IEN: 28
Section: 2.3.4.1

DRAFT INTERNETWORK PROTOCOL SPECIFICATION

Version 2

Jonathan B. Postel

February 1978

Information Sciences Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, California 90291

(213) 822-1511

TABLE OF CONTENTS

PREFACE

This is a very rough first cut at a specification of the "internet protocol". An attempt is made at following the outline suggested by the committee on protocol specifications at the recent TCP meeting.

Draft Internetwork Protocol Specification

Version 2

1. INTRODUCTION

The Internet Protocol is designed for use in interconnected systems of computer communication packet-switched networks. The internet protocol provides for transmitting segments of data from sources to destinations, where sources and destinations are identified by variable length addresses. The internet protocol also provides for fragmentation and reassembly of long segments, if necessary, for transmission through "small packet" networks.

1.1. History

This protocol has been developed as one result of the ARPA sponsored internetwork experiments program. The history until January 1978 is the history of the host-to-host protocol TCP.

The first publication of the ideas on which TCP is based was a paper in the IEEE Transactions on Communications by Cerf and Kahn in 1974 [1]. Later that year a protocol specification was published by a group led by Cerf at Stanford University [2]. A second specification was prepared in 1976 by a group led by Postel at SRI for the Defense Communication Agency for the AUTODIN II network [3]. In 1977 Cerf, at ARPA, prepared a substantial revision of the TCP specification [4]. Recently Postel revised Cerf's revision to distinguish the internet aspects from the host-to-host aspects [5].

1.2. Scope

The internet protocol is specifically limited in scope to provide the functions necessary to deliver a package of bits (an internet segment) from a source to a destination over an inconnected system of networks. There are no mechanism to promote relibility, flow control, sequencing, or other services commonly found in host-to-host protocols.

The protocol is intended to be utilized in gateways that interconnect sets of networks.

1.3. Other Documentation

No other documentation is known, however the TCP documents cited above provide some background, as do a series of working notes circulated in the ARPA research community. These notes are called Internetwork Experiment Notes (or IENs) and are collected into an Internet Notebook.

1.4. Interfaces

This protocol is called on by host-to-host protocols in an internet environment. This protocol calls on local network protocols to carry the internet packet to the next gateway or destination host.

For example a TCP module would call on the internet module to take a TCP segment (including the TCP header and user data) as the data portion of an internet segment. The TCP module would provide the addresses and other parameters in the internet header to the internet module as arguments of the call. The internet module would then create an internet segment and call on the local network interface to transmit the internet segment.

In the ARPANET case, for example, the internet module would call on a local net module which would add the 1822 leader [6] to the internet segment creating an ARPANET message to transmit to the IMP.

1.5. Operation

The internet protocol implements two basic functions: addressing, and fragmentation.

The internet modules use the addresses carried in the internet header to transmit the internet packets toward their destinations. The selection of a path for transmision is called routing. Routing is not a topic discussed by the internet protocol (at least not this version of it).

The internet modules use fields in the internet header to fragment and reassemble internet packets when necessary for transmission through "small packet" networks.

The model of operation is that an internet module resides in each host engaged in internet communication, and in each gateway that interconnects networks. These modules share common rules for interpreting address fields and for fragmenting and assembling internet packets. In addition these modules (especially in gateways) may have procedures for making routing decisions, and other functions.

The internet protocol does not provide a reliable communication facility. There are no acknowledgements either end-to-end or hop-by-hop. There is no error control, e.g. no checksums. There are no retransmissions. There is no flow control.

The internet protocol treats each internet segment as an independent entity unrelated to any other internet segment. There are no connections, or logical circuits.

2. PHILOSOPHY

2.1. History - Lessons Learned

???

2.2. Related Work

???

2.3. Mechanisms Explained

Addressing

A distinction is made between names, addresses, and routes. A name indicates what we seek. An address indicates where it is. A route indicates how to get there. The internet protocol deals only with addresses. It is the task of higher level (i.e. host-to-host or application) protocols to make the mapping from names to addresses. It is the task of lower level (i.e. local net or gateways) procedures to make the mapping from addresses to routes.

