MARS

IETF RFC 2022 1996-11 http://www.cis.ohio-state.edu/htbin/rfc/rfc2022.html

Multicasting is the process whereby a source host or protocol entity sends a packet to multiple destinations simultaneously using a single, local 'transmit' operation. ATM is being utilized as a new link layer technology to support a variety of protocols, including IP. The MARS protocol has two broad goals: to define a group address registration and membership distribution mechanism that allows UNI 3.0/3.1 based networks to support the multicast service of protocols such as IP and to define specific endpoint behaviors for managing point to multipoint VCs to achieve multicasting of layer 3 packets. The Multicast Address Resolution Server (MARS) is an extended analog of the ATM ARP Server. It acts as a registry, associating layer 3 multicast group identifiers with the ATM interfaces representing the group's members. MARS messages support the distribution of multicast group membership information between MARS and endpoints (hosts or routers). Endpoint address resolution entities query the MARS when a layer 3 address needs to be resolved to the set of ATM endpoints making up the group at any one time. Endpoints keep the MARS informed when they need to join or leave particular layer 3 groups. To provide for asynchronous notification of group membership changes, the MARS manages a point to multipoint VC out to all endpoints desiring multicast support. Each MARS manages a cluster of ATM-attached endpoints.

The format of the header is shown in the following illustration:

	Octets
Address family	1-2
Protocol identification	3-9
Reserved	10-12
Checksum	13-14
Extensions offset	15-16
Operation code	17-18
Type and length of source ATM number	19
Type and length of source ATM subaddress	20

MARS header structure

Address family
Defines the type of link layer addresses being carried.

Protocol ID
Contains 2 subfields:
16-bit protocol type.
40-bit optional SNAP extension to protocol type.

Reserved
This reserved field may be subdivided and assigned specific meanings for other control protocols indicated by the version number.

Checksum
This field carries a standard IP checksum calculated across the entire message.

Extension offset
This field identifies the existence and location of an optional supplementary parameters list.

Operation code
This field is divided into 2 sub fields: version and type. Version indicates the operation being performed, within the context of the control protocol version indicated by mar$op.version.

Type and length of ATM source number
Information regarding the source hardware address.

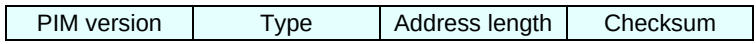
Type and length of ATM source subaddress
Information regarding the source hardware subaddress.

PIM

RFC 2117 http://www.cis.ohio-state.edu/htbin/rfc/rfc2117.html

Protocol Independent Multicast-Sparse Mode (PIM-SM) is a protocol for efficiently routing to multicast groups that may span wide-area (and inter-domain) internets. The protocol is not dependent on any particular unicast routing protocol, and is designed to support sparse groups.

The format of the PIM packet is shown in the following illustration:



PIM header structure

PIM version
Current PIM version is 2.

Type
Types for specific PIM messages.

Address length
Address length in bytes. The length of the address field throughout, in the specific message.

Checksum
The 16-bit one's complement, of the one's complement sum of the entire PIM message (excluding the data portion in the register message). For computing the checksum, the checksum field is zeroed.

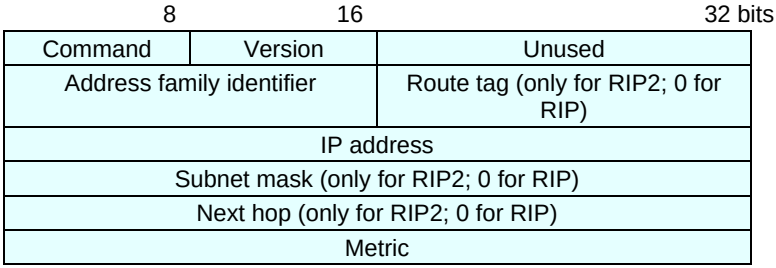
RIP

IETF RFC 1058 1988-06 http://www.cis.ohio-state.edu/htbin/rfc/rfc1058.html
IETF RFC 1723 1996-05 http://www.cis.ohio-state.edu/htbin/rfc/rfc1723.html
IETF RFC 1528 1994-02 http://www.cis.ohio-state.edu/htbin/rfc/rfc1528.html

RIP (Routing Information Protocol) is used by Berkeley 4BSD UNIX systems to exchange routing information. Implemented by a UNIX program, RIP derives from an earlier protocol of the same name developed by Xerox.

RIP is an extension of the Routing Information Protocol (RIP) intended to expand the amount of useful information carried in the RIP messages and to add a measure of security.

RIP is a UDP-based protocol. Each host that uses RIP has a routing process that sends and receives datagrams on UDP port number 520. The packet format of RIP is shown in the illustration below.



RIP packet structure

The portion of the datagram from Address Family Identifier through Metric may appear up to 25 times.

Command

The command field is used to specify the purpose of this datagram:

1 Request: A request for the responding system to send all or part of its routing table.

2 Response: A message containing all or part of the sender's routing table. This message may be sent in response to a request or poll, or it may be an update message generated by the sender.

3 Traceon: Obsolete. Messages containing this command are to be ignored.

4 Traceoff: Obsolete. Messages containing this command are to be ignored.
5 Reserved: Used by Sun Microsystems for its own purposes.

Version

The RIP version number. Datagrams are processed according to version number, as follows:

0 Datagrams whose version number is zero are to be ignored.
1 Datagrams whose version number is one are processed. All fields that are to be 0, are to be checked. If any such field contains a non-zero value, the entire message is ignored.
2 Specifies RIP messages which use authentication or carry information in any of the newly defined fields.
>2 Datagrams whose version numbers are greater than 1 are processed. All fields that are 0 are ignored.

