HARMONISED RESILIENCE, SECURITY AND MOBILITY CAPABILITY FOR IP

Randall Atkinson
Extreme Networks
RTP, NC, USA

Saleem Bhatti
University of St Andrews
St Andrews, UK

Stephen Hailes
UCL
London, UK

ABSTRACT

Military communications currently require secure end-to-end, resilient connectivity via multi-homed connections, and need to support both mobile hosts and mobile networks. Today, such functions are possible to some degree, but the functions are not harmonised. Standards that support these functions were designed independently and implemented in isolation. So, achieving converged capabilities for optimal communications in forward operating areas is a technical challenge, and results in a complex network landscape which is likely to be difficult to operate and manage, and brittle under failure conditions.

From our ongoing work, we present a new naming approach and use this to formulate a proposal to provide the following capability harmoniously: (a) multi-homed connectivity for traffic engineering and resilience; (b) true end-to-end network-layer security with high compatibility with the HAIPE architecture; (c) support for mobile hosts and mobile networks. Our approach is backwards compatible with IPv6 network equipment (existing IPv6 backbones can be used), and is also incrementally deployable.

I. INTRODUCTION

The data network is key to the Network Centric Operations (NCO) value chain. If we can improve the quality of the data network, we improve overall the degree of effectiveness. With an harmonised set of capabilities for site multi-homing, traffic engineering, end-to-end security, and support for mobile systems and networks, we can improve the underlying ability to deliver different mission capability packages (MCPs).

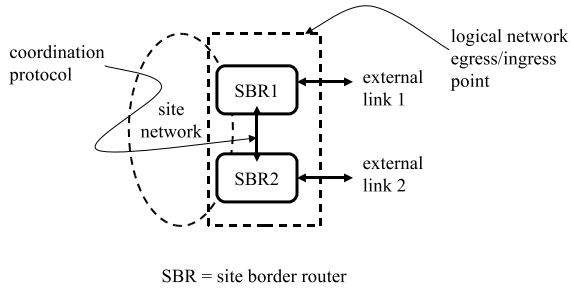

Fig. 1. General scenario: an example site network

We chose the abstraction for the site network of Figure 1 as it maps to many real scenarios, e.g. a warship (mobile network) with multiple satellite uplinks; an infantry platoon (mobile network) with multiple radio links; or a military base with multiple, redundant external links. We show only two external links, for simplicity, but a larger number of external links are possible, if desired.

The current independent design approaches to these functions create a very complex implementation, deployment, and operational environment for modern military networks, and are a hindrance to NCO. In many cases, the initial complexity is compounded by subsequent efforts to resolve limitations of the original approaches. The extensions often make things more complicated. This recombinant complexity impedes, rather than facilitates, NCO. For example:

- special purpose extensions are often needed to enable host mobility through NATs;
- use of IP mobility with IPsec has proven problematic, so work on key management extensions for mobility are being developed by the IETF;
- enabling current IPsec through a NAT device requires special mechanisms using UDP encapsulation;
- firewalls (and other middleboxes) typically need to be reprogrammed and/or reconfigured in order to be made aware of the operation of mobility and multi-homing.

In this paper, we present a set of harmonised functions (Section II) in a new naming approach (Section III), which would be used in the general scenario given in Figure 1, and could serve a number of MCPs. Here, a site network is connected to the rest of the world by one or more external links – site multi-homing (Section V) – for resilience and traffic engineering. The site network may be mobile (Section VI); it may use localised addressing internally for network management (Section IV); it may need to prioritise traffic flows – traffic engineering (Section VII); and it will require end-to-end security (Section III-F).

II. HARMONISED COMMUNICATION FUNCTIONS

The provision of harmonised, network functions across the protocol stack are essential for the provision of robust, scaleable and flexible Network Centric Operations (NCO). The current key capability requirements include site multi-homing, traffic engineering (TE), end-to-end communications security (COMSEC) and support for host mobility

978-1-4244-2677-5/08/$25.00 © 2008 IEEE

as well as network mobility using the Internet Protocol (IP). These capabilities are required today for the military, but have equally important application domains in civil and commercial scenarios. However, none of these requirements were considered when the current version of IP, IPv4, was first designed. So such capability tends to be retrofitted to IPv4 through engineering-based 'bolt-on' extensions. For IPv6, these additional requirements were raised by some. [1] However, current IPv6 standards do not offer harmonised support for these capabilities. Also, IPv6 continues with the IPv4 architecture, rather than adopting an improved architectural approach. Indeed, the recent *Routing & Addressing Workshop* held by the Internet Architecture Board (IAB) recognises that the current architecture is not adequate and needs to evolve. [2] In recognition of this, the IAB recently re-chartered the *Routing Research Group* of its *Internet Research Task Force*. The IETF Host Identity Protocol (HIP) WG [3], is working on some related issues. However, our approach differs to HIP in various ways, including (a) their approach *requires* identifiers for hosts to be a function of the respective public key of the host; (b) their approach *requires* the use of cryptographic authentication for all sessions.

