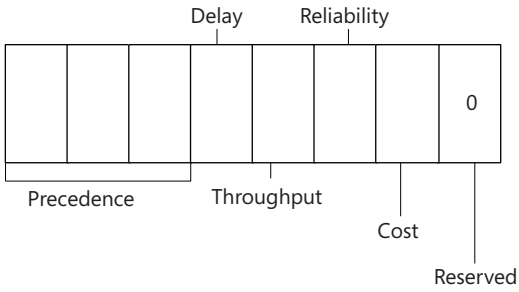


Figure 5-3 The structure of the RFC 791 IP Type Of Service field.

Normally, a sending host sends an IP datagram with the TOS field set to the value of 0x00: routine precedence, normal delay, normal throughput, normal reliability, and normal cost. Routers normally ignore the values in the TOS field and forward all datagrams as if the fields are not set. This is known as TOS0 routing. However, modern routing protocols such as Open Shortest Path First (OSPF) and Integrated Intermediate System-Intermediate System (IS-IS) now support the calculation of routes for each value of the TOS field.

The routers and the routing protocol determine how the various values in the TOS field are interpreted. In a properly configured network, packets with specific TOS values are forwarded over different paths. This can improve routing and delivery efficiency in a multipath IP internetwork. For example, an IP internetwork could have one path for general traffic, one for low-delay traffic, and another path for high-reliability traffic. When sending hosts set various combinations of TOS values, routers can choose among those paths. The TOS field is used for prioritized delivery, sometimes referred to as quality of service (QoS), in IP internetworks.

Precedence

The Precedence field is 3 bits long and is used to indicate the importance of the datagram. Table 5-2 lists the defined values of the Precedence field.

Table 5-2. Values of the IP Precedence Field

Precedence Value	Precedence
000	Routine
001	Priority
010	Immediate
011	Flash
100	Flash Override
101	CRITIC/ECP
110	Internetwork Control
111	Network Control

The Precedence field is set to 000 (Routine) by default.

Delay

The Delay field is a flag indicating either Normal Delay (when set to 0) or Low Delay (when set to 1). If Delay is set to 1, the IP router forwards the IP datagram along the path that has the lowest delay characteristics. An application can request the low delay path when sending either time-sensitive data, such as digitized voice or video, or interactive traffic, such as Telnet sessions. Based on the Delay flag, the router might choose the lower delay terrestrial wide area network (WAN) link over the higher delay satellite link, even if the satellite link has a higher bandwidth.

Throughput

The Throughput field is a flag indicating either Normal Throughput (when set to 0) or High Throughput (when set to 1). If the Throughput field is set to 1, the IP router forwards the IP datagram along the path that has the highest throughput characteristics. An application can request the high throughput path when sending bulk data. Based on the Throughput flag, the router can choose the higher throughput satellite link over the lower throughput terrestrial WAN link, even if the terrestrial link has a lower delay.

Reliability

The Reliability field is a flag indicating either Normal Reliability (when set to 0) or High Reliability (when set to 1). During periods of congestion at an IP router, the Reliability field is used to decide which IP datagrams to discard first. If the Reliability field is set to 1, the IP router discards these datagrams last. An application can request the high reliability path when sending time-sensitive data, so that it cannot be discarded. For example, with some methods of sending digital video, the digitized video is sent as two types of packets: The primary type is used to reconstruct the basic video image, and a secondary type is used to provide a higher resolution image. In this case, the primary packets are sent with the Reliability field set to 1 and the secondary packets are sent with the Reliability field set to 0. If congestion occurs at the router, the router discards the secondary packets first.

Cost

The Cost field is a flag indicating either Normal Cost (when set to 0) or Low Cost (when set to 1), where cost indicates monetary cost. If the Cost field is set to 1, the IP router forwards the IP datagram along the path that has the lowest cost characteristics. An application can request the low cost path when sending noncritical data. Based on the Cost flag, the

router can choose a lower cost terrestrial link over a higher cost satellite link, even if the terrestrial link has a lower bandwidth.

Reserved

The Reserved field is the last bit and must be set to 0. Routers ignore this field when forwarding IP datagrams.

RFC 2474 Definition of the TOS Field

To accommodate prioritized delivery of IP packets over an IP internetwork, RFC 2474 redefines the 8 bits in the TOS field in terms of a 6-bit Differentiated Services Code Point (DSCP) field and 2 unused bits. The DSCP value identifies the per-hop behavior that the receiving routers use to determine the special delivery handling for the packet. DSCP values are defined by network policy.

The RFC 2474–defined TOS field is shown in Figure 5-4.

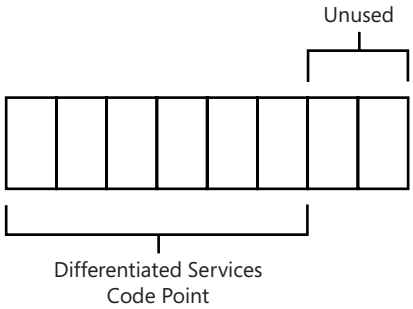


Figure 5-4 The structure of the RFC 2474 IP TOS field.

Differentiated services are an alternative to prioritized delivery mechanisms that use the Resource ReSerVation Protocol (RSVP). RSVP requires that communicating nodes use an initial signaling process and that intermediate routers maintain a flow state. With differentiated services, network policy determines the DSCP values and their corresponding delivery and queuing parameters. The network policy is propagated to both the routers and the communicating hosts. When a host needs prioritized delivery for a packet, it selects the appropriate DSCP value and places it in the TOS field in the IP header. The intermediate routers note the DSCP value and provide the corresponding prioritized delivery service.

TCP/IP for Windows Server 2008 and Windows Vista uses the RFC 2474 definition of the TOS field by default. Because the IP_TOS Winsock option has been removed, you can set its value with the QoS components of Windows Server 2008 and Windows Vista. You can use Group Policy-based QoS settings to set DSCP values and control application sending rates without having to use application programming interfaces (APIs) or modify existing applications. You can use the Generic QoS (GQoS) and Traffic Control (TC) APIs to set the DSCP value or the new QoS2 API, also known as Quality Windows Audio-Video Experience (qWAVE).

> **Note** IP for Windows Server 2008 and Windows Vista does not support the DisableUserTOSSetting registry value.

Explicit Congestion Notification and the TOS Field

To prevent the problems associated with dropped packets due to congested routers, the designers of TCP/IP created a new set of standards for both hosts and routers. These standards describe active queue management (AQM) on IP routers (RFC 2309) to allow the router to monitor that state of its forwarding queues and provide a mechanism to enable routers to report to sending hosts that congestion is occurring, allowing the sending hosts to lower their transmission rate before the router begins dropping packets. The router reporting and host response mechanism is known as Explicit Congestion Notification (ECN) and is defined in RFC 3168.

ECN support in IP uses the two unused bits of the RFC 2474-defined TOS field. Figure 5-5 shows the new definition of the TOS field with ECN.

