Workgroup: HIP
Internet-Draft:
draft-moskowitz-hip-fast-mobility-05
Updates: 8046 (if approved)
Published: 12 December 2022
Intended Status: Standards Track
Expires: 15 June 2023
Authors: R. Moskowitz S. Card A. Wiethuechter
HTT Consulting AX Enterprize AX Enterprize

Fast HIP Host Mobility

Abstract

This document describes mobility scenarios and how to aggressively support them in HIP. The goal is minimum lag in the mobility event.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 15 June 2023.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

This document expands on HIP Host Mobility [RFC8046] to describe a set of mobility scenarios that can be addressed by mechanisms that accelerate the basic HIP mobility UPDATE exchange.

HIP Host Mobility [RFC8046] performs a return address validation to ensure that the UPDATE address is reachable by the peer. Two reasons are given for this approach: middleboxes blocking return reachability and malicious peers providing false address updates to flood a target.

The approach here is to start using the new address while it is being validated. Worst case is a few packets are lost or sent to a

wrong target. These are acceptable risks while gaining a fast address update that works in most cases.

One mechanism used is to piggyback data using Next Header even while the mobile peer address is flagged UNVERIFIED. This is practical as the new peer address is authenticated by the HIP_MAC in UPDATE. The UPDATE can neither be forged nor can it be replayed. The verification is more to ensure reverse reachability particularly across NATs and firewalls.

Another mechanism expands the use of the VIA_RVS parameter to "shotgun" mobility UPDATEs. These and other optimizations will be covered in detail.

2. Terms and Definitions

2.1. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2.2. Definitions

MTU: The Maximum Transmit Unit or maximum number of bytes in a datagram that the Interface supports.

SPI: The Security Parameter Index.

3. Problem Space

3.1. Time to complete move

Most mobility environments are built with a "break then make" model for connectivity. Thus there is measurable time between the old address being unusable and the new address being functional. Adding mobility convergence times just further aggravates the delay which negatively impacts the user experience.

The "make then break" model for connectivity is supported via HIP multihoming and is the subject of a separate recommendation.

HIP mobility relies on a 3 packet UPDATE exchange which in some cases can be optimized to 2 packets. This can be further delayed in a "double-jump" scenario with waiting for the direct connection to fail before falling over to contacting the peer's RVS. These processes have resulted in other technologies to be preferred over

HIP as they have faster convergence even if they achieve this while sacrificing security.

3.2. Apriori move knowledge

A HIP Host that has the potential to 'move' (acquire a new address for an interface) during the lifetime of a HIP association SHOULD be registered to an RVS. Such a HIP host SHOULD always inform its peer of its RVS address, as it may experience a "Double-Jump" move as in Section 6.

In an RVS assisted base exchange, the Responder ensures the Initiator knows its RVS with the VIA_RVS parameter in the R1 as specified in HIP Rendezvous Extension [RFC8004]. However, the Responder has no mechanism to learn the Initiator's RVS address. Additionally, it is possible for an Initiator to directly contact the Responder and thus not learn about the Responder's RVS in the base exchange.

A host may not publish its RVS if it does not wish to be easily discovered. It still should notify its peers of its RVS if it expects to be found in some move scenarios.

The HIP base exchange needs to include more RVS information.

4. Enhanced availability of VIA_RVS

The VIA_RVS parameter is defined in HIP Rendezvous Extension [RFC8004] for use in R1, but only identifies the Responder's RVS to the Initiator when the I1 was routed through the RVS.

Firstly, a Responder SHOULD always provide its VIA_RVS information in R1 even when the I1 came directly from the Initiator. Secondly, the Initiator SHOULD always provide its VIA_RVS information in I2. The VIA_RVS address is always maintained as part of the HIT to IP addressing information. Through these two expansions in the availability of VIA_RVS, the hosts are assured to possess their peer's RVS address to "shotgun" UPDATEs and thus accelerate mobility.

5. Single move mobility

Data traffic between host A and B may use ESP with HIP [RFC7402] or any other tunneling protocol. In ESP the relationship of the tunnel SAs with the HIP SA puts a high level of trust on the following fast mobility.

5.1. Piggybacking impact on transport window size

The following sections define the operation of a HIP UPDATE payload followed by some transport (e.g. TCP or UDP) payload in a single IP datagram. This multicontent IP datagram works best with a smaller window size for the higher layer. The normal operation is to compare the size of the transport datagram plus HIP UPDATE payload and ensure it is less than the MTU. An implementation may be able to adjust the transport window size downward so that the higher layer could still fill it and the whole piece then still fit within the MTU.

5.2. Environment

- *Host A is mobile.
- *Host B may be mobile, but not changing IP address at this time.
- *Only Host A is moving in the network and changing its IP address.
- *Host A and B share a HIP Security Association.
- *Host A and B are registered to a RVS server, not necessarily the same and each has the others RVS address.

5.3. Scenario 1: Neither host has data to transmit

Host A triggers a HIP mobility UPDATE with Locator to inform Host B of new address. Host B, upon validating Host A HIP UPDATE, continues with Address Verification.

5.4. Scenario 2: Host A has data to transmit

5.4.1. IPv6 datagram + HIP UPDATE > MTU

Host A triggers a HIP mobility UPDATE with Locator to inform Host B of new address. As the UPDATE + datagram would exceed the MTU, the datagram is sent separately after receipt of the HIP UPDATE from Host B.

