The Host Identity Payload protocol: toward a secure solution to mobility and multi-homing

Francis.Dupont@enst-bretagne.fr

22 may 2002

Abstract

This paper presents the Host Identity Payload (HIP) protocol, which is a two-space network protocol where identities have cryptographical properties.

Addresses in the Internet Protocol serve two purposes: locator, i.e., "where" is the recipient, and identity, i.e., "who" is the recipient. The idea of two-space systems is to separate the two functions. Their immediate benefit is to make mobility and multi-homing far easier because a node can keep its identity and, at the same time, it can move to a different locator. Many innovative proposals for mobility (LIN6) or for multi-homing (MHTP) are two-space systems but the most famous one is the 8+8 (a.k.a. GSE) introduced by Mike O'Dell five years ago.

However from the security point of view two-space systems are weak: separating the locator function from the identity function obviously introduces vulnerabilities and authentication issues. The design of the HIP protocol includes security considerations from the beginning and does not suffer of this problem: an identity is a public key.

HIP can be considered too as a lightweight IPsec security association establishment protocol, i.e., a replacement of IKE, which is known for its outrageous complexity.

Introduction: address = locator + identity

Addresses in the Internet Protocol serve two purposes:

- **locator**: the location information of an interface. Routers use the destination address of a packet in deciding where to forward it to get it closer to its ultimate destination. Addresses specify "where" the intended recipient is located.
- **identity**: the unique identification of an interface. A sending node tells the identity of the intended recipient by the destination address. Addresses specify "who" is the intended recipient.

The idea of two-space systems is to separate the two functions in order to get easy solutions to mobility and multi-homing problems.

This paper will first describe the network protocol context with IPv6 and IPsec, then the two basic problems: mobility and multi-homing. After a short look at SCTP, a higher layer solution, a two-space solution from each previous point will be analyzed: LIN6 from mobility, MHTP from multi-homing, 8+8 from IPv6 and HIP from IPsec, finishing by further developments and works about HIP.

The purpose is not to detail HIP or IPsec, but to explain the design principles of HIP as a mobility and multi-homing solution (i.e., two-space system).

1 IPv6

IPv6 is the new version of the Internet Protocol, not really a new protocol. IPv6 shares many problems with IPv4 about mobility and multi-homing, as this paper will show because its addressing and routing architecture is basically the same (we do not know a better one). For instance, addresses are per interface (not per node) and routing is done following the longest matching prefix rule.

IPv6 was introduced in order to solve some problems:

- address space exhaustion
- routing table size explosion
- many badly integrated new features like autoconfiguration, multicast, mobility, security, ...

IPv6 should be the long-term solution, the short term is CIDR [1], i.e., an address allocation scheme for better address space conservation and aggregation.

IPv6 provides a large address space (over 128 bits in place of 32 bits), many nice features like a simple stateless autoconfiguration, new policies like multicast integration, mandatory IPsec support for compliant implementations, etc, but not an innovative support for mobility or a magical solution to keep routing tables small.

2 IPsec

IPsec is the security protocol in the network layer (the first global layer hence the best place to add security in). Its most used transform, ESP [2], provides confidentiality, authentication, integrity and anti-replay protection, both for payloads (the transport mode) or for whole packets by encapsulation (the tunnel mode which is used in Virtual Private Networks).

IPsec is based on a notion of Security Associations which are managed by the IKE [3] protocol. IKE is very complex to understand, to implement and to use so the ipsec working group at the IETF is developing a successor (a.k.a. the Son-Of-Ike) which should be simpler, more efficient (less messages), more resistant to Denial-of-Service (DoS) attacks, etc.

Technically, IKE is a Diffie-Hellman exchange to build a shared secret with strong authentication of the parties using commonly a public key mechanism (directly or with X.509 certificates).

DoS attacks are a major threat against security systems based on cryptography (one can argue that other systems can not be really secure). Without protection, a bad guy can easily make the system consume all its resources (for instance, CPU cycles, memory or batteries for wireless devices). Typical examples of a counter-measure are cookies: for instance, against a TCP SYN flood [4] attack, one can replace state by an encoded pseudo-random sequence number which will permit to recover the whole space when (if!) the third packet of the TCP three-way handshake will be received [5,6].