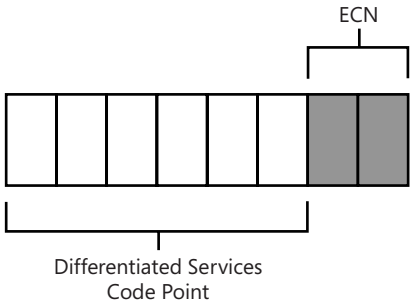


Figure 5-5 The structure of the RFC 3168 IP TOS field.

The two unused bits in the RFC 2474-defined TOS field are defined in RFC 3168 as the ECN field, which has the following values:

00

- The sending host does not support ECN.

01 or 10

- The sending host supports ECN.

11

- Congestion has been experienced by a router.

An ECN-capable host sends its packets with the ECN field set to 01 or 10. For packets sent by ECN-capable hosts, if a router in the path is ECN-capable and is experiencing congestion, it sets the ECN field to 11. If the ECN field has been set to 11, downstream routers in the path to the destination do not modify its value.

TCP/IP in Windows Server 2008 and Window Vista supports ECN but it is disabled by default. To enable ECN support, use the **netsh interface tcp set global ecncapability=enabled** command. Because ECN is using bits in the IP and TCP headers that were previously defined as unused or reserved, intermediate network devices such as routers and firewalls might silently discard packets when the ECN fields are set to nonzero values. To ensure that ECN-marked TCP/IP traffic will not be dropped from your network, survey your networking equipment and perform the appropriate configuration or upgrades to ensure that ECN-marked packets are not discarded.

Total Length

As Figure 5-2 shows, the Total Length field is 2 bytes long and is used to indicate the size of the IP datagram (IP header and IP payload) in bytes. With 16 bits, the maximum total length that can be indicated is 65,535 bytes. For typical maximum-sized IP datagrams, the total length is the same as the IP MTU for that Network Interface Layer technology.

Between the header length and the total length, the IP payload length can be determined from the following formula:

IP payload length (bytes) = Total Length value (bytes) – (4 × Header Length value (32-bit words))

Identification

The Identification field is 2 bytes long and is used to identify a specific IP packet sent between a source and destination node. The sending host sets the field's value, and the field is incremented for successive IP datagrams. The Identification field is used to identify the fragments of an original IP datagram.

Flags

The Flags field is 3 bits long and contains two flags for fragmentation. One flag is used to indicate whether the IP payload is eligible for fragmentation, and the other indicates whether or not there are more fragments to follow for this fragmented IP datagram.

More information on these flags and their uses can be found in the section entitled "Fragmentation," later in this chapter.

Fragment Offset

The Fragment Offset field is 13 bits long and is used to indicate the offset of where this fragment begins relative to the original unfragmented IP payload.

More information on the Fragment Offset field can be found in the section entitled "Fragmentation," later in this chapter.

Time-To-Live

The Time-To-Live (TTL) field is 1 byte long and is used to indicate how many links on which this IP datagram can travel before an IP router discards it. The TTL field was originally intended for use as a time counter, to indicate the number of seconds that the IP datagram could exist on the Internet. An IP router was intended to keep track of the time that it received the IP datagram and the time that it forwarded the IP datagram. The TTL was then decreased by the number of seconds that the packet resided at the router.

However, the latest modern standard (RFC 1812) specifies that IP routers decrement the TTL by 1 when forwarding an IP datagram. Therefore, the TTL is an inverse link count. The sending host sets the initial TTL, which acts as a maximum link count. The maximum value limits the number of links on which the datagram can travel and prevents a datagram from indefinitely looping.

Some additional aspects of the TTL field include the following:

- Routers decrement the TTL in received packets to be routed before consulting the routing table. If the TTL is less than 1, the packet is discarded and an ICMP Time Expired-TTL Expired In Transit message is sent back to the sending host.
- Unicast destination hosts do not check the TTL field.
- Sending hosts must send IP datagrams with a TTL greater than 0. The exact value of the TTL for sent IP datagrams is either an operating system default or is specified by the application. The maximum value of the TTL is 255.
- A recommended value of the TTL is twice the diameter of your internetwork. The diameter is the number of links between the farthest two nodes on the IP internetwork.
- The TTL is independent of routing protocol metrics such as the Routing Information Protocol (RIP) hop count and the OSPF cost.

Note The TTL can be mistakenly referred to as a hop count when in fact it is a link count. The difference is subtle but important. The hop count is the number of routers to cross to reach a given destination. Link count is the number of Network Interface Layer links to cross to reach a given destination. The difference between hop count and link count is 1. For example, if Host A and Host B are separated by five routers, the hop count is 5, but the link count is 6. An IP datagram sent from Host A to Host B with a TTL of 5 is discarded by the fifth router. An IP datagram sent from Host A to Host B with a TTL of 6 will arrive at Host B.

The default TTL for Windows Server 2008 and Windows Vista is 128. You can change the default value of the TTL field for sent packets with the following command:

netsh interface ipv4 set global defaultcurhoplimit=*TTL*

You can also use the following registry value:

DefaultTTL

```
Key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters
Value type: REG_DWORD
Valid range: 0 - 255
Default: 128
Present by default: No
```

The default value of DefaultTTL is set to 128 so that IP packets sent by a Windows Server 2008 or Windows Vista–based computer can reach locations on the Internet that might need to traverse many links. Changing the value of DefaultTTL is necessary only when the diameter of your network changes. Windows Sockets applications can override this default value.

Setting the TTL with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-i** option can be used to set the TTL value in ICMP Echo messages. The syntax is:

ping -i *TTLValue Destination*

For example, to ping 10.0.0.1 with a TTL field that is set to 7, use the following command:

ping -i 7 10.0.0.1

The default TTL for ICMP Echo messages sent by the Ping.exe tool is 128.

Protocol

The Protocol field is 1 byte long and is used to indicate the upper layer protocol contained within the IP payload. Some common values of the IP Protocol field are 1 for ICMP, 6 for TCP, and 17 (0x11) for UDP. The Protocol field acts as a multiplex identifier so that the payload can be passed to the proper upper layer protocol on receipt at the destination node.

Windows Sockets applications can refer to protocols by name. Protocol names are resolved to protocol numbers through the Protocol file stored in the %*SystemRoot*%\System32\Drivers\Etc directory.

Table 5-3 lists some of the values of the IP Protocol field for protocols that Windows Server 2008 and Windows Vista support.

Table 5-3. Values of the IP Protocol Field

Value	Protocol
1	ICMP
2	IGMP
6	TCP
17	UDP
41	IPv6
47	Generic Routing Encapsulation (GRE)
50	IP Security Encapsulating Security Payload (ESP)
51	IP Security Authentication Header (AH)

For a complete list of IP Protocol field values, see *http://www.iana.org/assignments/protocol-numbers*.

Header Checksum

The Header Checksum field is 2 bytes long and performs a bit-level integrity check on the IP header only. The IP payload is not included, and IP payloads must include their own checksums to check for bit-level integrity. The sending host performs an initial checksum in the sent IP datagram. Each router in the path between the source and destination verifies the Header Checksum field before processing the packet. If the verification fails, the router silently discards the IP datagram.

