Secure Mobile Networking
Seventh Quarterly Report – Spring 1997

James R. Binkley
John McHugh
Bill Trost
Portland State University

1 July 1997

Project Status Overview

Our Spring work was focused on finishing a couple of key infrastructure projects and there are now two pieces we can point to that are complete. Bill Trost and Jim Binkley ported our entire code base from FreeBSD 2.1.0 to 2.2.1. We have for the most port upgraded roughly 26 agent and mobile node systems with a very few remaining exceptions. This was a tremendous task and we are over the hump at this point. In addition, David Reeder finished the kernel work so that our IPSEC/route binding that includes Mobile–IP tunnels is now complete and in 2.2. We can now build on that work to include more IPSEC functionality in our Mobile–IP implementation.

We have made progress in other areas but the work is not complete yet. Our two graduate students continued their work on redundancy issues. Papers being worked on include one by Sarah Mocas and Jim Binkley, and one by Sarah and Tom Schubert.

We will probably make at least one summer release of our IPSEC/Mobile–IP and possibly two (say in early and late summer, depending on when ISAKMP has been successfully integrated into our Mobile–IP system).

Since the previous winter report was so long, and this quarter has for the most part been spent on implementation as opposed to design, this report will be briefer than average. It will however include a short summary of ISAKMP work by Bill Trost, and a commentary by Jim Binkley on Mobile–IP and firewall issues.

Accomplishments for Spring 1997

1. Bill Trost and Jim Binkley ported our kernel sub–systems including Mobile–IP and IPSEC, WaveLAN drivers, affected routing apps (`route(8)`, `arp(8)`,

etc.), and Mobile–IP code from FreeBSD 2.1.0 to FreeBSD 2.2.1. The port is complete. We will touch on it briefly below. This version of FreeBSD will form the basis for subsequent releases by us of our IPSEC/Mobile–IP system.

2. our WaveLAN driver ports are becoming more widely used. Michael Smith, a core member of the FreeBSD team, ported our ISA WaveLAN driver into FreeBSD 2.2 and we were able to reuse that work in our port. Tatsumi Hosokawa has included our PCMCIA driver into his FreeBSD PCMCIA package. We also reused that code in our 2.2 port.

3. David Reeder has finished the kernel IPSEC/route binding that we have been working on for quite some time. This means that AH/ESP in any combination can be used with kernel routes including IPIP tunnel routes of special interest to Mobile–IP. The latter feature is apt to be unique to our implementation.

4. Our FreeBSD 2.2.1 system (at least for the kernel including WaveLAN drivers) has been internally released to BBN. We have given accounts on our internal server to Greg Troxel and Matt Condell of BBN. We know that they are in the process of internal deployment of a WaveLAN based radio network.

5. Tom Schubert and Sarah Mocas have continued efforts on formal analysis of Mobile–IP protocols. These activities have included:

   (a) developing SMV (model checker) specification of Mobile IP.

   (b) continued development of journal paper.

   (c) preparation/presentation of 5 minute talk for Oakland.

   (d) preparation of DIMACS abstract (submitted mid June).

6. John McHugh prepared a position paper on initial key establishment to guide discussion at the Memphis IETF meeting.

7. A paper entitled *Dynamic Network Reconfiguration Support for Mobile Computers* by Jon Walpole, Jon Inouye, and James Binkley was accepted for MibiCom'97. The abstract of the paper is as follows:

   > Hot swapping technology combined with pervasive heterogeneous networks empowers mobile laptop users to select the most appropriate device for their current environment. Unfortunately, the majority of system software remains customized for a particular network configuration, and assumes many characteristics associated with the network environment remain invariant over the runtime of the software. Mobility produces changes in the environment and nullifies many of these assumptions. This leads to severe loss in system functionality when resources are lost and failure to benefit when resources are gained.