Addresses are variable length in multiples of octets, the source and destination addresses may be of different lengths. In this version of the internet protocol an address begins with an one octet network number. Following the network number the address is composed of fields appropriate to the specified network.

For example, in the ARPANET the network number is to be followed by an one octet IMP number and that followed by a two octet host number. It is resonable for individual hosts to specify additional address fields to distinguish different protocol services and applications.

Fragmentation

Fragmentation of an internet segment may be necessary when it originates in a local net that allows a large packet size, and must traverse a local net that limits packets to a smaller size, to reach its destination.

This protocol expects that all networks in the interconnected internet system will carry packets of at least 126 octets (1008 bits). While the total length field of the internet header allows for a maximum internet segment length of 2**16-1 octets, the fragmentation procedure limits the maximum internet segment size to 2**14-1 octets. This is because the fragmentation unit is 64 octets (2**6) and the maximum number of fragments is 256 (2**8).

An internet segment can be marked "don't fragment". Any internet segment so marked is not to be fragmented under any circumstances. If such an internet segment can not be delivered to its destination without fragmenting it, it is to be discarded instead.

The internet protocol fragmentation procedure utilizes information in three fields of the internet header: the identification, the not-end-of-segment-flag, and the fragment offset.

The sender of an internet segment sets the identification field to a value that must be unique for that source-destination pair using that format for the time the segment will be active in the internetwork system. The originator of a complete segment forces the not-end-of-segment-flag to zero, and the fragment offset to zero.

To fragment a long internet packet, an internet protocol module (for example in a gateway), creates two new internet packets and copies the contents of the internet header fields from the long packet into both new internet headers. The data of the long packet is divided into two portions on a 64 octet boundary (the second portion might not be an even multiple of 64 octets, but the first must be). Call the number of 64 octet blocks in the first portion NFB. The first portion of the data is placed in the first new internet packet and the total length field is set to the correct value. The not-end-of-segment-flag is set to one. The second portion of the data is placed in the second new internet packet and the total length field is set to the correct value. The not-end-of-segment-flag carries the same value as the long packet. The fragment offset field of the second new internet packet is set to the value of that field in the long packet plus NFB.

This procedure can be generalized for an n-way split, rather than the two-way split described.

To assemble the fragments of an internet segment an internet module (for example at a destination host) combines internet packets that all have the same value for the four fields: identification, format, destination, and source. The combination is done by placing the data portion of each fragment in the relative position indicated by the fragment offset in that fragment's internet header. The first fragment will have the fragment offset zero, and the last fragment will have the not-end-of-segment-flag reset to zero.

2.4. Functional Specification of Interfaces

???

```
+------+ +-----+ +-----+       +-----+
|Telnet| | FTP | |Voice|  ...  |     |
+------+ +-----+ +-----+       +-----+
    |   |         |              |
   +-----+     +-----+         +-----+
   | TCP |     | RTP |  ...    |     |
   +-----+     +-----+         +-----+
      |           |              |
   +-------------------------------+
   |       Internet Protocol       |
   +-------------------------------+
                  |
     +---------------------------+
     |   Local Network Protocol  |
     +---------------------------+
```

Protocol Relationships

Figure 1.

2.5. Problems Remaining

Internet Packet Corruption

The detection of erronous internet packets at a gateway or destination internet module would be useful. This would prevent the useless and resource consuming forwarding of corrupted data, or (even worse) acting on corrupted internet header fields.

Short Form Addresses

The addressing procedure of this version of the internet protocol requires that each address begin with the network field, it might be useful to allow short form of addressing when an internet message is to be delivered within a smaller environment.

2.6. Future Directions

???

2.7. Examples & Scenarios

???

3. SPECIFICATION

3.1. Formalisms Explained

???

3.2. Formal Specification

???

3.3. Internetwork Header Format

A terse descriptive summary of the contents of the internetwork header follows:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |   FLAGS   |Fragment Offset|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Format     |  DAL  |  SAL  |          Destination          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Destination continued     |            Source             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Source cont.  |            Options            |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Example Internet Packet Header

Note that each tick mark represents one bit position.

