An Integrated IPSEC and Mobile-IP for FreeBSD

Jim Binkley
Computer Science
Portland State University
Portland OR 97201
jrb@cs.pdx.edu

October 3, 2001

Abstract

Recently the Layer 2 802.11 link-layer security mechanism called WEP has been shown to be flawed. In this paper we present a combined layer 3 Mobile-IP and IPSEC routing architecture that is superior to WEP. We overview possible routing security architectures and then present two integrated IPv4 Mobile-IP/IPSEC routing daemon architectures that we refer to as "closely-coupled" and "loosely-coupled". The closely-coupled architecture relies on a direct binding of IPSEC policy attributes to routing table entries by Mobile-IP routing daemons. The loosely-coupled architecture is based on a more traditional access control list association between the current BSD IPSEC implementation and PSU Mobile-IP. Our discussion concludes with an architectural analysis of combined Mobile-IP/IPSEC and a call for the use of IPSEC as part of any mobile VPN scheme.

1 Introduction

Recently wireless security as found in the popular 802.11 [10] wireless protocol has suffered a series of failures due to the apparent collapse of part of the 802.11 specification called WEP, for "Wired Equivalent Privacy". WEP was intended to offer security services including encryption and authentication. Security experts have recently demonstrated substantial security problems with WEP. Borisov, et. al.[4], describe security and network architecture flaws in WEP. A recent cryptoanalytical paper [6] then described a theoretical attack against the RC4 stream cipher used in WEP. Worse, in a recent ATT technical report [17], the theoretical attack was implemented.

The ATT paper in its conclusions suggests that the 802.11 link layer be viewed as insecure. To be fair, the IEEE 802.11 specification stated that WEP was not only optional, but was intended to make wireless "at least as secure as a wire". Unfortunately, this may be misleading to naive users who assume that WEP would offer serious confidentiality services.

Borisov's paper, and the ATT technical report both suggest that users should consider using higher-level protocols; for example, IPSEC [9] and Secure Shell [18]. We concur and further suggest that IPSEC could be directly combined with Mobile-IP [12] in order to make a Virtual Private Network mechanism that is completely based on the layer 3 network layer. DARPA funded research along those lines at Portland State, between 1995-1999 [15]. In this paper, we intend to explore two different experimental system architectures for a combined Mobile-IP/IPSEC implementation. The earliest architecture was created on the FreeBSD 2.2.8 platform using an IPSEC, originally done for NetBSD by the Naval Research Labs, and the PSU Mobile-IP implementation. Our most recent architecture is based on the current BSD IPSEC, called KAME [8]. We have combined KAME, and our Mobile-IP to make an integrated IPSEC/Mobile-IP.

In this paper, we assume a basic understanding of Mobile-IP. However the reader should note the following terms:

- 1. *Mobile Node* - a system that may wander from one subnet point of attachment to another. Mobile-IP assumes that this system always has one fixed "home" IP subnet address. We will abbreviate this term as *MN*. We will call the fixed IP address, the *MN IP*.
- 2. *Home Agent* - a system that acts as a home router for a Mobile Node. We can assume that the fixed "home" subnet is present here. We will abbreviate this term as *HA*.
- 3. *Foreign Agent* - a system that acts as a non-local subnet point of attachment for a Mobile Node. We will abbreviate this term as *FA*. By definition, the Foreign Agent and MN do not share an IP subnet on a given link.

1.1 Mobile Routing Security Policies

During the DARPA-funded Secure Mobile Network project period, we decided that from a formal point of view, we could distinguish three very general architectural frameworks for mobile routing security. We will call these architectural constructs *secure mobile routing architectures.* These architectures are as follows:

- *MN to security gateway VPN*: A two-way VPN is setup between a Mobile Node and some security gateway that acts as an "entry point" into a secure enclave. In figure 1, on the lower left-hand side, we show an example of this architecture. From the point of view of traditional firewall thinking, the security gateway is a bastion host. In Mobile-IP terms, it may be co-located with the Home Agent (which is the assumption we make in our implementation). Using IPSEC, we setup a two-way layer 3 ESP tunnel, which might or might use dynamic keying. As we will present later in more detail, we have implemented this form of VPN in both of our combined Mobile-IP/IPSEC implementations. Of course, other possible VPN technologies may be used. A Mobile Node outside a secure enclave, has a two-way IPSEC VPN to and from its Home Agent. Foreign Agents are not involved directly in any security association and are merely tunneled over (as are any other layer 3 entities). The first link may be assumed to be wireless, and can be assumed to be outside the secure enclave. The path between the MN and security gateway may be multi-hop and may span the Internet or barring the first link, may be internal to the secure enclave. [1] In terms of the number and scalability of key associations, key management is linear; that is, for each HA, we have a set of MNs. Key management may be made more complex by security gateway (HA) redundancy issues. We do not rule out a centralized key management system within the secure enclave, that might, for example, use DNS or some other system.

