Network Working Group   T. Henderson, Ed.
Internet-Draft   University of Washington
Intended status: Standards Track   C. Vogt
Expires: July 16, 2015   J. Arkko
                                      Ericsson Research NomadicLab
                                              January 12, 2015

Host Multihoming with the Host Identity Protocol
draft-ietf-hip-multihoming-05

Abstract

   This document defines host multihoming extensions to the Host
   Identity Protocol (HIP), by leveraging protocol components defined
   for host mobility.

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 http://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 July 16, 2015.

Table of Contents

1. Introduction and Scope

The Host Identity Protocol [I-D.ietf-hip-rfc4423-bis] (HIP) supports
an architecture that decouples the transport layer (TCP, UDP, etc.)
from the internetworking layer (IPv4 and IPv6) by using public/
private key pairs, instead of IP addresses, as host identities. When
a host uses HIP, the overlying protocol sublayers (e.g., transport

layer sockets and Encapsulating Security Payload (ESP) Security
Associations (SAs)) are instead bound to representations of these
host identities, and the IP addresses are only used for packet
forwarding. However, each host must also know at least one IP
address at which its peers are reachable. Initially, these IP
addresses are the ones used during the HIP base exchange
[I-D.ietf-hip-rfc5201-bis].

One consequence of such a decoupling is that new solutions to
network-layer mobility and host multihoming are possible. Basic host
mobility is defined in [I-D.ietf-hip-rfc5206-bis] and covers the case
in which a host has a single address and changes its network point-
of-attachment while desiring to preserve the HIP-enabled security
association. Host multihoming is somewhat of a dual case to host
mobility, in that a host may simultaneously have more than one
network point-of-attachment. There are potentially many variations
of host multihoming possible. The scope of this document encompasses
messaging and elements of procedure for some basic host multihoming
scenarios of interest.

Another variation of multihoming that has been heavily studied is
site multihoming. Solutions for site multihoming in IPv6 networks
have been specified by the IETF shim6 working group. The shim6
protocol [RFC5533] bears many architectural similarities to HIP but
there are differences in the security model and in the protocol.

While HIP can potentially be used with transports other than the ESP
transport format [I-D.ietf-hip-rfc5202-bis], this document largely
assumes the use of ESP and leaves other transport formats for further
study.

There are a number of situations where the simple end-to-end
readdressing functionality defined herein is not sufficient. These
include the initial reachability of a multihomed host, location
privacy, simultaneous mobility of both hosts, and some modes of NAT
traversal. In these situations, there is a need for some helper
functionality in the network, such as a HIP rendezvous server
[I-D.ietf-hip-rfc5204-bis]. Such functionality is out of the scope
of this document. Finally, making underlying IP multihoming
transparent to the transport layer has implications on the proper
response of transport congestion control, path MTU selection, and
Quality of Service (QoS). Transport-layer mobility triggers, and the
proper transport response to a HIP multihoming address change, are
outside the scope of this document.

[2](#). Terminology and Conventions

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](#)].

The following terms used in this document are defined in [[I-D.ietf-hip-rfc5206-bis](#)]: LOCATOR_SET, Locator, Address, Preferred locator, and Credit Based Authorization.

[3](#). Protocol Model

The protocol model for HIP support of host multihoming extends the model for host mobility described in Section 3 of [[I-D.ietf-hip-rfc5206-bis](#)]. This section only highlights the differences.

In host multihoming, a host has multiple locators simultaneously rather than sequentially, as in the case of mobility. By using the LOCATOR_SET parameter defined in [[I-D.ietf-hip-rfc5206-bis](#)], a host can inform its peers of additional (multiple) locators at which it can be reached. When multiple locators are available and announced to the peer, a host can designate a particular locator as a "preferred" locator, meaning that the host prefers that its peer send packets to the designated address before trying an alternative address. Although this document defines a basic mechanism for multihoming, it does not define all possible policies and procedures, such as which locators to choose when more than one pair is available, the operation of simultaneous mobility and multihoming, source address selection policies (beyond those specified in [[RFC3484](#)]), and the implications of multihoming on transport protocols and ESP anti-replay windows.

[4](#). Protocol Overview

In this section, we briefly introduce a number of usage scenarios for HIP multihoming. These scenarios assume that HIP is being used with the ESP transform [[I-D.ietf-hip-rfc5202-bis](#)], although other scenarios may be defined in the future. To understand these usage scenarios, the reader should be at least minimally familiar with the HIP protocol specification [[I-D.ietf-hip-rfc5201-bis](#)]. However, for the (relatively) uninitiated reader, it is most important to keep in mind that in HIP the actual payload traffic is protected with ESP, and that the ESP SPI acts as an index to the right host-to-host context.

