Secure Mobile Networking
12th Quarterly Report*

James R. Binkley and John McHugh
Portland State University

4 December 1998

Project Status Overview

At this point our project has wound down. As a consequence this report will be short and in the form of a status report.

During Summer 98 Jim Binkley (faculty) did some pro bono project work and Bjorn Chambless (our last graduate student) remained employed as a M.S. level research assistant. We will briefly summarize our accomplishments for the summer in order of importance. The major accomplishment was our final source release which included student work from the previous year.

Status Items

Jim Binkley and Bjorn Chambless combined to produce our final major milestone, i.e., what we call the summer 1998 Secure Mobile Networks software release. This release is available on our project web page: http://www.cs.pdx.edu/research/SMN as the file mip-summer98.tar.gz. Please note that we reused the export-controlled bits at MIT, thus we did not have to re-release the kernel /netsec directory.

The release included the final MADRP, Multicast Adhoc Demand Routing Protocol work done by Jennifer Ye who graduated at the end of winter 1997/1998. It also included code done by Zhong Chen who made our Mobile-IP daemons work with DHCP as well with our simple IPSEC "away" policy scheme; i.e., we could combine DHCP/IP¡IPSEC¿IP tunnels so that a mobile node when away could tunnel home. This is one part of a "solution" for VPN-line mobile systems accessing home over anti IP-spoofing firewalls. Bjorn Chambless finished the Home Agent Redundancy Protocol and it was part of the release. More information on the release can be found in the summer release README

*Supported by th U.S. Department of Defense Advanced Research Projects Agency and the U.S. Air Force Rome Laboratory under contract F30602-95-1-0046.

file ftp://ftp.cs.pdx.edu/pub/mobile/mip-summer98.README. A copy of this appears as appendix A.

Wireless Network Real-Time Monitoring

Jim and Bjorn collaborated to deploy various server agents in order to instrument our wireless agent infrastructure. The infrastructure itself is now established in four buildings in the School of Engineering and Applied Sciences at PSU. It has a modest number of academic and staff users including system and network administrative staff as well as professors and graduate students in Computer Science and Electrical Engineering. The home page for the Seas Wireless Network can be found at:

http://guinness.cs.pdx.edu

The first link: "SEAS building agent map" shows currently deployed agents in the form of a very rough campus map (and is lacking information about a newly instrumented building).

The second link: "Wireless Agents in SEAS/Availablity - (via) Big Brother" shows the wireless agent infrastructure in terms of whether or not the mobile routers are operational. It uses the public domain system called Big Brother which displays graphical information about various system components on the web in "near" real-time. For example, if a system goes down, email will be sent to the adminstrators (Jim and Bjorn) in about five minutes time. Big Brother uses ping, disk checks, CPU process checks and other tests including testing the status of certain ASCII-oriented network servers like SMTP/sendmail. In our case, all our agents run snmpd and our mobile-ip daemon. The "procs" sub-field checks for the existance of those servers so we will be notified if they should crash. Big Brother is one of the two public domain applications we have installed across all of our agents (the other being the UCD snmp daemon).

The third link: "MRTG/SNMP - Network Traffic Display for SEAS Wireless Agents ...". Here we use the highly popular MRTG "Multi Router Traffic Grapher" tool which is essentially an SNMP-based system to monitor a few aspects of our more critical agents. We essentially use it to graph bytes in and bytes out on wireless and ethernet interfaces. MRTG[1] is a general tool and can actually be used to graph script-based applications as well as SNMP derived data variables. We have one link that is an attempt to graph our current Mobile-IP mobile node user count[2].

Linux Port

As a side project, Bjorn Chambless ported our mobile-node specific routing daemon code (not agents, just Mobile Nodes) to linux. He has made a release of this code on our home page. We are finding that many faculty members and students have linux on laptops and they may be able to make use of this code to join our Mobile-IP based wireless network. We have three candidate faculty members lined up and are in the process of making them "mobile".

[1](See http://ee-staff.ethz.ch/ oetiker/webtools/mrtg/mrtg.html)

[2]See http://guinness.cs.pdx.edu/mrtg/guinness.mip.html

Policy Work

As the work has come to a conclusion, we have spent a considerable amount of effort thinking about the policy implications of mobility. A journal publication is in progress, and several presentations have been given or are planned for the near future. Past presentations by John McHugh included ones at the Naval Postgraduate School in August and at Intel and Oregon State in November. A presentation at the CERT is planned for December.

Jim Binkley and John Richardson (of Intel) have prepared an IETF draft entitled "Security Considerations for Mobility and Firewalls" which appears as appendix B.

Just For Fun

As a demonstration of the mobile-IP linux laptop capability, we have hooked up a "quickcam" parallel-port based color camera to a linux laptop and have created a laptop/camera system that can be easily deployed anywhere within our wireless network infrastructure. We call the camera the "bjorncam"[3]. Of course, Bjorn may not actually be available at any given time. His bjorncam page records a few of his nearby mobile journeys and a few people encountered on his trips.

[3] The official homepage for the bjorncam is: http://bjorncam.cs.pdx.edu/bjorncam.

A Summer Release Information

```
A Mobile-IP Implementation based on rfc 2002
--------------------------------------------

Summer '98 Release for FreeBSD 2.2.1.

Jim Binkley,
John McHugh,
Sarah Mocas,
David Reeder,
Bill Trost,
Zhong Chen,
Bjorn Chambless,
Jennifer Ye

Portland State University
Computer Science Dept.  - Summer, 1998

email: mobileip@guinness.cs.pdx.edu

---------------------------------------------------------------
Introduction

Please see the file README for the main documentation.  This
README.summer98 file only covers new features in the latest release.
Thus README.summer98 is a supplement.  README discusses how
to install the basic release.

In general the work here represents student work done on
the Secure Mobile Network project by M.S. students at PSU.
This release represents a feature upgrade for our release of summer 97.
It mostly adds only functionality, some of it experimental,
to the previous integrated IPSEC-Mobile-IP system.
A lot of the work here uses architectural features resident
in the code from summer 97 and is built on top of it.

The contents of the original README are still important
and it should be read first.  Then read this README.summer97
for more information.  this release is still based on the
FreeBSD 2.2.1 kernel, but we hope to make a minor release
based on the latest FreeBSD kernel RSN.

New features for summer 98 include master degree student
work and some features added in late summer 98.  The main
features which will be discussed briefly here are:
```

0. an internal improvement in mnd and mipd so that they use the route(4) socket as opposed to route(8) and arp(8) calls. Thus they are much more efficient. We still supply modified route(8) and arp(8) binaries and source, as they can be used to manually setup Virtual Private Network connections along with IPSEC. Given that these improvements are internal, there will be no further documentation on this particular feature.

1. Jennifer Ye implemented a network-layer experimental ad hoc routing protocol called MADRP, for "Multicast Adhoc Demand Routing Protocol". The code is included in mnd and mipd but it is NOT compiled in by default (#ifdef MADRP). Functionally MADRP assumes that the previous ADHOC mode (ADHOC version ARP) is used for link-layer mobile communication. It allows Mobile Nodes that are separate from either Mobile-IP agents or other Mobile Nodes to set up host routes between them; i.e., packets are automatically forwarded across Mobile Nodes themselves. Connectivity between MNs and agents is also established; hence if a Mobile Node is separated from the wired infrastructure by other Mobile Nodes it can still find a routing path to Mobile-IP agents (and its own HA). See the MADRP section below for more information.

