HIP Resolution and Rendezvous Mechanisms

Lars Eggert, Julien Laganier, Marco Liebsch and Martin Stiemerling

***Abstract* — The Host Identity Protocol (HIP) decouples the name and locator roles that IP addresses serve in the current Internet architecture. This document discusses resolution mechanisms that map domain names into host identities and IP addresses and their effects of the overall HIP architecture, with a focus on rendezvous between nodes. It argues that HIP will benefit from removing its current dependencies on the presence of a deployed DNS infrastructure, resulting in a simpler, more modular system. Although such a system will require a new HIP resolution service to translate host identities into IP addresses, HIP will at the same time not require a dedicated rendezvous infrastructure anymore. Rendezvous servers become an optional component of the overall system that optimizes HIP performance in extreme situations, e.g., for highly mobile nodes, or enables advanced capabilities, such as location privacy.**

I. Introduction

THE current Internet uses two global namespaces: *domain names* and *IP addresses*. The first namespace – domain names – has a single use. Domain names, usually simply called names, are symbolic identifiers for sets of numeric IP addresses, chosen for their mnemonic properties: humans need to interact with them.

IP addresses form the Internet's second global namespace. They have two uses. First, they are topological *locators* for network attachment points, addressing a specific location in the network topology. Their second use is as *identifiers* for the network interfaces – and thus nodes – that attach to the addressed locations. In this role as identifiers, IP addresses loose their topological meaning and become simple names. Routing and other network-layer mechanisms use the locator aspects of IP addresses. Transport-layer protocols and mechanisms typically use IP addresses in their role as names for communication endpoints. (Saltzer [1] discusses these naming concepts in detail.)

This dual use of IP addresses as names and locators limits the flexibility of the Internet architecture. For example, the use of topology-dependent IP addresses as symbolic names for communication endpoints complicates node mobility. A mobile node changes its points of network attachment and hence its IP addresses dynamically. At the transport layer, this causes the logical endpoints of communication sessions to change dynamically as well. The Internet's transport protocols do not support changing the logical endpoints of an established communication session. Arguably, they should not, because the identity of the communicating nodes has not changed, simply their points of network attachment.

The *Host Identity Protocol* (HIP) architecture defines a third global namespace [3]. The new *host identity* namespace decouples the name and locator roles currently filled by IP addresses. Host identities take over the naming role, while IP addresses become pure locators. With HIP, transport-layer mechanisms operate on host identities instead of using IP addresses as endpoint names. Network-layer mechanisms continue to use IP addresses as pure locators.

Due to the introduction of a new global namespace, HIP also affects the Internet's current resolution services. The *Domain Name System* (DNS) is currently the Internet's single, global resolution service [2]. The DNS provides a two-way lookup service between domain names and their set of corresponding IP addresses. HIP requires an additional resolution step. Domain names now map into sets of host identities, which in turn map into sets of IP addresses.

The additional HIP resolution step complicates the *rendezvous* procedure by which two nodes establish a communication channel. In the current Internet, the DNS maps the domain name of a target remote node into its set of IP addresses, which the local node may then use to address packets. The address of each node's DNS server is preconfigured. In the absence of a preconfigured DNS server, nodes can still communicate by using IP addresses directly.

With HIP, the rendezvous procedure and resolution mechanisms are becoming more complex. The various alternatives for performing name and identity resolutions lead to rendezvous procedures that offer significantly different characteristics. This paper discusses these alternatives to aid the design of the overall HIP architecture. Section II presents different options for HIP resolution and rendezvous mechanisms. Section III discusses future work and concludes this document.

II. Resolution and Rendezvous

As mentioned in Section I, HIP complicates the Internet's simple resolution and rendezvous procedures. Currently, nodes use DNS servers at preconfigured, well-known IP addresses to resolve domain names into IP addresses, which they can then use to address packets. The left illustration in Figure 1 shows this resolution procedure. It also shows the *reverse resolution*, which resolves an IP address back into its associated domain name.