- *Agent boundary VPNs*: In this form, we restrict cryptographic services to the "external" link; that is, MNs are assumed to be outside the secure enclave, have two-way VPNs between themselves and a boundary agent, and the link connectivity is confined to only one link. The typical boundary agent could have one external link and one internal link. Boundary agents might be layer 3 entities as with Mobile-IP agents, or layer 2 entities as with 802.11/WEP access points. Please see the upper right-hand corner of figure 1 for an example of such an architecture. Boundary agents in a MIP system would insist that MNs must have an a-priori security association. Thus MNs that do not have local IPSEC keys would not be able to penetrate the secure enclave. In this

[1] The exact security status of the Home Agent as a security gateway is open to question. For the sake of argument, the reader may assume that the Home Agent is trusted by definition, as it is "at home". On the other hand, it is very reasonable to assume that the HA is a bastion host. Different levels of trust for the HA should be considered.

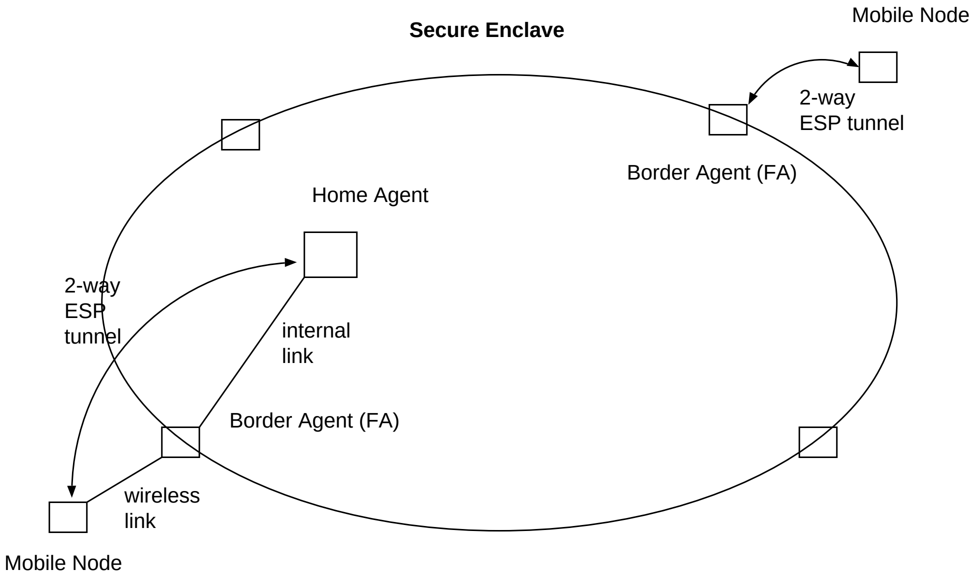

Figure 1: Two Layer 3 VPN architectures

security architecture, FAs, by definition must belong to your security enclave, and MN-FA security assocations must exist. Note that in the previous architecture, FAs were not part of the picture. Manual key administration here is fundamentally *not* scalable. as key associations are a function of the number of boundary agents times the number of Mobile Nodes. We believe that an internal tie-in between IKE daemons and centralized key service, possibly via DNSSEC is mandatory. The SMN project did not implement such an architecture, although we view it as possible future work. On the other hand, we did implement a layer 3 authentication system for Mobile-IP itself, that required authenticated ICMP advertisements from all network elements including agents and MNs [2]. Both BBN [19] and SMN also implemented layer 3 authentication systems based on per node digital signatures. Note that such authentication systems are not intended as replacements for higher-order confidentiality systems like IPSEC. They are merely supplemental.

- *Secure multi-hop ad hoc routing*: Multi-hop ad hoc routing refers to Mobile Nodes that setup multi-hop routing paths via a new class of dynamic routing algorithms; for example, please see [3]. The SMN project implemented a form of DSR [7] in which end host to end host IPSEC associations were manually available. Thus all packets between any two MNs could have IPSEC applied to them. It is important to note that "consenting" MNs in such an architecture, by definition, belong to the same security domain. Please see figure 2.

