Adoption Barriers of Network-layer Protocols: the Case of Host Identity Protocol[☆]

Tapio Levä[a,*], Miika Komu[b], Ari Keränen[c], Sakari Luukkainen[b]

[a] *Department of Communications and Networking, Aalto University, Espoo, Finland*
[b] *Department of Computer Science and Engineering, Aalto University, Espoo, Finland*
[c] *Ericsson Research, Jorvas, Finland*

Abstract

With increasing societal dependence on the Internet and new application areas emerging, the need for securing communications and identifying communication partners is expected to increase. However, the original Internet architecture is lacking these functionalities, and most of the protocols proposed to fix these issues have not been widely deployed. Often one of the reasons for such failure is that protocol designers have insufficient understanding of the potential adopters' economic incentives so one may end up designing protocols based on false or inaccurate assumptions. In this paper, we analyze the Host Identity Protocol (HIP) from this viewpoint. Based on 19 expert interviews, we identify six main reasons why HIP has not been widely deployed yet. Most importantly, 1) the demand for the functionalities of HIP has been low. Where demand would have existed, substitute solutions have been favored because 2) they were earlier on the market, 3) they have relative advantage due to some design choices of HIP, 4) HIP lacks early adopter benefits necessitating costly coordination among multiple stakeholders in public deployment scenarios, and 5) people have misconceptions about the deployability of HIP. Additionally, 6) the research-mindedness of HIP developers has lead to strategic mistakes and non-optimal design choices from the perspective of deployment. We also suggest strategies that HIP develop-

[☆]Accepted manuscript. Please cite this article as: Tapio Levä, Miika Komu, Ari Keränen, Sakari Luukkainen (2013). Adoption barriers of network layer protocols: The case of host identity protocol. Computer Networks, vol. 57, no. 10, pp. 2218-2232, DOI: 10.1016/j.comnet.2012.11.024.

[*]Corresponding author. Email address: tapio.leva@aalto.fi.

ers could take to foster the adoption of HIP. Besides providing value to HIP developers, the results propose some new adoption barriers and deployment strategies that could be taken into account when designing new protocols. Finally, the article also provides a template that could be followed when studying the feasibility of other protocols.

Keywords: Host Identity Protocol, Internet standard, adoption, deployment, stakeholder incentives

---

1. Introduction

The Internet was originally designed as a simple network that provides just the basic services for packet switching. During the last three decades, several attempts have been made towards extending the basic functionality of the Internet, for example, to support interdomain quality of service, IP multicast, IPv6 and DNS security. Although most networking professionals agree that the Internet would benefit from these evolutionary changes, they appear not to be forthcoming. As a result, the core protocols that comprise the Internet architecture have ossified [1], and the innovation has moved up to the application layer.

Semantic overload of IP addresses acting both as locators and identifiers has been identified as one of the core problems in the current Internet [2, 3]. Lack of persistent identities for hosts makes it difficult to support mobile devices and security. Even though these features were not considered important when the Internet was originally designed, the increasingly mobile users and business criticality of Internet communications increases the need for them. Furthermore, with advent of the Internet of Things and information-centric networking, identifying objects and data is expected to become increasingly important in the future.

The Host Identity Protocol (HIP) [4] proposes a persistent identity for a networking device to identify it independently of its location. With the help of the identity, network streams can be relocated when the device moves from a network to another to support end-host mobility and also multihoming. The identity is effectively the public key from private-public key pair, so it can be used to authenticate an end-host securely. Later, the HIP community discovered that the identity is also convenient because it is statistically unique and, thus, can be used to distinguish a host from others in overlapping private address realms as introduced by NAT devices. Separate extensions

exist [5] that support penetration of HIP through NAT boxes to support end-to-end connectivity and facilitate new types of P2P-oriented services and applications for the future Internet.

Despite of significant R&D efforts, the deployment of HIP has been minimal. Even though other authors have compared HIP to other protocols [6, 7], identified security risks [8], and analyzed the experiences gathered from the experiments [9], the adoption barriers of HIP have not been studied before from the perspective of stakeholders' incentives. This perspective is important because stakeholders' adoption decisions are driven by economic interests. Additionally, Internet protocols, including HIP, typically have strong network externalities [10], i.e. the benefit of adopting an innovation is a function of the number of current and potential adopters, due to which the adoption decisions are affected by the decisions made by other potential adopters.

Therefore, in this paper, we analyze the reasons why HIP has not been widely adopted yet and suggest strategies to foster its adoption in the future. We show how HIP's technical design affects its value as perceived by stakeholders by focusing on the adoption incentives and stakeholder interactions. The analysis is based on an extensive interview study covering 19 networking technology and business professionals, many of them leading experts in the field. After analyzing the HIP case in detail, we also discuss how the findings could provide value to the architects and developers of future protocols.

The remainder of this article is organized as follows: Section 2 presents the theoretical background on innovation diffusion in the Internet whereas Section 3 introduces Host Identity Protocol to the reader and Section 4 illustrates its deployment landscape. Then, Section 5 describes the interview study used as the research methodology. Section 6 analyzes the interview results concerning the reasons why HIP has not been widely deployed yet and the strategies to boost its deployment. Finally, Section 7 discusses about the limitations of the study and the potential wider meaning of the results before Section 8 concludes the paper.

2. Theoretical Background

As Hovav et al. [11] present, the factors affecting innovation diffusion can be divided into two main categories. Diffusion of innovation literature focuses on the attributes of the innovation (for a single adopter), whereas the economic perspective on innovation adoption focuses on the network

externalities' impact on the value of an innovation. When it comes to protocol diffusion, Internet as an environment has some special characteristics that affect the diffusion of Internet standards. The following sections elaborate on these three topics focusing on the Internet standards adoption.

2.1. Attribute-centric Perspective on Innovation Diffusion

Classical diffusion theory elaborates how the attributes of an innovation influence the adoption decision of a potential adopter. Rogers [12] has identified five generic characteristics of innovation that impact on the adoption decision: relative advantage, compatibility, complexity, trialability and observability. Other authors have created their own generic attributes that can be mapped back to Rogers' attributes, including, for example, Technology Acceptance Model (TAM) of Davis [13] and Unified Theory of Acceptance and Use of Technology (UTAUT) of Venkatesh et al. [14].

Relative advantage, which can be understood as economic profitability compared to substitutes and expressed in terms of costs and benefits, is an especially interesting attribute in a highly commercial environment as the Internet currently is. Rogers [12] defines relative advantage as the degree to which an innovation is better than the idea it supersedes. This is a necessary but not sufficient condition, because an innovation also needs to have relative advantage over the competing solutions being developed at the same time. Battle between VHS and Betamax is a legendary example (e.g., [15]) but the protocol world also has its own standard wars (e.g., SIP vs. H323 vs. Skype).

The Internet architecture board (IAB) has identified similar success factors for communication protocols by studying several protocol cases [16]. The pragmatic analysis finds the positive net value (benefits outweighing costs) and incremental deployability (early adopters gaining some benefit even though other Internet would not adopt) as the most critical factors for a protocol's success. The less critical factors relate to the IETF philosophy of openness (open code and specification availability, open maintenance process, and freedom from usage restrictions) and good technical design following proven design principles [17, 18]. IETF has also shown some interest towards deployment questions by establishing a wiki page [19] to analyze the success of the protocols.

Also the level of market uncertainty impacts the optimal design for standards. The distributed modular structure of standards affects positively on the adoption when the market uncertainty is high, becauses it increases the

flexibility by providing alternative options in the introductory phase of a new communications platform. The modular system structure also enables the successful experimentation of suitable new applications in niche segments especially when they are targeted at latent end user needs [20]. Centralized integrated architectures providing more generic applications can, however, be used later, when the market uncertainty related to the end-user needs is low [21, 22].

2.2. Economic Perspective on Innovation Diffusion

Economic perspective on innovation diffusion premises that the utility of an innovation depends on the number of existing or potential adopters [23, 10, 24]. These network externalities relate to a multitude of factors, such as the logarithmically increasing value of a network for an individual adopter [25], economies of scale [26] and improved knowledge from increased use [27].

Due to the significant network externalities associated with communication protocols, many authors have studied the evolution of utility when the number of adopters increases. The typical challenge to overcome is the so-called bootstrapping problem (or chicken-egg problem) where no users adopt the technology because the costs exceed the benefits until a certain level of adoption known as "critical mass" is reached. Joseph et al. [28] and Jin et al. [29] analyze how converters can be used to incentivize adopters to change from incumbent technology (IPv4) to new technology (IPv6). Iannone and Levä [30] conduct similar analysis for moving from BGP to LISP. Additionally, Ozment and Schechter [31] describe six different approaches that can stimulate adoption of security protocols (DNSSEC, SSH, HTTPS and IPsec) that face the bootstrapping problem. These include 1) global and 2) partial mandates, 3) bundling complements, 4) facilitating subnetwork adoption, 5) coordination, and 6) subsidization.

