Middlebox Traversal Issues of Host Identity Protocol (HIP) Communication
draft-stiemerling-hip-nat-03

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of Section 3 of RFC 3667. By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with RFC 3668. This document may not be modified, and derivative works of it may not be created, except to publish it as an RFC and to translate it into languages other than English.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on August 24, 2005.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

The Host Identity Protocol (HIP) fundamentally changes the way in which two Internet hosts communicate. One key advantage over other
schemes is that HIP does not require modifications to the traditional network-layer functionality of the Internet, i.e., its routers. In the current Internet, however, many devices other than routers modify the traditional network-layer behavior of the Internet. These "middleboxes" are intermediary devices that perform functions other than the standard functions of an IP router on the datagram path between source and destination hosts. Whereas some types of middleboxes may not interfere with HIP at all, others can affect some aspects of HIP communication and others can render HIP communication impossible. This document discusses the problems associated with HIP communication across network paths that include middleboxes. It pinpoints issues in the current HIP specifications that are the causes of problems for communicating across middleboxes.

Table of Contents

1. Introduction ............................................................... 3
2. HIP Across Middleboxes ................................................ 4
   2.1 Phase 1: HIP Base Exchange ....................................... 4
   2.1.1 IPv6 Base Exchange ............................................ 4
   2.1.2 IPv4 Base Exchange ............................................ 4
   2.2 Phase 2: IPsec Data Exchange ..................................... 5
   2.3 HIP across Twice-NAT ............................................... 5
3. Extensions to HIP ......................................................... 7
4. Extension to NATs ......................................................... 8
5. HIP unaware NATs ....................................................... 9
6. Firewalls ................................................................. 10
7. Security Considerations .............................................. 10
8. Acknowledgements ...................................................... 10
9. References .............................................................. 10
   9.1 Normative References .............................................. 10
   9.2 Informative References ........................................... 11
   Authors’ Addresses ..................................................... 12
   Intellectual Property and Copyright Statements .................. 14
1. Introduction

The current specification of the Host Identity Protocol (HIP) [I-D.ietf-hip-arch] assumes simple Internet paths, where routers forward globally routable IP packets based on their destination address alone. Over such paths, the HIP protocol performs well.

In the current Internet, such pure paths are becoming increasingly rare. For a number of reasons, several types of devices modify or extend the pure forwarding functionality the Internet’s network layer used to deliver. [RFC3234] coins the term middleboxes for such devices: "A middlebox is (...) any intermediary device performing functions other than the normal, standard functions of an IP router on the datagram path between a source host and destination host."

Middleboxes affect communication in a number of ways. For example, they may inspect the flows of some transport protocols, such as TCP, and selectively drop, insert or modify packets. If such devices encounter a higher-layer protocol they do not support, or even a variant of a supported protocol that they do not know how to handle, communication across the middlebox may become impossible for these kinds of traffic.

There are many different variants of middleboxes. The most common ones may be network address translators and firewalls. [RFC3234] identifies many other types of middleboxes. One broad way of classifying them is by behavior. The first group operates on packets, does not modify application-layer payloads and does not insert additional packets. This group includes NAT, NAT-PT, SOCKS gateways, IP tunnel endpoints, packet classifiers, markers, schedulers, transport relays, IP firewalls, application firewalls, involuntary packet redirectors and anonymizers. Other middleboxes exist, such as TCP performance-enhancing proxies, application-level gateways, gatekeepers and session control boxes, transcoders, proxies, caches, modified DNS servers, content and applications distribution boxes, load balancers that divert or modify URLs, application-level interceptors and application-level multicast systems.

Middleboxes can cause two different kinds of communication problems for HIP. They can interfere with the transmission of HIP control traffic or with the transmission of the IPsec’ed HIP data traffic.

This document does not propose a specific solution to the identified issues. It serves as a problem description that separate solution proposals can refer to. It also does not promote the use of any of the discussed types of middleboxes.
2. HIP Across Middleboxes

HIP operates in two phases. It first performs a HIP "base exchange" handshake before starting to exchange application data in the second phase. This section describes the problems that occur in each of the two phases when middleboxes are present along the path from the HIP initiator to the responder.

2.1 Phase 1: HIP Base Exchange


2.1.1 IPv6 Base Exchange

During a HIP base exchange over IPv6, current implementations use plain IPv6 packets without any payload other than the HIP extension header. This approach can cause problems in combination with middleboxes that translate or multiplex IP addresses, such as NATs, and packet filtering firewalls.

NATs typically operate on a combination of IP addresses and TCP or UDP port numbers. They cannot translate packets that do not contain TCP or UDP transport headers — such as the IPv6 HIP base exchange packets. Consequently, IPv6 HIP base exchanges through such NATs will fail and HIP communication cannot occur. (This issue is currently of minor concern, because IPv6 NATs are rare.)