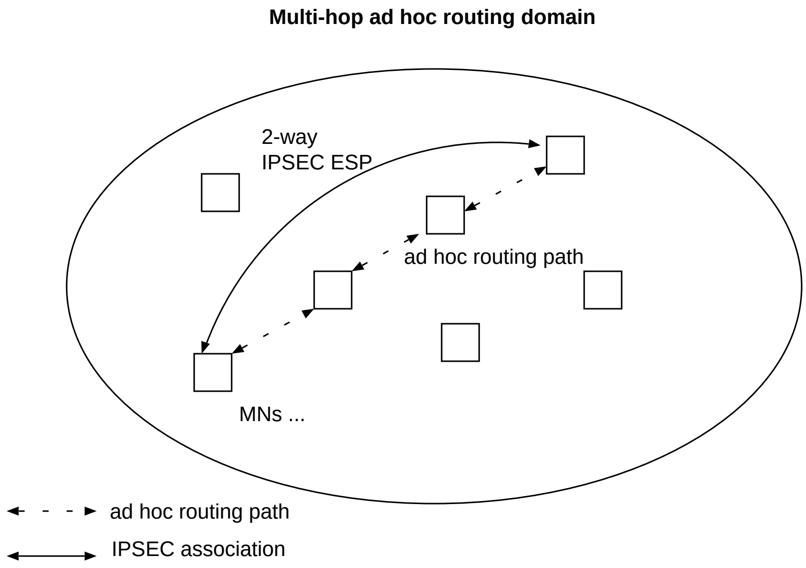

Figure 2: Secure multi-hop ad hoc routing

In section 2, we introduce the PSU mobile-ip routing architecture, and explain how the routing daemons function vis-a-vis the BSD routing table. In section 3, we present our older closely-coupled IPSEC/Mobile-IP architecture. In section 4, we present our newer loosely-coupled architecture in which we have combined our Mobile-IP with the current BSD KAME/IPSEC system. In section 5, we present some architectural analysis in terms of system organization, and finally in section 6, we present our conclusions.

2 PSU/FreeBSD Mobile-IP routing daemon architecture

In this section, we will briefly introduce the Mobile-IP architecture used in the PSU Mobile-IP routing daemons. This architecture is basic to understanding the two possible IPSEC implementations we later introduce.

Packets sent by other systems to the MN may be tunneled from the HA to the FA using tunneling technologies like IPIP encapsulation. Note that the FA has associated with it one fixed permanent address that must be learned by the MN, and registered with the HA. This address is called a *Care Of Address*, or *COA*. Thus the HA can tunnel packets to the MN via the FA.

IPIP encapsulation thus becomes:

```
outer IP       inner IP
IP src = HA IP IP src = peer IP
IP dst = COA   IP dst = MN IP
```

This IP tunnel architecture notion is crucial to Mobile-IP. MNs moving from one FA to another, must re-register so that the HA can move the tunnel destination from one COA to another COA. The IPIP tunnel can be viewed as a routing address two-tuple { COA IP, MN IP }, that conceptualizes the location of the MN. The COA is simply NULL when the MN is at home.

The PSU Mobile-IP routing daemons are named *mnd* and *mipd*. Mnd is used on MNs. Mipd provides functionality for both kinds of agents, Home and Foreign.

Mnd implements a state machine based on several event types. Events may be driven by clock timeouts, or by Mobile-IP messages which consist of UDP registration acknowledgements from the HA (which may be forwarded by any local FA), and ICMP router advertisements, sent by agents. The ICMP router advertisements are used by mnd to discover agents and to make decisions about which agents to use for the Mobile-IP registration and tunneling process. The basic state machine has 3 major states: *NOWHERE*, *AWAY*, *ATHOME* as pictured in figure 3, and a few minor states.

We do not have space here to cover the state machine in detail and only wish to convey a brief idea of its internal functionality. For example, a MN is in NOWHERE state when it cannot hear any agent beacons. It may then choose to try and act as its own Foreign Agent, and use DHCP to acquire a COA. If beacons are heard, it may transition to AWAY state (used at FAs) or at ATHOME state (used at HAs).