Economics of innovation diffusion have also been studied from the actual decision-making context within a firm. In their analysis about trade-offs between IPv4 and IPv6, Bohlin and Lindmark [32] state that path-dependency and lock-in problems [33] prevent adoption of new technologies, i.e., the experience with older standards may keep the firm trapped and make it difficult to shift to new and potentially better standards. Bottermann [34, 35, 36] has also surveyed the potential adopters of IPv6 in the regional Internet registry community for finding the reasons for non-deployment. In these studies the

biggest hurdles identified were the magnitude of costs to deploy IPv6, negative business case to non-technical decision makers, and poor availability of knowledgeable staff.

2.3. Internet as an Environment for Innovation Diffusion

The Internet constitutes a special environment for innovation diffusion due to its global, distributed and unregulated nature where control over resources is spread among a multitude of stakeholders with diverse economic goals. Therefore, adoption of IETF protocols is a market-based process where economic incentives define the success of a protocol. Consequently, the deployment of evolutionary changes to the Internet is often stymied by the lack of sufficient economic drivers, difficulty of achieving consensus among a plethora of stakeholders with imperfectly (if at all) aligned interests, and the inability of government to foster change [37].

Internet architecture is based on two fundamental principles: hourglass-shaped, layered structure with IP as the narrow waist, collaborated by the end-to-end principle [38]. Even though the layered architecture should allow each layer to evolve independently of each other, the end-to-end principle facilitates innovation only at the application layer while doing nothing to foster the evolution of the underlying functionality associated with the network layer and below [37]. Actually, as Sandvig [39] notes, the end-to-end principle seems to assume that the underlying network already provides all the functionality that will ever be necessary or desirable. Additionally, the recent EvoArch model [40] concerning the evolution of layered protocol stacks in a competitive environment suggests that challenging protocols close to the waist of the hourglass is almost impossible because differentiating with quality does not have strong impact there. As a consequence, new network and transport layer protocols should avoid competition with the incumbent protocols on the same layer.

3. Background on HIP

In this section, we go through the HIP functionality from the view point of deployment. It should be noted that we are referencing the experimental RFCs. They are being transitioned to the standards track at the IETF during year 2012. A more detailed overview of HIP is provided in [41].

3.1. What is HIP and Why is it Useful?

HIP has numerous extensions but its core purposes could be generalized around five use cases. In brief, HIP 1) secures network data flows of applications [42], 2) improves IPv6 interoperability [43, 9], 3) supports NAT traversal [5], and sustains network streaming of media for mobile devices when 4) they roam between different networks or 5) switch between different network interfaces [44]. The same functionality is also available in other protocols (see Section 3.4 for comparison) but HIP provides all of the five functionalities within a single protocol.

HIP supports cryptographic, network-level authentication of end-hosts. Even though the authentication is at a host level and transparent to the application, the authentication tokens can be percolated up to the application layer to establish strong channel bindings [45]. The authentication tokens in HIP are embedded into addresses; applications have to use "virtual" IPv4 or IPv6 addresses [46] in order to use HIP. The details of how to handle the application traffic is decoupled from the main HIP specification. However, IPsec ESP [42] is typically employed even though HIP is modular enough to accommodate other security mechanisms.

HIP facilitates IPv6 interoperability both at the application layer [9] and network layers [44, 43] because it introduces a new layer of indirection to the networking stack. A benefit of using HIP is that an IPv4 application can communicate with an IPv6 application. Independently of the addresses used at the application layer, the actual packets can be delivered over IPv4 or IPv6. This is possible because the HIP layer sits between the transport and network layers, and translates addresses of packets when they pass the HIP layer.

In many cases, wireless access points and broadband modems are configured to function as a NAT. Such devices typically block data flows for peer-to-peer software or for configurations where a server is located behind a NAT device. HIP with its NAT traversal extensions [5] automates and makes it easier to address hosts behind NAT devices.

Lastly, but not least importantly, HIP supports sustaining of, e.g., video, music or VoIP streaming when the network attachment point of a device changes [44]. For instance, the change can occur when a user switches from 3G to WLAN with his handheld device or when a laptop moves from one network to another. The former is usually referred to as multihoming and the latter as mobility. HIP also allows the server side to switch between networks for the purposes of, e.g., load balancing or fault tolerance.

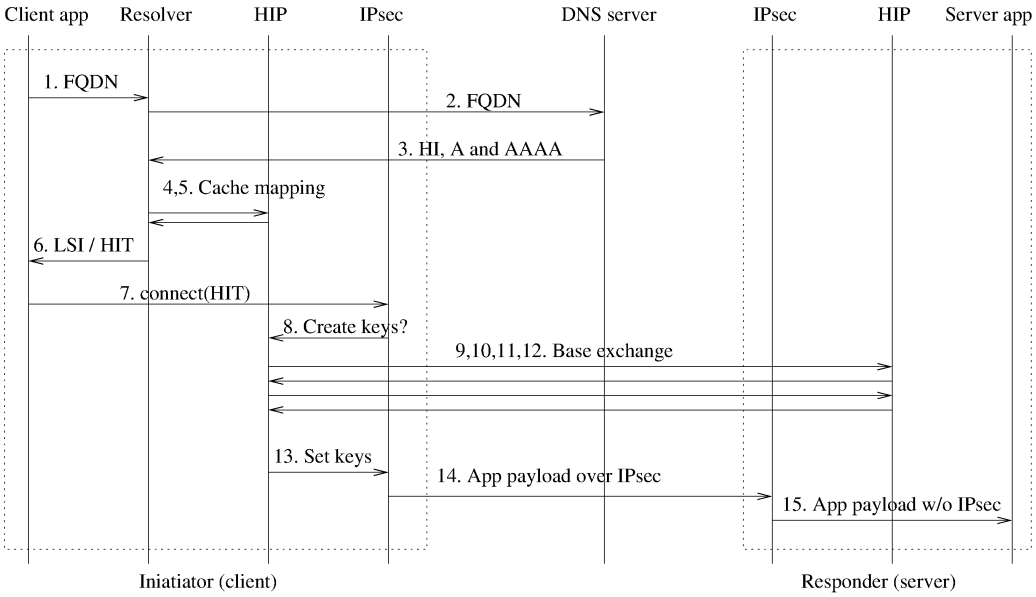

Figure 1: A flow diagram illustrating the process of sending a packet using HIP and IPsec

3.2. How HIP Works

While HIP can be used both with TCP- and UDP-based communications, Figure 1 illustrates how HIP works in practice with TCP. At the client host (or initiator in HIP terminology), a TCP-based application resolves Fully Qualified Domain Name (FQDN) of the server host (responder) in step 1. The resolver is either a DNS library or a local DNS proxy, but assumed to support HIP here, and it contacts the DNS server in step 2. In step 3, the DNS server returns the Host Identifier (HI), and A (IPv4) and AAAA (IPv6) records to the resolver. In the case the resolver did not have a HI record for the server, it would proceed normally and return the routable IPv4 or IPv6 address to the application.

Here, we assume that a HI record was found as the server is HIP capable in Figure 1. Continuing in steps 4 and 5, the resolver sends the records to the local HIP module that caches them to avoid resolving them again from DNS. In step 6, the resolver translates the variable size public key of the HI record into a fixed-size IPv4 or IPv6 address, where the former is referred as Local Scope Identifier (LSI) and the latter as Host Identity Tag (HIT). Then, the resolver delivers the LSI or HIT to the application depending on whether it requested an IPv4 or IPv6 address.

In step 7, the application triggers the TCP handshake using the connect() call to the HIT of the server which is intercepted by the local IPsec module that blocks the application call. As the connection is destined to a HIT, the IPsec module determines that the packet needs to be protected using an IPsec tunnel and requests local key management software to create symmetric keys in step 8. HIP module receives the request and fulfills it by triggering a four-way Diffie-Hellman key exchange, called the base exchange, with the server-side HIP module in steps 9-12. After completion in step 13, the HIP module at both sides configure the symmetric keys for IPsec. At the client, the IPsec module then resumes the blocked application call. In step 14, the TCP SYN is encapsulated over the secure IPsec tunnel and then received by the IPsec module at the server side. In step 15, the IPsec module decapsulates the TCP SYN and delivers to the application. Further application for the TCP connection will again reuse the same IPsec tunnel.

3.3. HIP Development and Deployment History

The first draft on HIP was published at the Internet Engineering Task Force (IETF) in 1999 and the official working group was formed in 2004. The main protocol specifications (RFC5201-5206) were published as experimental standards in 2008 and are presently being transitioned towards the standards track. The NAT traversal extensions [5] were published later in the process (during 2010). Internet Research Task Force (IRTF) had also a separate research group for HIP-related research which was concluded on July 2012.

A number of HIP implementations have emerged but the three well-known and interoperable open-source implementations are OpenHIP from Boeing, HIP for inter.net from Ericsson and HIP for Linux (HIPL) [47] that has been developed collaboratively by Helsinki Institute for Information Technology, Aalto University, and RWTH Aachen University. The implementations cover Windows, OS-X, BSD and Linux operating systems, but they are not included in any operating system distribution by default.

Despite the diversity of OS support, HIP has not been deployed widely in the Internet and consists mostly of small deployments by the implementers. For instance, Boeing uses HIP to secure traffic (and identify machines) with moving robots at their airplane factory [48]. Also, a so called Tofino product exists which uses HIP to facilitate layer-two Virtual Private Networks [49].