Address family identifier

Indicates what type of address is specified in this particular entry. This is used because RIP may carry routing information for several different protocols. The address family identifier for IP is 2.

When authentication is in use, the Address Family Identifier field will be set to 0xFFFF, the Route Tag field contains the authentication type and the remainder of the message contains the password.

Route tag

Attribute assigned to a route which must be preserved and readvertised with a route. The route tag provides a method of separating internal RIP routes (routes for networks within the RIP routing domain) from external RIP routes, which may have been imported from an EGP or another IGP.

IP address

The IP address of the destination.

Subnet mask

Value applied to the IP address to yield the non-host portion of the address. If zero, then no subnet mask has been included for this entry.

Next hop

Immediate next hop IP address to which packets to the destination specified by this route entry should be forwarded.

Metric

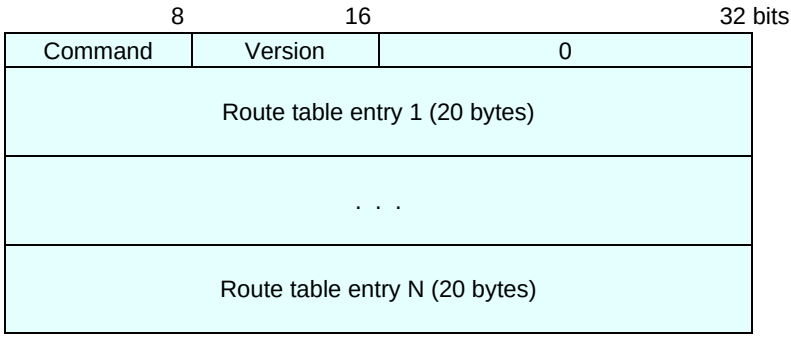
Represents the total cost of getting a datagram from the host to that destination. This metric is the sum of the costs associated with the networks that would be traversed in getting to the destination.

RIPng for IPv6

IETF RFC 2080 1997-01 http://www.cis.ohio-state.edu/htbin/rfc/rfc2080.html

RIPng for IPv6 is a routing protocol for the IPv6 Internet. It is based on protocols and algorithms used extensively in the IPv4 Internet.

The format of the header is shown in the following illustration:



RIPng for IPv6 header structure

Command

The purpose of the message. Possible commands are:

Request A request for the responding system to send all or part of its routing table

Response A message containing all or part of the sender's routing table.

Version

The version of the protocol. The current version is version 1.

Route table entry

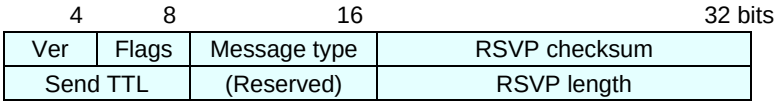
Each route table entry contains a destination prefix, the number of significant bits in the prefix and the cost of reaching that destination.

RSVP

IETF draft-ietf-rsvp-spec-13.txt 08-1996, IETF draft-ietf-rsvp-md5-02.txt 06-199

RSVP is a Resource ReSerVation setup Protocol designed for an integrated services Internet. It is used by a host on behalf of an application data stream to request a specific quality of service from the network for particular data streams or flows. It is also used by routers to deliver QoS control requests to all nodes.

The format of the header is shown in the following illustration:



RSVP header structure

Version
The protocol version number, this is version 1.

Flags
No flag bits are defined yet.

Message type
Possible values are:

1 Path.
2 Resv.
3 PathErr.
4 ResvErr.
5 PathTear.
6 ResvTear.
7 ResvConf.

RSVP checksum
The checksum.

Send TTL
The IP TTL value with which the message was sent.

RSVP length

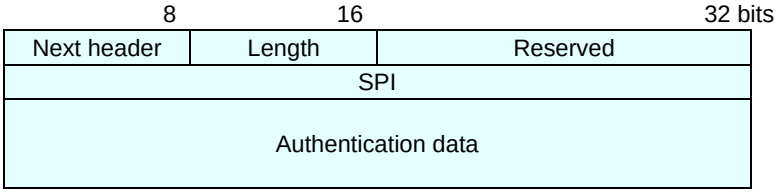
The total length of the RSVP message in bytes, including the common header and the variable length objects that follow.

AH

RFC 1826 http://www.cis.ohio-state.edu/htbin/rfc/rfc1826.html
RFC 1827 http://www.cis.ohio-state.edu/htbin/rfc/rfc1827.html

The IP Authentication Header seeks to provide security by adding authentication information to an IP datagram. This authentication information is calculated using all of the fields in the IP datagram (including not only the IP Header but also other headers and the user data) which do not change in transit. Fields or options which need to change in transit (e.g., hop count, time to live, ident, fragment offset or routing pointer) are considered to be zero for the calculation of the authentication data. This provides significantly more security than is currently present in IPv4 and might be sufficient for the needs of many users. When used with IPv6, the Authentication Header normally appears after the IPv6 Hop-by-Hop Header and before the IPv6 Destination Options. When used with IPv4, the Authentication Header normally follows the main IPv4 header.

The format of the header is shown in the following illustration:



AH header structure

Next header
The next payload after the authentication payload.

Length
The length of the authentication data field.

Reserved
Reserved for future use, must be set to zero.

Security parameters index (SPI)
Identifies the security association for this datagram.