Because each router in the path between the source and destination decrements the TTL, the header checksum changes at each router.

To compute the header checksum, each 16-bit quantity in the IP header is ones-complemented; bits within the 16-bit quantity that are set to 0 are changed to 1, bits within the 16-bit quantity that are set to 1 are changed to 0. The ones complemented 16-bit quantities are added together and the sum is ones-complemented. The result is placed in the Header Checksum field.

For the purposes of computing the header checksum over all the fields in the IP header, the value of the Header Checksum field is set to 0.

Source Address

The Source Address field is 4 bytes long and contains the IP address of the source host, unless a network address translator (NAT) is translating the IP datagram. A NAT is used to translate between public and private addresses when connecting to the Internet. NAT is defined in RFC 1631.

Destination Address

The Destination Address field is 4 bytes long and contains the IP address of the destination host, unless the IP datagram is being translated by a NAT or being loose-or strict-source routed. More information on IP source routing can be found in the section entitled "IP Options," later in this chapter.

Options and Padding

Options and padding can be added to the IP header, but must be done in 4-byte increments so that the size of the IP header can be indicated using the Header Length field.

For an example of the structure of the IP header, the following is frame 1 of Capture 05-01, a Network Monitor trace that is included in the \Captures folder on the companion CD-ROM, as displayed with Network Monitor 3.1:

```
  Frame:
+ Ethernet: Etype = Internet IP (IPv4)
- Ipv4: Next Protocol = ICMP, Packet ID = 13517, Total IP Length = 60
  - Versions: IPv4, Internet Protocol; Header Length = 20
     Version:      (0100....) IPv4, Internet Protocol
     HeaderLength: (....0101) 20 bytes (0x5)
  - DifferentiatedServicesField: DSCP: 0, ECN: 0
     DSCP: (000000..) Differentiated services codepoint 0
     ECT:  (......0.) ECN-Capable Transport not set
     CE:   (.......0) ECN-CE not set
    TotalLength: 60 (0x3C)
    Identification: 13517 (0x34CD)
  - FragmentFlags: 0 (0x0)
```

```
      Reserved: (0..............)
      DF:       (.0.............) Fragment if necessary
      MF:       (..0............) This is the last fragment
      Offset:   (...0000000000000) 0
    TimeToLive: 128 (0x80)
    NextProtocol: ICMP, 1(0x1)
    Checksum: 47209 (0xB869)
    SourceAddress: 157.59.11.19
    DestinationAddress: 157.59.8.1
+ Icmp: Echo Request Message, From 157.59.11.19 To 157.59.8.1
```

Fragmentation

When a source host or a router must transmit an IP datagram on a link and the MTU of the link is less than the IP datagram's size, the IP datagram must be fragmented. When IP fragmentation occurs, the IP payload is segmented and each segment is sent with its own IP header.

The IP header contains information required to reassemble the original IP payload at the destination host. Because IP is a datagram packet-switching technology and the fragments can arrive in a different order from which they were sent, the fragments must be grouped (using the Identification field), sequenced (using the Fragment Offset field), and delimited (using the More Fragments flag).

Fragmentation Fields

Figure 5-6 shows the fragmentation fields in the IP header, which are described in the following sections.

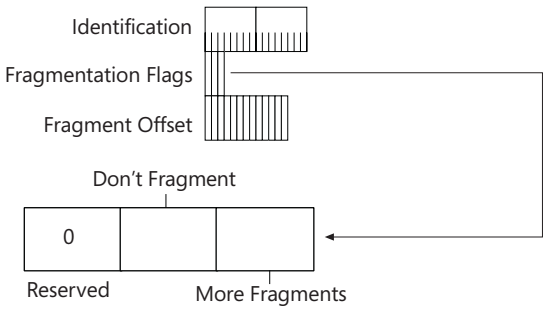


Figure 5-6 The fields in the IP header used for fragmentation.

Identification

The IP Identification field is used to group all the fragments of the payload of an original IP datagram together. The sending host sets the value of the Identification field, and this value is not changed during the fragmentation process. The Identification field is set even when fragmentation of the IP payload is not allowed by setting the Don't Fragment (DF) flag.

Don't Fragment Flag

The DF flag is set to 0 to allow fragmentation and set to 1 to prohibit fragmentation, so fragmentation occurs only if the DF flag is set to 0. If fragmentation is needed to forward the IP datagram and the DF flag is set to 1, the router should send an ICMP Destination Unreachable-Fragmentation Needed And DF Set message back to the source host and discards the IP datagram.

Fragmentation and reassembly is an expensive process at the routers and the destination host. The DF flag and the ICMP Destination Unreachable-Fragmentation Needed And DF Set message are the mechanisms by which a sending host discovers the MTU of the path between the source and the destination, or Path MTU Discovery. For more information, see Chapter 6, "Internet Control Message Protocol (ICMP)."

More Fragments Flag

The More Fragments (MF) flag is set to 0 if there are no more fragments that follow this fragment (this is the last fragment), and set to 1 if there are more fragments that follow this fragment (this is not the last fragment).

Fragment Offset

The Fragment Offset field is set to indicate the position of the fragment relative to the original IP payload. The Fragment Offset is an offset used for sequencing during reassembly, putting the incoming fragments in proper order to reconstruct the original payload. The Fragment Offset field is 13 bits long. With a maximum IP payload size of 65,515 bytes (the maximum IP MTU of 65,535 minus a minimum-sized IP header of 20 bytes), the Fragment Offset field cannot possibly indicate a byte offset. At 13 bits, the maximum value is 8191. The fragment offset must be 16 bits long to be a byte offset.

Because 16 bits are required to indicate a maximum-sized IP payload and only 13 bits are available in the Fragment Offset field, each value of the fragment offset must represent 3 bits. Therefore, the Fragment Offset field is defined in terms of 8-byte blocks, called *fragment blocks*.

During fragmentation, the payload is fragmented along 8-byte boundaries and the maximum number of 8-byte fragment blocks is placed in each fragment. The Fragment Offset field is set to indicate the starting fragment block for the fragment relative to the original IP payload.

For each fragment being fragmented by a router, the original IP header is copied and the following fields are changed:

Header Length

- Might or might not change depending on whether IP options are present and whether the options are copied to all fragments or just the first fragment. IP options are discussed in the section entitled "IP Options," later in this chapter.

TTL

- Decremented by 1.

Total Length

- Changed to reflect the new IP header and payload size.

MF

- Set to 1 for the first or middle fragments. Set to 0 for the last fragment.

Fragment Offset

- Set to indicate the position of the fragment in fragment blocks relative to the start of the original unfragmented payload.

Header Checksum

- Recalculated based on the changed fields in the IP header.

The Identification field does not change for any fragment.