The scenarios below assume that the two hosts have completed a single HIP base exchange with each other. Both of the hosts therefore have

one incoming and one outgoing SA. Further, each SA uses the same
pair of IP addresses, which are the ones used in the base exchange.

The readdressing protocol is an asymmetric protocol where a mobile or
multihomed host informs a peer host about changes of IP addresses on
affected SPIs. The readdressing exchange is designed to be
piggybacked on existing HIP exchanges. The majority of the packets
on which the LOCATOR_SET parameters are expected to be carried are
UPDATE packets. However, some implementations may want to experiment
with sending LOCATOR_SET parameters also on other packets, such as
R1, I2, and NOTIFY.

The scenarios below at times describe addresses as being in either an
ACTIVE, VERIFIED, or DEPRECATED state. From the perspective of a
host, newly-learned addresses of the peer must be verified before put
into active service, and addresses removed by the peer are put into a
deprecated state. Under limited conditions described in
[I-D.ietf-hip-rfc5206-bis], an UNVERIFIED address may be used.

Hosts that use link-local addresses as source addresses in their HIP
handshakes may not be reachable by a mobile peer. Such hosts SHOULD
provide a globally routable address either in the initial handshake
or via the LOCATOR_SET parameter.

4.1. Host Multihoming

A (mobile or stationary) host may sometimes have more than one
interface or global address. The host may notify the peer host of
the additional interface or address by using the LOCATOR_SET
parameter. To avoid problems with the ESP anti-replay window, a host
SHOULD use a different SA for each interface or address used to
receive packets from the peer host when multiple locator pairs are
being used simultaneously rather than sequentially.

When more than one locator is provided to the peer host, the host
SHOULD indicate which locator is preferred (the locator on which the
host prefers to receive traffic). By default, the addresses used in
the base exchange are preferred until indicated otherwise.

In the multihoming case, the sender may also have multiple valid
locators from which to source traffic. In practice, a HIP
association in a multihoming configuration may have both a preferred
peer locator and a preferred local locator, although rules for source
address selection should ultimately govern the selection of the
source locator based on the destination locator.

Although the protocol may allow for configurations in which there is
an asymmetric number of SAs between the hosts (e.g., one host has two

interfaces and two inbound SAs, while the peer has one interface and
one inbound SA), it is RECOMMENDED that inbound and outbound SAs be
created pairwise between hosts. When an ESP_INFO arrives to rekey a
particular outbound SA, the corresponding inbound SA should be also
rekeyed at that time. Although asymmetric SA configurations might be
experimented with, their usage may constrain interoperability at this
time. However, it is recommended that implementations attempt to
support peers that prefer to use non-paired SAs.

Consider the case between two hosts, one single-homed and one
multihomed. The multihomed host may decide to inform the single-
homed host about its other address. It is RECOMMENDED that the
multihomed host set up a new SA pair for use on this new address. To
do this, the multihomed host sends a LOCATOR_SET with an ESP_INFO,
indicating the request for a new SA by setting the OLD SPI value to
zero, and the NEW SPI value to the newly created incoming SPI. A
Locator Type of "1" is used to associate the new address with the new
SPI. The LOCATOR_SET parameter also contains a second Type "1"
locator, that of the original address and SPI. To simplify parameter
processing and avoid explicit protocol extensions to remove locators,
each LOCATOR_SET parameter MUST list all locators in use on a
connection (a complete listing of inbound locators and SPIs for the
host). The multihomed host waits for an ESP_INFO (new outbound SA)
from the peer and an ACK of its own UPDATE. As in the mobility case,
the peer host must perform an address verification before actively
using the new address. Figure 1 illustrates this scenario.

```
   Multi-homed Host                        Peer Host

        UPDATE(ESP_INFO, LOCATOR_SET, SEQ, [DIFFIE_HELLMAN])
    ------------------------------------>
        UPDATE(ESP_INFO, SEQ, ACK, [DIFFIE_HELLMAN,] ECHO_REQUEST)
    <------------------------------------
        UPDATE(ACK, ECHO_RESPONSE)
    ------------------------------------>
```

                 Figure 1: Basic Multihoming Scenario

In multihoming scenarios, it is important that hosts receiving
UPDATEs associate them correctly with the destination address used in
the packet carrying the UPDATE. When processing inbound LOCATOR_SETs
that establish new security associations on an interface with
multiple addresses, a host uses the destination address of the UPDATE
containing the LOCATOR_SET as the local address to which the
LOCATOR_SET plus ESP_INFO is targeted. This is because hosts may
send UPDATEs with the same (locator) IP address to different peer
addresses -- this has the effect of creating multiple inbound SAs
implicitly affiliated with different peer source addresses.

4.2. Site Multihoming

A host may have an interface that has multiple globally routable IP
addresses. Such a situation may be a result of the site having
multiple upper Internet Service Providers, or just because the site
provides all hosts with both IPv4 and IPv6 addresses. The host
should stay reachable at all or any subset of the currently available
global routable addresses, independent of how they are provided.