Authentication data

Variable number of 32-bit words.

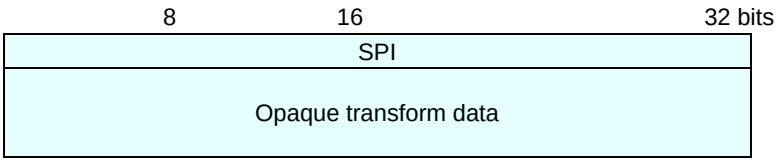
ESP

RFC 1826 http://www.cis.ohio-state.edu/htbin/rfc/rfc1826.html
RFC 1827 http://www.cis.ohio-state.edu/htbin/rfc/rfc2406.html

The IP Encapsulating Security Payload (ESP) seeks to provide confidentiality and integrity by encrypting data to be protected and placing the encrypted data in the data portion of the IP ESP. Depending on the user's security requirements, this mechanism may be used to encrypt either a transport-layer segment (e.g., TCP, UDP, ICMP, IGMP) or an entire IP datagram. Encapsulating the protected data is necessary to provide confidentiality for the entire original datagram.

ESP may appear anywhere after the IP header and before the final transport-layer protocol. The Internet Assigned Numbers Authority has assigned Protocol Number 50 to ESP. The header immediately preceding an ESP header will always contain the value 50 in its Next Header (IPv6) or Protocol (IPv4) field. ESP consists of an unencrypted header followed by encrypted data. The encrypted data includes both the protected ESP header fields and the protected user data, which is either an entire IP datagram or an upper-layer protocol frame (e.g., TCP or UDP).

The format of the header is shown in the following illustration:



ESP header structure

Security parameters index (SPI)

A 32-bit pseudo-random value identifying the security association for this datagram. If no security association has been established, the value of the SPI field is 0x00000000. An SPI is similar to the SAID used in other security protocols. The name has been changed because the semantics used here are not exactly the same as those used in other security protocols.

Opaque transform data

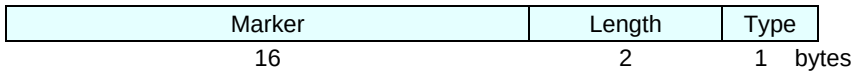
Variable length data field.

BGP-4

IETF RFC 1654 http://www.cis.ohio-state.edu/htbin/rfc/rfc1654.html

The Border Gateway Protocol (BGP) is an inter-Autonomous System routing protocol. The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems. BGP-4 provides a new set of mechanisms for supporting classes interdomain routing.

The format of the header is shown in the following illustration:



BGP-4 header structure

Marker

A 16-byte message containing a value predictable by the receiver of the message.

Length

The length of the message including the header.

Type

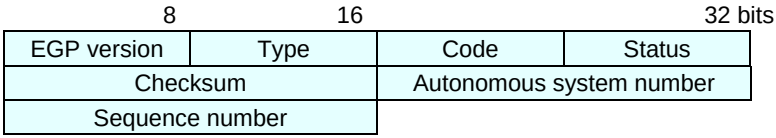
The message type. Possible messages are:
Open, Update, Notification, KeepAlive.

EGP

RFC 904 http://www.cis.ohio-state.edu/htbin/rfc/rfc904.html

The Exterior Gateway Protocol EGP exists in order to convey net-reachability information between neighboring gateways, possibly in different autonomous systems. The protocol includes mechanisms to acquire neighbors, monitor neighbor reachability and exchange net-reachability information in the form of Update messages. The protocol is based on periodic polling using Hello/I-Heard-You (I-H-U) message exchanges to monitor neighbor reachability and Poll commands to solicit Update responses.

The format of the header is shown in the following illustration:



EGP header structure

EGP Version

The version number.

Type

Identifies the message type. Possible types are as follows:

1 Update response/indication.
2 Poll command.
3 Neighbor acquisition message.
5 Neighbor reachability message.
8 Error response/indication.

Code

Identifies the message code.

Status

Contains message-dependent status information.

Checksum
The 16-bit one's complement of the one's complement sum of the EGP message starting with the EGP version number field. When computing the checksum the checksum field itself should be zero.

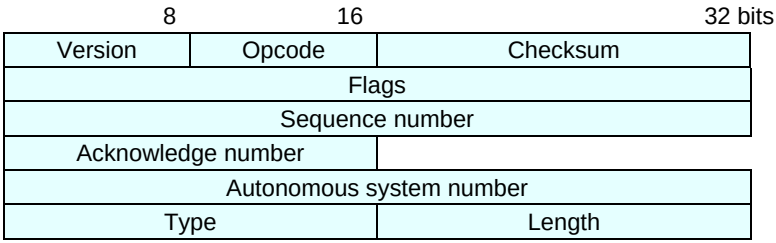
Autonomous system number
Assigned number identifying the particular autonomous system.

Sequence number
Send state variable (commands) or receive state variable (responses and indications).

EIGRP

Enhanced Interior Gateway Routing Protocol (EIGRP) is an enhanced version of IGRP. IGRP is Cisco's Interior Gateway Routing Protocol used in TCP/IP and OSI internets. It is regarded as an interior gateway protocol (IGP) but has also been used extensively as an exterior gateway protocol for inter-domain routing. IGRP uses distance vector routing technology. The same distance vector technology found in IGRP is also used in EIGRP, and the underlying distance information remains unchanged. The convergence properties and the operating efficiency of this protocol have improved significantly.

The format of the EIGRP header is shown in the following illustration.