Fragmentation Example

As an example of the fragmentation process, a node on a Token Ring network sends a fragmentable IP datagram with the IP Identification field set to 9999 to a node on an Ethernet network, as shown in Figure 5-7.

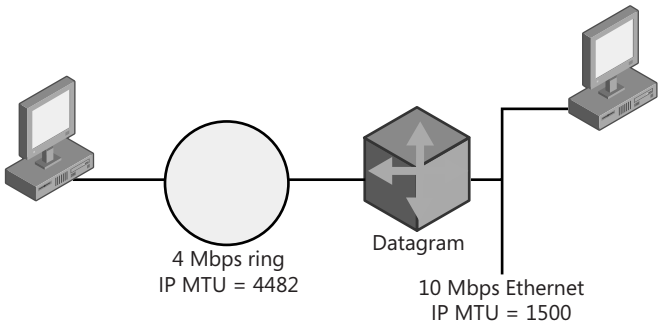


Figure 5-7 An example of a network where IP fragmentation can occur.

Assuming a 9-ms token holding time, a 4-Mbps ring, and no Token Ring source routing header, the IP MTU for the Token Ring network is 4482 bytes. The Ethernet IP MTU is 1500 bytes using Ethernet II encapsulation. Table 5-4 shows the fields relevant to fragmentation in the IP header and their values for the original IP datagram.

Table 5-4. Original IP Datagram

IP Header Field	Value
Total Length	4482
Identification	9999
DF	0
MF	0
Fragment Offset	0

The IP router connecting the two networks receives the IP datagram, checks its routing table, and notes that the interface on which to forward the datagram has a lower IP MTU than the datagram's size. The router then checks the DF flag. If set to 1, the router discards the IP datagram and then might send an ICMP Destination Unreachable-Fragmentation Needed And DF Set message back to the source host. If set to 0, the IP router fragments the 4462-byte IP payload (assuming no IP options are present) into four fragments, each of which can be sent on the 1500-byte Ethernet network.

IP payloads on an Ethernet network can be 1480 bytes long, assuming no IP options are present. Each 1480-byte payload is 185 fragment blocks (1480 ÷ 8 = 185). Therefore, the

four fragments are three fragments each with payloads of 1480 bytes and the last fragment with a payload of 22 bytes (4462 = 1480 + 1480 + 1480 + 22). Figure 5-8 shows the fragmentation process.

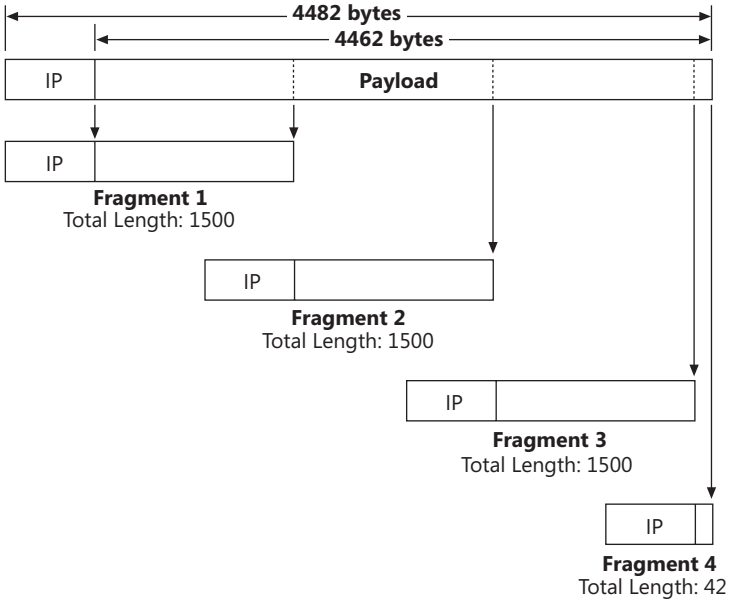


Figure 5-8 The IP fragmentation process when fragmenting from a 4482-byte IP MTU link to a 1500-byte IP MTU link.

Table 5-5 shows the fields relevant to fragmentation in the IP header of the four fragments.

Table 5-5. Fragments of the Original IP Datagram

IP Header Field	Value
Fragment 1	
Total Length	1500
Identification	9999
DF	0
MF	1
Fragment Offset	0
Fragment 2	
Total Length	1500
Identification	9999
DF	0
MF	1
Fragment Offset	185
Fragment 3	
Total Length	1500
Identification	9999
DF	0
MF	1
Fragment Offset	370

Fragment 4	
Total Length	42
Identification	9999
DF	0
MF	0
Fragment Offset	555

Note Token Ring is an older technology this is not in wide use today. This configuration is uncommon on modern networks and serves only as an example of a mixed-media network.

Reassembly Example

The fragments are forwarded by the intermediate IP router(s) to the destination host. Because IP is a datagram-based packet-switching technology, the fragments can take different paths to the destination and arrive in a different order from which the fragmenting router forwarded them. IP uses the Identification and Source IP Address fields to group the arriving fragments together.

After receiving a fragment (not necessarily the first fragment of the original IP payload), an IP implementation can allocate reassembly resources comprised of the following:

- A data buffer to contain the IP payload (65,515 bytes)
- A header buffer to contain the IP header (60 bytes)
- A fragment block bit table (1024 bytes or 8192 bits)
- A total length data variable
- A timer

IP can determine that a fragment arrived because either the MF flag or the Fragment Offset field has a nonzero value. An unfragmented IP datagram has the MF flag set to 0 and the Fragment Offset field set to 0. When the first fragment arrives (the Fragment Offset field is 0), its IP header is placed in the header buffer. When the last fragment arrives (the MF flag is 0), the total data length is computed.

For each arriving fragment, the IP payload is placed in the data buffer according to the values of the Fragment Offset and Total Length fields; the bits corresponding to the arriving fragment blocks are set in the fragment block bit table. When the final fragment arrives (which might not be the last fragment), all the bits in the fragment block bit table are set and reassembly of the original IP datagram is complete. IP delivers the IP payload to the appropriate upper layer protocol based on the Protocol field's value.

The reassembly timer is used to abandon the reassembly process within a certain amount of time. If all the fragments do not arrive before the reassembly timer expires, the IP datagram is discarded and the destination host can send an ICMP Time Exceeded-Fragmentation Time Expired message to the source host. RFC 791 recommends a default reassembly timer of 15 seconds; as fragments arrive, the reassembly timer is set to the maximum of the current value and the value of the arriving fragment's TTL field.

Figure 5-9 shows the reassembly process for our example fragmentation.