This paper presents a model for device availability where assumptions about a network device are made explicit. When these assumptions are invalidated, cross–layer notifications trigger reconfiguration operations. Each layer adjusts in an intelligent manner, adapting to a new set of assumptions based on a user–configurable policy. An implementation of this model demonstrates its effect on a variety of applications run while migrating between different network environments.

Current Personnel and Projects

Jim Binkley will act as a half–time manager for Bill Trost, David Reeder, and lead the research of our two graduate students Bjorn Chambless, and Jennie Ye.

John McHugh will as usual edit reports, represent us at any necessary off–site meetings, and consult on related security issues.

Sarah Mocas is going to lead the research for a new graduate student (replacing Xu Hao) named Zhong Chen, who will be joining us this summer. Sarah will begin investigating the use of a simple (and new) public–key based authentication sub–system to be used with both Mobile–IP and ad hoc routing protocols. Sarah is working with Jim Binkley on a paper that will summarize our Mobile–IP security views. We hope to submit that paper to an upcoming IEEE Transactions on Software Engineering, special issue on Mobility and Network Aware Computing.

Sarah Mocas and Tom Schubert have submitted an abstract entitled "Formal Analysis of IP Layer Security" to the DIMACS Workshop on Design and Formal Verification of Security Protocols, to be held September 3–5, 1997 at the DIMACS Center, Rutgers University. Tom and Sarah have modeled different security mechanisms and protocols that are currently being developed for use at the IP network layer. This work has been done as an interplay between formal logics and mechanized systems. In the context of Mobile IP, they are currently using formal logics to examine mobile registration. Specifically, mobile registration requires the use of an untrusted route to establish a secure tunnel. We model the security association implicit in mobile registration.

Bill Trost (programmer) has been working with Jim on 2.2.1 porting and deployment, and is now focusing on the integration of ISAKMP into our route–based IPSEC security system. Both Bill and David Reeder attended the Memphis IETF meetings. Bill's principle area of focus in addition to kernel internals is integration of our Mobile–IP IPSEC architecture with public–key cryptography–based protocol systems.

David Reeder (programmer) has just completed the kernel work needed for our IPSEC/route bindings. David will now focus on the actual use of those IPSEC bindings in our Mobile–IP system. We intend to finish the implementation (started long ago) of a manual configuration system that would cause Home Agents and

Foreign Agents to automatically install authenticated IPIP tunnels as opposed to non–authenticated IPIP tunnels. David will also fix various Interoperability problems found at the Spring FTP Interop and these fixes will be available in our next Internet release.

Bjorn Chambless, a Master's Student, is working on the implementation of HARP or Home Address Redudancy Protocol, which we hope to finish this summer. He will be full–time employed this summer by the project. This quarter we discovered that although we had designed a reasonable and simple protocol, the integration of the protocol into our Mobile–IP agent daemon infrastructure was anything but simple. As a result, we are in the process of rearchitecting the state machine side of the Home Agent in order to make the integration of HARP easier. When we finish the HARP work, we will write a draft RFC on HARP and submit it to the IETF. As Bjorn will not graduate for awhile, we will probably put him to work on Foreign Agent redundancy eventually.

Jennie Ye, has implemented the Mobile Node side of the Multicast Ad hoc Routing Protocol this quarter, and is beginning to test the protocol itself. She will be full–time employed this summer by the project. Our goal for summer with Jennie is to try and finish at least the Mobile Node side of the protocol. Eventual milestones in this project include:

1. MN to MN multi–hop routing discovery
2. MN to agent
3. integration of the protocol with Mobile–IP
4. integration of this protocol with the previous ad hoc beacon protocol to form one demand based (as opposed to broadcast based) protocol.
5. a written technical report on the protocol.

We expect this small sub–project to continue through next year.

Zhong Chen is a 1st year M.S. student at PSU and has considerable experience with TCP/IP, which he has taught in the past.

