



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Security Essentials: Network, Endpoint, and Cloud (Security 401)"
at <http://www.giac.org/registration/gsec>

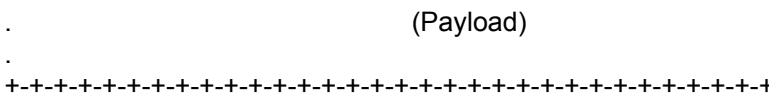


Figure 1 IPv4 Datagram Packet Header Format

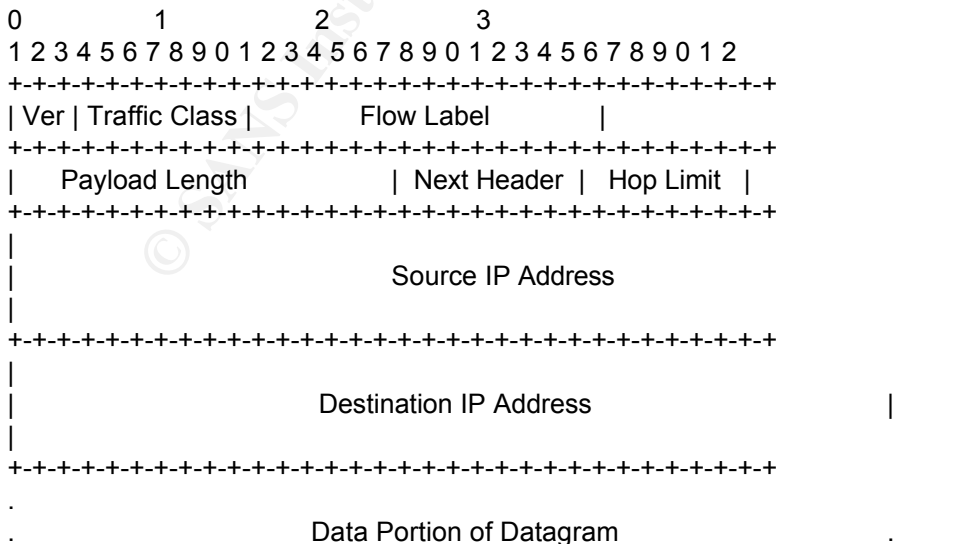
Flags and fragment offset fields work together to describe the specific location of the fragment within the original packet. One bit within the flag field indicates whether or not the packet may be fragmented. The fragment offset is a 13-bit field that displays what is known as the fragment block. The original datagram packet is broken into 8-bit blocks known as fragment blocks. The 13-bit field displays this fragment block value from 0 to 8191, which directly corresponds to a 0 to 65,528-bit offset from the original datagram packet. Another bit within the flag field shows if this is the last fragment of the original datagram or more come after this particular block.

With the present options available to IP version 4 the internet has proven very reliable for data transfer up through the 1980s and 1990s yet IPv4 has become less efficient as the internet has continued to expand into the new century. IPv4 was not designed to manage the extent of nodes it supports today. As the personal computer market has expanded in the past decade so has the requirement for a larger internetworking solution for TCP/IP. Over the past 5 years there has been efforts to create a next generation TCP/IP to incorporate requirements of today's internet. IPv6 offers many more distinctive attributes.

IPv6 Fragmentation / Enhancements for Packet Transfer

The actual operation process of IPv6 is the same as IPv4. In order to phase IPv6 into operation IPv6 can be utilized within an IPv4 environment. In order to anticipate an expanding internet IPv6 has included simplification of the IP header in order to realize increased operability from the internet. The fragmentation handling process with IPv6 has been simplified to create less packet processing required in transit. A brief description of IPv6 extension headers is required here for clarification on IPv6 fragmentation policy.

IPv4 utilizes standard header fields that offer the consistent options described above. All routers and hosts along the path as well as the destination host still must process all the fields within the IPv4 header whether or not they are utilized. IPv6 utilizes a slightly different model for its datagram header and IP processing than IPv4; it has fewer fields and is a standard size. Figure 2 depicts the fields required for an IPv6 datagram packet.



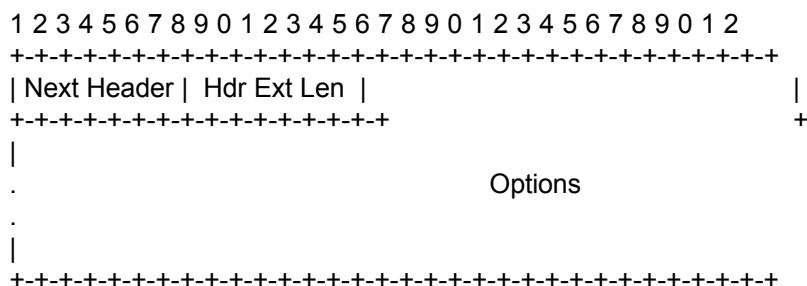


Figure 4 Standard hop-by-hop / destination Extension Header Format

Thus, the concept of datagram packet fragmentation for IPv6 is managed fundamentally different than for IPv4. IPv4 packets can be fragmented at any intermediate node along the path that does not allow packets as large as the original packet size. On the other hand, IPv6 allows fragmentation only at the originating or source node. With IPv6 the fragmentation option is utilized as an extension header with its own format.

Figure 5 portrays the fragmentation header format. The next header field (8 bits) is the format of the subsequent header field. The next field (8 bits) of the fragmentation header is reserved for future use. The fragment offset (13 bits) value tells the destination numbered in 8-bit segments where this portion fits within the fragmentable portion of the packets. The fragmentable portion includes only the payload and extension headers that are to be processed when the packet has arrived at its final destination. The next field is another field reserved (2 bits) for future use. The next field is the M flag which when the value is 1 indicates another fragment is forthcoming, a zero indicates this is the last fragment. The final identification field holds a 32-bit identifier that is intended to uniquely identify any packet sent recently.