3 Mobility

To begin with, there are two very different things which are named mobility:

- nomadism: a typical case is the way laptops are used today (until they become really shockproof): during movements which can last hours laptops are stopped or suspended. Applications, not connections, can stay alive and be resumed.
- true mobility: connections should stay alive, pauses from hand-offs should be short and the mobile-to-mobile case (i.e., simultaneous movements of the two parties) should be handled.

3.1 Nomadism

Nomadism is an easy problem: in fact for a good solution one needs only a good autoconfiguration (i.e., IPv6 or, in an IPv4 context, DHCP) and dynamic DNS update. At the Minneapolis IETF meeting at the spring of 2002, the "terminal room" (in fact the radio LANs) supported this kind of `DHCP + DNS` configuration so the question whether it was enough for mobility was raised in the IETF mailing-list (the answer is: this is only nomadism).

This can be considered as a two-space solution, the identity role is taken by the DNS names, the locator role by the dynamic IP addresses, and the identity to locator mapping is provided by the DNS itself. Of course, this implies some security requirements too:

- dynamic DNS updates must be secure by the way of symmetric key signature (TSIG [7]), asymmetric key signature (SIG(0) [8]) or another mechanism.

- data themselves (not only transactions, as in the previous item) should be secure but this implies the deployment of full DNSsec [9].

This event was an experiment of DNSsec / secure DNS updating with well-equiped and skilled users and its announce gave a pointer to an how-to [10]. Near all users used either SSH or IPsec or both to protect their communications to private (i.e., office or home) locations.

Note that if DNS is enough for nomadism, this is not the case for true mobility because DNS data have long times to live (reducing these times defeats cache mechanisms and severely infringes performances), DNS is currently not secure and in no case can support continuous connections.

3.2 Mobility

There are two ways to manage the true mobility:

- micro-mobility: the mobile node keeps its address across movements and a host route points to its current location. This can be considered as changing the interpretation of a locator. This works well, is very fast and easy (GSM uses this system, the host route table is the HLR) but unfortunately can not scale.

- macro-mobility: the mobile node has two addresses:

  - the home address which never changes. From the outside the node is known by its home address.
  - a transient care-of address which is allocated by the visited network.

  Mobile IP is a macro-mobility solution with a global scale and support of movements between interfaces with different technologies.

Mobility is a remote redirection mechanism so its signaling must be protected, but Mobile IP [11,12] has extra security problems due to its multiple addresses and their different statuses.

3.3 Mobile IPv6 Security

The three entities in Mobile IPv6 are:

- the Mobile Node (MN)

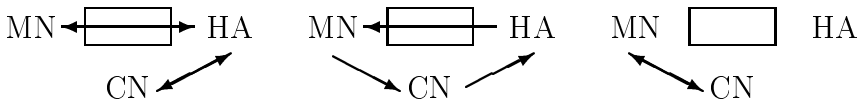
- a Correspondent Node (CN)

- the Home Agent (HA) which is a router on the home link

Mobile IPv6 can work in three different modes, from the most secure and least efficient to the most optimized and problematic:

- figure 1: a bidirectional tunnel encapsulates packets between the MN and the HA (inner MN address is the home address, outer is the care-of address). This is totally transparent to the CN.
- figure 2 (triangular routing): the MN sends packets directly to the CN with a special header containing a home address option: the source address of the packet is a care-of address (if it is not, ingress filtering [14], i.e., checking whether the source address is topologically plausible, should drop the packet) but is replaced at the CN by the inbound processing by the home address. The CN sends packets to the destination of the MN home address and, as in the previous case, they are intercepted and encapsulated to the MN (at its care-of address) by the HA.
- figure 3 (routing optimization): the CN has the knowledge of the binding and adds to packets to the MN a special routing header, having as the first destination in the IPv6 header the care-of address and as the final destination in the routing header the home address. The routing header takes the same role for the MN than the home address option, in fact both can be considered as special tunnels where a redundant address is removed.