A. *Current Approaches*

For our network functions of interest, the current state of implementation and deployment remains fragmented and lacks harmonisation. Site multi-homing and traffic engineering (TE) are in widespread operational use today in both military, civil and commercial networks. There are many ways in which these capabilities may be achieved for IPv4 based networks. End-to-end security is provided by IPsec [4], upon which High Assurance IP Encryptor (HAIPE) products are based. [5] Although the IETF has specified approaches to both host mobility (e.g. IPv6 [6]) and network mobility (i.e. NEMO [7]), neither has widespread deployment today, in part due to the complexity of the existing specifications. Myriad extensions have been proposed to address limitations of these specifications; sadly, the various extensions have the effect of increasing the complexity of implementation, deployment, and operation, and reducing backward compatibility to existing infrastructure.

The IETF's Mobile IP and Network Mobility (NEMO) approaches both overload the IP address for use as a location-independent identifier, while continuing to forward packets based on the IP address. These approaches also require the deployment and configuration of specialised infrastructure agents, known as Home Agents. The recently formed IETF MEXT (Mobile EXTensions) WG[1] seeks to combine features of Mobile IPv6 [6], NEMO for network mobility [7], along with IKEv2 for key management [8]. This will combine these mobility and security mechanisms, but of course will not be harmonised with multi-homing, localised addressing, or traffic engineering features.

[1] http://www.ietf.org/html.charters/mext-charter.html

B. *End-system Session State Transparency*

Existing standard transport-layer protocols, such as TCP and UDP, include the entire IP address of each endpoint node in their session state. So existing IETF work (e.g. Mobile IP) in all of these areas seeks to hide the real node location from this transport-layer session state. If the node's IP address changes in a way that is visible to TCP or UDP, then existing Transport-layer sessions will be lost. Some application protocols (e.g. FTP) also embed IP addresses. In turn, this means that those applications that embed IP addresses also might break if they become aware of the change in the node's real network location.

In the remainder of this paper we outline a proposal for harmonised support for multi-homing, traffic engineering, host mobility, and network mobility, based on our ongoing work. Essential to our approach is that:

- Changes in node location do not break existing transport-layer sessions or most applications.
- The proposal is fully backwards compatible with IPv6, so it can be introduced incrementally into existing IPv6 networks, without needing to upgrade components in the core network;
- This harmonised approach also simplifies the network engineering and network management requirements, relative to existing IP network deployments implementing the same functionality.

III. Overview of ILNP

In this section we present a brief overview of our proposed enhancements to the Internet Architecture, and also specifically to IPv6. We use the term *Identifier-Locator Network Protocol v6 (ILNPv6)* to refer to our proposal, as it can be engineered as enhancements to IPv6. [9]

A. *Naming Problems in IP today*

In our discussion below, we use the term *name* in a very general sense, to refer to any label that is attached to a network object. A summary is given in Table I.

It is important to recognise the two different functions for which the IP address is currently used – as a *locator* for naming an IP (sub)network, and as an *identifier* for naming a node. This overloading of the IP address causes entanglement across these functions and across protocol layers. The current use of the IP address is within applications,

TABLE I

TERMINOLOGY USED IN THIS PAPER

Term	DNS Record	Definition
Address	AAAA, A	Name used both for locating and identifying a network entity
Locator	L	Name that locates, topologically, a subnetwork
Identifier	I	Name that identifies a node, within the scope of a given locator

TABLE II

USE OF NAMES IN ILNP AND IP

Protocol layer	ILNP	IP
Application	FQDN	FQDN, IP address
Transport	Identifier, *I*	IP address
Network	Locator, *L*	IP address
Link	MAC address	MAC address

in the transport protocols (e.g. within the TCP pseudo-header checksum), and in the network layer to route packets between the end nodes – see Table II. The impacts include:

Localised Addressing: When a site uses Network Address Translation (NAT) to enable private addressing, harmonised use of multi-homing, mobility, traffic engineering, and end-to-end security become even more difficult, as the NAT introduces a discontinuity in the end-to-end state.