Figure 2.

About Addresses:

An address is a variable length quantity (in multiples of octets). It is intended for the first octet of an address to be interpreted as a network identifier, and that the rest of the address identifies a host within that network. The address field is allowed to be even longer than that with the view that a host may multiplex between several functions, or further route messages based on the additional address bits.

If a host were to support two instances of TCP they could be assigned distinct addresses by using an additional octet of address beyond that needed to identify the host. Other examples of such processes are the XNET (cross-network debugger) server process, the gateway control process, or the packet echoer process. There is also the possibility of placing another whole layer of addressing hierarchy in this position.

The 8 bit network number, which is the first octet of the variable length address, has a value as specified in RFC 739 [7]. In any case the latest information can be obtained from Jon Postel.

Version: 4 bits

There is a Version field which indicates the "shape", or format, of the internet portion. This is version 2.

IHL: 4 bits

Internet Header Length is the length of the internet header in 32 bit words, and thus points to the beginning of the data. (In the example of Figure 2 the value of this field is 5.)

Type of Service: 8 bits

Type of service.

Bits 0-6: To be defined later.
Bit 7: Don't Fragment This Segment (DF).

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|. . . . . . . D|
|. . . . . . . F|
+-+-+-+-+-+-+-+-+
```

Total Length: 16 bits

Total Length is the length of the packet in octets including internet header and data.

Identification: 16 bits

An identifying value assigned by the sender to aid in assembling the fragments of a segment.

FLAGS: 8 bits

Various Control Flags.

Bits 0-6: To be defined later.
Bit 7: Not End of Segment Flag (NES).

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|. . . . . . . N|
|. . . . . . . E|
|. . . . . . . S|
+-+-+-+-+-+-+-+-+
```

Fragment Offset: 8 bits

This field indicates where in the segment this fragment belongs. The fragment offset is measured in units of 64 octets (256 bits).

Format: 8 bits

There is a Format field which indicates the "shape", or format, of the data portion of the packet is (examples of format values are "TCP-3", "TCP-4", "DSP"). The format field has values as specified in RFC 739 [7].

DAL: 4 bits

Destination Address Length in octets. (In the example of Figure 2 this field has the value 3.)

SAL: 4 bits

Source Address Length in octets. (In the example of Figure 2 this field has the value 2.)

Destination: variable

The destination address, DAL octets in length.

Source: variable

The source address, SAL octets in length.

Options: variable

The options field may contain several options, and each option may be several octets in length. All options have the same basic format:

Option length: 8 bits

Length in octets (including the two octets of length and name).

Option name: 8 bits

There are two special cases for options.

The first is an option whose length field is zero. This marks the end of the option list. Only one octet is associated with this option, the length octet itself.

The second is an option whose length field is one. This option serves as padding and is also one octet long. This option does not terminate the option list.

Note that the list of options may be shorter than the header length field might imply. The content of the header beyond the end-of-option mark must be padding.

Currently defined options include:

Name	Length	Meaning
--	0	End of option list, occupies one octet only.
--	1	Padding, occupies one octet only.
000	---	Reserved.
200	---	Reserved.
204	variable	Internetwork timestamp field; used to accumulate timestamping information during internet transit.
205	variable	Satellite timestamp field; (as above).

Padding: variable

The Padding field is used to ensure that the data begins on 32 bit word boundary. The padding is zero.

4. VERIFICATION

Requires further research.

5. IMPLEMENTATION

5.1. What not to leave out.

???

5.2. User Interfaces

???

5.3. Mechanisms

???

5.4. Data Structures

???

5.5. Program sizes, performance data.

???

5.6. Test sequences, procedures, exerciser.

???

5.7. Parameter values: timeouts, segment sizes, buffer strategies.

???

5.8. Debugging

???

REFERENCES

[1] Vinton G. Cerf and Robert E. Kahn, "A Protocol for Packet Network Intercommunication," IEEE Transactions on Communications, volume COM-22, No. 5, May 1974, p. 637-648. (An early version of this paper appeared as INWG General Note 39, IFIP Working Group 6.1, September 1973).