Figure 1: Figure 2: Figure 3:



The main threat against security is to spoof the signaling message which manages the home address / care-of address binding: the binding update. With a spoofed binding, traffic to the mobile node is not sent to the proper location:

- the attacker can get all the packets sent to the mobile node
- the mobile node no more receives the traffic sent to it
- hijacked packets can flood a third party.

The security requirement for binding updates is that they must be authorized:

- the obvious solution is to use strong authentication with the proper authorization, but this relies on the availability of some kind of security infrastructure like a PKI, DNSsec, AAA, ...

- the current proposal (because no security infrastructure is available, some even argue a global PKI is not desirable) is the return routability check using multiple paths between the mobile node and its correspondent node. The idea is that a bad guy must be in a special place in the network to be able to perform an attack.

- the last idea is to use Cryptographically Generated Addresses (or Key-Based Addresses) which provide a proof of ownership. One of the more mature protocols of this class, SUCV addresses [13], is directly derived from HIP. But these systems assume a heavy cryptographic operation (signature checking) per verification and are encumbered by intellectual property rights.

From the two-space point of view, two remarks are interesting:

- the return routability check (using the direct MN-CN path and the MN-HA-CN path with an ESP protection on the MN-HA segment) is a locator check, i.e., it does not prove something about "who" is the MN, but only "where" it is. As the HA has higher requirements, one can try to use it but there is no reason to trust more the HA than the MN in the general case.

  The return routability check is a generalization of a well-known liveliness check in security protocols: one party puts a random token in a message to the other party and looks for it in the next messages. This proves the other party is reachable and can answer. If the token is a cookie (i.e., replaces the state) this severely limits some DoS attacks. But this is still vulnerable to Man-In-The-Middle (MITM) attacks: the added value of return routability is that the MITM should be in both paths, i.e., near the CN.

- in the triangular routing case the relationship between the two identities (from the home and the care-of addresses) is very weak and makes some new attacks possible, for instance, a distributed DoS where many attackers send traffic to many nodes supporting this mode with a home address option pointing to the victim to flood. CNs will answer (this is a reflection attack) to the victim without any trace of addresses of attackers.

  This problem makes the triangular routing unusable in the general case, so for the sake of performance Mobile IPv6 relies on a hypothetic heavy usage of the routing optimization...

4 Multi-Homing

A multi-homed site is a site that has more than one connection to the Internet, with those connections going through either the same or different Internet Service Providers (ISPs). Sites choose to multi-home for several reasons, especially to improve fault tolerance, perform load balancing, etc.

The central problem of multi-homing dealt with here is what happens in case of a failure of a path through an ISP in a short time scale:

- to a new connection from the outside to a node of the site
- to a new connection from a node of the site to the outside
- to an existing connection between a node of the site and the outside

There are three kind of solutions:

- solutions based on addressing
- solutions based on inter-domain routing
- solutions based on mobility-like mechanisms

4.1 Multi-Homing and Addressing

In the IPv4 space since CIDR, and in the IPv6 space, the address allocation system is hierarchical and sites are supposed to get their address spaces, a.k.a. prefixes, from their upstream ISPs. The benefit is an ISP can (so it should!) announce only one aggregated route for its whole prefix in place of one route per connected site, or, even worse, one route per connected site prefix.

The easy solution for a multi-homed site is to infringe this principle by using one prefix (from an ISP or for itself as a "provider independent" prefix) and to get it announced by some or all of its upstream ISPs and propagated to all default-free zone routers.

When at least a path through one of its ISPs works, the site has connectivity from and to the Internet. If the inter-domain routing system is fast enough, even existing connections can survive to a failure of one of its ISPs. So more and more sites are trying to use this kind of solution but, of course, from the ISP point of view things are very different.

The drawback of this solution is the route for each multi-homed site should be propagated to all routers of the default-free zone (i.e., routers not using a default route and receiving the global routing table from more than one of its peers). These routes contribute to the rapidly-increasing [1]

[1] the growth was exponential before CIDR, then linear but has taken again an exponential rate because of multi-homing

size of the global IPv4 routing table and places a significant stress on the inter-provider routing system.

