Workgroup: HIP
Updates: 7401 (if approved)
Published: 13 May 2020
Intended Status: Standards Track
Expires: 14 November 2020
Authors: R. Moskowitz HTT Consulting
S. Card AX Enterprize
A. Wiethuechter AX Enterprize

Hierarchical HITs for HIPv2

Abstract

This document describes using a hierarchical HIT to facilitate large deployments of managed devices. Hierarchical HITs differ from HIPv2 flat HITs by only using 64 bits for mapping the Host Identity, freeing 32 bits to bind in a hierarchy of Registering Entities that provide services to the consumers of hierarchical HITs.

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 14 November 2020.

Table of Contents

1. Introduction

This document expands on HIPv2 [RFC7401] to describe the structure of a hierarchical HIT (HHIT). Some of the challenges for large scale deployment addressed by HHITs are presented. The basics for the hierarchical HIT registries are defined here.

Including hierarchy information within the HIT is not a new concept. This was part of the original HIPv1 Architecture [draft.moskowitz-hip-arch-02]. It was dropped from the HIPv1 work for lack of a use case and concerns over the smaller HI mapping space. It was later brought up in the HIP Research Group (HIP-RG) in [draft.zhang-hip-hierarchical-parameter-00], but this never gained concensus.

Hierarchical HITs now have a solid use case with Public, mobile devices (e.g. Unmanned Aircraft). The math to evaluate the statistical collision risk is available, Appendix A. And finally, HHIT Registries [hhit-registries] provide a way to manage the hierarchy.

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

HDA (Hierarchical HIT Domain Authority):
The 16 bit field identifying the HIT Domain Authority under an RAA.

HID (Hierarchy ID):
The 32 bit field providing the HIT Hierarchy ID.

RAA (Registered Assigning Authority):
The 16 bit field identifying the Hierarchical HIT Assigning Authority.

RVS (Rendezvous Server):
The HIP Rendezvous Server for enabling mobility, as defined in [RFC8004].

3. Problem Space

3.1. Meeting the future of Mobile Devices in a public space

Public safety may impose a "right to know" what devices are in a public space. Public space use may only be permitted to devices that meet an exacting "who are you" query. This implies a device identity that can be quickly validated by public safety personal and even the general public in many situations.

Many proposals for mobile device identities are nothing more than a string of bits. These may provide information about the device but provide no assurance that the identity associated with a device really belongs to a particular device; they are highly susceptible to fraudulent use. Further they may impose a slow, complex method to discover the device owner to those with appropriate authorization.

The Host Identity Tag (HIT) from the Host Identity Protocol (HIP) provides a self-asserting Identity through a public key signing operation using the Host Identity's (HI) private key.

Although the HIT provides a "trust me, I am me" claim, it does not provide an assertion as to why the claim should be trusted and any additional side information about the device. The later could be distributed directly from the device in a secure manner, but again there is no 3rd-party assertion of such a claim.

3.2. Semi-permanency of Identities

A device Identity has some degree of permanency. A device creates its identity and registers it to some 3rd-party that will assert a level of trust for that identity. A device may have multiple identities to use in different contexts, and it may deprecate an identity for any number of reasons. The asserting 3rd-party may withdraw its assertion of an identity for any number of reasons. An identity system needs to facilitate all of this.

3.3. Managing a large flat address space

For HITs to be successfully used by a large population of mobile devices, they must support an Identity per device; potentially 10 billion Identities. Perhaps a Distributed Hash Table [RFC6537] can scale this large. There is still the operational challenges in establishing such a world-wide DHT implementation and how RVS [RFC8004] works with such a large population. There is also the challenge of how to turn this into a viable business. How can different controlling jurisdictions operate in such an environment?

Even though the probability of collisions with 7B HITs (one HIT per person) in a 96 bit flat address space is 3.9E-10, it is still real.

How are collisions managed? It is also possible that weak key uniqueness, as has been shown in deployed TLS certificates [WeakKeys], results in a much greater probability of collisions. Thus resolution of collisions needs to be a feature in a global namespace.

3.4. Defense against fraudulent HITs

How can a host protect against a fraudulent HIT? That is, a second pre-image attack on the HI hash that produces the HIT. A strong defense would require every HIT/HI registered and openly verifiable. This would best be done as part of the R1 and I2 validation. Or any other message that is signed by the HI private key.

4. The Hierarchical Host Identity Tag (HHIT)

The Hierarchical HIT (HHIT) is a small but important enhancement over the flat HIT space. By adding two levels of hierarchical administration control, the HHIT provides for device registration/ownership, thereby enhancing the trust framework for HITs.