Future Release Deliverables

We have essentially just made an internal release of our combined FreeBSD 2.2.1 and IPSEC/Mobile–IP system that includes 2–way DES tunnels (HA/MN). During the early part of the summer we will issue DES keys to our laptop users in order to stress this system and in order to see what actually happens when a number of users simultaneously use DES with a single Home Agent.

We are looking at probably two releases to the Internet over the summer and early fall depending on the exact system deliverables we wish to include. We suspect that an early release of a combined Mobile–IP/IPSEC would be a good idea simply to get it out. On the other hand, we want to eventually include ISAKMP and the HARP protocol, but it is hard to predict when those efforts will be done.

We are currently working towards a early summer release that will minimally have the following attributes:

1. Our IPSEC and Mobile–IP systems including WaveLAN drivers ported to FreeBSD 2.2.1

2. The now completed kernel route binding for IPSEC. This includes `arp(8)` and `route(8)` commands and various IPSEC combinations for routes including for example:

   (a) `route dst gateway -ah -spi ...`

   (b) `route dst gateway -esp -spi`

   (c) `route -ahtunnel dst gateway -ah -spi`

   (d) `route -esptunnel dst gateway -ah -spi`

   In other words, ordinary routes can have either ah/esp bound to them, and tunnel routes can have both those bindings and two more (-ahtunnel and -esptunnel) that apply to the appended COA outer IP header itself. Thus those bindings can be applied say to packets sent from a Home Agent to a Foreign Agent (or Mobile Node using a DHCP acquired COA).

3. A manually configured system that allows secure tunnels to be setup between Home Agent and Foreign Agents. Policy is expressed in the agent configuration file, and the routes between HA and FA are automatically installed with IPSEC bindings when needed.

4. Bug fixes in our Mobile–IP daemon system based on the FTP Interop experience.

Note that for any release we will have to cooperate with MIT again and rerelease the cryptographic part of the IPSEC code (/sys/netsec) as it has changed due to bug fixes and porting.

Later on when we finish ISAKMP work (i.e., change the kernel and daemons so that we can dynamically bind security associations to routes used in Mobile–IP) we will make a release that includes it. We would also hope to include the HARP protocol which should be done by the end of the summer and an integrated IPSEC/ARP mechanism.

One stumbling block that we hope to fix over the summer is that our Mobile–IP implementation changes routes by executing the `route(8)` and `arp(8)` binaries. This is not good in that it has always made error logging more difficult, and has certainly not made hand offs efficient. We intend to fix this problem over the summer and replace the Mobile–IP routing module with one that uses the `route(4)` routing socket directly. This will allow us to finally integrate our IPSEC/routing as used in Mobile–IP with traditional ARP as opposed to our current authenticated ad hoc beaconing protocol. In order to do IPSEC with ARP, we need to be able to detect when the kernel installs a low level link layer route. We will be able to do this by binding a rarely used BSD routing flag (XRESOLVE), which causes `route(4)` messages to be sent to daemons that have a route socket open. The per interface "clone" route (which is bound to a subnet) causes incoming ARP requests to create new link layer per IP destination routes. ARP itself fills in the MAC address as the route gateway. Since the clone route will have an XRESOLVE flag bound to it, each new instance of a ARP route will cause a message to be sent to the routing daemons. We will also bind a BLACKHOLE flag to initially created routes which will prevent packets from exiting the system until a security policy lookup and binding takes place. Thus our daemons can learn about ARP requests and bind IPSEC attributes safely to different link routes in a policy–based fashion. We hope to get IPSEC/ARP functionality into our late summer release.

Public–Key Cryptographic Work

Given that there were a number of possible avenues to go down in this area including ISAKMP, possible integration of BBN's MOIPS, and Fortezza, we decided to prioritize on getting ISAKMP integrated into our Mobile–IP architecture and system first. Once we have done that, we will reconsider what to do about Fortezza and MOIPS (although possibly BBN might adapt our Mobile–IP and do that work for us). Bill Trost has written a short report on his ISAKMP work which will follow in the next section.