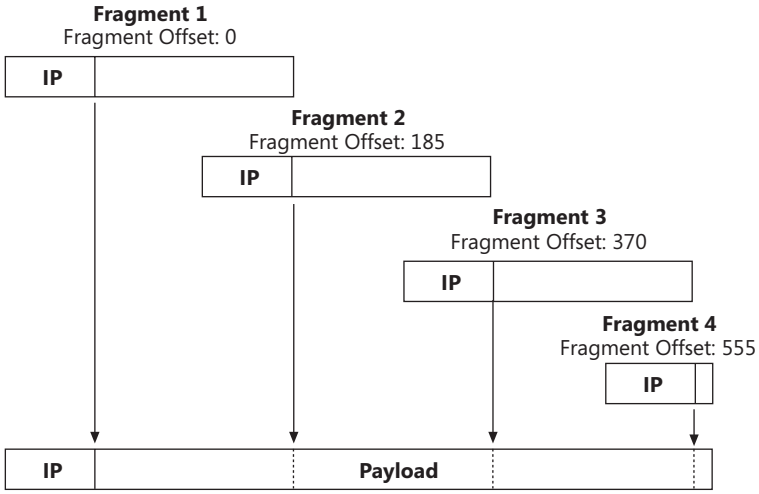


Figure 5-9 The IP reassembly process for the four fragments of the original IP datagram.

Fragmenting a Fragment

It is possible for fragments to become further fragmented. In this case, each fragmented payload is fragmented to fit the MTU of the link onto which it is being forwarded. The process of fragmenting a fragmented payload is slightly different from fragmenting an original IP payload in how the MF flag is set.

When fragmenting a previously fragmented payload, the MF flag is always set to 1, except when the fragment of the fragmented payload is the last fragment of the original payload.

- If an IP router fragments a previously fragmented first or middle fragment, all of the fragments have the MF flag set to 1.
- If an IP router fragments a previously fragmented last fragment, all of the fragments except the last fragment have the MF flag set to 1.

Therefore, regardless of how many times the IP datagram is fragmented, only one fragment has the MF flag set to 0, indicating the last fragment of the original IP payload.

Network Monitor Capture 05-02 (in the \Captures folder on the companion CD-ROM) provides an example of source-based IP fragmentation. The capture is the fragmentation of a 1500-byte IP datagram so that it fits on a subnet with a 576-byte IP MTU.

Avoiding Fragmentation

Although fragmentation allows IP nodes to communicate regardless of differing MTUs in intermediate subnets and without user intervention, IP fragmentation and reassembly is a relatively expensive process—both at the routers (or sending hosts) and at the destination host. On the modern Internet, fragmentation is highly discouraged; Internet routers are busy enough with the forwarding of IP traffic.

Fragmentation can be avoided by taking the following two measures:

- Discover the IP MTU that is supported by all of the links in the path between the source and the destination (the path MTU).
- Set the DF flag to 1 on all IP datagrams sent.

For more information on the Path MTU Discovery process, see Chapter 6, "Internet Control Message Protocol (ICMP)."

Setting the DF Flag with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-f** option can be used to set the DF flag to 1 in ICMP Echo messages. The syntax is

ping –f *Destination*

For example, to ping 10.0.0.1 and set the DF to 1, use the following command:

ping -f 10.0.0.1

By default, ICMP Echo messages sent by the Ping.exe tool have the DF flag set to 0 (fragmentation allowed).

Setting the IP Payload Size with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-l** option can be used to send IP packets with an arbitrary size by specifying the size of the Optional Data field in an ICMP Echo message. The syntax is:

ping -l *OptionalDataFieldSize Destination*

OptionalDataFieldSize is the size of the Optional Data field in an ICMP Echo message in bytes.

For example, to ping 10.0.0.1 with an Optional Data field size of 5000, use the following command:

ping -l 5000 10.0.0.1

The default Optional Data field size for Ping is 32 bytes.

The Optional Data field size is not the same as the IP payload size because ICMP Echo messages include an 8-byte ICMP header. Therefore, to calculate the IP payload's size, add 8 to the Optional Data field size. To calculate the IP datagram's size, add 20 to the size of the IP payload (or 28 to the size of the Optional Data field size). To ping with an ICMP Echo message at the maximum size allowed by the Network Interface technology, subtract 28 from the IP MTU. For example, to ping the address 10.0.0.1 with a maximum-sized ICMP Echo message on an Ethernet network (with an IP MTU of 1500), use the following Ping command:

ping -l 1472 10.0.0.1

Using Ping to Do Source Fragmentation

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-l** option can be used to do source fragmentation. Pinging with an Optional Data field size that is greater than (IP MTU – 28) bytes produces source-fragmented packets. For example, pinging from an Ethernet node with an Optional Data field size of 1472 or less does not produce fragmented packets. Pinging from an Ethernet node with an Optional Data field size greater than 1472 does produce fragmented packets.

Fragmentation and Translational Bridging Environments

Translational bridging is the interconnection of two different Network Interface Layer technologies on the same network by a Layer 2 device such as a bridge or switch. Translational bridges were used to connect an Ethernet segment to a Token Ring segment. In modern networks, switches use translational bridging to connect 10-Mbps or 100-Mbps Ethernet nodes to servers on high-speed ports. Common high-speed port technologies include FDDI, Gigabit Ethernet (GbE), and ATM.

The most serious obstacle to translational bridging is the difference in MTU between various Network Interface Layer technologies. Because there is no router involved, we cannot rely on either fragmentation or Path MTU Discovery processes to account for the differing MTUs. A translational bridge does not have the capability to fragment. Frames larger than the MTU of the link onto which they are to be forwarded are silently discarded by the bridge. As discussed in Chapter 10, "Transmission Control Protocol (TCP) Basics," when a TCP connection is established, both nodes communicate MTU information in the form of the TCP Maximum Segment Size (MSS) option. However, despite this indication, proper communication between all nodes in a translational bridging environment might require the modification of the IP MTU of specific nodes.

For example, Figure 5-10 shows two Ethernet switches connected on an Ethernet backbone. On each Ethernet switch is an FDDI port connected to an FDDI ring containing application servers. When the servers on the same FDDI ring communicate with each other, they can send packets with the FDDI MTU of 4352 bytes. When an Ethernet node on one of the switches uses TCP to connect to an application server on either FDDI ring, the TCP MSS option lowers the maximum size of TCP segments for IP datagrams of 1500 bytes.

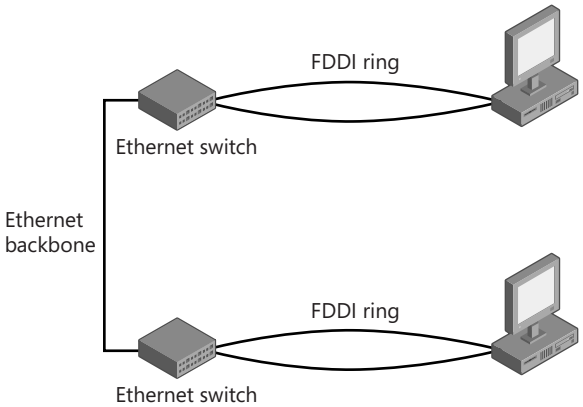


Figure 5-10 An MTU problem in a translational bridging environment caused by two FDDI hosts connected to two Ethernet switches.

However, consider the communication between application servers on different FDDI rings. In creating the TCP connection, each server indicates an FDDI-based TCP MSS. Therefore, Ethernet switches silently discard TCP-based IP datagrams sent between servers on different rings that have an IP total length greater than 1500.