The HIP UPDATE packets vary in length as follows:

Move notification: 302 bytes - UPDATE(ESP_INFO, LOCATOR, SEQ, HMAC, HIP_SIGNATURE)

Move verification: 286 bytes - UPDATE(ESP_INFO, SEQ, ACK, HMAC, HIP_SIGNATURE, ECHO_REQUEST_UNSIGNED)

Verification ack: 262 bytes - UPDATE(ESP_INFO, ACK, HMAC, HIP_SIGNATURE, ECHO_RESPONSE_UNSIGNED)

5.4.2. IPv6 datagram + HIP UPDATE <= MTU

Host A sends HIP UPDATE with Locator to inform Host B of new address. Datagram is appended to HIP UPDATE using Next Header. Host B, upon validating Host A HIP UPDATE, sends next header to proper module and continues with Address Verification. This datagram is processed even though the address is UNVERIFIED.

The ESP anti-replay window managed by its envelope sequence number can protect against replayed UPDATE+ESP packets prior to address verification.

5.5. Scenario 3: Host B has data to transmit

After Host B receives a HIP mobility UPDATE from A it has data to send to A. Or Host B may have been sending data to Host A while Host A was moving. The old data may have been lost; for example the data is over UDP with no keepalives during the move time. The old data may be in a retransmission state; for example the data is over TCP. Or the data reached the interface from the higher layer at the same time that the HIP UPDATE with new locator was successfully processed.

5.5.1. IPv6 datagram + HIP UPDATE > MTU

Host B sends the HIP UPDATE validation followed by the IPv6 datagram. Host B may place the address in ACTIVE state or wait from HIP UPDATE confirmation from Host A.

5.5.2. IPv6 datagram + HIP UPDATE <= MTU

Host B sends the HIP UPDATE validation within the IPv6 datagram. Host B may place the address in ACTIVE state or wait from HIP UPDATE confirmation from Host A.

6. Double-Jump mobility

The HIP mobility UPDATE will fail without the use of RVS. In fact both RVS are needed for both UPDATEs to find its peer. This is why the "shotgun" acceleration SHOULD always be used when the peer's RVS is known.

6.1. Environment

*Both host A and B are mobile.

*Host A and B share a HIP Security Association.

*Both hosts move in the network and change their IP addresses. Before either receives the others HIP mobility UPDATE.

*Host A and B are registered to a RVS server, not necessarily the same and each has the others RVS address.

6.2. Shotgunning UPDATEs

Shotgunning is the process of sending the same UPDATE to more than one LOCATOR. In particular it refers to sending the UPDATE to at least the peer's last known IP address and to its RVS address learned from the VIA_RVS for either the R1 or I2 packet.

A host MUST be prepared to receive and discard multiple HIP mobility UPDATEs. The duplicates will be readily identified as having the same SEQ (UPDATE sequence number).

Shotgunning SHOULD always be used when an RVS is known. A peer never knows of a "double-jump" event until after it receives its peer's UPDATE.

6.3. Neither host has data to transmit

Host A triggers a HIP mobility UPDATE with Locator to inform Host B of new address. Host B, upon validating Host A HIP UPDATE, continues with Address Verification.

No attempt should be made to piggyback the two UPDATE processes. They may run simultaneously but not in the same IP datagrams.

6.4. Either host has data to transmit

The following acceleration advice presents a number of challenges. The best rule of thumb is to send the data as soon as possible.

6.4.1. IPv6 datagram + HIP UPDATE > MTU

Same process as Section 6.3

6.4.2. IPv6 datagram + HIP UPDATE <= MTU

Host A sends HIP UPDATE with Locator to inform Host B of new address. Datagram is appended to HIP UPDATE using Next Header. Host B, may have already sent a datagram with its original HIP UPDATE. If since then a receipt of Host A's UPDATE it has more data to transmit, upon validating Host A HIP UPDATE, sends next header to proper module and continues with Address Verification. This datagram is processed even though the address is UNVERIFIED.

7. IANA Considerations

This document does not have any IANA requirements.

8. Security Considerations

The approach here is to start using the new address while it is being validated. One consequence is a few packets are lost or sent to a wrong target. These are acceptable risks while gaining a fast address update that works in most cases.

Another risk is there is a small window for malicious piggyback packets received during this pre-validation interval. But since all such packets should be ESP protected, the ESP process will catch these.

Beyond these items, HIP fast mobility does not introduce any new security considerations beyond that in HIP Host Mobility [RFC8046]. If anything its requirement to know and use the RVS for a peer improve the frequency of a successful mobility notification.

9. Acknowledgments

The term "shotgun" for fast mobility comes from the InfraHIP project. The HIP UPDATE lengths were supplied by Jeff Ahrenholz.

Sue Hares of Huawei and Jeff Ahrenholz of Tempered Networks contributed to the clarity in this document.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

10.2. Informative References

[RFC7402] Jokela, P., Moskowitz, R., and J. Melen, "Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP)", RFC 7402, DOI 10.17487/RFC7402, April 2015, <https://www.rfc-editor.org/info/rfc7402>.

[RFC8004] Laganier, J. and L. Eggert, "Host Identity Protocol (HIP) Rendezvous Extension", RFC 8004, DOI 10.17487/RFC8004, October 2016, <https://www.rfc-editor.org/info/rfc8004>.

[RFC8046]
Henderson, T., Ed., Vogt, C., and J. Arkko, "Host Mobility with the Host Identity Protocol", RFC 8046, DOI 10.17487/RFC8046, February 2017, <https://www.rfc-editor.org/info/rfc8046>.

Authors' Addresses

Robert Moskowitz
HTT Consulting
Oak Park, MI 48237
United States of America

Email: rgm@labs.htt-consult.com

Stuart W. Card
AX Enterprize
4947 Commercial Drive
Yorkville, NY 13495
United States of America

Email: stu.card@axenterprize.com

Adam Wiethuechter
AX Enterprize
4947 Commercial Drive
Yorkville, NY 13495
United States of America

Email: adam.wiethuechter@axenterprize.com