John McHugh, Jim Binkley, and Sarah Mocas have decided to use one new graduate student and attempt a small–scale authentication protocol that would simply use digital signatures. Sarah Mocas will lead this work and Jim and John will advise as needed. Given that BBN's protocol uses a session–key based approach, we thought it would be interesting to attempt something different and simple that would use signatures for authentication (e.g., RSA or DSA). Our goal would be to design a Tag–Length–Value based scheme that could easily replace our MD5–based authentication mechanisms used in both Mobile–IP registration and ad hoc routing protocols. The same scheme would be used in both cases. All these protocols do after all, is establish routing. ISAKMP would then be used once routing was established for all higher–level protocol security. In essence, we would have a two–tier security system (like our current ad hoc + static IPSEC system), but public–key cryptography would be available in both cases.

Such a protocol should minimize the number of messages needed and the size of the packets. Hence we would simply sign the packets, but not include certificates.

(This is in accord with BBN's thinking in MOIPS). We have briefly investigated the speed of signed MD5 hashes and think that a signature based mechanism is not unreasonable given the infrequency of registration packets and the current (and ever increasing) speeds of CPUs. In the first cut, we might simply use a flat file lookup system for public keys. Eventually such a system can be replaced with a certificate lookup system.

As an upper bound, http://www.rsa.com/rsa/developers/bench.htm, the RSA web page claims the following speeds for their proprietary BSAFE implementation for signing and verification on a Pentium–90 system (which is slow by currently available Pentium standards. We have several laptops that are faster).

512 bit keys	**seconds**	**ops/sec**
RSA/P90/sign	.024	41
RSA/P90/verify	.0027	370
DSA/P90/sign	.029	34
DSA/P90/verify	.052	15

These numbers suggest that signed packets when sent with the infrequency of current Mobile–IP registrations are computationally feasible. In our implementation, Mobile–IP nodes reregister on the order of once per minute. The more important number is probably the verification speed. This is because a given agent (especially a Home Agent) may be called upon to verify many Mobile Nodes. The scalability of verification at a Home Agent should be a concern. Of course, we have not yet factored in any costs that might be caused by a higher level certificate lookup system (perhaps available to the HA via DNS). It may be useful for us down the road to consider whether or not the HARP protocol can be modified to allow distributed authentication amongst a cluster of Home Agents. As a secondary consideration, verification speeds are probably also important for our new generation multi–hop ad hoc routing protocol. In our protocol, intermediate Mobile Nodes would have to both sign and verify multicast packets if the intermediate system were to modify the packet (e.g., appends its own path tuple).

We will have a design to present for this sub–protocol in the summer report.

ISAKMP (Bill Trost)

We have been working with the implementation of the Internet Security Association and Key Management Protocol (ISAKMP) and the Oakley protocol that Cisco systems is distributing. This implementation, called ikmpd, supports Oakley's Main Mode and Aggressive Mode exchanges for setting up the key management protocol, and Quick Mode exchanges for creating subsidiary security associations for use with data traffic. Under some circumstances, ikmpd will also use Informational Exchanges to let another party know when there is a problem with the creation of a security association. It may use DSS signatures, RSA encryption, or preshared keys to create the initial security association. The implementation does not provide for the expiration of security associations, but this is more a result of limitations of the IPSEC implementation being used than of Cisco's implementation of ISAKMP.

Cisco's most recent distribution of their ISAKMP implementation has a variety of problems. As distributed, the program is not even usable, as it has IP addresses hard–coded into the program in various places, and the program does not attempt to perform a DSS exchange, even if that is the only keying information available. When creating a security association in Quick Mode, the distributed program would use the IPSEC algorithm identifiers when instantiating the security association in the kernel, instead of using the operating system's algorithm identifiers.

Once these simple problems have been fixed, the program is still very brittle. Missing RSA keys cause the program to transmit malformed ISAKMP packets, and failure to create a security association sometimes induces fatal memory errors in ikmpd.