Today IPv6 is not deployed enough to encounter the same problem but already ISPs require from the Regional Internet Registries to give up any idea to provide IPv6 provider independent space. But as political pressure shall not be enough, the multi6 working group was created at the IETF and chartered to find a better solution to the multi-homing problem.

4.2 Multi-Homing and Inter-Domain Routing

So today a site may get many prefixes: we have three here on our network and at the IETF IPv6 working group interim meeting in Tokyo we got five. Nodes should have a smart address selection mechanism for both destination addresses (returned by the DNS) and source addresses. It seems a bit too much is assumed about the smartness of common hosts but there are not (yet) blocking problems.

The inter-domain routing is supposed to solve the first question: new connections from the outside. The remote node should try all possible addresses of the node of the site and at least one should work. This works if the failure is correctly detected and the information (a route removal) correctly and quickly propagated (this has an impact on performance). Of course, the routing system is supposed to have no bugs (no permanent incorrect information), to be reasonably stable and secure (this last point is the subject of new developments at the IETF because today the inter-domain routing system is very far from secure).

In order to solve the second question (new connections to the outside), every node should be informed when a source prefix should be used no more (because its return path is no longer available). We proposed many years ago to propagate the information to every internal router, for instance using the router renumbering protocol [15], and from routers to every host by "instantaneous deprecated" prefix informations in router advertisements. The effect is to make addresses of the prefix last-rank candidates (exactly what is needed) and can trigger a next section mechanism.

There is a partially applicable answer to the last question (existing connections) [16] (described with two ISPs A and B):

- the link between the site border router R^A_{site} and the ISP border router R^A_{ISP} is the primary link to the ISP A
- a tunnel from R^B_{site} via R^B_{ISP} to R^A_{ISP} (i.e., following first the other primary link then going between ISPs) is the secondary link to the ISP A
- the site announces its prefix A to the ISP A (i.e., to R^A_{ISP}) via both the primary and the secondary links with a preference for the primary link (to favour paths through it)

- same for the ISP B

The main drawbacks of this configuration are:

- it can become hairy [2] if an ISP performs ingress filtering, i.e., rejects packets with source addresses not in its prefix
- it works only for link failures between the site and the ISP routers (including the site router). For instance this solution is not applicable when the ISP puts its border router in the site
- in general ISPs highly dislike this kind of tunneling. If the two ISPs are in competition, they likely discourage this (for instance they can enable ingress filtering)

So this solution is nice but partially applicable as announced.

4.3 Multi-Homing and Mobility

We claim that most mobility mechanisms can be used for or adapted to multi-homing and, in the other way, most multi-homing mechanisms can be used for or adapted to mobility.

To solve the last (remaining) question about connection survival, Mobile IPv6 is usable (because of its capacity to support two addresses for the same node):

- the current source address takes the home address role
- a working alternate source address takes the care-of address role
- a movement is simulated in case of main path failure (packets are sent with a home address option, a binding update is sent to the other party as for routing optimization)

This does not use home agents: the result is the equivalent of simultaneous movement, the simultaneous failure, is handled only if it can be detected:

- for Mobile IPv6, when binding updates fail to reach the peer they are sent through the home agent (bindings are flushed on ICMP errors or no answer timeouts)
- multi-homing has no fallback like home agents but all peer addresses are known (so they are predictable)

As this paper will show, this is even easier for two-space systems.

[2] It needs to use source routing, i.e., routing in function of the source

5 SCTP

Network layer solutions are intrinsically better but there is a transport layer solution: the Stream Control Transmission Protocol (SCTP [6]).

STCP was developed as the transport protocol for SS7 signaling over IP. Its main feature is the support of many simultaneous streams with optional order-of-arrival delivery between two multi-homed endpoints.

In the initiation phase (protected by cookies), each endpoint can specify the set of addresses (IPv4, IPv6 addresses or DNS resolvable names) it will use. Using a "heartbeat" mechanism, the reachability of a destination address can be probed.

SCTP is designed for fault tolerance and has no real mobility support, for instance the address set must be specified at the association establishment only. It lacks a readdressing capability (i.e., address set management out of the establishment phase) and a rendezvous mechanism (for simultaneous address changes).

But the main limitation of SCTP is an application must be modified in order to use it and take advantage of its innovative features.