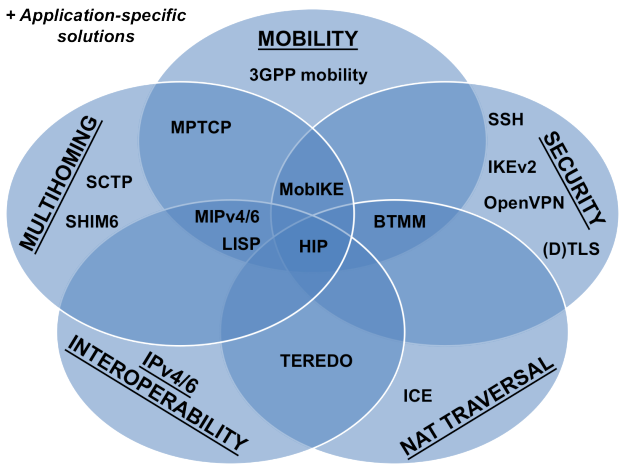

Figure 2: Substitute technologies and their position in relation to the features of HIP

3.4. Comparison of HIP to Substitute Technologies

HIP collects a number of features supported by other protocols under a single architecture as illustrated in Figure 2. Due to space limitations, we focus here on the most important deployed substitutes mentioned by the interviewees. Therefore, some of the latest developments in the IETF, such as SCTP, MPTCP and SHIM6, that may compete with HIP in the future, are not analyzed further.

Mobility (i.e., host movement between different networks) is also supported by Mobile IPv4 (MIPv4) [50] and Mobile IPv6 (MIPv6) [51]. Similarly as HIP, MIP has extensions [52] to support IPv4-IPv6 dual stack mobility. MIP requires extra infrastructure in the form of home agents to relay control and data plane traffic, whereas the extra infrastructure is optional for HIP. MIP is usually combined with IKE to support IPsec-based security which can result in twice as large code size compared to HIP according to some metrics [53, p. 162].

Cellular networks support mobility among the same link layer technology. Compared to cellular networks, HIP is agnostic to the underlying access technology. Besides lower layer solutions, certain network applications, such as web browsers and email clients, can tolerate certain aspects of mobility. For instance, modern web browsers support pausing and resuming of long downloads. Email applications, for one, try to automatically reconnect with the email server when the communications fail.

Virtual Private Networks (VPNs) can be set up using application-layer solutions such as OpenVPN or even with Secure SHell (SSH), or low-layer mechanisms such as Internet Key Exchange version 2 (IKEv2) [54]. Contrary to HIP, IKEv2 uses routable IP addresses instead of virtual ones and the authentication is based on certificates instead of public keys. IKEv2 also has an extension called MOBIKE [55] that supports client-side mobility and multihoming. Unlike (MOB)IKE, basic HIP does not make use of any gateways. Another difference is that MOBIKE does not support server-side mobility contrary to HIP and MobileIP.

The popular SSL/TLS [56] is used for securing TCP connections, typically with HTTP traffic. A separate protocol, DTLS [57], is needed when securing UDP-based communications. Both of the protocols require the application to use specific APIs. Compared to HIP, (D)TLS gives more control of the security to the application. When HIP is being used with unmodified legacy applications, they do not know when the communications is secured. While APIs for HIP have been specified [58, 59], they remain yet unsupported by the OS vendors and HIP implementations.

4. Deployment Landscape of HIP

Understanding the deployment actions is a prerequisite for identifying the involved stakeholders and their adoption incentives. To deploy HIP, both ends of communication need mandatory software modifications and also some changes to the infrastructure are required. We focus here on the modifications required by non-extended, basic HIP but also present how some of the standardized core extensions change the deployment landscape.

4.1. Deployment Actions at the End-hosts

Figure 3 visualizes the mandatory software modifications to the networking stack of an end-host as required by non-extended, basic HIP. It does not mandate any changes to the applications. Extensions to Sockets API for HIP-aware applications [58] require kernel changes but are optional and presently unsupported by the implementations. Transport and network layers do not mandate any changes for HIP.

HIP requires updates to the part of the networking stack that resolves names from DNS in order to support the new DNS records for HIP [60]. This is useful especially for HIP-capable clients so that they distinguish when to use HIP and when to use non-HIP communications with servers. HIPL

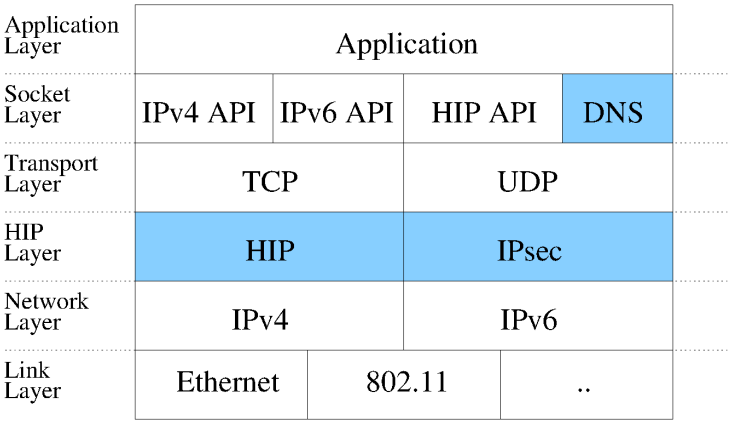

Figure 3: Layering and APIs in a HIP-capable network stack. The software components that require mandatory changes for HIP are highlighted.

and OpenHIP implementations have avoided this deployment hurdle and developed a separate DNS proxy module running locally at the userspace on the client host, intercepting DNS request and translating HIP records on the fly.

All implementations support HIP functionality as a userspace daemon. OpenHIP and HIPL support also the IPsec functionality in userspace to support other operating systems and Linux kernel versions older than 2.6.27. It should be noted that the HIP and IPsec software modules have to be deployed both at the client and server side.

4.2. Deployment Actions at the Infrastructure

HIP requires some modifications to the existing infrastructure as illustrated in Figure 4. The DNS records do not require changes in the actual DNS software itself, at least with one of the most popular DNS server software, *Bind* version 9, as it can support arbitrary types of new DNS records without modification to its software. Depending on the configuration, existing firewalls, VPNs and other mechanisms to control access at the network level may require some extra rules to allow HIP-related traffic to pass through. These changes need to be applied at the server side by the parties deploying HIP, including companies, ISPs, CDN providers, content providers and other organizations. However, HIP does not require any changes in network routers or switches. All HIP implementations support UDP encapsulation of HIP and IPsec traffic, so that modifications to NAT devices are not necessary.

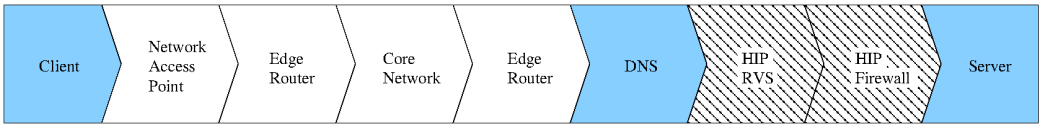

Figure 4: A simplified illustration of a HIP-based communication flow across the network. Devices that need modifications are colored and optional new infrastructure for HIP are filled with tilted lines.

Deploying new infrastructure is not required by the base specification of HIP, so it can be considered mostly optional. For example, HIP rendezvous servers or relays can be deployed when the network environment or the application scenario requires them. HIP-aware firewalls [61] can optionally be deployed to the perimeter network when e.g, a company is protecting their intranet. Alternatively, a server-side proxy could be used as a surrogate for a gateway-based VPN (MOBIKE or OpenVPN) to terminate the HIP tunnel before the servers. This way, access control for HIP would be centrally managed in the proxy instead of distributing the rules at the servers. The new DNS records are not needed in this kind of deployment model because a client is communicating with a single, fixed HIP host, the proxy. Also, a client-side proxy could be used to terminate the HIP tunnel to avoid modifying the clients [62, 63].

5. Methodology

As a research methodology, we use a single case study approach, where we focus on analyzing the HIP case in depth. By using the case study method, it is possible to study a specific phenomena in detail [64]. While the secondary sources mainly describe what has been done, an explanatory case study provides detailed insight into the questions of how and why certain decisions were made and what is the interconnection of different factors affecting related innovation diffusion. A qualitative single case study is especially suitable for studying an emerging technology case, such as HIP, where the goal is to identify relevant factors affecting to the case [65]. HIP is also ideal subject for adoption case study, because people have experiences from it due to its fairly long existence, but the minimal deployment prevents quantitative analysis.

As adoption decisions are often complex and rest on perceptions instead of hard facts, they cannot be easily studied with simulations, performance measurements or spreadsheet calculations. Therefore, the input for this pa-

per was collected in 19 expert interviews conducted between June 16 and September 1, 2011. This number of interviews does not provide statistical significance but is sufficient enough for identifying and prioritizing the most important adoption barriers. A formal, structured interview process with predefined set of questions asked in the same order was used to ensure the reliability of the results [65]. The data were treated comprehensively and appropriate tabulations were used to increase the validity of the results [66].