Multi-homing and Traffic Engineering: At present, multi-homing and traffic engineering require additional routing information to be kept in most or all backbone routers. Since IP routing uses longest-prefix match to select the preferred route to a destination, these two functions require additional more-specific IP routing prefixes to be advertised to all backbone routers globally. This is causing the size of the backbone routing tables to increase geometrically, raising scalability concerns. These concerns are sufficiently serious that the IAB has asked the Routing Research Group (Routing RG) of the Internet Research Task Force (IRTF) to investigate better approaches to these issues. [2]

End-to-End Security: IPsec Security Associations, which of course are also used by HAIPE products, include both the source and destination IP addresses. This means that if a node moves, or a network moves, then the existing IPsec Security Associations will cease to be valid. This constraint exacerbates existing concerns about the scalability of key management for IPsec devices. It also means that, regardless of what changes might be proposed for the Internet Key Exchange (IKEv2), support for mobility and multi-homing will remain limited and hard to deploy in the tactical environments where these capabilities are so crucial.

Support for Mobility: Both Mobile IP (v4 and v6), and NEMO, require that extra IP addresses, known as Care of Addresses (CoAs), be used with a special-purpose router, known as the Home Agent, using an IP-in-IP tunnel to forward packets sent by a correspondent from the mobile node's Home Address. To avoid *triangle routing* (i.e. packets travel from the correspondent node, via the Home Agent to the mobile node, but directly to the correspondent from the mobile node), Mobile IPv6 requires a control message be sent from the mobile node back to the correspondent node.

B. Naming Enhancements

Our approach recognises that an IP address has two very different roles – as a *locator* and as an *identifier*. So we replace the concept of the *address* with the concepts of an *Identifier* combined with a *Locator*. The *Locator* names an IP (sub)network: this is used only in routing, and not by the upper layers (e.g. TCP or UDP). The *Identifier* is only used for node identity (e.g. by TCP in the TCP pseudo-header checksum). This will be implemented such that the BSD Sockets API hides these details from applications, so existing applications generally should work without modification. For new applications, we propose the creation of an additional, more abstract, API that should simplify writing new applications.

The idea of an *Identifier/Locator* split is not a new idea, but our particular approach is new and is specified in more detail than preceding proposals. [10]–[12] We believe that applications should use fully-qualified domain names (FQDNs), wherever possible. A summary of the difference between the use of names in IP (v4 and v6) and the use in ILNP is given in Table II.

C. IPv6 Enhancements

While our approach above might seem abstract, we are implementing ILNP as an extension to IPv6, which we call *ILNPv6*. The similarities between the IPv6 packet header and the ILNPv6 packet header are deliberate. Essentially, the IPv6 address is broken into two separate components, a Locator (L) and an Identifier (I). Significantly, the IPv6 *Interface Identifier* is replaced by an ILNPv6 *Node Identifier* (I), with slightly different semantics.

The Locator (L) is an unsigned 64-bit value carried in the upper portion of the IPv6 address and is equivalent to an IPv6 address prefix. The (Node) Identifier (I) is an unsigned 64-bit value carried in the lower portion of the IPv6 address. The I value names a (virtual) node, rather than a network interface. An end-system may use multiple I values and multiple L values simultaneously. For the duration of a given session, its I value should remain constant. For practical reasons, the Identifier is normally formed from one of the MAC addresses associated with the node. This is represented in the IEEE's EUI-64 syntax,

```
IPv6:
        | 3 |     45 bits         |  16 bits  |       64 bits              |
        +---+---------------------+-----------+----------------------------+
        |001|global routing prefix| subnet ID |    Interface Identifier    |
        +---+---------------------+-----------+----------------------------+

ILNPv6:
        |             64 bits                 |       64 bits              |
        +---+---------------------+-----------+----------------------------+
        |             Locator                 |      Node Identifier       |
        +---+---------------------+-----------+----------------------------+
```

Fig. 2. IPv6 address format (from RFC3587 [14]) as used in ILNPv6

and is very likely to be globally unique as well. This usage is consistent with the IPv6 Addressing Architecture [13]. Strictly, the I value must be unique only within the scope of the L value with which it is used, but for practical purposes, having an I value that is likely to be globally unique is very useful, and allows us to dispense with IPv6 Duplicate Address Detection (DAD), which greatly reduces the time a node requires to execute a location change.