Mnd's model as a routing daemon is conceptually straight-forward. For the most part it dynamically manipulates the default route. When AWAY, it sets the default route to a FA's local link IP address. When at home, the HA becomes the default route. ARP table additions for agents are made or deleted as necessary. In general transitions are driven by MIP UDP registration or ICMP router advertisement messages. State changes

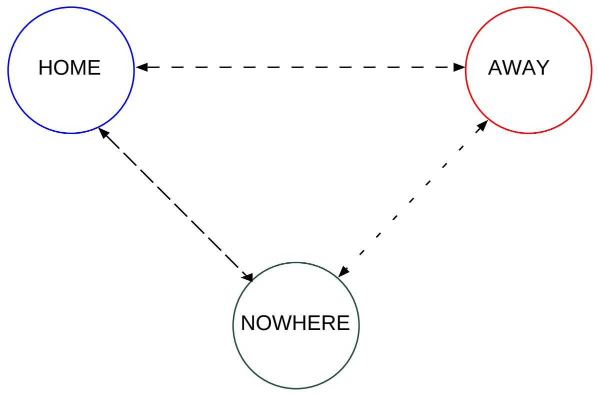

Figure 3: Mobile-IP MN states

to NOWHERE mode occur upon agent list timeouts. For example, if no agent beacons are heard for awhile or an individual registration times out, the daemon may delete its default route and move to NOWHERE state. The routing model for a foreign subnet can be summarized as follows:

```
default route:
to default (0.0.0.0)  via  FA IP

ARP table entry:
to FA IP         via  MAC address
```

This is "interesting" simply because the MN does not share an a priori subnet association with the FA.

The agent daemon, mipd, manipulates routes depending on its precise function as a HA or a FA. At the HA, link-layer routes to the MN are automatically added when the MN is at home, and a proxy ARP route is added when the MN is away. When the MN is away, an IPIP tunnel route [2] is installed.

The tunnel route is done via a special IPIP routing device called *mvif* for "mobile virtual interface". The mvif device takes packets sent to a particular MN IP and prepends a new outer IP header on the packet, using a HA address as IP source, and the COA (the routing table entry gateway) as the outer IP destination. The packet is then fed to the routing table again, and will be routed to the COA. Any number of mvif host entries may be

[2] This is a host route; that is, a /32 route to a particular MN IP address.

installed in the routing table. We felt this mechanism allowed us to take direct advantage of the BSD routing radix algorithm mechanism, which is very efficient [16].

Logically such a route looks like this:

```
to MN IP   via  COA IP  thru mvif device
```

Foreign agents are relatively simple (barring fundamental kernel modifications). They use the mvif device to decapsulate IPIP packets, and simply install (or remove) link-layer routes for local MNs.

A few kernel modifications are needed for PSU Mobile-IP support. The basic set of modifications follow:

- An IPIP driver is needed. The mvif driver itself must be installed. The mvif driver encapsulates and decapsulates IPIP packets. Typically it is used by the HA for a per MN host route, with the gateway set to the COA. It may also be used by MN's for tunneling all packets back to the HA.

- Non-local route binding. The BSD operating system does not allow the binding of non-local IP routes to a known interface. For example, if you need to route packets to a local link using the MN IP 10.0.0.1, BSD requires that you have an interface that a priori belongs to the 10/8 subnet. [3] Both MNs and FAs need to be able to bind arbitrary subnet routes to arbitrary interfaces, else Mobile-IP cannot function. We made a small extension to the routing architecture that conceptually allows the following routes to be installed:

  ```
  to dest IP  via  gateway IP  via specified interface.
  ```

  Consequently the route(4) socket can now specify a particular interface and routing table entries can directly be bound to existing interfaces by routing daemons without a priori subnet bindings.

Other minor modifications exist. For example, we have an additional optional security mechanism called *Authenticated Ad Hoc* routing [2] that replaces ARP with a less-insecure mechanism aimed at making ARP-spoofing more difficult. The kernel is modified so that the mobile routing daemons use the ARP table, but ARP itself is disabled.

[3] However Linux allows such bindings. See the Linux route(8) and arp(8) commands.

3 Closely-Coupled 2.2.8 FreeBSD Mobile-IP/IPSEC

We will briefly discuss some architectural aspects of our older 2.2.8 BSD combined Mobile-IP/IPSEC architecture. Much of the Mobile-IP architecture itself has proved portable over time, but the IPSEC aspects themselves did not survive abandonment of the Naval Research Labs (NRL) IPSEC mechanisms, based on older RFC 1825 IPSEC. [4] We will briefly describe a new IPSEC mechanism based on close coupling of routes and IPSEC policy. We call this *closely-coupled* because the route daemons directly manipulate the IPSEC policy.