2. Zhong Chen has added DHCP capability for Mobile Nodes which may or may not be combined with the pre-existing IPSEC tunnel functionality between Mobile Node and Home Agent. DHCP is basically viewed as another link mode. We assume that DHCP servers are available on a link (and that ARP is used ...), thus Foreign Agents are not available on a link. This feature applies only to the mnd daemon. See the DHCP section below for more information.

3. IPSEC integration with Mobile-IP code. There are two features worthy of note. IPSEC AH and ESP may be used simultaneously for the route binding. Also IPSEC AH may be used with manual keys between Home Agents and "same security domain" Foreign Agents; i.e., instead of IPIP tunnels between HAs and FAs, one may use IPAHIP tunnels. There will be a brief discussion of this feature set below.

4. Bjorn Chambless (PSU CS/MS student) has implemented and released HARP, the Home Agent Redundancy Protocol. HARP allows for Mobile-IP Home Agent redundancy; i.e., one can have TWO Home Agents without the rest of the Mobile-IP systems (MNs/FAs) having to change or being aware of the co-HAs. The co-HAs cause no changes in the MIP protocol itself. They use the HARP protocol between themselves to communicate MIP routing information (where MNs are registered). See the HARP section below for a high level description, and

configuration information.

Please note that for the supplied binaries and the source in mip.src, HARP, and DHCP are compiled in and turned on. The source for MADRP is included, but MADRP is experimental and the Makefile in mip.src has MADRP left out at this point. The MADRP code is not in the supplied binaries but of course they can be recompiled if you wish to experiment with it.

It should be possible to modify the Makefile and leave those #ifdef values in/out to produce binaries that have or do not have that functionality. Minimally we strongly urge that HARP and DHCP be left in the feature sets however. MADRP is interesting but is experimental.

----

1. MADRP - Jennifer Ye (programmer) and Xu Hao (designer)

MADRP is an experimental high-level ad hoc routing protocol. It may be viewed as an Interior Gateway Protocol. It allows Mobile Nodes to route packets over other Mobile Nodes which act as routers. It is very roughly based on or inspired by work done by Dave Johnson and Scott Corson in the distant past. MADRP contains certain design elements though of interest that include:

1. integration with Mobile-IP; i.e., Mobile Nodes that are more than one hop away from an agent can still use Mobile-IP to retain Internet connectivity.

2. the subnet-less nature of the previous link-layer ad hoc mechanism is still used; i.e., cooperating MNs do not assume they can only talk to other MNs of the same subnet.

3. MADRP hosts set up end to end host routes to other hosts they are talking to. Given that our IPSEC implementation is tied to routes in the routing table, this allows us to do end to end IPSEC between two participating laptops in so-called "tunnel mode".

Given that the Mobile Nodes are not gateways, such IPSEC tunnels are not subject to proposed plaintext attacks. For good or bad, all of the packets sent between the hosts would be covered by the same IPSEC security association. This is an interesting but experimental notion.

Note that MADRP assumes our link-level ad hoc scheme. It does not work with ARP. By definition, it uses any kernel routing mechanism involved with the link-level ad hoc scheme on which it is built. The protocol itself does not assume simple ad hoc or ARP, but the implementation certainly does.

Functionally MADRP uses UDP port 1066 and 224.0.0.11 as a multicast address. Given MADRP's experimental nature, these numbers are not cast in concrete and can be changed if desired (but at this time they are compiled in). See mip.src/sr_config.c for these and other configurable constants.

The protocol is actually fairly simple. If the kernel discovers that a route to a particular host does not exist (or mnd decides that a MADRP discovery protocol packet should be sent), e.g., due to a kernel route(4) socket upcall to the daemon ... mnd will send a MADRP multicast discovery packet. Participating hosts or agents may or may not forward the query decrementing the ttl (this depends on the limited flooding algorithm). Agents do not forward the query on internal (wired infrastructure) ports, but do forward it on wireless ports where the ad hoc link layer is used. The correct ip destination returns an answer (which may be multicast or unicast depending on configuration). Agents always return an answer (and thus in a sense act as a possible default route). Intermediate routing systems (other Mobile Nodes) setup host routes themselves typically back to the sender as MADRP queries (or multicast replies) are sent out and forwarded. As a result a routing path is setup.

One interesting trick is that Mobile-IP works simply because the mnd tries to find a host route to its Home Agent. If any agent can be found (possibly across intermediate mobile nodes), Internet access will thus be available. Another trick relies on TCP (the protocol) as an optimization. If TCP discovers that a round-trip "host route" fails due to the other end failing to return TCP ACKs, mnd is notified, and it will reinitiate the end host discovery mechanism.

MADRP at this time lacks much needed configuration and runtime status information.

Further MADRP documentation can be found in three places. The original ASCII specification as written by Xu Hao can be found in the papers directory. Jennifer Ye wrote a term paper on the implementation. Various descriptions of the protocol can be found in our project reports at http://www.cs.pdx.edu/research/SMN.

MADRP can only be compiled in or compiled out in the current implementation. See mip.src/Makefile and insert #ifdef MADRP as appropriate and recompile everything to add it.

----

2. DHCP/IPSEC - Zhong Chen

The DHCP mode causes mnd (the Mobile Node daemon) to be DHCP aware as well as Foreign Agent aware. This functionality only affects the Mobile Node, not Agents. DHCP is compiled in to the source base (#ifdef DHCP), but is not configured on by default. One and only one configuration line is necessary in /etc/mnd.conf in order to make mnd DHCP-aware.

```
dhcp mvif0 [lease-period-in-seconds]
```

The mvif device is used for "pushing" the Mobile-IP fixed node address (the home IP address) when a mnd system is resident at a DHCP server link. The lease-period-in-seconds value is optional. It defaults to 10 seconds. The MN will use the renewal period to try and detect movement; i.e., it will ask any local DHCP server for an address, and if the address is different, it will assume movement. A period of 60 seconds might be reasonable for casual mobility. Note that we assume that Foreign Agents are slightly better than DHCP-servers; i.e., if we don't have a FA, we will try and use DHCP (MN will make a DHCP query).

We should point out that the ISOC DHCP implementation was used on the servers and was of assistance in the creation of our DHCP code.

In addition to stock DHCP usage, mnd can also be IPSEC aware and tunnel packets to/from the Home Agent. When at a non-home (foreign) subnet, the security policy as specified with the mnd.conf line:

```
ipsec_mn bypass_fa ... etc
```

can be used; i.e., the MN will send IPSEC'ified packets to the Home Agent. The external IP header will use the DHCP local address as its IP source. The internal IP datagram will be shielded with ESP, hence the MN can send packets to/from home sans IP spoof filtering problems at nearby routers. HA IPSEC functionality is of course unchanged from the Foreign Agent (non-DHCP) IPSEC case).

As a security note, it should be pointed out that this mechanism solves one of the so-called Mobile-IP firewall security problems;

i.e., it allows Mobile-IP to work in the presence of ip-spoof filtering firewalls. It does *not* of course solve the problem of protecting ones own security enclave from alien and presumed hostile mobile nodes.

----

3. New IPSEC integration functionality (David Reeder)

3.1 two IPSEC bindings at once (combined AH/ESP)

Either route(8) or mnd may use combined AH+ESP headers (note: we have no support for the newer form of ESP, just the older separate forms) as opposed to (the older) ESP by itself. For use with route(8), please see the IPSEC-related modified man page on route (in our distribution tree, this is found at ../sbin/route/route.8). nroff -man route.8 | more. Note that route add can be used to add the first IPSEC association, and route change is used to add a second IPSEC association. One must a priori have two IPSEC associations (one for ESP and one for AH) in /etc/keys which have been loaded with keyadmin(8), and then one uses route(8) to tell the kernel to bind the IPSEC associations to the routes in the routing table. Packets sent through the routes will thus have the IPSEC modifications made to them.