This case is handled the same as if there were different IP
addresses, described above in Section 4.1. Note that a single
interface may experience site multihoming while the host itself may
have multiple interfaces.

Note that a host may be multihomed and mobile simultaneously, and
that a multihomed host may want to protect the location of some of
its interfaces while revealing the real IP address of some others.

This document does not presently additional site multihoming
extensions to HIP; such extensions are for further study.

4.3. Dual host multihoming

Consider the case in which both hosts would like to add an additional
address after the base exchange completes. In Figure 2, consider
that host1, which used address addr1a in the base exchange to set up
SPI1a and SPI2a, wants to add address addr1b. It would send an
UPDATE with LOCATOR_SET (containing the address addr1b) to host2,
using destination address addr2a, and a new set of SPIs would be
added between hosts 1 and 2 (call them SPI1b and SPI2b -- not shown
in the figure). Next, consider host2 deciding to add addr2b to the
relationship. Host2 must select one of host1's addresses towards
which to initiate an UPDATE. It may choose to initiate an UPDATE to
addr1a, addr1b, or both. If it chooses to send to both, then a full
mesh (four SA pairs) of SAs would exist between the two hosts. This
is the most general case; it often may be the case that hosts
primarily establish new SAs only with the peer's Preferred locator.
The readdressing protocol is flexible enough to accommodate this
choice.

```
        -<- SPI1a --                          -- SPI2a ->-
   host1 <             > addr1a <---> addr2a <            > host2
        ->- SPI2a --                          -- SPI1a -<-

                        addr1b <---> addr2a  (second SA pair)
                        addr1a <---> addr2b  (third SA pair)
                        addr1b <---> addr2b  (fourth SA pair)
```
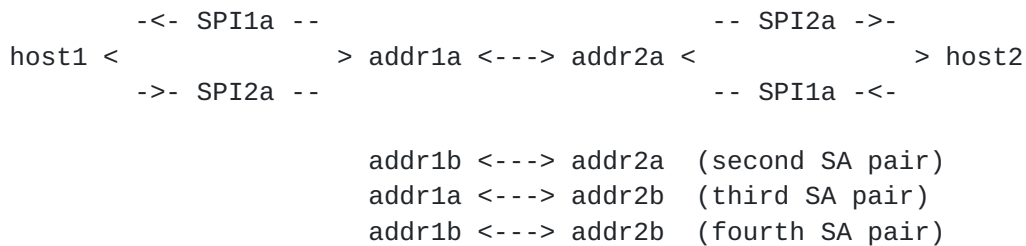
       Figure 2: Dual Multihoming Case in Which Each Host Uses LOCATOR_SET
                          to Add a Second Address

4.4. Combined Mobility and Multihoming

   It looks likely that in the future, many mobile hosts will be
   simultaneously mobile and multihomed, i.e., have multiple mobile
   interfaces. Furthermore, if the interfaces use different access
   technologies, it is fairly likely that one of the interfaces may
   appear stable (retain its current IP address) while some other(s) may
   experience mobility (undergo IP address change).

   The use of LOCATOR_SET plus ESP_INFO should be flexible enough to
   handle most such scenarios, although more complicated scenarios have
   not been studied so far.

4.5. Initiating the Protocol in R1 or I2

   A Responder host MAY include a LOCATOR_SET parameter in the R1 packet
   that it sends to the Initiator. This parameter MUST be protected by
   the R1 signature. If the R1 packet contains LOCATOR_SET parameters
   with a new Preferred locator, the Initiator SHOULD directly set the
   new Preferred locator to status ACTIVE without performing address
   verification first, and MUST send the I2 packet to the new Preferred
   locator. The I1 destination address and the new Preferred locator
   may be identical. All new non-preferred locators must still undergo
   address verification once the base exchange completes.

```
           Initiator                               Responder

                          R1 with LOCATOR_SET
                <-----------------------------------
    record additional addresses
    change responder address
                  I2 sent to newly indicated preferred address
                ----------------------------------->
                                                  (process normally)
                                R2
                <-----------------------------------
    (process normally, later verification of non-preferred locators)
```
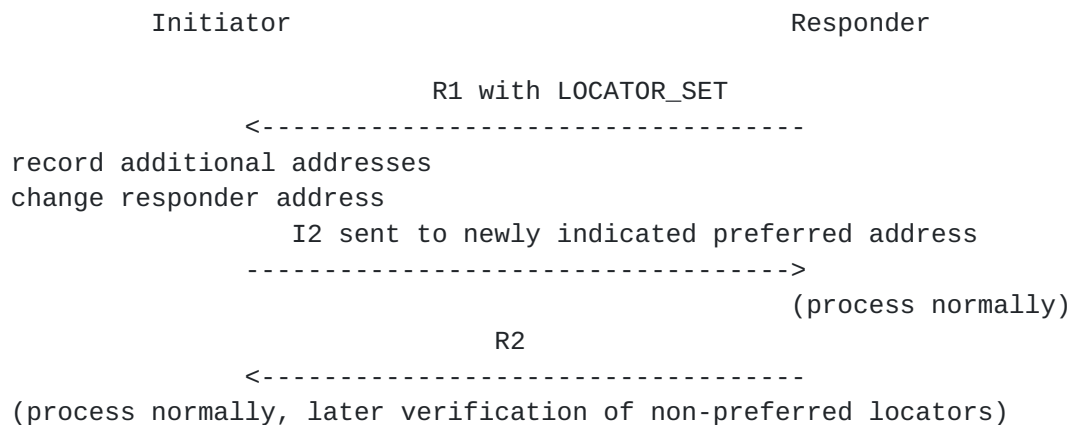
              Figure 3: LOCATOR_SET Inclusion in R1