With HIP, domain names map into sets of host identities, each of which maps into sets of IP addresses. This results in a logical two-step resolution process before a node knows the IP addresses associated with target domain name. The middle illustration in Figure 1 shows this two-step process. To maintain application compatibility, the first mapping – from names into host identities – should remain in the DNS. For the second mapping – from host identities into IP addresses – various alternatives are possible. Logically, this *HIP lookup* is

Manuscript received October 1, 2004. Parts of this work are a product of the *Ambient Networks*, *Daidalos* and *Enthrone* projects supported in part by the European Commission under its *Sixth Framework Programme*. It is provided "as is" and without any express or implied warranties, including, without limitation, the implied warranties of fitness for a particular purpose. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the *Ambient Networks*, *Daidalos* or *Enthrone* projects or the European Commission.

Lars Eggert, Marco Liebsch and Martin Stiemerling are with NEC Europe Ltd, Network Laboratories, Heidelberg, Germany (phone: +49-6221-905-1143; fax: +49-6221-905-1155; e-mail: {eggert, liebsch, stiemerling}@netlab.nec.de.)

Julien Laganier is with SUN Microsystems Laboratories, Grenoble, France and with the Laboratoire de l'Informatique du Parallelisme (UMR CNRS - ENS Lyon - UCB Lyon - INRIA), Ecole Normale Superieure de Lyon, France (phone: +33-476-188-815; fax: +33-476-188-888; e-mail: ju@sun.com.)

a completely separate operation from the initial *DNS lookup*, as shown in the middle illustration of Figure 1.

Currently deployed HIP prototypes choose to maintain the second mapping between host identities and IP addresses in the DNS as well. One proposal simply stores a node's host identities alongside its IP addresses in the node's DNS record [4]. A DNS resolution of a domain name thus returns a pair of host identities and IP addresses, as shown in the left illustration of Figure 1. This simplistic approach creates several problems that the following section discusses in more detail. Section II.B then discusses an explicit two-step resolution process and the final two sections of this document discuss the rendezvous of this approach.

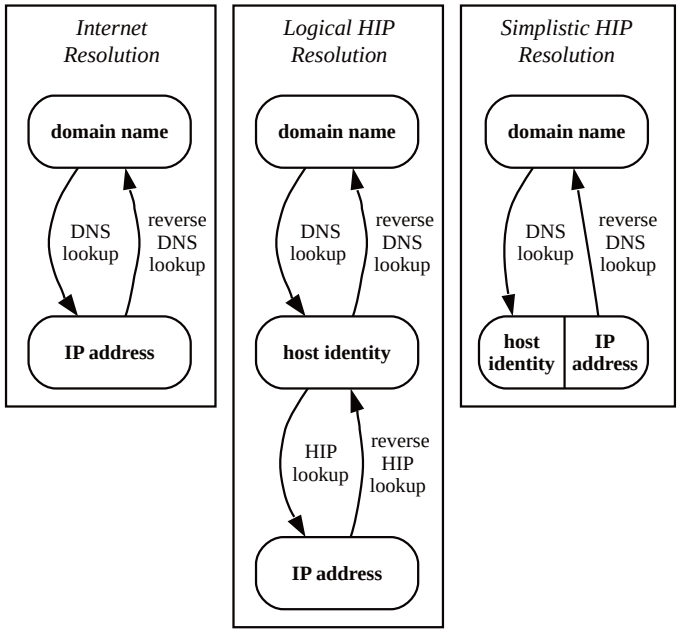

Figure 1. Domain name resolution without HIP (left) and with HIP (middle and right.)

A. Issues with Simplistic HIP Resolution

One critical problem is that storing host identities in a node's DNS record creates a dependency between HIP and the DNS. To communicate with HIP, DNS resolution of a domain name is required to obtain a peer's host identities and IP addresses. It is not possible to communicate with HIP based on host identities alone – no resolution mechanism exists to map host identities into IP addresses. This is a drastic change from the current Internet, where the DNS is an optional component and communication can occur based on IP addresses alone.