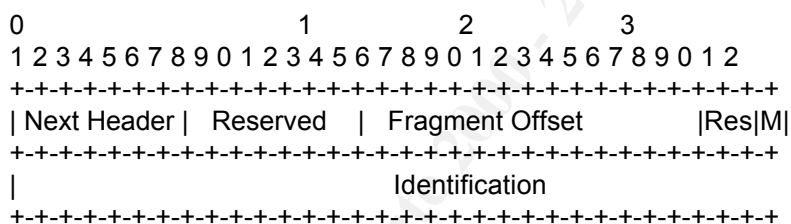


Figure 5 Fragmentation Header Format

Path MTU Discovery IPv4 / IPv6

Since fragmentation within IPv4 can potentially hamper throughput, and fragmentation in IPv6 requires path mapping, path Maximum Transmission Unit (MTU) discovery is imperative for proper operation of IP networks. Simply put, the path MTU discovery is a process where the originating node sends a packet to the destination. Along the path it checks the largest allowable MTU size of each node and delivers this information back to the originating node prior to releasing any datagram packets to the destination. The packets now sent from source to destination will be no larger than the intermediate nodes with the lowest capability of MTU size discovered through the MTU discovery process.

For IPv4 path MTU discovery was implemented through [RFC 1063](#) and later obsoleted by [RFC 1191](#). In order to utilize MTU discovery within IPv4 the *don't fragment* bit in the "flag" field must be set. The size of the packet sent is the assumed MTU, which equals the known MTU of the first hop. During relay of the packet if fragmentation is required at any node due to the size of the packet, the packet is discarded and an ICMP stating the destination is unreachable due to fragmentation required but DF (don't fragment) field set. The source sends another packet of smaller size. The path MTU discovery process is continued until the packet is

delivered without an ICMP reply. In short, path MTU discovery for IPv4 increases throughput of traffic between hosts on different LANs. Note that if the DF flag is not set then the packets will be fragmented along the path unbeknownst to the source host.

[RFC 1981](#) specifically suggests that “IPv6 nodes SHOULD implement Path MTU Discovery in order to discover and take advantage of paths with PMTU greater than the IPv6 minimum link MTU.” Since IPv6 does not allow fragmentation along the path, path MTU discovery is implemented by sending packets of varying size for the sole purpose of discovering the largest possible packet for a particular path between a source and destination. Note that when the source host has not implemented path MTU discovery, packets sent out that are too large and illicit an ICMPv6 reply will be rectified at the source node by its sending smaller packets. The difference here with MTU discovery is that the source node will not send these packets out only to map the packet’s path to destinations.

The [Computer Science Department of the Tel Aviv University](#) has suggested an addition to the IPv6 protocol to alleviate some of the shortcomings of the PMTU discovery algorithm. This addition to the IPv6 protocol would be the use of a new hop by hop extension headers known as a Detection Option and a destination extension header known as Indication Option. Both the Detection Option and Indication Option headers consist of three fields, option type, option data and affirmative PMTU.

For an initial detection of PMTU by a source node a Detection Option header would be sent within an IPv6 packet of minimum MTU size (576 bytes). The affirmative PMTU field within the Detection Option header would take on the value of the first hop link MTU value. Since it is a hop by hop extension header it will be processed at each node along the path. Each subsequent node along the path would compare the affirmative PMTU field with the next hop link and replace the affirmative PMTU field with the lower of the two values.

Once at the destination the affirmative PMTU field would hold the maximum size packet able to transverse the particular path. This value would be then be stored statically within the destination node. The destination node would then send an Indication Option header with the same data back to the source node. Since the Indication Option header is a destination extension header only the source and destination nodes would process the data within its fields.

Conclusion

Internet Protocol is the backbone of the world Internet. Because it is impossible to standardize capabilities of all the hosts and routers around the world packet fragmentation policies are necessary within IP standards to maintain structure. IPv4 and IPv6 were compared here to show their different strategies. Yet, both of these fragmentation policies expose hosts to potential service attacks through ICMP replies. It is important to investigate possibilities such as Option headers within the IPv6 protocol to subjugate the deficiencies of existing fragmentation policies in order to begin limiting Information Assurance attacks on our operational networks.

Works Cited

Feit, S., TCP/IP Signature Edition, McGraw-Hill, 1998.

Deering, S., Hinden, R., "Internet Protocol, Version 6 (IPv6) Specification," Request for Comments 2460, December 1998, URL:

<http://www.ietf.cnri.reston.va.us/rfc/rfc2460.txt?number=2460> (17 Nov 2000).

Loshin, P., IPv6 Clearly Explained, Morgan Kaufmann, 1999.

Mogul, K., Kent, C., Partridge, C., McCloghrie, K., "IP MTU Discovery Options," Request for Comments 1063, July 1988, URL: <http://www.ietf.cnri.reston.va.us/rfc/rfc1063.txt?number=1063> (20 Nov. 2000).

Mogul, K., Deering, S., "Path MTU Discovery," Request for Comments 1191, November 1990, URL: <http://www.ietf.cnri.reston.va.us/rfc/rfc1191.txt?number=1191> (18 Nov. 2000).

McCann, J., Deering, S., Mogul, K., "Path MTU Discovery for IP version 6," Request for Comments 1981, August 1996, URL: <http://cellworks.washington.edu/sage/1997/06/rfc1981.txt> (18 Nov. 2000).

Kreidenko, V., Burtman, D., "Protocols & Computer Networks Project Path MTU Discovery for IP version 6," URL: <http://www.rad.com/networks/1997/mtudisc/project.html> (13 Nov. 2000).

© SANS Institute 2000 - 2005. All rights reserved.