The ILNPv6 Locator is consistent with the IPv6 Addressing Architecture [13], specifically with section 2.5.4, which states that the sum of bits in the global routing prefix and the subnet ID is 64 bits. At present, IPv6 address allocation practices provide sites with IPv6 address blocks that are 48-bits long, so there are 16 bits left for sub-netting within the site. As the ILNPv6 network name (ILNPv6 Locator) is the same as an IPv6 network (IPv6 64-bit prefix), ILNPv6 packets travel across an unchanged IPv6 backbone, though the host IPv6 stack has to be enhanced to enable ILNPv6 on that host (to deal with Identifier values). ILNPv6 Neighbour Discovery (ND) would still use the full 128-bits of the combined I:L value. So IPv6 ND also can be used without change. *In short, already deployed IPv6 routers will support ILNPv6 without any changes.*

D. End-system Session State Invariance

Note that ILNP makes Identifiers visible at the top of the network layer, but only so that the Identifiers can be used by any transport layer protocol equally: Identifier values are not used by the network layer for routing.

Denoting the TCP, IP, and ILNP session state with the *tagged tuple* notation below, we consider a TCP connection for IPv4 or IPv6, with the end-system state represented as the tuples:

$$\langle TCP : a, p, b, q \rangle \tag{1}$$

$$\langle IP : w, x \rangle \tag{2}$$

$$\langle ILNP : y, z \rangle \tag{3}$$

where a and b are, respectively, the local and remote system names, and p and q are, respectively, the local and remote TCP port numbers. If the TCP tuple (1), which is the end-system state, can remain *invariant* during the operation of localised addressing, mobility, traffic engineering and end-to-end security, then it is clear that the end-to-end protocol is not affected and the operation of those functions are transparent (invisible) to the transport layer and so to the application.

For IPv6 or IPv4, a and b are, respectively, the local IP address and remote IP address. Further, w is equal to a, and x is equal to b. So, any changes to the IP addresses in use will cause end-system state to vary.

For ILNP a and b are the local Identifier and remote Identifier, respectively, while y and z are, respectively, the local and remote Locator values. So, changes in Locator values only will not affect end-system state. An internal cache at the top of the network-layer within the ILNPv6 implementation will keep track of the current I:L mappings for existing ILNPv6 sessions.

In this paper, we use the network scenario of Figure 1 with TCP as an example to show how the functions of end-to-end security, multi-homing, mobility, localised addressing and traffic engineering can be provided in harmony.

E. DNS Enhancements

To enable ILNPv6, several new DNS resource records are needed. We add the *I* record, which contains the unsigned 64-bit Identifier associated with a domain name. Similarly, the *L* record contains an unsigned 64-bit Locator associated with a domain name. As a node might have multiple Identifiers and multiple Locators, a given domain name also might have multiple *I* and multiple *L* records. The combination of a given L record and an associated I record is equivalent to the current IPv6 address.

Reverse lookups can be done as today with IPv6. As a performance optimisation, we also have a pair of new DNS records that could be used for reverse lookups. The *PTRL* record names an authoritative DNS server for an ILNPv6 subnetwork, while the *PTRI* record is used to obtain the name of a node using a given Identifier on a given subnetwork. This usage enables *PTRL* records to be cached, which is beneficial if performing reverse lookups for multiple nodes on the same subnetwork.

As a separate performance enhancement for managing site networks, we also introduce the *Locator Pointer* (*LP*) record. This record points to an *L* record. Nodes that are attached to a site network (which could be a mobile network) would typically have an *LP* record that pointed to the *L* record of that site network. So when the site network moves its point of Internet connection, only the network's own *L* record needs to be updated.

The existing Secure Dynamic DNS Update standard [15] permits a mobile node or multihomed node to update its *L* records when the node moves or its upstream connectivity

changes (e.g. due to a link fault). Separately, the DNS enhancements for ILNPv6 do not change the fundamental operation of the Domain Name System (DNS). So the the existing DNS Security (DNSsec) standards [16] can be used unchanged to authenticate these new DNS records. So our proposed enhancements do not create new security risks.

F. *IP Security Enhancements*

The High Assurance IP Encryptor (HAIPE) used to protect existing military IP networks is a US DoD profile of IETF standard IP Security [5], so our discussion of IPsec also addresses military deployments of IPsec. [17] In IPsec today, the IPsec Security Associations (SAs) are bound to full IP addresses at the local and remote sites, a and b, respectively, as a form of end-system identity. So, for tuple 2, IPsec requires that the IP addresses at each end-point of the communication remain fixed. For localised addressing, multi-homing and mobility, this may not remain true, and so IPsec has had to be modified, retrospectively, in order to cope with these functions.