In addition, our research environment has uncovered a problem that is relatively unlikely in a workstation environment. Many of our computers have two network interfaces, one wireless to provide support for the mobile laptops, and an ethernet interface to provide Internet connectivity for the computer and the laptops it serves. If a laptop attempts to establish a security association with the wired interface of a computer it was using for its Internet connectivity, responses from the wired computer would be transmitted using its wireless address. A Mobile Node, unable to determine that the wireless address corresponds to the same computer that it is trying to establish a security association with, discards the response it received and retransmits requests to the other computer's wired address, until it eventually abandons the attempt to establish the security association because it has seen no responses.

We have submitted fixes to these problems to Cisco, and the implementors there welcomed our suggestions and have indicated that our fixes will be included in the next release. In the case of the problem with multi–homed hosts, the solution we provided is to have ikmpd use the hostname assigned to the (agent) system when it starts up for all ISAKMP transmissions. This solution is adequate for our environment but will not work for computers which have multiple addresses assigned to the same name.

Cisco's implementation of ISAKMP is surprisingly slow. Profiling analysis indicated that the program spends a large amount of time generating random numbers for use in the protocol. FreeBSD provides a random number generator in the operating system based on keyboard activity and other interrupts received by the system. We modified ikmpd to use this source of random numbers should it be available. This change dramatically improved the program's performance. As Cisco does not use FreeBSD, they will be unable to take advantage of this improvement, although they may choose to distribute it.

The purpose of our work with ISAKMP and Cisco's implementation thereof is to integrate its dynamic, public key session creation mechanism with our changes to allow IPSEC security associations to be attached to routes. When this work is complete, a security association can be attached to a route using a special "wildcard" security association. This special security association will tell the operating system that it must request the key management service to negotiate and install a security association, which will then be used for all traffic on the route. Most of the changes to support the attachment of dynamically–negotiated security associations to routes must be made in the operating system itself; only one small change to ikmpd will be necessary. We hope to design and finish a ISAKMP route binding work this summer. Our spring work has helped us gain familiarization with the Cisco implementation.

Considerations on the Mobile–IP Firewall Problem

The Mobile–IP working group has spent some time attempting to deal with the perceived problem that current Internet border router filtering schemes may prevent Mobile–IP from working in a inter–routing domain context. There are security problems with the solution the working group has proposed. We believe we have a solution, and hope to pursue research and implementation in the next year that will demonstrate that the problem can be solved. To some extent, we feel we have already demonstrated a working solution to the problem.

The Mobile–IP "ingress filter" problem is this: CERT advisories suggested that routing filters should be installed that would prevent packets with IP source addresses that should be internal to a site, from crossing from the outside world into the site. The advisories were prompted by attacks on protocols like X, BSD lpr, NFS, BSD rsh and the like which have weak authentication systems based only on IP addresses. More recently an advisory was issued that suggested that packets that leave a routing domain possessing IP source addresses that are only externally valid should also be blocked. This was done due to the well–known TCP syn attack problem. A filter of this type may be called an "egress filter".

Both ingress and egress filtering will cause problems for Mobile–IP. By definition, Mobile–IP systems retain their "internal" IP source addresses. (Note that this does not apply to Mobile–IP itself as a registration protocol since Foreign Agents forward Mobile–IP packets as proxy systems; i.e., the UDP registration packet will have the FA's "outside" IP source address and hence that packet will not be filtered out.) Ingress filtering means that a Mobile–IP system in an external domain will not be able to talk to systems at home. The "egress filtering" problem is actually more serious in its consequences to a Mobile–IP "visitor". An "egress filter" would prevent the Mobile–IP system from talking to any external systems, at home or anywhere on the Internet.

The recently proposed Mobile–IP solution is that agents must support "reverse–tunneling" upon the request of a Mobile Node. Normal Mobile–IP tunnels might be called "forward–tunnels" and would apply to packets forwarded from a Home Agent to a Foreign Agent. "Reverse tunneling" means packets that are forwarded from the Mobile Node (or possibly Foreign Agent) back to the Mobile Node's home routing domain. IPIP (proto 4) would be used in either case.