This functionality may also be used with our Mobile-IP daemons. As before, one must first setup /etc/keys and use keyadmin(8) to load IPSEC associations into the kernel. The routing daemons (mipd) will then use this information when routes are setup or torn down automatically as Mobile-IP is used.

See the mnd.conf(5) man page for more details. The basic idea is that e.g., with the mnd.conf configuration file line:

```
ipsec_mn at_ha ha_ip_address ipsec-protocol SPI-pairs [off]
```

instead of using ONLY esp, ah for the ipsec-protocol, one can also specify ah+esp for the combined form. An additional SPI-pair is needed here in the configuration line. See the mnd.conf(5) man page for mnd functionality. See mipd.conf(5) for the related HA-side functionality (again ah+esp is used). Note that functionality may also be used with ipsec tunnels. Please see the mnd.conf and mipd.conf man pages for more details.

3.2 IPSEC HA/FA tunnels

We have worked out the kernel details so that it was possible to combine IPIP tunnels directly with IPSEC attributes. This

```
can be done manually with the route(8) command.  We also
integrated this functionality directly with the mipd Mobile-IP
daemon.  Of course, we still assume manual IPSEC keys (although
there is no reason a more general policy description could exist
that would allow ISAKMP (IKE) to automatically setup tunnels).  The basic
idea is that if a priori configuration exists in both
(IPSEC spis) /etc/keys loaded at boot, and mipd.conf (control
for MIPD), then mipd will install per Mobile Node IPIP tunnels
so that a AH security association exists between a Home Agent
and a Foreign Agent.  As a security PRO, this means that Foreign Agents
can be configured to
1. not accept plain IPIP packets and hence only accept IPAHIP
packets from hosts with which they share a security assocation.
(see the sysctl net.inet.ip.ipsecforwarding switch description in the
README file)

As a criticism, in truth, this feature sadly needs ISAKMP and
some one to centralize key storage in order to make the
IPAHIP tunnels dynamic in terms of configuration.  With only manual
keys and a few agents, there are too many manual keys to keep straight.

We assume that the net.inet.ip.ipsecforwarding switch is used at
agents to prevent the acceptance of plain IPIP packets.

2. Configure IPAHIP tunnels as opposed to IPIP tunnels
between agents (or HA/MNs acting as their own FAs).

In order to configure a "ipsec_tun" tunnel which is a one
way tunnel between a HA and a FA, one does the following:

2.1 add an AH association in /etc/keys .
in both HA and FA files add a one-way AH association between
        the two in /etc/keys and of course make sure that is loaded
        with keyadmin(8).

2.2 /etc/mipd.conf ipsec_tun switch

In mipd.conf files make sure that relevant ipsec_tun ha (at the HA)
and ipsec_tun fa (at the FA) lines exist so that mipd knows to
use the security association.  As a consequence IPIP packets will
be replaced with IPAHIP packets between the HA and FA.
See mipd.conf(5) for more details

-------------------------------------------------------------
4. HARP - Bjorn Chambless.

Overview:
```

The Home Agent Redundancy Protocol allows Mobile IP to maintain Mobile Node (MN) connectivity in the event that a Home Agent (HA) fails or some forms of network partition appear between the Home Agent and the current care-of-address of a MN.

The protocol operates by extending Mobile IP so as to allow multiple Home Agents to service a given set of MNs, thereby ensuring that the loss of a single HA does not lead to loss of network connectivity.

Cooperating Home Agents share registration information so that either HA may forward packets at any time to a Mobile Node's current care-of-address at any time.

The protocol is fairly simple. HARP uses three kinds of messages:
1. HARP registration forwarding. A Mobile Node registration at either HARP co-HA is internally forwarded as a HARP message from one co-HA to the other. As a result, they setup tunnels as necessary in parallel so that any received datagrams for the MN will be forwarded from either co-HA. This mechanism uses the HARP UDP port 1588 (look for HARP-PORT in mipconfig.h).

2. a boot time, HARP uses a TCP connection to exchange any internal MIP registration info with the other co-HA. This is an optimization and servers to speed up registration if one co-HA reboots due a local failure.

3. a HARP internal "ping" is used. Periodically one HARP co-HA will use the UDP HARP channel to determine if the other co-HA exists. Actions taken on failure may include:
1. a page of a local network admin OR
2. bringing up a local interface to act as a HOME for MNs if desired.

HARP is viewed as an Interior Gateway Protocol that ideally acts in conjunction with another IGP like OSPF or RIPv2. It may be used as static routes. Topologically we view co-HAs architected as follows:

```
Internet
|
OSPF router1 -------------  OSPF router2
  | exterior link          | exterior link
HA1 HA2
  | mobile-IP subnet .......     |
```

co-HAs must have two (possibly virtual) interfaces. E.g., one could use an Ethernet and wireless interface. Or one might use a single ethernet i/f with two ip addresses. We assume that the co-HAs somehow share ONE Mobile-IP subnet address, but that they also MUST have a separate unique IP address. Calls these two ip addresses:

1. the exterior (unique) IP address
2. the MIP (shared) IP address

From the IGP (exterior) point of view, the co-HAs can function as routers. They simply somehow inject the existance of the MIP subnet into the local IGP routing domain. In our implementation, we currently lack a dynamic routing daemon on FreeBSD (although we have some experimental routed code included that is insufficiently tested, but does show signs of co-existance with mipd). We recommend that the "real" router/s simply use static routes for now. If one co-HA fails, the static route path can be changed. Ideally however the co-HAs would be part of an OSPF system and Mobile-IP traffic could be load-balanced as appropriate through the routing infrastructure.

From the Mobile-IP subnet point of view, one has a number of topological possibilities. For example, one must act whether the interior Mobile-IP subnet (home) is partitioned or not. We recommend a partitioned subnet (ideal for wireless), but the system should work with a normal non-partitioned link (e.g., with ethernet). Keep in mind that the Mobile-IP subnet IP at the Home Agents (for the HAs themselves) must be the same. As a result, Mobile-IP does not change as a protocol and the MNs do not need to know about "two" HAs.

The internal Mobile-IP link topology is important. Ironically as HARP is an IGP, we *could* claim that how we deal with that is not important (it has little to do with HARP as a routing protocol). However it is important in practical terms. Thus we support two possible internal Mobile-IP link topologies.
1. partitioned (the HAs can NOT reach each other on the Mobile-IP link)
2. non-partitioned (the HAs can reach each other on the Mobile-IP link)

At PSU we have two co-HAs deployed. Each has a wireless and ethernet interface. The Mobile-IP subnet is bound to the wireless interface. The co-HAs can NOT hear each other on the wireless interfaces. Other topologies are possible

(e.g., two ethernet i/fs per HA or one ethernet i/f with two
ip addresses).

Setup:

This implementation of HARP only allows for the deployment of two Home Agents. ie. A co-HA pair.

HA locations should be chosen so as to be as far separated as possible, both geographically and in terms of network topology. This is in order to avoid the loss of both Home Agents due to some localized disturbance like a power loss or a local router outage. This may not be possible, but it is good if it can be done.