EIGRP header structure

Version

The version of the protocol.

Opcode

1 Update.
2 Reserved.
3 Query.
4 Hello.
5 IPX-SAP.

Type

1 EIGRP Parameters.
2 Reserved.
3 Sequence.
4 Software version.
5 Next Multicast sequence.

Length
Length of the frame.

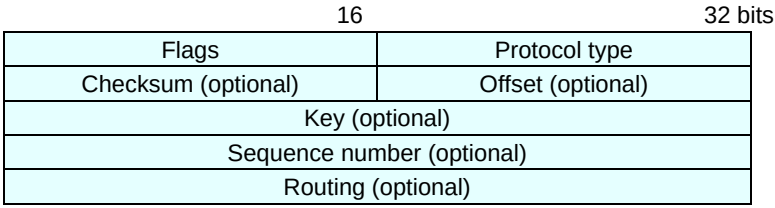
GRE

RFC 1701 http://www.cis.ohio-state.edu/htbin/rfc/rfc1701.html
RFC 1702 http://www.cis.ohio-state.edu/htbin/rfc/rfc1702.html

The Generic Routing Encapsulation (GRE) protocol provides a mechanism for encapsulating arbitrary packets within an arbitrary transport protocol. In the most general case, a system has a packet that needs to be encapsulated and routed (the payload packet). The payload is first encapsulated in a GRE packet, which possibly also includes a route. The resulting GRE packet is then encapsulated in some other protocol and forwarded (the delivery protocol).

GRE is also used with IP, using IP as the delivery protocol or the payload protocol. The GRE header used in PPTP is enhanced slightly from that specified in the current GRE protocol specification.

The format of the header is shown in the following illustration:



GRE header structure

Flags

The GRE flags are encoded in the first two octets. Bit 0 is the most significant bit, bit 12 is the least significant bit. Flags are as follows:

Checksum present (bit 0). When set to 1, the Checksum field is present and contains valid information.

Routing present (bit 1). When set to 1, the Offset and Routing fields are present and contain valid information.

Key present (bit 2). When set to 1, the Key field is present in the GRE header.

Sequence number present (bit 3). When set to 1, the Sequence number field is present.

Strict Source Route (bit 4). It is recommended that this bit only be set to 1 if all of the Routing Information consists of Strict Source Routes.

Recursion Control (bits 5-7). Contains a three bit unsigned integer which contains the number of additional encapsulations which are permissible.

Version Number (bits 13-15). Contains the value 0.

Protocol type
Contains the protocol type of the payload packet. In general, the value will be the Ethernet protocol type field for the packet.

Offset
Indicates the octet offset from the start of the Routing field to the first octet of the active Source Route Entry to be examined.

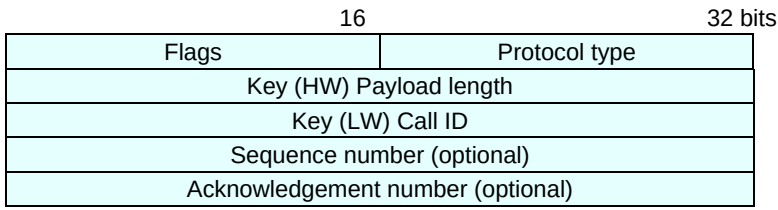
Checksum
Contains the IP (one's complement) checksum of the GRE header and the payload packet.

Key
Contains a four octet number which was inserted by the encapsulator. It may be used by the receiver to authenticate the source of the packet.

Sequence number
Contains an unsigned 32 bit integer which is inserted by the encapsulator. It may be used by the receiver to establish the order in which packets have been transmitted from the encapsulator to the receiver.

Routing
Contains data which may be used in routing this packet.

The format of the enhanced GRE header is as follows:



GRE header structure

Flags
Flags are defined as follows:
C (bit 0). Checksum Present.
R (bit 1). Routing Present.
K (bit 2). Key Present.
S (bit 3). Sequence Number Present.
s (bit 4). Strict source route present.
Recur (bits 5-7). Recursion control.
A (bit 8). Acknowledgment sequence number present.
Flags (bits 9-12). Must be set to zero.
Ver (bits 13-15). Must contain 1 (enhanced GRE).

Protocol type
Set to hex 880B.

Key
Use of the Key field is up to the implementation.

Sequence number
Contains the sequence number of the payload.

Acknowledgment number
Contains the sequence number of the highest numbered GRE packet received by the sending peer for this user session.

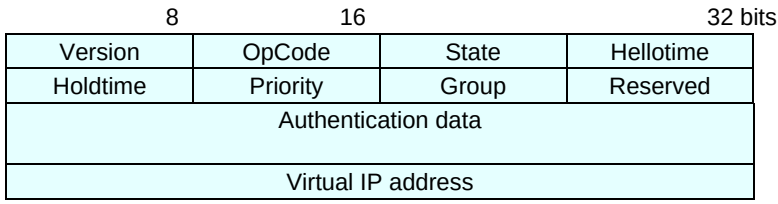
HSRP

RFC 2281 http://www.cis.ohio-state.edu/htbin/rfc/rfc2281.html

The Cisco Hot Standby Router Protocol (HSRP) provides a mechanism which is designed to support non-disruptive failover of IP traffic in certain circumstances. In particular, the protocol protects against the failure of the first hop router when the source host cannot learn the IP address of the first hop router dynamically. The protocol is designed for use over multi-access, multicast, or broadcast capable LANs (e.g., Ethernet). A large class of legacy host implementations that do not support dynamic discovery are capable of configuring a default router. HSRP provides failover services to those hosts.