The idea that a Foreign Agent might reverse tunnel packets back to a routing domain is not a good idea for security reasons. In our current PSU IPSEC/Mobile–IP implementation, we are capable of using IPSEC to reverse tunnel home directly from the MN to (and back) from the HA. We make sure that all (non Mobile IP registration) packets that go through a FA are encrypted. Hence the FA can only route and cannot "peek" at the MN's traffic. From a high level point of view, we believe that a Mobile Node should minimize its reliance on the FA and maximize self–reliance. A Mobile Node ideally should minimize security exposure and associations with external "unknown" hosts. It can (and must) always rely on its Home Agent, since all its traffic traverses the Home Agent. Hence we simply tunnel over the Foreign Agent using IPSEC.

Further, we have pointed out to the working group that such FA–based tunneling is not a solution since it simply violates the spirit of the CERT based filters. IPIP tunnels are a way to sneak in over the top of the CERT routing filters. IPIP itself as a protocol can easily be blocked (IP protocol 4) and may eventually be blocked on a wide basis if Mobile–IP becomes widespread. Indeed, in terms of current firewall infrastructure (cite Chapman) IPIP must be blocked in order to prevent a new mechanism for attacks via Mobile–IP agents on NFS and other protocols that rely on IP address–based authentication. It is hard to see a solution to this problem that does not involve strong cryptographic based solutions like IPSEC.

We have tested the ingress filtering problem by placing a Foreign Agent at Oregon Graduate Institute, which is external to us in routing terms, and where there has been at least one NorthWestNet border router with CERT ingress filters installed. The problem is real. Without extraordinary measures, a visiting PSU Mobile Node cannot talk to home.

We found that we could however set up Mobile Node based "reverse" tunnels and tunnel through our Home Agent at PSU to selected hosts. We combined IPSEC–based routing tunnels (using ESP/DES) with an IP src address stolen from the local OGI Foreign Agent. The packets sent to the Home Agent had the structure <IP><IP><ESP>, where the outer IP header used the FA's IP source address. These packets could cross the border router because the FA's IP source address was internal to OGI. The security association on the inner IP header was between the MN's "home" address and the Home Agent. Hence we will claim we were able to securely send packets home.

The notion of allowing a MN to borrow the FA's local address (as a COA) is a hack, and although it does seem to work, is not the best network engineering. The Mobile Node COA should be unique for a number of reasons; e.g., the IP identity field on the outer header should be unique per system and there might be clash if the same COA was used causing fragmentation to fail. However objecting to this mechanism does not disprove our solution, since FA's could easily be combined with DHCP or some other mechanism to give out unique local–link IP addresses (COAs) to visiting Mobile Nodes. During our testing we demonstrated that more than one Mobile Node could simultaneously use such a FA address and setup IPSEC/ESP host routes back to specific hosts and networks at PSU. Our current implementation is not elegant, but it allows us to suggest a more reasonable solution.

Our suggested solution has three fundamental principles:

1. When abroad, MNs should be able to acquire unique COAs (care–of addresses; i.e., local link IP addresses). It does not matter if an MN is at a Foreign Agent or not. We will call this the "COA acquisition" principle.

2. IPIP is a routing mechanism that allows tunneling across or through current CERT ingress filters. It must be made secure by the use of IPSEC. We will call this: "IPIP needs IPSEC".

3. IPIP itself must be blocked at firewalls for the foreseeable future. We plan to initiate such blockage on our own PSU engineering (and OGI) border routers within the next year.

COA acquisition

Mobile–IP has tended to view MNs at foreign links as possibly having two completely different operational modes. MNs may forward through Foreign Agents or they may act as their own FA if they can acquire a local COA (for example, via DHCP). These modes are not really mutually exclusive and there is no reason they cannot be combined.