HHITs represent the HI in only a 64 bit hash and uses the other 32 bits to create a hierarchical administration organization for HIT domains. Hierarchical HITs are "Using cSHAKE in ORCHIDs" [new-orchid]. The input values for the Encoding rules are in Section 4.3.5.

A HHIT is built from the following fields:

*28 bit IANA prefix

*4 bit HIT Suite ID

*32 bit Hierarchy ID (HID)

*64 bit ORCHID hash

4.1. HHIT prefix

A unique 28 bit prefix for HHITs is recommended. It clearly separates the flat-space HIT processing from HHIT processing per Section 4 of "Using cSHAKE in ORCHIDs" [new-orchid].

4.2. HHIT Suite IDs

The HIT Suite IDs specifies the HI and hash algorithms. Any HIT Suite ID can be used for HHITs, provided that the prefix for HHITs is different from flat space HITs. Without a unique prefix, Section 4.1, additional HIT Suite IDs would be needed for HHITs. This would risk exhausting the limited Suite ID space of only 15 IDs.

4.3. The Hierarchy ID (HID)

The Hierarchy ID (HID) provides the structure to organize HITs into administrative domains. HIDs are further divided into 2 fields:

*16 bit Registered Assigning Authority (RAA)

*16 bit Hierarchical HIT Domain Authority (HDA)

4.3.1. The Registered Assigning Authority (RAA)

An RAA is a business or organization that manages a registry of HDAs. For example, the Federal Aviation Authority (FAA) could be an RAA.

The RAA is a 16 bit field (65,536 RAAs) assigned by a numbers management organization, perhaps ICANN's IANA service. An RAA must provide a set of services to allocate HDAs to organizations. It must have a public policy on what is necessary to obtain an HDA. The RAA need not maintain any HIP related services. It must maintain a DNS zone minimally for discovering HID RVS servers.

This DNS zone may be a PTR for its RAA. It may be a zone in a HHIT specific DNS zone. Assume that the RAA is 100. The PTR record could be constructed:

```
100.hhit.arpa   IN PTR      raa.bar.com.
```

4.3.2. The Hierarchical HIT Domain Authority (HDA)

An HDA may be an ISP or any third party that takes on the business to provide RVS and other needed services for HIP enabled devices.

The HDA is an 16 bit field (65,536 HDAs per RAA) assigned by an RAA. An HDA should maintain a set of RVS servers that its client HIP-enabled customers use. How this is done and scales to the potentially millions of customers is outside the scope of this document. This service should be discoverable through the DNS zone maintained by the HDA's RAA.

An RAA may assign a block of values to an individual organization. This is completely up to the individual RAA's published policy for delegation.

4.3.3. Example of the HID DNS

HID related services should be discoverable via DNS. For example the RVS for a HID could be found via the following. Assume that the RAA is 100 and the HDA is 50. The PTR record is constructed as:

```
 50.100.hhit.arpa   IN PTR      rvs.foo.com.
```

The RAA is running its zone, 100.hhit.arpa under the hhit.arpa zone.

4.3.4. HHIT DNS Retrieval

The HDA SHOULD provide DNS retrieval per [RFC8005]. Assume that the Host_ID suite of EdDSA25519 (5), RAA of 10 and the HDA of 20 and the HHIT example is:

```
 2001:5:a:14:a3ad:1952:ad0:a69e
```

The HHIT FQDN is:

```
 2001:0005:a:14:a3ad:1952:0ad0:a69e.20.10.hhit.arpa.
```

The NS record for the HDA zone is constructed as:

```
 20.10.hhit.arpa   IN NS      registry.foo.com.
```

registry.foo.com returns a HIP RR with the HHIT and matching HI. The HDA sets its policy on TTL for caching the HIP RR. Optionally, the HDA may include RVS information. Including RVS in the HIP RR may impact the TTL for the response.

4.3.5. Changes to ORCHIDv2 to support Hierarchical HITs

A new format for ORCHIDs to support Hierarchical HITs is defined in "Using cSHAKE in ORCHIDs" [new-orchid]. For this use the following values apply:

```
Prefix     :=  HHIT Prefix
               Note: per section 4.1, this should be different
                     than the Prefix for RFC 7401
OGA ID     :=  4-bit Orchid Generation Algorithm identifier
               The HHIT Suite ID
Context ID :=  0x00B5 A69C 795D F5D5 F008 7F56 843F 2C40
Info (n)   :=  32 bit HID (Hierarchy ID)
Hash       :=  Hash_function specified in OGA ID
                   If hash is not a variable length output hash,
                   then en Encode_m, similar to ORCHID Encode_96
                   is used
m          :=  64
```

4.3.6. Collision risks with Hierarchical HITs

The 64 bit hash size does have an increased risk of collisions over the 96 bit hash size used for the other HIT Suites. There is a 0.01% probability of a collision in a population of 66 million. The probability goes up to 1% for a population of 663 million. See Appendix A for the collision probability formula.