6 LIN6

LIN6 [17] is the adaptation to IPv6 of the Location Independent Network Architecture (LINA). It is the heir of a long line of mobility proposals like [18]. There are similar ideas like LAR [19] but this paper uses LIN6 as a typical example of this class of two-space systems for mobility.

In LIN6 a node has a 64-bit identifier named the LIN6 ID which is used:

- to build a 128-bit generalized ID by concatenating of a fixed 64-bit prefix (the LIN6 prefix) to the LIN6 ID. The LIN6 generalized ID takes the role of IPv6 addresses for transport / upper layers but never appears on the wire
- to build per interface 128-bit LIN6 addresses from subnetwork (link in IPv6 terminology) prefixes and the LIN6 ID. They are used for IPv6 headers

The locators are the LIN6 addresses, the identities are the LIN6 IDs and the LIN6 generalized IDs are a way to easily reuse IPv6 code, and a way to provide coexistence with the standard IPv6 when the LIN6 generalized ID prefix is reserved and LIN6 IDs distinguishable from standard interface IDs.

Of course for IPsec LIN6 (generalized) IDs are used, so mobility and multi-homing features are transparent to IPsec and all transport / upper layers.

The resolution of IDs to addresses are provided by Mapping Agents (MA), MA discovery is done by a new DNS resource record pointing to a

MA associated to names of the nodes it serves (like the MX resource record). The resolution service can be insured by many MAs which do not need to stay on a particular link as for Mobile IPv6 (but details are not available). The resolution protocol (query, reply, update request, update reply) must be protected by IPsec. Address changes seem to be handled indirectly by errors and timeouts.

The main advantage of LIN6 is that the mapping between LIN6 generalized ID and LIN6 addresses removes the need for encapsulation (at the price of a resolution for connection to a new peer).

7 MHTP

Two-space systems are not only popular in the mobility community but also in the multi-homing one. The Multi-Homing Translation Protocol [20] (and its unpublished new version MHAP [21] where A stands for Aliasing) is a typical example of such systems.

MHAP is designed to be interoperable with the legacy IPv6 (named single-homed). Its identity space is Provider-Independent IPv6 address spaces using some special prefixes and allocated globally or locally (geographic (so aggregable) spaces) by /48 site prefixes. The locator space is the legacy (a.k.a. Provider-Aggregable) IPv6 address space.

A multi-homed end node knows (and is known by) only its PI address. An ingress router (a "client", between sites and the Default-Free Zone) translates the PI destination address into a PA address. Two kinds of clients are special:

- "endpoints" which translate back addresses to a PI space. They are the egress routers to multi-homed sites. They register "aliasings" too.

- "rendezvous points" which are the servers of the aliasing table management (they answer to aliasing queries). They offer a proxy translation service too.

Routing informations are carried by the legacy routing protocol (i.e., BGP4+) and aliasing resolution is provided by a protocol over UDP. Security is (very weakly) insured by a 64 bit request identifier (answers must reflect it).

The main choice in MHTP is to use Network Address Translation (NAT) in place of classical encapsulation which has the inconvenient of making MTHP a stateful system.

8 8+8 / GSE

MHTP is not the first attempt to solve once and for all the multi-homing problem for IPv6. The first notable proposal of this kind was "8+8" by Mike O'Dell of UUNET, renamed later GSE for Global–Site–End-system [22].

GSE splits the IPv6 address in three parts:

- a "routing goop" which specifies the public topology in the first 6 octets
- a "site topology partition" which specifies the private (i.e., in the site) topology in the next 2 octets
- an "end-system designator" which specifies an interface of the node in the last 8 octets. ESDs are globally unique.

This is the current format of global IPv6 addresses (a heritage of GSE) but in GSE the two-spaces are the two different parts (RG+STP a.k.a. the "routing stuff" and ESD) of addresses.

In GSE an end node knows only its STD and ESD (not its RG) but is known by its complete address outside of its site. The DNS maps names to a pair of prefix names which can be further resolved and the address in the prefix (i.e., last bits of the final address). This gave the A6 resource record [23] which is known to help re-homing (i.e., renumbering).