From a high–level routing point of view, the most important idea in Mobile–IP is that MNs, when away, prepend a COA IP address to their conventional "home" IP address. The MN must inform the Home Agent, which will tunnel packets for the inner address to the outer address. Where the COA comes from is immaterial. Hence we believe that Mobile Nodes can fundamentally solve the "ingress filtering" problem by simply acquiring a COA and making sure that packets sent abroad (and possibly routed through home on the way) leave with a reasonable COA.

The real question here is what routing implementation model is needed in terms of the Mobile Node's own operating system stack. As pointed out by cite Cheshire 4x4 routing paper, Mobile Node routing flexibility is needed and may even require user intervention in some cases. Our "theft" of the FA's COA works. Making DHCP available for "visitors" is a better solution. Teaching Mobile Nodes how to juggle both local COAs and their real "permanent" IP address (along with combining it with IPSEC) in terms of a given operating system routing implementation is the true challenge.

IPIP security issues

Mobile–IP might skirt the “ingress filtering” problem in a trivial way anyway since a site that has a firewall may well have a perimeter defense system and agents that have external access could easily be bastion hosts (be in the “no man’s land” between the external firewall and the internal harder firewall). This would solve the egress problem for visitors. Conventional site security administration would probably forbid same site Mobile Nodes from talking to systems within a perimeter defense anyway (unless strong cryptography is used, say with SSL, ssh, IPSEC, one time passwords, or the Microsoft PPTP tunneling scheme). The ingress problem is solved by insisting that all foreign hosts (including Mobile Nodes) talk only to bastion hosts and/or use protocols that involve strong cryptography. It is not clear that new firewall technology just for Mobile–IP is needed. Current firewall techniques (bastion hosts and carefully designed filters) can be used to make Mobile–IP safe for all. Frankly, it has never been clear that there would be that much inter–routing domain Mobile–IP anyway. Same domain use is much more likely.

IPIP itself is dangerous from a security point of view and should be blocked unless combined with IPSEC. We have implemented a simple filter in our agents that can be turned on when packets are detunneled. The filter asserts that IPIP packets must either already have successfully passed a IPSEC based security association check with the agent or have a security association yet to come in the packet header with the same agent. In effect, this means the sender must have a IPSEC security association with the agent. Packets might have either an <IP><IPSEC><IP> structure or a <IP><IP><IPSEC> structure. The former case could be of possible use between Home Agents and Foreign Agents for packets sent to remote Mobile Nodes. We are currently in the process of creating a manual–key based IPSEC security relationship system for Mobile IP that only applies to HA to FA packets. The latter case (<IP><IP><IPSEC>) applies to packets sent from Mobile Nodes (bypassing the FA) to the Home Agent, when the MN is away from home and is something our current implementation can do. In both cases, IPIP turns into IPIP + IPSEC and can hence be rendered safe for Internet use. Either system could eventually use public key cryptography and certificates for dynamic agent and Mobile Node security relationships.

Formal Methods for Protocol Analysis

To reason about Mobile-IP, a simple SMV specification is being developed that does not utilize IPSEC mechanisms. The specification describes agent discovery, registration and HA-FA tunneling. I hope (not) to find attacks by adding an adversary to the specification and modifying the specification to include the IPSEC mechanisms. I will then attempt to extend the specification to include ad hoc routing and redundancy mechanisms (use of multiple FAs, HARP).

After developing a simple SMV model of mobile registration, it was determined that the simplicity of SMV's specification language was not adequately expressive to define general routing structures (in particular, SMV does not provide arrays). A brief examination of the Murphi checker suggests that it might be a more appropriate tool. However, the focus of this summer's activity will be on further developing a model in the HOL90 theorem prover.

The abstract submitted to the DIMACS workshop appears below.