    An Initiator MAY include one or more LOCATOR_SET parameters in the I2
    packet, independent of whether or not there was a LOCATOR_SET
    parameter in the R1. These parameters MUST be protected by the I2
    signature. Even if the I2 packet contains LOCATOR_SET parameters,
    the Responder MUST still send the R2 packet to the source address of
    the I2. The new Preferred locator SHOULD be identical to the I2
    source address. If the I2 packet contains LOCATOR_SET parameters,
    all new locators must undergo address verification as usual, and the
    ESP traffic that subsequently follows should use the Preferred
    locator.

```
           Initiator                               Responder

                          I2 with LOCATOR_SET
                ----------------------------------->
                                                  (process normally)
                                          record additional addresses
                    R2 sent to source address of I2
                <-----------------------------------
    (process normally)
```
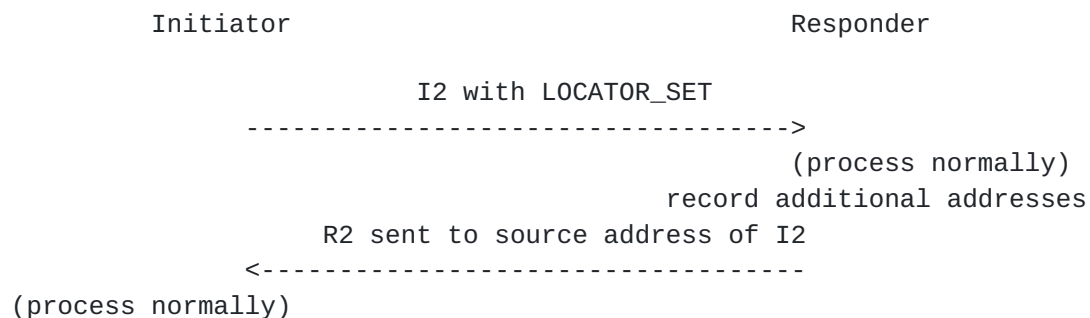
              Figure 4: LOCATOR_SET Inclusion in I2

    The I1 and I2 may be arriving from different source addresses if the
    LOCATOR_SET parameter is present in R1. In this case,
    implementations simultaneously using multiple pre-created R1s,
    indexed by Initiator IP addresses, may inadvertently fail the puzzle
    solution of I2 packets due to a perceived puzzle mismatch. See, for
    instance, the example in Appendix A of [I-D.ietf-hip-rfc5201-bis].
    As a solution, the Responder's puzzle indexing mechanism must be
    flexible enough to accommodate the situation when R1 includes a
    LOCATOR_SET parameter.

4.6. Using LOCATOR_SETs across Addressing Realms

   It is possible for HIP associations to migrate to a state in which
   both parties are only using locators in different addressing realms.
   For example, the two hosts may initiate the HIP association when both
   are using IPv6 locators, then one host may loose its IPv6
   connectivity and obtain an IPv4 address. In such a case, some type
   of mechanism for interworking between the different realms must be
   employed; such techniques are outside the scope of the present text.
   The basic problem in this example is that the host readdressing to
   IPv4 does not know a corresponding IPv4 address of the peer. This
   may be handled (experimentally) by possibly configuring this address
   information manually or in the DNS, or the hosts exchange both IPv4
   and IPv6 addresses in the locator.

5. Other Considerations

5.1. Address Verification

   An address verification method is specified in
   [I-D.ietf-hip-rfc5206-bis]. It is expected that addresses learned in
   multihoming scenarios also are subject to the same verification
   rules.

5.2. Preferred Locator

   When a host has multiple locators, the peer host must decide which to
   use for outbound packets. It may be that a host would prefer to
   receive data on a particular inbound interface. HIP allows a
   particular locator to be designated as a Preferred locator and
   communicated to the peer.