This causes a second issue: even if a node already knows the host identity of a peer, it cannot communicate with it without knowing and resolving the peer's domain name. With HIP, host identities should replace IP addresses everywhere above the network layer. Applications – and users – may substitute host identities wherever they now use IP addresses. A direct mechanism to resolve host identities into IP addresses, *i.e.*, one that does not depend on knowledge of the corresponding domain name, is required to enable this transparency. (Communication based on IP addresses alone is still possible with the simplistic HIP lookup, but obviously will not incur the benefits of HIP.)

Another problem with the simplistic HIP resolution shown on the left of Figure 1 is that no general mechanism exists to perform a reverse HIP lookup, *i.e.*, determine the domain name of a node based on its host identity. Only the traditional reverse DNS lookup exists, which operates on IP addresses, not host identities. Although this capability could be added to the DNS through a new root, similar to reverse lookups on IP addresses, this approach is cumbersome [4].

Rendezvous with the DNS infrastructure is a fourth issue with the simplistic HIP lookup. It may be useful to communicate with DNS servers using HIP instead of IP, *i.e.*, access a DNS server through its well-known host identity instead of its well-known IP address. This would enable DNS servers to benefit from HIP's mobility, multi-homing and security mechanisms. The simplistic HIP lookup requires a DNS infrastructure that remains accessible at well-known IP addresses.

B. Two-Step HIP Resolution

A resolution mechanism that follows the logical procedure shown in the middle of Figure 1 will address these issues. Under this scheme, the DNS maps domain names into host identities and back. Because host identities can be formatted to resemble IPv6 addresses, this is a simple modification. The DNS record types described for use with HIP may already support this [4]. HIP nodes only operate on the host identities a domain name lookup returns; they ignore any IP addresses that the record may contain. (The IP addresses in a HIP node's DNS record are only for communication with legacy, non-HIP nodes as described in Section II.D.)

To communicate with a peer, a HIP node resolves the peer's host identity into a set of IP addresses in a separate, second HIP lookup operation. Note that it is irrelevant how the node obtains the peer's host identity, be it from the DNS, is preconfigured, well-known or communicated in-band by another node. The result of the HIP lookup is a set of IP addresses the node may use to address packets to the peer.

With this two-step resolution, HIP no longer depends on the DNS. Communication based only on host identities is possible. Likewise, reverse lookups on host identities and IP addresses become possible, depending on the specifics of the resolution systems. Finally, accessing DNS infrastructure based on host identities becomes possible.

These capabilities do not come free. An explicit HIP lookup introduces a second, global resolution service into the Internet. Unlike the DNS, the HIP resolution service is mandatory – without it, no HIP communication can occur. The specifics of such a resolution service are currently not clear. A name service based on a distributed hash table [6], possibly accessed through anycast [7], might be a useful direction for further research.

C. Rendezvous: Resolution vs. Forwarding

Similar to the explicit HIP lookup step outlined in Section II.B, other proposals recognize the need for a HIP infrastructure that operates directly on host identities. The *Host Identity Indirection Infrastructure* (Hi3) describes a distributed forwarding plane for HIP control messages that routes based on host identities, not IP addresses.

Hi3 is mainly a rendezvous mechanism to deliver (some) HIP handshake packets. After the handshake, data communication flows end-to-end based on IP addresses. When nodes

move, they signal changes in addressing to the Hi3 infrastructure.

Although similar in some aspects to Hi3, the HIP lookup service is conceptually simpler. It is a pure lookup service that does not provide communication services of any kind. All communication, including HIP control traffic, occurs end-to-end based on IP addresses.

With the HIP lookup service, rendezvous is arguably simpler than with Hi3. There is no explicit rendezvous architecture. When nodes move, they update their information in the lookup service, similar to how they signal Hi3. It remains up to the end systems, however, to address the HIP handshake packets.