The solution to this problem is to manually configure the application servers' IP MTU for the smallest IP MTU of all the links within the translational bridged network.

Using our example, the IP MTU of the application servers on the FDDI rings are set to 1500, so translational bridges can forward IP datagrams between FDDI rings. Changing the application servers' MTU means that when sending packets to application servers on the same ring, the packets are sent at the lower MTU of 1500, a lower efficiency than the default FDDI MTU of 4352. However, it is better to have lower efficiency between servers on the same ring than zero efficiency between servers on different rings. For nodes running Windows Server 2008 or Windows Vista, use the MTU registry value to override the default MTU setting reported by NDIS.

Note FDDI is an older technology whose use has been made obsolete by 100 Mbps Ethernet. This configuration is unlikely on modern networks and serves only as an example of a mixed-media subnet.

Fragmentation and TCP/IP for Windows Server 2008 and Windows Vista

TCP/IP for Windows Server 2008 and Windows Vista supports IP fragmentation and reassembly with the following additional behaviors:

- IP can handle irregular fragments, which overlap either fully or partially, with already received fragments for the same payload.
- When forwarding fragments, IP can forward the individual fragments separately or hold all of the fragments and then send all of them when the last one arrives. The default behavior is to forward individual fragments. You can change this behavior with the **netsh interface ipv4 set global groupforwardedfragments=enabled** command.
- The maximum amount of memory that can be allocated for reassembly for all incoming IP packets is controlled by the **netsh interface ipv4 set global reassemblylimit=***MemorySize* command. You can view the current size of the reassembly buffer with the **netsh interface ipv4 show global** command.

IP Options

IP options are additional fields appended to the standard 20-byte IP header. Although IP options are not required on each IP header, the ability to process IP option fields is required. IP options are used infrequently and mostly for network testing purposes.

The IP options portion size of the IP header varies in length based on the IP options that are being used. The individual IP options also vary in length from a single byte to multiple four-byte quantities. Recall that the maximum-sized IP header that can be indicated with

the Header Length field is 60 bytes. With a standard IP header size of 20 bytes, 40 bytes are left for IP options.

The first byte of each IP option has the format shown in Figure 5-11.

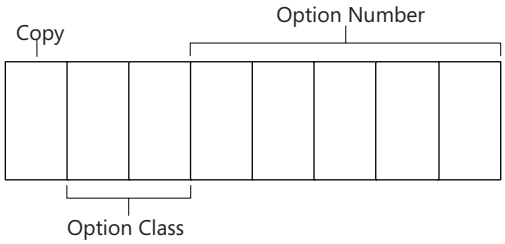


Figure 5-11 The structure of the first byte in an IP option.

Copy

The Copy field is 1 bit long and is used when a router or a sending host must fragment the IP datagram. When the Copy field is set to 0, the IP option should be copied only into the first fragment. When the Copy field is set to 1, the IP option should be copied into all fragments.

Option Class

The Option Class field is 2 bits long and is used to indicate the general class of the option. Table 5-6 lists the defined option classes.

Table 5-6. Option Classes

Option Class	Description
0	Network control
1	Reserved for future use
2	Debugging and measurement
3	Reserved for future use

Option Number

The Option Number field is 5 bits long and is used to indicate a specific option within the option class. Each option class can have up to 32 different option numbers.

Table 5-7 lists the defined option classes and numbers for nonmilitary computing.

Table 5-7. Option Classes and Numbers

Option Class	Option Number	Description
0	0	**End Of Option List** A one-byte option used to indicate the end of an option list
0	1	**No Operation** A one-byte option used to align bytes in a list of options
0	3	**Loose Source Routing** A variable-length option used to route a datagram through a specified path where alternate routes can be taken
0	7	**Record Route** A variable-length option used to trace a route through an IP internetwork

0	9	**Strict Source Routing** A variable-length option used to route a datagram through a specified path where alternate routes cannot be taken
0	20	**IP Router Alert** A fixed-length option used to inform the router that additional processing of the datagram is required
2	4	**Internet Timestamp** A variable-length option used to record a series of timestamps at each hop

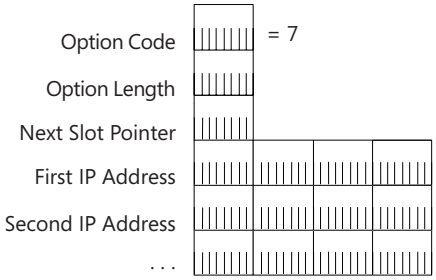
End Of Option List

The End Of Option List option is always a single byte in length and is used at the end of the IP options when they do not fall on a 4-byte boundary. This option is used only at the end of all the IP options, not at the end of each option.

No Operation

The No Operation option is always a single byte in length and is used between IP options when an IP option does not fall on a 4-byte boundary.

Record Route



The Record Route option is a variable-length option that is used to record the IP addresses of the far side interfaces of IP routers as it traverses the IP internetwork. The far side interface is the interface on the router on which the IP datagram is forwarded, presumed to be farthest from the sending host.

As the IP datagram is forwarded from router to router, each router adds its IP address to the list; each router also modifies the Next Slot Pointer field. The route from the source host to the destination host is recorded. To get the complete route, there must be enough room in the Record Route option. Unlike Token Ring source routing, the number of IP address slots is specified by the sending host and is fixed in the IP header.

The Record Route option contains the following fields:

Option Code

- Set to 7 (Copy Bit=0, Option Class=0, Option Number=7).

Option Length

- Set by the sending host to the number of bytes in the Record Route option.

Next Slot Pointer

- Set to the byte offset (starting at 1) within the Record Route option of the next available IP address. The minimum value of the Next Slot Pointer field is 4.

First IP Address, Second IP Address

- Set to the IP address of the far side interface by routers. With a maximum of 40 bytes in the IP options portion of the IP header, there is enough room for a maximum of nine IP addresses.

Record Route Processing

An IP router receiving an IP datagram with the Record Route option compares the Option Length and Next Slot Pointer fields. If the Next Slot Pointer field is less than the Option Length field, there are open IP address fields. The router records the IP address of the interface that is forwarding the datagram in the next available IP address field; the router also updates the Next Slot Pointer field by adding 4. If the value of the Next Slot Pointer field is greater than the Option Length field, routers have used all of the available IP address fields. The router then forwards the IP datagram without modifying the Record Route option.

Because the Record Route option size is not a multiple of 4 bytes, either an End Of Options option (if there are no more options) or a No Operation option (if there are more options) must be added to ensure that the IP header is an integral multiple of 4 bytes.

Setting the Record Route Option with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-r** option can be used to add the Record Route option and set the number of IP address slots in the Record Route option within an ICMP Echo message. The syntax is:

ping -r *IPAddressSlots Destination*

For example, to ping 10.0.0.1 with seven IP address slots, use the following command:

ping -r 7 10.0.0.1

When both hosts are computers running Windows Server 2008 or Windows Vista, the Record Route option records the IP addresses of the far side interfaces of forwarding routers in the ICMP Echo message. When the Echo message is received, the IP addresses recorded are maintained and the Echo Reply message is sent with the same Record Route option. The Echo Reply message contains the recorded route for the Echo message and the recorded route for the Echo Reply message.