[2] Vinton G. Cerf, Yogen K. Dalal, Carl Sunshine, "Specification of Internet Transmission Control Program," INWG General Note 72, IFIP Working Group 6.1, December 1974.

[3] Jonathan B. Postel, Larry L. Garlick, Raphael Rom, "Transmission Control Protocol Specification," Augmentation Research Center, Stanford Research Institute, Menlo Park, CA, 15 July 1976.

[4] Vinton G. Cerf, "Specification of Internet Transmission Control Program - TCP (Version 2)," March 1977.

[5] Vinton G. Cerf and Jonathan B. Postel, "Specification of Internetwork Transmission Control Program - TCP Version 3," Information Sciences Institute, January 1978.

[6] Bolt Beranek and Newman, "Specification for the Interconnection of a Host and an IMP," BBN technical Report 1822, January 1976 (Revised).

[7] J. Postel, "Assigned Numbers," RFC 739, NIC 42341, 11 November 1977.

GLOSSARY

1822
: BBN Report 1822, "The Specification of the Interconnection of a Host and an IMP". The specification of interface between a host and the ARPANET.

Address
: An address is a variable length quantity (in multiples of octets).

ARPANET message
: The unit of transmission between a host and an IMP in the ARPANET. The maximum size is about 1012 octets (8096 bits).

ARPANET packet
: A unit of transmission used internally in the ARPANET between IMPs. The maximum size is about 126 octets (1008 bits).

DAL
: Destination Address Length, an internet header field, which specifies the destination address length in octets.

Destination
: The (variable length) destination address, an internet header field.

DF
: The Don't Fragment bit carried in the type of service field.

FLAGS
: An internet header field carrying various control flags.

Format
: The internet header Format field, which indicates the format of the data portion of the internet segment.

fragment
: A portion of a logical unit of data, in particular an internet fragment is a portion of an internet segment.

Fragment Offset
: This internet header field indicates where in the internet segment this fragment belongs.

header

Control information at the beginning of a message, segment, packet or block of data.

Identification

An internet header field identifying value assigned by the sender to aid in assembling the fragments of a segment.

IHL

The internet header field Internet Header Length is the length of the internet header measured in 32 bit words.

IMP

The Interface Message Processor, the packet switch of the ARPANET.

internet fragment

A portion of the data of an internet segment with an internet header.

internet packet

Either an internet segment or an internet fragment.

internet segment

The unit of data exchanged between an internet module and the higher level protocol to gether with the internet header.

leader

Control information at the beginning of a message or block of data. In particular, in the ARPANET, the control information on an ARPANET message at the host-IMP interface.

module

An implementation, usually in software, of a protocol or other procedure.

NES

The Not-End-of-Segment Flag carried in the internet header FLAGS field.

NFB

The Number of Fragment Blocks in a portion of an internet packet. That is, the length of a portion of data measured in 64 octet units.

not-end-of-segment-flag

A flag indicating whether or not this internet packet contains the end of an internet segment, carried in the internet header FLAGS field.

octet

An eight bit byte.

Options

The internet header Options field may contain several options, and each option may be several octets in length. The options are used primarily in testing situations, for example to carry timestamps.

packet

A package of data with a header which may or may not be logically complete. More often a physical packaging than a logical packaging of data.

Padding

The internet header Padding field is used to ensure that the data begins on 32 bit word boundary. The padding is zero.

RTP

Real Time Protocol: A host-to-host protocol for communication of time critical information.

SAL

Source Address Length, an internet header field, which specifies the source address length in octets.

segment

A logical unit of data, in particular an internet segment is the unit of data transfered between the internet module and a higher level module.

Source

The (variable length) source address, an internet header field.

TCP

Transmission Control Protocol: A host-to-host protocol for reliable communication in internetwork environments.

Total Length

The internet header field Total Length is the length of the packet in octets including internet header and data.

Type of Service

An internet header field which indicates the type of service for this internet fragment.

Version

The Version field indicates the format of the internet header.

XNET

A cross-net debugging protocol.