However, this risk of collision is within a single HDA. Further, all HDAs are expected to provide a registration process for reverse lookup validation. This registration process would reject a collision, forcing the client to generate a new HI and thus hierarchical HIT and reapplying to the registration process.

5. IANA Considerations

Because HHIT use of ORCHIDv2 format is not compatible with [RFC7343], IANA is requested to allocated a new 28-bit prefix out of the IANA IPv6 Special Purpose Address Block, namely 2001:0000::/23, as per [RFC6890].

6. Security Considerations

A 64 bit hash space presents a real risk of second pre-image attacks. The HHIT Registry services effectively block attempts to "take over" a HHIT. It does not stop a rogue attempting to impersonate a known HHIT. This attack can be mitigated by the Responder using DNS to find the HI for the HHIT or the RVS for the HHIT that then provides the registered HI.

Another mitigation of HHIT hijacking is if the HI owner supplies an object containing the HHIT and signed by the HI private key of the HDA.

The two risks with hierarchical HITs are the use of an invalid HID and forced HIT collisions. The use of the "hhit.arpa." DNS zone is a strong protection against invalid HIDs. Querying an HDA's RVS for a HIT under the HDA protects against talking to unregistered clients. The Registry service has direct protection against forced or accidental HIT hash collisions.

7. Acknowledgments

The RDA/HDA 16/16 bit split, replacing the original 14/18 split was the result of discussions on lookup and implementation challenges of byte boundaries over nibble boundaries.

The initial versions of this document were developed with the assistance of Xiaohu Xu and Bingyang Liu of Huawei.

Sue Hares contributed to the clarity in this document.

8. References

8.1. Normative References

[new-orchid] Moskowitz, R., Card, S., and A. Wiethuechter, "Using cSHAKE in ORCHIDs", Work in Progress, Internet-Draft, draft-moskowitz-orchid-cshake-00, 11 December 2019, <https://tools.ietf.org/html/draft-moskowitz-orchid-cshake-00>.

[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>.

[RFC7401] Moskowitz, R., Ed., Heer, T., Jokela, P., and T. Henderson, "Host Identity Protocol Version 2 (HIPv2)", RFC 7401, DOI 10.17487/RFC7401, April 2015, <https://www.rfc-editor.org/info/rfc7401>.

[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>.

8.2. Informative References

[draft.moskowitz-hip-arch-02] Moskowitz, R., "Host Identity Payload", Superseded Internet-Draft, draft-moskowitz-hip-arch-02, 22 February 2001.

[draft.zhang-hip-hierarchical-parameter-00] Dacheng, Z. and X. Xiaohu, "Extensions of Host Identity Protocol (HIP) with Hierarchical Information", Abandoned Internet-Draft, draft-zhang-hip-hierarchical-parameter-00, 27 May 2009.

[hhit-registries] Moskowitz, R., Card, S., and A. Wiethuechter, "Hierarchical HIT Registries", Work in Progress, Internet-Draft, draft-moskowitz-hip-hhit-registries-02, 9 March 2020, <https://tools.ietf.org/html/draft-moskowitz-hip-hhit-registries-02>.

[RFC6537] Ahrenholz, J., "Host Identity Protocol Distributed Hash Table Interface", RFC 6537, DOI 10.17487/RFC6537, February 2012, <https://www.rfc-editor.org/info/rfc6537>.

[RFC6890] Cotton, M., Vegoda, L., Bonica, R., Ed., and B. Haberman, "Special-Purpose IP Address Registries", BCP 153, RFC 6890, DOI 10.17487/RFC6890, April 2013, <https://www.rfc-editor.org/info/rfc6890>.

[RFC7343] Laganier, J. and F. Dupont, "An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers Version 2 (ORCHIDv2)", RFC 7343, DOI 10.17487/RFC7343, September 2014, <https://www.rfc-editor.org/info/rfc7343>.

[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>.

[RFC8005] Laganier, J., "Host Identity Protocol (HIP) Domain Name System (DNS) Extension", RFC 8005, DOI 10.17487/RFC8005, October 2016, <https://www.rfc-editor.org/info/rfc8005>.

[WeakKeys] Heninger, N.H., Durumeric, Z.D., Wustrow, E.W., and J.A.H. Halderman, "Detection of Widespread Weak Keys in Network Devices", August 2012, <https://factorable.net/weakkeys12.extended.pdf>.

Appendix A. Calculating Collision Probabilities

The accepted formula for calculating the probability of a collision is:

```
p = 1 - e^{-k^2/(2n)}
```

```
P   Collision Probability
n   Total possible population
k   Actual population
```

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