HSRP runs on top of UDP, and uses port number 1985. Packets are sent to multicast address 224.0.0.2 with TTL 1. Routers use their actual IP address as the source address for protocol packets, not the virtual IP address. This is necessary so that the HSRP routers can identify each other.

The format of the data portion of the UDP datagram is shown in the following illustration:



HSRP header structure

Version

HSRP version number, 0 for this version.

OpCode

Type of message contained in the packet. Possible values are:

0 Hello, sent to indicate that a router is running and is capable of becoming the active or standby router.

1 Coup, sent when a router wishes to become the active router.

2 Resign, sent when a router no longer wishes to be the active router.

State

Internally, each router in the standby group implements a state machine. The State field describes the current state of the router sending the message. Possible values are:

0 Initial
1 Learn
2 Listen
4 Speak
8 Standby
16 Active

Hellotime

Approximate period between the Hello messages that the router sends (for Hello messages only). The time is given in seconds. If the Hellotime is not configured on a router, then it may be learned from the Hello message from the active router. The Hellotime should only be learned if no Hellotime is configured and the Hello message is authenticated. A router that sends a Hello message must insert the Hellotime that it is using in the Hellotime field in the Hello message. If the Hellotime is not learned from a Hello message from the active router and it is not manually configured, a default value of 3 seconds is recommended.

Holdtime

The amount of time, in seconds, that the current Hello message should be considered valid. (For Hello messages only.)

Priority

Used to elect the active and standby routers. When comparing priorities of two different routers, the router with the numerically higher priority wins. In the case of routers with equal priority the router with the higher IP address wins.

Group

Identifies the standby group. For Token Ring, values between 0 and 2 inclusive are valid. For other media, values between 0 and 255 inclusive are valid.

Authentication date

Clear-text 8 character reused password. If no authentication data is configured, the recommended default value is 0x63 0x69 0x73 0x63 0x6F 0x00 0x00 0x00.

Virtual IP Address

Virtual IP address used by this group. If the virtual IP address is not configured on a router, then it may be learned from the Hello message from the active router. An address should only be learned if no address was configured and the Hello message is authenticated.

IGRP

The Interior Gateway Routing Protocol (IGRP) was developed by the Cisco company. It is used to transfer routing information between routers.

IGRP is sent using IP datagrams with IP 9 (IGP). The packet begins with a header which starts immediately after the IP header.

	Octets
Version	1
Opcode	1
Edition	1
ASystem	1
Ninterior	1
Nsystem	1
Nexterior	1
Checksum	1

IGRP header structure

Version
Protocol version number (currently 1).

Opcode
Operation code indicating the message type:
1 Update.
2 Request.

Edition
Serial number which is incremented whenever there is a change in the routing table. The edition number allows gateways to avoid processing updates containing information that they have already seen.

ASystem
Autonomous system number. A gateway can participate in more than one autonomous system where each system runs its own IGRP. For each autonomous system, there are completely separate routing tables. This field allows the gateway to select which set of routing tables to use.

Ninterior, Nsystem, Nexterior

Indicate the number of entries in each of these three sections of update messages. The first entries (Ninterior) are taken to be interior, the next entries (Nsystem) as being system, and the final entries (Nexterior) as exterior.

Checksum

IP checksum which is computed using the same checksum algorithm as a UDP checksum. The checksum is computed on the IGRP header and any routing information that follows it. The checksum field is set to zero when computing the checksum. The checksum does not include the IP header and there is no virtual header as in UDP and TCP.

An IGRP request asks the recipient to send its routing table. The request message has only a header. Only the Version, Opcode and ASystem fields are used; all other fields are zero.

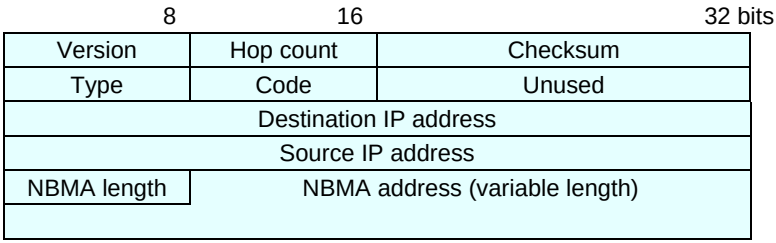
An IGRP update message contains a header, immediately followed by routing entries. As many routing entries as possible are included to fit into a 1500-byte datagram (including the IP header). With current structure declarations, this allows up to 104 entries. If more entries are needed, several update messages are sent.

NARP

RFC 1735 http://www.cis.ohio-state.edu/htbin/rfc/rfc1735.html

The NBMA Address Resolution Protocol (NARP) allows a source terminal (a host or router), wishing to communicate over a Non-Broadcast, Multi-Access link layer (NBMA) network, to find out the NBMA addresses of a destination terminal if the destination terminal is connected to the same NBMA network as the source.

The general format of the header is shown in the following illustration. The configuration varies according to the value of the type field to request type and reply type:



NARP header structure

Version

The NARP version number. Currently this value is 1.

Hop count

Indicates the maximum number of NASs that a request or reply is allowed to traverse before being discarded.

Checksum

The standard IP checksum over the entire NARP packet (starting with the fixed header).

Type

The NARP packet type. The NARP Request has a Type code 1, NARP Reply has a Type code 2.