IPv6 packet filtering firewalls are configured following the same filtering policies as IPv4 firewalls would do. A typical configuration is blocking all traffic belonging neither to TCP nor UDP. HIP base exchange packets will be blocked by such IPv6 firewalls since network administrators usually block IPv6 extension headers. In IPv4, network administrators configure firewalls to discard IP packets with IP options set (see [reference.fw_config]).

2.1.2 IPv4 Base Exchange

The HIP base specification suggests to encapsulate the HIP base exchange in IPv4 networks over UDP (see Appendix E in [I-D.ietf-hip-base]). This theoretically allows the HIP base exchange to traverse middleboxes more easily. However, UDP transport suffers from many well-known problems when traversing middleboxes, especially NATs, [RFC3715]. One general problem is that firewalls and NATs often block and discard UDP packets.
Even if middleboxes do allow UDP packets to pass, they may change the UDP port number and/or IP addresses of these packets. HIP over UDP currently mandates using a fixed port number (272) for both source and destination ports of HIP base exchange packets. If middleboxes change these port numbers, the base exchange fails.

2.2 Phase 2: IPsec Data Exchange

HIP uses IPsec to secure the data transmission between two HIP nodes after the base exchange is completed. [RFC3715] discusses issues with IPsec traversal across middleboxes. They fall into three distinct areas: NAT-intrinsic issues, NAT implementation issues and helper incompatibilities. The NAT intrinsic issues related to IPsec (IKE issues are omitted) are discussed in this section.

With ESP-encrypted data traffic, all upper-layer headers are invisible to a NAT. Thus, changes of the IP header during NAT traversal can invalidate upper-layer checksums contained within the ESP-protected payload. HIP hosts already avoid this problem by substituting HITs for IP addresses during checksum calculations [I-D.ietf-hip-base].

Although it is possible to transmit ESP-encrypted packets through a NAT, [RFC3715] notes that the used SPI values have only one-way significance. Thus, although SPIs could be used for demultiplexing different IPsec flows at the NAT, NATs can only observe the SPI values of outgoing ESP flows. They cannot determine the SPI value of the corresponding response traffic. Furthermore, SPI values may collide at the NAT, meaning that several different peers behind the NAT are using the same SPI value at the same time.

[I-D.ietf-ipsec-udp-encaps] describes a method for carrying IPsec traffic through NATs via UDP encapsulation.

2.3 HIP across Twice-NAT

A type of Network Address Translation is not mentioned in previous sections, these so-called twice-NATs (see [RFC2663]) are translating source and destination address at once while a datagram is translated from one address realm into another. Typically, twice-NATs are used when two private address realms have address collisions, for example, if two enterprises merge their networks and both of them are using the same address realm.

Twice-NATs are used in IPv4 networks and in some cases to translate from IPv6 to IPv4 and vice versa (NAT with protocol translation (NAT-PT, [RFC2766]).) This form of NAT is not considered within this memo, since HIP facilitates its own IPv4 to IPv6 mechanism. The
v6ops working group is considering NAT-PT as an experimental technique [I-D.ietf-v6ops-natpt-to-exprmntl].

Figure 1 sketches a scenario where HIP peers A and C are in overlapping address realms, due to address conflicts and HIP peer B is in the public Internet.

Twice-NATs are usually operated with application level support, such as DNS proxies or SIP proxies. These proxies intercept application signaling and configure the twice-NAT middlebox according the request. Configuring means allocating the first IP address in one address realm (e.g., realm1) and a second IP address in another address realm (e.g., realm2). The configuration assures that data from the private realm is sent to one of the twice-NAT’s addresses and afterwards mapped into the other address realm. Furthermore, the proxies will modify the application signaling to reflect the address changes. HIP peer A would see data from HIP peer C actually coming from the twice-NAT’s address in realm1. HIP peer C would see data coming from HIP peer A coming from the twice-NAT’s address in realm2. For HIP peer B data from peer A and B would come from one of the public IP addresses of the twice-NAT. For further studies it is assumed that the twice-NAT is running a n:m translation with n=m and n(realm1)=n(realm2), where n is the number of internal IP addresses in each realm and m the number of public IP addresses. Note that traffic either from peer A or C to peer B does not necessarily be traversed through twice-NAT; traditional NAT or NAPT is sufficient.

While using DNS queries to find another peer’s IP address behind the same twice-NAT, the DNS proxy is configuring the NAT to forward traffic between both peers (peer A and peer C in Figure 3). The HIP base exchange and IPSec traffic should be able to traverse the twice-NAT without any problem since only the address translation is
done, but it should be considered that both source and destination address will be changed. There is port mapping, since the address translation is 1:1 and the multiplexing is done on an IP address base. The IPsec traffic is valid for processing at both peers as long as ESP is used only, but the use of AH will result in broken IPsec checks, since the IP addresses were changed.