With ILNP, however, IPsec SAs are bound only to the Identifier (a and b, in tuple 1), never to the Locator. This makes it easy for the IPsec Security Association – and the related secure communications channel – to remain operational even if the end-points move. So, for ILNP, with respect to tuple 3, the invariance of the end-system identity, due to the use of Identifiers a and b in the TCP tuple, must be true for multi-homing, mobility, and traffic engineering for IPsec to work harmoniously with those functions. We will show that the property does indeed hold.

IV. Localised addressing

To support private, localised addressing, IP provides three well-known IP networks in a process known as Network Address Translation (NAT) [18]. NAT boxes reside at the site border router (SBR) of the privately addressed network and re-write addresses and checksums at the IP and TCP layer, translating between the privately used (local) address, A_L, and the globally unique (routable) address, A_G, for that site, and port numbers may also be re-written so that A_G can be shared amongst many nodes in the private network. Let us consider a TCP connection with the end-system state at the private network as:

$$\langle TCP : A_L, P_L, A_R, P_R \rangle \langle IP : A_L, A_R \rangle \tag{4}$$

where A_L is the local IP address, P_L is the local port number, A_R is the remote IP address and P_R is the remote port number. However, after traversing a NAT, the TCP state at the remote node (correspondent) will be:

$$\langle TCP : A_G, P_G, A_R, P_R \rangle \langle IP : A_G, A_R \rangle \tag{5}$$

where A_G and P_G are, respectively, the address and port number written by the NAT function: the end-system state is different at each end of the connection and the NAT holds the mapping. This can be disruptive to many applications and functions such as IPsec and mobility.

With ILNPv6, the end-system state is bound only to the Identifier, and only the Locator is used for routing. So, ILNP end-system state of any TCP connection would be:

$$\langle TCP : I_L, P_L, I_R, P_R \rangle \langle ILNP : L_L, L_R \rangle \tag{6}$$

where I_L and I_R are, respectively, the local and remote Identifier values. An ILNPv6 NAT would re-write only Locator values between, say, L_L, the local (private) Locator value, and L_G, the globally unique Locator value, which are only seen at the network layer packet. So, an ILNPv6 NAT is transparent (invisible) to the end-system connections. For example, if L_L is a local (private) Locator value for our end site, L_G is the global Locator value for our end site, and L_R is the remote Locator value, the TCP packet before the ILNPv6 NAT would be as in tuple (6), and after traversing the ILNPv6 NAT would be as in tuple (7).

$$\langle TCP : I_L, P_L, I_R, P_R \rangle \langle ILNP : L_R, L_G \rangle \tag{7}$$

This maintains the invariance requirement (Section III-D). We have examined this ability to re-write Locator values for ILNPv6 in the context of NATs. Although ILNPv6 NAT is not required to support site multi-homing, mobile networks, and traffic engineering, we shall see that it can be used to support efficiently and conveniently all these functions through a single (or multiple) ILNPv6-capable SBR(s).

V. Multi-homing

There are two kinds of multi-homing that need to be considered. Site multi-homing is when a given site has multiple upstream connections to different service providers. This, in combination with BGP and IP routing, can provide greater resilience and availability to all of the nodes within that site. This also enables an IP session to be maintained even if one uplink from the site fails and another begins use.

A. *Site Multi-homing*

Today, site multi-homing is implemented by advertising the site's more-specific IP routing prefix to the entire Internet and relying on the Internet's normal longest-prefix-match route selection algorithm. Unfortunately, this requires that IP routing prefixes to be de-aggregated. So instead of an

ISP advertising a single IP routing prefix that covers all of its customers, there are additional more-specific prefixes for each multi-homed site using that ISP. This practice is the largest source of entropy in the global routing table today. [19] Routing scalability has become a major concern, in large measure due to the current geometric growth in routing entropy. [2]

B. *Host Multi-homing*

At present, host multi-homing works by advertising multiple IP address records, either *A* records for IPv4 or *AAAA* records for IPv6, in the DNS to correspondents. However, this long-standing practice does not permit a given TCP session to continue working if the host interface used for that TCP session changes (e.g. due to a link fault).

C. *ILNP Site Multi-homing*

ILNPv6 uses the same mechanism to provide both site multi-homing and host multi-homing. With ILNP, the new DNS *L* or *LP* records are used to advertise the current reachability for a node or site. New correspondents perform a DNS lookup, as at present, to determine how to send packets initially to the target node(s). Whenever a node's currently valid Locator(s) change, the node sends *ICMP Locator Update (LU)* control messages to its existing correspondents. These messages can be authenticated either cryptographically using the IP Authentication Header, or non-cryptographically, as appropriate for the node's threat environment. The correspondent receives this update, validates it, and then begins using the new Locator(s) to send packets to the original node.

