HIP Working Group
Internet-Draft
Intended status: Experimental
Expires: January 7, 2010

G. Camarillo
A. Keranen
Ericsson

July 6, 2009

Host Identity Protocol (HIP) Multi-hop Routing Extension
draft-camarillo-hip-via-00.txt

Status of this Memo

Copyright Notice

Abstract

This document specifies two extensions to HIP to implement multi-hop routing. The first extension allows a HIP packet to carry the list of hosts that forwarded it. The second extension allows implementing

source routing in HIP. That is, a host sending a HIP packet can define a set of hosts that the HIP packet should traverse.

Table of Contents

1. Introduction

When HIP [RFC5201] is used in certain contexts (e.g., in a HIP BONE [I-D.ietf-hip-bone] overlay), hosts need the ability to perform source routing. That is, a host needs the ability to send a HIP packet that will traverse a set of hosts before reaching its destination. This document defines an extension that provides HIP with this functionality.

Additionally, when HIP packets are routed through multiple hosts, some of these hosts (e.g., the destination host) need the ability to know the hosts a particular packet traversed. This document defines another extension that provides HIP with this functionality.

These two extensions enable multi-hop routing in HIP. Before these extensions were specified, HIP only supported a single intermediate host (i.e., a rendezvous server [RFC5204]) between the source of a HIP packet and its destination.

2. Terminology

2.1. Requirements Language

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 RFC 2119 [RFC2119].

2.2. Definitions

Route list: A list of HITs of the hosts that a HIP packet has traversed or should traverse.

Symmetric routing: A response to a message is routed back using the same set of intermediary nodes as the original message used, except in reversed order.

3. Overview of Operations

TODO: this will be a non-normative section that will give a high-level overview about how these extensions work.

4. Protocol Definitions

4.1. Creating and Processing Via Route Lists

When a host sending a HIP packet needs to record the hosts that are on the path that the HIP packet traverses, it includes an empty ROUTE_VIA parameter to the packet.

A host that receives a packet with a ROUTE_VIA parameter adds its own HIT to the end of the ROUTE_VIA parameter, unless it is the receiver of the packet. If the host uses a different HIT on the HIP association it used for receiving the packet than for sending it forward, it should also add the receiving HIT to the route list before the sending HIT.

If the host is the receiver of the packet, and the received packet generates a response HIP packet, the host checks the SYMMETRIC flag from the ROUTE_VIA parameter. If the SYMMETRIC flag is set, the host MUST create a ROUTE_DST parameter from the ROUTE_VIA parameter as described in Section 4.2 and include it in the response packet. Also, if an intermediary host generates a new HIP packet (e.g, an error NOTIFY packet) due to a HIP packet that had a ROUTE_VIA parameter with SYMMETRIC flag set, and the new packet is intended for the sender of the original HIP packet, the host MUST construct and add a ROUTE_DST parameter into the new packet as in the previous case.

4.2. Creating Destination Route Lists

A host can add a ROUTE_DST parameter to a new HIP packet when it needs to define the hosts that should be on the path the HIP packet

traverses. The host may either decide the path independently, or it may create the path based on a ROUTE_VIA parameter. Only the originator of a signed HIP packet can add a ROUTE_DST parameter to the HIP packet since the parameter is covered by the signature.

When a host creates a ROUTE_DST parameter due to receiving a packet with a ROUTE_VIA parameter, it copies all the HITs in the ROUTE_VIA parameter to the ROUTE_DST parameter, but in reversed order. This results in HIP response packet being forwarded using the same set of hosts as the packet for which the response was generated for.

4.3. Processing Destination Route Lists

When a host receives a HIP packet that contains a ROUTE_DST parameter, it first looks up its own HIT from the route list. If host's own HIT is not in the list and the host is not the receiver of the packet, the packet was incorrectly forwarded and MUST be dropped. Next hop for the packet is the HIT after host's own HIT in the list. If the host's HIT was the last HIT in the list, the next hop is the

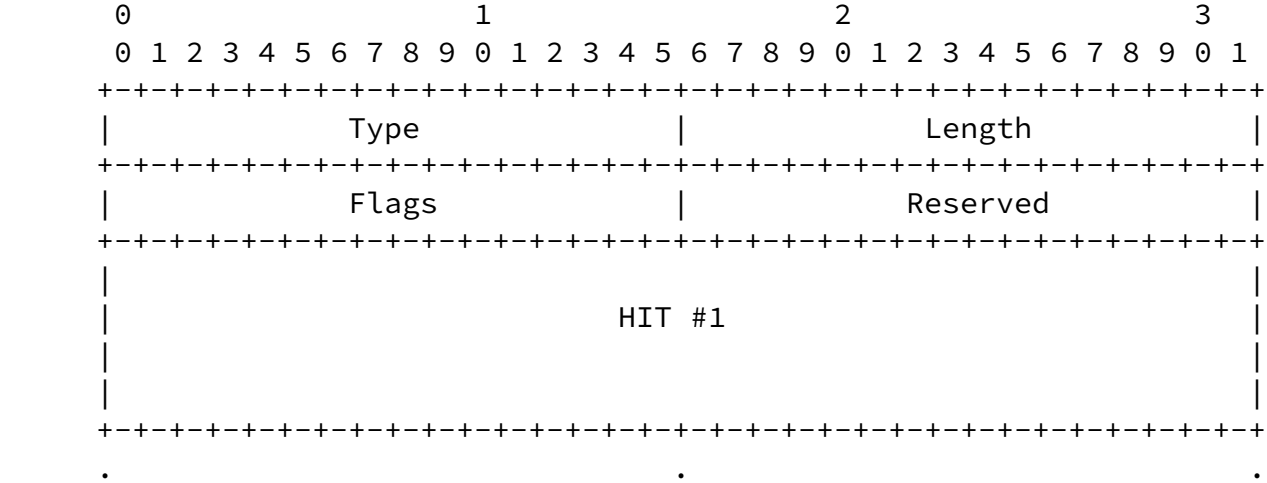
receiver's HIT in the HIP header.