We created an experimental IPSEC policy system based on modification of the route(4) socket. [5] IPSEC assumes two abstract databases in the operating system, that can be used for cryptographic operations on packets. One may be called a *policy database* with rules similar to:

```
use IPSEC (tunnel/transport/ESP/AH), on these IP addresses,
with a certain security association (algorithms/keys)
```

Formally this database is called the *Security Policy Database* or *SPD*. The other database provides key material; for example, use BLOWFISH, 3DES, with certain key bitstrings, and is called the *Security Association Database* or *SAD*. Our routing socket modifications allowed routes in the routing table to act as the SPD. We assumed key material had a priori been loaded into the SAD. Thus the SPD references the SAD for actual key materials.

Logically the route(8) command could be assumed to have the following form:

```
# route <ipsec-mode> -spi <SPI> -itsrc <SA-ipaddr> -itdst <SA-ipaddr>
```

The *ipsec-mode* could be any of *-ah, -esp, -ahtunnel, -esptunnel* . The modes defined a particular route as either transport or tunnel mode IPSEC. When a route was loaded, either manually or by a mobile routing daemon, internally a search was performed in the kernel for the SAD, and if an appropriate binding was found, a pointer was setup between the routing table, and the SAD.

[4] The BSD operating system has chosen to adopt the version of IPSEC called KAME, developed in Japan. KAME/IPSEC is based on the RFC 2053 version of IPSEC. Of course, it has also never been burdened with US export law problems.

[5] The reader should understand that most conventional IPSEC implementations are based on rulesets similar to firewall access control lists.

The Mobile-IP/IPSEC routing table feature was used for a number of different security architectures. For example, we implemented the basic *MN to security gateway VPN* as seen in figure 1. The HA to MN routing path was dealt with by creating an ESP tunnel on the IPIP tunnel device, resulting in packets with a { IP ESP (IP datagram) } header structure. Other security features included HA to FA 1-way authenticated tunnels with a IP AH IP datagram structure as opposed to the conventional IPIP tunnel. Also our mobile-node daemon was capable of using a combined form of DHCP, ESP, and Mobile-IP, when no foreign agents were to be found. This system allows a mobile node to retain its invariant MN IP when away from its home IP address area. The use of the DHCP IP address as the COA meant that any possible IP ingress address problems were avoided because the COA address did not belong to the MN's home addressing domain (see, for example [1], and [5]). Thus, Mobile-IP enabled systems can wander away from their home security enclaves without having to worry about the IP source ingress filter problem.

4 Loosely-coupled 4.3 FreeBSD Mobile-IP/IPSEC

The current architecture, based on 4.3 FreeBSD, combines the KAME IPSEC implementation and the PSU Mobile-IP daemons. We have re-implemented the basic *MN to security gateway VPN* as seen in figure 1. We call this architecture *loosely-coupled* because the Mobile-IP daemons do not directly manipulate the IPSEC policy. In KAME, IPSEC policy is setup more on the lines of traditional layer 3 access control lists. We assume initial IPSEC two-way tunnels are setup between the Home Agent and Mobile Node, and then run Mobile-IP on top of that configuration. In this section, we will explain the implementation setup in detail, and discuss some resulting implementation problems and solutions.

From the high level point of view, as routing consists of two 1 way problems, we must deal with 2 problems, 1. MN to HA, and 2. HA to MN. For IP datagrams sent from the MN to the HA, we tunnel conventional IP datagrams from the MN to the HA. Thus the IP outer header has an IP src = MN IP, and an IP dst = HA IP. The ESP header encrypts the interior datagram sent from the MN to some other host. For the HA to MN path, we first have packets tunneled via an IPSEC tunnel (IP ESP, IP datagram), where the outer IP header has an IP src = HA IP, and IP dst = MN IP. This packet is then encapsulated inside an IPIP datagram that deals with the COA. Conceptually the HA to MN path can thus be viewed as (IP (dst=COA), IP ESP, IP datagram).

4.1 Architectural Details

The BSD KAME/IPSEC system allows three levels of kernel control (see ipsec(4)). Sysctl(8) can be used for global policy. The manual setkey(8) command is used to set IPSEC packet-filter defaults – which are similar to traditional layer 3 access control lists implemented in routers. In addition, setsockopt(2) can be used for setting per socket IPSEC policy attributes. Thus routing daemons could choose to avoid more general policy when warranted. We make no use of the sysctl mechanism and instead use a combination of the setsockopt(3) and setkey(8) mechanisms. Below, we will briefly illustrate how a two-way ESP tunnel is setup between a Home Agent and a MN, and then we will present the mechanisms necessary to allow Mobile-IP to work with IPSEC.