   In general, when multiple locators are used for a session, there is
   the question of using multiple locators for failover only or for
   load-balancing. Due to the implications of load-balancing on the
   transport layer that still need to be worked out, this document
   assumes that multiple locators are used primarily for failover. An
   implementation may use ICMP interactions, reachability checks, or
   other means to detect the failure of a locator.

5.3. Interaction with Security Associations

   This document uses the HIP LOCATOR_SET protocol parameter, specified
   in [I-D.ietf-hip-rfc5206-bis]), that allows the hosts to exchange
   information about their locator(s) and any changes in their
   locator(s). The logical structure created with LOCATOR_SET
   parameters has three levels: hosts, Security Associations (SAs)
   indexed by Security Parameter Indices (SPIs), and addresses.

The relation between these levels for an association constructed as
defined in the base specification [I-D.ietf-hip-rfc5201-bis] and ESP
transform [I-D.ietf-hip-rfc5202-bis] is illustrated in Figure 5.

```
         -<- SPI1a --                          -- SPI2a ->-
   host1 <              > addr1a <---> addr2a <           > host2
         ->- SPI2a --                          -- SPI1a -<-
```

         Figure 5: Relation between Hosts, SPIs, and Addresses (Base
                              Specification)

In Figure 5, host1 and host2 negotiate two unidirectional SAs, and
each host selects the SPI value for its inbound SA. The addresses
addr1a and addr2a are the source addresses that the hosts use in the
base HIP exchange. These are the "preferred" (and only) addresses
conveyed to the peer for use on each SA. That is, although packets
sent to any of the hosts' interfaces may be accepted on the inbound
SA, the peer host in general knows of only the single destination
address learned in the base exchange (e.g., for host1, it sends a
packet on SPI2a to addr2a to reach host2), unless other mechanisms
exist to learn of new addresses.

In general, the bindings that exist in an implementation
corresponding to this document can be depicted as shown in Figure 6.
In this figure, a host can have multiple inbound SPIs (and, not
shown, multiple outbound SPIs) associated with another host.
Furthermore, each SPI may have multiple addresses associated with it.
These addresses that are bound to an SPI are not used to lookup the
incoming SA. Rather, the addresses are those that are provided to
the peer host, as hints for which addresses to use to reach the host
on that SPI. The LOCATOR_SET parameter is used to change the set of
addresses that a peer associates with a particular SPI.

```
                          address11
                        /
                  SPI1    - address12
               /
              /           address21
      host -- SPI2   <
              \           address22
               \
                  SPI3    - address31
                        \
                          address32
```

     Figure 6: Relation between Hosts, SPIs, and Addresses (General Case)

A host may establish any number of security associations (or SPIs)
with a peer. The main purpose of having multiple SPIs with a peer is
to group the addresses into collections that are likely to experience
fate sharing. For example, if the host needs to change its addresses
on SPI2, it is likely that both address21 and address22 will
simultaneously become obsolete. In a typical case, such SPIs may
correspond with physical interfaces; see below. Note, however, that
especially in the case of site multihoming, one of the addresses may
become unreachable while the other one still works. In the typical
case, however, this does not require the host to inform its peers
about the situation, since even the non-working address still
logically exists.

A basic property of HIP SAs is that the inbound IP address is not
used to lookup the incoming SA. Therefore, in Figure 6, it may seem
unnecessary for address31, for example, to be associated only with
SPI3 -- in practice, a packet may arrive to SPI1 via destination
address address31 as well. However, the use of different source and
destination addresses typically leads to different paths, with
different latencies in the network, and if packets were to arrive via
an arbitrary destination IP address (or path) for a given SPI, the
reordering due to different latencies may cause some packets to fall
outside of the ESP anti-replay window. For this reason, HIP provides
a mechanism to affiliate destination addresses with inbound SPIs,
when there is a concern that anti-replay windows might be violated.
In this sense, we can say that a given inbound SPI has an "affinity"
for certain inbound IP addresses, and this affinity is communicated
to the peer host. Each physical interface SHOULD have a separate SA,
unless the ESP anti-replay window is loose.

Moreover, even when the destination addresses used for a particular
SPI are held constant, the use of different source interfaces may
also cause packets to fall outside of the ESP anti-replay window,
since the path traversed is often affected by the source address or
interface used. A host has no way to influence the source interface
on which a peer sends its packets on a given SPI. A host SHOULD
consistently use the same source interface and address when sending
to a particular destination IP address and SPI. For this reason, a
host may find it useful to change its SPI or at least reset its ESP
anti-replay window when the peer host readdresses.

An address may appear on more than one SPI. This creates no
ambiguity since the receiver will ignore the IP addresses during SA
lookup anyway. However, this document does not specify such cases.