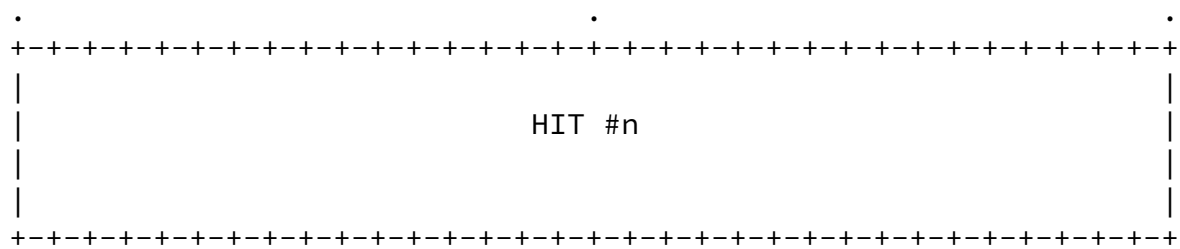
5. Packet Formats

This memo defines two new HIP parameters that are used for recording a route via multiple hosts (ROUTE_VIA) and to define a route a packet should traverse by the sender of the packet (ROUTE_DST).

The ROUTE_DST parameter is integrity protected with the signature (where present) but ROUTE_VIA is not so that intermediary hosts can add their own HITs to the list. Both parameters have critical type (as defined in Section 5.2.1 of [RFC5201]) since the packet will not be properly routed unless all hosts on path recognize the parameters.

5.1. Source and Destination Route List Parameters






```
Type      [ TBD by IANA
          ROUTE_DST: 971
          ROUTE_VIA: 65525 ]
Length    length in octets, excluding Type and Length
          (i.e., number-of-HITs * 16 + 4)
Flags     bit flags that can be used for requesting special
          handling of the parameter
Reserved  reserved for future use
HIT       Host Identity Tag of one of the hosts on the path
```

Figure 1: Format of the ROUTE_VIA and ROUTE_DST parameters

Figure 1 shows the format of both ROUTE_VIA and ROUTE_DST parameters. The ROUTE_DST parameter, if present, MUST have at least one HIT, but the ROUTE_VIA parameter can also have zero HITs. Both can contain at most 32 HITs. The Flags field is used for requesting special handling for certain Route lists. The flags defined in this document are shown in Table 1. The Reserved field can be used by future extensions; it MUST be zero when sending and ignored when receiving this parameter.

Pos	Name	Purpose
0	SYMMETRIC	The response packet MUST be sent using ROUTE_DST list made from this ROUTE_VIA list, i.e., using symmetric routing.

Table 1: Bit flags in ROUTE_VIA parameter

The "Pos" column in Table 1 shows the bit position of the flag (as in Figure 1) in the Flags field, "Name" gives the name of the flag used in this document, and "Purpose" gives brief description of the meaning of that flag.

6. IANA Considerations

This section is to be interpreted according to [RFC5226].

This document updates the IANA Registry for HIP Parameter Types [RFC5201] by assigning new HIP Parameter Type values for the new HIP Parameters: ROUTE_VIA and ROUTE_DST (defined in Section 5).

7. Security Considerations

7.1. Forwarding Loops

A malicious host could craft a destination route list that contains the same HIT more than once and thus create a forwarding loop. Since the IP layer TTL is decremented on each hop, the loop will be eventually broken, but hosts may additionally protect themselves against this attack by checking that their own HIT is in the destination list only once and drop invalid packets.

8. References

8.1. Normative References

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

[RFC5201] Moskowitz, R., Nikander, P., Jokela, P., and T. Henderson, "Host Identity Protocol", RFC 5201, April 2008.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an

IANA Considerations Section in RFCs", BCP 26, RFC 5226,

May 2008.

8.2. Informative References

[RFC5204] Laganier, J. and L. Eggert, "Host Identity Protocol (HIP) Rendezvous Extension", RFC 5204, April 2008.

[I-D.ietf-hip-bone]
Camarillo, G., Nikander, P., Hautakorpi, J., and A. Johnston, "HIP BONE: Host Identity Protocol (HIP) Based Overlay Networking Environment", draft-ietf-hip-bone-01 (work in progress), March 2009.

Authors' Addresses

Gonzalo Camarillo
Ericsson
Hirsalantie 11
02420 Jorvas
Finland

Email: Gonzalo.Camarillo@ericsson.com

Ari Keranen
Ericsson
Hirsalantie 11
02420 Jorvas
Finland

Email: ari.keranen@ericsson.com