Site border routers add the routing goop to the source address of outbound packets and remove it from the source address of inbound packets (i.e., GSE is "NAT made right"). Peer routing goop changes are accommodated by using the last received routing goop (so inter-site mobility and multi-homing are directly supported).

The GSE proposal was rejected but also analyzed in deep [24] as a rather mature example of a two-space system. Some weaknesses were discovered, mainly in the identity to locator mapping (a very hard problem in GSE because of the lack of structure of ESDs) and in the authentication of identifiers. They gave new DoS vulnerabilities: for instance, an attacker can break the connectivity to a target node by sending TCP packets with an invalid routing goop for the target in the source address.

9 HIP

Therefore in order to design a secure two-space system, two problems must be solved in a secure way:

- the identity-to-locator mapping: either directly by a security aware resolution system, or indirectly by a suitable structure of identities and DNSsec with names as a kind of abstract identities

- the authentication of identifiers: as return routability checks are not available as soon as locators and identities are split, the standard solution is to use cryptography

The Host Identity Payload protocol (HIP) fulfills these requirements so it is a (perhaps the only) secure two-space system.

Locators in HIP are the standard IP addresses and communications between two HIP nodes are done by end-to-end ESP in tunnel mode. HIP identities are:

- Host Identities: public key (of an asymmetric key pair)
- Host Identity Tags (HIT) on 128 bits: either a 126 bit hash of the HI or a 62 bit Host Assigning Authority followed by a 64 bit hash of the HI (the first two bits encode the format in an IPv6 compatible way)
- Local Scope Identities on 32 bits: localized IPv4 compatible representation of the HI, not considered in this paper

A Host Identity can be anonymous: it can not be strongly authenticated (the only provable property is ownership as for Mobile IPv6 CGAs) and is roughly equivalent to a self-signed certificate. Non-anonymous Host Identities must be stored in the DNS: HIP relies on DNSsec for strong authentication.

HIP packet formats are not yet definitively defined so no accurate and up-to-date description can be given but there are:

- **I1** (the Initiator packet): it contains only the fixed HIP header (with the two HITs of parties)
- **R1** (the Responder packet): it contains after the HIP header:
  - the birthday (a reboot counter)
  - the cookie proposal (I, K and challenge)
  - the responder Diffie-Hellman value
  - the proposed HIP and ESP transforms
  - the responder Host Identity
  - an optional external signature of the HI
- **I2** (the Second Initiator packet): this is the first packet where state is created, its content is:
  - the responder's LSI and SPI
  - the birthday
  - the cookie reply (I, J and challenge)
  - the initiator Diffie-Hellman value

  - the selected HIP and ESP transforms
  - the encrypted initiator HI (which can be optionally signed)
  - the signature of the whole HIP payload

- **R2** (the Second Responder packet):

  - the initiator's LSI and SPI
  - the signature of the whole HIP payload

  R2 is the last packet of the 4 packets / 2 exchanges setup.

- **NES** (the New SPI packet): this packet changes the SPI, i.e., the Security Association Index, of a SA to the sender. Optionally it provides rekeying, it contains:

  - the ESP sequence number, old and new SPIs
  - an optional Diffie-Hellman value
  - the signature of the whole HIP payload

- **REA** (the Readdress packet): a node sends it when it changes its address. This provides partial support for mobility and multi-homing because only one endpoint can change its address at a moment. Its content is:

  - the ESP sequence number and SPI
  - the new address information (interface ID, lifetime and address)
  - the signature of the whole HIP payload

- **BOS** (the Bootstrap packet): this packet can be broadcasted where some parts of the infrastructure are not available. It contains:

  - the announcer Host Identity
  - an optional external signature of the HI
  - the signature of the whole HIP payload

In order to complete the mobility support, HIP proposes rendezvous servers which track IP addresses of HIP nodes and forward I1 packets (in general, this kind of mechanism is needed to upgrade a readdressing tool to whole mobility / multi-homing).

10 Further developments and works

If HIP is so nice, why is it not used? The answer is terribly simple: the HIP specifications are not yet finished and first implementation works are just beginning (including here at ENST Bretagne).