To assure comprehensiveness, we interviewed different stakeholders, including operating system vendors (2 interviewees), software vendors providing VPN, mobility, security and sensor solutions (4), end-user companies (1), network equipment providers (5), Internet service providers (2), and professors with expertise in (wireless) networking and data security (5). Because HIP has been actively developed mostly at the IETF and the IRTF, most of the interviewees, including five Internet Architecture Board (IAB) members and two area directors of the IETF, were networking experts with extensive experience (224 RFCs in total) on IETF protocols. The interviewees included both HIP developers (7), expected to be enthusiastic about HIP, and other networking experts (12), called "HIP outsiders" in this paper, expected to have a more pessimistic view. In addition to technical expertise, many interviewees were also familiar with business requirements due to their positions as product managers, CTOs or entrepreneurs.

Each interview lasted 45-100 minutes and covered the following main categories: 1) background questions about the interviewee, 2) reasoning why HIP has not been widely deployed yet, and 3) discussion of deployment strategies and use cases for HIP. Most of the time was spent on discussing about reasons for the lack of success in HIP deployment. This was divided into two parts. In the first part, to avoid leading the interviewees, each interviewee was asked to elaborate freely why HIP has not been widely deployed yet. A set of high-level, open-ended questions was used to make sure that all the possibly relevant factor categories identified during the literature review were covered. These included characteristics of HIP, relative advantage compared to competing solutions, stakeholder incentives, the impact of network externalities, and Internet ecosystem specific questions related e.g. to standardization process and layer structure. The answers provided lots of insight but the prioritization would have been difficult based on that data due to the open nature of the questions. Therefore, in the second part, the interviewees were asked to evaluate the importance of 14 pre-identified reason candidates (i.e., our hypotheses, see Table 1) collected from literature and discussions

with HIP developers prior to the interviews. This provided additional data for the prioritization.

6. Results

The results of the interview study are reported and analyzed in this section. First, the interviewees' understanding of and attitudes towards HIP are introduced so that they can be taken into account when interpreting further results. Then, the bulk of this section describes and prioritizes the identified adoption barriers of HIP. After discussing the barriers, we look into the future by presenting strategies and promising use cases suggested by the interviewees that could foster the adoption of HIP.

6.1. Understanding of and Attitudes Towards HIP

Defining HIP or limiting the discussion to some specific use case was deliberately avoided during the interviews to obtain a holistic view and to learn also how interviewees perceive HIP. Thus, in the beginning of each interview, the interviewee was asked to define HIP. As expected, the answers varied significantly between interviewees. Besides mentioning the concept of the id-loc split, some interviewees (6 out of 19) emphasized the security aspects whereas others saw HIP primarily as a mobility protocol (4 out of 19). The security emphasis can partly be explained by the fact that HIP development was started by security experts, whereas mobility management was seen as the most interesting feature and differentiator from other solutions. Additionally, key management for security was raised as one of the main purposes, especially by HIP developers. This may stem from the fact that HIP development started because its inventors were frustrated with the heavyweightedness of IKE and its key management procedures. Interestingly, only HIP developers described NAT traversal or IPv4/v6 interoperability as features of HIP before the interviewer mentioned those. This was partly caused by unawareness and partly because HIP outsiders considered these features less important. Finally, most of the interviewees focused on public, multi-stakeholder deployment scenarios in their reasoning, which is reflected in the results. Therefore, we note that some of the identified adoption barriers may not be valid in private deployment scenarios.

We assumed that HIP developers would view HIP more positively than other interviewees, so we asked interviewees' attitudes towards HIP to better distinguish factual answers from emotional ones. Even though this held quite

true, none of the interviewees strongly objected to HIP. Actually, even the developers of substitute protocols saw HIP interesting as a research project that nonetheless lacks real-world deployability. As an emotionally neutral, skeptic attitude prevailed among interviewees (even with some HIP developers), the answers can be presumed to provide a rather realistic view on the deployment issues of HIP. Still, the attitudes are taken into account when reporting and analyzing the interview results as the answers were loaded here and there.

6.2. Adoption Barriers of HIP

The main purpose of the interviews was to identify and prioritize the adoption barriers, i.e., the reasons why HIP has not been widely deployed yet. As explained in Section 5, this was divided into two parts: 1) identifying non-deployment reasons using open-ended questions, and 2) prioritizing the importance of 14 pre-identified reasons. In this section, the synthesis of the results of both parts are used to summarize the most important reasons for non-deployment.

Table 1 presents the ranked list of the reasons as evaluated by the interviewees. N = 16, as three interviewees did not provide answers for this part of the interviews. The scores present the importance of each reason for non-deployment on the scale 1-5 (low-high). The median value is highlighted with grey background. Also "No answer" (NA) was provided as an option if the interviewee felt that he did not have sufficient knowledge to provide an opinion. This was used sparingly, though. Because the small N does not provide statistical significance and many reasons have high standard deviation partly explained by the heterogeneity in the interviewee's backgrounds and attitudes (see Section 6.1), these results do not provide definitive answers alone. Thus, also the answers to the open questions, and the impact of interviewees' backgrounds are taken into account in the analysis below.

Missing demand proved to be the most important adoption barries (#1, #7). However, the word "demand" was understood differently among interviewees, which explains large standard deviation for the reason #7. HIP developers considered "demand" existing if there are use cases where HIP would be useful independent of the potential adopters' need for HIP particularly, whereas other interviewees understood "demand" existing if potential adopters' have pressing, unsolved problems. The second most relevant reason, partly explaining the missing demand for HIP, was that substitute technologies are favored (#2, #4). As some interviewees explained, this is

Table 1: Importance of 14 pre-identified reasons for non-deployment on scale 1-5 (low-high) with "No answer" (NA) also as an option. Median value is highlighted with gray background.

#	**Reason for non-deployment**	**1**	**2**	**3**	**4**	**5**	**NA**
1	HIP is missing a killer application.	0	0	4	7	5	0
2	Substitute technologies (possibly already been deployed) are favored.	1	1	3	9	2	0
3	Chicken-egg problem (i.e., no one adopts HIP because early adopters do not gain immediate benefit).	2	2	4	4	3	1
4	HIP is a too big change and people favor point solutions solving a single problem.	1	5	3	3	4	0
5	Lack of stable mainstream implementation discourages adoption.	2	3	3	4	3	1
6	Real-world deployability not taken into account from the first day on due to research-mindedness.	4	1	4	6	0	1
7	There is no real demand for HIP.	5	2	2	4	3	0
8	Decision-makers are not aware of HIP or they have misinformation.	5	2	4	1	3	1
9	Company policies or personal frictions in the IETF hinder deployment.	6	2	4	2	2	0
10	Experimental track status discourages adoption.	7	2	4	2	1	0
11	Managing identities is too problematic.	7	2	1	3	0	3
12	HIP is too late.	7	3	0	3	0	3
13	ISPs (or other stakeholders) do not like encrypted traffic.	9	4	1	1	1	0
14	HIP is on the wrong layer of the protocol stack.	10	2	1	1	1	1

actually not a primary reason itself but a consequence from a combination of other reasons not receiving as high scores alone, such as HIP deployment requiring coordination among multiple stakeholders because it does not bring immediate (or standalone) benefit to early adopters (#3), lower net benefit of HIP for potential adopters due to some of its design choices (#4, #11, #13, #14) and being late compared to substitutes (#12). Furthermore, research-mindedness has lead to some strategic mistakes during the development and standardization of HIP (#5, #6, #9, #10). Finally, especially HIP developers suggested that the decision-makers may not have been aware of HIP or they have had misinformation (#8). The comparison of interviewees' opinions to the facts suggests that HIP developers may be right here. Thus, we indicate and correct the common misconceptions when reporting the more detailed results in the following subsections.

When the answers to the open questions and the ranked list of reasons were compared, it became obvious that some of the reasons in Table 1 were expressed ambiguously in the questionnaire form. Most importantly, term "demand" in reason #5 could have been explained better as potential adopters' "business demand" or "real need" for HIP. Reason #14 should have been phrased as "the layer choice complicates deployment" because now many interviewees did not consider the statement from the deployment perspective. Also, reason #12 received low scores partly because some interviewees thought that in some respect HIP was late (e.g., compared to substitutes) but in some respect it was early (e.g., the demand did not exist yet). Due to these limitations, Table 1 should not be taken as the ultimate truth but as an indicative record that needs to be elaborated.

Taking into account both the answers to open questions and Table 1 with its limitations, the reasons for non-deployment can be grouped under six high-level findings listed below, and described in more detail in the following subsections:

#1. Most importantly, demand for the functionalities of HIP has been low.

Where demand has existed, substitutes have been favored because:

#2. Substitutes were earlier in the market,
#3. Substitutes have relative advantage due to some design choices of HIP,
#4. Lack of early adopter benefits requires costly coordination,
#5. People have misconceptions about HIP, and
#6. Research-mindedness of HIP developers has lead to strategic mistakes.