4.1.1 KAME IPSEC setup

IPSEC is setup before mobile daemons are booted. The setkey(8) command implements both SPD, and SAD (policy and keys). The SPD informs the kernel that ESP with tunnel mode should be applied to certain IP address pairs. The SAD is used for determination of key material. For example, given a certain SPD entry, use ESP with BLOWFISH-CBC, and a certain SPI, manual key, etc.

We show how to setup a sample HA for one MN. Assume our HA has IP address 10.0.0.1, and our MN has address 10.0.0.2. Before we boot mipd for HA functionality, we execute the following shellscript.

```
#!/bin/sh
#/etc/mipipsec.sh
# 1. ipsec policy for mobile node 10.0.0.2
setkey -c <<EOF
spdadd 0.0.0.0/0 10.0.0.2/32 any -P out ipsec
esp/tunnel/10.0.0.1-10.0.0.2/require;
spdadd 10.0.0.2/32 10.0.0.1/32 any -P in ipsec
esp/tunnel/10.0.0.2-10.0.0.1/require;
add 10.0.0.2 10.0.0.1 esp 7000 -E blowfish-cbc "beefdeadbeefdeadbeefde
ad";
add 10.0.0.1 10.0.0.2 esp 7001 -E blowfish-cbc "deadbeefdeadbeefdeadbe
ef";
EOF
```

```
# 2. entry for next MN
```

Again as routing is 2 1-way problems, there are logically 4 entries, with input and output SPD pairs first, followed by SAD key setup. The first SPD output command states that any packets (IP addr == 0.0.0.0) sent to the MN must use tunnel-mode ESP. This particular access control hook is very important, as it is the key to implementing the HA to MN part of the two-way ESP VPN. Policy for input packets from the MN is supplied as well. This policy assumes that the MN will be setting up the tunnel part of the VPN. Finally, the "add" lines bind the key material to the SPD entries. Of course, as this is manual mode IPSEC, we must make sure that the manual keys are kept as securely as possible on the hosts in question, and that they match between the MN and the HA.

Equal and opposite setup occurs at the MN before the mnd daemon itself is brought up. We will neglect setup details here except for the following important setup feature:

```
setkey -c <<EOF
spdadd 131.252.222.4/32 0.0.0.0/0 any -P out ipsec
esp/tunnel/131.252.222.4-131.252.222.3/require;
```

This means that all outbound packets are ESP tunneled to the HA. Note that the setup is equal but opposite to the HA setup. It is important to understand that KAME/IPSEC here is doing the work of tunneling packets to the HA; that is, it is doing the IP ESP tunnel work from MN to HA.

4.1.2 Mobile-IP interoperation

The basic MN to HA two-way VPN policy requires several modifications to the Mobile-IP daemon implementation.

First of all, as one possible security policy choice, we chose to make the necessary UDP and ICMP sockets, *bypass* any and all IPSEC packet mechanisms in the kernel. This is done using ipsec_set_policy(3), and setsockopt(2) calls, This means that all Mobile-IP packets bypass local IPSEC, and must rely upon their own devices for security. Remember that we choose to ignore Foreign Agents, thus it is important that we be able to talk to them and not assume we speak IPSEC with them. Further, by definition, we cannot share secrets with agents from another security domain. Hence we choose to let Mobile-IP as a protocol stand on its own, otherwise MNs would wrap Mobile-IP registration packets in ESP, FAs might not understand them, and thus could not relay them to the Home Agent.