The advent of non-DNS based lookup mechanisms (such as described in [I-D.eggert-hiprg-rr-prob-desc]) is challenging for HIP with twice-NATs. Either a new type of proxy for this lookup mechanism must be installed at every twice-NAT device or as mentioned in Section 6 a new signaling protocol between HIP peer and twice-NAT can be used.

3. Extensions to HIP

This section evaluates changes to HIP so that it can traverse middleboxes with less problems. Two issues must be tackled, the reachability of HIP peers behind NATs and traversal of HIP’s base exchange through middleboxes.

Section 2 lists several problems with HIP’s base exchange encapsulation schemes for IPv4 and IPv6. It seems to be more appropriate to run the base exchange of HIP over UDP in general and not to rely on below transport level encapsulation schemes. The uses solves many issues in the presence of NATs and Firewalls. Other middleboxes may treat an UDP encapsulated HIP base exchange more friendly too. For example, load balancer that use IP and transport layer information for their load balancing algorithm. Running the base exchange in two different modes, one in plain IP and in one in UDP encapsulation, is not a viable solution. HIP peers would need to detect if they have to run one or the other mode depending on their network environment and, even worse, on the particular destination HIP peer’s network connection. A HIP peer behind a NAT could communicate with other HIP peers in the public Internet and with HIP peers in the same private IP address space at the same time.

HIP peers located behind NAT must notify other peers about its public reachable IP address and UDP port number (the contact address). Otherwise it is impossible to contact those NAT’ed HIP peers from the Internet. HIP requires an extension to notify other peers about their contact address. A kind of rendezvous point is required where NAT’ed peers can register their current location, meaning storage of their contact address, and an extension of the HIP base exchange is needed. This extension contains the contact address of the NAT’ed peer. The proposed REA packet exchange of [I-D.ietf-hip-mm] can be used for this. The original intention of this extension is support of for end host
mobility and multi-homing. HIP peers can notify by REA about their NAT’ed contact address and their private IP addresses. This behavior is similar to [I-D.ietf-mmusic-anat].

The mentioned rendezvous can be achieved by an extended rendezvous server. There are currently several approaches to this.

However, HIP peers must determine their contact address first before they can announced it within the REA packet to other peers.

4. Extension to NATs

As described in Section 2.2, IPsec SPIs have only one-way significance, meaning that NATs can learn SPI values of outgoing packets, but they cannot learn the corresponding SPI values of incoming packets. Therefore, a matching between SPI values used for outgoing flows to SPI values used for incoming flows is impossible.

NATs must learn the corresponding SPI values for outgoing and incoming IPsec flows. There are two ways in doing so. First, NATs can monitor the HIP base exchange and learn the SPI values agreed by both HIP peers. Afterwards, NATs can remember these values and map outgoing and incoming IPsec flows accordingly.

Second, HIP peers can use a NAT signaling protocol and signal the appropriate SPI values with IP addresses. NATs can so learn the SPI values out of the signaling protocol.

Both approaches require changes to NATs. The first approach would require changes that are only beneficial to HIP, the second one would be beneficial to other protocols as well. Possible solutions for signaling NATs SPI values are NSIS NAT/FW traversal [I-D.ietf-nsis-nslp-natfw] and MIDCOM [RFC3303]. Using MIDCOM in the context of HIP would require some additional knowledge about network topology, for instance, in multihomed environments with different border NATs, host must know which of the multiple NATs to signal for. Therefore, this solution is hard to deploy.

By using the NSIS NAT/FW traversal (NATFW NSLP) mechanism HIP nodes could signal the used SPI values for both directions. NATFW NSLP always ensures that signaling messages will reach an appropriate NAT and those messages follow exactly the data path (path-coupled signaling). NSIS requires usually both ends, i.e., both HIP peers, to support this new signaling protocol. However, NATFW NSLP offers support for only one end supporting this protocol, the so-called proxy mode. HIP peers behind a NATFW NSLP enabled NAT could so configure the local NATs without impacting other networks. An add-on is given through NATFW traversal, too, since on-path firewalls are
configured as well.

Requiring NATs to understand the HIP base exchange itself seems not be appropriate. Requiring a specialized signaling protocol between HIP peers and NAT/firewalls seems not be appropriate. Both require changes to those device would only being beneficial to HIP but no other protocol.

5. HIP unaware NATs

The solutions outlined in Section 4 require that NATs are updated to support new functions, such as HIP itself or NSIS NATFW signaling. By today’s measures, NATs are widely deployed and are currently getting a push through low-cost devices rolled out to broadband connection users, for instance, DSL lines. NATs are deployed in various places, at enterprise network borders, mobile phone networks offering IP-based services, etc. It will be impossible to upgrade or replace all of them to support HIP or HIP-related extensions since so many NATs are deployed. It is the intent of this section to explore how HIP works even in the presence of these HIP unaware NATs. Deployed NATs are currently IPv4 only, so that this section considers them only.