When the LOCATOR_SET parameter is sent in an UPDATE packet, then the
receiver will respond with an UPDATE acknowledgment. When the
LOCATOR_SET parameter is sent in an R1 or I2 packet, the base

exchange retransmission mechanism will confirm its successful
delivery. LOCATOR_SETs may experimentally be used in NOTIFY packets;
in this case, the recipient MUST consider the LOCATOR_SET as
informational and not immediately change the current preferred
address, but can test the additional locators when the need arises.
The use of the LOCATOR_SET in a NOTIFY message may not be compatible
with middleboxes.

6. Processing Rules

Basic processing rules for the LOCATOR_SET parameter are specified in
[I-D.ietf-hip-rfc5206-bis]. This document focuses on multihoming-
specific rules.

6.1. Sending LOCATOR_SETs

The decision of when to send a LOCATOR_SET, and which addresses to
include, is a local policy issue. [I-D.ietf-hip-rfc5206-bis]
recommends that a host send a LOCATOR_SET whenever it recognizes a
change of its IP addresses in use on an active HIP association, and
assumes that the change is going to last at least for a few seconds.
It is possible to delay the exposure of additional locators to the
peer, and to send data from previously unannounced locators, as might
arise in certain mobility or multihoming situations.

When a host decides to inform its peers about changes in its IP
addresses, it has to decide how to group the various addresses with
SPIs. The grouping should consider also whether middlebox
interaction requires sending the same LOCATOR_SET in separate UPDATEs
on different paths. Since each SPI is associated with a different
Security Association, the grouping policy may also be based on ESP
anti-replay protection considerations. In the typical case, simply
basing the grouping on actual kernel level physical and logical
interfaces may be the best policy. Grouping policy is outside of the
scope of this document.

Locators corresponding to tunnel interfaces (e.g. IPsec tunnel
interfaces or Mobile IP home addresses) or other virtual interfaces
MAY be announced in a LOCATOR_SET, but implementations SHOULD avoid
announcing such locators as preferred locators if more direct paths
may be obtained by instead preferring locators from non-tunneling
interfaces if such locators provide a more direct path to the HIP
peer.

Hosts MUST NOT announce broadcast or multicast addresses in
LOCATOR_SETs. Link-local addresses MAY be announced to peers that
are known to be neighbors on the same link, such as when the IP
destination address of a peer is also link-local. The announcement

of link-local addresses in this case is a policy decision; link-local
addresses used as Preferred locators will create reachability
problems when the host moves to another link. In any case, link-
local addresses MUST NOT be announced to a peer unless that peer is
known to be on the same link.

Once the host has decided on the groups and assignment of addresses
to the SPIs, it creates a LOCATOR_SET parameter that serves as a
complete representation of the addresses and affiliated SPIs intended
for active use. We now describe a few cases introduced in Section 4.
We assume that the Traffic Type for each locator is set to "0" (other
values for Traffic Type may be specified in documents that separate
the HIP control plane from data plane traffic). Other mobility and
multihoming cases are possible but are left for further
experimentation.

1. Host multihoming (addition of an address). We only describe the
    simple case of adding an additional address to a (previously)
    single-homed, non-mobile host. The host SHOULD set up a new SA
    pair between this new address and the preferred address of the
    peer host. To do this, the multihomed host creates a new inbound
    SA and creates a new SPI. For the outgoing UPDATE message, it
    inserts an ESP_INFO parameter with an OLD SPI field of "0", a NEW
    SPI field corresponding to the new SPI, and a KEYMAT Index as
    selected by local policy. The host adds to the UPDATE message a
    LOCATOR_SET with two Type "1" Locators: the original address and
    SPI active on the association, and the new address and new SPI
    being added (with the SPI matching the NEW SPI contained in the
    ESP_INFO). The Preferred bit SHOULD be set depending on the
    policy to tell the peer host which of the two locators is
    preferred. The UPDATE also contains a SEQ parameter and
    optionally a DIFFIE_HELLMAN parameter, and follows rekeying
    procedures with respect to this new address. The UPDATE message
    SHOULD be sent to the peer's Preferred address with a source
    address corresponding to the new locator.

The sending of multiple LOCATOR_SETs, locators with Locator Type "0",
and multiple ESP_INFO parameters is for further study. Note that the
inclusion of LOCATOR_SET in an R1 packet requires the use of Type "0"
locators since no SAs are set up at that point.

6.2. Handling Received LOCATOR_SETs

A host SHOULD be prepared to receive a LOCATOR_SET parameter in the
following HIP packets: R1, I2, UPDATE, and NOTIFY.

This document describes sending both ESP_INFO and LOCATOR_SET
parameters in an UPDATE. The ESP_INFO parameter is included when

there is a need to rekey or key a new SPI, and is otherwise included
for the possible benefit of HIP-aware middleboxes. The LOCATOR_SET
parameter contains a complete map of the locators that the host
wishes to make or keep active for the HIP association.

