1

Overview

COPYRIGHTED MATERIAL

The current Internet architecture, though hugely successful, faces many difficult challenges. The most important ones are the incorporation of mobile and multihomed terminals (hosts) and an overall lack of protection against Denial-of-Service attacks and other lacking security mechanisms. With gradual deployment of IPv6, interoperability between the "old" IPv4 Internet and the "new" IPv6 Internet becomes another challenging problem. Finally, the ongoing convergence on IP-based solutions for telecommunication networks, such as with UMTS/3G and its successors, increases the need to resolve these problems as soon as possible.

Although many of these problems have been widely recognized for some time, a complete and adequate solution is still missing. Most existing approaches are point-solutions that patch support for a subset of the required improvements into the current Internet architecture, but do not cleanly integrate with one another and do not present a stable base for future evolution. As an example, Mobile IP provides some support for host mobility, but still has major security flaws that prevent its widespread deployment.

This book is dedicated to the Host Identity Protocol (HIP), which is a promising new basis for a secure mobile Internet. The cornerstone of HIP is the idea of separating a host's identity from its present topological location in the Internet. This simple idea provides a solid basis for mobility and multihoming features. HIP also includes security as an inherent part of its design, because its "host identities" are cryptographic keys that can be used with many established security algorithms. For example, these cryptographic identities are used to encrypt all data traffic between two HIP hosts by default. Finally, the HIP specifications are patent-free (to our current best knowledge), which is an important criterion for adoption in both commercial and open source Internet platforms.

This chapter starts with Section 1.1 presenting the split in the host identifiers and locations as an important goal for future Internet. The place of HIP in the Internet architecture is discussed in Section 1.2. The origins and history of HIP development in IETF are given in Section 1.3. Section 1.4 walks through the book chapter by chapter to give the reader a general feeling on how the book is structured.

Host Identity Protocol (HIP): Towards the Secure Mobile Internet Andrei Gurtov
© 2008 John Wiley & Sons, Ltd

1.1 Identifier–locator split

Routing of IP datagrams in the Internet is based on network prefixes of destination IP addresses. It would be impractical to demand that each Internet router know the next hop for each individual IP address. It would make the routing tables excessively large, increasing the memory costs, and slowing down the lookups. Instead, IP addresses are aggregated and essentially located in a close geographical area. An IP address serves as a *locator* of the host in the Internet topology.

The host *identifier* is something that tells other communicating entities that the host is still the same despite a possible change of its location in the Internet. DNS name is one such widely-deployed form of identifier. In the early days of the Internet, when hosts were stationary, it was possible to use the IP address as a host identifier. In the present Internet architecture, the role of IP addresses as identifier and locators are still mixed, as shown in Figure 1.1, though the old assumptions are not valid anymore. Many hosts are mobile and multihomed. There, each separate service is using own socket as it should, but the endpoint identity is directly attached to the IP address of the interface.

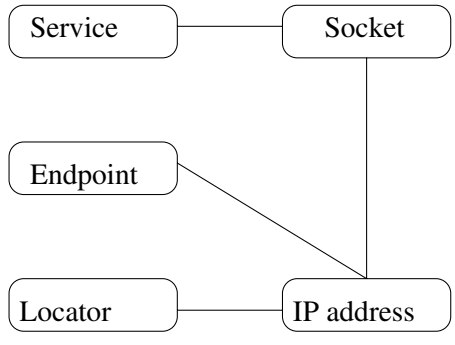

Figure 1.1 Location and identity of hosts are combined in the Internet.

Today many hosts use public IPv4 addresses assigned from a common DHCP pool or ephemeral IPv4 addresses assigned privately behind a NAT. Therefore, an IP address cannot often even be used as a robust identifier. As an example, blacklisting a certain IPv4 address as a source of SPAM messages can in fact disable an innocent host who received the same address later from DHCP or even prevent a large number of hosts using a public IPv4 address of a NAT.