6.2.1. Reason #1: Demand for the Functionalities of HIP Has Been Low

The fundamental problem of HIP seems to be that there has been no business demand for it. HIP was a visionary idea on the time it was first introduced in the late 90s, and the use cases, such as mobile Internet usage, that would make it essential were not realized yet. Even though similar ideas were suggested then also by MIP and have lately become more visible in other protocols too (e.g., in SHIM6), the need is still lower than originally anticipated by HIP developers. Additionally, HIP has been missing a killer application that could not be solved by other solutions.

As HIP solves multiple interrelated problems (such as strong authentication, data security and mobility) simultaneously, only few use cases share the entire problem space. Additionally, some interviewees suggested that the

way HIP solves some of the problems may not be the one demanded by applications. For instance, seamless HIP mobility that preserves TCP connections during breaks of connectivity may not be needed by many applications. For example, a majority (nearly 60%) of network traffic is HTTP according to some sources [67] and based on less than one second transactions according to Callahan et al. [68][1]. Furthermore, the solution HIP provides may not be sufficient enough in some cases because long connectivity breaks terminate the connection due to TCP timeouts [69].

The missing need should not come as a surprise to HIP developers, because their original motivation for HIP development was not to solve an unsolved problem but to build a simpler, better performing and architecturally more elegant solution than IKE and MIP. Actually, as one interviewee said, "everything HIP does can be done with other protocols as well". Therefore, many interviewees saw HIP as a research project where the technology enthusiastic participants have been more interested in architectural beauty than practical deployment[2].

6.2.2. Reason #2: Substitutes Were Earlier on the Market

HIP was introduced considerably later than its core substitutes IKE (First standards track RFC in 1998) [70], MIPv4 (1996) [71], MIPv6 (2004) [70] and TLS (1999) [72]. These substitute solutions had reached the incumbent status, i.e., acceptance of the standardization bodies and the industry, before HIP development even started. Thus, they have stolen the headlines and decreased the general public's knowledge of, interest towards and need for HIP. If HIP would have been available during the time its idea was initially presented (in 1999), there could have been better chances for its deployment. Currently, however, there are many other solutions available that are widely deployed, typically at the application layer. These are technologically mature, even though possibly sub-optimal solutions.

Substitutes have also evolved parallel with HIP development. The path

[1]However, the authors also state that TCP sessions with some services last longer, from quarter of a minute up to four minutes. Consequently, it appears that mobility mechanisms such as HIP are more useful for certain types of services than for others.

[2]Interviewees claimed that this has been visible first in neglecting the NAT traversal issues, later in the prolonged standardization process, and recently in HIP RG discussing new topics with more energy than HIP WG is putting into refining the details of the base specifications.

dependency has lead to a situation where potential adopters tend to stick with the solutions they are familiar with because it is easier to update and modify existing than to adopt something completely new. This has been the case with, for example, VPN providers that decided to add mobility to IKE by specifying MOBIKE instead of changing to HIP. Furthermore, some stakeholders have invested lots of R&D effort into alternative solutions (e.g., Nokia to MIP) and thus may not have been interested in investing into something new, even though it could have long-run benefits.

6.2.3. Reason #3: Substitutes Have Relative Advantage Due to Some Design Choices of HIP

Many interviewees believed that HIP has some advantages over substitute protocols. For instance, they thought that its complexity is smaller than the complexity of some of its substitutes, such as the IKE-MobileIP combination (evidence of this is referenced in Section 3.4). The interviewees also believed that HIP is better than application-specific solutions because its generality would free developers from implementing such functionality redundantly for each application. Still, interviewees argued that HIP has suffered from being a general solution because point solutions, such as IKE, MIP, or application-specific solutions, have been optimized to a single problem. Point solutions are also often fixing a very specific problem as a patch to the current solution, due to which they require less changes and are easier to deploy.

Based on the interviews, one of the most important factors complicating deployment and decreasing the net benefit of HIP is the location of HIP in the protocol stack. Even though the interviewees mostly agreed it is now positioned correctly from the theoretical perspective[3], practical reasons do not support this placement. Firstly, most interviewees claimed (incorrectly though, see Section 6.2.5 for further analysis) that breaking the tight binding of TCP/IP model implies changes to protocol stacks in OS kernels. These have high inertia for change due to long update cycles of OSes and OS vendors' conservativeness on what to include in their operating systems since each new protocol in the stack increases their maintainable codebase, and

[3]However, some interviewees also questioned technical validity of the layer choice. For example, they stated that IP layer may not be the right place to solve mobility and multihoming because the key challenges in mobility are not related to connectivity but rather to transport/application state. Therefore, solving mobility at higher layers may be more effective.

thus costs. As application providers cannot rely on the OS level solutions that are not widely available, the requirement of kernel updates prevents partial deployment for applications that could be interested in using HIP.

Secondly, the interviewees argued that the layer choice leads to problems in the presence of middleboxes due to non-TCP protocol number that may be blocked by, e.g., firewalls or NATs. Even though this problem has been solved by the NAT traversal extension [5] that uses UDP encapsulation, not all the interviewees were aware of this extension. This also implies that the wider audience may not have up-to-date information on the most recent developments of HIP that solve some critical deployment issues (see Section 6.2.5).

Another problem stemming from the layering choice relates to the interaction between HIP and application protocols. To enable interoperability with legacy applications, HIP has been designed to be transparent to applications by masquerading host identifiers as IP addresses. However, this transparency may not always be a virtue. First of all, it may cause problems if applications pass host identifiers to HIP-incapable hosts, e.g., in the application payload (known as the *referral problem* for addresses) [73, 9, 74]. More importantly, applications often want to influence, control or at least be aware of their traffic because they have better knowledge over transport and application state, and user requirements, than network layer protocols. A couple of interviewees stated that to enable communication between the HIP layer and applications, HIP developers should give up on full transparency and specify an API. This, however, has been standardized [58, 59] but remains unsupported by mainstream OSes.

IPsec encryption was also seen as a problematic design choice due to a number of reasons. Firstly, encryption may cost too much overhead in use cases where it is not needed. This was the case especially 10 years ago because the performance of devices was not sufficient. Even though client devices (except small sensors) are currently capable of handling encryption, and support for hardware-accelerated encryption alleviates the issues with processing overhead, the encryption still impairs performance especially at the server side that potentially needs to handle thousands of concurrent encrypted connections. Secondly, a couple of interviewees mentioned that encryption prevents ISPs from shaping the traffic, and complicates lawful interception, which may limit HIP's chances in ISP-centric use cases [75]. Thirdly, many interviewees stated that security is not a feature that sells (i.e., provides value to their customers) and that many people have asked if

HIP could provide mobility without security. Actually, not even IPsec itself has been taken widely into use outside of the VPN use case, which suggests that a protocol securing all the traffic is justified only in those rare use cases where its benefits truly outweigh its costs.

6.2.4. Reason #4: Lack of Early Adopter Benefits Requires Costly Coordination

Network externalities also create a challenge for HIP, especially in multistakeholder use cases in public deployment scenarios. Many interviewees explained that with its current basic design as a host-to-host protocol, both communicating partners have to support HIP before either of them gets any benefits[4]. This creates a bootstrapping problem [31] for the early adopters because the benefits realize only after the critical mass has adopted HIP and stakeholders are not willing to invest unless they are certain that the other stakeholders will also adopt the protocol. Using the terminology of Thaler and Aboba [16], HIP is not an *incrementally deployable* protocol as it is lacking the early adopter benefits due to which the deployment requires coordination among adopters. This is one of the main differences between MIP and HIP because with MIP one can support mobility regardless whether it is supported at the server side or not.

In general, most interviewees agreed that HIP deployment requires too many changes and involves too many stakeholders in public deployment scenarios. Even though the changes would be minor, such as configuring a new DNS record type, they still need to be deployed and often require coordination among multiple stakeholders. Beyond technical changes, HIP aspires to break the existing mental model concerning the tight binding between TCP and IP which may be even more challenging to get accepted by network and application designers than technical changes are.

6.2.5. Reason #5: People Have Misconceptions about HIP

When the responses of interviewees were compared with the facts about HIP, it became obvious that many interviewees had gaps in knowledge or misconceptions about HIP. For the most part, HIP outsiders were only familiar

[4]This problem may be circumvented in private deployments where both ends are controlled by the same party or by using client-side HIP proxies to avoid need for updating hosts on the client-side (or on the server-side in a MIP like deployment scenario).

with the basic ideas of HIP introduced in the early phases of its development, whereas their knowledge of the recent developments that solve many critical deployment bottlenecks was limited. Some interviewees explained this as being a result of the poor real-life deployability in the beginning of HIP development (e.g., NAT traversal issues where not considered), which made people to abandon HIP as unrealistic. Even though HIP developers have improved the deployability, changing prevailing opinions seems to be challenging.

The myth that has perhaps had the most negative impact on HIP adoption is that its deployment would require mandatory changes to the OS kernels, and thus require actions from OS vendors. Even though the conceptual location of the HIP layer between transport and network layers implies this as TCP/IP stack is typically implemented in the kernel, HIP can actually be implemented in the userspace as all of the HIP implementations demonstrate. A possible reason for this misconception is that networking architects strongly represented among interviewees have become estranged from implementing protocols and thus are not fully aware of the relationship between the protocol stack and OS architecture. Similarly, the need for infrastructure changes was overstated by HIP outsiders since many of them considered, e.g., rendezvous servers as a mandatory component.