For HIP over IPv4, UDP encapsulated HIP messages solve already some problems in traversing NATs. Usually, UDP packets can traverse NATs from inside (private Network) to outside (public network) and vice versa when they have been initiated from inside. The other way around, initiated from outside, will be blocked or impossible since the destination IP address of the NAT is not known a priory. In the case that UDP encapsulation works fine for the HIP message exchange, IPsec is still troublesome (see [RFC3715].) Some NAT implementations offer some sort of so-called VPN passthrough, where the NAT learns about IPsec flows and tries to correlate outgoing and incoming SPI values. This works probably only for a small numbers of nodes behind a single NAT, until there will be SPI collisions. The solution for running IPsec through NATs is documented in [I-D.ietf-ipsec-udp-encaps] and applicable here, too. HIP should support IPsec over UDP transport through an extension to the signaling. This extension would indicate when to use IPsec over UDP, instead of plain IPsec.

HIP peers should also consider additional NAT traversal mechanisms, such as the widely deployed Universal Plug and Play (UPNP, [reference.UPNP].) UPNP can be used to configure on-link home gateways to forward particular traffic.
6. Firewalls

IP firewalls, usually as known as packet filter firewalls, do inspection on a packet base and decided whether to forward or discard a packet. This type of middlebox is first of all not an obstacle for HIP, as long as the policy rule set defining the filtering allows the HIP base exchange and IPsec traffic to traverse. However, it is common to block traffic with unknown IPv6 extension headers, such as HIP is using, therefore preventing to exchange the HIP base exchange messages. Furthermore, outbound traffic initiated in the protected part is allowed to traverse and corresponding inbound traffic too. On the other hand, traffic initiated from outside and headed inbound is blocked by default. This issue is a problem for the IPsec traffic, since the correlation between outbound IPsec (defined through IP source, IP destination, outbound SPI value) and inbound (defined through IP source, IP destination, inbound SPI value) is impossible to learn for the IP firewall. No correlation is possible, as long as the firewall is unable to learn these corresponding outbound and inbound SPI values.

7. Security Considerations

To be done.

8. Acknowledgements

9. References

9.1 Normative References

[I-D.ietf-hip-arch]
Moskowitz, R., "Host Identity Protocol Architecture",

[I-D.ietf-hip-base]
Moskowitz, R., "Host Identity Protocol",

[I-D.ietf-ipsec-udp-encaps]
Huttunen, A., "UDP Encapsulation of IPsec Packets",

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address
Translator (NAT) Terminology and Considerations",
RFC 2663, August 1999.

[RFC2766] Tsirtsis, G. and P. Srisuresh, "Network Address
Translation - Protocol Translation (NAT-PT)", RFC 2766,
9.2 Informative References

[I-D.eggert-hiprg-rr-prob-desc]
Eggert, L. and J. Laganier, "HIP Resolution and Rendezvous Problem Description",
Internet-Draft draft-eggert-hiprg-rr-prob-desc-00, October 2004.

[I-D.ietf-hip-mm]
Nikander, P., "End-Host Mobility and Multi-Homing with Host Identity Protocol",
Internet-Draft draft-ietf-hip-mm-00, October 2004.

[I-D.ietf-mmusic-anat]
Camarillo, G., "The Alternative Network Address Types Semantics (ANAT) for the Session Description Protocol (SDP) Grouping Framework",

[I-D.ietf-nsis-nslp-natfw]
Stiemerling, M., "A NAT/Firewall NSIS Signaling Layer Protocol (NSLP)"

[I-D.ietf-v6ops-natpt-to-exprmntl]
Aoun, C. and E. Davies, "Reasons to Move NAT-PT to Experimental",
Internet-Draft draft-ietf-v6ops-natpt-to-exprmntl-00, February 2005.


[reference.NDSS-03.nikander]
Nikander, P., Ylitalo, J. and J. Wall, "Integrating

[reference.UPNP]

[reference.fw_config]

Authors’ Addresses

Martin Stiemerling  
NEC Network Laboratories  
Kurfuerstenanlage 36  
Heidelberg  69115  
Germany  

Phone: +49 6221 90511 13  
Fax:  +49 6221 90511 55  
Email: martin.stiemerling@netlab.nec.de  
URI:  http://www.netlab.nec.de/

Juergen Quittek  
NEC Network Laboratories  
Kurfuerstenanlage 36  
Heidelberg  69115  
Germany  

Phone: +49 6221 90511 15  
Fax:  +49 6221 90511 55  
Email: juergen.quittek@netlab.nec.de  
URI:  http://www.netlab.nec.de/
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.