The main architectural theme behind HIP is the split of host identifier and locator (Moskowitz and Nikander 2006). An appropriate security mechanism to enable the host to prove its identity is essential. A long randomly generating string would be sufficient to identify a host in a trusted environment, but not in a public Internet. In HIP architecture, a self-generated public–private key pair becomes the host identity. Figure 1.2 illustrates the positioning of host identity between socket and network interfaces. Now the sockets are bound to the host identity instead of a locator. In fact, there could be several locators

associated simultaneously with an identity. The locators can be dynamically added or removed from the active set without any changes on the socket interface.

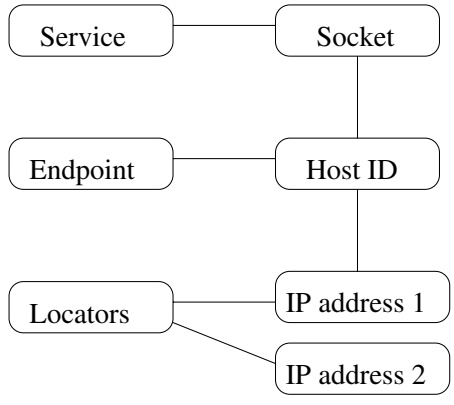

Figure 1.2 Separating location and identity of Internet hosts.

A single host can have multiple identities. Self-asserted unpublished identities are suitable for anonymous communication in a trustworthy way. The peer hosts can reliably verify that the new connection is coming from the same host as before. Host identifiers placed into Secure DNS (DNSSEC), asserted by Public Key Infrastructure (PKI) X.509 certificates, or Pretty Good Privacy (PGP) can be used to authenticate the host.

A single identity can be shared among a group of hosts that possess the same private key. Such hosts can, for example, form a multicast group or perform other distributed operations. Given the challenge of securely distributing private keys over the network, group host identities are still in the research phase and are not supported by the current HIP specifications.

1.2 HIP in the Internet architecture

Since the beginning of the Internet, the IP protocol has been the only routable network-layer protocol in use. Deployment of IPv4 occurred using a “flag day” when the old IP version was not routed anymore. Thanks to its simplicity, the IP protocol is able to run over a wide range of link technologies, including Ethernet (IEEE 802.3), Wireless LAN (IEEE 802.11), Token Ring (IEEE 802.2) and many others. On the other hand, multiple transport protocols can run on top of IP. The TCP and UDP protocols were the only ones widely deployed in practice. The number of applications using the transport protocols is large, with the most important ones being HTTP, SMTP, and FTP. The resulting model of the Internet protocol stack resembles the hourglass as shown in Figure 1.3. The IP is the narrowest part of the stack and sometimes called a waist of the Internet.

A major problem in the original Internet architecture is tight coupling between networking and transport layers. As an example, TCP uses a pseudoheader including IP addresses

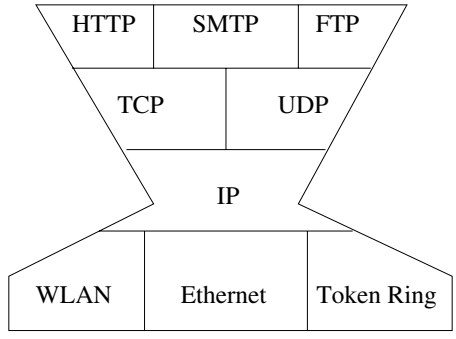

Figure 1.3 IP as a waist of the Internet protocol stack.

in checksum calculations. Therefore, independent evolution of two layers is not possible. Introduction of a new networking or transport protocol requires changes to other layer. In the proposed OSI model of a network stack, different protocol layers are mutually isolated. However, the model has not been realized in the Internet protocol stack.