Using Figure 1, consider an IPv6 site network using two routing prefixes, R_1 and R_2. Often, sites prefer the prefixes R_1 and R_2 to be provider independent, as the address prefixes are considered as part of the site's *identity* as well as providing routing information. Each SBR has to advertise both R_1 and R_2 on *both* links, i.e. four *additional* routing entries to advertise. In general for IPv6, the number of additional prefixes advertised is $N_P.N_L$, where N_P is the number of prefixes and N_L is the number of external links. For ILNPv6, we can use Locator values L_1 and L_2, respectively on external link 1 and external link 2. These can be taken simply from the upstream provider's Locator space and need not be Provider Independent: the site maintains names for identity by using Identifier values. As the Locator values are not part of the transport protocol state, we can use both Locator values simultaneously. *So, no additional prefixes need to be advertised.*

If we consider Localised Addressing (Section IV), using tuple (7) as the TCP packet state from our site network, then packets using SBR1 will have the state given in tuple (8) and packets using SBR2 will have state as in tuple (9).

$$\langle TCP : I_L, P_L, I_R, P_R \rangle \langle ILNP : L_1, L_R \rangle \tag{8}$$

$$\langle TCP : I_L, P_L, I_R, P_R \rangle \langle ILNP : L_2, L_R \rangle \tag{9}$$

Note that the TCP state at each end of the connection remains the same – the invariance requirement (Section III-D) is maintained. So, ILNPv6 can provide transparent multi-homing to the site-network. Whilst ILNPv6 does not need to use the SBR locator re-writing to support multi-homing, it provides an engineering optimisation and a good point for network management.

VI. Mobility

When considering mobility, it is important to understand that the military has both mobile nodes (e.g. a Humvee) and also mobile networks (e.g. aircraft, ships). Both kinds of mobility are important to tactical networking, so both kinds of mobility need to be natively supported in the deployed network.

We observe that mobility and multi-homing are so closely related that they can be difficult to distinguish from each other. For example, if a network supports node mobility, then the network can support node multi-homing with session resilience, e.g. a TCP session stays up even if the interface used for the session changes due to a link fault. Similarly, if mobile networks are fully supported, the same mechanisms can be used to provide network multi-homing. We recognise that the use of appropriate link-layer mobility mechanisms is important to providing a total solution to military mobility needs. Finally, we note that the use of *soft handoffs*, where the new uplink is established before the existing uplink disappears, is always recommended practice (both at the link-layer and separately at the network-layer) as this practice minimises the chances of packet loss during a mobile handoff. [20], [21]

A. *Host Mobility*

For the past two decades, the normal way to find the location of a remote system has been to look up its IP address from the Domain Name System (DNS). Use of DNS to find a remote node's location has worked well, is universally deployed, and is used by virtually all applications. We propose to extend this use for mobile nodes.

With ILNP, a mobile node uses the IETF's standard *Secure Domain Name System (DNS) Dynamic Update* to ensure that its currently valid Locators are kept current in the DNS. So new correspondents use the DNS to determine the current location of the mobile node.

For existing ILNP sessions, when a node moves its network location, the node will not only update the DNS as mentioned above, but also will send out new *ICMP*

Locator Update (LU) control messages. These new ICMP messages are sent to all existing correspondents and inform the recipients of the new (set of) Locator(s) that are valid for the mobile node sending the ICMP message. These ICMP messages are always authenticated, so this does not create an opportunity for an adversary to impersonate a mobile node or hijack an existing ILNP session.

ILNP has de-coupled the upper-layer protocols and applications from the location of the session endpoints. Locator values can change over time without causing existing transport-layer or application-layer sessions to be dropped. Additional details of our host mobility approach have been described elsewhere in more detail. [22] [9]

B. *Network Mobility*

For Mobile Networks, multiple approaches might be used. In one approach, the site uses private addressing *internally* (to the site network) and the network's SBR(s) rewrite the Locator values of nodes within the site as packets transit that SBR. In this model, nodes that are attached to the mobile network segment normally have DNS *LP* records that point to a common DNS *L* record covering the entire mobile subnetwork. The common *L* record would be updated by the SBR whenever its uplink moves to a different layer-3 ILNPv6 network.