Code

A response to an NARP request may contain cached information. If an authoritative answer is desired, then code 2 (NARP Request for Authoritative Information) should be used. Otherwise, a code value of 1 (NARP Request) should be used. NARP replies may be positive or negative. A Positive, Non- authoritative Reply carries a code of 1, while a Positive, Authoritative Reply carries a code of 2. A Negative, Non- authoritative Reply carries a code of 3 and a Negative, Authoritative reply carries a code of 4.

Source and destination IP addresses

Respectively, these are the IP addresses of the NARP requestor and the target terminal for which the NBMA address is desired.

NBMA length and NBMA address

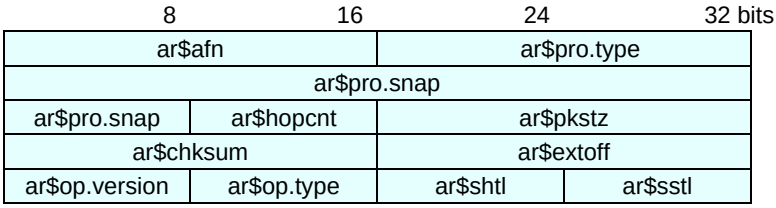
The NBMA length field is the length of the NBMA address of the source terminal in bits. The NBMA address itself is zero-filled to the nearest 32-bit boundary.

NHRP

RFC 2332 http://www.cis.ohio-state.edu/htbin/rfc/rfc2332.html
draft http://info.internet.isi.edu:80/in-drafts/files/draft-ietf-rolc-nhrp-15.txt

The NBMA Next Hop Resolution Protocol (NHRP) allows a source station (a host or router), wishing to communicate over a Non-Broadcast, Multi-Access (NBMA) subnetwork, to determine the internetworking layer addresses and NBMA addresses of suitable *NBMA next hops* toward a destination station.

The format of the header is shown in the following illustration:



NHRP header structure

ar$afn
Defines the type of link layer address being carried.

ar$pro.type
This field is a 16 bit unsigned integer.

ar$pro.snap
When ar$pro.type has a value of 0x0080, a snap encoded extension is being used to encode the protocol type. This snap extension is placed in the ar$pro.snap field; otherwise this field should be set to 0.

ar$hopcnt
The hop count. This indicates the maximum number of NHSs that an NHRP packet is allowed to traverse before being discarded.

ar$pktsz
The total length of the NHRP packet in octets.

ar$chksum
The standard IP checksum over the entire NHRP packet.

ar$extoff
This field identifies the existence and location of NHRP extensions.

ar$op.version
This field indicates what version of generic address mapping and management protocol is represented by this message.

ar$op.type
If the ar$op.version is 1 then this field represents the NHRP packet type. Possible values for packet types are:

1 NHRP Resolution Request.
2 NHRP Resolution Reply.
3 NHRP Registration Request.
4 NHRP Registration Reply.
5 NHRP Purge Request.
6 NHRP Purge Reply.
7 NHRP Error Indication.

ar$shtl
The type and length of the source NBMA address interpreted in the context of the *address family number*.

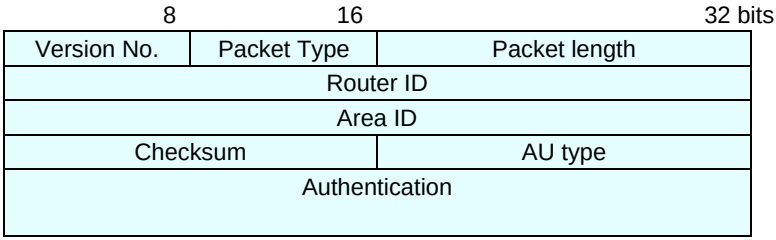
ar$sstl
The type and length of the source NBMA subaddress interpreted in the context of the "address family number".

OSPF

RFC1583 http://www.cis.ohio-state.edu/htbin/rfc/rfc1583.html

OSPF (Open Shortest Path First) protocol is a link-state routing protocol used for routing IP. It is an interior gateway protocol which is used for routing within a group of routers. It uses link-state technology in which routers send each other information about the direct connections and links which they have to other routers.

The OSPF header structure is shown in the illustration below.



OSPF header structure

Version number

Protocol version number (currently 1).

Packet type

Valid types are as follows:

1 Hello
2 Database Description
3 Link State Request
4 Link State Update
5 Link State Acknowledgment.

Packet length

The length of the protocol packet in bytes. This length includes the standard OSPF header.

Router ID

The router ID of the packet's source. In OSPF, the source and destination of a routing protocol packet are the two ends of an (potential) adjacency.

Area ID
A 32-bit number identifying the area that this packet belongs to. All OSPF packets are associated with a single area. Most travel a single hop only. Packets traveling over a virtual link are labeled with the back bone area ID of 0.0.0.0.

Checksum
The standard IP checksum of the entire contents of the packet, starting with the OSPF packet header, but excluding the 64-bit authentication field. This checksum is calculated as the 16-bit one's complement of the one's complement sum of all the 16-bit words in the packet, except for the authentication field. If the packet length is not an integral number of 16-bit words, the packet is padded with a byte of zero before checksumming.

AU type
Identifies the authentication scheme to be used for the packet.

Authentication
A 64-bit field for use by the authentication scheme.