Application developers have been also unaware of HIP despite all of the community efforts. For example, developers of the Back to my Mac service [76] told that they ended up building it without HIP because they did not know that HIP could be used to realize exactly the same service. One specific reason for their ignorance was that the ORCHID prefix [46] used by HIP was not (and is not even today) listed in IANA's IPv6 address space list.

6.2.6. Reason #6: Research-mindedness of HIP Developers Has Lead to Strategic Mistakes

Despite of some commercialization attempts, HIP has struggled to move from research and development phase to commercialization phase. Even though interviewees praised that HIP standardization has been done by the book (i.e., technical quality of specifications is fine and many interoperating reference implementations exist), they also suggested that the research-

mindedness[5] of HIP developers has lead to some strategic mistakes during HIP development that have had negative impact on its success.

First, HIP developers visualized HIP as something that changes the whole Internet, which, according to many HIP outsiders, was seen as a lack of realism by the people in the IETF. Also, the focus was too much on the id-loc split (instead of the practical benefits of the protocol) which was both a politically sensitive topic and difficult to understand for many. Second, OS vendors, who could have been key stakeholders during commercialization, were not involved in HIP development. Third, HIP developers have pursued perfection, which has lead to sub-optimal design choices from the deployment perspective and prolonged the standardization process so that the potential adopters have solved their problems with other solutions. Fourth, especially HIP developers considered not going directly to standards track as a major mistake, because it delayed standardization for four years and the experimental status of HIP RFCs has been used as an excuse to not deploy HIP[6]. Fifth, some interviewees also claimed that the HIP developers have been too "soft" and research-minded when success in getting HIP to commercial products, or to other protocols in the IETF, would have needed stubbornness, perseverance and sales attitude.

6.3. Strategies to Foster HIP Deployment

After discussing the adoption barriers, the interviewees were asked what could foster the deployment of HIP and which are the most promising use cases for it. As the missing demand was identified as the most important reason for non-deployment, interviewees suggested finding a "killer application"[7] and focusing on it as the "killer strategy". Even a single successful real life deployment in the public Internet, or another high profile private deployment in addition to Boeing [48], could demonstrate the costs and benefits and decrease the uncertainty related to the feasibility of HIP. To increase the success chances of this effort, the deployment should not be academia- but company-driven.

[5]The unusual co-existence of HIP WG and HIP RG supports this statement.

[6]This has had importance mostly inside the IETF when HIP has been suggested to be used as a part of other protocols, and in other standardization forums.

[7]The interviewees defined killer app as a use case where the net benefit of HIP is clearly higher than in other solutions, or other solutions do not exist.

The most promising use cases for HIP were seen to be found from the private deployment scenarios, such as military (e.g., battlefield communication), public safety (e.g., ambulances communicating with hospitals), industrial control systems (similar to the Boeing case [48]) or sensor (e.g., creating security context) fields. The overarching attribute of these use cases is that their deployment can be controlled by a single stakeholder, and, in most cases, the communications remains in the private network.

Interviewees suggested that also external events may trigger demand. For example, the increase of the mobile and multihomed device base may increase interest towards the mobility aspects of HIP, and a possible security meltdown of the Internet could make people more interested in security. Another driver for HIP could be an extremely robust, usable, and well-supported implementation that would work straight out of the box with minimal configuration. This may be too much required for the research funding that has historically funded HIP work, which suggests that HIP cannot be widely deployed before a professionally maintained (commercial) implementation is available. Therefore, inducing OS vendors, especially Microsoft, Apple and Google (Android), to implement and include HIP in their OSes, is still relevant, although a challenging task.

However, as explained earlier, OS vendors are currently lacking incentives for implementing HIP because their customers are not demanding it. Therefore, co-deployment of HIP with an application (e.g., integrated into a web browser) or as application-layer middleware (e.g., as a library[8]) without OS vendor support could be a better strategy to foster adoption. This would remove one deployment bottleneck and make HIP a legitimate option for application developers. A related, supporting action would be to enable better control to applications by improving the APIs between HIP and applications.

Finally, especially HIP developers underlined the importance of the ongoing transformation of HIP RFCs from experimental to standards track. This may increase the credibility of HIP inside the IETF and help in pushing HIP to other standardization forums, such as IEEE and 3GPP, but does not solve stakeholders' incentive problems. The improved RFCs would also document the accumulated knowledge of HIP and make it easier for other protocol developers to understand and learn from it. Additional education may also be needed to correct the prevailing misconceptions, this paper being one effort

[8]The results of this paper lead to a library implementation of HIP [77].

in this path. Hence, even if HIP would not have a future in its current form, the concepts and smaller pieces of HIP could be used as building blocks in other future security and identity solutions.

7. Discussion

The previous section provided a highly detailed reasoning why HIP has not succeeded yet and what could be done to foster its adoption in the future. However, the research method has some limitations that should be considered. The results illustrate interviewees' subjective opinions about the topic as interpreted by the interviewer. The number of interviews, though high for an explanatory interview study, does not provide statistical significance to guarantee that the results encompass each and every aspect. Additionally, the results of a single case study approach are not externally valid, i.e., they cannot be generalized beyond the HIP case [65].

Even though most of the identified adoption barriers and strategies to foster adoption have been reported in earlier studies in the context of other protocols (or innovations in general), the previous section contributed by explaining thoroughly their details and relative importance in the HIP case. In the rest of this section, however, we discuss about the potential wider meaning of the results to protocol development by concentrating on the findings that have not been discussed widely in the existing protocol adoption literature described in Section 2. Their generalization beyond HIP remains a topic for future research, though.

The underlying assumption of this paper is that the goal of the protocols standardized in the IETF, including HIP, is to develop solutions to real-life problems. There, deployment is the measure for success [16]. However, if a protocol is developed to research an interesting idea or write theses, the metrics for success are different, such as the number of published research papers. Without looking further into this direction, HIP could be seen more successful if the goal was mainly on research. This discussion suggests that protocol developers, and the investors spending money on protocol R&D, should consider carefully, what is the goal of the work.

Earlier studies (e.g., [16, 31, 34]) have identified the importance of value network issues during the deployment and adoption of the protocol. Our results propose that they may be important already during the protocol development. Because the research orientation of HIP development was found to be one reason for non-deployment, an interesting topic for future research

would be to study how the stakeholder composition during the protocol development (i.e., research-driven vs. business-driven development) affects the success rate of protocols. Moreover, too narrow stakeholder composition in the HIP development suggests that a failure to involve all the relevant stakeholders already during the protocol development may hamper the success chances of a protocol. This would also imply that the implications of the protocol design choices to the business of relevant stakeholders should be analyzed as early as possible. This perspective has gained acceptance after HIP development started, one proof being that at least the European commission research projects increasingly contain a work package focusing on socio-economic issues.

The trade-off between the generality and deployability of protocols in relation to the protocol stack layering was heavily discussed by the interviewees extending beyond the HIP case. Our results confirm the well known deployment challenges of network and transport layer protocols intented to be included in the networking stacks of operating systems. However, the logical architectural placement of the protocol in the networking stack and its software realization within the operating system is a topic that to our knowledge has not been widely discussed before. The HIP implementations show that with the present microkernel OSes, the protocols that conceptually belong to the network layer may not require kernel modifications but could be implemented in the userspace or as application-layer middleware. Especially application-layer middleware approach, successfully followed by, e.g., TLS, seems like a valid compromise between the generality and deployability as it can provide functionalities to applications as a service but is easier to deploy than changes to the OS kernels, while also giving better control to applications. Despite the implementation in the kernel would be desirable eventually due to, e.g., performance issues, implementing the protocol first closer to applications would enable easier experimentation for the potential adopters and thus help in demonstrating the feasibility of the protocol.

The innovation diffusion studies conducted in the context of consumer product adoption [12, 13, 14] acknowledge the determining role of potential adopters' subjective perception of the product's characteristics for the success of the product. Contrary to this, the earlier protocol adoption studies (e.g., [28, 29, 30, 31]) have taken a more objective view when studying the utility of the protocols. Our findings concerning the negative impact of misconceptions for HIP adoption challenge this perspective and propose that also protocol developers should consider more carefully how their protocols

are perceived by relevant stakeholders. This may be especially important with security protocols, that tend to be difficult to comprehend for others than security experts.

Finally, also the misconceptions among protocol developers seem to hamper protocol deployment. The HIP case confirms the finding of Rogers [12] that pro-innovation bias among enthusiastic developers often leads to a misconception that an innovation should be applied ubiquitously. Quite the contrary, our findings suggest that it is beneficial to focus on the problems and use cases with highest demand and net benefit, still keeping in mind the extendability of the protocol that would allow "wild success" beyond the original scope and scale [16]. To put this in the terms of algorithm designers, depth-first may be more favorable than breadth-first.