If we consider again Figure 1. Let us assume that, internally, all TCP packets have state as in tuple (6). Let us assume that the network is mobile and has two external links with Locators L_1 and L_2 respectively. These will be held in DNS *L* records pointed to by a DNS *LP* record. As a convenience for network management, the SBR provides Locator rewriting (Section IV) but this is not necessary for supporting mobile networks with ILNPv6. Now let us assume a hand-off is triggered for the link currently using L_1. A signal is detected in the new cell and a new Locator value, L_3 is attained. This can be done through normal IPv6 discovery mechanisms as Locator values are identical to IPv6 network prefixes. We will assume that the radio cells providing L_1 and L_3 overlap.

Now, the SBR updates the DNS *L* record to L_3 (for new sessions) and starts changing the state of connections using L_1 to L_3 by issuing *Locator Update (LU)* messages (synonymous to Binding Update message in IPv6) for correspondents using L_1. It then transitions sessions using Locator rewriting from L_1 to L_3. For any given session using a remote $I_R : L_R$ pair that was using L_1, when no more packets arrive from that remote location L_R using L_1 within a given time period (i.e. all sessions have transitioned to L_3), the connection is considered to have completed hand-off. This is a *soft hand-off* at the ILNPv6 layer, something that is not currently defined for IPv6. Note that the SBR is providing this capability efficiently for the whole mobile network. Note also that during this time, the link using Locator L_2 continues to operate as long as the external link 2 is sound, i.e. multi-homing is possible during mobility.

It is also possible to use ILNPv6 for normal hand-off, simply by switching to L_3 as soon as possible. Any packets in flight addressed to L_1 may be lost, but can be recovered through the retransmission capability in TCP. This may be considered inefficient, as it will invoke the congestion control behaviour of TCP (due to missing TCP ACKs).

Meanwhile, for both normal hand-off and soft hand-off, we maintain the invariance requirement (Section III-D), as no Identifier values have changed.

VII. Site Exterior Traffic Management

The approach to site traffic engineering (TE) exploits the ability to use multiple Locator values and multiple uplinks. Today's policy-based mechanisms for site TE can be used to filter flows (e.g. based on network layer or transport layer headers) and associate a TE policy with each flow, as required, and the selection of the correct egress interface. The same approach can be used to provide obfuscation for the interior topology of a site. This last capability is commonly desired by Internet-connected end sites that have high threat profiles, such as military or homeland security sites. For ILNPv6, policies can use node identity regardless of location, making it easier to configure and maintain TE policy. However, Locator values could be used to give conditional policy, if required.

A. *Locator Rewriting for Traffic Engineering*

Site Border Routers (SBRs) are permitted to rewrite both source and destination Locator value(s), after selecting the egress interface for a packet, but before forwarding the packet. If there are multiple SBRs in use, they will need to share session state among themselves. This is the same issue as arises for multi-homed sites with a firewall at each border. That distributed firewall synchronisation problem is already solved in commercially available products; the same solution approach can be applied here.

For example, consider two packet flows as in tuples (10) and (11). Each of these flows is from a separate host in the site network of Figure 1, using a local Locator value L_L, and have separate destination networks, identified by L_J and L_U, respectively.

$$\langle TCP : I_1, P_1, I_J, P_K \rangle \langle ILNP : L_L, L_J \rangle \tag{10}$$

$$\langle TCP : I_2, P_2, I_U, P_V \rangle \langle ILNP : L_L, L_U \rangle \tag{11}$$

As these traverse the SBR, an internal policy decides that the first flow should traverse link 1, using Locator L_1, and the second flow should traverse link 2, using Locator L_2:

$$\langle TCP : I_1, P_1, I_J, P_K \rangle \langle ILNP : L_1, L_J \rangle \quad (12)$$

$$\langle TCP : I_2, P_2, I_U, P_V \rangle \langle ILNP : L_2, L_U \rangle \quad (13)$$

So, the SBR simply rewrites the local Locator value, L_L as required. We note this maintains with invariance requirement (Section III-D).

B. Locator Rewriting for Topology Obfuscation

Many sites consider their internal site topology to be sensitive information, and want to retain the concept of obscuring their internal topology from external observers. The preceding Traffic Engineering approach can also be applied to this situation. In this case, referring to Figure 2, the 16 bits of the *subnet ID* in the Locator would also be changed, e.g. set to zero, and the mapping to the correct subnet (indexed by the Identifier value) maintained by the SBRs.

In this situation, the site has has only one Locator (L) record per upstream connection, and each node within the site has a Locator Pointer (LP) record for each upstream link pointing to the corresponding L record.

Packet processing for packets arriving at a site border router, whether from either inside or outside the site, is the same as described in the preceding section.