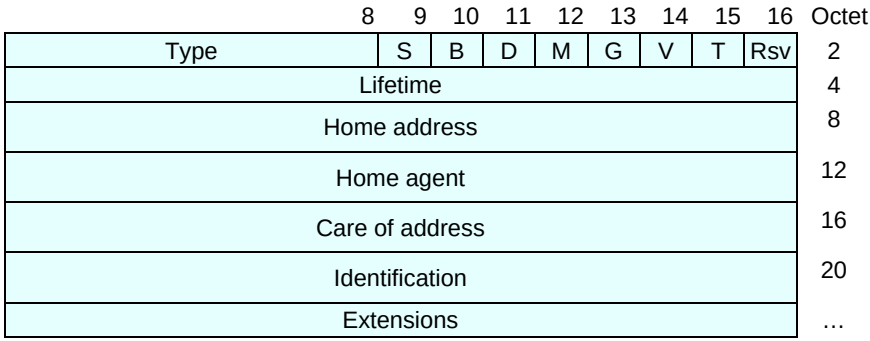
Mobile IP

RFC 2002: http://www.cis.ohio-state.edu/htbin/rfc/rfc2002.html
RFC 2290: http://www.cis.ohio-state.edu/htbin/rfc/rfc2290.html
RFC 2344: http://www.isi.edu/in-notes/rfc2344.txt

The Mobile IP protocol enables nodes to move from one IP subnet to another. Each mobile node is always identified by its home address, regardless of its current point of attachment to the Internet. While situated away from its home, a mobile node is also associated with a care-of address, which provides information about its current point of attachment to the Internet. The protocol allows registration of the care-of address with a home agent. The home agent sends datagrams destined for the mobile node through a tunnel to the care-of address. After arriving at the end of the tunnel, each datagram is then delivered to the mobile node. It can be used for mobility across both homogeneous and heterogeneous media. Mobile IP defines a set of new control messages, sent with UDP, Registration Request and Registration Reply.

The IP packet consists of the IP source and destination addresses, followed by the UDP source and destination ports, followed by the Mobile IP fields. Mobile IP packets can be either registration request or registration reply.

The format of the Mobile IP registration request message is shown in the following illustration:



Mobile IP registration request message structure

Type

1 signifies a registration request.

S
Simultaneous bindings. When set, the mobile node is requesting that the home agent retain its prior mobility bindings.

B
Broadcast datagrams. When set, the mobile node requests that the home agent tunnel to it any broadcast datagrams that it receives on the home network.

D
Decapsulation by mobile node. When set, the mobile node will itself decapsulate datagrams which are sent to the care-of address. In other words, the mobile node is using a co-located care-of address.

M
Minimal encapsulation. When set, the mobile node requests that its home agent use minimal encapsulation for datagrams tunneled to the mobile node.

G
GRE encapsulation. When set, the mobile node requests that its home agent use GRE encapsulation for datagrams tunneled to the mobile node.

V
The mobile node requests that its mobility agent use Van Jacobson header compression over its link with the mobile node.

T
When set, the mobile node asks its home agent to accept a reverse tunnel from the care-of address. Mobile nodes using a foreign agent care-of address ask the foreign agent to reverse-tunnel its packets.

Rsv
Reserved bit, set to zero.

Lifetime
The number of seconds remaining before the registration expires.

Home address
IP address of the mobile node.

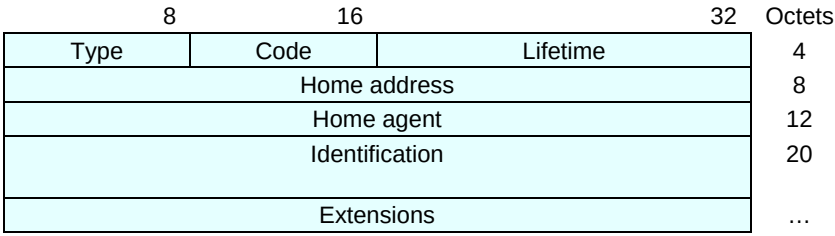
Home agent
IP address of the mobile node's home agent.

Care-of address
IP address for the end of the tunnel.

Identification
A 64-bit number, constructed by the mobile node, used for matching registration requests with registration replies, and for protecting against replay attacks of registration messages.

Extensions
The fixed portion of the registration request is followed by one or more of the extensions listed in Section 3.5 of RFC2002. The Mobile-Home Authentication Extension must be included in all registration requests.

The format of the Mobile IP registration reply message is shown in the following illustration:



Mobile IP registration reply message structure

Type
3 indicates a registration reply.

Code
A value indicating the result of the Registration Request. Values may be as follows:

Registration successful:

0 Registration accepted.

1 Registration accepted, but simultaneous mobility bindings unsupported.

Registration denied by the foreign agent:

64 Reason unspecified.

65 Administratively prohibited.

66 Insufficient resources.
67 Mobile node failed authentication.
68 Home agent failed authentication.
69 Requested Lifetime too long.
70 Poorly formed Request.
71 Poorly formed Reply.
72 Requested encapsulation unavailable.
73 Requested Van Jacobson compression unavailable.

Service denied by the foreign agent:

74 Requested reverse tunnel unavailable.
75 Reverse tunnel is mandatory and T bit not set.
76 Mobile node too distant

Registration denied by the home agent:

80 Home network unreachable (ICMP error received).
81 Home agent host unreachable (ICMP error received).
82 Home agent port unreachable (ICMP error received).
88 Home agent unreachable (other ICMP error received).

Service denied by the home agent:

137 Requested reverse tunnel unavailable.
138 Reverse tunnel is mandatory and T bit not set.
139 Requested encapsulation unavailable.