8. Conclusion

This paper studied the techno-economic adoption barriers of the Host Identity Protocol (HIP) based on 19 expert interviews conducted during the summer 2011. Low demand for the functionalities of HIP was identified as the most important adoption barrier. The substitute technologies have further limited the need for HIP in the cases where demand would have existed because they were either earlier on the market or are preferred due to their perceived relative advantage (i.e., higher net benefit). In reality, however, the relative advantage of substitutes may not be as significant as people perceive, because misconceptions about HIP prevailing among people, such as that a HIP implementation requires mandatory changes to the OS kernels, seem to unnecessarily hamper people's conception of HIP.

The increasingly mobile Internet usage, the emergence of the Internet of Things and the business criticality of the Internet services are such drivers that may increase the relevance of HIP in the future. Based on the interviews, the best chances for initial deployment seem to exist in private deployments controlled by a single stakeholder, such as military, public safety or sensor use cases. HIP developers should now focus their efforts on these use cases with the highest value and strive for company-driven deployment. To facilitate adoption, implementing HIP either directly in applications or as application-layer middleware could be a preferable strategy instead of OS level implementations in the kernel or userspace.

In general, the poor track record of new protocol deployment suggests that the knowledge in the Internet community about stakeholder incentives

and related adoption barriers is limited, or that the generic design guidelines and success factors do not provide sufficient guidance to protocol developers. This paper contributes to solving both of these challenges by distributing knowledge of, and suggesting extensions to protocol adoption barriers and deployment strategies, and by providing a template that could be followed to study other protocols. We believe that conducting these kinds of techno-economic adoption case studies already during protocol development could be one way to improve the low success rate of communication protocols.

Acknowledgments

The authors would like to thank Thomas Henderson, Jan Melen, Patrik Salmela, Henna Suomi, Mikko Särelä, Sasu Tarkoma and Samu Varjonen, for their comments and suggestions. They also thank the anonymous reviewers for their insightful critics.

This work was supported by TEKES as part of the Future Internet program of TIVIT (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT).

[1] M. Handley, Why the Internet only just works, BT Technology Journal 24 (3) (2006) 119–129, ISSN 1358-3948.

[2] D. Meyer, L. Zhang, K. Fall, Report from the IAB Workshop on Routing and Addressing, RFC 4984 (Informational), URL `http://www.ietf.org/rfc/rfc4984.txt`, 2007.

[3] T. Li, Design Goals for Scalable Internet Routing, RFC 6227 (Informational), URL `http://www.ietf.org/rfc/rfc6227.txt`, 2011.

[4] R. Moskowitz, P. Nikander, P. Jokela, T. Henderson, Host Identity Protocol, RFC 5201 (Experimental), URL `http://www.ietf.org/rfc/rfc5201.txt`, updated by RFC 6253, 2008.

[5] M. Komu, T. Henderson, H. Tschofenig, J. Melen, A. Keranen, Basic Host Identity Protocol (HIP) Extensions for Traversal of Network Address Translators, RFC 5770 (Experimental), URL `http://www.ietf.org/rfc/rfc5770.txt`, 2010.

[6] D. Thaler, A Comparison of IP Mobility-Related Protocols, IETF, expired Internet draft, 2006.

[7] T. R. Henderson, Host Mobility for IP Networks: A Comparison, IEEE Network Magazine 17 (6) (2003) 18–26.

[8] J. Sääskilahti, M. Särelä, Risk analysis of host identity protocol: using risk Identification Method based on Value Chain Dynamics Toolkit, in: Proceedings of the Fourth European Conference on Software Architecture: Companion Volume, ECSA '10, ACM, New York, NY, USA, ISBN 978-1-4503-0179-4, 213–220, 2010.

[9] T. Henderson, A. Gurtov, The Host Identity Protocol (HIP) Experiment Report, RFC 6538 (Informational), URL http://www.ietf.org/rfc/rfc6538.txt, 2012.

[10] M. L. Katz, C. Shapiro, Network Externalities, Competition, and Compatibility, The American Economic Review 75 (3) (1985) pp. 424–440, ISSN 00028282, URL http://www.jstor.org/stable/1814809.

[11] A. Hovav, R. Patnayakuni, D. Schuff, A model of Internet standards adoption: the case of IPv6, Information Systems Journal 14 (3) (2004) 265–294, ISSN 1365-2575, URL http://www3.interscience.wiley.com/journal/118804341/abstract.

[12] E. M. Rogers, Diffusion of innovations, Free Press, New York, NY [u.a.], 5th edn., ISBN 0-7432-2209-1, 978-0-7432-2209-9, 2003.

[13] F. Davis, Perceived usefulness, perceived ease of use, and user acceptance of information technology, MIS quarterly 13 (3) (1989) 319–340, URL http://www.jstor.org/stable/10.2307/249008.

[14] V. Venkatesh, M. Morris, G. Davis, User acceptance of information technology: Toward a unified view, MIS Quarterly 27 (3) (2003) 425–478, URL http://www.vvenkatesh.com/Files/Sciencewatch.pdf.

[15] M. Cusumano, Y. Mylonadis, R. Rosenbloom, Strategic Maneuvering and Mass-Market Dynamics: The Triumph of VHS over Beta, The Business History Review 66 (1) (1992) 51–94.

[16] D. Thaler, B. Aboba, What Makes For a Successful Protocol?, RFC 5218 (Informational), URL http://www.ietf.org/rfc/rfc5218.txt, 2008.

[17] B. Carpenter, Architectural Principles of the Internet, RFC 1958 (Informational), URL http://www.ietf.org/rfc/rfc1958.txt, updated by RFC 3439, 1996.

[18] R. Bush, D. Meyer, Some Internet Architectural Guidelines and Philosophy, RFC 3439 (Informational), URL http://www.ietf.org/rfc/rfc3439.txt, 2002.

[19] IETF, IETF OUTCOMES - Successes and Failures, URL http://trac.tools.ietf.org/misc/outcomes/, 2012.

[20] S. Luukkainen, C. Englund, Value creation for multimedia services on broadband networks, in: J. Harju, T. Karttunen, O. Martikainen (Eds.), SMARTNET, Chapman & Hall, ISBN 0-412-71730-1, 225–235, 1994.

[21] M. Gaynor, Network service investment guide: maximizing ROI in uncertain times, Wiley, ISBN 0471214752, URL http://www.lavoisier.fr/livre/notice.asp?depuis=e.lavoisier.fr&id=9780471214755, 2003.

[22] M. Gaynor, S. Bradner, The real options approach to standardization, in: System Sciences, 2001. Proceedings of the 34th Annual Hawaii International Conference on, IEEE, ISBN 0769509819, 1–10, URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=926526, 2001.

[23] J. Farrell, G. Saloner, Standardization, compatibility, and innovation, The RAND Journal of Economics 16 (1) (1985) 70–83, ISSN 0741-6261, URL http://www.jstor.org/stable/2555589.

[24] M. L. Katz, C. Shapiro, Technology Adoption in the Presence of Network Externalities, The Journal of Political Economy 94 (4) (1986) 822 – 841, URL http://www.jstor.org/stable/1833204.

[25] B. Briscoe, A. Odlyzko, B. Tilly, Metcalfe's law is wrong - communications networks increase in value as they add members-but by how much?, IEEE Spectrum 43 (7) (2006) 34–39.

[26] K. Arrow, The economic implications of learning by doing, The review of economic studies 29 (3) (1962) 155–173, URL http://www.jstor.org/stable/10.2307/2295952.

[27] N. Rosenberg, Inside the Black Box: Technology and Economics, Cambridge University Press, 1982.

[28] D. Joseph, N. Shetty, J. Chuang, I. Stoica, Modeling the adoption of new network architectures, Proceedings of the 2007 ACM CoNEXT conference on - CoNEXT '07 (2007) 1URL http://portal.acm.org/citation.cfm?doid=1364654.1364661.

[29] Y. Jin, S. Sen, R. Guérin, K. Hosanagar, Z.-L. Zhang, Dynamics of competition between incumbent and emerging network technologies, in: Proceedings of the 3rd international workshop on Economics of networked systems, ACM, Seattle, WA, USA, ISBN 978-1-60558-179-8, 49–54, URL http://portal.acm.org/citation.cfm?id=1403039, 2008.

[30] L. Iannone, T. Levä, Modeling the economics of Loc/ID Split for the Future Internet, in: Towards the Future Internet, IOS Press BV, 11–20, 2010.

[31] A. Ozment, S. Schechter, Bootstrapping the adoption of internet security protocols, in: Proc. Fifth Workshop on the Economics of Information Security, 2003, 1–19, URL http://www.icir.org/vern/cs294-28/papers/bootstrapping-security-protocols.pdf, 2006.

[32] E. Bohlin, S. Lindmark, Incentives to invest in next generation networks, in: 14th European Regional ITS Conference, 48, Helsinki, 1–19, 2003.

[33] W. Arthur, Competing technologies, increasing returns, and lock-in by historical events, The Economic Journal 99 (394) (1989) 116–131, URL http://www.jstor.org/stable/2234208.

[34] M. Bottermann, IPv6 Deployment Survey 2009, URL http://ripe59.ripe.net/presentations/botterman-v6-survey.pdf, 2009.

[35] M. Bottermann, IPv6 Deployment Survey 2010, URL http://www.nro.net/wp-content/uploads/2010/11/GlobalIPv6SurveySummaryv2.pdf, 2010.