The second implementation aspect is unfortunately far trickier. When a MN visits a FA, "all" packets in theory will be delivered via a HA tunnel encapsulation; that is, datagrams processed by the KAME IPSEC tunnel are formed as IP ESP { IP datagram }, with the outer IP src = MN IP, and the outer IP dst = HA IP. Unfortunately this runs full tilt into the BSD ARP table implementation. In the current BSD architecture, the ARP table is not separate from the routing table, and is implemented via a so-called clone route mechanism. When an interface uses ARP, and its IP address is configured, a clone route is placed in the routing table. For example, in the sample routing table below, we see a clone route (marked with the UC flags) that was loaded for a local ethernet interface when the interface was booted. One ARP table entry was instantianted in the routing table for local IP address 10.0.0.1, and later filled in by the ARP protocol itself with the MAC address of the local link host, 10.0.0.1.

```
host# netstat -rn

Destination         Gateway             Flags       Netif Expire
10.0.0.0/8           link#1              UC           0         0
10.0.0.1             0:d0:c0:5b:18:0    UHLW         4         3
```

This means that when a MN visits a Foreign Agent, and the first packet is sent via the IPSEC ESP tunnel from MN to HA, the outer IP header will of course, have IP dst = HA IP. This in turn, will cause the clone route to create an ARP table for the HA, because the MN after all, shares a local IP subnet association with the HA. Naturally since the HA is not nearby, this causes complete failure as no packets can reach the HA.

In order to fix this problem, we modified mnd to take advantage of the state machine. When configured for IPSEC, and in NOWHERE or AWAY states, it simply deletes all ARP table entries, and also deletes the clone route. When at home, the clone route is reinstalled. This is one possible policy choice, and the implementation might eventually allow more flexible configuration policies.

5 Architectural Analysis

In this section we wish to present an architectural analysis and briefly consider two questions: 1. what key ideas might be necessary in an operating system architecture to allow a combined Mobile-IP and IPSEC? and

2. What are the pros and cons of the two Mobile-IP+IPSEC architectural approaches, themselves?

We suggest that KAME IPSEC has provided us with two necessary features that we hope would be available with any IPSEC implementation. The first feature that was important is the ability to specify with the KAME packet filter mechanism that "all packets" should be sent over a tunnel to a tunnel endpoint. For example, the MN should be able to send "all" packets to the HA. It is hard to imagine that a IPSEC implementation would not have this capability, but it is fundamental and necessary.

The second important IPSEC capability is the ability to override higher-level "all packets must use IPSEC" packet filters on a per-socket basis. Without it, Mobile-IP registration packets could not be relayed by Foreign Agents that do not belong to the security domain. More generally, it is extremely reasonable for routing daemons using any routing protocol to be able to except themselves from a system-wide IPSEC policy. Most routing protocols have their own authentication mechanisms; for example, OSPF [11] has per link authentication.

We claim that our older 2.2.8-era BSD IPSEC/MIP experimental implementation was *strongly-coupled*, because the IPSEC policy was directly manipulated by the mobile routing daemons. On the other hand, the new 4.3 BSD KAME/IPSEC MIP is *loosely-coupled*. KAME IPSEC handles most of the IPSEC-based tunneling. We assume that the KAME IPSEC has been setup, and then run Mobile-IP which merely overrides any IPSEC policy in order to get Mobile-IP functions themselves accomplished.

So the bottom-line question then remains: which is better? It has been said in the past that any "packet filter" or access list mechanism vis-a-vis firewalls may be dangerous, because if the rule set is complex, it is easy to make mistakes. Our route-based mechanism however was probably more esoteric than any possible ACL mechanism. On the other hand, the bottom-line issue here may simply be portability. Our Mobile-IP implementation has been ported to Linux and makes a very few, reasonable assumptions about IP mobility features needed by an operating system. The old Mobile-IP/IPSEC implementation did not lend itself to simplicity or portability as it made complex assumptions about the host OS IPSEC and routing socket architectures. The newer system is much simpler.

6 Conclusions

In a narrow sense, we do not know of any related work, other than the comparison of our older version to our more modern version as previously presented in this paper. In a wider sense, we could consider competing link-layer, and network-layer systems that are somehow targeted at mobile security [6]

Such systems could include 802.11 WEP (link-layer), or other VPN systems like PPTP, which has been fairly well discredited by Bruce Schneier [14].

The critical question is this: *Why is IPSEC not chosen by default as the main vehicle for the delivery of end system to border gateway virtual private networks?* IPSEC has major virtues including: 1. it is not specific to any link-layer, and could be used for cellular telephony wireless, or over Ethernet for that matter. 2. it is not link-specific in terms of hop counts. It can easily be multi-hop across the Internet to a remote home security enclave. 3. it has been widely and opened reviewed in the IETF. and 4. over time, it will improve or at least keep up as it was designed for both replacement of its basic cryptographic algorithms, and key exchange algorithms. Thus it is more flexible than fragile algorithms like WEP.