At this point in time, we have experimented with a modified BSD routed on freebsd to make it coexist with mipd. This routing daemon would allow the FreeBSD systems to be part of a dynamic (but RIPv2) IGP infrastructure. Thus the infrastructure itself could determine if the interior routing path to the Mobile-IP subnet had gone down.

The experimental version of routed can be found in the usr.sbin/routed directory. We verified that it could work in conjunction with mipd using RIPv2. We do not feel that routed has been tested enough to be seriously deployed however and do not have the resources at the moment to do more testing.

Those wanting to actually use HARP can resort to static routes in local routers (e.g., Ciscos). We have tested changing static routes in neighborhood routers to make sure that HARP works; i.e., we take down one co-HA host to simulate loss of a Home Agent, and then change the Mobile-IP subnet route in a nearby Cisco by hand to point out the other HARP system to make sure HARP works. Although this is not the best model in terms of dynamic routing, it does provide proof of concept.

Re HARP configuration, first of all note that (almost) exactly the same mipd.conf file must appear on both HAs; i.e., /etc/mipd.conf. In general, key lines (e.g., for mip/mns) will most likely be the same. Of course other lines like coas/co-ha ip will be different.

There are three HARP config lines that can appear in /etc/mipd.conf

1. coha ... REQUIRED
2. harp_tunnel ... (optional)
3. harp2_arp_disable ... (optional)

(In addition there is an extra possible HARP protocol authentication line, which we will briefly discuss below.)

1. coha ip_address

Designates the IP address of the OTHER home agent in a co-HA pair. This line must be present for HARP to function. It must be present in both co-HAs and the IP address must be the "interior" (non-Mobile-IP) address that is unique per co-HA.

2. harp_tunnel < yes | no >

This configuration line applies to one of the two possible Mobile-IP link topologies. By default "no" is used (i.e., it is off). If a partitioned Mobile-IP link is desired, and you want Mobile Nodes to be able to be present at both Homes, set this value to "yes". As a result, the co-HA receiving packets for a MN present at the other co-HA will tunnels those packets to the other co-HA.

3. harp2_arp_disable <if_name>

This line should be used only when one desires to use a non-partitioned Mobile-IP home link. If used, harp_tunnel should be set to no (the default). It should only be set at one co-HA (not both). It causes that co-HA to dynamically enable ARP on the interface in question if and only if by using the HARP ping, that co-HA determines that the other co-HA is down. Hence confusion due to ARP and shared ip addresses will not occur.

HARP protocol authentication:

In terms of the implementation there are two possible mechanisms for HA to HA HARP protocol authentication. IPSEC AH at the application level may be used and in fact we assume that is available. mipd will simply try and use it with the current codebase. This means that an IPSEC/AH association must exist between the two Home Agent systems.

An optional mechanism that is like the Mobile-IP md5 authentication mechanism may be compiled into the code (this is not discussed in the HARP IETF draft). In order to use this, mipd must be recompiled. This form of authentication is compiled in with #ifdef MIPSEC for HARP. if IPSEC is not defined, then MIPSEC is compiled in by default and the haha_key item is used.

```
haha_key xxx.xxx.xxx.xxx md5 SPI Key
```

Again, haha_key must be compiled in and is only used as a stopgap so that some form of authentication always exists.

Additional documentation on HARP can be found in the ./papers directory. There is a student term paper written by Bjorn and an Internet draft authored by Bjorn and Jim Binkley.

Final configuration note:

In our current implementation for HARP in /etc/mipd.conf we actually use:

```
co_ha <other ip address>
harp_tunnel yes
```

This is because our agents are configured with two real physical interfaces (ethernet and wireless). Thus it is possible for our Mobile Nodes to be home in two places as well as take advantage of the redundancy provided by HARP.

In Conclusion:

Again see the README for the main installation details. This release only adds functionality (and fixes a bug or two).

As a reminder, see our web page: http://www.cs.pdx.edu/research/SMN for DARPA reports, which document the protocol and architectural aspects of a lot of this work.

In general project documentation consists of:

1. README* in this release
2. papers/ ... various papers generated at various times during the project that have bearing on particular parts of the project. Some of them are simply student reports and others are rough draft of either IETF documents or academic papers.
3. man/ ... man pages. Note that ps versions are supplied as well.
4. http://www.cs.pdx.edu/research/SMN

B Security Considerations for Mobility and Firewalls

small

Internet Engineering Task Force — Jim Binkley
INTERNET DRAFT — Portland State University
Category: Informational — John Richardson
Intel

Security Considerations for Mobility and Firewalls
<draft-binkrich-mobisec-00.txt>

Status of This Memo

Distribution of this memo is unlimited.

This document is an Internet-Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months, and may be updated, replaced, or made obsolete by other documents at any time. It is not appropriate to use Internet Drafts as reference material, or to cite them other than as a ‘‘working draft’’ or ‘‘work in progress.’’

To view the entire list of current Internet-Drafts, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Northern Europe), ftp.nic.it (Southern Europe), munnari.oz.au (Pacific Rim), ftp.ietf.org (US East Coast) or ftp.isi.edu (US West Coast).

Distribution of this memo is unlimited.

Abstract

In this paper we discuss various security issues concerning Mobile Hosts using Mobile-IP or other mobility systems (DHCP standalone) and current firewall technology. We first present some recent attacks on the Internet and what they might mean for mobile systems like Mobile-IP that rely on tunneling technologies. We point out that tunnels are a security threat and suggest how mobile systems

may be made "less insecure" with the use of IP layer security (IPSEC) as one means for creating Virtual Private Networks. The goal is to describe a security model wherein mobile systems can work across the Internet and not just as an interior routing protocol within one security and/or interior routing domain. Both the protection of Mobile Systems abroad and of Security Enclaves that tolerate mobile visitors must be considered.

Table of Contents

1. Introduction and Assumptions

With the rapid growth of Virtual Private Networks, tunneling protocols are assuming a high profile in the Internet. Our work with tunnels as applied to Mobile-IP has uncovered a vulnerability that most tunnels leave unprotected. Basically, while most of today's firewalls stop IP Spoofing attacks, tunnels "drilled" through those firewalls re-enable that class of attack.

Strong authentication of the remote systems by the tunnel endpoint, while necessary, is not sufficient to maintain the protection provided by the firewall complex.

More generally, if a tunnel server allows authenticated remote systems to become part of a "secure enclave", it must also provide the basic protection that the firewall provides for native hosts in that enclave.

The problem becomes more interesting if the secure enclave wishes to host "visiting" systems locally. For example, a company might wish to provide Internet connectivity in conference rooms and allow visitors to access the Internet (and not the secure enclave).

We will consider these problems below in more detail.

1.1 Assumptions

Networking, especially when done securely, has been developed from many different perspectives. Each community starts from a presumed base of common language and "normal" assumptions. To minimize confusion, we begin by stating our assumptions and provide a brief description of how commonly used terms are used in this document. We do not mean to imply anything about how they "should be used", just how we chose to use them here.

The focus of this document is on how a secure enclave (firewall protected area) may tolerate Mobile-IP [RFC-2002] or simpler mobility systems (for example, DHCP used standalone) and remain secure. By "secure enclave" we mean a conventional IP site with one management domain and a centralized security administration typically behind one IP firewall [Chapman]. By "firewall" we refer to one or more systems acting together to provide protection for a network. In particular, we assume that one (or more) endpoints of IP tunnels are part of the firewall complex.

Our focus here is on how a secure enclave can protect itself from foreign (non-local) Mobile Nodes. We also deal with IP spoofing issues and possible security problems that might occur due to naive implementations of IP tunneling [RFC-2003] when combined with such spoofing.