D. Rendezvous Service

Even with the HIP lookup service, a specialized infrastructure to establish node rendezvous may still be useful [10][11]. The next sections discuss a number of issues supporting such a rendezvous infrastructure. *Rendezvous servers* are fixed infrastructure that relay packets on behalf of mobile HIP nodes.

1) Update Latency

As with any name service, a HIP lookup service must find the right balance between lookup and update performance. Caching is an effective technique to decrease lookup delays. On the other hand, caching can increase update delays due to the involved cache consistency mechanisms, e.g., invalidation.

If a HIP node changes IP addresses faster than the propagation time of its address updates to the HIP lookup service and established peers, it can become unreachable. Dedicated rendezvous servers can improve operation in these cases. They establish a fixed target for peer nodes to send their packets to, while enabling the mobile node to use a faster update mechanism for the local forwarding state on the rendezvous server. The drawback of rendezvous servers is that they introduce triangle routing: packets no longer follow the direct path between two peers, but instead flow through the rendezvous server. Besides increasing the end-to-end latency, this may decrease the reliability of the connection.

2) Location Privacy and Traceability

Internet users are becoming more sensitive to privacy concerns. For example, the introduction of IPv6 already caused concern because of the possibility to trace users based on the unique EUI48 NIC identifiers included in their global IPv6 addresses.

HIP may potentially worsen the situation through its use of cryptographic, semi-permanent identifiers. One approach to mitigating these concerns is through the periodic regeneration of host identities. Instead of reusing the same identity, nodes will generate new identities on the fly, similar to a similar RFC 3041 [9]. This approach makes it more difficult to correlate a node's HIP associations and may thus reduce traceability concerns.

A second approach to increasing location privacy is concealing the IP addresses of two communicating nodes from one another. The *SPI-multiplexed NAT* (SPINAT) described as part of the BLIND framework [8] offers this ability. Besides relaying the rendezvous exchanges, they may also relay the following data traffic. Certain protocol features of HIP rendezvous servers may also eventually support similar levels of concealment [10].

Pushing the HIP lookup further into the network is another effective means of concealing the actual IP addresses of two communicating HIP nodes from one another [11]. Under this approach, HIP nodes do not resolve host identities into IP addresses themselves. They rather forward packets that contain unresolved host identities to a network entity that performs the HIP lookup on their behalf, as illustrated in Figure 2.

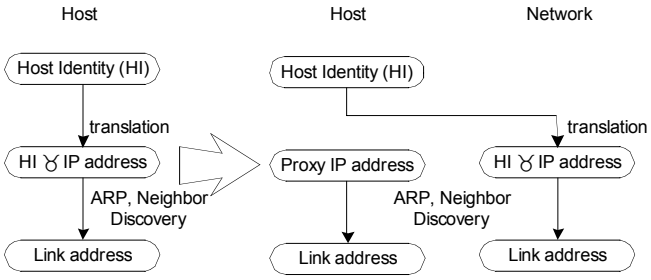

Figure 2: Relocating the HIP lookup into the network.

3) Interoperation with Non-HIP Nodes

HIP and non-HIP nodes may simply communicate using IP addresses directly. The downside is that their communication will then not receive the benefits associated with using HIP. One approach for communication with legacy hosts utilizes rendezvous servers as permanent relays [11]. HIP nodes obtain static, globally routed IP addresses from their rendezvous servers. Non-HIP nodes use these addresses to communicate with the HIP nodes. The rendezvous servers relay traffic to these addresses to the current network attachment points of their associated HIP nodes.

Non-HIP nodes expect the DNS resolution of a domain name to yield an IP address (i.e., an A or quad-A record) that may be used to address packets to the target. HIP nodes must consequently update their DNS records, entering the static IP addresses they obtain for legacy communication. Note that HIP-enabled nodes would ignore these address records. Instead, they perform a HIP lookup on the returned host identity the DNS lookup returns and use the IP addresses returned in this second lookup step to address packets to the destination.