As many of us have noticed, with passing years it becomes more difficult to control the waist volume. The same happened to the Internet with the introduction of IPv6. The scale of the Internet has grown dramatically and deployment of a new IP version with a flag day is not any more feasible. Now in dual-stack IP networks, both versions of IP protocol need to be simultaneously routed. This creates a challenge for inter-operating the legacy IPv4-only HIP applications with new IPv6 applications. Introduction of the HIP architecture can restore the original Internet hourglass model as shown in Figure 1.4. Then, HIP becomes a new waist of the Internet protocol stack replacing IPv4 in this role. IPv4 and IPv6 run underneath HIP, there as the transport protocols including the new SCTP and DCCP on top of HIP.

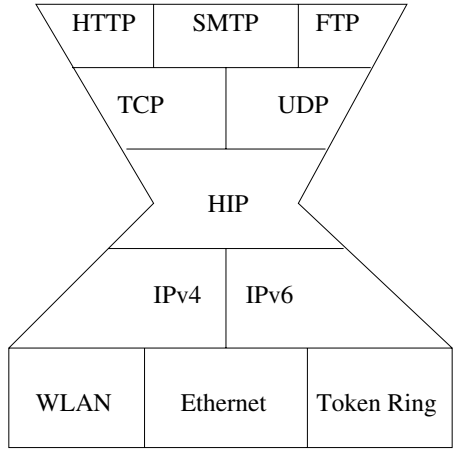

Figure 1.4 HIP as a new waist of the Internet protocol stack.

Another major problem of the present Internet is Denial-of-Service (DoS) attacks. During establishment of a TCP connection, the server creates a significant state by allocating TCP control block structure after replying to a SYN packet with a SYN-ACK packet. At that stage, the server has no assurance even that the SYN has arrived from the same host as stated in the SYN source IP address. Using this exploit, a moderate number of hosts can swamp the server with SYN messages, exhausting the server's memory or other system resources. Therefore, an important requirement of a new interworking protocol is to prevent creating the state at the server before the client is verified to respond from acclaimed IP source address. HIP implements this task and in addition uses cryptographic puzzles to prevent the client generating connection attempts at an overly fast rate. The puzzle is basically asking a client to reverse a hash function of a small length that statistically requires significant computational resources. On the other hand, verifying the puzzle at the server is a short operation.

Figure 1.5 shows a well-known structure of the present IP stack. The application creates a Berkeley socket using the IP address and transport protocol family of the peer. The state created at a transport layer is also using the IP address in addition to transport protocol-specific port numbers to deliver data segments to a correct application. The network layer uses the destination IP address to determine a right transmission link for a packet. The Network Interface Card (NIC) address is added on the link for transmission.

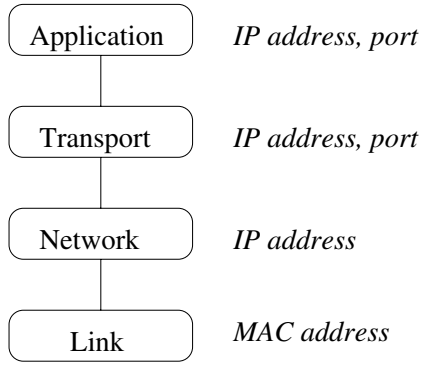

Figure 1.5 The IP protocol stack.

HIP takes place as a shim sub-layer between the network and transport layers, as shown in Figure 1.6. Now the applications and transport protocols use the host identity tag in their messages. The HIP sub-layer maps HITs to the best IP address of the destination internally before passing a packet to the networking layer. Transmission of the packet afterward follows the same pattern as in a regular IP stack.

1.3 Brief history of HIP

The problem of naming of hosts and data in the Internet has been discussed for a long time in the Internet Engineering Task Force (IETF). As an example, RFC 1498 from 1993 reprints the paper on naming and binding of network destinations originally from 1982 (Saltzer 1993).