The discussion is focused on the network layer. We are not considering higher-level authentication or confidentiality services that might be part of an application-level system. When we talk about firewalls, we are mostly talking about network layer access, and such mechanisms as packet-level firewalls with access control, Virtual Private Networks implemented as IP tunnels, and IP layer security (IPSEC) [RFC-1825]. We do not mean to discourage application or transport layer security in any way (Please see [RFC-2316] for the latest IAB discussion on network security in general). It is simply not our focus in this document.

Regarding firewalls, we assume Cisco access lists as a rough lingua franca for access control on routers and will use access list examples suitable for Cisco routers. Please see [Ballew] for discussion of Cisco access list mechanisms. We assume packet filter technology simply because accidental holes may indeed be poked through such a router if its manager is not careful.

When we cite IP addresses as examples in the text, we will use private IP addresses as mentioned in [RFC 1918]. These should be viewed as surrogates for public ("real") IP addresses associated with an interior routing domain. We use these addresses because we do not want to cite "real" addresses in any examples.

2. The Problem Space

Firewalls are designed to separate "inside" from "outside". A naive approach to protection would use the source IP address to make the distinction. Unfortunately, IP header information is unreliable as it can be set by the source (or any intermediary) to any arbitrary value. The attacker community knows this well and it forms the foundation for an entire class of attacks known as "Spoofing".

Spoofing has been used as the basis for a whole set of recent attacks (for example, see [RFC-2267]). Some of the attacks are denial of service oriented. Some seek to cause the attacked system to crash or hang. Many of the attacks can be characterized as single packets wherein the IP source and destination addresses in the IP header appear to originate within the attacked systems site.

2.1 Spoofing Attack Examples

To highlight the importance of spoofing attacks, we will briefly discuss three such attacks, TCP SYN [CA-96.21], "smurf" [CA-97.28], and "land" [CA-97.28].

In TCP SYN attacks, the attacker sends TCP initialization packets to a given site. The attacked system is tied up simply due to opening too many TCP control blocks which cause allocation of precious kernel memory. The attacker need only send one TCP SYN packet. The attacker may choose to use a spoofed IP source address so that tracing the attack back to its originating system is difficult.

In "smurf" attacks, the attacker sends one or many ping packets to an IP directed-broadcast address with an IP source address that may also be at the destination site. For example, if a site had a site specific class C address along the lines of 192.168.1.0, the attacking IP destination might be 192.168.1.255 or 192.168.1.0 (0 broadcast addresses may be used as well) with an IP source of 192.168.1.1. The result is that two systems (at least) may be attacked. The IP source itself is bombarded with ping reflections from all the systems at the directed broadcast address. Further the smurf vehicle could also be used for single packet "ping of death" attacks.

"Land" attacks involve one TCP SYN packet in which the IP source is set to be the same as the IP destination. The attack may cause the receiving machine to hang.

In general, note these attacks do not involve packets being returned, unless the packets are returned to another system that is being indirectly attacked itself ("smurf").

2.2 Prevention of Spoofing Attacks

One general technique that can be used is to disallow IP spoofing for internal source addresses [RFC-2267]. Filters can be put in place so that packets arriving on an "outside" interface must have an "outside" source address (or must NOT have an "inside" source address). One may also filter out spoofing attacks attempting to leave from the "inside" of a network.

We will briefly look at how spoofing may be prevented with a Cisco router which we assume is the interface between a site having 172.16.*.* as its internal IP address space and the Internet at large. Note that the access list entries shown here may be part of a more complex firewall policy and/or access list, but we only show the part relevant to IP spoofing.

The following access list entry may be bound to an external router interface and would be applied to packets entering the site.

```
access-list 101 ...

access-list 101 172.16.0.0 0.0.255.255 any
```

Packets entering the site with 172.16.*.* addresses will be discarded.

We apply the following access list to packets headed out on the external interface:

```
access-list 111 ...
access-list permit ip 172.16.0.0 0.0.255.255. any
access-list deny ip any any log
```

This blocks packets headed out that do not have IP src addresses in the 172.16.*.* range and logs any internal attempts at spoofing. Thus spoofing attacks cannot originate at this site. The result is that a site firewall or border router will neither send or receive packets over an interface when the packets do not belong to the source IP routing domain.

2.3 Anti-Spoofing Measures and Mobile-IP

The result of such anti-spoofing measures is that packets headed into the enclave to "foreign" Mobile Nodes; i.e., Mobile-Nodes from some other site than 172.16..*.* will not be permitted to enter. Packets trying to leave the site from systems from another site, say 192.168.1.0, will not be permitted to leave. Mobile-IP within the same routing domain, which we might call "interior Mobile-IP" would be permitted. Mobile-IP ("exterior Mobile-IP") between two interior routing domains would not be permitted.

Now we must consider what happens to packets sent from a "visiting" foreign Mobile Node that is somehow operating within the secure enclave. Please refer to the picture below.

First the UDP-based Mobile-IP registration protocol would still work if Foreign Agents were used as Foreign Agents act as UDP proxies for Mobile-IP registration; i.e., they will replace a Mobile Nodes IP source address with their own (legal) source address. A Mobile Node using DHCP as a source of local addresses could also succeed if it used the DHCP-obtained local address.

Data packets sent directly out of the domain from the visiting Mobile Node (unless tunneled via a local IP source address) would be discarded at the border. Data packets that somehow escaped the local secure enclave's border router could also be discarded by the "home" border router's spoofing filter as well, as it would not permit packets to enter that have "local" IP source addresses. Data packets tunneled from the (exterior) Home Agent to the (interior) Foreign Agent would be allowed through because the external encapsulation would get past the spoofing filter; i.e., Home Agent to Foreign Agent IPIP packets would have the legal interior IP source for the Foreign Agent as the exterior IP source address.


```
Home Agent    <-------> Border Router (for 192.168.1.X domain)
(192.168.1.X)
      ^
                |
                |  the Internet
      |
      v
Border Router (secure enclave)
 ^
             |
             |  secure enclave (172.16.*.*)
 V
```

```
Foreign Agent (172.16.*.* address for Care Of Address)
 ^
 |
 V
                Visiting Mobile Node  (192.168.1.X)
```

In addition to the obvious problems that anti-spoofing raises for Mobile-IP, one must also ask if tunnels raise additional security concerns and how one might address both those concerns and security for both the Mobile Node itself, its home domain, and the "visited" domain too.

3. Tunnels Considered Harmful

One mechanism that is part of Mobile-IP and in point of fact many other routing protocols are IP tunnels which might be implemented with IPIP, IPSEC Tunnel Mode, or Cisco's Generic Routing Encapsulation [RFC-1701]. It should be pointed out that IPIP tunnels are not peculiar to Mobile-IP. They are used in many routing protocols for many purposes including tunneling non-IETF protocols (e.g., Appletalk) or building virtual networks on top of the current Internet (MBONE, 6BONE).

Tunnels may mean encapsulated packets where one has one IP datagram inside another IP datagram and we will use IPIP (IP protocol 4) as our example here.

Mobile-IP uses tunnel mechanisms like IPIP to forward packets from the Home Agent to a remote "Care Of Address". The COA is a local site IP address that may represent a Mobile Node that has acquired a local IP address itself directly via DHCP or a router system that understands Mobile-IP called a Foreign Agent.