It may be argued that from the layer 1 and layer 2 IEEE points of view, the IEEE cannot assume that IETF protocols are in use. What would be wrong then with doing nothing? KISS has its virtues and trying to put complex functions like security into "in-firm" firmware or hardware may be best left to layers 3 and above.

One might argue that combined IPSEC/Mobile-IP is not a good combination, because perhaps Mobile-IP is not a good idea. There are those who argue that Mobile-IP may perhaps be inefficient or have other problems. It is not our goal here to argue for or against Mobile-IP. One could just as well combine IPSEC with DHCP. DHCP could be authenticated itself, or perhaps protected by IPSEC in local security domains, and then IPSEC could take care of the two-way tunnels to and from a home security agent. Obviousally with DHCP, and unlike with Mobile-IP, IPSEC cannot take advantage of a fixed IP address as a index mechanism because DHCP IP addresses may vary over time or over link reattachments. However, IPSEC provides for this possibility with its dynamic key management protocol called IKE. According to the IPSEC Domain of Interpretation [13], one can simply setup two-way

[6] Application-level systems like secure-shell are excellent, but they do not necessarily cover other applications; for example, secure-shell won't do much for UDP-based video applications.

tunnels with IPSEC using dynamic keying and a fixed higher level name a la "user@dnsname", or according to the DOI document, *ID_USER_FQDN*. Again, there is no point in avoiding IPSEC.

7 Acknowledgements

We wish to thank everyone who worked on the Secure Mobile Network project, including Bill Trost, David Reeder, John McHugh, and Madhusudhan Jujare who more recently brought up the KAME/PSU Mobile-IP implementation as a student project. Much thanks to Suresh Singh for a review of the initial draft.

References

[1] J. Binkley, and J. Richardson, Security Considerations for Mobility and Firewalls, IETF draft, 1998, http://www.cs.pdx.edu/~jrb/jrb.papers/firewall/draft.txt.

[2] J. Binkley, and W. Trost, Authenticated Ad Hoc Routing at the Link Layer for Mobile Systems, *Wireless Networks*, Vol. 7, No. 2, pp. 139-145, 2001.

[3] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols", *Mobicom*, Dallas, October 1998, pp. 85-97

[4] N. Borisov, I. Goldberg, D. Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11", *In Proceedings of MobiCom 2001*, July 2001.

[5] P. Ferguson, and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", RFC 2267, IETF, January 1998.

[6] S. Fluhrer, I. Mantin, A. Shamir, "Weaknesses in the key scheduling algorithm of RC4". Eighth Annual Workshop on Selected Areas in Cryptography, August 2001.

[7] David B. Johnson and David A. Maltz. "Dynamic source routing in ad hoc wireless networks", In Tomasz Imielinski and Henry F. Korth, editors, *Mobile Computing*, pages 153–181. Kluwer Academic Publishing, 1996.

[8] KAME IPv6 and IPSEC project,
http://www.kame.net , Sept. 21, 2001.

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

[10] Local and Metropolitan Area Networks, IEEE. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard 802.11, 1999.

[11] J. Moy, "OSPF Version 2", RFC 2328, IETF, 1998.

[12] C. Perkins, "IP Mobility Support", RFC 2002, IETF, 1996.

[13] D. Piper, "The Internet IP Security Domain of Interpretation for ISAKMP", RFC 2407, IETF, 1998.

[14] B. Schneier, and Mudge. "Cryptanalysis of Microsoft's Point-to-Point Tunneling Protocol (PPTP)", *5th ACM Conference on Computer and Communications Security*, pp. 132-140, San Francisco, California, November 1998. ACM Press.

[15] Secure Mobile Networks project,
http://www.cs.pdx.edu/research/SMN , Sept. 21, 2001.

[16] K. Sklower, "A Tree-Based Packet Routing Table for Berkeley UNIX", *Proceedings of the 1991 Winter USENIX Technical Conference*, January 1991.

[17] A. Stubblefield, J. Ioannidis, A. Rubin, "Using the Fluhrer, Mantin, and Shamir Attack to Break WEP", ATT Labs Technical Report, TD-4ZCPZZ, Revision 2, August 21, 2001.

[18] T. Ylonen, "SSH - Secure Login Connections over the Internet", *USENIX Security Conference VI*, 1996, pp. 37-42.

[19] J. Zao, S. Kent, J. Gahm, G. Troxel, M. Condell, P. Helinek, N. Yuan, and I. Castineya. A Public-Key Based Secure Mobile-IP. *MobiCom97*, September 1997.