In general, the processing of a LOCATOR_SET depends upon the packet
type in which it is included. Here, we describe only the case in
which ESP_INFO is present and a single LOCATOR_SET and ESP_INFO are
sent in an UPDATE message; other cases are for further study. The
steps below cover each of the cases described in Section 6.1.

The processing of ESP_INFO and LOCATOR_SET parameters is intended to
be modular and support future generalization to the inclusion of
multiple ESP_INFO and/or multiple LOCATOR_SET parameters. A host
SHOULD first process the ESP_INFO before the LOCATOR_SET, since the
ESP_INFO may contain a new SPI value mapped to an existing SPI, while
a Type "1" locator will only contain a reference to the new SPI.

When a host receives a validated HIP UPDATE with a LOCATOR_SET and
ESP_INFO parameter, it processes the ESP_INFO as follows. The
ESP_INFO parameter indicates whether an SA is being rekeyed, created,
deprecated, or just identified for the benefit of middleboxes. The
host examines the OLD SPI and NEW SPI values in the ESP_INFO
parameter:

1. (no rekeying) If the OLD SPI is equal to the NEW SPI and both
    correspond to an existing SPI, the ESP_INFO is gratuitous
    (provided for middleboxes) and no rekeying is necessary.

2. (rekeying) If the OLD SPI indicates an existing SPI and the NEW
    SPI is a different non-zero value, the existing SA is being
    rekeyed and the host follows HIP ESP rekeying procedures by
    creating a new outbound SA with an SPI corresponding to the NEW
    SPI, with no addresses bound to this SPI. Note that locators in
    the LOCATOR_SET parameter will reference this new SPI instead of
    the old SPI.

3. (new SA) If the OLD SPI value is zero and the NEW SPI is a new
    non-zero value, then a new SA is being requested by the peer.
    This case is also treated like a rekeying event; the receiving
    host must create a new SA and respond with an UPDATE ACK.

4. (deprecating the SA) If the OLD SPI indicates an existing SPI and
    the NEW SPI is zero, the SA is being deprecated and all locators
    uniquely bound to the SPI are put into the DEPRECATED state.

If none of the above cases apply, a protocol error has occurred and
the processing of the UPDATE is stopped.

Next, the locators in the LOCATOR_SET parameter are processed. For
each locator listed in the LOCATOR_SET parameter, check that the
address therein is a legal unicast or anycast address. That is, the
address MUST NOT be a broadcast or multicast address. Note that some
implementations MAY accept addresses that indicate the local host,
since it may be allowed that the host runs HIP with itself.

The below assumes that all locators are of Type "1" with a Traffic
Type of "0"; other cases are for further study.

For each Type "1" address listed in the LOCATOR_SET parameter, the
host checks whether the address is already bound to the SPI
indicated. If the address is already bound, its lifetime is updated.
If the status of the address is DEPRECATED, the status is changed to
UNVERIFIED. If the address is not already bound, the address is
added, and its status is set to UNVERIFIED. Mark all addresses
corresponding to the SPI that were NOT listed in the LOCATOR_SET
parameter as DEPRECATED.

As a result, at the end of processing, the addresses listed in the
LOCATOR_SET parameter have either a state of UNVERIFIED or ACTIVE,
and any old addresses on the old SA not listed in the LOCATOR_SET
parameter have a state of DEPRECATED.

Once the host has processed the locators, if the LOCATOR_SET
parameter contains a new Preferred locator, the host SHOULD initiate
a change of the Preferred locator. This requires that the host first
verifies reachability of the associated address, and only then
changes the Preferred locator; see Section 6.4.

If a host receives a locator with an unsupported Locator Type, and
when such a locator is also declared to be the Preferred locator for
the peer, the host SHOULD send a NOTIFY error with a Notify Message
Type of LOCATOR_TYPE_UNSUPPORTED, with the Notification Data field
containing the locator(s) that the receiver failed to process.
Otherwise, a host MAY send a NOTIFY error if a (non-preferred)
locator with an unsupported Locator Type is received in a LOCATOR_SET
parameter.

6.3. Verifying Address Reachability

Address verification is defined in [I-D.ietf-hip-rfc5206-bis].

When address verification is in progress for a new Preferred locator,
the host SHOULD select a different locator listed as ACTIVE, if one
such locator is available, to continue communications until address
verification completes. Alternatively, the host MAY use the new
Preferred locator while in UNVERIFIED status to the extent Credit-

Based Authorization permits. Credit-Based Authorization is explained
in [I-D.ietf-hip-rfc5206-bis]. Once address verification succeeds,
the status of the new Preferred locator changes to ACTIVE.