[36] M. Bottermann, IPv6 Deployment Survey 2011, URL http://www.nro.net/wp-content/uploads/ipv6_deployment_survey.pdf, 2011.

[37] J. Marcus, Evolving core capabilities of the internet, Journal on Telecommunications and High Technology Law 3 (2004) 121–161, URL http://papers.ssrn.com/sol3/papers.cfm?abstract_id=921903.

[38] J. H. Saltzer, D. Reed, D. D. Clark, End-to-End Arguments in System Design, ACM Transactions on Computer Systems 2 (4) (1984) 277–288.

[39] C. Sandvig, Communication Infrastructure and Innovation: The Internet as End-to-End Network that Isn't, 2002.

[40] S. Akhshabi, C. Dovrolis, The evolution of layered protocol stacks leads to an hourglass-shaped architecture, in: S. Keshav, J. Liebeherr, J. W. Byers, J. C. Mogul (Eds.), SIGCOMM, ACM, ISBN 978-1-4503-0797-0, 206–217, 2011.

[41] P. Nikander, A. Gurtov, T. R. Henderson, Host Identity Protocol (HIP): Connectivity, Mobility, Multi-Homing, Security, and Privacy over IPv4 and IPv6 Networks, IEEE Communications Surveys and Tutorials 12 (2) (2010) 186–204.

[42] P. Jokela, R. Moskowitz, P. Nikander, Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP), RFC 5202 (Experimental), URL http://www.ietf.org/rfc/rfc5202.txt, 2008.

[43] S. Varjonen, M. Komu, A. Gurtov, Secure and efficient IPv4/IPv6 handovers using host-based identifier-locator Split, in: Proceedings of the 17th international conference on Software, Telecommunications and Computer Networks, SoftCOM'09, IEEE Press, Piscataway, NJ, USA, ISBN 978-1-4244-4973-6, 111–115, URL http://dl.acm.org/citation.cfm?id=1719770.1719793, 2009.

[44] P. Nikander, T. Henderson, C. Vogt, J. Arkko, End-Host Mobility and Multihoming with the Host Identity Protocol, RFC 5206 (Experimental), URL http://www.ietf.org/rfc/rfc5206.txt, 2008.

[45] N. Williams, On the Use of Channel Bindings to Secure Channels, RFC 5056 (Proposed Standard), URL http://www.ietf.org/rfc/rfc5056.txt, 2007.

[46] P. Nikander, J. Laganier, F. Dupont, An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers (ORCHID), RFC 4843 (Experimental), URL http://www.ietf.org/rfc/rfc4843.txt, 2007.

[47] A. Pathak, M. Komu, A. Gurtov, Host Identity Protocol for Linux, URL http://www.linuxjournal.com/article/9129, 2009.

[48] R. H. Paine, Beyond HIP: The End to Hacking As We Know It, BookSurge Publishing, ISBN 1439256047, 9781439256046, 2009.

[49] T. Henderson, S. Venema, D. Mattes, HIP-based Virtual Private LAN Service (HIPLS), an expired Internet draft, 2011.

[50] C. Perkins, IP Mobility Support for IPv4, Revised, RFC 5944 (Proposed Standard), URL http://www.ietf.org/rfc/rfc5944.txt, 2010.

[51] C. Perkins, D. Johnson, J. Arkko, Mobility Support in IPv6, RFC 6275 (Proposed Standard), URL http://www.ietf.org/rfc/rfc6275.txt, 2011.

[52] H. Soliman, Mobile IPv6 Support for Dual Stack Hosts and Routers, RFC 5555 (Proposed Standard), URL http://www.ietf.org/rfc/rfc5555.txt, 2009.

[53] J. Ylitalo, Secure Mobility at Multiple Granularity Levels over Heterogeneous Datacom Networks, iSBN 978-951-22-9530-2, 2008.

[54] C. Kaufman, Internet Key Exchange (IKEv2) Protocol, RFC 4306 (Proposed Standard), URL http://www.ietf.org/rfc/rfc4306.txt, obsoleted by RFC 5996, updated by RFC 5282, 2005.

[55] P. Eronen, IKEv2 Mobility and Multihoming Protocol (MOBIKE), RFC 4555 (Proposed Standard), URL http://www.ietf.org/rfc/rfc4555.txt, 2006.

[56] T. Dierks, E. Rescorla, The Transport Layer Security (TLS) Protocol Version 1.2, RFC 5246 (Proposed Standard), URL http://www.ietf.org/rfc/rfc5246.txt, updated by RFCs 5746, 5878, 6176, 2008.

[57] E. Rescorla, N. Modadugu, Datagram Transport Layer Security, RFC 4347 (Proposed Standard), URL http://www.ietf.org/rfc/rfc4347.txt, obsoleted by RFC 6347, updated by RFC 5746, 2006.

[58] M. Komu, M. Bagnulo, K. Slavov, S. Sugimoto, Sockets Application Program Interface (API) for Multihoming Shim, RFC 6316 (Informational), URL http://www.ietf.org/rfc/rfc6316.txt, 2011.

[59] M. Komu, T. Henderson, Basic Socket Interface Extensions for the Host Identity Protocol (HIP), RFC 6317 (Experimental), URL http://www.ietf.org/rfc/rfc6317.txt, 2011.

[60] P. Nikander, J. Laganier, Host Identity Protocol (HIP) Domain Name System (DNS) Extensions, RFC 5205 (Experimental), URL http://www.ietf.org/rfc/rfc5205.txt, 2008.

[61] J. Lindqvist, E. Vehmersalo, J. Manner, M. Komu, Enterprise Network Packet Filtering for Mobile Cryptographic Identities, in: Journal of Handheld Computing Research (IJHCR), IGI Global (IGI, 701 E. Chocolate Avenue, Hershey, PA 17033, USA, 79 – 94, 2009.

[62] P. Salmela, J. Melén, Host identity protocol proxy, in: ICETE, 222–230, 2005.

[63] D. Zhang, X. Xu, J. Yao, Z. Cao, Investigation in HIP Proxies, work in progress, Internet draft, 2011.

[64] C. Robson, Real World Research - A Resource for Social Scientists and Practitioner-Researchers, Blackwell Publishing, Malden, second edn., 2002.

[65] R. Yin, Case Study Research: design and methods, Sage Publications Inc, 2003.

[66] D. Silverman, Doing Qualitative Research: Second Edition, Sage Publications Ltd, 2nd edn., 2004.

[67] G. Maier, A. Feldmann, V. Paxson, M. Allman, On dominant characteristics of residential broadband internet traffic, in: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference, IMC '09, ACM, New York, NY, USA, ISBN 978-1-60558-771-4, 90–102, URL http://doi.acm.org/10.1145/1644893.1644904, 2009.

[68] T. Callahan, M. Allman, V. Paxson, A longitudinal view of HTTP traffic, in: Proceedings of the 11th international conference on Passive and active measurement, PAM'10, Springer-Verlag, Berlin, Heidelberg, ISBN 3-642-12333-3, 978-3-642-12333-7, 222–231, URL http://dl.acm.org/citation.cfm?id=1889324.1889347, 2010.

[69] S. Schütz, L. Eggert, S. Schmid, M. Brunner, Protocol enhancements for intermittently connected hosts, ACM Computer Communications Review 35 (2005) 5–18.

[70] D. Harkins, D. Carrel, The Internet Key Exchange (IKE), RFC 2409 (Proposed Standard), URL http://www.ietf.org/rfc/rfc2409.txt, obsoleted by RFC 4306, updated by RFC 4109, 1998.

[71] C. Perkins, IP Mobility Support, RFC 2002 (Proposed Standard), URL http://www.ietf.org/rfc/rfc2002.txt, obsoleted by RFC 3220, updated by RFC 2290, 1996.

[72] T. Dierks, C. Allen, The TLS Protocol Version 1.0, RFC 2246 (Proposed Standard), URL http://www.ietf.org/rfc/rfc2246.txt, obsoleted by RFC 4346, updated by RFCs 3546, 5746, 6176, 1999.

[73] T. Henderson, P. Nikander, M. Komu, Using the Host Identity Protocol with Legacy Applications, RFC 5338 (Experimental), URL http://www.ietf.org/rfc/rfc5338.txt, 2008.

[74] M. Komu, S. Tarkoma, J. Kangasharju, A. Gurtov, Applying a cryptographic namespace to applications, in: DIN '05: Proceedings of the 1st ACM workshop on Dynamic interconnection of networks, ACM, New York, NY, USA, ISBN 1-59593-144-9, 23–27, 2005.

[75] T. Dietz, M. Brunner, N. Papadoglou, V. Raptis, K. Kypris, Issues of HIP in an Operators Networks, IETF, expired Internet draft, 2006.

[76] S. Cheshire, Z. Zhu, R. Wakikawa, L. Zhang, Understanding Apple's Back to My Mac (BTMM) Service, RFC 6281 (Informational), URL http://www.ietf.org/rfc/rfc6281.txt, 2011.

[77] X. Gu, Host Identity Protocol Version 2.5, Master's thesis, Aalto University, 2012.