Formal Analysis of IP Layer Security

Sarah Mocas and Tom Schubert
{sarah,schubert }@cs.pdx.edu
Department of Computer Science
Portland State University

The use of formal methods during the development of a security protocol permits the designer to identify subtle assumptions that can lead to security flaws. Automated support for using formal methods can greatly assist this process, however, for many applications, correct formal modeling of a protocol and identification of its underlying assumptions is initially of greater importance than automated proofs. With this in mind, we have modeled several different security mechanisms and protocols that are currently being developed for use at the IP network layer. This work has been done using an interplay between modeling protocols via formal logics and embedding protocols in a mechanized system.

The use of informal modeling has motivated the development of a mechanized system. We develop a framework to formally reason about cryptographic protocols using belief logics with the HOL theorem proving system. The prover allows identification of the sequence of inferences used to conclude why a property about a protocol does or does not hold. The automated procedure informs the user what rules are relevant and when rules are applied to a goal.

Using formalizations, we have examined key escrow protocols[4] and proposed IP security mechanisms. Current work pursues the analysis of secure mobile networks.

We have modeled several of the proposed IPSEC security extensions to determine the soundness of these propositions. Specifically, we examine the services that should be provided by the addition to an IP packet of the authentication header, AH, and the privacy header, ESP (encapsulation security payload). Through formal methods, we show where protocols using "secure packets" succeed in providing the service for which they are intended and where they fall short.

In the context of mobile IP, we are investigating the use of formal logics to examine IP layer security mechanisms. There are several security mechanisms needed for a reasonable implementation of a secure mobile network. Initially, mobile registration requires the use of an untrusted route to establish a secure tunnel. We model the security association implicit in mobile registration.

Belief logics are designed to consider what conclusions individual parties (principals) in a communication dialog can deduce based on messages received and a set of initial assumptions or beliefs. Belief analysis attempts to show that only desired properties are guaranteed by the communication (data confidentiality, message authorship, non-repudiation, no replayed transactions, etc). Note that proofs about idealized protocols are not a guarantee that the concrete protocols are correct. There are many implementation assumptions that if invalid, would cause a secure, idealized protocol to actually be insecure. For example, these logics all assume that the crypto algorithm is secure. Several logics for analyzing cryptographic protocols and authentication schemes have been proposed (see for example [1, 2, 3, 5, 6]).

To better support the use of mechanized belief logics, we have developed a general infrastructure for creating automated and interactive procedures to prove goals about inductively defined relations. The infrastructure provides tools to build interactive functions and goal-directed support functions, procedures to eliminate existentially quantified variables from terms, and tactics to generalize existentially quantified rules. Also provided is a tactic generating function that returns a tactic specialized for a list of inference rules and a list of combinators. The tactic searches for and then applies, an appropriate rule to apply based on the current goal. In practice, this tactic appears to provide a significant improvement over tactics that try every rule. The tactic also outputs what rules where chosen to apply in a given situation. The displayed list of decisions can assist the user understand why a proof is correct and often, why it fails.

References

[1] M. Abadi and M. Tuttle. A semantics for a logic of authentication. In *Proceedings of the Tenth ACM Symposium on Principles of Distributed Computing*, pages 201–216. ACM Press, 1991.

[2] Michael Burrows, Martin Abadi, and Roger Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8(1), Feb. 1990.

[3] L. Gong, R. Needham, and R. Yahalom. Reasoning about belief in cryptographic protocols. In *Proceedings of the 1990 IEEE Computer Society Symposium on Research in Security and Privacy.* IEEE Computer Society Press, 1990.

[4] Tom Schubert and Sarah Mocas. A mechanized logic for secure key escrow protocol verification. *International Workshop on the HOL Theorem Proving System and its Applications*, September 1995.

[5] Paul F. Syverson and Paul C. van Oorschot. On unifying some cryptographic protocol logics. In *Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy*, May 1994.

[6] P. vanOorschot. Extending cryptographic logics of cryptographic logics of belief to key agreement protocols. In *Proceedings of the First ACM Conference on Computers and Communications Security*, pages 232–243, 1993.