6.4. Changing the Preferred Locator

A host MAY want to change the Preferred outgoing locator for
different reasons, e.g., because traffic information or ICMP error
messages indicate that the currently used preferred address may have
become unreachable. Another reason may be due to receiving a
LOCATOR_SET parameter that has the "P" bit set.

To change the Preferred locator, the host initiates the following
procedure:

1. If the new Preferred locator has ACTIVE status, the Preferred
    locator is changed and the procedure succeeds.

2. If the new Preferred locator has UNVERIFIED status, the host
    starts to verify its reachability. The host SHOULD use a
    different locator listed as ACTIVE until address verification
    completes if one such locator is available. Alternatively, the
    host MAY use the new Preferred locator, even though in UNVERIFIED
    status, to the extent Credit-Based Authorization permits. Once
    address verification succeeds, the status of the new Preferred
    locator changes to ACTIVE and its use is no longer governed by
    Credit-Based Authorization.

3. If the peer host has not indicated a preference for any address,
    then the host picks one of the peer's ACTIVE addresses randomly
    or according to policy. This case may arise if, for example,
    ICMP error messages that deprecate the Preferred locator arrive,
    but the peer has not yet indicated a new Preferred locator.

4. If the new Preferred locator has DEPRECATED status and there is
    at least one non-deprecated address, the host selects one of the
    non-deprecated addresses as a new Preferred locator and
    continues. If the selected address is UNVERIFIED, the address
    verification procedure described above will apply.

7. Security Considerations

Security considerations are addressed in [I-D.ietf-hip-rfc5206-bis].

8. IANA Considerations

This document has no new IANA considerations.

9. Authors and Acknowledgments

This document contains content that was originally included in
RFC5206. Pekka Nikander and Jari Arkko originated RFC5206, and
Christian Vogt and Thomas Henderson (editor) later joined as co-
authors. Also in RFC5206, Greg Perkins contributed the initial draft
of the security section, and Petri Jokela was a co-author of the
initial individual submission.

The authors thank Miika Komu, Mika Kousa, Jeff Ahrenholz, and Jan
Melen for many improvements to the document.

10. References

10.1. Normative references

[I-D.ietf-hip-rfc5201-bis]
          Moskowitz, R., Heer, T., Jokela, P., and T. Henderson,
          "Host Identity Protocol Version 2 (HIPv2)", draft-ietf-
          hip-rfc5201-bis-20 (work in progress), October 2014.

[I-D.ietf-hip-rfc5202-bis]
          Jokela, P., Moskowitz, R., and J. Melen, "Using the
          Encapsulating Security Payload (ESP) Transport Format with
          the Host Identity Protocol (HIP)", draft-ietf-hip-
          rfc5202-bis-07 (work in progress), September 2014.

[I-D.ietf-hip-rfc5206-bis]
          Henderson, T., Vogt, C., and J. Arkko, "Host Mobility with
          the Host Identity Protocol", draft-ietf-hip-rfc5206-bis-07
          (work in progress), December 2014.

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

[RFC3484] Draves, R., "Default Address Selection for Internet
          Protocol version 6 (IPv6)", RFC 3484, February 2003.

10.2. Informative references

[I-D.ietf-hip-rfc4423-bis]
          Moskowitz, R. and M. Komu, "Host Identity Protocol
          Architecture", draft-ietf-hip-rfc4423-bis-09 (work in
          progress), October 2014.

   [I-D.ietf-hip-rfc5204-bis]
              Laganier, J. and L. Eggert, "Host Identity Protocol (HIP)
              Rendezvous Extension", draft-ietf-hip-rfc5204-bis-05 (work
              in progress), December 2014.

   [RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming
              Shim Protocol for IPv6", RFC 5533, June 2009.

Appendix A. Document Revision History

To be removed upon publication

+----------+---+
| Revision | Comments |
+----------+---+
draft-00	Initial version with multihoming text imported from
	RFC5206.
draft-01	Document refresh; no other changes.
draft-02	Document refresh; no other changes.
draft-03	Document refresh; no other changes.
draft-04	Document refresh; no other changes.
draft-05	Move remaining multihoming material from RFC5206-bis
	to this document
	Update lingering references to LOCATOR parameter to
	LOCATOR_SET
+----------+---+

Authors' Addresses

Thomas R. Henderson (editor)
University of Washington
Campus Box 352500
Seattle, WA
USA

EMail: tomhend@u.washington.edu

Christian Vogt
Ericsson Research NomadicLab
Hirsalantie 11
JORVAS FIN-02420
FINLAND

EMail: christian.vogt@ericsson.com

      Jari Arkko
      Ericsson Research NomadicLab
      JORVAS FIN-02420
      FINLAND

      Phone: +358 40 5079256
      EMail: jari.arkko@ericsson.com