This scheme combines the benefits of direct HIP-based communication, which may not involve a rendezvous server, with support for non-HIP nodes, which depend on the presence of valid endpoint IP addresses in the DNS.

4) Middlebox Traversal

Many middleboxes, such as firewalls and network address translators, protect a network by dropping apparently unsolicited inbound traffic. They often only permit inbound return traffic associated with previous, internally initiated communication [12]. This behavior generally restricts communication and is problematic for a large number of protocols. For HIP, this behavior prohibits peer nodes from establishing HIP associations with nodes behind such middleboxes.

Rendezvous servers can mitigate some of these issues. Registration with a rendezvous service is an internally initiated communication that may traverse middleboxes that protect a network more easily. Other peers can then initiate association establishment with a HIP node behind a middlebox through the rendezvous mechanism, allowing this externally originated traffic to reach the protected HIP node. This is similar to the *Teredo* mechanism for deploying IPv6 [13].

III. Conclusion

This document discussed the interdependencies of various lookup operations required for HIP communication and investigated how they interact with HIP rendezvous establishment. It argued that HIP would benefit from removing its current dependencies on the presence of a deployed DNS infrastructure, resulting in a simpler, more modular system. Although such a system will require a new HIP resolution service to translate host identities into IP addresses, HIP will at the same time not require a dedicated rendezvous infrastructure anymore. Rendezvous servers become an optional component of the overall system that optimizes HIP performance in extreme situations, e.g., for highly mobile nodes, or enables advanced capabilities, such as location privacy.

References

[1] Jerome Saltzer. On the Naming and Binding of Network Destinations. RFC 1498, August 1993.

[2] Paul Mockapetris. Domain names - concepts and facilities. STD 13, RFC 1034, November 1987.

[3] Robert Moskowitz and Pekka Nikander. Host Identity Protocol Architecture. Work in Progress (draft-moskowitz-hip-arch-06), June 2004.

[4] Julien Laganier and Pekka Nikander. Host Identity Protocol (HIP) Domain Name System (DNS) Extensions. Work in Progress (draft-nikander-hip-dns-00), May 2004.

[5] Pekka Nikander, Jari Arkko and Börje Ohlman. Host Identity Indirection Infrastructure (Hi3.) Work in Progress (draft-nikander-hiprg-hi3-00), June 24.

[6] Venugopalan Ramasubramanian and Emin Gün Sirer. The Design and Implementation of a Next Generation Name Service for the Internet. Proc. *ACM SIGCOMM*, Portland, OR, USA, August 30 - September 3, 2004, pp. 331-342.

[7] Craig Partridge, Trevor Mendez and Walter Milliken. Host Anycasting Service. RFC 1546, November 1993.

[8] Jukka Ylitalo and Pekka Nikander. BLIND: A Complete Identity Protection Framework for End-points. Proc. *Twelfth International Workshop on Security Protocols*, Cambridge, England, April 26-28, 2004.

[9] Thomas Narten and Richard Draves. Privacy Extensions for Stateless Address Autoconfiguration in IPv6. RFC 3041, January 2001.

[10] Lars Eggert and Julien Laganier. Host Identity Protocol (HIP) Rendezvous Extensions. Work in Progress (draft-eggert-hip-rvs-00), July 2004.

[11] Lars Eggert and Marco Liebsch. Host Identity Protocol (HIP) Rendezvous Mechanisms. Work in Progress (draft-eggert-hip-rendezvous-01), July 2004.

[12] Martin Stiemerling and Jürgen Quittek. Problem Statement: HIP operation over Network Address Translators. Work in Progress (draft-stiemerling-hip-nat-01), July 2004.

[13] Christian Huitema. Teredo: Tunneling IPv6 over UDP through NATs. Work in Progress (draft-huitema-v6ops-teredo-02), March 2004.