Any Mobile-IP system, including Mobile Nodes, Home Agents, or Foreign Agents, may source or sink tunnel packets. When a Home Agent forwards packets to a Mobile Node that is at a Foreign Agent, the use of IPIP in a datagram may appear as follows:

```
IP outer header          IP inner                     IP datagram
--------                 --------
ip src= Home Agent      ip src = peer end host     |  TCP, etc.
ip dst= Foreign Agent   ip dst = Mobile Node       |

|
| packets to MN
v
Home Agent ========= IPIP tunnel to COA ===>   FA and Mobile Node
```

One might ask if it is enough to simply use IPIP tunneling and somehow tunnel either from the Foreign Agent or Mobile Node back to the Home Agent and thus evade the anti-spoofing measures at a firewall? Unfortunately, this is an insecure approach.

In the first place, it is not enough to simply tunnel over the IP spoofing firewall. This is simply a new form of spoofing which we might call: "IPIP spoofing". The problem is that if one has a tunnel sink (be it any kind of agent or Mobile Node) that decapsulates packets and then forwards them, others can launch their spoofing attacks with IPIP too and thus have the spoofing emerge "inside" the enclave firewall. For example, smurfing might simply be redirected through a tunnel. The inner IP header might be directed broadcast with an interior IP source named as a target. All the previous attacks (TCP SYN, "smurf", "land") can thus be done through the firewall.

We suggest that one can block tunnels with current access list mechanisms and thus control tunnels so that tunneling from the outside can only be done to certain hosts that will be considered as "network-layer" bastion hosts.

For example, with Cisco IOS one can add the following statements to access list 101:

```
access-list 101 deny ipinip any any
access-list 101 deny gre any any
```

and thus block any IPIP or GRE packets coming in over the router. One may further add a permit statement to force any IPIP packets coming in to land at a certain host and then treat that host as a bastion host; i.e., a nexus of security focus.

For example,

```
...
access-list 101 permit ipinip any host 172.16.1.3
access-list 101 deny ipinip any any
access-list 101 deny gre any any
...
```

in a Cisco input access list, means IPIP will only be allowed

to the host 172.16.1.3, which we will assume is a tunnel
sink agent.

We have focused internal trust for IPIP on that one system. We next need to explore how to make sure that packets arriving at the tunnel sink agent are *not* attacks that can be made via a tunnel sink. This can be done with "tunnel-mode" IPSEC tied to IPIP. We will explore this idea further in the next section.

4. Problem Solution Space

We will discuss how to solve these problems from two topological points of view. First we look at the situation from the Mobile Node abroad's point of view. We assume it actually wants to get packets home and not compromise home security. Thus this point of view must necessarily include the Mobile Node's home enclave. We then look at the situation from the "foreign" security enclave's point of view. We assume that the foreign enclave wants to allow mobile service but protect itself. We also must consider the question of how both security enclaves (home and away) in general protect themselves from any tunnel-based attacks.

In this discussion we also try to contrast the use of Mobile-IP versus a simpler form of DHCP-only mobility that does not use Mobile-IP. Keep in mind that the main semantic for the use of Mobile-IP is that the Mobile Node retains at least one fixed IP address that is non-local for the subnet it is visiting.

A Mobile-IP system may have two IP addresses associated with it, a fixed permanent Mobile-IP address (call it the "Mobile-IP address"), and a locally acquired address (call it the "DHCP address"). A DHCP-only system would only have one locally acquired IP address.

4.1 Mobile Nodes Abroad Point of View

In this section we will consider the problems for a secure enclave if Mobile Nodes in that secure enclave go abroad; i.e., out to the Internet beyond the firewall. We must ask how the Mobile Node can secure its own traffic and in effect, take its security enclave with it. We must also ask how the home enclave can secure traffic coming from that Mobile Node back inside, which will extend the thinking about tunnels in the previous section.

Glass and Gupta [Gupta-draft] suggest that Mobile-IP Mobile Nodes abroad may use DHCP to acquire "local" IP addresses, Thus they can get by the anti-spoofing measures in the firewall router. This is indeed a reasonable possibility. Further, the Mobile Nodes can use IPSEC with two-way tunnels between the Home Agent as a classic bastion host and the Mobile Node. (See [http://www.cs.pdx.edu/research/SMN for a combined Mobile-IP IPSEC system in which Mobile Nodes can do two-way MN/HA ESP tunnels).

Please see the figure below:

```
                               Internet            172.16.1.1
Home Agent: <----> Firewall <-----------------> router/DHCP server
192.168.1.1                                       |
                                                      |
                                                      |
                                                   Mobile Node

   172.16.1.2/
                                                   192.168.1.2
```

4.1.1 Packets from the Mobile Node Out

If we assume Mobile-IP and two addresses in use by the Mobile Node, packets tunneled from the Mobile Node to the Home Agent might have the structure:

```
IP(1) | IP(2) | <IPSEC> | IP(3)
```

Each IP header has its own purpose. The most external header, IP(1) exists to get the packets to the Home Agent with the DHCP acquired address == 172.16.1.2. The IP destination would be 192.168.1.1. Thus header(1) allows transit of the Internet and any anti-spoofing firewalls. When the packet arrives at the Home Agent, that header is discarded and header(2), consisting of IP(2) combined with an IPSEC header is processed. Here we assume that the Mobile-IP address 192.168.1.2 is used for the source address and the Home Agent is again the destination.

The fixed Mobile-IP address may be needed here as it allows a-priori manual IPSEC keys to exist between the Mobile Node and the Home Agent. In effect, this is an IPSEC tunnel between the Mobile Node and the Home Agent. The interior header would contain the Mobile Nodes fixed address (192.168.1.2) as IP source and the address of any destination to which it is allowed to send packets.

The above triple-header system could be optimized by a higher-level protocol that could produce a dynamic binding between the local DHCP-acquired COA and the Home Agent's destination address. There is no reason Internet Key Exchange protocols [IKE] where non-IP naming schemes are used could not be deployed here. This would allow one header to be deleted.

For a DHCP-only form of mobility, the packet layout situation would be simpler. The Mobile Node would use a non-IP naming scheme with IKE to form a security association with a Home Security Agent. IP header (2) would not be needed.

4.1.2 Packets Coming From Home to the Mobile Node

For Mobile-IP, we must now consider packets coming back to the Mobile Node via a tunnel from the Home Agent. By definition these packets are tunneled to a local IP address and are not subject to problems caused by anti-spoofing filtering. However IPIP unadorned is a security threat to the receiving enclave. And of course, the Mobile Node may choose to have IPSEC-based security between itself and its home enclave.

One possible encapsulation scheme might take this form:

IP(1) | IP(2) | IPSEC | IP datagram

IP(1) exists to get the packets from the Home Agent to the remote Care Of Address which might be a Foreign Agent or a Mobile Node that has acquired a local IP address. The inner IP header would exist where manual keying is needed with IPSEC and the IP source would be the Home Agent. The IP destination would be the Mobile Node itself. Note that again IKE could be used to optimize out an IP header as long as IP addresses are not part of a manual configuration scheme.

It is highly likely that from a security policy point of view, one would not form security associations (especially confidentiality-based security associations) between random Home Agents and random enterprise-external Foreign Agents. As a policy consideration, unsecured IPIP might simply not be allowed to Foreign Agents. Foreign Agents might insist that all IPIP packets be sent to them from internal Home Agents with which they share an a priori security association. Alternatively Foreign Agents might exist "outside" a secure enclave, or unadorned IPIP packets when decapsulated might only be allowed to go "outside".