Therefore, with the Ping **-r** option, it is possible to record the far side router interfaces for the Echo message (the path from Host A to Host B) and the far side router interfaces for the Echo Reply message (the path from Host B to Host A). However, because there is only room for nine IP address slots, this is possible only if there are no more than four routers between hosts.

Network Monitor Capture 05-03 (in the \Captures folder on the companion CD-ROM) provides an example of Ping.exe tool traffic and the use of the Record Route option.

> **Note** The Tracert.exe tool does not use the Record Route option.

Strict and Loose Source Routing

The IP routing process at IP routers is performed through a comparison of the destination IP address with entries in a local routing table. Each router makes a forwarding decision. However, it is sometimes necessary to specify a path that an IP datagram is to take regardless of the router's routing table entries. The path is specified before the source host sends the datagram; this is known as *source routing*.

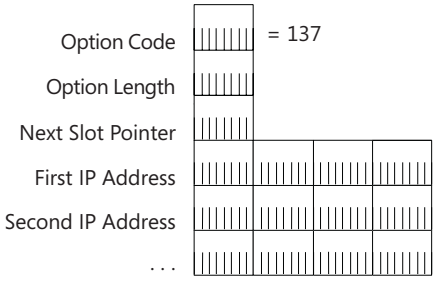
For example, in a multipath IP internetwork (where there is more than one path between IP networks), routers choose the best path based on a lowest cost metric. Once a router determines all of the best paths, the higher cost paths are not used unless the topology of the internetwork changes. To check that higher cost paths contain valid links, you must do source routing.

Source routing in IP is done by specifying the IP addresses of the near side interfaces of the desired routers between the source and its destination. At each leg of the journey, the destination IP address in the IP header is set to the IP address of the next near side router interface. IP supports both loose and strict source routing. In loose source routing, the next router's IP address does not have to be a neighboring router; it can be multiple hops away. In strict source routing, the next router's IP address must be a neighboring router (a single hop away).

IP source routing also records the path taken in the same way as the Record Route option. For each intermediate destination, the IP address of the interface on the router that forwarded the IP datagram is recorded.

Note To use IP source routing, it must be enabled on all the routers in the path between the source and destination hosts. It is a common practice to disable source routing on routers, especially those connected to the Internet.

Strict Source Route Option



The Strict Source Route option contains the following fields:

Option Code

- Set to 137 (Copy Bit=1, Option Class=0, Option Number=9).

Option Length

- Set by the sending host to the number of bytes in the Strict Source Route option.

Next Slot Pointer

- Set to the byte offset (starting at 1) within the Strict Source Route option for the next router. The Next Slot Pointer field's minimum value is 4. This field is used also in the

same manner as the Record Route option to determine the location of the next IP address slot for recording the route.

First IP Address, Second IP Address

- Set by the sending host for the series of IP addresses for successive router destinations in the strict source route; set also by IP routers to the IP address of the forwarding interface. With a maximum of 40 bytes in the IP options portion of the IP header, there is enough room for a maximum of nine IP addresses.

When a sending host sends an IP datagram with the Strict Source Route option, the sending host does the following:

1. Sets the Next Slot Pointer field's value to 4.
2. Places the first IP address in the strict source route in the IP header's Destination IP Address field.

When an IP router receives an IP datagram as the destination with the Strict Source Route option, it compares the Option Length and Next Slot Pointer fields. If the Next Slot Pointer field is less than the Option Length field, the router does the following:

1. Adds 4 to the Next Slot Pointer field's value.
2. Replaces the IP header's destination IP address with the IP address that is recorded in the next slot (based on the Next Slot Pointer field's new value).
3. Records the IP address of the forwarding interface in the previous slot.

If the next destination IP address is not reachable using a directly attached network (the IP address of a neighboring router or host), the IP datagram is discarded and an ICMP Destination Unreachable-Source Route Failed message is sent back to the source host.

If the Next Slot Pointer field's value is greater than the Option Length field's value, the IP datagram has reached its final destination.

Because the size of the Strict Source Route option is not a multiple of 4 bytes, either an End Of Options option (if there are no more options) or a No Operation option (if there are more options after the Strict Source Route option) must be added to ensure that the IP header is an integral multiple of 4 bytes. In Windows Server 2008 and Windows Vista, TCP/IP places the Strict Source Route option as the last option in the list and uses an End Of Options option to specify the end of the list of options.

Setting the Strict Source Route Option with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-k** option can be used to add the Strict Source Route option. The Ping.exe tool with the **–k** option also can be used to set the IP addresses of successive routers and the final destination in ICMP Echo messages. The syntax is:

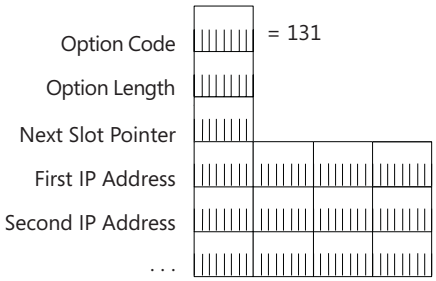
ping -k *FirstHopIPAddress SecondHopIPAddress …*
Destination

For example, to ping 10.0.0.1 through neighboring router interfaces 192.168.1.1 and 192.168.2.1, use the following command:

ping -k 192.168.1.1 192.168.2.1 10.0.0.1

Network Monitor Capture 05-04 (in the \Captures folder on the companion CD-ROM) provides an example of Ping.exe tool traffic and the use of the Strict Source Route option.

Loose Source Route Option



The Loose Source Route option contains the following fields:

Option Code

- Set to 131 (Copy Bit=1, Option Class=0, Option Number=3).

Option Length

- Set by the sending host to the number of bytes in the Loose Source Route option.

Next Slot Pointer

- Set to the byte offset (starting at 1) within the Loose Source Route option for the next router. The Next Slot Pointer field's minimum value is 4. The Next Slot Pointer field also is used in the same manner as the Record Route option to determine the location of the next IP address slot for recording the route.

First IP Address, Second IP Address

- Set by the sending host for the series of IP addresses for successive router destinations in the loose source route, and set by IP routers to the forwarding interface's IP address. With a maximum of 40 bytes in the IP options portion of the IP header, there is enough room for a maximum of nine IP addresses.

When a sending host sends an IP datagram with the Loose Source Route option, the sending host does the following:

1. Sets the Next Slot Pointer field's value to 4.
2. Places the first IP address in the loose source route in the IP header's Destination IP Address field.

When an IP router receives an IP datagram as the destination with the Loose Source Route option, it compares the Option Length and Next Slot Pointer fields. If the Next Slot Pointer field's value is less than the Option Length field's value, the router does the following:

1. Adds 4 to the Next Slot Pointer field's value.
2. Replaces the IP header's destination IP address with the IP address that is recorded in the next slot (based on the Next Slot Pointer field's new value).
3. Records the IP address of the forwarding interface in the previous slot.

