6man Working Group   S. Krishnan
Internet-Draft   Ericsson
Updates: 2460 (if approved)   July 2, 2009
Intended status: Standards Track
Expires: January 3, 2010

Handling of overlapping IPv6 fragments
draft-ietf-6man-overlap-fragment-03

Status of this Memo

Copyright Notice

Abstract

The fragmentation and reassembly algorithm specified in the base IPv6 specification allows fragments to overlap. This document demonstrates the security issues with allowing overlapping fragments and updates the IPv6 specification to explicitly forbid overlapping fragments.

Table of Contents

1. Introduction

Fragmentation is used in IPv6 when the IPv6 packet will not fit inside the path MTU to its destination. When fragmentation is performed an IPv6 node uses a fragment header as specified in section 4.5 of the IPv6 base specification [RFC2460] to break down the datagram into smaller fragments that will fit in the path MTU. The destination node receives these fragments and reassembles them. The algorithm specified for fragmentation in [RFC2460] does not prevent the fragments from overlapping, and this can lead to some security issues with firewalls [RFC4942]. This document explores the issues that can be caused by overlapping fragments.

1.1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL","SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Overlapping Fragments

Commonly used firewalls use the algorithm specified in [RFC1858] to weed out malicious packets that try to overwrite parts of the transport layer header to bypass inbound connection checks. [RFC1858] prevents an overlapping fragment attack on an upper layer protocol (in this case TCP) by recommending that packets with fragment offset 1 be dropped. While this works well for IPv4 fragments, it will not work for IPv6 fragments. This is because the fragmentable part of the IPv6 packet can contain extension headers before the TCP header, making this check less effective.

3. The attack

This attack describes how a malicious node can bypass a firewall using overlapping fragments. Consider a sufficiently large IPv6 packet that needs to be fragmented.

```
   +------------------+--------------------//-----------------------+
   |  Unfragmentable  |                 Fragmentable                |
   |       Part       |                     Part                    |
   +------------------+--------------------//-----------------------+
```

Figure 1: Large IPv6 packet

This packet is split into several fragments by the sender so that the packet can fit inside the path MTU. Let's say the packet is split into two fragments.

```
   +------------------+--------+--------------------+
   |  Unfragmentable  |Fragment|       first        |
   |       Part       | Header |      fragment      |
   +------------------+--------+--------------------+

   +------------------+--------+--------------------+
   |  Unfragmentable  |Fragment|       second       |
   |       Part       | Header |      fragment      |
   +------------------+--------+--------------------+
```

Figure 2: Fragmented IPv6 packet

Consider the first fragment. Let's say it contains a destination options header (DOH) 80 octets long and is followed by a TCP header.

```
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<==FH
 |NextHdr=DOH(60)|   Reserved    |   FragmentOffset = 0    |Res|1|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                Identification=aaaabbbb                        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<==DOH
 |NextHdr=TCP(6) | HdrExtLen = 9 |                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
 |                                                               |
 .                                                               .
 .                            Options                            .
 .                                                               .
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<==TCP
 |        Source Port            |       Destination Port        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Sequence Number                         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                    Acknowledgment Number                      |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Offset| Reserved  |U|A|P|R|S|F|           Window              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: First Fragment

The TCP header has the following values of the flags S(YN)=1 and A(CK)=1. This may make an inspecting stateful firewall think that it is a response packet for a connection request initiated from the trusted side of the firewall. Hence it will allow the fragment to pass. It will also allow the following fragments with the same Fragment Identification value in the fragment header to pass through.

A malicious node can form a second fragment with a TCP header that changes the flags and sets S(YN)=1 and A(CK)=0. This can change the packet on the receiving end to consider the packet as a connection request instead of a response. By doing this the malicious node has bypassed the firewall's access control to initiate a connection request to a node protected by a firewall.

```
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<==FH
 |NextHdr=DOH(60)|   Reserved    |   FragmentOffset = 10   |Res|0|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                Identification=aaaabbbb                        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<==TCP
 |        Source Port            |       Destination Port        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Sequence Number                         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                    Acknowledgment Number                      |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Offset| Reserved  |U|A|P|R|S|F|           Window              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Second Fragment

Note that this attack is much more serious in IPv6 than in IPv4. In IPv4 the overlapping part of the TCP header did not include the source and destination ports. In IPv6 the attack can easily work to replace the source or destination port with an overlapping fragment.

4. Recommendation

IPv6 nodes transmitting datagrams that need to be fragmented MUST NOT create overlapping fragments. When reassembling an IPv6 datagram, if one or more its constituent fragments is determined to be an overlapping fragment, the entire datagram (and any constituent fragments, including those not yet received), MUST be silently discarded.

5. Security Considerations

This document discusses an attack that can be used to bypass IPv6 firewalls using overlapping fragments. It recommends disallowing overlapping fragments in order to prevent this attack.

6. Acknowledgements

The author would like to thank Thomas Narten, Doug Montgomery, Gabriel Montenegro, Remi Denis-Courmont, Marla Azinger, Arnaud Ebalard, Seiichi Kawamura, Behcet Sarikaya, Vishwas Manral, Christian Vogt, and Alfred Hoenes for their reviews and suggestions that made this document better.

7. IANA Considerations

This document does not require any action from the IANA.

8. Normative References

[RFC1858] Ziemba, G., Reed, D., and P. Traina, "Security Considerations for IP Fragment Filtering", RFC 1858, October 1995.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC4942] Davies, E., Krishnan, S., and P. Savola, "IPv6 Transition/Co-existence Security Considerations", RFC 4942, September 2007.

Author's Address

Suresh Krishnan
Ericsson
8400 Blvd Decarie
Town of Mount Royal, Quebec
Canada

Email: suresh.krishnan@ericsson.com