4.1.3 Tunnel Security at Tunnel-Exit Agents

We suggest that a tunnel-sink agent like a Mobile-IP Home Agent may want to guarantee that all packets sent to it via a tunnel are cryptographically verified; e.g., shared secret keys might exist between it and the Mobile Node abroad. No packets forwarded to the tunnel-sink agent by the firewall will be internally decapsulated and forwarded until they have been cryptographically verified. This might be done with an access list mechanism tied to IPSEC or by simpler means. For example, the PSU system mentioned above has a BSD sysctl(8) switch:

```
# sysctl -w net.inet.ip.mvifipsecinput=1
```

that if set forces the IPIP driver to only forward packets if and only if IPSEC authentication or decryption has successfully occured between the remote system and this system. As a consequence, one may be sure that a Mobile-IP Home or Foreign Agent or any tunnel sink only forwards IPIP packets that have successfully passed IPSEC processing. Put another way, a security association must exist between the tunnel sink and the tunnel source system.

Packets coming from remote security-aware Mobile Nodes might have several forms:

IP(1) | IPSEC | IP datagram

or possibly

IP(1) | IP(2) | IPSEC | IP datagram

For example, the former packet architecture might occur with a remote Mobile Node that is only using DHCP and wants to securely tunnel home. The latter might be used by a remote Mobile Node that is using Mobile-IP and has also used DHCP to acquire a local COA.

The local anti-IP-spoofing firewall might then be configured in a number of possible manners depending on local security policies and the structure of external but acceptable packets. For example, with current Cisco access list technology, we could permit IP | IPSEC packets using ESP (ip proto 50) or AH (51) to the Home Agent:

```
...
access-list 101 permit 50 any host 172.16.1.3
access-list 101 permit 51 any host 172.16.1.3
access-list 101 deny ipinip any any
...
```

As in our previous example, the firewall might simply allow IPIP but only to a Home Agent. This would apply to the second IP | IP | IPSEC example.

We must point out that the security problems here are not terribly different from those encountered by current dialup clients into a secure enclave that access the enclave via an internal terminal multiplexor. The exterior host tunnels into a secure enclave and an agent in the secure enclave applies cryptographic measures to packets that have come in from the outside.

4.2 Hosting Visitors From Abroad

In the previous section we have focused on how to protect the Mobile Node abroad and also discussed the problem of how to make tunnel (exits) more secure. One must also worry about the security of the "other" enclave, else enclaves may not desire to host foreign Mobile Nodes. It makes little sense for a firewall-protected enclave to allow visitors to penetrate the enclave at will and thus enable possible attacks on internal systems by visitors.

Of course, we could start with a security policy that does not allow visitors to penetrate the firewall. In effect, that is the current security policy for many sites. However it is our goal here to discuss how we might tolerate "less trusted" visitors, not define them out of existance.

We suggest a topological approach based on network design measures that can be made with current (or near-current) technology and that should allow a secure enclave to remain secure.

Our basic principle is: "design the network so that visitor packets are not allowed inside". We observe that whatever is done to implement this goal will probably be similar to current systems that have two-level security enclaves.

For example, one might have a network designed as follows:

```
                           ^
                           |
```

```
Router(1)
   |
------------------------- internal less secure DMZ
        |          |             |
      Firewall(2) term mux DHCP/router for Mobile Visitors
|
     ------------------------
     | secure enclave       |
```


The above may be viewed as a logical (not necessarily physical) structure. We have a Router(1) that simply serves to allow access to the Internet. Inside the external router we have a less secure DMZ network that may serve to allow unfiltered access to the Internet. This network might include terminal multiplexors and local mobility servers. Behind it we could have at least one level of firewalls with bastion hosts (which may or may not be on the DMZ network). Firewall(2) would serve the function of the traditional firewall machine. One might observe that the above scheme does not force visiting laptops to acquire local addresses to bypass IP-spoofing filters. True, but unfortunately there may be other firewalls on the way home that possess such filters. Certainly the firewall at home may possess such a filter.

The above may be viewed as a physical network design that would tolerate visitors. However it is very likely that such a design may not be very convenient to implement. We suggest that various virtual network techniques may be used to approximate the physical structure above. Let us briefly consider two methods that may be used to logically separate networks and thus remove construction difficulties and make the deployment of mobile networks easier. Note that these techniques *require* careful network security design. The security onus here is certainly on both the network designer, and on the creators of both Mobile Agents and security gateways. Caution needs to be employed in the design of such systems and functionality must be clearly communicated by implementors to potential network designers.

First one may use a combination of simple tunneling sans IPSEC and/or authenticated tunnel packets (e.g., IPSEC AH) between a mobile router (MIP foreign agent or DHCP router/server) and the "outside" network. The basic idea would be that a router (e.g., a Mobile-IP Foreign Agent) could take all packets presented to an "exterior" interface and tunnel them (using IPIP or GRE) (possibly with additional IPSEC to alleviate paranoia) to a firewall-exterior interface on a border router. As a result, visitor packets would have no opportunity to access interior hosts. They would be tunneled "outside" and would be treated as external packets coming back through any existing firewall mechanism.

```
  | ----------------------------------- Internet accessible network
 Firewall(1)            exit tunnel router (2)
  |                               ^
  |                               |
  v                               |  tunnel to outside
secure enclave                    |
 v
DHCP server router or MIP agent (3)
                                  ^
                                  | external visitor link (4)
 v
                             visiting mobile node (5)
```

Note that we assume here that the agent(3) and the exit tunnel router(2) are under the control of the same network administration.
We suggest that careful combination of access lists with tunnel technology should allow the above picture to be collapsed in various ways. For example the Firewall(1) and exit router(2) systems could be the same system.

In addition, the router(3) that enables mobility could potentially optimize packet delivery. If IPSEC security associations existed at that router between a Mobile Node and the router itself, it might choose to NOT forward IPSEC-verified packets that show up via the external visitor link(4) over the internal tunnel. Thus IPSEC packets from "local" mobile systems that belong to the enclave itself could be allowed direct access to the local enclave. Of course, packets that lacked a security association with the mobile agent router would be forced over the tunnel to the "outside" world.

We will not go into details, but link-layer switching technologies can also be of use here. For example, Virtual Lans [3com] when assumed to be 1-1 with IP subnets could be used as a way to funnel visitor packets back to a router that might apply access list technology to packets trying to cross from an "exterior" subnet to an "interior" subnet.

5. Miscellaneous Considerations

5.1 Firewall Discovery

Although we cannot attribute such discussion, some have suggested that some sort of Firewall Discovery system might allow Mobile Nodes to dynamically tunnel to and from firewalls. There are several problems with this notion:

1. It is unnecessary since our solution here will work with current or near-current firewall technology.

2. It is not very likely from a security point of view. Security people and network managers may not care for notions that involve poking holes dynamically through firewalls. Complexities involved in cross-security domain certification are likely beyond near-term scope. Further the security folks "at-home" may not care for schemes that involve key exchange with strangers; i.e., a Mobile Node from home somehow secures packets between itself and a foreign firewall at a different enterprise. After all, that firewall might choose to store all data traffic, and enable a classic "man-in-the-middle" attack.

3. Traditional notions of IP fate sharing (considered bad) may apply here. Mobile-IP systems are already tied to the fate of their Home Agent. Additional ties between systems that are not related from interal routing or security enclave considerations may be complex. After all, it is hard to predict how many firewalls that rule out IP spoofing to/from a given site may exist.