If the Next Slot Pointer field's value is greater than the Option Length field's value, the IP datagram has reached its final destination.

Because the size of the Loose Source Route option is not a multiple of 4 bytes, either an End Of Options option (if there are no more options) or a No Operation option (if there are more options) must be added to ensure that the IP header is an integral multiple of 4 bytes.

Setting the Loose Source Route Option with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool with the **-j** option can be used to add the Loose Source Route option. Additionally, it is used to set the IP addresses of successive routers and the final destination in ICMP Echo messages. The syntax is:

ping -j *FirstHopIPAddress SecondHopIPAddress …*
Destination

For example, to ping 10.0.0.1 through neighboring router interfaces 192.168.1.1 and 192.168.2.1, use the following command:

ping -j 192.168.1.1 192.168.2.1 10.0.0.1

Network Monitor Capture 05-05 (in the \Captures folder on the companion CD-ROM) provides an example of Ping.exe tool traffic and the use of the Loose Source Route option.

By default, an IP router running Windows Server 2008 or Windows Vista does not forward source-routed IP packets. You can change the behavior of IP for source-routed IP packets with the following command:

netsh interface ipv4 set global sourceroutingbehavior=drop|forward|dontforward

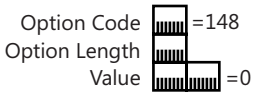
You can also use the following registry value:

DisableIPSourceRouting

```
Key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters
Value type: REG_DWORD
Valid range: 0 - 2
Default: 1
Present by default: No
```

Set the DisableIPSourceRouting registry value to 0 to forward source-routed packets, to 1 to not forward source-routed packets (for packets being forwarded), or to 2 to drop all incoming source-routed packets (for packets being forwarded and for packets destined to the node).

IP Router Alert



The IP Router Alert option is used to indicate to IP routers that additional processing of the IP datagram is required even when the IP datagram is not addressed to the router. The IP Router Alert option is used for the Resource Reservation Protocol (RSVP), IGMP version 2, and IGMP version 3. For example, when a router receives an IP datagram with the IP Router Alert option, it looks at the IP Protocol field to see if the IP payload requires

additional processing before making a forwarding decision. RFC 2113 describes the IP Router Alert option.

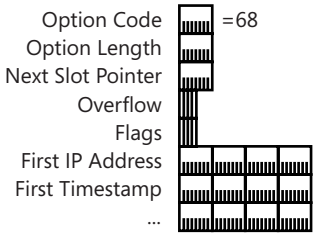
The IP Router Alert option contains the following fields:

Option Code
- Set to 148 (Copy Bit=1, Option Class=0, Option Number=20).

Option Length
- Set to the fixed length of 4.

Value
- A 2-byte field set to 0. All other values are reserved. The value of 0 indicates that the router must examine the packet.

Internet Timestamp



The Internet Timestamp option is used to record the time that an IP datagram arrived at each IP router in the path between the source and destination host. The Internet Timestamp option is similar to the Record Route option in that the sending node creates blank entries in the IP header that routers fill out as the packet travels through the IP internetwork. Each entry consists of the router's IP address and a 32-bit integer timestamp that indicates the number of milliseconds since midnight, Universal Time. If Universal Time is not being used, the high-order bit of the timestamp field is set to 1.

> **Note** To use Internet timestamps, Internet timestamping must be enabled on all the routers in the path between the source and destination hosts. It is common for routers to either not support Internet timestamping or have it disabled.

The Internet Timestamp option contains the following fields:

Option Code
- Set to 68 (Copy Bit=0, Option Class=2, Option Number=4).

Option Length
- Set by the sending host to the number of bytes in the Internet Timestamp option.

Next Slot Pointer
- Set to the byte offset (starting at 1) within the Internet Timestamp option of the next slot for the recording of the IP address and timestamp. The Next Slot Pointer field's minimum value is 5.

Overflow
- Set by routers to indicate the number of routers that were unable to record their IP address and timestamp.

Flags

- Set by the sending host to indicate the format of the IP Address/Timestamp slots. When Flags is set to 0, the IP address is omitted. This allows up to nine timestamps to be recorded. When Flags is set to 1, the IP address is recorded, allowing up to four IP address/timestamp pairs to be recorded. The Internet Timestamp option format shown assumes Flags is set to 1. When Flags is set to 3, the sending node specifies the IP addresses of successive routers: A timestamp is recorded only if the IP address in the slot matches the router's IP address.

First IP Address/First Timestamp

- Set by routers to record the IP address and timestamp of the routers encountered (when Flags is set to 1) or specified (when Flags is set to 3).

When a sending host sends an IP datagram with the Internet Timestamp option, the sending host does the following:

1. Sets the Next Slot Pointer field's value to 5.
2. For a specified route (when Flags is set to 3), places the series of IP addresses in the Internet Timestamp option.

When an IP router receives an IP datagram with the Internet Timestamp option, it compares the Option Length and Next Slot Pointer fields. If the Next Slot Pointer field's value is less than the Option Length field's value, it does the following:

- If Flags is set to 3, the router replaces the IP header's destination IP address with the IP address that is recorded in the next slot (based on the Next Slot Pointer field).
- If Flags is set to 1 or 3, the router records the IP address of the interface on which the IP datagram was received in the same slot.
- If Flags is set to 0, the router records the timestamp and adds 4 to the Next Slot Pointer field. If Flags is set to 1, the router records the timestamp after the IP address and adds 8 to the Next Slot Pointer field. If Flags is set to 3, the router replaces the IP address and adds 4 to the Next Slot Pointer field.

If the Next Slot Pointer field's value is greater than the Option Length field's value, the router increments the Overflow field. If the Overflow field is 15 before incrementing, an ICMP Parameter Problem is sent back to the source host.

Setting the Internet Timestamp Option with Ping

The Windows Server 2008 and Windows Vista Ping.exe tool and the **-s** option can be used to send ICMP Echo messages with the Internet timestamp. The syntax is the following:

ping -s *Slots Destination*

For example, to ping the IP address of 10.9.1.1 using Internet timestamps with three slots, use the following command:

ping -s 3 10.9.1.1

Network Monitor Capture 05-06 (in the \Captures folder on the companion CD-ROM) provides an example of Ping.exe tool traffic and the use of the Internet Timestamp option.

Summary

IP provides the internetworking building block for all other Internet Layer and higher protocols in the TCP/IP suite. IP provides a best effort, unreliable, connectionless datagram delivery service between networks of an IP internetwork. The IP header provides addressing, type of delivery, maximum link count, fragmentation, and checksum services. IP fragmentation provides a way for IP datagrams to travel over links with a lower IP MTU than the original IP datagram. The basic services of the IP header are extended through IP options, the most common of which provide source routing, path recording, router alert, and timestamping functions.