As noted previously, the separation of Identifier from Locator means that this ILNPv6 locator rewriting capability differs from traditional NAT in that the transport-protocol session state, which now binds only to Identifiers and never to Locators, is not affected.

VIII. Conclusion and Further Work

We have presented an evolutionary approach to providing harmonised resilience, security, and mobility for IPv6. This architectural change leads to significant improvement in operational capability for military networks, particularly for mobile and tactical networks. This enhanced capability does not sacrifice security, and in fact can enhance the deployability and capability of militarised IP Security, such as the High Assurance IP Encryptor (HAIPE) products. Our approach is incrementally deployable and backwards compatible with existing IPv6 implementations.

To date, our research has not specifically addressed Mobile Ad-Hoc Network (MANET) capabilities. Informal conversations about our architecture with leaders in the military MANET area lead us to believe that our enhanced architecture can also yield benefits within the MANET environment. A logical future direction is to examine application of our improved Internet Architecture to MANET technologies.

Recent UK research into a *Distributed MANET DNS* seems particularly complementary to our work. We believe that this enhanced DNS approach can be integrated with our overall architecture and could be an important part of our future MANET capabilities. [20]

References

[1] B. Adamson, "Tactical Radio Frequency Communication Requirements for IPng," IETF, RFC 1677, Aug. 1994.

[2] D. Meyer, L. Zhang, and K. Fall, "Report from the IAB Workshop on Routing and Addressing," IAB, RFC 4984, Sept. 2007.

[3] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host Identity Protocol," IETF, RFC 5201, Apr. 2008.

[4] S. Kent and K. Seo, "Security Architecture for the Internet Protocol," IETF, RFC 4301, Dec. 2005.

[5] US DoD, "High-Assurance IP Encryption Interoperability Specification (HAIPE IS), Version 1.3.5," May 2004.

[6] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," IETF, RFC 3775, June 2004.

[7] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol," IETF, RFC 3963, Jan. 2005.

[8] V. Devarapalli and F. Dupont, "Mobile IPv6 Operation with IKEv2 and the Revised IPsec Architecture," IETF, RFC 4877, Apr. 2007.

[9] R. Atkinson, S. Bhatti, and S. Hailes, "A Proposal for Unifying Mobility with Multi-Homing, NAT, and Security," in *Proceedings of 5th ACM International Workshop on Mobility Management and Wireless Access*. Chania, Crete, Greece: ACM, Oct. 2007.

[10] C. Bennett, S. Edge, and A. Hinchley, "Issues in the Interconnection of Datagram Networks," ARPA Network Working Group, Internet Experiment Note (IEN) 1, July 1977.

[11] I. Castineyra, N. Chiappa, and M. Steenstrup, "The Nimrod Routing Architecture," IETF, RFC 1992, Aug. 1996.

[12] M. O'Dell, "GSE - An Alternate Addressing Architecture for IPv6," IETF, Internet-Draft draft-ipng-gseaddr-00.txt, Feb. 1997.

[13] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," IETF, RFC 4291, Feb. 2006.

[14] R. Hinden, S. Deering, and E. Nordmark, "IPv6 Global Unicast Address Format," IETF, RFC 3587, Aug. 2003.

[15] B. Wellington, "Secure Domain Name System (DNS) Dynamic Update," IETF, RFC 3007, Nov. 2000.

[16] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, "DNS Security Introduction and Requirements," RFC 4033, IETF, RFC 4033, Mar. 2005.

[17] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol," IETF, RFC 2401, Nov. 1998.

[18] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," IETF, RFC 1918, Feb. 1996.

[19] X.-Q. Meng, Z.-G. Xu, B.-C. Zhang, G. Huston, S.-W. Lu, and L.-X. Zhang, "IPv4 Address Allocation and the BGP Routing Table Evolution," *ACM Computer Communications Review*, vol. 35, no. 1, pp. 71–80, January 2005.

[20] J. Weston, "Interoperable Networks for Secure Communications, Phase 2, Task 3," North Atlantic Treaty Organisation (NATO), Final Report INSC-TASK3, July 2006.

[21] J. Macker, "Interoperable Networks for Secure Communications, Phase 1, Task 6," North Atlantic Treaty Organisation (NATO), Final Report INSC-TASK6, Dec. 2003.

[22] R. Atkinson, S. Bhatti, and S. Hailes, "Mobility as an Integrated Service Through the Use of Naming," in *Proceedings of 2nd ACM International Workshop on Mobility in the Evolving Internet Architecture*. Kyoto, Japan: ACM, Aug. 2007.