Schemes that allow trusted locals to poke holes through firewalls are perilous by definition since "untrusted" people may crack the scheme. It is unlikely that dynamic mechanisms that allow random visitors access will prove widely acceptable.

5.2 The Role of a Mobile-IP Foreign Agent

In the previous discussion, we suggested that DHCP can be used to simply allow Mobile Nodes abroad to obtain a local address. Using that address they can then send packets wherever they choose. As a result, it might seem that there is little role for a Mobile-IP Foreign Agent in a security system. Ultimately the roles that mobility systems play depend upon policy considerations. One could suggest a policy wherein Mobile Nodes abroad are not allowed to speak directly to (as opposed to through) or exchange cryptographic material with "foreign agents". This is certainly a reasonable policy. However the focus of such a policy is on the Mobile Node. We need to also consider Foreign Agent oriented policy and how a Foreign Agent might serve as a border router for a secure enclave.

Foreign Agents may serve as routers that simply do not allow foreign visitors any access to an internal enclave and only allow authorized local Mobile Nodes entrance. Many techniques exist for such screening including the pre-existing Mobile-IP manually keyed registration that can secure Mobile Node access via a given Foreign Agent. However, security techniques should apply to all packets and not just Mobile-IP registration packets.

As another possibility (and there are probably others), IPSEC-aware Foreign Agents can discriminate between locals (who hold security credentials) and non-locals (who lack security credentials). Once a Mobile Node has identified itself to a Foreign Agent as belonging to the agent's secure enclave, it could use an IPSEC security tunnel between itself and the Foreign Agent. Any packets verified by the Foreign Agent as belonging to the local secure enclave could thus be delivered locally and not pushed out of the routing domain via a tunnel. Non-local visitor packets might be unceremoniously escorted off the premises via another kind of tunnel and would have no access to internal resources. Thus local mobile systems and visitors could both be tolerated at the same agent link.

Of course a paranoid enclave might choose for policy reasons to force all wireless visitors to be "foreign". "Locals" could be always treated as remote visitors and tunneled outside, thus having to use secure means to come back inside. Or foreigners might simply not be permitted entrance at a given agent. Both policy considerations are possible and should be considered in implementations.

6. Security Considerations

The entire document is about security, mobility, and dangers inherent in IP tunnels. We focus specifically on issues arising out of the interaction between firewalls and any tunneled protocol and highlight security concerns regarding Mobile-IP or simple DHCP for foreign visitors "beyond" the home firewall.

We should point out at least one more specific security consideration for tunnel entrances. If IPSEC is used in "tunnel-mode" at a router or forwarding system that is neither the IP source or IP destination, it is possible that the security system may be subject to "proposed plaintext attacks". Please refer to the following figure:

```
Home Agent (1) (or tunnel entrance)
|                        ^
|                        |    plaintext packets
|  attacking system (3) ---
|
| cryptotext packets to Mobile Node
                V
remote Mobile Node (2) (tunnel exit)
```

If a attacking system(3) can present plaintext packets to (1), and then read them back after encryption in the tunnel between (1) and (2), the potential for proposed plaintext attacks exists. This liability exists for a number of proposed combined tunnel and security systems, as long as network-layer forwarding combined with IPSEC (or cryptography) is part of the architecture. Solutions for the problem include session keys [IKE] and possible restriction of communication between the Home Agent(1) and the Mobile Node(2) to exclude IP sources that do not lie within the home enclave. By definition, this problem is found only with network-layer forwarding (i.e., at IPSEC gateways), and is not present in any end-system to end-system IPSEC.

7. Conclusions

In this document we have presented proposals that will enable Mobile Nodes from abroad or nearby to less insecurely access the Internet. Such systems are not dissimilar from current dialup systems that involve a remote PPP-based dialup client and a local terminal multiplexor. IPSEC-enabled tunnel mechanisms may be used between the Mobile Node system and its home security companion. Very simply put, the Mobile Node is an extension of the local security domain. However, in addition to securing the Mobile Node and its home enclave, one must also give thought both to the dangers of tunnels and to how a local enclave may enable its own security and still tolerate visitors.

In summary, we will make the following suggestions:

1. DHCP to acquire a local COA solves problems caused by IP spoofing prevention for visiting Mobile Nodes abroad and may

or may not be combined with Mobile-IP.

2. Suitable two-way cryptographic tunnels between a system abroad and a routing system at home will allow a Mobile Node's own traffic to be securely tunneled over the Internet.

3. IPIP tunnels sans cryptographic safeguards should be viewed with caution. If an IPIP tunnel sink does not guarantee cryptographically controlled access, an attacker may tunnel various one-way attacks (land, etc.) into an enclave. The tunnel sink may be logically regarded as an extension of the firewall itself. It may be co-located. If it is *not* co-located, firewall filtering mechanisms may need to be duplicated at the tunnel-exit point.

4. Flexibility in routing, access list mechanisms, and encapsulation possibly with authentication should be considered by implementors so that a secure enclave can securely escort visitor packets off-site without threat to secured systems within the site.

5. Security considerations must apply both to Mobile Nodes abroad, their own home enclave itself, and also to how enclaves may be designed to tolerate visitors.

8. Acknowledgements

We would like to thank David Reeder of Trusted Information Systems and Mark Morrissey of Oregon Graduate Institute for their comments.

9. References

[Ballew] Ballew, Scott, "Managing IP Networks", O'Reilly and Associates, Inc., 1997; ISBN 1-56592-320-0

[Chapman] Chapman, D.B., and Zwicky, E.D., "Building Internet Firewalls", O'Reilly and Associates, Inc., 1995; ISBN 1-56592-124-0

[RFC-1701] Hanks, S., Li, T., Farinacci, D., Traina, P., "Generic Routing Encapsulation", October 1994.

[RFC-1825] Atkinson, R., "Security Architecture for the Internet Protocol", August 1995.

[RFC-1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., de Groot, G., and

E. Lear, "Address Allocation for Private Internets",
February 1996.

[RFC-2002] Perkins, C., "IP Mobility Support", October 1996.

[RFC-2003] Perkins, C., "IP Encapsulation within IP",
October 1996.

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

[RFC-2316] Bellovin, Steve, "Report of the IAB Security Architecture
Workshop", April 1998.

[IKE drafts] Internet Key Exchange (ISAKMP/Oakley). Works in progress.

[CA-96.21] CERT Advisory CA-96.21; TCP SYN Flooding and IP Spoofing
Attacks; September 24, 1996.
http://www.cert.org/advisories/CA-96.21.tcp_syn_flooding.html

[CA-97.28] CERT Advisory CA-97.28; "Teardrop/Land" IP Denial-of-Service
Attacks; December 16, 1997.
http://www.cert.org/advisories/CA-97.28.Teardrop_Land.html

[CA-98.01] CERT Advisory CA-98.01; "smurf" IP Denial-of-Service Attacks;
January 5, 1998. http://www.cert.org/advisories/CA-98.01.smurf.html

[3com] http://www.3com.com/nsc/200374.html;
Passmore, David, and Freeman, John. "The Virtual Lan Technology".
3com Inc.

[Gupta-draft]
Gupta, V., Glass, S., "Firewall Traversal for Mobile IP:
Guidelines for Firewalls and Mobile Ip entities",
draft-ietf-mobileip-firewall-trav-00.txt, work in progress,
March 17, 1997.

10. Contact Information

Jim Binkley
Computer Science Department

Portland State University
Email: jrb@cs.px.edu

John Richardson
Intel
Email: jwr@intel.com