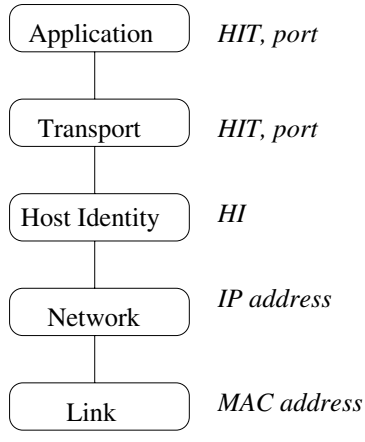

Figure 1.6 The protocol stack of HIP.

The paper starts from considering a resource name, address and route as the basic definitions used by John Shoch. The definitions are extended to cover services and users, network nodes, network attachment points, and paths. The paper identified three successive and changeable bindings of a service to a node, a node to an attachment point, and an attachment point to a route.

The Name Space Research Group (NSRG) has been active in IRTF from 1999 to 2003. It produced a report on the use of other namespaces than the 32-bit IPv4 addresses.

The original inventor of HIP was Robert Moskowitz from ICSA, Inc. The draft-moskowitz-hip-00 was published as an individual submission in the IETF in May 1999. Quoting its acknowledgment section: "The drive to create HIP came to being after attending the MALLOC meeting at IETF 43. It is distilled from many conversations from the IPsec mailing list and the IPsec workshops. Particularly Rodney Thayer should be mentioned for giving this protocol its initial push. Steve Bellovin assisted on some of the public key and replay concerns. Baiju Patel and Hilarie Orman gave extensive comments on the initial format, resulting in the present document. Hugh Daniels and IPsec implementers have kept after me to see that HIP moved beyond concept to spec".

From 1999 to 2002, Robert Moskowitz has held several informal meetings during the IETFs. Several revisions of the HIP architecture and protocol specifications were published as individual submissions.

In 2002, Pekka Nikander became interested in HIP and took over the leading of the standardization effort from Robert Moskowitz, who had other commitments. Together with a number of people at Ericsson NomadicLab, Boeing and HIIT he developed new packet structure, the state machine and the protocol details. The specifications were published as individual submissions until 2004.

After a period of background development, an IETF working group on HIP was created and draft-ietf-hip-base-00 was published in June 2004. The HIP WG is chaired by David Ward (Cisco) and Gonzalo Camarillo (Ericsson). The purpose of the WG was "to define the minimal elements that are needed for HIP experimentation on a wide scale".

The first outcome of the group was the overview of HIP architecture. Originally the group focused on creating the HIP base exchange and ESP encapsulation specifications, mobility and multihoming extensions, DNS and rendezvous, and registration extensions.

In late 2006, the WG was re-charted to include legacy NAT traversal, the application support and native API as WG items. At the time of writing (early 2007) the HIP specifications are under review process in IESG and their publication is expected shortly. The document quality meets the requirements for standard-track RFCs, but their status is "experimental" due to unknown implications on HIP deployment to the Internet. It is expected that after more experience with the use of HIP in real networks is obtained, the specifications could be elevated to the standard-track status. The document specifying an IPv6 prefix for HIP is published as an independent submission.

A HIP Research Group was chartered at the Internet Research Task Force (IRTF) in 2004. From the establishment Pekka Nikander (Ericsson) and Tom Henderson (Boeing) had served as the chairs of HIP RG. In 2005, Andrei Gurtov (HIIT) replaced Pekka Nikander after his election to the Internet Architecture Board (IAB). The task of HIP RG is to evaluate the impact of wider HIP deployment on the Internet and develop experimental protocol extensions that are not yet ready for standardization in the IETF.

1.4 Organization of the book

The book is organized into four parts: introduction, protocol specifications, infrastructure support, and applications. The first part includes a general HIP overview and background on cryptography and Internet protocols. The second part covers the base protocol, the core extensions for mobility and multihoming, rendezvous and DNS, as well as advanced extensions such as the HIP opportunistic mode. The third part describes changes to the Internet infrastructure useful for wider HIP deployment, such as middlebox traversal and name resolution. The last part covers practical use cases for HIP and its possible use with other protocols, such as SIP and Mobile IP.