There are many points to improve:

- mobility and multi-homing need a smarter address management. The REA has already been modified with the introduction of interface IDs (peers have the capacity to track the current addresses of the node per interface). The next step should be the addition of "virtual interfaces" which is even more powerful than the notion of home address.
- address set management should enable path management, something very useful for multi-homing where blind address selection as mentioned can not be qualified to be smart.
- inner headers in ESP tunnels have a lot of fields which can be classified as "implicit" in a header compression framework [25] because they are selectors of the security association and may not change. ESP is traditionally associated to payload compression but header compression in general should be efficient too, and reducing (or even suppressing or reversing) the space overhead of ESP tunnels should have many benefits like the simplification of MTU problems.

So HIP is still evolving toward a long term solution, in two to five years, to Mobile IPv6 problems and, more important, in a near desperate situation because of the lack of interest from the R&D community, to multi-homing. Even if HIP will not succeed, it should be a major help in the understanding of mobility and multi-homing security issues.

At ENST Bretagne, we actively participate to HIP developments and we are starting an implementation on BSD operating systems.

References

[1] Y. Rekhter, T. Li, *An Architecture for IP Address Allocation with CIDR*, `RFC 1518`, September 1993.

[2] S. Kent, R. Atkinson, *IP Encapsulating Security Payload (ESP)*, `RFC 2406`, November 1998.

[3] D. Harkins, D. Carrel, *The Internet Key Exchange (IKE)*, `RFC 2409`, November 1998.

[4] CERT, *TCP SYN Flooding and IP Spoofing Attacks*, `Advisory CA-96.21`, September 1996.

[5] http://cr.yp.to/syncookies.html

[6] R. Stewart, Q. Xie & all, *Stream Control Transmission Protocol*, RFC 2960, October 2000.

[7] P. Vixie & all, *Secret Key Transaction Authentication for DNS (TSIG)*, RFC 2845, May 2000.

[8] D. Eastlake, *DNS Request and Transaction Signatures (SIG(0)s)*, RFC 2931, September 2000.

[9] D. Eastlake, *Domain Name System Security Extensions*, RFC 2535, March 1999.

[10] http://ops.ietf.org/dns/dynupd/secure-ddns-howto.html

[11] C. Perkins (ed.), *IP Mobility Support for IPv4*, RFC 3220, January 2002.

[12] D. Johnson, C. Perkins, J. Arkko, *Mobility Support in IPv6*, draft-ietf-mobileip-ipv6-17.txt, May 2002.

[13] G. Montenegro, C. Castelluccia, *SUCV Identifiers and Addresses*, draft-montenegro-sucv-02.txt, November 2001.

[14] P. Ferguson, D. Senie, *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing*, RFC 2827 - BCP 38, May 2000.

[15] M. Crawford, *Router Renumbering for IPv6*, RFC 2894, August 2000.

[16] J. Hagino, H. Snyder, *IPv6 Multihoming Support at Site Exit Routers*, RFC 3178, October 2001.

[17] F. Teraoka & all, *LIN6: A Solution to Mobility and Multi-Homing in IPv6*, draft-teraoka-ipng-lin6-01.txt, August 2001.

[18] http://www.csl.sony.co.jp/person/tera.html

[19] http://www-r2.u-strasbg.fr/~noel/

[20] M. Py, *Multi Homing Translation Protocol (MHTP)*, draft-py-multi6-mhtp-01.txt, November 2001.

[21] http://arneill-py.sacramento.ca.us/ipv6mh/

[22] M. O'Dell, *GSE - An Alternate Addressing Architecture for IPv6*, draft-ietf-ipngwg-gseaddr-00.txt, February 1997.

[23] M. Crawford, C. Huitema, *DNS Extensions to Support IPv6 Address Aggregation and Renumbering*, RFC 2874, July 2000.

[24] M. Crawford, A. Mankin, T. Narten, J. Stewart, L. Zhang, *Separating Identifiers and Locators in Addresses: An Analysis of the GSE Proposal for IPv6*, `draft-ietf-ipngwg-esd-analysis-05.txt`, October 1999.

[25] M. Degermark, B. Nordgren, S. Pink, *IP Header Compression*, `RFC 2507`, February 1999.