Lifetime

If the Code field indicates that the registration was accepted, the Lifetime field is set to the number of seconds remaining before the registration expires. A value of zero indicates that the mobile node has been deregistered. A value of 0xffff indicates infinity. If the Code field indicates that the registration was denied, the contents of the Lifetime field are unspecified and are ignored on reception.

The following fields are as described for the registration request message.

Van Jacobson

IETF RFC 1144 http://www.cis.ohio-state.edu/htbin/rfc/rfc1144.html

Van Jacobson is a compressed TCP protocol which thereby improves the TCP/IP performance over low speed (300 to 19,200 bps) serial links.

The format of the compressed TCP is as follows:

	C	I	P	S	A	W	U	Octets
Connection number (C)								1
TCP checksum								2
Urgent pointer (U)								1
Δ Window (W)								1
Δ Ack (A)								1
Δ Sequence (S)								1
Δ IP ID (I)								1
data								

Compressed TCP structure

C, I, P, S, A, W, U
Change mask. Identifies which of the fields expected to change per-packet actually changed.

Connection number
Used to locate the saved copy of the last packet for this TCP connection.

TCP checksum
Included so that the end-to-end data integrity check will still be valid.

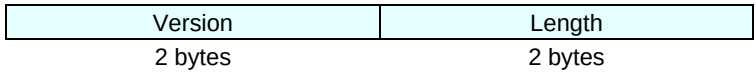
Urgent pointer
This is sent if URG is set.

Δ values for each field
Represent the amount the associated field changed from the original TCP(for each field specified in the change mask).

XOT

IETF RFC 1613 http://www.cis.ohio-state.edu/htbin/rfc/rfc1613.html

XOT is Cisco Systems X.25 over TCP. The format of the header is shown in the following illustration:



XOT header structure

Version
The version number.

Length
The length of the packet.

MGCP

IETF draft: http://www.ietf.org/internet-drafts/draft-huitema-mgcp-test1-00.txt

Media Gateway Control Protocol (MGCP) is used for controlling telephony gateways from external call control elements called media gateway controllers or call agents. A telephony gateway is a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks.

MGCP assumes a call control architecture where the call control intelligence is outside the gateways and handled by external call control elements. The MGCP assumes that these call control elements, or Call Agents, will synchronize with each other to send coherent commands to the gateways under their control. MGCP is, in essence, a master/slave protocol, where the gateways are expected to execute commands sent by the Call Agents.

The MGCP implements the media gateway control interface as a set of transactions. The transactions are composed of a command and a mandatory response. There are eight types of commands:

- CreateConnection.
- ModifyConnection.
- DeleteConnection.
- NotificationRequest.
- Notify.
- AuditEndpoint.
- AuditConnection.
- RestartInProgress.

The first four commands are sent by the Call Agent to a gateway. The Notify command is sent by the gateway to the Call Agent. The gateway may also send a DeleteConnection. The Call Agent may send either of the Audit commands to the gateway. The Gateway may send a RestartInProgress command to the Call Agent.

All commands are composed of a command header, optionally followed by a session description. All responses are composed of a response header, optionally followed by a session description. Headers and session descriptions are encoded as a set of text lines, separated by a carriage return and line feed character (or, optionally, a single line-feed character). The headers are separated from the session description by an empty line.

MGCP uses a transaction identifier to correlate commands and responses. Transaction identifiers have values between 1 and 999999999. An MGCP entity cannot reuse a transaction identifier sooner than 3 minutes after completion of the previous command in which the identifier was used.

The command header is composed of:

- A command line, identifying the requested action or verb, the transaction identifier, the endpoint towards which the action is requested, and the MGCP protocol version,
- A set of parameter lines, composed of a parameter name followed by a parameter value.

The command line is composed of:

- Name of the requested verb.
- Transaction identifier correlates commands and responses. Values may be between 1 and 999999999. An MGCP entity cannot reuse a transaction identifier sooner than 3 minutes after completion of the previous command in which the identifier was used.
- Name of the endpoint that should execute the command (in notifications, the name of the endpoint that is issuing the notification).
- Protocol version.

These four items are encoded as strings of printable ASCII characters, separated by white spaces, i.e., the ASCII space (0x20) or tabulation (0x09) characters. It is recommended to use exactly one ASCII space separator.

SGCP

IETF draft: http://www.ietf.org/internet-drafts/draft-huitema-sgcp-v1-02.txt

Simple Gateway Control Protocol (SGCP) is used to control telephony gateways from external call control elements. A telephony gateway is a network element that provides conversion between the audio signals carried on telephone circuits and data packets carried over the Internet or over other packet networks.

The SGCP assumes a call control architecture where the call control intelligence is outside the gateways and handled by external call control elements. The SGCP assumes that these call control elements, or Call Agents, will synchronize with each other to send coherent commands to the gateways under their control.

The SGCP implements the simple gateway control interface as a set of transactions. The transactions are composed of a command and a mandatory response. There are five types of commands:

- CreateConnection.
- ModifyConnection.
- DeleteConnection.
- NotificationRequest.
- Notify.

The first four commands are sent by the Call Agent to a gateway. The Notify command is sent by the gateway to the Call Agent. The gateway may also send a DeleteConnection.

All commands are composed of a Command header, optionally followed by a session description. All responses are composed of a Response header, optionally followed by a session description. Headers and session descriptions are encoded as a set of text lines, separated by a line feed character. The headers are separated from the session description by an empty line.

The command header is composed of:

- Command line.
- A set of parameter lines, composed of a parameter name followed by a parameter value.