In the overview (Chapter 1), we outlined the problems that the current Internet faces and provided the motivation for HIP, as well as the range of existing solutions and their shortcomings. Chapter 2 presents the necessary background on public–private key and symmetric cryptography and Internet protocols (IPsec, Internet Key Exchange). Readers that are familiar with all or some of this background information can skip these sections.

Part II starts with Chapter 3 on the place of HIP in the Internet architecture. It presents the concept of Internet namespaces, position of HIP within the TCP/IP stack, and construction of host identifiers. Chapter 4 describes the core parts of the HIP specifications that establish a secure connection between two hosts in a way that is DoS-resistant and makes spoofing difficult. The HIP base exchange packets I1, R1, I2 and R2 are explained in detail, as well as NOTIFY, UPDATE, CLOSE, and CLOSE_ACK HIP control packets. The IPsec encapsulation of HIP data packets within a Bound End-to-End Tunnel (BEET) is presented.

Chapter 5 then proceeds to describe extensions to HIP that enable host mobility and multihoming (the UPDATE packet and the LOCATOR parameter), the role of a rendezvous server and registering with it, and DNS extensions. Chapter 6 covers advanced extensions. The HIP opportunistic mode enables a HIP host to establish a HIP association to another

host without prior knowledge of its identity. Piggybacking of data to HIP control packets can reduce the HIP association establishment time. HIP service discovery extensions enable a HIP host to locate available service, such as a rendezvous server, in the local network. Simultaneous multiaccess enables efficient utilization of multiple network paths between the HIP host. The SIMPLE presence protocol can be used to exchange host identities for HIP. Chapter 7 presents implementation and performance results of using HIP on lightweight hardware, a Linux PDA. Chapter 8 concludes the second part with a description of lightweight HIP that suits well for Internet hosts with modest hardware resources.

Part III describes extensions to the Internet infrastructure that enable wide-scale HIP deployment. The part starts with Chapter 9, which focuses on the interactions between the end-to-end HIP and middleboxes located on the path between two HIP hosts. Because the base HIP has difficulties in passing through NATs and firewalls in the Internet, the NAT traversal extensions for HIP form an important part of this chapter. The chapter also considers HIP-aware middleboxes that are designed to explicitly support HIP.

Chapter 10 covers the process of mapping of host name to its current locator, also known as the name resolution. The chapter starts with describing the requirements for the name resolution service and an overview of Distributed Hash Tables (DHTs). The interface between OpenDHT and the HIP host is described to insert and lookup HIT-IP mappings. Next, the use of overlay networks for routing of HIP control packets is introduced. The Host Internet Indirection Infrastructure is described in detail. In the initial HIP deployment phase, many of the infrastructure services are running on PlanetLab, a distributed research testbed of roughly 700 servers worldwide. Chapter 11 introduces extensions to HIP architecture that enable host micromobility. The third part is concluded with Chapter 12 outlining protocol and infrastructure extensions to support location privacy of HIP hosts.

Part IV starts with Chapter 13 listing several practical applications that benefit from the use of HIP protocol. As a real-world case study, we describe the HIP deployment in a wireless testbed on a Boeing airplane factory. Chapter 14 presents an interface between HIP and applications. First, compatibility mechanisms that let existing applications benefit from communicating over HIP without the need for application modifications are described. The Native API enabling the use of wider functionality for new applications is presented next.

Chapter 15 discusses the integration of HIP with other protocols, such as SIP and Mobile IP. Experimental proposals for replacing the HIP base exchange or IPsec encapsulation with IKEv2 or SRTP are presented. The chapter concludes with description of a HIP proxy that enables the legacy hosts to benefit from HIP.

Appendix A describes how HIP can be gradually deployed on the readers' own computers and what the benefits are. In particular, this chapter includes practical advice and tutorial-style sections on downloading and installing an existing HIP implementation, utilizing PlanetLab's HIP services, and simple "first steps" experiments with HIP using the Wireshark network analyzer.